

2006-03-27

IEEE 802.16i-06/001r1, March 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*

Ken Stanwood, *Vice Chair*

Dean Chang, *Secretary*

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

Phillip Barber, *Chair*

Changhoi Koo, *Vice Chair*

Itzik Kitroser, *Vice 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
9. Configuration	6
9.4 Mobile MIB for SNMP	6
9.4.1 MIB-II integration	6
9.4.2 Usage of MIB-II tables	6
15. IRP Definitions	11
15.1 NRM IRP IS	11
15.1.1 Information Service Models	11
15.1.1.1 Information entities imported and local labels	11
15.1.1.2 Class diagram	12
15.1.1.3 Information object classes definition	14
15.1.1.4 Information relationships definition	15
15.1.1.5 Notifications	15
15.1.1.6 Information attributes definition	15
15.1.2 Proposal for BS Related Objects NRM Definitions	16
15.1.2.1 Information entities imported and local labels:	16
15.1.2.2 Class diagram	17
15.1.2.3 Information object classes definition	27
15.1.2.4 Information relationships definition	34
15.1.2.5 Notifications	35
15.1.2.6 Information attributes definition	35
15.2 NRM IRP SNMP Solution Set	42
15.2.1 NRM IRP SNMP Solution Set	42
15.2.2 ASN.1 Definitions of 802.16 MIB for SNMP	44
Annex E.	250
Annex F. Proposal for Adding Mobility Handover and Paging group MIBs	251
1. Introduction	251
2. Proposed Text Introduction	251
2.1 wmanIfBsObjects	251
2.1.1 wmanIfBsMobility	251
2.1.1.1 wmanIfBsHandoverConfiguration	251
2.1.1.2 wmanIfBsPagingGroupTable	251
3. ASN.1 Definitions of 802.16 MIB for SNMP	252

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 Layer Topology	2
Figure 2—Mobile BWA Network Management Architecture (I)	3
Figure 3—Mobile BWA Network Management Architecture (II)	4
Figure 4—SS / MS DL Synchronization	8
Figure 5—SS / MS Network Entry	9
Figure 6—Containment and Naming Diagram	12
Figure 7—Inheritance Diagram	13
Figure 8—General View	18
Figure 9—Segment view Neighbouring BS NRM Containment/Naming Diagram	19
Figure 10—Segment view PacketCs NRM Containment/Naming Diagram	20
Figure 11—Segment view SecurityMng NRM Containment/Naming Diagram	21
Figure 12—Segment view CPSMngMng NRM Containment/Naming Diagram	22
Figure 13—Segment view PhyMng NRM Containment/Naming Diagram	23
Figure 14—Inheritance Diagram	24
Figure 15—IEEE 802.16 NRM PacketCS Inheritance Diagram	25
Figure 16—IEEE 802.16 NRM CPS Inheritance Diagram	25
Figure 17—IEEE 802.16 NRM Security Sublayer Inheritance Diagram	26
Figure 18—IEEE802.16 NRM PHY Layer Inheritance Diagram	26
Figure 19—wmanIfBsPhy structure	42

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	7
Table 2—Example of the Usage of ifTable objects for SS.....	7
Table 3—Information entities imported and local labels	12
Table 4—Attributes	14
Table 5—Attributes	15
Table 7—Information entities imported and local labels	16
Table 6—Definition and legal values	16
Table 8—Attributes of BSFunction.....	27
Table 9—Attributes of ExternalBSFunction	28
Table 10—Attributes of BSRelation.....	29
Table 11—Attributes of PagingGroup.....	29
Table 12—PowerCtrl Attributes.....	30
Table 13—Attributes of SecurityManagmentFunction	31
Table 14—Attributes of PkmBase.....	32
Table 15—Attributes of PkmTek	33
Table 16—Attributes of MSPkmAuth.....	34
Table 17—Roles of the relation ExternalNeighbourBSRelation.....	35
Table 18—Information attributes definition.....	35

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

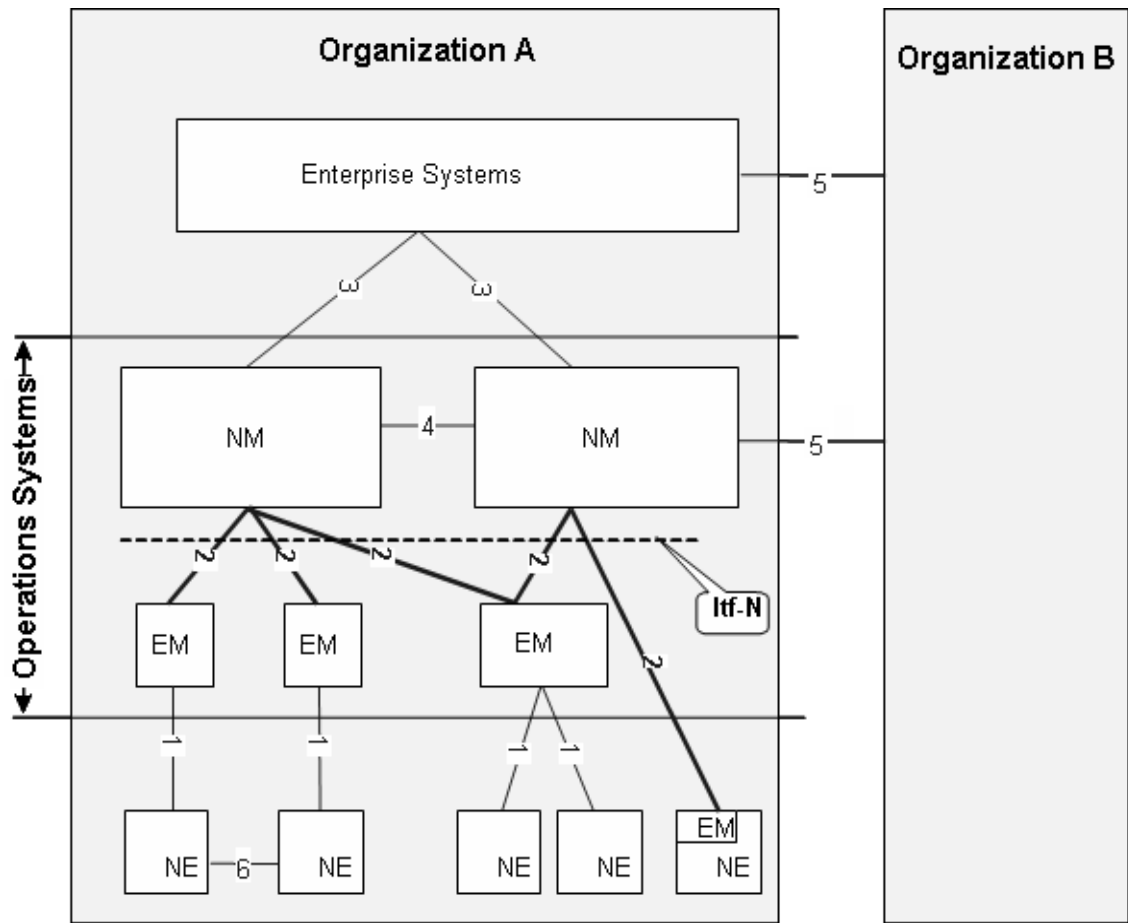


Figure 1—Mobile BWA Network Management Layer Topology

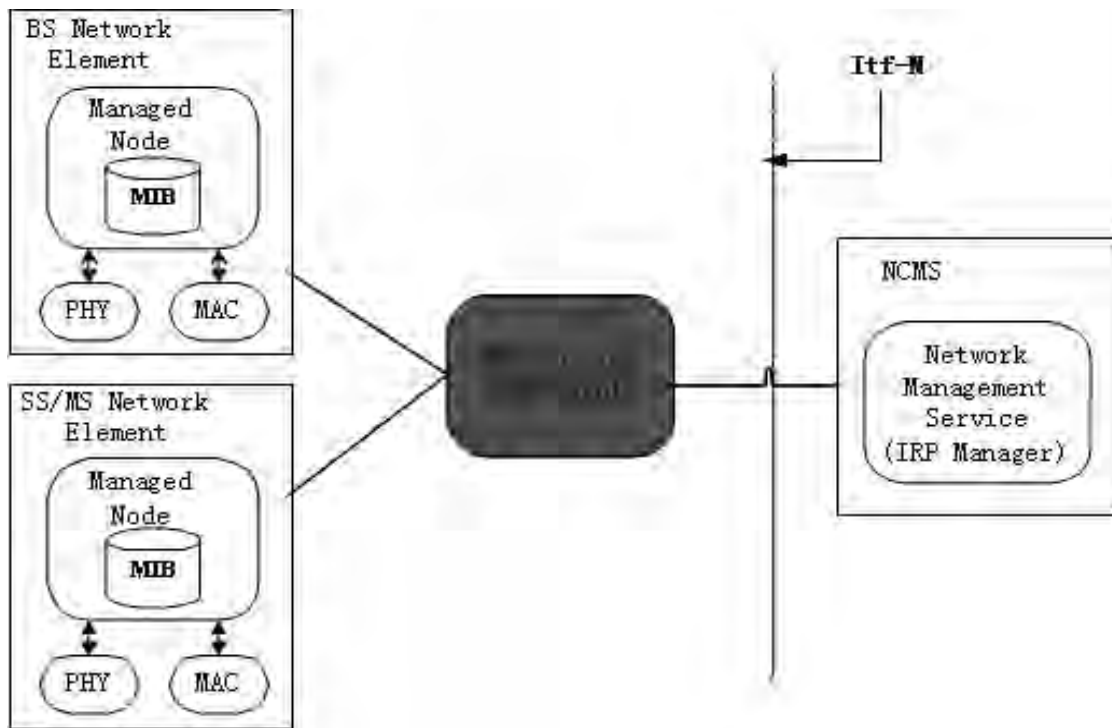


Figure 2—Mobile BWA Network Management Architecture (I)

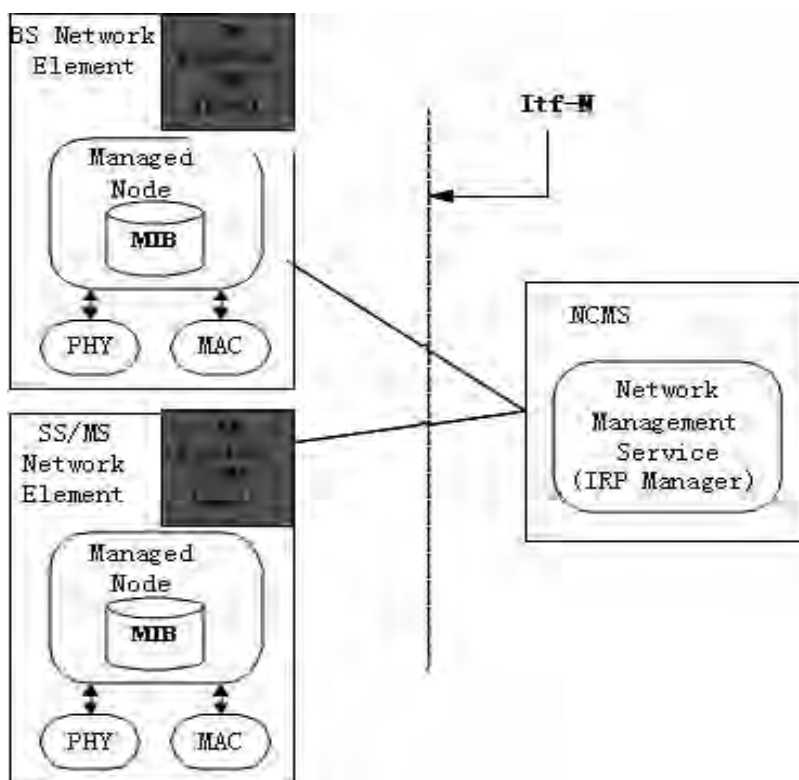


Figure 3—Mobile BWA Network Management Architecture (II)

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.

9. Configuration

[Insert a new subclause 9.4]

9.4 Mobile MIB for SNMP

9.4.1 MIB-II integration

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

```

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

```

Pending on IETF approval, wmanIfMib 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:

- IEEE 802.16-2004, OFDM 256
- 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 RFC286

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 4 shows a procedure describing how BS can determine the FFT size of a SS or MS during the DL synchronization for.

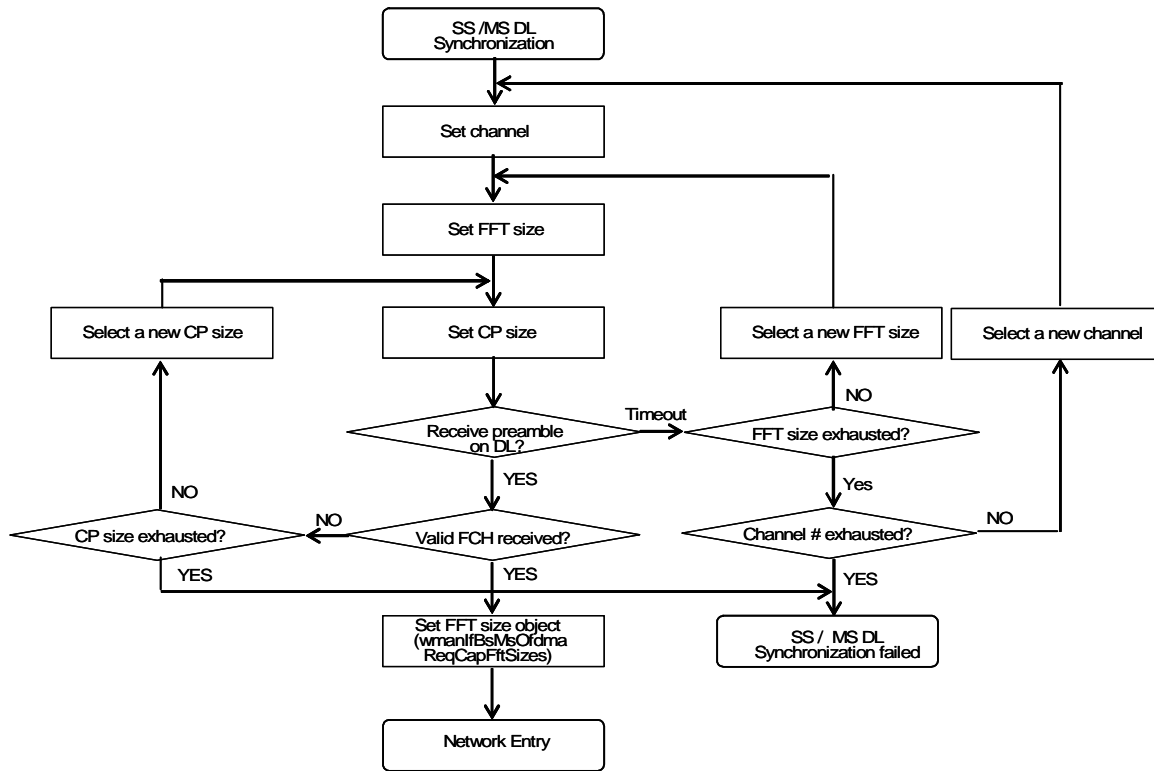


Figure 4—SS / MS DL Synchronization

1. Set the Rx channel (Select a frequency for receiving DL channel)
2. Set the FFT size
3. Set the CP size
4. If a preamble is received successfully, then go to step 5; otherwise,
 - a) If FFT size is not exhausted, then select a new FFT size, and go to step 2; otherwise,
 - 1) If channel to be scanned is exhausted, then declare SS / MS DL synchronization failed; otherwise, select a new channel, and go step 1
5. Set the CP size
6. If a FCH (Frame Control Header) is received successfully, then go to network entry; otherwise,
 - a) If CP size is not exhausted, then select a new CP size, and go to step 3; otherwise, declare SS / MS DL synchronization failed
 - a) b) Set FFT size object

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

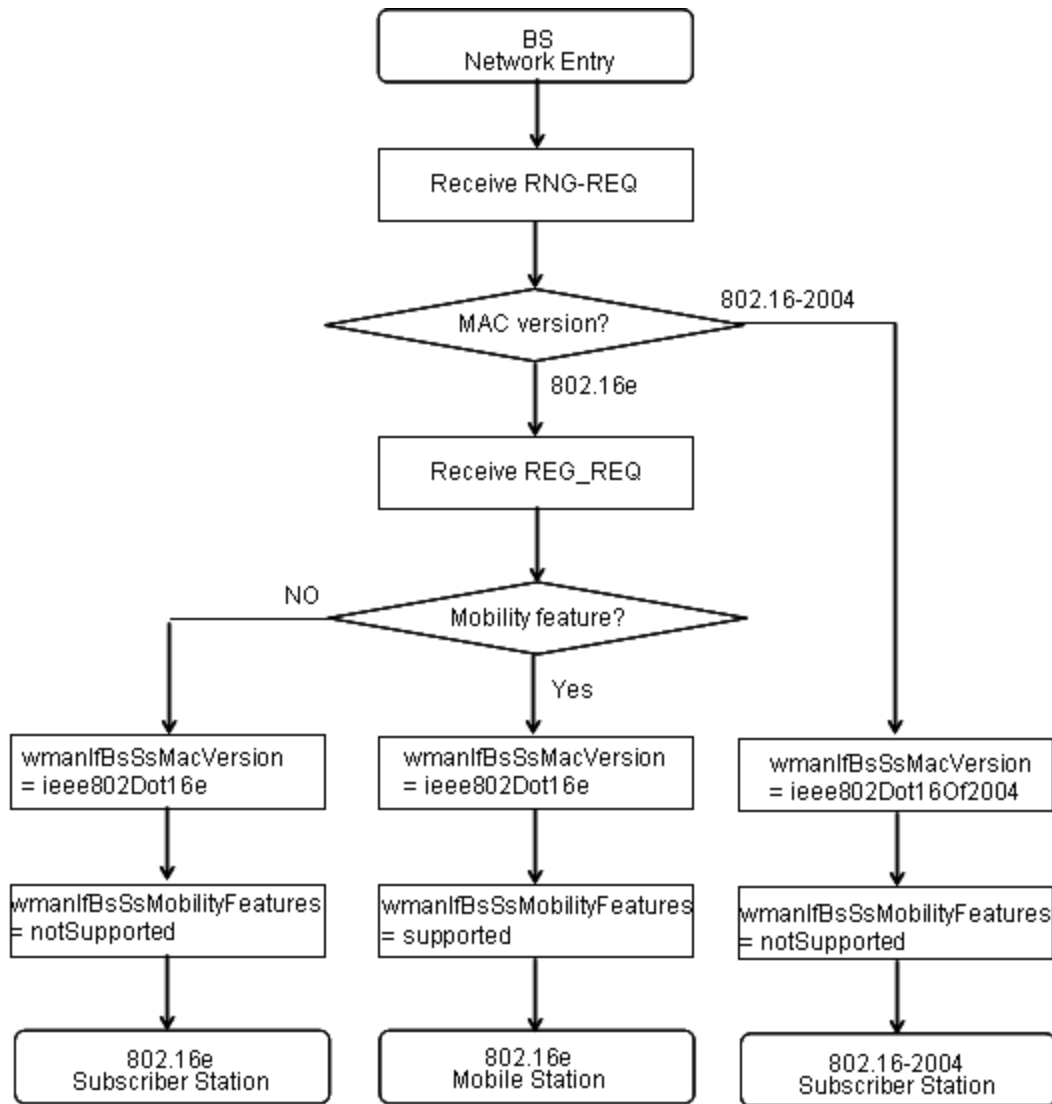


Figure 5—SS / MS Network Entry

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
4. If Mobility Feature is supported, then
 - a) wmanIfBsSsMacVersion = ieee802Dot16e
 - b) wmanIfBsSsMobilityFeatures = Supported
 otherwise

- 1 a) wmanIfBsSsMacVersion = ieee802Dot16e
- 2
- 3 b) wmanIfBsSsMobilityFeatures = Not Supported
- 4
- 5 5. Continue network entry procedure
- 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

For the purpose of Management Interface development an Interface Methodology known as Integration Reference Point (IRP) was developed to promote the wider adoption of standardized Management interfaces in telecommunication networks. The IRP methodology employs Protocol & Technology Neutral modeling methods as well as protocol specific solution sets to help achieve its goals. The Integration Reference Point is a methodology to aid a modular approach to the development of standards interfaces.

There are three cornerstones to the IRP approach:

1. Top-down, process-driven modeling approach

The process begins with a requirements phase, the aim at this step is to provide conceptual and use case definitions for a specific interface aspect as well as defining subsequent requirements for this IRP.

2. Technology-independent modeling

The second phase of the process is the development of a protocol independent model of the interface. This protocol independent model is specified in the IRP Information Service.

3. Standards-based technology-dependent modeling

The third phase of the process is to create one or more interface technology and protocol dependent models from the Information Service model. This is specified in the IRP Solution Set(s).

15.1 NRM IRP IS

15.1.1 Information Service Models

Information Service Models refer to both Interface IRPs and NRM IRPs.

This section is providing the IEEE 802.16 protocol neutral (IS) resource model (NRM/MIB) definitions.

15.1.1.1 Information entities imported and local labels

Table 3—Information entities imported and local labels

Label reference	Local label
information object class, ManagedElement	ManagedElement
information object class, ManagedFunction	ManagedFunction
information object class, SubNetwork	SubNetwork
information object class, Top	Top

15.1.1.2 Class diagram

15.1.1.2.1 Attributes and relationships

Figure 1. establishes the naming and containment for the protocol neutral network management models of the 802.16 standard. The inheritance diagram show in Figure 2. is based on 802.16e and 802.16-2004. This diagram establishes the context of the IOC and shows ME's as inventory items and MF's as the functions that perform functions in the 802.16 network.

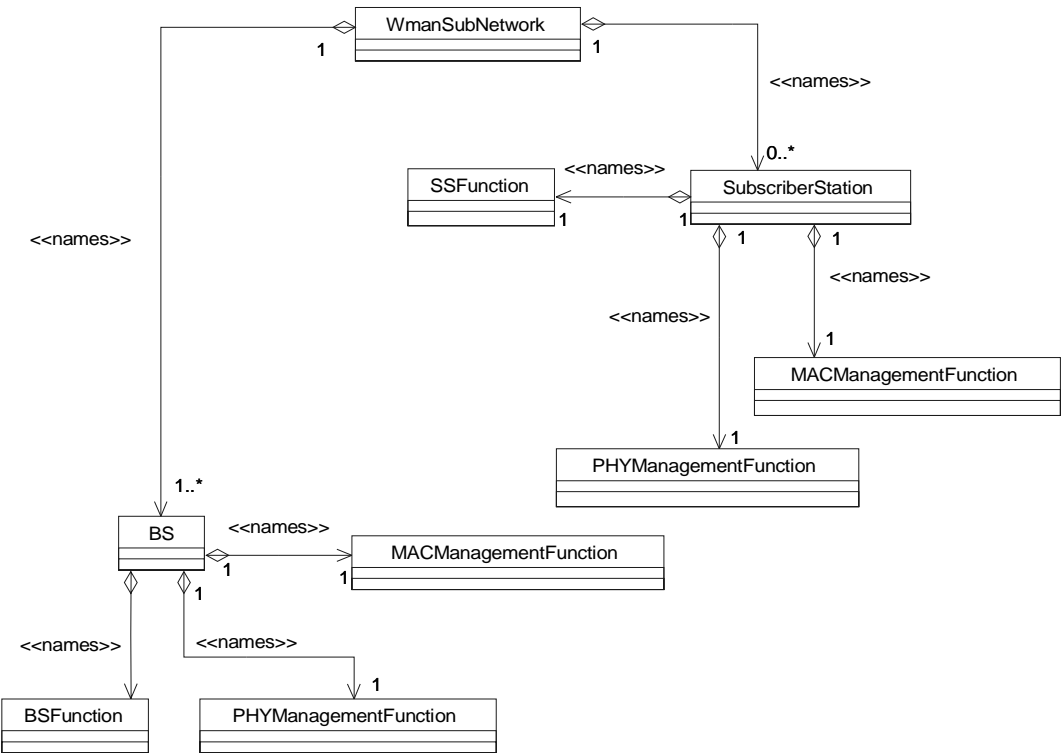


Figure 6—Containment and Naming Diagram

15.1.1.2.2 Inheritance

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

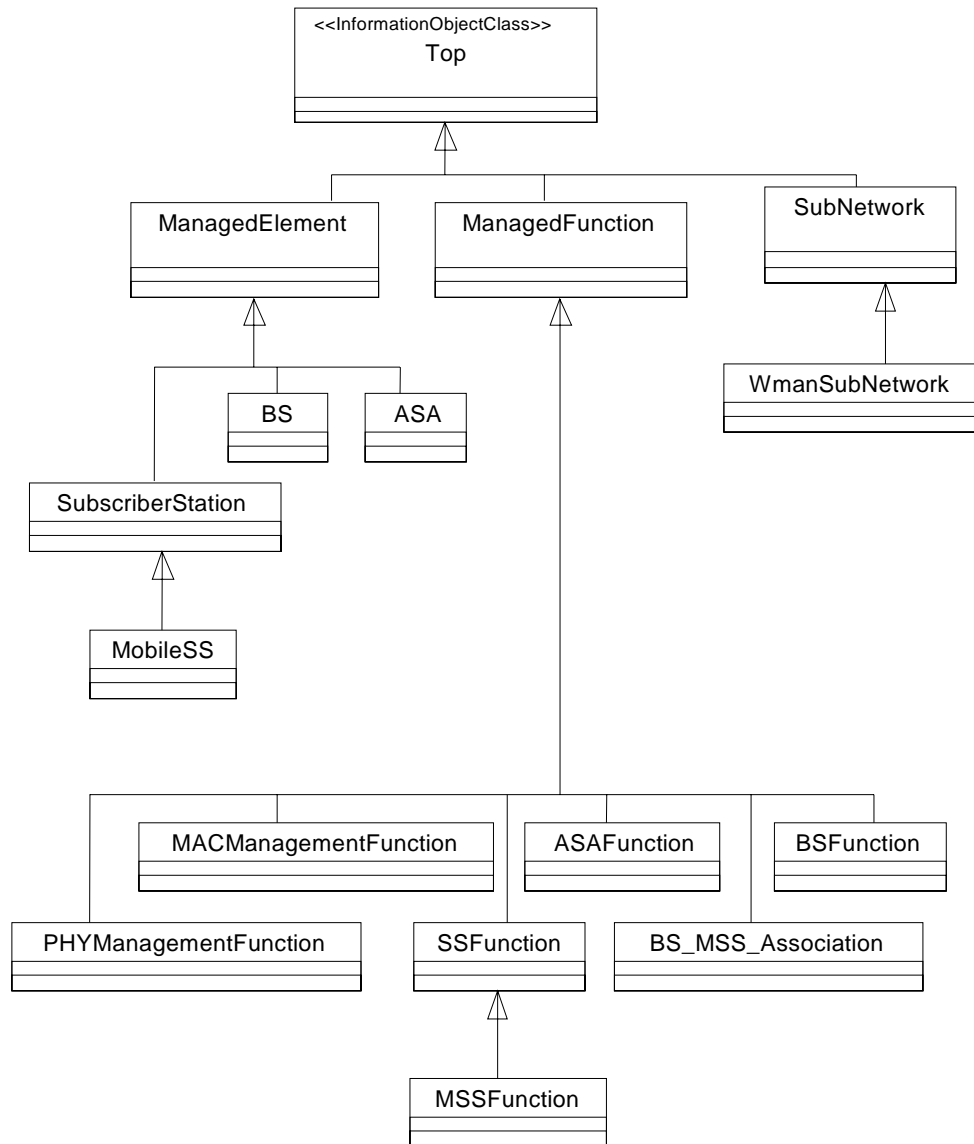


Figure 7—Inheritance Diagram

15.1.1.3 Information object classes definition

15.1.1.3.1 IOC BsFunction

15.1.1.3.1.1 Definition

This IOC represents a WMAN base station. For more information, see [zz]. It is derived from Managed-Function.

<Section Note: This table is just a template for reference.>

15.1.1.3.1.2 Attributes

Table 4—Attributes

Attribute name	Defined in	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
BsFunctionId	--	+	M	M	--
objectClass	Top	+ _{inherited}	M ^{inherited}	M ^{inherited}	-- _{inherited}
objectInstance	Top	+ _{inherited}	M ^{inherited}	M ^{inherited}	-- _{inherited}
userLabel	ManagedFunction	+ _{inherited}	M ^{inherited}	M ^{inherited}	M ^{inherited}
aaa	--	+	O	M	--
bbb	--	+	O	M	--
yyy	--	+	O	M	--
zzz	--	+	O	M	--

15.1.1.3.2 IOC WmanSsFunction

15.1.1.3.2.1 Definition

This IOC represents a WMAN subscriber station. For more information, see [tbd]. It is derived from ManagedFunction.

15.1.1.3.2.2 Attributes

Table 5—Attributes

Attribute name	Defined in	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
SsFunctionId	--	+	M	M	--
objectClass	Top	+ _{inherited}	M _{inherited}	M _{inherited}	-- _{inherited}
objectInstance	Top	+ _{inherited}	M _{inherited}	M _{inherited}	-- _{inherited}
userLabel	ManagedFunction	+ _{inherited}	M _{inherited}	M _{inherited}	M _{inherited}
ccc	--	+	O	M	--
ddd	--	+	O	M	--
www	--	+	O	M	--
xxx	--	+	O	M	--

15.1.1.3.3 IOC xxx

15.1.1.3.4 IOC yyy

15.1.1.4 Information relationships definition

15.1.1.5 Notifications

15.1.1.6 Information attributes definition

15.1.1.6.1 Definition and legal values

Table 6—Definition and legal values

Attribute name	Definition	Legal Values
BsFunctionId	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.	--
SsFunctionId		--
ZzzId		--
aaa	tbd	tbd
bbb	tbd	tbd
ccc	tbd	tbd
ddd	tbd	tbd
objectClass	As defined in [zz]: An attribute which captures the name of the class from which the object instance is an occurrence of.	--

15.1.2 Proposal for BS Related Objects NRM Definitions

15.1.2.1 Information entities imported and local labels:

Table 7—Information entities imported and local labels

Label reference	Local label
information object class, ManagedElement	ManagedElement
information object class, ManagedFunction	ManagedFunction
information object class, SubNetwork	SubNetwork
information object class, Top	Top
information object class, BS	BS
Information object class, BSFunction	BSFunction
information object class, ExternalBSFunction	ExternalBSFunction
information object class, BSRelation	BSRelation
Information object class, PagingGroup	PagingGroup
Information object class, CommonFunction	CommonFunction
Information object class, MSFunction	MSFunction
information object class, ExternalBSFunction	ExternalBSFunction

Table 7—Information entities imported and local labels

Label reference	Local label
information object class, BSRelation	BSRelation
Information object class, PacketCSMngtFunction	PacketCSMngtFunction
Information object class, ClassifierRule	ClassifierRule
Information object class, ProvisionedSS	ProvisionedSS
Information object class, PHSRule	PHSRule
Information object class, SSProvisionedForSF	SSProvisionedForSF
Information object class, ServiceClass	ServiceClass
Information object class, SecurityMngFunction	SecurityMngFunction
Information object class, PKMBase	PKMBase
Information object class, PKMTEK	PKMTEK
Information object class, SSPKMAuth	SSPKMAuth
Information object class, CryptoSuite	CryptoSuite
Information object class, PHYMngFunction	PHYMngFunction
Information object class, UCDBurstProfile	UCDBurstProfile
Information object class, DCDBurstProfile	DCDBurstProfile
Information object class, PowerCtrl	PowerCtrl
Information object class, DownLinkChannel	DownLinkChannel
Information object class, UplinkChannel	UplinkChannel
Information object class, CPSMngFunction	CPSMngFunction
Information object class, BasicCapabilities	BasicCapabilities
Information object class, PowerSavingClass	PowerSavingClass
Information object class, MBSZone	MBSZone
Information object class, MBSServiceFlow	MBSServiceFlow
Information object class, RegisteredMS	RegisteredMS
Information object class, IdleModeMS	IdleModeMS
Information object class, SleepModeMS	SleepModeMS

15.1.2.2 Class diagram

15.1.2.2.1 Attributes and relationships

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.

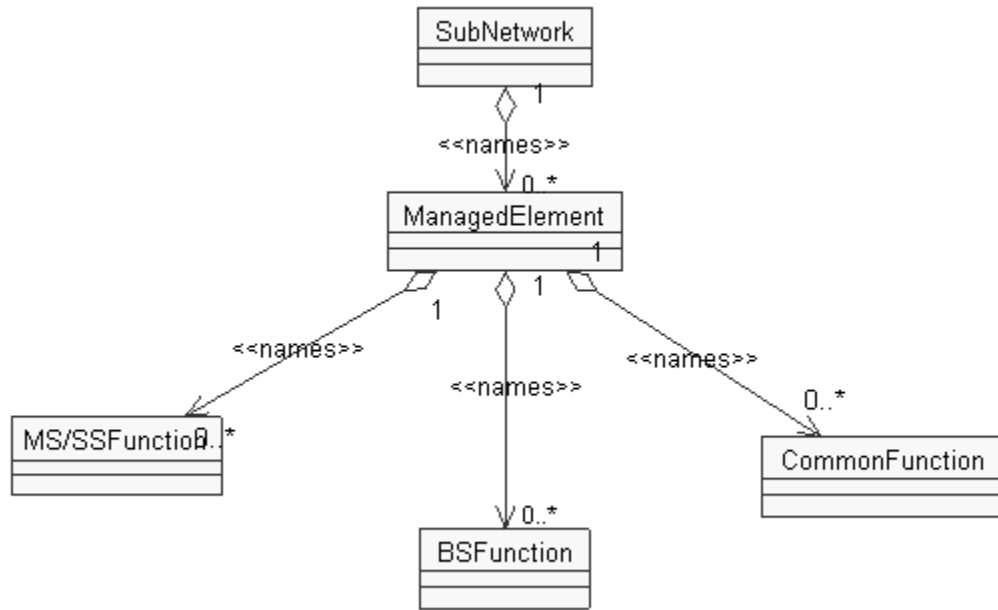


Figure 8—General View

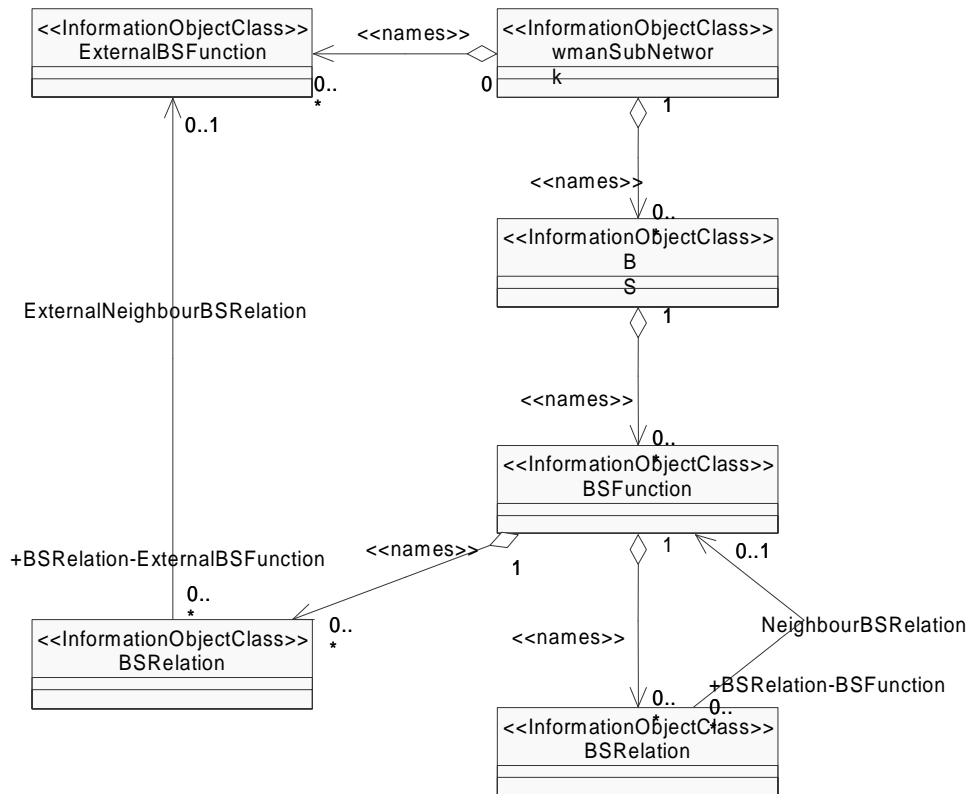


Figure 9—Segment view Neighbouring BS NRM Containment/Naming Diagram

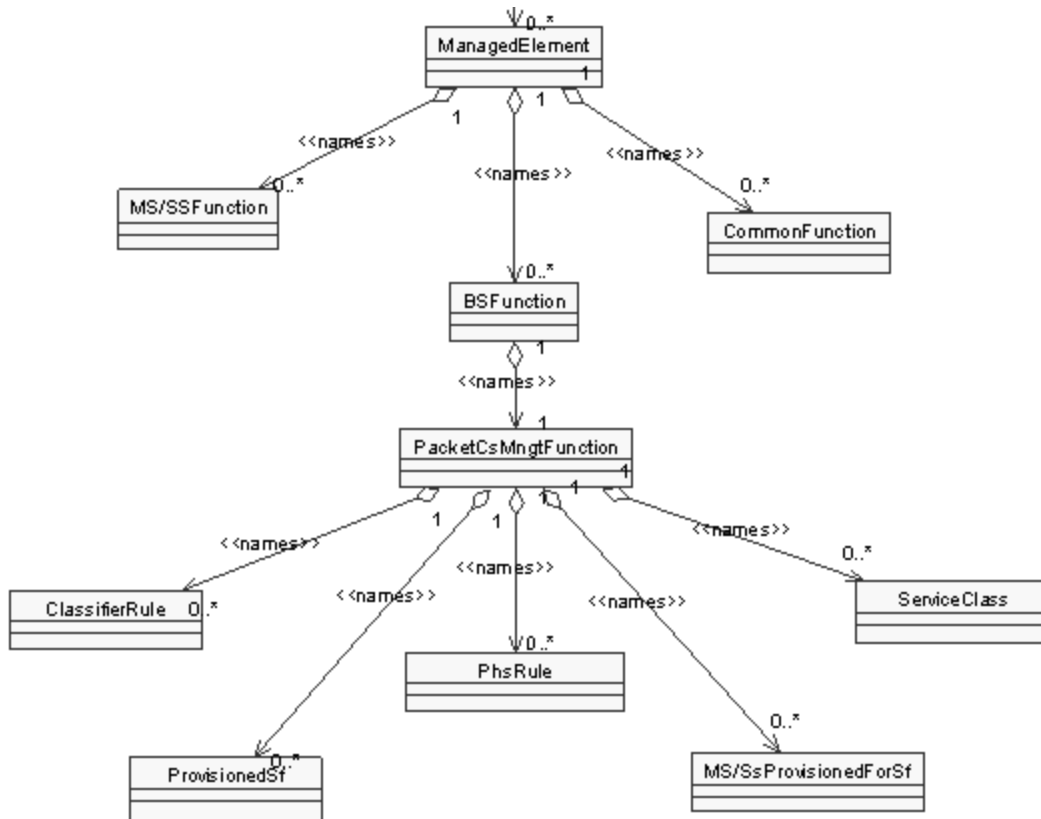


Figure 10—Segment view PacketCs NRM Containment/Naming Diagram

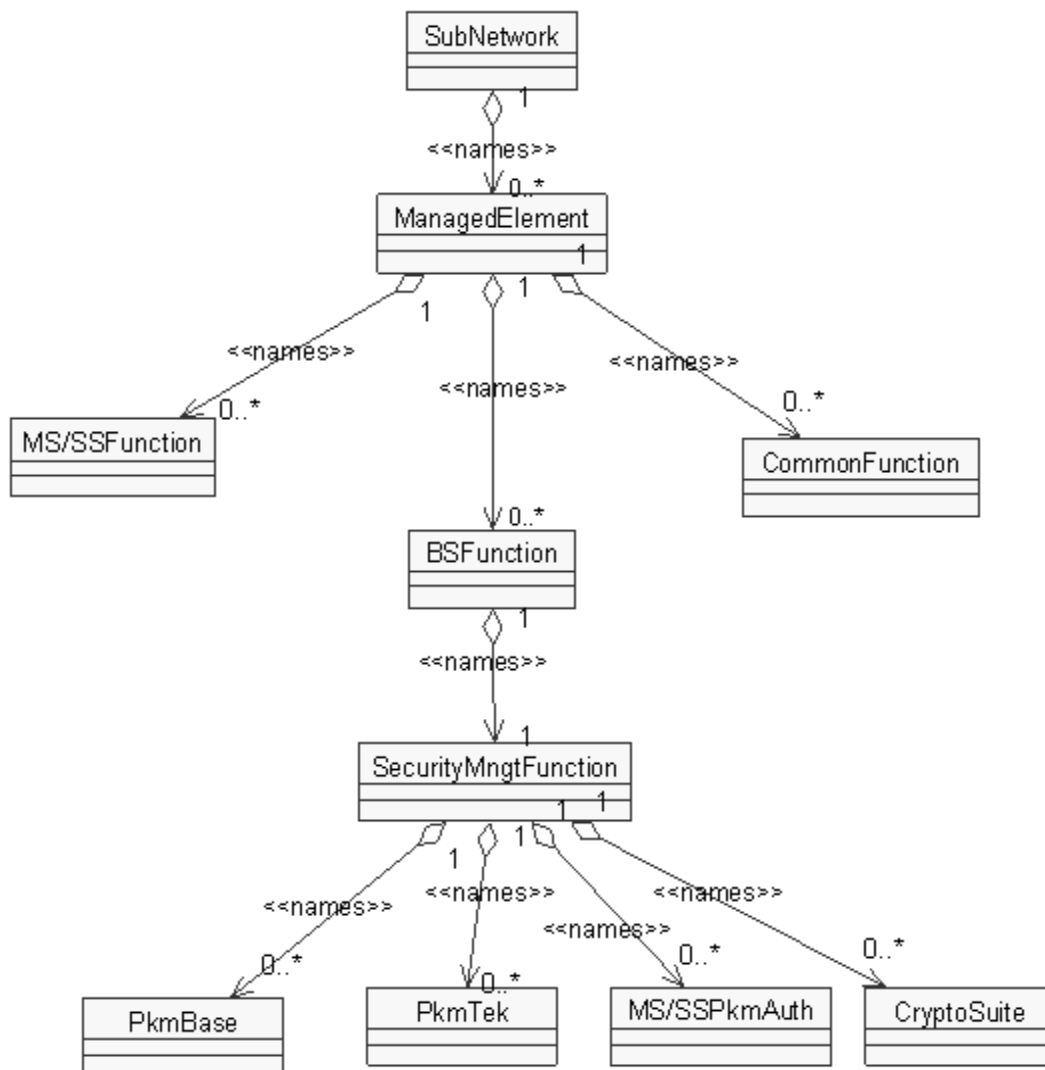


Figure 11—Segment view SecurityMng NRM Containment/Naming Diagram

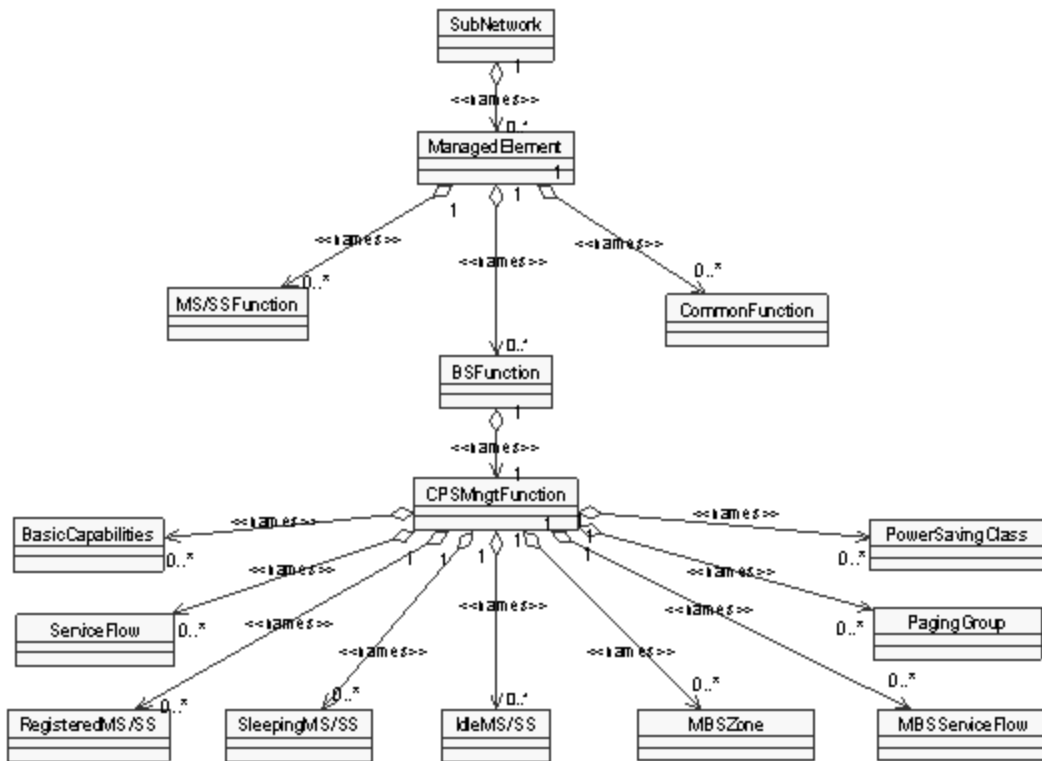


Figure 12—Segment view CPSMngMng NRM Containment/Naming Diagram

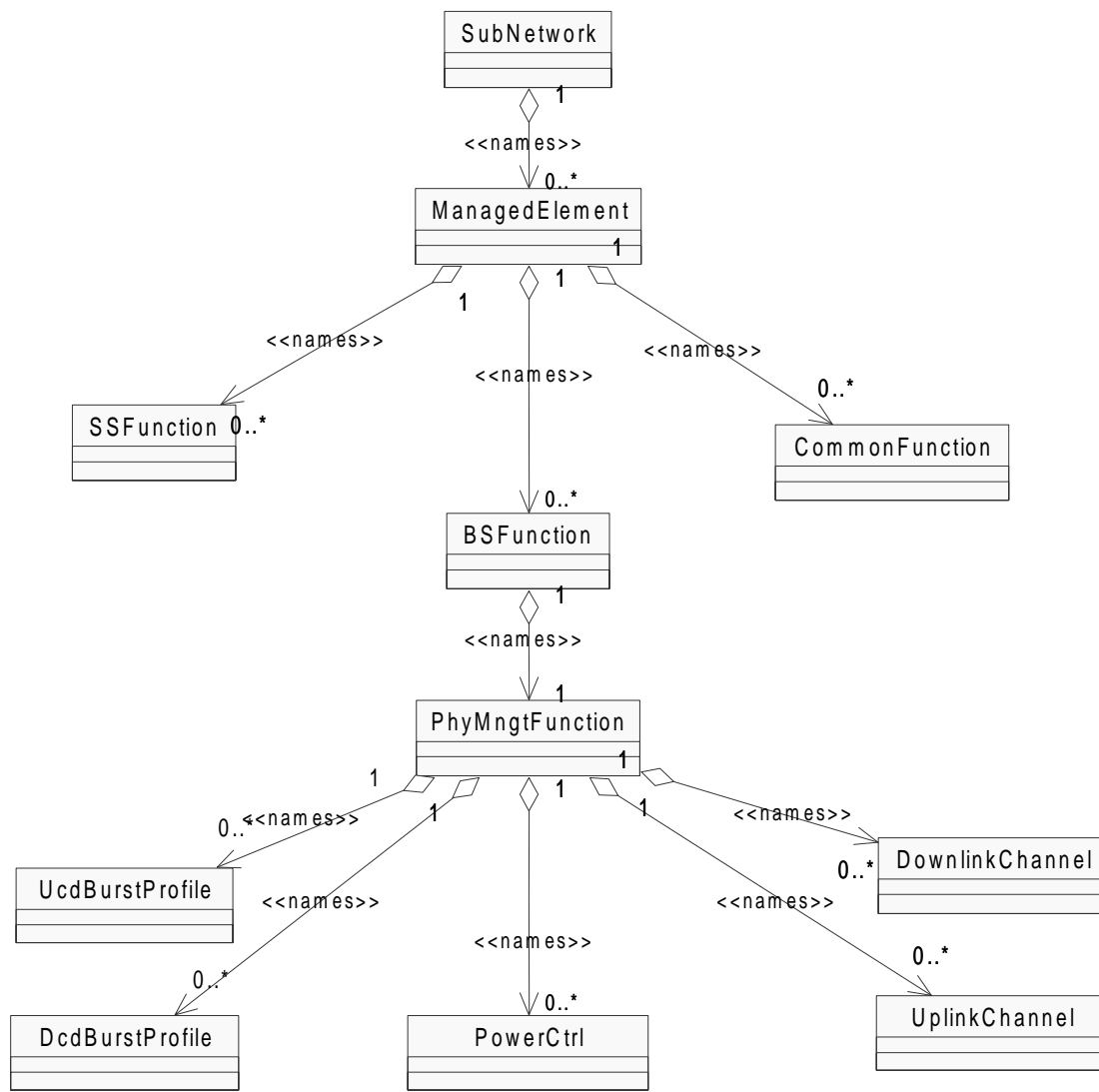


Figure 13—Segment view PhyMng NRM Containment/Naming Diagram

15.1.2.2.2 Inheritance

The inheritance diagram show below is, is based on 802.16e and 802.16-2004. This diagram establishes the context of the IOC and shows ME's as inventory items and MF's as the functions that perform functions in the 802.16 network.

