White Paper

Remote Management: SNMP Sub-agent for Intel® Server Hardware

Simple Network Management Protocol (SNMP) is used ubiquitously to monitor and control any device over the network by exchanging SNMP messages. The SNMP, defined by IETF [1], has gained significant popularity among the software, hardware, telecom, and network industries. SNMP was derived from its predecessor SGMP (Simple Gateway Management Protocol). This is considered as a de facto and standard client-server protocol to transfer the management information over the network. In 1988, the initial version of SNMP (version 1) used TCP/IP suite and supported GET, GET-NEXT, and SET methods to get management information from the managed node. The SNMPv2c was introduced with improvements in performance, security, and confidentiality with support of GET-BULK. The SNMPv3 was recognized by IETF [1] in 2004 with three new important features - message integrity, authentication, and encryption. SNMPv3 notification message can be sent as trap or inform. Traps are unreliable. On the other hand, the SNMP inform messages are reliable. The management application on the receiving-end sends the acknowledgement to the sender SNMP agent.

Remote management of devices in a network is a prime requirement for IT management. Management of Intel EPSD servers is a use-case for such requirement. SNMP, a network management protocol, is a widely accepted protocol for transferring management information over the network. The overall system health information and alerts related to sensors from EPSD servers are available by using SNMP. The purpose of this whitepaper is to describe different solutions and features of SNMP based remote management of Intel server hardware. These solutions are out-of-band SNMP agent in BMC and in-band Intel® Baseboard SNMP sub-agent.

1. Basics of SNMP

The SNMP defines how the communication between two network nodes happens, message type and format, and message interpretation in accordance with its standards. The SNMP management application communicates to the SNMP agent to get management information from the managed device. The SNMP message is wrapped in the UDP packet and that inturn is wrapped in the IP packet. The SNMP resides in the application layer. The layered communication model for SNMP message is shown in **Exhibit 1**. The SNMP manager issues GET, GETNEXT and SET messages to the agent. The GET and GET-NEXT messages are used by management application as requests to the agent. The agent issues GET-RESPONSE message to the manager with either the information requested or an error indication as to why the request could not be processed. A SET message allows management application to modify the value of a specific variable. Again agent would respond with the GET-RESPONSE message with an indication whether the variable is modified or not. The general format for all types of SNMP message is shown in **Exhibit 2**.

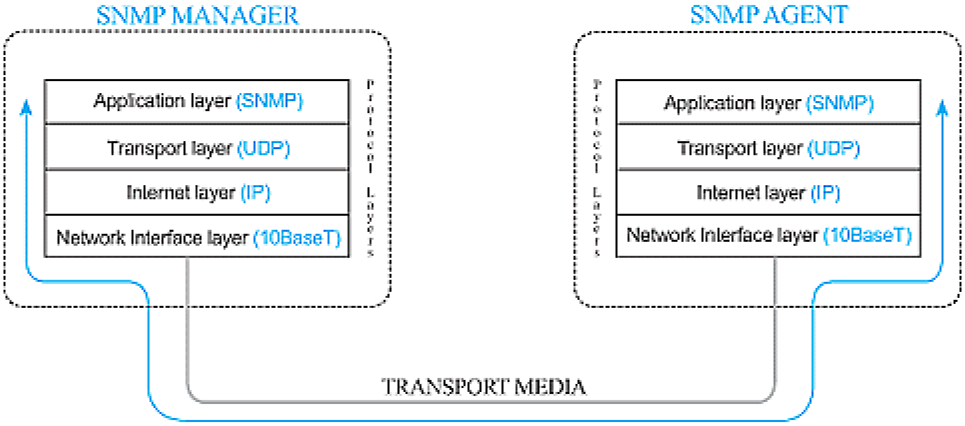


Exhibit 1: Layered Communication of SNMP Messages

SNMP manager and agent use management information base (MIB) along with a small set of commands to communicate with each other. On the managed device, there are few variables in which management application is interested. These variables are represented in the tree structure in the MIB. Each variable in the MIB is termed as OID (Object Identifier). The OID for every variable is unique within the MIB. The MIB is also a guide the capabilities of managed device or node.

2. SNMP Notification Messages

The SNMP management application may be responsible for managing a large number of network devices. The management application is always interested in the change of state of managed device. This is certainly not practical for management application to poll all the managed devices for a specific event. Therefore, the SNMP agent on each managed device notifies the management application without solicitation. For SNMP v1 and v2c, these event messages are SNMP trap of the event. For SNMP v3, trap and inform are two types of notification message for such event. The SNMP inform message is reliable. The receiver management application sends the acknowledgement. The default values for retries and timeout is 3 and 30 seconds, respectively. This is the only message that is generated by the agent to inform or alert the management application if a specific variable is behaving unexpectedly. After receiving the SNMP trap, the management application can choose to take an action. The SNMP traps and details of its format are defined in RFC 1157 **[2]**.

