Project	**IEEE 802.16 Broadband Wireless Access Working Group <http://ieee802.org/16>**
Title	**[**Operation Support System Interface Specification for 802.16 fixed Wireless Systems**]**
Date Submitted	**[2004-08-24]**
Source(s)	Radu Selea, Bogdan Moldoveanu Redline Communications 302 Town Center Blvd., Suite 100 Markham, ON, Canada L3R 0E8 Voice: 905 479 8344 (ext 223) Fax: 905 479 5333 mailto: radu@redlinecommunications.com
Re:	IEEE 802.16g-04/01
Abstract	This document contains a baseline proposal of an OSS Interface draft.
Purpose	[Description of what the author wants 802.16 to do with the information in the document.]
Notice	This document has been prepared to assist IEEE 802.16. It is offered as a basis for discussion and is not binding on the contributing individual(s) or organization(s). The material in this document is subject to change in form and content after further study. The contributor(s) reserve(s) the right to add, amend or withdraw material contained herein.
Release	The contributor grants a free, irrevocable license to the IEEE to incorporate material contained in this contribution, and any modifications thereof, in the creation of an IEEE Standards publication; to copyright in the IEEE's name any IEEE Standards publication even though it may include portions of this contribution; and at the IEEE's sole discretion to permit others to reproduce in whole or in part the resulting IEEE Standards publication. The contributor also acknowledges and accepts that this contribution may be made public by IEEE 802.16.
Patent Policy and Procedures	The contributor is familiar with the IEEE 802.16 Patent Policy and Procedures <http://ieee802.org/16/ipr/patents/policy.html>, including the statement "IEEE standards may include the known use of patent(s), including patent applications, provided the IEEE receives assurance from the patent holder or applicant with respect to patents essential for compliance with both mandatory and optional portions of the standard." Early disclosure to the Working Group of patent information that might be relevant to the standard is essential to reduce the possibility for delays in the development process and increase the likelihood that the draft publication will be approved for publication. Please notify the Chair <mailto:chair@wirelessman.org> as early as possible, in written or electronic form, if patented technology (or technology under patent application) might be incorporated into a draft standard being developed within the IEEE 802.16 Working Group. The Chair will disclose this notification via the IEEE 802.16 web site <http://ieee802.org/16/ipr/patents/notices>.

Operation Support System Interface Specification

Radu Selea
Bogdan Moldoveanu
Redline Communications Inc.

1 Scope

The scope of the document is to define a recommended practice for WiMAX compliant systems on Operation Support System interfaces.
Basic network management requirements to support 802.16 systems are included:

- Device provisioning
- User provisioning
- User account management
- Configuration management
- Fault management
- General BS/SS management and monitoring (SNMP)
- MIB's requirements

The proposal is mostly based on SNMP as the basic management interface but other possible interfaces are nominated or proposed for consideration along the document.

2 Network Management and Provisioning in 802.16 Networks

The paragraph has an informational purpose. In Figure 1 is presented a high level generic architecture of an access network management infrastructure.

The intent is to define the required interfaces at network elements level in order to support OSS functions and to allow 802.16 systems integration in existing service provider infrastructures.
Goals:

- Reduction at a minimum of truck roll overhead
- Seamless user connectivity

- Access to QoS and multiple services that allows triple play (Data/Voice/Video)
  - Multiple Service Flows per user
- Dynamic services environment
  - Easy and incremental service adjustments
  - Bandwidth management
- Flexible service portfolio
  - Services Classes
  - Billing per Time of Day concept
  - Multiple Service Flows to the user
- Specify the SS/BS interfaces in order to allow connectivity to any Operator OSS infrastructure.

The picture can include others options like Radius, SYSLOG servers but for simplicity and flexibility the focus are on designing a framework for the interfaces.

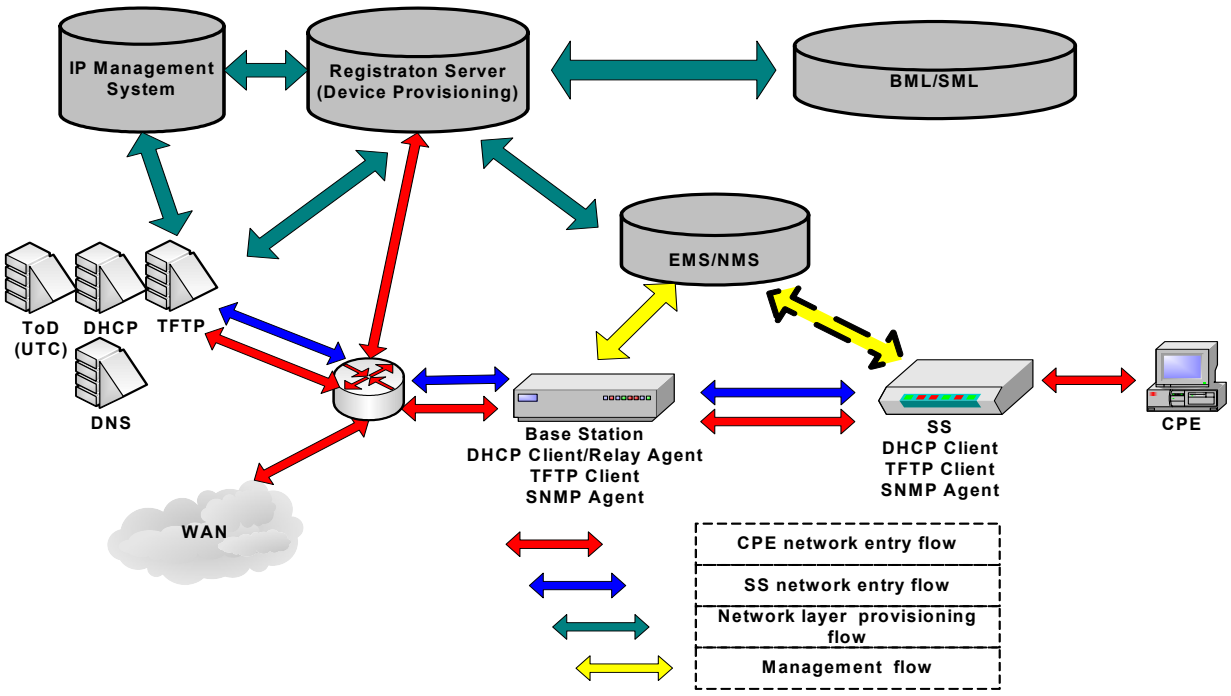

Figure 1 Provisioning and Network Management Components

2.1 Network Element Layer

- BS – Base Station

- SS – Subscriber Station
- Any manageable Network Element present in the network (routers, switches)

2.1.1 Standard Internet Elements

Standard Internet Elements are part of the Network Element Layer.

- DNS – Domain Name System Server maps IP addresses to ASCII domain names
- DHCP – Dynamic Host Configuration Protocol Server dynamically allocates IP addresses and client configuration information to IP devices.
    - DHCP server is used by Network Elements (BS, SS, routers, switches)
    - User devices use DHCP server in order to get a public IP address and access the Internet.
- TOD – Time of Day Server used by BS and SS.
- TFTP – Trivial File Transfer Protocol Server to transfer Configuration Files and any required SW upgrades to network elements.

2.2 Element Management Layer

EMS typically is an SNMP manager that communicates directly with the 802.16 network elements SNMP agents.
They can reside on Base Station or in Network Operation Centre (NOC). The EMS's for other types of equipment are outside the scope of the document.
They communicate directly to the 802.16 network elements.

2.3 Network Management Layer

Network Management layer has multiple components that have to be detailed.

2.3.1 Network Management System

NMS is the system that manages and aggregates EMS systems activity. It usually contain an SNMP manager and required network tools for management purposes:

- Configuration Management
- Fault Management
- Monitoring
- Data collection (performance, billing, statistics)
- Provisioning

The functionality of NMS can be limited or expanded function of Service Provider infrastructure.

2.3.2 IP Management System

The IP Management Server manages the IP servers required to provision the 802.16 network elements and the user Internet access. The typical components managed by the Internet management system are the ToD, TFTP, DNS and DHCP servers.

The IP Management System may support beside 802.16 systems and user devices, other type of access equipment.
It should control and manage:

- IP address ranges and numbering.
- TFTP server
    - Track of SW upgrades and Configuration File names
    - Update DHCP server with proper information to include in DHCP messages
        - Upgrade file name/path
        - Configuration File name/path
- DNS updates
- DHCP Server
    - IP address range
    - IP address pools (private/public)
    - Management of unregistered versus registered users
- Updates all servers based on information from Registration /Provisioning Server
- Updates Registration/Provisioning Server of network elements/users IP address assignment, for provisioning purposes

For simple scenarios, a network administrator using the standard interfaces to the ToD, TFTP, DNS and DHCP servers may perform the IP Management System functionality manually.

2.3.3 Registration/Provisioning Server

The Server resides at the "border" of Network Management Layer and Business/Service Management Layer and it is mainly responsible on network elements (SS/BS) provisioning and user provisioning.
It should have following generic components:

- **Registration Server**
    - Subscriber registration/deregistration process and service updates
    - Managing collected information and forwarding them to:
        - Service/user related info to Business Service Management Layer
            - User subscription info
            - Billing related info
            - Service availability
        - Service provisioning information to IP Management System
            - SS MAC / CPE's MAC / IP addresses
            - User service registration status
            - Configuration File info (if needed)
        - Service provisioning info to provisioning application
            - Service Class (or service description)
            - SS MAC /IP address
            - CPE's MAC / IP addresses

- User/client ID (if the case)

- **Provisioning Server**
  - Provisioning Application - is responsible for coordinating the SS/BS and user provisioning process. This application has an associated SNMP Entity.
  - Provisioning SNMP Entity – The provisioning SNMP entity MUST include a trap/inform handler for provisioning enrollment and the provisioning status traps/informs as well as an SNMP engine for retrieving 802.16 elements information. This entity has to provide the means to provision subscribed service flows (class of service) attributes into the BS/SS. 802.16 specification favors BS centric service flow provisioning.
    The SNMP entity can reside on SNMP manager (NMS).

The server can include an LDAP database dedicated to these purposes.
Any of these elements are not bound to a certain structure or specific functionality, but their main features and some minimal interfaces are required to any OSS and Network Management System.
The structure described in this paragraph is used along to document in order to outline desired features of WiMax equipment and range of services that can be provided.

2.4 Business Management Layer & Service Management Layer

- **BML** layer is made up of a number of business processes such as billing systems and customer care or help desk systems.
- **SML** layer is made up of a number of service processes such as policy management, call management, and order entry.

These layers are not subjects of this document.

3 OSSI for IEEE 802.16 Radio Interface

3.1 Fault management

The goals of fault management are remote monitoring/detection, diagnosis, and correction of problems. Network Management operators rely on the ability to monitor and detect problems (such as ability to trace and identify faults, accept and act on error-detection events), as well as the ability to diagnose and correct problems (such as perform a sequences of diagnostic tests, correct faults, and display/maintain event logs).
This section defines what should be available to support remote monitoring/detection, and diagnosis and correction of problems.

3.1.1 SNMP Usage

The goals of fault management are the remote detection, diagnosis, and correction of network problems. Therefore, the requirements related to SNMP are very similar to the ones described in 3.2.5.

SS SNMP access may be restricted to support policy goals.
SS installation technicians can use SNMP queries from a station on the Ethernet side to perform on-site SS diagnostics and fault classification (note that this may require temporary provisioning of the SS from a local DHCP server or assignment of static IP address).
Further, future Ethernet side customer applications, using SNMP queries, can diagnose simple post-installation issues, avoiding visits from service personnel and minimizing help desk telephone queries.
Standard mib-2 support MUST be implemented to instrument interface status, packet corruption, protocol errors, etc. The transmission MIB for Ethernet-like objects [RFC 2665] MUST be implemented on each SS Ethernet and Fast Ethernet port.
The BS/SS SHOULD support managed objects for fault management of the PHY and MAC layers.

3.1.2 Event Notification

An 802.16 device BS/SS MUST generate asynchronous events that indicate malfunction situations and notify about important non-fault events. Events could be stored in BS/SS device internal event LOG file, in nonvolatile memory and:

- Get reported to other SNMP entities (as TRAP or INFORM SNMP messages)
- Be sent as a SYSLOG event message to a pre-defined SYSLOG server. (Optional)
- Events MAY also be sent to the BS/SS console as a duplicate (identical) message to the optional console destination.

Event notification implemented by BS/SS SHOULD be fully configurable, by priority class; including the ability to disable SNMP Trap, SYSLOG transmission, and local logging.
A BS/SS device MUST support one of the following event notification mechanisms (regardless of the device's SNMP mode):

- Local event logging
- SNMP TRAP/INFORM (trap-versions/targets/limiting/throttling)
- SYSLOG (targets/limiting/throttling)

3.1.3 Local Event Logging

A SS MUST maintain local-log events in both local-volatile storage and local-nonvolatile storage. A BS must maintain local-log events in local-volatile storage or local-nonvolatile storage or both.
BS/SS events designated for local-nonvolatile storage MAY also be retained in local-volatile storage. Data from local-volatile log and local-nonvolatile log is reported through an xxxDevEventTable.

- The 802.16 device event log should be organized as a cyclic buffer with a minimum of ten entries, and may persist across reboots. The event log table must be accessible through the xxxDevEventTable of the MIB's.

- Event descriptions should appear in English and must not be longer than 255 characters, which is the maximum defined for SnmpAdminString.
- Events are identical if their EventIds are identical. For identical events occurring consecutively, the SS may choose to store only a single event. In such a case, the event description recorded must reflect the most recent event.
- The EventId digit is a 32-bit unsigned integer. The EventId must be converted from the error codes to be defined.
-

The unused range of EventIds can be used as vendor-specific EventIds using the following format:

- Bit 31 is set to indicate vendor-specific event
- Bits 30-16 contain the lower 15 bits of the vendor's SNMP enterprise number
- Bits 15-0 are used by the vendor to number events

TBD.