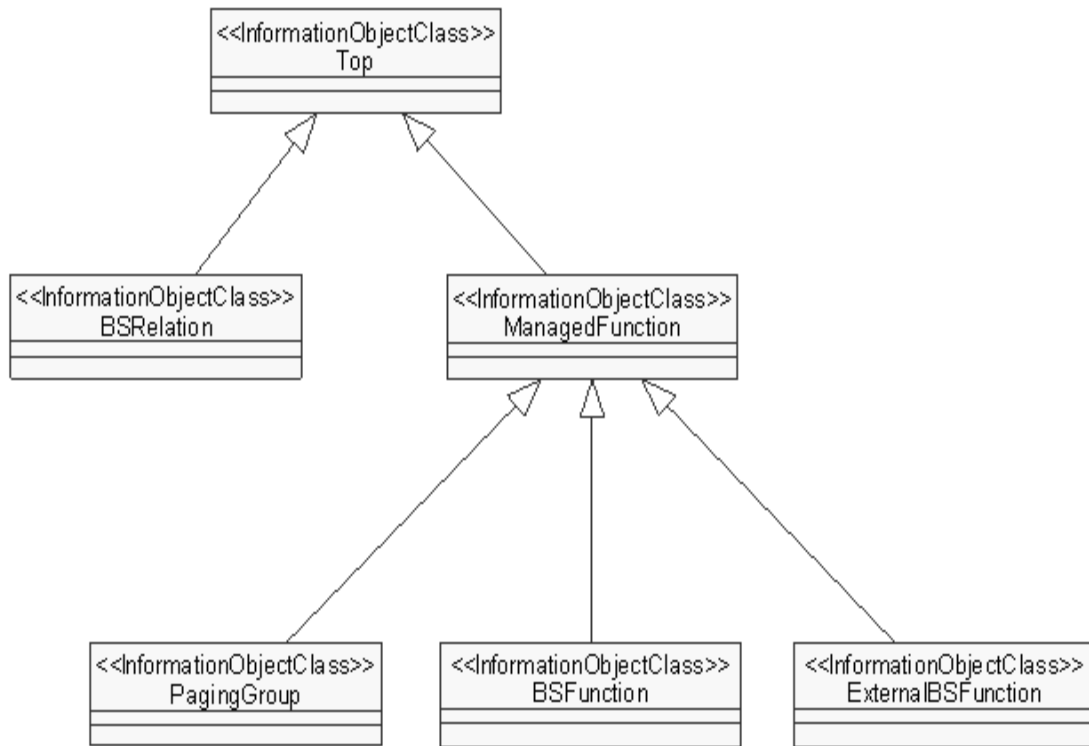


Figure 14—Inheritance Diagram

The PacketCS IOC inheritance hierarchy:

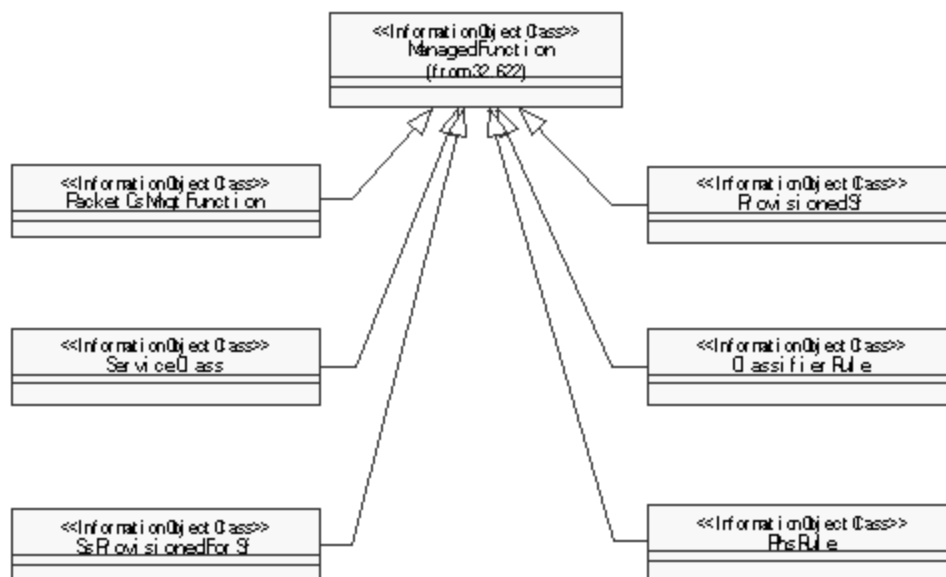
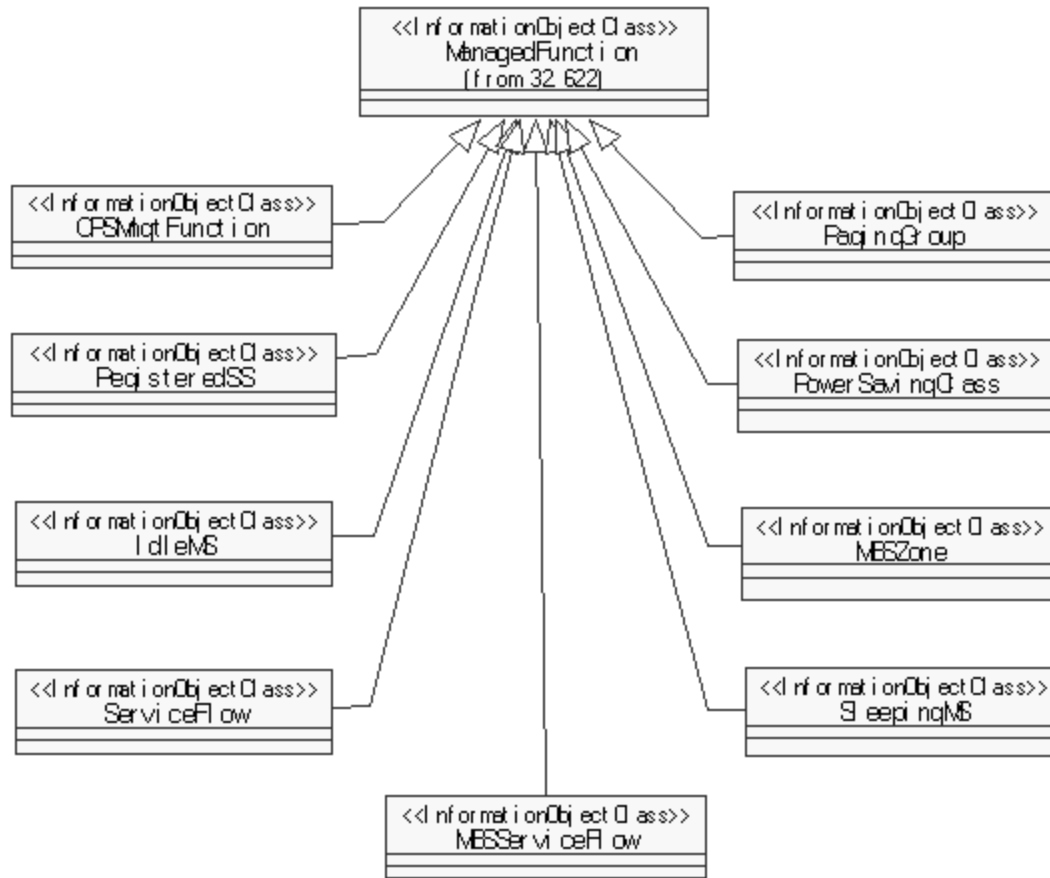


Figure 15—IEEE 802.16 NRM PacketCS Inheritance Diagram

The CPS IOC inheritance hierarchy:

**Figure 16—IEEE 802.16 NRM CPS Inheritance Diagram**

The Security sublayer IOC inheritance hierarchy:

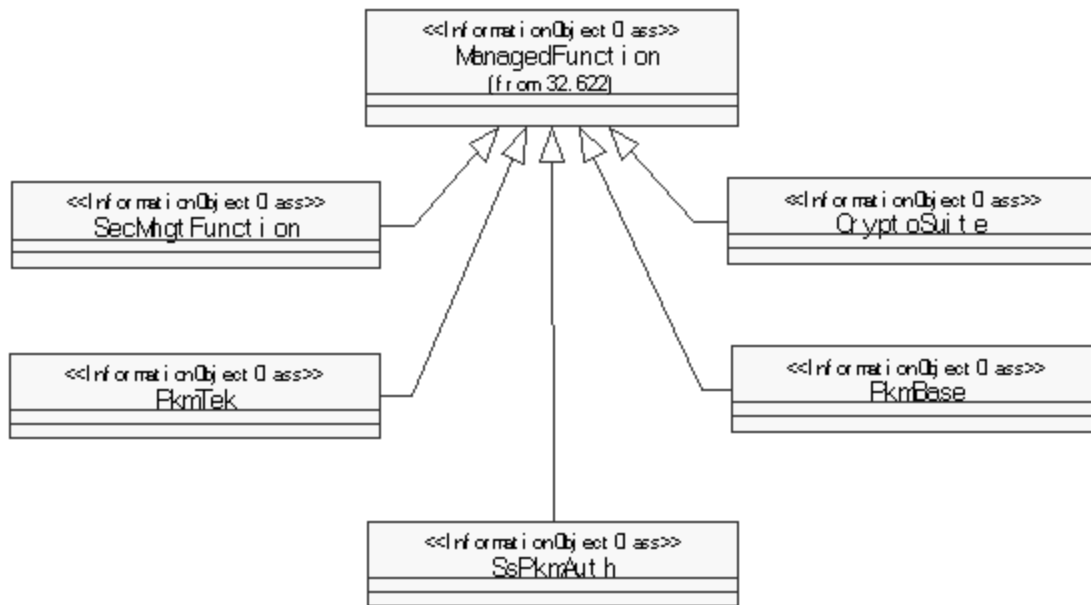


Figure 17—IEEE 802.16 NRM Security Sublayer Inheritance Diagram

The PHY layer IOC inheritance hierarchy:

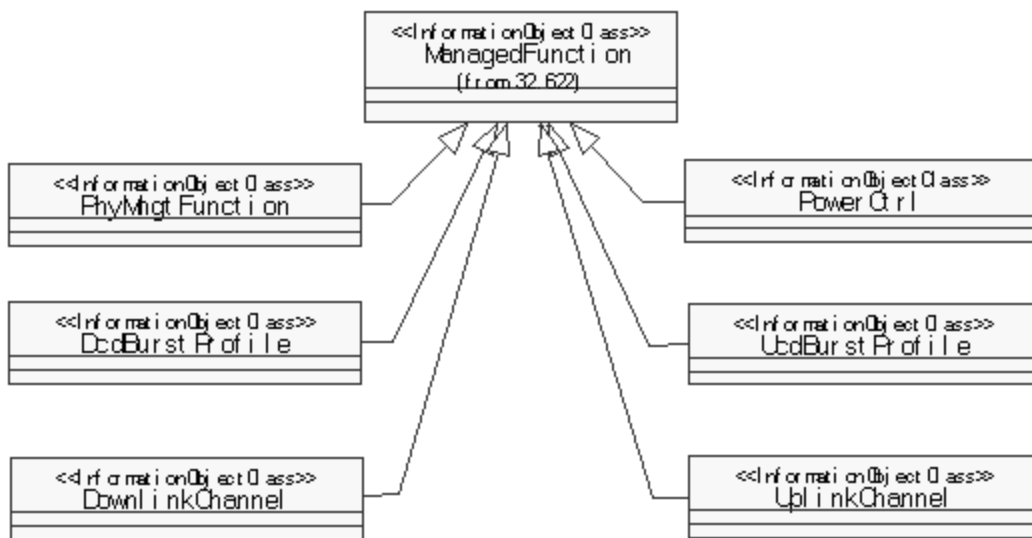


Figure 18—IEEE802.16 NRM PHY Layer Inheritance Diagram

15.1.2.3 Information object classes definition

15.1.2.3.1 IOC BSFunction

15.1.2.3.1.1 Definition

This IOC represents a WMAN base station. It is derived from ManagedFunction

15.1.2.3.1.2 Attributes

Table 8—Attributes of BSFunction

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
MSHOCOnnectionProcessTime	+	M	M	M
MSHOTeKProcessTime	+	M	M	M
ULPermutationBase	+	M	M	M
DLPermutationBase	+	M	M	M
PreambleIndex	+	M	M	M
SegmentNumber	+	M	M	M

15.1.2.3.2 IOC ExternalBSFunction

15.1.2.3.2.1 Definition

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

15.1.2.3.2.2 Attributes

Table 9—Attributes of ExternalBSFunction

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

15.1.2.3.3 IOC BSRelation

15.1.2.3.3.1 Definition

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

15.1.2.3.3.2 Attributes

Table 10—Attributes of BSRelation

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

15.1.2.3.4 IOC PagingGroup

15.1.2.3.4.1 Definition

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

15.1.2.3.4.2 Attributes

Table 11—Attributes of PagingGroup

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

Table 11—Attributes of PagingGroup

Attribute name	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
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

15.1.2.3.5 IOC PowerCtrl**15.1.2.3.5.1 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 ManagedFunction.

15.1.2.3.5.2 Attributes**Table 12—PowerCtrl Attributes**

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	ManagedFunction	+inherited	Minherited	Minherited	Minherited
msUpPwrAdjStep	--	+	M	M	O
msDownPwrAdjStep	--	+	M	M	O
minPwrAdjLever	--	+	M	M	O
maxPwrAdjLever	--	+	M	M	O
txPwrRepThreshold	--	+	M	M	O

Table 12—PowerCtrl Attributes

Attribute name	Defined in	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
txPwrRepInterval	--	+	M	M	O
alphaPAvg	--	+	M	M	O
txPwrRepThresholdCQI	--	+	M	M	O
txPwrRepIntervalCQI	--	+	M	M	O
alphaPAvgCQI	--	+	M	M	O

15.1.2.3.6 IOC SecurityManagementFunction**15.1.2.3.6.1 Definition**

This IOC represents a SecurityManagementFunction object. It is derived from ManagedFunction

15.1.2.3.6.2 Attributes**Table 13—Attributes of SecurityManagementFunction**

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

15.1.2.3.7 IOC PkmBase**15.1.2.3.7.1 Definition**

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

15.1.2.3.7.2 Attributes

Table 14—Attributes of PkmBase

Attribute name	Defined in	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
objectClass	Top	+inherited	Minherited	Minherited	--inherited
objectInstance	Top	+inherited	Minherited	Minherited	--inherited
userLabel	Managed-Function	+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

15.1.2.3.8 IOC PkmTek

15.1.2.3.8.1 Definition

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

15.1.2.3.8.2 Attributes

Table 15—Attributes of PkmTek

Attribute name	Defined in	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
objectClass	Top	+inherited	Minherited	Minherited	--inherited
objectInstance	Top	+inherited	Minherited	Minherited	--inherited
userLabel	Managed-Function	+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	-

15.1.2.3.9 IOC MS/SSPkmAuth

15.1.2.3.9.1 Definition

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

15.1.2.3.9.2 Attributes

Table 16—Attributes of MSPkmAuth

Attribute name	Defined in	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
objectClass	Top	+inherited	Minherited	Minherited	--inherited
objectInstance	Top	+inherited	Minherited	Minherited	--inherited
userLabel	Managed-Function	+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	

15.1.2.4 Information relationships definition

15.1.2.4.1 ExternalNeighbourBSRelation

15.1.2.4.1.1 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.

15.1.2.4.1.2 Roles

Table 17—Roles of the relation ExternalNeighbourBSRelation

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

15.1.2.4.1.3 Constraints

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

15.1.2.5 Notifications

15.1.2.6 Information attributes definition

15.1.2.6.1 Definition and legal values

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

Table 18—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 combin- ing supported monitor- ing single MAP from anchor BS(2), MDHO DL soft combin- ing supported monitor- ing MAPS from active BSs(3))
SystemResourceRetain-Time	The Resource_Retain_Time is the duration for MS s con- nection information that will be retained in serving BS. BS shall start Resource_Retain_Time timer at MS notifi- cation of pending HO attempt through MOB_HO-IND or by detecting an MS drop. The unit of this value is 100 milliseconds.	

Table 18—Information attributes definition

Attribute Name	Definition	Legal Values
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.	
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.	
FFTSsize	Indicate neighbouring BS FFT size	
CyclePrefix	indicate neighbouring BS Cycle Prefix	

Table 18—Information attributes definition

Attribute Name	Definition	Legal Values
FramDurationCode	Indicate neighbouring BS Frame duration code	
ULPermutationBase	Indicate neighbouring BS uplink permutation base.	
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 18—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 18—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)		

Table 18—Information attributes definition

Attribute Name	Definition	Legal Values
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.	
wmanIfBsPkmTekExpiresOld	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.	
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.	

Table 18—Information attributes definition

Attribute Name	Definition	Legal Values
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.	
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	
wmanIfBsKeyPushCounter	Protect for replay attack.	

15.2 NRM IRP SNMP Solution Set

15.2.1 NRM IRP SNMP Solution Set

15.2.1.1.5 wmanIfBsPhy



Figure 19—wmanIfBsPhy structure

15.2.1.1.5.1 wmanIfBsOfdmaConfigurationTable

wmanIfBsOfdmaConfigurationTable contains BS configuration objects, specific to OFDMA PHY.

15.2.1.1.5.2 wmanIfBsSsOfdmaReqCapabilitiesTable

wmanIfBsSsOfdmaReqCapabilitiesTable 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.

15.2.1.1.5.3 wmanIfBsSsOfdmaRspCapabilitiesTable

wmanIfBsSsOfdmaRspCapabilitiesTable 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 wmanIfBsRegisteredSsTable.

15.2.1.1.5.4 wmanIfBsOfdmaCapabilitiesTable

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

15.2.1.1.5.5 wmanIfBsOfdmaCapabilitiesConfigTable

wmanIfBsOfdmaCapabilitiesConfigTable 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.

15.2.2 ASN.1 Definitions of 802.16 MIB for SNMP

WMAN-IF-MIB DEFINITIONS ::= BEGIN

IMPORTS

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

wmanIfMib MODULE-IDENTITY

LAST-UPDATED "200603270000Z" -- March 27, 2006
 ORGANIZATION "IEEE 802.16"
 CONTACT-INFO
 "WG E-mail: stds-802-16@ieee.org
 WG Chair: Roger B. Marks
 Postal: (U.S.) National Institute
 of Standards and Technology
 E-mail: r.b.marks@ieee.org

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

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

DESCRIPTION

"This material is from IEEE Std 802.16f
 Copyright (c) 2005 IEEE.
 This MIB Module defines managed objects for
 IEEE 802.16-2004 based Subscriber Station
 and Base Station."

REVISION "200603270000Z"

DESCRIPTION

"The second version of WMAN-IF-MIB module that is

```

1          enhanced to support IEEE 802.16e-2005 standard."
2  REVISION          "200508020000Z"
3  DESCRIPTION
4
5      "The first approved version of WMAN-IF-MIB module."
6  ::= { transmission 184 }
7
8  wmanIfMibObjects OBJECT IDENTIFIER ::= { wmanIfMib 1 }
9  wmanIfBsObjects OBJECT IDENTIFIER ::= { wmanIfMibObjects 1 }
10 wmanIfSsObjects OBJECT IDENTIFIER ::= { wmanIfMibObjects 2 }
11 wmanIfCommonObjects OBJECT IDENTIFIER ::= { wmanIfMibObjects 3 }
12
13
14 -- Textual Conventions
15 WmanIfSfSchedulingType ::= 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 WmanIfPhsRuleVerify ::= 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 WmanIfClassifierBitMap ::= 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 WmanIfSfState ::= TEXTUAL-CONVENTION
15     STATUS      current
16     DESCRIPTION
17         "WmanIfSfState defines the state of a service flow."
18     SYNTAX      INTEGER {authorized(1),
19                     admitted(2),
20                     active(3)}
21
22 WmanIfServClassName ::= TEXTUAL-CONVENTION
23     STATUS      current
24     DESCRIPTION
25         "WmanIfServClassName defines the type of service
26         class name."
27     SYNTAX      OCTET STRING (SIZE(2..128))
28
29 WmanIfCsSpecification ::= TEXTUAL-CONVENTION
30     STATUS      current
31     DESCRIPTION
32         "WmanIfCsSpecification defines the types of convergence
33         sublayer."
34     SYNTAX      INTEGER {noCs(0),
35                     packetIPv4(1),
36                     packetIPv6(2),
37                     packet802dot3Ethernet(3),
38                     packet802dot1QVlan(4),
39                     packetIPv4Over802dot3(5),
40                     packetIPv6Over802dot3(6),
41                     packetIPv4Over802dot1Q(7),
42                     packetIPv6Over802dot1Q(8),
43                     atm(9)}
44
45 WmanIfMacVersion ::= TEXTUAL-CONVENTION
46     STATUS      current
47     DESCRIPTION
48         "Version number of IEEE 802.16."
49     SYNTAX      INTEGER {ieee802Dot16Of2001(1),
50                     ieee802Dot16cOf2002(2),
51                     ieee802Dot16aOf2003(3),
52                     ieee802Dot16Of2004(4)}
53
54 WmanIfCidType ::= TEXTUAL-CONVENTION

```

```

1      STATUS      current
2      DESCRIPTION
3          "Type of CID."
4      SYNTAX      INTEGER (0 .. 65535)
5
6
7      WmanIfDataEncryptAlgId ::= TEXTUAL-CONVENTION
8          STATUS      current
9          DESCRIPTION
10             "Data encryption algorithm identifiers."
11          REFERENCE
12             "Table 375 in IEEE Std 802.16-2004"
13          SYNTAX      INTEGER {none(0),
14                               des56BitCbcMode(1),
15                               aesCcmMode(2)}
16
17
18
19      WmanIfDataAuthAlgId ::= TEXTUAL-CONVENTION
20          STATUS      current
21          DESCRIPTION
22             "Data authentication algorithm identifiers."
23          REFERENCE
24             "Table 376 in IEEE Std 802.16-2004"
25          SYNTAX      INTEGER {noDataAuthentication(0),
26                               reserved(1)}
27
28
29
30      WmanIfTekEncryptAlgId ::= TEXTUAL-CONVENTION
31          STATUS      current
32          DESCRIPTION
33             "TEK encryption algorithm identifiers."
34          REFERENCE
35             "Table 377 in IEEE Std 802.16-2004"
36          SYNTAX      INTEGER {tripleDes128BitKey(1),
37                               rsa1024BitKey(2),
38                               aes128BitKey(3)}
39
40
41
42      WmanIfChannelNumber ::= TEXTUAL-CONVENTION
43          STATUS      current
44          DESCRIPTION
45             "Physical channel number"
46          SYNTAX      INTEGER (0 .. 199)
47
48
49
50      WmanIfOfdmFecCodeType ::= TEXTUAL-CONVENTION
51          STATUS      current
52          DESCRIPTION
53             "FEC code type and modulation type"
54          REFERENCE
55             "Table 356 and Table 362 in IEEE Std 802.16-2004"
56          SYNTAX      INTEGER {bpskCc1Over2(0),
57                               qpskRsCcCc1Over2(1),
58                               qpskRsCcCc3Over4(2),
59                               sixteenQamRsCcCc1Over2(3),
60                               sixteenQamRsCcCc3Over4(4),
61                               sixtyFourQamRsCcCc2Over3(5),
62                               sixtyFourQamRsCcCc3Over4(6),
63                               qpskBtclOver2(7),
64
65

```

```

1          qpskBtc3Over4(8),
2          sixteenQamBtc3Over4(9),
3          sixteenQamBtc4Over5(10),
4          sixtyFourQamBtc2Over3(11),
5          sixtyFourQamBtc5Over6(12),
6          qpskCtc1Over2(13),
7          qpskCtc2Over3(14),
8          qpskCtc3Over4(15),
9          sixteenQamCtc1Over2(16),
10         sixteenQamCtc3Over4(17),
11         sixtyFourQamCtc2Over3(18),
12         sixtyFourQamCtc3Over4(19) }
13
14
15
16
17 WmanIfOfdmaFecCodeType ::= TEXTUAL-CONVENTION
18     STATUS          current
19     DESCRIPTION
20         "FEC code type and modulation type"
21     REFERENCE
22         "Table 356 and Table 362 in IEEE Std 802.16-2004"
23     SYNTAX          INTEGER { qpskCc1Over2(0),
24                             qpskCc3Over4(1),
25                             sixteenQamCc1Over2(2),
26                             sixteenQamCc3Over4(3),
27                             sixtyFourQamCc2Over3(4),
28                             sixtyFourQamCc3Over4(5),
29                             qpskBtc1Over2(6),
30                             qpskBtc2Over3(7),
31                             sixteenQamBtc3Over5(8),
32                             sixteenQamBtc4Over5(9),
33                             sixtyFourQamBtc5Over8(10),
34                             sixtyFourQamBtc4Over5(11),
35                             qpskCtc1Over2(12),
36                             qpskCtc2Over3(13),
37                             qpskCtc3Over4(14),
38                             sixteenQamCtc1Over2(15),
39                             sixteenQamCtc3Over4(16),
40                             sixtyFourQamCtc2Over3(17),
41                             sixtyFourQamCtc3Over4(18),
42                             sixtyFourQamCtc5Over6(19),
43                             qpskZtCc1Over2(20),
44                             qpskZtCc3Over4(21),
45                             sixteenQamZtCc1Over2(22),
46                             sixteenQamZtCc3Over4(23),
47                             sixtyFourQamZtCc2Over3(24),
48                             sixtyFourQamZtCc3Over4(25) }
49
50
51
52
53
54
55
56 -- Textual convention for capabilities encodings
57 WmanIfNumOfUplinkCid ::= TEXTUAL-CONVENTION
58     STATUS          current
59     DESCRIPTION
60         "The object of this type shows the number of Uplink CIDs
61         the SS can support."
62     REFERENCE
63         "Subclause 11.7.4 in IEEE Std 802.16-2004"
64
65

```

```

1          SYNTAX          INTEGER (2..65535)
2
3
4 WmanIfArqSupportType ::= TEXTUAL-CONVENTION
5     STATUS          current
6     DESCRIPTION
7         "The object of this type indicates whether the SS support
8         ARQ."
9
10    REFERENCE
11        "Subclause 11.7.8.1 in IEEE Std 802.16-2004"
12    SYNTAX          INTEGER {arqNotSupported(0),
13                          arqSupported(1)}
14
15 WmanIfMaxDsxFlowType ::= TEXTUAL-CONVENTION
16     STATUS          current
17     DESCRIPTION
18         "The object of this type specifies the maximum number of
19         concurrent DSA, DSC, or DSD transactions that may be
20         outstanding."
21
22    REFERENCE
23        "Subclause 11.7.8.2 in IEEE Std 802.16-2004"
24    SYNTAX          INTEGER (0..255)
25
26
27 WmanIfMacCrcSupport ::= TEXTUAL-CONVENTION
28     STATUS          current
29     DESCRIPTION
30         "The object of this type indicates whether or not the SS
31         supports MAC level CRC."
32
33    REFERENCE
34        "Subclause 11.7.8.3 in IEEE Std 802.16-2004"
35    SYNTAX          INTEGER {noMacCrcSupport(0),
36                          macCrcSupport(1)}
37
38
39 WmanIfMaxMcaFlowType ::= TEXTUAL-CONVENTION
40     STATUS          current
41     DESCRIPTION
42         "The object of this type specifies the maximum number of
43         concurrent MCA transactions that may be outstanding."
44
45    REFERENCE
46        "Subclause 11.7.8.4 in IEEE Std 802.16-2004"
47    SYNTAX          INTEGER (0..255)
48
49
50 WmanIfMaxMcpGroupCid ::= TEXTUAL-CONVENTION
51     STATUS          current
52     DESCRIPTION
53         "The object of this type indicates the maximum number of
54         simultaneous Multicast Polling Groups the SS is
55         capable of belonging to."
56
57    REFERENCE
58        "Subclause 11.7.8.5 in IEEE Std 802.16-2004"
59    SYNTAX          INTEGER (0..255)
60
61
62 WmanIfMaxPkmFlowType ::= TEXTUAL-CONVENTION
63     STATUS          current
64     DESCRIPTION
65

```

```

1         "The object of this type specifies the maximum number of
2         concurrent PKM transactions that may be outstanding."
3     REFERENCE
4         "Subclause 11.7.8.6 in IEEE Std 802.16-2004"
5     SYNTAX      INTEGER (0..255)
6
7
8 WmanIfAuthPolicyType ::= TEXTUAL-CONVENTION
9     STATUS      current
10    DESCRIPTION
11        "The object of this type specifies authorization policy
12        that both SS and BS need to negotiate and synchronize.
13        A bit value of 0 = not supported, 1 = supported. If this
14        field is omitted, then both SS and BS shall use the IEEE
15        802.16 security, constituting X.509 digital certificates
16        and the RSA public key encryption algorithm, as
17        authorization policy."
18    REFERENCE
19        "Subclause 11.7.8.7 in IEEE Std 802.16-2004"
20    SYNTAX      BITS {ieee802Dot16PrivacySupported(0),
21                    reserved1(1),
22                    reserved2(2),
23                    reserved3(3),
24                    reserved4(4),
25                    reserved5(5),
26                    reserved6(6),
27                    reserved7(7)}
28
29
30 WmanIfMaxNumOfSaType ::= TEXTUAL-CONVENTION
31     STATUS      current
32     DESCRIPTION
33         "This field specifies maximum number of supported
34         security association of the SS."
35     REFERENCE
36         "Subclause 11.7.8.8 in IEEE Std 802.16-2004"
37     SYNTAX      INTEGER (0..255)
38
39
40 WmanIfIpVersionType ::= TEXTUAL-CONVENTION
41     STATUS      current
42     DESCRIPTION
43         "The object of this type indicates the version of IP used
44         on the Secondary Management Connection. The value should
45         be undefined if the 2nd management CID doesn't exist."
46     REFERENCE
47         "Subclause 11.7.4 in IEEE Std 802.16-2004"
48     SYNTAX      INTEGER {undefined(0),
49                    ipv4(1),
50                    ipv6(2)}
51
52
53 WmanIfMacCsBitMap ::= TEXTUAL-CONVENTION
54     STATUS      current
55     DESCRIPTION
56         "The object of this type indicates the set of MAC
57         convergence sublayer support. When a bit is set, it
58         indicates the corresponding CS feature is supported."
59
60
61

```

```

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

```

```

1          1 - only 11-bit FSN values are supported
2          All unspecified and reserved bits should be set to zero."
3
4      REFERENCE
5          "Subclause 11.8.2 in IEEE Std 802.16-2004"
6      SYNTAX      BITS {piggybackedRequests(0),
7                   fsnValuesSize(1)}
8
9
10     WmanIfSsTransitionGap ::= TEXTUAL-CONVENTION
11         STATUS      current
12         DESCRIPTION
13             "This field indicates the transition speed SSTTG and SSRTG
14             for TDD and H-FDD SSs. Allowed values are:
15                 OFDM mode: TDD and H-FDD 0..100
16                 Other modes: TDD: 0..50; H-FDD: 0..100"
17         REFERENCE
18             "Subclause 11.8.3.1 in IEEE Std 802.16-2004"
19         SYNTAX      INTEGER (0..100)
20
21
22
23     WmanIfMaxTxPowerType ::= TEXTUAL-CONVENTION
24         STATUS      current
25         DESCRIPTION
26             "This type is used to define maximum available power for
27             BPSK, QPSK, 16-QAM and 64-QAM constellations. The maximum
28             power parameters are reported in dBm and quantized in 0.5
29             dBm steps ranging from -64 dBm (encoded 0x00) to 63.5 dBm
30             (encoded 0xFF). Values outside this range shall be
31             assigned the closest extreme. SSs that do not support
32             QAM64 shall report the value of 0x00 in the maximum QAM64
33             power field."
34         REFERENCE
35             "Subclause 11.8.3.2 in IEEE Std 802.16-2004"
36         SYNTAX      INTEGER (0..255)
37
38
39
40
41     WmanIfOfdmFftSizes ::= TEXTUAL-CONVENTION
42         STATUS      current
43         DESCRIPTION
44             "This field indicates the FFT sizes supported by the SS/MS.
45             For each FFT size, a bit value of 0 indicates
46             'not supported' while 1 indicates 'supported'."
47         REFERENCE
48             "Subclause 11.8.3.7.1 in IEEE 802.16e-2005"
49         SYNTAX      BITS {fft256(0),
50                   fft2048(1),
51                   fft128(2),
52                   fft512(3),
53                   fft1024(4)}
54
55
56
57     WmanIfOfdmSsDeModType ::= TEXTUAL-CONVENTION
58         STATUS      current
59         DESCRIPTION
60             "This field indicates the different demodulator options
61             supported by a WirelessMAN-OFDM PHY SS for downlink. This
62             field is not used for other PHY specifications. A bit
63             value of 0 indicates 'not supported' while 1 indicates
64             'supported'."
65

```

```

1         'supported'."
2     REFERENCE
3         "Subclause 11.8.3.6.2 in IEEE Std 802.16-2004"
4     SYNTAX      BITS {qam64(0),
5                   btc(1),
6                   ctc(2),
7                   stc(3),
8                   aac(4)}
9
10
11
12 WmanIfOfdmSsModType ::= TEXTUAL-CONVENTION
13     STATUS      current
14     DESCRIPTION
15         "This field indicates the different modulator options
16         supported by a WirelessMAN-OFDM PHY SS for uplink. This
17         field is not used for other PHY specifications. A bit
18         value of 0 indicates 'not supported' while 1 indicates
19         'supported'."
20     REFERENCE
21         "Subclause 11.8.3.6.3 in IEEE Std 802.16-2004"
22     SYNTAX      BITS {qam64(0),
23                   btc(1),
24                   ctc(2),
25                   subchannellization(3),
26                   focusedCtBwReq(4)}
27
28
29
30
31 WmanIfOfdmFocusedCt ::= TEXTUAL-CONVENTION
32     STATUS      current
33     DESCRIPTION
34         "This field indicates whether the SS supports Focused
35         Contention (see 8.3.7.3.3). A bit value of 0 indicates
36         'not supported' while 1 indicates 'supported'."
37     REFERENCE
38         "Subclause 11.8.3.6.4 in IEEE Std 802.16-2004"
39     SYNTAX      BITS {focusedCtSupport(0)}
40
41
42
43 WmanIfOfdmTcSublayer ::= TEXTUAL-CONVENTION
44     STATUS      current
45     DESCRIPTION
46         "This field indicates whether or not the SS supports the
47         TC sublayer (see 8.3.4). A bit value of 0 indicates
48         'not supported' while 1 indicates 'supported'."
49     REFERENCE
50         "Subclause 11.8.3.6.5 in IEEE Std 802.16-2004"
51     SYNTAX      BITS {tcSublayerSupport(0)}
52
53
54
55 WmanIfBsIdType ::= TEXTUAL-CONVENTION
56     STATUS      current
57     DESCRIPTION
58         "Defines the encoding of BSID. The BSID is a 6 byte number
59         and follows the encoding rules of MacAddress textual
60         convention, i.e. as if it were transmitted
61         least-significant bit first. The value should be displayed
62         with 2 parts clearly separated by a colon e.g:
63         001DFF:00003A. The most significant part is representing
64
65

```

```

1         the Operator ID. "
2     SYNTAX      OCTET STRING (SIZE(6))
3
4
5 WmanIfIpv6FlowLabel ::= TEXTUAL-CONVENTION
6     STATUS      current
7     DESCRIPTION
8         "The value of this field specifies the matching values for
9         the IPv6 Flow label field. As the flow label field has a
10        length of 20 bits, the first 4 bits of the most
11        significant byte shall be set to 0x0 and disregarded."
12
13     SYNTAX      OCTET STRING (SIZE(3))
14
15 WmanIfOfdmaMsDeModType ::= TEXTUAL-CONVENTION
16     STATUS      current
17     DESCRIPTION
18         "This field indicates the different demodulator options
19         supported by a WirelessMAN-OFDMA PHY SS for downlink.
20         A bit value of 0 indicates 'not supported' while 1
21         indicates 'supported'."
22
23     REFERENCE
24         "Subclause 11.8.3.7.2 in IEEE 802.16e"
25
26     SYNTAX      BITS {qam64(0),
27                      btc(1),
28                      ctc(2),
29                      stc(3),
30                      aasDiversityMapScan(4),
31                      harqChase(5),
32                      harqCtcIr(6),
33                      reserved(7),
34                      harqCcIr(8),
35                      ldpc(9)}
36
37
38
39 WmanIfOfdmaMsModType ::= TEXTUAL-CONVENTION
40     STATUS      current
41     DESCRIPTION
42         "This field indicates the different modulator options
43         supported by a WirelessMAN-OFDMA PHY SS for uplink. A bit
44         value of 0 indicates 'not supported' while 1 indicates
45         'supported'."
46
47     REFERENCE
48         "Subclause 11.8.3.7.3 in IEEE 802.16e"
49
50     SYNTAX      BITS {qam64(0),
51                      btc(1),
52                      ctc(2),
53                      stc(3),
54                      harqChase(4),
55                      ctcIr(5),
56                      ccIr(6),
57                      ldpc(7)}
58
59
60
61 WmanIfOfdmaPermutation ::= TEXTUAL-CONVENTION
62     STATUS      current
63     DESCRIPTION
64         "This field indicates the OFDMA SS Permutation support
65

```

```

1           A bit value of 0 indicates 'not supported' while 1
2           indicates 'supported'."
3
4       REFERENCE
5           "Subclause 11.8.3.7.5 in IEEE 802.16e"
6       SYNTAX      BITS {optionalPuscSupport(0),
7                       optionalFuscSupport(1),
8                       amcOneBySixSupport(2),
9                       amcTwoByThreeSupport(3),
10                      amcThreeByTwoSupport(4),
11                      amcSupportWithHarqMap(5),
12                      tusclSupport(6),
13                      tusc2(7)}
14
15
16 WmanIfOfdmaMobility ::= TEXTUAL-CONVENTION
17     STATUS      current
18     DESCRIPTION
19         "This field indicates whether or not the MS supports
20         mobility hand-over, Sleepmode, and Idle-mode. A bit
21         value of 0 indicates 'not supported' while 1 indicates
22         it is supported."
23     REFERENCE
24         "Subclause 11.8.3.7.5 in IEEE 802.16e"
25     SYNTAX      BITS {handoverSupport(0),
26                     sleepModeSupport(1),
27                     idleModeSupport(2)}
28
29
30
31
32 --
33 -- BS object group - containing tables and objects to be implemented in
34 -- the Base station
35 --
36 --
37 -- wmanIfBsPacketCs contain the Base Station Packet Convergence
38 -- Sublayer objects
39 --
40
41 wmanIfBsPacketCs OBJECT IDENTIFIER ::= { wmanIfBsObjects 1 }
42
43 wmanIfBsProvisionedSfTable OBJECT-TYPE
44     SYNTAX      SEQUENCE OF WmanIfBsProvisionedSfEntry
45     MAX-ACCESS  not-accessible
46     STATUS      current
47     DESCRIPTION
48         "This table contains service flow profiles provisioned by
49         NMS. The service flow should be created with SS(s)
50         following instruction given by wmanIfBsSfState object.
51         1. The QoS parameters of the service flow are provisioned
52         in wmanIfBsServiceClassTable and referenced by
53         wmanIfBsServiceClassIndex.
54         2. The classifier rules of the service flow are provisioned
55         in wmanIfBsClassifierRuleTable, where they refer to SF
56         via wmanIfBsSfId.
57
58         The MAC addresses of SSs the service flow is created with
59         are provisioned in wmanIfBsSsProvisionedForSfTable, where
60         they refer to SF via wmanIfBsSfId."
61     REFERENCE
62
63
64
65

```

```

1         "Subclause 6.3.13 and 6.3.14 in IEEE Std 802.16-2004"
2         ::= { wmanIfBsPacketCs 1 }
3
4
5 wmanIfBsProvisionedSfEntry OBJECT-TYPE
6     SYNTAX      WmanIfBsProvisionedSfEntry
7     MAX-ACCESS  not-accessible
8     STATUS      current
9     DESCRIPTION
10        "This table provides one row for each service flow
11         provisioned by NMS. The table is indexed by ifIndex and
12         wmanIfBsSfId. ifIndex is associated with the BS sector."
13     INDEX { ifIndex, wmanIfBsSfId }
14     ::= { wmanIfBsProvisionedSfTable 1 }
15
16
17 WmanIfBsProvisionedSfEntry ::= SEQUENCE {
18     wmanIfBsSfId                Unsigned32,
19     wmanIfBsSfDirection         INTEGER,
20     wmanIfBsServiceClassIndex   INTEGER,
21     wmanIfBsSfState             WmanIfSfState,
22     wmanIfBsSfProvisionedTime   TimeStamp,
23     wmanIfBsSfCsSpecification   WmanIfCsSpecification,
24     wmanIfBsProvisionedSfRowStatus RowStatus}
25
26
27
28 wmanIfBsSfId OBJECT-TYPE
29     SYNTAX      Unsigned32 (1 .. 4294967295)
30     MAX-ACCESS  not-accessible
31     STATUS      current
32     DESCRIPTION
33        "A 32 bit quantity that uniquely identifies a service flow
34         to both the subscriber station and base station (BS)."
35     ::= { wmanIfBsProvisionedSfEntry 1 }
36
37
38
39 wmanIfBsSfDirection OBJECT-TYPE
40     SYNTAX      INTEGER {downstream(1),
41                          upstream(2)}
42     MAX-ACCESS  read-create
43     STATUS      current
44     DESCRIPTION
45        "An attribute indicating the service flow is downstream or
46         upstream."
47     ::= { wmanIfBsProvisionedSfEntry 2 }
48
49
50
51 wmanIfBsServiceClassIndex OBJECT-TYPE
52     SYNTAX      INTEGER (1..65535)
53     MAX-ACCESS  read-create
54     STATUS      current
55     DESCRIPTION
56        "The index in wmanIfBsServiceClassTable describing the
57         service class or QoS parameters for such service flow.
58         If no associated entry in wmanIfBsServiceClassTable
59         exists, this object returns a value of zero."
60     ::= { wmanIfBsProvisionedSfEntry 3 }
61
62
63
64 wmanIfBsSfState OBJECT-TYPE
65

```

```

1      SYNTAX      WmanIfSfState
2      MAX-ACCESS  read-create
3      STATUS      current
4      DESCRIPTION
5          "wmanIfBsSfState determines the requested state of a service
6          flow.
7          - authorized state: A service flow is provisioned but
8            not resource is reserved yet
9          - admitted state: service flow has resources reserved.
10         - active state: has resources committed by the BS (e.g., is
11           actively sending maps containing unsolicited grants for a
12           UGS-based service flow),"
13
14      REFERENCE
15          "Subclause 6.3.14.6, in IEEE Std 802.16-2004"
16      ::= { wmanIfBsProvisionedSfEntry 4 }
17
18  wmanIfBsSfProvisionedTime OBJECT-TYPE
19      SYNTAX      TimeStamp
20      MAX-ACCESS  read-create
21      STATUS      current
22      DESCRIPTION
23          "Indicates the date and time when the service flow is
24          provisioned."
25      ::= { wmanIfBsProvisionedSfEntry 5 }
26
27  wmanIfBsSfCsSpecification OBJECT-TYPE
28      SYNTAX      WmanIfCsSpecification
29      MAX-ACCESS  read-create
30      STATUS      current
31      DESCRIPTION
32          "This parameter specifies the convergence sublayer
33          encapsulation mode."
34      REFERENCE
35          "Subclause 11.13.19.1 in IEEE Std 802.16-2004"
36      ::= { wmanIfBsProvisionedSfEntry 6 }
37
38  wmanIfBsProvisionedSfRowStatus OBJECT-TYPE
39      SYNTAX      RowStatus
40      MAX-ACCESS  read-create
41      STATUS      current
42      DESCRIPTION
43          "This object is used to create a new row or modify or
44          delete an existing row in this table.
45
46          If the implementator of this MIB has choosen not
47          to implement 'dynamic assignment' of profiles, this
48          object is not useful and should return noSuchName
49          upon SNMP request."
50      ::= { wmanIfBsProvisionedSfEntry 7 }
51
52  wmanIfBsSsProvisionedForSfTable OBJECT-TYPE
53      SYNTAX      SEQUENCE OF WmanIfBsSsProvisionedForSfEntry
54      MAX-ACCESS  not-accessible
55      STATUS      current

```

```

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

```

```

1      STATUS      current
2      DESCRIPTION
3          "This table is provisioned and is indexed by
4          wmanIfBsQoSProfileIndex. Each entry of the table contains
5          corresponding service flow characteristic attributes
6          (e.g. QoS parameter set). The value of
7          wmanIfBsQoSProfileIndex is obtained from
8          wmanIfBsServiceClassIndex in wmanIfBsProvisionedSfTable"
9
10     REFERENCE
11         "Subclause 6.3.14.4 in IEEE Std 802.16-2004"
12     ::= { wmanIfBsPacketCs 3 }
13
14
15     wmanIfBsServiceClassEntry OBJECT-TYPE
16         SYNTAX      WmanIfBsServiceClassEntry
17         MAX-ACCESS   not-accessible
18         STATUS      current
19         DESCRIPTION
20             "This table provides one row for each service class"
21         INDEX { ifIndex, wmanIfBsQoSProfileIndex }
22     ::= { wmanIfBsServiceClassTable 1 }
23
24
25
26     WmanIfBsServiceClassEntry ::= SEQUENCE {
27         wmanIfBsQoSProfileIndex      INTEGER,
28         wmanIfBsQoSServiceClassName  WmanIfServClassName,
29         wmanIfBsQoSSTrafficPriority   INTEGER,
30         wmanIfBsQoSMaxSustainedRate   Unsigned32,
31         wmanIfBsQoSMaxTrafficBurst    Unsigned32,
32         wmanIfBsQoSMinReservedRate    Unsigned32,
33         wmanIfBsQOSToleratedJitter    Unsigned32,
34         wmanIfBsQoSMaxLatency         Unsigned32,
35         wmanIfBsQoSFixedVsVariableSduInd  INTEGER,
36         wmanIfBsQOSSduSize            Unsigned32,
37         wmanIfBsQoSScSchedulingType    WmanIfSfSchedulingType,
38         wmanIfBsQoSScArqEnable         TruthValue,
39         wmanIfBsQoSScArqWindowSize     INTEGER,
40         wmanIfBsQoSScArqBlockLifetime  INTEGER,
41         wmanIfBsQoSScArqSyncLossTimeout  INTEGER,
42         wmanIfBsQoSScArqDeliverInOrder  TruthValue,
43         wmanIfBsQoSScArqRxPurgeTimeout  INTEGER,
44         wmanIfBsQoSScArqBlockSize      INTEGER,
45         wmanIfBsQoSSCMinRsVdTolerableRate  Unsigned32,
46         wmanIfBsQoSReqTxPolicy         BITS,
47         wmanIfBsQOSServiceClassRowStatus  RowStatus }
48
49
50
51     wmanIfBsQoSProfileIndex OBJECT-TYPE
52         SYNTAX      INTEGER (1 .. 65535)
53         MAX-ACCESS   not-accessible
54         STATUS      current
55         DESCRIPTION
56             "The index value which uniquely identifies an entry
57             in the wmanIfBsServiceClassTable"
58     ::= { wmanIfBsServiceClassEntry 1 }
59
60
61
62     wmanIfBsQoSServiceClassName OBJECT-TYPE
63
64
65

```

```

1      SYNTAX      WmanIfServClassName
2      MAX-ACCESS  read-create
3      STATUS      current
4      DESCRIPTION
5          "Refers to the Service Class Name"
6      REFERENCE
7          "Subclause 11.13.3 in IEEE Std 802.16-2004"
8      ::= { wmanIfBsServiceClassEntry 2 }
9
10
11
12  wmanIfBsQoStraffPriority OBJECT-TYPE
13      SYNTAX      INTEGER (0..7)
14      MAX-ACCESS  read-create
15      STATUS      current
16      DESCRIPTION
17          "The value of this parameter specifies the priority
18           assigned to a service flow. For uplink service flows,
19           the BS should use this parameter when determining
20           precedence in request service and grant generation,
21           and the SS shall preferentially select contention
22           Request opportunities for Priority Request CIDs
23           based on this priority. Higher numbers indicate higher
24           priority"
25      REFERENCE
26          "Subclause 11.13.5 in IEEE Std 802.16-2004"
27      ::= { wmanIfBsServiceClassEntry 3 }
28
29
30
31
32  wmanIfBsQoSMaxSustainedRate OBJECT-TYPE
33      SYNTAX      Unsigned32
34      UNITS       "b/s"
35      MAX-ACCESS  read-create
36      STATUS      current
37      DESCRIPTION
38          "This parameter defines the peak information rate
39           of the service. The rate is expressed in bits per
40           second and pertains to the SDUs at the input to
41           the system."
42      REFERENCE
43          "Subclause 11.13.6 in IEEE Std 802.16-2004"
44      ::= { wmanIfBsServiceClassEntry 4 }
45
46
47
48
49  wmanIfBsQoSMaxTrafficBurst OBJECT-TYPE
50      SYNTAX      Unsigned32
51      UNITS       "byte"
52      MAX-ACCESS  read-create
53      STATUS      current
54      DESCRIPTION
55          "This parameter defines the maximum burst size that
56           must be accommodated for the service."
57      REFERENCE
58          "Subclause 11.13.7 in IEEE Std 802.16-2004"
59      ::= { wmanIfBsServiceClassEntry 5 }
60
61
62
63  wmanIfBsQoSMinReservedRate OBJECT-TYPE
64      SYNTAX      Unsigned32
65

```

```

1      UNITS      "b/s"
2      MAX-ACCESS  read-create
3      STATUS      current
4      DESCRIPTION
5          "This parameter specifies the minimum rate reserved
6          for this service flow."
7      REFERENCE
8          "Subclause 11.13.8 in IEEE Std 802.16-2004"
9      ::= { wmanIfBsServiceClassEntry 6 }
10
11
12
13  wmanIfBsQoS toleratedJitter OBJECT-TYPE
14      SYNTAX      Unsigned32
15      UNITS        "millisecond"
16      MAX-ACCESS  read-create
17      STATUS      current
18      DESCRIPTION
19          "This parameter defines the Maximum delay
20          variation (jitter) for the connection."
21      REFERENCE
22          "Subclause 11.13.13 in IEEE Std 802.16-2004"
23      ::= { wmanIfBsServiceClassEntry 7 }
24
25
26
27  wmanIfBsQoSMaxLatency OBJECT-TYPE
28      SYNTAX      Unsigned32
29      UNITS        "millisecond"
30      MAX-ACCESS  read-create
31      STATUS      current
32      DESCRIPTION
33          "The value of this parameter specifies the maximum
34          latency between the reception of a packet by the BS
35          or SS on its network interface and the forwarding
36          of the packet to its RF Interface."
37      REFERENCE
38          "Subclause 11.13.14 in IEEE Std 802.16-2004"
39      ::= { wmanIfBsServiceClassEntry 8 }
40
41
42
43
44  wmanIfBsQoSFixedVsVariableSduInd OBJECT-TYPE
45      SYNTAX      INTEGER {variableLength(0),
46                          fixedLength(1)}
47      MAX-ACCESS  read-create
48      STATUS      current
49      DESCRIPTION
50          "The value of this parameter specifies whether the SDUs
51          on the service flow are variable-length (0) or
52          fixed-length (1). The parameter is used only if
53          packing is on for the service flow. The default value
54          is 0, i.e., variable-length SDUs."
55      REFERENCE
56          "Subclause 11.13.15 in IEEE Std 802.16-2004"
57      DEFVAL      { variableLength }
58      ::= { wmanIfBsServiceClassEntry 9 }
59
60
61
62
63  wmanIfBsQoS SduSize OBJECT-TYPE
64      SYNTAX      Unsigned32
65

```

```

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

```

```

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

```

```

1      DEFVAL      {0}
2      ::= { wmanIfBsServiceClassEntry 17 }
3
4
5  wmanIfBsQosScArqBlockSize OBJECT-TYPE
6      SYNTAX      INTEGER (1..2040)
7      UNITS       "byte"
8      MAX-ACCESS  read-create
9      STATUS      current
10     DESCRIPTION
11         "The value of this parameter specifies the size of an
12         ARQ block. This parameter shall be established by
13         negotiation during the connection creation dialog."
14     REFERENCE
15         "Subclause 11.13.18.8 in IEEE Std 802.16-2004"
16     ::= { wmanIfBsServiceClassEntry 18 }
17
18
19
20  wmanIfBsQosSCMinRsvdTolerableRate OBJECT-TYPE
21      SYNTAX      Unsigned32
22      UNITS       "b/s"
23      MAX-ACCESS  read-create
24      STATUS      current
25      DESCRIPTION
26         "Minimum Tolerable Traffic Rate = R (bits/sec) with
27         time base T(sec) means the following. Let S denote
28         additional demand accumulated at the MAC SAP of the
29         transmitter during an arbitrary time interval of the
30         length T. Then the amount of data forwarded at the
31         receiver to CS (in bits) during this interval should
32         be not less than min {S, R * T}."
33     REFERENCE
34         "Subclause 11.13.9 in IEEE Std 802.16-2004"
35     ::= { wmanIfBsServiceClassEntry 19 }
36
37
38
39
40
41  wmanIfBsQoSReqTxPolicy OBJECT-TYPE
42      SYNTAX      BITS {noBroadcastBwReq(0),
43                        reserved1(1),
44                        noPiggybackReq(2),
45                        noFragmentData(3),
46                        noPHS(4),
47                        noSduPacking(5),
48                        noCrc(6),
49                        reserved2(7)}
50      MAX-ACCESS  read-create
51      STATUS      current
52      DESCRIPTION
53         "The value of this parameter provides the capability to
54         specify certain attributes for the associated service
55         flow. An attribute is enabled by setting the
56         corresponding bit position to 1."
57     REFERENCE
58         "Subclause 11.13.12 in IEEE Std 802.16-2004"
59     ::= { wmanIfBsServiceClassEntry 20 }
60
61
62
63  wmanIfBsQoSServiceClassRowStatus OBJECT-TYPE
64      SYNTAX      RowStatus
65

```

```

1      MAX-ACCESS    read-create
2      STATUS        current
3      DESCRIPTION
4          "This object is used to create a new row or modify or
5          delete an existing row in this table.
6
7          If the implementator of this MIB has choosen not
8          to implement 'dynamic assignment' of profiles, this
9          object is not useful and should return noSuchName
10         upon SNMP request."
11
12     ::= { wmanIfBsServiceClassEntry 21 }
13
14
15 wmanIfBsClassifierRuleTable OBJECT-TYPE
16     SYNTAX          SEQUENCE OF WmanIfBsClassifierRuleEntry
17     MAX-ACCESS      not-accessible
18     STATUS          current
19     DESCRIPTION
20         "This table contains packet classifier rules associated
21         with service flows."
22     REFERENCE
23         "Subclause 11.13.19.3.4 in IEEE Std 802.16-2004"
24     ::= { wmanIfBsPacketCs 4 }
25
26
27
28
29 wmanIfBsClassifierRuleEntry OBJECT-TYPE
30     SYNTAX          WmanIfBsClassifierRuleEntry
31     MAX-ACCESS      not-accessible
32     STATUS          current
33     DESCRIPTION
34         "This table provides one row for each packet classifier
35         rule, and is indexed by ifIndex, wmanIfBsSfId, and
36         wmanIfBsClassifierRuleIndex. IfIdex is associated with
37         the BS sector. wmanIfBsSfId identifies the service flow,
38         while wmanIfBsClassifierRuleIndex identifies the packet
39         classifier rule."
40     INDEX { ifIndex, wmanIfBsSfId, wmanIfBsClassifierRuleIndex }
41     ::= { wmanIfBsClassifierRuleTable 1 }
42
43
44
45 WmanIfBsClassifierRuleEntry ::= SEQUENCE {
46     wmanIfBsClassifierRuleIndex      Unsigned32,
47     wmanIfBsClassifierRulePriority    INTEGER,
48     wmanIfBsClassifierRuleIpTosLow   INTEGER,
49     wmanIfBsClassifierRuleIpTosHigh  INTEGER,
50     wmanIfBsClassifierRuleIpTosMask  INTEGER,
51     wmanIfBsClassifierRuleIpProtocol Integer32,
52     wmanIfBsClassifierRuleIpSourceAddr InetAddress,
53     wmanIfBsClassifierRuleIpSourceMask InetAddress,
54     wmanIfBsClassifierRuleIpDestAddr InetAddress,
55     wmanIfBsClassifierRuleIpDestMask InetAddress,
56     wmanIfBsClassifierRuleSourcePortStart Integer32,
57     wmanIfBsClassifierRuleSourcePortEnd Integer32,
58     wmanIfBsClassifierRuleDestPortStart Integer32,
59     wmanIfBsClassifierRuleDestPortEnd Integer32,
60     wmanIfBsClassifierRuleDestMacAddr MacAddress,
61     wmanIfBsClassifierRuleDestMacMask MacAddress,
62
63
64
65

```

```

1      wmanIfBsClassifierRuleSourceMacAddr      MacAddress,
2      wmanIfBsClassifierRuleSourceMacMask      MacAddress,
3      wmanIfBsClassifierRuleEnetProtocolType    INTEGER,
4      wmanIfBsClassifierRuleEnetProtocol        Integer32,
5      wmanIfBsClassifierRuleUserPriLow          Integer32,
6      wmanIfBsClassifierRuleUserPriHigh         Integer32,
7      wmanIfBsClassifierRuleVlanId              Integer32,
8      wmanIfBsClassifierRuleState               INTEGER,
9      wmanIfBsClassifierRulePhsSize              Integer32,
10     wmanIfBsClassifierRulePhsMask              OCTET STRING,
11     wmanIfBsClassifierRulePhsVerify            WmanIfPhsRuleVerify,
12     wmanIfBsClassifierRuleIpv6FlowLabel        WmanIfIpv6FlowLabel,
13     wmanIfBsClassifierRuleBitMap               WmanIfClassifierBitMap,
14     wmanIfBsClassifierRuleRowStatus            RowStatus}
15
16
17
18
19     wmanIfBsClassifierRuleIndex OBJECT-TYPE
20         SYNTAX      Unsigned32 (1..4294967295)
21         MAX-ACCESS  not-accessible
22         STATUS      current
23         DESCRIPTION
24             "An index is assigned to a classifier in BS classifiers
25              table"
26         ::= { wmanIfBsClassifierRuleEntry 1 }
27
28
29
30     wmanIfBsClassifierRulePriority OBJECT-TYPE
31         SYNTAX      INTEGER (0..255)
32         MAX-ACCESS  read-create
33         STATUS      current
34         DESCRIPTION
35             "The value specifies the priority for the Classifier, which
36              is used for determining the order of the Classifier. A
37              higher value indicates higher priority. Classifiers may
38              have priorities in the range 0..255."
39         REFERENCE
40             "Subclause 11.13.19.3.4.1 in IEEE Std 802.16-2004"
41         DEFVAL      { 0 }
42         ::= { wmanIfBsClassifierRuleEntry 2 }
43
44
45
46
47     wmanIfBsClassifierRuleIpTosLow OBJECT-TYPE
48         SYNTAX      INTEGER (0..255)
49         MAX-ACCESS  read-create
50         STATUS      current
51         DESCRIPTION
52             "The low value of a range of TOS byte values. If the
53              referenced parameter is not present in a classifier, this
54              object reports the value of 0."
55         REFERENCE
56             "Subclause 11.13.19.3.4.2 in IEEE Std 802.16-2004"
57         ::= { wmanIfBsClassifierRuleEntry 3 }
58
59
60
61     wmanIfBsClassifierRuleIpTosHigh OBJECT-TYPE
62         SYNTAX      INTEGER (0..255)
63         MAX-ACCESS  read-create
64         STATUS      current
65

```

```

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

```

```

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

```

```

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

```

```

1         "An Ethernet packet matches an entry when its destination
2         MAC address bitwise ANDed with
3         wmanIfBsClassifierRuleDestMacMask equals the value of
4         wmanIfBsClassifierRuleDestMacAddr. If the referenced
5         parameter is not present in a classifier, this object
6         reports the value of '000000000000'H."
7
8     REFERENCE
9         "Subclause 11.13.19.3.4.8 in IEEE Std 802.16-2004"
10    ::= { wmanIfBsClassifierRuleEntry 15 }
11
12
13 wmanIfBsClassifierRuleDestMacMask OBJECT-TYPE
14     SYNTAX      MacAddress
15     MAX-ACCESS  read-create
16     STATUS      current
17     DESCRIPTION
18         "An Ethernet packet matches an entry when its destination
19         MAC address bitwise ANDed with
20         wmanIfBsClassifierRuleDestMacMask equals the value of
21         wmanIfBsClassifierRuleDestMacAddr. If the referenced
22         parameter is not present in a classifier, this object
23         reports the value of '000000000000'H."
24
25     REFERENCE
26         "Subclause 11.13.19.3.4.8 in IEEE Std 802.16-2004"
27    ::= { wmanIfBsClassifierRuleEntry 16 }
28
29
30
31 wmanIfBsClassifierRuleSourceMacAddr OBJECT-TYPE
32     SYNTAX      MacAddress
33     MAX-ACCESS  read-create
34     STATUS      current
35     DESCRIPTION
36         "An Ethernet packet matches this entry when its source
37         MAC address bitwise ANDed with
38         wmanIfBsClassifierRuleSourceMacMask equals the value
39         of wmanIfBsClassifierRuleSourceMacAddr. If the
40         referenced parameter is not present in a classifier,
41         this object reports the value of '000000000000'H."
42
43     REFERENCE
44         "Subclause 11.13.19.3.4.9 in IEEE Std 802.16-2004"
45    ::= { wmanIfBsClassifierRuleEntry 17 }
46
47
48
49 wmanIfBsClassifierRuleSourceMacMask OBJECT-TYPE
50     SYNTAX      MacAddress
51     MAX-ACCESS  read-create
52     STATUS      current
53     DESCRIPTION
54         "An Ethernet packet matches an entry when its source
55         MAC address bitwise ANDed with
56         wmanIfBsClassifierRuleSourceMacMask equals the value of
57         wmanIfBsClassifierRuleSourceMacAddr. If the referenced
58         parameter is not present in a classifier, this object
59         reports the value of '000000000000'H."
60
61     REFERENCE
62         "Subclause 11.13.19.3.4.9 in IEEE Std 802.16-2004"
63    ::= { wmanIfBsClassifierRuleEntry 18 }
64
65

```

1
2 wmanIfBsClassifierRuleEnetProtocolType OBJECT-TYPE

3 SYNTAX INTEGER {none(0),
4 ethertype(1),
5 dsap(2)}

6
7 MAX-ACCESS read-create

8 STATUS current

9 DESCRIPTION

10 "This object indicates the format of the layer 3 protocol
11 id in the Ethernet packet. A value of none(0) means that
12 the rule does not use the layer 3 protocol type as a
13 matching criteria. A value of ethertype(1) means that the
14 rule applies only to frames which contains an EtherType
15 value. Ethertype values are contained in packets using
16 the Dec-Intel-Xerox (DIX) encapsulation or the RFC1042
17 Sub-Network Access Protocol (SNAP) encapsulation formats.
18 A value of dsap(2) means that the rule applies only to
19 frames using the IEEE802.3 encapsulation format with a
20 Destination Service Access Point (DSAP) other than 0xAA
21 (which is reserved for SNAP). If the Ethernet frame
22 contains an 802.1P/Q Tag header (i.e. EtherType 0x8100),
23 this object applies to the embedded EtherType field within
24 the 802.1P/Q header. If the referenced parameter is not
25 present in a classifier, this object reports the value of
26 0."
27
28
29
30

31 REFERENCE

32 "Subclause 11.13.19.3.4.10 in IEEE Std 802.16-2004"
33 ::= { wmanIfBsClassifierRuleEntry 19 }
34

35
36 wmanIfBsClassifierRuleEnetProtocol OBJECT-TYPE

37 SYNTAX Integer32 (0..65535)

38 MAX-ACCESS read-create

39 STATUS current

40 DESCRIPTION

41 "If wmanIfBsClassifierRuleEnetProtocolType is none(0),
42 this object is ignored when considering whether a packet
43 matches the current rule.
44 If wmanIfBsClassifierRuleEnetProtocolType is ethertype(1),
45 this object gives the 16-bit value of the EtherType that
46 the packet must match in order to match the rule.
47 If wmanIfBsClassifierRuleEnetProtocolType is dsap(2), the
48 lower 8 bits of this object's value must match the DSAP
49 byte of the packet in order to match the rule.
50 If the Ethernet frame contains an 802.1P/Q Tag header
51 (i.e. EtherType 0x8100), this object applies to the
52 embedded EtherType field within the 802.1P/Q header.
53 If the referenced parameter is not present in the
54 classifier, the value of this object is reported as 0."
55
56
57
58

59 REFERENCE

60 "Subclause 11.13.19.3.4.10 in IEEE Std 802.16-2004"
61 ::= { wmanIfBsClassifierRuleEntry 20 }
62

63 wmanIfBsClassifierRuleUserPriLow OBJECT-TYPE

64 SYNTAX Integer32 (0..7)
65

```

1      MAX-ACCESS    read-create
2      STATUS        current
3      DESCRIPTION
4          "This object applies only to Ethernet frames using the
5              802.1P/Q tag header (indicated with EtherType 0x8100).
6              Such frames include a 16-bit Tag that contains a 3 bit
7              Priority field and a 12 bit VLAN number.
8              Tagged Ethernet packets must have a 3-bit Priority field
9              within the range of wmanIfBsClassifierRuleUserPriLow and
10             wmanIfBsClassifierRuleUserPriHigh in order to match this
11             rule.
12             If the referenced parameter is not present in the
13             classifier, the value of this object is reported as 0."
14      REFERENCE
15          "Subclause 11.13.19.3.4.11 in IEEE Std 802.16-2004"
16      ::= { wmanIfBsClassifierRuleEntry 21 }
17
21  wmanIfBsClassifierRuleUserPriHigh OBJECT-TYPE
22      SYNTAX          Integer32 (0..7)
23      MAX-ACCESS      read-create
24      STATUS          current
25      DESCRIPTION
26          "This object applies only to Ethernet frames using the
27              802.1P/Q tag header (indicated with EtherType 0x8100).
28              Such frames include a 16-bit Tag that contains a 3 bit
29              Priority field and a 12 bit VLAN number.
30              Tagged Ethernet packets must have a 3-bit Priority
31              field within the range of wmanIfBsClassifierRuleUserPriLow
32              and wmanIfBsClassifierRuleUserPriHigh in order to match
33              this rule.
34              If the referenced parameter is not present in the
35              classifier, the value of this object is reported as 7."
36      REFERENCE
37          "Subclause 11.13.19.3.4.11 in IEEE Std 802.16-2004"
38      ::= { wmanIfBsClassifierRuleEntry 22 }
39
43  wmanIfBsClassifierRuleVlanId OBJECT-TYPE
44      SYNTAX          Integer32 (0..4095)
45      MAX-ACCESS      read-create
46      STATUS          current
47      DESCRIPTION
48          "This object applies only to Ethernet frames using the
49              802.1P/Q tag header.
50              If this object's value is nonzero, tagged packets must
51              have a VLAN Identifier that matches the value in order
52              to match the rule.
53              Only the least significant 12 bits of this object's
54              value are valid.
55              If the referenced parameter is not present in the
56              classifier, the value of this object is reported as 0."
57      REFERENCE
58          "Subclause 11.13.19.3.4.12 in IEEE Std 802.16-2004"
59      ::= { wmanIfBsClassifierRuleEntry 23 }
60
61
62
63
64
65

```

```

1  wmanIfBsClassifierRuleState OBJECT-TYPE
2      SYNTAX      INTEGER {active(1),
3                      inactive(2)}
4
5      MAX-ACCESS   read-create
6      STATUS      current
7      DESCRIPTION
8          "This object indicates whether or not the classifier is
9           enabled to classify packets to a Service Flow.
10          If the referenced parameter is not present in the
11          classifier, the value of this object is reported
12          as active(1)."
```

::= { wmanIfBsClassifierRuleEntry 24 }

```

16 wmanIfBsClassifierRulePhsSize OBJECT-TYPE
17     SYNTAX      Integer32
18     UNITS       "byte"
19     MAX-ACCESS   read-create
20     STATUS      current
21     DESCRIPTION
22         "This object is used to configure the PHS rule for this
23          classifier. The value of this field - PHSS is the total
24          number of bytes in the header to be suppressed and then
25          restored in a service flow that uses PHS. If the value of
26          this field is 0 bytes then PHS is disabled for this
27          classifier. If flag phsMask in wmanIfBsClassifierRuleBitMap
28          is set to 0 and flag phsSize in
29          wmanIfBsClassifierRuleBitMap is set to 0, then BS can still
30          create PHS rules using its own custom mask (i.e. the rule
31          is not configured by NMS)."
```

REFERENCE

"Subclause 11.13.19.3.7.4 in IEEE Std 802.16-2004"

```

38     DEFVAL      {0}
39     ::= { wmanIfBsClassifierRuleEntry 25 }
```

```

42 wmanIfBsClassifierRulePhsMask OBJECT-TYPE
43     SYNTAX      OCTET STRING (SIZE(0..65535))
44     MAX-ACCESS   read-create
45     STATUS      current
46     DESCRIPTION
47         "This object is used to configure the PHS rule for this
48          classifier. It is encoded as follows:
49          bit 0:
50              0 = don't suppress the 1st byte of the suppression field
51              1 = suppress first byte of the suppression field
52          bit 1:
53              0 = don't suppress the 2nd byte of the suppression field
54              1 = suppress second byte of the suppression field
55          bit x:
56              0 = don't suppress the (x+1) byte of the suppression
57                  field
58              1 = suppress (x+1) byte of the suppression field
59          where the length of the octet string is ceiling
60          (wmanIfBsClassifierRulePhsSize/8). BS should use this value
61          to create a new PHS rule index (PHSI) and field (PHSF) as
```

defined in the standard. If flag phsMask in wmanIfBsClassifierRuleBitMap is set to 0 and flag phsSize in wmanIfBsClassifierRuleBitMap 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)."

