

Netra Modular System Developer's Guide

ORACLE®

Part No: E62890-01
August 2015

Part No: E62890-01

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS. Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

Référence: E62890-01

Copyright © 2015, Oracle et/ou ses affiliés. Tous droits réservés.

Ce logiciel et la documentation qui l'accompagne sont protégés par les lois sur la propriété intellectuelle. Ils sont concédés sous licence et soumis à des restrictions d'utilisation et de divulgation. Sauf stipulation expresse de votre contrat de licence ou de la loi, vous ne pouvez pas copier, reproduire, traduire, diffuser, modifier, breveter, transmettre, distribuer, exposer, exécuter, publier ou afficher le logiciel, même partiellement, sous quelque forme et par quelque procédé que ce soit. Par ailleurs, il est interdit de procéder à toute ingénierie inverse du logiciel, de le désassembler ou de le décompiler, excepté à des fins d'interopérabilité avec des logiciels tiers ou tel que prescrit par la loi.

Les informations fournies dans ce document sont susceptibles de modification sans préavis. Par ailleurs, Oracle Corporation ne garantit pas qu'elles soient exemptes d'erreurs et vous invite, le cas échéant, à lui en faire part par écrit.

Si ce logiciel, ou la documentation qui l'accompagne, est concédé sous licence au Gouvernement des Etats-Unis, ou à toute entité qui délivre la licence de ce logiciel ou l'utilise pour le compte du Gouvernement des Etats-Unis, la notice suivante s'applique:

U.S. GOVERNMENT END USERS. Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

Ce logiciel ou matériel a été développé pour un usage général dans le cadre d'applications de gestion des informations. Ce logiciel ou matériel n'est pas conçu ni n'est destiné à être utilisé dans des applications à risque, notamment dans des applications pouvant causer des dommages corporels. Si vous utilisez ce logiciel ou matériel dans le cadre d'applications dangereuses, il est de votre responsabilité de prendre toutes les mesures de secours, de sauvegarde, de redondance et autres mesures nécessaires à son utilisation dans des conditions optimales de sécurité. Oracle Corporation et ses affiliés déclinent toute responsabilité quant aux dommages causés par l'utilisation de ce logiciel ou matériel pour ce type d'applications.

Oracle et Java sont des marques déposées d'Oracle Corporation et/ou de ses affiliés. Tout autre nom mentionné peut correspondre à des marques appartenant à d'autres propriétaires qu'Oracle.

Intel et Intel Xeon sont des marques ou des marques déposées d'Intel Corporation. Toutes les marques SPARC sont utilisées sous licence et sont des marques ou des marques déposées de SPARC International, Inc. AMD, Opteron, le logo AMD et le logo AMD Opteron sont des marques ou des marques déposées d'Advanced Micro Devices. UNIX est une marque déposée d'The Open Group.

Ce logiciel ou matériel et la documentation qui l'accompagne peuvent fournir des informations ou des liens donnant accès à des contenus, des produits et des services émanant de tiers. Oracle Corporation et ses affiliés déclinent toute responsabilité ou garantie expresse quant aux contenus, produits ou services émanant de tiers, sauf mention contraire stipulée dans un contrat entre vous et Oracle. En aucun cas, Oracle Corporation et ses affiliés ne sauraient être tenus pour responsables des pertes subies, des coûts occasionnés ou des dommages causés par l'accès à des contenus, produits ou services tiers, ou à leur utilisation, sauf mention contraire stipulée dans un contrat entre vous et Oracle.

Accessibilité de la documentation

Pour plus d'informations sur l'engagement d'Oracle pour l'accessibilité à la documentation, visitez le site Web Oracle Accessibility Program, à l'adresse <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Accès au support électronique

Les clients Oracle qui ont souscrit un contrat de support ont accès au support électronique via My Oracle Support. Pour plus d'informations, visitez le site <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> ou le site <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> si vous êtes malentendant.

Contents

- Using This Documentation** 9
 - Product Documentation Library 9
 - Feedback 9

- Understanding the Interfaces and Software** 11
 - Understanding Interface Options 11
 - CLI Overview 11
 - SNMP Overview 13
 - IPMI Overview 13
 - Oracle ILOM Overview 14
 - SAF HPI Overview 15
 - OpenHPI Overview 16
 - Management Software 17

- Programming API Functions** 19
 - General Functions 19
 - Domain Functions 19
 - Session Management 20
 - Domain Discovery 20
 - Resource Presence 21
 - Event Logs 21
 - Events 22
 - Domain Alarms 22
 - Resource Functions 22
 - Resource Data Record Repository 23
 - Sensors 23
 - Controls 24
 - Inventory Data Repositories 24
 - Watchdog Timers 25
 - FUMI 25

