

2006-02-07

IEEE 802.16i-06/001, February 2006

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

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

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
15. IRP Definitions .....	7
15.1 NRM IRP IS .....	7
15.1.1 Information Service Models .....	7
15.1.1.1 Information entities imported and local labels .....	7
15.1.1.2 Class diagram .....	8
15.1.1.3 Information object classes definition .....	10
15.1.1.4 Information relationships definition .....	11
15.1.1.5 Notifications .....	11
15.1.1.6 Information attributes definition .....	11
15.1.2 Proposal for BS Related Objects NRM Definitions .....	12
15.1.2.1 Information entities imported and local labels: .....	12
15.1.2.2 Class diagram .....	13
15.1.2.3 Information object classes definition .....	20
15.1.2.4 Information relationships definition .....	23
15.1.2.5 Notifications .....	24
15.1.2.6 Information attributes definition .....	24
15.2 NRM IRP SNMP Solution Set .....	27
15.3 NRM IRP CORBA Solution Set .....	27
15.4 NRM IRP XML Solution Set .....	27
15.5 Interface IRP Considerations .....	27
Annex E. ASN.1 Definition of wmanIfMib in 802.16f .....	28
Annex F. Proposal for Adding Mobility Handover and Paging group MIBs .....	227
1. Introduction .....	227
2. Proposed Text Introduction .....	227
2.1 wmanIfBsObjects .....	227
2.1.1 wmanIfBsMobility .....	227
2.1.1.1 wmanIfBsHandoverConfiguration .....	227
2.1.1.2 wmanIfBsPagingGroupTable .....	227
3. ASN.1 Definitions of 802.16 MIB for SNMP .....	228

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—Containment and Naming Diagram.....	8
Figure 5—Inheritance Diagram .....	9
Figure 6—General View.....	14
Figure 7—Segment view Neighbouring BS NRM Containment/Naming Diagram .....	15
Figure 8—Segment view PacketCs NRM Containment/Naming Diagram.....	16
Figure 9—Segment view SecurityMng NRM Containment/Naming Diagram.....	17
Figure 10—Segment view CPSPMngMng NRM Containment/Naming Diagram .....	18
Figure 11—Segment view PhyMng NRM Containment/Naming Diagram.....	19
Figure 12—Inheritance Diagram .....	20

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—Information entities imported and local labels .....	8
Table 2—Attributes .....	10
Table 3—Attributes .....	11
Table 5—Information entities imported and local labels .....	12
Table 4—Definition and legal values .....	12
Table 6—Attributes of BSFunction .....	20
Table 7—Attributes of ExternalBSFunction .....	21
Table 8—Attributes of BSRelation .....	22
Table 9—Attributes of PagingGroup .....	23
Table 10—Roles of the relation ExternalNeighbourBSRelation .....	23
Table 11—Information attributes definition .....	24

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: Mobile Management Information Base

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

### 1.2 Purpose

### 1.3 Reference Models

1.3.1 management Reference Models

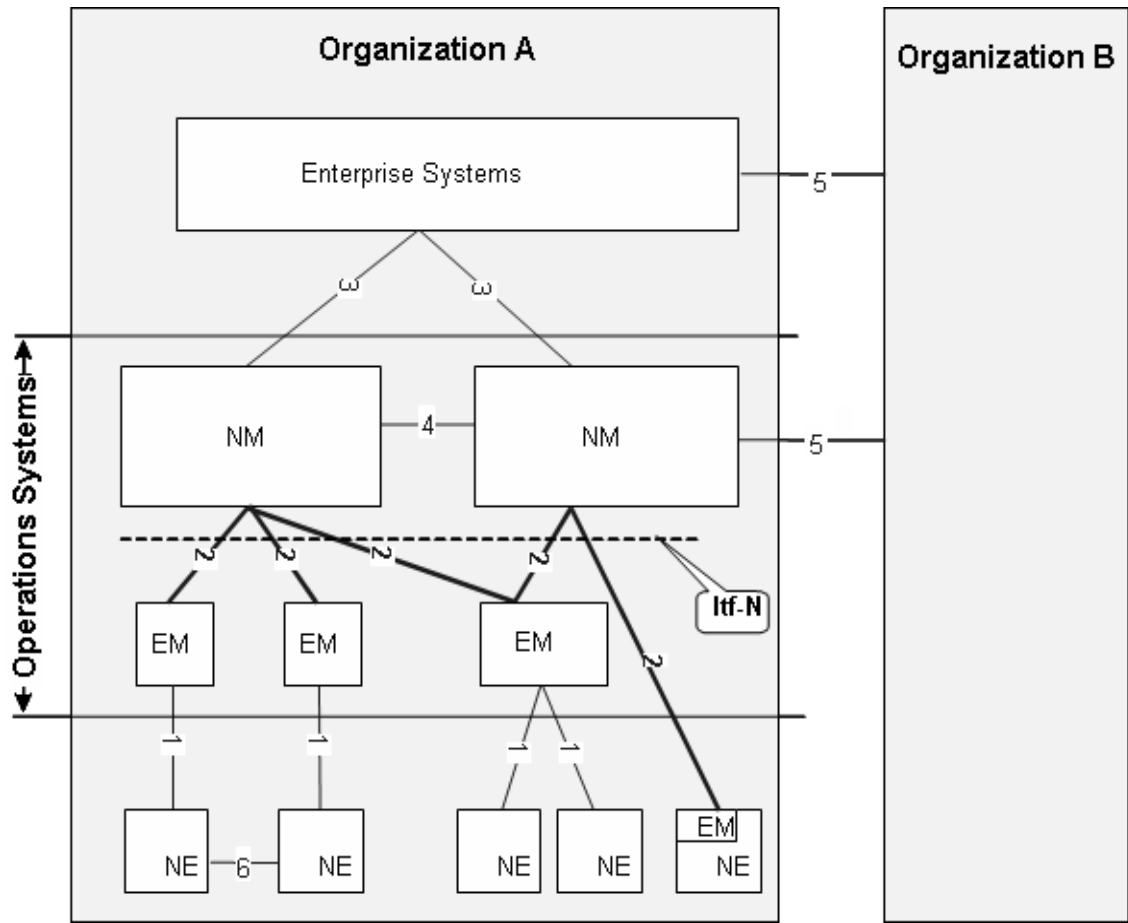


Figure 1—Mobile BWA Network Management Layer Topology

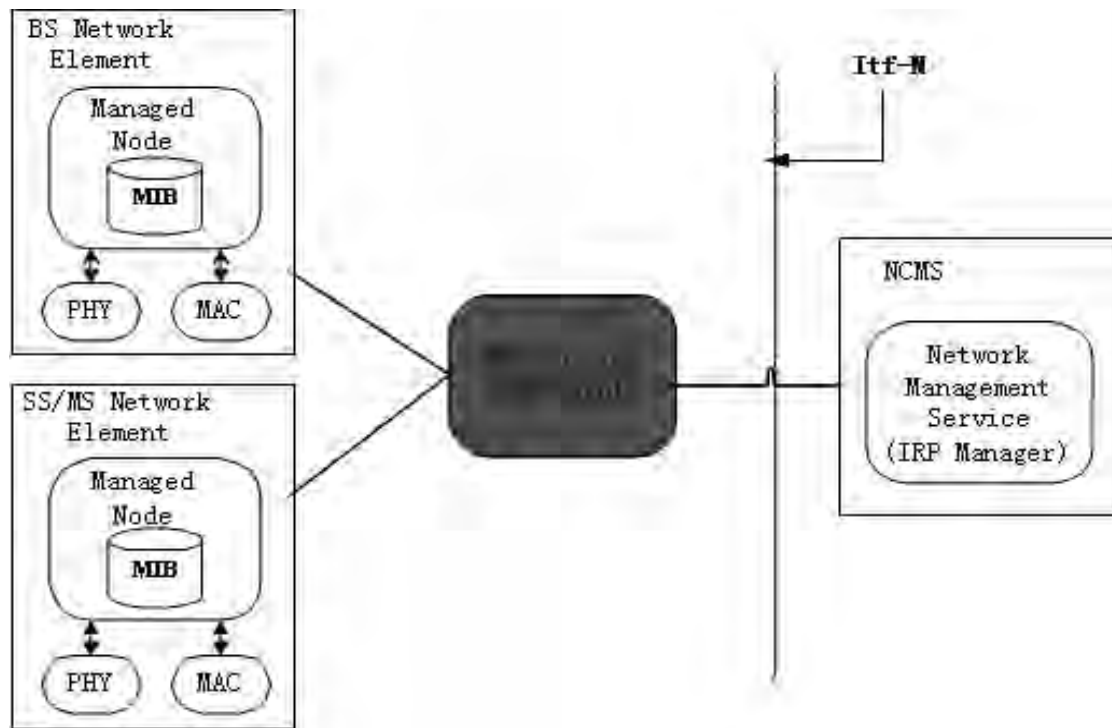


Figure 2—Mobile BWA Network Management Architecture (I)

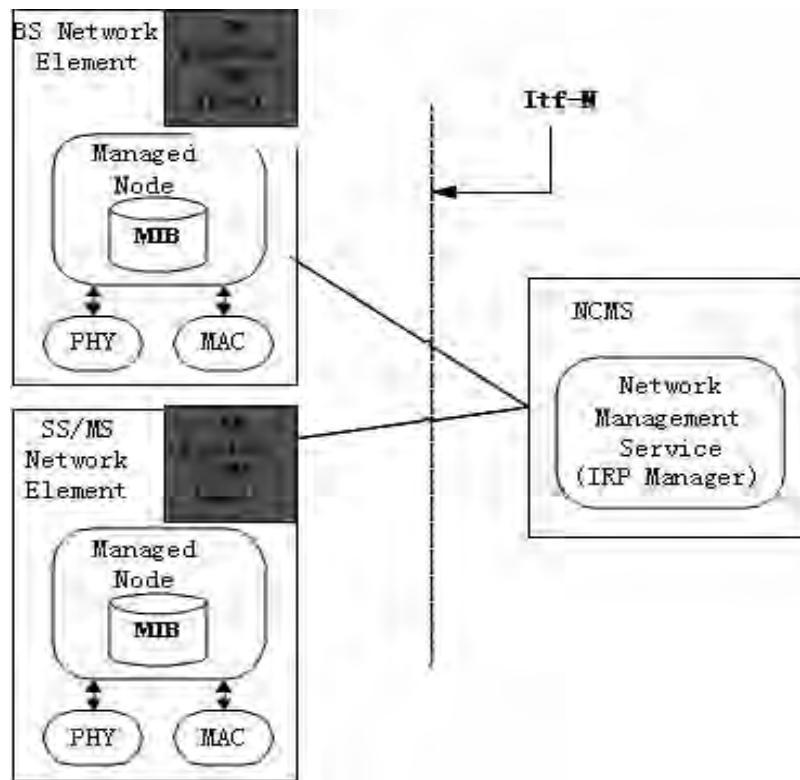


Figure 3—Mobile BWA Network Management Architecture (II)



## 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.

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

**9. Configuration**

## 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 1—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.

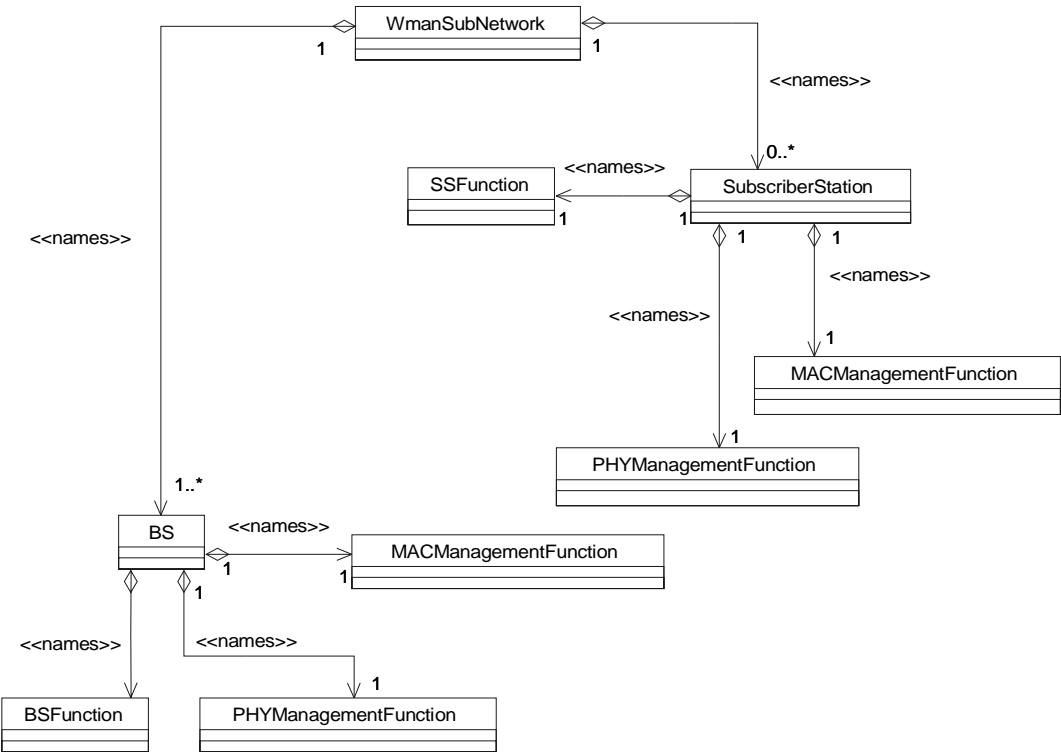
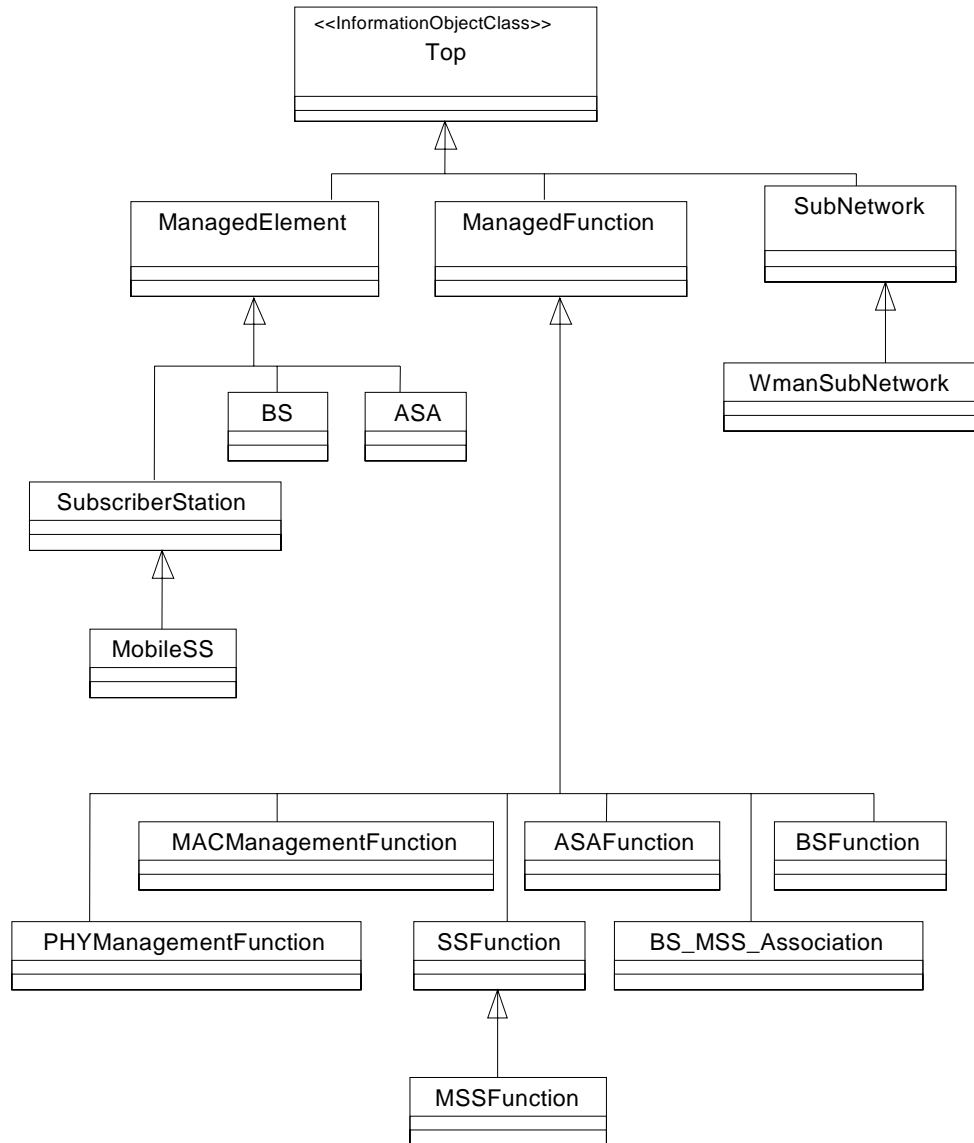


Figure 4—Containment and Naming Diagram

### 15.1.1.2.2 Inheritance

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



**Figure 5—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 2—Attributes**

Attribute name	Defined in	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
BsFunctionId	--	+	M	M	--
objectClass	Top	+ <sub>inherited</sub>	M <sub>inherited</sub>	M <sub>inherited</sub>	-- <sub>inherited</sub>
objectInstance	Top	+ <sub>inherited</sub>	M <sub>inherited</sub>	M <sub>inherited</sub>	-- <sub>inherited</sub>
userLabel	ManagedFunction	+ <sub>inherited</sub>	M <sub>inherited</sub>	M <sub>inherited</sub>	M <sub>inherited</sub>
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 3—Attributes**

Attribute name	Defined in	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
SsFunctionId	--	+	M	M	--
objectClass	Top	+ <sub>inherited</sub>	M <sub>inherited</sub>	M <sub>inherited</sub>	-- <sub>inherited</sub>
objectInstance	Top	+ <sub>inherited</sub>	M <sub>inherited</sub>	M <sub>inherited</sub>	-- <sub>inherited</sub>
userLabel	ManagedFunction	+ <sub>inherited</sub>	M <sub>inherited</sub>	M <sub>inherited</sub>	M <sub>inherited</sub>
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 4—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 5—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 5—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.

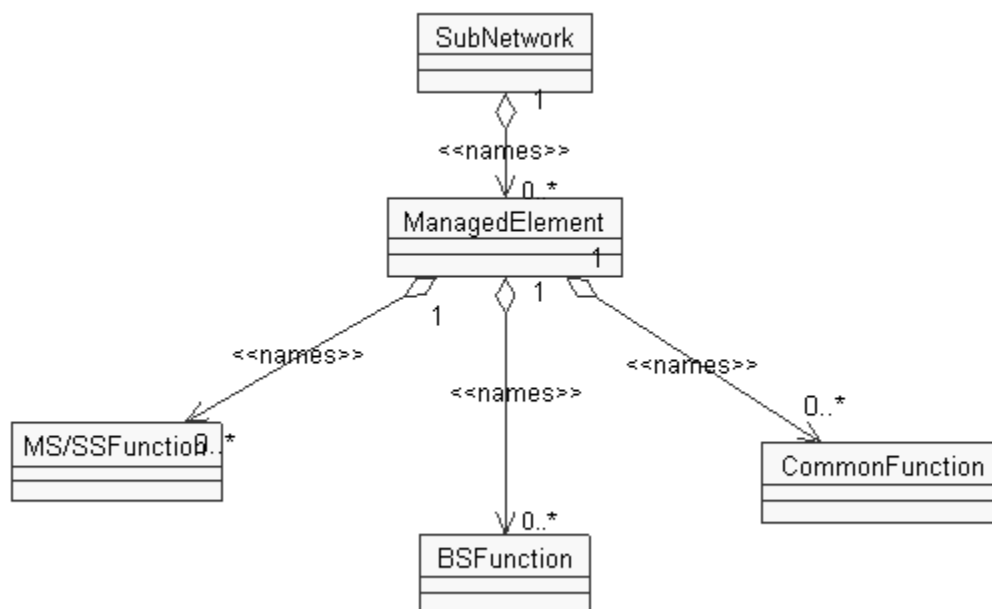
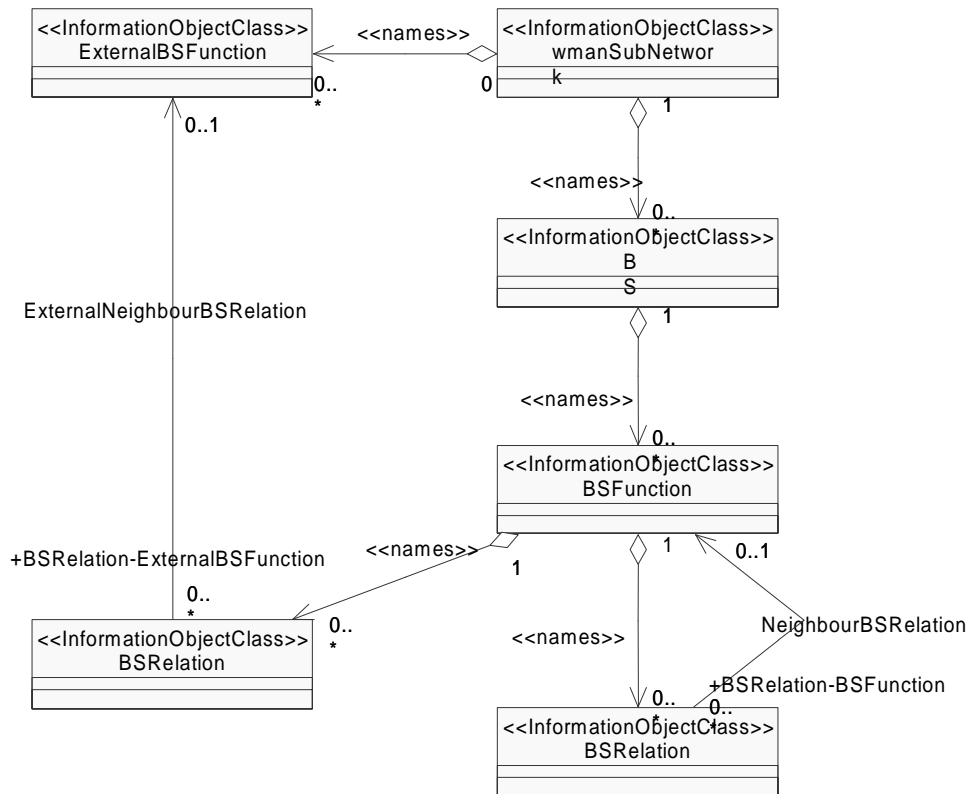
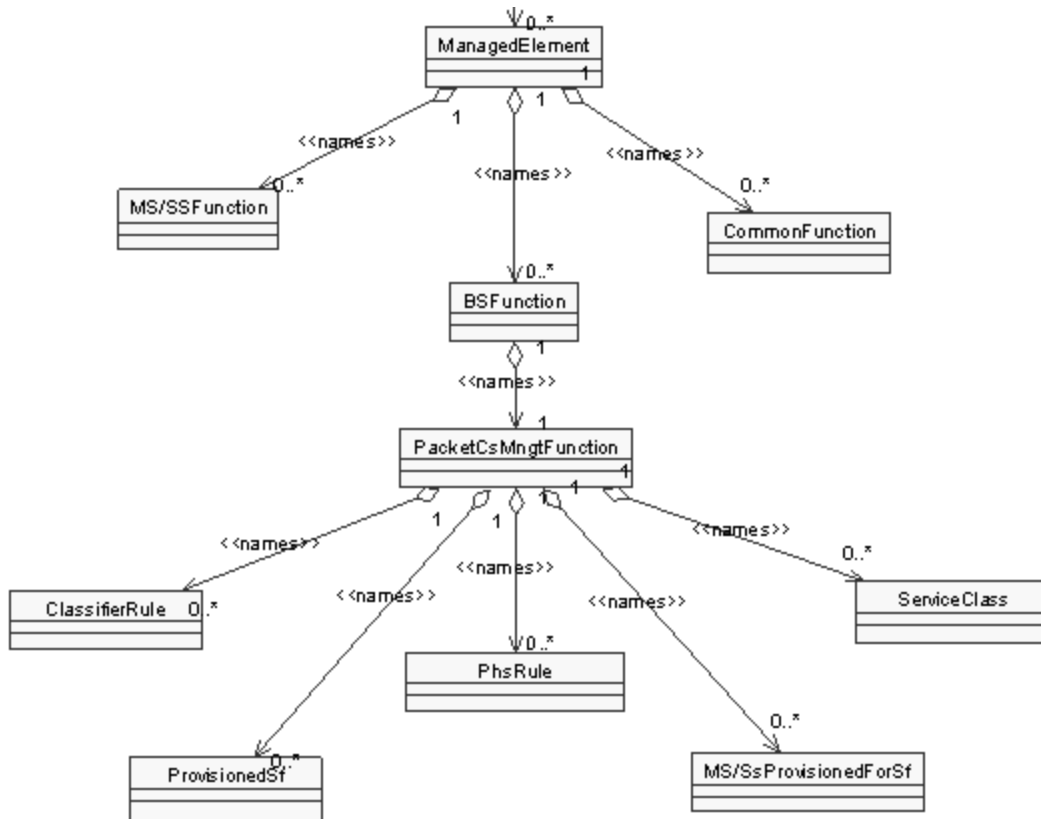


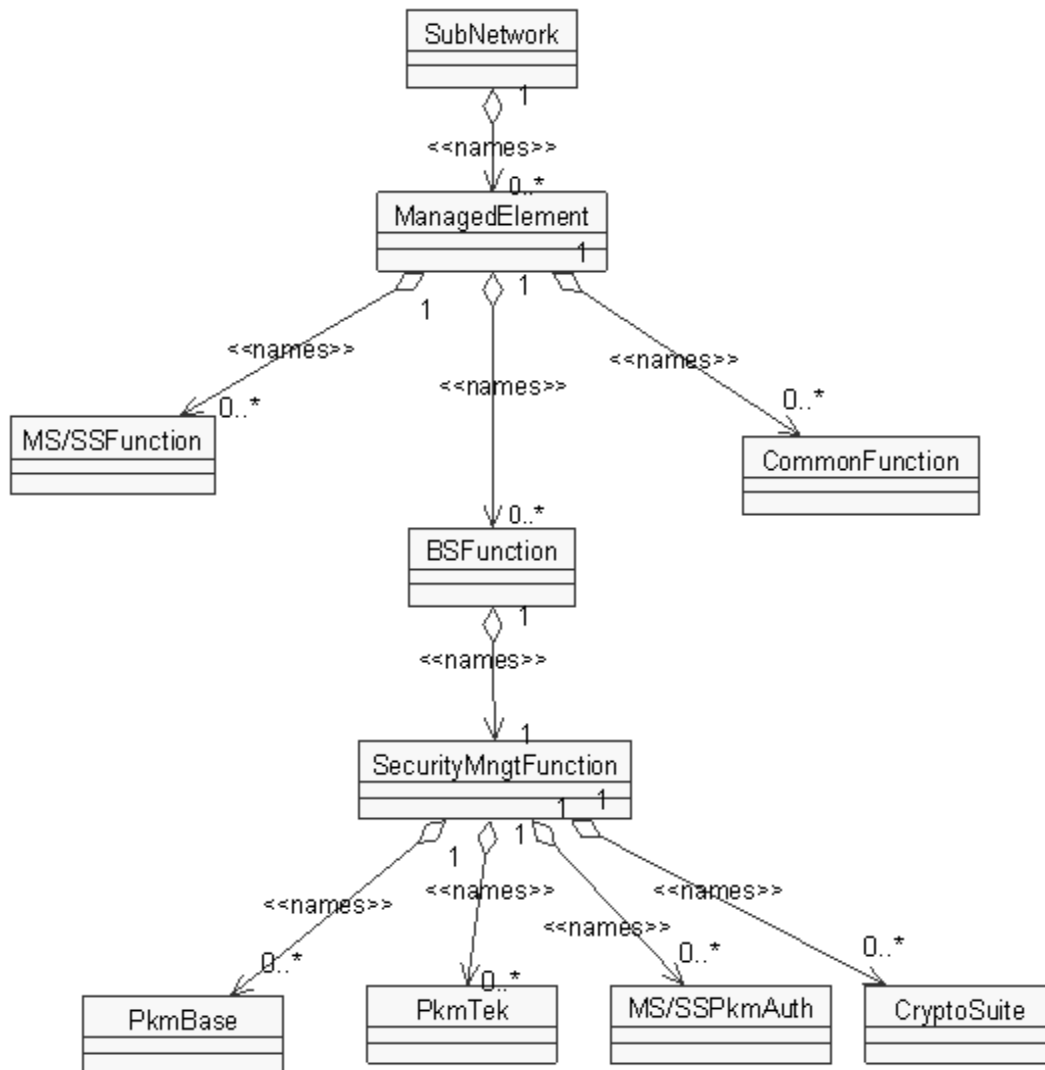
Figure 6—General View



**Figure 7—Segment view Neighbouring BS NRM Containment/Naming Diagram**



**Figure 8—Segment view PacketCs NRM Containment/Naming Diagram**



**Figure 9—Segment view SecurityMng NRM Containment/Naming Diagram**

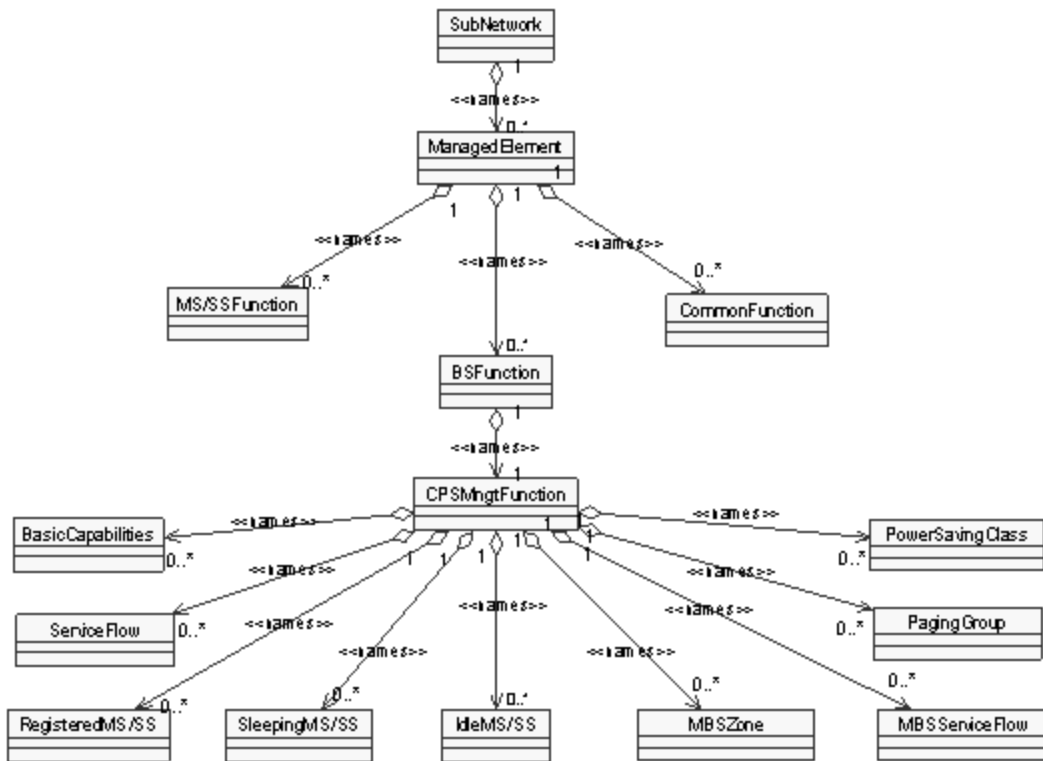
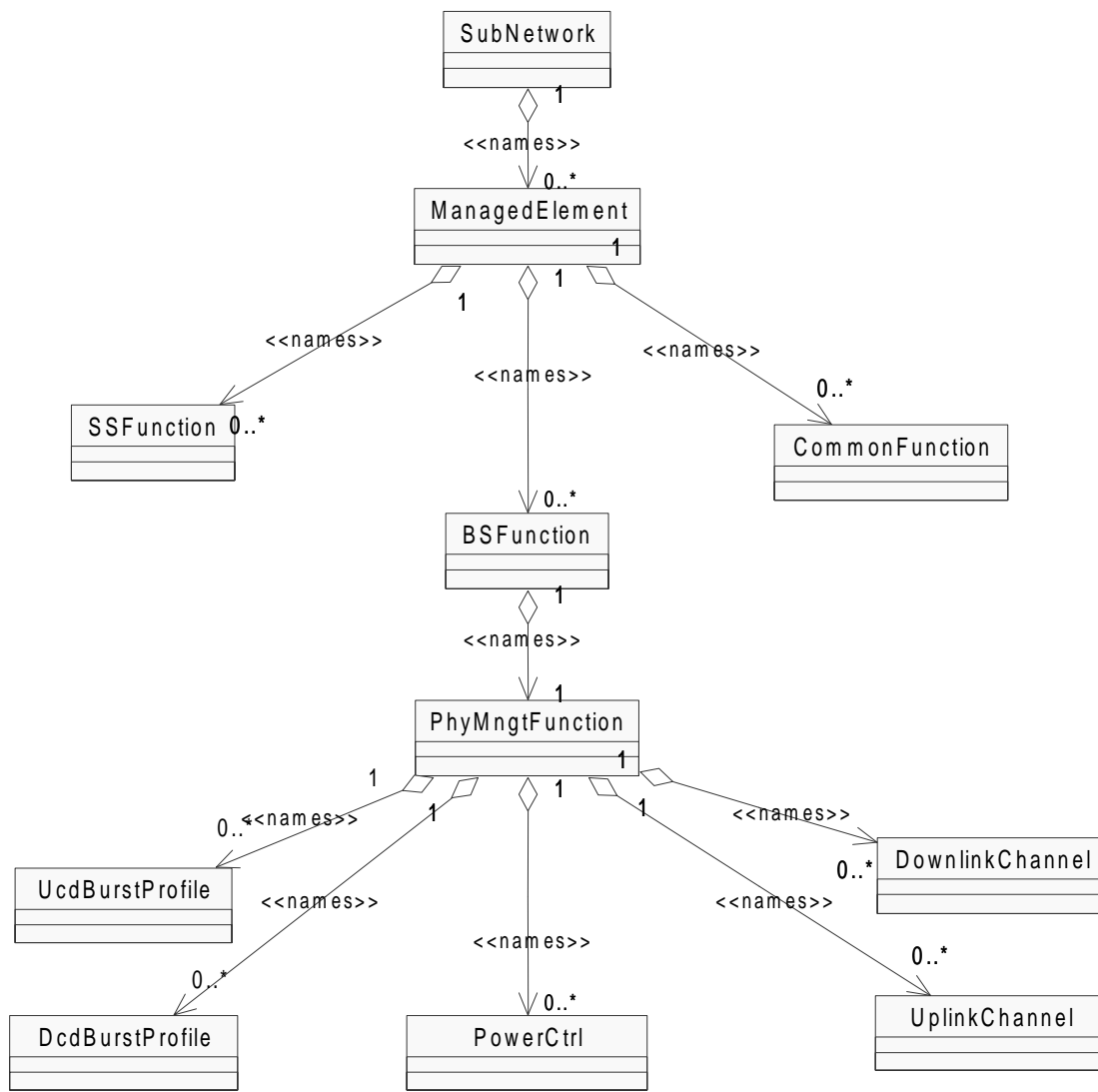


Figure 10—Segment view CPSMngMng NRM Containment/Naming Diagram



**Figure 11—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.

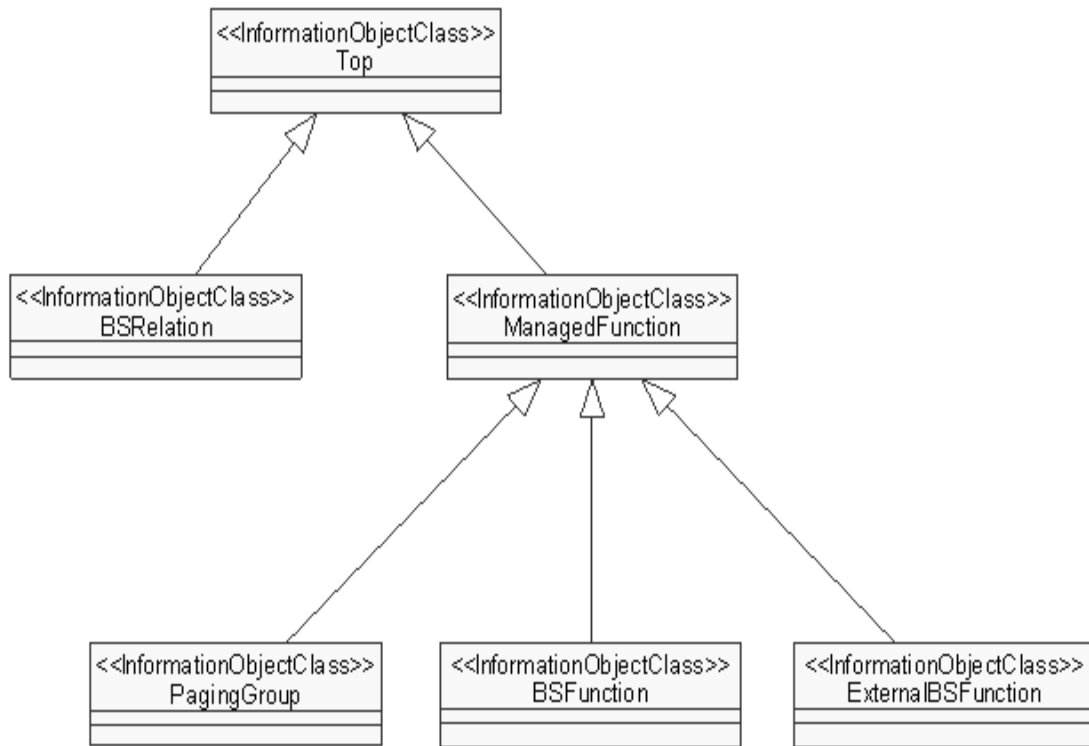


Figure 12—Inheritance Diagram

### 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 6—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



**Table 6—Attributes of BSFunction**

Attribute name	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
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 7—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
HOProcessOptimization	+	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

**Table 7—Attributes of ExternalBSFunction**

Attribute name	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
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 8—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 9—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
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

#### 15.1.2.4 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 10—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.adjacentBS 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 11—Information attributes definition**

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

**Table 11—Information attributes definition**

Attribute Name	Definition	Legal Values
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))
MSHOConnectionProcessTime	Time in ms the MS needs to process information on connections provided in RNGRSP or REG-RSP message during HO	
MSHOTeKProcessTime	Time in ms the MS needs to completely process TEK information during HO	
ULPermutationBase	Uplink subcarrier allocation	
DLPermutationBase	Downlink subcarrier allocation	
PreambleIndex	Downlink synchronization by MS	
SegmentNumber	An unique segment identifier	
ExternalBSId	External BS Identifier	
FAIndex	Frequency Assignment Index	
BSEIRP	Neighbour BS EIRP	
HOProcessOptimization	Identifies re-entry process management messages that may be omitted during the current HO attempt due to the availability of MS service and operational context information, and the MS service and operational status post-HO completion.	
SchedulingServiceSupported	Indicate neighbouring BS scheduling service type.	Type: Enumerated value Range: ( Non-real-time Polling Service(0), Real-time Polling Service(0), Extended real-time Polling Service(0), Unsolicited Grant Service(0), Best Effort(3))
Bandwidth	Indicate neighbouring BS bandwidth.	
FFTSize	Indicate neighbouring BS FFT size	
CyclePrefix	indicate neighbouring BS Cycle Prefix	
FramDurationCode	Indicate neighbouring BS Frame duration code	
ULPermutationBase	Indicate neighbouring BS uplink permutation base.	
DLPermutationBase	Indicate neighbouring BS uplink permutation base.	
SegmentNumber	Indicate neighbouring BS segment number.	
PreambleIndex	Indicate neighbouring BS preamble index.	

**Table 11—Information attributes definition**

Attribute Name	Definition	Legal Values
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 retransmission(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.	