Section 'Format of Events' describes rules to generate unique EventIds from the error code.
The MIB xxxDevEvIndex object should provide indication of relative ordering of events in the log.

In case of creation of local volatile and local-nonvolatile logs, it is required a method for synchronizing xxxDevEvIndex values between the two local logs after reboot.
In such case the following procedure must be used after reboot:

- The values of xxxDevEvIndex maintained in the local non-volatile log must be renumbered beginning with 1.
- The local-volatile log must then be initialized with the contents of the local non-volatile log.
- The first event recorded in the new active session's local-volatile log must use as its xxxDevEvIndex the value of (last restored non-volatile xxxDevEvIndex + 1).

A reset of the log initiated through an SNMP SET of the xxxDevEvControl object must clear both the local-volatile and local-nonvolatile logs.

3.1.3.1 Format of Events

The following sections explain in detail how to report these events via any of the three mechanisms (local event logging, SNMP trap and syslog).

3.1.3.1.1 SNMP TRAP/INFORM

A SS/BS MUST send the following generic SNMP traps, as defined in standard MIB [RFC 3418] and [RFC 2863]:

- ColdStart (WarmStart is optional) [RFC 3418]
- Linkup [RFC 2863]
- LinkDown [RFC 2863]
- SNMP authentication-Failure [RFC 3418]

A SS/BS MUST implement SNMP traps defined in the xxx-DEVICE-TRAPMIB.

- BS/SS should support INFORM.

INFORM is a variation of trap and requires the receiving host to acknowledge the arrival of an InformRequest-PDU with an InformResponse-PDU. An InformRequest-PDU is exactly the same as a trap-PDU except that the value in the PDU-type field is 6 for InformRequest-PDU instead of 7 for SNMPv2-trap-PDU.

- When an SNMP trap defined in the xxx-DEVICE-TRAP-MIB is enabled in a SS, it MUST send notifications for any event in its category whose priority is either “error” or “notice”.
- When the SNMP trap defined in the xxx-DEVICE-TRAP-MIB is enabled in a BS, it MUST send notifications for an event whose priority is “critical” or “error” or “warning” or “notice”.

Vendor-specific events reportable via SNMP TRAP must be described in the vendor documents. Vendors can also define vendor-specific SNMP traps and MUST do so in the private MIBs.

3.1.3.1.2 SYSLOG message format

For 802.16 events, the SS’s Syslog message MUST be sent in the following format:

- <*level*>SS [*vendor*]: <*eventId*> *text vendor-specific-text*
    - *Level* is an ASCII representation of the event priority.

The BS’s Syslog message MUST be sent in the following format:

- <*level*>*TIMESTAMP HOSTNAME B*S [*vendor*]: <*eventId*> *text vendor-specific-text*

Where:

- *Level* is an ASCII representation of the event priority
- *TIMESTAMP* and *HOSTNAME* MAY be sent after <level> by the BS. If the *TIMESTAMP* and *HOSTNAME* fields are sent, they MUST sent together. The one space after *TIMESTAMP* is part of the *TIMESTAMP* field. The one space after *HOSTNAME* is part of the *HOSTNAME* field.
- If no *HOSTNAME* available then we have to discuss a unique identifier for the BS in such situations.
- *Vendor* is the vendor name for the vendor-specific SYSLOG

- *EventId* is an ASCII representation of the INTEGER number in decimal format, enclosed in angle brackets, which uniquely identifies the type of event. For the standard events this number is converted from the error code using the following rules:
  - The number is an eight-digit decimal number.
  - The first two digits (left-most) are the ASCII code for the letter in the Error code.
  - The next four digits are filled by 2 or 3 digits between the letter and the dot in the Error code with zero filling in the gap in the left side.
  - The number fills the last two digits after the dot in the Error code with zero filling in the gap in the left side.
  - The first letter of an error code is always in upper case.
- *Text*: for the standard 802.16 messages, this string MUST contain the textual description as defined in the future event table.
- *Vendors for vendor-specific information may provide vendor-specific-text*.

3.1.3.2 Proposed Events for SS's

Emergency event (priority 1) - Reserved for vendor-specific 'fatal' hardware or software errors that prevents normal system operation and causes reporting system to reboot. Vendors may define their own set of emergency events.

Alert event (priority 2) - A serious failure, which causes reporting system to reboot but it is not caused by h/w or s/w malfunctioning. After recovering from the critical event, the system MUST send a cold/warm start notification. The alert event could not be reported as a Trap or SYSLOG message and MUST be stored in the internal log file. The code of this event MUST be saved in non-volatile memory and reported later.

Critical event (priority 3) - A serious failure that requires attention and prevents the device from transmitting data but could be recovered without rebooting the system. After recovering from the error event SS MUST send the Link Up notification. Critical events could not be reported as a Trap or SYSLOG message and MUST be stored in the internal log file. The code of this event MUST be reported later.

Error event (priority 4) - A failure occurred that could interrupt the normal data flow but will not cause the SS to re-register. Error events could be reported in real time by using the trap or SYSLOG mechanism.

Warning event (priority 5) - A failure occurred that could interrupt the normal data flow but will not cause the SS to re-register. 'Warning' level is assigned to events both SS and BS have information about. To prevent sending the same event both from the SS and the BS, the trap and Syslog reporting mechanism is disabled by default for this level.

Notice event (priority 6) - The event is important, but is not a failure and could be reported in real time by using the trap or SYSLOG mechanism.

Informational event (priority 7) - The event is of marginal importance, and is not failure, but could be helpful for tracing the normal modem operation.

Debug event (priority 8) Reserved for vendor-specific non-critical events.

Event Priority	Local-Log Non-volatile (bit-0)	SNMP Trap (bit-1)	SYSLOG (bit-2)	Local-Log Volatile (bit-3)
1 Emergency	Yes	No	No	No or Yes*
2 Alert	Yes	No	No	No or Yes*
3 Critical	Yes	No	No	No or Yes*
4 Error	No or Yes**	Yes	Yes	Yes
5 Warning	No or Yes**	No	No	Yes
6 Notice	No or Yes**	Yes	Yes	Yes
7 Informational	No or Yes**	No	No	No
8 Debug	No	No	No	No

Table 1. Proposed reporting mechanism on SS

3.1.3.3 Proposed Events for BS's

BS uses the same levels of the event priorities as a SS; however, the severity definition of the events is different.

- Events with the severity level of Warning and less specify problems that could affect individual user.
- Severity level of 'Error' indicates problems with a group of SSs (for example SSs that share the same sector if the case).
- Severity level of 'Critical' indicates problem that affects whole system operation, but is not a faulty condition of the BS device.

In all these cases the BS MUST be able to send SYSLOG event and (or) SNMP TRAP to the NMS.

- Severity level of 'Emergency' is vendor-specific and indicates problems with the BS hardware or software, which prevents BS operation.

Event Priority	Local-Log Non-volatile (bit-0)	SNMP Trap (bit-1)	SYSLOG (bit-2)	Local-Log Volatile (bit-3)
1 Emergency	Yes	No	No	No or Yes*
2 Alert	Yes	No	No	No or Yes*
3 Critical	Yes	Yes	Yes	No or Yes*
4 Error	No or Yes**	Yes	Yes	Yes
5 Warning	No or Yes**	Yes	Yes	Yes
6 Notice	No or Yes**	Yes	Yes	Yes
7 Informational	No	No	No	No
8 Debug	No	No	No	No

Table 2. Proposed reporting mechanism on BS

3.1.4 Event Table

Event table splits into two sections:

- Standard events
- Vendor specific events

Procedure could be:

- Scanning & Synchronization
- Ranging
  - RNG-REQ – is a state
- Registration
- DHCP
- DSx
- SW Upgrade
  - Upgrade Fail – is a state

The format for defining all events could be this one:

Table 3.Log Events presented below in its first draft version contains only network initialization and periodic ranging events (a full list of events is to be discussed)

	Procedure	State	SS priority	BS priority	Event Message	Notes	Error Code Set	Event ID	Trap Name
1	init	SYNC	critical		Lost DL-MAP		S01.0	83000100	
2	init	SYNC	critical		Lost DL Sync		S02.0	83000200	
3	init	SYNC	notice		Sync completed	For SS SYSLOG only, append: MAC Addr: <P2>. P2 = BS MAC address	S03.0	83000300	
4	init	SYNC	critical		No UCD Received - Timeout		S04.0	83000400	
5	init	SYNC	critical		UCD invalid or channel unusable		S05.0	83000500	
6	init	SYNC	notice		Uplink Params Aquired		S06.0	83000600	
7	init	SYNC	critical		Invalid UL Parameter		S07.0	83000700	

8	init	RANG	notice		Received Broadcast for Ranging Opportunity	For SS SYSLOG only, append: MAC Addr: <P2>. P2 = BS MAC address	R01.0	82000100	
9	init	RANG	critical		No Maintenance Broadcasts for Ranging opportunities received - T2 time-out		R02.0	82000200	
10	init	RANG	notice		Unicast Initial Ranging Started		R03.0	82000300	
11	init	RANG	critical		Unicast Initial Ranging – No Response received - T3 timeout		R04.0	82000400	
12	init	RANG	critical		Unicast Initial Ranging attempted – No response – Retries exhausted		R05.0	82000500	
13	init	RANG		notice	Initial Ranging Request Received (report MAC Address)	For BS SYSLOG only, append: MAC Addr: <P1>. P1 = SS MAC address	R06.0	82000600	
14	per	RANG	critical		Received Response to Broadcast Maintenance Request, But no Unicast Maintenance opportunities received - T4 timeout		R07.0	82000700	
15	init	RANG	critical		Unicast Initial Ranging Request Retries exhausted		R08.0	82000800	
16	init	RANG	notice		Initial Ranging - Subscriber goes to minimum power		R09.0	82000900	
17	init	RANG	notice		Initial Ranging - Subscriber Power Increased		R10.0	82001000	
18	per	RANG	critical		Unicast Ranging Received Abort Response - Re-initializing MAC	For SS SYSLOG only, append: MAC Addr: <P2>. P2 = BS MAC address	R11.0	82001100	
19	per	RANG		warning	Unicast Ranging Sent Abort Response (report MAC address)	For BS SYSLOG only, append: MAC Addr: <P1>. P1 = SS MAC address	R12.0	82001200	wmanBsSsStatusNotificationTrap,wmanIfBsSsStatusValue=2
20	init	RANG	notice	notice	Initial Ranging Successful (report MAC Address in case of BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG append: MAC Addr: <P2>. P2 = BS MAC address	R13.0	82001300	wmanBsSsStatusNotificationTrap,wmanIfBsSsStatusValue=1
21	init	RANG	notice		Initial Ranging Local Params adjusted after RNG-RSP	For SS SYSLOG only, append: MAC Addr: <P2>. P2 = BS MAC address	R14.0	82001400	

22	per	RANG		warning	Maintainance Ranging T27 Timeout for Subscriber (report MAC address). Ranging opportunity issued	For BS SYSLOG only, append: MAC Addr: <P1>. P1 = SS MAC address	R15.0	82001500	
23	per	RANG		warning	Maintainance Ranging Signal not present in unicast uplink burst grant for Subscriber (report MAC address). Invited counter incremented	For BS SYSLOG only, append: MAC Addr: <P1>. P1 = SS MAC address	R16.0	82001600	
24	per	RANG		warning	Maintainance Ranging Invited counter maximum reached - remove Subscriber (report MAC address)	For BS SYSLOG only, append: MAC Addr: <P1>. P1 = SS MAC address	R17.0	82001700	wmanBsSsStatusNotificationTrap,wmanIfBsSsStatusValue=2
25	per	RANG		warning	Maintainance Ranging Signal Detected not good enough. Correction counter reached maximum - remove Subscriber (report MAC address)	For BS SYSLOG only, append: MAC Addr: <P1>. P1 = SS MAC address	R18.0	82001800	wmanBsSsStatusNotificationTrap,wmanIfBsSsStatusValue=2
26	per	RANG		warning	Maintainance Ranging Signal Detected not good enough. Correction required for Subscriber (report MAC address)	For BS SYSLOG only, append: MAC Addr: <P1>. P1 = SS MAC address	R19.0	82001900	
27	per	RANG	notice	notice	Maintainance Ranging Successful (report MAC Address of Subscriber in case BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG append: MAC Addr: <P2>. P2 = BS MAC address	R20.0	82002000	wmanBsSsStatusNotificationTrap,wmanIfBsSsStatusValue=1
28	per	RANG	notice		Maintainance Ranging Signal Corrections made successfully	For SS SYSLOG only, append: MAC Addr: <P2>. P2 = BS MAC address	R21.0	82002100	
29	per	RANG	warning		Maintainance Ranging Signal Correction anomaly has occurred	For SS SYSLOG only, append: MAC Addr: <P2>. P2 = BS MAC address	R22.0	82002200	
30	per	RANG	notice	notice	Data Grant (report MAC Address of Subscriber in case BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG append: MAC Addr: <P2>. P2 = BS MAC address	R23.0	82002300	