Hot-Swap	26
Reset Management	27
Power Management	27
Programming OEM Controls	29
LED Controls	29
LED Get	30
LED Set	30
FRU Configuration Controls	30
FRU Get	31
FRU Set	31
IP Controls	32
IP Controls Get	32
IP Controls Set	33
Point-to-Point Get Controls	33
Alarm Profile Controls	34
Alarm Profile Get	35
Alarm Profile Set	35
Understanding Netra Modular System Resources	37
Modular System Resources	37
Chassis Resources	38
TOP_LEVEL_PRODUCT Controls	38
Modular System Inventory Data Repositories	39
FMM System Board Resources	39
FMM System Board Sensors	39
FMM System Board Inventory Data Repositories	40
FMM Service Processor Sensors	41
FMM Service Processor Controls	41
FMM Service Processor Inventory Data Repositories	42
Switch Inventory Data Repositories	42
BMC Inventory Data Repositories	42
FBN Resources	43
FBN Sensors	43
FBN Controls	45
FBN Inventory Data Repositories	45
Understanding Oracle Server X5-2M Resources	47
Node Resources	47

Node System Resources	50
System Sensors	51
System Controls	51
System Inventory Data Repositories	52
Motherboard Resources	52
Motherboard Sensors	53
Motherboard Inventory Data Repositories	53
Motherboard Riser Card Sensors	55
Motherboard/BIOS Inventory Data Repositories	55
Node SP Resources	55
SP Controls	56
SP Inventory Data Repositories	56
Node Processor Resources	57
Processor Sensors	57
Processor Controls	57
Processor Inventory Data Repositories	58
DIMM Resources	58
DIMM Sensors	58
DIMM Controls	59
DIMM Inventory Data Repositories	59
Disk Drive Resources	59
Disk Drive Sensors	60
Disk Drive Controls	60
Disk Drive Backplane Inventory Data Repositories	61
Node Power Supply Resources	61
Power Supply Sensors	61
Power Supply Inventory Data Repositories	62
Node PCIe Slot Resources	63
Node Fan Module Resources	63
Fan Module Sensors	64
Fan Module Controls	64
Node BMC Resources	64
Understanding Switch Resources	67
Switch Resources	67
Switch System Resources	68
Switch System Sensors	68
Switch System Controls	72
Switch System Inventory Data Repositories	72

Switch Power Supply Resources	73
Power Supply Sensors	73
Power Supply Inventory Data Repositories	74
Switch Fan Module Resources	74
Fan Module Sensors	74
Fan Module Controls	75
Switch SP Resources	75
SP Sensors	75
SP Controls	76
SP Inventory Data Repositories	77
Switch BMC Inventory Data Repositories Resources	77
Switch Chassis Resources	78
Chassis Sensors	78
Chassis Inventory Data Repositories	78
Glossary	81
Index	85

Using This Documentation

- **Overview** – Provides reference information about the interfaces and APIs of Oracle's Netra Modular System.
- **Audience** – Programmers and developers with experience coding and porting applications using APIs.
- **Required Knowledge** – Advanced experience programming and using interfaces.

Product Documentation Library

Documentation and resources for this product and related products are available at <https://www.oracle.com/goto/netra-modular-system/docs>.

Feedback

Provide feedback about this documentation at <http://www.oracle.com/goto/docfeedback>.

Understanding the Interfaces and Software

These topics provide an overview of the software environment that forms the basis for managing and developing applications for the modular system.

- [“Understanding Interface Options” on page 11](#)
- [“Management Software” on page 17](#)

Related Information

- [“Programming API Functions” on page 19](#)
- [“Programming OEM Controls” on page 29](#)

Understanding Interface Options

The following interfaces are supported.

- [“CLI Overview” on page 11](#)
- [“SNMP Overview” on page 13](#)
- [“IPMI Overview” on page 13](#)
- [“Oracle ILOM Overview” on page 14](#)
- [“SAF HPI Overview” on page 15](#)
- [“OpenHPI Overview” on page 16](#)

Related Information

- [“Management Software” on page 17](#)
- [“Programming API Functions” on page 19](#)
- [“Programming OEM Controls” on page 29](#)

CLI Overview

The [CLI](#) provides a comprehensive set of textual commands that you can issue to the system through [Oracle ILOM](#).

The Oracle ILOM CLI is based on the Distributed Management Task Force specification, Server Management Command-Line Protocol Specification (DMTF CLP). The entire specification is available at: <http://www.dmtf.org/>.

The DMTF CLP provides a management user interface for servers regardless of server state, method of access, or installed operating system. The DMTF CLP architecture models a hierarchical namespace, which is a predefined tree that contains every managed object in the system.

In this model, commands operate on a namespace of targets, which you can modify by using options and properties. This namespace defines the targets for each command. The command syntax is as follows:

```
action [options] [target] [properties]
```

The *action* refers to the command purpose or action taking place. The standard actions are set, show, create, delete, reset, start, stop, cd, version, exit, and help. The target indicates the address or path of the target of the command.

From the CLI, you can access the /FMM namespace, which includes /FMM/FB n .

In the /FMM namespace, you can manage and configure the FMM. In the /FMM namespaces, you can access and configure properties for managed components.

Namespace	CLI Management Target Description
/FMM	Use to monitor the SP on the FMM. The targets and properties in this target type are for configuring the FMM and viewing the configuration. The /FMM replaces the /CMM used in the Sun Blade 6000 Modular System.
/Frame	Use to provide inventory status, environmental conditions, and hardware management at the frame level. The /Frame address space replaces the /CH on the Sun Blade 6000 Modular System.
/Frame/FB n	Use to access and configure /Frame FBN properties and options from the FMM CLI session.