1  
2  
3  
4 **15.2 NRM IRP SNMP Solution Set**  
5  
6  
7

8  
9  
10 **15.3 NRM IRP CORBA Solution Set**  
11  
12  
13

14  
15 **15.4 NRM IRP XML Solution Set**  
16  
17  
18

19  
20 **15.5 Interface IRP Considerations**  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65

**Annex E. ASN.1 Definition of wmanIfMib in 802.16f**

```

1  WMAN-IF-MIB DEFINITIONS ::= BEGIN
2
3
4
5  IMPORTS
6
7      MODULE-IDENTITY,
8      OBJECT-TYPE,
9      NOTIFICATION-TYPE,
10     Unsigned32, Integer32, Counter32,
11     Counter64, transmission
12     FROM SNMPv2-SMI
13     SnmpAdminString
14     FROM SNMP-FRAMEWORK-MIB
15     TEXTUAL-CONVENTION,
16     MacAddress, RowStatus, TruthValue,
17     TimeStamp, DateAndTime
18     FROM SNMPv2-TC
19     InetAddressType, InetAddress
20     FROM INET-ADDRESS-MIB
21     OBJECT-GROUP,
22     MODULE-COMPLIANCE,
23     NOTIFICATION-GROUP
24     FROM SNMPv2-CONF
25     ifIndex
26     FROM IF-MIB;
27
28
29
30
31
32  wmanIfMib MODULE-IDENTITY
33      LAST-UPDATED      "200508020000Z" -- August 02, 2005
34      ORGANIZATION      "IEEE 802.16"
35      CONTACT-INFO
36          "WG E-mail: stds-802-16@ieee.org
37           WG Chair: Roger B. Marks
38           Postal: (U.S.) National Institute
39                 of Standards and Technology
40           E-mail: r.b.marks@ieee.org
41
42           TGf Chair: Phillip Barber
43           Postal: Huawei Technologies Co., Ltd
44           E-mail: pbarber@futurewei.com
45
46           Editor: Joey Chou
47           Postal: Intel Corporation
48                 5000 W. Chandler Blvd,
49                 Chandler, AZ 85227, USA
50           E-mail: joey.chou@intel.com"
51
52  DESCRIPTION
53      "This material is from IEEE Std 802.16f
54      Copyright (c) 2005 IEEE.
55      This MIB Module defines managed objects for
56      IEEE 802.16-2004 based Subscriber Station
57      and Base Station."
58
59  REVISION      "200508020000Z"
60
61  DESCRIPTION
62      "The first approved version of WMAN-IF-MIB module."
63
64
65

```



```

1      ::= { transmission 184 }
2
3      wmanIfMibObjects OBJECT IDENTIFIER ::= { wmanIfMib 1 }
4      wmanIfBsObjects OBJECT IDENTIFIER ::= { wmanIfMibObjects 1 }
5      wmanIfSsObjects OBJECT IDENTIFIER ::= { wmanIfMibObjects 2 }
6      wmanIfCommonObjects OBJECT IDENTIFIER ::= { wmanIfMibObjects 3 }
7
8
9      -- Textual Conventions
10     WmanIfSfsSchedulingType ::= TEXTUAL-CONVENTION
11         STATUS          current
12         DESCRIPTION
13             "The scheduling service provided by a SC for an
14             upstream service flow. If the parameter is omitted
15             from an upstream QOS Parameter Set, this object takes
16             the value of bestEffort (2). This parameter must be
17             reported as undefined (1) for downstream QOS Parameter
18             Sets."
19         SYNTAX          INTEGER {undefined(1),
20                             bestEffort(2),
21                             nonRealTimePollingService(3),
22                             realTimePollingService(4),
23                             reserved(5),
24                             unsolicitedGrantService(6)}
25
26
27     WmanIfPhsRuleVerify ::= TEXTUAL-CONVENTION
28         STATUS          current
29         DESCRIPTION
30             "The value of this field indicates to the sending entity
31             whether or not the packet header contents are to be
32             verified prior to performing suppression. If PHSV is
33             enabled, the sender shall compare the bytes in the packet
34             header with the bytes in the PHSF that are to be
35             suppressed as indicated by the PHSM."
36         REFERENCE
37             "Subclause 11.13.19.3.7.5 in IEEE Std 802.16-2004"
38         SYNTAX          INTEGER {phsVerifyEnable(0),
39                                 phsVerifyDisable(1)}
40
41
42     WmanIfClassifierBitMap ::= TEXTUAL-CONVENTION
43         STATUS          current
44         DESCRIPTION
45             "A bit of of this object is set to 1 if the parameter
46             indicated by the comment was present in the classifier
47             encoding, and 0 otherwise.
48             Note: that BITS are encoded most significant bit first,
49             so that if e.g. bits 6 and 7 are set, this object is
50             encoded as the octet string '030000'H."
51         REFERENCE
52             "Subclause 11.13.19.3.4 in IEEE Std 802.16-2004"
53         SYNTAX          BITS {priority(0),
54                               ipTos(1),
55                               ipProtocol(2),
56                               ipMaskedSrcAddr(3),
57                               ipMaskedDestAddr(4),
58

```

```

1          srcPort(5),
2          destPort(6),
3          destMacAddr(7),
4          srcMacAddr(8),
5          ethernetProtocol(9),
6          userPriority(10),
7          vlanId(11),
8          ipv6FlowLabel(12)}
9
10
11
12 WmanIfSfState ::= TEXTUAL-CONVENTION
13     STATUS      current
14     DESCRIPTION
15         "WmanIfSfState defines the state of a service flow."
16     SYNTAX      INTEGER {authorized(1),
17                     admitted(2),
18                     active(3)}
19
20
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
30 WmanIfCsSpecification ::= TEXTUAL-CONVENTION
31     STATUS      current
32     DESCRIPTION
33         "WmanIfCsSpecification defines the types of convergence
34         sublayer."
35     SYNTAX      INTEGER {noCs(0),
36                     packetIPv4(1),
37                     packetIPv6(2),
38                     packet802dot3Ethernet(3),
39                     packet802dot1QVlan(4),
40                     packetIPv4Over802dot3(5),
41                     packetIPv6Over802dot3(6),
42                     packetIPv4Over802dot1Q(7),
43                     packetIPv6Over802dot1Q(8),
44                     atm(9)}
45
46
47
48
49 WmanIfMacVersion ::= TEXTUAL-CONVENTION
50     STATUS      current
51     DESCRIPTION
52         "Version number of IEEE 802.16."
53     SYNTAX      INTEGER {ieee802Dot16Of2001(1),
54                     ieee802Dot16cOf2002(2),
55                     ieee802Dot16aOf2003(3),
56                     ieee802Dot16Of2004(4)}
57
58
59
60 WmanIfCidType ::= TEXTUAL-CONVENTION
61     STATUS      current
62     DESCRIPTION
63         "Type of CID."
64     SYNTAX      INTEGER (0 .. 65535)
65

```

```

1
2 WmanIfDataEncryptAlgId ::= TEXTUAL-CONVENTION
3     STATUS          current
4     DESCRIPTION
5         "Data encryption algorithm identifiers."
6     REFERENCE
7         "Table 375 in IEEE Std 802.16-2004"
8     SYNTAX          INTEGER {none(0),
9                     des56BitCbcMode(1),
10                    aesCcmMode(2) }
11
12
13
14 WmanIfDataAuthAlgId ::= TEXTUAL-CONVENTION
15     STATUS          current
16     DESCRIPTION
17         "Data authentication algorithm identifiers."
18     REFERENCE
19         "Table 376 in IEEE Std 802.16-2004"
20     SYNTAX          INTEGER {noDataAuthentication(0),
21                    reserved(1) }
22
23
24
25 WmanIfTekEncryptAlgId ::= TEXTUAL-CONVENTION
26     STATUS          current
27     DESCRIPTION
28         "TEK encryption algorithm identifiers."
29     REFERENCE
30         "Table 377 in IEEE Std 802.16-2004"
31     SYNTAX          INTEGER {tripleDes128BitKey(1),
32                    rsa1024BitKey(2),
33                    aes128BitKey(3) }
34
35
36
37 WmanIfChannelNumber ::= TEXTUAL-CONVENTION
38     STATUS          current
39     DESCRIPTION
40         "Physical channel number"
41     SYNTAX          INTEGER (0 .. 199)
42
43
44 WmanIfOfdmFecCodeType ::= TEXTUAL-CONVENTION
45     STATUS          current
46     DESCRIPTION
47         "FEC code type and modulation type"
48     REFERENCE
49         "Table 356 and Table 362 in IEEE Std 802.16-2004"
50     SYNTAX          INTEGER {bpskCc1Over2(0),
51                    qpskRsCcCc1Over2(1),
52                    qpskRsCcCc3Over4(2),
53                    sixteenQamRsCcCc1Over2(3),
54                    sixteenQamRsCcCc3Over4(4),
55                    sixtyFourQamRsCcCc2Over3(5),
56                    sixtyFourQamRsCcCc3Over4(6),
57                    qpskBtc1Over2(7),
58                    qpskBtc3Over4(8),
59                    sixteenQamBtc3Over4(9),
60                    sixteenQamBtc4Over5(10),
61                    sixtyFourQamBtc2Over3(11),
62
63
64
65

```

```

1          sixtyFourQamBtc5Over6(12),
2          qpskCtc1Over2(13),
3          qpskCtc2Over3(14),
4          qpskCtc3Over4(15),
5          sixteenQamCtc1Over2(16),
6          sixteenQamCtc3Over4(17),
7          sixtyFourQamCtc2Over3(18),
8          sixtyFourQamCtc3Over4(19)}
9
10
11
12 WmanIfOfdmaFecCodeType ::= TEXTUAL-CONVENTION
13     STATUS          current
14     DESCRIPTION
15         "FEC code type and modulation type"
16     REFERENCE
17         "Table 356 and Table 362 in IEEE Std 802.16-2004"
18     SYNTAX          INTEGER {qpskCc1Over2(0),
19                             qpskCc3Over4(1),
20                             sixteenQamCc1Over2(2),
21                             sixteenQamCc3Over4(3),
22                             sixtyFourQamCc2Over3(4),
23                             sixtyFourQamCc3Over4(5),
24                             qpskBtc1Over2(6),
25                             qpskBtc2Over3(7),
26                             sixteenQamBtc3Over5(8),
27                             sixteenQamBtc4Over5(9),
28                             sixtyFourQamBtc5Over8(10),
29                             sixtyFourQamBtc4Over5(11),
30                             qpskCtc1Over2(12),
31                             qpskCtc2Over3(13),
32                             qpskCtc3Over4(14),
33                             sixteenQamCtc1Over2(15),
34                             sixteenQamCtc3Over4(16),
35                             sixtyFourQamCtc2Over3(17),
36                             sixtyFourQamCtc3Over4(18),
37                             sixtyFourQamCtc5Over6(19),
38                             qpskZtCc1Over2(20),
39                             qpskZtCc3Over4(21),
40                             sixteenQamZtCc1Over2(22),
41                             sixteenQamZtCc3Over4(23),
42                             sixtyFourQamZtCc2Over3(24),
43                             sixtyFourQamZtCc3Over4(25)}
44
45
46
47
48
49
50
51 -- Textual convention for capabilities encodings
52 WmanIfNumOfUplinkCid ::= TEXTUAL-CONVENTION
53     STATUS          current
54     DESCRIPTION
55         "The object of this type shows the number of Uplink CIDs
56         the SS can support."
57     REFERENCE
58         "Subclause 11.7.4 in IEEE Std 802.16-2004"
59     SYNTAX          INTEGER (2..65535)
60
61
62
63 WmanIfArgSupportType ::= TEXTUAL-CONVENTION
64     STATUS          current
65

```

```

1      DESCRIPTION
2          "The object of this type indicates whether the SS support
3          ARQ."
4
5      REFERENCE
6          "Subclause 11.7.8.1 in IEEE Std 802.16-2004"
7      SYNTAX      INTEGER {arqNotSupported(0),
8                      arqSupported(1)}
9
10
11  WmanIfMaxDsxFloType ::= TEXTUAL-CONVENTION
12      STATUS      current
13      DESCRIPTION
14          "The object of this type specifies the maximum number of
15          concurrent DSA, DSC, or DSD transactions that may be
16          outstanding."
17      REFERENCE
18          "Subclause 11.7.8.2 in IEEE Std 802.16-2004"
19      SYNTAX      INTEGER (0..255)
20
21
22
23  WmanIfMacCrcSupport ::= TEXTUAL-CONVENTION
24      STATUS      current
25      DESCRIPTION
26          "The object of this type indicates whether or not the SS
27          supports MAC level CRC."
28      REFERENCE
29          "Subclause 11.7.8.3 in IEEE Std 802.16-2004"
30      SYNTAX      INTEGER {noMacCrcSupport(0),
31                      macCrcSupport(1)}
32
33
34
35  WmanIfMaxMcaFloType ::= TEXTUAL-CONVENTION
36      STATUS      current
37      DESCRIPTION
38          "The object of this type specifies the maximum number of
39          concurrent MCA transactions that may be outstanding."
40      REFERENCE
41          "Subclause 11.7.8.4 in IEEE Std 802.16-2004"
42      SYNTAX      INTEGER (0..255)
43
44
45
46  WmanIfMaxMcpGroupCid ::= TEXTUAL-CONVENTION
47      STATUS      current
48      DESCRIPTION
49          "The object of this type indicates the maximum number of
50          simultaneous Multicast Polling Groups the SS is
51          capable of belonging to."
52      REFERENCE
53          "Subclause 11.7.8.5 in IEEE Std 802.16-2004"
54      SYNTAX      INTEGER (0..255)
55
56
57
58  WmanIfMaxPkmFloType ::= TEXTUAL-CONVENTION
59      STATUS      current
60      DESCRIPTION
61          "The object of this type specifies the maximum number of
62          concurrent PKM transactions that may be outstanding."
63      REFERENCE
64          "Subclause 11.7.8.6 in IEEE Std 802.16-2004"
65

```

```

1          SYNTAX          INTEGER (0..255)
2
3
4  WmanIfAuthPolicyType ::= TEXTUAL-CONVENTION
5      STATUS              current
6      DESCRIPTION
7          "The object of this type specifies authorization policy
8           that both SS and BS need to negotiate and synchronize.
9           A bit value of 0 = not supported, 1 = supported. If this
10          field is omitted, then both SS and BS shall use the IEEE
11          802.16 security, constituting X.509 digital certificates
12          and the RSA public key encryption algorithm, as
13          authorization policy."
14
15      REFERENCE
16          "Subclause 11.7.8.7 in IEEE Std 802.16-2004"
17
18      SYNTAX              BITS {ieee802Dot16PrivacySupported(0),
19                               reserved1(1),
20                               reserved2(2),
21                               reserved3(3),
22                               reserved4(4),
23                               reserved5(5),
24                               reserved6(6),
25                               reserved7(7)}
26
27
28
29  WmanIfMaxNumOfSaType ::= TEXTUAL-CONVENTION
30      STATUS              current
31      DESCRIPTION
32          "This field specifies maximum number of supported
33          security association of the SS."
34
35      REFERENCE
36          "Subclause 11.7.8.8 in IEEE Std 802.16-2004"
37
38      SYNTAX              INTEGER (0..255)
39
40
41  WmanIfIpVersionType ::= TEXTUAL-CONVENTION
42      STATUS              current
43      DESCRIPTION
44          "The object of this type indicates the version of IP used
45          on the Secondary Management Connection. The value should
46          be undefined if the 2nd management CID doesn't exist."
47
48      REFERENCE
49          "Subclause 11.7.4 in IEEE Std 802.16-2004"
50
51      SYNTAX              INTEGER {undefined(0),
52                                   ipv4(1),
53                                   ipv6(2)}
54
55
56  WmanIfMacCsBitMap ::= TEXTUAL-CONVENTION
57      STATUS              current
58      DESCRIPTION
59          "The object of this type indicates the set of MAC
60          convergence sublayer support. When a bit is set, it
61          indicates the corresponding CS feature is supported."
62
63      REFERENCE
64          "Subclause 11.7.7.1 in IEEE Std 802.16-2004"
65
66      SYNTAX              BITS {atm(0),
67                               packetIpv4(1),

```

```

1          packetIpv6(2),
2          packet802Dot3(3),
3          packet802Dot1Q(4),
4          packetIpv4Over802Dot3(5),
5          packetIpv6Over802Dot3(6),
6          packetIpv4Over802Dot1Q(7),
7          packetIpv6Over802Dot1Q(8)}
8
9
10
11 WmanIfMaxClassifiers ::= TEXTUAL-CONVENTION
12     STATUS          current
13     DESCRIPTION
14         "The object of this type indicates the maximum number of
15         admitted Classifiers that the SS is allowed to have."
16     REFERENCE
17         "Subclause 11.7.7.2 in IEEE Std 802.16-2004"
18     SYNTAX          INTEGER (0..65535)
19
20
21 WmanIfPhsSupportType ::= TEXTUAL-CONVENTION
22     STATUS          current
23     DESCRIPTION
24         "The object of this type indicates the level
25         of PHS support."
26     REFERENCE
27         "Subclause 11.7.7.3 in IEEE Std 802.16-2004"
28     SYNTAX          INTEGER {noPhsSupport(0),
29                        atmPhsSupport(1),
30                        packetPhsSupport(2)}
31
32
33
34 WmanIfBwAllocSupport ::= TEXTUAL-CONVENTION
35     STATUS          current
36     DESCRIPTION
37         "This field indicates properties of the SS that the BS
38         needs to know for bandwidth allocation purposes. When
39         a bit is set, it indicates the corresponding feature
40         is supported. All unspecified and reserved bits should
41         be set to zero."
42     REFERENCE
43         "Subclause 11.8.1 in IEEE Std 802.16-2004"
44     SYNTAX          BITS {reserved(0),
45                        halfDuplexFdd(1),
46                        fullDuplexFdd(2)}
47
48
49
50
51 WmanIfPduConstruction ::= TEXTUAL-CONVENTION
52     STATUS          current
53     DESCRIPTION
54         "Specifies capabilities for construction and transmission
55         of MAC PDUs. When piggybackedRequests bit is set, it
56         indicates that the piggybacked requests are supported. The
57         fsnValuesSize bit is coded as follows:
58         0 - only 3-bit FSN values are supported
59         1 - only 11-bit FSN values are supported
60         All unspecified and reserved bits should be set to zero."
61     REFERENCE
62         "Subclause 11.8.2 in IEEE Std 802.16-2004"
63
64
65

```

```

1          SYNTAX      BITS {piggybackedRequests(0),
2                        fsnValuesSize(1)}
3
4
5  WmanIfSsTransitionGap ::= TEXTUAL-CONVENTION
6      STATUS          current
7      DESCRIPTION
8          "This field indicates the transition speed SSTTG and SSRTG
9          for TDD and H-FDD SSs. Allowed values are:
10             OFDM mode: TDD and H-FDD 0..100
11             Other modes: TDD: 0..50; H-FDD: 0..100"
12
13      REFERENCE
14          "Subclause 11.8.3.1 in IEEE Std 802.16-2004"
15      SYNTAX          INTEGER (0..100)
16
17
18  WmanIfMaxTxPowerType ::= TEXTUAL-CONVENTION
19      STATUS          current
20      DESCRIPTION
21          "This type is used to define maximum available power for
22          BPSK, QPSK, 16-QAM and 64-QAM constellations. The maximum
23          power parameters are reported in dBm and quantized in 0.5
24          dBm steps ranging from -64 dBm (encoded 0x00) to 63.5 dBm
25          (encoded 0xFF). Values outside this range shall be
26          assigned the closest extreme. SSs that do not support
27          QAM64 shall report the value of 0x00 in the maximum QAM64
28          power field."
29
30      REFERENCE
31          "Subclause 11.8.3.2 in IEEE Std 802.16-2004"
32      SYNTAX          INTEGER (0..255)
33
34
35  WmanIfOfdmFftSizes ::= TEXTUAL-CONVENTION
36      STATUS          current
37      DESCRIPTION
38          "This field indicates the FFT sizes supported by the SS.
39          For each FFT size, a bit value of 0 indicates
40          'not supported' while 1 indicates 'supported'."
41
42      REFERENCE
43          "Subclause 11.8.3.6.1 in IEEE Std 802.16-2004"
44      SYNTAX          BITS {fft256(0),
45                          fft2048(1)}
46
47
48  WmanIfOfdmSsDeModType ::= TEXTUAL-CONVENTION
49      STATUS          current
50      DESCRIPTION
51          "This field indicates the different demodulator options
52          supported by a WirelessMAN-OFDM PHY SS for downlink. This
53          field is not used for other PHY specifications. A bit
54          value of 0 indicates 'not supported' while 1 indicates
55          'supported'."
56
57      REFERENCE
58          "Subclause 11.8.3.6.2 in IEEE Std 802.16-2004"
59      SYNTAX          BITS {qam64(0),
60                          btc(1),
61                          ctc(2),
62                          stc(3),
63
64
65

```



```

1          aac(4) }
2
3 WmanIfOfdmSsModType ::= TEXTUAL-CONVENTION
4     STATUS          current
5     DESCRIPTION
6         "This field indicates the different modulator options
7         supported by a WirelessMAN-OFDM PHY SS for uplink. This
8         field is not used for other PHY specifications. A bit
9         value of 0 indicates 'not supported' while 1 indicates
10        'supported'."
11    REFERENCE
12        "Subclause 11.8.3.6.3 in IEEE Std 802.16-2004"
13    SYNTAX          BITS {qam64(0),
14                        btc(1),
15                        ctc(2),
16                        subchannellization(3),
17                        focusedCtBwReq(4)}
18
19 WmanIfOfdmFocusedCt ::= TEXTUAL-CONVENTION
20     STATUS          current
21     DESCRIPTION
22         "This field indicates whether the SS supports Focused
23         Contention (see 8.3.7.3.3). A bit value of 0 indicates
24         'not supported' while 1 indicates 'supported'."
25    REFERENCE
26        "Subclause 11.8.3.6.4 in IEEE Std 802.16-2004"
27    SYNTAX          BITS {focusedCtSupport(0)}
28
29 WmanIfOfdmTcSublayer ::= TEXTUAL-CONVENTION
30     STATUS          current
31     DESCRIPTION
32         "This field indicates whether or not the SS supports the
33         TC sublayer (see 8.3.4). A bit value of 0 indicates
34         'not supported' while 1 indicates 'supported'."
35    REFERENCE
36        "Subclause 11.8.3.6.5 in IEEE Std 802.16-2004"
37    SYNTAX          BITS {tcSublayerSupport(0)}
38
39 WmanIfBsIdType ::= TEXTUAL-CONVENTION
40     STATUS          current
41     DESCRIPTION
42         "Defines the encoding of BSID. The BSID is a 6 byte number
43         and follows the encoding rules of MacAddress textual
44         convention, i.e. as if it were transmitted
45         least-significant bit first. The value should be displayed
46         with 2 parts clearly separated by a colon e.g:
47         001DFF:00003A. The most significant part is representing
48         the Operator ID. "
49    SYNTAX          OCTET STRING (SIZE(6))
50
51 WmanIfIpv6FlowLabel ::= TEXTUAL-CONVENTION
52     STATUS          current
53     DESCRIPTION
54         "The value of this field specifies the matching values for
55

```

```

1         the IPv6 Flow label field. As the flow label field has a
2         length of 20 bits, the first 4 bits of the most
3         significant byte shall be set to 0x0 and disregarded."
4     SYNTAX      OCTET STRING (SIZE(3))
5
6
7     --
8     -- BS object group - containing tables and objects to be implemented in
9     -- the Base station
10    --
11    --
12    -- wmanIfBsPacketCs contain the Base Station Packet Convergence
13    -- Sublayer objects
14    --
15    wmanIfBsPacketCs OBJECT IDENTIFIER ::= { wmanIfBsObjects 1 }
16
17    wmanIfBsProvisionedSfTable OBJECT-TYPE
18        SYNTAX      SEQUENCE OF WmanIfBsProvisionedSfEntry
19        MAX-ACCESS   not-accessible
20        STATUS       current
21        DESCRIPTION
22            "This table contains service flow profiles provisioned by
23            NMS. The service flow should be created with SS(s)
24            following instruction given by wmanIfBsSfState object.
25            1. The QoS parameters of the service flow are provisioned
26            in wmanIfBsServiceClassTable and referenced by
27            wmanIfBsServiceClassIndex.
28            2. The classifier rules of the service flow are provisioned
29            in wmanIfBsClassifierRuleTable, where they refer to SF
30            via wmanIfBsSfId.
31
32            The MAC addresses of SSs the service flow is created with
33            are provisioned in wmanIfBsSsProvisionedForSfTable, where
34            they refer to SF via wmanIfBsSfId."
35
36        REFERENCE
37            "Subclause 6.3.13 and 6.3.14 in IEEE Std 802.16-2004"
38        ::= { wmanIfBsPacketCs 1 }
39
40    wmanIfBsProvisionedSfEntry OBJECT-TYPE
41        SYNTAX      WmanIfBsProvisionedSfEntry
42        MAX-ACCESS   not-accessible
43        STATUS       current
44        DESCRIPTION
45            "This table provides one row for each service flow
46            provisioned by NMS. The table is indexed by ifIndex and
47            wmanIfBsSfId. ifIndex is associated with the BS sector."
48        INDEX { ifIndex, wmanIfBsSfId }
49        ::= { wmanIfBsProvisionedSfTable 1 }
50
51    WmanIfBsProvisionedSfEntry ::= SEQUENCE {
52        wmanIfBsSfId                Unsigned32,
53        wmanIfBsSfDirection          INTEGER,
54        wmanIfBsServiceClassIndex    INTEGER,
55        wmanIfBsSfState              WmanIfSfState,
56        wmanIfBsSfProvisionedTime    TimeStamp,
57        wmanIfBsSfCsSpecification    WmanIfCsSpecification,

```

```

1          wmanIfBsProvisionedSfRowStatus          RowStatus}
2
3
4  wmanIfBsSfId OBJECT-TYPE
5      SYNTAX      Unsigned32 (1 .. 4294967295)
6      MAX-ACCESS  not-accessible
7      STATUS      current
8      DESCRIPTION
9          "A 32 bit quantity that uniquely identifies a service flow
10         to both the subscriber station and base station (BS)."
```

::= { wmanIfBsProvisionedSfEntry 1 }

```

13
14  wmanIfBsSfDirection OBJECT-TYPE
15      SYNTAX      INTEGER {downstream(1),
16                      upstream(2)}
17      MAX-ACCESS  read-create
18      STATUS      current
19      DESCRIPTION
20          "An attribute indicating the service flow is downstream or
21         upstream."
```

::= { wmanIfBsProvisionedSfEntry 2 }

```

25
26  wmanIfBsServiceClassIndex OBJECT-TYPE
27      SYNTAX      INTEGER (1..65535)
28      MAX-ACCESS  read-create
29      STATUS      current
30      DESCRIPTION
31          "The index in wmanIfBsServiceClassTable describing the
32         service class or QoS parameters for such service flow.
33         If no associated entry in wmanIfBsServiceClassTable
34         exists, this object returns a value of zero."
```

::= { wmanIfBsProvisionedSfEntry 3 }

```

38
39  wmanIfBsSfState OBJECT-TYPE
40      SYNTAX      WmanIfSfState
41      MAX-ACCESS  read-create
42      STATUS      current
43      DESCRIPTION
44          "wmanIfBsSfState determines the requested state of a service
45         flow.
46         - authorized state: A service flow is provisioned but
47         not resource is reserved yet
48         - admitted state: service flow has resources reserved.
49         - active state: has resources committed by the BS (e.g., is
50         actively sending maps containing unsolicited grants for a
51         UGS-based service flow),"
```

REFERENCE

"Subclause 6.3.14.6, in IEEE Std 802.16-2004"

::= { wmanIfBsProvisionedSfEntry 4 }

```

58
59  wmanIfBsSfProvisionedTime OBJECT-TYPE
60      SYNTAX      TimeStamp
61      MAX-ACCESS  read-create
62      STATUS      current
63      DESCRIPTION
64
65
```

```

1         "Indicates the date and time when the service flow is
2         provisioned."
3         ::= { wmanIfBsProvisionedSfEntry 5 }
4
5
6 wmanIfBsSfCsSpecification OBJECT-TYPE
7     SYNTAX      WmanIfCsSpecification
8     MAX-ACCESS  read-create
9     STATUS      current
10    DESCRIPTION
11        "This parameter specifies the convergence sublayer
12        encapsulation mode."
13    REFERENCE
14        "Subclause 11.13.19.1 in IEEE Std 802.16-2004"
15    ::= { wmanIfBsProvisionedSfEntry 6 }
16
17
18
19 wmanIfBsProvisionedSfRowStatus OBJECT-TYPE
20     SYNTAX      RowStatus
21     MAX-ACCESS  read-create
22     STATUS      current
23     DESCRIPTION
24        "This object is used to create a new row or modify or
25        delete an existing row in this table.
26
27        If the implementator of this MIB has choosen not
28        to implement 'dynamic assignment' of profiles, this
29        object is not useful and should return noSuchName
30        upon SNMP request."
31    ::= { wmanIfBsProvisionedSfEntry 7 }
32
33
34
35
36 wmanIfBsSsProvisionedForSfTable OBJECT-TYPE
37     SYNTAX      SEQUENCE OF WmanIfBsSsProvisionedForSfEntry
38     MAX-ACCESS  not-accessible
39     STATUS      current
40     DESCRIPTION
41        "This table maps the MAC addresses of SSs to the service
42        flows provisioned in wmanIfBsProvisionedSfTable."
43    REFERENCE
44        "Subclause 6.3.14 in IEEE Std 802.16-2004"
45    ::= { wmanIfBsPacketCs 2 }
46
47
48
49 wmanIfBsSsProvisionedForSfEntry OBJECT-TYPE
50     SYNTAX      WmanIfBsSsProvisionedForSfEntry
51     MAX-ACCESS  not-accessible
52     STATUS      current
53     DESCRIPTION
54        "This table is indexed by wmanIfBsSsProvMacAddress and
55        wmanIfBsProvSfId."
56    INDEX { wmanIfBsSsProvMacAddress, wmanIfBsProvSfId }
57    ::= { wmanIfBsSsProvisionedForSfTable 1 }
58
59
60
61 WmanIfBsSsProvisionedForSfEntry ::= SEQUENCE {
62     wmanIfBsSsProvMacAddress      MacAddress,
63     wmanIfBsProvSfId              Unsigned32,
64     wmanIfBsSsProvisionedForSfRowStatus RowStatus}
65

```

```

1
2 wmanIfBsSsProvMacAddress OBJECT-TYPE
3     SYNTAX      MacAddress
4     MAX-ACCESS  not-accessible
5     STATUS      current
6     DESCRIPTION
7         "The MAC address of the SS, the service flow is created
8         with."
9     ::= { wmanIfBsSsProvisionedForSfEntry 1 }
10
11
12
13 wmanIfBsProvSfId OBJECT-TYPE
14     SYNTAX      Unsigned32 (1 .. 4294967295)
15     MAX-ACCESS  not-accessible
16     STATUS      current
17     DESCRIPTION
18         "A 32 bit quantity that uniquely identifies a service flow.
19         The value of this object can be used by BS to index the
20         wmanBsProvisionedSfTable."
21     ::= { wmanIfBsSsProvisionedForSfEntry 2 }
22
23
24
25 wmanIfBsSsProvisionedForSfRowStatus OBJECT-TYPE
26     SYNTAX      RowStatus
27     MAX-ACCESS  read-create
28     STATUS      current
29     DESCRIPTION
30         "This object is used to ensure that the write, create,
31         delete operation to multiple columns is guaranteed to
32         be treated as atomic operation by agent."
33     ::= { wmanIfBsSsProvisionedForSfEntry 3 }
34
35
36
37 wmanIfBsServiceClassTable OBJECT-TYPE
38     SYNTAX      SEQUENCE OF WmanIfBsServiceClassEntry
39     MAX-ACCESS  not-accessible
40     STATUS      current
41     DESCRIPTION
42         "This table is provisioned and is indexed by
43         wmanIfBsQoSProfileIndex. Each entry of the table contains
44         corresponding service flow characteristic attributes
45         (e.g. QoS parameter set). The value of
46         wmanIfBsQoSProfileIndex is obtained from
47         wmanIfBsServiceClassIndex in wmanIfBsProvisionedSfTable"
48     REFERENCE
49         "Subclause 6.3.14.4 in IEEE Std 802.16-2004"
50     ::= { wmanIfBsPacketCs 3 }
51
52
53
54
55 wmanIfBsServiceClassEntry OBJECT-TYPE
56     SYNTAX      WmanIfBsServiceClassEntry
57     MAX-ACCESS  not-accessible
58     STATUS      current
59     DESCRIPTION
60         "This table provides one row for each service class"
61     INDEX { ifIndex, wmanIfBsQoSProfileIndex }
62     ::= { wmanIfBsServiceClassTable 1 }
63
64
65

```

```

1  WmanIfBsServiceClassEntry ::= SEQUENCE {
2      wmanIfBsQoSProfileIndex          INTEGER,
3      wmanIfBsQoSServiceClassName      WmanIfServClassName,
4      wmanIfBsQoSSTrafficPriority       INTEGER,
5      wmanIfBsQoSMaxSustainedRate      Unsigned32,
6      wmanIfBsQoSMaxTrafficBurst       Unsigned32,
7      wmanIfBsQoSMinReservedRate       Unsigned32,
8      wmanIfBsQOSToleratedJitter       Unsigned32,
9      wmanIfBsQoSMaxLatency            Unsigned32,
10     wmanIfBsQoSFixedVsVariableSduInd  INTEGER,
11     wmanIfBsQOSSduSize               Unsigned32,
12     wmanIfBsQoSScSchedulingType       WmanIfSfSchedulingType,
13     wmanIfBsQoSScArqEnable            TruthValue,
14     wmanIfBsQoSScArqWindowSize        INTEGER,
15     wmanIfBsQoSScArqBlockLifetime     INTEGER,
16     wmanIfBsQoSScArqSyncLossTimeout   INTEGER,
17     wmanIfBsQoSScArqDeliverInOrder    TruthValue,
18     wmanIfBsQoSScArqRxPurgeTimeout    INTEGER,
19     wmanIfBsQoSScArqBlockSize         INTEGER,
20     wmanIfBsQoSMinRsvdTolerableRate   Unsigned32,
21     wmanIfBsQoSReqTxPolicy            BITS,
22     wmanIfBsQOSServiceClassRowStatus  RowStatus}
23
24
25
26
27
28
29  wmanIfBsQoSProfileIndex OBJECT-TYPE
30      SYNTAX      INTEGER (1 .. 65535)
31      MAX-ACCESS  not-accessible
32      STATUS      current
33      DESCRIPTION
34          "The index value which uniquely identifies an entry
35           in the wmanIfBsServiceClassTable"
36      ::= { wmanIfBsServiceClassEntry 1 }
37
38
39
40  wmanIfBsQoSServiceClassName OBJECT-TYPE
41      SYNTAX      WmanIfServClassName
42      MAX-ACCESS  read-create
43      STATUS      current
44      DESCRIPTION
45          "Refers to the Service Class Name"
46      REFERENCE
47          "Subclause 11.13.3 in IEEE Std 802.16-2004"
48      ::= { wmanIfBsServiceClassEntry 2 }
49
50
51
52  wmanIfBsQoSSTrafficPriority OBJECT-TYPE
53      SYNTAX      INTEGER (0..7)
54      MAX-ACCESS  read-create
55      STATUS      current
56      DESCRIPTION
57          "The value of this parameter specifies the priority
58           assigned to a service flow. For uplink service flows,
59           the BS should use this parameter when determining
60           precedence in request service and grant generation,
61           and the SS shall preferentially select contention
62           Request opportunities for Priority Request CIDs
63           based on this priority. Higher numbers indicate higher
64
65

```

```

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

```

```

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

```



```

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

REFERENCE

```

8           "Subclause 11.13.11 in IEEE Std 802.16-2004"
9
10          DEFVAL    {bestEffort}
11          ::= { wmanIfBsServiceClassEntry 11 }
```

wmanIfBsQosScArqEnable OBJECT-TYPE

```

14          SYNTAX      TruthValue
15          MAX-ACCESS   read-create
16          STATUS       current
17          DESCRIPTION
18              "True(1) ARQ enabling is requested for the connection."
19          REFERENCE
20              "Subclause 11.13.18 in IEEE Std 802.16-2004"
21          ::= { wmanIfBsServiceClassEntry 12 }
```

wmanIfBsQosScArqWindowSize OBJECT-TYPE

```

26          SYNTAX      INTEGER (1 .. 1024)
27          MAX-ACCESS   read-create
28          STATUS       current
29          DESCRIPTION
30              "Indicates the maximum number of unacknowledged
31              fragments at any time."
32          REFERENCE
33              "Subclause 11.13.18 in IEEE Std 802.16-2004"
34          ::= { wmanIfBsServiceClassEntry 13 }
```

wmanIfBsQosScArqBlockLifetime OBJECT-TYPE

```

39          SYNTAX      INTEGER (0 .. 65535)
40          UNITS        "10 us"
41          MAX-ACCESS   read-create
42          STATUS       current
43          DESCRIPTION
44              "The maximum time interval an ARQ fragment will be
45              managed by the transmitter ARQ machine, once
46              initial transmission of the fragment has occurred.
47              If transmission or retransmission of the fragment
48              is not acknowledged by the receiver before the
49              time limit is reached, the fragment is discarded.
50              A value of 0 means Infinite."
51          REFERENCE
52              "Subclause 11.13.18 in IEEE Std 802.16-2004"
53          DEFVAL      {0}
54          ::= { wmanIfBsServiceClassEntry 14 }
```

wmanIfBsQosScArqSyncLossTimeout OBJECT-TYPE

```

61          SYNTAX      INTEGER (0 .. 65535 )
62          UNITS        "10 us"
63          MAX-ACCESS   read-create
64          STATUS       current
```

```

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      REFERENCE
6          "Subclause 11.13.18 in IEEE Std 802.16-2004"
7      DEFVAL    {0}
8      ::= { wmanIfBsServiceClassEntry 15 }
9
10
11
12  wmanIfBsQosScArqDeliverInOrder OBJECT-TYPE
13      SYNTAX      TruthValue
14      MAX-ACCESS  read-create
15      STATUS      current
16      DESCRIPTION
17          "Indicates whether or not data is to be delivered
18            by the receiving MAC to its client application
19            in the order in which data was handed off to the
20            originating MAC."
21      REFERENCE
22          "Subclause 11.13.18 in IEEE Std 802.16-2004"
23      ::= { wmanIfBsServiceClassEntry 16 }
24
25
26
27  wmanIfBsQosScArqRxPurgeTimeout OBJECT-TYPE
28      SYNTAX      INTEGER (0 .. 65535)
29      UNITS        "10 us"
30      MAX-ACCESS  read-create
31      STATUS      current
32      DESCRIPTION
33          "Indicates the time interval the ARQ window is advanced
34            after a fragment is received. A value of 0 means
35            Infinite."
36      REFERENCE
37          "Subclause 11.13.18 in IEEE Std 802.16-2004"
38      DEFVAL    {0}
39      ::= { wmanIfBsServiceClassEntry 17 }
40
41
42
43
44  wmanIfBsQosScArqBlockSize OBJECT-TYPE
45      SYNTAX      INTEGER (1..2040)
46      UNITS        "byte"
47      MAX-ACCESS  read-create
48      STATUS      current
49      DESCRIPTION
50          "The value of this parameter specifies the size of an
51            ARQ block. This parameter shall be established by
52            negotiation during the connection creation dialog."
53      REFERENCE
54          "Subclause 11.13.18.8 in IEEE Std 802.16-2004"
55      ::= { wmanIfBsServiceClassEntry 18 }
56
57
58
59
60  wmanIfBsQosSCMinRsvdTolerableRate OBJECT-TYPE
61      SYNTAX      Unsigned32
62      UNITS        "b/s"
63      MAX-ACCESS  read-create
64      STATUS      current
65

```

```

1      DESCRIPTION
2          "Minimum Tolerable Traffic Rate =  $R$  (bits/sec) with
3          time base  $T$ (sec) means the following. Let  $S$  denote
4          additional demand accumulated at the MAC SAP of the
5          transmitter during an arbitrary time interval of the
6          length  $T$ . Then the amount of data forwarded at the
7          receiver to CS (in bits) during this interval should
8          be not less than  $\min \{S, R * T\}$ ."
9
10     REFERENCE
11         "Subclause 11.13.9 in IEEE Std 802.16-2004"
12     ::= { wmanIfBsServiceClassEntry 19 }
13
14
15     wmanIfBsQoSReqTxPolicy OBJECT-TYPE
16         SYNTAX      BITS {noBroadcastBwReq(0),
17                         reserved1(1),
18                         noPiggybackReq(2),
19                         noFragmentData(3),
20                         noPHS(4),
21                         noSduPacking(5),
22                         noCrc(6),
23                         reserved2(7)}
24
25     MAX-ACCESS      read-create
26     STATUS           current
27     DESCRIPTION
28         "The value of this parameter provides the capability to
29         specify certain attributes for the associated service
30         flow. An attribute is enabled by setting the
31         corresponding bit position to 1."
32     REFERENCE      "Subclause 11.13.12 in IEEE Std 802.16-2004"
33     ::= { wmanIfBsServiceClassEntry 20 }
34
35
36     wmanIfBsQoSServiceClassRowStatus OBJECT-TYPE
37         SYNTAX      RowStatus
38         MAX-ACCESS      read-create
39         STATUS           current
40         DESCRIPTION
41             "This object is used to create a new row or modify or
42             delete an existing row in this table.
43
44             If the implementator of this MIB has choosen not
45             to implement 'dynamic assignment' of profiles, this
46             object is not useful and should return noSuchName
47             upon SNMP request."
48         ::= { wmanIfBsServiceClassEntry 21 }
49
50
51     wmanIfBsClassifierRuleTable OBJECT-TYPE
52         SYNTAX      SEQUENCE OF WmanIfBsClassifierRuleEntry
53         MAX-ACCESS      not-accessible
54         STATUS           current
55         DESCRIPTION
56             "This table contains packet classifier rules associated
57             with service flows."
58     REFERENCE      "Subclause 11.13.19.3.4 in IEEE Std 802.16-2004"
59
60
61
62
63
64
65

```

```

1      ::= { wmanIfBsPacketCs 4 }
2
3  wmanIfBsClassifierRuleEntry OBJECT-TYPE
4      SYNTAX      WmanIfBsClassifierRuleEntry
5      MAX-ACCESS  not-accessible
6      STATUS      current
7      DESCRIPTION
8          "This table provides one row for each packet classifier
9          rule, and is indexed by ifIndex, wmanIfBsSfId, and
10         wmanIfBsClassifierRuleIndex. IfIdex is associated with
11         the BS sector. wmanIfBsSfId identifies the service flow,
12         while wmanIfBsClassifierRuleIndex identifies the packet
13         classifier rule."
14     INDEX { ifIndex, wmanIfBsSfId, wmanIfBsClassifierRuleIndex }
15     ::= { wmanIfBsClassifierRuleTable 1 }
16
17  WmanIfBsClassifierRuleEntry ::= SEQUENCE {
18      wmanIfBsClassifierRuleIndex      Unsigned32,
19      wmanIfBsClassifierRulePriority    INTEGER,
20      wmanIfBsClassifierRuleIpTosLow   INTEGER,
21      wmanIfBsClassifierRuleIpTosHigh  INTEGER,
22      wmanIfBsClassifierRuleIpTosMask  INTEGER,
23      wmanIfBsClassifierRuleIpProtocol Integer32,
24      wmanIfBsClassifierRuleIpSourceAddr InetAddress,
25      wmanIfBsClassifierRuleIpSourceMask InetAddress,
26      wmanIfBsClassifierRuleIpDestAddr  InetAddress,
27      wmanIfBsClassifierRuleIpDestMask  InetAddress,
28      wmanIfBsClassifierRuleSourcePortStart Integer32,
29      wmanIfBsClassifierRuleSourcePortEnd Integer32,
30      wmanIfBsClassifierRuleDestPortStart Integer32,
31      wmanIfBsClassifierRuleDestPortEnd Integer32,
32      wmanIfBsClassifierRuleDestMacAddr  MacAddress,
33      wmanIfBsClassifierRuleDestMacMask  MacAddress,
34      wmanIfBsClassifierRuleSourceMacAddr MacAddress,
35      wmanIfBsClassifierRuleSourceMacMask MacAddress,
36      wmanIfBsClassifierRuleEnetProtocolType INTEGER,
37      wmanIfBsClassifierRuleEnetProtocol Integer32,
38      wmanIfBsClassifierRuleUserPriLow   Integer32,
39      wmanIfBsClassifierRuleUserPriHigh  Integer32,
40      wmanIfBsClassifierRuleVlanId       Integer32,
41      wmanIfBsClassifierRuleState        INTEGER,
42      wmanIfBsClassifierRulePhsSize      Integer32,
43      wmanIfBsClassifierRulePhsMask      OCTET STRING,
44      wmanIfBsClassifierRulePhsVerify    WmanIfPhsRuleVerify,
45      wmanIfBsClassifierRuleIpv6FlowLabel WmanIfIpv6FlowLabel,
46      wmanIfBsClassifierRuleBitMap       WmanIfClassifierBitMap,
47      wmanIfBsClassifierRuleRowStatus    RowStatus}
48
49  wmanIfBsClassifierRuleIndex OBJECT-TYPE
50      SYNTAX      Unsigned32 (1..4294967295)
51      MAX-ACCESS  not-accessible
52      STATUS      current
53      DESCRIPTION
54          "An index is assigned to a classifier in BS classifiers
55
56
57
58
59
60
61
62
63
64
65

```

```

1         table"
2         ::= { wmanIfBsClassifierRuleEntry 1 }
3
4
5 wmanIfBsClassifierRulePriority OBJECT-TYPE
6     SYNTAX      INTEGER (0..255)
7     MAX-ACCESS  read-create
8     STATUS      current
9     DESCRIPTION
10        "The value specifies the priority for the Classifier, which
11        is used for determining the order of the Classifier. A
12        higher value indicates higher priority. Classifiers may
13        have priorities in the range 0..255."
14    REFERENCE
15        "Subclause 11.13.19.3.4.1 in IEEE Std 802.16-2004"
16    DEFVAL      { 0 }
17    ::= { wmanIfBsClassifierRuleEntry 2 }
18
19
20
21 wmanIfBsClassifierRuleIpTosLow OBJECT-TYPE
22     SYNTAX      INTEGER (0..255)
23     MAX-ACCESS  read-create
24     STATUS      current
25     DESCRIPTION
26        "The low value of a range of TOS byte values. If the
27        referenced parameter is not present in a classifier, this
28        object reports the value of 0."
29    REFERENCE
30        "Subclause 11.13.19.3.4.2 in IEEE Std 802.16-2004"
31    ::= { wmanIfBsClassifierRuleEntry 3 }
32
33
34
35 wmanIfBsClassifierRuleIpTosHigh OBJECT-TYPE
36     SYNTAX      INTEGER (0..255)
37     MAX-ACCESS  read-create
38     STATUS      current
39     DESCRIPTION
40        "The 8-bit high value of a range of TOS byte values.
41        If the referenced parameter is not present in a classifier,
42        this object reports the value of 0."
43    REFERENCE
44        "Subclause 11.13.19.3.4.2 in IEEE Std 802.16-2004"
45    ::= { wmanIfBsClassifierRuleEntry 4 }
46
47
48
49 wmanIfBsClassifierRuleIpTosMask OBJECT-TYPE
50     SYNTAX      INTEGER (0..255)
51     MAX-ACCESS  read-create
52     STATUS      current
53     DESCRIPTION
54        "The value of this object specifies the matching parameter
55        for the IP type of service/DSCP [IETF RFC 2474] byte mask.
56        An IP packet with IP type of service (ToS) byte value
57        ip-tos matches this parameter if tos-low less than or
58        equal (ip-tos AND tos-mask) less than or equal tos-high."
59    REFERENCE
60        "Subclause 11.13.19.3.4.2 in IEEE Std 802.16-2004"
61    ::= { wmanIfBsClassifierRuleEntry 5 }
62
63
64
65

```

