

|                              |   |
|------------------------------|---|
| Project                      | <b>IEEE 802.16 Broadband Wireless Access Working Group</b> < <a href="http://ieee802.org/16">http://ieee802.org/16</a> >  |
| Title                        | <b>BS and SS Event Log for wmanIfMib</b>  |
| Date Submitted               | <b>2004-11-05</b>   |
| Source(s)                    | Joey Chou <a href="mailto:joey.chou@intel.com">[mailto:joey.chou@intel.com]</a><br>Intel Corporation<br>5000 W. Chandler Blvd.<br>Chandler, AZ 85226  |
| Re:                          |   |
| Abstract                     | Event logging provides a standard and centralized way to record important software and hardware events. It is instrumental to fault mitigation, system debugging, and the monitoring of the system operation, performance. This contribution proposed the BS and SS event log MIB to be included wmanIfMib in IEEE P802.16f/D1.   |
| Purpose                      | Adoption  |
| 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 | <p>The contributor is familiar with the IEEE 802.16 Patent Policy and Procedures (Version 1.0) &lt;<a href="http://ieee802.org/16/ipr/patents/policy.html">http://ieee802.org/16/ipr/patents/policy.html</a>&gt;, including the statement "IEEE standards may include the known use of patent(s), including patent applications, if there is technical justification in the opinion of the standards-developing committee and provided the IEEE receives assurance from the patent holder that it will license applicants under reasonable terms and conditions for the purpose of implementing the standard."</p> <p>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 &lt;<a href="mailto:r.b.marks@ieee.org">mailto:r.b.marks@ieee.org</a>&gt; as early as possible, in written or electronic form, of any patents (granted or under application) that may cover technology that is under consideration by or has been approved by IEEE 802.16. The Chair will disclose this notification via the IEEE 802.16 web site &lt;<a href="http://ieee802.org/16/ipr/patents/notices">http://ieee802.org/16/ipr/patents/notices</a>&gt;.</p> |

*Table of Content*

**1. Introduction ..... 3**

**2. Event Log Requirements..... 3**

**3. Event Log ASN.1 Definition..... 4**

1

## 2 1. Introduction

3 Event logging provides a standard and centralized way to record important software and  
4 hardware events. Event Log MIB records the transient information associated with an  
5 event against the possibility that the Notification message can be lost. It is instrumental to  
6 fault mitigation, system debugging, and the monitoring of the system operation,  
7 performance. This contribution proposed the BS and SS event log MIB to be included in  
8 wmanIfMib in IEEE P802.16f/D1.

## 9 2. Event Log Requirements

10 The Event Log consists of the following requirements:

- 11
- 12     ▪ Event log uses the wrap-around buffers to store events. When the buffer is full,  
13     the oldest entry will be removed to make room for the new entry.
- 14     ▪ The sizes of the buffers are configurable.
- 15     ▪ Events in the log have a lifespan that is also configurable..
- 16     ▪ The content of each entry will be retained after the power reset.
- 17     ▪ Certain events can trigger notifications that will be sent to NMS.
- 18     ▪ It is flexible to add vendor specific events or WiMAX certification events

19

20 Each entry consists of the following objects:

- 21
- 22     ▪ wmanIfSsEventIdentifier – the event ID.
- 23     ▪ wmanIfSsEventLoggedTime – the time when the event occurred.
- 24     ▪ wmanIfSsEventDescription – a string description of the event.
- 25     ▪ wmanIfSsEventSeverity – the severity of the event.
- 26         • **Emergency** – Reserved for vendor-specific ‘fatal’ hardware or software  
27         errors that prevents normal system operation and causes reporting  
28         system to reboot. Vendors may define their own set of emergency events.
- 29         • **Alert** – A serious failure, which causes reporting system to reboot but it is  
30         not caused by hardware or software malfunctioning. After recovering from  
31         the critical event, the system **MUST** send a cold/warm start notification.  
32         The alert event could not be reported as a Trap or SYSLOG message  
33         and **MUST** be stored in the internal log file. The code of this event **MUST**  
34         be saved in non-volatile memory and reported later.
- 35         • **Critical** – A serious failure that requires attention and prevents the device  
36         from transmitting data but could be recovered without rebooting the  
37         system. After recovering from the error event SS **MUST** send the Link Up  
38         notification. Critical events could not be reported as a Trap or SYSLOG  
39         message and **MUST** be stored in the internal log file. The code of this  
40         event **MUST** be reported later.