For mapping management tasks to CLI targets on the FMM, refer to 3.2.x *Oracle ILOM User's Guide for System Monitoring and Diagnostics* at: https://docs.oracle.com/cd/E37444_01/html/E37447/z40000091420195.html#scrolltoc

For information about CLI commands, refer to the *Oracle ILOM Quick Reference for CLI Commands* at: https://docs.oracle.com/cd/E37444_01/html/E37448/z4000068165586.html#scrolltoc

Related Information

- [“SNMP Overview” on page 13](#)

- “IPMI Overview” on page 13
- “Oracle ILOM Overview” on page 14
- “SAF HPI Overview” on page 15
- “OpenHPI Overview” on page 16

SNMP Overview

SNMP is a standard protocol for monitoring networked components. The SNMP is part of an internet protocol suite, defined by the Internet Engineering Task Force (IETF).

SNMP is employed by management systems to monitor networked components for conditions that need system administrative action. SNMP is of a suite of standards for network management that includes an application layer protocol, a database schema, and data objects.

To be managed, a device must have an SNMP associated agent. The agent manages requests for data representing a component's state and generates an appropriate response. The agent controls the component's state. Additionally, the agent generates SNMP traps (unsolicited messages to network management stations) to signal significant events relating to the component.

For more information, refer to SNMP specifications (RFCs 3410-3417) at:

<https://www.ietf.org/>

Related Information

- “CLI Overview” on page 11
- “IPMI Overview” on page 13
- “Oracle ILOM Overview” on page 14
- “SAF HPI Overview” on page 15
- “OpenHPI Overview” on page 16

IPMI Overview

RMCP is a standard network interface to an **IPMI** controller through the LAN and is defined by the IPMI specification. Access to the IPMI subsystem is usually through RMCP.

For the supported features, the IPMI driver user interface is compatible with the Linux OpenIPMI driver user interface.

All of the IPMI features are supported except for:

- Power on Host (IPMITool)

- Power off Host (IPMITool)
- PowerCycle Host (IPMITool)
- Shutdown Host Gracefully (IPMITool)
- Receive a message, such as a shut down request, from other IPMI clients

An IPMI driver is required to communicate with the local IPMI controller or other IPMI clients. For instance, with the IPMI driver, you can:

- Program the node front panel LEDs.
- Program the watchdog timer in the IPMI controller.
- Receive a message, like a shutdown request, from other IPMI clients.

For more information, refer to the IPMI specification at:

<http://www.intel.com/content/www/us/en/servers/ipmi/ipmi-specifications.html>

Related Information

- “CLI Overview” on page 11
- “SNMP Overview” on page 13
- “Oracle ILOM Overview” on page 14
- “SAF HPI Overview” on page 15
- “OpenHPI Overview” on page 16

Oracle ILOM Overview

Through [Oracle ILOM](#), you can manage systems and resources. Compute nodes in the modular system support all of the standard Oracle ILOM interfaces and features that are documented in the Oracle ILOM documentation. For the modular system, use the Oracle ILOM for managing the compute nodes, management nodes, [FSA](#), and [FMM](#).

Note - Compute nodes in the modular system do not have a serial console available, because this port is used to detect the node's presence in the modular system and to read configuration data. The only physical access to the SP is through the network (NET_MGT) port through a switch in the [FMM](#).

A new target is available under the /SYS target for the FMM. You can get the error and presence status of the FMM with other properties, such as the IP address of the FMM. However, this target might not be visible in the default mode, which masks legacy targets. By default, the Oracle ILOM CLI interface shows the [SDM](#), which brings uniformity to the CLI on all systems.

For detailed information, refer to the Oracle ILOM documentation:

Oracle ILOM User's Guide for System Monitoring and Diagnostics https://docs.oracle.com/cd/E37444_01/html/E37447/z40000091420195.html#scrolltoc

Oracle ILOM Quick Reference for CLI Commands https://docs.oracle.com/cd/E37444_01/html/E37448/z4000068165586.html#scrolltoc

Related Information

- “CLI Overview” on page 11
- “SNMP Overview” on page 13
- “IPMI Overview” on page 13
- “SAF HPI Overview” on page 15
- “OpenHPI Overview” on page 16

SAF HPI Overview

The **SAF HPI** provides platform-independent C APIs to use as building blocks for monitoring and controlling highly-available systems. The **HPI** standards specify generic data structures, functions, and resources that you can program to interact with manageable components of the modular system.

Using the standard interfaces, you can develop applications and middleware to access and manage hardware components. The HPI allows portability of HPI code across many hardware platforms.

The SAF HPI is based upon the **IPMI** specification, which defines platform-independent capabilities and data formats. However, HPI is a generic interface specification, therefore, you can implement it on any platform with compatible platform management technology.

For more information about the HPI specification and usage examples, refer to <http://www.saforum.org>.

The SAF HPI model defines four concepts: sessions, domains, resources, and entities.

Sessions – Provide all access to an HPI implementation by **HPI user**. An HPI session is opened on a single domain; one HPI user can have multiple sessions open at once, and there can be multiple sessions open on any given domain at once. Sessions also provide access to events created or forwarded by the domain accessed by the session. An HPI user accesses the system through sessions, where each session is opened on a domain. A session provides access to domain functions and to a set of resources that are accessible through the domain.