```

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

```

```

1         packet matches the rule when the packet IP destination
2         address bitwise ANDed with the
3         wmanIfBsClassifierRuleIpDestMask value equals the
4         wmanIfBsClassifierRuleIpDestAddr value.
5         If the referenced parameter is not present in a
6         classifier, this object reports the value of 0.0.0.0."
7
8     REFERENCE
9         "Subclause 11.13.19.3.4.5 in IEEE Std 802.16-2004"
10    ::= { wmanIfBsClassifierRuleEntry 9 }
11
12
13    wmanIfBsClassifierRuleIpDestMask OBJECT-TYPE
14        SYNTAX      InetAddress
15        MAX-ACCESS   read-create
16        STATUS       current
17        DESCRIPTION
18            "This object specifies which bits of a packet's IP
19             Destination Address that are compared to match this rule.
20             An IP packet matches the rule when the packet destination
21             address bitwise ANDed with the
22             wmanIfBsClassifierRuleIpDestMask value equals the
23             wmanIfBsClassifierRuleIpDestAddr value.
24             If the referenced parameter is not present in a classifier
25             , this object reports the value of 0.0.0.0."
26
27        REFERENCE
28            "Subclause 11.13.19.3.4.5 in IEEE Std 802.16-2004"
29        ::= { wmanIfBsClassifierRuleEntry 10 }
30
31
32
33    wmanIfBsClassifierRuleSourcePortStart OBJECT-TYPE
34        SYNTAX      Integer32 (0..65535)
35        MAX-ACCESS   read-create
36        STATUS       current
37        DESCRIPTION
38            "This object specifies the low end inclusive range of
39             TCP/UDP source port numbers to which a packet is compared.
40             This object is irrelevant for non-TCP/UDP IP packets.
41             If the referenced parameter is not present in a
42             classifier, this object reports the value of 0."
43
44        REFERENCE
45            "Subclause 11.13.19.3.4.6 in IEEE Std 802.16-2004"
46        ::= { wmanIfBsClassifierRuleEntry 11 }
47
48
49
50    wmanIfBsClassifierRuleSourcePortEnd OBJECT-TYPE
51        SYNTAX      Integer32 (0..65535)
52        MAX-ACCESS   read-create
53        STATUS       current
54        DESCRIPTION
55            "This object specifies the high end inclusive range of
56             TCP/UDP source port numbers to which a packet is compared.
57             This object is irrelevant for non-TCP/UDP IP packets.
58             If the referenced parameter is not present in a classifier,
59             this object reports the value of 65535."
60
61        REFERENCE
62            "Subclause 11.13.19.3.4.6 in IEEE Std 802.16-2004"
63        ::= { wmanIfBsClassifierRuleEntry 12 }
64
65

```

```

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

```



## REFERENCE

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

## wmanIfBsClassifierRuleSourceMacAddr OBJECT-TYPE

SYNTAX        MacAddress  
 MAX-ACCESS   read-create  
 STATUS        current  
 DESCRIPTION

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

## REFERENCE

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

## wmanIfBsClassifierRuleSourceMacMask OBJECT-TYPE

SYNTAX        MacAddress  
 MAX-ACCESS   read-create  
 STATUS        current  
 DESCRIPTION

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

## REFERENCE

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

## wmanIfBsClassifierRuleEnetProtocolType OBJECT-TYPE

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

MAX-ACCESS   read-create  
 STATUS        current  
 DESCRIPTION

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

```

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

```

```

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
7            field within the range of wmanIfBsClassifierRuleUserPriLow
8            and wmanIfBsClassifierRuleUserPriHigh in order to match
9            this rule.
10           If the referenced parameter is not present in the
11           classifier, the value of this object is reported as 7."
12      REFERENCE
13          "Subclause 11.13.19.3.4.11 in IEEE Std 802.16-2004"
14          ::= { wmanIfBsClassifierRuleEntry 22 }
15
16      wmanIfBsClassifierRuleVlanId OBJECT-TYPE
17          SYNTAX      Integer32 (0..4095)
18          MAX-ACCESS  read-create
19          STATUS      current
20          DESCRIPTION
21              "This object applies only to Ethernet frames using the
22                802.1P/Q tag header.
23                If this object's value is nonzero, tagged packets must
24                have a VLAN Identifier that matches the value in order
25                to match the rule.
26                Only the least significant 12 bits of this object's
27                value are valid.
28                If the referenced parameter is not present in the
29                classifier, the value of this object is reported as 0."
30          REFERENCE
31              "Subclause 11.13.19.3.4.12 in IEEE Std 802.16-2004"
32              ::= { wmanIfBsClassifierRuleEntry 23 }
33
34      wmanIfBsClassifierRuleState OBJECT-TYPE
35          SYNTAX      INTEGER {active(1),
36                           inactive(2)}
37          MAX-ACCESS  read-create
38          STATUS      current
39          DESCRIPTION
40              "This object indicates whether or not the classifier is
41                enabled to classify packets to a Service Flow.
42                If the referenced parameter is not present in the
43                classifier, the value of this object is reported
44                as active(1)."
45          ::= { wmanIfBsClassifierRuleEntry 24 }
46
47      wmanIfBsClassifierRulePhsSize OBJECT-TYPE
48          SYNTAX      Integer32
49          UNITS        "byte"
50          MAX-ACCESS  read-create
51          STATUS      current
52          DESCRIPTION
53              "This object is used to configure the PHS rule for this
54                classifier. The value of this field - PHSS is the total
55

```

```

1         number of bytes in the header to be suppressed and then
2         restored in a service flow that uses PHS. If the value of
3         this field is 0 bytes then PHS is disabled for this
4         classifier. If flag phsMask in wmanIfBsClassifierRuleBitMap
5         is set to 0 and flag phsSize in
6         wmanIfBsClassifierRuleBitMap is set to 0, then BS can still
7         create PHS rules using its own custom mask (i.e. the rule
8         is not configured by NMS)."
```

REFERENCE

```

11         "Subclause 11.13.19.3.7.4 in IEEE Std 802.16-2004"
12
13     DEFVAL        {0}
14     ::= { wmanIfBsClassifierRuleEntry 25 }
```

wmanIfBsClassifierRulePhsMask OBJECT-TYPE

```

17     SYNTAX          OCTET STRING (SIZE(0..65535))
18     MAX-ACCESS      read-create
19     STATUS           current
20     DESCRIPTION
21         "This object is used to configure the PHS rule for this
22         classifier. It is encoded as follows:
23         bit 0:
24             0 = don't suppress the 1st byte of the suppression field
25             1 = suppress first byte of the suppression field
26         bit 1:
27             0 = don't suppress the 2nd byte of the suppression field
28             1 = suppress second byte of the suppression field
29         bit x:
30             0 = don't suppress the (x+1) byte of the suppression
31             field
32             1 = suppress (x+1) byte of the suppression field
33         where the length of the octet string is ceiling
34         (wmanIfBsClassifierRulePhsSize/8). BS should use this value
35         to create a new PHS rule index (PHSI) and field (PHSF) as
36         defined in the standard. If flag phsMask in
37         wmanIfBsClassifierRuleBitMap is set to 0 and flag phsSize
38         in wmanIfBsClassifierRuleBitMap is set to 0, then BS can
39         still create PHS rules using its own custom mask (i.e. the
40         rule is not configured by NMS)."
```

REFERENCE

```

47         "Subclause 11.13.19.3.7.3 in IEEE Std 802.16-2004"
48
49     ::= { wmanIfBsClassifierRuleEntry 26 }
```

wmanIfBsClassifierRulePhsVerify OBJECT-TYPE

```

52     SYNTAX          WmanIfPhsRuleVerify
53     MAX-ACCESS      read-create
54     STATUS           current
55     DESCRIPTION
56         "The value of this field indicates to the sending entity
57         whether or not the packet header contents are to be
58         verified prior to performing suppression."
59
60     DEFVAL          { phsVerifyEnable }
61     ::= { wmanIfBsClassifierRuleEntry 27 }
```

wmanIfBsClassifierRuleIpv6FlowLabel OBJECT-TYPE

```

1      SYNTAX      WmanIfIpv6FlowLabel
2      MAX-ACCESS  read-create
3      STATUS      current
4      DESCRIPTION
5          "The value of this field specifies the matching values for
6          the IPv6 Flow label field."
7      ::= { wmanIfBsClassifierRuleEntry 28 }
8
9
10
11  wmanIfBsClassifierRuleBitMap OBJECT-TYPE
12      SYNTAX      WmanIfClassifierBitMap
13      MAX-ACCESS  read-create
14      STATUS      current
15      DESCRIPTION
16          "This object indicates which parameter encodings were
17          actually present in the entry. A bit set to '1' indicates
18          the corresponding classifier encoding is present, and '0'
19          means otherwise"
20      ::= { wmanIfBsClassifierRuleEntry 29 }
21
22
23
24  wmanIfBsClassifierRuleRowStatus OBJECT-TYPE
25      SYNTAX      RowStatus
26      MAX-ACCESS  read-create
27      STATUS      current
28      DESCRIPTION
29          "This object is used to create a new row or modify or
30          delete an existing row in this table.
31
32          If the implementator of this MIB has choosen not
33          to implement 'dynamic assignment' of profiles, this
34          object is not useful and should return noSuchName
35          upon SNMP request."
36      ::= { wmanIfBsClassifierRuleEntry 30 }
37
38
39
40
41  wmanIfBsSsPacketCounterTable OBJECT-TYPE
42      SYNTAX      SEQUENCE OF WmanIfBsSsPacketCounterEntry
43      MAX-ACCESS  not-accessible
44      STATUS      current
45      DESCRIPTION
46          "This table contains counters to keep track of the number
47          of packets and octets that have been received or
48          transmitted on the per service flow basis."
49      ::= { wmanIfBsPacketCs 5 }
50
51
52
53  wmanIfBsSsPacketCounterEntry OBJECT-TYPE
54      SYNTAX      WmanIfBsSsPacketCounterEntry
55      MAX-ACCESS  not-accessible
56      STATUS      current
57      DESCRIPTION
58          "This table provides one row for each service flow, and
59          is indexed by ifIndex, wmanIfCmnCpsSfMacAddress, and
60          wmanIfCmnCpsSfId."
61      INDEX { ifIndex, wmanIfCmnCpsSfMacAddress,
62              wmanIfCmnCpsSfId }
63      ::= { wmanIfBsSsPacketCounterTable 1 }
64
65

```

```

1
2 WmanIfBsSsPacketCounterEntry ::= SEQUENCE {
3     wmanIfBsSsMacSduCount          Counter64,
4     wmanIfBsSsOctetCount           Counter64,
5     wmanIfBsSsResetCounter         INTEGER,
6     wmanIfBsSsResetCounterTime     TimeStamp}
7
8
9
10 wmanIfBsSsMacSduCount OBJECT-TYPE
11     SYNTAX          Counter64
12     MAX-ACCESS      read-only
13     STATUS          current
14     DESCRIPTION
15         "This object counts the number of MAC SDUs that have
16         been transmitted or received."
17     ::= { wmanIfBsSsPacketCounterEntry 1 }
18
19
20 wmanIfBsSsOctetCount OBJECT-TYPE
21     SYNTAX          Counter64
22     MAX-ACCESS      read-only
23     STATUS          current
24     DESCRIPTION
25         "This object counts the number of octets of MAC SDUs
26         that have been transmitted or received."
27     ::= { wmanIfBsSsPacketCounterEntry 2 }
28
29
30
31 wmanIfBsSsResetCounter OBJECT-TYPE
32     SYNTAX          INTEGER {null(0),
33                        resetCounter(1)}
34     MAX-ACCESS      read-write
35     STATUS          current
36     DESCRIPTION
37         "When this attribute is SET to resetCounter(1), the
38         corresponding entry of packet counters will be reset.
39         A GET operation performed on this object will always
40         return null(0). The counter is normally reset after
41         the packet count information is retrieved. "
42     ::= { wmanIfBsSsPacketCounterEntry 3 }
43
44
45
46 wmanIfBsSsResetCounterTime OBJECT-TYPE
47     SYNTAX          TimeStamp
48     MAX-ACCESS      read-only
49     STATUS          current
50     DESCRIPTION
51         "Indicates the date and time when the counter is
52         reset."
53     ::= { wmanIfBsSsPacketCounterEntry 4 }
54
55
56
57 --
58 -- wmanIfBsCps contain the Base Station Common Part Sublayer objects
59 --
60
61 wmanIfBsCps OBJECT IDENTIFIER ::= { wmanIfBsObjects 2 }
62
63
64 wmanIfBsRegisteredSsTable OBJECT-TYPE
65     SYNTAX          SEQUENCE OF WmanIfBsRegisteredSsEntry

```

```

1      MAX-ACCESS    not-accessible
2      STATUS        current
3      DESCRIPTION
4          "This table contains the basic capability information
5          of SSs that have been negotiated and agreed between
6          BS and SS via REG-REQ and REG-RSP messages. An entry
7          in this table indicates the SS has entered and registered
8          into the BS."
9
10     REFERENCE
11         "Subclause 6.3.2.3.7 in IEEE Std 802.16-2004"
12     ::= { wmanIfBsCps 1 }
13
14
15     wmanIfBsRegisteredSsEntry OBJECT-TYPE
16         SYNTAX      WmanIfBsRegisteredSsEntry
17         MAX-ACCESS  not-accessible
18         STATUS      current
19         DESCRIPTION
20             "This table provides one row for each SS that has been
21             registered in the BS, and is indexed by
22             wmanIfBsSsMacAddress. The primary index is the ifIndex
23             with an ifType of propBWAmp2Mp, indicating the BS sector
24             with which the SS is associated. wmanIfBsSsMacAddress
25             identifies the SS being registered."
26         INDEX { ifIndex, wmanIfBsSsMacAddress }
27         ::= { wmanIfBsRegisteredSsTable 1 }
28
29
30     WmanIfBsRegisteredSsEntry ::= SEQUENCE {
31         wmanIfBsSsMacAddress          MacAddress,
32         wmanIfBsSsBasicCid             WmanIfCidType,
33         wmanIfBsSsPrimaryCid           WmanIfCidType,
34         wmanIfBsSsSecondaryCid         WmanIfCidType,
35         wmanIfBsSsManagementSupport    INTEGER,
36         wmanIfBsSsIpManagementMode     INTEGER,
37         wmanIfBsSs2ndMgmtArqEnable     TruthValue,
38         wmanIfBsSs2ndMgmtArqWindowSize INTEGER,
39         wmanIfBsSs2ndMgmtArqDnLinkTxDelay INTEGER,
40         wmanIfBsSs2ndMgmtArqUpLinkTxDelay INTEGER,
41         wmanIfBsSs2ndMgmtArqDnLinkRxDelay INTEGER,
42         wmanIfBsSs2ndMgmtArqUpLinkRxDelay INTEGER,
43         wmanIfBsSs2ndMgmtArqBlockLifetime INTEGER,
44         wmanIfBsSs2ndMgmtArqSyncLossTimeout INTEGER,
45         wmanIfBsSs2ndMgmtArqDeliverInOrder TruthValue,
46         wmanIfBsSs2ndMgmtArqRxPurgeTimeout INTEGER,
47         wmanIfBsSs2ndMgmtArqBlockSize INTEGER,
48         wmanIfBsSsVendorIdEncoding     OCTET STRING,
49         wmanIfBsSsAasBroadcastPermission INTEGER,
50         wmanIfBsSsMaxTxPowerBpsk       WmanIfMaxTxPowerType,
51         wmanIfBsSsMaxTxPowerQpsk       WmanIfMaxTxPowerType,
52         wmanIfBsSsMaxTxPower16Qam      WmanIfMaxTxPowerType,
53         wmanIfBsSsMaxTxPower64Qam      WmanIfMaxTxPowerType,
54         wmanIfBsSsMacVersion            WmanIfMacVersion}
55
56     wmanIfBsSsMacAddress OBJECT-TYPE
57         SYNTAX      MacAddress
58
59
60
61
62
63
64
65

```

```

1      MAX-ACCESS    not-accessible
2      STATUS        current
3      DESCRIPTION
4          "The MAC address of SS is received from the RNG-REQ
5          message. When SS registers, this MAC address is entered
6          into the table, and used as the identifier to the SS."
7      REFERENCE
8          "Subclause 6.3.2.3.5 in IEEE Std 802.16-2004"
9      ::= { wmanIfBsRegisteredSsEntry 1 }
10
11
12
13  wmanIfBsSsBasicCid OBJECT-TYPE
14      SYNTAX          WmanIfCidType
15      MAX-ACCESS      read-only
16      STATUS          current
17      DESCRIPTION
18          "The value of this object indicates the SS's basic CID
19          that was sent in the RNG-RSP message."
20      REFERENCE
21          "Subclause 6.3.2.3.6 in IEEE Std 802.16-2004"
22      ::= { wmanIfBsRegisteredSsEntry 2 }
23
24
25
26  wmanIfBsSsPrimaryCid OBJECT-TYPE
27      SYNTAX          WmanIfCidType
28      MAX-ACCESS      read-only
29      STATUS          current
30      DESCRIPTION
31          "The value of this object indicates the primary CID of the
32          SS received from the RNG-RSP message."
33      REFERENCE
34          "Subclause 6.3.2.3.6 in IEEE Std 802.16-2004"
35      ::= { wmanIfBsRegisteredSsEntry 3 }
36
37
38
39  wmanIfBsSsSecondaryCid OBJECT-TYPE
40      SYNTAX          WmanIfCidType
41      MAX-ACCESS      read-only
42      STATUS          current
43      DESCRIPTION
44          "The value of this object indicates the secondary
45          management CID present in the REG-RSP message. The value
46          should be null if the 2nd management channel is not
47          available."
48      REFERENCE
49          "Subclause 6.4.2.3.8 in IEEE Std 802.16-2004"
50      ::= { wmanIfBsRegisteredSsEntry 4 }
51
52
53
54  wmanIfBsSsManagementSupport OBJECT-TYPE
55      SYNTAX          INTEGER {unmanagedSs(0),
56                          managedSs(1)}
57      MAX-ACCESS      read-only
58      STATUS          current
59      DESCRIPTION
60          "This object indicates whether or not the SS is managed."
61      REFERENCE
62          "Subclause 11.7.2 in IEEE Std 802.16-2004"
63
64
65

```



```

1      ::= { wmanIfBsRegisteredSsEntry 5 }
2
3  wmanIfBsSsIpManagementMode OBJECT-TYPE
4      SYNTAX      INTEGER {unmanaged(0),
5                      ipManaged(1)}
6
7      MAX-ACCESS  read-only
8      STATUS      current
9      DESCRIPTION
10         "The IP management mode parameter dictates whether
11         the provider intends to manage the SS on an ongoing
12         basis via IP-based mechanisms."
13
14     REFERENCE
15         "Subclause 11.7.3 in IEEE Std 802.16-2004"
16     ::= { wmanIfBsRegisteredSsEntry 6 }
17
18
19  wmanIfBsSs2ndMgmtArqEnable OBJECT-TYPE
20      SYNTAX      TruthValue
21      MAX-ACCESS  read-only
22      STATUS      current
23      DESCRIPTION
24         "True(1) ARQ enabling is requested for the 2nd
25         management channel."
26
27     REFERENCE
28         "Subclause 11.13.18.1 in IEEE Std 802.16-2004"
29     ::= { wmanIfBsRegisteredSsEntry 7 }
30
31
32  wmanIfBsSs2ndMgmtArqWindowSize OBJECT-TYPE
33      SYNTAX      INTEGER (1 .. 1024)
34      MAX-ACCESS  read-only
35      STATUS      current
36      DESCRIPTION
37         "Indicates the maximum number of unacknowledged
38         fragments at any time for 2nd management channel."
39
40     REFERENCE
41         "Subclause 11.13.18.2 in IEEE Std 802.16-2004"
42     ::= { wmanIfBsRegisteredSsEntry 8 }
43
44
45  wmanIfBsSs2ndMgmtArqDnLinkTxDelay OBJECT-TYPE
46      SYNTAX      INTEGER (0 .. 65535)
47      UNITS        "us"
48      MAX-ACCESS  read-only
49      STATUS      current
50      DESCRIPTION
51         "The object defines the ARQ transmitter delay for
52         downlink transmission."
53
54     REFERENCE
55         "Subclause 11.13.18.3 in IEEE Std 802.16-2004"
56     ::= { wmanIfBsRegisteredSsEntry 9 }
57
58
59  wmanIfBsSs2ndMgmtArqUpLinkTxDelay OBJECT-TYPE
60      SYNTAX      INTEGER (0 .. 65535)
61      UNITS        "us"
62      MAX-ACCESS  read-only
63      STATUS      current
64
65

```

```

1      DESCRIPTION
2          "The object defines the ARQ transmitter delay for
3          uplink transmission."
4
5      REFERENCE
6          "Subclause 11.13.18.3 in IEEE Std 802.16-2004"
7          ::= { wmanIfBsRegisteredSsEntry 10 }
8
9
10     wmanIfBsSs2ndMgmtArqDnLinkRxDelay OBJECT-TYPE
11         SYNTAX      INTEGER (0 .. 65535)
12         UNITS        "us"
13         MAX-ACCESS   read-only
14         STATUS        current
15         DESCRIPTION
16             "The object defines the ARQ receiver delay for
17             downlink transmission."
18
19         REFERENCE
20             "Subclause 11.13.18.3 in IEEE Std 802.16-2004"
21             ::= { wmanIfBsRegisteredSsEntry 11 }
22
23
24     wmanIfBsSs2ndMgmtArqUpLinkRxDelay OBJECT-TYPE
25         SYNTAX      INTEGER (0 .. 65535)
26         UNITS        "us"
27         MAX-ACCESS   read-only
28         STATUS        current
29         DESCRIPTION
30             "The object defines the ARQ receiver delay for
31             uplink transmission."
32
33         REFERENCE
34             "Subclause 11.13.18.3 in IEEE Std 802.16-2004"
35             ::= { wmanIfBsRegisteredSsEntry 12 }
36
37
38     wmanIfBsSs2ndMgmtArqBlockLifetime OBJECT-TYPE
39         SYNTAX      INTEGER (0 .. 65535)
40         UNITS        "10 us"
41         MAX-ACCESS   read-only
42         STATUS        current
43         DESCRIPTION
44             "The maximum time interval an ARQ fragment will be
45             managed by the transmitter ARQ machine, once
46             initial transmission of the fragment has occurred.
47             If transmission or retransmission of the fragment
48             is not acknowledged by the receiver before the
49             time limit is reached, the fragment is discarded.
50             A value of 0 means Infinite."
51
52         REFERENCE
53             "Subclause 11.13.18.4 in IEEE Std 802.16-2004"
54
55         DEFVAL      {0}
56         ::= { wmanIfBsRegisteredSsEntry 13 }
57
58
59     wmanIfBsSs2ndMgmtArqSyncLossTimeout OBJECT-TYPE
60         SYNTAX      INTEGER (0 .. 65535)
61         UNITS        "10 us"
62         MAX-ACCESS   read-only
63         STATUS        current
64
65

```

```

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      REFERENCE
6          "Subclause 11.13.18.5 in IEEE Std 802.16-2004"
7      DEFVAL    {0}
8      ::= { wmanIfBsRegisteredSsEntry 14 }
9
10
11
12  wmanIfBsSs2ndMgmtArqDeliverInOrder OBJECT-TYPE
13      SYNTAX      TruthValue
14      MAX-ACCESS  read-only
15      STATUS      current
16      DESCRIPTION
17          "Indicates whether or not data is to be delivered
18            by the receiving MAC to its client application
19            in the order in which data was handed off to the
20            originating MAC."
21      REFERENCE
22          "Subclause 11.13.18.6 in IEEE Std 802.16-2004"
23      ::= { wmanIfBsRegisteredSsEntry 15 }
24
25
26
27  wmanIfBsSs2ndMgmtArqRxPurgeTimeout OBJECT-TYPE
28      SYNTAX      INTEGER (0 .. 65535)
29      UNITS        "10 us"
30      MAX-ACCESS  read-only
31      STATUS      current
32      DESCRIPTION
33          "Indicates the time interval the ARQ window is advanced
34            after a fragment is received. A value of 0 means Infinite."
35      REFERENCE
36          "Subclause 11.13.18.7 in IEEE Std 802.16-2004"
37      DEFVAL    {0}
38      ::= { wmanIfBsRegisteredSsEntry 16 }
39
40
41
42
43  wmanIfBsSs2ndMgmtArqBlockSize OBJECT-TYPE
44      SYNTAX      INTEGER (1 .. 2040)
45      MAX-ACCESS  read-only
46      STATUS      current
47      DESCRIPTION
48          "This parameter specifies the size of a ARQ block. This
49            parameter shall be established by negotiation during the
50            connection setup. The requester includes its desired
51            setting in the REQ message. The receiver of the REQ
52            message shall take the smaller of the value it prefers and
53            value in the REQ message. The minimum value is included in
54            the RSP message."
55      REFERENCE
56          "Subclause 11.13.18.8 in IEEE Std 802.16-2004"
57      ::= { wmanIfBsRegisteredSsEntry 17 }
58
59
60
61
62  wmanIfBsSsVendorIdEncoding OBJECT-TYPE
63      SYNTAX      OCTET STRING (SIZE(3))
64      MAX-ACCESS  read-only
65

```

```

1      STATUS      current
2      DESCRIPTION
3          "The value field contains the vendor identification
4          specified by the 3 byte vendor-specific organizationally
5          unique identifier of the SS or BS MAC address. A vendor ID
6          used in a REG-REQ shall be the Vendor ID of the SS sending
7          the request. A vendor ID used in a REG-RSP shall be the
8          Vendor ID of the BS sending the response."
9
10     REFERENCE
11         "Subclause 11.1.5 in IEEE Std 802.16-2004"
12     ::= { wmanIfBsRegisteredSsEntry 18 }
13
14
15     wmanIfBsSsAasBroadcastPermission OBJECT-TYPE
16     SYNTAX      INTEGER {contBasedBwReqPermitted(0),
17                     contBasedBwReqNotPermitted(1)}
18     MAX-ACCESS  read-only
19     STATUS      current
20     DESCRIPTION
21         "This parameter specifies if SS can issue contention-based
22         bandwidth request or not."
23     REFERENCE
24         "Subclause 11.6 in IEEE Std 802.16-2004"
25     ::= { wmanIfBsRegisteredSsEntry 19 }
26
27
28     wmanIfBsSsMaxTxPowerBpsk OBJECT-TYPE
29     SYNTAX      WmanIfMaxTxPowerType
30     MAX-ACCESS  read-only
31     STATUS      current
32     DESCRIPTION
33         "The maximum available power for BPSK. The maximum power
34         parameters are reported in dBm and quantized in 0.5 dBm
35         steps ranging from -64 dBm (encoded 0x00) to 63.5 dBm
36         (encoded 0xFF). Values outside this range shall be assigned
37         the closest extreme. This parameter is only applicable to
38         systems supporting the Sca, OFDM or OFDMA PHY."
39     REFERENCE
40         "Subclause 11.8.3.2 in IEEE Std 802.16-2004"
41     ::= { wmanIfBsRegisteredSsEntry 20 }
42
43
44     wmanIfBsSsMaxTxPowerQpsk OBJECT-TYPE
45     SYNTAX      WmanIfMaxTxPowerType
46     MAX-ACCESS  read-only
47     STATUS      current
48     DESCRIPTION
49         "The maximum available power for QPSK. The maximum power
50         parameters are reported in dBm and quantized in 0.5 dBm
51         steps ranging from -64 dBm (encoded 0x00) to 63.5 dBm
52         (encoded 0xFF). Values outside this range shall be assigned
53         to closest extreme. This parameter is only applicable to
54         systems supporting the Sca, OFDM or OFDMA PHY."
55     REFERENCE
56         "Subclause 11.8.3.2 in IEEE Std 802.16-2004"
57     ::= { wmanIfBsRegisteredSsEntry 21 }
58
59
60
61
62
63
64
65

```

```

1  wmanIfBsSsMaxTxPower16Qam OBJECT-TYPE
2      SYNTAX          WmanIfMaxTxPowerType
3      MAX-ACCESS      read-only
4      STATUS          current
5      DESCRIPTION
6          "The maximum available power for 16-QAM constellations.
7          The maximum power parameters are reported in dBm and
8          quantized in 0.5 dBm steps ranging from -64 dBm (encoded
9          0x00) to 63.5 dBm (encoded 0xFF). Values outside this
10         range shall be assigned the closest extreme. This parameter
11         is only applicable to systems supporting the SCa, OFDM or
12         OFDMA PHY."
13     REFERENCE
14         "Subclause 11.8.3.2 in IEEE Std 802.16-2004"
15     ::= { wmanIfBsRegisteredSsEntry 22 }
16
17  wmanIfBsSsMaxTxPower64Qam OBJECT-TYPE
18      SYNTAX          WmanIfMaxTxPowerType
19      MAX-ACCESS      read-only
20      STATUS          current
21      DESCRIPTION
22          "The maximum available power for 64-QAM constellations.
23          The maximum power parameters are reported in dBm and
24          quantized in 0.5 dBm steps ranging from -64 dBm (encoded
25          0x00) to 63.5 dBm (encoded 0xFF). Values outside this
26          range shall be assigned the closest extreme. SSs that do
27          not support QAM64 shall report the value of 0x00. This
28          parameter is only applicable to systems supporting the SCa,
29          OFDM or OFDMA PHY."
30     REFERENCE
31         "Subclause 11.8.3.2 in IEEE Std 802.16-2004"
32     ::= { wmanIfBsRegisteredSsEntry 23 }
33
34  wmanIfBsSsMacVersion 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 the
40          message originator conforms."
41     REFERENCE
42         "Subclause 11.1.3 in IEEE Std 802.16-2004"
43     ::= { wmanIfBsRegisteredSsEntry 24 }
44
45  --
46  -- wmanIfBsConfigurationTable contains global parameters common in BS
47  --
48  wmanIfBsConfigurationTable OBJECT-TYPE
49      SYNTAX          SEQUENCE OF WmanIfBsConfigurationEntry
50      MAX-ACCESS      not-accessible
51      STATUS          current
52      DESCRIPTION
53          "This table provides one row for each BS sector that
54          contains the BS system parameters as defined in Subclause

```

10.1 of [3]. The objects in this table define the default behaviour of the BS for 2nd Management Channel scheduling and SFID allocation as well as configuration parameters of the CPS scheduler and AAS system."

REFERENCE

"Subclause 10.1 in IEEE Std 802.16-2004"

::= { wmanIfBsCps 2 }

wmanIfBsConfigurationEntry OBJECT-TYPE

SYNTAX WmanIfBsConfigurationEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This table is indexed by ifIndex with an ifType of propBWA2Mp."

INDEX { ifIndex }

::= { wmanIfBsConfigurationTable 1 }

WmanIfBsConfigurationEntry ::= SEQUENCE {

wmanIfBsDcdInterval	INTEGER,
wmanIfBsUcdInterval	INTEGER,
wmanIfBsUcdTransition	INTEGER,
wmanIfBsDcdTransition	INTEGER,
wmanIfBsInitialRangingInterval	INTEGER,
wmanIfBsSsULMapProcTime	Unsigned32,
wmanIfBsSsRangRespProcTime	Unsigned32,
wmanIfBsT5Timeout	INTEGER,
wmanIfBsT9Timeout	INTEGER,
wmanIfBsT13Timeout	INTEGER,
wmanIfBsT15Timeout	INTEGER,
wmanIfBsT17Timeout	INTEGER,
wmanIfBsT27IdleTimer	Unsigned32,
wmanIfBsT27ActiveTimer	Unsigned32,
wmanIfBs2ndMgmtDlQoSProfileIndex	INTEGER,
wmanIfBs2ndMgmtUlQoSProfileIndex	INTEGER,
wmanIfBsAutoSfidEnabled	INTEGER,
wmanIfBsAutoSfidRangeMin	Unsigned32,
wmanIfBsAutoSfidRangeMax	Unsigned32,
wmanIfBsAasChanFbckReqFreq	INTEGER,
wmanIfBsAasBeamSelectFreq	INTEGER,
wmanIfBsAasChanFbckReqResolution	INTEGER,
wmanIfBsAasBeamReqResolution	INTEGER,
wmanIfBsAasNumOptDiversityZones	INTEGER,
wmanIfBsResetSector	INTEGER}

```

1  wmanIfBsUcdInterval OBJECT-TYPE
2      SYNTAX          INTEGER (0..10000)
3      UNITS           "milliseconds"
4      MAX-ACCESS      read-write
5      STATUS          current
6      DESCRIPTION
7          "Time between transmission of UCD messages in ms."
8      ::= { wmanIfBsConfigurationEntry 2 }
9
10
11  wmanIfBsUcdTransition OBJECT-TYPE
12      SYNTAX          INTEGER (2..65535)
13      UNITS           "Number of MAC Frames"
14      MAX-ACCESS      read-write
15      STATUS          current
16      DESCRIPTION
17          "The time the BS shall wait after transmitting a UCD message
18          with an incremented Configuration Change Count before
19          issuing a UL-MAP message referring to
20          Uplink_Burst_Profiles defined in that UCD message."
21      ::= { wmanIfBsConfigurationEntry 3 }
22
23
24  wmanIfBsDcdTransition OBJECT-TYPE
25      SYNTAX          INTEGER (2..65535)
26      UNITS           "Number of MAC Frames"
27      MAX-ACCESS      read-write
28      STATUS          current
29      DESCRIPTION
30          "The time the BS shall wait after transmitting a DCD message
31          with an incremented Configuration Change Count before
32          issuing a DL-MAP message referring to
33          Downlink_Burst_Profiles defined in that DCD message."
34      ::= { wmanIfBsConfigurationEntry 4 }
35
36
37  wmanIfBsInitialRangingInterval OBJECT-TYPE
38      SYNTAX          INTEGER(0..2000)
39      UNITS           "milliseconds"
40      MAX-ACCESS      read-write
41      STATUS          current
42      DESCRIPTION
43          "Time between Initial Ranging regions assigned by the BS
44          in ms."
45      ::= { wmanIfBsConfigurationEntry 5 }
46
47
48  wmanIfBsSsULMapProcTime OBJECT-TYPE
49      SYNTAX          Unsigned32 (200 .. 4294967295)
50      UNITS           "micro seconds"
51      MAX-ACCESS      read-write
52      STATUS          current
53      DESCRIPTION
54          "Time provided between arrival of the last bit of a UL-MAP
55          at an SS and effectiveness of that map in us."
56      ::= { wmanIfBsConfigurationEntry 6 }
57
58
59  wmanIfBsSsRangRespProcTime OBJECT-TYPE
60
61
62
63
64
65

```

```

1      SYNTAX      Unsigned32 (10000 .. 4294967295)
2      UNITS       "micro seconds"
3      MAX-ACCESS  read-write
4      STATUS      current
5      DESCRIPTION
6          "Time allowed for an SS following receipt of a ranging
7          response before it is expected to reply to an invited
8          ranging request in us."
9      ::= { wmanIfBsConfigurationEntry 7 }
10
11
12
13  wmanIfBsT5Timeout OBJECT-TYPE
14      SYNTAX      INTEGER (0 .. 2000)
15      UNITS       "milliseconds"
16      MAX-ACCESS  read-write
17      STATUS      current
18      DESCRIPTION
19          "Wait for Uplink Channel Change Response in ms."
20      ::= { wmanIfBsConfigurationEntry 8 }
21
22
23
24  wmanIfBsT9Timeout OBJECT-TYPE
25      SYNTAX      INTEGER (300 .. 65535)
26      UNITS       "milliseconds"
27      MAX-ACCESS  read-write
28      STATUS      current
29      DESCRIPTION
30          "Registration Timeout, the time allowed between the BS
31          sending a RNG-RSP (success) to an SS, and receiving a
32          SBC-REQ from that same SS in ms."
33      ::= { wmanIfBsConfigurationEntry 9 }
34
35
36
37  wmanIfBsT13Timeout OBJECT-TYPE
38      SYNTAX      INTEGER (15 .. 65535)
39      UNITS       "minutes"
40      MAX-ACCESS  read-write
41      STATUS      current
42      DESCRIPTION
43          "The time allowed for an SS, following receipt of a
44          REG-RSP message to send a TFTP-CPLT message to the BS
45          in min."
46      ::= { wmanIfBsConfigurationEntry 10 }
47
48
49
50  wmanIfBsT15Timeout OBJECT-TYPE
51      SYNTAX      INTEGER (20 .. 65535)
52      UNITS       "milliseconds"
53      MAX-ACCESS  read-write
54      STATUS      current
55      DESCRIPTION
56          "Wait for MCA-RSP in ms."
57      ::= { wmanIfBsConfigurationEntry 11 }
58
59
60
61  wmanIfBsT17Timeout OBJECT-TYPE
62      SYNTAX      INTEGER (5 .. 65535)
63      UNITS       "minutes"
64      MAX-ACCESS  read-write
65

```



```

1      STATUS      current
2      DESCRIPTION
3          "Time allowed for SS to complete SS Authorization and
4          Key Exchange in minutes."
5      ::= { wmanIfBsConfigurationEntry 12 }
6
7
8      wmanIfBsT27IdleTimer OBJECT-TYPE
9
10     SYNTAX      Unsigned32 (10000 .. 4294967295)
11     UNITS       "us"
12     MAX-ACCESS  read-write
13     STATUS      current
14     DESCRIPTION
15         "Maximum time between unicast grants to SS when BS believes
16         SS uplink transmission quality is good enough."
17     ::= { wmanIfBsConfigurationEntry 13 }
18
19
20     wmanIfBsT27ActiveTimer OBJECT-TYPE
21
22     SYNTAX      Unsigned32 (10000 .. 4294967295)
23     UNITS       "us"
24     MAX-ACCESS  read-write
25     STATUS      current
26     DESCRIPTION
27         "Maximum time between unicast grants to SS when BS believes
28         SS uplink transmission quality is not good enough."
29     ::= { wmanIfBsConfigurationEntry 14 }
30
31
32
33     wmanIfBs2ndMgmtDlQoSProfileIndex OBJECT-TYPE
34
35     SYNTAX      INTEGER (1..65535)
36     MAX-ACCESS  read-write
37     STATUS      current
38     DESCRIPTION
39         "This object defines the index of a row in
40         wmanIfBsServiceClassTable which is used to obtain all QoS
41         parameters required for the BS downlink scheduler to
42         properly allocate and manage the bandwidth and schedule
43         the 2nd Management Connection traffic. The 2nd Management
44         Connection traffic doesn't differ from Traffic Connection
45         traffic in the area of QoS management."
46     ::= { wmanIfBsConfigurationEntry 15 }
47
48
49
50     wmanIfBs2ndMgmtUlQoSProfileIndex OBJECT-TYPE
51
52     SYNTAX      INTEGER (1..65535)
53     MAX-ACCESS  read-write
54     STATUS      current
55     DESCRIPTION
56         "This object defines the index of a row in
57         wmanIfBsServiceClassTable which is used to obtain all QoS
58         parameters required for the BS uplink scheduler to
59         properly allocate and manage the bandwidth and schedule
60         the 2nd Management Connection traffic. The 2nd Management
61         Connection traffic doesn't differ from Traffic Connection
62         traffic in the area of QoS management."
63     ::= { wmanIfBsConfigurationEntry 16 }
64
65

```

```

1
2 wmanIfBsAutoSfidEnabled OBJECT-TYPE
3     SYNTAX      INTEGER {autoSfidDisabled(0),
4                   autoSfidEnabled(1)}
5
6     MAX-ACCESS  read-write
7     STATUS      current
8     DESCRIPTION
9         "This object defines whether the BS is allowed to
10        autonomously allocate SFIDs. When the object is set to
11        autoSfidEnabled, the BS is allowed to autonomously allocate
12        SFIDs from the range of allowed values defined by
13        wmanIfBsConfigExtAutoSfidRangeMin and
14        wmanIfBsConfigExtAutoSfidRangeMax. A SF is created
15        autonomously when it has not been provisioned in the
16        wmanIfBsProvisionedSfTable and may be initiated by either
17        the SS or BS. The BS should always initiate SF creation
18        based on the provisioned Service flows configured in
19        wmanIfBsProvisionedSfTable."
20
21     REFERENCE
22         "Subclause 11.13.1 in IEEE Std 802.16-2004"
23         ::= { wmanIfBsConfigurationEntry 17 }
24
25
26
27 wmanIfBsAutoSfidRangeMin OBJECT-TYPE
28     SYNTAX      Unsigned32 ( 1 .. 4294967295)
29     MAX-ACCESS  read-write
30     STATUS      current
31     DESCRIPTION
32         "This object defines the minimum value of the range of SFID
33        values allocated for the BS sector for the purpose of
34        autonomous creation of service flows. This value is used
35        when the object wmanIfBsAutoSfidEnabled allows
36        autonomous creation of SFIDs."
37
38     REFERENCE
39         "Subclause 11.13.1 in IEEE Std 802.16-2004"
40         ::= { wmanIfBsConfigurationEntry 18 }
41
42
43
44 wmanIfBsAutoSfidRangeMax OBJECT-TYPE
45     SYNTAX      Unsigned32 ( 1 .. 4294967295)
46     MAX-ACCESS  read-write
47     STATUS      current
48     DESCRIPTION
49         "This object defines the maximum value of the range of SFID
50        values allocated for the BS sector for the purpose of
51        autonomous creation of the service flows. This value is
52        used when the object wmanIfBsAutoSfidEnabled allows
53        autonomous creation of SFIDs."
54
55     REFERENCE
56         "Subclause 11.13.1 in IEEE Std 802.16-2004"
57         ::= { wmanIfBsConfigurationEntry 19 }
58
59
60
61 wmanIfBsAasChanFbckReqFreq OBJECT-TYPE
62     SYNTAX      INTEGER (5..10000)
63     UNITS        "ms"
64     MAX-ACCESS  read-write
65

```

```

1      STATUS      current
2      DESCRIPTION
3          "This object defines AAS channel feedback request frequency.
4          It controls the frequency of downlink beam measurements.
5          The relevant MAC messages are AAS-FBCK-REQ/RSP"
6      REFERENCE
7          "Subclause 6.3.2.3.40 in IEEE Std 802.16-2004"
8      ::= { wmanIfBsConfigurationEntry 20 }
9
10
11
12      wmanIfBsAasBeamSelectFreq OBJECT-TYPE
13          SYNTAX      INTEGER (5..10000)
14          UNITS        "ms"
15          MAX-ACCESS   read-write
16          STATUS       current
17          DESCRIPTION
18              "This object defines AAS beam select frequency.
19              It controls how often SS issues beam select messages.
20              The relevant MAC message is AAS_Beam_Select"
21          REFERENCE
22              "Subclause 6.3.2.3.41 in IEEE Std 802.16-2004"
23          ::= { wmanIfBsConfigurationEntry 21 }
24
25
26
27      wmanIfBsAasChanFbckReqResolution OBJECT-TYPE
28          SYNTAX      INTEGER {aasChanFbckRes00(0),
29                               aasChanFbckRes01(1),
30                               aasChanFbckRes10(2),
31                               aasChanFbckRes11(3)}
32          MAX-ACCESS   read-write
33          STATUS       current
34          DESCRIPTION
35              "This object defines AAS feedback request frequency
36              measurements resolution. It is coded as follows:
37              aasChanFbckRes00 - every 4th carrier
38                               (-100, -96, -92, ..., 100)
39              aasChanFbckRes01 - every 8th carrier
40                               (-100, -92, -84, ..., 100)
41              aasChanFbckRes10 - every 16th carrier
42                               (-100, -84, -68, ..., 100)
43              aasChanFbckRes11 - every 32th carrier
44                               (-100, -68, -36, ..., 100)"
45          REFERENCE
46              "Subclause 8.3.6.4 in IEEE Std 802.16-2004"
47          ::= { wmanIfBsConfigurationEntry 22 }
48
49
50
51
52
53
54      wmanIfBsAasBeamReqResolution OBJECT-TYPE
55          SYNTAX      INTEGER {aasBeamReqRes000(0),
56                               aasBeamReqRes001(1),
57                               aasBeamReqRes010(2),
58                               aasBeamReqRes011(3),
59                               aasBeamReqRes100(4)}
60          MAX-ACCESS   read-write
61          STATUS       current
62          DESCRIPTION
63              "This object defines AAS beam select request resolution
64
65

```