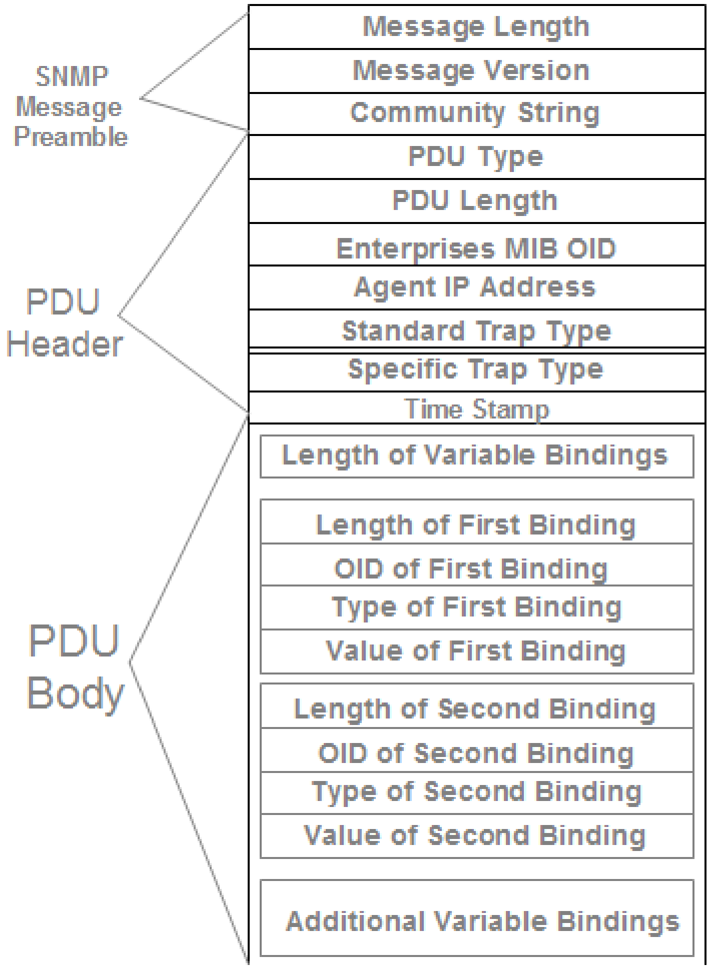


Exhibit 2: SNMP Message Format

3. Baseboard Management Controller

Intelligent Platform Management Interface (IPMI) defines a standardized, abstracted, message-based interface to server management hardware [4]. The key characteristics of IPMI for inventory, monitoring, logging and recovery control functions are available independent of the main processors, BIOS and operating system. These functions can also be made available when the system is in a powered down state; however system should be connected to AC power and network. This interface is exposed through standard management interfaces such as SNMP, CIM, WMI, etc. The heart of the IPMI architecture is a microcontroller that is Baseboard Management Controller (BMC). BMC contains a firmware that manages the interface between system management software and the platform management hardware. The key functions provided by this interface are SEL (System Event Log), SDR (Sensor Data Record) repository, PEF (Platform Event Filtering), etc. The complete management interface and functions of BMC are mentioned in the IPMI specification [6].

4. System Event Log

BMC provides a centralized, non-volatile SEL and adds event messages to it. These messages can be significant or critical management events raised by the management controllers. Event receiver writes message into the SEL. Each SEL entry has a unique ID field that can be used for retrieving it.

These can be deleted and complete repository may be cleared by the management applications. SEL event record follows a format [6]. The management applications parse this information, get the time stamp, data on event, and sensor, etc.

5. SDR Repository and Sensor Model

The SDR repository is a single non-volatile storage area that contains all the system SDRs. Its records contain information about the type and number of platform sensors, threshold support, event generation capabilities and type of readings provided by the sensor. SDR also include records describing the number and type of devices that are connected to the system.

The monitoring information from sensors such as temperature, voltages, fan status, etc is provided by the IPMI Sensor Model. This provides an abstraction between management software and hardware. Sensors are classified according to the type of reading they provide and/or the type of events they generate; sensor can return either an analog or discrete reading. Sensors can be also be classified into two types; linear and non-linear sensors. Linear sensors have constant accuracy, tolerance and resolution over their raw readings. Using a linear conversion formula, linear sensors readings can be converted into the desired sensor units such as temperature, voltage, etc. Non-linear sensors do not have constant conversion factors, accuracy, tolerance and/or resolution over the raw readings. This means that the polling of such sensor is two-step process. After getting the raw reading, the get sensor reading factors should be retrieved by the management software.