24	init	SBC	critical		SBC Failed - Request		N01.0	78000100	

				Timeout, T18				
init	SBC	critical		SBC Failed - Request Retries Exhausted		N02.0	78000200	
init	SBC		warning	SBC Failed - Wait Timeout, T9 (report MAC Address)	For BS SYSLOG only, append: MAC Addr: <P1>. P1 = SS MAC address	N03.0	78000300	wmanBsSsStatusNotificationTrap,wmanIfBsSsStatusValue=7
init	SBC	critical	warning	SBC - None of the capabilities required are supported (report MAC Address in case BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG append: MAC Addr: <P2>. P2 = BS MAC address	N04.0	78000400	wmanBsSsStatusNotificationTrap,wmanIfBsSsStatusValue=7
init	SBC	notice	notice	SBC Successful (report MAC Address in case of BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG append: MAC Addr: <P2>. P2 = BS MAC address	N05.0	78000500	wmanBsSsStatusNotificationTrap,wmanIfBsSsStatusValue=6
init	AUTH	notice	notice	Auth General - Communication Established (report MAC Address in case of BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG append: MAC Addr: <P2>. P2 = BS MAC address	A01.0	65000100	
init	AUTH	warning		Auth General - Request Timeout		A02.0	65000200	
init	AUTH	warning	error	Auth General - Grace Timeout (report MAC Address in case of BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG append: MAC Addr: <P2>. P2 = BS MAC address	A03.0	65000300	
init	AUTH	notice	notice	Auth General - Re-Auth (report MAC Address in case of BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG append: MAC Addr: <P2>. P2 = BS MAC address	A04.0	65000400	
init	AUTH	warning	error	Auth General - Unsupported Crypto Suite (report MAC Address in case of BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG append: MAC Addr: <P2>. P2 = BS MAC address	A05.0	65000500	

34	init	AUTH	warning	error	Auth Invalid – No Information (report MAC Address in case of BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG append: MAC Addr: <P2>. P2 = BS MAC address	A06.0	65000600	
35	init	AUTH	warning	error	Auth Invalid – Unauthorized Subscriber (report MAC Address in case of BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG append: MAC Addr: <P2>. P2 = BS MAC address	A07.0	65000700	
36	init	AUTH	warning	error	Auth Invalid - Unsolicited (report MAC Address in case of BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG append: MAC Addr: <P2>. P2 = BS MAC address	A08.0	65000800	
37	init	AUTH	warning	error	Auth Invalid – Invalid Key Sequence Number (report MAC Address in case of BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG append: MAC Addr: <P2>. P2 = BS MAC address	A09.0	65000900	
38	init	AUTH	warning	error	Auth Invalid - Message (Key Request) Authentication Failure (report MAC Address in case of BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG append: MAC Addr: <P2>. P2 = BS MAC address	A10.0	65001000	
39	init	AUTH	error	error	Permanent Auth Reject – Unknown Manufacturer (report MAC Address in case of BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG append: MAC Addr: <P2>. P2 = BS MAC address	A11.0	65001100	wmanBsSsStatusNotificationTrap,wmanIfBsSsStatusValue=9
40	init	AUTH	error	error	Permanent Auth Reject – Invalid signature on Subscriber certificate (report MAC Address in case of BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG append: MAC Addr: <P2>. P2 = BS MAC address	A12.0	65001200	wmanBsSsStatusNotificationTrap,wmanIfBsSsStatusValue=9
41	init	AUTH	error	error	Permanent Auth Reject – ASN.1 parsing failure (report MAC Address in case of BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG	A13.0	65001300	wmanBsSsStatusNotificationTrap,wmanIfBsSsStatusValue=9

					append: MAC Addr: <P2>. P2 = BS MAC address			
init	AUTH	error	error	Permanent Auth Reject – Inconsistencies between certificate and PKM attributes (report MAC Address in case of BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG append: MAC Addr: <P2>. P2 = BS MAC address	A14.0	65001400	wmanBsSsStatusNotificationTrap,wmanIfBsSsStatusValue=9
init	AUTH	error	error	Permanent Auth Reject – Incompatible security capabilities (report MAC Address in case of BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG append: MAC Addr: <P2>. P2 = BS MAC address	A15.0	65001500	wmanBsSsStatusNotificationTrap,wmanIfBsSsStatusValue=9
init	AUTH	warning	error	Auth Reject – No Information (report MAC Address in case of BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG append: MAC Addr: <P2>. P2 = BS MAC address	A16.0	65001600	wmanBsSsStatusNotificationTrap,wmanIfBsSsStatusValue=9
init	AUTH	warning	error	Auth Reject – Unauthorized Subscriber (report MAC Address in case of BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG append: MAC Addr: <P2>. P2 = BS MAC address	A17.0	65001700	wmanBsSsStatusNotificationTrap,wmanIfBsSsStatusValue=9
init	AUTH	warning	error	Auth Reject – Unauthorized SAID (report MAC Address in case of BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG append: MAC Addr: <P2>. P2 = BS MAC address	A18.0	65001800	wmanBsSsStatusNotificationTrap,wmanIfBsSsStatusValue=9
init	AUTH	alert	error	Auth Reject - Subscriber Certificate Error (report MAC Address in case of BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG append: MAC Addr: <P2>. P2 = BS MAC address	A19.0	65001900	wmanBsSsStatusNotificationTrap,wmanIfBsSsStatusValue=9
init	AUTH	notice	notice	Authorized (report MAC Address in case of BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG append: MAC Addr: <P2>. P2 = BS MAC address	A20.0	65002000	wmanBsSsStatusNotificationTrap,wmanIfBsSsStatusValue=8

49	init	AUTH	notice		Auth Pending	For SS SYSLOG only, append: MAC Addr: <P2>. P2 = BS MAC address	A21.0	65002100	
50	init	AUTH	notice		Auth Complete	For SS SYSLOG only, append: MAC Addr: <P2>. P2 = BS MAC address	A22.0	65002200	
51	init	AUTH	notice		Auth Stop	For SS SYSLOG only, append: MAC Addr: <P2>. P2 = BS MAC address	A23.0	65002300	
52	init	AUTH	warning		TEK General - Request Timeout	For SS SYSLOG only, append: MAC Addr: <P2>. P2 = BS MAC address	A24.0	65002400	
53	init	AUTH	warning		TEK General - Refresh Timeout	For SS SYSLOG only, append: MAC Addr: <P2>. P2 = BS MAC address	A25.0	65002500	
54	init	AUTH	warning	error	Key Reject – No Information (report MAC Address in case of BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG append: MAC Addr: <P2>. P2 = BS MAC address	A26.0	65002600	
55	init	AUTH	warning	error	Key Reject – Unauthorized SAID (report MAC Address in case of BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG append: MAC Addr: <P2>. P2 = BS MAC address	A27.0	65002700	
56	init	AUTH	warning	error	TEK Invalid – No Information (report MAC Address in case of BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG append: MAC Addr: <P2>. P2 = BS MAC address	A28.0	65002800	
57	init	AUTH	warning	error	TEK Invalid – Invalid Key Sequence Number (report MAC Address in case of BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG append: MAC Addr: <P2>. P2 = BS MAC address	A29.0	65002900	

58	init	REG	critical		Registration Response Timeout, T6	For SS SYSLOG only, append: MAC Addr: <P2>. P2 = BS MAC address	G01.0	71000100	

59	init	REG		warning	Registration Request Timeout,T17 (report MAC Address)	For BS SYSLOG only, append: MAC Addr: <P1>. P1 = SS MAC address	G02.0	71000200	wmanBsSsStatusNotificationTrap,wmanIfBsSsStatusValue=4
60	init	REG	critical		Registration Retries Exhausted	For SS SYSLOG only, append: MAC Addr: <P2>. P2 = BS MAC address	G03.0	71000300	
61	init	REG	notice	notice	Registration Completed (report MAC Address in case of BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG append: MAC Addr: <P2>. P2 = BS MAC address	G04.0	71000400	wmanBsSsStatusNotificationTrap,wmanIfBsSsStatusValue=3
62	init	REG	notice	notice	Registration Completed - Subscriber NOT Managed (report MAC Address in case of BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG append: MAC Addr: <P2>. P2 = BS MAC address	G05.0	71000500	wmanBsSsStatusNotificationTrap,wmanIfBsSsStatusValue=3
63	init	REG	critical		Registration Failed - Reinitializing MAC	For SS SYSLOG only, append: MAC Addr: <P2>. P2 = BS MAC address	G06.0	71000600	
64	init	REG	critical	warning	Registration Failed - Invalid HMAC (report MAC Address in case of BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG append: MAC Addr: <P2>. P2 = BS MAC address	G07.0	71000700	wmanBsSsStatusNotificationTrap,wmanIfBsSsStatusValue=4
65	init	REG		error	Required IP Version Not Supported by the Base Station (report MAC Address)	For BS SYSLOG only, append: MAC Addr: <P1>. P1 = SS MAC address	G08.0	71000800	

66	init	DHCP	critical		DHCP FAILED - Discover sent, no offer received		D01.0	68000100	
67	init	DHCP	critical		DHCP FAILED - Request sent, No response		D02.0	68000200	
68	init	DHCP	critical		DHCP FAILED - Requested Info not supported.		D03.0	68000300	
69	init	DHCP	critical		DHCP FAILED – Response doesn't contain ALL the valid fields		D04.0	68000400	

70	init	TOD	warning		TOD Warning ToD request sent - No Response received		T01.0	84000100	
71	init	TOD	warning		TOD Warning ToD		T02.0	84000200	

				Response received – Invalid data format				
72	init	TFTP	critical		TFTP Failed - Request sent - No Response	For SS SYSLOG only: append: File name = <P1> P1 = requested file name	F01.0	70000100	
73	init	TFTP	critical		TFTP Failed - configuration file NOT FOUND	For SS SYSLOG only: append: File name = <P1> P1 = requested file name	F02.0	70000200	
74	init	TFTP	critical		TFTP Failed - OUT OF ORDER packets	For SS SYSLOG only: append: File name = <P1> P1 = requested file name	F03.0	70000300	
75	init	TFTP	critical		TFTP File complete - but failed Message Integrity check MIC	For SS SYSLOG only: append: File name = <P1> P1 = requested file name	F04.0	70000400	
76	init	TFTP	critical		TFTP File complete - but missing mandatory TLV	For SS SYSLOG only: append: File name = <P1> P1 = requested file name	F05.0	70000500	
77	init	TFTP	critical		TFTP Failed - file too big	For SS SYSLOG only: append: File name = <P1> P1 = requested file name	F06.0	70000600	
78	init	TFTP		warning	TFTP Failed - TFTP-CPLT not received,T13 Timeout	For BS SYSLOG only, append: MAC Addr: <P1>. P1 = SS MAC address	F07.0	70000700	wmanBsSsStatusNotificationTrap,wmanIfBsSsStatusValue=11
79	init	TFTP	notice	notice	TFTP Successful (TFTP-CPLT received in case of BS)	For BS SYSLOG append: MAC Addr: <P1>. P1 = SS MAC address For SS SYSLOG append: File name = <P1> P1 = requested file name	F08.0	70000800	wmanBsSsStatusNotificationTrap,wmanIfBsSsStatusValue=10

Table 3.Log Events

3.1.5 Minimum Requirements

- BS/SS must comply to the events structure specified above
- BS/SS must support at least SNMP and fault management reporting traps.
- BS/SS must support at least one of the choices of volatile/non-volatile memory storage
- BS/SS must implement MIB's that support fault management

3.1.6 MIB Status

Traps are events escalated to NMS.MIB variables that refers to traps and their associated object variables as per WMAN-IF-MIB version 8-16 are defined below:

Base station traps:
wmanBsSsStatusNotificationTrap
wmanBsSsDynamicServiceFailTrap
wmanBsSsRssiStatusChangeTrap
wmanSsBPKMFailTrap
wmanBsPowerStatusChangeTrap
wmanBsFanStatusTrap
wmanBsTemperatureChangeTrap

And their associated variables:
wmanIfBsThresholdConfigTable
wmanIfBsSsNotificationObjectsTable
wmanIfBsNotificationObjectsTable
wmanIfBsTrapControlRegister

Subscriber Station traps:
wmanSsTLVUnknownTrap
wmanSsDynamicServiceFailTrap
wmanSsDHCPSuccessTrap
wmanSsRssiStatusChangeTrap

And their associated variables:
wmanIfSsMacAddress
wmanIfSsUnknownTlv
wmanIfSsDynamicServiceType
wmanIfSsDynamicServiceFailReason
wmanIfSsRssiStatus
wmanIfSsRssiStatusInfo
wmanIfSsTrapControlRegister
wmanIfSsRssiLowThreshold
wmanIfSsRssiHighThreshold

3.2 Configuration

3.2.1 Software Control and Management

Configuration management is concerned with initializing, maintaining, adding and updating network components. Unlike performance, fault, and account management, which emphasize network monitoring, configuration management is primarily concerned with network control. Network control, as defined by this interface specification, is concerned with modifying parameters in and causing actions to be taken by the SS and/or BS. Configuration parameters could include both identifiable physical resources (for example, RF or Ethernet Interface) and logical objects.
Modifying the configuration information of a SS and/or BS can be categorized as *non-operational* or *operational*.

- **Non-operational** changes occur when a manager issues a modify command to a SS/BS, and the change doesn't affect the operating environment. For example, a manager may change contact information, such as the name and address of the person responsible for a BS.
- **Operational** changes occur when a manager issues a modify command to a SS/BS, and the change affects the underlying resource or environment. For example, a manager may a reset command, which in turn will cause the SS to reboot.

To adjust the necessary attribute values, the SS and BS MUST support MIB objects.
While the network is in operation, configuration management is responsible for monitoring the configuration and making changes in response to commands via SNMP or in response to other network management functions.
A fault management function may detect and isolate a fault and may issue a configuration management change to bypass the fault.