```

1         parameter. It is coded as follows:
2         aasBeamReqRes000 - every 4th carrier
3         aasBeamReqRes001 - every 8th carrier
4         aasBeamReqRes010 - every 16th carrier
5         aasBeamReqRes011 - every 32th carrier
6         aasBeamReqRes100 - every 64th carrier"
7
8     REFERENCE
9
10        "Subclause 8.3.6.5 in IEEE Std 802.16-2004"
11        ::= { wmanIfBsConfigurationEntry 23 }
12
13 wmanIfBsAasNumOptDiversityZones OBJECT-TYPE
14     SYNTAX      INTEGER (0..65535)
15     MAX-ACCESS  read-write
16     STATUS      current
17     DESCRIPTION
18         "This object defines the number of optional diversity zones
19         transmitted in downlink."
20     REFERENCE
21
22        "Figure 209 in IEEE Std 802.16-2004"
23        ::= { wmanIfBsConfigurationEntry 24 }
24
25
26 wmanIfBsResetSector OBJECT-TYPE
27     SYNTAX      INTEGER {actionResetSectorNoAction(0),
28                        actionResetSector(1)}
29     MAX-ACCESS  read-write
30     STATUS      current
31     DESCRIPTION
32         "This object should be implemented as follows:
33         - When set to actionsResetSector value, instructs BS to
34         reset the sector identified by ifIndex. As a result of
35         this action the Phy and Mac of this sector should be
36         reinitialised.
37         - When set to value different than actionsResetSector it
38         should be ignored
39         - When read it should return actionsResetSectorNoAction"
40     ::= { wmanIfBsConfigurationEntry 25 }
41
42
43
44
45 --
46 -- Base Station Channel Measurement Table
47 --
48
49 wmanIfBsChannelMeasurementTable OBJECT-TYPE
50     SYNTAX      SEQUENCE OF WmanIfBsChannelMeasurementEntry
51     MAX-ACCESS  not-accessible
52     STATUS      current
53     DESCRIPTION
54         "This table contains channel measurement information as
55         derived from BS measurement of uplink signal from SS,
56         and the downlink signal as reported from SS using
57         REP-REQ/RSP messages. The table shall be maintained as
58         FIFO to store measurement samples that can be used to
59         create RSSI and CINR histogram report. When the
60         measurement entry for a SS reaches the limit, the oldest
61         entry shall be deleted as the new entry is added to the
62         table."
63
64
65

```

```

1      REFERENCE
2          "6.3.2.3.33 in IEEE Std 802.16-2004"
3      ::= { wmanIfBsCps 3 }
4
5
6      wmanIfBsChannelMeasurementEntry OBJECT-TYPE
7          SYNTAX      WmanIfBsChannelMeasurementEntry
8          MAX-ACCESS   not-accessible
9          STATUS       current
10         DESCRIPTION
11             "Each entry in the table contains RSSI and CINR
12             signal quality measurement on signal received from the SS.
13             The primary index is the ifIndex with ifType of propBWA2Mp
14             identifying the BS sector. wmanIfBsSsMacAddress identifies
15             the SS from which the signal was received.
16             wmanIfBsChannelDirection is the index to the direction of
17             the channel. wmanIfBsHistogramIndex is the index to
18             histogram samples. Since there is no time stamp in the
19             table, wmanIfBsHistogramIndex should be increased
20             monotonically, and wraps around when it reaches the
21             implementation specific limit."
22         INDEX          { ifIndex,
23                         wmanIfBsSsMacAddress,
24                         wmanIfBsChannelDirection,
25                         wmanIfBsHistogramIndex }
26         ::= { wmanIfBsChannelMeasurementTable 1 }
27
28
29      WmanIfBsChannelMeasurementEntry ::= SEQUENCE {
30          wmanIfBsChannelDirection      INTEGER,
31          wmanIfBsHistogramIndex         Unsigned32,
32          wmanIfBsChannelNumber          WmanIfChannelNumber,
33          wmanIfBsStartFrame             INTEGER,
34          wmanIfBsDuration               INTEGER,
35          wmanIfBsBasicReport            BITS,
36          wmanIfBsMeanCinrReport         INTEGER,
37          wmanIfBsMeanRssiReport         INTEGER,
38          wmanIfBsStdDeviationCinrReport INTEGER,
39          wmanIfBsStdDeviationRssiReport INTEGER}
40
41
42      wmanIfBsChannelDirection OBJECT-TYPE
43          SYNTAX      INTEGER {downstream(1),
44                               upstream(2)}
45          MAX-ACCESS   not-accessible
46          STATUS       current
47          DESCRIPTION
48              "wmanIfBsChannelDirection identifies the direction of a
49              a channel where the measurement takes place."
50          ::= { wmanIfBsChannelMeasurementEntry 1 }
51
52
53      wmanIfBsHistogramIndex OBJECT-TYPE
54          SYNTAX      Unsigned32 (1 .. 4294967295)
55          MAX-ACCESS   read-only
56          STATUS       current
57          DESCRIPTION
58              "wmanIfBsHistogramIndex identifies the histogram samples

```

```

1         in the table for each subscriber station."
2         ::= { wmanIfBsChannelMeasurementEntry 2 }
3
4
5 wmanIfBsChannelNumber OBJECT-TYPE
6     SYNTAX      WmanIfChannelNumber
7     MAX-ACCESS  read-only
8     STATUS      current
9     DESCRIPTION
10        "Physical channel number to be reported on is only
11        applicable to license exempt band. For licensed band,
12        this parameter should be null."
13
14     REFERENCE
15        "Subclause 11.12 in IEEE Std 802.16-2004"
16     ::= { wmanIfBsChannelMeasurementEntry 3 }
17
18
19 wmanIfBsStartFrame OBJECT-TYPE
20     SYNTAX      INTEGER (0..65535)
21     MAX-ACCESS  read-only
22     STATUS      current
23     DESCRIPTION
24        "Frame number in which measurement for this channel
25        started."
26
27     REFERENCE
28        "Subclause 11.12 in IEEE Std 802.16-2004"
29     ::= { wmanIfBsChannelMeasurementEntry 4 }
30
31
32 wmanIfBsDuration OBJECT-TYPE
33     SYNTAX      INTEGER (0 .. 16777215)
34     MAX-ACCESS  read-only
35     STATUS      current
36     DESCRIPTION
37        "Cumulative measurement duration on the channel in
38        multiples of Ts. For any value exceeding 0xFFFFFFFF,
39        report 0xFFFFFFFF."
40
41     REFERENCE
42        "Subclause 11.12 in IEEE Std 802.16-2004"
43     ::= { wmanIfBsChannelMeasurementEntry 5 }
44
45
46 wmanIfBsBasicReport OBJECT-TYPE
47     SYNTAX      BITS {wirelessHuman(0),
48                      unknownTransmission(1),
49                      primaryUser(2),
50                      channelNotMeasured(3)}
51
52     MAX-ACCESS  read-only
53     STATUS      current
54     DESCRIPTION
55        "Bit #0: WirelessHUMAN detected on the channel
56        Bit #1: Unknown transmissions detected on the channel
57        Bit #2: Primary User detected on the channel
58        Bit #3: Unmeasured. Channel not measured"
59
60     REFERENCE
61        "Subclause 11.12 in IEEE Std 802.16-2004"
62     ::= { wmanIfBsChannelMeasurementEntry 6 }
63
64
65

```

```

1  wmanIfBsMeanCinrReport OBJECT-TYPE
2      SYNTAX      INTEGER (0 .. 41)
3      UNITS       "dB"
4      MAX-ACCESS  read-only
5      STATUS      current
6      DESCRIPTION
7          "Mean CINR report."
8
9      REFERENCE
10         "Subclause 8.2.2, 8.3.9 in IEEE Std 802.16-2004"
11         ::= { wmanIfBsChannelMeasurementEntry 7 }
12
13
14  wmanIfBsMeanRssiReport OBJECT-TYPE
15      SYNTAX      INTEGER (0 .. 83)
16      UNITS       "dBm"
17      MAX-ACCESS  read-only
18      STATUS      current
19      DESCRIPTION
20          "Mean RSSI report."
21
22      REFERENCE
23         "Subclause 8.2.2, 8.3.9 in IEEE Std 802.16-2004"
24         ::= { wmanIfBsChannelMeasurementEntry 8 }
25
26
27  wmanIfBsStdDeviationCinrReport OBJECT-TYPE
28      SYNTAX      INTEGER (0 .. 41)
29      UNITS       "dB"
30      MAX-ACCESS  read-only
31      STATUS      current
32      DESCRIPTION
33          "Standard deviation CINR report."
34
35      REFERENCE
36         "Subclause 8.2.2, 8.3.9 in IEEE Std 802.16-2004"
37         ::= { wmanIfBsChannelMeasurementEntry 9 }
38
39
40  wmanIfBsStdDeviationRssiReport OBJECT-TYPE
41      SYNTAX      INTEGER (0 .. 83)
42      UNITS       "dB"
43      MAX-ACCESS  read-only
44      STATUS      current
45      DESCRIPTION
46          "Standard deviation RSSI report."
47
48      REFERENCE
49         "Subclause 8.2.2, 8.3.9 in IEEE Std 802.16-2004"
50         ::= { wmanIfBsChannelMeasurementEntry 10 }
51
52
53  --
54  -- Base Station capabilities
55  --
56
57  wmanIfBsCapabilities OBJECT IDENTIFIER ::= { wmanIfBsCps 4 }
58
59
60  wmanIfBsSsReqCapabilitiesTable OBJECT-TYPE
61      SYNTAX      SEQUENCE OF WmanIfBsSsReqCapabilitiesEntry
62      MAX-ACCESS  not-accessible
63      STATUS      current
64      DESCRIPTION
65

```

```

1          "This table contains the basic capability information of SSs
2          that have been reported by SSs to BS using RNG-REQ, SBC-REQ
3          and REG-REQ messages. Entries in this table should be
4          created when an SS registers with a BS."
5      ::= { wmanIfBsCapabilities 1 }
6
7
8      wmanIfBsSsReqCapabilitiesEntry OBJECT-TYPE
9          SYNTAX      WmanIfBsSsReqCapabilitiesEntry
10         MAX-ACCESS  not-accessible
11         STATUS      current
12         DESCRIPTION
13             "This table provides one row for each SS that has been
14             registered in the BS. This table augments the table
15             wmanIfBsRegisteredSsTable."
16         AUGMENTS { wmanIfBsRegisteredSsEntry }
17         ::= { wmanIfBsSsReqCapabilitiesTable 1 }
18
19
20
21     WmanIfBsSsReqCapabilitiesEntry ::= SEQUENCE {
22         wmanIfBsSsReqCapUplinkCidSupport      WmanIfNumOfUplinkCid,
23         wmanIfBsSsReqCapArqSupport            WmanIfArqSupportType,
24         wmanIfBsSsReqCapDsxFlowControl        WmanIfMaxDsxFlowType,
25         wmanIfBsSsReqCapMacCrcSupport         WmanIfMacCrcSupport,
26         wmanIfBsSsReqCapMcaFlowControl        WmanIfMaxMcaFlowType,
27         wmanIfBsSsReqCapMcpGroupCidSupport    WmanIfMaxMcpGroupCid,
28         wmanIfBsSsReqCapPkmFlowControl        WmanIfMaxPkmFlowType,
29         wmanIfBsSsReqCapAuthPolicyControl     WmanIfAuthPolicyType,
30         wmanIfBsSsReqCapMaxNumOfSupportedSA   WmanIfMaxNumOfSaType,
31         wmanIfBsSsReqCapIpVersion             WmanIfIpVersionType,
32         wmanIfBsSsReqCapMacCsSupportBitMap    WmanIfMacCsBitMap,
33         wmanIfBsSsReqCapMaxNumOfClassifier    WmanIfMaxClassifiers,
34         wmanIfBsSsReqCapPhsSupport            WmanIfPhsSupportType,
35         wmanIfBsSsReqCapBandwidthAllocSupport WmanIfBwAllocSupport,
36         wmanIfBsSsReqCapPduConstruction      WmanIfPduConstruction,
37         wmanIfBsSsReqCapTtgTransitionGap      WmanIfSsTransitionGap,
38         wmanIfBsSsReqCapRtgTransitionGap      WmanIfSsTransitionGap}
39
40
41
42
43
44     wmanIfBsSsReqCapUplinkCidSupport OBJECT-TYPE
45         SYNTAX      WmanIfNumOfUplinkCid
46         MAX-ACCESS  read-only
47         STATUS      current
48         DESCRIPTION
49             "This object shows the number of Uplink CIDs the SS can
50             support."
51         ::= { wmanIfBsSsReqCapabilitiesEntry 1 }
52
53
54
55     wmanIfBsSsReqCapArqSupport OBJECT-TYPE
56         SYNTAX      WmanIfArqSupportType
57         MAX-ACCESS  read-only
58         STATUS      current
59         DESCRIPTION
60             "This object indicates whether the SS supports ARQ."
61         ::= { wmanIfBsSsReqCapabilitiesEntry 2 }
62
63
64
65     wmanIfBsSsReqCapDsxFlowControl OBJECT-TYPE

```



```

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

```

```

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

```

```

1      DESCRIPTION
2          "This object indicates indicates the level of SS support
3          for PHS."
4      DEFVAL      { noPhsSupport }
5      ::= { wmanIfBsSsReqCapabilitiesEntry 13 }
6
7
8      wmanIfBsSsReqCapBandwidthAllocSupport OBJECT-TYPE
9          SYNTAX      WmanIfBwAllocSupport
10         MAX-ACCESS  read-only
11         STATUS      current
12         DESCRIPTION
13             "This field indicates the bandwidth allocation
14             capabilities of the SS. The usage is defined by
15             WmanIfBwAllocSupport."
16             ::= { wmanIfBsSsReqCapabilitiesEntry 14 }
17
18
19
20     wmanIfBsSsReqCapPduConstruction OBJECT-TYPE
21         SYNTAX      WmanIfPduConstruction
22         MAX-ACCESS  read-only
23         STATUS      current
24         DESCRIPTION
25             "This field indicates the SS's capabilities for
26             construction and transmission of MAC PDUs. The usage
27             is defined by WmanIfPduConstruction."
28             ::= { wmanIfBsSsReqCapabilitiesEntry 15 }
29
30
31
32     wmanIfBsSsReqCapTtgTransitionGap OBJECT-TYPE
33         SYNTAX      WmanIfSsTransitionGap
34         UNITS        "us"
35         MAX-ACCESS  read-only
36         STATUS      current
37         DESCRIPTION
38             "This field indicates the SS's transition speed SSTTG
39             for TDD and H-FDD SSs. The usage is defined by
40             WmanIfSsTransitionGap."
41             ::= { wmanIfBsSsReqCapabilitiesEntry 16 }
42
43
44
45     wmanIfBsSsReqCapRtgTransitionGap OBJECT-TYPE
46         SYNTAX      WmanIfSsTransitionGap
47         UNITS        "us"
48         MAX-ACCESS  read-only
49         STATUS      current
50         DESCRIPTION
51             "This field indicates the SS's transition speed SSRTG
52             for TDD and H-FDD SSs. The usage is defined by
53             WmanIfSsTransitionGap."
54             ::= { wmanIfBsSsReqCapabilitiesEntry 17 }
55
56
57
58     wmanIfBsSsRspCapabilitiesTable OBJECT-TYPE
59         SYNTAX      SEQUENCE OF WmanIfBsSsRspCapabilitiesEntry
60         MAX-ACCESS  not-accessible
61         STATUS      current
62         DESCRIPTION
63             "This table contains the basic capability information of SSs
64
65

```

```

1         that have been negotiated and agreed between BS and SS via
2         RNG-REQ/RSP, SBC-REQ/RSP and REG-REQ/RSP messages.
3         This table augments the wmanIfBsRegisteredSsTable."
4
5     REFERENCE
6         "Subclause 6.3.2.3.7 in IEEE Std 802.16-2004"
7         ::= { wmanIfBsCapabilities 2 }
8
9
10    wmanIfBsSsRspCapabilitiesEntry OBJECT-TYPE
11        SYNTAX      WmanIfBsSsRspCapabilitiesEntry
12        MAX-ACCESS   not-accessible
13        STATUS      current
14        DESCRIPTION
15            "This table provides one row for each SS that has been
16             registered in the BS. This table augments the
17             wmanIfBsRegisteredSsTable. "
18        AUGMENTS { wmanIfBsRegisteredSsEntry }
19        ::= { wmanIfBsSsRspCapabilitiesTable 1 }
20
21
22
23    WmanIfBsSsRspCapabilitiesEntry ::= SEQUENCE {
24        wmanIfBsSsRspCapUplinkCidSupport      WmanIfNumOfUplinkCid,
25        wmanIfBsSsRspCapArqSupport            WmanIfArqSupportType,
26        wmanIfBsSsRspCapDsxFlowControl        WmanIfMaxDsxFlowType,
27        wmanIfBsSsRspCapMacCrcSupport         WmanIfMacCrcSupport,
28        wmanIfBsSsRspCapMcaFlowControl        WmanIfMaxMcaFlowType,
29        wmanIfBsSsRspCapMcpGroupCidSupport    WmanIfMaxMcpGroupCid,
30        wmanIfBsSsRspCapPkmFlowControl        WmanIfMaxPkmFlowType,
31        wmanIfBsSsRspCapAuthPolicyControl     WmanIfAuthPolicyType,
32        wmanIfBsSsRspCapMaxNumOfSupportedSA   WmanIfMaxNumOfSaType,
33        wmanIfBsSsRspCapIpVersion            WmanIfIpVersionType,
34        wmanIfBsSsRspCapMacCsSupportBitMap    WmanIfMacCsBitMap,
35        wmanIfBsSsRspCapMaxNumOfClassifier    WmanIfMaxClassifiers,
36        wmanIfBsSsRspCapPhsSupport            WmanIfPhsSupportType,
37        wmanIfBsSsRspCapBandwidthAllocSupport WmanIfBwAllocSupport,
38        wmanIfBsSsRspCapPduConstruction      WmanIfPduConstruction,
39        wmanIfBsSsRspCapTtgTransitionGap      WmanIfSsTransitionGap,
40        wmanIfBsSsRspCapRtgTransitionGap      WmanIfSsTransitionGap
41    }
42
43
44
45    wmanIfBsSsRspCapUplinkCidSupport OBJECT-TYPE
46        SYNTAX      WmanIfNumOfUplinkCid
47        MAX-ACCESS   read-only
48        STATUS      current
49        DESCRIPTION
50            "Negotiated number of Uplink CIDs the SS can support."
51        ::= { wmanIfBsSsRspCapabilitiesEntry 1 }
52
53
54
55    wmanIfBsSsRspCapArqSupport OBJECT-TYPE
56        SYNTAX      WmanIfArqSupportType
57        MAX-ACCESS   read-only
58        STATUS      current
59        DESCRIPTION
60            "This object indicates whether the SS is allowed to use ARQ
61             as a result of the capabilities negotiation."
62        ::= { wmanIfBsSsRspCapabilitiesEntry 2 }
63
64
65

```

```

1  wmanIfBsSsRspCapDsxFwControl OBJECT-TYPE
2      SYNTAX      WmanIfMaxDsxFwType
3      MAX-ACCESS  read-only
4      STATUS      current
5      DESCRIPTION
6          "Negotiated maximum number of concurrent DSA, DSC, or DSD
7           transactions that may be outstanding."
8      ::= { wmanIfBsSsRspCapabilitiesEntry 3 }
9
10
11  wmanIfBsSsRspCapMacCrcSupport OBJECT-TYPE
12      SYNTAX      WmanIfMacCrcSupport
13      MAX-ACCESS  read-only
14      STATUS      current
15      DESCRIPTION
16          "This object indicates whether or not the SS is allowed to
17           use MAC level CRC as a result of the capabilities
18           negotiation."
19      DEFVAL      { macCrcSupport }
20      ::= { wmanIfBsSsRspCapabilitiesEntry 4 }
21
22
23  wmanIfBsSsRspCapMcaFlowControl OBJECT-TYPE
24      SYNTAX      WmanIfMaxMcaFlowType
25      MAX-ACCESS  read-only
26      STATUS      current
27      DESCRIPTION
28          "Negotiated maximum number of concurrent
29           MCA transactions that may be outstanding."
30      DEFVAL      { 0 }
31      ::= { wmanIfBsSsRspCapabilitiesEntry 5 }
32
33
34  wmanIfBsSsRspCapMcpGroupCidSupport OBJECT-TYPE
35      SYNTAX      WmanIfMaxMcpGroupCid
36      MAX-ACCESS  read-only
37      STATUS      current
38      DESCRIPTION
39          "Negotiated maximum number of simultaneous Multicast
40           Polling Groups the SS is capable of belonging to."
41      DEFVAL      { 0 }
42      ::= { wmanIfBsSsRspCapabilitiesEntry 6 }
43
44
45  wmanIfBsSsRspCapPkmFlowControl OBJECT-TYPE
46      SYNTAX      WmanIfMaxPkmFlowType
47      MAX-ACCESS  read-only
48      STATUS      current
49      DESCRIPTION
50          "Negotiated maximum number of concurrent PKM
51           transactions that may be outstanding."
52      DEFVAL      { 0 }
53      ::= { wmanIfBsSsRspCapabilitiesEntry 7 }
54
55
56  wmanIfBsSsRspCapAuthPolicyControl OBJECT-TYPE
57      SYNTAX      WmanIfAuthPolicyType
58      MAX-ACCESS  read-only
59      STATUS      current
60

```

```

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

```

```

1         "This object indicates the negotiated level of PHS
2         support."
3     DEFVAL      { noPhsSupport }
4     ::= { wmanIfBsSsRspCapabilitiesEntry 13 }
5
6
7     wmanIfBsSsRspCapBandwidthAllocSupport OBJECT-TYPE
8         SYNTAX      WmanIfBwAllocSupport
9         MAX-ACCESS   read-only
10        STATUS      current
11        DESCRIPTION
12            "This field indicates negotiated properties of the SS
13            for bandwidth allocation purposes. The usage is defined
14            by WmanIfBwAllocSupport."
15        ::= { wmanIfBsSsRspCapabilitiesEntry 14 }
16
17
18
19     wmanIfBsSsRspCapPduConstruction OBJECT-TYPE
20         SYNTAX      WmanIfPduConstruction
21         MAX-ACCESS   read-only
22         STATUS      current
23         DESCRIPTION
24             "Specifies negotiated capabilities for construction and
25             transmission of MAC PDUs. The usage is defined by
26             WmanIfPduConstruction."
27         ::= { wmanIfBsSsRspCapabilitiesEntry 15 }
28
29
30
31     wmanIfBsSsRspCapTtgTransitionGap OBJECT-TYPE
32         SYNTAX      WmanIfSsTransitionGap
33         UNITS        "us"
34         MAX-ACCESS   read-only
35         STATUS      current
36         DESCRIPTION
37             "This field indicates the negotiated transition speed
38             SSTTG for TDD and H-FDD SSs. The usage is defined by
39             WmanIfSsTransitionGap."
40         ::= { wmanIfBsSsRspCapabilitiesEntry 16 }
41
42
43
44     wmanIfBsSsRspCapRtgTransitionGap OBJECT-TYPE
45         SYNTAX      WmanIfSsTransitionGap
46         UNITS        "us"
47         MAX-ACCESS   read-only
48         STATUS      current
49         DESCRIPTION
50             "This field indicates the negotiated transition speed
51             SSRTG for TDD and H-FDD SSs. The usage is defined by
52             WmanIfSsTransitionGap."
53         ::= { wmanIfBsSsRspCapabilitiesEntry 17 }
54
55
56
57     wmanIfBsBasicCapabilitiesTable OBJECT-TYPE
58         SYNTAX      SEQUENCE OF WmanIfBsBasicCapabilitiesEntry
59         MAX-ACCESS   not-accessible
60         STATUS      current
61         DESCRIPTION
62             "This table contains the basic capabilities of the BS as
63             implemented in BS hardware and software. These capabilities
64
65

```

```

1         along with the configuration for them
2         (wmanIfBsCapabilitiesConfigTable) are used for negotiation
3         of basic capabilities with SS using RNG-RSP, SBC-RSP and
4         REG-RSP messages. The negotiated capabilities are obtained
5         by interSubclause of SS raw reported capabilities, BS raw
6         capabilities and BS configured capabilities. The objects in
7         the table have read-only access. The table is maintained
8         by BS."
9
10        ::= { wmanIfBsCapabilities 3 }
11
12
13    wmanIfBsBasicCapabilitiesEntry OBJECT-TYPE
14        SYNTAX      WmanIfBsBasicCapabilitiesEntry
15        MAX-ACCESS   not-accessible
16        STATUS      current
17        DESCRIPTION
18            "This table provides one row for each BS sector and is
19             indexed by ifIndex."
20        INDEX { ifIndex }
21        ::= { wmanIfBsBasicCapabilitiesTable 1 }
22
23
24
25    WmanIfBsBasicCapabilitiesEntry ::= SEQUENCE {
26        wmanIfBsCapUplinkCidSupport      WmanIfNumOfUplinkCid,
27        wmanIfBsCapArqSupport             WmanIfArqSupportType,
28        wmanIfBsCapDsxFLOWControl         WmanIfMaxDsxFLOWType,
29        wmanIfBsCapMacCrcSupport          WmanIfMacCrcSupport,
30        wmanIfBsCapMcaFLOWControl         WmanIfMaxMcaFLOWType,
31        wmanIfBsCapMcpGroupCidSupport     WmanIfMaxMcpGroupCid,
32        wmanIfBsCapPkmFLOWControl         WmanIfMaxPkmFLOWType,
33        wmanIfBsCapAuthPolicyControl      WmanIfAuthPolicyType,
34        wmanIfBsCapMaxNumOfSupportedSA    WmanIfMaxNumOfSaType,
35        wmanIfBsCapIpVersion              WmanIfIpVersionType,
36        wmanIfBsCapMacCsSupportBitMap     WmanIfMacCsBitMap,
37        wmanIfBsCapMaxNumOfClassifier     WmanIfMaxClassifiers,
38        wmanIfBsCapPhsSupport             WmanIfPhsSupportType,
39        wmanIfBsCapBandwidthAllocSupport  WmanIfBwAllocSupport,
40        wmanIfBsCapPduConstruction        WmanIfPduConstruction,
41        wmanIfBsCapTtgTransitionGap       WmanIfSsTransitionGap,
42        wmanIfBsCapRtgTransitionGap       WmanIfSsTransitionGap}
43
44
45
46
47
48    wmanIfBsCapUplinkCidSupport OBJECT-TYPE
49        SYNTAX      WmanIfNumOfUplinkCid
50        MAX-ACCESS   read-only
51        STATUS      current
52        DESCRIPTION
53            "This object shows the number of Uplink CIDs the BS can
54             support per SS."
55        ::= { wmanIfBsBasicCapabilitiesEntry 1 }
56
57
58
59    wmanIfBsCapArqSupport OBJECT-TYPE
60        SYNTAX      WmanIfArqSupportType
61        MAX-ACCESS   read-only
62        STATUS      current
63        DESCRIPTION
64            "This object indicates whether the BS supports ARQ."
65

```



```

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

```

```

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

```

```

1
2 wmanIfBsCapPhsSupport OBJECT-TYPE
3     SYNTAX      WmanIfPhsSupportType
4     MAX-ACCESS  read-only
5     STATUS      current
6     DESCRIPTION
7         "This object indicates the level of BS support for PHS.
8         The usage is defined by WmanIfPhsSupportType."
9     DEFVAL      { noPhsSupport }
10    ::= { wmanIfBsBasicCapabilitiesEntry 13 }
11
12
13
14 wmanIfBsCapBandwidthAllocSupport OBJECT-TYPE
15     SYNTAX      WmanIfBwAllocSupport
16     MAX-ACCESS  read-only
17     STATUS      current
18     DESCRIPTION
19         "This field indicates the bandwidth allocation properties
20         that the BS permits SSs to use. The usage is defined by
21         WmanIfBwAllocSupport."
22     ::= { wmanIfBsBasicCapabilitiesEntry 14 }
23
24
25
26 wmanIfBsCapPduConstruction OBJECT-TYPE
27     SYNTAX      WmanIfPduConstruction
28     MAX-ACCESS  read-only
29     STATUS      current
30     DESCRIPTION
31         "Specifies the capabilities for construction and
32         transmission of MAC PDUs allowed by the BS. The usage is
33         defined by WmanIfPduConstruction."
34     ::= { wmanIfBsBasicCapabilitiesEntry 15 }
35
36
37
38 wmanIfBsCapTtgTransitionGap OBJECT-TYPE
39     SYNTAX      WmanIfSsTransitionGap
40     UNITS        "us"
41     MAX-ACCESS  read-only
42     STATUS      current
43     DESCRIPTION
44         "This field indicates the transition speed SSTTG for TDD
45         and H-FDD SSs allowed by the BS. The usage is defined by
46         WmanIfSsTransitionGap."
47     ::= { wmanIfBsBasicCapabilitiesEntry 16 }
48
49
50
51 wmanIfBsCapRtgTransitionGap OBJECT-TYPE
52     SYNTAX      WmanIfSsTransitionGap
53     UNITS        "us"
54     MAX-ACCESS  read-only
55     STATUS      current
56     DESCRIPTION
57         "This field indicates the transition speed SSRTG for TDD
58         and H-FDD SSs allowed by the BS. The usage is defined
59         by WmanIfSsTransitionGap."
60     ::= { wmanIfBsBasicCapabilitiesEntry 17 }
61
62
63
64 wmanIfBsCapabilitiesConfigTable OBJECT-TYPE
65

```

```

1      SYNTAX      SEQUENCE OF WmanIfBsCapabilitiesConfigEntry
2      MAX-ACCESS  not-accessible
3      STATUS      current
4      DESCRIPTION
5
6          "This table contains the configuration for basic
7          capabilities of BS. The table is intended to be used to
8          restrict the Capabilities implemented by BS, for example in
9          order to comply with local regulatory requirements. The BS
10         should use the configuration along with the implemented
11         Capabilities (wmanIfBsBasicCapabilitiesTable) for
12         negotiation of basic capabilities with SS using RNG-RSP,
13         SBC-RSP and REG-RSP messages. The negotiated capabilities
14         are obtained by interSubclause of SS reported capabilities,
15         BS raw capabilities and BS configured capabilities. The
16         objects in the table have read-write access. The rows are
17         created by BS as a copy of wmanIfBsBasicCapabilitiesTable
18         and can be modified by NMS."
19
20         ::= { wmanIfBsCapabilities 4 }
21
22
23
24     wmanIfBsCapabilitiesConfigEntry OBJECT-TYPE
25         SYNTAX      WmanIfBsCapabilitiesConfigEntry
26         MAX-ACCESS  not-accessible
27         STATUS      current
28         DESCRIPTION
29
30             "This table provides one row for each BS sector and is
31             indexed by ifIndex."
32         INDEX { ifIndex }
33         ::= { wmanIfBsCapabilitiesConfigTable 1 }
34
35
36     WmanIfBsCapabilitiesConfigEntry ::= SEQUENCE {
37         wmanIfBsCapCfgUplinkCidSupport      WmanIfNumOfUplinkCid,
38         wmanIfBsCapCfgArqSupport            WmanIfArqSupportType,
39         wmanIfBsCapCfgDsxFlowControl        WmanIfMaxDsxFlowType,
40         wmanIfBsCapCfgMacCrcSupport         WmanIfMacCrcSupport,
41         wmanIfBsCapCfgMcaFlowControl        WmanIfMaxMcaFlowType,
42         wmanIfBsCapCfgMcpGroupCidSupport    WmanIfMaxMcpGroupCid,
43         wmanIfBsCapCfgPkmFlowControl        WmanIfMaxPkmFlowType,
44         wmanIfBsCapCfgAuthPolicyControl     WmanIfAuthPolicyType,
45         wmanIfBsCapCfgMaxNumOfSupportedSA   WmanIfMaxNumOfSaType,
46         wmanIfBsCapCfgIpVersion            WmanIfIpVersionType,
47         wmanIfBsCapCfgMacCsSupportBitMap    WmanIfMacCsBitMap,
48         wmanIfBsCapCfgMaxNumOfClassifier    WmanIfMaxClassifiers,
49         wmanIfBsCapCfgPhsSupport            WmanIfPhsSupportType,
50         wmanIfBsCapCfgBandwidthAllocSupport WmanIfBwAllocSupport,
51         wmanIfBsCapCfgPduConstruction       WmanIfPduConstruction,
52         wmanIfBsCapCfgTtgTransitionGap      WmanIfSsTransitionGap,
53         wmanIfBsCapCfgRtgTransitionGap      WmanIfSsTransitionGap}
54
55
56
57
58
59     wmanIfBsCapCfgUplinkCidSupport OBJECT-TYPE
60         SYNTAX      WmanIfNumOfUplinkCid
61         MAX-ACCESS  read-write
62         STATUS      current
63         DESCRIPTION
64
65             "This object shows the configured number of Uplink CIDs the

```

```

1         BS can support per SS."
2         ::= { wmanIfBsCapabilitiesConfigEntry 1 }
3
4
5 wmanIfBsCapCfgArqSupport OBJECT-TYPE
6     SYNTAX      WmanIfArqSupportType
7     MAX-ACCESS  read-write
8     STATUS      current
9     DESCRIPTION
10        "This object indicates whether the BS is configured to
11        support ARQ."
12        ::= { wmanIfBsCapabilitiesConfigEntry 2 }
13
14
15 wmanIfBsCapCfgDsxFowControl OBJECT-TYPE
16     SYNTAX      WmanIfMaxDsxFowType
17     MAX-ACCESS  read-write
18     STATUS      current
19     DESCRIPTION
20        "This object specifies the configured maximum number of
21        concurrent DSA, DSC, or DSD transactions that BS allows
22        each SS to have outstanding."
23        DEFVAL   { 0 }
24        ::= { wmanIfBsCapabilitiesConfigEntry 3 }
25
26
27 wmanIfBsCapCfgMacCrcSupport OBJECT-TYPE
28     SYNTAX      WmanIfMacCrcSupport
29     MAX-ACCESS  read-write
30     STATUS      current
31     DESCRIPTION
32        "This object indicates whether BS is configured to support
33        MAC level CRC."
34        DEFVAL   { macCrcSupport }
35        ::= { wmanIfBsCapabilitiesConfigEntry 4 }
36
37
38 wmanIfBsCapCfgMcaFlowControl OBJECT-TYPE
39     SYNTAX      WmanIfMaxMcaFlowType
40     MAX-ACCESS  read-write
41     STATUS      current
42     DESCRIPTION
43        "This object specifies the maximum number of concurrent
44        MCA transactions that BS is configured to allow each SS to
45        have."
46        DEFVAL   { 0 }
47        ::= { wmanIfBsCapabilitiesConfigEntry 5 }
48
49
50 wmanIfBsCapCfgMcpGroupCidSupport OBJECT-TYPE
51     SYNTAX      WmanIfMaxMcpGroupCid
52     MAX-ACCESS  read-write
53     STATUS      current
54     DESCRIPTION
55        "This object indicates the maximum number of simultaneous
56        Multicast Polling Groups the BS is configured to allow
57        each SS to belong to."
58        DEFVAL   { 0 }
59        ::= { wmanIfBsCapabilitiesConfigEntry 6 }
60
61
62
63
64
65

```

```

1
2 wmanIfBsCapCfgPkmFlowControl OBJECT-TYPE
3     SYNTAX      WmanIfMaxPkmFlowType
4     MAX-ACCESS  read-write
5     STATUS      current
6     DESCRIPTION
7         "This object specifies the maximum number of concurrent
8         PKM transactions that BS is configured to allow each SS
9         to have."
10    DEFVAL      { 0 }
11    ::= { wmanIfBsCapabilitiesConfigEntry 7 }
12
13 wmanIfBsCapCfgAuthPolicyControl OBJECT-TYPE
14     SYNTAX      WmanIfAuthPolicyType
15     MAX-ACCESS  read-write
16     STATUS      current
17     DESCRIPTION
18         "This object specifies authorization policy that BS is
19         configured to be capable of. A bit value of 0 = not
20         supported, 1 = upported. If this field is omitted, then
21         both SS and BS shall use the IEEE 802.16 security,
22         constituting X.509 digital certificates and the RSA
23         public key encryption algorithm, as authorization policy."
24    ::= { wmanIfBsCapabilitiesConfigEntry 8 }
25
26 wmanIfBsCapCfgMaxNumOfSupportedSA OBJECT-TYPE
27     SYNTAX      WmanIfMaxNumOfSaType
28     MAX-ACCESS  read-write
29     STATUS      current
30     DESCRIPTION
31         "This field specifies configured maximum number of supported
32         security association per SS."
33    DEFVAL      { 1 }
34    ::= { wmanIfBsCapabilitiesConfigEntry 9 }
35
36 wmanIfBsCapCfgIpVersion OBJECT-TYPE
37     SYNTAX      WmanIfIpVersionType
38     MAX-ACCESS  read-write
39     STATUS      current
40     DESCRIPTION
41         "This object indicates the configured version of IP that the
42         BS allows each SS to use on the 2nd Management Connection.
43         The value 'undefined' should not be used in this field."
44    ::= { wmanIfBsCapabilitiesConfigEntry 10 }
45
46 wmanIfBsCapCfgMacCsSupportBitMap OBJECT-TYPE
47     SYNTAX      WmanIfMacCsBitMap
48     MAX-ACCESS  read-write
49     STATUS      current
50     DESCRIPTION
51         "This object indicates BS configured set of MAC convergence
52         sublayer support. When a bit is set, it indicates
53         the corresponding CS feature is supported."
54    ::= { wmanIfBsCapabilitiesConfigEntry 11 }

```

```

1
2 wmanIfBsCapCfgMaxNumOfClassifier OBJECT-TYPE
3     SYNTAX      WmanIfMaxClassifiers
4     MAX-ACCESS  read-write
5     STATUS      current
6     DESCRIPTION
7         "This object indicates the configured maximum number of
8         admitted Classifiers per SS that the BS can support."
9     DEFVAL      { 0 }
10    ::= { wmanIfBsCapabilitiesConfigEntry 12 }
11
12
13
14 wmanIfBsCapCfgPhsSupport OBJECT-TYPE
15     SYNTAX      WmanIfPhsSupportType
16     MAX-ACCESS  read-write
17     STATUS      current
18     DESCRIPTION
19         "This object indicates the configured level of BS support
20         for PHS."
21     DEFVAL      { noPhsSupport }
22    ::= { wmanIfBsCapabilitiesConfigEntry 13 }
23
24
25
26 wmanIfBsCapCfgBandwidthAllocSupport OBJECT-TYPE
27     SYNTAX      WmanIfBwAllocSupport
28     MAX-ACCESS  read-write
29     STATUS      current
30     DESCRIPTION
31         "This field indicates configured properties of the BS for
32         bandwidth allocation purposes. The usage is defined by
33         WmanIfCapBwAllocSupport."
34     ::= { wmanIfBsCapabilitiesConfigEntry 14 }
35
36
37
38 wmanIfBsCapCfgPduConstruction OBJECT-TYPE
39     SYNTAX      WmanIfPduConstruction
40     MAX-ACCESS  read-write
41     STATUS      current
42     DESCRIPTION
43         "Specifies configured capabilities for construction and
44         transmission of MAC PDUs. The usage is defined by
45         WmanIfPduConstruction."
46     ::= { wmanIfBsCapabilitiesConfigEntry 15 }
47
48
49
50 wmanIfBsCapCfgTtgTransitionGap OBJECT-TYPE
51     SYNTAX      WmanIfSsTransitionGap
52     UNITS        "us"
53     MAX-ACCESS  read-write
54     STATUS      current
55     DESCRIPTION
56         "This field indicates the configured transition speed
57         SSTTG for TDD and H-FDD SSs. The usage is defined by
58         WmanIfSsTransitionGap."
59     ::= { wmanIfBsCapabilitiesConfigEntry 16 }
60
61
62
63 wmanIfBsCapCfgRtgTransitionGap OBJECT-TYPE
64     SYNTAX      WmanIfSsTransitionGap
65

```

```

1      UNITS          "us"
2      MAX-ACCESS     read-write
3      STATUS         current
4      DESCRIPTION
5
6          "This field indicates the configured transition speed
7          SSRTG for TDD and H-FDD SSs. The usage is defined by
8          WmanIfSsTransitionGap."
9
10     ::= { wmanIfBsCapabilitiesConfigEntry 17 }
11
12 wmanIfBsSsActionsTable OBJECT-TYPE
13     SYNTAX          SEQUENCE OF WmanIfBsSsActionsEntry
14     MAX-ACCESS     not-accessible
15     STATUS         current
16     DESCRIPTION
17
18         "This table contains all the actions specified for SSs in
19         the standard. The actions are routed down to SS using
20         unsolicited MAC messages: REG-RSP, DREG-REQ and RES-CMD.
21         The table also contains the parameters of the actions in
22         cases where they are specified by the standard."
23
24     ::= { wmanIfBsCps 5 }
25
26 wmanIfBsSsActionsEntry OBJECT-TYPE
27     SYNTAX          WmanIfBsSsActionsEntry
28     MAX-ACCESS     not-accessible
29     STATUS         current
30     DESCRIPTION
31
32         "This table is indexed by wmanIfBsSsActionsMacAddress. The
33         action can be requested for SS in any state not only those
34         registered. However BS will decide whether the action is
35         applicable to the SS based on its current state and execute
36         it or skip it as defined in each action definition."
37
38     INDEX { wmanIfBsSsActionsMacAddress }
39     ::= { wmanIfBsSsActionsTable 1 }
40
41
42 WmanIfBsSsActionsEntry ::= SEQUENCE {
43     wmanIfBsSsActionsMacAddress      MacAddress,
44     wmanIfBsSsActionsResetSs         INTEGER,
45     wmanIfBsSsActionsAbortSs         INTEGER,
46     wmanIfBsSsActionsOverrideDnFreq  Unsigned32,
47     wmanIfBsSsActionsOverrideChannelId  INTEGER,
48     wmanIfBsSsActionsDeReRegSs       INTEGER,
49     wmanIfBsSsActionsDeReRegSsCode   INTEGER,
50     wmanIfBsSsActionsRowStatus       RowStatus}
51
52
53
54 wmanIfBsSsActionsMacAddress OBJECT-TYPE
55     SYNTAX          MacAddress
56     MAX-ACCESS     not-accessible
57     STATUS         current
58     DESCRIPTION
59
60         "This object uniquely identifies the SS as an action
61         target."
62
63     ::= { wmanIfBsSsActionsEntry 1 }
64
65 wmanIfBsSsActionsResetSs OBJECT-TYPE

```



```

1      SYNTAX      INTEGER {actionsResetSsNoAction(0),
2                      actionsResetSs(1)}
3
4      MAX-ACCESS   read-create
5      STATUS      current
6      DESCRIPTION
7          "This object should be implemented as follows:
8              - When set to actionsResetSs value, instructs BS to send
9                RES-CMD to SS
10             - When set to value different than actionsResetSs it
11               should be ignored
12             - When read it should return actionsResetSsNoAction
13             The RES-CMD message shall be transmitted by the BS on an
14             SS Basic CID to force the SS to reset itself,
15             reinitialize its MAC, and repeat initial system access."
16
17      REFERENCE
18          "Subclause 6.3.2.3.22 in IEEE Std 802.16-2004"
19      ::= { wmanIfBsSsActionsEntry 2 }
20
21
22
23      wmanIfBsSsActionsAbortSs OBJECT-TYPE
24          SYNTAX      INTEGER {actionsAbortSsNoAction(0),
25                      actionsAbortSs(1),
26                      actionAbortSsParams(2)}
27
28
29          MAX-ACCESS   read-create
30          STATUS      current
31          DESCRIPTION
32              "This object should be implemented as follows:
33                  - When set to actionsAbortSs value, it instructs BS to send
34                    unsolicited RNG-RSP with Ranging Status equal to 'abort'
35                    without override parameters
36                  - When set to actionAbortSsParams value, it instructs BS to
37                    send unsolicited RNG-RSP with Ranging Status equal to
38                    'abort' and with 'Downlink Frequency Override' and
39                    'Uplink Channel ID Override' parameters.
40                  - When set to any other value it should be ignored
41                  - When read it should returned actionsAbortSsNoAction"
42
43          REFERENCE
44              "Subclause 11.6, Table 365 in IEEE Std 802.16-2004"
45          ::= { wmanIfBsSsActionsEntry 3 }
46
47
48
49      wmanIfBsSsActionsOverrideDnFreq OBJECT-TYPE
50          SYNTAX      Unsigned32
51          UNITS       "kHz"
52          MAX-ACCESS   read-create
53          STATUS      current
54          DESCRIPTION
55              "This object is used as a parmeter of the AbortSs action
56              with the code actionAbortSsParams. It is used for licensed
57              bands only. It defines the Center frequency, in kHz, of
58              new downlink channel where the SS should redo initial
59              ranging."
60
61          REFERENCE
62              "Subclause 11.6, Table 365 in IEEE Std 802.16-2004"
63          ::= { wmanIfBsSsActionsEntry 4 }
64
65

```