6. Out-of-Band SNMP Agent in BMC

The BMC has SNMP agent running with a limited capability to support the generation of SNMP traps. This has OOB support for management applications. If supported, PEF (Platform Event Filtering) provides a mechanism for configuring this agent to take selected actions on the event messages. The BMC maintains the PEF event filter and policy tables, those can be configured by management applications. For every event message (for example, a new entry into the SEL), the agent performs the event filtering task based on the event data to select which action should be triggered. After successful match of event from event filter table, the corresponding action is performed. These actions include operations such as system power off, system reset, as well as triggering the generation of IPMI Alerts. The actions are sorted and high priority action (for example, power off) is performed first. If action is sending an alert, the agent looks for Alert in the Alert policy table; this policy may have collection of one or more alert destinations. OOB agent sends the SEL event in the PET format, [7], as a SNMP trap, to all the alert destinations. The fields of SNMP massage are shown in Exhibit 2. The specific-trap and variable-bindings fields carry the heart of the PET information. The content and definition of these fields are specification is specified Exhibit 3 in and Exhibit 4 respectively.

Field #	Name	size/ type	Description
1	Event	integer	31:24 reserved. 0000_0000b 23:16 Event Sensor Type An *Event Sensor* is a logical entity that is responsible for detecting events. The *Event Sensor Type* field indicates what types of events the sensor is monitoring. E.g. *temperature, voltage, current, BIOS, POST, processor, fan, etc.* (This field corresponds to the IPMI 'Sensor Type' field, and conceptually maps to the 'cause of trap' field in the Phoenix proposal.) 15:8 Event Type Code indicating what type of transition / state change triggered the trap. (Corresponds to IPMI 'Event Type' field) The code is split into the following ranges: 00-0Bh = generic - can be used with any type of sensor 6Fh = sensor specific 70h-7Fh = OEM all other = reserved See Table 4, below, for generic event type codes 7:0 Event Offset Indicates which particular event occurred for a given Event Type. This field allows events to be extended on a per Event Type basis—making it easier to manage the Event Type 'name space'. 7 0 = Assertion Event. (Event occurred when state became asserted) 1 = Deassertion Event. 6:4 reserved. 000b. 3:0 Offset Value. Per IPMI, up to 15 different discrete states are allowed per each Event Type. 0Fh = unspecified.

Exhibit 3: Specific Trap Field

octet #	Name	size/ type	Description
1:16	GUID	16 bytes	GUID for the platform, per SMBIOS 2.2 / PXE specifications. All 0's = unspecified (need to use agent-addr to identify the platform that generated the trap). The following specifies the octet ordering of sub-fields within the GUID field: Octet offset time_low 0:3 msbyte in offset 0 time_mid 4:5 msbyte in offset 4 time_hi_and_version 6:7 msbyte in offset 6 clock_seq_hi_and_reserved 8 clock_seq_low 9 node 10-15 msbyte in offset 10
17:18	Sequence # / Cookie	word	0000h = unspecified. The function of this field is specific to the trap source type. The intent of the field is to provide a 'sequence #' that can be used to differentiate a re-transmitted (re-tried) trap from a new trap instance. It may also be used for applications that know how to respond to the trap source to give a positive acknowledge. There are restrictions on the use of this field: • An application must not be required to interpret this field in order to accept the trap or decode the fields in the trap. • The trap source must not rely on getting a response or other action from an application that interprets this field. [I.e. must be designed with the assumption that it may not get a response.] • All trap source types must support a 0000h=unspecified content for this field.
19:22	Local Timestamp	dword	Differs from SNMP trap timestamp in that this is platform local time based. Encoded as number of seconds from 0:00 1/1/98. 0000 0000 = unspecified.
23:24	UTC Offset	word	UTC Offset in minutes (two's complement, signed. -720 to +720, 0xFFFF=unspecified).

Exhibit 4: Variable-Bindings Field

7. Intel® Baseboard In-band SNMP Sub-Agent

The Intel®® Baseboard SNMP Sub-agent supports SNMPv1 and SNMPv2c versions. This is in-band agent. It communicates to the BMC using the driver that supports the IPMI specification. This way sub-agent runs on an Intel server in the OS, and gets information on all the platform sensors. Different flavors of Microsoft® Windows*, RHEL* and SUSE* operating systems are supported. The key features supported by in-band agent for the EPSD server boards include information on the sensors, chassis, memory, and processor. The details of MIB for voltage and temperature sensors are shown in **Exhibit 5**. It also provides the health of each sensor and overall health of Intel server board. It supports GET, GET-NEXT and SET commands to the SNMP management application. The SET command is supported to modify the sensor thresholds. SNMP traps are generated whenever the state of sensor is changed. See section 7.1 for details. The out-of-band SNMP agent runs on the BMC FW, and has limited support; see section 6 for details.