3.2.1.1 Version Control

The SS SHOULD support software revision and operational parameter configuration interrogation.
The SS MUST include at least the hardware version, Boot ROM image version, vendor name, software version, and model number in the sysDescr object (from [RFC 3418]). The SS MUST support xxxDevSwCurrentVers MIB object and the object MUST contain the same software revision information as shown in the software information included in the sysDescr object.
The format of the specific information contained in the sysDescr MUST be as follows:

- **To report Format of each field**
  - Hardware Version HW_REV: <Hardware version>
  - Vendor Name VENDOR: <Vendor name>
  - Boot ROM BOOTR: <Boot ROM Version>
  - Software Version SW_REV: <Software version>
  - Model Number MODEL: <Model number>

Each type-value pair MUST be separated with a colon and blank space. Each pair is separated by a ";" followed by a blank. For instance, a sysDescr of a SS of vendor X, hardware version 5.2, Boot ROM version 1.4, SW version 2.2, and model number X MUST appear as follows:

Any text<<HW_REV: 5.2; VENDOR: X; BOOTR: 1.4; SW_REV 2.2; MODEL: X>>any text

The SS MUST report at least all of the information necessary in determining what SW the SS is capable of being upgraded to. If any fields are not applicable, the SS MUST report "NONE" as the value. For example SS with no BOOTR, SS will report BOOTR: NONE.
The SS MUST implement the xxxDevSwCurrentVers object in the MIB in order to report the current software version.
The intent of specifying the format of sysObjectID and sysDescr is to define how to report information in a consistent manner so that sysObjectID and sysDescr field information can be programmatically parsed.
This format specification does not intend to restrict the vendor's hardware version numbering policy.
The BS MUST implement the sysDescr object (from [RFC 3418]). For the BS, the format and content of the information in sysDescr is vendor-dependent.

3.2.2 System Initialization and Configuration

There are several methods available to configure SS and BS including console port, SNMP set, or possibly configuration file, and configuration-file-based SNMP encoded object.

- The SS SHOULD support system initialization and configuration via at least SNMP set. Configuration file, configuration-file-based SNMP encoded object could be options.
- SS SHOULD support initialization and configuration via both RF and Ethernet interface.
- SS local capability of initialization and configuration must be available for debugging purposes. (No general requirements)
- The BS SHOULD support system initialization and configuration via telnet connection, console port, and SNMP set.

The SS and BS (only BS that support configuration by configuration file) MUST support any valid configuration file regardless of configuration file size.

3.2.3 Secure Software Upgrades

TBD.

3.2.4 SS Device Provisioning (Network Entry)

SS network entry is described in 802.16 specification [1].
See **Subscriber station** under wmanIfSsConfigFileEncodingTable

As SS is using DHCP provisioning mode this document has to detail and add if necessary any parameter or option, that must be used during a DHCP session.
TBD. Reevaluate options present in DHCP messages.
If BS will use DHCP Relay Agent, document must complete a section related to that.
Any particular requirement to DHCP servers set of features and supported RFC's has to be provided here including options to be followed by SS.

3.2.4.1 DHCP Relay Agent

Optional implementation of DHCP Relay Agent on BS <TBD>
If DHCP Relay implemented DHCP message fields should be reviewed.

3.2.5 Minimal Requirements

- **SS/BS must support SNMP**
- **SS must include in sysDescr object fields mentioned in 3.2.1.1**
- **SS/BS must support SW upgrades by TFTP**
- **SS SHOULD support SNMP management traffic across both the RFI and Ethernet interfaces regardless of the SS's connectivity state.**
- **SS/BS must support at least one other method of Initialization and Configuration independent of NMS resources.**

3.2.6 MIB Status

MIB variables addressing **configuration** in 802.16 are found within the following nodes
For Base station
under wmanIfBsConfigurationTable
under wmanIfBsRegisteredSsTable

For Subscriber station
under wmanIfSsConfigFileEncodingTable
under wmanIfSsConfigurationTable

Common (BS and SS) Configuration Objects:
under wmanIfCmnBsSsConfigurationTable

3.3 User Account Management Interface Specification

The Subscriber Account Management Interface Specification is defined to enable prospective vendors of SS's and BS's to address the operational requirements of subscriber account management in a uniform and consistent

manner. It is the intention that this would enable operators and other interested parties to define, design and develop Operations Support Systems necessary for the commercial deployment of different class of services over wireless networks with accompanying usage-based billing of services for each individual subscriber.
Subscriber Account Management described here refers to the following business processes and terms:

- Class of Service Provisioning Processes, which are involved in the automatic and dynamic provisioning and enforcement of subscribed class of policy-based service level agreements (SLAs);
- Usage-Based Billing Processes, which are involved in the processing of bills based on services rendered to and consumed by paying subscribers.

This Specification focuses primarily on bandwidth-centric usage-based billing scenarios.
In order to develop Subscriber Account Management Specification, it is necessary to consider high-level tasks common to wireless operators and the associated operational scenarios.
This paragraph proposes usage of IPDR Network model as interface .The model is used in:

- WiFi and public WLAN access
- DOCSIS
- VoIP
- IPDR is suitable for nomadic/roaming scenarios
- It scales well to 802.16 mobility extension

For details see, http://www.ipdr.org/

3.3.1 Service Flows and User Usage Billing

As defined in 5.1.2 Service Class Name (SCN) provides a handle to an associated QoS Parameter Set (QPS) template. Service Flows that are created using an SCN are considered to be "named" Service Flows.
The SCN identifies the service characteristics of a Service Flow to external systems such as a billing system or customer service system. For consistency in billing, operators should ensure that SCNs are unique within an area serviced by the same OSS that utilizes this interface.

A **Service Package** implements a Service Level Agreement (SLA) between the Service Provider and its Subscribers. A Service Package might be known by a name such as Gold, Silver, or Bronze.
Service Package is itself implemented by the set of named Service Flows (using SCNs) that are sent to the SS. Note that many Subscribers can have assigned the same Service Package.

Internally, each active Service Flow is identified by a 32-bit SFID assigned by the BS to a specific SS (relative to the RFI interface). For billing purposes, however, the SFID is not sufficient as the only identifier of a Service Flow because the billing system cannot distinguish the class of service being delivered by one SFID from another.

Therefore, the SCN is necessary, in addition to the SFID, to identify the Service Flow’s class of service characteristics to the billing system.

The billing system can then rate the charges differently for each of the Service Flow traffic counts based on its Service Class (e.g. Gold octet counts are likely to be charged more than Bronze octet counts). Thus, the billing system obtains from the BS the traffic counts for each named Service Flow (identified by SFID and SCN) that a subscriber uses during the billing data collection interval.
Note that the SFID is the primary key to the Service Flow.
When an active Service Flow exists across multiple sequential billing files the SFID allows the sequence of recorded counter values to be correlated to the same Service Flow instance.

3.3.2 High-Level Requirements for Subscriber Usage Billing Records

The BS, or its supporting Element Management System (EMS), must provide formatted Subscriber Usage Billing Records for all subscribers attached to the BS on demand to a mediation system or a billing system. The following are the requirements for processing and transmitting Subscriber Usage Billing Records:

1.The minimum billing record collection interval that must be supported by a BS should be around 15 minutes or TBD.

2. The Subscriber Usage Billing File must identify the BS by host name and IP address and the time that the billing file was created. The sysUpTime value for the BS must also be recorded.

3. Subscriber usage billing records must be identified by SS MAC address (but not necessarily sorted). The Subscriber's current SS IP address must also be present in the billing record for the Subscriber. If the BS is tracking CPE IP addresses behind the Subscriber's SS, then these CPE IP addresses must also be present in the billing record.

4. Subscriber usage billing records must have entries for each active Service Flow (identified by SFID and Service Class Name) used by all SS's during the collection interval.
This includes all currently running Service Flows as well as all terminated Service Flows that were deleted and logged during the collection interval.
Note well that a provisioned or admitted state SF that was deleted before it became active is not recorded in the billing file, even though the BS logged it.

5. It must be possible to distinguish running Service Flows from terminated Service Flows in the billing records.
Internal BS Service Flow log records should not be deleted from the BS until after they have been recorded in a billing file stored in non-volatile storage.
The BS must maintain a separate view of the internal Service Flow log for SNMP access via the QOS-MIB. It must not be possible to delete internal Service Flow log entries via SNMP until; the billing formatter has released them. A terminated Service Flow must be reported into a Billing File exactly once.

6. It must be possible to identify the Service Flow direction as upstream or downstream without reference to the Service Class Name. The number of packets and octets passed must be collected for each upstream and downstream Service Flow.
Note that since it is possible for a Subscriber to change from one service package to another and back again or to have dynamic service flows occur multiple times, it is possible that there will be multiple entries for a given SCN within a Subscriber's billing record for the collection period.
This could also occur if a SS re-registers for any reason (such as SS power failure or lost link).

7. All traffic counters must be based on absolute 64-bit counters as maintained by the BS. These counters must be reset to zero by the BS if it re-initializes its management interface. The BS sysUpTime value is used to determine if the management interface has been reset between adjacent collection intervals. It is expected that the 64-bit counters will not roll over within the service lifetime of the BS.
Note: Subscriber billing records are a method of byte usage accounting only. Some types of Service Flows can consume system resources without bytes actually being passed (e.g. an active RTPS flow or an admitted UGS flow). Billing for these types of resources should be clarified.

8. To facilitate processing of the Subscriber Usage Billing Records by a large number of diverse billing and mediation systems an Extensible Markup Language (XML) format would be required. Specifically, the IP Detail Record (IPDR) standard as described in IPDR.org's Network Data Management - Usage, Version 3.1 ([NDMU 3.1]) could be used. See also http://www.ipdr.org for more information on the NDM-U specification and Service Specification Guidelines. Need some work in order to customize the generic format if needed.

9. To improve the performance of storage and transmission of the NDM-U XML format billing records a compressed file format would be required. Loss-less compression in GZIP 4.3 format as described in [[RFC 1952] could be used to store and transmit the billing file. It is expected that an IPDRv3 XML format-billing file will compress on the order of 30:1 or better. See also http://www.gnu.org/software/gzip for more information.

10. To improve the network performance of the billing collection activity, a reliable high-throughput TCP stream must be used to transfer billing records between the record formatter and the collection system. Standard FTP GET of the compressed (and optionally encrypted) billing file from the record formatter by the collection system should be supported.

11.To allow for decoupled scheduling, the billing collection cycle must be driven by the collection system through the standard FTP GET and FTP DELETE operations. Since the collection interval may vary over time, the record formatter is only required to maintain one current billing file in its FTP file system. The Collection system (operating on its own schedule) may retrieve the current billing file using FTP GET at any time after it has been constructed and placed in the FTP file system by the record formatter. The collection system must explicitly FTP DELETE the billing file when it no longer needs it. The retrieval model is detailed in one of the next paragraphs.

3.3.3 IP Detail Record (IPDR) Standard

The IPDR Organization (see http://www.ipdr.org) has defined a generic model for using XML Schema in IP Detail Recording applications. Industry specific IP billing applications such as the 802.16 Subscriber Usage Billing Record can be added to the IPDR standard by mapping the application semantics onto the NDM-U XML Schema syntax.

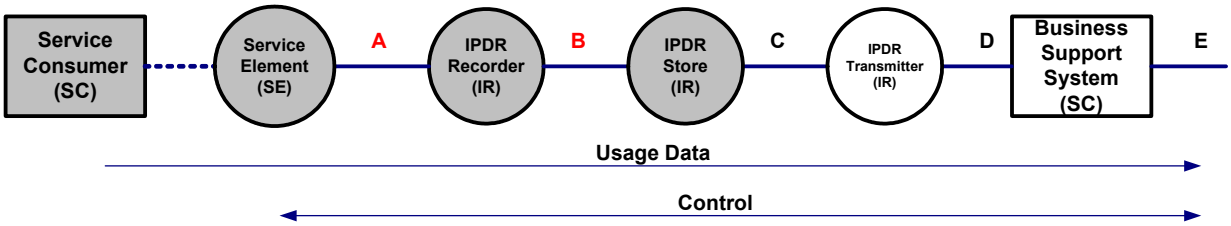

Figure 2. Basic Network Model (ref. [NDM-U 3.1] from www.ipdr.org)

3.3.3.1 IPDR Network Model

The IPDR Network Model is given in the [NDM-U 3.1] specification and is portrayed in Figure 2 above. Note that the highlighted blocks and interfaces are the only ones defined here. In this network model:

- The Service Consumer (SC) is the Wireless Service Subscriber identified by their 802.16 SS MAC address, current SS IP address, and current CPE IP addresses
- The Service Element (SE) is the BS identified by its host name, IP address, and current value of its sysUpTime object.
- The IPDR Recorder (IR) is the billing record formatter function that creates the [NDM-U 3.1] schema format XML IPDRs from the internal counters maintained by the BS for each Subscriber's running and terminated Service Flows.
- The IPDR Store (IS) is the function that maintains the billing file in the FTP file system and detects that the billing collector has deleted the billing file.

The IPDR Recorder and the IPDR Store are functions that may be implemented within the BS or hosted on another platform such as an Element Management System (EMS) or Record Keeping Server (RKS).

- The IPDR Transmitter (IT) represents the billing record collectors that retrieve the billing records from the IPDR Store.

In this specification the IT retrieves the compressed and possibly encrypted billing file from the IS on a collection cycle determined by the IT.

- Note that the A-interface is not specified by the NDM-U specification because it is an internal interface between the SE and the IR components.

- The B-interface between the IR and the IS component is also internal to the implementation and is not specified here.
- The C-interface is specified by the NDM-U specification as **a file of IPDR records formatted according to the IPDRdoc XML Schema (.xsd) files**. In addition, the billing file in the C-interface could be compressed.
- The D- and E interfaces are beyond the scope of this specification.

That assumes:

- **SE, IPDR Recorder, IPDR Store reside on BS (EMS)**

Note: The C-interface billing file MUST be implemented using the Data Systems Subscriber Usage Billing Record that has to be determined.

3.3.3.2 IPDR Record Structure

The [NDM-U 3.1] specification specifies the IPDRDoc record structure. The IPDRDoc XML schema defines the hierarchy of elements within the IPDR document that MUST be supported by the BS.
Note that the Type presented in figure bellow is the application specific implementation of the IPDR element. Thus, the 802.16 specific elements are sub elements of the IPDR element.