```

1
2 wmanIfBsSsActionsOverrideChannelId OBJECT-TYPE
3     SYNTAX      INTEGER (0..199)
4     MAX-ACCESS  read-create
5     STATUS      current
6     DESCRIPTION
7         "This object is used as a parameter of the AbortSs action
8         with the code actionAbortSsParams. It is coded as follows:
9         - Licensed bands: The identifier of the uplink channel
10        with which the SS is to redo initial ranging (not used
11        with PHYs without channelized uplinks).
12        - License-exempt bands: The Channel Nr (see 8.5.1) where
13        the SS should redo initial ranging."
14
15 REFERENCE
16     "Subclause 11.6, Table 365 in IEEE Std 802.16-2004"
17 ::= { wmanIfBsSsActionsEntry 5 }
18
19
20
21 wmanIfBsSsActionsDeReRegSs OBJECT-TYPE
22     SYNTAX      INTEGER {actionsDeReRegSsNoAction(0),
23                        actionsDeReRegSs(1)}
24     MAX-ACCESS  read-create
25     STATUS      current
26     DESCRIPTION
27         "This object should be implemented as follows:
28         - When set to actionsDeReRegSs value, instructs BS to
29         send DREG-CMD to SS with specified action code
30         - When set to value different than actionsDeReRegSs it
31         should be ignored
32         - When read it should return actionsDeReRegSsNoAction
33         The DREG-CMD message shall be transmitted by the BS on an
34         SS Basic CID to force the SS to change its access state.
35         Upon receiving a DREG-CMD, the SS shall take the action
36         indicated by the action code defined by
37         wmanIfBsSsActionsDeReRegSsCode."
38
39 REFERENCE
40     "Subclause 6.3.2.3.26 in IEEE Std 802.16-2004"
41 ::= { wmanIfBsSsActionsEntry 6 }
42
43
44
45
46 wmanIfBsSsActionsDeReRegSsCode OBJECT-TYPE
47     SYNTAX      INTEGER {actionsDeReRegSsCodeChangeChan(0),
48                        actionsDeReRegSsCodeNoTransmit(1),
49                        actionsDeReRegSsCodeLtdTransmit(2),
50                        actionsDeReRegSsCodeResume(3)}
51     MAX-ACCESS  read-create
52     STATUS      current
53     DESCRIPTION
54         "This object defines the action code for
55         wmanIfBsSsActionsDeReRegSs action. The codes are defined
56         as follows:
57         actionsDeReRegSsCodeChangeChan - SS shall leave the
58         current channel and attempt to access another channel.
59         actionsDeReRegSsCodeNoTransmit - SS shall listen to the
60         current channel but shall not transmit until an
61         RES-CMD message or DREG_CMD with an Action Code that
62

```

```

1         allows transmission is received.
2         actionsDeReRegSsCodeLtdTransmit - SS shall listen to the
3         current channel but only transmit on the Basic,
4         Primary Management and 2nd Management Connections.
5         actionsDeReRegSsCodeResume - SS shall return to normal
6         operation and may transmit on any of its active
7         connections."
8
9     REFERENCE
10        "Subclause 6.3.2.3.26, Table 55 in IEEE Std 802.16-2004"
11    ::= { wmanIfBsSsActionsEntry 7 }
12
13
14    wmanIfBsSsActionsRowStatus OBJECT-TYPE
15        SYNTAX      RowStatus
16        MAX-ACCESS   read-create
17        STATUS       current
18        DESCRIPTION
19            "This object is used to ensure that the write operation to
20            multiple columns is guaranteed to be treated as atomic
21            operation by agent."
22    ::= { wmanIfBsSsActionsEntry 8 }
23
24
25
26    --
27    -- Base station PKM group
28    -- wmanIfBsPkmObjects contain the Base Station Privacy Sublayer objects
29    --
30    wmanIfBsPkmObjects OBJECT IDENTIFIER ::= { wmanIfBsObjects 3 }
31
32
33    --
34    -- Table wmanIfBsPkmBaseTable
35    --
36    wmanIfBsPkmBaseTable OBJECT-TYPE
37        SYNTAX      SEQUENCE OF WmanIfBsPkmBaseEntry
38        MAX-ACCESS   not-accessible
39        STATUS       current
40        DESCRIPTION
41            "This table describes the basic PKM attributes of each Base
42            Station wireless interface."
43    ::= { wmanIfBsPkmObjects 1 }
44
45
46
47    wmanIfBsPkmBaseEntry OBJECT-TYPE
48        SYNTAX      WmanIfBsPkmBaseEntry
49        MAX-ACCESS   not-accessible
50        STATUS       current
51        DESCRIPTION
52            "Each entry contains objects describing attributes of one
53            BS wireless interface."
54        INDEX       { ifIndex }
55    ::= { wmanIfBsPkmBaseTable 1 }
56
57
58
59    WmanIfBsPkmBaseEntry ::= SEQUENCE {
60        wmanIfBsPkmDefaultAuthLifetime      Integer32,
61        wmanIfBsPkmDefaultTekLifetime        Integer32,
62        wmanIfBsPkmDefaultSelfSigManufCertTrust INTEGER,
63        wmanIfBsPkmCheckCertValidityPeriods TruthValue,
64
65

```

```

1          wmanIfBsPkmAuthentInfos          Counter32,
2          wmanIfBsPkmAuthRequests          Counter32,
3          wmanIfBsPkmAuthReplies          Counter32,
4          wmanIfBsPkmAuthRejects          Counter32,
5          wmanIfBsPkmAuthInvalids          Counter32}
6
7
8
9  wmanIfBsPkmDefaultAuthLifetime OBJECT-TYPE
10     SYNTAX      Integer32 (86400..6048000)
11     UNITS       "seconds"
12     MAX-ACCESS  read-write
13     STATUS      current
14     DESCRIPTION
15         "The value of this object is the default lifetime, in
16         seconds, the BS assigns to a new authorization key."
17     REFERENCE
18         "Table 341 in IEEE Std 802.16-2004"
19     DEFVAL      { 604800 }
20     ::= { wmanIfBsPkmBaseEntry 1 }
21
22
23
24  wmanIfBsPkmDefaultTekLifetime OBJECT-TYPE
25     SYNTAX      Integer32 (1800..604800)
26     UNITS       "seconds"
27     MAX-ACCESS  read-write
28     STATUS      current
29     DESCRIPTION
30         "The value of this object is the default lifetime, in
31         seconds, the BS assigns to a new Traffic Encryption
32         Key (TEK)."
33     REFERENCE
34         "Table 341 in IEEE Std 802.16-2004"
35     DEFVAL      { 43200 }
36     ::= { wmanIfBsPkmBaseEntry 2 }
37
38
39
40
41
42  wmanIfBsPkmDefaultSelfSigManufCertTrust OBJECT-TYPE
43     SYNTAX      INTEGER {trusted (1),
44                      untrusted (2)}
45     MAX-ACCESS  read-write
46     STATUS      current
47     DESCRIPTION
48         "This object determines the default trust of all (new)
49         self-signed manufacturer certificates obtained after
50         setting the object."
51     ::= { wmanIfBsPkmBaseEntry 3 }
52
53
54
55  wmanIfBsPkmCheckCertValidityPeriods OBJECT-TYPE
56     SYNTAX      TruthValue
57     MAX-ACCESS  read-write
58     STATUS      current
59     DESCRIPTION
60         "Setting this object to TRUE causes all certificates
61         received thereafter to have their validity periods (and
62         their chain's validity periods) checked against the current
63         time of day. A FALSE setting will cause all certificates
64
65

```

```

1         received Thereafter to not have their validity periods
2         (nor their chain's validity periods) checked against the
3         current time of day."
4     ::= { wmanIfBsPkmBaseEntry 4 }
5
6
7 wmanIfBsPkmAuthentInfos OBJECT-TYPE
8     SYNTAX      Counter32
9     MAX-ACCESS  read-only
10    STATUS      current
11    DESCRIPTION
12        "The value of this object is the count of times the BS has
13        received an Authentication Information message from any
14        SS."
15    ::= { wmanIfBsPkmBaseEntry 5 }
16
17
18
19 wmanIfBsPkmAuthRequests OBJECT-TYPE
20     SYNTAX      Counter32
21     MAX-ACCESS  read-only
22     STATUS      current
23     DESCRIPTION
24         "The value of this object is the count of times the BS has
25         received an Authorization Request message from any SS"
26     ::= { wmanIfBsPkmBaseEntry 6 }
27
28
29
30 wmanIfBsPkmAuthReplies OBJECT-TYPE
31     SYNTAX      Counter32
32     MAX-ACCESS  read-only
33     STATUS      current
34     DESCRIPTION
35         "The value of this object is the count of times the BS has
36         transmitted an Authorization Reply message to any SS."
37     ::= { wmanIfBsPkmBaseEntry 7 }
38
39
40
41 wmanIfBsPkmAuthRejects 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 BS has
47         transmitted an Authorization Reject message to any SS."
48     ::= { wmanIfBsPkmBaseEntry 8 }
49
50
51
52 wmanIfBsPkmAuthInvalids OBJECT-TYPE
53     SYNTAX      Counter32
54     MAX-ACCESS  read-only
55     STATUS      current
56     DESCRIPTION
57         "The value of this object is the count of times the BS has
58         transmitted an Authorization Invalid message to any SS."
59     ::= { wmanIfBsPkmBaseEntry 9 }
60
61
62 --
63 -- Table wmanIfBsSsPkmAuthTable
64 --
65

```

```

1  wmanIfBsSsPkmAuthTable OBJECT-TYPE
2      SYNTAX      SEQUENCE OF WmanIfBsSsPkmAuthEntry
3      MAX-ACCESS  not-accessible
4      STATUS      current
5      DESCRIPTION
6          "This table describes PKM attributes related
7          to the authorization for each SS. The BS maintains one
8          Primary Security Association with each Baseline
9          Privacy-enabled SS on each BS wireless interface."
10         ::= { wmanIfBsPkmObjects 2 }
11
12
13
14  wmanIfBsSsPkmAuthEntry OBJECT-TYPE
15      SYNTAX      WmanIfBsSsPkmAuthEntry
16      MAX-ACCESS  not-accessible
17      STATUS      current
18      DESCRIPTION
19          "The BS MUST create one entry per SS per wireless
20          interface, based on the receipt of an Authorization
21          Request message and MUST not delete the entry before
22          the SS authorization permanently expires."
23      INDEX      { ifIndex, wmanIfBsSsPkmAuthMacAddress }
24      ::= { wmanIfBsSsPkmAuthTable 1 }
25
26
27
28
29  WmanIfBsSsPkmAuthEntry ::= SEQUENCE {
30      wmanIfBsSsPkmAuthMacAddress      MacAddress,
31      wmanIfBsSsPkmAuthKeySequenceNumber  Integer32,
32      wmanIfBsSsPkmAuthExpiresOld      DateAndTime,
33      wmanIfBsSsPkmAuthExpiresNew      DateAndTime,
34      wmanIfBsSsPkmAuthLifetime        Integer32,
35      wmanIfBsSsPkmAuthReset           INTEGER,
36      wmanIfBsSsPkmAuthInfos           Counter64,
37      wmanIfBsSsPkmAuthRequests        Counter64,
38      wmanIfBsSsPkmAuthReplies         Counter64,
39      wmanIfBsSsPkmAuthRejects         Counter64,
40      wmanIfBsSsPkmAuthInvalids        Counter64,
41      wmanIfBsSsPkmAuthRejectErrorCode  INTEGER,
42      wmanIfBsSsPkmAuthRejectErrorString SnmpAdminString,
43      wmanIfBsSsPkmAuthInvalidErrorCode INTEGER,
44      wmanIfBsSsPkmAuthInvalidErrorString SnmpAdminString,
45      wmanIfBsSsPkmAuthPrimarySAId     INTEGER,
46      wmanIfBsSsPkmAuthValidStatus     INTEGER}
47
48
49
50
51
52  wmanIfBsSsPkmAuthMacAddress OBJECT-TYPE
53      SYNTAX      MacAddress
54      MAX-ACCESS  not-accessible
55      STATUS      current
56      DESCRIPTION
57          "The value of this object is the physical address of the SS
58          to which the authorization association applies."
59      ::= { wmanIfBsSsPkmAuthEntry 1 }
60
61
62
63  wmanIfBsSsPkmAuthKeySequenceNumber OBJECT-TYPE
64      SYNTAX      Integer32 (0..15)
65      MAX-ACCESS  read-only

```

```

1      STATUS      current
2      DESCRIPTION
3          "The value of this object is the most recent authorization
4          key sequence number for this SS."
5      ::= { wmanIfBsSsPkmAuthEntry 2 }
6
7
8      wmanIfBsSsPkmAuthExpiresOld OBJECT-TYPE
9          SYNTAX      DateAndTime
10         MAX-ACCESS  read-only
11         STATUS      current
12         DESCRIPTION
13             "The value of this object is the actual clock time for
14             expiration of the immediate predecessor of the most recent
15             authorization key for this FSM. If this FSM has only one
16             authorization key, then the value is the time of activation
17             of this FSM."
18         ::= { wmanIfBsSsPkmAuthEntry 3 }
19
20
21
22      wmanIfBsSsPkmAuthExpiresNew OBJECT-TYPE
23          SYNTAX      DateAndTime
24          MAX-ACCESS  read-only
25          STATUS      current
26          DESCRIPTION
27              "The value of this object is the actual clock time for
28              expiration of the most recent authorization key for this
29              FSM"
30          ::= { wmanIfBsSsPkmAuthEntry 4 }
31
32
33
34      wmanIfBsSsPkmAuthLifetime OBJECT-TYPE
35          SYNTAX      Integer32 (86400..6048000)
36          UNITS        "seconds"
37          MAX-ACCESS  read-only
38          STATUS      current
39          DESCRIPTION
40              "The vaue of this object is the lifetime, in seconds, the
41              BS assigns to an authorization key for this SS."
42          REFERENCE
43              "Table 341 in IEEE Std 802.16-2004"
44          DEFVAL      { 604800 }
45          ::= { wmanIfBsSsPkmAuthEntry 5 }
46
47
48
49
50      wmanIfBsSsPkmAuthReset OBJECT-TYPE
51          SYNTAX      INTEGER {noResetRequested(1),
52                               invalidateAuth(2),
53                               sendAuthInvalid(3),
54                               invalidateTeks(4)}
55          MAX-ACCESS  read-write
56          STATUS      current
57          DESCRIPTION
58              "Setting this object to invalidateAuth(2) causes the BS to
59              invalidate the current SS authorization key(s), but not to
60              transmit an Authorization Invalid message nor to invalidate
61              unicast TEKs. Setting this object to sendAuthInvalid(3)
62              causes the BS to invalidate the current SS authorization
63
64
65

```

```

1         key(s), and to transmit an Authorization Invalid message to
2         the SS, but not to invalidate unicast TEKs. Setting this
3         object to invalidateTek(s) causes the BS to invalidate the
4         current SS authorization key(s), to transmit an
5         Authorization Invalid message to the SS, and to
6         invalidate all unicast TEKs associated with this SS
7         authorization. Reading this object returns the
8         most-recently-set value of this object, or returns
9         noResetRequested(1) if the object has not been set since
10        the last BS reboot."
11        ::= { wmanIfBsSsPkmAuthEntry 6 }
12
13        wmanIfBsSsPkmAuthInfos OBJECT-TYPE
14            SYNTAX      Counter64
15            MAX-ACCESS   read-only
16            STATUS       current
17            DESCRIPTION
18                "The value of this object is the count of times the BS has
19                received an Authentication Information message from this
20                SS."
21            ::= { wmanIfBsSsPkmAuthEntry 7 }
22
23        wmanIfBsSsPkmAuthRequests OBJECT-TYPE
24            SYNTAX      Counter64
25            MAX-ACCESS   read-only
26            STATUS       current
27            DESCRIPTION
28                "The value of this object is the count of times the BS has
29                received an Authorization Request message from this SS."
30            ::= { wmanIfBsSsPkmAuthEntry 8 }
31
32        wmanIfBsSsPkmAuthReplies OBJECT-TYPE
33            SYNTAX      Counter64
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                transmitted an Authorization Reply message to this SS."
39            ::= { wmanIfBsSsPkmAuthEntry 9 }
40
41        wmanIfBsSsPkmAuthRejects 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                transmitted an Authorization Reject message to this SS."
48            ::= { wmanIfBsSsPkmAuthEntry 10 }
49
50        wmanIfBsSsPkmAuthInvalids OBJECT-TYPE
51            SYNTAX      Counter64
52            MAX-ACCESS   read-only
53            STATUS       current
54            DESCRIPTION

```



```

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

```

```

1         message has been transmitted to the SS."
2     ::= { wmanIfBsSsPkmAuthEntry 15 }
3
4
5 wmanIfBsSsPkmAuthPrimarySAId OBJECT-TYPE
6     SYNTAX      INTEGER (0..65535)
7     MAX-ACCESS  read-only
8     STATUS      current
9     DESCRIPTION
10        "The value of this object is the Primary Security
11        Association identifier."
12
13     REFERENCE
14        "IEEE Std 802.16-2004; 11.9.7"
15     ::= { wmanIfBsSsPkmAuthEntry 16 }
16
17
18 wmanIfBsSsPkmAuthValidStatus OBJECT-TYPE
19     SYNTAX      INTEGER {unknown (0),
20                      validSsChained (1),
21                      validSsTrusted (2),
22                      invalidSsUntrusted (3),
23                      invalidCAUntrusted (4),
24                      invalidSsOther (5),
25                      invalidCAOther (6)}
26
27     MAX-ACCESS  read-only
28     STATUS      current
29     DESCRIPTION
30        "Contains the reason why a SS's certificate is deemed valid
31        or invalid. Return unknown if the SS is running PKM mode.
32        ValidSsChained means the certificate is valid because it
33        chains to a valid certificate. ValidSsTrusted means the
34        certificate is valid because it has been provisioned to be
35        trusted. InvalidSsUntrusted means the certificate is
36        invalid because it has been provisioned to be untrusted.
37        InvalidCAUntrusted means the certificate is invalid
38        because it chains to an untrusted certificate.
39        InvalidSsOther and InvalidCAOther refer to errors in
40        parsing, validity periods, etc, which are attributable to
41        the SS certificate or its chain respectively."
42     ::= { wmanIfBsSsPkmAuthEntry 17 }
43
44
45
46
47
48 --
49 -- Table wmanIfBsPkmTekTable
50 --
51
52 wmanIfBsPkmTekTable OBJECT-TYPE
53     SYNTAX      SEQUENCE OF WmanIfBsPkmTekEntry
54     MAX-ACCESS  not-accessible
55     STATUS      current
56     DESCRIPTION
57        "This table describes the attributes of each Traffic
58        Encryption Key (TEK) association. The BS maintains one TEK
59        association per SAID on each BS wireless interface."
60     ::= { wmanIfBsPkmObjects 3 }
61
62
63
64 wmanIfBsPkmTekEntry OBJECT-TYPE
65     SYNTAX      WmanIfBsPkmTekEntry

```

```

1      MAX-ACCESS    not-accessible
2      STATUS        current
3      DESCRIPTION
4          "Each entry contains objects describing attributes of one
5          TEK association on a particular BS wireless interface. The
6          BS MUST create one entry per SAID per wireless interface,
7          based on the receipt of a Key Request message, and MUST not
8          delete the entry before the SS authorization for the SAID
9          permanently expires."
10
11      INDEX          { ifIndex, wmanIfBsPkmTekSAId }
12      ::= { wmanIfBsPkmTekTable 1 }
13
14
15      WmanIfBsPkmTekEntry ::= SEQUENCE {
16          wmanIfBsPkmTekSAId          INTEGER,
17          wmanIfBsPkmTekSAType        INTEGER,
18          wmanIfBsPkmTekDataEncryptAlg WmanIfDataEncryptAlgId,
19          wmanIfBsPkmTekDataAuthentAlg WmanIfDataAuthAlgId,
20          wmanIfBsPkmTekEncryptAlg    WmanIfTekEncryptAlgId,
21          wmanIfBsPkmTekLifetime      Integer32,
22          wmanIfBsPkmTekKeySequenceNumber Integer32,
23          wmanIfBsPkmTekExpiresOld    DateAndTime,
24          wmanIfBsPkmTekExpiresNew    DateAndTime,
25          wmanIfBsPkmTekReset         TruthValue,
26          wmanIfBsPkmKeyRequests      Counter32,
27          wmanIfBsPkmKeyReplies       Counter32,
28          wmanIfBsPkmKeyRejects       Counter32,
29          wmanIfBsPkmTekInvalids      Counter32,
30          wmanIfBsPkmKeyRejectErrorCode INTEGER,
31          wmanIfBsPkmKeyRejectErrorString SnmpAdminString,
32          wmanIfBsPkmTekInvalidErrorCode INTEGER,
33          wmanIfBsPkmTekInvalidErrorString SnmpAdminString}
34
35
36      wmanIfBsPkmTekSAId OBJECT-TYPE
37          SYNTAX          INTEGER (0..65535)
38          MAX-ACCESS      not-accessible
39          STATUS          current
40          DESCRIPTION
41              "The value of this object is the Security Association
42              ID (SAID)."
```

REFERENCE

```

43          "IEEE Std 802.16-2004; 11.9.7"
44          ::= { wmanIfBsPkmTekEntry 1 }
45
46
47      wmanIfBsPkmTekSAType OBJECT-TYPE
48          SYNTAX          INTEGER {primarySA(0),
49                                  staticSA(1),
50                                  dynamicSA(2)}
51          MAX-ACCESS      read-only
52          STATUS          current
53          DESCRIPTION
54              "The value of this object is the type of security
55              association. Dynamic does not apply to SSs running in PKM
56              mode."
```

REFERENCE

```

1         "IEEE Std 802.16-2004; subclause 11.9.18"
2     ::= { wmanIfBsPkmTekEntry 2 }
3
4
5 wmanIfBsPkmTekDataEncryptAlg OBJECT-TYPE
6     SYNTAX      WmanIfDataEncryptAlgId
7     MAX-ACCESS  read-only
8     STATUS      current
9     DESCRIPTION
10        "The value of this object is the data encryption algorithm
11        being utilized."
12
13     REFERENCE
14        "Table 375, IEEE Std 802.16-2004"
15     ::= { wmanIfBsPkmTekEntry 3 }
16
17
18 wmanIfBsPkmTekDataAuthentAlg OBJECT-TYPE
19     SYNTAX      WmanIfDataAuthAlgId
20     MAX-ACCESS  read-only
21     STATUS      current
22     DESCRIPTION
23        "The value of this object is the data authentication
24        algorithm being utilized."
25
26     REFERENCE
27        "Table 376, IEEE Std 802.16-2004"
28     ::= { wmanIfBsPkmTekEntry 4 }
29
30
31 wmanIfBsPkmTekEncryptAlg OBJECT-TYPE
32     SYNTAX      WmanIfTekEncryptAlgId
33     MAX-ACCESS  read-only
34     STATUS      current
35     DESCRIPTION
36        "The value of this object is the TEK key encryption
37        algorithm being utilized."
38
39     REFERENCE
40        "Table 377, IEEE Std 802.16-2004"
41     ::= { wmanIfBsPkmTekEntry 5 }
42
43
44 wmanIfBsPkmTekLifetime OBJECT-TYPE
45     SYNTAX      Integer32 (1800..604800)
46     UNITS       "seconds"
47     MAX-ACCESS  read-only
48     STATUS      current
49     DESCRIPTION
50        "The value of this object is the lifetime, in seconds, the
51        BS assigns to keys for this TEK association."
52
53     REFERENCE
54        "Table 341 in IEEE Std 802.16-2004"
55
56     DEFVAL      { 43200 }
57     ::= { wmanIfBsPkmTekEntry 6 }
58
59
60 wmanIfBsPkmTekKeySequenceNumber OBJECT-TYPE
61     SYNTAX      Integer32 (0..3)
62     MAX-ACCESS  read-only
63     STATUS      current
64     DESCRIPTION
65

```

```

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

```

```

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

```

```

1      REFERENCE
2          "IEEE Std 802.16-2004; Table 371"
3      ::= { wmanIfBsPkmTekEntry 17 }
4
5
6      wmanIfBsPkmTekInvalidErrorString OBJECT-TYPE
7          SYNTAX      SnmpAdminString (SIZE (0..128))
8          MAX-ACCESS  read-only
9          STATUS      current
10         DESCRIPTION
11             "The value of this object is the Display-String in the most
12              recent TEK Invalid message sent in association with this
13              SAID. This is a zero length string if no TEK Invalid
14              message has been received since reboot."
15         ::= { wmanIfBsPkmTekEntry 18 }
16
17
18
19     --
20     -- Base station Notification Group
21     -- wmanIfBsNotificationObjects contains the BS SNMP Trap objects
22     --
23
24     wmanIfBsNotification OBJECT IDENTIFIER ::= { wmanIfBsObjects 4 }
25     wmanIfBsTrapControl   OBJECT IDENTIFIER ::= { wmanIfBsNotification 1 }
26     wmanIfBsTrapDefinitions OBJECT IDENTIFIER ::= { wmanIfBsNotification 2 }
27
28
29     -- This object groups all NOTIFICATION-TYPE objects for BS.
30     -- It is defined following RFC2758 sections 8.5 and 8.6
31     -- for the compatibility with SNMPv1.
32     wmanIfBsTrapPrefix OBJECT IDENTIFIER ::= { wmanIfBsTrapDefinitions 0 }
33
34
35     wmanIfBsTrapControlRegister OBJECT-TYPE
36         SYNTAX      BITS {wmanIfBsSsStatusNotification      (0),
37                          wmanIfBsSsDynamicServiceFail      (1),
38                          wmanIfBsSsRssiStatusChange         (2),
39                          wmanIfBsSsRegistrer                 (3),
40                          wmanIfBsSsPkmFail                   (4)}
41         MAX-ACCESS  read-write
42         STATUS      current
43         DESCRIPTION
44             "The object is used to enable or disable Base Station traps.
45              From left to right, the set bit indicates the corresponding
46              Base Station trap is enabled."
47         ::= { wmanIfBsTrapControl 1 }
48
49
50
51     wmanIfBsStatusTrapControlRegister OBJECT-TYPE
52         SYNTAX      BITS {unused(0),
53                          ssInitRangingSucc(1),
54                          ssInitRangingFail(2),
55                          ssRegistered(3),
56                          ssRegistrationFail(4),
57                          ssDeregistered(5),
58                          ssBasicCapabilitySucc(6),
59                          ssBasicCapabilityFail(7),
60                          ssAuthorizationSucc(8),
61                          ssAuthorizationFail(9),
62                          tftpSucc(10),
63

```

```

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

```



```

1
2  --
3  -- Subscriber station Notification Objects Definitions
4  --
5
6  wmanIfBsSsNotificationObjectsTable OBJECT-TYPE
7      SYNTAX      SEQUENCE OF WmanIfBsSsNotificationObjectsEntry
8      MAX-ACCESS  not-accessible
9      STATUS      current
10     DESCRIPTION
11         "This table contains SS notification objects that have been
12         reported by the trap."
13     ::= { wmanIfBsTrapDefinitions 1 }
14
15
16
17  wmanIfBsSsNotificationObjectsEntry OBJECT-TYPE
18      SYNTAX      WmanIfBsSsNotificationObjectsEntry
19      MAX-ACCESS  not-accessible
20      STATUS      current
21      DESCRIPTION
22         "This table provides one row for each SS that has
23         generated traps, and is double indexed by
24         wmanIfBsSsNotificationMacAddr and ifIndex for BS sector."
25     INDEX        { ifIndex, wmanIfBsSsNotificationMacAddr }
26     ::= { wmanIfBsSsNotificationObjectsTable 1 }
27
28
29
30  WmanIfBsSsNotificationObjectsEntry ::= SEQUENCE {
31      wmanIfBsSsNotificationMacAddr      MacAddress,
32      wmanIfBsSsStatusValue               INTEGER,
33      wmanIfBsSsStatusInfo                OCTET STRING,
34      wmanIfBsDynamicServiceType          INTEGER,
35      wmanIfBsDynamicServiceFailReason    OCTET STRING,
36      wmanIfBsSsRssiStatus                INTEGER,
37      wmanIfBsSsRssiStatusInfo            OCTET STRING,
38      wmanIfBsSsRegisterStatus            INTEGER}
39
40
41
42  wmanIfBsSsNotificationMacAddr OBJECT-TYPE
43      SYNTAX      MacAddress
44      MAX-ACCESS  read-only
45      STATUS      current
46      DESCRIPTION
47         "The MAC address of the SS, reporing the notofiation."
48     ::= { wmanIfBsSsNotificationObjectsEntry 1 }
49
50
51
52  wmanIfBsSsStatusValue OBJECT-TYPE
53      SYNTAX      INTEGER {ssInitRangingSucc(1),
54                          ssInitRangingFail(2),
55                          ssRegistered(3),
56                          ssRegistrationFail(4),
57                          ssDeregistered(5),
58                          ssBasicCapabilitySucc(6),
59                          ssBasicCapabilityFail(7),
60                          ssAuthorizationSucc(8),
61                          ssAuthorizationFail(9),
62                          tftpSucc(10),
63                          tftpFail(11),
64
65

```

```

1          sfCreationSucc(12),
2          sfCreationFail(13)}
3
4      MAX-ACCESS    read-only
5      STATUS        current
6      DESCRIPTION
7          "This object indicates the status of a SS, as it goes
8          through network entry and initialization procedure."
9      ::= { wmanIfBsSsNotificationObjectsEntry 2 }
10
11
12      wmanIfBsSsStatusInfo OBJECT-TYPE
13          SYNTAX      OCTET STRING (SIZE(0..255))
14          MAX-ACCESS    read-only
15          STATUS        current
16          DESCRIPTION
17              "This object indicates the reason of SS's status change."
18          ::= { wmanIfBsSsNotificationObjectsEntry 3 }
19
20
21      wmanIfBsDynamicServiceType OBJECT-TYPE
22          SYNTAX      INTEGER {bsSfCreationReq(1),
23                          bsSfCreationRsp(2),
24                          bsSfCreationAck(3)}
25          MAX-ACCESS    read-only
26          STATUS        current
27          DESCRIPTION
28              "This object indicates the dynamic service flow
29              creation command type."
30          ::= { wmanIfBsSsNotificationObjectsEntry 4 }
31
32
33
34      wmanIfBsDynamicServiceFailReason OBJECT-TYPE
35          SYNTAX      OCTET STRING (SIZE(0..255))
36          MAX-ACCESS    read-only
37          STATUS        current
38          DESCRIPTION
39              "This object indicates the reason why the service flow
40              creation has failed."
41          ::= { wmanIfBsSsNotificationObjectsEntry 5 }
42
43
44
45      wmanIfBsSsRssiStatus OBJECT-TYPE
46          SYNTAX      INTEGER {bsRssiAlarm(1),
47                          bsRssiNoAlarm(2)}
48          MAX-ACCESS    read-only
49          STATUS        current
50          DESCRIPTION
51              "A RSSI alarm is generated when RSSI becomes lower than
52              wmanIfBsLowRssiThreshold and is cleared when RSSI becomes
53              higher than wmanIfBsLowRssiThreshold."
54          ::= { wmanIfBsSsNotificationObjectsEntry 6 }
55
56
57
58      wmanIfBsSsRssiStatusInfo OBJECT-TYPE
59          SYNTAX      OCTET STRING (SIZE(0..255))
60          MAX-ACCESS    read-only
61          STATUS        current
62          DESCRIPTION
63              "This object indicates the reason why RSSI alarm is
64
65

```

```

1         generated."
2         ::= { wmanIfBsSsNotificationObjectsEntry 7 }
3
4
5 wmanIfBsSsRegisterStatus OBJECT-TYPE
6     SYNTAX      INTEGER {ssRegister(1),
7                     ssDeregister(2)}
8     MAX-ACCESS  read-only
9     STATUS      current
10    DESCRIPTION
11        "This object indicates the status of SS registration."
12    ::= { wmanIfBsSsNotificationObjectsEntry 8 }
13
14
15 --
16 -- Subscriber station Notification Trap Definitions
17 --
18
19 wmanIfBsSsStatusNotificationTrap NOTIFICATION-TYPE
20     OBJECTS      {ifIndex,
21                     wmanIfBsSsNotificationMacAddr,
22                     wmanIfBsSsStatusValue,
23                     wmanIfBsSsStatusInfo}
24     STATUS      current
25     DESCRIPTION
26        "This trap reports the status of a SS. Based on this
27        notification the NMS will issue an alarm with certain
28        severity depending on the status and the reason received."
29    ::= { wmanIfBsTrapPrefix 1 }
30
31
32
33 wmanIfBsSsDynamicServiceFailTrap NOTIFICATION-TYPE
34     OBJECTS      {ifIndex,
35                     wmanIfBsSsNotificationMacAddr,
36                     wmanIfBsDynamicServiceType,
37                     wmanIfBsDynamicServiceFailReason}
38     STATUS      current
39     DESCRIPTION
40        "An event to report the failure of a dynamic service
41        operation happened during the dynamic services process
42        and detected in the Bs side."
43    ::= { wmanIfBsTrapPrefix 2 }
44
45
46
47
48 wmanIfBsSsRssiStatusChangeTrap NOTIFICATION-TYPE
49     OBJECTS      {ifIndex,
50                     wmanIfBsSsNotificationMacAddr,
51                     wmanIfBsSsRssiStatus,
52                     wmanIfBsSsRssiStatusInfo}
53     STATUS      current
54     DESCRIPTION
55        "An event to report that the uplink RSSI is below
56        wmanIfBsLowRssiThreshold, or above
57        wmanIfBsHighRssiThreshold after restore."
58    ::= { wmanIfBsTrapPrefix 3 }
59
60
61
62 wmanIfBsSsPkmFailTrap NOTIFICATION-TYPE
63     OBJECTS      {wmanIfBsSsNotificationMacAddr}
64     STATUS      current
65

```

```

1      DESCRIPTION
2          "An event to report the failure of a Pkm operation."
3      ::= { wmanIfBsTrapPrefix 4 }
4
5
6      wmanIfBsSsRegistrerTrap NOTIFICATION-TYPE
7          OBJECTS      {wmanIfBsSsNotificationMacAddr,
8                        wmanIfBsSsRegisterStatus}
9          STATUS      current
10         DESCRIPTION
11             "An event to report SS registration status."
12         ::= { wmanIfBsTrapPrefix 5 }
13
14
15     --
16     -- Base station PHY Group
17     --
18     --
19     wmanIfBsPhy OBJECT IDENTIFIER ::= { wmanIfBsObjects 6 }
20
21     --
22     -- BS OFDM PHY objects
23     --
24     --
25     wmanIfBsOfdmPhy OBJECT IDENTIFIER ::= { wmanIfBsPhy 1 }
26
27     wmanIfBsOfdmUplinkChannelTable OBJECT-TYPE
28         SYNTAX      SEQUENCE OF WmanIfBsOfdmUplinkChannelEntry
29         MAX-ACCESS  not-accessible
30         STATUS      current
31         DESCRIPTION
32             "This table contains UCD channel attributes, defining the
33             transmission characteristics of uplink channels"
34         REFERENCE
35             "Table 349 and Table 352, in IEEE Std 802.16-2004"
36         ::= { wmanIfBsOfdmPhy 1 }
37
38
39     wmanIfBsOfdmUplinkChannelEntry OBJECT-TYPE
40         SYNTAX      WmanIfBsOfdmUplinkChannelEntry
41         MAX-ACCESS  not-accessible
42         STATUS      current
43         DESCRIPTION
44             "This table provides one row for each uplink channel of
45             multi-sector BS, and is indexed by BS ifIndex. An entry
46             in this table exists for each ifEntry of BS with an
47             ifType of propBWAmp2Mp."
48         INDEX { ifIndex }
49         ::= { wmanIfBsOfdmUplinkChannelTable 1 }
50
51     WmanIfBsOfdmUplinkChannelEntry ::= SEQUENCE {
52         wmanIfBsOfdmCtBasedResvTimeout      INTEGER,
53         wmanIfBsOfdmBwReqOppSize            INTEGER,
54         wmanIfBsOfdmRangReqOppSize          INTEGER,
55         wmanIfBsOfdmUplinkCenterFreq        Unsigned32,
56         wmanIfBsOfdmNumSubChReqRegionFull   INTEGER,
57         wmanIfBsOfdmNumSymbolsReqRegionFull INTEGER,
58         wmanIfBsOfdmSubChFocusCtCode        INTEGER,
59         wmanIfBsOfdmUpLinkChannelId         INTEGER}
60
61
62
63
64
65

```

```

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

```

```

1          SYNTAX          INTEGER {oneSubchannel(0),
2                                twoSubchannels(1),
3                                fourSubchannels(2),
4                                eightSubchannels(3),
5                                sixteenSubchannels(4)}
6
7          MAX-ACCESS      read-write
8          STATUS          current
9          DESCRIPTION
10             "Number of subchannels used by each transmit
11             opportunity when REQ Region-Full is allocated in
12             subchannelization region."
13
14          REFERENCE
15             "Table 352, in IEEE Std 802.16-2004"
16             ::= { wmanIfBsOfdmUplinkChannelEntry 5 }
17
18
19          wmanIfBsOfdmNumSymbolsReqRegionFull OBJECT-TYPE
20             SYNTAX          INTEGER (0..31)
21             MAX-ACCESS      read-write
22             STATUS          current
23             DESCRIPTION
24                 "Number of OFDM symbols used by each transmit
25                 opportunity when REQ Region-Full is allocated in
26                 subchannelization region."
27
28             REFERENCE
29                 "Table 352, in IEEE Std 802.16-2004"
30                 ::= { wmanIfBsOfdmUplinkChannelEntry 6 }
31
32
33          wmanIfBsOfdmSubChFocusCtCode OBJECT-TYPE
34             SYNTAX          INTEGER (0..8)
35             MAX-ACCESS      read-write
36             STATUS          current
37             DESCRIPTION
38                 "Number of contention codes (CSE) that shall only be used to
39                 request a subchannelized allocation. Default value 0.
40                 Allowed values 0-8."
41
42             REFERENCE
43                 "Table 352, in IEEE Std 802.16-2004"
44
45             DEFVAL          { 0 }
46             ::= { wmanIfBsOfdmUplinkChannelEntry 7 }
47
48
49          wmanIfBsOfdmUpLinkChannelId OBJECT-TYPE
50             SYNTAX          INTEGER (0..255)
51             MAX-ACCESS      read-write
52             STATUS          current
53             DESCRIPTION
54                 "The identifier of the uplink channel to which the relevant
55                 RNG-RSP or RNG-REQ message refers."
56
57             REFERENCE
58                 "Subclause 6.3.2.3.4. Table 16, in IEEE Std 802.16-2004"
59                 ::= { wmanIfBsOfdmUplinkChannelEntry 8 }
60
61
62          wmanIfBsOfdmDownlinkChannelTable OBJECT-TYPE
63             SYNTAX          SEQUENCE OF WmanIfBsOfdmDownlinkChannelEntry
64             MAX-ACCESS      not-accessible
65

```

```

1      STATUS      current
2      DESCRIPTION
3          "This table contains DCD channel attributes, defining the
4              transmission characteristics of downlink channels"
5
6      REFERENCE
7          "Table 358, in IEEE Std 802.16-2004"
8      ::= { wmanIfBsOfdmPhy 2 }
9
10
11  wmanIfBsOfdmDownlinkChannelEntry OBJECT-TYPE
12      SYNTAX      WmanIfBsOfdmDownlinkChannelEntry
13      MAX-ACCESS  not-accessible
14      STATUS      current
15      DESCRIPTION
16          "This table provides one row for each downlink 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      ::= { wmanIfBsOfdmDownlinkChannelTable 1 }
22
23
24
25  WmanIfBsOfdmDownlinkChannelEntry ::= SEQUENCE {
26      wmanIfBsOfdmBseIRP          INTEGER,
27      wmanIfBsOfdmChannelNumber   WmanIfChannelNumber,
28      wmanIfBsOfdmTTG             INTEGER,
29      wmanIfBsOfdmRTG             INTEGER,
30      wmanIfBsOfdmInitRngMaxRSS   INTEGER,
31      wmanIfBsOfdmDownlinkCenterFreq Unsigned32,
32      wmanIfBsOfdmBsId           WmanIfBsIdType,
33      wmanIfBsOfdmMacVersion      WmanIfMacVersion,
34      wmanIfBsOfdmFrameDurationCode INTEGER,
35      wmanIfBsOfdmDownLinkChannelId INTEGER}
36
37
38
39  wmanIfBsOfdmBseIRP OBJECT-TYPE
40      SYNTAX      INTEGER (0..65535)
41      UNITS       "dBm"
42      MAX-ACCESS  read-write
43      STATUS      current
44      DESCRIPTION
45          "The EIRP is the equivalent isotropic radiated power of
46              the base station, which is computed for a simple
47              single-antenna transmitter."
48      REFERENCE
49          "Table 358, in IEEE Std 802.16-2004"
50      ::= { wmanIfBsOfdmDownlinkChannelEntry 1 }
51
52
53
54
55  wmanIfBsOfdmChannelNumber OBJECT-TYPE
56      SYNTAX      WmanIfChannelNumber
57      MAX-ACCESS  read-write
58      STATUS      current
59      DESCRIPTION
60          "Downlink channel number as defined in 8.5.
61              Used for license-exempt operation only."
62      REFERENCE
63          "Table 358, in IEEE Std 802.16-2004"
64
65

```

```

1      ::= { wmanIfBsOfdmDownlinkChannelEntry 2 }
2
3
4  wmanIfBsOfdmTTG OBJECT-TYPE
5      SYNTAX      INTEGER (0..255)
6      MAX-ACCESS  read-write
7      STATUS      current
8      DESCRIPTION
9          "Transmit / Receive Transition Gap."
10     REFERENCE
11         "Table 358, in IEEE Std 802.16-2004"
12     ::= { wmanIfBsOfdmDownlinkChannelEntry 3 }
13
14
15  wmanIfBsOfdmRTG OBJECT-TYPE
16      SYNTAX      INTEGER (0..255)
17      MAX-ACCESS  read-write
18      STATUS      current
19      DESCRIPTION
20          "Receive / Transmit Transition Gap."
21     REFERENCE
22         "Table 358, in IEEE Std 802.16-2004"
23     ::= { wmanIfBsOfdmDownlinkChannelEntry 4 }
24
25
26
27  wmanIfBsOfdmInitRngMaxRSS OBJECT-TYPE
28      SYNTAX      INTEGER (0..65535)
29      UNITS        "dBm"
30      MAX-ACCESS  read-write
31      STATUS      current
32      DESCRIPTION
33          "Initial Ranging Max. Received Signal Strength at BS
34          Signed in units of 1 dBm."
35     REFERENCE
36         "Table 358, in IEEE Std 802.16-2004"
37     ::= { wmanIfBsOfdmDownlinkChannelEntry 5 }
38
39
40
41
42  wmanIfBsOfdmDownlinkCenterFreq OBJECT-TYPE
43      SYNTAX      Unsigned32
44      UNITS        "kHz"
45      MAX-ACCESS  read-write
46      STATUS      current
47      DESCRIPTION
48          "Downlink center frequency (kHz)."
49     REFERENCE
50         "Table 358, in IEEE Std 802.16-2004"
51     ::= { wmanIfBsOfdmDownlinkChannelEntry 6 }
52
53
54
55  wmanIfBsOfdmBsId OBJECT-TYPE
56      SYNTAX      WmanIfBsIdType
57      MAX-ACCESS  read-write
58      STATUS      current
59      DESCRIPTION
60          "Base station ID."
61     REFERENCE
62         "Table 358, in IEEE Std 802.16-2004"
63     ::= { wmanIfBsOfdmDownlinkChannelEntry 7 }
64
65

```