- 1           • **Error** – A failure occurred that could interrupt the normal data flow but will
- 2           not cause the SS to re-register. Error events could be reported in real time
- 3           by using the trap or SYSLOG mechanism.
- 4           • **Warning** – A failure occurred that could interrupt the normal data flow but
- 5           will not cause the SS to re-register. ‘Warning’ level is assigned to events
- 6           both SS and BS have information about. To prevent sending the same
- 7           event both from the SS and the BS, the trap and Syslog reporting
- 8           mechanism is disabled by default for this level.
- 9           • **Notice** – The event is important, but is not a failure and could be reported
- 10          in real time by using the trap or SYSLOG mechanism.
- 11          • **Informational** – The event is of marginal importance, and is not failure,
- 12          but could be helpful for tracing the normal modem operation.
- 13          • **Debug** – Reserved for vendor-specific non-critical events.
- 14          ▪ wmanIfSsEventNotification – a Boolean value determines if a trap should be
- 15          reported.
- 16          ▪ wmanIfSsEventNotificationOid – the object identifier of the event.

### 17 **3. Event Log ASN.1 Definition**

```

18
19 WmanIfEventSeverity ::= TEXTUAL-CONVENTION
20     STATUS      current
21     DESCRIPTION
22         "WmanIfEventSeverity defines the alarm Severity of an
23         event."
24     SYNTAX      INTEGER {emergency(1),
25                    alert(2),
26                    critical(3),
27                    error(4),
28                    warning(5),
29                    notice(6),
30                    informational(7),
31                    debug(8) }
32
33 WmanIfBsEventId ::= TEXTUAL-CONVENTION
34     STATUS      current
35     DESCRIPTION
36         "WmanIfBsEventId defines the identifier of BS events."
37     SYNTAX      INTEGER {
38 --         Event Id                               Severity Trap
39 -- Synchronization, and UL/DL acquisition
40
41 -- Initial ranging and periodic ranging
42         initRangRecv(0100), -- Initial RNG-REQ w MAC address
43         -- received                               notice    no
44
45 -- SBC
46
47
```

```

1  -- Authorization, authentication
2
3
4  -- Registration
5
6
7  -- Dynamic Services Add, Change, and Delete
8      dsxFailed(0500),      -- DSx operation failed  critical  yes
9                          -- wmanBsSsDynamicServiceFailTrap
10
11 -- Environment
12     powerStatChange(0600), -- Primary / secondary power ON or OFF
13                             --                               critical  yes
14                             -- wmanBsPowerStatusChangeTrap
15     fanStstChange(0601),   -- Fan ON or OFF status change
16                             --                               critical  yes
17                             -- wmanBsFanStatusTrap
18     tempatureChange(0602), -- Temperature change across
19                             -- high / low threshold  critical  yes
20                             -- wmanBsTemperatureChangeTrap
21
22 -- Radio
23     rssiStatChange(0700)   -- RSSI change across high / low
24                             -- threshold              critical  yes
25                             -- wmanBsSsRssiStatusChangeTrap
26 -- Vendor specific events
27
28 -- WiMAX certification events
29     }
30
31 WmanIfSsEventId ::= TEXTUAL-CONVENTION
32     STATUS      current
33     DESCRIPTION
34         "WmanIfSsEventId defines the identifier of SS events."
35     SYNTAX      INTEGER {
36 --         Event Id                                     Severity Trap
37 -- Synchronization, and UL/DL acquisition
38     lostDlMap(0001),      -- No DL-MAP received within 'Lost
39                             -- DL-MAP Interval',      critical, no
40     lostUlMap(0002),      -- No UL-MAP received within 'Lost
41                             -- UL-MAP Interval',      critical, no
42     lostUcd(0003),        -- No UCD received within
43                             -- 'UCD Interval'        critical, no
44     invlaidUcd(0004),     -- Invalid UCD received,
45                             --                               error,    no
46     ucdTimeout(0005),    -- No DCD received within T12,
47                             --                               critical  no
48     lostDcd(0006)        -- No DCD received within
49                             -- 'DCD Interval'        critical, no
50     invalidDcd(0007),    -- Invalid DCD received,
51                             --                               error,    no
52     dcdTimeout(0008),    -- No DCD received within T1,
53                             --                               critical  no
54     syncCompleted(0009), -- Sync completed          notice   no