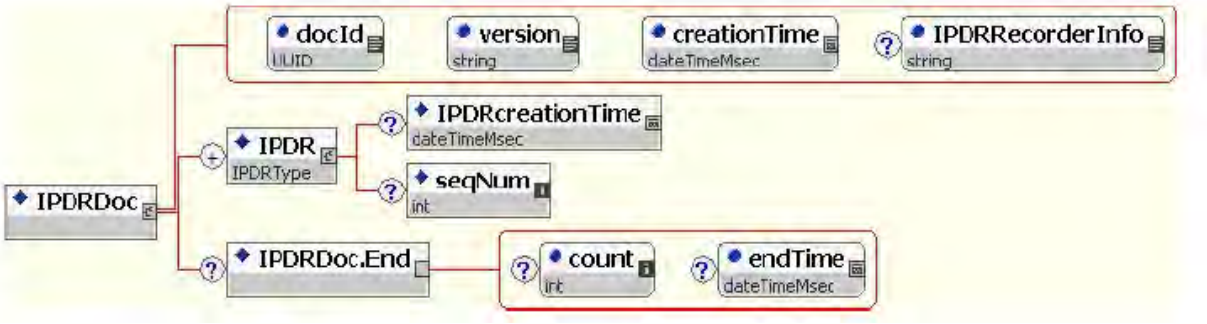

Figure 3. IPDRDoc Generic Structure

The structure presented above is independent of application.

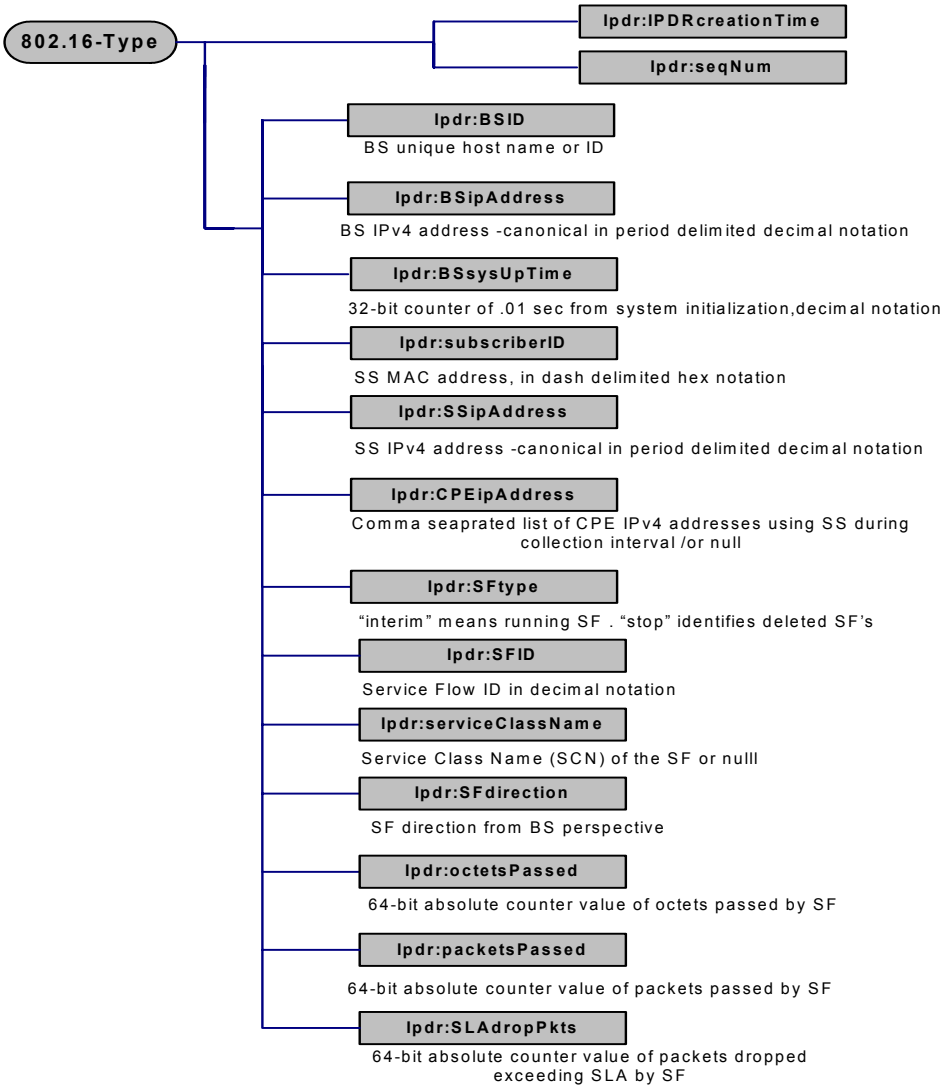

Figure 4. 802.16 IPDR Draft Schema

1. The IPDRDoc element is the outermost element that describes the IPDR billing file itself. It defines the XML namespace, the identity of the XML schema document, and the version of the specification, and the timestamp for the file, a unique document identifier, and the identity of the IPDR recorder. An IPDRDoc is composed of multiple IPDR records. The attributes for the IPDRDoc element MUST be as follows:

- Xmlns=http://www.ipdr.org/namespaces/ipdr. Constant: the XML namespace identifier. Defined by ipdr.org.

- Xmlns: xsi=http://www.w3.org/2001/XMLSchema-instance. Constant: the XML base schema identifier. Defined by ipdr.org.
- Xsi: schemaLocation="802.16-xx-xx.xsd". Constant: the name of the 802.16 application specific schema file.
- Version="3.1" Constant: the version of the IPDR document. Defined by ipdr.org.
- CreationTime ="yyyy-mm-ddThh: mm: ssZ". UTC time stamp at the time the billing file is created (in ISO format). For example: creationTime="2002-06-12T21: 11:21Z". Note that IPDR timestamps MUST always be in UTC/GMT (Z).
- DocId="<32-bit UTC timestamp>-0000-0000-0000-<48-bit MAC address>" .The unique document identifier.
    - The docId is in a simplified format that is compatible with the Universal Unique Identifier (UUID) format required by the IPDR NDM-U 3.1 specification.
    - The 32-bit UTC timestamp component MUST be the IPDRDoc creationTime in seconds since the epoch 1 Jan 1970 UTC formatted as eight hex digits.
    - The 48-bit MAC address component MUST be the Ethernet address of the BS management interface formatted as 12 hex digits.

All other components MUST be set to zero**. In the context of the minimum 15-minute IPDR billing file collection cycle specified in this document, this simplified UUID is guaranteed to be unique across all BS's and for the foreseeable future.**
For example: docId="3d07b8f9-0000-0000-0000-00015c11bfbe".

- IPDRRecorderInfo - identifies the IPDR Recorder (IR) from the network model (see figure above). This attribute MUST identify the billing record formatter .At this point has to be determined on how we use it:
    - Some implementations use fully qualified hostname of the BS or the EMS where the formatter resides.
    - If a hostname is not available, then IPv4 address of the BS (EMS) could be used, formatted in dotted decimal notation.

2. An IPDR element MUST describe a single Subscriber Usage Billing Record for a single 802.16 service flow. The IPDR is further structured into 802.16 specific sub elements that describe the details of the BS, the subscriber (SS and CPE), and the service flow itself. While the generic IPDR record structure is designed to describe most time-based and event-oriented IP services, this feature is not particularly relevant to the 802.16Data Service Subscriber Usage Billing Records and is largely ignored. This is because a service session at the BS is just the aggregate usage of an active Service Flow during the billing collection interval. Another way to look at it is as if there is really only one event being recorded: the billing collection event itself.
The attributes for the IPDR element are:

- Xsi: type="802.16-Type". Constant: identifies the application specific type of the IPDR record.

3. The IPDRcreationTime element identifies the time associated with the counters for this service flow. The format MUST be the same as the IPDRDoc creationTime attribute.

- IPDRcreationTime MUST be the same as the IPDRDoc creationTime when the service flow is still running (i.e. SFtype = Interim).
- IPDRcreationTime MUST point the time the service flow was deleted when the service flow has been terminated (i.e. SFtype = Stop). Note that a Stop IPDR is always earlier than the IPDRDoc creationTime.

Also, note that this sub element is optional in the basic IPDR 3.1 schema, but should be REQUIRED for all 802.16 IPDRs.

4. The seqNum element is an optional sub element of the basic IPDR 3.1 schema. It MUST NOT be used in 802.16 IPDRs. Note that there is no ordering implied in 802.16 IPDRs within an IPDRDoc.

5. The BSID element is a REQUIRED element that contains the fully qualified domain name (FQDN) of the BS or a unique ID if it exists. For example: 802.16bs01.mso.com. This element MUST be null if no FQDN exists. TBD.

6. The BSipAddress element contains the IP address of the management interface of the BS. This element is REQUIRED and MUST be represented in standard IPv4 decimal dotted notation (for example: 10.10.10.1).

7. The BSsysUpTime element contains the value of the sysUpTime SNMP object in the BS taken at the IPDRDoc creationTime. This element is REQUIRED and MUST be the count of 100ths of seconds since the BS management interface was initialized. If the BSsysUpTime regresses between adjacent IPDRDocs, then the BS management interface has been reset and all service flow counters have been reset to zero. Note well: this value MUST be the same for each IPDR within a given IPDRDoc file, regardless of the IPDRcreationTime of a given IPDR.

8. The subscriberId element contains the unique identifier of the subscriber. This element is REQUIRED and MUST be the subscriber's SS 48-bit MAC address formatted as dash delimited hex digits. For Example: 11-11-11-11-11-11.

9.The SSipAddress element contains the current IP address of the subscriber's SS. This element is REQUIRED and MUST be represented in standard IPv4 decimal dotted notation (for example, 10.100.100.123). Note that this address can change over a set of IPDRDoc files if the operator's DHCP server reassigns IP addresses to SS's.

10. The CPEipAddress element MUST contain a comma delimited list of the current IP addresses of all of the subscriber's CPE using this SSmodem **or null if there are none being tracked by the BS** (i.e.

<CPEipAddress></CPEipAddress> or <CPEipAddress/>). If there are multiple CPE using the SS, then there MUST be multiple CPE IP addresses in the list. Each CPE IP address MUST be represented in standard IPv4 decimal dotted notation (for example: 12.12.12.123 or 12.12.12.123, 12.12.12.124, 12.12.12.125).

11.The SFtype element identifies the kind of service flow being described by this IPDR. This element is REQUIRED and MUST have either of two values:

- "Interim" identifies this SF as currently running in the BS.
- "Stop" identifies this SF as having been terminated in the BS.

A running service flow has active counters in the BS and this IPDR MUST contain the current sample of these counters.
A terminated service flow has logged counters in the BS and this IPDR MUST contain the final counter values for this service flow.

Note: the internal logged SF counters on the BS MUST NOT be deleted until after the terminated service flow has been recorded into an IPDR record that has been stored in non-volatile memory, regardless of any other capability to manage them via SNMP through the QOS-MIB.

13.The SFID element contains the service flow identifier known to the BS. This element is REQUIRED and is needed to correlate the IPDRs for an individual service flow between adjacent IPDRDoc files when computing delta counters between samples. SFID element works here as long as is relative to the network scope. Otherwise, the SFID element MUST be formatted as ifIndex.

14.The serviceClassName element contains the name associated with the QoS parameter set for this service flow in the BS. The SCN is an ASCII string identifier, such as "GoldUp" or "SilverDn", that can be used by external operations systems to assign, monitor, and bill for different levels of bandwidth service without having to interpret the details of the QoS parameter set itself. A service flow is associated with an SCN whenever a SCN is used to define an active service flow. A dynamic service flow application such as VoIP may also assign an SCN to a service flow as a parameter during the dynamic creation of the service flow.

Note that use of SCNs is optional within the context of the 802.16 specifications, however, for operational purposes, especially when billing for tiered data services per this specification, their use often becomes mandatory. Since this policy is within the control of the operator, the use of SCNs is not mandatory in this specification, but rather highly recommended.

Note l: this element is REQUIRED in the IPDR record, but if no SCN is used to identify the service flow in the BS, then this element MUST have a null value (that is <serviceClassName></serviceClassName> or <serviceClassName/>).

15.The SFdirection element identifies the service flow direction relative to the BS RFI interface. This element is REQUIRED and MUST have one of two values:

- "Upstream" identifies service flows passing packets from the SS to the BS.
- "Downstream" identifies service flows passing packets from the BS to the SS.

16.The octetsPassed element MUST contain the current 64-bit count of the number of octets passed by this service flow formatted in decimal notation. This element is REQUIRED. If the SFtype is Interim, then this is the current value of the running counter. If the SFtype is Stop, then this is the final value of the terminated counter. The 64-bit counter value will not wrap around within the service lifetime of the BS.

17.The pktsPassed element MUST contain the current 64-bit count of the number of packets passed by this service flow formatted in decimal notation. This element is REQUIRED. If the SFtype is Interim, then this is the current value of the running counter. If the SFtype is Stop, then this is the final value of the terminated counter. The 64-bit counter value will not wrap around within the service lifetime of the BS

18.The SLAdropPkts element contains the current 64-bit count of the number of packets dropped by this service flow due to enforcement of the maximum throughput limit specified by the Service Level Agreement (SLA) as implemented by the QoS parameter set. This element should be REQUIRED for all service flows. For upstream service flows, the counter record only the SLA enforcement performed by the BS. Upstream packets dropped or delayed at the SS are not recorded here. The counter is formatted in decimal notation. If the SFtype is Interim, then this is the current value of the running counter. If the SFtype is Stop, then this is the final value of the terminated counter. The 64-bit counter value will not wrap around within the service lifetime of the BS.

Note that these values are provided to aid the operator in identifying subscribers who are attempting to use more bandwidth than their SLA provides. This may be an opportunity to offer the subscriber a higher capacity SLA consistent with his/her demonstrated needs.