Domains – All HPI user functions are accessed through sessions, and each session is associated with a single domain. A domain provides access to zero or more resources and provides a set of associated services and capabilities. The latter are logically grouped into an abstraction

called a *domain controller*. The resources that are accessible through a domain are listed in the domain's Resource Presence Table (RPT). The contents of this table can change over time, and the domain's session management capability rejects any attempt to access a resource that is not currently listed in the domain's [RPT](#).

Resources – Provide management access to the entities within the system. Frequently, resources represent functions performed by a local control processor used for management of the entity's hardware. Each resource is responsible for presenting a set of management instruments and management capabilities to the HPI User. Resources can be dynamically added and removed in a system as hot-swappable system components.

Entities – Represent the physical components of the system. Each entity has a unique identifier, called an entity path, which is defined by the component's location in the physical containment hierarchy of the system.

Related Information

- [“CLI Overview” on page 11](#)
- [“SNMP Overview” on page 13](#)
- [“IPMI Overview” on page 13](#)
- [“Oracle ILOM Overview” on page 14](#)
- [“OpenHPI Overview” on page 16](#)

OpenHPI Overview

[OpenHPI](#) implements of the SAF HPI C application programming interface to provide platform management capabilities, such as:

- Configuration – The components in the system.
- Inventory – The vendor, model, version, and serial number of the components.
- Status – The temperature, voltage, fan speed, and state of the LEDs.
- Control – The ability to power on, power off, and reset the system, along with setting the WDT.

OpenHPI is an open source implementation of the SAF HPI. OpenHPI implements the APIs with a client library that interfaces with a daemon. The OpenHPI client library communicates with the OpenHPI daemon using a TCP-based OpenHPI-specific protocol. The daemon has a modular, configurable, plug-in architecture. Plug-ins are dynamically loaded, shared libraries, that interface with the systems that are being monitored and managed.

An OpenHPI plugin developed specifically for the Netra Modular System uses the [RMCP](#) to interface with the [FMM](#) and the Oracle ILOM interfaces of the nodes and switches in the system.

For a detailed description of the OpenHPI, with supported return codes, refer to the OpenHPI specification at: <http://www.openhpi.org/>.

Related Information

- [“CLI Overview” on page 11](#)
- [“SNMP Overview” on page 13](#)
- [“IPMI Overview” on page 13](#)
- [“Oracle ILOM Overview” on page 14](#)
- [“SAF HPI Overview” on page 15](#)
- [“OpenHPI Overview” on page 16](#)

Management Software

The modular system has two components that provide management access, the [FMM](#) and the [FSA](#). The FMM is the access point for [OOB](#) management. The FSA is the single-point of [IB](#) management.

For in-service systems management operations, all external access (such as element management systems) occurs through the FSA, with the FSA accessing the FMM.

Direct access to the FMM is limited to the service operations needed to bring the FSA in-service. The FSA is a [RMS](#) that hosts the system management software stack for frame-level management.

The system management software:

- Interfaces with the [FMM](#) and the networked components through [Oracle ILOM](#).
- Can run on a dedicated management node or a virtualized domain.
- Provides semantics-to-solution application and data center manager.
- Provides service management API to the upper-layer service manager.
- Provides hardware model sensors and messages via [JMX](#).
- Bases the equipment modeling part of the management devices on [CIM](#).

The equipment model and Java API documentation are delivered as part of modular system release package, in a `.tar` file. After you expand the `.tar` file, you can view the documentation with a web browser.

Related Information

- [“Understanding Interface Options” on page 11](#)

- [“Understanding Netra Modular System Resources” on page 37](#)

Programming API Functions

Note - The functions described are SAF HPI Functions. These functions are not supported by the other interfaces.

These topics list the API functions that are supported with this product:

- [“General Functions” on page 19](#)
- [“Domain Functions” on page 19](#)
- [“Resource Functions” on page 22](#)

Related Information

- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming OEM Controls” on page 29](#)
- [“Understanding Netra Modular System Resources” on page 37](#)

General Functions

- `saHpiVersionGet()`
- `saHpiInitialize()`
- `saHpiFinalize()`

Related Information

- [“Domain Functions” on page 19](#)
- [“Resource Functions” on page 22](#)

Domain Functions

- [“Session Management” on page 20](#)

- [“Domain Discovery” on page 20](#)
- [“Resource Presence” on page 21](#)
- [“Event Logs” on page 21](#)
- [“Events” on page 22](#)
- [“Domain Alarms” on page 22](#)

Related Information

- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming OEM Controls” on page 29](#)
- [“Understanding Netra Modular System Resources” on page 37](#)

Session Management

- `saHpiSessionOpen()`
- `saHpiSessionClose()`
- `saHpiDiscover()`

Related Information

- [“Domain Discovery” on page 20](#)
- [“Resource Presence” on page 21](#)
- [“Event Logs” on page 21](#)
- [“Events” on page 22](#)
- [“Domain Alarms” on page 22](#)

Domain Discovery

- `saHpiDomainInfoGet()`
- `saHpiDrtEntryGet()`
- `saHpiDomainTagSet()`

Related Information

- [“Session Management” on page 20](#)
- [“Resource Presence” on page 21](#)