Following is the list of sensors supported by the in-band SNMP sub-agent.

- Chassis intrusions
- Processor
- Memory
- Power supply
- Voltage
- System FAN
- Temperature
- Drive Slot Presence

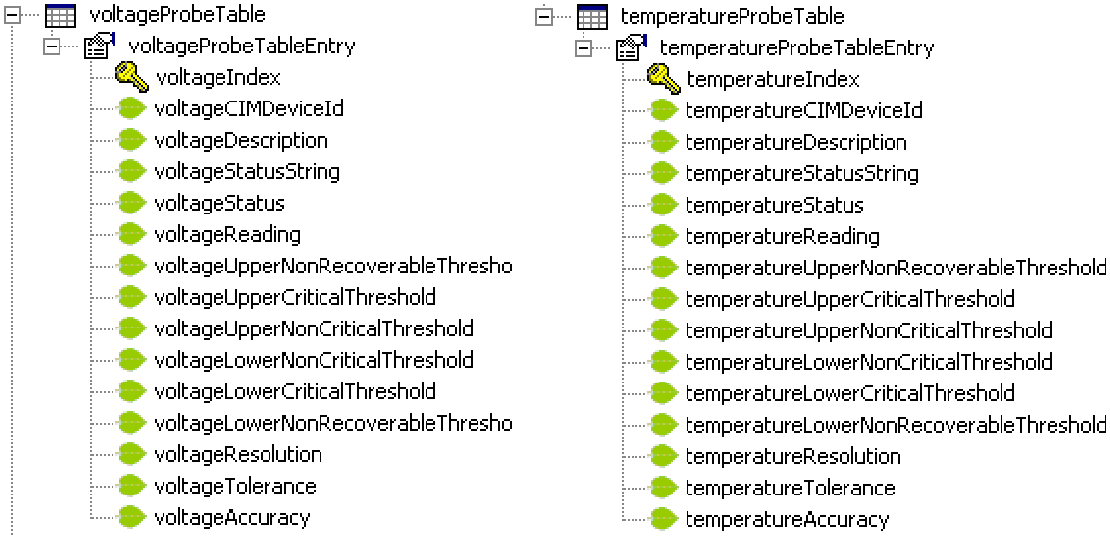
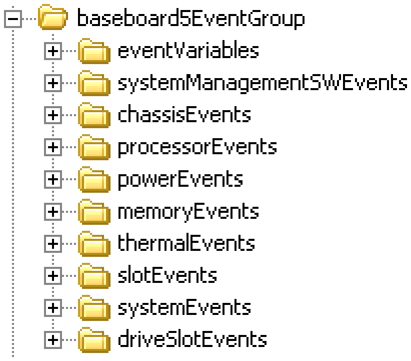

voltageProbeTable
voltageProbeTableEntry
voltageIndex
voltageCIMDeviceId
voltageDescription
voltageStatusString
voltageStatus
voltageReading
voltageUpperNonRecoverableThresho
voltageUpperCriticalThreshold
voltageUpperNonCriticalThreshold
voltageLowerNonCriticalThreshold
voltageLowerCriticalThreshold
voltageLowerNonRecoverableThresho
voltageResolution
voltageTolerance
voltageAccuracy
temperatureProbeTable
temperatureProbeTableEntry
temperatureIndex
temperatureCIMDeviceId
temperatureDescription
temperatureStatusString
temperatureStatus
temperatureReading
temperatureUpperNonRecoverableThreshold
temperatureUpperCriticalThreshold
temperatureUpperNonCriticalThreshold
temperatureLowerNonCriticalThreshold
temperatureLowerCriticalThreshold
temperatureLowerNonRecoverableThreshold
temperatureResolution
temperatureTolerance
temperatureAccuracy