19. IPDRDoc.End element MUST be the last object inside IPDRDoc that describes the IPDR billing file itself. It defines the count of IPDRs that are contained in the file and the ending timestamp for the file creation.

- Count="nnnn". Where nnnn MUST be the decimal count of the number of IPDR records in this IPDRDoc.
- EndTime ="yyyy-mm-ddThh: mm: ssZ". MUST be the UTC time stamp at the time the billing file is completed (formatted as above). For example: endTime=" 2002-06-12T21: 11:23Z".

3.3.4 Billing Collection Interval

Subscriber Usage Billing Records report the absolute traffic counter values for each Service Flow by a SS (Subscriber) that has become active during the billing collection interval as seen at the end of the interval. The collection interval is defined as the time between the creation of the previous billing file (Tprev) and the

creation of the current billing file (Tnow). See Figure below. There are two kinds of Service Flows that are reported in the current billing file:

- SFs that are still running at the time the billing file are created.
- Terminated SFs that have been deleted and logged during the collection interval.

A provisioned or admitted state SF that was deleted before it became active MUST NOT be recorded in the billing file, even though the BS logged it.
The BS (or supporting EMS) MUST record any currently running SFs using Tnow as the timestamp for its counters and MUST identify them in the IPDR SFtype element as "Interim".
Terminated SFs that have a deletion time (Tdel) later than Tprev are the only ones recorded in the current billing file (i.e. a terminated SF MUST BE reported exactly once).
A BS MUST record a terminated SF using its Tdel from the log as the timestamp for its counters and MUST identify it in the IPDR SFtype element as "Stop".

Note that the timestamps are based on the formatter's recording times, not the collection system's retrieval times. Since the collection cycle may vary over time, the recording times in the billing file can be used to construct an accurate time base over sequences of billing files.

In the example shown in Figure below there are four Service Flows recorded for a Subscriber in the current billing file being created at Tnow.

- SFa is a long running SF that was running during the previous collection interval (it has the same SFID in both the current and the previous billing files). SFa was recorded as type Interim at Tprev in the previous billing file and is recorded again as type Interim at Tnow in the current file.
- SFb is a running SF that was created during the current collection interval. SFb is recorded as type Interim for the first time at Tnow in the current file.
- SFc is a terminated SF that was running during the previous collection interval but was deleted and logged during the current collection interval. SFc was recorded as type Interim at Tprev in the previous billing file and is recorded as type Stop at the logged Tdel(c) in the current file.
- SFd is a terminated SF that was both created and deleted during the current collection interval. SFd is recorded only once as type Stop at the logged Tdel (d) in the current billing file only.

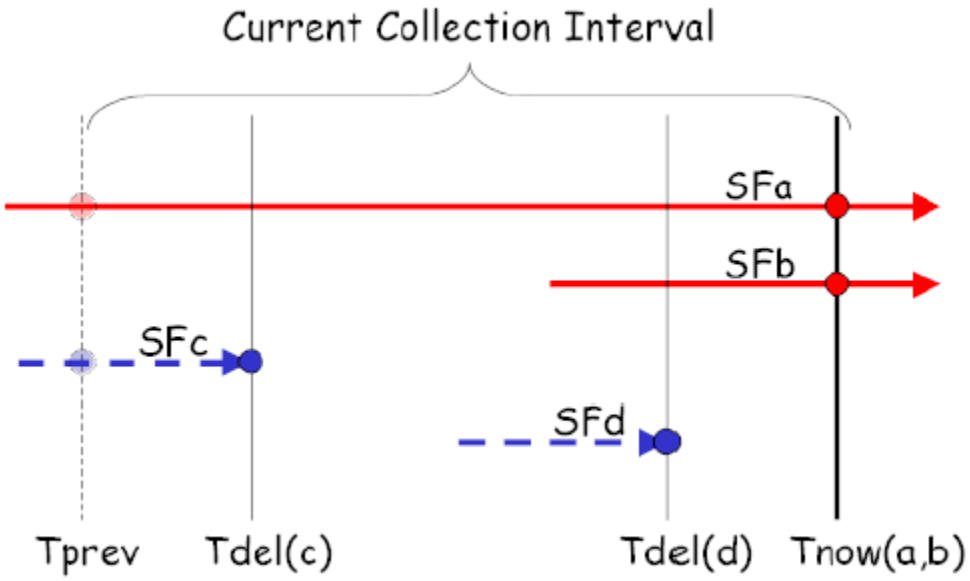

Figure 5. Billing Collection Interval Example

3.3.5 Billing File Retrieval Model

Billing files are built by the record formatter on the BS (or supporting EMS) and are then retrieved by the collection system in a decoupled manner using FTP semantics. There is no explicit signaling protocol between them and no prior arrangement regarding the frequency of billing collection.

- The BS (or supporting EMS) is responsible for creating the current billing file and must place it into its FTP file system only when the file is completely built. The formatter only creates one billing file that it has to be protected until the collection system is done with it.
- The collection system may retrieve the current billing file via FTP GET at any time after the file becomes available in the formatter's FTP file system.
- When the collection system has successfully retrieved the billing file, it must remove the file via FTP DELETE from the formatter's FTP file system.
- The formatter must monitor the existence of the billing file in its FTP file system and when it no longer exists, the formatter must begin to create the next billing file.
- The formatter MUST finish constructing the next billing file and have it ready for retrieval in its FTP file system within 15 minutes of the previous file's deletion.
- If the billing file does not yet exist in the formatter's FTP file system when the collection system comes to retrieve it, the collection system must back off and return later to try again. The specific timeout for collection system retries is implementation dependent, however, the collection system must not make more than 3 retrieval attempts within any 5-minute period.

Note that if the collection system fails for any reason, the formatter will retain and protect the last billing file created until the collection system returns to retrieve the file.

In this case, even though the recording timestamps in the current billing file may be quite old, the collection system will still retrieve the current file and delete it in the standard manner. The formatter will then immediately begin construction of a new billing file based on the current values of the BS's internal absolute 64-bit counters and the current timestamp. The collection system may then return at any time after the minimum cycle time (i.e. 15 minutes) and retrieve the new billing file with the current timestamps. The BS will always preserve the absolute values of the counters while it is operating, only the collection interval will be extended due to the outage on the collection system. The billing system can use the recording timestamps in the two files to accurately reconstruct the time base of the counters.

Furthermore, the collection system may deliberately vary its collection cycles based on time of day or day of week. This decoupled billing file retrieval model works well for this case also.

The decoupled billing file retrieval model also supports multiple retrievals by multiple collection systems so long as the last collection system deletes the billing file when it is done with it. However, there is no requirement to support multiple simultaneous file transfers from the formatter. How the multiple collection systems coordinate this between themselves is beyond the scope of this specification.

3.3.6 Billing File Security Model (FUTURE OPTIONAL)

The billing file security model has two components: 1) secure authentication to control access to the billing file in the formatter's FTP file system and 2) secure file transfer to ensure the privacy and the integrity of the billing file while it is in transit. Both of these components are provided by the Secure Shell protocol version 2 (SSH2) and its Secure FTP (SFTP) subsystem as described by Internet drafts maintained by the IETF's SECSH working group at:

www.ietf.org/html.charters/secsh-charter.html.

Additional information may be obtained from:

www.openssh.org,

Site provides an open source implementation of SSH2 and SFTP. A BS (or supporting EMS) hosting the billing formatter MUST provide secure access to its FTP file system via SSH2 and SFTP. It is also strongly recommended that the operator disable network access to the formatter's platform via legacy insecure Telnet and FTP when SSH2/SFTP are active.

The billing collector MUST have it's own user id and password for access to the billing file directory via SSH2/SFTP and this userid MUST NOT be shared with any other applications or users hosted on the formatter's platform. SSH2 user public key authentication is OPTIONAL for the billing collector's userid. How userids, keys, and passwords are administered on the formatter's platform is beyond the scope of this specification. Note also that the collection system requires both read and delete access permissions to the billing file directory in the formatter's FTP file system.

While the formatter's platform MUST provide secure authentication and file transfer capabilities, the operator may elect to not utilize them. In this case, the formatter's platform MUST provide access to the billing file

directory via legacy insecure FTP and the billing collector MUST have its own userid and password for legacy FTP access as well.

3.3.7 Minimal Requirements

WiMax compliant systems should comply with a set of rules in order to be able to support this method of Subscriber Account Management:

- **BS must implement IPDR recorder (IR) and IPDR store (IS) functionality**
- **BS must interact with BML/SML based on interface "C"**
- **BS must implement file of IPDR records as specified in 3.3.3.2**
- **BS must implement Billing Collection procedure as in 3.3.4**
- **BS must implement File retrieval model as in 3.3.5**
- **BS may implement File security model as in 3.3.6**
- **BS must implement MIB objects to support IPDR elements**

Most of these components have open source code and standard interfaces.
The interface proposal is only a draft but it can be detailed and extended to 802.16 specifics.

3.3.8 MIB Status

Some of the MIB variables for accounting are **TBD**. Statistics and performance data on service flows are not defined yet in WMAN-IF-MIB.

3.4 Performance

At the PHY and MAC layer level, performance management should focus on monitoring the effectiveness of system functionality. The required tool is provided in the form of standard interface statistics [RFC 2863] as well as the following variables nominated bellow. RFC 2863 is targeted to a generic interface and can be used to any interface of the system.

3.4.1 MIB Status

MIB variables addressing **performance** in 802.16 are found within the following nodes
For Base station
under wmanIfBsChMeasurementTable

For Subscriber station
not defined

Common (BS and SS) Performance Objects:
under wmanIfCmnSsChMeasurementTable

3.5 Security

MIB variables addressing **security** in 802.16 are found within the following nodes
For Base station
under wmanIfBsPkmObjects

For Subscriber station
under wmanIfSsPkmObjects

Common (BS and SS) PKM Objects:
under wmanIfCmnPkmObjects

4 OSSI for Digital Certificate management process

<TBD>

5 User Service provisioning

802.16/WiMax systems can offer a wide range of services based on dynamic services and multiple service flows capability:

- Different SLA's
- Data/Voice/Video Service Bundle
- Multiple Service Flows/Users per SS with different QoS level

Let's screen some possible models.

5.1 Services

5.1.1 Data/Voice/Video Services (SLA's)

Any service can be described by a set of QoS parameters such:

- CIR → Minimum Reserved Traffic Rate
- PIR → Maximum Sustained Traffic Rate
- Maximum Latency (if the case)
- Tolerated Jitter (if the case)

The requirements for Quality of Service may include:

- A configuration and registration function for pre-configuring SS-based QoS Service Flows and traffic parameters.
- Utilization of QoS traffic parameters for downstream Service Flows.
- Classification of packets arriving from the upper layer service interface to a specific active Service Flow

- Grouping of Service Flow properties into named Service Classes, so upper layer entities and external applications (at both the SS and the BS) can request Service Flows with desired QoS parameters in a **globally consistent** way.

Any packet is associated to a particular Service Flow and QoS policy by a set of **CLASSIFIER RULES.**
There are two major aspects of Service Flow definition and provisioning:

- SLA QoS parameters
- Classifier set

5.1.2 Class of Service

While designing different class of service offerings, a service operator might consider the following framework:

- Class of Service by account type: business vs. residential accounts
- Class of Service by guaranteed service levels
- Class of Service by time of day and/or day of week
- "On Demand" Service by special order
- Global Service Class concept introduced in 802.16e draft.

A **Service Class Name (SCN)** is defined in the BS via provisioning. An SCN provides a handle to an associated QoS Parameter Set (QPS) template. Service Flows that are created using an SCN are considered to be "named" Service Flows.
The SCN identifies the service characteristics of a Service Flow to external systems such as a billing system or customer service system. For consistency in billing, operators should ensure that SCNs are unique within an area serviced by the same OSS that utilizes this interface.

A descriptive SCN might be something that indicate the nature and direction of the Service Flow to the external system:

- PrimaryUp
- GoldUp
- VoiceDn
- BronzeDn

A good alternative to this known model is the definition of global service flows as in [11].
Global Service Class Name—A rules based, composite name parsed in six, one-byte parts of format ISBRLS, elements reference extensible look-up tables. Byte placeholders must be expressed values; may not be omitted.
The Global Service Class Names are not ‘descriptive’ but very useful and not last unique in 802.16 domains.

A Service Package implements a Service Level Agreement (SLA) between the Service Provider and its Subscribers. A Service Package might be known by a name such as Gold, Silver, or Bronze.
Service Package is itself implemented by the set of named Service Flows (using SCNs) that are provisioned to the BS/SS. Note that many Subscribers are assigned to the same Service Package.
The following is **a plausible sample of service classes** on downstream part of a service package:

- "Best Effort" Service Without Minimum Guarantee. This class of "Best Effort Only" service is the normal practice of today where subscribers of this class of service are allocated only excess channel bandwidth available at the time while each subscriber's access is capped at a maximum bandwidth (for example at 256 kilobit per second).
- Platinum Service for Business and High-Access Residential Accounts. Business accounts subscribing to this service are guaranteed a minimum data rate of downstream bandwidth 512 kilobit per second - and if excess bandwidth is available, they are allowed to burst to 2 megabit per second.
- Gold Service for Business Accounts. This class of service guarantees subscribers a 256-kilobit per second downstream data rate during business hours (for example from 8 a.m. to 6 p.m.) and 128 kilobit per second at other times. If excess bandwidth is available at any time, data is allowed to burst to 1 megabit per second.
- Gold Service for Residential Accounts. Residential subscribers of this service are guaranteed 128 kilobit per second downstream bandwidth during business hours and 256 kilobit per second at other times (for example from 6 p.m. to 8 a.m.), and a maximum data burst rate of 1 megabit per second with available excess bandwidth.
- Silver Service for Business Accounts. Business accounts subscribing to this service are guaranteed 128 kilobit per second downstream data rate during business hours and 64 kilobit per second during other times, and a maximum burst rate of 512 kilobit per second.
- Silver Service for Residential Accounts. Subscribers are guaranteed 64 kilobit per second downstream bandwidth during business hours and 128 kilobit per second at other times, with a maximum burst rate of 512 kilobit per second.
- "On Demand" Service by Special Order. This class of "on demand" service allows a subscriber to request additional bandwidth available for a specific period of time. For example, a subscriber can go to operator's web site and requests for increased guaranteed bandwidth service levels from his registered subscribed class of service from the normal 256 kilobit per second to 1 megabit per second from 2 p.m. to 4 p.m. the following day only, after which his service levels returns to the original subscribed class. The provisioning server will check the bandwidth commitment and utilization history to decide whether such "on demand" service is granted.
- Business accounts can require sometime more than one uplink and downlink service flow per SS. The same situation is encountered in MTU (multi-tenant unit) scenario where different accounts are open for service flows carried by one SS.
- TDM/VoIP terminals require multiple uplink and downlink service flows as well.