- [“Event Logs” on page 21](#)
- [“Events” on page 22](#)
- [“Domain Alarms” on page 22](#)

Resource Presence

- `saHpiRptEntryGet()`
- `saHpiRptEntryGetResourceId()`
- `saHpiRptEntryGetByResourceId()`
- `saHpiMyEntityPathGet()`
- `saHpiResourceIdGet()`
- `saHpiGetIdByEntityPath()`
- `saHpiGetcChildEntityPath()`

Related Information

- [“Session Management” on page 20](#)
- [“Domain Discovery” on page 20](#)
- [“Event Logs” on page 21](#)
- [“Events” on page 22](#)
- [“Domain Alarms” on page 22](#)

Event Logs

- `saHpiEventLogInfoGet()`
- `saHpiEventLogCapabilities()`
- `saHpiEventLogEntryGet()`
- `saHpiEventLogClear()`

Related Information

- [“Session Management” on page 20](#)
- [“Domain Discovery” on page 20](#)
- [“Resource Presence” on page 21](#)
- [“Events” on page 22](#)
- [“Domain Alarms” on page 22](#)

Events

- `saHpiSubscribe()`
- `saHpiUnsubscribe()`
- `saHpiEventGet()`

Related Information

- [“Session Management” on page 20](#)
- [“Domain Discovery” on page 20](#)
- [“Resource Presence” on page 21](#)
- [“Event Logs” on page 21](#)
- [“Domain Alarms” on page 22](#)

Domain Alarms

- `saHpiAlarmGetNext()`
- `saHpiAlarmGet()`

Related Information

- [“Session Management” on page 20](#)
- [“Domain Discovery” on page 20](#)
- [“Resource Presence” on page 21](#)
- [“Event Logs” on page 21](#)
- [“Events” on page 22](#)

Resource Functions

- [“Resource Data Record Repository” on page 23](#)
- [“Sensors” on page 23](#)
- [“Controls” on page 24](#)
- [“Inventory Data Repositories” on page 24](#)
- [“Watchdog Timers” on page 25](#)
- [“FUMI” on page 25](#)
- [“Hot-Swap” on page 26](#)
- [“Reset Management” on page 27](#)

- [“Power Management” on page 27](#)

Related Information

- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming OEM Controls” on page 29](#)
- [“Understanding Netra Modular System Resources” on page 37](#)

Resource Data Record Repository

- `saHpiRdrGet()`
- `saHpiRdrGetByInstrumentId()`
- `saHpiRdrUpdateCountGet()`

Related Information

- [“Sensors” on page 23](#)
- [“Controls” on page 24](#)
- [“Inventory Data Repositories” on page 24](#)
- [“Watchdog Timers” on page 25](#)
- [“FUMI” on page 25](#)
- [“Hot-Swap” on page 26](#)
- [“Reset Management” on page 27](#)
- [“Power Management” on page 27](#)

Sensors

- `saHpiSensorReadingGet()`
- `saHpiSensorThresholdsGet()`
- `saHpiSensorThresholdsSet()`
- `saHpiSensorTypeGet()`
- `saHpiSensorEnableGet()`
- `saHpiSensorEventEnableGet()`

Related Information

- [“Resource Data Record Repository” on page 23](#)

- [“Controls” on page 24](#)
- [“Inventory Data Repositories” on page 24](#)
- [“Watchdog Timers” on page 25](#)
- [“FUMI” on page 25](#)
- [“Hot-Swap” on page 26](#)
- [“Reset Management” on page 27](#)
- [“Power Management” on page 27](#)

Controls

- `saHpiControlTypeGet()`
- `saHpiControlGet()`
- `saHpiControlSet()`

Related Information

- [“Resource Data Record Repository” on page 23](#)
- [“Sensors” on page 23](#)
- [“Inventory Data Repositories” on page 24](#)
- [“Watchdog Timers” on page 25](#)
- [“FUMI” on page 25](#)
- [“Hot-Swap” on page 26](#)
- [“Reset Management” on page 27](#)
- [“Power Management” on page 27](#)

Inventory Data Repositories

- `saHpiIdrInfoGet()`
- `saHpiIdrAreaHeaderGet()`
- `saHpiIdrFieldGet()`

Related Information

- [“Resource Data Record Repository” on page 23](#)
- [“Sensors” on page 23](#)
- [“Controls” on page 24](#)
- [“Watchdog Timers” on page 25](#)

- [“FUMI” on page 25](#)
- [“Hot-Swap” on page 26](#)
- [“Reset Management” on page 27](#)
- [“Power Management” on page 27](#)

Watchdog Timers

- `saHpiWatchdogTimerGet()`
- `saHpiWatchdogTimerSet()`
- `saHpiWatchdogTimerReset()`

Related Information

- [“Resource Data Record Repository” on page 23](#)
- [“Sensors” on page 23](#)
- [“Controls” on page 24](#)
- [“Inventory Data Repositories” on page 24](#)
- [“FUMI” on page 25](#)
- [“Hot-Swap” on page 26](#)
- [“Reset Management” on page 27](#)
- [“Power Management” on page 27](#)

FUMI

Most of the FUMI functions are not supported. Use the FUMI functions to perform firmware upgrades of the FMM firmware, Oracle ILOM firmware, and BIOS.