Exhibit 5: Voltage and Temperature Sensor MIB

7.1 SNMP Trap Events

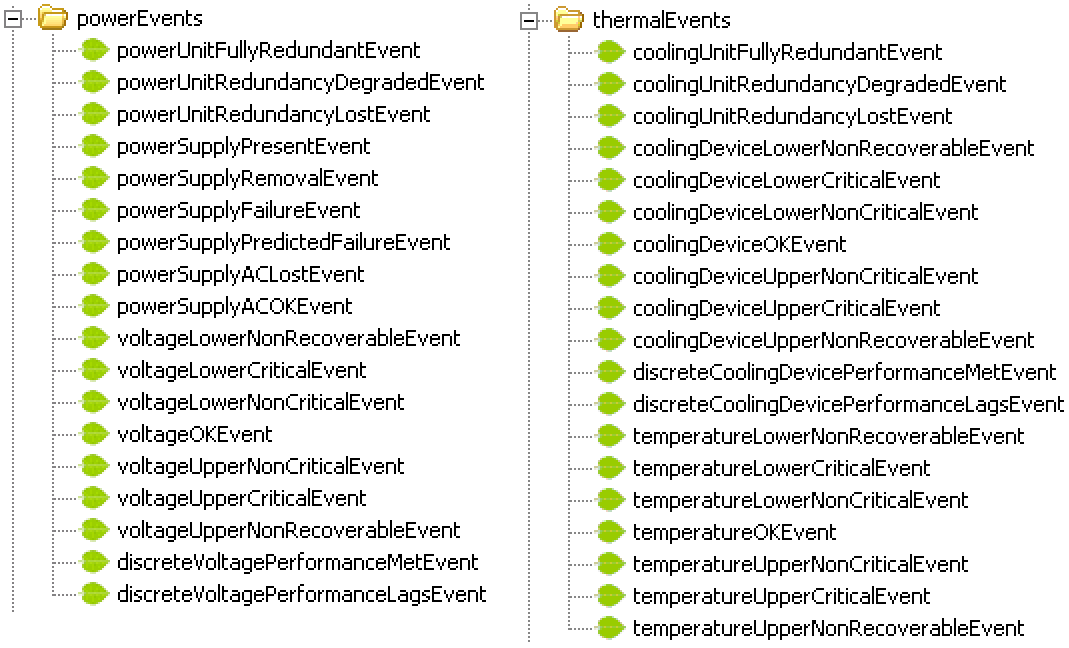
In order to parse and understand the SNMP trap, the management application must know the MIB of the generated trap such that it can take appropriate action. Additionally, the management application should be configured, by its IP address, in the SNMP agent to receive the trap. The Intel® SNMP subagent sends the trap for every new entry in the SEL inserted by the BMC. One or more SNMP traps are generated for each SEL event. The different types of events based on the devices present on the server are shown in **Exhibit 6**.



- baseboard5EventGroup
  - eventVariables
  - systemManagementSWEvents
  - chassisEvents
  - processorEvents
  - powerEvents
  - memoryEvents
  - thermalEvents
  - slotEvents
  - systemEvents
  - driveSlotEvents

Exhibit 6: Event Group MIB

The in-band agent polls for the new SEL entries and generates the SNMP trap to the registered SNMP client systems. The number of events generated for each SEL entry depends on the type of SEL entry and Intel platforms. It also depends on the firmware and type of platform. The list of events for few sensors, for example, power unit, power supply, voltage, cooling unit, and fan, are shown in **Exhibit 7**.



- powerEvents
  - powerUnitFullyRedundantEvent
  - powerUnitRedundancyDegradedEvent
  - powerUnitRedundancyLostEvent
  - powerSupplyPresentEvent
  - powerSupplyRemovalEvent
  - powerSupplyFailureEvent
  - powerSupplyPredictedFailureEvent
  - powerSupplyACLostEvent
  - powerSupplyACOKEvent
  - voltageLowerNonRecoverableEvent
  - voltageLowerCriticalEvent
  - voltageLowerNonCriticalEvent
  - voltageOKEvent
  - voltageUpperNonCriticalEvent
  - voltageUpperCriticalEvent
  - voltageUpperNonRecoverableEvent
  - discreteVoltagePerformanceMetEvent
  - discreteVoltagePerformanceLagsEvent

- thermalEvents
  - coolingUnitFullyRedundantEvent
  - coolingUnitRedundancyDegradedEvent
  - coolingUnitRedundancyLostEvent
  - coolingDeviceLowerNonRecoverableEvent
  - coolingDeviceLowerCriticalEvent
  - coolingDeviceLowerNonCriticalEvent
  - coolingDeviceOKEvent
  - coolingDeviceUpperNonCriticalEvent
  - coolingDeviceUpperCriticalEvent
  - coolingDeviceUpperNonRecoverableEvent
  - discreteCoolingDevicePerformanceMetEvent
  - discreteCoolingDevicePerformanceLagsEvent
  - temperatureLowerNonRecoverableEvent
  - temperatureLowerCriticalEvent
  - temperatureLowerNonCriticalEvent
  - temperatureOKEvent
  - temperatureUpperNonCriticalEvent
  - temperatureUpperCriticalEvent
  - temperatureUpperNonRecoverableEvent

Exhibit 7: Power and Thermal Events MIB

The variables in the variable bindings in SNMP trap message generated by Intel® SNMP Subagent are event description, severity, slot description, and slow CIM device ID. Each variable is mapped with an OID in the MIB.