The same template specification can be done on upstream depending on the service requirement.

The proposal here would be to use:

- **Global Service Class Names as defined in [11].**
- **Use of descriptive Service Package names as defined above that encapsulates multiple global service classes.**
- **The operator can use any combination of global Service Class Names in order to define its set of Service Package.**

5.2 Service Flow Provisioning Model

Service Flow Provisioning is the process of provisioning service flow related parameters that allows service activation. A service flow is partially characterized by the following attributes:

- Identifier
- QoS Parameter Set
- Filtering Rules Set (Classification parameter set)
- Specific Parameters (ARQ….)

For more clarity , here are some definitions of:

- **Pre-provisioning** – user account and subscribed services are defined prior SS network entry and reside on the registration server. Pre-provisioned Service Flow information is sent to BS after SS is registered and authenticated. BS must inform NMS that a certain SS registered to it.
- **Self-provisioning** – user account and subscribed services are set after system installation. User gets limited access (walled garden) to the Provisioning Server during initialization process and set his own account. At the end of the process the network elements are provisioned and the user become registered and get access to the services.

None of the methods assumes a specific solution on classifier setting. If the classifiers cannot be provisioned in the same time with otherService Flow parameters a two-step service flow activation process is used.
Classifiers can be set by:

- Specific techniques used in network element domain (learning for instance).
- Higher network management layers through a management interface. How classifier is defined for a specific service flow is outside of the scope of the document.

5.2.1 SF Provisioning Basics

There is a set of rules to follow in order to be able to support provisioning and management of Service Flows:

- **Minimal requirement of a Service Flow identification is SFID and SCN**
- **SFID must be unique at network management level to meet the above requirement**
- **Any Service Flow (SFID, SCN) must be associated to an SS and one SS can provide multiple flows.**

- **Any user device (CPE/router/gateway) must be associated to a Service Flow (SFID, SCN) and one Service Flow (SFID, SCN) can support multiple devices.**
- **One CPE can be associated to multiple upstream and downstream Service Flows**
- **Classification rules assign packets to different Service Flows.**
- **Provisioning system must maintain a correct association of SS → Service Flows (SFID, SCN) → CPE (user device) and make possible classifier creation and setting.**

Here is a possible logical mapping:

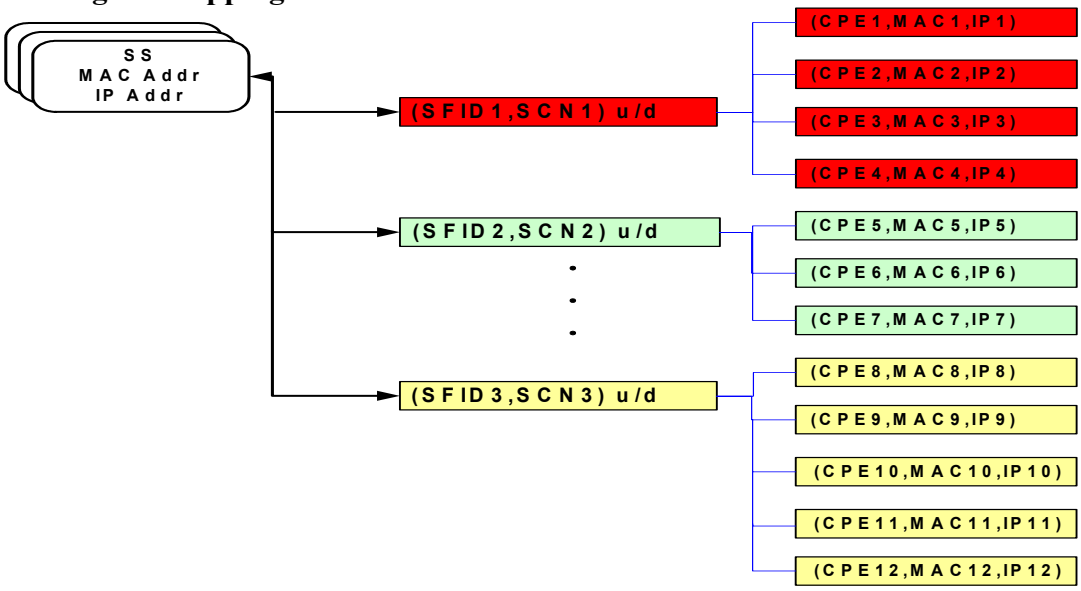

Figure 6.Service Flow Mapping Example

In the provisioning database the above mapping should exist for any account or user having multiple accounts.

5.2.2 Pre-provisioning Model

The right column abbreviations of the figure have the significance of:

- Session Initiator
- Ordering number

For instance: NMS3 means that:

- NMS is the initiator
- Third step of provisioning flow

Each step represents a session that performs a specific function.
All sessions can be composed as multiple messages coming back and forth. On both 5.2.2 and 5.2.3 all these sessions have to be detailed as far as BS or SS are concerned.

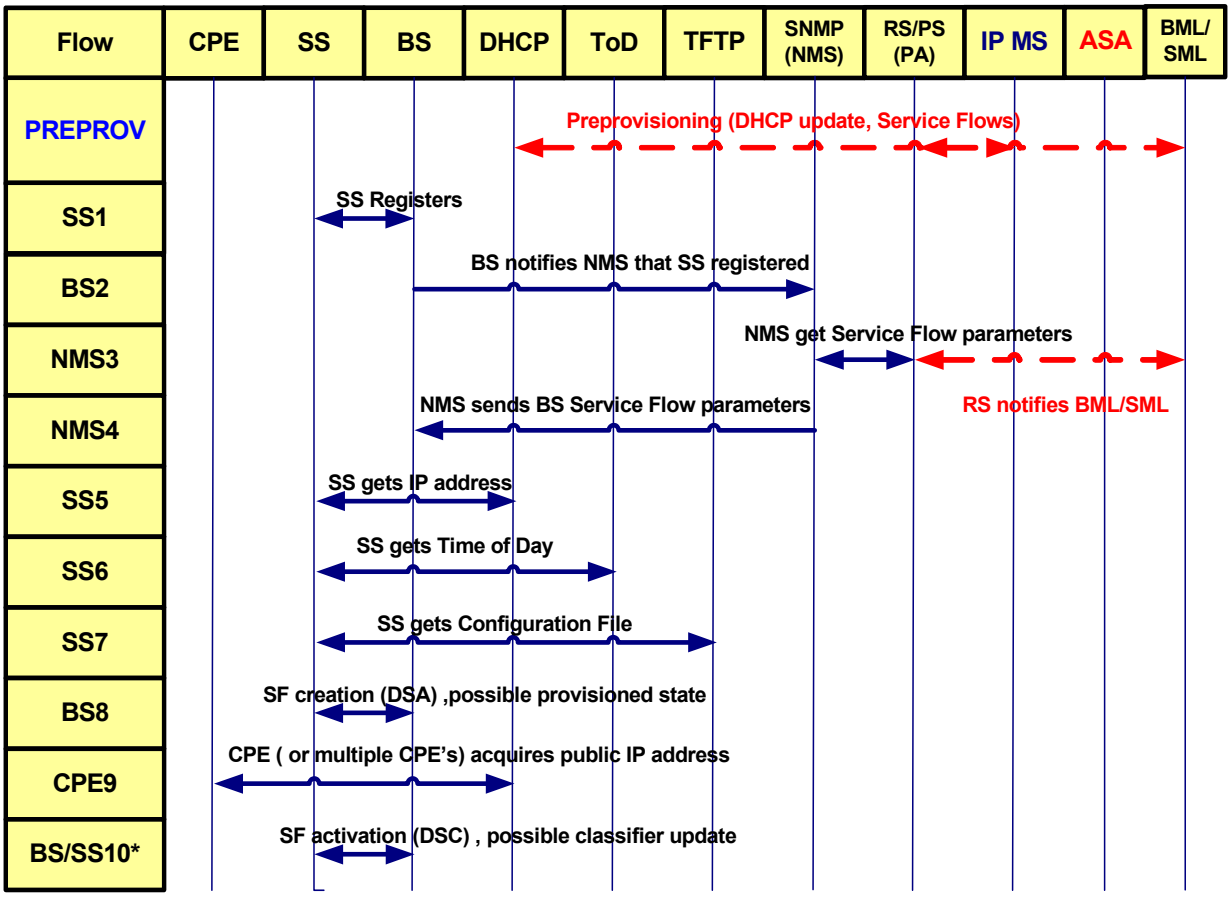

Figure 7 Service Provisioning Flow (pre-provisioned method)

The assumption on this case is that user account and service flow are provisioned into Service Provider infrastructure prior to SS connectivity.

Pre-provisioning assumes that client account and profile is built on BML/SML layer and based on that:

- Registration Server is updated
- Provisioning Application gets Service flow information
- Standard Internet Elements are configured (DHCP, TFTP, DNS)

At the time SS register, the entire pre-provisioning process is done at NML, BML/SML layers and Standard Internet Elements are configured.

During step 1 to step 4, SS is registered and authorized at network management level, and BS is provisioned with correct service flows.

This step cannot be done before hand as Registration Server must know to what BS, SS registered.

Authorization and network registration procedure can be expanded or customized depending on:

- BS features
- ISP infrastructure

The **BS/SS10** step is optional and happens in a two stages activation process when classifiers are not pre-provisioned for instance.
The basic interface requirement of service flow provisioning from NMS to Network Elements (BS/SS) is SNMP.
802.16 provisioning process is BS centric:

- BS8 assumes that service creation is initiated by BS
- The second step BS/SS10 (if the case) can be initiated be either one.

5.2.3 Self-Provisioning Model

Note: This model is based on a DHCP scenario. PPPoE can offer a self-provisioning method that is based on existence of a PPPoE client installed on the CPE.

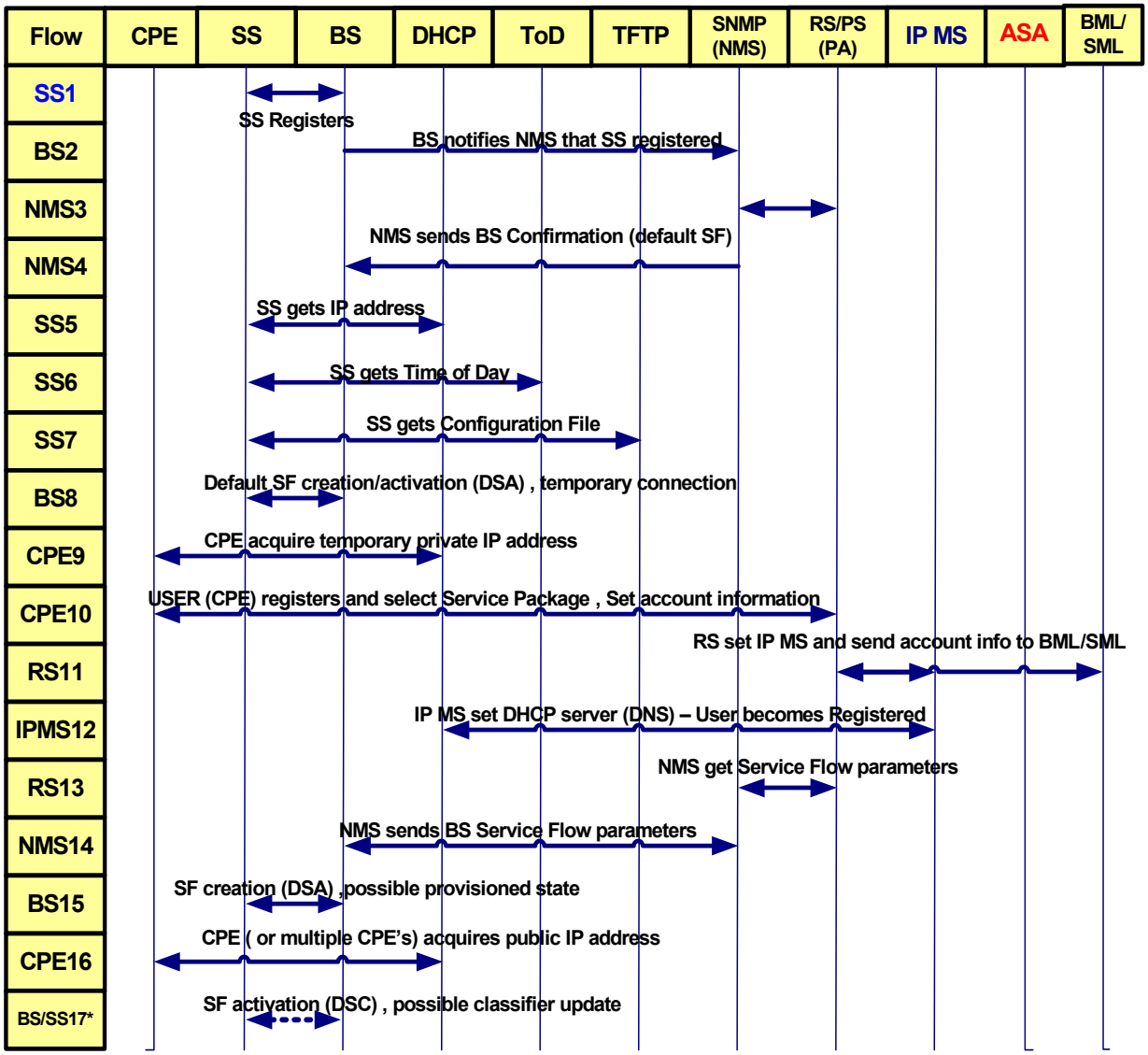

Figure 8.Service Provisioning Flow (self-provisioning method)

From Step 1 to Step 4:

- SS registers
- Network Management layer is informed that SS is associated to particular BS

When SS enters the network a default temporary service is activated in order to allow user registration. The default service flow has minimal QoS requirements and it is active for a limited time (X minutes).