- `saHpiFumiSpecInfoGet()`
- `saHpiFumiServiceImpactGet()`
- `saHpiFumiSourceSet()`
- `saHpiFumiSourceInfoGet()`
- `saHpiFumiInstallStart()`
- `saHpiFumiUpgradeStatusGet()`

Related Information

- [“Resource Data Record Repository” on page 23](#)

- [“Sensors” on page 23](#)
- [“Controls” on page 24](#)
- [“Inventory Data Repositories” on page 24](#)
- [“Watchdog Timers” on page 25](#)
- [“Hot-Swap” on page 26](#)
- [“Reset Management” on page 27](#)
- [“Power Management” on page 27](#)

Hot-Swap

Only the `saHpiHotSwapStateGet()` function is supported.

All resources except the `SYSTEM_CHASSIS` resource for rackmount servers follow the simple two-state hot-swap (`NOT_PRESENT` or `ACTIVE`). The `SYSTEM_CHASSIS` resource for a server can transition through the five-state HPI hot-swap model (`NOT_PRESENT`, `INACTIVE`, `INSERTION_PENDING`, `ACTIVE`, or `EXTRACTION_PENDING`).

When you insert a server into the modular system, the modular system attempts to create an RMCP session with the Oracle ILOM of the server. Once the session is established, the plugin discovers the resources, sensors, controls, and inventories on the server.

After the discovery is complete, first `NOT_PRESENT` then `ACTIVE` hot-swap events are sent for each resource associated with the server, except for the `SYSTEM_CHASSIS` resource. The `SYSTEM_CHASSIS` resource follows the five-state HPI hot-swap model described in the HPI specification.

If a server is removed from the NMS, the `ACTIVE` to `NOT_PRESENT` hot-swap events are generated for each resource associated with the server. The `SYSTEM_CHASSIS` resource for the server follows the five-state HPI hot-swap model described in the HPI specification.

Related Information

- [“Resource Data Record Repository” on page 23](#)
- [“Sensors” on page 23](#)
- [“Controls” on page 24](#)
- [“Inventory Data Repositories” on page 24](#)
- [“Watchdog Timers” on page 25](#)
- [“FUMI” on page 25](#)
- [“Reset Management” on page 27](#)
- [“Power Management” on page 27](#)

Reset Management

- `saHpiResourceResetStateGet()`
Returns SAHPI_RESET_DEASSERT for SYSTEM_BOARD resources.
- `saHpiResourceResetStateSet()`
Supported for SYSTEM_BOARD resources.
SAHPI_COLD_RESET – Implemented with the IPMI chassis control power cycle command.
SAHPI_WARM_RESET – Implemented with the IPMI chassis control warm reset command.

Related Information

- [“Resource Data Record Repository” on page 23](#)
- [“Sensors” on page 23](#)
- [“Controls” on page 24](#)
- [“Inventory Data Repositories” on page 24](#)
- [“Watchdog Timers” on page 25](#)
- [“FUMI” on page 25](#)
- [“Hot-Swap” on page 26](#)
- [“Power Management” on page 27](#)

Power Management

- `saHpiResourcePowerStateGet()`
Supported for SYSTEM_BOARD resources.
- `saHpiResourcePowerStateSet()`
Supported for SYSTEM_BOARD resources.
SAHPI_POWER_CYCLE – Implemented with the IPMI chassis control power cycle command.
SAHPI_POWER_ON – Implemented with the IPMI chassis control power up command.
SAHPI_POWER_OFF – Use the IPMI chassis control soft shutdown command first. If the resource does not power off, then use the IPMI chassis control power down command.

Related Information

- [“Resource Data Record Repository” on page 23](#)
- [“Sensors” on page 23](#)
- [“Controls” on page 24](#)

- [“Inventory Data Repositories” on page 24](#)
- [“Watchdog Timers” on page 25](#)
- [“FUMI” on page 25](#)
- [“Hot-Swap” on page 26](#)
- [“Reset Management” on page 27](#)

Programming OEM Controls

These topics describe how to program OEM controls using the `saHpiControlGet()` and `saHpiControlSet()` functions:

- [“LED Controls” on page 29](#)
- [“FRU Configuration Controls” on page 30](#)
- [“IP Controls” on page 32](#)
- [“Point-to-Point Get Controls” on page 33](#)
- [“Alarm Profile Controls” on page 34](#)

Related Information

- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming API Functions” on page 19](#)

LED Controls

Use the LED controls to get and set the state of LEDs.

- [“LED Get” on page 30](#)
- [“LED Set” on page 30](#)

Related Information

- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming API Functions” on page 19](#)

LED Get

Parameter	Input	Output
SessionId	Session ID	
ResourceId	Resource ID	
CtrlNum	LED control number	
CtrlMode		SAHP_CTRL_MODE_AUTO
CtrlState.Type		SAHPI_CTRL_TYPE_OEM
CtrlState.StateUnion.Oem.Body[0]		■ 0 if the LED is off ■ 1 if the LED is on

Related Information

- [“LED Set” on page 30](#)

LED Set

Parameter	Input
SessionId	Session ID
ResourceId	Resource ID
CtrlNum	LED control number
CtrlMode	SAHP_CTRL_MODE_MANUAL
CtrlState.Type	SAHPI_CTRL_TYPE_OEM
CtrlState.StateUnion.Oem.Body[0]	■ 0 if the LED is off ■ 1 if the LED is on

Related Information

- [“LED Get” on page 30](#)

FRU Configuration Controls

Use the FRU configuration controls to get and set FRU information using the `saHpiControlGet()` and `saHpiControlSet()` functions.