```

```

1
2 -- Initial ranging and periodic ranging
3     rcvBcRangOpp(0100), -- Received broadcast ranging
4     -- opportunity          notice    no
5     rcvUcRangOpp(0101), -- Received unicast ranging
6     -- opportunity          notice    no
7     bcRangTimeout(0102), -- Broadcast ranging timeout (T2)
8     --                      warning    no
9     rangRspTimeout(0103), -- RNG-RSP timeout (T3)
10    --                      warning    no
11    contRangRetry(0104), -- Exceed the number of 'Contention
12    -- Ranging Retry          critical no
13    invRangRetry(0105), -- Exceed the number of 'Invited
14    -- Ranging Retry'         critical no
15    noUcRangOpp(0106), -- No unicast ranging opportunity
16    -- within T4              critical no
17    initRangAbort(0107), -- Initial ranging aborted
18    --                      error      no
19    initRangSucc(0108), -- Initial ranging succeed
20    --                      notice     no
21    periodRangSucc(0109), -- Periodic ranging succeed
22    --                      notice     no
23    periodRangFail(0110), -- Periodic ranging failed
24    --                      critical   no
25 -- SBC
26     sbcRspTimeout(0200), -- SBC-RSP not received within T18
27     --                      critical   no
28     sbcReqRetry(0201), -- Exceed the number of 'SBC Request
29     -- Retries'              critical no
30     sbcRspFail(0202), -- SBC-RSP failed          critical no
31     sbcRspSucc(0203), -- SBC-RSP succeed        notice   no
32
33 -- Authorization, authentication
34     authReqTimeout(0300), -- Auth request timeout  warning  no
35     reauthTimeout(0301), -- Reauth REQ timeout    warning  no
36
37 -- Registration
38     regRspTimeout(0400), -- REG-RSP not received within T6
39     --                      critical   no
40     regReqRetry(0401), -- Exceed the number of 'Registration
41     -- Request Retries'     critical no
42     regCompleted(0402), -- Registration completed
43     --                      notice     yes
44     -- wmanBsSsRegisterTrap
45     regUnmanagedSs(0403), -- Unmanaged SS          notice   no
46     regFailed(0404), -- Registration failed    critical no
47     regInvalidHmac(0405), -- Invalid HMAC in REG    critical no
48     regIpUnsupport(0406), -- IP version in REG not supported
49     --                      error      no
50
51 -- Dynamic Services Add, Change, and Delete
52     ssDsxFail(0500), -- DSx operation failed  critical yes
53     -- wmanSsDynamicServiceFailTrap
54

```

```

1  -- Radio
2      ssRssiChange(0700)    -- RSSI change across high / low
3                          -- threshold                critical  yes
4                          -- wmanSsRssiStatusChangeTrap
5  -- Vendor specific events
6
7  -- WiMAX certification events
8      }
9
10
11 -- Base station Event Log Group
12 --
13 wmanIfBsEventLog OBJECT IDENTIFIER ::= { wmanIfBsObjects 5 }
14
15 --
16 -- BS Event log configuration
17 --
18 wmanIfBsEventLogEntryLimit OBJECT-TYPE
19     SYNTAX      INTEGER
20     MAX-ACCESS  read-write
21     STATUS      current
22     DESCRIPTION
23         "The maximum number of event entries that may be held
24         in wmanIfBsEventLogTable. If an application changes
25         the limit while there are events in the log, the
26         oldest events must be discarded to bring the log down
27         to the new limit."
28     DEFVAL      { 200 }
29     ::= { wmanIfBsEventLog 1 }
30
31 wmanIfBsEventLifeTimeLimit OBJECT-TYPE
32     SYNTAX      INTEGER
33     UNITS       "minutes"
34     MAX-ACCESS  read-write
35     STATUS      current
36     DESCRIPTION
37         "The number of minutes an event should be kept in the log
38         before it is automatically removed. If an application
39         changes the value of wmanIfBsEventLifeTimeLimit, events
40         that are older than the new time may be discarded to meet
41         the new lifetime. A value of 0 means lifetime limit."
42     DEFVAL      { 1440 }
43     ::= { wmanIfBsEventLog 2 }
44
45 wmanIfBsEventLogTable OBJECT-TYPE
46     SYNTAX      SEQUENCE OF WmanIfBsEventLogEntry
47     MAX-ACCESS  not-accessible
48     STATUS      current
49     DESCRIPTION
50         "This is the Syslog table that is used to store Bs local
51         events. This table should reside in the non-volatile
52         memory that should persist after power cycle or reboot.
53         The number of entries in this table is determined by
54         wmanIfBsEventLogEntryLimit. It is a wrap around buffer.