7.2 Modify Sensor Thresholds

Some of the sensors thresholds can be modified by the management application. These sensors thresholds are written to the SDR (Sensor Data Record) repository for each device accessible through the BMC. The voltage, temperature and fan sensors have modifiable thresholds. The SNMP SET command is used to modify the sensor thresholds. The support for the SET command for the SNMP sub-agent is configurable, for example, whether a user can change a sensor's threshold value or not etc.

7.3 Memory and Processor Information

Information about all the memory arrays and also every memory device on each array is available. The status, location, max capacity, number of memory devices, mirroring status, sparing status, and RAID status are exposed to the SNMP application. For each memory device, its status, total width, data width, size, speed, bank location, and memory type along with other information is made available from SMBIOS table.

7.4 Overall System Health Status

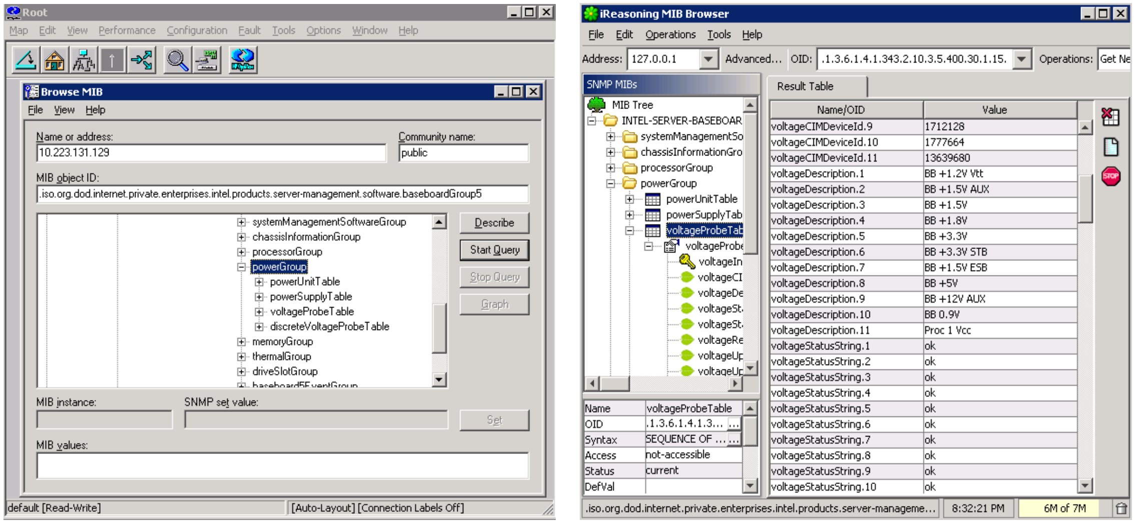
The overall system health status is based on the status of sensors supported by the platform. The health of each sensor is monitored based on its upper and lower thresholds limits. If all the devices are in healthy state, the overall system health would be shown as "OK". The details of information in the trap message are given in section 7.1.

8. SNMP Application – a client-side perspective

Organizations always seek to improve their network availability and performance, increase staff efficiency, and reduce operating expenses. The benefits of Network Management Software (NMS) are enhanced by integration with other device management software. This enables optimized utilization of resources including people, processes, and technology. A server with Intel® SNMP Subagent can be managed by any client software through SNMP interface. This section illustrates few NMS products that can be used to manage Intel Server Hardware.

8.1 HP Software & Solutions*

The HP Software & Solutions is a new name of HP Open View [8]. HP Network Node Manager (NNM), a part of Open View suite, has integration capability of different NMS. The MIB browser, as shown in Exhibit 8a, is also supplied as part of NNM. After loading the MIB, user can perform the SNMP Get/Set operations on an Intel® SNMP Subagent.



Name or address: 10.223.131.129

Community name: public

MIB object ID: .iso.org.dod.internet.private.enterprises.intel.products.server-management.software.baseboardGroup5

MIB instance:

SNMP set value:

MIB values:

default [Read-Write]   [Auto-Layout] [Connection Labels Off]

Address: 127.0.0.1   Advanced...   OID: .1.3.6.1.4.1.343.2.10.3.5.400.30.1.15.   Operations: Get Ne

Name	voltageProbeTable
OID	.1.3.6.1.4.1.3...
Syntax	SEQUENCE OF ...
Access	not-accessible
Status	current
DefVal	