- [“FRU Get” on page 31](#)

- [“FRU Set” on page 31](#)

Related Information

- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming API Functions” on page 19](#)

FRU Get

Parameter	Input	Output
SessionId	Session ID	
ResourceId	Resource ID	
CtrlNum	FRU control number	
CtrlMode		SAHP_CTRL_MODE_MANUAL
CtrlType		
CtrlState.StateUnion. Oem.MId		0x2A
CtrlState.StateUnion. Oem.BodyLength		The number of bytes of data
CtrlState.StateUnion. Oem.Body [0]	<ul style="list-style-type: none"> ▪ 0 get the static part of the FRU configuration ▪ 1 to get the product description string ▪ 2 to get the firmware requirements 	
CtrlState.StateUnion. Oem.Body [1:x]		Requested FRU information

Related Information

- [“FRU Set” on page 31](#)

FRU Set

Parameter	Input
SessionId	Session ID
ResourceId	Resource ID
CtrlNum	FRU control number

Parameter	Input
CtrlMode	SAHP_CTRL_MODE_MANUAL
CtrlType	
CtrlState.StateUnion.Oem.MId	
CtrlState.StateUnion.Oem.BodyLength	The number of bytes of data
CtrlState.StateUnion.Oem.Body[0]	<ul style="list-style-type: none"> ■ 0 to set the static part of the FRU configuration ■ 1 to set the product description string ■ 2 to set the firmware requirements
CtrlState.StateUnion.Oem.Body[1:x]	

Related Information

- [“FRU Get” on page 31](#)

IP Controls

- [“IP Controls Get” on page 32](#)
- [“IP Controls Set” on page 33](#)

Related Information

- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming API Functions” on page 19](#)

IP Controls Get

Parameter	Input	Output
SessionId	Session ID	
ResourceId	Resource ID	
CtrlNum	IP control number	
CtrlMode		SAHP_CTRL_MODE_MANUAL
CtrlType		
CtrlState.StateUnion.Oem.MId		0x2A
CtrlState.StateUnion.Oem.BodyLength		5
CtrlState.StateUnion.Oem.Body[0]	<ul style="list-style-type: none"> ■ 0 to get the IP address 	

Parameter	Input	Output
	<ul style="list-style-type: none"> ■ 1 to get the subnet ■ 2 to get the netmask 	
CtrlState.StateUnion.Oem.Body [1:x]		Requested IP information

Related Information

- [“IP Controls Set” on page 33](#)

IP Controls Set

Parameter	Input
SessionId	Open session ID
ResourceId	Resource ID
CtrlNum	IP control number
CtrlMode	SAHP_CTRL_MODE_MANUAL
CtrlType	
CtrlState.StateUnion.Oem.MId	
CtrlState.StateUnion.Oem.BodyLength	A number from 1 to 5
CtrlState.StateUnion.Oem.Body[0]	<ul style="list-style-type: none"> ■ 0 to set the IP address ■ 1 to set the subnet ■ 2 to set the netmask
CtrlState.StateUnion.Oem.Body[1:x]	IP information

Related Information

- [“IP Controls Get” on page 32](#)

Point-to-Point Get Controls

Parameter	Input	Output
SessionId	Session ID	
ResourceId	Resource ID	
CtrlNum	Point-to-point control number	

Parameter	Input	Output
CtrlMode		SAHP_CTRL_MODE_MANUAL
CtrlType		
CtrlState.StateUnion. Oem.MId		0x2A
CtrlState.StateUnion. Oem.BodyLength		0xf
CtrlState.StateUnion. Oem.Body [0]	Connection record number, starting from 1	
CtrlState.StateUnion. Oem.Body [1]	Connection list entry number, starting from 1	
CtrlState.StateUnion. Oem.Body [2]		Total number of connection records
CtrlState.StateUnion. Oem.Body [3]		The next connection list entry number, or 0xff to indicate that there are no more entries in the list
CtrlState.StateUnion. Oem.Body [4:5]		Connection ID
CtrlState.StateUnion. Oem.Body [6:7]		Location ID A
CtrlState.StateUnion. Oem.Body [8:9]		Port ID A
CtrlState.StateUnion. Oem.Body [10:11]		Location ID B
CtrlState.StateUnion. Oem.Body [12:13]		Port ID B
CtrlState.StateUnion. Oem.Body [15]		Policy byte

Related Information

- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming API Functions” on page 19](#)

Alarm Profile Controls

- [“Alarm Profile Get” on page 35](#)
- [“Alarm Profile Set” on page 35](#)

Related Information

- [“Understanding the Interfaces and Software” on page 11](#)

- [“Programming API Functions” on page 19](#)