REFERENCE

"Subclause 11.13.19.3.7.3 in IEEE Std 802.16-2004"
 ::= { wmanIfBsClassifierRuleEntry 26 }

wmanIfBsClassifierRulePhsVerify OBJECT-TYPE

SYNTAX WmanIfPhsRuleVerify
 MAX-ACCESS read-create
 STATUS current
 DESCRIPTION
 "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."
 DEFVAL { phsVerifyEnable }
 ::= { wmanIfBsClassifierRuleEntry 27 }

wmanIfBsClassifierRuleIpv6FlowLabel OBJECT-TYPE

SYNTAX WmanIfIpv6FlowLabel
 MAX-ACCESS read-create
 STATUS current
 DESCRIPTION
 "The value of this field specifies the matching values for the IPv6 Flow label field."
 ::= { wmanIfBsClassifierRuleEntry 28 }

wmanIfBsClassifierRuleBitMap OBJECT-TYPE

SYNTAX WmanIfClassifierBitMap
 MAX-ACCESS read-create
 STATUS current
 DESCRIPTION
 "This object indicates which parameter encodings were actually present in the entry. A bit set to '1' indicates the corresponding classifier encoding is present, and '0' means otherwise"
 ::= { wmanIfBsClassifierRuleEntry 29 }

wmanIfBsClassifierRuleRowStatus OBJECT-TYPE

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

 If the implementator of this MIB has choosen not to implement 'dynamic assignment' of profiles, this object is not useful and should return noSuchName upon SNMP request."
 ::= { wmanIfBsClassifierRuleEntry 30 }

```

1  wmanIfBsSsPacketCounterTable OBJECT-TYPE
2      SYNTAX          SEQUENCE OF WmanIfBsSsPacketCounterEntry
3      MAX-ACCESS      not-accessible
4      STATUS          current
5      DESCRIPTION
6          "This table contains counters to keep track of the number
7          of packets and octets that have been received or
8          transmitted on the per service flow basis."
9      ::= { wmanIfBsPacketCs 5 }
10
11
12
13  wmanIfBsSsPacketCounterEntry OBJECT-TYPE
14      SYNTAX          WmanIfBsSsPacketCounterEntry
15      MAX-ACCESS      not-accessible
16      STATUS          current
17      DESCRIPTION
18          "This table provides one row for each service flow, and
19          is indexed by ifIndex, wmanIfCmnCpsSfMacAddress, and
20          wmanIfCmnCpsSfId."
21      INDEX { ifIndex, wmanIfCmnCpsSfMacAddress,
22              wmanIfCmnCpsSfId }
23      ::= { wmanIfBsSsPacketCounterTable 1 }
24
25
26
27  WmanIfBsSsPacketCounterEntry ::= SEQUENCE {
28      wmanIfBsSsMacSduCount          Counter64,
29      wmanIfBsSsOctetCount           Counter64,
30      wmanIfBsSsResetCounter         INTEGER,
31      wmanIfBsSsResetCounterTime     TimeStamp}
32
33
34
35  wmanIfBsSsMacSduCount OBJECT-TYPE
36      SYNTAX          Counter64
37      MAX-ACCESS      read-only
38      STATUS          current
39      DESCRIPTION
40          "This object counts the number of MAC SDUs that have
41          been transmitted or received."
42      ::= { wmanIfBsSsPacketCounterEntry 1 }
43
44
45
46  wmanIfBsSsOctetCount OBJECT-TYPE
47      SYNTAX          Counter64
48      MAX-ACCESS      read-only
49      STATUS          current
50      DESCRIPTION
51          "This object counts the number of octets of MAC SDUs
52          that have been transmitted or received."
53      ::= { wmanIfBsSsPacketCounterEntry 2 }
54
55
56
57  wmanIfBsSsResetCounter OBJECT-TYPE
58      SYNTAX          INTEGER {null(0),
59                          resetCounter(1)}
60      MAX-ACCESS      read-write
61      STATUS          current
62      DESCRIPTION
63          "When this attribute is SET to resetCounter(1), the
64          corresponding entry of packet counters will be reset."
65

```

```

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

```

```

1      wmanIfBsSs2ndMgmtArqEnable      TruthValue,
2      wmanIfBsSs2ndMgmtArqWindowSize  INTEGER,
3      wmanIfBsSs2ndMgmtArqDnLinkTxDelay  INTEGER,
4      wmanIfBsSs2ndMgmtArqUpLinkTxDelay  INTEGER,
5      wmanIfBsSs2ndMgmtArqDnLinkRxDelay  INTEGER,
6      wmanIfBsSs2ndMgmtArqUpLinkRxDelay  INTEGER,
7      wmanIfBsSs2ndMgmtArqBlockLifetime  INTEGER,
8      wmanIfBsSs2ndMgmtArqSyncLossTimeout  INTEGER,
9      wmanIfBsSs2ndMgmtArqDeliverInOrder  TruthValue,
10     wmanIfBsSs2ndMgmtArqRxPurgeTimeout  INTEGER,
11     wmanIfBsSs2ndMgmtArqBlockSize       INTEGER,
12     wmanIfBsSsVendorIdEncoding          OCTET STRING,
13     wmanIfBsSsAasBroadcastPermission    INTEGER,
14     wmanIfBsSsMaxTxPowerBpsk            WmanIfMaxTxPowerType,
15     wmanIfBsSsMaxTxPowerQpsk            WmanIfMaxTxPowerType,
16     wmanIfBsSsMaxTxPower16Qam           WmanIfMaxTxPowerType,
17     wmanIfBsSsMaxTxPower64Qam           WmanIfMaxTxPowerType,
18     wmanIfBsSsMacVersion                WmanIfMacVersion}
19
20
21
22
23
24     wmanIfBsSsMacAddress OBJECT-TYPE
25         SYNTAX      MacAddress
26         MAX-ACCESS  not-accessible
27         STATUS      current
28         DESCRIPTION
29             "The MAC address of SS is received from the RNG-REQ
30             message. When SS registers, this MAC address is entered
31             into the table, and used as the identifier to the SS."
32         REFERENCE
33             "Subclause 6.3.2.3.5 in IEEE Std 802.16-2004"
34             ::= { wmanIfBsRegisteredSsEntry 1 }
35
36
37
38     wmanIfBsSsBasicCid OBJECT-TYPE
39         SYNTAX      WmanIfCidType
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             ::= { wmanIfBsRegisteredSsEntry 2 }
48
49
50
51     wmanIfBsSsPrimaryCid OBJECT-TYPE
52         SYNTAX      WmanIfCidType
53         MAX-ACCESS  read-only
54         STATUS      current
55         DESCRIPTION
56             "The value of this object indicates the primary CID of the
57             SS received from the RNG-RSP message."
58         REFERENCE
59             "Subclause 6.3.2.3.6 in IEEE Std 802.16-2004"
60             ::= { wmanIfBsRegisteredSsEntry 3 }
61
62
63
64     wmanIfBsSsSecondaryCid OBJECT-TYPE
65

```

```

1      SYNTAX      WmanIfCidType
2      MAX-ACCESS  read-only
3      STATUS      current
4      DESCRIPTION
5          "The value of this object indicates the secondary
6          management CID present in the REG-RSP message. The value
7          should be null if the 2nd management channel is not
8          available."
9      REFERENCE
10         "Subclause 6.4.2.3.8 in IEEE Std 802.16-2004"
11         ::= { wmanIfBsRegisteredSsEntry 4 }
12
13 wmanIfBsSsManagementSupport OBJECT-TYPE
14     SYNTAX      INTEGER {unmanagedSs(0),
15                          managedSs(1)}
16     MAX-ACCESS  read-only
17     STATUS      current
18     DESCRIPTION
19         "This object indicates whether or not the SS is managed."
20     REFERENCE
21         "Subclause 11.7.2 in IEEE Std 802.16-2004"
22         ::= { wmanIfBsRegisteredSsEntry 5 }
23
24 wmanIfBsSsIpManagementMode OBJECT-TYPE
25     SYNTAX      INTEGER {unmanaged(0),
26                          ipManaged(1)}
27     MAX-ACCESS  read-only
28     STATUS      current
29     DESCRIPTION
30         "The IP management mode parameter dictates whether
31         the provider intends to manage the SS on an ongoing
32         basis via IP-based mechanisms."
33     REFERENCE
34         "Subclause 11.7.3 in IEEE Std 802.16-2004"
35         ::= { wmanIfBsRegisteredSsEntry 6 }
36
37 wmanIfBsSs2ndMgmtArqEnable OBJECT-TYPE
38     SYNTAX      TruthValue
39     MAX-ACCESS  read-only
40     STATUS      current
41     DESCRIPTION
42         "True(1) ARQ enabling is requested for the 2nd
43         management channel."
44     REFERENCE
45         "Subclause 11.13.18.1 in IEEE Std 802.16-2004"
46         ::= { wmanIfBsRegisteredSsEntry 7 }
47
48 wmanIfBsSs2ndMgmtArqWindowSize OBJECT-TYPE
49     SYNTAX      INTEGER (1 .. 1024)
50     MAX-ACCESS  read-only
51     STATUS      current
52     DESCRIPTION
53         "Indicates the maximum number of unacknowledged
54         fragments at any time for 2nd management channel."
55

```

```

1      REFERENCE
2          "Subclause 11.13.18.2 in IEEE Std 802.16-2004"
3      ::= { wmanIfBsRegisteredSsEntry 8 }
4
5
6  wmanIfBsSs2ndMgmtArqDnLinkTxDelay OBJECT-TYPE
7      SYNTAX      INTEGER (0 .. 65535)
8      UNITS       "us"
9      MAX-ACCESS  read-only
10     STATUS      current
11     DESCRIPTION
12         "The object defines the ARQ transmitter delay for
13         downlink transmission."
14     REFERENCE
15         "Subclause 11.13.18.3 in IEEE Std 802.16-2004"
16     ::= { wmanIfBsRegisteredSsEntry 9 }
17
18
19
20  wmanIfBsSs2ndMgmtArqUpLinkTxDelay OBJECT-TYPE
21      SYNTAX      INTEGER (0 .. 65535)
22      UNITS       "us"
23      MAX-ACCESS  read-only
24      STATUS      current
25      DESCRIPTION
26         "The object defines the ARQ transmitter delay for
27         uplink transmission."
28     REFERENCE
29         "Subclause 11.13.18.3 in IEEE Std 802.16-2004"
30     ::= { wmanIfBsRegisteredSsEntry 10 }
31
32
33
34
35  wmanIfBsSs2ndMgmtArqDnLinkRxDelay OBJECT-TYPE
36      SYNTAX      INTEGER (0 .. 65535)
37      UNITS       "us"
38      MAX-ACCESS  read-only
39      STATUS      current
40      DESCRIPTION
41         "The object defines the ARQ receiver delay for
42         downlink transmission."
43     REFERENCE
44         "Subclause 11.13.18.3 in IEEE Std 802.16-2004"
45     ::= { wmanIfBsRegisteredSsEntry 11 }
46
47
48
49  wmanIfBsSs2ndMgmtArqUpLinkRxDelay OBJECT-TYPE
50      SYNTAX      INTEGER (0 .. 65535)
51      UNITS       "us"
52      MAX-ACCESS  read-only
53      STATUS      current
54      DESCRIPTION
55         "The object defines the ARQ receiver delay for
56         uplink transmission."
57     REFERENCE
58         "Subclause 11.13.18.3 in IEEE Std 802.16-2004"
59     ::= { wmanIfBsRegisteredSsEntry 12 }
60
61
62
63  wmanIfBsSs2ndMgmtArqBlockLifetime OBJECT-TYPE
64      SYNTAX      INTEGER (0 .. 65535)
65

```

```

1      UNITS          "10 us"
2      MAX-ACCESS    read-only
3      STATUS        current
4      DESCRIPTION
5
6          "The maximum time interval an ARQ fragment will be
7          managed by the transmitter ARQ machine, once
8          initial transmission of the fragment has occurred.
9          If transmission or retransmission of the fragment
10         is not acknowledged by the receiver before the
11         time limit is reached, the fragment is discarded.
12         A value of 0 means Infinite."
13
14     REFERENCE
15         "Subclause 11.13.18.4 in IEEE Std 802.16-2004"
16     DEFVAL        {0}
17     ::= { wmanIfBsRegisteredSsEntry 13 }
18
19
20 wmanIfBsSs2ndMgmtArqSyncLossTimeout OBJECT-TYPE
21     SYNTAX          INTEGER (0 .. 65535)
22     UNITS            "10 us"
23     MAX-ACCESS      read-only
24     STATUS           current
25     DESCRIPTION
26
27         "The maximum interval before declaring a loss
28         of synchronization of the sender and receiver
29         state machines. A value of 0 means Infinite."
30
31     REFERENCE
32         "Subclause 11.13.18.5 in IEEE Std 802.16-2004"
33     DEFVAL          {0}
34     ::= { wmanIfBsRegisteredSsEntry 14 }
35
36
37 wmanIfBsSs2ndMgmtArqDeliverInOrder OBJECT-TYPE
38     SYNTAX          TruthValue
39     MAX-ACCESS      read-only
40     STATUS           current
41     DESCRIPTION
42
43         "Indicates whether or not data is to be delivered
44         by the receiving MAC to its client application
45         in the order in which data was handed off to the
46         originating MAC."
47
48     REFERENCE
49         "Subclause 11.13.18.6 in IEEE Std 802.16-2004"
50     ::= { wmanIfBsRegisteredSsEntry 15 }
51
52
53 wmanIfBsSs2ndMgmtArqRxPurgeTimeout OBJECT-TYPE
54     SYNTAX          INTEGER (0 .. 65535)
55     UNITS            "10 us"
56     MAX-ACCESS      read-only
57     STATUS           current
58     DESCRIPTION
59
60         "Indicates the time interval the ARQ window is advanced
61         after a fragment is received. A value of 0 means Infinite."
62
63     REFERENCE
64         "Subclause 11.13.18.7 in IEEE Std 802.16-2004"
65     DEFVAL          {0}

```

```

1      ::= { wmanIfBsRegisteredSsEntry 16 }
2
3
4  wmanIfBsSs2ndMgmtArqBlockSize OBJECT-TYPE
5      SYNTAX      INTEGER (1 .. 2040)
6      MAX-ACCESS  read-only
7      STATUS      current
8      DESCRIPTION
9          "This parameter specifies the size of a ARQ block. This
10         parameter shall be established by negotiation during the
11         connection setup. The requester includes its desired
12         setting in the REQ message. The receiver of the REQ
13         message shall take the smaller of the value it prefers and
14         value in the REQ message. The minimum value is included in
15         the RSP message."
16
17  REFERENCE
18      "Subclause 11.13.18.8 in IEEE Std 802.16-2004"
19
20  ::= { wmanIfBsRegisteredSsEntry 17 }
21
22
23  wmanIfBsSsVendorIdEncoding OBJECT-TYPE
24      SYNTAX      OCTET STRING (SIZE(3))
25      MAX-ACCESS  read-only
26      STATUS      current
27      DESCRIPTION
28          "The value field contains the vendor identification
29         specified by the 3 byte vendor-specific organizationally
30         unique identifier of the SS or BS MAC address. A vendor ID
31         used in a REG-REQ shall be the Vendor ID of the SS sending
32         the request. A vendor ID used in a REG-RSP shall be the
33         Vendor ID of the BS sending the response."
34
35  REFERENCE
36      "Subclause 11.1.5 in IEEE Std 802.16-2004"
37
38  ::= { wmanIfBsRegisteredSsEntry 18 }
39
40
41  wmanIfBsSsAasBroadcastPermission OBJECT-TYPE
42      SYNTAX      INTEGER {contBasedBwReqPermitted(0),
43                          contBasedBwReqNotPermitted(1)}
44      MAX-ACCESS  read-only
45      STATUS      current
46      DESCRIPTION
47          "This parameter specifies if SS can issue contention-based
48         bandwidth request or not."
49
50  REFERENCE
51      "Subclause 11.6 in IEEE Std 802.16-2004"
52
53  ::= { wmanIfBsRegisteredSsEntry 19 }
54
55  wmanIfBsSsMaxTxPowerBpsk OBJECT-TYPE
56      SYNTAX      WmanIfMaxTxPowerType
57      MAX-ACCESS  read-only
58      STATUS      current
59      DESCRIPTION
60          "The maximum available power for BPSK. The maximum power
61         parameters are reported in dBm and quantized in 0.5 dBm
62         steps ranging from -64 dBm (encoded 0x00) to 63.5 dBm
63         (encoded 0xFF). Values outside this range shall be assigned
64
65

```

the closest extreme. This parameter is only applicable to systems supporting the SCa, OFDM or OFDMA PHY."

REFERENCE

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

wmanIfBsSsMaxTxPowerQpsk OBJECT-TYPE

SYNTAX WmanIfMaxTxPowerType

MAX-ACCESS read-only

STATUS current

DESCRIPTION

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

REFERENCE

"Subclause 11.8.3.2 in IEEE Std 802.16-2004"
 ::= { wmanIfBsRegisteredSsEntry 21 }

wmanIfBsSsMaxTxPower16Qam OBJECT-TYPE

SYNTAX WmanIfMaxTxPowerType

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The maximum available power for 16-QAM constellations. The maximum power parameters are reported in dBm and quantized in 0.5 dBm steps ranging from -64 dBm (encoded 0x00) to 63.5 dBm (encoded 0xFF). Values outside this range shall be assigned the closest extreme. This parameter is only applicable to systems supporting the SCa, OFDM or OFDMA PHY."

REFERENCE

"Subclause 11.8.3.2 in IEEE Std 802.16-2004"
 ::= { wmanIfBsRegisteredSsEntry 22 }

wmanIfBsSsMaxTxPower64Qam OBJECT-TYPE

SYNTAX WmanIfMaxTxPowerType

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The maximum available power for 64-QAM constellations. The maximum power parameters are reported in dBm and quantized in 0.5 dBm steps ranging from -64 dBm (encoded 0x00) to 63.5 dBm (encoded 0xFF). Values outside this range shall be assigned the closest extreme. Ss that do not support QAM64 shall report the value of 0x00. This parameter is only applicable to systems supporting the SCa, OFDM or OFDMA PHY."

REFERENCE

"Subclause 11.8.3.2 in IEEE Std 802.16-2004"
 ::= { wmanIfBsRegisteredSsEntry 23 }

```

1  wmanIfBsSsMacVersion OBJECT-TYPE
2      SYNTAX      WmanIfMacVersion
3      MAX-ACCESS  read-only
4      STATUS      current
5      DESCRIPTION
6          "This parameter specifies the version of 802.16 to which the
7          message originator conforms."
8      REFERENCE
9          "Subclause 11.1.3 in IEEE Std 802.16-2004"
10     ::= { wmanIfBsRegisteredSsEntry 24 }
11
12  --
13  -- wmanIfBsConfigurationTable contains global parameters common in BS
14  --
15  wmanIfBsConfigurationTable OBJECT-TYPE
16      SYNTAX      SEQUENCE OF WmanIfBsConfigurationEntry
17      MAX-ACCESS  not-accessible
18      STATUS      current
19      DESCRIPTION
20          "This table provides one row for each BS sector that
21          contains the BS system parameters as defined in Subclause
22          10.1 of [3]. The objects in this table define the default
23          behaviour of the BS for 2nd Management Channel scheduling
24          and SFID allocation as well as configuration parameters
25          of the CPS scheduler and AAS system."
26      REFERENCE
27          "Subclause 10.1 in IEEE Std 802.16-2004"
28     ::= { wmanIfBsCps 2 }
29
30  wmanIfBsConfigurationEntry OBJECT-TYPE
31      SYNTAX      WmanIfBsConfigurationEntry
32      MAX-ACCESS  not-accessible
33      STATUS      current
34      DESCRIPTION
35          "This table is indexed by ifIndex with an ifType of
36          propBWA2Mp."
37      INDEX { ifIndex }
38     ::= { wmanIfBsConfigurationTable 1 }
39
40  WmanIfBsConfigurationEntry ::= SEQUENCE {
41      wmanIfBsDcdInterval          INTEGER,
42      wmanIfBsUcdInterval          INTEGER,
43      wmanIfBsUcdTransition        INTEGER,
44      wmanIfBsDcdTransition        INTEGER,
45      wmanIfBsInitialRangingInterval INTEGER,
46      wmanIfBsSsULMapProcTime      Unsigned32,
47      wmanIfBsSsRangRespProcTime   Unsigned32,
48      wmanIfBsT5Timeout            INTEGER,
49      wmanIfBsT9Timeout            INTEGER,
50      wmanIfBsT13Timeout           INTEGER,
51      wmanIfBsT15Timeout           INTEGER,
52      wmanIfBsT17Timeout           INTEGER,
53      wmanIfBsT27IdleTimer         Unsigned32,
54      wmanIfBsT27ActiveTimer       Unsigned32,

```

```

1      wmanIfBs2ndMgmtDlQoSProfileIndex      INTEGER,
2      wmanIfBs2ndMgmtUlQoSProfileIndex      INTEGER,
3      wmanIfBsAutoSfidEnabled                INTEGER,
4      wmanIfBsAutoSfidRangeMin              Unsigned32,
5      wmanIfBsAutoSfidRangeMax              Unsigned32,
6      wmanIfBsAasChanFbckReqFreq            INTEGER,
7      wmanIfBsAasBeamSelectFreq             INTEGER,
8      wmanIfBsAasChanFbckReqResolution      INTEGER,
9      wmanIfBsAasBeamReqResolution           INTEGER,
10     wmanIfBsAasNumOptDiversityZones        INTEGER,
11     wmanIfBsResetSector                    INTEGER}
12
13
14
15     wmanIfBsDcdInterval OBJECT-TYPE
16         SYNTAX      INTEGER (0..10000)
17         UNITS        "milliseconds"
18         MAX-ACCESS   read-write
19         STATUS        current
20         DESCRIPTION
21             "Time between transmission of DCD messages in ms."
22             ::= { wmanIfBsConfigurationEntry 1 }
23
24
25
26     wmanIfBsUcdInterval OBJECT-TYPE
27         SYNTAX      INTEGER (0..10000)
28         UNITS        "milliseconds"
29         MAX-ACCESS   read-write
30         STATUS        current
31         DESCRIPTION
32             "Time between transmission of UCD messages in ms."
33             ::= { wmanIfBsConfigurationEntry 2 }
34
35
36
37     wmanIfBsUcdTransition 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             ::= { wmanIfBsConfigurationEntry 3 }
48
49
50
51     wmanIfBsDcdTransition OBJECT-TYPE
52         SYNTAX      INTEGER (2..65535)
53         UNITS        "Number of MAC Frames"
54         MAX-ACCESS   read-write
55         STATUS        current
56         DESCRIPTION
57             "The time the BS shall wait after transmitting a DCD message
58             with an incremented Configuration Change Count before
59             issuing a DL-MAP message referring to
60             Downlink_Burst_Profiles defined in that DCD message."
61             ::= { wmanIfBsConfigurationEntry 4 }
62
63
64
65

```

```

1  wmanIfBsInitialRangingInterval OBJECT-TYPE
2      SYNTAX      INTEGER(0..2000)
3      UNITS       "milliseconds"
4      MAX-ACCESS  read-write
5      STATUS      current
6      DESCRIPTION
7          "Time between Initial Ranging regions assigned by the BS
8              in ms."
9      ::= { wmanIfBsConfigurationEntry 5 }
10
11
12
13  wmanIfBsSsULMapProcTime OBJECT-TYPE
14      SYNTAX      Unsigned32 (200 .. 4294967295)
15      UNITS       "micro seconds"
16      MAX-ACCESS  read-write
17      STATUS      current
18      DESCRIPTION
19          "Time provided between arrival of the last bit of a UL-MAP
20              at an SS and effectiveness of that map in us."
21      ::= { wmanIfBsConfigurationEntry 6 }
22
23
24
25  wmanIfBsSsRangRespProcTime OBJECT-TYPE
26      SYNTAX      Unsigned32 (10000 .. 4294967295)
27      UNITS       "micro seconds"
28      MAX-ACCESS  read-write
29      STATUS      current
30      DESCRIPTION
31          "Time allowed for an SS following receipt of a ranging
32              response before it is expected to reply to an invited
33              ranging request in us."
34      ::= { wmanIfBsConfigurationEntry 7 }
35
36
37
38  wmanIfBsT5Timeout OBJECT-TYPE
39      SYNTAX      INTEGER (0 .. 2000)
40      UNITS       "milliseconds"
41      MAX-ACCESS  read-write
42      STATUS      current
43      DESCRIPTION
44          "Wait for Uplink Channel Change Response in ms."
45      ::= { wmanIfBsConfigurationEntry 8 }
46
47
48
49  wmanIfBsT9Timeout OBJECT-TYPE
50      SYNTAX      INTEGER (300 .. 65535)
51      UNITS       "milliseconds"
52      MAX-ACCESS  read-write
53      STATUS      current
54      DESCRIPTION
55          "Registration Timeout, the time allowed between the BS
56              sending a RNG-RSP (success) to an SS, and receiving a
57              SBC-REQ from that same SS in ms."
58      ::= { wmanIfBsConfigurationEntry 9 }
59
60
61
62  wmanIfBsT13Timeout OBJECT-TYPE
63      SYNTAX      INTEGER (15 .. 65535)
64      UNITS       "minutes"
65

```

```

1      MAX-ACCESS  read-write
2      STATUS      current
3      DESCRIPTION
4          "The time allowed for an SS, following receipt of a
5          REG-RSP message to send a TFTP-CPLT message to the BS
6          in min."
7      ::= { wmanIfBsConfigurationEntry 10 }
8
9
10
11 wmanIfBsT15Timeout OBJECT-TYPE
12     SYNTAX      INTEGER (20 .. 65535)
13     UNITS        "milliseconds"
14     MAX-ACCESS  read-write
15     STATUS      current
16     DESCRIPTION
17         "Wait for MCA-RSP in ms."
18     ::= { wmanIfBsConfigurationEntry 11 }
19
20
21
22 wmanIfBsT17Timeout OBJECT-TYPE
23     SYNTAX      INTEGER (5 .. 65535)
24     UNITS        "minutes"
25     MAX-ACCESS  read-write
26     STATUS      current
27     DESCRIPTION
28         "Time allowed for SS to complete SS Authorization and
29         Key Exchange in minutes."
30     ::= { wmanIfBsConfigurationEntry 12 }
31
32
33
34 wmanIfBsT27IdleTimer OBJECT-TYPE
35     SYNTAX      Unsigned32 (10000 .. 4294967295)
36     UNITS        "us"
37     MAX-ACCESS  read-write
38     STATUS      current
39     DESCRIPTION
40         "Maximum time between unicast grants to SS when BS believes
41         SS uplink transmission quality is good enough."
42     ::= { wmanIfBsConfigurationEntry 13 }
43
44
45
46 wmanIfBsT27ActiveTimer OBJECT-TYPE
47     SYNTAX      Unsigned32 (10000 .. 4294967295)
48     UNITS        "us"
49     MAX-ACCESS  read-write
50     STATUS      current
51     DESCRIPTION
52         "Maximum time between unicast grants to SS when BS believes
53         SS uplink transmission quality is not good enough."
54     ::= { wmanIfBsConfigurationEntry 14 }
55
56
57
58
59 wmanIfBs2ndMgmtDlQoSProfileIndex OBJECT-TYPE
60     SYNTAX      INTEGER (1..65535)
61     MAX-ACCESS  read-write
62     STATUS      current
63     DESCRIPTION
64         "This object defines the index of a row in
65

```

```

1          wmanIfBsServiceClassTable which is used to obtain all QoS
2          parameters required for the BS downlink scheduler to
3          properly allocate and manage the bandwidth and schedule
4          the 2nd Management Connection traffic. The 2nd Management
5          Connection traffic doesn't differ from Traffic Connection
6          traffic in the area of QoS management."
7
8      ::= { wmanIfBsConfigurationEntry 15 }
9
10
11 wmanIfBs2ndMgmtUlQoSProfileIndex OBJECT-TYPE
12     SYNTAX      INTEGER (1..65535)
13     MAX-ACCESS   read-write
14     STATUS       current
15     DESCRIPTION
16         "This object defines the index of a row in
17         wmanIfBsServiceClassTable which is used to obtain all QoS
18         parameters required for the BS uplink scheduler to
19         properly allocate and manage the bandwidth and schedule
20         the 2nd Management Connection traffic. The 2nd Management
21         Connection traffic doesn't differ from Traffic Connection
22         traffic in the area of QoS management."
23
24     ::= { wmanIfBsConfigurationEntry 16 }
25
26
27 wmanIfBsAutoSfidEnabled OBJECT-TYPE
28     SYNTAX      INTEGER {autoSfidDisabled(0),
29                        autoSfidEnabled(1)}
30     MAX-ACCESS   read-write
31     STATUS       current
32     DESCRIPTION
33         "This object defines whether the BS is allowed to
34         autonomously allocate SFIDs. When the object is set to
35         autoSfidEnabled, the BS is allowed to autonomously allocate
36         SFIDs from the range of allowed values defined by
37         wmanIfBsConfigExtAutoSfidRangeMin and
38         wmanIfBsConfigExtAutoSfidRangeMax. A SF is created
39         autonomously when it has not been provisioned in the
40         wmanIfBsProvisionedSfTable and may be initiated by either
41         the SS or BS. The BS should always initiate SF creation
42         based on the provisioned Service flows configured in
43         wmanIfBsProvisionedSfTable."
44
45     REFERENCE
46         "Subclause 11.13.1 in IEEE Std 802.16-2004"
47
48     ::= { wmanIfBsConfigurationEntry 17 }
49
50
51
52 wmanIfBsAutoSfidRangeMin OBJECT-TYPE
53     SYNTAX      Unsigned32 ( 1 .. 4294967295)
54     MAX-ACCESS   read-write
55     STATUS       current
56     DESCRIPTION
57         "This object defines the minimum value of the range of SFID
58         values allocated for the BS sector for the purpose of
59         autonomous creation of service flows. This value is used
60         when the object wmanIfBsAutoSfidEnabled allows
61         autonomous creation of SFIDs."
62
63     REFERENCE
64
65

```

```

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

```

```

1          (-100, -96, -92, ..., 100)
2      aasChanFbckRes01 - every 8th carrier
3          (-100, -92, -84, ..., 100)
4      aasChanFbckRes10 - every 16th carrier
5          (-100, -84, -68, ..., 100)
6      aasChanFbckRes11 - every 32th carrier
7          (-100, -68, -36, ..., 100)"
8
9  REFERENCE
10     "Subclause 8.3.6.4 in IEEE Std 802.16-2004"
11     ::= { wmanIfBsConfigurationEntry 22 }
12
13
14  wmanIfBsAasBeamReqResolution OBJECT-TYPE
15      SYNTAX      INTEGER {aasBeamReqRes000(0),
16                      aasBeamReqRes001(1),
17                      aasBeamReqRes010(2),
18                      aasBeamReqRes011(3),
19                      aasBeamReqRes100(4)}
20
21      MAX-ACCESS   read-write
22      STATUS       current
23      DESCRIPTION
24          "This object defines AAS beam select request resolution
25          parameter. It is coded as follows:
26          aasBeamReqRes000 - every 4th carrier
27          aasBeamReqRes001 - every 8th carrier
28          aasBeamReqRes010 - every 16th carrier
29          aasBeamReqRes011 - every 32th carrier
30          aasBeamReqRes100 - every 64th carrier"
31
32  REFERENCE
33     "Subclause 8.3.6.5 in IEEE Std 802.16-2004"
34     ::= { wmanIfBsConfigurationEntry 23 }
35
36
37
38  wmanIfBsAasNumOptDiversityZones OBJECT-TYPE
39      SYNTAX      INTEGER (0..65535)
40      MAX-ACCESS   read-write
41      STATUS       current
42      DESCRIPTION
43          "This object defines the number of optional diversity zones
44          transmitted in downlink."
45
46  REFERENCE
47     "Figure 209 in IEEE Std 802.16-2004"
48     ::= { wmanIfBsConfigurationEntry 24 }
49
50
51
52  wmanIfBsResetSector OBJECT-TYPE
53      SYNTAX      INTEGER {actionResetSectorNoAction(0),
54                      actionResetSector(1)}
55      MAX-ACCESS   read-write
56      STATUS       current
57      DESCRIPTION
58          "This object should be implemented as follows:
59          - When set to actionsResetSector value, instructs BS to
60            reset the sector identified by ifIndex. As a result of
61            this action the Phy and Mac of this sector should be
62            reinitialised.
63          - When set to value different than actionsResetSector it
64
65

```

```

1          should be ignored
2          - When read it should return actionsResetSectorNoAction"
3      ::= { wmanIfBsConfigurationEntry 25 }
4
5
6      --
7      -- Base Station Channel Measurement Table
8      --
9      wmanIfBsChannelMeasurementTable OBJECT-TYPE
10         SYNTAX      SEQUENCE OF WmanIfBsChannelMeasurementEntry
11         MAX-ACCESS   not-accessible
12         STATUS       current
13         DESCRIPTION
14             "This table contains channel measurement information as
15             derived from BS measurement of uplink signal from SS,
16             and the downlink signal as reported from SS using
17             REP-REQ/RSP messages. The table shall be maintained as
18             FIFO to store measurement samples that can be used to
19             create RSSI and CINR histogram report. When the
20             measurement entry for a SS reaches the limit, the oldest
21             entry shall be deleted as the new entry is added to the
22             table."
23         REFERENCE
24             "6.3.2.3.33 in IEEE Std 802.16-2004"
25         ::= { wmanIfBsCps 3 }
26
27
28      wmanIfBsChannelMeasurementEntry OBJECT-TYPE
29         SYNTAX      WmanIfBsChannelMeasurementEntry
30         MAX-ACCESS   not-accessible
31         STATUS       current
32         DESCRIPTION
33             "Each entry in the table contains RSSI and CINR
34             signal quality measurement on signal received from the SS.
35             The primary index is the ifIndex with ifType of propBWAmp2Mp
36             identifying the BS sector. wmanIfBsSsMacAddress identifies
37             the SS from which the signal was received.
38             wmanIfBsChannelDirection is the index to the direction of
39             the channel. wmanIfBsHistogramIndex is the index to
40             histogram samples. Since there is no time stamp in the
41             table, wmanIfBsHistogramIndex should be increased
42             monotonically, and wraps around when it reaches the
43             implementation specific limit."
44         INDEX      { ifIndex,
45                     wmanIfBsSsMacAddress,
46                     wmanIfBsChannelDirection,
47                     wmanIfBsHistogramIndex }
48         ::= { wmanIfBsChannelMeasurementTable 1 }
49
50
51      WmanIfBsChannelMeasurementEntry ::= SEQUENCE {
52          wmanIfBsChannelDirection      INTEGER,
53          wmanIfBsHistogramIndex        Unsigned32,
54          wmanIfBsChannelNumber         WmanIfChannelNumber,
55          wmanIfBsStartFrame            INTEGER,
56          wmanIfBsDuration              INTEGER,
57          wmanIfBsBasicReport           BITS,

```

```

1          wmanIfBsMeanCinrReport          INTEGER,
2          wmanIfBsMeanRssiReport          INTEGER,
3          wmanIfBsStdDeviationCinrReport  INTEGER,
4          wmanIfBsStdDeviationRssiReport  INTEGER}
5
6
7  wmanIfBsChannelDirection OBJECT-TYPE
8      SYNTAX      INTEGER {downstream(1),
9                      upstream(2)}
10
11     MAX-ACCESS   not-accessible
12     STATUS      current
13     DESCRIPTION
14         "wmanIfBsChannelDirection identifies the direction of a
15         a channel where the measurement takes place."
16     ::= { wmanIfBsChannelMeasurementEntry 1 }
17
18
19  wmanIfBsHistogramIndex OBJECT-TYPE
20      SYNTAX      Unsigned32 (1 .. 4294967295)
21     MAX-ACCESS   read-only
22     STATUS      current
23     DESCRIPTION
24         "wmanIfBsHistogramIndex identifies the histogram samples
25         in the table for each subscriber station."
26     ::= { wmanIfBsChannelMeasurementEntry 2 }
27
28
29
30  wmanIfBsChannelNumber OBJECT-TYPE
31      SYNTAX      WmanIfChannelNumber
32     MAX-ACCESS   read-only
33     STATUS      current
34     DESCRIPTION
35         "Physical channel number to be reported on is only
36         applicable to license exempt band. For licensed band,
37         this parameter should be null."
38     REFERENCE
39         "Subclause 11.12 in IEEE Std 802.16-2004"
40     ::= { wmanIfBsChannelMeasurementEntry 3 }
41
42
43
44  wmanIfBsStartFrame OBJECT-TYPE
45      SYNTAX      INTEGER (0..65535)
46     MAX-ACCESS   read-only
47     STATUS      current
48     DESCRIPTION
49         "Frame number in which measurement for this channel
50         started."
51     REFERENCE
52         "Subclause 11.12 in IEEE Std 802.16-2004"
53     ::= { wmanIfBsChannelMeasurementEntry 4 }
54
55
56
57  wmanIfBsDuration OBJECT-TYPE
58      SYNTAX      INTEGER (0 .. 16777215)
59     MAX-ACCESS   read-only
60     STATUS      current
61     DESCRIPTION
62         "Cumulative measurement duration on the channel in
63         multiples of Ts. For any value exceeding 0xFFFFF,
64
65

```

```

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

```

```

1  wmanIfBsStdDeviationRssiReport OBJECT-TYPE
2      SYNTAX      INTEGER (0 .. 83)
3      UNITS       "dB"
4      MAX-ACCESS  read-only
5      STATUS      current
6      DESCRIPTION
7          "Standard deviation RSSI report."
8
9      REFERENCE
10         "Subclause 8.2.2, 8.3.9 in IEEE Std 802.16-2004"
11
12     ::= { wmanIfBsChannelMeasurementEntry 10 }
13
14 --
15 -- Base Station capabilities
16 --
17
18 wmanIfBsCapabilities OBJECT IDENTIFIER ::= { wmanIfBsCps 4 }
19
20 wmanIfBsSsReqCapabilitiesTable OBJECT-TYPE
21     SYNTAX      SEQUENCE OF WmanIfBsSsReqCapabilitiesEntry
22     MAX-ACCESS  not-accessible
23     STATUS      current
24     DESCRIPTION
25         "This table contains the basic capability information of SSs
26         that have been reported by SSs to BS using RNG-REQ, SBC-REQ
27         and REG-REQ messages. Entries in this table should be
28         created when an SS registers with a BS."
29
30     ::= { wmanIfBsCapabilities 1 }
31
32
33 wmanIfBsSsReqCapabilitiesEntry OBJECT-TYPE
34     SYNTAX      WmanIfBsSsReqCapabilitiesEntry
35     MAX-ACCESS  not-accessible
36     STATUS      current
37     DESCRIPTION
38         "This table provides one row for each SS that has been
39         registered in the BS. This table augments the table
40         wmanIfBsRegisteredSsTable."
41
42     AUGMENTS { wmanIfBsRegisteredSsEntry }
43
44     ::= { wmanIfBsSsReqCapabilitiesTable 1 }
45
46
47 WmanIfBsSsReqCapabilitiesEntry ::= SEQUENCE {
48     wmanIfBsSsReqCapUplinkCidSupport      WmanIfNumOfUplinkCid,
49     wmanIfBsSsReqCapArqSupport            WmanIfArqSupportType,
50     wmanIfBsSsReqCapDsxFowControl         WmanIfMaxDsxFowType,
51     wmanIfBsSsReqCapMacCrcSupport         WmanIfMacCrcSupport,
52     wmanIfBsSsReqCapMcaFlowControl        WmanIfMaxMcaFlowType,
53     wmanIfBsSsReqCapMcpGroupCidSupport    WmanIfMaxMcpGroupCid,
54     wmanIfBsSsReqCapPkmFlowControl        WmanIfMaxPkmFlowType,
55     wmanIfBsSsReqCapAuthPolicyControl     WmanIfAuthPolicyType,
56     wmanIfBsSsReqCapMaxNumOfSupportedSA   WmanIfMaxNumOfSaType,
57     wmanIfBsSsReqCapIpVersion             WmanIfIpVersionType,
58     wmanIfBsSsReqCapMacCsSupportBitMap    WmanIfMacCsBitMap,
59     wmanIfBsSsReqCapMaxNumOfClassifier    WmanIfMaxClassifiers,
60     wmanIfBsSsReqCapPhsSupport            WmanIfPhsSupportType,
61     wmanIfBsSsReqCapBandwidthAllocSupport WmanIfBwAllocSupport,
62     wmanIfBsSsReqCapPduConstruction       WmanIfPduConstruction,

```

```

1          wmanIfBsSsReqCapTtgTransitionGap          WmanIfSsTransitionGap,
2          wmanIfBsSsReqCapRtgTransitionGap          WmanIfSsTransitionGap}
3
4
5  wmanIfBsSsReqCapUplinkCidSupport OBJECT-TYPE
6      SYNTAX      WmanIfNumOfUplinkCid
7      MAX-ACCESS  read-only
8      STATUS      current
9      DESCRIPTION
10         "This object shows the number of Uplink CIDs the SS can
11         support."
12         ::= { wmanIfBsSsReqCapabilitiesEntry 1 }
13
14
15  wmanIfBsSsReqCapArqSupport OBJECT-TYPE
16      SYNTAX      WmanIfArqSupportType
17      MAX-ACCESS  read-only
18      STATUS      current
19      DESCRIPTION
20         "This object indicates whether the SS supports ARQ."
21         ::= { wmanIfBsSsReqCapabilitiesEntry 2 }
22
23
24
25  wmanIfBsSsReqCapDsxFlowControl OBJECT-TYPE
26      SYNTAX      WmanIfMaxDsxFlowType
27      MAX-ACCESS  read-only
28      STATUS      current
29      DESCRIPTION
30         "This object specifies the maximum number of concurrent
31         DSA, DSC, or DSD transactions that SS is capable of having
32         outstanding."
33         DEFVAL    { 0 }
34         ::= { wmanIfBsSsReqCapabilitiesEntry 3 }
35
36
37
38  wmanIfBsSsReqCapMacCrcSupport OBJECT-TYPE
39      SYNTAX      WmanIfMacCrcSupport
40      MAX-ACCESS  read-only
41      STATUS      current
42      DESCRIPTION
43         "This object indicates whether or not the SS supports MAC
44         level CRC."
45         DEFVAL    { macCrcSupport }
46         ::= { wmanIfBsSsReqCapabilitiesEntry 4 }
47
48
49
50  wmanIfBsSsReqCapMcaFlowControl OBJECT-TYPE
51      SYNTAX      WmanIfMaxMcaFlowType
52      MAX-ACCESS  read-only
53      STATUS      current
54      DESCRIPTION
55         "This object specifies the maximum number of concurrent MCA
56         transactions that SS is capable of having outstanding."
57         DEFVAL    { 0 }
58         ::= { wmanIfBsSsReqCapabilitiesEntry 5 }
59
60
61
62  wmanIfBsSsReqCapMcpGroupCidSupport OBJECT-TYPE
63      SYNTAX      WmanIfMaxMcpGroupCid
64      MAX-ACCESS  read-only
65

```

```

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

```

```

1      STATUS      current
2      DESCRIPTION
3          "This object indicates SS reported set of MAC convergence
4          sublayer support. When a bit is set, it indicates
5          the corresponding CS feature is supported."
6      ::= { wmanIfBsSsReqCapabilitiesEntry 11 }
7
8
9
10     wmanIfBsSsReqCapMaxNumOfClassifier OBJECT-TYPE
11         SYNTAX      WmanIfMaxClassifiers
12         MAX-ACCESS   read-only
13         STATUS       current
14         DESCRIPTION
15             "This object indicates the maximum number of admitted
16             Classifiers that the SS can support."
17         DEFVAL       { 0 }
18         ::= { wmanIfBsSsReqCapabilitiesEntry 12 }
19
20
21
22     wmanIfBsSsReqCapPhsSupport OBJECT-TYPE
23         SYNTAX      WmanIfPhsSupportType
24         MAX-ACCESS   read-only
25         STATUS       current
26         DESCRIPTION
27             "This object indicates indicates the level of SS support
28             for PHS."
29         DEFVAL       { noPhsSupport }
30         ::= { wmanIfBsSsReqCapabilitiesEntry 13 }
31
32
33
34     wmanIfBsSsReqCapBandwidthAllocSupport OBJECT-TYPE
35         SYNTAX      WmanIfBwAllocSupport
36         MAX-ACCESS   read-only
37         STATUS       current
38         DESCRIPTION
39             "This field indicates the bandwidth allocation
40             capabilities of the SS. The usage is defined by
41             WmanIfBwAllocSupport."
42         ::= { wmanIfBsSsReqCapabilitiesEntry 14 }
43
44
45
46     wmanIfBsSsReqCapPduConstruction OBJECT-TYPE
47         SYNTAX      WmanIfPduConstruction
48         MAX-ACCESS   read-only
49         STATUS       current
50         DESCRIPTION
51             "This field indicates the SS's capabilities for
52             construction and transmission of MAC PDUs. The usage
53             is defined by WmanIfPduConstruction."
54         ::= { wmanIfBsSsReqCapabilitiesEntry 15 }
55
56
57
58     wmanIfBsSsReqCapTtgTransitionGap OBJECT-TYPE
59         SYNTAX      WmanIfSsTransitionGap
60         UNITS        "us"
61         MAX-ACCESS   read-only
62         STATUS       current
63         DESCRIPTION
64             "This field indicates the SS's transition speed SSTTG
65

```

```

1           for TDD and H-FDD SSs. The usage is defined by
2           WmanIfSsTransitionGap."
3           ::= { wmanIfBsSsReqCapabilitiesEntry 16 }
4
5
6 wmanIfBsSsReqCapRtgTransitionGap OBJECT-TYPE
7     SYNTAX      WmanIfSsTransitionGap
8     UNITS        "us"
9     MAX-ACCESS   read-only
10    STATUS       current
11    DESCRIPTION
12      "This field indicates the SS's transition speed SSRTG
13      for TDD and H-FDD SSs. The usage is defined by
14      WmanIfSsTransitionGap."
15      ::= { wmanIfBsSsReqCapabilitiesEntry 17 }
16
17
18
19 wmanIfBsSsRspCapabilitiesTable OBJECT-TYPE
20     SYNTAX      SEQUENCE OF WmanIfBsSsRspCapabilitiesEntry
21     MAX-ACCESS   not-accessible
22     STATUS       current
23     DESCRIPTION
24       "This table contains the basic capability information of SSs
25       that have been negotiated and agreed between BS and SS via
26       RNG-REQ/RSP, SBC-REQ/RSP and REG-REQ/RSP messages.
27       This table augments the wmanIfBsRegisteredSsTable."
28     REFERENCE
29       "Subclause 6.3.2.3.7 in IEEE Std 802.16-2004"
30     ::= { wmanIfBsCapabilities 2 }
31
32
33
34
35 wmanIfBsSsRspCapabilitiesEntry OBJECT-TYPE
36     SYNTAX      WmanIfBsSsRspCapabilitiesEntry
37     MAX-ACCESS   not-accessible
38     STATUS       current
39     DESCRIPTION
40       "This table provides one row for each SS that has been
41       registered in the BS. This table augments the
42       wmanIfBsRegisteredSsTable. "
43     AUGMENTS { wmanIfBsRegisteredSsEntry }
44     ::= { wmanIfBsSsRspCapabilitiesTable 1 }
45
46
47
48 WmanIfBsSsRspCapabilitiesEntry ::= SEQUENCE {
49     wmanIfBsSsRspCapUplinkCidSupport      WmanIfNumOfUplinkCid,
50     wmanIfBsSsRspCapArqSupport            WmanIfArqSupportType,
51     wmanIfBsSsRspCapDsxFlowControl        WmanIfMaxDsxFlowType,
52     wmanIfBsSsRspCapMacCrcSupport         WmanIfMacCrcSupport,
53     wmanIfBsSsRspCapMcaFlowControl        WmanIfMaxMcaFlowType,
54     wmanIfBsSsRspCapMcpGroupCidSupport    WmanIfMaxMcpGroupCid,
55     wmanIfBsSsRspCapPkmFlowControl        WmanIfMaxPkmFlowType,
56     wmanIfBsSsRspCapAuthPolicyControl     WmanIfAuthPolicyType,
57     wmanIfBsSsRspCapMaxNumOfSupportedSA   WmanIfMaxNumOfSaType,
58     wmanIfBsSsRspCapIpVersion            WmanIfIpVersionType,
59     wmanIfBsSsRspCapMacCsSupportBitMap    WmanIfMacCsBitMap,
60     wmanIfBsSsRspCapMaxNumOfClassifier    WmanIfMaxClassifiers,
61     wmanIfBsSsRspCapPhsSupport            WmanIfPhsSupportType,
62     wmanIfBsSsRspCapBandwidthAllocSupport WmanIfBwAllocSupport,
63

```

```

1          wmanIfBsSsRspCapPduConstruction          WmanIfPduConstruction,
2          wmanIfBsSsRspCapTtgTransitionGap          WmanIfSsTransitionGap,
3          wmanIfBsSsRspCapRtgTransitionGap          WmanIfSsTransitionGap}
4
5
6  wmanIfBsSsRspCapUplinkCidSupport OBJECT-TYPE
7      SYNTAX      WmanIfNumOfUplinkCid
8      MAX-ACCESS  read-only
9      STATUS      current
10     DESCRIPTION
11         "Negotiated number of Uplink CIDs the SS can support."
12     ::= { wmanIfBsSsRspCapabilitiesEntry 1 }
13
14
15  wmanIfBsSsRspCapArqSupport OBJECT-TYPE
16      SYNTAX      WmanIfArqSupportType
17      MAX-ACCESS  read-only
18      STATUS      current
19      DESCRIPTION
20         "This object indicates whether the SS is allowed to use ARQ
21         as a result of the capabilities negotiation."
22     ::= { wmanIfBsSsRspCapabilitiesEntry 2 }
23
24
25
26  wmanIfBsSsRspCapDsxFowControl OBJECT-TYPE
27      SYNTAX      WmanIfMaxDsxFowType
28      MAX-ACCESS  read-only
29      STATUS      current
30      DESCRIPTION
31         "Negotiated maximum number of concurrent DSA, DSC, or DSD
32         transactions that may be outstanding."
33     ::= { wmanIfBsSsRspCapabilitiesEntry 3 }
34
35
36
37  wmanIfBsSsRspCapMacCrcSupport OBJECT-TYPE
38      SYNTAX      WmanIfMacCrcSupport
39      MAX-ACCESS  read-only
40      STATUS      current
41      DESCRIPTION
42         "This object indicates whether or not the SS is allowed to
43         use MAC level CRC as a result of the capabilities
44         negotiation."
45     DEFVAL      { macCrcSupport }
46     ::= { wmanIfBsSsRspCapabilitiesEntry 4 }
47
48
49
50  wmanIfBsSsRspCapMcaFlowControl OBJECT-TYPE
51      SYNTAX      WmanIfMaxMcaFlowType
52      MAX-ACCESS  read-only
53      STATUS      current
54      DESCRIPTION
55         "Negotiated maximum number of concurrent
56         MCA transactions that may be outstanding."
57     DEFVAL      { 0 }
58     ::= { wmanIfBsSsRspCapabilitiesEntry 5 }
59
60
61
62  wmanIfBsSsRspCapMcpGroupCidSupport OBJECT-TYPE
63      SYNTAX      WmanIfMaxMcpGroupCid
64      MAX-ACCESS  read-only
65