```

1
2 wmanIfBsOfdmMacVersion OBJECT-TYPE
3     SYNTAX      WmanIfMacVersion
4     MAX-ACCESS  read-write
5     STATUS      current
6     DESCRIPTION
7         "This parameter specifies the version of 802.16 to which
8         the message originator conforms."
9     REFERENCE
10        "Table 358, in IEEE Std 802.16-2004"
11        ::= { wmanIfBsOfdmDownlinkChannelEntry 8 }
12
13 wmanIfBsOfdmFrameDurationCode OBJECT-TYPE
14     SYNTAX      INTEGER {duration2dot5ms(0),
15                          duration4ms(1),
16                          duration5ms(2),
17                          duration8ms(3),
18                          duration10ms(4),
19                          duration12dot5ms(5),
20                          duration20ms(6)}
21     MAX-ACCESS  read-write
22     STATUS      current
23     DESCRIPTION
24         "The duration of the frame. The frame duration code
25         values are specified in Table 230."
26     REFERENCE
27        "Subclause 11.4.1, Table 230, in IEEE Std 802.16-2004"
28        ::= { wmanIfBsOfdmDownlinkChannelEntry 9 }
29
30 wmanIfBsOfdmDownLinkChannelId OBJECT-TYPE
31     SYNTAX      INTEGER (0..255)
32     MAX-ACCESS  read-write
33     STATUS      current
34     DESCRIPTION
35         "The identifier of the downlink channel to which this
36         message refers."
37     REFERENCE
38        "Subclause 6.3.2.3.1. Table 15, in IEEE Std 802.16-2004"
39        ::= { wmanIfBsOfdmDownlinkChannelEntry 10 }
40
41 wmanIfBsOfdmUcdBurstProfileTable OBJECT-TYPE
42     SYNTAX      SEQUENCE OF WmanIfBsOfdmUcdBurstProfileEntry
43     MAX-ACCESS  not-accessible
44     STATUS      current
45     DESCRIPTION
46         "This table contains UCD burst profiles for each uplink
47         channel"
48     REFERENCE
49        "Table 356, in IEEE Std 802.16-2004"
50        ::= { wmanIfBsOfdmPhy 3 }
51
52 wmanIfBsOfdmUcdBurstProfileEntry OBJECT-TYPE
53     SYNTAX      WmanIfBsOfdmUcdBurstProfileEntry
54     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 propBWAmp2Mp. The secondary index
6          is wmanIfBsOfdmUiucIndex."
7
8      INDEX { ifIndex, wmanIfBsOfdmUiucIndex }
9      ::= { wmanIfBsOfdmUcdBurstProfileTable 1 }
10
11
12      WmanIfBsOfdmUcdBurstProfileEntry ::= SEQUENCE {
13          wmanIfBsOfdmUiucIndex      INTEGER,
14          wmanIfBsOfdmUcdFecCodeType  WmanIfOfdmFecCodeType,
15          wmanIfBsOfdmFocusCtPowerBoost  INTEGER,
16          wmanIfBsOfdmUcdTcsEnable    INTEGER,
17          wmanIfBsOfdmUcdBurstProfileRowStatus  RowStatus}
18
19
20      wmanIfBsOfdmUiucIndex OBJECT-TYPE
21          SYNTAX      INTEGER (5 .. 12)
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              wmanIfBsOfdmUcdBurstProfileTable."
29          REFERENCE
30              "Subclause 8.3.6.3.1, in IEEE Std 802.16-2004"
31          ::= { wmanIfBsOfdmUcdBurstProfileEntry 1 }
32
33
34
35
36      wmanIfBsOfdmUcdFecCodeType OBJECT-TYPE
37          SYNTAX      WmanIfOfdmFecCodeType
38          MAX-ACCESS  read-create
39          STATUS      current
40          DESCRIPTION
41              "Uplink FEC code type and modulation type"
42          REFERENCE
43              "Table 356, in IEEE Std 802.16-2004"
44          ::= { wmanIfBsOfdmUcdBurstProfileEntry 2 }
45
46
47
48
49      wmanIfBsOfdmFocusCtPowerBoost OBJECT-TYPE
50          SYNTAX      INTEGER (0 .. 255)
51          MAX-ACCESS  read-create
52          STATUS      current
53          DESCRIPTION
54              "The power boost in dB of focused contention carriers, as
55              described in 8.3.6.3.3."
56          REFERENCE
57              "Table 356, in IEEE Std 802.16-2004"
58          ::= { wmanIfBsOfdmUcdBurstProfileEntry 3 }
59
60
61
62      wmanIfBsOfdmUcdTcsEnable OBJECT-TYPE
63          SYNTAX      INTEGER {tcsDisabled(0),
64                      tcsEnabled(1)}
65          MAX-ACCESS  read-create

```

```

1      STATUS      current
2      DESCRIPTION
3          "This parameter determines the transmission convergence
4          sublayer, as described in 8.1.4.3, can be enabled on a
5          per-burst basis for both uplink and downlink. through
6          DIUC/UIUC messages."
7
8      REFERENCE
9          "Table 356, in IEEE Std 802.16-2004"
10     ::= { wmanIfBsOfdmUcdBurstProfileEntry 4 }
11
12
13 wmanIfBsOfdmUcdBurstProfileRowStatus OBJECT-TYPE
14     SYNTAX      RowStatus
15     MAX-ACCESS  read-create
16     STATUS      current
17     DESCRIPTION
18         "This object is used to create a new row or modify or
19         delete an existing row in this table.
20
21         If the implementator of this MIB has choosen not
22         to implement 'dynamic assignment' of profiles, this
23         object is not useful and should return noSuchName
24         upon SNMP request."
25     ::= { wmanIfBsOfdmUcdBurstProfileEntry 5 }
26
27
28
29
30 wmanIfBsOfdmDcdBurstProfileTable OBJECT-TYPE
31     SYNTAX      SEQUENCE OF WmanIfBsOfdmDcdBurstProfileEntry
32     MAX-ACCESS  not-accessible
33     STATUS      current
34     DESCRIPTION
35         "This table provides one row for each DCD burst profile.
36         This table is double indexed. The primary index is an
37         ifIndex with an ifType of propBWA2Mp. The secondary
38         index is wmanIfBsOfdmDiucIndex."
39     REFERENCE
40         "Table 362, in IEEE Std 802.16-2004"
41     ::= { wmanIfBsOfdmPhy 4 }
42
43
44
45 wmanIfBsOfdmDcdBurstProfileEntry OBJECT-TYPE
46     SYNTAX      WmanIfBsOfdmDcdBurstProfileEntry
47     MAX-ACCESS  not-accessible
48     STATUS      current
49     DESCRIPTION
50         "This table provides one row for each DCD burst profile.
51         This table is double indexed. The primary index is an
52         ifIndex with an ifType of propBWA2Mp. The secondary index
53         is wmanIfBsOfdmDiucIndex."
54     INDEX { ifIndex, wmanIfBsOfdmDiucIndex }
55     ::= { wmanIfBsOfdmDcdBurstProfileTable 1 }
56
57
58
59 WmanIfBsOfdmDcdBurstProfileEntry ::= SEQUENCE {
60     wmanIfBsOfdmDiucIndex          INTEGER,
61     wmanIfBsOfdmDownlinkFrequency Unsigned32,
62     wmanIfBsOfdmDcdFecCodeType    WmanIfOfdmFecCodeType,
63     wmanIfBsOfdmDiucMandatoryExitThresh INTEGER,
64

```

```

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

```

```

1      DESCRIPTION
2          "DIUC minimum entry threshold: 0 - 63.75 dB The minimum CINR
3          required to start using this DIUC when changing from a more
4          robust DIUC is required, in 0.25 dB units."
5
6      REFERENCE
7          "Table 362, in IEEE Std 802.16-2004"
8      ::= { wmanIfBsOfdmDcdBurstProfileEntry 5 }
9
10
11  wmanIfBsOfdmTcsEnable OBJECT-TYPE
12      SYNTAX      INTEGER {tcsDisabled (0),
13                      tcsEnabled (1)}
14      MAX-ACCESS  read-create
15      STATUS      current
16      DESCRIPTION
17          "Indicates whether Transmission Convergence Sublayer
18          is enabled or disabled."
19
20      REFERENCE
21          "Table 362, in IEEE Std 802.16-2004"
22      ::= { wmanIfBsOfdmDcdBurstProfileEntry 6 }
23
24
25  wmanIfBsOfdmDcdBurstProfileRowStatus OBJECT-TYPE
26      SYNTAX      RowStatus
27      MAX-ACCESS  read-create
28      STATUS      current
29      DESCRIPTION
30          "This object is used to create a new row or modify or
31          delete an existing row in this table.
32
33          If the implementator of this MIB has choosen not
34          to implement 'dynamic assignment' of profiles, this
35          object is not useful and should return noSuchName
36          upon SNMP request."
37      ::= { wmanIfBsOfdmDcdBurstProfileEntry 7 }
38
39
40
41
42  wmanIfBsOfdmConfigurationTable OBJECT-TYPE
43      SYNTAX      SEQUENCE OF WmanIfBsOfdmConfigurationEntry
44      MAX-ACCESS  not-accessible
45      STATUS      current
46      DESCRIPTION
47          "This table contains BS configuration objects, specific to
48          OFDM PHY."
49      ::= { wmanIfBsOfdmPhy 5 }
50
51
52
53  wmanIfBsOfdmConfigurationEntry OBJECT-TYPE
54      SYNTAX      WmanIfBsOfdmConfigurationEntry
55      MAX-ACCESS  not-accessible
56      STATUS      current
57      DESCRIPTION
58          "This table is indexed by ifIndex with an ifType of
59          propBWA2Mp."
60      INDEX { ifIndex }
61      ::= { wmanIfBsOfdmConfigurationTable 1 }
62
63
64  WmanIfBsOfdmConfigurationEntry ::= SEQUENCE {
65

```

```

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

```

```

1          limits that are deemed tolerable by the BS. If the SS does
2          not transmit within these limits after a number of
3          correction attempts then the BS aborts Initial Ranging."
4
5      REFERENCE
6          "Figure 63 and Table 365 in IEEE Std 802.16-2004"
7      ::= { wmanIfBsOfdmConfigurationEntry 4 }
8
9
10     wmanIfBsOfdmRangeAbortPowerThold OBJECT-TYPE
11         SYNTAX      INTEGER (0..255)
12         UNITS        "0.25dB"
13         MAX-ACCESS   read-write
14         STATUS       current
15         DESCRIPTION
16             "This object defines Tolerable Power Offset. BS performs
17             Initial Ranging until the SS transmissions are within
18             limits that are deemed tolerable by the BS. If the SS does
19             not transmit within these limits after a number of
20             correction attempts then the BS aborts Initial Ranging."
21
22     REFERENCE
23         "Figure 63 and Table 365 in IEEE Std 802.16-2004"
24     ::= { wmanIfBsOfdmConfigurationEntry 5 }
25
26
27     wmanIfBsOfdmRangeAbortFreqThold OBJECT-TYPE
28         SYNTAX      INTEGER (0..255)
29         UNITS        "Hz"
30         MAX-ACCESS   read-write
31         STATUS       current
32         DESCRIPTION
33             "This object defines Tolerable Frequency Offset. BS performs
34             Initial Ranging until the SS transmissions are within
35             limits that are deemed tolerable by the BS. If the SS does
36             not transmit within these limits after a number of
37             correction attempts then the BS aborts Initial Ranging."
38
39     REFERENCE
40         "Figure 63 and Table 365 in IEEE Std 802.16-2004"
41     ::= { wmanIfBsOfdmConfigurationEntry 6 }
42
43
44
45     wmanIfBsOfdmDnlnkRateId OBJECT-TYPE
46         SYNTAX      INTEGER {dnlnkRateIdBpsk1Over2(0),
47                               dnlnkRateIdQpsk1Over2(1),
48                               dnlnkRateIdQpsk3Over4(2),
49                               dnlnkRateId16Qam1Over2(3),
50                               dnlnkRateId16Qam3Over4(4),
51                               dnlnkRateId64Qam2Over3(5),
52                               dnlnkRateId64Qam3Over4(6)}
53         MAX-ACCESS   read-write
54         STATUS       current
55         DESCRIPTION
56             "The Rate ID to be used in the first downlink burst
57             immediately following the FCH. The Rate ID encoding is
58             static and cannot be changed during system operation. The
59             change of the Rate ID should be applied on system
60             re-intialisation (e.g. following sector or BS reset)."

```

```

1          "Subclause 8.3.3.4.3 in IEEE Std 802.16-2004"
2      DEFVAL      { dn1kRateIdBpsk1Over2 }
3      ::= { wmanIfBsOfdmConfigurationEntry 7 }
4
5
6      wmanIfBsOfdmRatioG OBJECT-TYPE
7          SYNTAX  INTEGER {ratio1To4(0),
8                      ratio1To8(1),
9                      ratio1To16(2),
10                     ratio1To32(3)}
11
12          MAX-ACCESS  read-write
13          STATUS      current
14          DESCRIPTION
15              "The ratio of CP time to 'useful' time.Values
16              are 1/4, 1/8, 1/16 or 1/32."
17
18          REFERENCE
19              "Subclause 8.3.1.1.1 in IEEE Std 802.16-2004"
20      DEFVAL      { ratio1To4 }
21      ::= { wmanIfBsOfdmConfigurationEntry 8 }
22
23
24      wmanIfBsSsOfdmReqCapabilitiesTable OBJECT-TYPE
25          SYNTAX      SEQUENCE OF WmanIfBsSsOfdmReqCapabilitiesEntry
26          MAX-ACCESS  not-accessible
27          STATUS      current
28          DESCRIPTION
29              "This table contains the basic capability information,
30              specific to OFDM Phy, of SSs that have been reported by
31              SSs to BS using RNG-REQ, SBC-REQ and REG-REQ messages.
32              Entries in this table should be created when an SS
33              registers with a BS."
34
35      ::= { wmanIfBsOfdmPhy 6 }
36
37
38      wmanIfBsSsOfdmReqCapabilitiesEntry OBJECT-TYPE
39          SYNTAX      WmanIfBsSsOfdmReqCapabilitiesEntry
40          MAX-ACCESS  not-accessible
41          STATUS      current
42          DESCRIPTION
43              "This table provides one row for each SS that has been
44              registered in the BS. This table augments the table
45              wmanIfBsRegisteredSsTable."
46
47          AUGMENTS { wmanIfBsRegisteredSsEntry }
48      ::= { wmanIfBsSsOfdmReqCapabilitiesTable 1 }
49
50
51      WmanIfBsSsOfdmReqCapabilitiesEntry ::= SEQUENCE {
52          wmanIfBsSsOfdmReqCapFftSizes      WmanIfOfdmFftSizes,
53          wmanIfBsSsOfdmReqCapSsDemodulator  WmanIfOfdmSsDeModType,
54          wmanIfBsSsOfdmReqCapSsModulator     WmanIfOfdmSsModType,
55          wmanIfBsSsOfdmReqCapFocusedCtSupport WmanIfOfdmFocusedCt,
56          wmanIfBsSsOfdmReqCapTcSublayerSupport WmanIfOfdmTcSublayer}
57
58
59      wmanIfBsSsOfdmReqCapFftSizes OBJECT-TYPE
60          SYNTAX      WmanIfOfdmFftSizes
61          MAX-ACCESS  read-only
62          STATUS      current
63          DESCRIPTION
64
65

```



```

1         "This field indicates the FFT sizes supported by SS.
2         The usage is defined by WmanIfOfdmFftSizes."
3         ::= { wmanIfBsSsOfdmReqCapabilitiesEntry 1 }
4
5
6 wmanIfBsSsOfdmReqCapSsDemodulator OBJECT-TYPE
7     SYNTAX      WmanIfOfdmSsDeModType
8     MAX-ACCESS  read-only
9     STATUS      current
10    DESCRIPTION
11        "This field indicates the different demodulator options
12        supported by SS for downlink.
13        The usage is defined by WmanIfOfdmSsDeModType."
14        ::= { wmanIfBsSsOfdmReqCapabilitiesEntry 2 }
15
16
17
18 wmanIfBsSsOfdmReqCapSsModulator OBJECT-TYPE
19     SYNTAX      WmanIfOfdmSsModType
20     MAX-ACCESS  read-only
21     STATUS      current
22     DESCRIPTION
23        "This field indicates the different modulator options
24        supported by SS for uplink.
25        The usage is defined by WmanIfOfdmSsModType."
26        ::= { wmanIfBsSsOfdmReqCapabilitiesEntry 3 }
27
28
29
30 wmanIfBsSsOfdmReqCapFocusedCtSupport OBJECT-TYPE
31     SYNTAX      WmanIfOfdmFocusedCt
32     MAX-ACCESS  read-only
33     STATUS      current
34     DESCRIPTION
35        "This field indicates whether the SS supports Focused
36        Contention. The usage is defined by
37        WmanIfOfdmFocusedCt."
38        ::= { wmanIfBsSsOfdmReqCapabilitiesEntry 4 }
39
40
41
42 wmanIfBsSsOfdmReqCapTcSublayerSupport OBJECT-TYPE
43     SYNTAX      WmanIfOfdmTcSublayer
44     MAX-ACCESS  read-only
45     STATUS      current
46     DESCRIPTION
47        "This field indicates whether or not the SS supports
48        the TC sublayer. The usage is defined by
49        WmanIfOfdmTcSublayer."
50        ::= { wmanIfBsSsOfdmReqCapabilitiesEntry 5 }
51
52
53
54 wmanIfBsSsOfdmRspCapabilitiesTable OBJECT-TYPE
55     SYNTAX      SEQUENCE OF WmanIfBsSsOfdmRspCapabilitiesEntry
56     MAX-ACCESS  not-accessible
57     STATUS      current
58     DESCRIPTION
59        "This table contains the basic capability information,
60        specific to OFDM Phy, of SSs that have been negotiated
61        and agreed between BS and SS via RNG-REQ/RSP,
62        SBC-REQ/RSP and REG-REQ/RSP messages. This table
63        augments the wmanIfBsRegisteredSsTable."
64
65

```

```

1      REFERENCE
2          "Subclause 6.3.2.3.7 in IEEE Std 802.16-2004"
3      ::= { wmanIfBsOfdmPhy 7 }
4
5
6      wmanIfBsSsOfdmRspCapabilitiesEntry OBJECT-TYPE
7          SYNTAX      WmanIfBsSsOfdmRspCapabilitiesEntry
8          MAX-ACCESS   not-accessible
9          STATUS       current
10         DESCRIPTION
11             "This table provides one row for each SS that has been
12              registered in the BS. This table augments the
13              wmanIfBsRegisteredSsTable. "
14         AUGMENTS { wmanIfBsRegisteredSsEntry }
15         ::= { wmanIfBsSsOfdmRspCapabilitiesTable 1 }
16
17
18
19      WmanIfBsSsOfdmRspCapabilitiesEntry ::= SEQUENCE {
20          wmanIfBsSsOfdmRspCapFftSizes      WmanIfOfdmFftSizes,
21          wmanIfBsSsOfdmRspCapSsDemodulator WmanIfOfdmSsDeModType,
22          wmanIfBsSsOfdmRspCapSsModulator   WmanIfOfdmSsModType,
23          wmanIfBsSsOfdmRspCapFocusedCtSupport WmanIfOfdmFocusedCt,
24          wmanIfBsSsOfdmRspCapTcSublayerSupport WmanIfOfdmTcSublayer}
25
26
27
28      wmanIfBsSsOfdmRspCapFftSizes OBJECT-TYPE
29          SYNTAX      WmanIfOfdmFftSizes
30          MAX-ACCESS   read-only
31          STATUS       current
32          DESCRIPTION
33              "This field indicates the FFT sizes negotiated with the
34              SS. The usage is defined by WmanIfOfdmFftSizes."
35          ::= { wmanIfBsSsOfdmRspCapabilitiesEntry 1 }
36
37
38
39      wmanIfBsSsOfdmRspCapSsDemodulator OBJECT-TYPE
40          SYNTAX      WmanIfOfdmSsDeModType
41          MAX-ACCESS   read-only
42          STATUS       current
43          DESCRIPTION
44              "This field indicates the different demodulator options
45              negotiated for SS for downlink. The usage is defined by
46              WmanIfOfdmSsDeModType."
47          ::= { wmanIfBsSsOfdmRspCapabilitiesEntry 2 }
48
49
50
51      wmanIfBsSsOfdmRspCapSsModulator OBJECT-TYPE
52          SYNTAX      WmanIfOfdmSsModType
53          MAX-ACCESS   read-only
54          STATUS       current
55          DESCRIPTION
56              "This field indicates the different modulator options
57              negotiated for SS for uplink. The usage is defined by
58              WmanIfOfdmSsModType."
59          ::= { wmanIfBsSsOfdmRspCapabilitiesEntry 3 }
60
61
62
63      wmanIfBsSsOfdmRspCapFocusedCtSupport OBJECT-TYPE
64          SYNTAX      WmanIfOfdmFocusedCt
65          MAX-ACCESS   read-only

```

```

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

```

```

1      STATUS      current
2      DESCRIPTION
3          "This field indicates the FFT sizes supported by the BS.
4          The usage is defined by WmanIfOfdmCapFftSizes."
5          ::= { wmanIfBsOfdmCapabilitiesEntry 1 }
6
7
8      wmanIfBsOfdmCapSsDemodulator OBJECT-TYPE
9          SYNTAX      WmanIfOfdmSsDeModType
10         MAX-ACCESS  read-only
11         STATUS      current
12         DESCRIPTION
13             "This field indicates the different BS demodulator options
14             for uplink supported by the BS. The usage is defined by
15             WmanIfOfdmSsDeModType."
16             ::= { wmanIfBsOfdmCapabilitiesEntry 2 }
17
18
19
20     wmanIfBsOfdmCapSsModulator OBJECT-TYPE
21         SYNTAX      WmanIfOfdmSsModType
22         MAX-ACCESS  read-only
23         STATUS      current
24         DESCRIPTION
25             "This field indicates the different BS modulator options
26             for downlink supported by the BS. The usage is defined by
27             WmanIfOfdmSsModType."
28             ::= { wmanIfBsOfdmCapabilitiesEntry 3 }
29
30
31
32     wmanIfBsOfdmCapFocusedCtSupport OBJECT-TYPE
33         SYNTAX      WmanIfOfdmFocusedCt
34         MAX-ACCESS  read-only
35         STATUS      current
36         DESCRIPTION
37             "This field indicates the BS support for Focused
38             Contention. The usage is defined by
39             WmanIfOfdmFocusedCt."
40             ::= { wmanIfBsOfdmCapabilitiesEntry 4 }
41
42
43
44     wmanIfBsOfdmCapTcSublayerSupport OBJECT-TYPE
45         SYNTAX      WmanIfOfdmTcSublayer
46         MAX-ACCESS  read-only
47         STATUS      current
48         DESCRIPTION
49             "This field indicates the BS supports for TC sublayer. The
50             usage is defined by WmanIfOfdmTcSublayer."
51             ::= { wmanIfBsOfdmCapabilitiesEntry 5 }
52
53
54
55     wmanIfBsOfdmCapabilitiesConfigTable OBJECT-TYPE
56         SYNTAX      SEQUENCE OF WmanIfBsOfdmCapabilitiesConfigEntry
57         MAX-ACCESS  not-accessible
58         STATUS      current
59         DESCRIPTION
60             "This table contains the configuration for basic
61             capabilities of BS, specific to OFDM Phy. The table is
62             intended to be used to restrict the Capabilities
63             implemented by BS, for example in order to comply with
64
65

```

```

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

```

```

1           "This field indicates the different BS modulator options
2           configured for downlink. The usage is defined by
3           WmanIfOfdmSsModType."
4           ::= { wmanIfBsOfdmCapabilitiesConfigEntry 3 }
5
6
7 wmanIfBsOfdmCapCfgFocusedCtSupport OBJECT-TYPE
8     SYNTAX      WmanIfOfdmFocusedCt
9     MAX-ACCESS  read-write
10    STATUS      current
11    DESCRIPTION
12      "This field indicates the BS support configured for
13      Focused Contention. The usage is defined by
14      WmanIfOfdmFocusedCt."
15      ::= { wmanIfBsOfdmCapabilitiesConfigEntry 4 }
16
17
18
19 wmanIfBsOfdmCapCfgTcSublayerSupport OBJECT-TYPE
20     SYNTAX      WmanIfOfdmTcSublayer
21     MAX-ACCESS  read-write
22     STATUS      current
23     DESCRIPTION
24       "This field indicates the BS support configured for TC
25       sublayer. The usage is defined by
26       WmanIfOfdmTcSublayer."
27       ::= { wmanIfBsOfdmCapabilitiesConfigEntry 5 }
28
29
30
31 --
32 -- BS OFDMA PHY objects
33 --
34
35 wmanIfBsOfdmaPhy OBJECT IDENTIFIER ::= { wmanIfBsPhy 2 }
36
37
38 wmanIfBsOfdmaUplinkChannelTable OBJECT-TYPE
39     SYNTAX      SEQUENCE OF WmanIfBsOfdmaUplinkChannelEntry
40     MAX-ACCESS  not-accessible
41     STATUS      current
42     DESCRIPTION
43       "This table contains UCD channel attributes, defining the
44       transmission characteristics of uplink channels"
45     REFERENCE
46       "Table 349 and Table 353, in IEEE Std 802.16-2004"
47     ::= { wmanIfBsOfdmaPhy 1 }
48
49
50
51 wmanIfBsOfdmaUplinkChannelEntry OBJECT-TYPE
52     SYNTAX      WmanIfBsOfdmaUplinkChannelEntry
53     MAX-ACCESS  not-accessible
54     STATUS      current
55     DESCRIPTION
56       "This table provides one row for each uplink channel of
57       multi-sector BS, and is indexed by BS ifIndex. An entry
58       in this table exists for each ifEntry of BS with an
59       ifType of propBWA2Mp."
60     INDEX      { ifIndex }
61     ::= { wmanIfBsOfdmaUplinkChannelTable 1 }
62
63
64 WmanIfBsOfdmaUplinkChannelEntry ::= SEQUENCE {
65

```

```

1      wmanIfBsOfdmaCtBasedResvTimeout      INTEGER,
2      wmanIfBsOfdmaBwReqOppSize            INTEGER,
3      wmanIfBsOfdmaRangReqOppSize          INTEGER,
4      wmanIfBsOfdmaUplinkCenterFreq        Unsigned32,
5      wmanIfBsOfdmaInitRngCodes            INTEGER,
6      wmanIfBsOfdmaPeriodicRngCodes        INTEGER,
7      wmanIfBsOfdmaBWReqCodes              INTEGER,
8      wmanIfBsOfdmaPerRngBackoffStart       INTEGER,
9      wmanIfBsOfdmaPerRngBackoffEnd         INTEGER,
10     wmanIfBsOfdmaStartOfRngCodes          INTEGER,
11     wmanIfBsOfdmaPermutationBase          INTEGER,
12     wmanIfBsOfdmaULAllocSubchBitmap        OCTET STRING,
13     wmanIfBsOfdmaOptPermULAllocSubchBitmap OCTET STRING,
14     wmanIfBsOfdmaBandAMCAllocThreshold    INTEGER,
15     wmanIfBsOfdmaBandAMCReleaseThreshold  INTEGER,
16     wmanIfBsOfdmaBandAMCAllocTimer         INTEGER,
17     wmanIfBsOfdmaBandAMCReleaseTimer       INTEGER,
18     wmanIfBsOfdmaBandStatRepMAXPeriod      INTEGER,
19     wmanIfBsOfdmaBandAMCRetryTimer         INTEGER,
20     wmanIfBsOfdmaSafetyChAllocThreshold    INTEGER,
21     wmanIfBsOfdmaSafetyChReleaseThreshold  INTEGER,
22     wmanIfBsOfdmaSafetyChAllocTimer        INTEGER,
23     wmanIfBsOfdmaSafetyChReleaseTimer      INTEGER,
24     wmanIfBsOfdmaBinStatRepMAXPeriod       INTEGER,
25     wmanIfBsOfdmaSafetyChaRetryTimer       INTEGER,
26     wmanIfBsOfdmaHARQAackDelayULBurst      INTEGER,
27     wmanIfBsOfdmaCQICHBandAMCTranaDelay    INTEGER}
28
29
30
31
32
33
34
35     wmanIfBsOfdmaCtBasedResvTimeout OBJECT-TYPE
36         SYNTAX      INTEGER (1..255)
37         MAX-ACCESS   read-write
38         STATUS       current
39         DESCRIPTION
40             "The number of UL-MAPS to receive before contention-based
41             reservation is attempted again for the same connection."
42         REFERENCE
43             "Table 349, in IEEE Std 802.16-2004"
44             ::= { wmanIfBsOfdmaUplinkChannelEntry 1 }
45
46
47
48     wmanIfBsOfdmaBwReqOppSize OBJECT-TYPE
49         SYNTAX      INTEGER (1..65535)
50         UNITS       "PS"
51         MAX-ACCESS   read-write
52         STATUS       current
53         DESCRIPTION
54             "Size (in units of PS) of PHY payload that SS may use to
55             format and transmit a bandwidth request message in a
56             contention request opportunity. The value includes all
57             PHY overhead as well as allowance for the MAC data the
58             message may hold."
59         REFERENCE
60             "Table 349, in IEEE Std 802.16-2004"
61             ::= { wmanIfBsOfdmaUplinkChannelEntry 2 }
62
63
64
65

```

```

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

```



```

1  wmanIfBsOfdmaBWReqCodes OBJECT-TYPE
2      SYNTAX      INTEGER (0..255)
3      MAX-ACCESS  read-write
4      STATUS      current
5      DESCRIPTION
6          "Number of bandwidth request codes. Possible values are
7              0..255. The total number of wmanIfBsOfdmaInitRngCodes,
8              wmanIfBsOfdmaPeriodicRngCodes and wmanIfBsOfdmaBWReqCodes
9              shall be equal or less than 256."
10
11      REFERENCE
12          "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
13
14      DEFVAL      { 30 }
15      ::= { wmanIfBsOfdmaUplinkChannelEntry 7 }
16
17
18  wmanIfBsOfdmaPerRngBackoffStart OBJECT-TYPE
19      SYNTAX      INTEGER (0..15)
20      MAX-ACCESS  read-write
21      STATUS      current
22      DESCRIPTION
23          "Initial backoff window size for periodic ranging contention,
24              , expressed as a power of 2. Range: 0..15 (the highest order
25              bits shall be unused and set to 0)."
26
27      REFERENCE
28          "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
29
30      DEFVAL      { 0 }
31      ::= { wmanIfBsOfdmaUplinkChannelEntry 8 }
32
33
34  wmanIfBsOfdmaPerRngBackoffEnd OBJECT-TYPE
35      SYNTAX      INTEGER (0 .. 15)
36      MAX-ACCESS  read-write
37      STATUS      current
38      DESCRIPTION
39          "Final backoff window size for periodic ranging contention,
40              expressed as a power of 2. Range: 0..15 (the highest order
41              bits shall be unused and set to 0)."
42
43      REFERENCE
44          "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
45
46      DEFVAL      { 15 }
47      ::= { wmanIfBsOfdmaUplinkChannelEntry 9 }
48
49
50  wmanIfBsOfdmaStartOfRngCodes OBJECT-TYPE
51      SYNTAX      INTEGER (0..255)
52      MAX-ACCESS  read-write
53      STATUS      current
54      DESCRIPTION
55          "Indicates the starting number, S, of the group of codes
56              used for this uplink. All the ranging codes used on this
57              uplink will be between S and ((S+N+M+L) mod 256). Where,
58              N is the number of initial-ranging codes M is the number
59              of periodic-ranging codes L is the number of
60              bandwidth-request codes The range of values is 0 S255"
61
62      REFERENCE
63          "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
64
65      DEFVAL      { 0 }

```

```

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

```

```

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

```

```

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

```

```

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

```

```

1      MAX-ACCESS    not-accessible
2      STATUS        current
3      DESCRIPTION
4          "This table contains DCD channel attributes, defining the
5          transmission characteristics of downlink channels"
6      REFERENCE
7          "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
8      ::= { wmanIfBsOfdmaPhy 2 }
9
10
11
12  wmanIfBsOfdmaDownlinkChannelEntry OBJECT-TYPE
13      SYNTAX          WmanIfBsOfdmaDownlinkChannelEntry
14      MAX-ACCESS      not-accessible
15      STATUS          current
16      DESCRIPTION
17          "This table provides one row for each downlink channel of
18          multi-sector BS, and is indexed by BS ifIndex. An entry in
19          this table exists for each ifEntry of BS with an ifType of
20          propBWA2Mp."
21      INDEX           { ifIndex }
22      ::= { wmanIfBsOfdmaDownlinkChannelTable 1 }
23
24
25
26  WmanIfBsOfdmaDownlinkChannelEntry ::= SEQUENCE {
27      wmanIfBsOfdmaBsEIRP                INTEGER,
28      wmanIfBsOfdmaChannelNumber          WmanIfChannelNumber,
29      wmanIfBsOfdmaTTG                    INTEGER,
30      wmanIfBsOfdmaRTG                    INTEGER,
31      wmanIfBsOfdmaInitRngMaxRSS           INTEGER,
32      wmanIfBsOfdmaDownlinkCenterFreq     Unsigned32,
33      wmanIfBsOfdmaBsId                   WmanIfBsIdType,
34      wmanIfBsOfdmaMacVersion              WmanIfMacVersion,
35      wmanIfBsOfdmaFrameDurationCode      INTEGER,
36      wmanIfBsOfdmaSizeCqichIdField       INTEGER,
37      wmanIfBsOfdmaHARQAackDelayBurst     INTEGER}
38
39
40
41
42  wmanIfBsOfdmaBsEIRP OBJECT-TYPE
43      SYNTAX          INTEGER (0..65535)
44      UNITS            "dBm"
45      MAX-ACCESS      read-write
46      STATUS          current
47      DESCRIPTION
48          "The EIRP is the equivalent isotropic radiated power of
49          the base station, which is computed for a simple
50          single-antenna transmitter."
51      REFERENCE
52          "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
53      ::= { wmanIfBsOfdmaDownlinkChannelEntry 1 }
54
55
56
57  wmanIfBsOfdmaChannelNumber OBJECT-TYPE
58      SYNTAX          WmanIfChannelNumber
59      MAX-ACCESS      read-write
60      STATUS          current
61      DESCRIPTION
62          "Downlink channel number as defined in 8.5. Used for
63          license-exempt operation only."
64
65

```

```

1      REFERENCE
2      "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
3      ::= { wmanIfBsOfdmaDownlinkChannelEntry 2 }
4
5
6  wmanIfBsOfdmaTTG OBJECT-TYPE
7      SYNTAX      INTEGER (0..255)
8      MAX-ACCESS  read-write
9      STATUS      current
10     DESCRIPTION
11         "Transmit / Receive Transition Gap."
12     REFERENCE
13         "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
14     ::= { wmanIfBsOfdmaDownlinkChannelEntry 3 }
15
16
17
18  wmanIfBsOfdmaRTG OBJECT-TYPE
19      SYNTAX      INTEGER (0..255)
20      MAX-ACCESS  read-write
21      STATUS      current
22      DESCRIPTION
23         "Receive / Transmit Transition Gap."
24     REFERENCE
25         "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
26     ::= { wmanIfBsOfdmaDownlinkChannelEntry 4 }
27
28
29
30  wmanIfBsOfdmaInitRngMaxRSS OBJECT-TYPE
31      SYNTAX      INTEGER (0..65535)
32      UNITS       "dBm"
33      MAX-ACCESS  read-write
34      STATUS      current
35      DESCRIPTION
36         "Initial Ranging Max. Received Signal Strength at BS
37         Signed in units of 1 dBm."
38     REFERENCE
39         "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
40     ::= { wmanIfBsOfdmaDownlinkChannelEntry 5 }
41
42
43
44  wmanIfBsOfdmaDownlinkCenterFreq OBJECT-TYPE
45      SYNTAX      Unsigned32
46      UNITS       "kHz"
47      MAX-ACCESS  read-write
48      STATUS      current
49      DESCRIPTION
50         "Downlink center frequency (kHz)."
51     REFERENCE
52         "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
53     ::= { wmanIfBsOfdmaDownlinkChannelEntry 6 }
54
55
56
57  wmanIfBsOfdmaBsId OBJECT-TYPE
58      SYNTAX      WmanIfBsIdType
59      MAX-ACCESS  read-write
60      STATUS      current
61      DESCRIPTION
62         "Base station ID."
63     REFERENCE
64
65

```

```

1         "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
2         ::= { wmanIfBsOfdmaDownlinkChannelEntry 7 }
3
4
5 wmanIfBsOfdmaMacVersion OBJECT-TYPE
6     SYNTAX      WmanIfMacVersion
7     MAX-ACCESS  read-write
8     STATUS      current
9     DESCRIPTION
10        "This parameter specifies the version of 802.16 to which
11         the message originator conforms."
12
13     REFERENCE
14        "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
15        ::= { wmanIfBsOfdmaDownlinkChannelEntry 8 }
16
17
18 wmanIfBsOfdmaFrameDurationCode OBJECT-TYPE
19     SYNTAX      INTEGER { aASGap(0),
20                          duration2ms(1),
21                          duration2dot5ms(2),
22                          duration4ms(3),
23                          duration5ms(4),
24                          duration8ms(5),
25                          duration10ms(6),
26                          duration12dot5ms(7),
27                          duration20ms(8) }
28
29     MAX-ACCESS  read-write
30     STATUS      current
31     DESCRIPTION
32        "The duration of the frame. The frame duration code values
33         are specified in Table 274."
34
35     REFERENCE
36        "Table 273, in IEEE Std 802.16-2004"
37        ::= { wmanIfBsOfdmaDownlinkChannelEntry 9 }
38
39
40
41 wmanIfBsOfdmaSizeCqichIdField OBJECT-TYPE
42     SYNTAX      INTEGER { threebits(1),
43                          fourbits(2),
44                          fivebits(3),
45                          sixbits(4),
46                          sevenbits(5),
47                          eightbits(6),
48                          ninebits(7) }
49
50     MAX-ACCESS  read-write
51     STATUS      current
52     DESCRIPTION
53        "This object defines the size of CQICH ID field.
54         0 = Reserved
55         1 = 3 bits
56         2 = 4 bits
57         3 = 5 bits
58         4 = 6 bits
59         5 = 7 bits
60         6 = 8 bits
61         7 = 9 bits
62         8...255 = Reserved"
63
64
65

```



```

1      REFERENCE
2          "Subclause 11.3.1, Table 358, in IEEE Std 802.16-2004"
3      ::= { wmanIfBsOfdmaDownlinkChannelEntry 10 }
4
5
6      wmanIfBsOfdmaHARQAackDelayBurst OBJECT-TYPE
7          SYNTAX      INTEGER {oneframeoffset(1),
8                          twoframesoffset(2),
9                          threeframesoffset(3)}
10
11          MAX-ACCESS  read-write
12          STATUS      current
13          DESCRIPTION
14              "This object defines the OFDMA H-ARQ ACK delay for DL burst.
15               1 = one frame offset
16               2 = two frames offset
17               3 = three frames offset"
18
19          REFERENCE
20              "Subclause 11.3.1, Table 358, in IEEE Std 802.16-2004"
21          ::= { wmanIfBsOfdmaDownlinkChannelEntry 11 }
22
23
24      wmanIfBsOfdmaUcdBurstProfileTable OBJECT-TYPE
25          SYNTAX      SEQUENCE OF WmanIfBsOfdmaUcdBurstProfileEntry
26          MAX-ACCESS  not-accessible
27          STATUS      current
28          DESCRIPTION
29              "This table contains UCD burst profiles for each uplink
30               channel"
31
32          REFERENCE
33              "Table 356, in IEEE Std 802.16-2004"
34          ::= { wmanIfBsOfdmaPhy 3 }
35
36
37      wmanIfBsOfdmaUcdBurstProfileEntry OBJECT-TYPE
38          SYNTAX      WmanIfBsOfdmaUcdBurstProfileEntry
39          MAX-ACCESS  not-accessible
40          STATUS      current
41          DESCRIPTION
42              "This table provides one row for each UCD burst profile.
43               This table is double indexed. The primary index is an
44               ifIndex with an ifType of propBWAmp2Mp. The secondary index
45               is wmanIfBsOfdmaUiucIndex."
46
47          INDEX      { ifIndex, wmanIfBsOfdmaUiucIndex }
48          ::= { wmanIfBsOfdmaUcdBurstProfileTable 1 }
49
50
51      WmanIfBsOfdmaUcdBurstProfileEntry ::= SEQUENCE {
52          wmanIfBsOfdmaUiucIndex      INTEGER,
53          wmanIfBsOfdmaUcdFecCodeType WmanIfOfdmaFecCodeType,
54          wmanIfBsOfdmaRangingDataRatio      INTEGER,
55          wmanIfBsOfdmaNorCOVerNOVERRIDE      OCTET STRING,
56          wmanIfBsOfdmaUcdBurstProfileRowStatus      RowStatus}
57
58
59      wmanIfBsOfdmaUiucIndex OBJECT-TYPE
60          SYNTAX      INTEGER (1 .. 10)
61          MAX-ACCESS  not-accessible
62          STATUS      current
63          DESCRIPTION
64
65

```

```

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

```

```

1          MIB has choosen not to implement 'dynamic assignment' of
2          profiles, this object is not useful and should return
3          noSuchName upon SNMP request."
4      ::= { wmanIfBsOfdmaUcdBurstProfileEntry 5 }
5
6
7      wmanIfBsOfdmaDcdBurstProfileTable OBJECT-TYPE
8          SYNTAX      SEQUENCE OF WmanIfBsOfdmaDcdBurstProfileEntry
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 propBWA2Mp. The secondary index
15             is wmanIfBsOfdmaDiucIndex."
16         ::= { wmanIfBsOfdmaPhy 4 }
17
18
19
20      wmanIfBsOfdmaDcdBurstProfileEntry OBJECT-TYPE
21          SYNTAX      WmanIfBsOfdmaDcdBurstProfileEntry
22          MAX-ACCESS   not-accessible
23          STATUS      current
24          DESCRIPTION
25              "This table provides one row for each DCD burst profile.
26              This table is double indexed. The primary index is an
27              ifIndex with an ifType of propBWA2Mp. The secondary index
28              is wmanIfBsOfdmaDiucIndex."
29          INDEX       { ifIndex, wmanIfBsOfdmaDiucIndex }
30          ::= { wmanIfBsOfdmaDcdBurstProfileTable 1 }
31
32
33
34      WmanIfBsOfdmaDcdBurstProfileEntry ::= SEQUENCE {
35          wmanIfBsOfdmaDiucIndex          INTEGER,
36          wmanIfBsOfdmaDownlinkFrequency  Unsigned32,
37          wmanIfBsOfdmaDcdFecCodeType     WmanIfOfdmaFecCodeType,
38          wmanIfBsOfdmaDiucMandatoryExitThresh  INTEGER,
39          wmanIfBsOfdmaDiucMinEntryThresh  INTEGER,
40          wmanIfBsOfdmaDcdBurstProfileRowStatus  RowStatus}
41
42
43
44      wmanIfBsOfdmaDiucIndex OBJECT-TYPE
45          SYNTAX      INTEGER (0 .. 12)
46          MAX-ACCESS   not-accessible
47          STATUS      current
48          DESCRIPTION
49              "The Downlink Interval Usage Code indicates the downlink
50              burst profile in the DCD message, and is used along with
51              ifIndex to identify an entry in the
52              wmanIfBsOfdmaDcdBurstProfileTable."
53          REFERENCE
54              "Subclause 8.4.5.3.1, in IEEE Std 802.16-2004"
55          ::= { wmanIfBsOfdmaDcdBurstProfileEntry 1 }
56
57
58
59
60      wmanIfBsOfdmaDownlinkFrequency OBJECT-TYPE
61          SYNTAX      Unsigned32
62          UNITS        "kHz"
63          MAX-ACCESS   read-create
64          STATUS      current
65

```