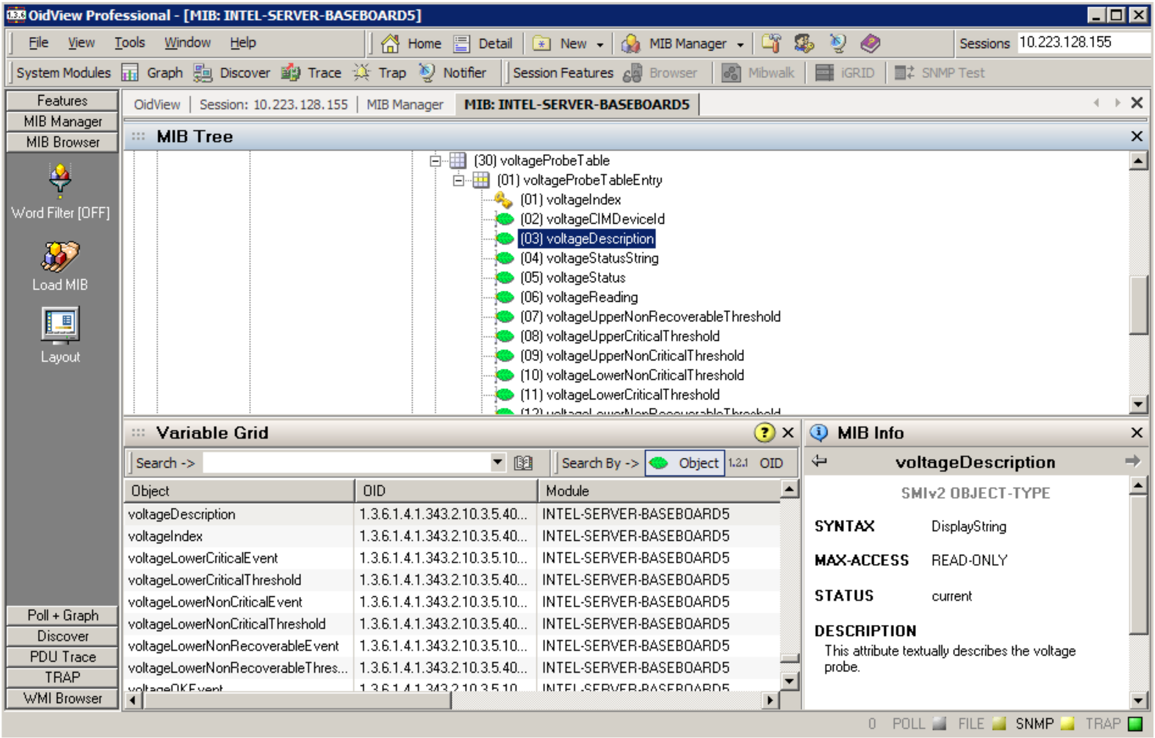
Name/OID	Value
voltageCIMDeviceId.9	1712128
voltageCIMDeviceId.10	1777664
voltageCIMDeviceId.11	13639680
voltageDescription.1	BB +1.2V Vtt
voltageDescription.2	BB +1.5V AUX
voltageDescription.3	BB +1.5V
voltageDescription.4	BB +1.8V
voltageDescription.5	BB +3.3V
voltageDescription.6	BB +3.3V STB
voltageDescription.7	BB +1.5V ESB
voltageDescription.8	BB +5V
voltageDescription.9	BB +12V AUX
voltageDescription.10	BB 0.9V
voltageDescription.11	Proc 1 Vcc
voltageStatusString.1	ok
voltageStatusString.2	ok
voltageStatusString.3	ok
voltageStatusString.4	ok
voltageStatusString.5	ok
voltageStatusString.6	ok
voltageStatusString.7	ok
voltageStatusString.8	ok
voltageStatusString.9	ok
voltageStatusString.10	ok

.iso.org.dod.internet.private.enterprises.intel.products.server-manageme...   8:32:21 PM   6M of 7M

Exhibit 8: (a) HP Open View and (b) iReasoning MIB Browsers

8.2 iReasoning Networks*

iReasoning Networks **[9]** provides "*SysUpTime Network Monitor*" and "*iReasoning MIB Browser*" network management solutions. The SysUpTime MSP edition provides a complete solution for Managed Service Providers (MSP) and organizations with multiple sites to monitor any number of networks from a single central site. The "*iReasoning Browser*", as shown in **Exhibit 8b**, allows user to load multiple MIBs, send SNMP requests and receive traps.



OidView Professional - [MIB: INTEL-SERVER-BASEBOARD5]
File View Tools Window Help
Home Detail New MIB Manager
Sessions 10.223.128.155
System Modules Graph Discover Trace Trap Notifier Session Features Browser Mibwalk iGRID SNMP Test
Features
MIB Manager
MIB Browser
Word Filter [OFF]
Load MIB
Layout
Poll + Graph
Discover
PDU Trace
TRAP
WMI Browser
OidView
Session: 10.223.128.155
MIB Manager
MIB: INTEL-SERVER-BASEBOARD5
MIB Tree
Variable Grid
MIB Info
0 POLL FILE SNMP TRAP