```

```

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

```

```

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

```

```

1      WmanIfSsTransitionGap."
2      ::= { wmanIfBsSsRspCapabilitiesEntry 16 }
3
4
5  wmanIfBsSsRspCapRtgTransitionGap OBJECT-TYPE
6      SYNTAX      WmanIfSsTransitionGap
7      UNITS       "us"
8      MAX-ACCESS  read-only
9      STATUS      current
10     DESCRIPTION
11         "This field indicates the negotiated transition speed
12         SSRTG for TDD and H-FDD SSs. The usage is defined by
13         WmanIfSsTransitionGap."
14     ::= { wmanIfBsSsRspCapabilitiesEntry 17 }
15
16
17
18  wmanIfBsBasicCapabilitiesTable OBJECT-TYPE
19      SYNTAX      SEQUENCE OF WmanIfBsBasicCapabilitiesEntry
20      MAX-ACCESS  not-accessible
21      STATUS      current
22      DESCRIPTION
23         "This table contains the basic capabilities of the BS as
24         implemented in BS hardware and software. These capabilities
25         along with the configuration for them
26         (wmanIfBsCapabilitiesConfigTable) are used for negotiation
27         of basic capabilities with SS using RNG-RSP, SBC-RSP and
28         REG-RSP messages. The negotiated capabilities are obtained
29         by interSubclause of SS raw reported capabilities, BS raw
30         capabilities and BS configured capabilities. The objects in
31         the table have read-only access. The table is maintained
32         by BS."
33     ::= { wmanIfBsCapabilities 3 }
34
35
36
37
38  wmanIfBsBasicCapabilitiesEntry OBJECT-TYPE
39      SYNTAX      WmanIfBsBasicCapabilitiesEntry
40      MAX-ACCESS  not-accessible
41      STATUS      current
42      DESCRIPTION
43         "This table provides one row for each BS sector and is
44         indexed by ifIndex."
45     INDEX { ifIndex }
46     ::= { wmanIfBsBasicCapabilitiesTable 1 }
47
48
49
50  WmanIfBsBasicCapabilitiesEntry ::= SEQUENCE {
51      wmanIfBsCapUplinkCidSupport          WmanIfNumOfUplinkCid,
52      wmanIfBsCapArqSupport                WmanIfArqSupportType,
53      wmanIfBsCapDsxFlowControl            WmanIfMaxDsxFlowType,
54      wmanIfBsCapMacCrcSupport             WmanIfMacCrcSupport,
55      wmanIfBsCapMcaFlowControl            WmanIfMaxMcaFlowType,
56      wmanIfBsCapMcpGroupCidSupport        WmanIfMaxMcpGroupCid,
57      wmanIfBsCapPkmFlowControl            WmanIfMaxPkmFlowType,
58      wmanIfBsCapAuthPolicyControl         WmanIfAuthPolicyType,
59      wmanIfBsCapMaxNumOfSupportedSA       WmanIfMaxNumOfSaType,
60      wmanIfBsCapIpVersion                 WmanIfIpVersionType,
61      wmanIfBsCapMacCsSupportBitMap        WmanIfMacCsBitMap,
62      wmanIfBsCapMaxNumOfClassifier        WmanIfMaxClassifiers,
63
64
65

```

```

1      wmanIfBsCapPhsSupport      WmanIfPhsSupportType,
2      wmanIfBsCapBandwidthAllocSupport  WmanIfBwAllocSupport,
3      wmanIfBsCapPduConstruction    WmanIfPduConstruction,
4      wmanIfBsCapTtgTransitionGap    WmanIfSsTransitionGap,
5      wmanIfBsCapRtgTransitionGap    WmanIfSsTransitionGap}
6
7
8      wmanIfBsCapUplinkCidSupport OBJECT-TYPE
9          SYNTAX      WmanIfNumOfUplinkCid
10         MAX-ACCESS  read-only
11         STATUS      current
12         DESCRIPTION
13             "This object shows the number of Uplink CIDs the BS can
14             support per SS."
15         ::= { wmanIfBsBasicCapabilitiesEntry 1 }
16
17
18      wmanIfBsCapArqSupport OBJECT-TYPE
19          SYNTAX      WmanIfArqSupportType
20          MAX-ACCESS  read-only
21          STATUS      current
22          DESCRIPTION
23              "This object indicates whether the BS supports ARQ."
24          ::= { wmanIfBsBasicCapabilitiesEntry 2 }
25
26
27      wmanIfBsCapDsxFowControl OBJECT-TYPE
28          SYNTAX      WmanIfMaxDsxFowType
29          MAX-ACCESS  read-only
30          STATUS      current
31          DESCRIPTION
32              "This object specifies the maximum number of concurrent
33              DSA, DSC, or DSD transactions that BS allows each SS to
34              have outstanding."
35          DEFVAL      { 0 }
36          ::= { wmanIfBsBasicCapabilitiesEntry 3 }
37
38
39      wmanIfBsCapMacCrcSupport OBJECT-TYPE
40          SYNTAX      WmanIfMacCrcSupport
41          MAX-ACCESS  read-only
42          STATUS      current
43          DESCRIPTION
44              "This object indicates whether or not the BS supports MAC
45              level CRC."
46          DEFVAL      { macCrcSupport }
47          ::= { wmanIfBsBasicCapabilitiesEntry 4 }
48
49
50      wmanIfBsCapMcaFlowControl OBJECT-TYPE
51          SYNTAX      WmanIfMaxMcaFlowType
52          MAX-ACCESS  read-only
53          STATUS      current
54          DESCRIPTION
55              "This object specifies the maximum number of concurrent
56              MCA transactions that BS allows each SS to have."
57          DEFVAL      { 0 }
58          ::= { wmanIfBsBasicCapabilitiesEntry 5 }
59
60
61
62
63
64
65

```

```

1  wmanIfBsCapMcpGroupCidSupport OBJECT-TYPE
2      SYNTAX      WmanIfMaxMcpGroupCid
3      MAX-ACCESS  read-only
4      STATUS      current
5      DESCRIPTION
6          "This object indicates the maximum number of simultaneous
7          Multicast Polling Groups the BS allows each SS to belong
8          to."
9      DEFVAL      { 0 }
10     ::= { wmanIfBsBasicCapabilitiesEntry 6 }
11
12  wmanIfBsCapPkmFlowControl OBJECT-TYPE
13      SYNTAX      WmanIfMaxPkmFlowType
14      MAX-ACCESS  read-only
15      STATUS      current
16      DESCRIPTION
17          "This object specifies the maximum number of concurrent
18          PKM transactions that BS allows each SS to have."
19      DEFVAL      { 0 }
20     ::= { wmanIfBsBasicCapabilitiesEntry 7 }
21
22  wmanIfBsCapAuthPolicyControl OBJECT-TYPE
23      SYNTAX      WmanIfAuthPolicyType
24      MAX-ACCESS  read-only
25      STATUS      current
26      DESCRIPTION
27          "This object specifies authorization policy that BS is
28          capable of. A bit value of 0 = not supported,
29          1 = upported. If this field is omitted, then both SS and
30          BS shall use the IEEE 802.16 security, constituting X.509
31          digital certificates and the RSA public key encryption
32          algorithm, as authorization policy."
33     ::= { wmanIfBsBasicCapabilitiesEntry 8 }
34
35  wmanIfBsCapMaxNumOfSupportedSA OBJECT-TYPE
36      SYNTAX      WmanIfMaxNumOfSaType
37      MAX-ACCESS  read-only
38      STATUS      current
39      DESCRIPTION
40          "This field specifies maximum number of supported security
41          associations per SS that the BS allows."
42      DEFVAL      { 1 }
43     ::= { wmanIfBsBasicCapabilitiesEntry 9 }
44
45  wmanIfBsCapIpVersion OBJECT-TYPE
46      SYNTAX      WmanIfIpVersionType
47      MAX-ACCESS  read-only
48      STATUS      current
49      DESCRIPTION
50          "This object indicates the version of IP BS allows each SS
51          to use on the 2nd Management Connection. The value
52          'undefined' should not be used for this field."
53      REFERENCE
54          "Subclause 11.7.4 in IEEE Std 802.16-2004"
55

```

```

1      ::= { wmanIfBsBasicCapabilitiesEntry 10 }
2
3
4  wmanIfBsCapMacCsSupportBitMap OBJECT-TYPE
5      SYNTAX      WmanIfMacCsBitMap
6      MAX-ACCESS  read-only
7      STATUS      current
8      DESCRIPTION
9          "This object indicates BS set of MAC convergence
10         sublayer support. When a bit is set, it indicates
11         the corresponding CS feature is supported."
12
13     ::= { wmanIfBsBasicCapabilitiesEntry 11 }
14
15
16  wmanIfBsCapMaxNumOfClassifier OBJECT-TYPE
17      SYNTAX      WmanIfMaxClassifiers
18      MAX-ACCESS  read-only
19      STATUS      current
20      DESCRIPTION
21          "This object indicates the maximum number of admitted
22         Classifiers per SS that the BS allows."
23
24      DEFVAL      { 0 }
25
26     ::= { wmanIfBsBasicCapabilitiesEntry 12 }
27
28
29  wmanIfBsCapPhsSupport OBJECT-TYPE
30      SYNTAX      WmanIfPhsSupportType
31      MAX-ACCESS  read-only
32      STATUS      current
33      DESCRIPTION
34          "This object indicates the level of BS support for PHS.
35         The usage is defined by WmanIfPhsSupportType."
36
37      DEFVAL      { noPhsSupport }
38
39     ::= { wmanIfBsBasicCapabilitiesEntry 13 }
40
41
42  wmanIfBsCapBandwidthAllocSupport OBJECT-TYPE
43      SYNTAX      WmanIfBwAllocSupport
44      MAX-ACCESS  read-only
45      STATUS      current
46      DESCRIPTION
47          "This field indicates the bandwidth allocation properties
48         that the BS permits SSs to use. The usage is defined by
49         WmanIfBwAllocSupport."
50
51     ::= { wmanIfBsBasicCapabilitiesEntry 14 }
52
53
54  wmanIfBsCapPduConstruction OBJECT-TYPE
55      SYNTAX      WmanIfPduConstruction
56      MAX-ACCESS  read-only
57      STATUS      current
58      DESCRIPTION
59          "Specifies the capabilities for construction and
60         transmission of MAC PDUs allowed by the BS. The usage is
61         defined by WmanIfPduConstruction."
62
63     ::= { wmanIfBsBasicCapabilitiesEntry 15 }
64
65
66  wmanIfBsCapTtgTransitionGap OBJECT-TYPE
67      SYNTAX      WmanIfSsTransitionGap

```

```

1      UNITS          "us"
2      MAX-ACCESS    read-only
3      STATUS        current
4      DESCRIPTION
5          "This field indicates the transition speed SSTTG for TDD
6          and H-FDD SSs allowed by the BS. The usage is defined by
7          WmanIfSsTransitionGap."
8      ::= { wmanIfBsBasicCapabilitiesEntry 16 }
9
10
11
12  wmanIfBsCapRtgTransitionGap OBJECT-TYPE
13      SYNTAX          WmanIfSsTransitionGap
14      UNITS           "us"
15      MAX-ACCESS      read-only
16      STATUS          current
17      DESCRIPTION
18          "This field indicates the transition speed SSRTG for TDD
19          and H-FDD SSs allowed by the BS. The usage is defined
20          by WmanIfSsTransitionGap."
21      ::= { wmanIfBsBasicCapabilitiesEntry 17 }
22
23
24
25  wmanIfBsCapabilitiesConfigTable OBJECT-TYPE
26      SYNTAX          SEQUENCE OF WmanIfBsCapabilitiesConfigEntry
27      MAX-ACCESS      not-accessible
28      STATUS          current
29      DESCRIPTION
30          "This table contains the configuration for basic
31          capabilities of BS. The table is intended to be used to
32          restrict the Capabilities implemented by BS, for example in
33          order to comply with local regulatory requirements. The BS
34          should use the configuration along with the implemented
35          Capabilities (wmanIfBsBasicCapabilitiesTable) for
36          negotiation of basic capabilities with SS using RNG-RSP,
37          SBC-RSP and REG-RSP messages. The negotiated capabilities
38          are obtained by interSubclause of SS reported capabilities,
39          BS raw capabilities and BS configured capabilities. The
40          objects in the table have read-write access. The rows are
41          created by BS as a copy of wmanIfBsBasicCapabilitiesTable
42          and can be modified by NMS."
43      ::= { wmanIfBsCapabilities 4 }
44
45
46
47
48
49  wmanIfBsCapabilitiesConfigEntry OBJECT-TYPE
50      SYNTAX          WmanIfBsCapabilitiesConfigEntry
51      MAX-ACCESS      not-accessible
52      STATUS          current
53      DESCRIPTION
54          "This table provides one row for each BS sector and is
55          indexed by ifIndex."
56      INDEX { ifIndex }
57      ::= { wmanIfBsCapabilitiesConfigTable 1 }
58
59
60
61  WmanIfBsCapabilitiesConfigEntry ::= SEQUENCE {
62      wmanIfBsCapCfgUplinkCidSupport      WmanIfNumOfUplinkCid,
63      wmanIfBsCapCfgArqSupport            WmanIfArqSupportType,
64      wmanIfBsCapCfgDsxFlowControl        WmanIfMaxDsxFlowType,
65

```

```

1      wmanIfBsCapCfgMacCrcSupport          WmanIfMacCrcSupport,
2      wmanIfBsCapCfgMcaFlowControl          WmanIfMaxMcaFlowType,
3      wmanIfBsCapCfgMcpGroupCidSupport      WmanIfMaxMcpGroupCid,
4      wmanIfBsCapCfgPkmFlowControl          WmanIfMaxPkmFlowType,
5      wmanIfBsCapCfgAuthPolicyControl       WmanIfAuthPolicyType,
6      wmanIfBsCapCfgMaxNumOfSupportedSA     WmanIfMaxNumOfSaType,
7      wmanIfBsCapCfgIpVersion               WmanIfIpVersionType,
8      wmanIfBsCapCfgMacCsSupportBitMap      WmanIfMacCsBitMap,
9      wmanIfBsCapCfgMaxNumOfClassifier      WmanIfMaxClassifiers,
10     wmanIfBsCapCfgPhsSupport               WmanIfPhsSupportType,
11     wmanIfBsCapCfgBandwidthAllocSupport    WmanIfBwAllocSupport,
12     wmanIfBsCapCfgPduConstruction          WmanIfPduConstruction,
13     wmanIfBsCapCfgTtgTransitionGap         WmanIfSsTransitionGap,
14     wmanIfBsCapCfgRtgTransitionGap         WmanIfSsTransitionGap}
15
16
17
18
19     wmanIfBsCapCfgUplinkCidSupport OBJECT-TYPE
20         SYNTAX      WmanIfNumOfUplinkCid
21         MAX-ACCESS  read-write
22         STATUS      current
23         DESCRIPTION
24             "This object shows the configured number of Uplink CIDs the
25             BS can support per SS."
26         ::= { wmanIfBsCapabilitiesConfigEntry 1 }
27
28
29
30     wmanIfBsCapCfgArqSupport OBJECT-TYPE
31         SYNTAX      WmanIfArqSupportType
32         MAX-ACCESS  read-write
33         STATUS      current
34         DESCRIPTION
35             "This object indicates whether the BS is configured to
36             support ARQ."
37         ::= { wmanIfBsCapabilitiesConfigEntry 2 }
38
39
40
41     wmanIfBsCapCfgDsxFowControl OBJECT-TYPE
42         SYNTAX      WmanIfMaxDsxFowType
43         MAX-ACCESS  read-write
44         STATUS      current
45         DESCRIPTION
46             "This object specifies the configured maximum number of
47             concurrent DSA, DSC, or DSD transactions that BS allows
48             each SS to have outstanding."
49         DEFVAL      { 0 }
50         ::= { wmanIfBsCapabilitiesConfigEntry 3 }
51
52
53
54     wmanIfBsCapCfgMacCrcSupport OBJECT-TYPE
55         SYNTAX      WmanIfMacCrcSupport
56         MAX-ACCESS  read-write
57         STATUS      current
58         DESCRIPTION
59             "This object indicates whether BS is configured to support
60             MAC level CRC."
61         DEFVAL      { macCrcSupport }
62         ::= { wmanIfBsCapabilitiesConfigEntry 4 }
63
64
65

```

```

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

```

```

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

```

```

1      MAX-ACCESS  read-write
2      STATUS      current
3      DESCRIPTION
4          "Specifies configured capabilities for construction and
5           transmission of MAC PDUs. The usage is defined by
6           WmanIfPduConstruction."
7      ::= { wmanIfBsCapabilitiesConfigEntry 15 }
8
9
10
11  wmanIfBsCapCfgTtgTransitionGap OBJECT-TYPE
12      SYNTAX      WmanIfSsTransitionGap
13      UNITS        "us"
14      MAX-ACCESS  read-write
15      STATUS      current
16      DESCRIPTION
17          "This field indicates the configured transition speed
18           SSTTG for TDD and H-FDD SSs. The usage is defined by
19           WmanIfSsTransitionGap."
20      ::= { wmanIfBsCapabilitiesConfigEntry 16 }
21
22
23
24  wmanIfBsCapCfgRtgTransitionGap OBJECT-TYPE
25      SYNTAX      WmanIfSsTransitionGap
26      UNITS        "us"
27      MAX-ACCESS  read-write
28      STATUS      current
29      DESCRIPTION
30          "This field indicates the configured transition speed
31           SSRTG for TDD and H-FDD SSs. The usage is defined by
32           WmanIfSsTransitionGap."
33      ::= { wmanIfBsCapabilitiesConfigEntry 17 }
34
35
36
37  wmanIfBsSsActionsTable OBJECT-TYPE
38      SYNTAX      SEQUENCE OF WmanIfBsSsActionsEntry
39      MAX-ACCESS  not-accessible
40      STATUS      current
41      DESCRIPTION
42          "This table contains all the actions specified for SSs in
43           the standard. The actions are routed down to SS using
44           unsolicited MAC messages: REG-RSP, DREG-REQ and RES-CMD.
45           The table also contains the parameters of the actions in
46           cases where they are specified by the standard."
47      ::= { wmanIfBsCps 5 }
48
49
50
51  wmanIfBsSsActionsEntry OBJECT-TYPE
52      SYNTAX      WmanIfBsSsActionsEntry
53      MAX-ACCESS  not-accessible
54      STATUS      current
55      DESCRIPTION
56          "This table is indexed by wmanIfBsSsActionsMacAddress. The
57           action can be requested for SS in any state not only those
58           registered. However BS will decide whether the action is
59           applicable to the SS based on its current state and execute
60           it or skip it as defined in each action definition."
61      INDEX { wmanIfBsSsActionsMacAddress }
62      ::= { wmanIfBsSsActionsTable 1 }
63
64
65

```

```

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

```

```

1          'Uplink Channel ID Override' parameters.
2          - When set to any other value it should be ignored
3          - When read it should returned actionsAbortSsNoAction"
4
5      REFERENCE
6          "Subclause 11.6, Table 365 in IEEE Std 802.16-2004"
7      ::= { wmanIfBsSsActionsEntry 3 }
8
9
10     wmanIfBsSsActionsOverrideDnFreq OBJECT-TYPE
11         SYNTAX      Unsigned32
12         UNITS        "kHz"
13         MAX-ACCESS   read-create
14         STATUS       current
15         DESCRIPTION
16             "This object is used as a parmeter of the AbortSs action
17             with the code actionAbortSsParams. It is used for licensed
18             bands only. It defines the Center frequency, in kHz, of
19             new downlink channel where the SS should redo initial
20             ranging."
21
22     REFERENCE
23         "Subclause 11.6, Table 365 in IEEE Std 802.16-2004"
24     ::= { wmanIfBsSsActionsEntry 4 }
25
26
27     wmanIfBsSsActionsOverrideChannelId OBJECT-TYPE
28         SYNTAX      INTEGER (0..199)
29         MAX-ACCESS   read-create
30         STATUS       current
31         DESCRIPTION
32             "This object is used as a parmeter of the AbortSs action
33             with the code actionAbortSsParams. It is coded as follows:
34             - Licensed bands: The identifier of the uplink channel
35             with which the SS is to redo initial ranging (not used
36             with PHYs without channelized uplinks).
37             - License-exempt bands: The Channel Nr (see 8.5.1) where
38             the SS should redo initial ranging."
39
40     REFERENCE
41         "Subclause 11.6, Table 365 in IEEE Std 802.16-2004"
42     ::= { wmanIfBsSsActionsEntry 5 }
43
44
45
46     wmanIfBsSsActionsDeReRegSs OBJECT-TYPE
47         SYNTAX      INTEGER {actionsDeReRegSsNoAction(0),
48                             actionsDeReRegSs(1)}
49         MAX-ACCESS   read-create
50         STATUS       current
51         DESCRIPTION
52             "This object should be implemented as follows:
53             - When set to actionsDeReRegSs value, instructs BS to
54             send DREG-CMD to SS with specified action code
55             - When set to value different than actionsDeReRegSs it
56             should be ignored
57             - When read it should return actionsDeReRegSsNoAction
58             The DREG-CMD message shall be transmitted by the BS on an
59             SS Basic CID to force the SS to change its access state.
60             Upon receiving a DREG-CMD, the SS shall take the action
61             indicated by the action code defined by
62
63
64
65

```

```

1          wmanIfBsSsActionsDeReRegSsCode."
2  REFERENCE
3          "Subclause 6.3.2.3.26 in IEEE Std 802.16-2004"
4          ::= { wmanIfBsSsActionsEntry 6 }
5
6
7  wmanIfBsSsActionsDeReRegSsCode OBJECT-TYPE
8      SYNTAX      INTEGER {actionsDeReRegSsCodeChangeChan(0),
9                      actionsDeReRegSsCodeNoTransmit(1),
10                     actionsDeReRegSsCodeLtdTransmit(2),
11                     actionsDeReRegSsCodeResume(3)}
12
13  MAX-ACCESS    read-create
14  STATUS        current
15  DESCRIPTION
16      "This object defines the action code for
17      wmanIfBsSsActionsDeReRegSs action. The codes are defined
18      as follows:
19      actionsDeReRegSsCodeChangeChan - SS shall leave the
20      current channel and attempt to access another channel.
21      actionsDeReRegSsCodeNoTransmit - SS shall listen to the
22      current channel but shall not transmit until an
23      RES-CMD message or DREG_CMD with an Action Code that
24      allows transmission is received.
25      actionsDeReRegSsCodeLtdTransmit - SS shall listen to the
26      current channel but only transmit on the Basic,
27      Primary Management and 2nd Management Connections.
28      actionsDeReRegSsCodeResume - SS shall return to normal
29      operation and may transmit on any of its active
30      connections."
31
32  REFERENCE
33      "Subclause 6.3.2.3.26, Table 55 in IEEE Std 802.16-2004"
34      ::= { wmanIfBsSsActionsEntry 7 }
35
36
37  wmanIfBsSsActionsRowStatus OBJECT-TYPE
38      SYNTAX      RowStatus
39      MAX-ACCESS  read-create
40      STATUS      current
41      DESCRIPTION
42          "This object is used to ensure that the write operation to
43          multiple columns is guaranteed to be treated as atomic
44          operation by agent."
45          ::= { wmanIfBsSsActionsEntry 8 }
46
47
48  --
49  -- Base station PKM group
50  -- wmanIfBsPkmObjects contain the Base Station Privacy Sublayer objects
51  --
52  wmanIfBsPkmObjects OBJECT IDENTIFIER ::= { wmanIfBsObjects 3 }
53
54
55  --
56  -- Table wmanIfBsPkmBaseTable
57  --
58  wmanIfBsPkmBaseTable OBJECT-TYPE
59      SYNTAX      SEQUENCE OF WmanIfBsPkmBaseEntry
60      MAX-ACCESS  not-accessible
61

```

```

1      STATUS      current
2      DESCRIPTION
3          "This table describes the basic PKM attributes of each Base
4          Station wireless interface."
5      ::= { wmanIfBsPkmObjects 1 }
6
7
8      wmanIfBsPkmBaseEntry OBJECT-TYPE
9          SYNTAX      WmanIfBsPkmBaseEntry
10         MAX-ACCESS  not-accessible
11         STATUS      current
12         DESCRIPTION
13             "Each entry contains objects describing attributes of one
14             BS wireless interface."
15         INDEX        { ifIndex }
16         ::= { wmanIfBsPkmBaseTable 1 }
17
18
19
20     WmanIfBsPkmBaseEntry ::= SEQUENCE {
21         wmanIfBsPkmDefaultAuthLifetime      Integer32,
22         wmanIfBsPkmDefaultTekLifetime        Integer32,
23         wmanIfBsPkmDefaultSelfSigManufCertTrust INTEGER,
24         wmanIfBsPkmCheckCertValidityPeriods  TruthValue,
25         wmanIfBsPkmAuthentInfos              Counter32,
26         wmanIfBsPkmAuthRequests              Counter32,
27         wmanIfBsPkmAuthReplies               Counter32,
28         wmanIfBsPkmAuthRejects               Counter32,
29         wmanIfBsPkmAuthInvalids              Counter32}
30
31
32
33     wmanIfBsPkmDefaultAuthLifetime OBJECT-TYPE
34         SYNTAX      Integer32 (86400..604800)
35         UNITS        "seconds"
36         MAX-ACCESS  read-write
37         STATUS      current
38         DESCRIPTION
39             "The value of this object is the default lifetime, in
40             seconds, the BS assigns to a new authorization key."
41         REFERENCE
42             "Table 341 in IEEE Std 802.16-2004"
43         DEFVAL       { 604800 }
44         ::= { wmanIfBsPkmBaseEntry 1 }
45
46
47
48
49     wmanIfBsPkmDefaultTekLifetime OBJECT-TYPE
50         SYNTAX      Integer32 (1800..604800)
51         UNITS        "seconds"
52         MAX-ACCESS  read-write
53         STATUS      current
54         DESCRIPTION
55             "The value of this object is the default lifetime, in
56             seconds, the BS assigns to a new Traffic Encryption
57             Key (TEK) ."
58         REFERENCE
59             "Table 341 in IEEE Std 802.16-2004"
60         DEFVAL       { 43200 }
61         ::= { wmanIfBsPkmBaseEntry 2 }
62
63
64
65

```

```

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

```

```

1  wmanIfBsPkmAuthRejects 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          transmitted an Authorization Reject message to any SS."
8      ::= { wmanIfBsPkmBaseEntry 8 }
9
10
11  wmanIfBsPkmAuthInvalids 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 Invalid message to any SS."
18      ::= { wmanIfBsPkmBaseEntry 9 }
19
20
21  --
22  -- Table wmanIfBsSsPkmAuthTable
23  --
24  wmanIfBsSsPkmAuthTable OBJECT-TYPE
25      SYNTAX          SEQUENCE OF WmanIfBsSsPkmAuthEntry
26      MAX-ACCESS      not-accessible
27      STATUS          current
28      DESCRIPTION
29          "This table describes PKM attributes related
30          to the authorization for each SS. The BS maintains one
31          Primary Security Association with each Baseline
32          Privacy-enabled SS on each BS wireless interface."
33      ::= { wmanIfBsPkmObjects 2 }
34
35  wmanIfBsSsPkmAuthEntry OBJECT-TYPE
36      SYNTAX          WmanIfBsSsPkmAuthEntry
37      MAX-ACCESS      not-accessible
38      STATUS          current
39      DESCRIPTION
40          "The BS MUST create one entry per SS per wireless
41          interface, based on the receipt of an Authorization
42          Request message and MUST not delete the entry before
43          the SS authorization permanently expires."
44      INDEX          { ifIndex, wmanIfBsSsPkmAuthMacAddress }
45      ::= { wmanIfBsSsPkmAuthTable 1 }
46
47  WmanIfBsSsPkmAuthEntry ::= SEQUENCE {
48      wmanIfBsSsPkmAuthMacAddress      MacAddress,
49      wmanIfBsSsPkmAuthKeySequenceNumber Integer32,
50      wmanIfBsSsPkmAuthExpiresOld      DateAndTime,
51      wmanIfBsSsPkmAuthExpiresNew      DateAndTime,
52      wmanIfBsSsPkmAuthLifetime        Integer32,
53      wmanIfBsSsPkmAuthReset           INTEGER,
54      wmanIfBsSsPkmAuthInfos           Counter64,
55      wmanIfBsSsPkmAuthRequests        Counter64,
56      wmanIfBsSsPkmAuthReplies         Counter64,

```

```

1          wmanIfBsSsPkmAuthRejects          Counter64,
2          wmanIfBsSsPkmAuthInvalids          Counter64,
3          wmanIfBsSsPkmAuthRejectErrorCode    INTEGER,
4          wmanIfBsSsPkmAuthRejectErrorString  SnmpAdminString,
5          wmanIfBsSsPkmAuthInvalidErrorCode    INTEGER,
6          wmanIfBsSsPkmAuthInvalidErrorString SnmpAdminString,
7          wmanIfBsSsPkmAuthPrimarySAid        INTEGER,
8          wmanIfBsSsPkmAuthValidStatus        INTEGER}
9
10
11
12 wmanIfBsSsPkmAuthMacAddress OBJECT-TYPE
13     SYNTAX      MacAddress
14     MAX-ACCESS  not-accessible
15     STATUS      current
16     DESCRIPTION
17         "The value of this object is the physical address of the SS
18         to which the authorization association applies."
19     ::= { wmanIfBsSsPkmAuthEntry 1 }
20
21
22
23 wmanIfBsSsPkmAuthKeySequenceNumber OBJECT-TYPE
24     SYNTAX      Integer32 (0..15)
25     MAX-ACCESS  read-only
26     STATUS      current
27     DESCRIPTION
28         "The value of this object is the most recent authorization
29         key sequence number for this SS."
30     ::= { wmanIfBsSsPkmAuthEntry 2 }
31
32
33
34 wmanIfBsSsPkmAuthExpiresOld OBJECT-TYPE
35     SYNTAX      DateAndTime
36     MAX-ACCESS  read-only
37     STATUS      current
38     DESCRIPTION
39         "The value of this object is the actual clock time for
40         expiration of the immediate predecessor of the most recent
41         authorization key for this FSM. If this FSM has only one
42         authorization key, then the value is the time of activation
43         of this FSM."
44     ::= { wmanIfBsSsPkmAuthEntry 3 }
45
46
47
48 wmanIfBsSsPkmAuthExpiresNew OBJECT-TYPE
49     SYNTAX      DateAndTime
50     MAX-ACCESS  read-only
51     STATUS      current
52     DESCRIPTION
53         "The value of this object is the actual clock time for
54         expiration of the most recent authorization key for this
55         FSM"
56     ::= { wmanIfBsSsPkmAuthEntry 4 }
57
58
59
60 wmanIfBsSsPkmAuthLifetime OBJECT-TYPE
61     SYNTAX      Integer32 (86400..6048000)
62     UNITS        "seconds"
63     MAX-ACCESS  read-only
64     STATUS      current
65

```

```

1      DESCRIPTION
2          "The value of this object is the lifetime, in seconds, the
3          BS assigns to an authorization key for this SS."
4
5      REFERENCE
6          "Table 341 in IEEE Std 802.16-2004"
7
8      DEFVAL      { 604800 }
9      ::= { wmanIfBsSsPkmAuthEntry 5 }
10
11  wmanIfBsSsPkmAuthReset OBJECT-TYPE
12      SYNTAX      INTEGER {noResetRequested(1),
13                          invalidateAuth(2),
14                          sendAuthInvalid(3),
15                          invalidateTeks(4)}
16
17      MAX-ACCESS  read-write
18      STATUS      current
19      DESCRIPTION
20          "Setting this object to invalidateAuth(2) causes the BS to
21          invalidate the current SS authorization key(s), but not to
22          transmit an Authorization Invalid message nor to invalidate
23          unicast TEKs. Setting this object to sendAuthInvalid(3)
24          causes the BS to invalidate the current SS authorization
25          key(s), and to transmit an Authorization Invalid message to
26          the SS, but not to invalidate unicast TEKs. Setting this
27          object to invalidateTeks(4) causes the BS to invalidate the
28          current SS authorization key(s), to transmit an
29          Authorization Invalid message to the SS, and to
30          invalidate all unicast TEKs associated with this SS
31          authorization. Reading this object returns the
32          most-recently-set value of this object, or returns
33          noResetRequested(1) if the object has not been set since
34          the last BS reboot."
35      ::= { wmanIfBsSsPkmAuthEntry 6 }
36
37
38
39
40
41  wmanIfBsSsPkmAuthInfos OBJECT-TYPE
42      SYNTAX      Counter64
43      MAX-ACCESS  read-only
44      STATUS      current
45      DESCRIPTION
46          "The value of this object is the count of times the BS has
47          received an Authentication Information message from this
48          SS."
49      ::= { wmanIfBsSsPkmAuthEntry 7 }
50
51
52
53  wmanIfBsSsPkmAuthRequests 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      ::= { wmanIfBsSsPkmAuthEntry 8 }
61
62
63  wmanIfBsSsPkmAuthReplies OBJECT-TYPE
64      SYNTAX      Counter64
65

```

```

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

```

```

1          unsolicited(3),
2          invalidKeySequence(4),
3          keyRequestAuthenticationFailure(5)}
4
5      MAX-ACCESS    read-only
6      STATUS        current
7      DESCRIPTION
8          "The value of this object is the enumerated description of
9          the Error-Code in most recent Authorization Invalid message
10         transmitted to the SS."
11
12     REFERENCE
13         "IEEE Std 802.16-2004; Table 371"
14     ::= { wmanIfBsSsPkmAuthEntry 14 }
15
16 wmanIfBsSsPkmAuthInvalidErrorString OBJECT-TYPE
17     SYNTAX          SnmpAdminString (SIZE (0..128))
18     MAX-ACCESS      read-only
19     STATUS           current
20     DESCRIPTION
21         "The value of this object is the Display-String in most
22         recent Authorization Invalid message transmitted to the SS.
23         This is a zero length string if no Authorization Invalid
24         message has been transmitted to the SS."
25     ::= { wmanIfBsSsPkmAuthEntry 15 }
26
27
28
29
30 wmanIfBsSsPkmAuthPrimarySAId 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     ::= { wmanIfBsSsPkmAuthEntry 16 }
40
41
42
43 wmanIfBsSsPkmAuthValidStatus OBJECT-TYPE
44     SYNTAX          INTEGER {unknown (0),
45                          validSsChained (1),
46                          validSsTrusted (2),
47                          invalidSsUntrusted (3),
48                          invalidCAUntrusted (4),
49                          invalidSsOther (5),
50                          invalidCAOther (6)}
51
52     MAX-ACCESS      read-only
53     STATUS           current
54     DESCRIPTION
55         "Contains the reason why a SS's certificate is deemed valid
56         or invalid. Return unknown if the SS is running PKM mode.
57         ValidSsChained means the certificate is valid because it
58         chains to a valid certificate. ValidSsTrusted means the
59         certificate is valid because it has been provisioned to be
60         trusted. InvalidSsUntrusted means the certificate is
61         invalid because it has been provisioned to be untrusted.
62         InvalidCAUntrusted means the certificate is invalid
63
64
65

```

```

1         because it chains to an untrusted certificate.
2         InvalidSsOther and InvalidCAOther refer to errors in
3         parsing, validity periods, etc, which are attributable to
4         the SS certificate or its chain respectively."
5     ::= { wmanIfBsSsPkmAuthEntry 17 }
6
7
8     --
9     -- Table wmanIfBsPkmTekTable
10    --
11    wmanIfBsPkmTekTable OBJECT-TYPE
12        SYNTAX      SEQUENCE OF WmanIfBsPkmTekEntry
13        MAX-ACCESS   not-accessible
14        STATUS       current
15        DESCRIPTION
16            "This table describes the attributes of each Traffic
17             Encryption Key (TEK) association. The BS maintains one TEK
18             association per SAID on each BS wireless interface."
19        ::= { wmanIfBsPkmObjects 3 }
20
21
22    wmanIfBsPkmTekEntry OBJECT-TYPE
23        SYNTAX      WmanIfBsPkmTekEntry
24        MAX-ACCESS   not-accessible
25        STATUS       current
26        DESCRIPTION
27            "Each entry contains objects describing attributes of one
28             TEK association on a particular BS wireless interface. The
29             BS MUST create one entry per SAID per wireless interface,
30             based on the receipt of a Key Request message, and MUST not
31             delete the entry before the SS authorization for the SAID
32             permanently expires."
33        INDEX       { ifIndex, wmanIfBsPkmTekSAId }
34        ::= { wmanIfBsPkmTekTable 1 }
35
36
37    WmanIfBsPkmTekEntry ::= SEQUENCE {
38        wmanIfBsPkmTekSAId                INTEGER,
39        wmanIfBsPkmTekSAType               INTEGER,
40        wmanIfBsPkmTekDataEncryptAlg       WmanIfDataEncryptAlgId,
41        wmanIfBsPkmTekDataAuthAlg          WmanIfDataAuthAlgId,
42        wmanIfBsPkmTekEncryptAlg           WmanIfTekEncryptAlgId,
43        wmanIfBsPkmTekLifetime              Integer32,
44        wmanIfBsPkmTekKeySequenceNumber    Integer32,
45        wmanIfBsPkmTekExpiresOld            DateAndTime,
46        wmanIfBsPkmTekExpiresNew           DateAndTime,
47        wmanIfBsPkmTekReset                 TruthValue,
48        wmanIfBsPkmKeyRequests              Counter32,
49        wmanIfBsPkmKeyReplies               Counter32,
50        wmanIfBsPkmKeyRejects               Counter32,
51        wmanIfBsPkmTekInvalids              Counter32,
52        wmanIfBsPkmKeyRejectErrorCode        INTEGER,
53        wmanIfBsPkmKeyRejectErrorString     SnmpAdminString,
54        wmanIfBsPkmTekInvalidErrorCode      INTEGER,
55        wmanIfBsPkmTekInvalidErrorString    SnmpAdminString}
56
57    wmanIfBsPkmTekSAId OBJECT-TYPE
58
59
60
61
62
63
64
65

```

```

1      SYNTAX      INTEGER (0..65535)
2      MAX-ACCESS  not-accessible
3      STATUS      current
4      DESCRIPTION
5          "The value of this object is the Security Association
6          ID (SAID)."
```

REFERENCE

```

9          "IEEE Std 802.16-2004; 11.9.7"
10         ::= { wmanIfBsPkmTekEntry 1 }
```

wmanIfBsPkmTekSAType OBJECT-TYPE

```

14      SYNTAX      INTEGER {primarySA(0),
15                          staticSA(1),
16                          dynamicSA(2)}
17      MAX-ACCESS  read-only
18      STATUS      current
19      DESCRIPTION
20          "The value of this object is the type of security
21          association. Dynamic does not apply to SSs running in PKM
22          mode."
```

REFERENCE

```

26          "IEEE Std 802.16-2004; subclause 11.9.18"
27         ::= { wmanIfBsPkmTekEntry 2 }
```

wmanIfBsPkmTekDataEncryptAlg OBJECT-TYPE

```

31      SYNTAX      WmanIfDataEncryptAlgId
32      MAX-ACCESS  read-only
33      STATUS      current
34      DESCRIPTION
35          "The value of this object is the data encryption algorithm
36          being utilized."
```

REFERENCE

```

39          "Table 375, IEEE Std 802.16-2004"
40         ::= { wmanIfBsPkmTekEntry 3 }
```

wmanIfBsPkmTekDataAuthentAlg OBJECT-TYPE

```

44      SYNTAX      WmanIfDataAuthAlgId
45      MAX-ACCESS  read-only
46      STATUS      current
47      DESCRIPTION
48          "The value of this object is the data authentication
49          algorithm being utilized."
```

REFERENCE

```

52          "Table 376, IEEE Std 802.16-2004"
53         ::= { wmanIfBsPkmTekEntry 4 }
```

wmanIfBsPkmTekEncryptAlg OBJECT-TYPE

```

57      SYNTAX      WmanIfTekEncryptAlgId
58      MAX-ACCESS  read-only
59      STATUS      current
60      DESCRIPTION
61          "The value of this object is the TEK key encryption
62          algorithm being utilized."
```

REFERENCE

```

65
```

```

1         "Table 377, IEEE Std 802.16-2004"
2         ::= { wmanIfBsPkmTekEntry 5 }
3
4
5 wmanIfBsPkmTekLifetime OBJECT-TYPE
6     SYNTAX      Integer32 (1800..604800)
7     UNITS       "seconds"
8     MAX-ACCESS  read-only
9     STATUS      current
10    DESCRIPTION
11        "The value of this object is the lifetime, in seconds, the
12         BS assigns to keys for this TEK association."
13    REFERENCE
14        "Table 341 in IEEE Std 802.16-2004"
15    DEFVAL      { 43200 }
16    ::= { wmanIfBsPkmTekEntry 6 }
17
18
19
20 wmanIfBsPkmTekKeySequenceNumber OBJECT-TYPE
21     SYNTAX      Integer32 (0..3)
22     MAX-ACCESS  read-only
23     STATUS      current
24     DESCRIPTION
25        "The value of this object is the most recent TEK key
26         sequence number for this SAID."
27    REFERENCE
28        "IEEE Std 802.16-2004; subclause 11.9.5"
29    ::= { wmanIfBsPkmTekEntry 7 }
30
31
32
33 wmanIfBsPkmTekExpiresOld OBJECT-TYPE
34     SYNTAX      DateAndTime
35     MAX-ACCESS  read-only
36     STATUS      current
37     DESCRIPTION
38        "The value of this object is the actual clock time for
39         expiration of the immediate predecessor of the most recent
40         TEK for this FSM. If this FSM has only one TEK, then the
41         value is the time of activation of this FSM."
42    ::= { wmanIfBsPkmTekEntry 8 }
43
44
45
46 wmanIfBsPkmTekExpiresNew OBJECT-TYPE
47     SYNTAX      DateAndTime
48     MAX-ACCESS  read-only
49     STATUS      current
50     DESCRIPTION
51        "The value of this object is the actual clock time for
52         expiration of the most recent TEK for this FSM."
53    ::= { wmanIfBsPkmTekEntry 9 }
54
55
56
57 wmanIfBsPkmTekReset OBJECT-TYPE
58     SYNTAX      TruthValue
59     MAX-ACCESS  read-write
60     STATUS      current
61     DESCRIPTION
62        "Setting this object to TRUE causes the BS to invalidate
63         the current active TEK(s) (plural due to key transition
64
65

```

```

1         periods), and to generate a new TEK for the associated
2         SAID; the BS MAY also generate an unsolicited TEK Invalid
3         message, to optimize the TEK synchronization between the BS
4         and the SS. Reading this object always returns FALSE."
5         ::= { wmanIfBsPkmTekEntry 10 }
6
7
8     wmanIfBsPkmKeyRequests OBJECT-TYPE
9         SYNTAX      Counter32
10        MAX-ACCESS  read-only
11        STATUS      current
12        DESCRIPTION
13            "The value of this object is the count of times the BS has
14             received a Key Request message."
15        ::= { wmanIfBsPkmTekEntry 11 }
16
17
18     wmanIfBsPkmKeyReplies OBJECT-TYPE
19         SYNTAX      Counter32
20        MAX-ACCESS  read-only
21        STATUS      current
22        DESCRIPTION
23            "The value of this object is the count of times the BS has
24             transmitted a Key Reply message."
25        ::= { wmanIfBsPkmTekEntry 12 }
26
27
28     wmanIfBsPkmKeyRejects OBJECT-TYPE
29         SYNTAX      Counter32
30        MAX-ACCESS  read-only
31        STATUS      current
32        DESCRIPTION
33            "The value of this object is the count of times the BS has
34             transmitted a Key Reject message."
35        ::= { wmanIfBsPkmTekEntry 13 }
36
37
38     wmanIfBsPkmTekInvalids 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 BS has
44             transmitted a TEK Invalid message."
45        ::= { wmanIfBsPkmTekEntry 14 }
46
47
48     wmanIfBsPkmKeyRejectErrorCode OBJECT-TYPE
49         SYNTAX      INTEGER {noInformation(0),
50                             unauthorizedSaid(2)}
51        MAX-ACCESS  read-only
52        STATUS      current
53        DESCRIPTION
54            "The value of this object is the enumerated; description of
55             the Error-Code in the most recent Key Reject message sent
56             in response to a Key Request for this SAID."
57        REFERENCE
58            "IEEE Std 802.16-2004; Table 371"
59        ::= { wmanIfBsPkmTekEntry 15 }
60
61
62
63
64
65

```

```

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

```

```

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

```

```

1  WmanIfBsThresholdConfigEntry ::= SEQUENCE {
2      wmanIfBsRssiLowThreshold      Integer32,
3      wmanIfBsRssiHighThreshold     Integer32}
4
5
6  wmanIfBsRssiLowThreshold OBJECT-TYPE
7      SYNTAX      Integer32
8      UNITS       "dBm"
9      MAX-ACCESS  read-write
10     STATUS      current
11     DESCRIPTION
12         "Low threshold for generating the RSSI alarm."
13     ::= { wmanIfBsThresholdConfigEntry 1 }
14
15
16  wmanIfBsRssiHighThreshold OBJECT-TYPE
17     SYNTAX      Integer32
18     UNITS       "dBm"
19     MAX-ACCESS  read-write
20     STATUS      current
21     DESCRIPTION
22         "High threshold for clearing the RSSI alarm."
23     ::= { wmanIfBsThresholdConfigEntry 2 }
24
25
26
27  --
28  -- Subscriber station Notification Objects Definitions
29  --
30
31  wmanIfBsSsNotificationObjectsTable OBJECT-TYPE
32     SYNTAX      SEQUENCE OF WmanIfBsSsNotificationObjectsEntry
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     ::= { wmanIfBsTrapDefinitions 1 }
39
40
41
42  wmanIfBsSsNotificationObjectsEntry OBJECT-TYPE
43     SYNTAX      WmanIfBsSsNotificationObjectsEntry
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 double indexed by
49         wmanIfBsSsNotificationMacAddr and ifIndex for BS sector."
50     INDEX      { ifIndex, wmanIfBsSsNotificationMacAddr }
51     ::= { wmanIfBsSsNotificationObjectsTable 1 }
52
53
54
55  WmanIfBsSsNotificationObjectsEntry ::= SEQUENCE {
56      wmanIfBsSsNotificationMacAddr  MacAddress,
57      wmanIfBsSsStatusValue          INTEGER,
58      wmanIfBsSsStatusInfo           OCTET STRING,
59      wmanIfBsDynamicServiceType     INTEGER,
60      wmanIfBsDynamicServiceFailReason OCTET STRING,
61      wmanIfBsSsRssiStatus           INTEGER,
62      wmanIfBsSsRssiStatusInfo       OCTET STRING,
63      wmanIfBsSsRegisterStatus       INTEGER}
64
65

```

```

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

```

```

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

```

```

1      DESCRIPTION
2          "An event to report the failure of a dynamic service
3          operation happened during the dynamic services process
4          and detected in the Bs side."
5      ::= { wmanIfBsTrapPrefix 2 }
6
7
8      wmanIfBsSsRssiStatusChangeTrap NOTIFICATION-TYPE
9          OBJECTS      {ifIndex,
10                      wmanIfBsSsNotificationMacAddr,
11                      wmanIfBsSsRssiStatus,
12                      wmanIfBsSsRssiStatusInfo}
13          STATUS      current
14          DESCRIPTION
15              "An event to report that the uplink RSSI is below
16              wmanIfBsLowRssiThreshold, or above
17              wmanIfBsHighRssiThreshold after restore."
18          ::= { wmanIfBsTrapPrefix 3 }
19
20
21      wmanIfBsSsPkmFailTrap NOTIFICATION-TYPE
22          OBJECTS      {wmanIfBsSsNotificationMacAddr}
23          STATUS      current
24          DESCRIPTION
25              "An event to report the failure of a Pkm operation."
26          ::= { wmanIfBsTrapPrefix 4 }
27
28
29      wmanIfBsSsRegistrerTrap NOTIFICATION-TYPE
30          OBJECTS      {wmanIfBsSsNotificationMacAddr,
31                      wmanIfBsSsRegisterStatus}
32          STATUS      current
33          DESCRIPTION
34              "An event to report SS registration status."
35          ::= { wmanIfBsTrapPrefix 5 }
36
37
38      --
39      -- Base station PHY Group
40      --
41      wmanIfBsPhy OBJECT IDENTIFIER ::= { wmanIfBsObjects 6 }
42
43
44      --
45      -- BS OFDM PHY objects
46      --
47      wmanIfBsOfdmPhy OBJECT IDENTIFIER ::= { wmanIfBsPhy 1 }
48
49
50      wmanIfBsOfdmUplinkChannelTable OBJECT-TYPE
51          SYNTAX      SEQUENCE OF WmanIfBsOfdmUplinkChannelEntry
52          MAX-ACCESS  not-accessible
53          STATUS      current
54          DESCRIPTION
55              "This table contains UCD channel attributes, defining the
56              transmission characteristics of uplink channels"
57          REFERENCE
58              "Table 349 and Table 352, in IEEE Std 802.16-2004"
59          ::= { wmanIfBsOfdmPhy 1 }
60
61
62
63
64
65

```

```

1  wmanIfBsOfdmUplinkChannelEntry OBJECT-TYPE
2      SYNTAX      WmanIfBsOfdmUplinkChannelEntry
3      MAX-ACCESS  not-accessible
4      STATUS      current
5      DESCRIPTION
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     ::= { wmanIfBsOfdmUplinkChannelTable 1 }
12
13  WmanIfBsOfdmUplinkChannelEntry ::= SEQUENCE {
14      wmanIfBsOfdmCtBasedResvTimeout      INTEGER,
15      wmanIfBsOfdmBwReqOppSize             INTEGER,
16      wmanIfBsOfdmRangReqOppSize           INTEGER,
17      wmanIfBsOfdmUplinkCenterFreq         Unsigned32,
18      wmanIfBsOfdmNumSubChReqRegionFull    INTEGER,
19      wmanIfBsOfdmNumSymbolsReqRegionFull  INTEGER,
20      wmanIfBsOfdmSubChFocusCtCode         INTEGER,
21      wmanIfBsOfdmUpLinkChannelId          INTEGER}
22
23  wmanIfBsOfdmCtBasedResvTimeout OBJECT-TYPE
24      SYNTAX      INTEGER (1..255)
25      MAX-ACCESS  read-write
26      STATUS      current
27      DESCRIPTION
28          "The number of UL-MAPs to receive before contention-based
29          reservation is attempted again for the same connection."
30      REFERENCE
31          "Table 349, in IEEE Std 802.16-2004"
32      ::= { wmanIfBsOfdmUplinkChannelEntry 1 }
33
34  wmanIfBsOfdmBwReqOppSize OBJECT-TYPE
35      SYNTAX      INTEGER (1..65535)
36      UNITS       "PS"
37      MAX-ACCESS  read-write
38      STATUS      current
39      DESCRIPTION
40          "Size (in units of PS) of PHY payload that SS may use to
41          format and transmit a bandwidth request message in a
42          contention request opportunity. The value includes all
43          PHY overhead as well as allowance for the MAC data the
44          message may hold."
45      REFERENCE
46          "Table 349, in IEEE Std 802.16-2004"
47      ::= { wmanIfBsOfdmUplinkChannelEntry 2 }
48
49  wmanIfBsOfdmRangReqOppSize OBJECT-TYPE
50      SYNTAX      INTEGER (1..65535)
51      UNITS       "PS"
52      MAX-ACCESS  read-write
53      STATUS      current
54      DESCRIPTION

```

```

1         "Size (in units of PS) of PHY payload that SS may use to
2         format and transmit a RNG-REQ message in a contention
3         request opportunity. The value includes all PHY overhead
4         as well as allowance for the MAC data the message may
5         hold and the maximum SS/BS roundtrip propagation delay."
6
7     REFERENCE
8         "Table 349, in IEEE Std 802.16-2004"
9     ::= { wmanIfBsOfdmUplinkChannelEntry 3 }
10
11
12 wmanIfBsOfdmUplinkCenterFreq OBJECT-TYPE
13     SYNTAX      Unsigned32
14     UNITS       "kHz"
15     MAX-ACCESS  read-write
16     STATUS      current
17     DESCRIPTION
18         " Uplink center frequency (kHz)"
19     REFERENCE
20         "Table 349, in IEEE Std 802.16-2004"
21     ::= { wmanIfBsOfdmUplinkChannelEntry 4 }
22
23
24
25 wmanIfBsOfdmNumSubChReqRegionFull OBJECT-TYPE
26     SYNTAX      INTEGER {oneSubchannel(0),
27                        twoSubchannels(1),
28                        fourSubchannels(2),
29                        eightSubchannels(3),
30                        sixteenSubchannels(4)}
31     MAX-ACCESS  read-write
32     STATUS      current
33     DESCRIPTION
34         "Number of subchannels used by each transmit
35         opportunity when REQ Region-Full is allocated in
36         subchannelization region."
37     REFERENCE
38         "Table 352, in IEEE Std 802.16-2004"
39     ::= { wmanIfBsOfdmUplinkChannelEntry 5 }
40
41
42
43
44 wmanIfBsOfdmNumSymbolsReqRegionFull OBJECT-TYPE
45     SYNTAX      INTEGER (0..31)
46     MAX-ACCESS  read-write
47     STATUS      current
48     DESCRIPTION
49         "Number of OFDM symbols used by each transmit
50         opportunity when REQ Region-Full is allocated in
51         subchannelization region."
52     REFERENCE
53         "Table 352, in IEEE Std 802.16-2004"
54     ::= { wmanIfBsOfdmUplinkChannelEntry 6 }
55
56
57
58
59 wmanIfBsOfdmSubChFocusCtCode OBJECT-TYPE
60     SYNTAX      INTEGER (0..8)
61     MAX-ACCESS  read-write
62     STATUS      current
63     DESCRIPTION
64         "Number of contention codes (CSE) that shall only be used to
65

```

```

1         request a subchannelized allocation. Default value 0.
2         Allowed values 0-8."
3
4     REFERENCE
5         "Table 352, in IEEE Std 802.16-2004"
6     DEFVAL      { 0 }
7     ::= { wmanIfBsOfdmUplinkChannelEntry 7 }
8
9
10    wmanIfBsOfdmUpLinkChannelId OBJECT-TYPE
11        SYNTAX      INTEGER (0..255)
12        MAX-ACCESS   read-write
13        STATUS       current
14        DESCRIPTION
15            "The identifier of the uplink channel to which the relevant
16             RNG-RSP or RNG-REQ message refers."
17        REFERENCE
18            "Subclause 6.3.2.3.4. Table 16, in IEEE Std 802.16-2004"
19        ::= { wmanIfBsOfdmUplinkChannelEntry 8 }
20
21
22
23    wmanIfBsOfdmDownlinkChannelTable OBJECT-TYPE
24        SYNTAX      SEQUENCE OF WmanIfBsOfdmDownlinkChannelEntry
25        MAX-ACCESS   not-accessible
26        STATUS       current
27        DESCRIPTION
28            "This table contains DCD channel attributes, defining the
29             transmission characteristics of downlink channels"
30        REFERENCE
31            "Table 358, in IEEE Std 802.16-2004"
32        ::= { wmanIfBsOfdmPhy 2 }
33
34
35
36    wmanIfBsOfdmDownlinkChannelEntry OBJECT-TYPE
37        SYNTAX      WmanIfBsOfdmDownlinkChannelEntry
38        MAX-ACCESS   not-accessible
39        STATUS       current
40        DESCRIPTION
41            "This table provides one row for each downlink channel of
42             multi-sector BS, and is indexed by BS ifIndex. An entry
43             in this table exists for each ifEntry of BS with an
44             ifType of propBWAp2Mp."
45        INDEX { ifIndex }
46        ::= { wmanIfBsOfdmDownlinkChannelTable 1 }
47
48
49
50    WmanIfBsOfdmDownlinkChannelEntry ::= SEQUENCE {
51        wmanIfBsOfdmBsEIRP                INTEGER,
52        wmanIfBsOfdmChannelNumber          WmanIfChannelNumber,
53        wmanIfBsOfdmTTG                    INTEGER,
54        wmanIfBsOfdmRTG                    INTEGER,
55        wmanIfBsOfdmInitRngMaxRSS           INTEGER,
56        wmanIfBsOfdmDownlinkCenterFreq     Unsigned32,
57        wmanIfBsOfdmBsId                   WmanIfBsIdType,
58        wmanIfBsOfdmMacVersion              WmanIfMacVersion,
59        wmanIfBsOfdmFrameDurationCode      INTEGER,
60        wmanIfBsOfdmDownLinkChannelId      INTEGER}
61
62
63
64
65    wmanIfBsOfdmBsEIRP OBJECT-TYPE