```

1      DESCRIPTION
2          "Downlink Frequency (kHz)."
```

3

```
4      REFERENCE
5          "Subclause 11.4.2, Table 359, in IEEE Std 802.16-2004"
```

6

```
7      ::= { wmanIfBsOfdmaDcdBurstProfileEntry 2 }
```

7

```
8      wmanIfBsOfdmaDcdFecCodeType OBJECT-TYPE
9          SYNTAX      WmanIfOfdmaFecCodeType
10         MAX-ACCESS   read-create
11         STATUS      current
12         DESCRIPTION
13             "Downlink FEC code type and modulation type"
```

14

```
15         REFERENCE
16             "Subclause 11.4.2, Table 363, in IEEE Std 802.16-2004"
```

17

```
18         ::= { wmanIfBsOfdmaDcdBurstProfileEntry 3 }
```

19

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

29

```
30         REFERENCE
31             "Subclause 11.4.2, Table 363, in IEEE Std 802.16-2004"
```

32

```
33         ::= { wmanIfBsOfdmaDcdBurstProfileEntry 4 }
```

34

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

43

```
44         REFERENCE
45             "Subclause 11.4.2, Table 363, in IEEE Std 802.16-2004"
```

46

```
47         ::= { wmanIfBsOfdmaDcdBurstProfileEntry 5 }
```

48

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

59

```
60         ::= { wmanIfBsOfdmaDcdBurstProfileEntry 6 }
```

61

```
62
63
64
65     --
```

```

1  -- SS object group - containing tables and objects to be implemented in
2  -- the Subscriber station
3
4
5  --
6  -- wmanIfSsCps contain the SS Common Part Sublayer objects
7  --
8  wmanIfSsCps OBJECT IDENTIFIER ::= { wmanIfSsObjects 1 }
9
10
11
12 -- wmanIfSsConfigurationTable contains global parameters for SS
13 --
14 wmanIfSsConfigurationTable OBJECT-TYPE
15     SYNTAX      SEQUENCE OF WmanIfSsConfigurationEntry
16     MAX-ACCESS  not-accessible
17     STATUS      current
18     DESCRIPTION
19         "This table contains one row for the SS system
20         parameters."
21     REFERENCE
22         "Subclause 10.1 in IEEE Std 802.16-2004"
23     ::= { wmanIfSsCps 1 }
24
25
26
27 wmanIfSsConfigurationEntry OBJECT-TYPE
28     SYNTAX      WmanIfSsConfigurationEntry
29     MAX-ACCESS  not-accessible
30     STATUS      current
31     DESCRIPTION
32         "This table is indexed by ifIndex."
33     INDEX { ifIndex }
34     ::= { wmanIfSsConfigurationTable 1 }
35
36
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,
57     wmanIfSsT14Timeout             INTEGER,
58     wmanIfSsT16Timeout             INTEGER,
59     wmanIfSsT18Timeout             INTEGER,
60     wmanIfSsT19Timeout             INTEGER,

```

```

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

```

```

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

```

```

1      DESCRIPTION
2          "The retry period to re-establishing the Time of Day, as
3          describe in the network entry procedure."
4      ::= { wmanIfSsConfigurationEntry 12 }
5
6
7      wmanIfSsT1Timeout OBJECT-TYPE
8          SYNTAX      INTEGER (0..50000)
9          UNITS        "milliseconds"
10         MAX-ACCESS   read-write
11         STATUS       current
12         DESCRIPTION
13             "Wait for DCD timeout in ms."
14         ::= { wmanIfSsConfigurationEntry 13 }
15
16
17
18         wmanIfSsT2Timeout OBJECT-TYPE
19             SYNTAX      INTEGER (0..10000)
20             UNITS        "milliseconds"
21             MAX-ACCESS   read-write
22             STATUS       current
23             DESCRIPTION
24                 "Wait for broadcast ranging timeout in ms."
25             ::= { wmanIfSsConfigurationEntry 14 }
26
27
28
29         wmanIfSsT3Timeout OBJECT-TYPE
30             SYNTAX      INTEGER (0..200)
31             UNITS        "milliseconds"
32             MAX-ACCESS   read-write
33             STATUS       current
34             DESCRIPTION
35                 "Ranging Response reception timeout following the
36                 transmission of a Ranging Request in ms."
37             ::= { wmanIfSsConfigurationEntry 15 }
38
39
40
41         wmanIfSsT4Timeout OBJECT-TYPE
42             SYNTAX      INTEGER (30..35)
43             UNITS        "seconds"
44             MAX-ACCESS   read-write
45             STATUS       current
46             DESCRIPTION
47                 "Wait for unicast ranging opportunity. If the pending until
48                 complete field was used earlier by this SS, then the value
49                 of that field shall be added to this interval in second."
50             ::= { wmanIfSsConfigurationEntry 16 }
51
52
53
54         wmanIfSsT6Timeout OBJECT-TYPE
55             SYNTAX      INTEGER (0..3000)
56             UNITS        "milliseconds"
57             MAX-ACCESS   read-write
58             STATUS       current
59             DESCRIPTION
60                 "Wait for registration response in ms."
61             ::= { wmanIfSsConfigurationEntry 17 }
62
63
64
65         wmanIfSsT12Timeout OBJECT-TYPE

```



```

1          SYNTAX      INTEGER (0..50000)
2          UNITS        "milliseconds"
3          MAX-ACCESS   read-write
4          STATUS       current
5          DESCRIPTION
6              "Wait for UCD descriptor in ms."
7          ::= { wmanIfSsConfigurationEntry 18 }
8
9
10         wmanIfSsTl4Timeout OBJECT-TYPE
11             SYNTAX      INTEGER (0..200)
12             UNITS        "milliseconds"
13             MAX-ACCESS   read-write
14             STATUS       current
15             DESCRIPTION
16                 "Wait for DSX-RVD Timeout in ms."
17             ::= { wmanIfSsConfigurationEntry 19 }
18
19
20
21         wmanIfSsTl6Timeout OBJECT-TYPE
22             SYNTAX      INTEGER (10..65535)
23             UNITS        "milliseconds"
24             MAX-ACCESS   read-write
25             STATUS       current
26             DESCRIPTION
27                 "wait for bandwidth request grant in ms."
28             ::= { wmanIfSsConfigurationEntry 20 }
29
30
31
32         wmanIfSsTl8Timeout OBJECT-TYPE
33             SYNTAX      INTEGER (0..65535)
34             UNITS        "milliseconds"
35             MAX-ACCESS   read-write
36             STATUS       current
37             DESCRIPTION
38                 "wait for SBC-RSP timeout in ms."
39             ::= { wmanIfSsConfigurationEntry 21 }
40
41
42
43         wmanIfSsTl9Timeout OBJECT-TYPE
44             SYNTAX      INTEGER (0..1048575)
45             UNITS        "milliseconds"
46             MAX-ACCESS   read-write
47             STATUS       current
48             DESCRIPTION
49                 "Time DL-channel remains unusable in ms."
50             ::= { wmanIfSsConfigurationEntry 22 }
51
52
53
54         wmanIfSsT20Timeout OBJECT-TYPE
55             SYNTAX      INTEGER (0..65535)
56             UNITS        "milliseconds"
57             MAX-ACCESS   read-write
58             STATUS       current
59             DESCRIPTION
60                 "Time SS searches for preambles on a given channel in ms."
61             ::= { wmanIfSsConfigurationEntry 23 }
62
63
64
65         wmanIfSsT21Timeout OBJECT-TYPE

```

```

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

```

```

1      REFERENCE
2          "6.3.2.3.33 in IEEE Std 802.16-2004"
3      ::= { wmanIfSsCps 2 }
4
5
6      wmanIfSsChannelMeasurementEntry OBJECT-TYPE
7          SYNTAX      WmanIfSsChannelMeasurementEntry
8          MAX-ACCESS   not-accessible
9          STATUS       current
10         DESCRIPTION
11             "Each entry in the table contains RSSI and CINR
12             signal quality measurement taken from the SS. The primary
13             index is the ifIndex pointing to SS.
14             wmanIfCmnHistogramIndex is the index to histogram
15             samples. Since there is no time stamp in the table,
16             wmanIfCmnHistogramIndex should be increased monotonically,
17             and wraps around when it reaches the limit.
18             When the measurement entry for a SS reaches the limit,
19             the oldest entry shall be deleted as the new entry is
20             added to the table."
21         INDEX          { ifIndex, wmanIfSsHistogramIndex }
22         ::= { wmanIfSsChannelMeasurementTable 1 }
23
24
25      WmanIfSsChannelMeasurementEntry ::= SEQUENCE {
26          wmanIfSsHistogramIndex      Unsigned32,
27          wmanIfSsChannelNumber        WmanIfChannelNumber,
28          wmanIfSsStartFrame           INTEGER,
29          wmanIfSsDuration             INTEGER,
30          wmanIfSsBasicReport          BITS,
31          wmanIfSsMeanCinrReport       INTEGER,
32          wmanIfSsStdDeviationCinrReport INTEGER,
33          wmanIfSsMeanRssiReport       INTEGER,
34          wmanIfSsStdDeviationRssiReport INTEGER}
35
36
37      wmanIfSsHistogramIndex OBJECT-TYPE
38          SYNTAX      Unsigned32 (1 .. 4294967295)
39          MAX-ACCESS   not-accessible
40          STATUS       current
41          DESCRIPTION
42              "wmanIfSsHistogramIndex identifies the histogram samples
43              in the table for each subscriber station."
44          ::= { wmanIfSsChannelMeasurementEntry 1 }
45
46
47      wmanIfSsChannelNumber OBJECT-TYPE
48          SYNTAX      WmanIfChannelNumber
49          MAX-ACCESS   read-only
50          STATUS       current
51          DESCRIPTION
52              "Physical channel number to be reported on."
53          REFERENCE
54              "Subclause 8.5.1 in IEEE Std 802.16-2004"
55          ::= { wmanIfSsChannelMeasurementEntry 2 }
56
57
58      wmanIfSsStartFrame OBJECT-TYPE
59          SYNTAX      INTEGER (0 .. 65535)
60
61
62
63
64
65

```

```

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

```

```

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

```

```

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

```

```

1         "The value of this object is the actual clock time for
2         expiration of the immediate predecessor of the most recent
3         authorization key for this FSM. If this FSM has only one
4         authorization key, then the value is the time of activation
5         of this FSM."
6         ::= { wmanIfSsPkmAuthEntry 3 }
7
8
9
10        wmanIfSsPkmAuthExpiresNew OBJECT-TYPE
11            SYNTAX      DateAndTime
12            MAX-ACCESS  read-only
13            STATUS      current
14            DESCRIPTION
15                "The value of this object is the actual clock time for
16                expiration of the most recent authorization key for this
17                FSM."
18            ::= { wmanIfSsPkmAuthEntry 4 }
19
20
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
33
34        wmanIfSsPkmAuthentInfos OBJECT-TYPE
35            SYNTAX      Counter32
36            MAX-ACCESS  read-only
37            STATUS      current
38            DESCRIPTION
39                "The value of this object is the count of times the SS has
40                transmitted an Authentication Information message."
41            ::= { wmanIfSsPkmAuthEntry 6 }
42
43
44
45        wmanIfSsPkmAuthRequests OBJECT-TYPE
46            SYNTAX      Counter32
47            MAX-ACCESS  read-only
48            STATUS      current
49            DESCRIPTION
50                "The value of this object is the count of times the SS has
51                transmitted an Authorization Request message."
52            ::= { wmanIfSsPkmAuthEntry 7 }
53
54
55
56        wmanIfSsPkmAuthReplies OBJECT-TYPE
57            SYNTAX      Counter32
58            MAX-ACCESS  read-only
59            STATUS      current
60            DESCRIPTION
61                "The value of this object is the count of times the SS has
62                received an Authorization Reply message."
63            ::= { wmanIfSsPkmAuthEntry 8 }
64
65

```

```

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

```



```

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

```

```

1      STATUS      current
2      DESCRIPTION
3          "The value of this object is the Authorize Wait Timeout."
4      REFERENCE
5          "Table 341 in IEEE Std 802.16-2004"
6      DEFVAL      { 10 }
7      ::= { wmanIfSsPkmAuthEntry 17 }
8
9
10
11  wmanIfSsPkmReauthWaitTimeout OBJECT-TYPE
12      SYNTAX      Integer32 (2..30)
13      UNITS       "seconds"
14      MAX-ACCESS  read-only
15      STATUS      current
16      DESCRIPTION
17          "The value of this object is the Reauthorize Wait Timeout
18           in seconds."
19      REFERENCE
20          "Table 341 in IEEE Std 802.16-2004"
21      DEFVAL      { 10 }
22      ::= { wmanIfSsPkmAuthEntry 18 }
23
24
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
40
41
42  wmanIfSsPkmRekeyWaitTimeout OBJECT-TYPE
43      SYNTAX      Integer32 (1..10)
44      UNITS       "seconds"
45      MAX-ACCESS  read-only
46      STATUS      current
47      DESCRIPTION
48          "The value of this object is the Rekey Wait Timeout in
49           seconds."
50      REFERENCE
51          "Table 341 in IEEE Std 802.16-2004"
52      DEFVAL      { 1 }
53      ::= { wmanIfSsPkmAuthEntry 20 }
54
55
56
57  wmanIfSsPkmAuthRejectWaitTimeout OBJECT-TYPE
58      SYNTAX      Integer32 (10..600)
59      UNITS       "seconds"
60      MAX-ACCESS  read-only
61      STATUS      current
62      DESCRIPTION
63          "The value of this object is the Authorization Reject Wait
64
65

```

```

1          Timeout in seconds."
2  REFERENCE
3          "Table 341 in IEEE Std 802.16-2004"
4  DEFVAL      { 60 }
5  ::= { wmanIfSsPkmAuthEntry 21 }
6
7
8  --
9  -- Table wmanIfSsPkmTekTable
10 --
11
12 wmanIfSsPkmTekTable OBJECT-TYPE
13     SYNTAX      SEQUENCE OF WmanIfSsPkmTekEntry
14     MAX-ACCESS  not-accessible
15     STATUS      current
16     DESCRIPTION
17         "This table describes the attributes of each SS Traffic
18         Encryption Key (TEK) association. The SS maintains (no more
19         than) one TEK association per SAID per SS wireless
20         interface."
21     ::= { wmanIfSsPkmObjects 2 }
22
23
24
25 wmanIfSsPkmTekEntry OBJECT-TYPE
26     SYNTAX      WmanIfSsPkmTekEntry
27     MAX-ACCESS  not-accessible
28     STATUS      current
29     DESCRIPTION
30         "Each entry contains objects describing the TEK association
31         attributes of one SAID. The SS MUST create one entry per
32         SAID, regardless of whether the SAID was obtained from a
33         Registration Response message, from an Authorization Reply
34         message, or from any dynamic SAID establishment
35         mechanisms."
36     INDEX      { ifIndex, wmanIfSsPkmTekSAId }
37     ::= { wmanIfSsPkmTekTable 1 }
38
39
40
41
42 WmanIfSsPkmTekEntry ::= SEQUENCE {
43     wmanIfSsPkmTekSAId          INTEGER,
44     wmanIfSsPkmTekSAType        INTEGER,
45     wmanIfSsPkmTekDataEncryptAlg WmanIfDataEncryptAlgId,
46     wmanIfSsPkmTekDataAuthAlg   WmanIfDataAuthAlgId,
47     wmanIfSsPkmTekEncryptAlg     WmanIfTekEncryptAlgId,
48     wmanIfSsPkmTekState          INTEGER,
49     wmanIfSsPkmTekKeySequenceNumber Integer32,
50     wmanIfSsPkmTekExpiresOld     DateAndTime,
51     wmanIfSsPkmTekExpiresNew     DateAndTime,
52     wmanIfSsPkmTekKeyRequests    Counter32,
53     wmanIfSsPkmTekKeyReplies     Counter32,
54     wmanIfSsPkmTekKeyRejects     Counter32,
55     wmanIfSsPkmTekInvalids       Counter32,
56     wmanIfSsPkmTekAuthPends      Counter32,
57     wmanIfSsPkmTekKeyRejectErrorCode INTEGER,
58     wmanIfSsPkmTekKeyRejectErrorString SnmpAdminString,
59     wmanIfSsPkmTekInvalidErrorCode INTEGER,
60     wmanIfSsPkmTekInvalidErrorString SnmpAdminString}
61
62
63
64
65

```

```

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

::= { wmanIfSsPkmTekEntry 1 }

```

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

::= { wmanIfSsPkmTekEntry 2 }

```

27 wmanIfSsPkmTekDataEncryptAlg OBJECT-TYPE
28     SYNTAX          WmanIfDataEncryptAlgId
29     MAX-ACCESS      read-only
30     STATUS          current
31     DESCRIPTION
32         "The value of this object is the data encryption algorithm
33             being utilized."
```

::= { wmanIfSsPkmTekEntry 3 }

```

40 wmanIfSsPkmTekDataAuthentAlg OBJECT-TYPE
41     SYNTAX          WmanIfDataAuthAlgId
42     MAX-ACCESS      read-only
43     STATUS          current
44     DESCRIPTION
45         "The value of this object is the data authentication
46             algorithm being utilized."
```

::= { wmanIfSsPkmTekEntry 4 }

```

53 wmanIfSsPkmTekEncryptAlg OBJECT-TYPE
54     SYNTAX          WmanIfTekEncryptAlgId
55     MAX-ACCESS      read-only
56     STATUS          current
57     DESCRIPTION
58         "The value of this object is the TEK key encryption
59             algorithm for this cryptographic suite capability."
```

::= { wmanIfSsPkmTekEntry 5 }

```

1
2  wmanIfSsPkmTekState OBJECT-TYPE
3      SYNTAX          INTEGER {start(1),
4                          opWait(2),
5                          opReauthWait(3),
6                          operational(4),
7                          rekeyWait(5),
8                          rekeyReauthWait(6)}
9
10     MAX-ACCESS      read-only
11     STATUS          current
12     DESCRIPTION
13         "The value of this object is the state of the indicated TEK
14         FSM. The start(1) state indicates that FSM is in its
15         initial state."
16     ::= { wmanIfSsPkmTekEntry 6 }
17
18
19
20  wmanIfSsPkmTekKeySequenceNumber 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 TEK FSM."
27      REFERENCE
28         "IEEE Std 802.16-2004; 11.9.5"
29      ::= { wmanIfSsPkmTekEntry 7 }
30
31
32
33  wmanIfSsPkmTekExpiresOld 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      ::= { wmanIfSsPkmTekEntry 8 }
43
44
45
46  wmanIfSsPkmTekExpiresNew 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      ::= { wmanIfSsPkmTekEntry 9 }
54
55
56
57  wmanIfSsPkmTekKeyRequests OBJECT-TYPE
58      SYNTAX          Counter32
59      MAX-ACCESS      read-only
60      STATUS          current
61      DESCRIPTION
62         "The value of this object is the count of times the SS has
63         transmitted a Key Request message."
64
65

```

```

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

```

```

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

```

```

1      DESCRIPTION
2          "Each entry contains the device certificate of one SS."
3      INDEX          { ifIndex }
4      ::= { wmanIfSsDeviceCertTable 1 }
5
6
7      WmanIfSsDeviceCertEntry ::= SEQUENCE {
8          wmanIfSsDeviceCert          OCTET STRING,
9          wmanIfSsDeviceManufCert     OCTET STRING}
10
11
12      wmanIfSsDeviceCert OBJECT-TYPE
13          SYNTAX          OCTET STRING (SIZE(0..65535))
14          MAX-ACCESS      read-only
15          STATUS          current
16          DESCRIPTION
17              "The X509 DER-encoded subscriber station certificate."
18          ::= { wmanIfSsDeviceCertEntry 1 }
19
20
21      wmanIfSsDeviceManufCert OBJECT-TYPE
22          SYNTAX          OCTET STRING (SIZE(0..65535))
23          MAX-ACCESS      read-only
24          STATUS          current
25          DESCRIPTION
26              "The X509 DER-encoded manufacturer certificate which is
27              signed by the CA root authority certificate."
28          ::= { wmanIfSsDeviceCertEntry 2 }
29
30
31      --
32      -- Subscriber station Notification Group
33      -- wmanIfSsNotificationObjects contains the SS SNMP Trap objects
34      --
35      wmanIfSsNotification OBJECT IDENTIFIER ::= { wmanIfSsObjects 3 }
36      wmanIfSsTrapControl OBJECT IDENTIFIER ::= { wmanIfSsNotification 1 }
37      wmanIfSsTrapDefinitions OBJECT IDENTIFIER ::= { wmanIfSsNotification 2 }
38
39      -- This object groups all NOTIFICATION-TYPE objects for SS.
40      -- It is defined following RFC2758 sections 8.5 and 8.6
41      -- for the compatibility with SNMPv1.
42      wmanIfSsTrapPrefix OBJECT IDENTIFIER ::= { wmanIfSsTrapDefinitions 0 }
43
44      wmanIfSsTrapControlRegister OBJECT-TYPE
45          SYNTAX          BITS {wmanIfSsTlvUnknown(0),
46                               wmanIfSsDynamicServiceFail(1),
47                               wmanIfSsDhcpSuccess(2),
48                               wmanIfSsRssiStatusChange(3)}
49          MAX-ACCESS      read-write
50          STATUS          current
51          DESCRIPTION
52              "The object is used to enable Subscriber Station traps.
53              From left to right, the set bit indicates the corresponding
54              Subscriber Station trap is enabled."
55          ::= { wmanIfSsTrapControl 1 }
56
57      wmanIfSsThresholdConfigTable OBJECT-TYPE
58          SYNTAX          SEQUENCE OF WmanIfSsThresholdConfigEntry
59
60
61
62
63
64
65

```



```

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

```

```

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

```

```

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

```

```

1          "This object provides additional information about RSSI
2          alarm. It is implementation specific"
3          ::= { wmanIfSsNotificationObjectsEntry 6 }
4
5
6      --
7      -- Subscriber station PHY Group
8      --
9      wmanIfSsPhy OBJECT IDENTIFIER ::= { wmanIfSsObjects 5 }
10
11
12      --
13      -- SS OFDM PHY objects
14      --
15      wmanIfSsOfdmPhy OBJECT IDENTIFIER ::= { wmanIfSsPhy 1 }
16
17      wmanIfSsOfdmUplinkChannelTable OBJECT-TYPE
18          SYNTAX      SEQUENCE OF WmanIfSsOfdmUplinkChannelEntry
19          MAX-ACCESS  not-accessible
20          STATUS      current
21          DESCRIPTION
22              "This table contains UCD channel attributes, defining the
23              transmission characteristics of uplink channels"
24          REFERENCE
25              "Table 349 and Table 352, in IEEE Std 802.16-2004"
26          ::= { wmanIfSsOfdmPhy 1 }
27
28      wmanIfSsOfdmUplinkChannelEntry OBJECT-TYPE
29          SYNTAX      WmanIfSsOfdmUplinkChannelEntry
30          MAX-ACCESS  not-accessible
31          STATUS      current
32          DESCRIPTION
33              "This table provides one row for each uplink channel of
34              multi-sector BS, and is indexed by BS ifIndex. An entry
35              in this table exists for each ifEntry of BS with an
36              ifType of propBWA2Mp."
37          INDEX { ifIndex }
38          ::= { wmanIfSsOfdmUplinkChannelTable 1 }
39
40      WmanIfSsOfdmUplinkChannelEntry ::= SEQUENCE {
41          wmanIfSsOfdmCtBasedResvTimeout      INTEGER,
42          wmanIfSsOfdmBwReqOppSize            INTEGER,
43          wmanIfSsOfdmRangReqOppSize          INTEGER,
44          wmanIfSsOfdmUplinkCenterFreq        Unsigned32,
45          wmanIfSsOfdmNumSubChReqRegionFull   INTEGER,
46          wmanIfSsOfdmNumSymbolsReqRegionFull INTEGER,
47          wmanIfSsOfdmSubChFocusCtCode        INTEGER,
48          wmanIfSsOfdmUpLinkChannelId         INTEGER}
49
50      wmanIfSsOfdmCtBasedResvTimeout OBJECT-TYPE
51          SYNTAX      INTEGER (1..255)
52          MAX-ACCESS  read-only
53          STATUS      current
54          DESCRIPTION
55              "The number of UL-MAPs to receive before contention-based
56              reservation is attempted again for the same connection."
57

```

```

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

```

```

1          "Number of subchannels used by each transmit
2          opportunity when REQ Region-Full is allocated in
3          subchannelization region."
4
5      REFERENCE
6          "Table 352, in IEEE Std 802.16-2004"
7      ::= { wmanIfSsOfdmUplinkChannelEntry 5 }
8
9
10     wmanIfSsOfdmNumSymbolsReqRegionFull OBJECT-TYPE
11         SYNTAX      INTEGER (0..31)
12         MAX-ACCESS   read-only
13         STATUS       current
14         DESCRIPTION
15             "Number of OFDM symbols used by each transmit
16             opportunity when REQ Region-Full is allocated in
17             subchannelization region."
18         REFERENCE
19             "Table 352, in IEEE Std 802.16-2004"
20         ::= { wmanIfSsOfdmUplinkChannelEntry 6 }
21
22
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
38
39     wmanIfSsOfdmUpLinkChannelId OBJECT-TYPE
40         SYNTAX      INTEGER (0..255)
41         MAX-ACCESS   read-only
42         STATUS       current
43         DESCRIPTION
44             "The identifier of the uplink channel to which this
45             message refers."
46         REFERENCE
47             "Subclause 6.3.2.3.4. Table 16, in IEEE Std 802.16-2004"
48         ::= { wmanIfSsOfdmUplinkChannelEntry 8 }
49
50
51
52     wmanIfSsOfdmDownlinkChannelTable OBJECT-TYPE
53         SYNTAX      SEQUENCE OF WmanIfSsOfdmDownlinkChannelEntry
54         MAX-ACCESS   not-accessible
55         STATUS       current
56         DESCRIPTION
57             "This table contains DCD channel attributes, defining the
58             transmission characteristics of downlink channels"
59         REFERENCE
60             "Table 358, in IEEE Std 802.16-2004"
61         ::= { wmanIfSsOfdmPhy 2 }
62
63
64
65

```

```

1  wmanIfSsOfdmDownlinkChannelEntry OBJECT-TYPE
2      SYNTAX      WmanIfSsOfdmDownlinkChannelEntry
3      MAX-ACCESS  not-accessible
4      STATUS      current
5      DESCRIPTION
6          "This table provides one row for each downlink 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     ::= { wmanIfSsOfdmDownlinkChannelTable 1 }
12
13  WmanIfSsOfdmDownlinkChannelEntry ::= SEQUENCE {
14      wmanIfSsOfdmBsEIRP                INTEGER,
15      wmanIfSsOfdmChannelNumber          WmanIfChannelNumber,
16      wmanIfSsOfdmTTG                    INTEGER,
17      wmanIfSsOfdmRTG                    INTEGER,
18      wmanIfSsOfdmInitRngMaxRSS          INTEGER,
19      wmanIfSsOfdmDownlinkCenterFreq     Unsigned32,
20      wmanIfSsOfdmBsId                   WmanIfBsIdType,
21      wmanIfSsOfdmMacVersion              WmanIfMacVersion,
22      wmanIfSsOfdmFrameDurationCode      INTEGER,
23      wmanIfSsOfdmDownLinkChannelId      INTEGER}
24
25  wmanIfSsOfdmBsEIRP OBJECT-TYPE
26      SYNTAX      INTEGER (0..65535)
27      UNITS       "dBm"
28      MAX-ACCESS  read-only
29      STATUS      current
30      DESCRIPTION
31          "The EIRP is the equivalent isotropic radiated power of
32          the base station, which is computed for a simple
33          single-antenna transmitter."
34      REFERENCE
35          "Table 358, in IEEE Std 802.16-2004"
36      ::= { wmanIfSsOfdmDownlinkChannelEntry 1 }
37
38  wmanIfSsOfdmChannelNumber OBJECT-TYPE
39      SYNTAX      WmanIfChannelNumber
40      MAX-ACCESS  read-only
41      STATUS      current
42      DESCRIPTION
43          "Downlink channel number as defined in 8.5.
44          Used for license-exempt operation only."
45      REFERENCE
46          "Table 358, in IEEE Std 802.16-2004"
47      ::= { wmanIfSsOfdmDownlinkChannelEntry 2 }
48
49  wmanIfSsOfdmTTG OBJECT-TYPE
50      SYNTAX      INTEGER (0..255)
51      MAX-ACCESS  read-only
52      STATUS      current
53      DESCRIPTION
54          "Transmit / Receive Transition Gap."

```

```

1      REFERENCE
2          "Table 358, in IEEE Std 802.16-2004"
3      ::= { wmanIfSsOfdmDownlinkChannelEntry 3 }
4
5
6  wmanIfSsOfdmRTG OBJECT-TYPE
7      SYNTAX      INTEGER (0..255)
8      MAX-ACCESS  read-only
9      STATUS      current
10     DESCRIPTION
11         "Receive / Transmit Transition Gap."
12     REFERENCE
13         "Table 358, in IEEE Std 802.16-2004"
14     ::= { wmanIfSsOfdmDownlinkChannelEntry 4 }
15
16
17
18  wmanIfSsOfdmInitRngMaxRSS OBJECT-TYPE
19      SYNTAX      INTEGER (0..65535)
20      UNITS       "dBm"
21      MAX-ACCESS  read-only
22      STATUS      current
23      DESCRIPTION
24          "Initial Ranging Max. Received Signal Strength at BS
25           Signed in units of 1 dBm."
26      REFERENCE
27          "Table 358, in IEEE Std 802.16-2004"
28      ::= { wmanIfSsOfdmDownlinkChannelEntry 5 }
29
30
31
32  wmanIfSsOfdmDownlinkCenterFreq OBJECT-TYPE
33      SYNTAX      Unsigned32
34      UNITS       "kHz"
35      MAX-ACCESS  read-only
36      STATUS      current
37      DESCRIPTION
38          "Downlink center frequency (kHz)."
39      REFERENCE
40          "Table 358, in IEEE Std 802.16-2004"
41      ::= { wmanIfSsOfdmDownlinkChannelEntry 6 }
42
43
44
45  wmanIfSsOfdmBsId OBJECT-TYPE
46      SYNTAX      WmanIfBsIdType
47      MAX-ACCESS  read-only
48      STATUS      current
49      DESCRIPTION
50          "Base station ID."
51      REFERENCE
52          "Table 358, in IEEE Std 802.16-2004"
53      ::= { wmanIfSsOfdmDownlinkChannelEntry 7 }
54
55
56
57  wmanIfSsOfdmMacVersion OBJECT-TYPE
58      SYNTAX      WmanIfMacVersion
59      MAX-ACCESS  read-only
60      STATUS      current
61      DESCRIPTION
62          "This parameter specifies the version of 802.16 to which
63           the message originator conforms."
64
65

```



```

1      REFERENCE
2          "Table 358, in IEEE Std 802.16-2004"
3      ::= { wmanIfSsOfdmDownlinkChannelEntry 8 }
4
5
6  wmanIfSsOfdmFrameDurationCode OBJECT-TYPE
7      SYNTAX      INTEGER {duration2dot5ms(0),
8                      duration4ms(1),
9                      duration5ms(2),
10                     duration8ms(3),
11                     duration10ms(4),
12                     duration12dot5ms(5),
13                     duration20ms(6)}
14
15      MAX-ACCESS  read-only
16      STATUS      current
17      DESCRIPTION
18          "The duration of the frame. The frame duration code
19          values are specified in Table 230."
20
21      REFERENCE
22          "Subclause 11.4.1, Table 230, in IEEE Std 802.16-2004"
23      ::= { wmanIfSsOfdmDownlinkChannelEntry 9 }
24
25
26  wmanIfSsOfdmDownLinkChannelId OBJECT-TYPE
27      SYNTAX      INTEGER (0..255)
28      MAX-ACCESS  read-only
29      STATUS      current
30      DESCRIPTION
31          "The identifier of the downlink channel to which this
32          message refers."
33
34      REFERENCE
35          "Subclause 6.3.2.3.1, Table 15, in IEEE Std 802.16-2004"
36      ::= { wmanIfSsOfdmDownlinkChannelEntry 10 }
37
38
39  wmanIfSsOfdmUcdBurstProfileTable OBJECT-TYPE
40      SYNTAX      SEQUENCE OF WmanIfSsOfdmUcdBurstProfileEntry
41      MAX-ACCESS  not-accessible
42      STATUS      current
43      DESCRIPTION
44          "This table contains UCD burst profiles for each uplink
45          channel"
46
47      REFERENCE
48          "Table 356, in IEEE Std 802.16-2004"
49      ::= { wmanIfSsOfdmPhy 3 }
50
51
52  wmanIfSsOfdmUcdBurstProfileEntry OBJECT-TYPE
53      SYNTAX      WmanIfSsOfdmUcdBurstProfileEntry
54      MAX-ACCESS  not-accessible
55      STATUS      current
56      DESCRIPTION
57          "This table provides one row for each UCD burst profile.
58          This table is double indexed. The primary index is an
59          ifIndex with an ifType of propBWAmp2Mp. The secondary index
60          is wmanIfSsOfdmOfdmUcdBurstProfIndex."
61
62      INDEX { ifIndex, wmanIfSsOfdmUiucIndex }
63      ::= { wmanIfSsOfdmUcdBurstProfileTable 1 }
64
65

```

```

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

```

```

1
2 wmanIfSsOfdmDcdBurstProfileTable OBJECT-TYPE
3     SYNTAX      SEQUENCE OF WmanIfSsOfdmDcdBurstProfileEntry
4     MAX-ACCESS  not-accessible
5     STATUS      current
6     DESCRIPTION
7         "This table provides one row for each DCD burst profile.
8         This table is double indexed. The primary index is an
9         ifIndex with an ifType of propBWAmp2Mp. The secondary
10        index is wmanIfSsOfdmDiucIndex."
11
12 REFERENCE
13     "Table 362, in IEEE Std 802.16-2004"
14 ::= { wmanIfSsOfdmPhy 4 }
15
16
17 wmanIfSsOfdmDcdBurstProfileEntry OBJECT-TYPE
18     SYNTAX      WmanIfSsOfdmDcdBurstProfileEntry
19     MAX-ACCESS  not-accessible
20     STATUS      current
21     DESCRIPTION
22         "This table provides one row for each DCD burst profile.
23         This table is double indexed. The primary index is an
24         ifIndex with an ifType of propBWAmp2Mp. The secondary index
25         is wmanIfSsOfdmDcdBurstProfIndex."
26
27 INDEX { ifIndex, wmanIfSsOfdmDiucIndex }
28 ::= { wmanIfSsOfdmDcdBurstProfileTable 1 }
29
30
31 WmanIfSsOfdmDcdBurstProfileEntry ::= SEQUENCE {
32     wmanIfSsOfdmDiucIndex          INTEGER,
33     wmanIfSsOfdmDownlinkFrequency Unsigned32,
34     wmanIfSsOfdmDcdFecCodeType     WmanIfOfdmFecCodeType,
35     wmanIfSsOfdmDiucMandatoryExitThresh INTEGER,
36     wmanIfSsOfdmDiucMinEntryThresh INTEGER,
37     wmanIfSsOfdmTcsEnable          INTEGER}
38
39
40 wmanIfSsOfdmDiucIndex OBJECT-TYPE
41     SYNTAX      INTEGER (1..11)
42     MAX-ACCESS  not-accessible
43     STATUS      current
44     DESCRIPTION
45         "The Downlink Interval Usage Code indicates the downlink
46         burst profile in the DCD message, and is used along with
47         ifIndex to identify an entry in the
48         wmanIfSsOfdmDcdBurstProfileTable."
49
50 REFERENCE
51     "Subclause 8.3.6.3.1, in IEEE Std 802.16-2004"
52 ::= { wmanIfSsOfdmDcdBurstProfileEntry 1 }
53
54
55 wmanIfSsOfdmDownlinkFrequency OBJECT-TYPE
56     SYNTAX      Unsigned32
57     UNITS       "kHz"
58     MAX-ACCESS  read-only
59     STATUS      current
60     DESCRIPTION
61         "Downlink Frequency (kHz)."
```

```

1      REFERENCE
2      "Table 362, in IEEE Std 802.16-2004"
3      ::= { wmanIfSsOfdmDcdBurstProfileEntry 2 }
4
5
6  wmanIfSsOfdmDcdFecCodeType OBJECT-TYPE
7      SYNTAX      WmanIfOfdmFecCodeType
8      MAX-ACCESS  read-only
9      STATUS      current
10     DESCRIPTION
11         "Downlink FEC code type and modulation type"
12     REFERENCE
13         "Table 362, in IEEE Std 802.16-2004"
14     ::= { wmanIfSsOfdmDcdBurstProfileEntry 3 }
15
16
17
18  wmanIfSsOfdmDiucMandatoryExitThresh OBJECT-TYPE
19      SYNTAX      INTEGER (0..255)
20      MAX-ACCESS  read-only
21      STATUS      current
22      DESCRIPTION
23          "DIUC mandatory exit threshold: 0 - 63.75 dB CINR at or
24          below where this DIUC can no longer be used and where this
25          change to a more robust DIUC is required in 0.25 dB units."
26      REFERENCE
27          "Table 362, in IEEE Std 802.16-2004"
28      ::= { wmanIfSsOfdmDcdBurstProfileEntry 4 }
29
30
31
32  wmanIfSsOfdmDiucMinEntryThresh OBJECT-TYPE
33      SYNTAX      INTEGER (0..255)
34      MAX-ACCESS  read-only
35      STATUS      current
36      DESCRIPTION
37          "DIUC minimum entry threshold: 0 - 63.75 dB The minimum CINR
38          required to start using this DIUC when changing from a more
39          robust DIUC is required, in 0.25 dB units."
40      REFERENCE
41          "Table 362, in IEEE Std 802.16-2004"
42      ::= { wmanIfSsOfdmDcdBurstProfileEntry 5 }
43
44
45
46  wmanIfSsOfdmTcsEnable OBJECT-TYPE
47      SYNTAX      INTEGER {tcsDisabled (0),
48                          tcsEnabled (1)}
49      MAX-ACCESS  read-only
50      STATUS      current
51      DESCRIPTION
52          "Indicates whether Transmission Convergence Sublayer
53          is enabled or disabled."
54      REFERENCE
55          "Table 362, in IEEE Std 802.16-2004"
56      ::= { wmanIfSsOfdmDcdBurstProfileEntry 6 }
57
58
59
60
61  --
62  -- SS OFDMA PHY objects
63  --
64  wmanIfSsOfdmaPhy OBJECT IDENTIFIER ::= { wmanIfSsPhy 2 }
65

```

```

1
2 wmanIfSsOfdmaUplinkChannelTable OBJECT-TYPE
3     SYNTAX      SEQUENCE OF WmanIfSsOfdmaUplinkChannelEntry
4     MAX-ACCESS  not-accessible
5     STATUS      current
6     DESCRIPTION
7         "This table contains UCD channel attributes, defining the
8         transmission characteristics of uplink channels"
9     REFERENCE
10        "Subclause 11.3.1, Table 349 and Table 353, in IEEE Std
11 802.16-2004"
12 ::= { wmanIfSsOfdmaPhy 1 }
13
14 wmanIfSsOfdmaUplinkChannelEntry OBJECT-TYPE
15     SYNTAX      WmanIfSsOfdmaUplinkChannelEntry
16     MAX-ACCESS  not-accessible
17     STATUS      current
18     DESCRIPTION
19         "This table provides one row for each uplink channel of
20         multi-sector BS, and is indexed by BS ifIndex. An entry
21         in this table exists for each ifEntry of BS with an
22         ifType of propBWA2Mp."
23     INDEX       { ifIndex }
24 ::= { wmanIfSsOfdmaUplinkChannelTable 1 }
25
26 WmanIfSsOfdmaUplinkChannelEntry ::= SEQUENCE {
27     wmanIfSsOfdmaCtBasedResvTimeout      INTEGER,
28     wmanIfSsOfdmaBwReqOppSize            INTEGER,
29     wmanIfSsOfdmaRangReqOppSize          INTEGER,
30     wmanIfSsOfdmaUplinkCenterFreq        Unsigned32,
31     wmanIfSsOfdmaInitRngCodes            INTEGER,
32     wmanIfSsOfdmaPeriodicRngCodes        INTEGER,
33     wmanIfSsOfdmaBWReqCodes              INTEGER,
34     wmanIfSsOfdmaPerRngBackoffStart       INTEGER,
35     wmanIfSsOfdmaPerRngBackoffEnd        INTEGER,
36     wmanIfSsOfdmaStartOfRngCodes         INTEGER,
37     wmanIfSsOfdmaPermutationBase         INTEGER,
38     wmanIfSsOfdmaULAllocSubchBitmap      OCTET STRING,
39     wmanIfSsOfdmaOptPermULAllocSubchBitmap OCTET STRING,
40     wmanIfSsOfdmaBandAMCAllocThreshold   INTEGER,
41     wmanIfSsOfdmaBandAMCReleaseThreshold INTEGER,
42     wmanIfSsOfdmaBandAMCAllocTimer       INTEGER,
43     wmanIfSsOfdmaBandAMCReleaseTimer     INTEGER,
44     wmanIfSsOfdmaBandStatRepMAXPeriod    INTEGER,
45     wmanIfSsOfdmaBandAMCRetryTimer       INTEGER,
46     wmanIfSsOfdmaSafetyChAllocThreshold INTEGER,
47     wmanIfSsOfdmaSafetyChReleaseThreshold INTEGER,
48     wmanIfSsOfdmaSafetyChAllocTimer     INTEGER,
49     wmanIfSsOfdmaSafetyChReleaseTimer    INTEGER,
50     wmanIfSsOfdmaBinStatRepMAXPeriod     INTEGER,
51     wmanIfSsOfdmaSafetyChARetryTimer     INTEGER,
52     wmanIfSsOfdmaHARQAackDelayULBurst    INTEGER,
53     wmanIfSsOfdmaCQICHBandAMCTranaDelay  INTEGER }
54
55
56
57
58
59
60
61
62
63
64
65

```

```

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

```

```

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

```

```

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

```



```

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

```

```

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

```

```

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

```

1                                     threeframesoffset(3)}
2
3     MAX-ACCESS    read-only
4     STATUS        current
5     DESCRIPTION
6         "This object defines the OFDMA H-ARQ ACK delay for UL burst.
7             1 = one frame offset
8             2 = two frames offset
9             3 = three frames offset"
10
11     REFERENCE
12         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
13         ::= { wmanIfSsOfdmaUplinkChannelEntry 26 }
14
15 wmanIfSsOfdmaCQICHBandAMCTranaDelay OBJECT-TYPE
16     SYNTAX        INTEGER (0 .. 255)
17     UNITS          "Frame"
18     MAX-ACCESS    read-only
19     STATUS        current
20     DESCRIPTION
21         "This object defines the OFDMA CQICH band AMC transition
22         delay."
23     REFERENCE
24         "Subclause 11.3.1, Table 353, in IEEE Std 802.16-2004"
25         ::= { wmanIfSsOfdmaUplinkChannelEntry 27 }
26
27 wmanIfSsOfdmaDownlinkChannelTable OBJECT-TYPE
28     SYNTAX        SEQUENCE OF WmanIfSsOfdmaDownlinkChannelEntry
29     MAX-ACCESS    not-accessible
30     STATUS        current
31     DESCRIPTION
32         "This table contains DCD channel attributes, defining the
33         transmission characteristics of downlink channels"
34     REFERENCE
35         "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
36         ::= { wmanIfSsOfdmaPhy 2 }
37
38 wmanIfSsOfdmaDownlinkChannelEntry OBJECT-TYPE
39     SYNTAX        WmanIfSsOfdmaDownlinkChannelEntry
40     MAX-ACCESS    not-accessible
41     STATUS        current
42     DESCRIPTION
43         "This table provides one row for each downlink channel of
44         multi-sector BS, and is indexed by BS ifIndex. An entry in
45         this table exists for each ifEntry of BS with an ifType of
46         propBWAp2Mp."
47     INDEX          { ifIndex }
48     ::= { wmanIfSsOfdmaDownlinkChannelTable 1 }
49
50 WmanIfSsOfdmaDownlinkChannelEntry ::= SEQUENCE {
51     wmanIfSsOfdmaBsEIRP                INTEGER,
52     wmanIfSsOfdmaChannelNumber          WmanIfChannelNumber,
53     wmanIfSsOfdmaTTG                    INTEGER,
54     wmanIfSsOfdmaRTG                    INTEGER,
55     wmanIfSsOfdmaInitRngMaxRSS           INTEGER,
56     wmanIfSsOfdmaDownlinkCenterFreq     Unsigned32,

```

```

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

```

```

1      STATUS      current
2      DESCRIPTION
3          "Initial Ranging Max. Received Signal Strength at BS
4          Signed in units of 1 dBm."
5      REFERENCE
6          "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
7          ::= { wmanIfSsOfdmaDownlinkChannelEntry 5 }
8
9
10     wmanIfSsOfdmaDownlinkCenterFreq OBJECT-TYPE
11         SYNTAX      Unsigned32
12         UNITS        "kHz"
13         MAX-ACCESS   read-only
14         STATUS       current
15         DESCRIPTION
16             "Downlink center frequency (kHz)."
17         REFERENCE
18             "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
19             ::= { wmanIfSsOfdmaDownlinkChannelEntry 6 }
20
21
22     wmanIfSsOfdmaBsId OBJECT-TYPE
23         SYNTAX      WmanIfBsIdType
24         MAX-ACCESS   read-only
25         STATUS       current
26         DESCRIPTION
27             "Base station ID."
28         REFERENCE
29             "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
30             ::= { wmanIfSsOfdmaDownlinkChannelEntry 7 }
31
32
33     wmanIfSsOfdmaMacVersion OBJECT-TYPE
34         SYNTAX      WmanIfMacVersion
35         MAX-ACCESS   read-only
36         STATUS       current
37         DESCRIPTION
38             "This parameter specifies the version of 802.16 to which
39             the message originator conforms."
40         REFERENCE
41             "Subclause 11.4.1, Table 358, in IEEE Std 802.16-2004"
42             ::= { wmanIfSsOfdmaDownlinkChannelEntry 8 }
43
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         MAX-ACCESS   read-only
56         STATUS       current
57         DESCRIPTION
58             "The duration of the frame. The frame duration code values
59

```

are specified in Table 232 in IEEE Std 802.16-2004."