```

```

1           When the buffer is full, the oldest entry will be removed
2           to make room for the newest entry."
3       ::= { wmanIfBsEventLog 3 }
4
5 wmanIfBsEventLogEntry OBJECT-TYPE
6     SYNTAX      WmanIfBsEventLogEntry
7     MAX-ACCESS  not-accessible
8     STATUS      current
9     DESCRIPTION
10            "Entries appear in this table when events occur, and are
11            removed to make ways for new entries when buffer is full,
12            the entry passes the lifetime limit. This table is
13            indexed by wmanIfBsEventIndex."
14     INDEX       { wmanIfBsEventIndex }
15     ::= { wmanIfBsEventLogTable 1 }
16
17 WmanIfBsEventLogEntry ::= SEQUENCE {
18     wmanIfBsEventIndex      Unsigned32,
19     wmanIfBsEventIdentifier  WmanIfBsEventId,
20     wmanIfBsEventLoggedTime  TimeStamp,
21     wmanIfBsEventDescription SnmpAdminString,
22     wmanIfBsEventSeverity    WmanIfEventSeverity,
23     wmanIfBsEventNotification TruthValue,
24     wmanIfBsEventNotificationOid OBJECT IDENTIFIER}
25
26 wmanIfBsEventIndex OBJECT-TYPE
27     SYNTAX      Unsigned32 (1..4294967295)
28     MAX-ACCESS  not-accessible
29     STATUS      current
30     DESCRIPTION
31            "A monotonically increasing integer for the sole purpose
32            of indexing entries within the event log. When it
33            reaches the maximum value, the agent wraps the value
34            back to 1."
35     ::= { wmanIfBsEventLogEntry 1 }
36
37 wmanIfBsEventIdentifier OBJECT-TYPE
38     SYNTAX      WmanIfBsEventId
39     MAX-ACCESS  read-only
40     STATUS      current
41     DESCRIPTION
42            "The identifier of a BS event."
43     ::= { wmanIfBsEventLogEntry 2 }
44
45 wmanIfBsEventLoggedTime OBJECT-TYPE
46     SYNTAX      TimeStamp
47     MAX-ACCESS  read-only
48     STATUS      current
49     DESCRIPTION
50            "The value of sysUpTime when the entry was placed in the
51            log. If the entry occurred before the most recent
52            management system initialization this object value must
53            be set to zero."
54     ::= { wmanIfBsEventLogEntry 3 }

```

```

1
2 wmanIfBsEventDescription OBJECT-TYPE
3     SYNTAX      SnmpAdminString
4     MAX-ACCESS  read-only
5     STATUS      current
6     DESCRIPTION
7         "This object describes the event."
8     ::= { wmanIfBsEventLogEntry 4 }
9
10 wmanIfBsEventSeverity OBJECT-TYPE
11     SYNTAX      WmanIfEventSeverity
12     MAX-ACCESS  read-only
13     STATUS      current
14     DESCRIPTION
15         "This object describes the severity of such event."
16     ::= { wmanIfBsEventLogEntry 5 }
17
18 wmanIfBsEventNotification OBJECT-TYPE
19     SYNTAX      TruthValue
20     MAX-ACCESS  read-only
21     STATUS      current
22     DESCRIPTION
23         "An event notification will be reported when it is
24         True (1)."