```

```

1      SYNTAX      INTEGER (0..65535)
2      UNITS       "dBm"
3      MAX-ACCESS  read-write
4      STATUS      current
5      DESCRIPTION
6          "The EIRP is the equivalent isotropic radiated power of
7           the base station, which is computed for a simple
8           single-antenna transmitter."
9
10     REFERENCE
11         "Table 358, in IEEE Std 802.16-2004"
12     ::= { wmanIfBsOfdmDownlinkChannelEntry 1 }
13
14
15 wmanIfBsOfdmChannelNumber OBJECT-TYPE
16     SYNTAX      WmanIfChannelNumber
17     MAX-ACCESS  read-write
18     STATUS      current
19     DESCRIPTION
20         "Downlink channel number as defined in 8.5.
21          Used for license-exempt operation only."
22     REFERENCE
23         "Table 358, in IEEE Std 802.16-2004"
24     ::= { wmanIfBsOfdmDownlinkChannelEntry 2 }
25
26
27 wmanIfBsOfdmTTG OBJECT-TYPE
28     SYNTAX      INTEGER (0..255)
29     MAX-ACCESS  read-write
30     STATUS      current
31     DESCRIPTION
32         "Transmit / Receive Transition Gap."
33     REFERENCE
34         "Table 358, in IEEE Std 802.16-2004"
35     ::= { wmanIfBsOfdmDownlinkChannelEntry 3 }
36
37
38 wmanIfBsOfdmRTG OBJECT-TYPE
39     SYNTAX      INTEGER (0..255)
40     MAX-ACCESS  read-write
41     STATUS      current
42     DESCRIPTION
43         "Receive / Transmit Transition Gap."
44     REFERENCE
45         "Table 358, in IEEE Std 802.16-2004"
46     ::= { wmanIfBsOfdmDownlinkChannelEntry 4 }
47
48
49 wmanIfBsOfdmInitRngMaxRSS OBJECT-TYPE
50     SYNTAX      INTEGER (0..65535)
51     UNITS       "dBm"
52     MAX-ACCESS  read-write
53     STATUS      current
54     DESCRIPTION
55         "Initial Ranging Max. Received Signal Strength at BS
56          Signed in units of 1 dBm."
57     REFERENCE
58         "Table 358, in IEEE Std 802.16-2004"
59     ::= { wmanIfBsOfdmDownlinkChannelEntry 5 }
60
61
62
63
64
65

```

```

1
2 wmanIfBsOfdmDownlinkCenterFreq OBJECT-TYPE
3     SYNTAX      Unsigned32
4     UNITS       "kHz"
5     MAX-ACCESS  read-write
6     STATUS      current
7     DESCRIPTION
8         "Downlink center frequency (kHz)."
```

REFERENCE

```

11     "Table 358, in IEEE Std 802.16-2004"
12     ::= { wmanIfBsOfdmDownlinkChannelEntry 6 }
```

wmanIfBsOfdmBsId OBJECT-TYPE

```

15     SYNTAX      WmanIfBsIdType
16     MAX-ACCESS  read-write
17     STATUS      current
18     DESCRIPTION
19         "Base station ID."
```

REFERENCE

```

22     "Table 358, in IEEE Std 802.16-2004"
23     ::= { wmanIfBsOfdmDownlinkChannelEntry 7 }
```

wmanIfBsOfdmMacVersion OBJECT-TYPE

```

27     SYNTAX      WmanIfMacVersion
28     MAX-ACCESS  read-write
29     STATUS      current
30     DESCRIPTION
31         "This parameter specifies the version of 802.16 to which
32         the message originator conforms."
```

REFERENCE

```

35     "Table 358, in IEEE Std 802.16-2004"
36     ::= { wmanIfBsOfdmDownlinkChannelEntry 8 }
```

wmanIfBsOfdmFrameDurationCode OBJECT-TYPE

```

40     SYNTAX      INTEGER {duration2dot5ms(0),
41                     duration4ms(1),
42                     duration5ms(2),
43                     duration8ms(3),
44                     duration10ms(4),
45                     duration12dot5ms(5),
46                     duration20ms(6) }
```

MAX-ACCESS read-write

```

50     STATUS      current
51     DESCRIPTION
52         "The duration of the frame. The frame duration code
53         values are specified in Table 230."
```

REFERENCE

```

56     "Subclause 11.4.1, Table 230, in IEEE Std 802.16-2004"
57     ::= { wmanIfBsOfdmDownlinkChannelEntry 9 }
```

wmanIfBsOfdmDownLinkChannelId OBJECT-TYPE

```

61     SYNTAX      INTEGER (0..255)
62     MAX-ACCESS  read-write
63     STATUS      current
```

```

1      DESCRIPTION
2          "The identifier of the downlink channel to which this
3          message refers."
4
5      REFERENCE
6          "Subclause 6.3.2.3.1. Table 15, in IEEE Std 802.16-2004"
7      ::= { wmanIfBsOfdmDownlinkChannelEntry 10 }
8
9
10     wmanIfBsOfdmUcdBurstProfileTable OBJECT-TYPE
11         SYNTAX      SEQUENCE OF WmanIfBsOfdmUcdBurstProfileEntry
12         MAX-ACCESS  not-accessible
13         STATUS      current
14         DESCRIPTION
15             "This table contains UCD burst profiles for each uplink
16             channel"
17         REFERENCE
18             "Table 356, in IEEE Std 802.16-2004"
19         ::= { wmanIfBsOfdmPhy 3 }
20
21
22
23     wmanIfBsOfdmUcdBurstProfileEntry OBJECT-TYPE
24         SYNTAX      WmanIfBsOfdmUcdBurstProfileEntry
25         MAX-ACCESS  not-accessible
26         STATUS      current
27         DESCRIPTION
28             "This table provides one row for each UCD burst profile.
29             This table is double indexed. The primary index is an
30             ifIndex with an ifType of propBWA2Mp. The secondary index
31             is wmanIfBsOfdmUiucIndex."
32         INDEX { ifIndex, wmanIfBsOfdmUiucIndex }
33         ::= { wmanIfBsOfdmUcdBurstProfileTable 1 }
34
35
36
37     WmanIfBsOfdmUcdBurstProfileEntry ::= SEQUENCE {
38         wmanIfBsOfdmUiucIndex          INTEGER,
39         wmanIfBsOfdmUcdFecCodeType     WmanIfOfdmFecCodeType,
40         wmanIfBsOfdmFocusCtPowerBoost  INTEGER,
41         wmanIfBsOfdmUcdTcsEnable       INTEGER,
42         wmanIfBsOfdmUcdBurstProfileRowStatus  RowStatus }
43
44
45     wmanIfBsOfdmUiucIndex OBJECT-TYPE
46         SYNTAX      INTEGER (5 .. 12)
47         MAX-ACCESS  not-accessible
48         STATUS      current
49         DESCRIPTION
50             "The Uplink Interval Usage Code indicates the uplink burst
51             profile in the UCD message, and is used along with ifIndex
52             to identify an entry in the
53             wmanIfBsOfdmUcdBurstProfileTable."
54         REFERENCE
55             "Subclause 8.3.6.3.1, in IEEE Std 802.16-2004"
56         ::= { wmanIfBsOfdmUcdBurstProfileEntry 1 }
57
58
59
60
61     wmanIfBsOfdmUcdFecCodeType OBJECT-TYPE
62         SYNTAX      WmanIfOfdmFecCodeType
63         MAX-ACCESS  read-create
64         STATUS      current
65

```

```

1      DESCRIPTION
2          "Uplink FEC code type and modulation type"
3      REFERENCE
4          "Table 356, in IEEE Std 802.16-2004"
5      ::= { wmanIfBsOfdmUcdBurstProfileEntry 2 }
6
7
8      wmanIfBsOfdmFocusCtPowerBoost OBJECT-TYPE
9          SYNTAX      INTEGER (0 .. 255)
10         MAX-ACCESS  read-create
11         STATUS      current
12         DESCRIPTION
13             "The power boost in dB of focused contention carriers, as
14             described in 8.3.6.3.3."
15         REFERENCE
16             "Table 356, in IEEE Std 802.16-2004"
17         ::= { wmanIfBsOfdmUcdBurstProfileEntry 3 }
18
19
20      wmanIfBsOfdmUcdTcsEnable OBJECT-TYPE
21          SYNTAX      INTEGER {tcsDisabled(0),
22                               tcsEnabled(1)}
23          MAX-ACCESS  read-create
24          STATUS      current
25          DESCRIPTION
26              "This parameter determines the transmission convergence
27              sublayer, as described in 8.1.4.3, can be enabled on a
28              per-burst basis for both uplink and downlink. through
29              DIUC/UIUC messages."
30          REFERENCE
31              "Table 356, in IEEE Std 802.16-2004"
32          ::= { wmanIfBsOfdmUcdBurstProfileEntry 4 }
33
34
35      wmanIfBsOfdmUcdBurstProfileRowStatus OBJECT-TYPE
36          SYNTAX      RowStatus
37          MAX-ACCESS  read-create
38          STATUS      current
39          DESCRIPTION
40              "This object is used to create a new row or modify or
41              delete an existing row in this table.
42
43              If the implementator of this MIB has choosen not
44              to implement 'dynamic assignment' of profiles, this
45              object is not useful and should return noSuchName
46              upon SNMP request."
47          ::= { wmanIfBsOfdmUcdBurstProfileEntry 5 }
48
49
50      wmanIfBsOfdmDcdBurstProfileTable OBJECT-TYPE
51          SYNTAX      SEQUENCE OF WmanIfBsOfdmDcdBurstProfileEntry
52          MAX-ACCESS  not-accessible
53          STATUS      current
54          DESCRIPTION
55              "This table provides one row for each DCD burst profile.
56              This table is double indexed. The primary index is an
57              ifIndex with an ifType of propBWA2Mp. The secondary
58              index is wmanIfBsOfdmDiucIndex."
59

```

```

1      REFERENCE
2          "Table 362, in IEEE Std 802.16-2004"
3      ::= { wmanIfBsOfdmPhy 4 }
4
5
6  wmanIfBsOfdmDcdBurstProfileEntry OBJECT-TYPE
7      SYNTAX      WmanIfBsOfdmDcdBurstProfileEntry
8      MAX-ACCESS  not-accessible
9      STATUS      current
10     DESCRIPTION
11         "This table provides one row for each DCD burst profile.
12         This table is double indexed. The primary index is an
13         ifIndex with an ifType of propBWA2Mp. The secondary index
14         is wmanIfBsOfdmDiucIndex."
15     INDEX { ifIndex, wmanIfBsOfdmDiucIndex }
16     ::= { wmanIfBsOfdmDcdBurstProfileTable 1 }
17
18
19
20  WmanIfBsOfdmDcdBurstProfileEntry ::= SEQUENCE {
21      wmanIfBsOfdmDiucIndex          INTEGER,
22      wmanIfBsOfdmDownlinkFrequency  Unsigned32,
23      wmanIfBsOfdmDcdFecCodeType     WmanIfOfdmFecCodeType,
24      wmanIfBsOfdmDiucMandatoryExitThresh  INTEGER,
25      wmanIfBsOfdmDiucMinEntryThresh  INTEGER,
26      wmanIfBsOfdmTcsEnable           INTEGER,
27      wmanIfBsOfdmDcdBurstProfileRowStatus  RowStatus }
28
29
30
31  wmanIfBsOfdmDiucIndex OBJECT-TYPE
32      SYNTAX      INTEGER (1..11)
33      MAX-ACCESS  not-accessible
34      STATUS      current
35      DESCRIPTION
36          "The Downlink Interval Usage Code indicates the downlink
37          burst profile in the DCD message, and is used along with
38          ifIndex to identify an entry in the
39          wmanIfBsOfdmDcdBurstProfileTable."
40      REFERENCE
41          "Subclause 8.3.6.3.1, in IEEE Std 802.16-2004"
42      ::= { wmanIfBsOfdmDcdBurstProfileEntry 1 }
43
44
45
46  wmanIfBsOfdmDownlinkFrequency OBJECT-TYPE
47      SYNTAX      Unsigned32
48      UNITS       "kHz"
49      MAX-ACCESS  read-create
50      STATUS      current
51      DESCRIPTION
52          "Downlink Frequency (kHz)."
53      REFERENCE
54          "Table 359, in IEEE Std 802.16-2004"
55      ::= { wmanIfBsOfdmDcdBurstProfileEntry 2 }
56
57
58
59  wmanIfBsOfdmDcdFecCodeType OBJECT-TYPE
60      SYNTAX      WmanIfOfdmFecCodeType
61      MAX-ACCESS  read-create
62      STATUS      current
63      DESCRIPTION
64
65

```

```

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

```

```

1
2 wmanIfBsOfdmConfigurationTable OBJECT-TYPE
3     SYNTAX      SEQUENCE OF WmanIfBsOfdmConfigurationEntry
4     MAX-ACCESS  not-accessible
5     STATUS      current
6     DESCRIPTION
7         "This table contains BS configuration objects, specific to
8         OFDM PHY."
9     ::= { wmanIfBsOfdmPhy 5 }
10
11
12
13 wmanIfBsOfdmConfigurationEntry OBJECT-TYPE
14     SYNTAX      WmanIfBsOfdmConfigurationEntry
15     MAX-ACCESS  not-accessible
16     STATUS      current
17     DESCRIPTION
18         "This table is indexed by ifIndex with an ifType of
19         propBWA2Mp."
20     INDEX { ifIndex }
21     ::= { wmanIfBsOfdmConfigurationTable 1 }
22
23
24
25 WmanIfBsOfdmConfigurationEntry ::= SEQUENCE {
26     wmanIfBsOfdmMinReqRegionFullTxOpp      INTEGER,
27     wmanIfBsOfdmMinFocusedCtTxOpp           INTEGER,
28     wmanIfBsOfdmMaxRoundTripDelay           INTEGER,
29     wmanIfBsOfdmRangeAbortTimingThold       INTEGER,
30     wmanIfBsOfdmRangeAbortPowerThold        INTEGER,
31     wmanIfBsOfdmRangeAbortFreqThold         INTEGER,
32     wmanIfBsOfdmDnlkRateId                  INTEGER,
33     wmanIfBsOfdmRatioG                       INTEGER}
34
35
36
37 wmanIfBsOfdmMinReqRegionFullTxOpp OBJECT-TYPE
38     SYNTAX      INTEGER (1..65535)
39     UNITS        "1/sec"
40     MAX-ACCESS  read-write
41     STATUS      current
42     DESCRIPTION
43         "The minimum number of Full bandwidth Req-Region Full
44         Transmit opportunities scheduled in the UL per second."
45     REFERENCE
46         "Subclause 6.3.7.4.3 in IEEE Std 802.16-2004"
47     ::= { wmanIfBsOfdmConfigurationEntry 1 }
48
49
50
51 wmanIfBsOfdmMinFocusedCtTxOpp OBJECT-TYPE
52     SYNTAX      INTEGER (0..65535)
53     UNITS        "1/sec"
54     MAX-ACCESS  read-write
55     STATUS      current
56     DESCRIPTION
57         "The minimum number of focused contention Transmit
58         opportunities scheduled in the UL per second. The value may
59         be 0 if the focused contention is not implemented."
60     REFERENCE
61         "Subclauses 6.3.6.4 and 8.3.7.3.3 in IEEE Std 802.16-2004"
62     ::= { wmanIfBsOfdmConfigurationEntry 2 }
63
64
65

```

1
2 wmanIfBsOfdmMaxRoundTripDelay OBJECT-TYPE

3 SYNTAX INTEGER (1..65535)

4 UNITS "us"

5 MAX-ACCESS read-write

6 STATUS current

7 DESCRIPTION

8 "Maximum supported round trip delay.

9 It is required to limit the cell size."

10 REFERENCE

11 "Subclause 8.3.5.1 in IEEE Std 802.16-2004"

12 ::= { wmanIfBsOfdmConfigurationEntry 3 }

13
14
15
16 wmanIfBsOfdmRangeAbortTimingThold OBJECT-TYPE

17 SYNTAX INTEGER (0..255)

18 UNITS "1/Fs"

19 MAX-ACCESS read-write

20 STATUS current

21 DESCRIPTION

22 "This object defines Tolerable Timing Offset. BS performs

23 Initial Ranging until the SS transmissions are within

24 limits that are deemed tolerable by the BS. If the SS does

25 not transmit within these limits after a number of

26 correction attempts then the BS aborts Initial Ranging."

27 REFERENCE

28 "Figure 63 and Table 365 in IEEE Std 802.16-2004"

29 ::= { wmanIfBsOfdmConfigurationEntry 4 }

30
31
32
33 wmanIfBsOfdmRangeAbortPowerThold OBJECT-TYPE

34 SYNTAX INTEGER (0..255)

35 UNITS "0.25dB"

36 MAX-ACCESS read-write

37 STATUS current

38 DESCRIPTION

39 "This object defines Tolerable Power Offset. BS performs

40 Initial Ranging until the SS transmissions are within

41 limits that are deemed tolerable by the BS. If the SS does

42 not transmit within these limits after a number of

43 correction attempts then the BS aborts Initial Ranging."

44 REFERENCE

45 "Figure 63 and Table 365 in IEEE Std 802.16-2004"

46 ::= { wmanIfBsOfdmConfigurationEntry 5 }

47
48
49
50 wmanIfBsOfdmRangeAbortFreqThold OBJECT-TYPE

51 SYNTAX INTEGER (0..255)

52 UNITS "Hz"

53 MAX-ACCESS read-write

54 STATUS current

55 DESCRIPTION

56 "This object defines Tolerable Frequency Offset. BS performs

57 Initial Ranging until the SS transmissions are within

58 limits that are deemed tolerable by the BS. If the SS does

59 not transmit within these limits after a number of

60 correction attempts then the BS aborts Initial Ranging."

```

1      REFERENCE
2          "Figure 63 and Table 365 in IEEE Std 802.16-2004"
3      ::= { wmanIfBsOfdmConfigurationEntry 6 }
4
5
6  wmanIfBsOfdmDnlkRateId OBJECT-TYPE
7      SYNTAX      INTEGER {dnlkRateIdBpsk1Over2(0),
8                          dnlkRateIdQpsk1Over2(1),
9                          dnlkRateIdQpsk3Over4(2),
10                         dnlkRateId16Qam1Over2(3),
11                         dnlkRateId16Qam3Over4(4),
12                         dnlkRateId64Qam2Over3(5),
13                         dnlkRateId64Qam3Over4(6)}
14
15  MAX-ACCESS    read-write
16  STATUS        current
17  DESCRIPTION
18      "The Rate ID to be used in the first downlink burst
19      immediately following the FCH. The Rate ID encoding is
20      static and cannot be changed during system operation. The
21      change of the Rate ID should be applied on system
22      re-intialisation (e.g. following sector or BS reset)."

```

```

1      MAX-ACCESS    not-accessible
2      STATUS        current
3      DESCRIPTION
4          "This table provides one row for each SS that has been
5          registered in the BS. This table augments the table
6          wmanIfBsRegisteredSsTable."
7      AUGMENTS { wmanIfBsRegisteredSsEntry }
8      ::= { wmanIfBsSsOfdmReqCapabilitiesTable 1 }
9
10
11
12  WmanIfBsSsOfdmReqCapabilitiesEntry ::= SEQUENCE {
13      wmanIfBsSsOfdmReqCapFftSizes          WmanIfOfdmFftSizes,
14      wmanIfBsSsOfdmReqCapSsDemodulator      WmanIfOfdmSsDeModType,
15      wmanIfBsSsOfdmReqCapSsModulator         WmanIfOfdmSsModType,
16      wmanIfBsSsOfdmReqCapFocusedCtSupport    WmanIfOfdmFocusedCt,
17      wmanIfBsSsOfdmReqCapTcSublayerSupport   WmanIfOfdmTcSublayer}
18
19
20  wmanIfBsSsOfdmReqCapFftSizes OBJECT-TYPE
21      SYNTAX      WmanIfOfdmFftSizes
22      MAX-ACCESS   read-only
23      STATUS        current
24      DESCRIPTION
25          "This field indicates the FFT sizes supported by SS.
26          The usage is defined by WmanIfOfdmFftSizes."
27      ::= { wmanIfBsSsOfdmReqCapabilitiesEntry 1 }
28
29
30
31  wmanIfBsSsOfdmReqCapSsDemodulator OBJECT-TYPE
32      SYNTAX      WmanIfOfdmSsDeModType
33      MAX-ACCESS   read-only
34      STATUS        current
35      DESCRIPTION
36          "This field indicates the different demodulator options
37          supported by SS for downlink.
38          The usage is defined by WmanIfOfdmSsDeModType."
39      ::= { wmanIfBsSsOfdmReqCapabilitiesEntry 2 }
40
41
42
43  wmanIfBsSsOfdmReqCapSsModulator OBJECT-TYPE
44      SYNTAX      WmanIfOfdmSsModType
45      MAX-ACCESS   read-only
46      STATUS        current
47      DESCRIPTION
48          "This field indicates the different modulator options
49          supported by SS for uplink.
50          The usage is defined by WmanIfOfdmSsModType."
51      ::= { wmanIfBsSsOfdmReqCapabilitiesEntry 3 }
52
53
54
55  wmanIfBsSsOfdmReqCapFocusedCtSupport OBJECT-TYPE
56      SYNTAX      WmanIfOfdmFocusedCt
57      MAX-ACCESS   read-only
58      STATUS        current
59      DESCRIPTION
60          "This field indicates whether the SS supports Focused
61          Contention. The usage is defined by
62          WmanIfOfdmFocusedCt."
63      ::= { wmanIfBsSsOfdmReqCapabilitiesEntry 4 }
64
65

```

```

1
2 wmanIfBsSsOfdmReqCapTcSublayerSupport OBJECT-TYPE
3     SYNTAX          WmanIfOfdmTcSublayer
4     MAX-ACCESS      read-only
5     STATUS          current
6     DESCRIPTION
7         "This field indicates whether or not the SS supports
8         the TC sublayer. The usage is defined by
9         WmanIfOfdmTcSublayer."
10    ::= { wmanIfBsSsOfdmReqCapabilitiesEntry 5 }
11
12
13
14 wmanIfBsSsOfdmRspCapabilitiesTable OBJECT-TYPE
15     SYNTAX          SEQUENCE OF WmanIfBsSsOfdmRspCapabilitiesEntry
16     MAX-ACCESS      not-accessible
17     STATUS          current
18     DESCRIPTION
19         "This table contains the basic capability information,
20         specific to OFDM Phy, of SSs that have been negotiated
21         and agreed between BS and SS via RNG-REQ/RSP,
22         SBC-REQ/RSP and REG-REQ/RSP messages. This table
23         augments the wmanIfBsRegisteredSsTable."
24     REFERENCE
25         "Subclause 6.3.2.3.7 in IEEE Std 802.16-2004"
26    ::= { wmanIfBsOfdmPhy 7 }
27
28
29
30
31 wmanIfBsSsOfdmRspCapabilitiesEntry OBJECT-TYPE
32     SYNTAX          WmanIfBsSsOfdmRspCapabilitiesEntry
33     MAX-ACCESS      not-accessible
34     STATUS          current
35     DESCRIPTION
36         "This table provides one row for each SS that has been
37         registered in the BS. This table augments the
38         wmanIfBsRegisteredSsTable. "
39     AUGMENTS { wmanIfBsRegisteredSsEntry }
40    ::= { wmanIfBsSsOfdmRspCapabilitiesTable 1 }
41
42
43
44 WmanIfBsSsOfdmRspCapabilitiesEntry ::= SEQUENCE {
45     wmanIfBsSsOfdmRspCapFftSizes          WmanIfOfdmFftSizes,
46     wmanIfBsSsOfdmRspCapSsDemodulator      WmanIfOfdmSsDeModType,
47     wmanIfBsSsOfdmRspCapSsModulator         WmanIfOfdmSsModType,
48     wmanIfBsSsOfdmRspCapFocusedCtSupport    WmanIfOfdmFocusedCt,
49     wmanIfBsSsOfdmRspCapTcSublayerSupport  WmanIfOfdmTcSublayer}
50
51
52
53 wmanIfBsSsOfdmRspCapFftSizes OBJECT-TYPE
54     SYNTAX          WmanIfOfdmFftSizes
55     MAX-ACCESS      read-only
56     STATUS          current
57     DESCRIPTION
58         "This field indicates the FFT sizes negotiated with the
59         SS. The usage is defined by WmanIfOfdmFftSizes."
60    ::= { wmanIfBsSsOfdmRspCapabilitiesEntry 1 }
61
62
63
64 wmanIfBsSsOfdmRspCapSsDemodulator OBJECT-TYPE
65     SYNTAX          WmanIfOfdmSsDeModType

```

```

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

```

```

1
2 wmanIfBsOfdmCapabilitiesEntry OBJECT-TYPE
3     SYNTAX      WmanIfBsOfdmCapabilitiesEntry
4     MAX-ACCESS  not-accessible
5     STATUS      current
6     DESCRIPTION
7         "This table provides one row for each BS sector and is
8         indexed by ifIndex."
9     INDEX { ifIndex }
10    ::= { wmanIfBsOfdmCapabilitiesTable 1 }
11
12
13
14 wmanIfBsOfdmCapabilitiesEntry ::= SEQUENCE {
15     wmanIfBsOfdmCapFftSizes          WmanIfOfdmFftSizes,
16     wmanIfBsOfdmCapSsDemodulator     WmanIfOfdmSsDeModType,
17     wmanIfBsOfdmCapSsModulator       WmanIfOfdmSsModType,
18     wmanIfBsOfdmCapFocusedCtSupport  WmanIfOfdmFocusedCt,
19     wmanIfBsOfdmCapTcSublayerSupport WmanIfOfdmTcSublayer}
20
21
22
23 wmanIfBsOfdmCapFftSizes OBJECT-TYPE
24     SYNTAX      WmanIfOfdmFftSizes
25     MAX-ACCESS  read-only
26     STATUS      current
27     DESCRIPTION
28         "This field indicates the FFT sizes supported by the BS.
29         The usage is defined by WmanIfOfdmCapFftSizes."
30     ::= { wmanIfBsOfdmCapabilitiesEntry 1 }
31
32
33
34 wmanIfBsOfdmCapSsDemodulator OBJECT-TYPE
35     SYNTAX      WmanIfOfdmSsDeModType
36     MAX-ACCESS  read-only
37     STATUS      current
38     DESCRIPTION
39         "This field indicates the different BS demodulator options
40         for uplink supported by the BS. The usage is defined by
41         WmanIfOfdmSsDeModType."
42     ::= { wmanIfBsOfdmCapabilitiesEntry 2 }
43
44
45
46 wmanIfBsOfdmCapSsModulator OBJECT-TYPE
47     SYNTAX      WmanIfOfdmSsModType
48     MAX-ACCESS  read-only
49     STATUS      current
50     DESCRIPTION
51         "This field indicates the different BS modulator options
52         for downlink supported by the BS. The usage is defined by
53         WmanIfOfdmSsModType."
54     ::= { wmanIfBsOfdmCapabilitiesEntry 3 }
55
56
57
58 wmanIfBsOfdmCapFocusedCtSupport OBJECT-TYPE
59     SYNTAX      WmanIfOfdmFocusedCt
60     MAX-ACCESS  read-only
61     STATUS      current
62     DESCRIPTION
63         "This field indicates the BS support for Focused
64         Contention. The usage is defined by
65

```

```

1      WmanIfOfdmFocusedCt."
2      ::= { wmanIfBsOfdmCapabilitiesEntry 4 }
3
4
5  wmanIfBsOfdmCapTcSublayerSupport OBJECT-TYPE
6      SYNTAX      WmanIfOfdmTcSublayer
7      MAX-ACCESS  read-only
8      STATUS      current
9      DESCRIPTION
10         "This field indicates the BS supports for TC sublayer. The
11         usage is defined by WmanIfOfdmTcSublayer."
12      ::= { wmanIfBsOfdmCapabilitiesEntry 5 }
13
14
15  wmanIfBsOfdmCapabilitiesConfigTable OBJECT-TYPE
16      SYNTAX      SEQUENCE OF WmanIfBsOfdmCapabilitiesConfigEntry
17      MAX-ACCESS  not-accessible
18      STATUS      current
19      DESCRIPTION
20         "This table contains the configuration for basic
21         capabilities of BS, specific to OFDM Phy. The table is
22         intended to be used to restrict the Capabilities
23         implemented by BS, for example in order to comply with
24         local regulatory requirements. The BS should use the
25         configuration along with the implemented Capabilities
26         (wmanIfBsOfdmPhyTable) for negotiation of basic
27         capabilities with SS using RNG-RSP, SBC-RSP and REG-RSP
28         messages. The negotiated capabilities are obtained by
29         interSubclause of SS reported capabilities, BS raw
30         capabilities and BS configured capabilities. The objects
31         in the table have read-write access. The rows are created
32         by BS as a copy of wmanIfBsBasicCapabilitiesTable
33         and can be modified by NMS."
34      ::= { wmanIfBsOfdmPhy 9 }
35
36
37  wmanIfBsOfdmCapabilitiesConfigEntry OBJECT-TYPE
38      SYNTAX      WmanIfBsOfdmCapabilitiesConfigEntry
39      MAX-ACCESS  not-accessible
40      STATUS      current
41      DESCRIPTION
42         "This table provides one row for each BS sector and is
43         indexed by ifIndex."
44      INDEX { ifIndex }
45      ::= { wmanIfBsOfdmCapabilitiesConfigTable 1 }
46
47
48  WmanIfBsOfdmCapabilitiesConfigEntry ::= SEQUENCE {
49      wmanIfBsOfdmCapCfgFftSizes      WmanIfOfdmFftSizes,
50      wmanIfBsOfdmCapCfgSsDemodulator WmanIfOfdmSsDeModType,
51      wmanIfBsOfdmCapCfgSsModulator   WmanIfOfdmSsModType,
52      wmanIfBsOfdmCapCfgFocusedCtSupport WmanIfOfdmFocusedCt,
53      wmanIfBsOfdmCapCfgTcSublayerSupport WmanIfOfdmTcSublayer}
54
55  wmanIfBsOfdmCapCfgFftSizes OBJECT-TYPE
56      SYNTAX      WmanIfOfdmFftSizes
57      MAX-ACCESS  read-write
58      STATUS      current
59
60
61
62
63
64
65

```

```

1      DESCRIPTION
2          "This field indicates the FFT sizes support configured for
3          the BS. The usage is defined by
4          WmanIfOfdmCapFftSizes."
5      ::= { wmanIfBsOfdmCapabilitiesConfigEntry 1 }
6
7
8  wmanIfBsOfdmCapCfgSsDemodulator OBJECT-TYPE
9      SYNTAX      WmanIfOfdmSsDeModType
10     MAX-ACCESS  read-write
11     STATUS      current
12     DESCRIPTION
13         "This field indicates the different BS demodulator options
14         configured for uplink. The usage is defined by
15         WmanIfOfdmSsDeModType."
16     ::= { wmanIfBsOfdmCapabilitiesConfigEntry 2 }
17
18
19
20  wmanIfBsOfdmCapCfgSsModulator OBJECT-TYPE
21     SYNTAX      WmanIfOfdmSsModType
22     MAX-ACCESS  read-write
23     STATUS      current
24     DESCRIPTION
25         "This field indicates the different BS modulator options
26         configured for downlink. The usage is defined by
27         WmanIfOfdmSsModType."
28     ::= { wmanIfBsOfdmCapabilitiesConfigEntry 3 }
29
30
31
32  wmanIfBsOfdmCapCfgFocusedCtSupport OBJECT-TYPE
33     SYNTAX      WmanIfOfdmFocusedCt
34     MAX-ACCESS  read-write
35     STATUS      current
36     DESCRIPTION
37         "This field indicates the BS support configured for
38         Focused Contention. The usage is defined by
39         WmanIfOfdmFocusedCt."
40     ::= { wmanIfBsOfdmCapabilitiesConfigEntry 4 }
41
42
43
44  wmanIfBsOfdmCapCfgTcSublayerSupport OBJECT-TYPE
45     SYNTAX      WmanIfOfdmTcSublayer
46     MAX-ACCESS  read-write
47     STATUS      current
48     DESCRIPTION
49         "This field indicates the BS support configured for TC
50         sublayer. The usage is defined by
51         WmanIfOfdmTcSublayer."
52     ::= { wmanIfBsOfdmCapabilitiesConfigEntry 5 }
53
54
55
56  --
57  -- BS OFDMA PHY objects
58  --
59
60  wmanIfBsOfdmaPhy OBJECT IDENTIFIER ::= { wmanIfBsPhy 2 }
61
62
63  wmanIfBsOfdmaUplinkChannelTable OBJECT-TYPE
64     SYNTAX      SEQUENCE OF WmanIfBsOfdmaUplinkChannelEntry
65     MAX-ACCESS  not-accessible

```

```

1      STATUS      current
2      DESCRIPTION
3          "This table contains UCD channel attributes, defining the
4          transmission characteristics of uplink channels"
5      REFERENCE
6          "Table 349 and Table 353, in IEEE Std 802.16-2004"
7      ::= { wmanIfBsOfdmaPhy 1 }
8
9
10
11  wmanIfBsOfdmaUplinkChannelEntry OBJECT-TYPE
12      SYNTAX      WmanIfBsOfdmaUplinkChannelEntry
13      MAX-ACCESS  not-accessible
14      STATUS      current
15      DESCRIPTION
16          "This table provides one row for each uplink channel of
17          multi-sector BS, and is indexed by BS ifIndex. An entry
18          in this table exists for each ifEntry of BS with an
19          ifType of propBWA2Mp."
20      INDEX      { ifIndex }
21      ::= { wmanIfBsOfdmaUplinkChannelTable 1 }
22
23
24
25  WmanIfBsOfdmaUplinkChannelEntry ::= SEQUENCE {
26      wmanIfBsOfdmaCtBasedResvTimeout      INTEGER,
27      wmanIfBsOfdmaBwReqOppSize             INTEGER,
28      wmanIfBsOfdmaRangReqOppSize           INTEGER,
29      wmanIfBsOfdmaUplinkCenterFreq         Unsigned32,
30      wmanIfBsOfdmaInitRngCodes             INTEGER,
31      wmanIfBsOfdmaPeriodicRngCodes         INTEGER,
32      wmanIfBsOfdmaBWReqCodes              INTEGER,
33      wmanIfBsOfdmaPerRngBackoffStart       INTEGER,
34      wmanIfBsOfdmaPerRngBackoffEnd         INTEGER,
35      wmanIfBsOfdmaStartOfRngCodes          INTEGER,
36      wmanIfBsOfdmaPermutationBase          INTEGER,
37      wmanIfBsOfdmaULAllocSubchBitmap       OCTET STRING,
38      wmanIfBsOfdmaOptPermULAllocSubchBitmap OCTET STRING,
39      wmanIfBsOfdmaBandAMCAllocThreshold    INTEGER,
40      wmanIfBsOfdmaBandAMCReleaseThreshold  INTEGER,
41      wmanIfBsOfdmaBandAMCAllocTimer        INTEGER,
42      wmanIfBsOfdmaBandAMCReleaseTimer      INTEGER,
43      wmanIfBsOfdmaBandStatRepMAXPeriod     INTEGER,
44      wmanIfBsOfdmaBandAMCRetryTimer        INTEGER,
45      wmanIfBsOfdmaSafetyChAllocThreshold   INTEGER,
46      wmanIfBsOfdmaSafetyChReleaseThreshold INTEGER,
47      wmanIfBsOfdmaSafetyChAllocTimer       INTEGER,
48      wmanIfBsOfdmaSafetyChReleaseTimer     INTEGER,
49      wmanIfBsOfdmaBinStatRepMAXPeriod      INTEGER,
50      wmanIfBsOfdmaSafetyChRetryTimer       INTEGER,
51      wmanIfBsOfdmaHARQAckDelayULBurst      INTEGER,
52      wmanIfBsOfdmaCQICHBandAMCTranaDelay   INTEGER}
53
54
55
56
57
58
59
60  wmanIfBsOfdmaCtBasedResvTimeout OBJECT-TYPE
61      SYNTAX      INTEGER (1..255)
62      MAX-ACCESS  read-write
63      STATUS      current
64      DESCRIPTION
65

```

```

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

```

```

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

```

expressed as a power of 2. Range: 0..15 (the highest order bits shall be unused and set to 0)."

REFERENCE

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

DEFVAL { 15 }

::= { wmanIfBsOfdmaUplinkChannelEntry 9 }

wmanIfBsOfdmaStartOfRngCodes OBJECT-TYPE

SYNTAX INTEGER (0..255)

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Indicates the starting number, S, of the group of codes used for this uplink. All the ranging codes used on this uplink will be between S and ((S+N+M+L) mod 256). Where, N is the number of initial-ranging codes M is the number of periodic-ranging codes L is the number of bandwidth-request codes The range of values is 0 S255"

REFERENCE

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

DEFVAL { 0 }

::= { wmanIfBsOfdmaUplinkChannelEntry 10 }

wmanIfBsOfdmaPermutationBase OBJECT-TYPE

SYNTAX INTEGER (0..255)

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Determines the UL_IDcell parameter for the subcarrier permutation to be used on this uplink channel"

REFERENCE

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

DEFVAL { 0 }

::= { wmanIfBsOfdmaUplinkChannelEntry 11 }

wmanIfBsOfdmaULAllocSubchBitmap OBJECT-TYPE

SYNTAX OCTET STRING (SIZE (9))

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"This is a bitmap describing the sub-channels allocated to the segment in the UL, when using the uplink PUSC permutation. The LSB of the first byte shall correspond to subchannel 0. For any bit that is not set, the corresponding subchannel shall not be used by the SS on that segment"

REFERENCE

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

::= { wmanIfBsOfdmaUplinkChannelEntry 12 }

wmanIfBsOfdmaOptPermULAllocSubchBitmap OBJECT-TYPE

SYNTAX OCTET STRING (SIZE (13))

MAX-ACCESS read-write

STATUS current

```

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

```

```

1      REFERENCE
2          "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
3      ::= { wmanIfBsOfdmaUplinkChannelEntry 17 }
4
5
6  wmanIfBsOfdmaBandStatRepMAXPeriod OBJECT-TYPE
7      SYNTAX      INTEGER (0 .. 255)
8      UNITS       "Frame"
9      MAX-ACCESS  read-write
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     ::= { wmanIfBsOfdmaUplinkChannelEntry 18 }
17
18
19
20  wmanIfBsOfdmaBandAMCRetryTimer OBJECT-TYPE
21      SYNTAX      INTEGER (0 .. 255)
22      UNITS       "Frame"
23      MAX-ACCESS  read-write
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      ::= { wmanIfBsOfdmaUplinkChannelEntry 19 }
31
32
33
34
35  wmanIfBsOfdmaSafetyChAllocThreshold OBJECT-TYPE
36      SYNTAX      INTEGER (0 .. 255)
37      UNITS       "dB"
38      MAX-ACCESS  read-write
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      ::= { wmanIfBsOfdmaUplinkChannelEntry 20 }
46
47
48
49  wmanIfBsOfdmaSafetyChReleaseThreshold OBJECT-TYPE
50      SYNTAX      INTEGER (0 .. 255)
51      UNITS       "dB"
52      MAX-ACCESS  read-write
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      ::= { wmanIfBsOfdmaUplinkChannelEntry 21 }
60
61
62
63  wmanIfBsOfdmaSafetyChAllocTimer OBJECT-TYPE
64      SYNTAX      INTEGER (0 .. 255)
65

```

```

1      UNITS          "Frame"
2      MAX-ACCESS read-write
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      ::= { wmanIfBsOfdmaUplinkChannelEntry 22 }
10
11
12
13 wmanIfBsOfdmaSafetyChReleaseTimer OBJECT-TYPE
14     SYNTAX      INTEGER (0 .. 255)
15     UNITS       "Frame"
16     MAX-ACCESS  read-write
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     ::= { wmanIfBsOfdmaUplinkChannelEntry 23 }
24
25
26
27 wmanIfBsOfdmaBinStatRepMAXPeriod OBJECT-TYPE
28     SYNTAX      INTEGER (0 .. 255)
29     UNITS       "Frame"
30     MAX-ACCESS  read-write
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     ::= { wmanIfBsOfdmaUplinkChannelEntry 24 }
38
39
40
41
42 wmanIfBsOfdmaSafetyChaRetryTimer OBJECT-TYPE
43     SYNTAX      INTEGER (0 .. 255)
44     UNITS       "Frame"
45     MAX-ACCESS  read-write
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     ::= { wmanIfBsOfdmaUplinkChannelEntry 25 }
53
54
55
56 wmanIfBsOfdmaHARQAackDelayULBurst OBJECT-TYPE
57     SYNTAX      INTEGER { oneframeoffset(1),
58                          twoframesoffset(2),
59                          threeframesoffset(3) }
60     MAX-ACCESS  read-write
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          ::= { wmanIfBsOfdmaUplinkChannelEntry 26 }
8
9
10     wmanIfBsOfdmaCQICHBandAMCTranaDelay 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 CQICH band AMC transition
17              delay."
18         REFERENCE
19             "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
20             ::= { wmanIfBsOfdmaUplinkChannelEntry 27 }
21
22
23
24     wmanIfBsOfdmaDownlinkChannelTable OBJECT-TYPE
25         SYNTAX      SEQUENCE OF WmanIfBsOfdmaDownlinkChannelEntry
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             ::= { wmanIfBsOfdmaPhy 2 }
34
35
36
37     wmanIfBsOfdmaDownlinkChannelEntry OBJECT-TYPE
38         SYNTAX      WmanIfBsOfdmaDownlinkChannelEntry
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         ::= { wmanIfBsOfdmaDownlinkChannelTable 1 }
48
49
50
51     WmanIfBsOfdmaDownlinkChannelEntry ::= SEQUENCE {
52         wmanIfBsOfdmaBsEIRP                INTEGER,
53         wmanIfBsOfdmaChannelNumber          WmanIfChannelNumber,
54         wmanIfBsOfdmaTTG                    INTEGER,
55         wmanIfBsOfdmaRTG                    INTEGER,
56         wmanIfBsOfdmaInitRngMaxRSS          INTEGER,
57         wmanIfBsOfdmaDownlinkCenterFreq    Unsigned32,
58         wmanIfBsOfdmaBsId                   WmanIfBsIdType,
59         wmanIfBsOfdmaMacVersion             WmanIfMacVersion,
60         wmanIfBsOfdmaFrameDurationCode     INTEGER,
61         wmanIfBsOfdmaSizeCqichIdField      INTEGER,
62         wmanIfBsOfdmaHARQAckDelayBurst     INTEGER}
63
64
65

```

```

1
2 wmanIfBsOfdmaBsEIRP OBJECT-TYPE
3     SYNTAX      INTEGER (0..65535)
4     UNITS       "dBm"
5     MAX-ACCESS  read-write
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    ::= { wmanIfBsOfdmaDownlinkChannelEntry 1 }
14
15 wmanIfBsOfdmaChannelNumber OBJECT-TYPE
16     SYNTAX      WmanIfChannelNumber
17     MAX-ACCESS  read-write
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    ::= { wmanIfBsOfdmaDownlinkChannelEntry 2 }
25
26 wmanIfBsOfdmaTTG OBJECT-TYPE
27     SYNTAX      INTEGER (0..255)
28     MAX-ACCESS  read-write
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    ::= { wmanIfBsOfdmaDownlinkChannelEntry 3 }
35
36 wmanIfBsOfdmaRTG OBJECT-TYPE
37     SYNTAX      INTEGER (0..255)
38     MAX-ACCESS  read-write
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    ::= { wmanIfBsOfdmaDownlinkChannelEntry 4 }
45
46 wmanIfBsOfdmaInitRngMaxRSS OBJECT-TYPE
47     SYNTAX      INTEGER (0..65535)
48     UNITS       "dBm"
49     MAX-ACCESS  read-write
50     STATUS      current
51     DESCRIPTION
52         "Initial Ranging Max. Received Signal Strength at BS
53         Signed in units of 1 dBm."
54    REFERENCE
55
56
57
58
59
60
61
62
63
64
65

```

```

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

```

12    REFERENCE
13        "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
14    ::= { wmanIfBsOfdmaDownlinkChannelEntry 6 }
15
16
17
18 wmanIfBsOfdmaBsId OBJECT-TYPE
19     SYNTAX      WmanIfBsIdType
20     MAX-ACCESS  read-write
21     STATUS      current
22     DESCRIPTION
23        "Base station ID."
```

```

24    REFERENCE
25        "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
26    ::= { wmanIfBsOfdmaDownlinkChannelEntry 7 }
27
28
29
30 wmanIfBsOfdmaMacVersion OBJECT-TYPE
31     SYNTAX      WmanIfMacVersion
32     MAX-ACCESS  read-write
33     STATUS      current
34     DESCRIPTION
35        "This parameter specifies the version of 802.16 to which
36         the message originator conforms."
```

```

37    REFERENCE
38        "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
39    ::= { wmanIfBsOfdmaDownlinkChannelEntry 8 }
40
41
42
43 wmanIfBsOfdmaFrameDurationCode OBJECT-TYPE
44     SYNTAX      INTEGER { aASGap(0),
45                          duration2ms(1),
46                          duration2dot5ms(2),
47                          duration4ms(3),
48                          duration5ms(4),
49                          duration8ms(5),
50                          duration10ms(6),
51                          duration12dot5ms(7),
52                          duration20ms(8) }
53
54     MAX-ACCESS  read-write
55     STATUS      current
56     DESCRIPTION
57        "The duration of the frame. The frame duration code values
58         are specified in Table 274."
```

```

59    REFERENCE
60        "Table 273, in IEEE Std 802.16-2004"
61    ::= { wmanIfBsOfdmaDownlinkChannelEntry 9 }
62
63
64
65

```

```

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

```

```

1      STATUS      current
2      DESCRIPTION
3          "This table provides one row for each UCD burst profile.
4          This table is double indexed. The primary index is an
5          ifIndex with an ifType of propBWA2Mp. The secondary index
6          is wmanIfBsOfdmaUiucIndex."
7      INDEX      { ifIndex, wmanIfBsOfdmaUiucIndex }
8      ::= { wmanIfBsOfdmaUcdBurstProfileTable 1 }
9
10
11
12      WmanIfBsOfdmaUcdBurstProfileEntry ::= SEQUENCE {
13          wmanIfBsOfdmaUiucIndex      INTEGER,
14          wmanIfBsOfdmaUcdFecCodeType  WmanIfOfdmaFecCodeType,
15          wmanIfBsOfdmaRangingDataRatio  INTEGER,
16          wmanIfBsOfdmaNorCOverNOVERRIDE OCTET STRING,
17          wmanIfBsOfdmaUcdBurstProfileRowStatus RowStatus}
18
19
20      wmanIfBsOfdmaUiucIndex OBJECT-TYPE
21          SYNTAX      INTEGER (1 .. 10)
22          MAX-ACCESS  not-accessible
23          STATUS      current
24          DESCRIPTION
25              "The Uplink Interval Usage Code indicates the uplink burst
26              profile in the UCD message, and is used along with ifIndex
27              to identify an entry in the
28              wmanIfBsOfdmaUcdBurstProfileTable."
29          REFERENCE
30              "Subclause 8.4.5.4.1, in IEEE Std 802.16-2004"
31          ::= { wmanIfBsOfdmaUcdBurstProfileEntry 1 }
32
33
34
35      wmanIfBsOfdmaUcdFecCodeType OBJECT-TYPE
36          SYNTAX      WmanIfOfdmaFecCodeType
37          MAX-ACCESS  read-create
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          ::= { wmanIfBsOfdmaUcdBurstProfileEntry 2 }
44
45
46
47      wmanIfBsOfdmaRangingDataRatio OBJECT-TYPE
48          SYNTAX      INTEGER (0 .. 255)
49          MAX-ACCESS  read-create
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          ::= { wmanIfBsOfdmaUcdBurstProfileEntry 3 }
57
58
59
60      wmanIfBsOfdmaNorCOverNOVERRIDE OBJECT-TYPE
61          SYNTAX      OCTET STRING (SIZE (5))
62          MAX-ACCESS  read-create
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
5            IEEE 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      ::= { wmanIfBsOfdmaUcdBurstProfileEntry 4 }
14
15  wmanIfBsOfdmaUcdBurstProfileRowStatus OBJECT-TYPE
16      SYNTAX      RowStatus
17      MAX-ACCESS   read-create
18      STATUS      current
19      DESCRIPTION
20          "This object is used to create a new row or modify or delete
21            an existing row in this table. If the implementator of this
22            MIB has choosen not to implement 'dynamic assignment' of
23            profiles, this object is not useful and should return
24            noSuchName upon SNMP request."
25      ::= { wmanIfBsOfdmaUcdBurstProfileEntry 5 }
26
27  wmanIfBsOfdmaDcdBurstProfileTable OBJECT-TYPE
28      SYNTAX      SEQUENCE OF WmanIfBsOfdmaDcdBurstProfileEntry
29      MAX-ACCESS   not-accessible
30      STATUS      current
31      DESCRIPTION
32          "This table provides one row for each DCD burst profile.
33            This table is double indexed. The primary index is an
34            ifIndex with an ifType of propBWA2Mp. The secondary index
35            is wmanIfBsOfdmaDiucIndex."
36      ::= { wmanIfBsOfdmaPhy 4 }
37
38  wmanIfBsOfdmaDcdBurstProfileEntry OBJECT-TYPE
39      SYNTAX      WmanIfBsOfdmaDcdBurstProfileEntry
40      MAX-ACCESS   not-accessible
41      STATUS      current
42      DESCRIPTION
43          "This table provides one row for each DCD burst profile.
44            This table is double indexed. The primary index is an
45            ifIndex with an ifType of propBWA2Mp. The secondary index
46            is wmanIfBsOfdmaDiucIndex."
47      INDEX      { ifIndex, wmanIfBsOfdmaDiucIndex }
48      ::= { wmanIfBsOfdmaDcdBurstProfileTable 1 }
49
50  WmanIfBsOfdmaDcdBurstProfileEntry ::= SEQUENCE {
51      wmanIfBsOfdmaDiucIndex          INTEGER,
52      wmanIfBsOfdmaDownlinkFrequency  Unsigned32,
53      wmanIfBsOfdmaDcdFecCodeType     WmanIfOfdmaFecCodeType,
54      wmanIfBsOfdmaDiucMandatoryExitThresh  INTEGER,
55      wmanIfBsOfdmaDiucMinEntryThresh  INTEGER,

```

```

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

```

```

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

```

```

1         "This field indicates the FFT sizes supported by MS."
2         ::= { wmanIfBsMsOfdmaReqCapabilitiesEntry 1 }
3
4
5 wmanIfBsMsOfdmaReqCapDemodulator OBJECT-TYPE
6     SYNTAX      WmanIfOfdmaMsDeModType
7     MAX-ACCESS  read-only
8     STATUS      current
9     DESCRIPTION
10        "This field indicates the different demodulator options
11        supported by MS for downlink."
12        ::= { wmanIfBsMsOfdmaReqCapabilitiesEntry 2 }
13
14
15 wmanIfBsMsOfdmaReqCapModulator OBJECT-TYPE
16     SYNTAX      WmanIfOfdmaMsModType
17     MAX-ACCESS  read-only
18     STATUS      current
19     DESCRIPTION
20        "This field indicates the different modulator options
21        supported by MS for uplink."
22        ::= { wmanIfBsMsOfdmaReqCapabilitiesEntry 3 }
23
24
25
26 wmanIfBsMsOfdmaReqCapPermutation OBJECT-TYPE
27     SYNTAX      WmanIfOfdmaPermutation
28     MAX-ACCESS  read-only
29     STATUS      current
30     DESCRIPTION
31        "This field indicates the OFDMA MS Permutation support"
32        ::= { wmanIfBsMsOfdmaReqCapabilitiesEntry 4 }
33
34
35
36 wmanIfBsMsOfdmaReqCapMobilityFeature OBJECT-TYPE
37     SYNTAX      WmanIfOfdmaMobility
38     MAX-ACCESS  read-only
39     STATUS      current
40     DESCRIPTION
41        "The field indicates whether or not the MS supports
42        mobility hand-over, Sleepmode, and Idle-mode."
43        ::= { wmanIfBsMsOfdmaReqCapabilitiesEntry 5 }
44
45
46
47 wmanIfBsMsOfdmaRspCapabilitiesTable OBJECT-TYPE
48     SYNTAX      SEQUENCE OF WmanIfBsMsOfdmaRspCapabilitiesEntry
49     MAX-ACCESS  not-accessible
50     STATUS      current
51     DESCRIPTION
52        "This table contains the basic capability information,
53        specific to OFDMA Phy, of MSs that have been reported by
54        MSs to BS using RNG-REQ, SBC-REQ and REG-REQ messages.
55        Entries in this table should be created when an MS
56        registers with a BS."
57        ::= { wmanIfBsOfdmaPhy 6 }
58
59
60
61 wmanIfBsMsOfdmaRspCapabilitiesEntry OBJECT-TYPE
62     SYNTAX      WmanIfBsMsOfdmaRspCapabilitiesEntry
63     MAX-ACCESS  not-accessible
64     STATUS      current
65

```

```

1      DESCRIPTION
2          "This table provides one row for each MS that has been
3          registered in the BS. This table augments the table
4          wmanIfBsRegisteredSsTable."
5      AUGMENTS { wmanIfBsRegisteredSsEntry }
6      ::= { wmanIfBsMsOfdmaRspCapabilitiesTable 1 }
7
8
9
10     WmanIfBsMsOfdmaRspCapabilitiesEntry ::= SEQUENCE {
11         wmanIfBsMsOfdmaRspCapFftSizes          WmanIfOfdmFftSizes,
12         wmanIfBsMsOfdmaRspCapDemodulator        WmanIfOfdmaMsDeModType,
13         wmanIfBsMsOfdmaRspCapModulator          WmanIfOfdmaMsModType,
14         wmanIfBsMsOfdmaRspCapPermutation        WmanIfOfdmaPermutation,
15         wmanIfBsMsOfdmaRspCapMobilityFeature    WmanIfOfdmaMobility}
16
17
18     wmanIfBsMsOfdmaRspCapFftSizes OBJECT-TYPE
19         SYNTAX          WmanIfOfdmFftSizes
20         MAX-ACCESS      read-only
21         STATUS          current
22         DESCRIPTION
23             "This field indicates the FFT sizes negotiated with the
24             MS."
25         ::= { wmanIfBsMsOfdmaRspCapabilitiesEntry 1 }
26
27
28
29     wmanIfBsMsOfdmaRspCapDemodulator OBJECT-TYPE
30         SYNTAX          WmanIfOfdmaMsDeModType
31         MAX-ACCESS      read-only
32         STATUS          current
33         DESCRIPTION
34             "This field indicates the different demodulator options
35             negotiated for MS for downlink."
36         ::= { wmanIfBsMsOfdmaRspCapabilitiesEntry 2 }
37
38
39
40     wmanIfBsMsOfdmaRspCapModulator OBJECT-TYPE
41         SYNTAX          WmanIfOfdmaMsModType
42         MAX-ACCESS      read-only
43         STATUS          current
44         DESCRIPTION
45             "This field indicates the different modulator options
46             negotiated for MS for uplink."
47         ::= { wmanIfBsMsOfdmaRspCapabilitiesEntry 3 }
48
49
50
51     wmanIfBsMsOfdmaRspCapPermutation OBJECT-TYPE
52         SYNTAX          WmanIfOfdmaPermutation
53         MAX-ACCESS      read-only
54         STATUS          current
55         DESCRIPTION
56             "This field indicates the OFDMA MS Permutation support
57             negotiated for MS."
58         ::= { wmanIfBsMsOfdmaRspCapabilitiesEntry 4 }
59
60
61
62     wmanIfBsMsOfdmaRspCapMobilityFeature OBJECT-TYPE
63         SYNTAX          WmanIfOfdmaMobility
64         MAX-ACCESS      read-only
65         STATUS          current

```