When user CPE will access DHCP server, the server shall assign the user a temporary private/restricted IP address from unregistered address pool that gives user access only to ISP Registration Server (Walled Garden). This is Step 9.Step 9 can happen multiple times during service selection stage.
Once user accessed RS (step 10), is walked through service selection stage.
At the end of user registration and service selection stage, RS server updates higher layer with required account information and Internet standard elements with provisioning information.
As a result DHCP server will move new user into registered users pool (based on user identifier – client identifier, MAC address, host name…). When temporary restricted IP address will expire, user will receive a public IP address, and gain access to Internet.
Once **service selection** is done NMS (NMS14) shall provision service flow parameters to the BS.
The **BS/SS17** step is optional and happens in a two stages activation process when classifiers are not provisioned during BS15.
The basic interface requirement of service flow provisioning from NMS to Network Elements (BS/SS) is SNMP.

Note: Both models can be used for user provisioning in either DHCP or PPPoE environment.

5.3 Minimal Requirements

WiMax compliant systems should comply to a set of rules in order to be able to cope with any of these provisioning scenarios:

- **802.16 network elements should use SNMP for Service Flow provisioning purposes**
  - **At network element layer such changes in the MIB's must trigger DSx sessions as mentioned above**
- **802.16 network elements must implement a set of trap/inform messages in order to support SNMP based Service Flow provisioning.**
- **802.16 network elements must be able to inform NMS at least of SS ←→ BS association by means of SNMP (trap definition)**
- **802.16 network elements must support at least two-stage service activation process (provisioned to active).**
- **A recommended practice document must detail service flow changes (SNMP) and related DSx sessions.**
- **802.16 network elements should have learning capabilities at least at Layer 2 level.**

All these capabilities have to be defined. TBD.

5.4 MIB Status

MIB variables addressing **User Service Provisioning** are found within the following nodes
In Base station
under wmanIfBsProvisionedSfTable

under wmanIfBsServiceClassTable
under wmanIfBsClassifierRuleTable

In Subscriber station
Not needed

Common (BS and SS) Provisioning Objects:
under wmanIfCmnClassifierRuleTable
under wmanIfCmnCpsServiceFlowTable

6 Management of SS Ethernet Interface

The SS should have either an Ethernet 10/100 Base-T, or PCI/PCMCIA type bus for transparent bidirectional IP traffic forwarding in case of embedded wireless card.
This section describes the case of stand-alone SS that has an Ethernet Interface (EI).

6.1 SNMP Access via Ethernet Interface

SNMP access from the SS EI regardless activation stage, must comply with the access requirements specified by SNMP mode version (TBD)
The SS should support SNMP access through the following IP addresses:

- The SS DHCP-acquired IP can accept an SNMP request from EI only after completing IP connectivity.
- The SS should support 192.X.X.X as a well-known diagnostic IP address accessible only from the EI interfaces regardless of the SS registration state.

6.2 SS Diagnostic Capabilities

The SS may have a diagnostic interface for debugging and troubleshooting purposes. The interface must be limited by default to the requirements described (TBD), and should be disabled by default after registration has been completed. Additional controls may be provided that will enable the service operator to alter or customize the diagnostic interface, such as via the configuration process or later management by the MSO through the setting of a proprietary MIB.

6.3 Management Information Base (MIB) Requirements

All SS must implement the MIBs detailed in [x –802.16 reference] specification, and:

- SS must implement [RFC 2665], the Ethernet Interface MIB.

7 Appendix A DHCP versus PPPoE

Currently, many Service Providers are using PPPoE tunneling to provision services to the users. Both DHCP and PPPoE have advantages and drawbacks. This section does not advocate DHCP or PPPoE, but lists some advantages and drawbacks on each of them and depicts possible provisioning schemes.

7.1.1.1 PPPoE Advantages

- Largely deployed (DSL, Cable...), PPP is deployed for over a decade.
- The PPP architecture also incorporates the standard RADIUS protocols already deployed in a lot of service provider solutions.
- The PPPoE session concept can apply smooth to DSx services.

7.1.1.2 PPPoE Drawbacks

- Since it requires a network login/password, PPPoE is not "plug and play". This can be a real burden if you are using a laptop to connect to different networks at home, work, school, etc. To automate the connection you must store your password on your computer.
- Related to the above, DHCP makes it easy to plug a hub into a modem and share a connection to the ISP. With PPPoE you must either notify the ISP of each computer you intend to use (so they can bill you accordingly) or you must trade in your hub for a PPPoE-capable NAT device.
- PPPoE encapsulates TCP/IP. This reduces throughput by adding overhead to each packet.
- 802.16 does not have specified in the CS support for PPPoE encapsulation and classification.

7.1.1.3 DHCP Advantages

- An architecture based on DHCP offers increased flexibility and potential plug-and-play in certain applications. Process is transparent to the users.
- There are a lot of adds-on and improvements at IETF level in order to provide authentication and additional options that allows a better provisioning process.
- The so-called "Walled garden" is one example of an architecture. The broadband user is unknown to the network when first granted access through the DHCP server, so limited access is assigned while the authentication process is completed. In other words, the user is granted access to the network, but it is restricted to a specific area. The model is specified in the self-provisioning model detailed in 5.2.3.
- DHCP seems more suited to future extensions like mobility /nomadic applications.

- 802.16 use anyway DHCP on SS provisioning. According to 802.16 network entry before SS DHCP session, SS is already authorized.

7.1.1.4 DHCP Drawbacks

- Newer in the context of service provisioning. Multiple service flows raise additional issues.
- Walled garden is complex, requiring interfaces between the DHCP servers, a RADIUS server (for authentication), the Registration Server (IP Management System).
- DHCP walled garden also presents maintenance and administrative challenges because so many different applications must be tightly integrated in order to perform the provisioning procedure.
- The fact that user has granted some level of access before the authentication procedure can begin—clearly a potential security problem. The new IETF effort can potentially solve the issues and according to 802.16 the SS that bridge the provisioning is already authorized into the Network.

8 Annex B Alternative Management Protocols

8.1 COPS-PR

Common Open Policy Service protocol for support of policy provisioning (COPS-PR), see [12]. COPS-PR was designed to provision complex and continuously changing device configurations generated from policy based management systems.

- Used as standard interface in DOCSIS/PacketCable
- Used as standard interface in 3GPP as QoS policy control interface (Go Interface PDF←→ GPRS GSN)
- Both standard and proprietary data from PIBs
- TCP/IP based. Improvement over SNMP's use of UDP/IP
- Simple query/request and response/decision protocol
- Exchanges "policy" information between a Policy Decision Point (PDP) and its clients (Policy Enforcement Points, PEPs)
- Not widely implemented

8.2 SOAP/XML over HTTP (or BEEP)

Simple Object Access Protocol [13] is a member of the family of XML-associated protocols:

- Used in some OSS systems for Wi-Fi
- Standard interface in CableHome/home LAN realm
- Defined by the W3C, part of Web Services Web services consist of several building blocks built on top of XML (used as information format)
- Specification for RPC-like interactions and message communications using XML and HTTP

- Three main parts:
  - Message format that uses an envelope, header and body metaphor to wrap XML data
  - Restricted definition of XML data for making strict RPC-like calls, without using a predefined XML schema
  - Binding for SOAP messages to HTTP and extended HTTP
- Most people currently envision using SOAP over HTTP/1.1.
- May also run over a non-HTTP protocol (BEEP for instance)

8.3 NETCONF

- IETF defined XML based Network Management with focus on configuration
- Still in drafting stage based on XML [14]
- Addresses configuration aspects of network devices in vendor independent way
- Includes concepts of locking and transactions
- Existent Issues:
  - Unable to select single mandatory transport /"session" choice.
    - SSH - operators like it
    - SOAP/HTTP - developers like it
    - BEEP - protocol designers (router vendors) like it
  - No standard data, just commands and limited in functionality
- Extent of alignment to Web Services protocols SOAP/WSDL
- Transport layer must offer security.

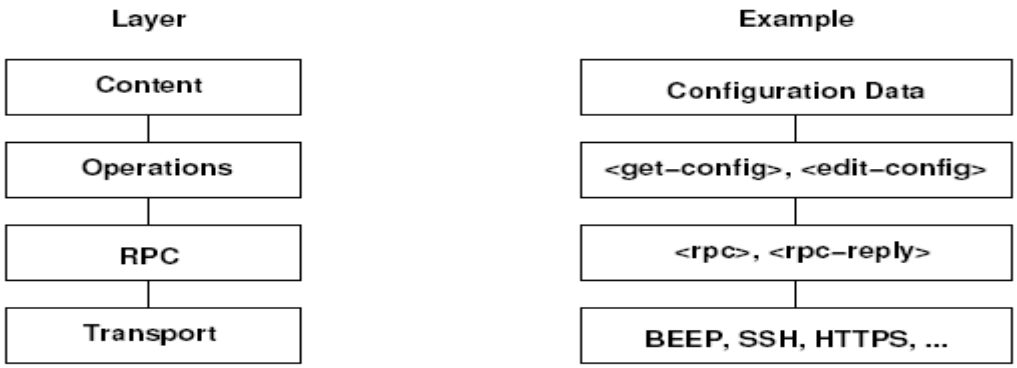

Figure 9 NetConf Layering Model

9 Glossary of Terms

Base station (BS): generalized equipment set providing connectivity, management, and control of the Subscriber Station (SS).

Customer (End User): A human being, organization, or telecommunications system that accesses the network in order to communicate via the services provided by the network.

Customer Premises Equipment (CPE): Equipment at the end user's premises; MAY be provided by the end user or the service provider.

DHCP: Dynamic Host Configuration Protocol - An Internet protocol used for assigning network-layer (IP) addresses.

Dynamic service: The set of messages and protocols that allow the base station (BS) and subscriber station
(SS) to add, modify, or delete the characteristics of a service flow.

Ethernet Interface (EI): the 'wired' network side interface of the Subscriber Station.

IPDR : Internet Protocol Detail Record –used for the representation and encapsulation of Internet Protocol (IP)-based events for use by business, operations and decision support systems.

MIB: Management Information Base

Network Management: The functions related to the management of data link layer and physical layer resources and their stations across the data network.

OSSI: Operations Support System Interface

Request For Comments (RFC): A technical policy document of the IETF; these documents can be accessed on the World Wide Web at http://www.rfc-editor.org/.

SCN: Service Class Name

Service flow (SF): A unidirectional flow of medium access control (MAC) service data units (SDUs) on a connection that is provided a particular Quality of Service (QoS).

Service flow identifier (SFID): A 32 bit quantity that uniquely identifies a service flow to both the subscriber station and base station (BS).
SLA: Service Level Agreement

Subscriber station (SS): A generalized equipment set providing connectivity between subscriber's equipment and a base station (BS).

Type/Length/Value (TLV): An encoding of three fields, in which the first field indicates the type of element, the second the length of the element, and the third field the value.

XML: Extensible Markup Language

10 References

[1] **IEEE P802.16-REVd/D5-2004,** Air Interface for Fixed Broadband Wireless Access Systems
[2] **SP-OSSIv2.0-I05-040407**, Data-Over-Cable Service Interface Specifications DOCSIS 2.0
[3] **Element Management Systems,** http://www.iec.org/
[4] **Network Data Management – Usage (NDM-U) For IP-Based Services, Version 3.1.1,** http://www.ipdr.org/
[5] **Service Specification – Public WLAN Access – IPDR,** http://www.ipdr.org/
[6] **RFC 1157 ,** Schoffstall, M., Fedor, M., Davin, J. and Case, J., A Simple Network Management Protocol (SNMP), IETF RFC 1157, May, 1990
[7] **RFC 1213 ,** K. McCloghrie and M. Rose. Management Information Base for Network Management of TCP/IP-base internets: MIB-II, IETF RFC 1213, March, 1991
[8] **RFC 1952** Deutsch, P., "GZIP file format specification version 4.3", RFC 1952, May, 1996.
[9] **RFC 2132** S. Alexander, R. Droms. DHCP Options and BOOTP Vendor Extensions. IETF RFC 2132. March, 1997.
[10] **RFC 3411** D. Harrington, R. Presuhn, B. Wijnen, An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks, December 2002.

[11] **IEEE P802.16e/D3, 31 May 2004**, Air Interface for Fixed and Mobile Broadband Wireless Access Systems — Amendment for Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands.
[12] **RFC 3084** - COPS Usage for Policy Provisioning (COPS-PR)
[13] **SOAP Version 1.2**, W3C Working Draft, World Wide Web Consortium (W3C), December 19, 2002, http://www.w3.org/2000/xp/Group/#drafts.
[14] http://www.ops.ietf.org/netconf/