(30) voltageProbeTable
(01) voltageProbeTableEntry
(01) voltageIndex
(02) voltageCIMDeviceId
(03) voltageDescription
(04) voltageStatusString
(05) voltageStatus
(06) voltageReading
(07) voltageUpperNonRecoverableThreshold
(08) voltageUpperCriticalThreshold
(09) voltageUpperNonCriticalThreshold
(10) voltageLowerNonCriticalThreshold
(11) voltageLowerCriticalThreshold
(12) voltageLowerNonRecoverableThreshold

Search ->   Search By -> Object 1.2.1 OID

Object	OID	Module
voltageDescription	1.3.6.1.4.1.343.2.10.3.5.40...	INTEL-SERVER-BASEBOARD5
voltageIndex	1.3.6.1.4.1.343.2.10.3.5.40...	INTEL-SERVER-BASEBOARD5
voltageLowerCriticalEvent	1.3.6.1.4.1.343.2.10.3.5.10...	INTEL-SERVER-BASEBOARD5
voltageLowerCriticalThreshold	1.3.6.1.4.1.343.2.10.3.5.40...	INTEL-SERVER-BASEBOARD5
voltageLowerNonCriticalEvent	1.3.6.1.4.1.343.2.10.3.5.10...	INTEL-SERVER-BASEBOARD5
voltageLowerNonCriticalThreshold	1.3.6.1.4.1.343.2.10.3.5.40...	INTEL-SERVER-BASEBOARD5
voltageLowerNonRecoverableEvent	1.3.6.1.4.1.343.2.10.3.5.10...	INTEL-SERVER-BASEBOARD5
voltageLowerNonRecoverableThres...	1.3.6.1.4.1.343.2.10.3.5.40...	INTEL-SERVER-BASEBOARD5
voltageOKEvent	1.3.6.1.4.1.343.2.10.3.5.10...	INTEL-SERVER-BASEBOARD5

voltageDescription

SMIv2 OBJECT-TYPE

SYNTAX DisplayString

MAX-ACCESS READ-ONLY

STATUS current

DESCRIPTION
This attribute textually describes the voltage probe.

Exhibit 9: OIDView SNMP MIB Browser

8.3 OIDView SNMP Tools

The OidView SNMP Tools from ByteSphere is a suite of network management products that includes MIB Browser, Trap Manager and SNMP Tester etc. [10]. OidView MIB Browser, as shown in Exhibit 1Exhibit 9, is SNMP management application and network management analyzer. It allows users to send SNMP requests and connect to manage network devices using telnet. It analyzes the MIB walk files from multiple vendors.

9. Summary

The remote management of devices in a network is the prime requirement from IT management. This paper describes remote management of Intel® EPSD servers using SNMP. The overall system health information and alerts related to sensors from EPSD servers are available using SNMP management protocol. We presented different solutions and features of SNMP based remote management of Intel ® server hardware. These solutions are out-of-band SNMP agent in BMC and in-band Intel® Baseboard SNMP sub-agent.

10. Bibliography

[1]. The Internet Engineering Task Force (IETF) http://www.ietf.org.

[2]. For information on the SNMP protocol (RFC 1157), versions and different standards associated with this technology is available on Internet Engineering Task Force (http://www.ietf.org).

[3]. The user guide on the Intel® SNMP subagent to manage supported hardware components is available on http://www.intel.com/support/motherboards/server/sysmgmt/sb/CS-029304.htm.

[4]. What is IPMI? http://www.intel.com/design/servers/ipmi/ipmi.htm

[5]. http://download.intel.com/design/servers/ipmi/IPMI_and_CIM_Spring2005_IDF.pps

[6]. http://download.intel.com/design/servers/ipmi/IPMIv2_0rev1_0.pdf

[7]. http://download.intel.com/design/servers/ipmi/PET100.pdf

[8]. HP Network Node Manager - BTO software.

[9]. iReasoning – Network management and application management software.

[10]. OIDView SNMP Tools - http://www.oidview.com/

Disclaimers

Information in this document is provided in connection with Intel® products. No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted by this document. Except as provided in Intel's Terms and Conditions of Sale for such products, Intel assumes no liability whatsoever, and Intel disclaims any express or implied warranty, relating to sale and/or use of Intel products including liability or warranties relating to fitness for a particular purpose, merchantability, or infringement of any patent, copyright or other intellectual property right. Intel products are not intended for use in medical, life saving, or life sustaining applications. Intel may make changes to specifications and product descriptions at any time, without notice.

This document contains information on products in the design phase of development. Do not finalize a design with this information. Revised information will be published when the product is available. Verify with your local sales office that you have the latest datasheet before finalizing a design. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them.

The document may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Intel, Pentium, Celeron, and Xeon are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Copyright © Intel Corporation 2010

*Other names and brands may be claimed as the property of others.