```

1      DESCRIPTION
2          "The field indicates the mobility hand-over, Sleepmode,
3          and Idle-mode negotiated for MS."
4      ::= { wmanIfBsOfdmaRspCapabilitiesEntry 5 }
5
6
7      wmanIfBsOfdmaCapabilitiesTable OBJECT-TYPE
8          SYNTAX      SEQUENCE OF WmanIfBsOfdmaCapabilitiesEntry
9          MAX-ACCESS  not-accessible
10         STATUS      current
11         DESCRIPTION
12             "This table contains the basic capabilities, specific to
13             OFDMA Phy, of the BS as implemented in BS hardware and
14             software. These capabilities along with the configuration
15             for them (wmanIfBsOfdmaCapabilitiesConfigTable) are used
16             for negotiation of basic capabilities with SS using
17             RNG-RSP, SBC-RSP and REG-RSP messages. The negotiated
18             capabilities are obtained by interSubclause of MS raw
19             reported capabilities, BS raw capabilities and BS
20             configured capabilities. The objects in the table have
21             read-only access. The table is maintained by BS."
22         ::= { wmanIfBsOfdmaPhy 7 }
23
24
25      wmanIfBsOfdmaCapabilitiesEntry OBJECT-TYPE
26          SYNTAX      WmanIfBsOfdmaCapabilitiesEntry
27          MAX-ACCESS  not-accessible
28          STATUS      current
29          DESCRIPTION
30              "This table provides one row for each BS sector and is
31              indexed by ifIndex."
32          INDEX { ifIndex }
33          ::= { wmanIfBsOfdmaCapabilitiesTable 1 }
34
35
36      WmanIfBsOfdmaCapabilitiesEntry ::= SEQUENCE {
37          wmanIfBsOfdmaCapFftSizes          WmanIfOfdmFftSizes,
38          wmanIfBsOfdmaCapDemodulator       WmanIfOfdmaMsDeModType,
39          wmanIfBsOfdmaCapModulator         WmanIfOfdmaMsModType,
40          wmanIfBsOfdmaCapPermutation       WmanIfOfdmaPermutation,
41          wmanIfBsOfdmaCapMobilityFeature   WmanIfOfdmaMobility}
42
43
44      wmanIfBsOfdmaCapFftSizes OBJECT-TYPE
45          SYNTAX      WmanIfOfdmFftSizes
46          MAX-ACCESS  read-only
47          STATUS      current
48          DESCRIPTION
49              "This field indicates the FFT sizes supported by BS."
50          ::= { wmanIfBsOfdmaCapabilitiesEntry 1 }
51
52
53      wmanIfBsOfdmaCapDemodulator OBJECT-TYPE
54          SYNTAX      WmanIfOfdmaMsDeModType
55          MAX-ACCESS  read-only
56          STATUS      current
57          DESCRIPTION
58              "This field indicates the different demodulator options
59              supported by BS."
60

```

```

1      ::= { wmanIfBsOfdmaCapabilitiesEntry 2 }
2
3  wmanIfBsOfdmaCapModulator OBJECT-TYPE
4      SYNTAX      WmanIfOfdmaMsModType
5      MAX-ACCESS  read-only
6      STATUS      current
7      DESCRIPTION
8          "This field indicates the different modulator options
9           supported by BS."
10     ::= { wmanIfBsOfdmaCapabilitiesEntry 3 }
11
12  wmanIfBsOfdmaCapPermutation OBJECT-TYPE
13      SYNTAX      WmanIfOfdmaPermutation
14      MAX-ACCESS  read-only
15      STATUS      current
16      DESCRIPTION
17          "This field indicates the OFDMA MS Permutation support
18           supported by BS."
19     ::= { wmanIfBsOfdmaCapabilitiesEntry 4 }
20
21  wmanIfBsOfdmaCapMobilityFeature OBJECT-TYPE
22      SYNTAX      WmanIfOfdmaMobility
23      MAX-ACCESS  read-only
24      STATUS      current
25      DESCRIPTION
26          "The field indicates the mobility hand-over, Sleepmode,
27           and Idle-mode supported by BS."
28     ::= { wmanIfBsOfdmaCapabilitiesEntry 5 }
29
30  wmanIfBsOfdmaCapabilitiesConfigTable OBJECT-TYPE
31      SYNTAX      SEQUENCE OF WmanIfBsOfdmaCapabilitiesConfigEntry
32      MAX-ACCESS  not-accessible
33      STATUS      current
34      DESCRIPTION
35          "This table contains the configuration for basic
36           capabilities of BS, specific to OFDMA Phy. The table is
37           intended to be used to restrict the Capabilities
38           implemented by BS, for example in order to comply with
39           local regulatory requirements. The BS should use the
40           configuration along with the implemented Capabilities
41           (wmanIfBsOfdmaPhyTable) for negotiation of basic
42           capabilities with SS using RNG-RSP, SBC-RSP and REG-RSP
43           messages. The negotiated capabilities are obtained by
44           interSubclause of MS reported capabilities, BS raw
45           capabilities and BS configured capabilities. The objects
46           in the table have read-write access. The rows are created
47           by BS as a copy of wmanIfBsBasicCapabilitiesTable
48           and can be modified by NMS."
49     ::= { wmanIfBsOfdmaPhy 8 }
50
51  wmanIfBsOfdmaCapabilitiesConfigEntry OBJECT-TYPE
52      SYNTAX      WmanIfBsOfdmaCapabilitiesConfigEntry
53      MAX-ACCESS  not-accessible
54      STATUS      current
55
56
57
58
59
60
61
62
63
64
65

```

```

1      DESCRIPTION
2          "This table provides one row for each BS sector and is
3          indexed by ifIndex."
4      INDEX { ifIndex }
5      ::= { wmanIfBsOfdmaCapabilitiesConfigTable 1 }
6
7
8      WmanIfBsOfdmaCapabilitiesConfigEntry ::= SEQUENCE {
9          wmanIfBsOfdmaCapCfgFftSizes          WmanIfOfdmFftSizes,
10         wmanIfBsOfdmaCapCfgDemodulator        WmanIfOfdmaMsDeModType,
11         wmanIfBsOfdmaCapCfgModulator          WmanIfOfdmaMsModType,
12         wmanIfBsOfdmaCapCfgPermutation        WmanIfOfdmaPermutation,
13         wmanIfBsOfdmaCapCfgMobilityFeature    WmanIfOfdmaMobility}
14
15
16      wmanIfBsOfdmaCapCfgFftSizes OBJECT-TYPE
17          SYNTAX      WmanIfOfdmFftSizes
18          MAX-ACCESS  read-only
19          STATUS      current
20          DESCRIPTION
21              "This field indicates the FFT sizes configured for the BS."
22          ::= { wmanIfBsOfdmaCapabilitiesConfigEntry 1 }
23
24
25      wmanIfBsOfdmaCapCfgDemodulator OBJECT-TYPE
26          SYNTAX      WmanIfOfdmaMsDeModType
27          MAX-ACCESS  read-only
28          STATUS      current
29          DESCRIPTION
30              "This field indicates the different demodulator options
31              configured for the BS."
32          ::= { wmanIfBsOfdmaCapabilitiesConfigEntry 2 }
33
34
35      wmanIfBsOfdmaCapCfgModulator OBJECT-TYPE
36          SYNTAX      WmanIfOfdmaMsModType
37          MAX-ACCESS  read-only
38          STATUS      current
39          DESCRIPTION
40              "This field indicates the different modulator options
41              configured for the BS."
42          ::= { wmanIfBsOfdmaCapabilitiesConfigEntry 3 }
43
44
45      wmanIfBsOfdmaCapCfgPermutation OBJECT-TYPE
46          SYNTAX      WmanIfOfdmaPermutation
47          MAX-ACCESS  read-only
48          STATUS      current
49          DESCRIPTION
50              "This field indicates the OFDMA MS Permutation support
51              configured for the BS."
52          ::= { wmanIfBsOfdmaCapabilitiesConfigEntry 4 }
53
54
55      wmanIfBsOfdmaCapCfgMobilityFeature OBJECT-TYPE
56          SYNTAX      WmanIfOfdmaMobility
57          MAX-ACCESS  read-only
58          STATUS      current
59          DESCRIPTION
60              "The field indicates the mobility hand-over, Sleepmode,
61

```

```

1         and Idle-mode configured for the BS."
2         ::= { wmanIfBsOfdmaCapabilitiesConfigEntry 5 }
3
4
5     --
6     -- SS object group - containing tables and objects to be implemented in
7     -- the Subscriber station
8
9     --
10
11     -- wmanIfSsCps contain the SS Common Part Sublayer objects
12     --
13     wmanIfSsCps OBJECT IDENTIFIER ::= { wmanIfSsObjects 1 }
14
15     --
16     -- wmanIfSsConfigurationTable contains global parameters for SS
17     --
18     wmanIfSsConfigurationTable OBJECT-TYPE
19         SYNTAX      SEQUENCE OF WmanIfSsConfigurationEntry
20         MAX-ACCESS   not-accessible
21         STATUS       current
22         DESCRIPTION
23             "This table contains one row for the SS system
24             parameters."
25         REFERENCE
26             "Subclause 10.1 in IEEE Std 802.16-2004"
27         ::= { wmanIfSsCps 1 }
28
29     wmanIfSsConfigurationEntry OBJECT-TYPE
30         SYNTAX      WmanIfSsConfigurationEntry
31         MAX-ACCESS   not-accessible
32         STATUS       current
33         DESCRIPTION
34             "This table is indexed by ifIndex."
35         INDEX { ifIndex }
36         ::= { wmanIfSsConfigurationTable 1 }
37
38     WmanIfSsConfigurationEntry ::= SEQUENCE {
39         wmanIfSsLostDLMapInterval          INTEGER,
40         wmanIfSsLostULMapInterval          INTEGER,
41         wmanIfSsContentionRangRetries      INTEGER,
42         wmanIfSsRequestRetries             INTEGER,
43         wmanIfSsRegRequestRetries          INTEGER,
44         wmanIfSsTftpBackoffStart           INTEGER,
45         wmanIfSsTftpBackoffEnd             INTEGER,
46         wmanIfSsTftpRequestRetries         INTEGER,
47         wmanIfSsTftpDownloadRetries        INTEGER,
48         wmanIfSsTftpWait                   INTEGER,
49         wmanIfSsToDRetries                 INTEGER,
50         wmanIfSsToDRetryPeriod             INTEGER,
51         wmanIfSsT1Timeout                  INTEGER,
52         wmanIfSsT2Timeout                  INTEGER,
53         wmanIfSsT3Timeout                  INTEGER,
54         wmanIfSsT4Timeout                  INTEGER,
55         wmanIfSsT6Timeout                  INTEGER,
56         wmanIfSsT12Timeout                 INTEGER,

```

```

1      wmanIfSsT14Timeout      INTEGER,
2      wmanIfSsT16Timeout      INTEGER,
3      wmanIfSsT18Timeout      INTEGER,
4      wmanIfSsT19Timeout      INTEGER,
5      wmanIfSsT20Timeout      INTEGER,
6      wmanIfSsT21Timeout      INTEGER,
7      wmanIfSsSBCRequestRetries      INTEGER,
8      wmanIfSsTftpCpltRetries      INTEGER,
9      wmanIfSsT26Timeout      INTEGER,
10     wmanIfSsDLManagProcTime      INTEGER}
11
12
13
14     wmanIfSsLostDLMapInterval OBJECT-TYPE
15         SYNTAX      INTEGER (0..600)
16         UNITS        "milliseconds"
17         MAX-ACCESS   read-write
18         STATUS        current
19         DESCRIPTION
20             "Time since last received DL-MAP message before downlink
21             synchronization is considered lost in ms."
22             ::= { wmanIfSsConfigurationEntry 1 }
23
24
25
26     wmanIfSsLostULMapInterval OBJECT-TYPE
27         SYNTAX      INTEGER (0..600)
28         UNITS        "milliseconds"
29         MAX-ACCESS   read-write
30         STATUS        current
31         DESCRIPTION
32             "Time since last received UL-MAP message before uplink
33             synchronization is considered lost in ms."
34             ::= { wmanIfSsConfigurationEntry 2 }
35
36
37
38     wmanIfSsContentionRangRetries OBJECT-TYPE
39         SYNTAX      INTEGER (16..65535)
40         MAX-ACCESS   read-write
41         STATUS        current
42         DESCRIPTION
43             "Number of retries on contention Ranging Requests."
44             ::= { wmanIfSsConfigurationEntry 3 }
45
46
47
48     wmanIfSsRequestRetries OBJECT-TYPE
49         SYNTAX      INTEGER (16..65535)
50         MAX-ACCESS   read-write
51         STATUS        current
52         DESCRIPTION
53             "Number of retries on bandwidth allocation requests."
54             ::= { wmanIfSsConfigurationEntry 4 }
55
56
57
58     wmanIfSsRegRequestRetries OBJECT-TYPE
59         SYNTAX      INTEGER (3..65535)
60         MAX-ACCESS   read-write
61         STATUS        current
62         DESCRIPTION
63             "Number of retries on registration requests."
64             ::= { wmanIfSsConfigurationEntry 5 }
65

```

```

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

```

```

1          SYNTAX          INTEGER (5..65535)
2          UNITS            "minutes"
3          MAX-ACCESS      read-write
4          STATUS          current
5          DESCRIPTION
6              "The retry period to re-establishing the Time of Day, as
7              describe in the network entry procedure."
8          ::= { wmanIfSsConfigurationEntry 12 }
9
10
11
12 wmanIfSsT1Timeout OBJECT-TYPE
13     SYNTAX          INTEGER (0..50000)
14     UNITS            "milliseconds"
15     MAX-ACCESS      read-write
16     STATUS          current
17     DESCRIPTION
18         "Wait for DCD timeout in ms."
19     ::= { wmanIfSsConfigurationEntry 13 }
20
21
22
23 wmanIfSsT2Timeout OBJECT-TYPE
24     SYNTAX          INTEGER (0..10000)
25     UNITS            "milliseconds"
26     MAX-ACCESS      read-write
27     STATUS          current
28     DESCRIPTION
29         "Wait for broadcast ranging timeout in ms."
30     ::= { wmanIfSsConfigurationEntry 14 }
31
32
33
34 wmanIfSsT3Timeout OBJECT-TYPE
35     SYNTAX          INTEGER (0..200)
36     UNITS            "milliseconds"
37     MAX-ACCESS      read-write
38     STATUS          current
39     DESCRIPTION
40         "Ranging Response reception timeout following the
41         transmission of a Ranging Request in ms."
42     ::= { wmanIfSsConfigurationEntry 15 }
43
44
45
46 wmanIfSsT4Timeout OBJECT-TYPE
47     SYNTAX          INTEGER (30..35)
48     UNITS            "seconds"
49     MAX-ACCESS      read-write
50     STATUS          current
51     DESCRIPTION
52         "Wait for unicast ranging opportunity. If the pending until
53         complete field was used earlier by this SS, then the value
54         of that field shall be added to this interval in second."
55     ::= { wmanIfSsConfigurationEntry 16 }
56
57
58
59 wmanIfSsT6Timeout OBJECT-TYPE
60     SYNTAX          INTEGER (0..3000)
61     UNITS            "milliseconds"
62     MAX-ACCESS      read-write
63     STATUS          current
64     DESCRIPTION
65

```

```

1         "Wait for registration response in ms."
2         ::= { wmanIfSsConfigurationEntry 17 }
3
4
5 wmanIfSsT12Timeout OBJECT-TYPE
6     SYNTAX      INTEGER (0..50000)
7     UNITS       "milliseconds"
8     MAX-ACCESS  read-write
9     STATUS      current
10    DESCRIPTION
11        "Wait for UCD descriptor in ms."
12    ::= { wmanIfSsConfigurationEntry 18 }
13
14
15 wmanIfSsT14Timeout OBJECT-TYPE
16     SYNTAX      INTEGER (0..200)
17     UNITS       "milliseconds"
18     MAX-ACCESS  read-write
19     STATUS      current
20     DESCRIPTION
21        "Wait for DSX-RVD Timeout in ms."
22    ::= { wmanIfSsConfigurationEntry 19 }
23
24
25
26 wmanIfSsT16Timeout OBJECT-TYPE
27     SYNTAX      INTEGER (10..65535)
28     UNITS       "milliseconds"
29     MAX-ACCESS  read-write
30     STATUS      current
31     DESCRIPTION
32        "wait for bandwidth request grant in ms."
33    ::= { wmanIfSsConfigurationEntry 20 }
34
35
36
37 wmanIfSsT18Timeout OBJECT-TYPE
38     SYNTAX      INTEGER (0..65535)
39     UNITS       "milliseconds"
40     MAX-ACCESS  read-write
41     STATUS      current
42     DESCRIPTION
43        "wait for SBC-RSP timeout in ms."
44    ::= { wmanIfSsConfigurationEntry 21 }
45
46
47
48 wmanIfSsT19Timeout OBJECT-TYPE
49     SYNTAX      INTEGER (0..1048575)
50     UNITS       "milliseconds"
51     MAX-ACCESS  read-write
52     STATUS      current
53     DESCRIPTION
54        "Time DL-channel remains unusable in ms."
55    ::= { wmanIfSsConfigurationEntry 22 }
56
57
58
59 wmanIfSsT20Timeout OBJECT-TYPE
60     SYNTAX      INTEGER (0..65535)
61     UNITS       "milliseconds"
62     MAX-ACCESS  read-write
63     STATUS      current
64     DESCRIPTION
65

```

```

1         "Time SS searches for preambles on a given channel in ms."
2         ::= { wmanIfSsConfigurationEntry 23 }
3
4
5 wmanIfSsT21Timeout OBJECT-TYPE
6     SYNTAX      INTEGER (0..10000)
7     UNITS       "milliseconds"
8     MAX-ACCESS  read-write
9     STATUS      current
10    DESCRIPTION
11        "Time SS searches for DL-MAP on a given channel in ms."
12        ::= { wmanIfSsConfigurationEntry 24 }
13
14
15 wmanIfSsSBCRequestRetries OBJECT-TYPE
16     SYNTAX      INTEGER (3..16)
17     MAX-ACCESS  read-write
18     STATUS      current
19     DESCRIPTION
20        "Number of retries on SBC Request."
21        ::= { wmanIfSsConfigurationEntry 25 }
22
23
24
25 wmanIfSsTftpCpltRetries OBJECT-TYPE
26     SYNTAX      INTEGER (3..16)
27     MAX-ACCESS  read-write
28     STATUS      current
29     DESCRIPTION
30        "Number of retries on TFTP-CPLT."
31        ::= { wmanIfSsConfigurationEntry 26 }
32
33
34
35 wmanIfSsT26Timeout OBJECT-TYPE
36     SYNTAX      INTEGER (10..200)
37     UNITS       "milliseconds"
38     MAX-ACCESS  read-write
39     STATUS      current
40     DESCRIPTION
41        "Wait for TFTP-RSP in ms."
42        ::= { wmanIfSsConfigurationEntry 27 }
43
44
45
46 wmanIfSsDLManagProcTime OBJECT-TYPE
47     SYNTAX      INTEGER (0..200)
48     UNITS       "micro seconds"
49     MAX-ACCESS  read-write
50     STATUS      current
51     DESCRIPTION
52        "Max. time between reception of Fast Power Control
53         management message and compliance to its instructions
54         by SS in us."
55        ::= { wmanIfSsConfigurationEntry 28 }
56
57
58
59 --
60 -- Subscriber Channel Measurement Table
61 --
62 wmanIfSsChannelMeasurementTable OBJECT-TYPE
63     SYNTAX      SEQUENCE OF WmanIfSsChannelMeasurementEntry
64     MAX-ACCESS  not-accessible
65

```

```

1      STATUS      current
2      DESCRIPTION
3          "This table contains downlink channel measurement
4          information for each SS."
5      REFERENCE
6          "6.3.2.3.33 in IEEE Std 802.16-2004"
7      ::= { wmanIfSsCps 2 }
8
9
10
11 wmanIfSsChannelMeasurementEntry OBJECT-TYPE
12     SYNTAX      WmanIfSsChannelMeasurementEntry
13     MAX-ACCESS  not-accessible
14     STATUS      current
15     DESCRIPTION
16         "Each entry in the table contains RSSI and CINR
17         signal quality measurement taken from the SS. The primary
18         index is the ifIndex pointing to SS.
19         wmanIfCmnHistogramIndex is the index to histogram
20         samples. Since there is no time stamp in the table,
21         wmanIfCmnHistogramIndex should be increased monotonically,
22         and wraps around when it reaches the limit.
23         When the measurement entry for a SS reaches the limit,
24         the oldest entry shall be deleted as the new entry is
25         added to the table."
26     INDEX       { ifIndex, wmanIfSsHistogramIndex }
27     ::= { wmanIfSsChannelMeasurementTable 1 }
28
29
30
31
32 WmanIfSsChannelMeasurementEntry ::= SEQUENCE {
33     wmanIfSsHistogramIndex      Unsigned32,
34     wmanIfSsChannelNumber       WmanIfChannelNumber,
35     wmanIfSsStartFrame          INTEGER,
36     wmanIfSsDuration            INTEGER,
37     wmanIfSsBasicReport         BITS,
38     wmanIfSsMeanCinrReport      INTEGER,
39     wmanIfSsStdDeviationCinrReport INTEGER,
40     wmanIfSsMeanRssiReport      INTEGER,
41     wmanIfSsStdDeviationRssiReport INTEGER}
42
43
44
45 wmanIfSsHistogramIndex OBJECT-TYPE
46     SYNTAX      Unsigned32 (1 .. 4294967295)
47     MAX-ACCESS  not-accessible
48     STATUS      current
49     DESCRIPTION
50         "wmanIfSsHistogramIndex identifies the histogram samples
51         in the table for each subscriber station."
52     ::= { wmanIfSsChannelMeasurementEntry 1 }
53
54
55
56 wmanIfSsChannelNumber OBJECT-TYPE
57     SYNTAX      WmanIfChannelNumber
58     MAX-ACCESS  read-only
59     STATUS      current
60     DESCRIPTION
61         "Physical channel number to be reported on."
62     REFERENCE
63         "Subclause 8.5.1 in IEEE Std 802.16-2004"
64
65

```

```

1      ::= { wmanIfSsChannelMeasurementEntry 2 }
2
3
4  wmanIfSsStartFrame OBJECT-TYPE
5      SYNTAX      INTEGER (0 .. 65535)
6      MAX-ACCESS  read-only
7      STATUS      current
8      DESCRIPTION
9          "Frame number in which measurement for this channel
10         started."
11
12     REFERENCE
13         "Subclause 11.12 in IEEE Std 802.16-2004"
14     ::= { wmanIfSsChannelMeasurementEntry 3 }
15
16
17  wmanIfSsDuration OBJECT-TYPE
18      SYNTAX      INTEGER (0..16777215)
19      MAX-ACCESS  read-only
20      STATUS      current
21      DESCRIPTION
22          "Cumulative measurement duration on the channel in
23         multiples of Ts. For any value exceeding 0xFFFFF,
24         report 0xFFFFF."
25
26     REFERENCE
27         "Subclause 11.12 in IEEE Std 802.16-2004"
28     ::= { wmanIfSsChannelMeasurementEntry 4 }
29
30
31  wmanIfSsBasicReport OBJECT-TYPE
32      SYNTAX      BITS {wirelessHuman(0),
33                        unknownTransmission(1),
34                        primaryUser(2),
35                        channelNotMeasured(3)}
36
37      MAX-ACCESS  read-only
38      STATUS      current
39      DESCRIPTION
40          "Bit #0: WirelessHUMAN detected on the channel
41         Bit #1: Unknown transmissions detected on the channel
42         Bit #2: Primary User detected on the channel
43         Bit #3: Unmeasured. Channel not measured"
44
45     REFERENCE
46         "Subclause 11.12 in IEEE Std 802.16-2004"
47     ::= { wmanIfSsChannelMeasurementEntry 5 }
48
49
50  wmanIfSsMeanCinrReport OBJECT-TYPE
51      SYNTAX      INTEGER (0 .. 41)
52      UNITS        "dB"
53      MAX-ACCESS  read-only
54      STATUS      current
55      DESCRIPTION
56          "Mean CINR report."
57
58     REFERENCE
59         "Subclause 8.2.2, 8.3.9 in IEEE Std 802.16-2004"
60     ::= { wmanIfSsChannelMeasurementEntry 6 }
61
62
63  wmanIfSsStdDeviationCinrReport OBJECT-TYPE
64      SYNTAX      INTEGER (0 .. 41)
65

```

```

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

```

```

1      DESCRIPTION
2          "Each entry contains objects describing attributes of one
3          SS wireless interface."
4      INDEX          { ifIndex }
5      ::= { wmanIfSsPkmAuthTable 1 }
6
7
8      WmanIfSsPkmAuthEntry ::= SEQUENCE {
9          wmanIfSsPkmAuthState          INTEGER,
10         wmanIfSsPkmAuthKeySequenceNumber Integer32,
11         wmanIfSsPkmAuthExpiresOld      DateAndTime,
12         wmanIfSsPkmAuthExpiresNew      DateAndTime,
13         wmanIfSsPkmAuthReset           TruthValue,
14         wmanIfSsPkmAuthentInfos         Counter32,
15         wmanIfSsPkmAuthRequests         Counter32,
16         wmanIfSsPkmAuthReplies          Counter32,
17         wmanIfSsPkmAuthRejects          Counter32,
18         wmanIfSsPkmAuthInvalids         Counter32,
19         wmanIfSsPkmAuthRejectErrorCode  INTEGER,
20         wmanIfSsPkmAuthRejectErrorString SnmpAdminString,
21         wmanIfSsPkmAuthInvalidErrorCode INTEGER,
22         wmanIfSsPkmAuthInvalidErrorString SnmpAdminString,
23         wmanIfSsPkmAuthGraceTime        Integer32,
24         wmanIfSsPkmTekGraceTime          Integer32,
25         wmanIfSsPkmAuthWaitTimeout       Integer32,
26         wmanIfSsPkmReauthWaitTimeout     Integer32,
27         wmanIfSsPkmOpWaitTimeout         Integer32,
28         wmanIfSsPkmRekeyWaitTimeout      Integer32,
29         wmanIfSsPkmAuthRejectWaitTimeout Integer32}
30
31
32      wmanIfSsPkmAuthState OBJECT-TYPE
33      SYNTAX          INTEGER {start(1),
34                          authWait(2),
35                          authorized(3),
36                          reauthWait(4),
37                          authRejectWait(5),
38                          silent(6)}
39
40      MAX-ACCESS      read-only
41      STATUS          current
42      DESCRIPTION
43          "The value of this object is the state of the SS
44          authorization FSM. The start state indicates that FSM is
45          in its initial state."
46      ::= { wmanIfSsPkmAuthEntry 1 }
47
48
49      wmanIfSsPkmAuthKeySequenceNumber OBJECT-TYPE
50      SYNTAX          Integer32 (0..15)
51      MAX-ACCESS      read-only
52      STATUS          current
53      DESCRIPTION
54          "The value of this object is the most recent authorization
55          key sequence number for this FSM."
56      ::= { wmanIfSsPkmAuthEntry 2 }
57
58
59      wmanIfSsPkmAuthExpiresOld OBJECT-TYPE
60
61
62
63
64
65

```

```

1      SYNTAX      DateAndTime
2      MAX-ACCESS  read-only
3      STATUS      current
4      DESCRIPTION
5          "The value of this object is the actual clock time for
6          expiration of the immediate predecessor of the most recent
7          authorization key for this FSM. If this FSM has only one
8          authorization key, then the value is the time of activation
9          of this FSM."
10     ::= { wmanIfSsPkmAuthEntry 3 }
11
12 wmanIfSsPkmAuthExpiresNew OBJECT-TYPE
13     SYNTAX      DateAndTime
14     MAX-ACCESS  read-only
15     STATUS      current
16     DESCRIPTION
17         "The value of this object is the actual clock time for
18         expiration of the most recent authorization key for this
19         FSM."
20     ::= { wmanIfSsPkmAuthEntry 4 }
21
22 wmanIfSsPkmAuthReset OBJECT-TYPE
23     SYNTAX      TruthValue
24     MAX-ACCESS  read-write
25     STATUS      current
26     DESCRIPTION
27         "Setting this object to TRUE generates a Reauthorize event
28         in the authorization FSM. Reading this object always
29         returns FALSE."
30     ::= { wmanIfSsPkmAuthEntry 5 }
31
32 wmanIfSsPkmAuthentInfos OBJECT-TYPE
33     SYNTAX      Counter32
34     MAX-ACCESS  read-only
35     STATUS      current
36     DESCRIPTION
37         "The value of this object is the count of times the SS has
38         transmitted an Authentication Information message."
39     ::= { wmanIfSsPkmAuthEntry 6 }
40
41 wmanIfSsPkmAuthRequests OBJECT-TYPE
42     SYNTAX      Counter32
43     MAX-ACCESS  read-only
44     STATUS      current
45     DESCRIPTION
46         "The value of this object is the count of times the SS has
47         transmitted an Authorization Request message."
48     ::= { wmanIfSsPkmAuthEntry 7 }
49
50 wmanIfSsPkmAuthReplies OBJECT-TYPE
51     SYNTAX      Counter32
52     MAX-ACCESS  read-only
53     STATUS      current
54     DESCRIPTION
55         "The value of this object is the count of times the SS has
56         transmitted an Authorization Request message."
57     ::= { wmanIfSsPkmAuthEntry 7 }
58
59
60
61
62
63
64
65

```

```

1         "The value of this object is the count of times the SS has
2         received an Authorization Reply message."
3         ::= { wmanIfSsPkmAuthEntry 8 }
4
5
6 wmanIfSsPkmAuthRejects 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 Reject message."
13        ::= { wmanIfSsPkmAuthEntry 9 }
14
15
16 wmanIfSsPkmAuthInvalids OBJECT-TYPE
17     SYNTAX      Counter32
18     MAX-ACCESS  read-only
19     STATUS      current
20     DESCRIPTION
21        "The value of this object is the count of times the SS has
22        received an Authorization Invalid message."
23        ::= { wmanIfSsPkmAuthEntry 10 }
24
25
26 wmanIfSsPkmAuthRejectErrorCode OBJECT-TYPE
27     SYNTAX      INTEGER {none(1),
28                     unknown(2),
29                     unauthorizedSs(3),
30                     unauthorizedSaid(4),
31                     permanentAuthorizationFailure(8),
32                     timeOfDayNotAcquired(11)}
33     MAX-ACCESS  read-only
34     STATUS      current
35     DESCRIPTION
36        "The value of this object is the enumerated description of
37        the Error-Code in most recent Authorization Reject message
38        received by the SS. This has value unknown(2) if the last
39        Error-Code value was 0, and none(1) if no Authorization
40        Reject message has been received since reboot."
41        ::= { wmanIfSsPkmAuthEntry 11 }
42
43
44 wmanIfSsPkmAuthRejectErrorString OBJECT-TYPE
45     SYNTAX      SnmpAdminString (SIZE (0..128))
46     MAX-ACCESS  read-only
47     STATUS      current
48     DESCRIPTION
49        "The value of this object is the Display-String in most
50        recent Authorization Reject message received by the SS.
51        This is a zero length string if no Authorization Reject
52        message has been received since reboot."
53        ::= { wmanIfSsPkmAuthEntry 12 }
54
55
56 wmanIfSsPkmAuthInvalidErrorCode OBJECT-TYPE
57     SYNTAX      INTEGER {none(1),
58                     unknown(2),
59                     unauthorizedSs(3),
60

```

```

1          unsolicited(5),
2          invalidKeySequence(6),
3          keyRequestAuthenticationFailure(7)}
4
5      MAX-ACCESS    read-only
6      STATUS        current
7      DESCRIPTION
8          "The value of this object is the enumerated description of
9           the Error-Code in most recent Authorization Invalid message
10          received by the SS. This has value unknown(2) if the last
11          Error-Code value was 0, and none(1) if no Authorization
12          Invalid message has been received since reboot."
13      ::= { wmanIfSsPkmAuthEntry 13 }
14
15
16      wmanIfSsPkmAuthInvalidErrorString OBJECT-TYPE
17          SYNTAX      SnmpAdminString (SIZE (0..128))
18          MAX-ACCESS    read-only
19          STATUS        current
20          DESCRIPTION
21              "The value of this object is the Display-String in most
22               recent Authorization Invalid message received by the SS.
23               This is a zero length string if no Authorization Invalid
24               message has been received since reboot."
25          ::= { wmanIfSsPkmAuthEntry 14 }
26
27
28      wmanIfSsPkmAuthGraceTime OBJECT-TYPE
29          SYNTAX      Integer32 (300..3024000)
30          UNITS        "seconds"
31          MAX-ACCESS    read-only
32          STATUS        current
33          DESCRIPTION
34              "The value of this object is the grace time for an
35               authorization key. A SS is expected to start trying to get
36               a new authorization key beginning AuthGraceTime seconds
37               before the authorization key actually expires."
38          REFERENCE
39              "Table 341 in IEEE Std 802.16-2004"
40          DEFVAL       { 600 }
41          ::= { wmanIfSsPkmAuthEntry 15 }
42
43
44      wmanIfSsPkmTekGraceTime OBJECT-TYPE
45          SYNTAX      Integer32 (300..3024000)
46          UNITS        "seconds"
47          MAX-ACCESS    read-only
48          STATUS        current
49          DESCRIPTION
50              "The value of this object is the grace time for the TEK in
51               seconds. The SS is expected to start trying to acquire a
52               new TEK beginning TEK GraceTime seconds before the
53               expiration of the most recent TEK."
54          REFERENCE
55              "Table 341 in IEEE Std 802.16-2004"
56          DEFVAL       { 3600 }
57          ::= { wmanIfSsPkmAuthEntry 16 }
58
59
60
61
62
63
64
65

```

```

1  wmanIfSsPkmAuthWaitTimeout OBJECT-TYPE
2      SYNTAX      Integer32 (2..30)
3      UNITS       "seconds"
4      MAX-ACCESS  read-only
5      STATUS      current
6      DESCRIPTION
7          "The value of this object is the Authorize Wait Timeout."
8      REFERENCE
9          "Table 341 in IEEE Std 802.16-2004"
10     DEFVAL      { 10 }
11     ::= { wmanIfSsPkmAuthEntry 17 }
12
13  wmanIfSsPkmReauthWaitTimeout OBJECT-TYPE
14      SYNTAX      Integer32 (2..30)
15      UNITS       "seconds"
16      MAX-ACCESS  read-only
17      STATUS      current
18      DESCRIPTION
19          "The value of this object is the Reauthorize Wait Timeout
20          in seconds."
21      REFERENCE
22          "Table 341 in IEEE Std 802.16-2004"
23     DEFVAL      { 10 }
24     ::= { wmanIfSsPkmAuthEntry 18 }
25
26  wmanIfSsPkmOpWaitTimeout OBJECT-TYPE
27      SYNTAX      Integer32 (1..10)
28      UNITS       "seconds"
29      MAX-ACCESS  read-only
30      STATUS      current
31      DESCRIPTION
32          "The value of this object is the Operational Wait Timeout
33          in seconds."
34      REFERENCE
35          "Table 341 in IEEE Std 802.16-2004"
36     DEFVAL      { 1 }
37     ::= { wmanIfSsPkmAuthEntry 19 }
38
39  wmanIfSsPkmRekeyWaitTimeout OBJECT-TYPE
40      SYNTAX      Integer32 (1..10)
41      UNITS       "seconds"
42      MAX-ACCESS  read-only
43      STATUS      current
44      DESCRIPTION
45          "The value of this object is the Rekey Wait Timeout in
46          seconds."
47      REFERENCE
48          "Table 341 in IEEE Std 802.16-2004"
49     DEFVAL      { 1 }
50     ::= { wmanIfSsPkmAuthEntry 20 }
51
52  wmanIfSsPkmAuthRejectWaitTimeout OBJECT-TYPE
53      SYNTAX      Integer32 (10..600)
54      UNITS       "seconds"

```

```

1      MAX-ACCESS    read-only
2      STATUS        current
3      DESCRIPTION
4          "The value of this object is the Authorization Reject Wait
5          Timeout in seconds."
6      REFERENCE
7          "Table 341 in IEEE Std 802.16-2004"
8      DEFVAL        { 60 }
9      ::= { wmanIfSsPkmAuthEntry 21 }
10
11
12
13  --
14  -- Table wmanIfSsPkmTekTable
15  --
16  wmanIfSsPkmTekTable OBJECT-TYPE
17      SYNTAX          SEQUENCE OF WmanIfSsPkmTekEntry
18      MAX-ACCESS      not-accessible
19      STATUS          current
20      DESCRIPTION
21          "This table describes the attributes of each SS Traffic
22          Encryption Key (TEK) association. The SS maintains (no more
23          than) one TEK association per SAID per SS wireless
24          interface."
25      ::= { wmanIfSsPkmObjects 2 }
26
27
28
29  wmanIfSsPkmTekEntry OBJECT-TYPE
30      SYNTAX          WmanIfSsPkmTekEntry
31      MAX-ACCESS      not-accessible
32      STATUS          current
33      DESCRIPTION
34          "Each entry contains objects describing the TEK association
35          attributes of one SAID. The SS MUST create one entry per
36          SAID, regardless of whether the SAID was obtained from a
37          Registration Response message, from an Authorization Reply
38          message, or from any dynamic SAID establishment
39          mechanisms."
40      INDEX            { ifIndex, wmanIfSsPkmTekSAId }
41      ::= { wmanIfSsPkmTekTable 1 }
42
43
44
45  WmanIfSsPkmTekEntry ::= SEQUENCE {
46      wmanIfSsPkmTekSAId          INTEGER,
47      wmanIfSsPkmTekSAType        INTEGER,
48      wmanIfSsPkmTekDataEncryptAlg WmanIfDataEncryptAlgId,
49      wmanIfSsPkmTekDataAuthAlg   WmanIfDataAuthAlgId,
50      wmanIfSsPkmTekEncryptAlg    WmanIfTekEncryptAlgId,
51      wmanIfSsPkmTekState         INTEGER,
52      wmanIfSsPkmTekKeySequenceNumber Integer32,
53      wmanIfSsPkmTekExpiresOld    DateAndTime,
54      wmanIfSsPkmTekExpiresNew    DateAndTime,
55      wmanIfSsPkmTekKeyRequests   Counter32,
56      wmanIfSsPkmTekKeyReplies    Counter32,
57      wmanIfSsPkmTekKeyRejects    Counter32,
58      wmanIfSsPkmTekInvalids      Counter32,
59      wmanIfSsPkmTekAuthPends     Counter32,
60      wmanIfSsPkmTekKeyRejectErrorCode INTEGER,
61

```

```

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

```

13     ::= { wmanIfSsPkmTekEntry 1 }
14
15
16  wmanIfSsPkmTekSAType OBJECT-TYPE
17      SYNTAX      INTEGER {primarySA(0),
18                          staticSA(1),
19                          dynamicSA(2)}
20      MAX-ACCESS  read-only
21      STATUS      current
22      DESCRIPTION
23         "The value of this object is the type of security
24         association."
```

```

25     REFERENCE
26         "IEEE Std 802.16-2004; 11.9.18"
```

```

27     ::= { wmanIfSsPkmTekEntry 2 }
28
29
30  wmanIfSsPkmTekDataEncryptAlg OBJECT-TYPE
31      SYNTAX      WmanIfDataEncryptAlgId
32      MAX-ACCESS  read-only
33      STATUS      current
34      DESCRIPTION
35         "The value of this object is the data encryption algorithm
36         being utilized."
```

```

37     REFERENCE
38         "Table 375, IEEE Std 802.16-2004"
```

```

39     ::= { wmanIfSsPkmTekEntry 3 }
40
41
42  wmanIfSsPkmTekDataAuthentAlg OBJECT-TYPE
43      SYNTAX      WmanIfDataAuthAlgId
44      MAX-ACCESS  read-only
45      STATUS      current
46      DESCRIPTION
47         "The value of this object is the data authentication
48         algorithm being utilized."
```

```

49     REFERENCE
50         "Table 376, IEEE Std 802.16-2004"
```

```

51     ::= { wmanIfSsPkmTekEntry 4 }
52
53
54  wmanIfSsPkmTekEncryptAlg OBJECT-TYPE
55      SYNTAX      WmanIfTekEncryptAlgId
56      MAX-ACCESS  read-only
57      STATUS      current
58      DESCRIPTION
59         "The value of this object is the TEK key encryption
60         algorithm being utilized."
```

```

1         algorithm for this cryptographic suite capability."
2     REFERENCE
3         "Table 377, IEEE Std 802.16-2004"
4     ::= { wmanIfSsPkmTekEntry 5 }
5
6
7     wmanIfSsPkmTekState OBJECT-TYPE
8         SYNTAX      INTEGER {start(1),
9                        opWait(2),
10                       opReauthWait(3),
11                       operational(4),
12                       rekeyWait(5),
13                       rekeyReauthWait(6)}
14
15     MAX-ACCESS      read-only
16     STATUS          current
17     DESCRIPTION
18         "The value of this object is the state of the indicated TEK
19         FSM. The start(1) state indicates that FSM is in its
20         initial state."
21     ::= { wmanIfSsPkmTekEntry 6 }
22
23
24
25     wmanIfSsPkmTekKeySequenceNumber 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 TEK FSM."
32         REFERENCE
33             "IEEE Std 802.16-2004; 11.9.5"
34     ::= { wmanIfSsPkmTekEntry 7 }
35
36
37
38     wmanIfSsPkmTekExpiresOld 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     ::= { wmanIfSsPkmTekEntry 8 }
48
49
50
51     wmanIfSsPkmTekExpiresNew 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     ::= { wmanIfSsPkmTekEntry 9 }
59
60
61
62     wmanIfSsPkmTekKeyRequests 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              transmitted a Key Request message."
5      ::= { wmanIfSsPkmTekEntry 10 }
6
7
8      wmanIfSsPkmTekKeyReplies OBJECT-TYPE
9          SYNTAX      Counter32
10         MAX-ACCESS  read-only
11         STATUS      current
12         DESCRIPTION
13             "The value of this object is the count of times the SS has
14                 received a Key Reply message, including a message whose
15                 authentication failed."
16         ::= { wmanIfSsPkmTekEntry 11 }
17
18
19
20     wmanIfSsPkmTekKeyRejects OBJECT-TYPE
21         SYNTAX      Counter32
22         MAX-ACCESS  read-only
23         STATUS      current
24         DESCRIPTION
25             "The value of this object is the count of times the SS has
26                 received a Key Reject message, including a message whose
27                 authentication failed."
28         ::= { wmanIfSsPkmTekEntry 12 }
29
30
31
32     wmanIfSsPkmTekInvalids OBJECT-TYPE
33         SYNTAX      Counter32
34         MAX-ACCESS  read-only
35         STATUS      current
36         DESCRIPTION
37             "The value of this object is the count of times the SS has
38                 received a TEK Invalid message, including a message whose
39                 authentication failed."
40         ::= { wmanIfSsPkmTekEntry 13 }
41
42
43
44     wmanIfSsPkmTekAuthPends OBJECT-TYPE
45         SYNTAX      Counter32
46         MAX-ACCESS  read-only
47         STATUS      current
48         DESCRIPTION
49             "The value of this object is the count of times an
50                 Authorization Pending (Auth Pend) event occurred in this
51                 FSM."
52         ::= { wmanIfSsPkmTekEntry 14 }
53
54
55
56     wmanIfSsPkmTekKeyRejectErrorCode OBJECT-TYPE
57         SYNTAX      INTEGER { none(1),
58                               unknown(2),
59                               unauthorizedSaid(4) }
60         MAX-ACCESS  read-only
61         STATUS      current
62         DESCRIPTION
63             "The value of this object is the enumerated description of
64
65

```

```

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

```

```

1  wmanIfSsDeviceCertEntry OBJECT-TYPE
2      SYNTAX      WmanIfSsDeviceCertEntry
3      MAX-ACCESS  not-accessible
4      STATUS      current
5      DESCRIPTION
6          "Each entry contains the device certificate of one SS."
7      INDEX      { ifIndex }
8      ::= { wmanIfSsDeviceCertTable 1 }
9
10
11  WmanIfSsDeviceCertEntry ::= SEQUENCE {
12      wmanIfSsDeviceCert          OCTET STRING,
13      wmanIfSsDeviceManufCert     OCTET STRING}
14
15
16  wmanIfSsDeviceCert OBJECT-TYPE
17      SYNTAX      OCTET STRING (SIZE(0..65535))
18      MAX-ACCESS  read-only
19      STATUS      current
20      DESCRIPTION
21          "The X509 DER-encoded subscriber station certificate."
22      ::= { wmanIfSsDeviceCertEntry 1 }
23
24
25
26  wmanIfSsDeviceManufCert OBJECT-TYPE
27      SYNTAX      OCTET STRING (SIZE(0..65535))
28      MAX-ACCESS  read-only
29      STATUS      current
30      DESCRIPTION
31          "The X509 DER-encoded manufacturer certificate which is
32          signed by the CA root authority certificate."
33      ::= { wmanIfSsDeviceCertEntry 2 }
34
35
36
37  --
38  -- Subscriber station Notification Group
39  -- wmanIfSsNotificationObjects contains the SS SNMP Trap objects
40  --
41
42  wmanIfSsNotification OBJECT IDENTIFIER ::= { wmanIfSsObjects 3 }
43  wmanIfSsTrapControl OBJECT IDENTIFIER ::= { wmanIfSsNotification 1 }
44  wmanIfSsTrapDefinitions OBJECT IDENTIFIER ::= { wmanIfSsNotification 2 }
45
46
47  -- This object groups all NOTIFICATION-TYPE objects for SS.
48  -- It is defined following RFC2758 sections 8.5 and 8.6
49  -- for the compatibility with SNMPv1.
50  wmanIfSsTrapPrefix OBJECT IDENTIFIER ::= { wmanIfSsTrapDefinitions 0 }
51
52
53  wmanIfSsTrapControlRegister OBJECT-TYPE
54      SYNTAX      BITS {wmanIfSsTlvUnknown(0),
55                      wmanIfSsDynamicServiceFail(1),
56                      wmanIfSsDhcpSuccess(2),
57                      wmanIfSsRssiStatusChange(3)}
58      MAX-ACCESS  read-write
59      STATUS      current
60      DESCRIPTION
61          "The object is used to enable Subscriber Station traps.
62          From left to right, the set bit indicates the corresponding
63          Subscriber Station trap is enabled."
64
65

```

```

1      ::= { wmanIfSsTrapControl 1 }
2
3
4  wmanIfSsThresholdConfigTable OBJECT-TYPE
5      SYNTAX      SEQUENCE OF WmanIfSsThresholdConfigEntry
6      MAX-ACCESS  not-accessible
7      STATUS      current
8      DESCRIPTION
9          "This table contains threshold objects that can be set to
10         detect the threshold crossing events."
11      ::= { wmanIfSsTrapControl 2 }
12
13
14  wmanIfSsThresholdConfigEntry OBJECT-TYPE
15      SYNTAX      WmanIfSsThresholdConfigEntry
16      MAX-ACCESS  not-accessible
17      STATUS      current
18      DESCRIPTION
19          "This table provides one row for each Ss, and is indexed
20         by ifIndex."
21      INDEX      { ifIndex }
22      ::= { wmanIfSsThresholdConfigTable 1 }
23
24
25
26  WmanIfSsThresholdConfigEntry ::= SEQUENCE {
27      wmanIfSsRssiLowThreshold      Integer32,
28      wmanIfSsRssiHighThreshold     Integer32}
29
30
31  wmanIfSsRssiLowThreshold OBJECT-TYPE
32      SYNTAX      Integer32
33      UNITS       "dBm"
34      MAX-ACCESS  read-write
35      STATUS      current
36      DESCRIPTION
37          "Low RSSI threshold for generating the RSSI alarm trap."
38      ::= { wmanIfSsThresholdConfigEntry 1 }
39
40
41
42  wmanIfSsRssiHighThreshold OBJECT-TYPE
43      SYNTAX      Integer32
44      UNITS       "dBm"
45      MAX-ACCESS  read-write
46      STATUS      current
47      DESCRIPTION
48          "High RSSI threshold for generating a trap to indicate
49         the RSSI is restored."
50      ::= { wmanIfSsThresholdConfigEntry 2 }
51
52
53
54  wmanIfSsTlvUnknownTrap NOTIFICATION-TYPE
55      OBJECTS      {ifIndex,
56                  wmanIfSsMacAddress,
57                  wmanIfSsUnknownTlv}
58      STATUS      current
59      DESCRIPTION
60          "Event that notifies detection of unknown TLV during
61         the TLV parsing process."
62      ::= { wmanIfSsTrapPrefix 1 }
63
64
65

```

```

1  wmanIfSsDynamicServiceFailTrap NOTIFICATION-TYPE
2      OBJECTS      {ifIndex,
3                    wmanIfSsMacAddress,
4                    wmanIfSsDynamicServiceType,
5                    wmanIfSsDynamicServiceFailReason}
6
7      STATUS      current
8
9      DESCRIPTION
10         "An event to report the failure of a dynamic service
11         operation happened during the dynamic services process
12         and detected in the BS side."
13         ::= { wmanIfSsTrapPrefix 2 }
14
15  wmanIfSsDhcpSuccessTrap NOTIFICATION-TYPE
16      OBJECTS      {ifIndex,
17                    wmanIfSsMacAddress}
18
19      STATUS      current
20
21      DESCRIPTION
22         "An event to report a successful Handshake to establish IP
23         connectivity."
24         ::= { wmanIfSsTrapPrefix 3 }
25
26  wmanIfSsRssiStatusChangeTrap NOTIFICATION-TYPE
27      OBJECTS      {ifIndex,
28                    wmanIfSsMacAddress,
29                    wmanIfSsRssiStatus,
30                    wmanIfSsRssiStatusInfo}
31
32      STATUS      current
33
34      DESCRIPTION
35         "An event to report that the downlink RSSI is below
36         wmanIfSsRssiLowThreshold, or above
37         wmanIfSsRssiHighThreshold after restore."
38         ::= { wmanIfSsTrapPrefix 4 }
39
40
41  wmanIfSsNotificationObjectsTable OBJECT-TYPE
42      SYNTAX      SEQUENCE OF WmanIfSsNotificationObjectsEntry
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         ::= { wmanIfSsTrapDefinitions 1 }
49
50
51  wmanIfSsNotificationObjectsEntry OBJECT-TYPE
52      SYNTAX      WmanIfSsNotificationObjectsEntry
53      MAX-ACCESS  not-accessible
54      STATUS      current
55      DESCRIPTION
56         "This table provides one row for each SS that has
57         generated traps, and is indexed by ifIndex."
58      INDEX      { ifIndex }
59      ::= { wmanIfSsNotificationObjectsTable 1 }
60
61
62  WmanIfSsNotificationObjectsEntry ::= SEQUENCE {
63      wmanIfSsMacAddress      MacAddress,
64
65

```

```

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

```

```

1      SYNTAX      OCTET STRING (SIZE(0..255))
2      MAX-ACCESS  read-only
3      STATUS      current
4      DESCRIPTION
5          "This object provides additional information about RSSI
6          alarm. It is implementation specific"
7      ::= { wmanIfSsNotificationObjectsEntry 6 }
8
9
10     --
11     -- Subscriber station PHY Group
12     --
13     wmanIfSsPhy OBJECT IDENTIFIER ::= { wmanIfSsObjects 5 }
14
15     --
16     -- SS OFDM PHY objects
17     --
18     wmanIfSsOfdmPhy OBJECT IDENTIFIER ::= { wmanIfSsPhy 1 }
19
20     wmanIfSsOfdmUplinkChannelTable OBJECT-TYPE
21         SYNTAX      SEQUENCE OF WmanIfSsOfdmUplinkChannelEntry
22         MAX-ACCESS  not-accessible
23         STATUS      current
24         DESCRIPTION
25             "This table contains UCD channel attributes, defining the
26             transmission characteristics of uplink channels"
27         REFERENCE
28             "Table 349 and Table 352, in IEEE Std 802.16-2004"
29         ::= { wmanIfSsOfdmPhy 1 }
30
31     wmanIfSsOfdmUplinkChannelEntry OBJECT-TYPE
32         SYNTAX      WmanIfSsOfdmUplinkChannelEntry
33         MAX-ACCESS  not-accessible
34         STATUS      current
35         DESCRIPTION
36             "This table provides one row for each uplink channel of
37             multi-sector BS, and is indexed by BS ifIndex. An entry
38             in this table exists for each ifEntry of BS with an
39             ifType of propBWAp2Mp."
40         INDEX { ifIndex }
41         ::= { wmanIfSsOfdmUplinkChannelTable 1 }
42
43     WmanIfSsOfdmUplinkChannelEntry ::= SEQUENCE {
44         wmanIfSsOfdmCtBasedResvTimeout      INTEGER,
45         wmanIfSsOfdmBwReqOppSize             INTEGER,
46         wmanIfSsOfdmRangReqOppSize           INTEGER,
47         wmanIfSsOfdmUplinkCenterFreq        Unsigned32,
48         wmanIfSsOfdmNumSubChReqRegionFull    INTEGER,
49         wmanIfSsOfdmNumSymbolsReqRegionFull  INTEGER,
50         wmanIfSsOfdmSubChFocusCtCode         INTEGER,
51         wmanIfSsOfdmUpLinkChannelId          INTEGER}
52
53     wmanIfSsOfdmCtBasedResvTimeout OBJECT-TYPE
54         SYNTAX      INTEGER (1..255)
55         MAX-ACCESS  read-only
56

```

```

1      STATUS      current
2      DESCRIPTION
3          "The number of UL-MAPs to receive before contention-based
4          reservation is attempted again for the same connection."
5      REFERENCE
6          "Table 349, in IEEE Std 802.16-2004"
7          ::= { wmanIfSsOfdmUplinkChannelEntry 1 }
8
9
10     wmanIfSsOfdmBwReqOppSize 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             ::= { wmanIfSsOfdmUplinkChannelEntry 2 }
24
25     wmanIfSsOfdmRangReqOppSize OBJECT-TYPE
26         SYNTAX      INTEGER (1..65535)
27         UNITS        "PS"
28         MAX-ACCESS   read-only
29         STATUS       current
30         DESCRIPTION
31             "Size (in units of PS) of PHY payload that SS may use to
32             format and transmit a RNG-REQ message in a contention
33             request opportunity. The value includes all PHY overhead
34             as well as allowance for the MAC data the message may
35             hold and the maximum SS/BS roundtrip propagation delay."
36         REFERENCE
37             "Table 349, in IEEE Std 802.16-2004"
38             ::= { wmanIfSsOfdmUplinkChannelEntry 3 }
39
40     wmanIfSsOfdmUplinkCenterFreq OBJECT-TYPE
41         SYNTAX      Unsigned32
42         UNITS        "kHz"
43         MAX-ACCESS   read-only
44         STATUS       current
45         DESCRIPTION
46             " Uplink center frequency (kHz)"
47         REFERENCE
48             "Table 349, in IEEE Std 802.16-2004"
49             ::= { wmanIfSsOfdmUplinkChannelEntry 4 }
50
51     wmanIfSsOfdmNumSubChReqRegionFull OBJECT-TYPE
52         SYNTAX      INTEGER {oneSubchannel(0),
53                             twoSubchannels(1),
54                             fourSubchannels(2),
55                             eightSubchannels(3),

```

```

1                               sixteenSubchannels(4) }
2   MAX-ACCESS    read-only
3   STATUS        current
4   DESCRIPTION
5       "Number of subchannels used by each transmit
6       opportunity when REQ Region-Full is allocated in
7       subchannelization region."
8   REFERENCE
9       "Table 352, in IEEE Std 802.16-2004"
10  ::= { wmanIfSsOfdmUplinkChannelEntry 5 }
11
12  wmanIfSsOfdmNumSymbolsReqRegionFull OBJECT-TYPE
13      SYNTAX      INTEGER (0..31)
14      MAX-ACCESS  read-only
15      STATUS      current
16      DESCRIPTION
17          "Number of OFDM symbols used by each transmit
18          opportunity when REQ Region-Full is allocated in
19          subchannelization region."
20      REFERENCE
21          "Table 352, in IEEE Std 802.16-2004"
22      ::= { wmanIfSsOfdmUplinkChannelEntry 6 }
23
24  wmanIfSsOfdmSubChFocusCtCode OBJECT-TYPE
25      SYNTAX      INTEGER (0..8)
26      MAX-ACCESS  read-only
27      STATUS      current
28      DESCRIPTION
29          "Number of contention codes (CSE) that shall only be used to
30          request a subchannelized allocation. Default value 0.
31          Allowed values 0-8."
32      REFERENCE
33          "Table 352, in IEEE Std 802.16-2004"
34      DEFVAL      { 0 }
35      ::= { wmanIfSsOfdmUplinkChannelEntry 7 }
36
37  wmanIfSsOfdmUpLinkChannelId OBJECT-TYPE
38      SYNTAX      INTEGER (0..255)
39      MAX-ACCESS  read-only
40      STATUS      current
41      DESCRIPTION
42          "The identifier of the uplink channel to which this
43          message refers."
44      REFERENCE
45          "Subclause 6.3.2.3.4. Table 16, in IEEE Std 802.16-2004"
46      ::= { wmanIfSsOfdmUplinkChannelEntry 8 }
47
48  wmanIfSsOfdmDownlinkChannelTable OBJECT-TYPE
49      SYNTAX      SEQUENCE OF WmanIfSsOfdmDownlinkChannelEntry
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
57
58
59
60
61
62
63
64
65

```

```

1      REFERENCE
2          "Table 358, in IEEE Std 802.16-2004"
3      ::= { wmanIfSsOfdmPhy 2 }
4
5
6  wmanIfSsOfdmDownlinkChannelEntry OBJECT-TYPE
7      SYNTAX      WmanIfSsOfdmDownlinkChannelEntry
8      MAX-ACCESS  not-accessible
9      STATUS      current
10     DESCRIPTION
11         "This table provides one row for each downlink channel of
12         multi-sector BS, and is indexed by BS ifIndex. An entry
13         in this table exists for each ifEntry of BS with an
14         ifType of propBWA2Mp."
15     INDEX { ifIndex }
16     ::= { wmanIfSsOfdmDownlinkChannelTable 1 }
17
18
19
20  WmanIfSsOfdmDownlinkChannelEntry ::= SEQUENCE {
21      wmanIfSsOfdmBsEIRP                INTEGER,
22      wmanIfSsOfdmChannelNumber          WmanIfChannelNumber,
23      wmanIfSsOfdmTTG                    INTEGER,
24      wmanIfSsOfdmRTG                    INTEGER,
25      wmanIfSsOfdmInitRngMaxRSS          INTEGER,
26      wmanIfSsOfdmDownlinkCenterFreq    Unsigned32,
27      wmanIfSsOfdmBsId                  WmanIfBsIdType,
28      wmanIfSsOfdmMacVersion              WmanIfMacVersion,
29      wmanIfSsOfdmFrameDurationCode      INTEGER,
30      wmanIfSsOfdmDownLinkChannelId      INTEGER}
31
32
33
34
35  wmanIfSsOfdmBsEIRP OBJECT-TYPE
36      SYNTAX      INTEGER (0..65535)
37      UNITS       "dBm"
38      MAX-ACCESS  read-only
39      STATUS      current
40      DESCRIPTION
41          "The EIRP is the equivalent isotropic radiated power of
42          the base station, which is computed for a simple
43          single-antenna transmitter."
44      REFERENCE
45          "Table 358, in IEEE Std 802.16-2004"
46      ::= { wmanIfSsOfdmDownlinkChannelEntry 1 }
47
48
49
50  wmanIfSsOfdmChannelNumber OBJECT-TYPE
51      SYNTAX      WmanIfChannelNumber
52      MAX-ACCESS  read-only
53      STATUS      current
54      DESCRIPTION
55          "Downlink channel number as defined in 8.5.
56          Used for license-exempt operation only."
57      REFERENCE
58          "Table 358, in IEEE Std 802.16-2004"
59      ::= { wmanIfSsOfdmDownlinkChannelEntry 2 }
60
61
62
63  wmanIfSsOfdmTTG OBJECT-TYPE
64      SYNTAX      INTEGER (0..255)
65

```

```

1      MAX-ACCESS  read-only
2      STATUS      current
3      DESCRIPTION
4          "Transmit / Receive Transition Gap."
5      REFERENCE
6          "Table 358, in IEEE Std 802.16-2004"
7      ::= { wmanIfSsOfdmDownlinkChannelEntry 3 }
8
9
10
11  wmanIfSsOfdmRTG OBJECT-TYPE
12      SYNTAX      INTEGER (0..255)
13      MAX-ACCESS  read-only
14      STATUS      current
15      DESCRIPTION
16          "Receive / Transmit Transition Gap."
17      REFERENCE
18          "Table 358, in IEEE Std 802.16-2004"
19      ::= { wmanIfSsOfdmDownlinkChannelEntry 4 }
20
21
22
23  wmanIfSsOfdmInitRngMaxRSS OBJECT-TYPE
24      SYNTAX      INTEGER (0..65535)
25      UNITS       "dBm"
26      MAX-ACCESS  read-only
27      STATUS      current
28      DESCRIPTION
29          "Initial Ranging Max. Received Signal Strength at BS
30           Signed in units of 1 dBm."
31      REFERENCE
32          "Table 358, in IEEE Std 802.16-2004"
33      ::= { wmanIfSsOfdmDownlinkChannelEntry 5 }
34
35
36
37  wmanIfSsOfdmDownlinkCenterFreq OBJECT-TYPE
38      SYNTAX      Unsigned32
39      UNITS       "kHz"
40      MAX-ACCESS  read-only
41      STATUS      current
42      DESCRIPTION
43          "Downlink center frequency (kHz)."