```

```

1
2 wmanIfSsEventLifeTimeLimit OBJECT-TYPE
3     SYNTAX      INTEGER
4     UNITS       "minutes"
5     MAX-ACCESS  read-write
6     STATUS      current
7     DESCRIPTION
8         "The number of minutes an event should be kept in the log
9         before it is automatically removed. If an application
10        changes the value of wmanIfSsEventLifeTimeLimit, events
11        that are older than the new time may be discarded to meet
12        the new lifetime. A value of 0 means lifetime limit."
13     DEFVAL      { 1440 }
14     ::= { wmanIfSsEventLog 2 }
15
16 wmanIfSsEventLogTable OBJECT-TYPE
17     SYNTAX      SEQUENCE OF WmanIfSsEventLogEntry
18     MAX-ACCESS  not-accessible
19     STATUS      current
20     DESCRIPTION
21         "This is the Syslog table that is used to store SS local
22         events. This table should reside in the non-volatile
23         memory that should persist after power cycle or reboot.
24         The number of entries in this table is determined by
25         wmanIfSsEventLogEntryLimit. It is a wrap around buffer.
26         When the buffer is full, the oldest entry will be removed
27         to make room for the newest entry."
28     ::= { wmanIfSsEventLog 3 }
29
30 wmanIfSsEventLogEntry OBJECT-TYPE
31     SYNTAX      WmanIfSsEventLogEntry
32     MAX-ACCESS  not-accessible
33     STATUS      current
34     DESCRIPTION
35         "Entries appear in this table when events occur, and are
36         removed to make ways for new entries when buffer is full,
37         the entry passes the lifetime limit. This table is
38         indexed by wmanIfSsEventIndex."
39     INDEX       { wmanIfSsEventIndex }
40     ::= { wmanIfSsEventLogTable 1 }
41
42 WmanIfSsEventLogEntry ::= SEQUENCE {
43     wmanIfSsEventIndex          Unsigned32,
44     wmanIfSsEventIdentifier     WmanIfSsEventId,
45     wmanIfSsEventLoggedTime     TimeStamp,
46     wmanIfSsEventDescription    SnmpAdminString,
47     wmanIfSsEventSeverity       WmanIfEventSeverity,
48     wmanIfSsEventNotification   TruthValue,
49     wmanIfSsEventNotificationOid OBJECT IDENTIFIER}
50
51 wmanIfSsEventIndex OBJECT-TYPE
52     SYNTAX      Unsigned32 (1..4294967295)
53     MAX-ACCESS  not-accessible
54     STATUS      current

```

```
1      DESCRIPTION
2          "A monotonically increasing integer for the sole purpose
3          of indexing entries within the event log. When it
4          reaches the maximum value, the agent wraps the value
5          back to 1."
6      ::= { wmanIfSsEventLogEntry 1 }
7
8 wmanIfSsEventIdentifier OBJECT-TYPE
9     SYNTAX      WmanIfSsEventId
10    MAX-ACCESS  read-only
11    STATUS      current
12    DESCRIPTION
13        "The identifier of a SS event."
14    ::= { wmanIfSsEventLogEntry 2 }
15
16 wmanIfSsEventLoggedTime OBJECT-TYPE
17     SYNTAX      TimeStamp
18    MAX-ACCESS  read-only
19    STATUS      current
20    DESCRIPTION
21        "The value of sysUpTime when the entry was placed in the
22        log. If the entry occurred before the most recent
23        management system initialization this object value must
24        be set to zero."
25    ::= { wmanIfSsEventLogEntry 3 }
26
27 wmanIfSsEventDescription OBJECT-TYPE
28     SYNTAX      SnmpAdminString
29    MAX-ACCESS  read-only
30    STATUS      current
31    DESCRIPTION
32        "This object describes the event."
33    ::= { wmanIfSsEventLogEntry 4 }
34
35 wmanIfSsEventSeverity OBJECT-TYPE
36     SYNTAX      WmanIfEventSeverity
37    MAX-ACCESS  read-only
38    STATUS      current
39    DESCRIPTION
40        "This object describes the severity of such event."
41    ::= { wmanIfSsEventLogEntry 5 }
42
43 wmanIfSsEventNotification OBJECT-TYPE
44     SYNTAX      TruthValue
45    MAX-ACCESS  read-only
46    STATUS      current
47    DESCRIPTION
48        "An event notification will be reported when it is
49        True (1)."
```

```
50    ::= { wmanIfSsEventLogEntry 6 }
51
52 wmanIfSsEventNotificationOid OBJECT-TYPE
53     SYNTAX      OBJECT IDENTIFIER
54    MAX-ACCESS  read-only
```

```
1      STATUS      current
2      DESCRIPTION
3          "The NOTIFICATION-TYPE object identifier of the event
4          that occurred."
5      ::= { wmanIfSsEventLogEntry 7 }
6
7
```