#### REFERENCE

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

::= { wmanIfSsOfdmaDownlinkChannelEntry 9 }

#### wmanIfSsOfdmaSizeCqichIdField OBJECT-TYPE

SYNTAX INTEGER {threebits(1),  
fourbits(2),  
fivebits(3),  
sixbits(4),  
sevenbits(5),  
eightbits(6),  
ninebits(7)}

MAX-ACCESS read-only

STATUS current

#### DESCRIPTION

"This object defines the size of CQICH ID field.

0 = Reserved

1 = 3 bits

2 = 4 bits

3 = 5 bits

4 = 6 bits

5 = 7 bits

6 = 8 bits

7 = 9 bits

8...255 = Reserved"

#### REFERENCE

"Subclause 11.3.1, Table 358, in IEEE Std 802.16-2004"

::= { wmanIfSsOfdmaDownlinkChannelEntry 10 }

#### wmanIfSsOfdmaHARQAackDelayBurst OBJECT-TYPE

SYNTAX INTEGER {oneframeoffset(1),  
twoframesoffset(2),  
threeframesoffset(3)}

MAX-ACCESS read-only

STATUS current

#### DESCRIPTION

"This object defines the OFDMA H-ARQ ACK delay for DL burst.

1 = one frame offset

2 = two frames offset

3 = three frames offset"

#### REFERENCE

"Subclause 11.3.1, Table 358, in IEEE Std 802.16-2004"

::= { wmanIfSsOfdmaDownlinkChannelEntry 11 }

#### wmanIfSsOfdmaUcdBurstProfileTable OBJECT-TYPE

SYNTAX SEQUENCE OF WmanIfSsOfdmaUcdBurstProfileEntry

MAX-ACCESS not-accessible

STATUS current

#### DESCRIPTION

"This table contains UCD burst profiles for each uplink channel"

#### REFERENCE

"Subclause 11.3.1.1, Table 288 and Table 357, in IEEE Std

```

1      802.16-2004"
2          ::= { wmanIfSsOfdmaPhy 3 }
3
4
5      wmanIfSsOfdmaUcdBurstProfileEntry OBJECT-TYPE
6          SYNTAX      WmanIfSsOfdmaUcdBurstProfileEntry
7          MAX-ACCESS   not-accessible
8          STATUS      current
9          DESCRIPTION
10             "This table provides one row for each UCD burst profile.
11             This table is double indexed. The primary index is an
12             ifIndex with an ifType of propBWA2Mp. The secondary index
13             is wmanIfSsOfdmaUiucIndex."
14             INDEX      { ifIndex, wmanIfSsOfdmaUiucIndex }
15             ::= { wmanIfSsOfdmaUcdBurstProfileTable 1 }
16
17
18
19      WmanIfSsOfdmaUcdBurstProfileEntry ::= SEQUENCE {
20          wmanIfSsOfdmaUiucIndex      INTEGER,
21          wmanIfSsOfdmaUcdFecCodeType  WmanIfOfdmaFecCodeType,
22          wmanIfSsOfdmaRangingDataRatio  INTEGER,
23          wmanIfSsOfdmaNorCOverNOverride OCTET STRING}
24
25
26      wmanIfSsOfdmaUiucIndex OBJECT-TYPE
27          SYNTAX      INTEGER (1 .. 10)
28          MAX-ACCESS   read-only
29          STATUS      current
30          DESCRIPTION
31             "The Uplink Interval Usage Code indicates the uplink burst
32             profile in the UCD message, and is used along with ifIndex
33             to identify an entry in the
34             wmanIfSsOfdmaUcdBurstProfileTable."
35             REFERENCE
36             "Subclause 8.4.5.4.1, in IEEE Std 802.16-2004"
37             ::= { wmanIfSsOfdmaUcdBurstProfileEntry 1 }
38
39
40
41
42      wmanIfSsOfdmaUcdFecCodeType OBJECT-TYPE
43          SYNTAX      WmanIfOfdmaFecCodeType
44          MAX-ACCESS   read-only
45          STATUS      current
46          DESCRIPTION
47             "Uplink FEC code type and modulation type"
48             REFERENCE
49             "Subclause 11.3.1.1, Table 357, in IEEE Std 802.16-2004"
50             ::= { wmanIfSsOfdmaUcdBurstProfileEntry 2 }
51
52
53
54      wmanIfSsOfdmaRangingDataRatio OBJECT-TYPE
55          SYNTAX      INTEGER (0 .. 255)
56          MAX-ACCESS   read-only
57          STATUS      current
58          DESCRIPTION
59             "Reducing factor in units of 1 dB, between the power used
60             for this burst and power should be used for CDMA Ranging."
61             REFERENCE
62             "Subclause 11.3.1.1, Table 357, in IEEE Std 802.16-2004"
63             ::= { wmanIfSsOfdmaUcdBurstProfileEntry 3 }
64
65

```



```

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

```

```

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

```

```

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

```

```

1          wmanIfCmnClassifierRuleUserPriHigh      Integer32,
2          wmanIfCmnClassifierRuleVlanId          Integer32,
3          wmanIfCmnClassifierRuleState           INTEGER,
4          wmanIfCmnClassifierRulePkts            Counter64,
5          wmanIfCmnClassifierRuleIpv6FlowLabel    WmanIfIpv6FlowLabel,
6          wmanIfCmnClassifierRuleBitMap           WmanIfClassifierBitMap}
7
8
9
10         wmanIfCmnClassifierRuleIndex OBJECT-TYPE
11             SYNTAX      Unsigned32 (1..4294967295)
12             MAX-ACCESS   not-accessible
13             STATUS       current
14             DESCRIPTION
15                 "An index is assigned to each classifier in the classifiers
16                 table"
17             ::= { wmanIfCmnClassifierRuleEntry 1 }
18
19
20         wmanIfCmnClassifierRulePriority OBJECT-TYPE
21             SYNTAX      INTEGER (0..255)
22             MAX-ACCESS   read-only
23             STATUS       current
24             DESCRIPTION
25                 "The value specifies the order of evaluation of the
26                 classifiers. The higher the value the higher the
27                 priority. The value of 0 is used as default in
28                 provisioned service flows classifiers. The default
29                 value of 64 is used for dynamic service flow classifiers.
30                 If the referenced parameter is not present in a classifier
31                 , this object reports the default value as defined above"
32             REFERENCE
33                 "Subclause 11.13.19.3.4.1 in IEEE Std 802.16-2004"
34             DEFVAL      { 0 }
35             ::= { wmanIfCmnClassifierRuleEntry 2 }
36
37
38         wmanIfCmnClassifierRuleIpTosLow OBJECT-TYPE
39             SYNTAX      INTEGER (0 .. 255)
40             MAX-ACCESS   read-only
41             STATUS       current
42             DESCRIPTION
43                 "The low value of a range of TOS byte values. If the
44                 referenced parameter is not present in a classifier, this
45                 object reports the value of 0."
46             REFERENCE
47                 "Subclause 11.13.19.3.4.2 in IEEE Std 802.16-2004"
48             ::= { wmanIfCmnClassifierRuleEntry 3 }
49
50
51         wmanIfCmnClassifierRuleIpTosHigh OBJECT-TYPE
52             SYNTAX      INTEGER (0 .. 255)
53             MAX-ACCESS   read-only
54             STATUS       current
55             DESCRIPTION
56                 "The 8-bit high value of a range of TOS byte values.
57                 If the referenced parameter is not present in a classifier
58                 , this object reports the value of 0."
59             REFERENCE
60
61
62
63
64
65

```

```

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

```

```

1         ANDed with the
2         wmanIfCmnClassifierRuleIpSourceMask value equals the
3         wmanIfCmnClassifierRuleIpSourceAddr value.
4         If the referenced parameter is not present in a classifier
5         , this object reports the value of 0.0.0.0."
6     REFERENCE
7         "Subclause 11.13.19.3.4.4 in IEEE Std 802.16-2004"
8     ::= { wmanIfCmnClassifierRuleEntry 8 }
9
10
11
12 wmanIfCmnClassifierRuleIpDestAddr OBJECT-TYPE
13     SYNTAX      InetAddress
14     MAX-ACCESS  read-only
15     STATUS      current
16     DESCRIPTION
17         "This object specifies the value of the IP Destination
18         Address required for packets to match this rule. An IP
19         packet matches the rule when the packet IP destination
20         address bitwise ANDed with the
21         wmanIfCmnClassifierRuleIpDestMask value equals the
22         wmanIfCmnClassifierRuleIpDestAddr value.
23         If the referenced parameter is not present in a
24         classifier, this object reports the value of 0.0.0.0."
25     REFERENCE
26         "Subclause 11.13.19.3.4.5 in IEEE Std 802.16-2004"
27     ::= { wmanIfCmnClassifierRuleEntry 9 }
28
29
30
31
32 wmanIfCmnClassifierRuleIpDestMask OBJECT-TYPE
33     SYNTAX      InetAddress
34     MAX-ACCESS  read-only
35     STATUS      current
36     DESCRIPTION
37         "This object specifies which bits of a packet's IP
38         Destination Address that are compared to match this rule.
39         An IP packet matches the rule when the packet destination
40         address bitwise ANDed with the
41         wmanIfCmnClassifierRuleIpDestMask value equals the
42         wmanIfCmnClassifierRuleIpDestAddr value.
43         If the referenced parameter is not present in a classifier
44         , this object reports the value of 0.0.0.0."
45     REFERENCE
46         "Subclause 11.13.19.3.4.5 in IEEE Std 802.16-2004"
47     ::= { wmanIfCmnClassifierRuleEntry 10 }
48
49
50
51
52
53 wmanIfCmnClassifierRuleSourcePortStart OBJECT-TYPE
54     SYNTAX      Integer32 (0..65535)
55     MAX-ACCESS  read-only
56     STATUS      current
57     DESCRIPTION
58         "This object specifies the low end inclusive range of
59         TCP/UDP source port numbers to which a packet is compared
60         . This object is irrelevant for non-TCP/UDP IP packets.
61         If the referenced parameter is not present in a
62         classifier, this object reports the value of 0."
63     REFERENCE
64
65

```

```

1         "Subclause 11.13.19.3.4.6 in IEEE Std 802.16-2004"
2         ::= { wmanIfCmnClassifierRuleEntry 11 }
3
4
5 wmanIfCmnClassifierRuleSourcePortEnd OBJECT-TYPE
6     SYNTAX      Integer32 (0..65535)
7     MAX-ACCESS  read-only
8     STATUS      current
9     DESCRIPTION
10        "This object specifies the high end inclusive range of
11        TCP/UDP source port numbers to which a packet is compared.
12        This object is irrelevant for non-TCP/UDP IP packets.
13        If the referenced parameter is not present in a classifier,
14        this object reports the value of 65535."
15
16    REFERENCE
17        "Subclause 11.13.19.3.4.6 in IEEE Std 802.16-2004"
18        ::= { wmanIfCmnClassifierRuleEntry 12 }
19
20
21 wmanIfCmnClassifierRuleDestPortStart OBJECT-TYPE
22     SYNTAX      Integer32 (0..65535)
23     MAX-ACCESS  read-only
24     STATUS      current
25     DESCRIPTION
26        "This object specifies the low end inclusive range of
27        TCP/UDP destination port numbers to which a packet is
28        compared. If the referenced parameter is not present
29        in a classifier, this object reports the value of 0."
30
31    REFERENCE
32        "Subclause 11.13.19.3.4.7 in IEEE Std 802.16-2004"
33        ::= { wmanIfCmnClassifierRuleEntry 13 }
34
35
36 wmanIfCmnClassifierRuleDestPortEnd OBJECT-TYPE
37     SYNTAX      Integer32 (0..65535)
38     MAX-ACCESS  read-only
39     STATUS      current
40     DESCRIPTION
41        "This object specifies the high end inclusive range of
42        TCP/UDP destination port numbers to which a packet is
43        compared. If the referenced parameter is not present
44        in a classifier, this object reports the value of
45        65535."
46
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        wmanIfCmnClassifierRuleDestMacMask equals the value of
60        wmanIfCmnClassifierRuleDestMacAddr. If the referenced
61        parameter is not present in a classifier, this object
62

```

```

1         reports the value of '000000000000'H."
2     REFERENCE
3         "Subclause 11.13.19.3.4.8 in IEEE Std 802.16-2004"
4     ::= { wmanIfCmnClassifierRuleEntry 15 }
5
6
7     wmanIfCmnClassifierRuleDestMacMask OBJECT-TYPE
8         SYNTAX      MacAddress
9         MAX-ACCESS   read-only
10        STATUS      current
11        DESCRIPTION
12            "An Ethernet packet matches an entry when its destination
13             MAC address bitwise ANDed with
14             wmanIfCmnClassifierRuleDestMacMask equals the value of
15             wmanIfCmnClassifierRuleDestMacAddr. If the referenced
16             parameter is not present in a classifier, this object
17             reports the value of '000000000000'H."
18        REFERENCE
19            "Subclause 11.13.19.3.4.8 in IEEE Std 802.16-2004"
20        ::= { wmanIfCmnClassifierRuleEntry 16 }
21
22
23     wmanIfCmnClassifierRuleSourceMacAddr OBJECT-TYPE
24         SYNTAX      MacAddress
25         MAX-ACCESS   read-only
26         STATUS      current
27         DESCRIPTION
28             "An Ethernet packet matches this entry when its source
29             MAC address bitwise ANDed with
30             wmanIfCmnClassifierRuleSourceMacMask equals the value
31             of wmanIfCmnClassifierRuleSourceMacAddr. If the
32             referenced parameter is not present in a classifier,
33             this object reports the value of '000000000000'H."
34        REFERENCE
35            "Subclause 11.13.19.3.4.9 in IEEE Std 802.16-2004"
36        ::= { wmanIfCmnClassifierRuleEntry 17 }
37
38
39     wmanIfCmnClassifierRuleSourceMacMask OBJECT-TYPE
40         SYNTAX      MacAddress
41         MAX-ACCESS   read-only
42         STATUS      current
43         DESCRIPTION
44             "An Ethernet packet matches an entry when its destination
45             MAC address bitwise ANDed with
46             wmanIfCmnClassifierRuleSourceMacMask equals the value of
47             wmanIfCmnClassifierRuleSourceMacAddr. If the referenced
48             parameter is not present in a classifier, this object
49             reports the value of '000000000000'H."
50        REFERENCE
51            "Subclause 11.13.19.3.4.9 in IEEE Std 802.16-2004"
52        ::= { wmanIfCmnClassifierRuleEntry 18 }
53
54
55     wmanIfCmnClassifierRuleEnetProtocolType OBJECT-TYPE
56         SYNTAX      INTEGER {none(0),
57                        ethertype(1),
58                        dsap(2)}
59
60
61
62
63
64
65

```



```

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

```

```

1      Such frames include a 16-bit Tag that contains a 3 bit
2      Priority field and a 12 bit VLAN number.
3      Tagged Ethernet packets must have a 3-bit Priority field
4      within the range of wmanIfCmnClassifierRulePriLow and
5      wmanIfCmnClassifierRulePriHigh in order to match this
6      rule.
7      If the referenced parameter is not present in the
8      classifier, the value of this object is reported as 0."
9
10     REFERENCE
11         "Subclause 11.13.19.3.4.11 in IEEE Std 802.16-2004"
12     ::= { wmanIfCmnClassifierRuleEntry 21 }
13
14
15     wmanIfCmnClassifierRuleUserPriHigh OBJECT-TYPE
16         SYNTAX      Integer32 (0..7)
17         MAX-ACCESS   read-only
18         STATUS       current
19         DESCRIPTION
20             "This object applies only to Ethernet frames using the
21             802.1P/Q tag header (indicated with EtherType 0x8100).
22             Such frames include a 16-bit Tag that contains a 3 bit
23             Priority field and a 12 bit VLAN number.
24             Tagged Ethernet packets must have a 3-bit Priority
25             field within the range of wmanIfCmnClassifierRulePriLow
26             and wmanIfCmnClassifierRulePriHigh in order to match
27             this rule.
28             If the referenced parameter is not present in the
29             classifier, the value of this object is reported as 7."
30
31     REFERENCE
32         "Subclause 11.13.19.3.4.11 in IEEE Std 802.16-2004"
33     ::= { wmanIfCmnClassifierRuleEntry 22 }
34
35
36     wmanIfCmnClassifierRuleVlanId OBJECT-TYPE
37         SYNTAX      Integer32 (0..4095)
38         MAX-ACCESS   read-only
39         STATUS       current
40         DESCRIPTION
41             "This object applies only to Ethernet frames using the
42             802.1P/Q tag header.
43             If this object's value is nonzero, tagged packets must
44             have a VLAN Identifier that matches the value in order
45             to match the rule.
46             Only the least significant 12 bits of this object's
47             value are valid.
48             If the referenced parameter is not present in the
49             classifier, the value of this object is reported as 0."
50
51     REFERENCE
52         "Subclause 11.13.19.3.4.12 in IEEE Std 802.16-2004"
53     ::= { wmanIfCmnClassifierRuleEntry 23 }
54
55
56     wmanIfCmnClassifierRuleState OBJECT-TYPE
57         SYNTAX      INTEGER {active(1),
58                             inactive(2)}
59         MAX-ACCESS   read-only
60         STATUS       current
61

```

```

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

::= { wmanIfCmnClassifierRuleEntry 24 }

```

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

wmanIfCmnClassifierRuleIpv6FlowLabel OBJECT-TYPE

```

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

wmanIfCmnClassifierRuleBitMap OBJECT-TYPE

```

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

wmanIfCmnPhsRuleTable OBJECT-TYPE

```

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

REFERENCE

```

64         "Subclause 5.2.3 in IEEE Std 802.16-2004"
```

```

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

```

```

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

```

```

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

```

```

1         "When this table is implemented on the basestation, this
2         object contains the SS Mac address, the reported service
3         flow was created for. On the SS, the value returned is
4         the SS's own Mac address."
5         ::= { wmanIfCmnCpsServiceFlowEntry 1 }
6
7
8 wmanIfCmnCpsSfId OBJECT-TYPE
9     SYNTAX      Unsigned32 ( 1 .. 4294967295)
10    MAX-ACCESS  read-only
11    STATUS      current
12    DESCRIPTION
13        "A 32 bit quantity that uniquely identifies a service flow
14        to both the subscriber station and base station (BS)."
```

```

15    ::= { wmanIfCmnCpsServiceFlowEntry 2 }
16
17
18 wmanIfCmnCpsSfCid OBJECT-TYPE
19     SYNTAX      WmanIfCidType
20     MAX-ACCESS  read-only
21     STATUS      current
22     DESCRIPTION
23         "A 16 bit channel identifier to identify the connection
24         being created by DSA."
25     ::= { wmanIfCmnCpsServiceFlowEntry 3 }
26
27
28 wmanIfCmnCpsSfDirection OBJECT-TYPE
29     SYNTAX      INTEGER {downstream(1),
30                        upstream(2)}
31     MAX-ACCESS  read-only
32     STATUS      current
33     DESCRIPTION
34         "An attribute indicating the service flow is downstream or
35         upstream."
36     ::= { wmanIfCmnCpsServiceFlowEntry 4 }
37
38
39 wmanIfCmnCpsSfState OBJECT-TYPE
40     SYNTAX      WmanIfSfState
41     MAX-ACCESS  read-only
42     STATUS      current
43     DESCRIPTION
44         "wmanIfCmnCpsSfState indicates the service flow state:
45         Authorized (1), Admitted (2), and Active (3) service
46         flow state."
47     REFERENCE
48         "Subclause 6.3.14.6, in IEEE Std 802.16-2004"
49     ::= { wmanIfCmnCpsServiceFlowEntry 5 }
50
51
52 wmanIfCmnCpsTrafficPriority OBJECT-TYPE
53     SYNTAX      INTEGER (0 .. 7)
54     MAX-ACCESS  read-only
55     STATUS      current
56     DESCRIPTION
57         "The value of this parameter specifies the priority
58         assigned to a service flow. For uplink service flows,
59         the BS should use this parameter when determining
60
61
62
63
64
65
```

```

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

```



```

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

```

```

1      MAX-ACCESS    read-only
2      STATUS        current
3      DESCRIPTION
4          "Specifies the upstream scheduling service used for
5          upstream service flow. If the referenced parameter
6          is not present in the corresponding 802.16 QoS
7          Parameter Set of an upstream service flow, the
8          default value of this object is bestEffort(2)."
```

REFERENCE

```

11         "Subclause 11.13.11 in IEEE Std 802.16-2004"
12     DEFVAL          { bestEffort }
13     ::= { wmanIfCmnCpsServiceFlowEntry 14 }
```

wmanIfCmnCpsArqEnable OBJECT-TYPE

```

17     SYNTAX          TruthValue
18     MAX-ACCESS      read-only
19     STATUS          current
20     DESCRIPTION
21         "True(1) ARQ enabling is requested for the connection."
22     ::= { wmanIfCmnCpsServiceFlowEntry 15 }
```

wmanIfCmnCpsArqWindowSize OBJECT-TYPE

```

26     SYNTAX          INTEGER (1..1024)
27     MAX-ACCESS      read-only
28     STATUS          current
29     DESCRIPTION
30         "Indicates the maximum number of unacknowledged
31         fragments at any time."
32     ::= { wmanIfCmnCpsServiceFlowEntry 16 }
```

wmanIfCmnCpsArqBlockLifetime OBJECT-TYPE

```

37     SYNTAX          INTEGER (0 .. 65535)
38     UNITS            "10 us"
39     MAX-ACCESS      read-only
40     STATUS          current
41     DESCRIPTION
42         "The maximum time interval an ARQ fragment will be
43         managed by the transmitter ARQ machine, once
44         initial transmission of the fragment has occurred.
45         If transmission or retransmission of the fragment
46         is not acknowledged by the receiver before the
47         time limit is reached, the fragment is discarded.
48         A value of 0 means Infinite."
49     ::= { wmanIfCmnCpsServiceFlowEntry 17 }
```

wmanIfCmnCpsArqSyncLossTimeout OBJECT-TYPE

```

55     SYNTAX          INTEGER (0 .. 65535 )
56     UNITS            "10 us"
57     MAX-ACCESS      read-only
58     STATUS          current
59     DESCRIPTION
60         "The maximum interval before declaring a loss
61         of synchronization of the sender and receiver
62         state machines. A value of 0 means Infinite."
63
64
65
```

```

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

```

```

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

```

```

1         to which the SSs are associated."
2     REFERENCE
3         "Subclause 10.1 in IEEE Std 802.16-2004"
4     ::= { wmanIfCmnCps 2 }
5
6
7     wmanIfCmnBsSsConfigurationEntry OBJECT-TYPE
8         SYNTAX      WmanIfCmnBsSsConfigurationEntry
9         MAX-ACCESS   not-accessible
10        STATUS      current
11        DESCRIPTION
12            "This table is indexed by ifIndex, indicating BS
13             sector."
14        INDEX        { ifIndex }
15        ::= { wmanIfCmnBsSsConfigurationTable 1 }
16
17
18
19     WmanIfCmnBsSsConfigurationEntry ::= SEQUENCE {
20         wmanIfCmnInvitedRangRetries      INTEGER,
21         wmanIfCmnDSxReqRetries           Unsigned32,
22         wmanIfCmnDSxRespRetries          Unsigned32,
23         wmanIfCmnT7Timeout                INTEGER,
24         wmanIfCmnT8Timeout                INTEGER,
25         wmanIfCmnT10Timeout               INTEGER,
26         wmanIfCmnT22Timeout               INTEGER}
27
28
29
30     wmanIfCmnInvitedRangRetries OBJECT-TYPE
31         SYNTAX      INTEGER (16..65535)
32         MAX-ACCESS   read-write
33         STATUS      current
34         DESCRIPTION
35             "Number of retries on inviting Ranging Requests."
36         ::= { wmanIfCmnBsSsConfigurationEntry 1 }
37
38
39
40     wmanIfCmnDSxReqRetries OBJECT-TYPE
41         SYNTAX      Unsigned32
42         MAX-ACCESS   read-write
43         STATUS      current
44         DESCRIPTION
45             "Number of Timeout Retries on DSA/DSC/DSD Requests."
46         DEFVAL      { 3 }
47         ::= { wmanIfCmnBsSsConfigurationEntry 2 }
48
49
50
51     wmanIfCmnDSxRespRetries OBJECT-TYPE
52         SYNTAX      Unsigned32
53         MAX-ACCESS   read-write
54         STATUS      current
55         DESCRIPTION
56             "Number of Timeout Retries on DSA/DSC/DSD Responses."
57         DEFVAL      { 3 }
58         ::= { wmanIfCmnBsSsConfigurationEntry 3 }
59
60
61
62     wmanIfCmnT7Timeout OBJECT-TYPE
63         SYNTAX      INTEGER (0 .. 1000)
64         UNITS        "milliseconds"
65         MAX-ACCESS   read-write

```

```

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

```

```

1      DESCRIPTION
2          "Each entry contains the cryptographic suite pair that SS
3          or BS supports."
4      INDEX          { ifIndex, wmanIfCmnCryptoSuiteIndex }
5      ::= { wmanIfCmnCryptoSuiteTable 1 }
6
7
8      WmanIfCmnCryptoSuiteEntry ::= SEQUENCE {
9          wmanIfCmnCryptoSuiteIndex          Integer32,
10         wmanIfCmnCryptoSuiteDataEncryptAlg  WmanIfDataEncryptAlgId,
11         wmanIfCmnCryptoSuiteDataAuthentAlg  WmanIfDataAuthAlgId,
12         wmanIfCmnCryptoSuiteTekEncryptAlg   WmanIfTekEncryptAlgId}
13
14
15     wmanIfCmnCryptoSuiteIndex OBJECT-TYPE
16         SYNTAX          Integer32 (1 .. 1000)
17         MAX-ACCESS      not-accessible
18         STATUS          current
19         DESCRIPTION
20             "The index for a cryptographic suite row."
21         ::= { wmanIfCmnCryptoSuiteEntry 1 }
22
23
24
25     wmanIfCmnCryptoSuiteDataEncryptAlg OBJECT-TYPE
26         SYNTAX          WmanIfDataEncryptAlgId
27         MAX-ACCESS      read-only
28         STATUS          current
29         DESCRIPTION
30             "The value of this object is the data encryption algorithm
31             for this cryptographic suite capability."
32         REFERENCE
33             "Table 375, IEEE Std 802.16-2004"
34         ::= { wmanIfCmnCryptoSuiteEntry 2 }
35
36
37
38     wmanIfCmnCryptoSuiteDataAuthentAlg OBJECT-TYPE
39         SYNTAX          WmanIfDataAuthAlgId
40         MAX-ACCESS      read-only
41         STATUS          current
42         DESCRIPTION
43             "The value of this object is the data authentication
44             algorithm for this cryptographic suite capability."
45         REFERENCE
46             "Table 376, IEEE Std 802.16-2004"
47         ::= { wmanIfCmnCryptoSuiteEntry 3 }
48
49
50
51     wmanIfCmnCryptoSuiteTekEncryptAlg OBJECT-TYPE
52         SYNTAX          WmanIfTekEncryptAlgId
53         MAX-ACCESS      read-only
54         STATUS          current
55         DESCRIPTION
56             "The value of this object is the TEK key encryption
57             algorithm for this cryptographic suite capability."
58         REFERENCE
59             "Table 377, IEEE Std 802.16-2004"
60         ::= { wmanIfCmnCryptoSuiteEntry 4 }
61
62
63
64     --
65

```

```

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

```



```

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

```

```

1          wmanIfCmnT7Timeout,
2          wmanIfCmnT8Timeout,
3          wmanIfCmnT10Timeout,
4          wmanIfCmnT22Timeout,
5          wmanIfCmnCryptoSuiteDataEncryptAlg,
6          wmanIfCmnCryptoSuiteDataAuthentAlg,
7          wmanIfCmnCryptoSuiteTekEncryptAlg}
8
9      STATUS          current
10
11      DESCRIPTION
12          "This group contains objects for both BS and SS,
13          and are independent of PHY."
14      ::= { wmanIfMibGroups 1 }
15
16  wmanIfMibQoSGroup      OBJECT-GROUP
17      OBJECTS {wmanIfCmnCpsSfId,
18              wmanIfCmnCpsSfCid,
19              wmanIfCmnCpsSfDirection,
20              wmanIfCmnCpsSfState,
21              wmanIfCmnCpsTrafficPriority,
22              wmanIfCmnCpsMaxSustainedRate,
23              wmanIfCmnCpsMaxTrafficBurst,
24              wmanIfCmnCpsMinReservedRate,
25              wmanIfCmnCpsToleratedJitter,
26              wmanIfCmnCpsMaxLatency,
27              wmanIfCmnCpsFixedVsVariableSduInd,
28              wmanIfCmnCpsSduSize,
29              wmanIfCmnCpsSfSchedulingType,
30              wmanIfCmnCpsArqEnable,
31              wmanIfCmnCpsArqWindowSize,
32              wmanIfCmnCpsArqBlockLifetime,
33              wmanIfCmnCpsArqSyncLossTimeout,
34              wmanIfCmnCpsArqDeliverInOrder,
35              wmanIfCmnCpsArqRxPurgeTimeout,
36              wmanIfCmnCpsArqBlockSize,
37              wmanIfCmnCpsMinRsvdTolerableRate,
38              wmanIfCmnCpsReqTxPolicy,
39              wmanIfCmnSfCsSpecification}
40
41      STATUS          current
42
43      DESCRIPTION
44          "This group contains QoS objects for both BS and SS."
45      ::= { wmanIfMibGroups 2 }
46
47
48  wmanIfMibBsGroup      OBJECT-GROUP
49      OBJECTS {-- Service classes
50              wmanIfBsSfDirection,
51              wmanIfBsServiceClassIndex,
52              wmanIfBsSfState,
53              wmanIfBsSfProvisionedTime,
54              wmanIfBsProvisionedSfRowStatus,
55              wmanIfBsSsProvisionedForSfRowStatus,
56              wmanIfBsSfCsSpecification,
57              wmanIfBsQoSServiceClassName,
58              wmanIfBsQOSTrafficPriority,
59              wmanIfBsQOSMaxSustainedRate,
60
61
62
63
64
65

```

```

1      wmanIfBsQoSMaxTrafficBurst,
2      wmanIfBsQoSMinReservedRate,
3      wmanIfBsQOSToleratedJitter,
4      wmanIfBsQoSMaxLatency,
5      wmanIfBsQOSFixedVsVariableSduInd,
6      wmanIfBsQOSSduSize,
7      wmanIfBsQoSScSchedulingType,
8      wmanIfBsQoSScArqEnable,
9      wmanIfBsQoSScArqWindowSize,
10     wmanIfBsQoSScArqBlockLifetime,
11     wmanIfBsQoSScArqSyncLossTimeout,
12     wmanIfBsQoSScArqDeliverInOrder,
13     wmanIfBsQoSScArqRxPurgeTimeout,
14     wmanIfBsQoSScArqBlockSize,
15     wmanIfBsQoSSCMinRsvdTolerableRate,
16     wmanIfBsQoSReqTxPolicy,
17     wmanIfBsQOSServiceClassRowStatus,
18
19     -- Classification
20     wmanIfBsClassifierRulePriority,
21     wmanIfBsClassifierRuleIpTosLow,
22     wmanIfBsClassifierRuleIpTosHigh,
23     wmanIfBsClassifierRuleIpTosMask,
24     wmanIfBsClassifierRuleIpProtocol,
25     wmanIfBsClassifierRuleIpSourceAddr,
26     wmanIfBsClassifierRuleIpSourceMask,
27     wmanIfBsClassifierRuleIpDestAddr,
28     wmanIfBsClassifierRuleIpDestMask,
29     wmanIfBsClassifierRuleSourcePortStart,
30     wmanIfBsClassifierRuleSourcePortEnd,
31     wmanIfBsClassifierRuleDestPortStart,
32     wmanIfBsClassifierRuleDestPortEnd,
33     wmanIfBsClassifierRuleDestMacAddr,
34     wmanIfBsClassifierRuleDestMacMask,
35     wmanIfBsClassifierRuleSourceMacAddr,
36     wmanIfBsClassifierRuleSourceMacMask,
37     wmanIfBsClassifierRuleEnetProtocolType,
38     wmanIfBsClassifierRuleEnetProtocol,
39     wmanIfBsClassifierRuleUserPriLow,
40     wmanIfBsClassifierRuleUserPriHigh,
41     wmanIfBsClassifierRuleVlanId,
42     wmanIfBsClassifierRuleState,
43     wmanIfBsClassifierRulePhsSize,
44     wmanIfBsClassifierRulePhsMask,
45     wmanIfBsClassifierRulePhsVerify,
46     wmanIfBsClassifierRuleIpv6FlowLabel,
47     wmanIfBsClassifierRuleBitMap,
48     wmanIfBsClassifierRuleRowStatus,
49
50     -- Packet counters
51     wmanIfBsSsMacSduCount,
52     wmanIfBsSsOctetCount,
53     wmanIfBsSsResetCounter,
54     wmanIfBsSsResetCounterTime,
55
56
57
58
59
60
61
62
63
64
65

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```



```

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

```

```

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

```

```

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

```

```

1          wmanIfBsOfdmaBsEIRP,
2          wmanIfBsOfdmaChannelNumber,
3          wmanIfBsOfdmaTTG,
4          wmanIfBsOfdmaRTG,
5          wmanIfBsOfdmaInitRngMaxRSS,
6          wmanIfBsOfdmaDownlinkCenterFreq,
7          wmanIfBsOfdmaBsId,
8          wmanIfBsOfdmaMacVersion,
9          wmanIfBsOfdmaFrameDurationCode,
10         wmanIfBsOfdmaSizeCqichIdField,
11         wmanIfBsOfdmaHARQAackDelayBurst,
12         wmanIfBsOfdmaUcdFecCodeType,
13         wmanIfBsOfdmaRangingDataRatio,
14         wmanIfBsOfdmaNorCOverNOVERRIDE,
15         wmanIfBsOfdmaUcdBurstProfileRowStatus,
16         wmanIfBsOfdmaDownlinkFrequency,
17         wmanIfBsOfdmaDcdFecCodeType,
18         wmanIfBsOfdmaDiucMandatoryExitThresh,
19         wmanIfBsOfdmaDiucMinEntryThresh,
20         wmanIfBsOfdmaDcdBurstProfileRowStatus}
21
22 STATUS          current
23 DESCRIPTION
24     "This group contains objects for BS and OFDMA PHY."
25 ::= { wmanIfMibGroups 8 }
26
27
28
29
30
31 wmanIfMibSsOfdmaGroup      OBJECT-GROUP
32     OBJECTS {wmanIfSsOfdmaCtBasedResvTimeout,
33             wmanIfSsOfdmaBwReqOppSize,
34             wmanIfSsOfdmaRangReqOppSize,
35             wmanIfSsOfdmaUplinkCenterFreq,
36             wmanIfSsOfdmaInitRngCodes,
37             wmanIfSsOfdmaPeriodicRngCodes,
38             wmanIfSsOfdmaBWReqCodes,
39             wmanIfSsOfdmaPerRngBackoffStart,
40             wmanIfSsOfdmaPerRngBackoffEnd,
41             wmanIfSsOfdmaStartOfRngCodes,
42             wmanIfSsOfdmaPermutationBase,
43             wmanIfSsOfdmaULAllocSubchBitmap,
44             wmanIfSsOfdmaOptPermULAllocSubchBitmap,
45             wmanIfSsOfdmaBandAMCAllocThreshold,
46             wmanIfSsOfdmaBandAMCReleaseThreshold,
47             wmanIfSsOfdmaBandAMCAllocTimer,
48             wmanIfSsOfdmaBandAMCReleaseTimer,
49             wmanIfSsOfdmaBandStatRepMAXPeriod,
50             wmanIfSsOfdmaBandAMCRetryTimer,
51             wmanIfSsOfdmaSafetyChAllocThreshold,
52             wmanIfSsOfdmaSafetyChReleaseThreshold,
53             wmanIfSsOfdmaSafetyChAllocTimer,
54             wmanIfSsOfdmaSafetyChReleaseTimer,
55             wmanIfSsOfdmaBinStatRepMAXPeriod,
56             wmanIfSsOfdmaSafetyChARetryTimer,
57             wmanIfSsOfdmaHARQAackDelayULBurst,
58             wmanIfSsOfdmaCQICHBandAMCTranaDelay,
59             wmanIfSsOfdmaBsEIRP,
60
61
62
63
64
65

```

```

1          wmanIfSsOfdmaChannelNumber,
2          wmanIfSsOfdmaTTG,
3          wmanIfSsOfdmaRTG,
4          wmanIfSsOfdmaInitRngMaxRSS,
5          wmanIfSsOfdmaDownlinkCenterFreq,
6          wmanIfSsOfdmaBsId,
7          wmanIfSsOfdmaMacVersion,
8          wmanIfSsOfdmaFrameDurationCode,
9          wmanIfSsOfdmaSizeCqichIdField,
10         wmanIfSsOfdmaHARQAackDelayBurst,
11         wmanIfSsOfdmaUiucIndex,
12         wmanIfSsOfdmaUcdFecCodeType,
13         wmanIfSsOfdmaRangingDataRatio,
14         wmanIfSsOfdmaNorCOverNOVERRIDE,
15         wmanIfSsOfdmaDiucIndex,
16         wmanIfSsOfdmaDownlinkFrequency,
17         wmanIfSsOfdmaDcdFecCodeType,
18         wmanIfSsOfdmaDiucMandatoryExitThresh,
19         wmanIfSsOfdmaDiucMinEntryThresh}
20
21     STATUS          current
22     DESCRIPTION
23         "This group contains objects for SS and OFDMA PHY."
24     ::= { wmanIfMibGroups 9 }
25
26 wmanIfMibBsNotificationGroup    NOTIFICATION-GROUP
27     NOTIFICATIONS {wmanIfBsSsStatusNotificationTrap,
28                     wmanIfBsSsDynamicServiceFailTrap,
29                     wmanIfBsSsRssiStatusChangeTrap,
30                     wmanIfBsSsPkmFailTrap,
31                     wmanIfBsSsRegistrerTrap}
32
33     STATUS          current
34     DESCRIPTION
35         "This group contains event notifications for BS."
36     ::= { wmanIfMibGroups 10 }
37
38 wmanIfMibSsNotificationGroup    NOTIFICATION-GROUP
39     NOTIFICATIONS {wmanIfSsTlvUnknownTrap,
40                     wmanIfSsDynamicServiceFailTrap,
41                     wmanIfSsDhcpSuccessTrap,
42                     wmanIfSsRssiStatusChangeTrap}
43
44     STATUS          current
45     DESCRIPTION
46         "This group contains event notifications for SS."
47     ::= { wmanIfMibGroups 11 }
48
49 wmanIfMibCmnPhsGroup           OBJECT-GROUP
50     OBJECTS {-- Payload header supression
51               wmanIfCmnPhsRulePhsField,
52               wmanIfCmnPhsRulePhsMask,
53               wmanIfCmnPhsRulePhsSize,
54               wmanIfCmnPhsRulePhsVerify}
55
56     STATUS          current
57     DESCRIPTION
58         "This group contains common objects for PHS."
59
60
61
62
63
64
65

```

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

## **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         wmanIfBsPagingGroupTable OBJECT-TYPE
26             SYNTAX SEQUENCE OF WmanIfBsPagingGroupEntry
27             MAX-ACCESS not-accessible
28             STATUS current
29             DESCRIPTION
30                 "This table contains paging group related parameters."
31             ::= { wmanIfBsMobility 3 }
32
33
34         wmanIfBsPagingGroupEntry OBJECT-TYPE
35             SYNTAX WmanIfBsPagingGroupEntry
36             MAX-ACCESS not-accessible
37             STATUS current
38             DESCRIPTION
39                 "This table is indexed by wmanIfBsPagingGroupId."
40             INDEX { wmanIfBsPagingGroupId }
41             ::= { wmanIfBsPagingGroupTable 1 }
42
43
44         wmanIfBsPagingGroupEntry ::= SEQUENCE {
45             wmanIfBsPagingControlId          IpAddress,
46             wmanIfBsPagingGroupId            INTEGER,
47             wmanIfBsMgmtResourceHoldingTimerInteger32,
48             wmanIfBsT46Timer                  Integer32,
49             wmanIfBsPagingRetryCount          INTEGER,
50             wmanIfBsREQDuration               INTEGER,
51             wmanIfBsMACHashSkipThresholdInteger32,
52             wmanIfBsCDMATransmissionOpportunityAssignmentINTEGER,
53             wmanIfBsPagingResponseWindow     INTEGER,
54             wmanIfBsIdleModeTimer             INTEGER,
55             wmanIfBsIdleModeSystemTimer      INTEGER,
56             wmanIfBsPagingIntervallLength    INTEGER,
57             wmanIfBsPagingCycle               INTEGER
58         }
59
60
61
62
63
64
65

```

```

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

```

```

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

```



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

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

21  
 22       wmanIfBsPagingCycle OBJECT-TYPE  
 23           SYNTAX INTEGER  
 24           MAX-ACCESS read-write  
 25           STATUS current  
 26           DESCRIPTION  
 27               "Cycle in which the paging message is transmitted  
 28               within the paging group."  
 29  
 30           ::= { wmanIfBsPagingGroupEntry 13 }  
 31  
 32  
 33  
 34  
 35  
 36  
 37  
 38  
 39  
 40  
 41  
 42  
 43  
 44  
 45  
 46  
 47  
 48  
 49  
 50  
 51  
 52  
 53  
 54  
 55  
 56  
 57  
 58  
 59  
 60  
 61  
 62  
 63  
 64  
 65