```

```

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

```

```

1         ifIndex with an ifType of propBWA2Mp. The secondary index
2         is wmanIfSsOfdmOfdmUcdBurstProfIndex."
3     INDEX { ifIndex, wmanIfSsOfdmUiucIndex }
4     ::= { wmanIfSsOfdmUcdBurstProfileTable 1 }
5
6
7     WmanIfSsOfdmUcdBurstProfileEntry ::= SEQUENCE {
8         wmanIfSsOfdmUiucIndex          INTEGER,
9         wmanIfSsOfdmUcdFecCodeType     WmanIfOfdmFecCodeType,
10        wmanIfSsOfdmFocusCtPowerBoost   INTEGER,
11        wmanIfSsOfdmUcdTcsEnable        INTEGER}
12
13
14    wmanIfSsOfdmUiucIndex OBJECT-TYPE
15        SYNTAX      INTEGER (5 .. 12)
16        MAX-ACCESS   not-accessible
17        STATUS       current
18        DESCRIPTION
19            "The Uplink Interval Usage Code indicates the uplink burst
20             profile in the UCD message, and is used along with ifIndex
21             to identify an entry in the
22             wmanIfSsOfdmUcdBurstProfileTable."
23        REFERENCE
24            "Subclause 8.3.6.3.1, in IEEE Std 802.16-2004"
25        ::= { wmanIfSsOfdmUcdBurstProfileEntry 1 }
26
27
28    wmanIfSsOfdmUcdFecCodeType OBJECT-TYPE
29        SYNTAX      WmanIfOfdmFecCodeType
30        MAX-ACCESS   read-only
31        STATUS       current
32        DESCRIPTION
33            "Uplink FEC code type and modulation type"
34        REFERENCE
35            "Table 356, in IEEE Std 802.16-2004"
36        ::= { wmanIfSsOfdmUcdBurstProfileEntry 2 }
37
38
39    wmanIfSsOfdmFocusCtPowerBoost OBJECT-TYPE
40        SYNTAX      INTEGER (0 .. 255)
41        MAX-ACCESS   read-only
42        STATUS       current
43        DESCRIPTION
44            "The power boost in dB of focused contention carriers, as
45             described in 8.3.6.3.3."
46        REFERENCE
47            "Table 356, in IEEE Std 802.16-2004"
48        ::= { wmanIfSsOfdmUcdBurstProfileEntry 3 }
49
50
51    wmanIfSsOfdmUcdTcsEnable OBJECT-TYPE
52        SYNTAX      INTEGER {tcsDisabled(0),
53                             tcsEnabled(1)}
54        MAX-ACCESS   read-only
55        STATUS       current
56        DESCRIPTION
57            "This parameter determines the transmission convergence
58             sublayer, as described in 8.1.4.3, can be enabled on a
59             per-burst basis for both uplink and downlink. through

```

```

1          DIUC/UIUC messages."
2  REFERENCE
3      "Table 356, in IEEE Std 802.16-2004"
4      ::= { wmanIfSsOfdmUcdBurstProfileEntry 4 }
5
6
7  wmanIfSsOfdmDcdBurstProfileTable OBJECT-TYPE
8      SYNTAX      SEQUENCE OF WmanIfSsOfdmDcdBurstProfileEntry
9      MAX-ACCESS  not-accessible
10     STATUS      current
11     DESCRIPTION
12         "This table provides one row for each DCD burst profile.
13         This table is double indexed. The primary index is an
14         ifIndex with an ifType of propBWAp2Mp. The secondary
15         index is wmanIfSsOfdmDiucIndex."
16     REFERENCE
17         "Table 362, in IEEE Std 802.16-2004"
18         ::= { wmanIfSsOfdmPhy 4 }
19
20
21
22
23  wmanIfSsOfdmDcdBurstProfileEntry OBJECT-TYPE
24      SYNTAX      WmanIfSsOfdmDcdBurstProfileEntry
25      MAX-ACCESS  not-accessible
26      STATUS      current
27      DESCRIPTION
28          "This table provides one row for each DCD burst profile.
29          This table is double indexed. The primary index is an
30          ifIndex with an ifType of propBWAp2Mp. The secondary index
31          is wmanIfSsOfdmDcdBurstProfIndex."
32      INDEX { ifIndex, wmanIfSsOfdmDiucIndex }
33      ::= { wmanIfSsOfdmDcdBurstProfileTable 1 }
34
35
36
37  WmanIfSsOfdmDcdBurstProfileEntry ::= SEQUENCE {
38      wmanIfSsOfdmDiucIndex          INTEGER,
39      wmanIfSsOfdmDownlinkFrequency  Unsigned32,
40      wmanIfSsOfdmDcdFecCodeType     WmanIfOfdmFecCodeType,
41      wmanIfSsOfdmDiucMandatoryExitThresh  INTEGER,
42      wmanIfSsOfdmDiucMinEntryThresh  INTEGER,
43      wmanIfSsOfdmTcsEnable          INTEGER}
44
45
46
47  wmanIfSsOfdmDiucIndex OBJECT-TYPE
48      SYNTAX      INTEGER (1..11)
49      MAX-ACCESS  not-accessible
50      STATUS      current
51      DESCRIPTION
52          "The Downlink Interval Usage Code indicates the downlink
53          burst profile in the DCD message, and is used along with
54          ifIndex to identify an entry in the
55          wmanIfSsOfdmDcdBurstProfileTable."
56      REFERENCE
57          "Subclause 8.3.6.3.1, in IEEE Std 802.16-2004"
58          ::= { wmanIfSsOfdmDcdBurstProfileEntry 1 }
59
60
61
62  wmanIfSsOfdmDownlinkFrequency OBJECT-TYPE
63      SYNTAX      Unsigned32
64      UNITS       "kHz"
65

```

```

1      MAX-ACCESS    read-only
2      STATUS        current
3      DESCRIPTION
4          "Downlink Frequency (kHz)."
```

REFERENCE

```

7          "Table 362, in IEEE Std 802.16-2004"
8      ::= { wmanIfSsOfdmDcdBurstProfileEntry 2 }
```

wmanIfSsOfdmDcdFecCodeType OBJECT-TYPE

```

11     SYNTAX          WmanIfOfdmFecCodeType
12     MAX-ACCESS      read-only
13     STATUS          current
14     DESCRIPTION
15         "Downlink FEC code type and modulation type"
```

REFERENCE

```

18     "Table 362, in IEEE Std 802.16-2004"
19     ::= { wmanIfSsOfdmDcdBurstProfileEntry 3 }
```

wmanIfSsOfdmDiucMandatoryExitThresh OBJECT-TYPE

```

22     SYNTAX          INTEGER (0..255)
23     MAX-ACCESS      read-only
24     STATUS          current
25     DESCRIPTION
26         "DIUC mandatory exit threshold: 0 - 63.75 dB CINR at or
27         below where this DIUC can no longer be used and where this
28         change to a more robust DIUC is required in 0.25 dB units."
```

REFERENCE

```

31     "Table 362, in IEEE Std 802.16-2004"
32     ::= { wmanIfSsOfdmDcdBurstProfileEntry 4 }
```

wmanIfSsOfdmDiucMinEntryThresh OBJECT-TYPE

```

35     SYNTAX          INTEGER (0..255)
36     MAX-ACCESS      read-only
37     STATUS          current
38     DESCRIPTION
39         "DIUC minimum entry threshold: 0 - 63.75 dB The minimum CINR
40         required to start using this DIUC when changing from a more
41         robust DIUC is required, in 0.25 dB units."
```

REFERENCE

```

44     "Table 362, in IEEE Std 802.16-2004"
45     ::= { wmanIfSsOfdmDcdBurstProfileEntry 5 }
```

wmanIfSsOfdmTcsEnable OBJECT-TYPE

```

48     SYNTAX          INTEGER {tcsDisabled (0),
49                        tcsEnabled (1)}
50     MAX-ACCESS      read-only
51     STATUS          current
52     DESCRIPTION
53         "Indicates whether Transmission Convergence Sublayer
54         is enabled or disabled."
```

REFERENCE

```

57     "Table 362, in IEEE Std 802.16-2004"
58     ::= { wmanIfSsOfdmDcdBurstProfileEntry 6 }
```

```

1  --
2  -- SS OFDMA PHY objects
3  --
4
5  wmanIfSsOfdmaPhy OBJECT IDENTIFIER ::= { wmanIfSsPhy 2 }
6
7  wmanIfSsOfdmaUplinkChannelTable OBJECT-TYPE
8      SYNTAX      SEQUENCE OF WmanIfSsOfdmaUplinkChannelEntry
9      MAX-ACCESS  not-accessible
10     STATUS      current
11     DESCRIPTION
12         "This table contains UCD channel attributes, defining the
13         transmission characteristics of uplink channels"
14     REFERENCE
15         "Subclause 11.3.1, Table 349 and Table 353, in IEEE Std
16     802.16-2004"
17     ::= { wmanIfSsOfdmaPhy 1 }
18
19  wmanIfSsOfdmaUplinkChannelEntry OBJECT-TYPE
20      SYNTAX      WmanIfSsOfdmaUplinkChannelEntry
21      MAX-ACCESS  not-accessible
22      STATUS      current
23      DESCRIPTION
24          "This table provides one row for each uplink channel of
25          multi-sector BS, and is indexed by BS ifIndex. An entry
26          in this table exists for each ifEntry of BS with an
27          ifType of propBWAp2Mp."
28      INDEX      { ifIndex }
29      ::= { wmanIfSsOfdmaUplinkChannelTable 1 }
30
31  WmanIfSsOfdmaUplinkChannelEntry ::= SEQUENCE {
32      wmanIfSsOfdmaCtBasedResvTimeout      INTEGER,
33      wmanIfSsOfdmaBwReqOppSize             INTEGER,
34      wmanIfSsOfdmaRangReqOppSize           INTEGER,
35      wmanIfSsOfdmaUplinkCenterFreq         Unsigned32,
36      wmanIfSsOfdmaInitRngCodes             INTEGER,
37      wmanIfSsOfdmaPeriodicRngCodes        INTEGER,
38      wmanIfSsOfdmaBWReqCodes              INTEGER,
39      wmanIfSsOfdmaPerRngBackoffStart       INTEGER,
40      wmanIfSsOfdmaPerRngBackoffEnd        INTEGER,
41      wmanIfSsOfdmaStartOfRngCodes         INTEGER,
42      wmanIfSsOfdmaPermutationBase         INTEGER,
43      wmanIfSsOfdmaULAllocSubchBitmap      OCTET STRING,
44      wmanIfSsOfdmaOptPermULAllocSubchBitmap OCTET STRING,
45      wmanIfSsOfdmaBandAMCAllocThreshold   INTEGER,
46      wmanIfSsOfdmaBandAMCReleaseThreshold INTEGER,
47      wmanIfSsOfdmaBandAMCAllocTimer       INTEGER,
48      wmanIfSsOfdmaBandAMCReleaseTimer     INTEGER,
49      wmanIfSsOfdmaBandStatRepMAXPeriod    INTEGER,
50      wmanIfSsOfdmaBandAMCRetryTimer       INTEGER,
51      wmanIfSsOfdmaSafetyChAllocThreshold  INTEGER,
52      wmanIfSsOfdmaSafetyChReleaseThreshold INTEGER,
53      wmanIfSsOfdmaSafetyChAllocTimer      INTEGER,
54      wmanIfSsOfdmaSafetyChReleaseTimer    INTEGER,
55      wmanIfSsOfdmaBinStatRepMAXPeriod     INTEGER,
56  }

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

1
2 wmanIfSsOfdmaHARQAackDelayULBurst OBJECT-TYPE
3     SYNTAX      INTEGER {oneframeoffset(1),
4                   twoframesoffset(2),
5                   threeframesoffset(3)}
6
7     MAX-ACCESS  read-only
8     STATUS      current
9     DESCRIPTION
10        "This object defines the OFDMA H-ARQ ACK delay for UL burst.
11          1 = one frame offset
12          2 = two frames offset
13          3 = three frames offset"
14
15     REFERENCE
16        "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
17        ::= { wmanIfSsOfdmaUplinkChannelEntry 26 }
18
19
20 wmanIfSsOfdmaCQICHBandAMCTranaDelay 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 CQICH band AMC transition
27          delay."
28
29     REFERENCE
30        "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
31        ::= { wmanIfSsOfdmaUplinkChannelEntry 27 }
32
33
34 wmanIfSsOfdmaDownlinkChannelTable OBJECT-TYPE
35     SYNTAX      SEQUENCE OF WmanIfSsOfdmaDownlinkChannelEntry
36     MAX-ACCESS  not-accessible
37     STATUS      current
38     DESCRIPTION
39        "This table contains DCD channel attributes, defining the
40          transmission characteristics of downlink channels"
41
42     REFERENCE
43        "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
44        ::= { wmanIfSsOfdmaPhy 2 }
45
46
47 wmanIfSsOfdmaDownlinkChannelEntry OBJECT-TYPE
48     SYNTAX      WmanIfSsOfdmaDownlinkChannelEntry
49     MAX-ACCESS  not-accessible
50     STATUS      current
51     DESCRIPTION
52        "This table provides one row for each downlink channel of
53          multi-sector BS, and is indexed by BS ifIndex. An entry in
54          this table exists for each ifEntry of BS with an ifType of
55          propBWA2Mp."
56
57     INDEX       { ifIndex }
58     ::= { wmanIfSsOfdmaDownlinkChannelTable 1 }
59
60
61 WmanIfSsOfdmaDownlinkChannelEntry ::= SEQUENCE {
62     wmanIfSsOfdmaBsEIRP          INTEGER,
63     wmanIfSsOfdmaChannelNumber   WmanIfChannelNumber,
64
65

```

```

1      wmanIfSsOfdmaTTG                      INTEGER,
2      wmanIfSsOfdmaRTG                      INTEGER,
3      wmanIfSsOfdmaInitRngMaxRSS            INTEGER,
4      wmanIfSsOfdmaDownlinkCenterFreq       Unsigned32,
5      wmanIfSsOfdmaBsId                    WmanIfBsIdType,
6      wmanIfSsOfdmaMacVersion               WmanIfMacVersion,
7      wmanIfSsOfdmaFrameDurationCode        INTEGER,
8      wmanIfSsOfdmaSizeCqichIdField         INTEGER,
9      wmanIfSsOfdmaHARQAckDelayBurst        INTEGER}
10
11
12
13  wmanIfSsOfdmaBsEIRP OBJECT-TYPE
14      SYNTAX      INTEGER (0..65535)
15      UNITS       "dBm"
16      MAX-ACCESS  read-only
17      STATUS      current
18      DESCRIPTION
19          "The EIRP is the equivalent isotropic radiated power of
20           the base station, which is computed for a simple
21           single-antenna transmitter."
22      REFERENCE
23          "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
24      ::= { wmanIfSsOfdmaDownlinkChannelEntry 1 }
25
26
27
28  wmanIfSsOfdmaChannelNumber OBJECT-TYPE
29      SYNTAX      WmanIfChannelNumber
30      MAX-ACCESS  read-only
31      STATUS      current
32      DESCRIPTION
33          "Downlink channel number as defined in 8.5. Used for
34           license-exempt operation only."
35      REFERENCE
36          "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
37      ::= { wmanIfSsOfdmaDownlinkChannelEntry 2 }
38
39
40
41  wmanIfSsOfdmaTTG OBJECT-TYPE
42      SYNTAX      INTEGER (0..255)
43      MAX-ACCESS  read-only
44      STATUS      current
45      DESCRIPTION
46          "Transmit / Receive Transition Gap."
47      REFERENCE
48          "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
49      ::= { wmanIfSsOfdmaDownlinkChannelEntry 3 }
50
51
52
53  wmanIfSsOfdmaRTG OBJECT-TYPE
54      SYNTAX      INTEGER (0..255)
55      MAX-ACCESS  read-only
56      STATUS      current
57      DESCRIPTION
58          "Receive / Transmit Transition Gap."
59      REFERENCE
60          "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
61      ::= { wmanIfSsOfdmaDownlinkChannelEntry 4 }
62
63
64
65

```

```

1  wmanIfSsOfdmaInitRngMaxRSS OBJECT-TYPE
2      SYNTAX      INTEGER (0..65535)
3      UNITS       "dBm"
4      MAX-ACCESS  read-only
5      STATUS      current
6      DESCRIPTION
7          "Initial Ranging Max. Received Signal Strength at BS
8           Signed in units of 1 dBm."
9      REFERENCE
10         "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
11         ::= { wmanIfSsOfdmaDownlinkChannelEntry 5 }
12
13  wmanIfSsOfdmaDownlinkCenterFreq OBJECT-TYPE
14      SYNTAX      Unsigned32
15      UNITS       "kHz"
16      MAX-ACCESS  read-only
17      STATUS      current
18      DESCRIPTION
19          "Downlink center frequency (kHz)."
```

```

20      REFERENCE
21         "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
22         ::= { wmanIfSsOfdmaDownlinkChannelEntry 6 }
23
24  wmanIfSsOfdmaBsId OBJECT-TYPE
25      SYNTAX      WmanIfBsIdType
26      MAX-ACCESS  read-only
27      STATUS      current
28      DESCRIPTION
29          "Base station ID."
30      REFERENCE
31         "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
32         ::= { wmanIfSsOfdmaDownlinkChannelEntry 7 }
33
34  wmanIfSsOfdmaMacVersion OBJECT-TYPE
35      SYNTAX      WmanIfMacVersion
36      MAX-ACCESS  read-only
37      STATUS      current
38      DESCRIPTION
39          "This parameter specifies the version of 802.16 to which
40           the message originator conforms."
41      REFERENCE
42         "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
43         ::= { wmanIfSsOfdmaDownlinkChannelEntry 8 }
44
45  wmanIfSsOfdmaFrameDurationCode OBJECT-TYPE
46      SYNTAX      INTEGER { aASGap(0),
47                           duration2ms(1),
48                           duration2dot5ms(2),
49                           duration4ms(3),
50                           duration5ms(4),
51                           duration8ms(5),
52                           duration10ms(6),
53                           duration12dot5ms(7),
54                           duration20ms(8) }
55

```

```

1      MAX-ACCESS    read-only
2      STATUS        current
3      DESCRIPTION
4          "The duration of the frame. The frame duration code values
5          are specified in Table 232 in IEEE Std 802.16-2004."
6      REFERENCE
7          "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
8      ::= { wmanIfSsOfdmaDownlinkChannelEntry 9 }
9
10
11
12  wmanIfSsOfdmaSizeCqichIdField OBJECT-TYPE
13      SYNTAX          INTEGER {threebits(1),
14                          fourbits(2),
15                          fivebits(3),
16                          sixbits(4),
17                          sevenbits(5),
18                          eightbits(6),
19                          ninebits(7)}
20
21      MAX-ACCESS    read-only
22      STATUS        current
23      DESCRIPTION
24          "This object defines the size of CQICH ID field.
25          0 = Reserved
26          1 = 3 bits
27          2 = 4 bits
28          3 = 5 bits
29          4 = 6 bits
30          5 = 7 bits
31          6 = 8 bits
32          7 = 9 bits
33          8...255 = Reserved"
34      REFERENCE
35          "Subclause 11.3.1, Table 358, in IEEE Std 802.16-2004"
36      ::= { wmanIfSsOfdmaDownlinkChannelEntry 10 }
37
38
39
40
41
42  wmanIfSsOfdmaHARQAackDelayBurst OBJECT-TYPE
43      SYNTAX          INTEGER {oneframeoffset(1),
44                          twoframesoffset(2),
45                          threeframesoffset(3)}
46
47      MAX-ACCESS    read-only
48      STATUS        current
49      DESCRIPTION
50          "This object defines the OFDMA H-ARQ ACK delay for DL burst.
51          1 = one frame offset
52          2 = two frames offset
53          3 = three frames offset"
54      REFERENCE
55          "Subclause 11.3.1, Table 358, in IEEE Std 802.16-2004"
56      ::= { wmanIfSsOfdmaDownlinkChannelEntry 11 }
57
58
59
60  wmanIfSsOfdmaUcdBurstProfileTable OBJECT-TYPE
61      SYNTAX          SEQUENCE OF WmanIfSsOfdmaUcdBurstProfileEntry
62      MAX-ACCESS    not-accessible
63      STATUS        current
64      DESCRIPTION
65

```

```

1           "This table contains UCD burst profiles for each uplink
2           channel"
3
4       REFERENCE
5           "Subclause 11.3.1.1, Table 288 and Table 357, in IEEE Std
6       802.16-2004"
7           ::= { wmanIfSsOfdmaPhy 3 }
8
9
10      wmanIfSsOfdmaUcdBurstProfileEntry OBJECT-TYPE
11          SYNTAX      WmanIfSsOfdmaUcdBurstProfileEntry
12          MAX-ACCESS   not-accessible
13          STATUS       current
14          DESCRIPTION
15              "This table provides one row for each UCD burst profile.
16              This table is double indexed. The primary index is an
17              ifIndex with an ifType of propBWA2Mp. The secondary index
18              is wmanIfSsOfdmaUiucIndex."
19          INDEX        { ifIndex, wmanIfSsOfdmaUiucIndex }
20          ::= { wmanIfSsOfdmaUcdBurstProfileTable 1 }
21
22
23
24      WmanIfSsOfdmaUcdBurstProfileEntry ::= SEQUENCE {
25          wmanIfSsOfdmaUiucIndex          INTEGER,
26          wmanIfSsOfdmaUcdFecCodeType     WmanIfOfdmaFecCodeType,
27          wmanIfSsOfdmaRangingDataRatio   INTEGER,
28          wmanIfSsOfdmaNorCOverNOVERRIDE  OCTET STRING}
29
30
31      wmanIfSsOfdmaUiucIndex OBJECT-TYPE
32          SYNTAX      INTEGER (1 .. 10)
33          MAX-ACCESS   read-only
34          STATUS       current
35          DESCRIPTION
36              "The Uplink Interval Usage Code indicates the uplink burst
37              profile in the UCD message, and is used along with ifIndex
38              to identify an entry in the
39              wmanIfSsOfdmaUcdBurstProfileTable."
40          REFERENCE
41              "Subclause 8.4.5.4.1, in IEEE Std 802.16-2004"
42          ::= { wmanIfSsOfdmaUcdBurstProfileEntry 1 }
43
44
45
46      wmanIfSsOfdmaUcdFecCodeType OBJECT-TYPE
47          SYNTAX      WmanIfOfdmaFecCodeType
48          MAX-ACCESS   read-only
49          STATUS       current
50          DESCRIPTION
51              "Uplink FEC code type and modulation type"
52          REFERENCE
53              "Subclause 11.3.1.1, Table 357, in IEEE Std 802.16-2004"
54          ::= { wmanIfSsOfdmaUcdBurstProfileEntry 2 }
55
56
57
58      wmanIfSsOfdmaRangingDataRatio OBJECT-TYPE
59          SYNTAX      INTEGER (0 .. 255)
60          MAX-ACCESS   read-only
61          STATUS       current
62          DESCRIPTION
63              "Reducing factor in units of 1 dB, between the power used
64

```

```

1         for this burst and power should be used for CDMA Ranging."
2     REFERENCE
3     "Subclause 11.3.1.1, Table 357, in IEEE Std 802.16-2004"
4     ::= { wmanIfSsOfdmaUcdBurstProfileEntry 3 }
5
6
7     wmanIfSsOfdmaNorCoverNOVERRIDE OBJECT-TYPE
8         SYNTAX OCTET STRING (SIZE (5))
9         MAX-ACCESS read-only
10        STATUS current
11        DESCRIPTION
12            "This is a list of numbers, where each number is encoded by
13             one nibble, and interpreted as a signed integer. The nibbles
14             correspond in order to the list define by Table 334 in
15             IEEE Std 802.16-2004 starting from the second line, such that
16             the LS nibble of the first byte corresponds to the second
17             line in the table. The number encoded by each nibble
18             represents the difference in normalized C/N relative to the
19             previous line in the table"
20        REFERENCE
21            "Subclause 11.3.1.1, Table 357, in IEEE Std 802.16-2004"
22        ::= { wmanIfSsOfdmaUcdBurstProfileEntry 4 }
23
24
25        wmanIfSsOfdmaDcdBurstProfileTable OBJECT-TYPE
26            SYNTAX SEQUENCE OF WmanIfSsOfdmaDcdBurstProfileEntry
27            MAX-ACCESS not-accessible
28            STATUS current
29            DESCRIPTION
30                "This table provides one row for each DCD burst profile.
31                 This table is double indexed. The primary index is an
32                 ifIndex with an ifType of propBWA2Mp. The secondary index
33                 is wmanIfSsOfdmaDiucIndex."
34            ::= { wmanIfSsOfdmaPhy 4 }
35
36
37        wmanIfSsOfdmaDcdBurstProfileEntry OBJECT-TYPE
38            SYNTAX WmanIfSsOfdmaDcdBurstProfileEntry
39            MAX-ACCESS not-accessible
40            STATUS current
41            DESCRIPTION
42                "This table provides one row for each DCD burst profile,
43                 and is double indexed. The primary index is an ifIndex
44                 with an ifType of propBWA2Mp. The secondary index is
45                 wmanIfSsOfdmaDiucIndex."
46            INDEX { ifIndex, wmanIfSsOfdmaDiucIndex }
47            ::= { wmanIfSsOfdmaDcdBurstProfileTable 1 }
48
49
50        WmanIfSsOfdmaDcdBurstProfileEntry ::= SEQUENCE {
51            wmanIfSsOfdmaDiucIndex INTEGER,
52            wmanIfSsOfdmaDownlinkFrequency Unsigned32,
53            wmanIfSsOfdmaDcdFecCodeType WmanIfOfdmaFecCodeType,
54            wmanIfSsOfdmaDiucMandatoryExitThresh INTEGER,
55            wmanIfSsOfdmaDiucMinEntryThresh INTEGER}
56
57
58        wmanIfSsOfdmaDiucIndex OBJECT-TYPE
59            SYNTAX INTEGER (0 .. 12)
60
61

```

```

1      MAX-ACCESS    read-only
2      STATUS        current
3      DESCRIPTION
4          "The Downlink Interval Usage Code indicates the downlink burst
5      profile in the DCD message, and is used
6      along with ifIndex to identify an entry in the
7          wmanIfSsOfdmaDcdBurstProfileTable."
8      REFERENCE
9          "Subclause 8.4.5.3.1, in IEEE Std 802.16-2004"
10     ::= { wmanIfSsOfdmaDcdBurstProfileEntry 1 }
11
12 wmanIfSsOfdmaDownlinkFrequency OBJECT-TYPE
13     SYNTAX          Unsigned32
14     UNITS            "kHz"
15     MAX-ACCESS      read-only
16     STATUS          current
17     DESCRIPTION
18         "Downlink Frequency (kHz)."

```

```

1      ::= { wmanIfSsOfdmaDcdBurstProfileEntry 5 }
2
3
4      --
5      -- Common object group - containing common tables and objects to be
6      -- implemented in both Base Station and Subscriber Station
7      --
8      -- wmanIfCmnPacketCs contain the Packet Convergence Sublayer objects
9      -- that are common to both Base Station and Subscriber Station
10     --
11
12     wmanIfCmnPacketCs OBJECT IDENTIFIER ::= { wmanIfCommonObjects 1 }
13
14     wmanIfCmnClassifierRuleTable OBJECT-TYPE
15         SYNTAX      SEQUENCE OF WmanIfCmnClassifierRuleEntry
16         MAX-ACCESS   not-accessible
17         STATUS       current
18         DESCRIPTION
19             "This table contains packet classifier rules associated
20             with service flows."
21         ::= { wmanIfCmnPacketCs 1 }
22
23
24
25     wmanIfCmnClassifierRuleEntry OBJECT-TYPE
26         SYNTAX      WmanIfCmnClassifierRuleEntry
27         MAX-ACCESS   not-accessible
28         STATUS       current
29         DESCRIPTION
30             "This table provides one row for each packet classifier
31             rule, and is indexed by ifIndex, wmanIfCmnCpsSfId, and
32             wmanIfCmnClassifierRuleIndex. ifIndex is associated with
33             the BS sector. wmanIfCmnCpsSfId identifies the service
34             flow, and wmanIfCmnClassifierRuleIndex identifies the
35             packet classifier rule."
36         INDEX { ifIndex, wmanIfCmnCpsSfId,
37                 wmanIfCmnClassifierRuleIndex }
38         ::= { wmanIfCmnClassifierRuleTable 1 }
39
40
41
42
43     WmanIfCmnClassifierRuleEntry ::= SEQUENCE {
44         wmanIfCmnClassifierRuleIndex      Unsigned32,
45         wmanIfCmnClassifierRulePriority    INTEGER,
46         wmanIfCmnClassifierRuleIpTosLow   INTEGER,
47         wmanIfCmnClassifierRuleIpTosHigh  INTEGER,
48         wmanIfCmnClassifierRuleIpTosMask  INTEGER,
49         wmanIfCmnClassifierRuleIpProtocol Integer32,
50         wmanIfCmnClassifierRuleIpSourceAddr InetAddress,
51         wmanIfCmnClassifierRuleIpSourceMask InetAddress,
52         wmanIfCmnClassifierRuleIpDestAddr  InetAddress,
53         wmanIfCmnClassifierRuleIpDestMask  InetAddress,
54         wmanIfCmnClassifierRuleSourcePortStart Integer32,
55         wmanIfCmnClassifierRuleSourcePortEnd Integer32,
56         wmanIfCmnClassifierRuleDestPortStart Integer32,
57         wmanIfCmnClassifierRuleDestPortEnd Integer32,
58         wmanIfCmnClassifierRuleDestMacAddr MacAddress,
59         wmanIfCmnClassifierRuleDestMacMask MacAddress,
60         wmanIfCmnClassifierRuleSourceMacAddr MacAddress,
61         wmanIfCmnClassifierRuleSourceMacMask MacAddress,
62
63
64
65

```

```

1      wmanIfCmnClassifierRuleEnetProtocolType INTEGER,
2      wmanIfCmnClassifierRuleEnetProtocol   Integer32,
3      wmanIfCmnClassifierRuleUserPriLow     Integer32,
4      wmanIfCmnClassifierRuleUserPriHigh   Integer32,
5      wmanIfCmnClassifierRuleVlanId        Integer32,
6      wmanIfCmnClassifierRuleState          INTEGER,
7      wmanIfCmnClassifierRulePkts          Counter64,
8
9      wmanIfCmnClassifierRuleIpv6FlowLabel  WmanIfIpv6FlowLabel,
10     wmanIfCmnClassifierRuleBitMap          WmanIfClassifierBitMap}
11
12
13 wmanIfCmnClassifierRuleIndex OBJECT-TYPE
14     SYNTAX      Unsigned32 (1..4294967295)
15     MAX-ACCESS  not-accessible
16     STATUS      current
17     DESCRIPTION
18         "An index is assigned to each classifier in the classifiers
19         table"
20     ::= { wmanIfCmnClassifierRuleEntry 1 }
21
22
23 wmanIfCmnClassifierRulePriority OBJECT-TYPE
24     SYNTAX      INTEGER (0..255)
25     MAX-ACCESS  read-only
26     STATUS      current
27     DESCRIPTION
28         "The value specifies the order of evaluation of the
29         classifiers. The higher the value the higher the
30         priority. The value of 0 is used as default in
31         provisioned service flows classifiers. The default
32         value of 64 is used for dynamic service flow classifiers.
33         If the referenced parameter is not present in a classifier
34         , this object reports the default value as defined above"
35     REFERENCE
36         "Subclause 11.13.19.3.4.1 in IEEE Std 802.16-2004"
37     DEFVAL      { 0 }
38     ::= { wmanIfCmnClassifierRuleEntry 2 }
39
40
41 wmanIfCmnClassifierRuleIpTosLow OBJECT-TYPE
42     SYNTAX      INTEGER (0 .. 255)
43     MAX-ACCESS  read-only
44     STATUS      current
45     DESCRIPTION
46         "The low value of a range of TOS byte values. If the
47         referenced parameter is not present in a classifier, this
48         object reports the value of 0."
49     REFERENCE
50         "Subclause 11.13.19.3.4.2 in IEEE Std 802.16-2004"
51     ::= { wmanIfCmnClassifierRuleEntry 3 }
52
53
54 wmanIfCmnClassifierRuleIpTosHigh OBJECT-TYPE
55     SYNTAX      INTEGER (0 .. 255)
56     MAX-ACCESS  read-only
57     STATUS      current
58     DESCRIPTION
59         "The 8-bit high value of a range of TOS byte values.
60

```

```

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

```

```

1      "This object specifies which bits of a packet's IP Source
2      Address that are compared to match this rule. An IP packet
3      matches the rule when the packet source address bitwise
4      ANDed with the
5      wmanIfCmnClassifierRuleIpSourceMask value equals the
6      wmanIfCmnClassifierRuleIpSourceAddr value.
7      If the referenced parameter is not present in a classifier
8      , this object reports the value of 0.0.0.0."
9
10     REFERENCE
11         "Subclause 11.13.19.3.4.4 in IEEE Std 802.16-2004"
12     ::= { wmanIfCmnClassifierRuleEntry 8 }
13
14
15     wmanIfCmnClassifierRuleIpDestAddr OBJECT-TYPE
16         SYNTAX      InetAddress
17         MAX-ACCESS   read-only
18         STATUS       current
19         DESCRIPTION
20             "This object specifies the value of the IP Destination
21             Address required for packets to match this rule. An IP
22             packet matches the rule when the packet IP destination
23             address bitwise ANDed with the
24             wmanIfCmnClassifierRuleIpDestMask value equals the
25             wmanIfCmnClassifierRuleIpDestAddr value.
26             If the referenced parameter is not present in a
27             classifier, this object reports the value of 0.0.0.0."
28         REFERENCE
29             "Subclause 11.13.19.3.4.5 in IEEE Std 802.16-2004"
30     ::= { wmanIfCmnClassifierRuleEntry 9 }
31
32
33     wmanIfCmnClassifierRuleIpDestMask OBJECT-TYPE
34         SYNTAX      InetAddress
35         MAX-ACCESS   read-only
36         STATUS       current
37         DESCRIPTION
38             "This object specifies which bits of a packet's IP
39             Destination Address that are compared to match this rule.
40             An IP packet matches the rule when the packet destination
41             address bitwise ANDed with the
42             wmanIfCmnClassifierRuleIpDestMask value equals the
43             wmanIfCmnClassifierRuleIpDestAddr value.
44             If the referenced parameter is not present in a classifier
45             , this object reports the value of 0.0.0.0."
46         REFERENCE
47             "Subclause 11.13.19.3.4.5 in IEEE Std 802.16-2004"
48     ::= { wmanIfCmnClassifierRuleEntry 10 }
49
50
51     wmanIfCmnClassifierRuleSourcePortStart OBJECT-TYPE
52         SYNTAX      Integer32 (0..65535)
53         MAX-ACCESS   read-only
54         STATUS       current
55         DESCRIPTION
56             "This object specifies the low end inclusive range of
57             TCP/UDP source port numbers to which a packet is compared
58             . This object is irrelevant for non-TCP/UDP IP packets."
59
60
61

```

```

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

```

```

1          wmanIfCmnClassifierRuleDestMacMask equals the value of
2          wmanIfCmnClassifierRuleDestMacAddr. If the referenced
3          parameter is not present in a classifier, this object
4          reports the value of '000000000000'H."
5
6      REFERENCE
7          "Subclause 11.13.19.3.4.8 in IEEE Std 802.16-2004"
8      ::= { wmanIfCmnClassifierRuleEntry 15 }
9
10
11  wmanIfCmnClassifierRuleDestMacMask OBJECT-TYPE
12      SYNTAX      MacAddress
13      MAX-ACCESS  read-only
14      STATUS      current
15      DESCRIPTION
16          "An Ethernet packet matches an entry when its destination
17          MAC address bitwise ANDed with
18          wmanIfCmnClassifierRuleDestMacMask equals the value of
19          wmanIfCmnClassifierRuleDestMacAddr. If the referenced
20          parameter is not present in a classifier, this object
21          reports the value of '000000000000'H."
22
23      REFERENCE
24          "Subclause 11.13.19.3.4.8 in IEEE Std 802.16-2004"
25      ::= { wmanIfCmnClassifierRuleEntry 16 }
26
27
28
29  wmanIfCmnClassifierRuleSourceMacAddr OBJECT-TYPE
30      SYNTAX      MacAddress
31      MAX-ACCESS  read-only
32      STATUS      current
33      DESCRIPTION
34          "An Ethernet packet matches this entry when its source
35          MAC address bitwise ANDed with
36          wmanIfCmnClassifierRuleSourceMacMask equals the value
37          of wmanIfCmnClassifierRuleSourceMacAddr. If the
38          referenced parameter is not present in a classifier,
39          this object reports the value of '000000000000'H."
40
41      REFERENCE
42          "Subclause 11.13.19.3.4.9 in IEEE Std 802.16-2004"
43      ::= { wmanIfCmnClassifierRuleEntry 17 }
44
45
46
47  wmanIfCmnClassifierRuleSourceMacMask OBJECT-TYPE
48      SYNTAX      MacAddress
49      MAX-ACCESS  read-only
50      STATUS      current
51      DESCRIPTION
52          "An Ethernet packet matches an entry when its destination
53          MAC address bitwise ANDed with
54          wmanIfCmnClassifierRuleSourceMacMask equals the value of
55          wmanIfCmnClassifierRuleSourceMacAddr. If the referenced
56          parameter is not present in a classifier, this object
57          reports the value of '000000000000'H."
58
59      REFERENCE
60          "Subclause 11.13.19.3.4.9 in IEEE Std 802.16-2004"
61      ::= { wmanIfCmnClassifierRuleEntry 18 }
62
63
64
65  wmanIfCmnClassifierRuleEnetProtocolType OBJECT-TYPE

```

```

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

```

```

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

```

1                               inactive(2) }
2   MAX-ACCESS    read-only
3   STATUS        current
4   DESCRIPTION
5       "This object indicates whether or not the classifier is
6       enabled to classify packets to a Service Flow.
7       If the referenced parameter is not present in the
8       classifier, the value of this object is reported
9       as active(1)."
```

::= { wmanIfCmnClassifierRuleEntry 24 }

```

14 wmanIfCmnClassifierRulePkts OBJECT-TYPE
15     SYNTAX      Counter64
16     MAX-ACCESS  read-only
17     STATUS      current
18     DESCRIPTION
19         "This object counts the number of packets that have
20         been classified using this entry."
21     ::= { wmanIfCmnClassifierRuleEntry 25 }
```

wmanIfCmnClassifierRuleIpv6FlowLabel OBJECT-TYPE

```

25     SYNTAX      WmanIfIpv6FlowLabel
26     MAX-ACCESS  read-only
27     STATUS      current
28     DESCRIPTION
29         "The value of this field specifies the matching values for
30         the IPv6 Flow label field."
31     ::= { wmanIfCmnClassifierRuleEntry 26 }
```

wmanIfCmnClassifierRuleBitMap OBJECT-TYPE

```

36     SYNTAX      WmanIfClassifierBitMap
37     MAX-ACCESS  read-only
38     STATUS      current
39     DESCRIPTION
40         "This object indicates which parameter encodings were
41         actually present in the entry. A bit set to '1' indicates
42         the corresponding classifier encoding is present, and '0'
43         means otherwise"
44     ::= { wmanIfCmnClassifierRuleEntry 27 }
```

wmanIfCmnPhsRuleTable OBJECT-TYPE

```

49     SYNTAX      SEQUENCE OF WmanIfCmnPhsRuleEntry
50     MAX-ACCESS  not-accessible
51     STATUS      current
52     DESCRIPTION
53         "This table contains PHS rule dictionary entries. Each
54         entry contains the data of the header to be suppressed
55         along with its identification - PHSI. The classifier
56         uniquely maps packets to its associated PHS Rule. The
57         receiving entity uses the CID and the PHSI to restore the
58         PHSF. Once a PHSF has been assigned to a PHSI, it shall
59         not be changed. To change the value of a PHSF on a
60         service flow, a new PHS rule shall be defined, the old
61         rule is removed from the service flow, and the new rule
62         is added. When a classifier is deleted, any associated
```

```

1           PHS rule shall also be deleted."
2 REFERENCE
3       "Subclause 5.2.3 in IEEE Std 802.16-2004"
4 ::= { wmanIfCmnPacketCs 2 }
5
6
7 wmanIfCmnPhsRuleEntry OBJECT-TYPE
8     SYNTAX      WmanIfCmnPhsRuleEntry
9     MAX-ACCESS  not-accessible
10    STATUS      current
11    DESCRIPTION
12        "This table provides one row for each PHS rule created
13        dynamically by the BS and SS on a given service flow. The
14        PHS rule is defined by the pair (PHSS, PHSM) for each
15        distinct header data. It is indexed by IfIndex,
16        wmanIfCmnCpsSfId, and wmanIfCmnPhsIndex. The table is
17        read-only for NMS. "
18    INDEX        { ifIndex, wmanIfCmnCpsSfId,
19                  wmanIfCmnPhsRulePhsIndex }
20 ::= { wmanIfCmnPhsRuleTable 1 }
21
22
23
24
25 WmanIfCmnPhsRuleEntry ::= SEQUENCE {
26     wmanIfCmnPhsRulePhsIndex      INTEGER,
27     wmanIfCmnPhsRulePhsField      OCTET STRING,
28     wmanIfCmnPhsRulePhsMask       OCTET STRING,
29     wmanIfCmnPhsRulePhsSize       Integer32,
30     wmanIfCmnPhsRulePhsVerify     WmanIfPhsRuleVerify}
31
32
33 wmanIfCmnPhsRulePhsIndex OBJECT-TYPE
34     SYNTAX      INTEGER (1..255)
35     MAX-ACCESS  not-accessible
36     STATUS      current
37     DESCRIPTION
38         "The PHSI (PHS Index) has a value between 1 and 255, which
39         uniquely references the suppressed byte string. The index
40         is unique per service flow. The uplink and downlink PHSI
41         values are independent of each other."
42     REFERENCE
43         "Subclause 11.13.19.3.7.1 in IEEE Std 802.16-2004"
44 ::= { wmanIfCmnPhsRuleEntry 1 }
45
46
47
48
49 wmanIfCmnPhsRulePhsField OBJECT-TYPE
50     SYNTAX      OCTET STRING (SIZE(0..65535))
51     MAX-ACCESS  read-only
52     STATUS      current
53     DESCRIPTION
54         "The PHSF (PHS Field) is a string of bytes containing the
55         header information to be suppressed by the sending CS and
56         reconstructed by the receiving CS. The most significant
57         byte of the string corresponds to the first byte of the
58         CS-SDU."
59     REFERENCE
60         "Subclause 11.13.19.3.7.2 in IEEE Std 802.16-2004"
61 ::= { wmanIfCmnPhsRuleEntry 2 }
62
63
64
65

```

```

1  wmanIfCmnPhsRulePhsMask OBJECT-TYPE
2      SYNTAX      OCTET STRING (SIZE(0..65535))
3
4      MAX-ACCESS  read-only
5
6      STATUS      current
7
8      DESCRIPTION
9          "The PHSM An 8-bit mask that indicates which bytes in the
10         PHS Field (PHSF) to suppress and which bytes to not
11         suppress. The PHSM allows fields, such as sequence numbers
12         or checksums (which vary in value), to be excluded from
13         suppression with the constant bytes around them suppressed.
14         It is encoded as follows:
15         bit 0:
16             0 = don't suppress the 1st byte of the suppression field
17             1 = suppress first byte of the suppression field
18         bit 1:
19             0 = don't suppress the 2nd byte of the suppression field
20             1 = suppress second byte of the suppression field
21         bit x:
22             0 = don't suppress the (x+1) byte of the suppression
23             field
24             1 = suppress (x+1) byte of the suppression field
25         where the length of the octet string is ceiling
26         (wmanIfCmnPhsRulePhsSize/8)."
27
28     REFERENCE
29         "Subclause 11.13.19.3.7.3 in IEEE Std 802.16-2004"
30     ::= { wmanIfCmnPhsRuleEntry 3 }
31
32
33  wmanIfCmnPhsRulePhsSize OBJECT-TYPE
34      SYNTAX      Integer32
35
36      UNITS       "byte"
37
38      MAX-ACCESS  read-only
39
40      STATUS      current
41
42      DESCRIPTION
43          "The value of this field - PHSS is the total number of bytes
44         in the header to be suppressed and then restored in a
45         service flow that uses PHS."
46
47     REFERENCE
48         "Subclause 11.13.19.3.7.4 in IEEE Std 802.16-2004"
49
50     DEFVAL      { 0 }
51     ::= { wmanIfCmnPhsRuleEntry 4 }
52
53
54  wmanIfCmnPhsRulePhsVerify OBJECT-TYPE
55      SYNTAX      WmanIfPhsRuleVerify
56
57      MAX-ACCESS  read-only
58
59      STATUS      current
60
61      DESCRIPTION
62          "The value of this field indicates to the sending entity
63         whether or not the packet header contents are to be
64         verified prior to performing suppression."
65
66     DEFVAL      { phsVerifyEnable }
67     ::= { wmanIfCmnPhsRuleEntry 5 }
68
69
70  --
71  -- wmanIfCmnCps contain the Common Part Sublayer objects that are

```

```

1  -- common to both Base Station and Subscriber Station
2  --
3  wmanIfCmnCps OBJECT IDENTIFIER ::= { wmanIfCommonObjects 2 }
4
5
6  wmanIfCmnCpsServiceFlowTable OBJECT-TYPE
7      SYNTAX      SEQUENCE OF WmanIfCmnCpsServiceFlowEntry
8      MAX-ACCESS  not-accessible
9      STATUS      current
10     DESCRIPTION
11         "This table contains Service Flow managed objects that
12         are common in BS and SS."
13     ::= { wmanIfCmnCps 1 }
14
15
16  wmanIfCmnCpsServiceFlowEntry OBJECT-TYPE
17      SYNTAX      WmanIfCmnCpsServiceFlowEntry
18      MAX-ACCESS  not-accessible
19      STATUS      current
20      DESCRIPTION
21         "This table provides one row for each created service
22         flow for a given MacAddress, and is indexed by ifIndex,
23         wmanIfCmnCpsCpsSfMacAddress, and wmanIfCmnCpsSfId.
24         IfIndex is associated with the BS sector."
25     INDEX      { ifIndex, wmanIfCmnCpsSfMacAddress,
26                  wmanIfCmnCpsSfId }
27     ::= { wmanIfCmnCpsServiceFlowTable 1 }
28
29
30  WmanIfCmnCpsServiceFlowEntry ::= SEQUENCE {
31      wmanIfCmnCpsSfMacAddress      MacAddress,
32      wmanIfCmnCpsSfId              Unsigned32,
33      wmanIfCmnCpsSfCid             WmanIfCidType,
34      wmanIfCmnCpsSfDirection      INTEGER,
35      wmanIfCmnCpsSfState           WmanIfSfState,
36      wmanIfCmnCpsTrafficPriority   INTEGER,
37      wmanIfCmnCpsMaxSustainedRate  Unsigned32,
38      wmanIfCmnCpsMaxTrafficBurst   Unsigned32,
39      wmanIfCmnCpsMinReservedRate   Unsigned32,
40      wmanIfCmnCpsToleratedJitter   Unsigned32,
41      wmanIfCmnCpsMaxLatency        Unsigned32,
42      wmanIfCmnCpsFixedVsVariableSduInd  INTEGER,
43      wmanIfCmnCpsSduSize           Unsigned32,
44      wmanIfCmnCpsSfsSchedulingType WmanIfSfsSchedulingType,
45      wmanIfCmnCpsArqEnable         TruthValue,
46      wmanIfCmnCpsArqWindowSize     INTEGER,
47      wmanIfCmnCpsArqBlockLifetime  INTEGER,
48      wmanIfCmnCpsArqSyncLossTimeout  INTEGER,
49      wmanIfCmnCpsArqDeliverInOrder TruthValue,
50      wmanIfCmnCpsArqRxPurgeTimeout  INTEGER,
51      wmanIfCmnCpsArqBlockSize      INTEGER,
52      wmanIfCmnCpsMinRsvdTolerableRate  Unsigned32,
53      wmanIfCmnCpsReqTxPolicy        BITS,
54      wmanIfCmnSfCsSpecification    WmanIfCsSpecification,
55      wmanIfCmnCpsTargetSaid        INTEGER}
56
57
58  wmanIfCmnCpsSfMacAddress OBJECT-TYPE
59
60
61
62
63
64
65

```

```

1      SYNTAX      MacAddress
2      MAX-ACCESS  not-accessible
3      STATUS      current
4      DESCRIPTION
5          "When this table is implemented on the basestation, this
6          object contains the SS Mac address, the reported service
7          flow was created for. On the SS, the value returned is
8          the SS's own Mac address."
9      ::= { wmanIfCmnCpsServiceFlowEntry 1 }
10
11
12
13 wmanIfCmnCpsSfId OBJECT-TYPE
14     SYNTAX      Unsigned32 ( 1 .. 4294967295)
15     MAX-ACCESS  read-only
16     STATUS      current
17     DESCRIPTION
18         "A 32 bit quantity that uniquely identifies a service flow
19         to both the subscriber station and base station (BS)."
```

```

20     ::= { wmanIfCmnCpsServiceFlowEntry 2 }
21
22
23
24 wmanIfCmnCpsSfCid OBJECT-TYPE
25     SYNTAX      WmanIfCidType
26     MAX-ACCESS  read-only
27     STATUS      current
28     DESCRIPTION
29         "A 16 bit channel identifier to identify the connection
30         being created by DSA."
31     ::= { wmanIfCmnCpsServiceFlowEntry 3 }
32
33
34
35 wmanIfCmnCpsSfDirection OBJECT-TYPE
36     SYNTAX      INTEGER {downstream(1),
37                        upstream(2)}
38     MAX-ACCESS  read-only
39     STATUS      current
40     DESCRIPTION
41         "An attribute indicating the service flow is downstream or
42         upstream."
43     ::= { wmanIfCmnCpsServiceFlowEntry 4 }
44
45
46
47 wmanIfCmnCpsSfState OBJECT-TYPE
48     SYNTAX      WmanIfSfState
49     MAX-ACCESS  read-only
50     STATUS      current
51     DESCRIPTION
52         "wmanIfCmnCpsSfState indicates the service flow state:
53         Authorized (1), Admitted (2), and Active (3) service
54         flow state."
55     REFERENCE
56         "Subclause 6.3.14.6, in IEEE Std 802.16-2004"
```

```

57     ::= { wmanIfCmnCpsServiceFlowEntry 5 }
58
59
60
61 wmanIfCmnCpsTrafficPriority OBJECT-TYPE
62     SYNTAX      INTEGER (0 .. 7)
63     MAX-ACCESS  read-only
64     STATUS      current
65
```

```

1      DESCRIPTION
2          "The value of this parameter specifies the priority
3            assigned to a service flow. For uplink service flows,
4            the BS should use this parameter when determining
5            precedence in request service and grant generation,
6            and the SS shall preferentially select contention
7            Request opportunities for Priority Request CIDs
8            based on this priority"
9
10     REFERENCE
11         "Subclause 11.13.5 in IEEE Std 802.16-2004"
12         ::= { wmanIfCmnCpsServiceFlowEntry 6 }
13
14
15     wmanIfCmnCpsMaxSustainedRate OBJECT-TYPE
16         SYNTAX      Unsigned32
17         UNITS        "b/s"
18         MAX-ACCESS   read-only
19         STATUS        current
20         DESCRIPTION
21             "This parameter defines the peak information rate
22               of the service. The rate is expressed in bits per
23               second and pertains to the SDUs at the input to
24               the system."
25         REFERENCE
26             "Subclause 11.13.6 in IEEE Std 802.16-2004"
27             ::= { wmanIfCmnCpsServiceFlowEntry 7 }
28
29
30     wmanIfCmnCpsMaxTrafficBurst OBJECT-TYPE
31         SYNTAX      Unsigned32
32         UNITS        "byte"
33         MAX-ACCESS   read-only
34         STATUS        current
35         DESCRIPTION
36             "This parameter defines the maximum burst size that
37               must be accommodated for the service."
38         REFERENCE
39             "Subclause 11.13.7 in IEEE Std 802.16-2004"
40             ::= { wmanIfCmnCpsServiceFlowEntry 8 }
41
42
43     wmanIfCmnCpsMinReservedRate OBJECT-TYPE
44         SYNTAX      Unsigned32
45         UNITS        "byte"
46         MAX-ACCESS   read-only
47         STATUS        current
48         DESCRIPTION
49             "This parameter specifies the minimum rate reserved
50               for this service flow."
51         REFERENCE
52             "Subclause 11.13.8 in IEEE Std 802.16-2004"
53             ::= { wmanIfCmnCpsServiceFlowEntry 9 }
54
55
56     wmanIfCmnCpsToleratedJitter OBJECT-TYPE
57         SYNTAX      Unsigned32
58         UNITS        "millisecond"
59         MAX-ACCESS   read-only
60
61
62
63
64
65

```

```

1      STATUS      current
2      DESCRIPTION
3          "This parameter defines the Maximum delay
4          variation (jitter) for the connection."
5      REFERENCE
6          "Subclause 11.13.13 in IEEE Std 802.16-2004"
7      ::= { wmanIfCmnCpsServiceFlowEntry 10 }
8
9
10     wmanIfCmnCpsMaxLatency OBJECT-TYPE
11         SYNTAX      Unsigned32
12         UNITS        "millisecond"
13         MAX-ACCESS   read-only
14         STATUS       current
15         DESCRIPTION
16             "The value of this parameter specifies the maximum
17             latency between the reception of a packet by the BS
18             or SS on its network interface and the forwarding
19             of the packet to its RF Interface."
20         REFERENCE
21             "Subclause 11.13.14 in IEEE Std 802.16-2004"
22         ::= { wmanIfCmnCpsServiceFlowEntry 11 }
23
24     wmanIfCmnCpsFixedVsVariableSduInd OBJECT-TYPE
25         SYNTAX      INTEGER {variableLength(0),
26                         fixedLength(1)}
27         MAX-ACCESS   read-only
28         STATUS       current
29         DESCRIPTION
30             "The value of this parameter specifies whether the SDUs
31             on the service flow are variable-length (0) or
32             fixed-length (1). The parameter is used only if
33             packing is on for the service flow. The default value
34             is 0, i.e., variable-length SDUs."
35         REFERENCE
36             "Subclause 11.13.15 in IEEE Std 802.16-2004"
37         DEFVAL      { variableLength }
38         ::= { wmanIfCmnCpsServiceFlowEntry 12 }
39
40     wmanIfCmnCpsSduSize OBJECT-TYPE
41         SYNTAX      Unsigned32
42         UNITS        "byte"
43         MAX-ACCESS   read-only
44         STATUS       current
45         DESCRIPTION
46             "The value of this parameter specifies the length of the
47             SDU for a fixed-length SDU service flow. This parameter
48             is used only if packing is on and the service flow is
49             indicated as carrying fixed-length SDUs. The default
50             value is 49 bytes, i.e., VC-switched ATM cells with PHS.
51             The parameter is relevant for both ATM and Packet
52             Convergence Sublayers."
53         REFERENCE
54             "Subclause 11.13.16 in IEEE Std 802.16-2004"
55         DEFVAL      { 49 }
56
57
58
59
60
61
62
63
64
65

```

```

1      ::= { wmanIfCmnCpsServiceFlowEntry 13 }
2
3
4  wmanIfCmnCpsSfsSchedulingType OBJECT-TYPE
5      SYNTAX      WmanIfSfsSchedulingType
6      MAX-ACCESS  read-only
7      STATUS      current
8      DESCRIPTION
9          "Specifies the upstream scheduling service used for
10         upstream service flow. If the referenced parameter
11         is not present in the corresponding 802.16 QOS
12         Parameter Set of an upstream service flow, the
13         default value of this object is bestEffort(2)."

```

```

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

```

```

1      REFERENCE
2          "Subclause 11.13.9 in IEEE Std 802.16-2004"
3      ::= { wmanIfCmnCpsServiceFlowEntry 22 }
4
5
6      wmanIfCmnCpsReqTxPolicy OBJECT-TYPE
7          SYNTAX      BITS {noBroadcastBwReq(0),
8                          reserved1(1),
9                          noPiggybackReq(2),
10                         noFragmentData(3),
11                         noPHS(4),
12                         noSduPacking(5),
13                         noCrc(6),
14                         reserved2(7)}
15
16      MAX-ACCESS      read-only
17      STATUS          current
18      DESCRIPTION
19          "The value of this parameter provides the capability to
20          specify certain attributes for the associated service
21          flow. An attribute is enabled by setting the
22          corresponding bit position to 1."
23
24      REFERENCE
25          "Subclause 11.13.12 in IEEE Std 802.16-2004"
26      ::= { wmanIfCmnCpsServiceFlowEntry 23 }
27
28
29
30      wmanIfCmnSfCsSpecification OBJECT-TYPE
31          SYNTAX      WmanIfCsSpecification
32          MAX-ACCESS      read-only
33          STATUS          current
34          DESCRIPTION
35              "This parameter specifies the convergence sublayer
36              encapsulation mode."
37
38      REFERENCE
39          "Subclause 11.13.19.1 in IEEE Std 802.16-2004"
40      ::= { wmanIfCmnCpsServiceFlowEntry 24 }
41
42
43      wmanIfCmnCpsTargetSaid OBJECT-TYPE
44          SYNTAX      INTEGER (0 .. 65535)
45          MAX-ACCESS      read-only
46          STATUS          current
47          DESCRIPTION
48              "The target SAID parameter indicates the SAID onto
49              which the service flow being set up shall be mapped."
50
51      REFERENCE
52          "Subclause 11.13.17 in IEEE Std 802.16-2004"
53      ::= { wmanIfCmnCpsServiceFlowEntry 25 }
54
55
56      --
57      -- wmanIfCmnBsSsConfigurationTable contains global parameters
58      -- common in BS and SS
59      --
60
61      wmanIfCmnBsSsConfigurationTable OBJECT-TYPE
62          SYNTAX      SEQUENCE OF WmanIfCmnBsSsConfigurationEntry
63          MAX-ACCESS      not-accessible
64          STATUS          current
65

```

```

1      DESCRIPTION
2          "This table provides one row for each BS sector that
3          contains the system parameters common in both SS and
4          BS. All SSs shall have the same parameters as the BS
5          to which the SSs are associated."
6      REFERENCE
7          "Subclause 10.1 in IEEE Std 802.16-2004"
8      ::= { wmanIfCmnCps 2 }
9
10
11
12  wmanIfCmnBsSsConfigurationEntry OBJECT-TYPE
13      SYNTAX      WmanIfCmnBsSsConfigurationEntry
14      MAX-ACCESS  not-accessible
15      STATUS      current
16      DESCRIPTION
17          "This table is indexed by ifIndex, indicating BS
18          sector."
19      INDEX      { ifIndex }
20      ::= { wmanIfCmnBsSsConfigurationTable 1 }
21
22
23
24  WmanIfCmnBsSsConfigurationEntry ::= SEQUENCE {
25      wmanIfCmnInvitedRangRetries      INTEGER,
26      wmanIfCmnDSxReqRetries           Unsigned32,
27      wmanIfCmnDSxRespRetries         Unsigned32,
28      wmanIfCmnT7Timeout               INTEGER,
29      wmanIfCmnT8Timeout               INTEGER,
30      wmanIfCmnT10Timeout              INTEGER,
31      wmanIfCmnT22Timeout              INTEGER}
32
33
34
35  wmanIfCmnInvitedRangRetries OBJECT-TYPE
36      SYNTAX      INTEGER (16..65535)
37      MAX-ACCESS  read-write
38      STATUS      current
39      DESCRIPTION
40          "Number of retries on inviting Ranging Requests."
41      ::= { wmanIfCmnBsSsConfigurationEntry 1 }
42
43
44  wmanIfCmnDSxReqRetries OBJECT-TYPE
45      SYNTAX      Unsigned32
46      MAX-ACCESS  read-write
47      STATUS      current
48      DESCRIPTION
49          "Number of Timeout Retries on DSA/DSC/DSD Requests."
50      DEFVAL      { 3 }
51      ::= { wmanIfCmnBsSsConfigurationEntry 2 }
52
53
54
55  wmanIfCmnDSxRespRetries OBJECT-TYPE
56      SYNTAX      Unsigned32
57      MAX-ACCESS  read-write
58      STATUS      current
59      DESCRIPTION
60          "Number of Timeout Retries on DSA/DSC/DSD Responses."
61      DEFVAL      { 3 }
62      ::= { wmanIfCmnBsSsConfigurationEntry 3 }
63
64
65

```

```

1  wmanIfCmnT7Timeout OBJECT-TYPE
2      SYNTAX          INTEGER (0 .. 1000)
3      UNITS           "milliseconds"
4      MAX-ACCESS      read-write
5      STATUS          current
6      DESCRIPTION
7          "Wait for DSA/DSC/DSD Response Timeout in ms."
8      ::= { wmanIfCmnBsSsConfigurationEntry 4 }
9
10
11  wmanIfCmnT8Timeout OBJECT-TYPE
12      SYNTAX          INTEGER (0 .. 300)
13      UNITS           "milliseconds"
14      MAX-ACCESS      read-write
15      STATUS          current
16      DESCRIPTION
17          "Wait for DSA/DSC/DSD Acknowledge Timeout in ms."
18      ::= { wmanIfCmnBsSsConfigurationEntry 5 }
19
20
21  wmanIfCmnT10Timeout OBJECT-TYPE
22      SYNTAX          INTEGER (0 .. 3000)
23      UNITS           "milliseconds"
24      MAX-ACCESS      read-write
25      STATUS          current
26      DESCRIPTION
27          "Wait for Transaction End timeout in ms."
28      ::= { wmanIfCmnBsSsConfigurationEntry 6 }
29
30
31  wmanIfCmnT22Timeout OBJECT-TYPE
32      SYNTAX          INTEGER (0 .. 500)
33      UNITS           "milliseconds"
34      MAX-ACCESS      read-write
35      STATUS          current
36      DESCRIPTION
37          "Wait for ARQ Reset in ms."
38      ::= { wmanIfCmnBsSsConfigurationEntry 7 }
39
40
41  -- Common PKM group
42  -- wmanIfCmnPkmObjects contain the Privacy Sublayer objects that are
43  -- common to both Base Station and Subscriber Station
44  --
45  wmanIfCmnPkmObjects OBJECT IDENTIFIER ::= { wmanIfCommonObjects 3 }
46
47
48  --
49  -- Table wmanIfCmnCryptoSuiteTable
50  --
51  wmanIfCmnCryptoSuiteTable OBJECT-TYPE
52      SYNTAX          SEQUENCE OF WmanIfCmnCryptoSuiteEntry
53      MAX-ACCESS      not-accessible
54      STATUS          current
55      DESCRIPTION
56          "This table describes the PKM cryptographic suite
57          capabilites for each SS or BS wireless interface."
58      ::= { wmanIfCmnPkmObjects 1 }
59
60
61
62
63
64
65

```

```

1  wmanIfCmnCryptoSuiteEntry OBJECT-TYPE
2      SYNTAX          WmanIfCmnCryptoSuiteEntry
3      MAX-ACCESS      not-accessible
4      STATUS          current
5      DESCRIPTION
6          "Each entry contains the cryptographic suite pair that SS
7          or BS supports."
8      INDEX            { ifIndex, wmanIfCmnCryptoSuiteIndex }
9      ::= { wmanIfCmnCryptoSuiteTable 1 }
10
11
12
13  WmanIfCmnCryptoSuiteEntry ::= SEQUENCE {
14      wmanIfCmnCryptoSuiteIndex          Integer32,
15      wmanIfCmnCryptoSuiteDataEncryptAlg WmanIfDataEncryptAlgId,
16      wmanIfCmnCryptoSuiteDataAuthentAlg WmanIfDataAuthAlgId,
17      wmanIfCmnCryptoSuiteTekEncryptAlg  WmanIfTekEncryptAlgId}
18
19
20  wmanIfCmnCryptoSuiteIndex OBJECT-TYPE
21      SYNTAX          Integer32 (1 .. 1000)
22      MAX-ACCESS      not-accessible
23      STATUS          current
24      DESCRIPTION
25          "The index for a cryptographic suite row."
26      ::= { wmanIfCmnCryptoSuiteEntry 1 }
27
28
29
30  wmanIfCmnCryptoSuiteDataEncryptAlg OBJECT-TYPE
31      SYNTAX          WmanIfDataEncryptAlgId
32      MAX-ACCESS      read-only
33      STATUS          current
34      DESCRIPTION
35          "The value of this object is the data encryption algorithm
36          for this cryptographic suite capability."
37      REFERENCE
38          "Table 375, IEEE Std 802.16-2004"
39      ::= { wmanIfCmnCryptoSuiteEntry 2 }
40
41
42
43  wmanIfCmnCryptoSuiteDataAuthentAlg OBJECT-TYPE
44      SYNTAX          WmanIfDataAuthAlgId
45      MAX-ACCESS      read-only
46      STATUS          current
47      DESCRIPTION
48          "The value of this object is the data authentication
49          algorithm for this cryptographic suite capability."
50      REFERENCE
51          "Table 376, IEEE Std 802.16-2004"
52      ::= { wmanIfCmnCryptoSuiteEntry 3 }
53
54
55
56  wmanIfCmnCryptoSuiteTekEncryptAlg OBJECT-TYPE
57      SYNTAX          WmanIfTekEncryptAlgId
58      MAX-ACCESS      read-only
59      STATUS          current
60      DESCRIPTION
61          "The value of this object is the TEK key encryption
62          algorithm for this cryptographic suite capability."
63      REFERENCE
64
65

```

```

1         "Table 377, IEEE Std 802.16-2004"
2         ::= { wmanIfCmnCryptoSuiteEntry 4 }
3
4
5     --
6     -- Conformance Information
7     --
8     wmanIfMibConformance OBJECT IDENTIFIER ::= {wmanIfMib 2}
9     wmanIfMibGroups      OBJECT IDENTIFIER ::= {wmanIfMibConformance 1}
10    wmanIfMibCompliances OBJECT IDENTIFIER ::= {wmanIfMibConformance 2}
11
12
13    -- compliance statements
14    wmanIfMibCompliance MODULE-COMPLIANCE
15        STATUS          current
16        DESCRIPTION
17            "The compliance statement for devices that implement
18            Wireless MAN interfaces as defined in IEEE Std 802.16-2004."
19
20
21    MODULE -- wmanIfMib
22
23
24    MANDATORY-GROUPS          -- unconditionally mandatory groups
25        { wmanIfMibCommonGroup }
26
27
28    GROUP wmanIfMibQoSGroup    -- unconditionally mandatory group
29    DESCRIPTION
30        "This group is mandatory for Base Station and subscriber
31        station."
32
33
34    GROUP wmanIfMibBsGroup     -- conditionally mandatory group
35    DESCRIPTION
36        "This group is mandatory for Base Station."
37
38
39    GROUP wmanIfMibBsAasGroup  -- optional group
40    DESCRIPTION
41        "This group is mandatory for Base Station."
42
43
44    GROUP wmanIfMibSsGroup     -- conditionally mandatory group
45    DESCRIPTION
46        "This group is mandatory for Subscriber Station."
47
48
49    GROUP wmanIfMibBsOfdmGroup -- conditionally mandatory group
50    DESCRIPTION
51        "This group is mandatory for Base Station
52        implementaing the OFDM PHY."
53
54
55    GROUP wmanIfMibSsOfdmGroup -- conditionally mandatory group
56    DESCRIPTION
57        "This group is mandatory for Subscriber Station
58        implementing the OFDM PHY."
59
60
61    GROUP wmanIfMibBsOfdmaGroup -- conditionally mandatory group
62    DESCRIPTION
63        "This group is mandatory for Base Station
64        implementaing the OFDMA PHY."
65

```

```

1      GROUP wmanIfMibSsOfdmaGroup -- conditionally mandatory group
2      DESCRIPTION
3          "This group is mandatory for Subscriber Station
4          implementing the OFDMA PHY."
5
6
7      GROUP wmanIfMibBsNotificationGroup -- unconditionally
8                                          -- mandatory groups
9
10     DESCRIPTION
11         "This group is mandatory for Base Station."
12
13     GROUP wmanIfMibSsNotificationGroup -- optional group
14     DESCRIPTION
15         "This group is optional for Subscriber Station."
16
17
18     GROUP wmanIfMibCmnPhsGroup -- optional group
19     DESCRIPTION
20         "This group is optional for Base Station and
21         Subscriber Station."
22
23
24     GROUP wmanIfMibBsPhsGroup -- optional group
25     DESCRIPTION
26         "This group is optional for Base Station."
27     ::= { wmanIfMibCompliances 1 }
28
29
30 wmanIfMibCommonGroup      OBJECT-GROUP
31     OBJECTS {-- Classification
32         wmanIfCmnClassifierRulePriority,
33         wmanIfCmnClassifierRuleIpTosLow,
34         wmanIfCmnClassifierRuleIpTosHigh,
35         wmanIfCmnClassifierRuleIpTosMask,
36         wmanIfCmnClassifierRuleIpProtocol,
37         wmanIfCmnClassifierRuleIpSourceAddr,
38         wmanIfCmnClassifierRuleIpSourceMask,
39         wmanIfCmnClassifierRuleIpDestAddr,
40         wmanIfCmnClassifierRuleIpDestMask,
41         wmanIfCmnClassifierRuleSourcePortStart,
42         wmanIfCmnClassifierRuleSourcePortEnd,
43         wmanIfCmnClassifierRuleDestPortStart,
44         wmanIfCmnClassifierRuleDestPortEnd,
45         wmanIfCmnClassifierRuleDestMacAddr,
46         wmanIfCmnClassifierRuleDestMacMask,
47         wmanIfCmnClassifierRuleSourceMacAddr,
48         wmanIfCmnClassifierRuleSourceMacMask,
49         wmanIfCmnClassifierRuleEnetProtocolType,
50         wmanIfCmnClassifierRuleEnetProtocol,
51         wmanIfCmnClassifierRuleUserPriLow,
52         wmanIfCmnClassifierRuleUserPriHigh,
53         wmanIfCmnClassifierRuleVlanId,
54         wmanIfCmnClassifierRuleState,
55         wmanIfCmnClassifierRulePkts,
56         wmanIfCmnClassifierRuleIpv6FlowLabel,
57         wmanIfCmnClassifierRuleBitMap,
58
59         -- Configuration parameters
60
61
62
63
64
65

```

```

1          wmanIfCmnCpsTargetSaid,
2          wmanIfCmnInvitedRangRetries,
3          wmanIfCmnDSxReqRetries,
4          wmanIfCmnDSxRespRetries,
5          wmanIfCmnT7Timeout,
6          wmanIfCmnT8Timeout,
7          wmanIfCmnT10Timeout,
8          wmanIfCmnT22Timeout,
9          wmanIfCmnCryptoSuiteDataEncryptAlg,
10         wmanIfCmnCryptoSuiteDataAuthentAlg,
11         wmanIfCmnCryptoSuiteTekEncryptAlg}
12
13     STATUS          current
14
15     DESCRIPTION
16         "This group contains objects for both BS and SS,
17         and are independent of PHY."
18     ::= { wmanIfMibGroups 1 }
19
20
21 wmanIfMibQoSGroup      OBJECT-GROUP
22     OBJECTS {wmanIfCmnCpsSfId,
23             wmanIfCmnCpsSfCid,
24             wmanIfCmnCpsSfDirection,
25             wmanIfCmnCpsSfState,
26             wmanIfCmnCpsTrafficPriority,
27             wmanIfCmnCpsMaxSustainedRate,
28             wmanIfCmnCpsMaxTrafficBurst,
29             wmanIfCmnCpsMinReservedRate,
30             wmanIfCmnCpsToleratedJitter,
31             wmanIfCmnCpsMaxLatency,
32             wmanIfCmnCpsFixedVsVariableSduInd,
33             wmanIfCmnCpsSduSize,
34             wmanIfCmnCpsSfSchedulingType,
35             wmanIfCmnCpsArqEnable,
36             wmanIfCmnCpsArqWindowSize,
37             wmanIfCmnCpsArqBlockLifetime,
38             wmanIfCmnCpsArqSyncLossTimeout,
39             wmanIfCmnCpsArqDeliverInOrder,
40             wmanIfCmnCpsArqRxPurgeTimeout,
41             wmanIfCmnCpsArqBlockSize,
42             wmanIfCmnCpsMinRsvdTolerableRate,
43             wmanIfCmnCpsReqTxPolicy,
44             wmanIfCmnSfCsSpecification}
45
46     STATUS          current
47
48     DESCRIPTION
49         "This group contains QoS objects for both BS and SS."
50     ::= { wmanIfMibGroups 2 }
51
52
53 wmanIfMibBsGroup      OBJECT-GROUP
54     OBJECTS {-- Service classes
55             wmanIfBsSfDirection,
56             wmanIfBsServiceClassIndex,
57             wmanIfBsSfState,
58             wmanIfBsSfProvisionedTime,
59             wmanIfBsProvisionedSfRowStatus,
60             wmanIfBsSsProvisionedForSfRowStatus,
61

```

```

1      wmanIfBsSfCsSpecification,
2      wmanIfBsQoSServiceClassName,
3      wmanIfBsQOSTrafficPriority,
4      wmanIfBsQOSMaxSustainedRate,
5      wmanIfBsQOSMaxTrafficBurst,
6      wmanIfBsQOSMinReservedRate,
7      wmanIfBsQOSToleratedJitter,
8      wmanIfBsQOSMaxLatency,
9      wmanIfBsQOSFixedVsVariableSduInd,
10     wmanIfBsQOSSduSize,
11     wmanIfBsQOSScSchedulingType,
12     wmanIfBsQOSScArqEnable,
13     wmanIfBsQOSScArqWindowSize,
14     wmanIfBsQOSScArqBlockLifetime,
15     wmanIfBsQOSScArqSyncLossTimeout,
16     wmanIfBsQOSScArqDeliverInOrder,
17     wmanIfBsQOSScArqRxPurgeTimeout,
18     wmanIfBsQOSScArqBlockSize,
19     wmanIfBsQOSSCMinRsvdTolerableRate,
20     wmanIfBsQOSReqTxPolicy,
21     wmanIfBsQOSServiceClassRowStatus,
22
23     -- Classification
24     wmanIfBsClassifierRulePriority,
25     wmanIfBsClassifierRuleIpTosLow,
26     wmanIfBsClassifierRuleIpTosHigh,
27     wmanIfBsClassifierRuleIpTosMask,
28     wmanIfBsClassifierRuleIpProtocol,
29     wmanIfBsClassifierRuleIpSourceAddr,
30     wmanIfBsClassifierRuleIpSourceMask,
31     wmanIfBsClassifierRuleIpDestAddr,
32     wmanIfBsClassifierRuleIpDestMask,
33     wmanIfBsClassifierRuleSourcePortStart,
34     wmanIfBsClassifierRuleSourcePortEnd,
35     wmanIfBsClassifierRuleDestPortStart,
36     wmanIfBsClassifierRuleDestPortEnd,
37     wmanIfBsClassifierRuleDestMacAddr,
38     wmanIfBsClassifierRuleDestMacMask,
39     wmanIfBsClassifierRuleSourceMacAddr,
40     wmanIfBsClassifierRuleSourceMacMask,
41     wmanIfBsClassifierRuleEnetProtocolType,
42     wmanIfBsClassifierRuleEnetProtocol,
43     wmanIfBsClassifierRuleUserPriLow,
44     wmanIfBsClassifierRuleUserPriHigh,
45     wmanIfBsClassifierRuleVlanId,
46     wmanIfBsClassifierRuleState,
47     wmanIfBsClassifierRulePhsSize,
48     wmanIfBsClassifierRulePhsMask,
49     wmanIfBsClassifierRulePhsVerify,
50     wmanIfBsClassifierRuleIpv6FlowLabel,
51     wmanIfBsClassifierRuleBitMap,
52     wmanIfBsClassifierRuleRowStatus,
53
54     -- Packet counters
55
56
57
58
59
60
61
62
63
64
65

```

```

1      wmanIfBsSsMacSduCount,
2      wmanIfBsSsOctetCount,
3      wmanIfBsSsResetCounter,
4      wmanIfBsSsResetCounterTime,
5
6
7      -- Capability negotiation
8      wmanIfBsSsBasicCid,
9      wmanIfBsSsPrimaryCid,
10     wmanIfBsSsSecondaryCid,
11     wmanIfBsSsManagementSupport,
12     wmanIfBsSsIpManagementMode,
13     wmanIfBs2ndMgmtDlQoSProfileIndex,
14     wmanIfBs2ndMgmtUlQoSProfileIndex,
15     wmanIfBsAutoSfidEnabled,
16     wmanIfBsAutoSfidRangeMin,
17     wmanIfBsAutoSfidRangeMax,
18     wmanIfBsResetSector,
19     wmanIfBsSs2ndMgmtArqEnable,
20     wmanIfBsSs2ndMgmtArqWindowSize,
21     wmanIfBsSs2ndMgmtArqDnLinkTxDelay,
22     wmanIfBsSs2ndMgmtArqUpLinkTxDelay,
23     wmanIfBsSs2ndMgmtArqDnLinkRxDelay,
24     wmanIfBsSs2ndMgmtArqUpLinkRxDelay,
25     wmanIfBsSs2ndMgmtArqBlockLifetime,
26     wmanIfBsSs2ndMgmtArqSyncLossTimeout,
27     wmanIfBsSs2ndMgmtArqDeliverInOrder,
28     wmanIfBsSs2ndMgmtArqRxPurgeTimeout,
29     wmanIfBsSs2ndMgmtArqBlockSize,
30     wmanIfBsSsVendorIdEncoding,
31     wmanIfBsSsAasBroadcastPermission,
32     wmanIfBsSsMaxTxPowerBpsk,
33     wmanIfBsSsMaxTxPowerQpsk,
34     wmanIfBsSsMaxTxPower16Qam,
35     wmanIfBsSsMaxTxPower64Qam,
36
37
38     -- Configuration parameters
39     wmanIfBsSsMacVersion,
40     wmanIfBsDcdInterval,
41     wmanIfBsUcdInterval,
42     wmanIfBsUcdTransition,
43     wmanIfBsDcdTransition,
44     wmanIfBsInitialRangingInterval,
45     wmanIfBsSsULMapProcTime,
46     wmanIfBsSsRangRespProcTime,
47     wmanIfBsT5Timeout,
48     wmanIfBsT9Timeout,
49     wmanIfBsT13Timeout,
50     wmanIfBsT15Timeout,
51     wmanIfBsT17Timeout,
52     wmanIfBsT27IdleTimer,
53     wmanIfBsT27ActiveTimer,
54
55
56     -- Performance monitoring
57     wmanIfBsHistogramIndex,
58
59
60
61
62
63
64
65

```

```

1      wmanIfBsChannelNumber,
2      wmanIfBsStartFrame,
3      wmanIfBsDuration,
4      wmanIfBsBasicReport,
5      wmanIfBsMeanCinrReport,
6      wmanIfBsMeanRssiReport,
7      wmanIfBsStdDeviationCinrReport,
8      wmanIfBsStdDeviationRssiReport,
9
10
11
12      -- Capability negotiation
13      wmanIfBsSsReqCapUplinkCidSupport,
14      wmanIfBsSsReqCapArqSupport,
15      wmanIfBsSsReqCapDsxFlowControl,
16      wmanIfBsSsReqCapMacCrcSupport,
17      wmanIfBsSsReqCapMcaFlowControl,
18      wmanIfBsSsReqCapMcpGroupCidSupport,
19      wmanIfBsSsReqCapPkmFlowControl,
20      wmanIfBsSsReqCapAuthPolicyControl,
21      wmanIfBsSsReqCapMaxNumOfSupportedSA,
22      wmanIfBsSsReqCapIpVersion,
23      wmanIfBsSsReqCapMacCsSupportBitMap,
24      wmanIfBsSsReqCapMaxNumOfClassifier,
25      wmanIfBsSsReqCapPhsSupport,
26      wmanIfBsSsReqCapBandwidthAllocSupport,
27      wmanIfBsSsReqCapPduConstruction,
28      wmanIfBsSsReqCapTtgTransitionGap,
29      wmanIfBsSsReqCapRtgTransitionGap,
30      wmanIfBsSsRspCapUplinkCidSupport,
31      wmanIfBsSsRspCapArqSupport,
32      wmanIfBsSsRspCapDsxFlowControl,
33      wmanIfBsSsRspCapMacCrcSupport,
34      wmanIfBsSsRspCapMcaFlowControl,
35      wmanIfBsSsRspCapMcpGroupCidSupport,
36      wmanIfBsSsRspCapPkmFlowControl,
37      wmanIfBsSsRspCapAuthPolicyControl,
38      wmanIfBsSsRspCapMaxNumOfSupportedSA,
39      wmanIfBsSsRspCapIpVersion,
40      wmanIfBsSsRspCapMacCsSupportBitMap,
41      wmanIfBsSsRspCapMaxNumOfClassifier,
42      wmanIfBsSsRspCapPhsSupport,
43      wmanIfBsSsRspCapBandwidthAllocSupport,
44      wmanIfBsSsRspCapPduConstruction,
45      wmanIfBsSsRspCapTtgTransitionGap,
46      wmanIfBsSsRspCapRtgTransitionGap,
47      wmanIfBsCapUplinkCidSupport,
48      wmanIfBsCapArqSupport,
49      wmanIfBsCapDsxFlowControl,
50      wmanIfBsCapMacCrcSupport,
51      wmanIfBsCapMcaFlowControl,
52      wmanIfBsCapMcpGroupCidSupport,
53      wmanIfBsCapPkmFlowControl,
54      wmanIfBsCapAuthPolicyControl,
55      wmanIfBsCapMaxNumOfSupportedSA,
56      wmanIfBsCapIpVersion,
57
58
59
60
61
62
63
64
65

```

```

1      wmanIfBsCapMacCsSupportBitMap,
2      wmanIfBsCapMaxNumOfClassifier,
3      wmanIfBsCapPhsSupport,
4      wmanIfBsCapBandwidthAllocSupport,
5      wmanIfBsCapPduConstruction,
6      wmanIfBsCapTtgTransitionGap,
7      wmanIfBsCapRtgTransitionGap,
8      wmanIfBsCapCfgUplinkCidSupport,
9      wmanIfBsCapCfgArqSupport,
10     wmanIfBsCapCfgDsxFLOWControl,
11     wmanIfBsCapCfgMacCrcSupport,
12     wmanIfBsCapCfgMcaFlowControl,
13     wmanIfBsCapCfgMcpGroupCidSupport,
14     wmanIfBsCapCfgPkmFlowControl,
15     wmanIfBsCapCfgAuthPolicyControl,
16     wmanIfBsCapCfgMaxNumOfSupportedSA,
17     wmanIfBsCapCfgIpVersion,
18     wmanIfBsCapCfgMacCsSupportBitMap,
19     wmanIfBsCapCfgMaxNumOfClassifier,
20     wmanIfBsCapCfgPhsSupport,
21     wmanIfBsCapCfgBandwidthAllocSupport,
22     wmanIfBsCapCfgPduConstruction,
23     wmanIfBsCapCfgTtgTransitionGap,
24     wmanIfBsCapCfgRtgTransitionGap,
25     wmanIfBsSsActionsResetSs,
26     wmanIfBsSsActionsAbortSs,
27     wmanIfBsSsActionsOverrideDnFreq,
28     wmanIfBsSsActionsOverrideChannelId,
29     wmanIfBsSsActionsDeReRegSs,
30     wmanIfBsSsActionsDeReRegSsCode,
31     wmanIfBsSsActionsRowStatus,
32
33     -- Privacy sublayer
34     wmanIfBsPkmDefaultAuthLifetime,
35     wmanIfBsPkmDefaultTekLifetime,
36     wmanIfBsPkmDefaultSelfSigManufCertTrust,
37     wmanIfBsPkmCheckCertValidityPeriods,
38     wmanIfBsPkmAuthentInfos,
39     wmanIfBsPkmAuthRequests,
40     wmanIfBsPkmAuthReplies,
41     wmanIfBsPkmAuthRejects,
42     wmanIfBsPkmAuthInvalids,
43     wmanIfBsSsPkmAuthKeySequenceNumber,
44     wmanIfBsSsPkmAuthExpiresOld,
45     wmanIfBsSsPkmAuthExpiresNew,
46     wmanIfBsSsPkmAuthLifetime,
47     wmanIfBsSsPkmAuthReset,
48     wmanIfBsSsPkmAuthInfos,
49     wmanIfBsSsPkmAuthRequests,
50     wmanIfBsSsPkmAuthReplies,
51     wmanIfBsSsPkmAuthRejects,
52     wmanIfBsSsPkmAuthInvalids,
53     wmanIfBsSsPkmAuthRejectErrorCode,
54     wmanIfBsSsPkmAuthRejectErrorString,
55
56
57
58
59
60
61
62
63
64
65

```

```

1      wmanIfBsSsPkmAuthInvalidErrorCode,
2      wmanIfBsSsPkmAuthInvalidErrorString,
3      wmanIfBsSsPkmAuthPrimarySAId,
4      wmanIfBsSsPkmAuthValidStatus,
5      wmanIfBsPkmTekSAType,
6      wmanIfBsPkmTekDataEncryptAlg,
7      wmanIfBsPkmTekDataAuthentAlg,
8      wmanIfBsPkmTekEncryptAlg,
9      wmanIfBsPkmTekLifetime,
10     wmanIfBsPkmTekKeySequenceNumber,
11     wmanIfBsPkmTekExpiresOld,
12     wmanIfBsPkmTekExpiresNew,
13     wmanIfBsPkmTekReset,
14     wmanIfBsPkmKeyRequests,
15     wmanIfBsPkmKeyReplies,
16     wmanIfBsPkmKeyRejects,
17     wmanIfBsPkmTekInvalids,
18     wmanIfBsPkmKeyRejectErrorCode,
19     wmanIfBsPkmKeyRejectErrorString,
20     wmanIfBsPkmTekInvalidErrorCode,
21     wmanIfBsPkmTekInvalidErrorString,
22
23     -- Notification
24     wmanIfBsTrapControlRegister,
25     wmanIfBsStatusTrapControlRegister,
26     wmanIfBsRssiLowThreshold,
27     wmanIfBsRssiHighThreshold,
28     wmanIfBsSsNotificationMacAddr,
29     wmanIfBsSsStatusValue,
30     wmanIfBsSsStatusInfo,
31     wmanIfBsDynamicServiceType,
32     wmanIfBsDynamicServiceFailReason,
33     wmanIfBsSsRssiStatus,
34     wmanIfBsSsRssiStatusInfo,
35     wmanIfBsSsRegisterStatus}
36
37 STATUS      current
38
39 DESCRIPTION
40     "This group contains objects for BS, and are
41     independent of PHY."
42 ::= { wmanIfMibGroups 3 }
43
44 wmanIfMibBsAasGroup      OBJECT-GROUP
45     OBJECTS {-- AAS Configuration parameters
46         wmanIfBsAasChanFbckReqFreq,
47         wmanIfBsAasBeamSelectFreq,
48         wmanIfBsAasChanFbckReqResolution,
49         wmanIfBsAasBeamReqResolution,
50         wmanIfBsAasNumOptDiversityZones}
51
52 STATUS      current
53
54 DESCRIPTION
55     "This group contains objects for AAS in BS."
56 ::= { wmanIfMibGroups 4 }
57
58 wmanIfMibSsGroup         OBJECT-GROUP
59
60
61
62
63
64
65

```

```

1      OBJECTS {-- Configuration parameters
2          wmanIfSsLostDLMapInterval,
3          wmanIfSsLostULMapInterval,
4          wmanIfSsContentionRangRetries,
5          wmanIfSsRequestRetries,
6          wmanIfSsRegRequestRetries,
7          wmanIfSsTftpBackoffStart,
8          wmanIfSsTftpBackoffEnd,
9          wmanIfSsTftpRequestRetries,
10         wmanIfSsTftpDownloadRetries,
11         wmanIfSsTftpWait,
12         wmanIfSsToDRetries,
13         wmanIfSsToDRetryPeriod,
14         wmanIfSsT1Timeout,
15         wmanIfSsT2Timeout,
16         wmanIfSsT3Timeout,
17         wmanIfSsT4Timeout,
18         wmanIfSsT6Timeout,
19         wmanIfSsT12Timeout,
20         wmanIfSsT14Timeout,
21         wmanIfSsT16Timeout,
22         wmanIfSsT18Timeout,
23         wmanIfSsT19Timeout,
24         wmanIfSsT20Timeout,
25         wmanIfSsT21Timeout,
26         wmanIfSsSBCRequestRetries,
27         wmanIfSsTftpCpltRetries,
28         wmanIfSsT26Timeout,
29         wmanIfSsDLManagProcTime,
30
31         -- Performance monitoring
32         wmanIfSsChannelNumber,
33         wmanIfSsStartFrame ,
34         wmanIfSsDuration,
35         wmanIfSsBasicReport,
36         wmanIfSsMeanCinrReport,
37         wmanIfSsStdDeviationCinrReport,
38         wmanIfSsMeanRssiReport,
39         wmanIfSsStdDeviationRssiReport,
40
41         -- Privacy sublayer
42         wmanIfSsPkmAuthState,
43         wmanIfSsPkmAuthKeySequenceNumber,
44         wmanIfSsPkmAuthExpiresOld,
45         wmanIfSsPkmAuthExpiresNew ,
46         wmanIfSsPkmAuthReset,
47         wmanIfSsPkmAuthentInfos,
48         wmanIfSsPkmAuthRequests,
49         wmanIfSsPkmAuthReplies,
50         wmanIfSsPkmAuthRejects,
51         wmanIfSsPkmAuthInvalids,
52         wmanIfSsPkmAuthRejectErrorCode,
53         wmanIfSsPkmAuthRejectErrorString,
54         wmanIfSsPkmAuthInvalidErrorCode,
55
56
57
58
59
60
61
62
63
64
65

```

```

1      wmanIfSsPkmAuthInvalidErrorString ,
2      wmanIfSsPkmAuthGraceTime,
3      wmanIfSsPkmTekGraceTime,
4      wmanIfSsPkmAuthWaitTimeout,
5      wmanIfSsPkmReauthWaitTimeout,
6      wmanIfSsPkmOpWaitTimeout,
7      wmanIfSsPkmRekeyWaitTimeout,
8      wmanIfSsPkmAuthRejectWaitTimeout,
9
10     wmanIfSsPkmTekSAType,
11     wmanIfSsPkmTekDataEncryptAlg,
12     wmanIfSsPkmTekDataAuthentAlg,
13     wmanIfSsPkmTekEncryptAlg,
14     wmanIfSsPkmTekState,
15     wmanIfSsPkmTekKeySequenceNumber,
16     wmanIfSsPkmTekExpiresOld,
17     wmanIfSsPkmTekExpiresNew,
18     wmanIfSsPkmTekKeyRequests,
19     wmanIfSsPkmTekKeyReplies,
20     wmanIfSsPkmTekKeyRejects,
21     wmanIfSsPkmTekInvalids,
22     wmanIfSsPkmTekAuthPends,
23     wmanIfSsPkmTekKeyRejectErrorCode,
24     wmanIfSsPkmTekKeyRejectErrorString,
25     wmanIfSsPkmTekInvalidErrorCode,
26     wmanIfSsPkmTekInvalidErrorString,
27     wmanIfSsDeviceCert,
28     wmanIfSsDeviceManufCert,
29
30     -- Notofocation
31     wmanIfSsTrapControlRegister,
32     wmanIfSsRssiLowThreshold,
33     wmanIfSsRssiHighThreshold,
34     wmanIfSsMacAddress,
35     wmanIfSsUnknownTlv,
36     wmanIfSsDynamicServiceType,
37     wmanIfSsDynamicServiceFailReason,
38     wmanIfSsRssiStatus,
39     wmanIfSsRssiStatusInfo}
40
41 STATUS      current
42
43 DESCRIPTION
44     "This group contains objects for SS, and are
45     independent of PHY."
46 ::= { wmanIfMibGroups 5 }
47
48
49 wmanIfMibBsOfdmGroup      OBJECT-GROUP
50     OBJECTS {wmanIfBsOfdmCtBasedResvTimeout,
51             wmanIfBsOfdmBwReqOppSize,
52             wmanIfBsOfdmRangReqOppSize,
53             wmanIfBsOfdmUplinkCenterFreq,
54             wmanIfBsOfdmNumSubChReqRegionFull,
55             wmanIfBsOfdmNumSymbolsReqRegionFull,
56             wmanIfBsOfdmSubChFocusCtCode,
57             wmanIfBsOfdmUpLinkChannelId,
58             wmanIfBsOfdmBsEIRP,

```

```

1      wmanIfBsOfdmChannelNumber,
2      wmanIfBsOfdmTTG,
3      wmanIfBsOfdmRTG,
4      wmanIfBsOfdmInitRngMaxRSS,
5      wmanIfBsOfdmDownlinkCenterFreq,
6      wmanIfBsOfdmBsId,
7      wmanIfBsOfdmMacVersion,
8      wmanIfBsOfdmFrameDurationCode,
9      wmanIfBsOfdmDownLinkChannelId,
10     wmanIfBsOfdmUcdFecCodeType,
11     wmanIfBsOfdmFocusCtPowerBoost,
12     wmanIfBsOfdmUcdTcsEnable,
13     wmanIfBsOfdmUcdBurstProfileRowStatus,
14     wmanIfBsOfdmDownlinkFrequency,
15     wmanIfBsOfdmDcdFecCodeType,
16     wmanIfBsOfdmDiucMandatoryExitThresh,
17     wmanIfBsOfdmDiucMinEntryThresh,
18     wmanIfBsOfdmTcsEnable,
19     wmanIfBsOfdmDcdBurstProfileRowStatus,
20     wmanIfBsOfdmMinReqRegionFullTxOpp,
21     wmanIfBsOfdmMinFocusedCtTxOpp,
22     wmanIfBsOfdmMaxRoundTripDelay,
23     wmanIfBsOfdmRangeAbortTimingThold,
24     wmanIfBsOfdmRangeAbortPowerThold ,
25     wmanIfBsOfdmRangeAbortFreqThold,
26     wmanIfBsOfdmDnlkRateId,
27     wmanIfBsOfdmRatioG,
28     wmanIfBsSsOfdmReqCapFftSizes,
29     wmanIfBsSsOfdmReqCapSsDemodulator,
30     wmanIfBsSsOfdmReqCapSsModulator,
31     wmanIfBsSsOfdmReqCapFocusedCtSupport,
32     wmanIfBsSsOfdmReqCapTcSublayerSupport,
33     wmanIfBsSsOfdmRspCapFftSizes,
34     wmanIfBsSsOfdmRspCapSsDemodulator,
35     wmanIfBsSsOfdmRspCapSsModulator,
36     wmanIfBsSsOfdmRspCapFocusedCtSupport,
37     wmanIfBsSsOfdmRspCapTcSublayerSupport,
38     wmanIfBsOfdmCapFftSizes,
39     wmanIfBsOfdmCapSsDemodulator,
40     wmanIfBsOfdmCapSsModulator,
41     wmanIfBsOfdmCapFocusedCtSupport,
42     wmanIfBsOfdmCapTcSublayerSupport,
43     wmanIfBsOfdmCapCfgFftSizes,
44     wmanIfBsOfdmCapCfgSsDemodulator,
45     wmanIfBsOfdmCapCfgSsModulator,
46     wmanIfBsOfdmCapCfgFocusedCtSupport,
47     wmanIfBsOfdmCapCfgTcSublayerSupport}
48
49 STATUS      current
50
51 DESCRIPTION
52     "This group contains objects for BS and OFDM PHY."
53     ::= { wmanIfMibGroups 6 }
54
55 wmanIfMibSsOfdmGroup      OBJECT-GROUP
56     OBJECTS {wmanIfSsOfdmCtBasedResvTimeout,
57

```

```

1      wmanIfSsOfdmBwReqOppSize,
2      wmanIfSsOfdmRangReqOppSize,
3      wmanIfSsOfdmUplinkCenterFreq,
4      wmanIfSsOfdmNumSubChReqRegionFull,
5      wmanIfSsOfdmNumSymbolsReqRegionFull,
6      wmanIfSsOfdmSubChFocusCtCode,
7      wmanIfSsOfdmUpLinkChannelId,
8      wmanIfSsOfdmBsEIRP,
9      wmanIfSsOfdmChannelNumber,
10     wmanIfSsOfdmTTG,
11     wmanIfSsOfdmRTG,
12     wmanIfSsOfdmInitRngMaxRSS,
13     wmanIfSsOfdmDownlinkCenterFreq,
14     wmanIfSsOfdmBsId,
15     wmanIfSsOfdmMacVersion,
16     wmanIfSsOfdmFrameDurationCode,
17     wmanIfSsOfdmDownLinkChannelId,
18     wmanIfSsOfdmUcdFecCodeType,
19     wmanIfSsOfdmFocusCtPowerBoost,
20     wmanIfSsOfdmUcdTcsEnable,
21     wmanIfSsOfdmDownlinkFrequency,
22     wmanIfSsOfdmDcdFecCodeType,
23     wmanIfSsOfdmDiucMandatoryExitThresh,
24     wmanIfSsOfdmDiucMinEntryThresh,
25     wmanIfSsOfdmTcsEnable}
26
27 STATUS      current
28
29 DESCRIPTION
30     "This group contains objects for SS and OFDM PHY."
31     ::= { wmanIfMibGroups 7 }
32
33 wmanIfMibBsOfdmaGroup      OBJECT-GROUP
34     OBJECTS {wmanIfBsOfdmaCtBasedResvTimeout,
35             wmanIfBsOfdmaBwReqOppSize,
36             wmanIfBsOfdmaRangReqOppSize,
37             wmanIfBsOfdmaUplinkCenterFreq,
38             wmanIfBsOfdmaInitRngCodes,
39             wmanIfBsOfdmaPeriodicRngCodes,
40             wmanIfBsOfdmaBWReqCodes,
41             wmanIfBsOfdmaPerRngBackoffStart,
42             wmanIfBsOfdmaPerRngBackoffEnd,
43             wmanIfBsOfdmaStartOfRngCodes,
44             wmanIfBsOfdmaPermutationBase,
45             wmanIfBsOfdmaULAllocSubchBitmap,
46             wmanIfBsOfdmaOptPermULAllocSubchBitmap,
47             wmanIfBsOfdmaBandAMCAllocThreshold,
48             wmanIfBsOfdmaBandAMCReleaseThreshold,
49             wmanIfBsOfdmaBandAMCAllocTimer,
50             wmanIfBsOfdmaBandAMCReleaseTimer,
51             wmanIfBsOfdmaBandStatRepMAXPeriod,
52             wmanIfBsOfdmaBandAMCRetryTimer,
53             wmanIfBsOfdmaSafetyChAllocThreshold,
54             wmanIfBsOfdmaSafetyChReleaseThreshold,
55             wmanIfBsOfdmaSafetyChAllocTimer,
56             wmanIfBsOfdmaSafetyChReleaseTimer,
57
58
59
60
61
62
63
64
65

```

```

1      wmanIfBsOfdmaBinStatRepMAXPeriod,
2      wmanIfBsOfdmaSafetyChaRetryTimer,
3      wmanIfBsOfdmaHARQAackDelayULBurst,
4      wmanIfBsOfdmaCQICHBandAMCTranaDelay,
5      wmanIfBsOfdmaBsEIRP,
6      wmanIfBsOfdmaChannelNumber,
7      wmanIfBsOfdmaTTG,
8      wmanIfBsOfdmaRTG,
9      wmanIfBsOfdmaInitRngMaxRSS,
10     wmanIfBsOfdmaDownlinkCenterFreq,
11     wmanIfBsOfdmaBsId,
12     wmanIfBsOfdmaMacVersion,
13     wmanIfBsOfdmaFrameDurationCode,
14     wmanIfBsOfdmaSizeCqichIdField,
15     wmanIfBsOfdmaHARQAackDelayBurst,
16     wmanIfBsOfdmaUcdFecCodeType,
17     wmanIfBsOfdmaRangingDataRatio,
18     wmanIfBsOfdmaNorCOverNOVERRIDE,
19     wmanIfBsOfdmaUcdBurstProfileRowStatus,
20     wmanIfBsOfdmaDownlinkFrequency,
21     wmanIfBsOfdmaDcdFecCodeType,
22     wmanIfBsOfdmaDiucMandatoryExitThresh,
23     wmanIfBsOfdmaDiucMinEntryThresh,
24     wmanIfBsOfdmaDcdBurstProfileRowStatus}
25
26 STATUS      current
27
28 DESCRIPTION
29     "This group contains objects for BS and OFDMA PHY."
30     ::= { wmanIfMibGroups 8 }
31
32 wmanIfMibSsOfdmaGroup      OBJECT-GROUP
33     OBJECTS {wmanIfSsOfdmaCtBasedResvTimeout,
34             wmanIfSsOfdmaBwReqOppSize,
35             wmanIfSsOfdmaRangReqOppSize,
36             wmanIfSsOfdmaUplinkCenterFreq,
37             wmanIfSsOfdmaInitRngCodes,
38             wmanIfSsOfdmaPeriodicRngCodes,
39             wmanIfSsOfdmaBWReqCodes,
40             wmanIfSsOfdmaPerRngBackoffStart,
41             wmanIfSsOfdmaPerRngBackoffEnd,
42             wmanIfSsOfdmaStartOfRngCodes,
43             wmanIfSsOfdmaPermutationBase,
44             wmanIfSsOfdmaULAllocSubchBitmap,
45             wmanIfSsOfdmaOptPermULAllocSubchBitmap,
46             wmanIfSsOfdmaBandAMCAllocThreshold,
47             wmanIfSsOfdmaBandAMCReleaseThreshold,
48             wmanIfSsOfdmaBandAMCAllocTimer,
49             wmanIfSsOfdmaBandAMCReleaseTimer,
50             wmanIfSsOfdmaBandStatRepMAXPeriod,
51             wmanIfSsOfdmaBandAMCRetryTimer,
52             wmanIfSsOfdmaSafetyChAllocThreshold,
53             wmanIfSsOfdmaSafetyChReleaseThreshold,
54             wmanIfSsOfdmaSafetyChAllocTimer,
55             wmanIfSsOfdmaSafetyChReleaseTimer,
56             wmanIfSsOfdmaBinStatRepMAXPeriod,

```

```

1      wmanIfSsOfdmaSafetyChaRetryTimer,
2      wmanIfSsOfdmaHARQAackDelayULBurst,
3      wmanIfSsOfdmaCQICHBandAMCTranaDelay,
4      wmanIfSsOfdmaBsEIRP,
5      wmanIfSsOfdmaChannelNumber,
6      wmanIfSsOfdmaTTG,
7      wmanIfSsOfdmaRTG,
8      wmanIfSsOfdmaInitRngMaxRSS,
9      wmanIfSsOfdmaDownlinkCenterFreq,
10     wmanIfSsOfdmaBsId,
11     wmanIfSsOfdmaMacVersion,
12     wmanIfSsOfdmaFrameDurationCode,
13     wmanIfSsOfdmaSizeCqichIdField,
14     wmanIfSsOfdmaHARQAackDelayBurst,
15     wmanIfSsOfdmaUiucIndex,
16     wmanIfSsOfdmaUcdFecCodeType,
17     wmanIfSsOfdmaRangingDataRatio,
18     wmanIfSsOfdmaNorCOverNOVERRIDE,
19     wmanIfSsOfdmaDiucIndex,
20     wmanIfSsOfdmaDownlinkFrequency,
21     wmanIfSsOfdmaDcdFecCodeType,
22     wmanIfSsOfdmaDiucMandatoryExitThresh,
23     wmanIfSsOfdmaDiucMinEntryThresh}
24
25     STATUS          current
26     DESCRIPTION
27         "This group contains objects for SS and OFDMA PHY."
28     ::= { wmanIfMibGroups 9 }
29
30 wmanIfMibBsNotificationGroup    NOTIFICATION-GROUP
31     NOTIFICATIONS {wmanIfBsSsStatusNotificationTrap,
32                     wmanIfBsSsDynamicServiceFailTrap,
33                     wmanIfBsSsRssiStatusChangeTrap,
34                     wmanIfBsSsPkmFailTrap,
35                     wmanIfBsSsRegistrerTrap}
36
37     STATUS          current
38     DESCRIPTION
39         "This group contains event notifications for BS."
40     ::= { wmanIfMibGroups 10 }
41
42 wmanIfMibSsNotificationGroup    NOTIFICATION-GROUP
43     NOTIFICATIONS {wmanIfSsTlvUnknownTrap,
44                     wmanIfSsDynamicServiceFailTrap,
45                     wmanIfSsDhcpSuccessTrap,
46                     wmanIfSsRssiStatusChangeTrap}
47
48     STATUS          current
49     DESCRIPTION
50         "This group contains event notifications for SS."
51     ::= { wmanIfMibGroups 11 }
52
53 wmanIfMibCmnPhsGroup           OBJECT-GROUP
54     OBJECTS {-- Payload header supression
55               wmanIfCmnPhsRulePhsField,
56               wmanIfCmnPhsRulePhsMask,
57               wmanIfCmnPhsRulePhsSize,
58

```

```

1          wmanIfCmnPhsRulePhsVerify}
2      STATUS          current
3      DESCRIPTION
4          "This group contains common objects for PHS."
5      ::= { wmanIfMibGroups 12 }
6
7
8      wmanIfMibBsPhsGroup      OBJECT-GROUP
9          OBJECTS {-- Payload header supression
10              wmanIfBsClassifierRulePhsSize,
11              wmanIfBsClassifierRulePhsMask,
12              wmanIfBsClassifierRulePhsVerify,
13              wmanIfBsClassifierRuleBitMap}
14
15      STATUS          current
16      DESCRIPTION
17          "This group contains BS objects for PHS."
18      ::= { wmanIfMibGroups 13 }
19
20  END

```

Annex E.

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

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

1. Introduction

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

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

2. Proposed Text Introduction

2.1 wmanIfBsObjects

2.1.1 wmanIfBsMobility

2.1.1.1 wmanIfBsHandoverConfiguration

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

2.1.1.2 wmanIfBsPagingGroupTable

wmanIfBsPagingGroupTable contains paging group related parameters

3. ASN.1 Definitions of 802.16 MIB for SNMP

```

1  wmanIfBsMobility OBJECT IDENTIFIER ::= { wmanIfBsObjects 1 }
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 }

wmanIfBsMsHOCOnnectProcessingTime 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 { 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 wmanIfNeighbourBsPreambleIndex OBJECT-TYPE
17     SYNTAX Integer32
18     MAX-ACCESS read-write
19     STATUS current
20     DESCRIPTION
21        "This parameter is used to indicate neighbouring BS preamble index."
22    ::= { wmanIfNeighbourBsEntry 13 }
23
24
25
26 wmanIfBsPagingGroupTable OBJECT-TYPE
27     SYNTAX SEQUENCE OF WmanIfBsPagingGroupEntry
28     MAX-ACCESS not-accessible
29     STATUS current
30     DESCRIPTION
31        "This table contains paging group related parameters."
32    ::= { wmanIfBsMobility 3 }
33
34
35
36 wmanIfBsPagingGroupEntry OBJECT-TYPE
37     SYNTAX WmanIfBsPagingGroupEntry
38     MAX-ACCESS not-accessible
39     STATUS current
40     DESCRIPTION
41        "This table is indexed by wmanIfBsPagingGroupId."
42     INDEX { wmanIfBsPagingGroupId }
43     ::= { wmanIfBsPagingGroupTable 1 }
44
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     wmanIfBsPagingIntervallLength INTEGER,
60     wmanIfBsPagingCycle          INTEGER
61 }
62
63
64
65

```

```

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

```

1 DESCRIPTION
2 "Waiting value for the DREG-REQ message re-transmission
3 (measured in frames)."
4 ::= { wmanIfBsPagingGroupEntry 6 }

7 wmanIfBsMACHashSkipThreshold OBJECT-TYPE
8 SYNTAX Integer32
9 MAX-ACCESS read-write
10 STATUS current
11 DESCRIPTION
12 "Maximum number of successive MOB_PAG-ADV messages
13 that may be sent from a BS without individual notification for
14 an MS for which BS is allowed to skip MS MAC Address Hash
15 when the Action Code for the MS is 0b00,'No Action Required'.
16 ::= { wmanIfBsPagingGroupEntry 7 }

19 wmanIfBsCDMATransmissionOpportunityAssignment OBJECT-TYPE
20 SYNTAX INTEGER
21 MAX-ACCESS read-write
22 STATUS current
23 DESCRIPTION
24 "The CDMA code and transmission opportunity
25 assignment field indicates the assigned code
26 and transmission opportunity for a MS who is
27 paged to use over dedicated CDMA ranging region."
28 ::= { wmanIfBsPagingGroupEntry 8 }

31 wmanIfBsPagingResponseWindow OBJECT-TYPE
32 SYNTAX INTEGER
33 MAX-ACCESS read-write
34 STATUS current
35 DESCRIPTION
36 "The Page-Response Window indicates the Page-Response window for a MS
37 who is paged to transmit
38 the assigned code for CDMA ranging channel."
39 ::= { wmanIfBsPagingGroupEntry 9 }

42 wmanIfBsIdleModeTimer OBJECT-TYPE
43 SYNTAX INTEGER (128..65536)
44 MAX-ACCESS read-write
45 STATUS current
46 DESCRIPTION
47 "MS timed interval to conduct
48 Location Update. Set timer to MS
49 Idle Mode Timeout capabilities
50 setting. Timer recycles on successful
51 Idle Mode Location Update."
52 ::= { wmanIfBsPagingGroupEntry 10 }

55 wmanIfBsIdleModeSystemTimer OBJECT-TYPE
56 SYNTAX INTEGER (128..65536)
57 MAX-ACCESS read-write
58 STATUS current
59 DESCRIPTION

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

::= { wmanIfBsPagingGroupEntry 11 }

wmanIfBsPagingIntervalLength OBJECT-TYPE

SYNTAX INTEGER (2..5)

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"time duration of Paging Interval
of the BS."

::= { wmanIfBsPagingGroupEntry 12 }

wmanIfBsPagingCycle OBJECT-TYPE

SYNTAX INTEGER

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Cycle in which the paging message is transmitted
within the paging group."

::= { wmanIfBsPagingGroupEntry 13 }