Alarm Profile Get

Parameter	Input	Output
SessionId	Session ID	
ResourceId	Resource ID	
CtrlNum	Alarm control number	
CtrlMode		
CtrlType		
CtrlState.StateUnion.Oem.Body[0]		Blade profile
CtrlState.StateUnion.Oem.Body[1]		Switch profile
CtrlState.StateUnion.Oem.Body[2]		Processor speed profile
CtrlState.StateUnion.Oem.Body[3]		Fan module profile
CtrlState.StateUnion.Oem.Body[4]		Temperature nonrecoverable profile
CtrlState.StateUnion.Oem.Body[5]		Temperature critical profile
CtrlState.StateUnion.Oem.Body[6]		Temperature noncritical profile
CtrlState.StateUnion.Oem.Body[7]		Fan speed profile

Related Information

- [“Alarm Profile Set” on page 35](#)

Alarm Profile Set

Parameter	Input
SessionId	Session ID
ResourceId	Resource ID
CtrlNum	Alarm control number
CtrlMode	N/A
CtrlType	N/A
CtrlState.StateUnion.Oem.Body[0]	Blade profile
CtrlState.StateUnion.Oem.Body[1]	Switch profile
CtrlState.StateUnion.Oem.Body[2]	Processor speed profile

Parameter	Input
CtrlState.StateUnion.0em.Body[3]	Fan module profile
CtrlState.StateUnion.0em.Body[4]	Temperature nonrecoverable profile
CtrlState.StateUnion.0em.Body[5]	Temperature critical profile
CtrlState.StateUnion.0em.Body[6]	Temperature noncritical profile
CtrlState.StateUnion.0em.Body[7]	Fan speed profile

Related Information

- [“Alarm Profile Get” on page 35](#)

Understanding Netra Modular System Resources

These topics describe the supported resources for the Netra Modular System:

- [“Modular System Resources” on page 37](#)
- [“FMM System Board Resources” on page 39](#)
- [“Switch Inventory Data Repositories” on page 42](#)
- [“BMC Inventory Data Repositories” on page 42](#)
- [“FBN Resources” on page 43](#)

Related Information

- [“Programming API Functions” on page 19](#)
- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming OEM Controls” on page 29](#)

Modular System Resources

These topics describe the modular system's top-level components that operate the system and provide status, alerts, and so on.

- [“Chassis Resources” on page 38](#)
- [“TOP_LEVEL_PRODUCT Controls” on page 38](#)
- [“Modular System Inventory Data Repositories” on page 39](#)

Related Information

- [“Programming API Functions” on page 19](#)
- [“Understanding the Interfaces and Software” on page 11](#)

- [“Programming OEM Controls” on page 29](#)

Chassis Resources

Tag	Entity Path	Capabilities
FMM	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}	FRU, INVENTORY, POWER, RDR, RESOURCE, SENSOR, RESET
FMM/SP	{SYSTEM_CHASSIS,0}{SYS_MGMT_MODULE,0}	CONTROL, FRU, INVENTORY, RDR, RESOURCE, SENSOR
SWITCH	{SYSTEM_CHASSIS,0}{OTHER_SYSTEM_BOARD,0}	FRU, INVENTORY, RDR, RESOURCE
TOP_LEVEL_PROD	{SYSTEM_CHASSIS,0}	CONTROL, FRU, RDR, RESOURCE, INVENTORY
BMC	{SYSTEM_CHASSIS,0}{MC_FIRMWARE,0}	EVENT_LOG, FRU, FUMI, INVENTORY, RDR, RESOURCE, WATCHDOG
/FBNx x = [2-32]	{SYSTEM_CHASSIS,0}{OTHER_CHASSIS_BOARD,x}	CONTROL, FRU, INVENTORY, RDR, RESOURCE, SENSOR

Related Information

- [“TOP_LEVEL_PRODUCT Controls” on page 38](#)
- [“Modular System Inventory Data Repositories” on page 39](#)

TOP_LEVEL_PRODUCT Controls

ID	Type	Output Type	Default Mode
USER_ALARM	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_LED	SAHPI_CTRL_MODE_AUTO
MINOR_ALARM	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_LED	SAHPI_CTRL_MODE_AUTO
MAJOR_ALARM	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_LED	SAHPI_CTRL_MODE_AUTO
CRITICAL_ALARM	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_LED	SAHPI_CTRL_MODE_AUTO
FAULT	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_LED	SAHPI_CTRL_MODE_AUTO
OK	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_LED	SAHPI_CTRL_MODE_AUTO

Related Information

- [“Chassis Resources” on page 38](#)
- [“Modular System Inventory Data Repositories” on page 39](#)

Modular System Inventory Data Repositories

ID	Area Type	Field Types
TOP_LEVEL_PROD	PRODUCT_INFO	MANUFACTURER
		PRODUCT_NAME
		PART_NUMBER
		SERIAL_NUMBER

Related Information

- [“Chassis Resources” on page 38](#)
- [“TOP_LEVEL_PRODUCT Controls” on page 38](#)

FMM System Board Resources

- [“FMM System Board Sensors” on page 39](#)
- [“FMM System Board Inventory Data Repositories” on page 40](#)
- [“FMM Service Processor Sensors” on page 41](#)
- [“FMM Service Processor Controls” on page 41](#)
- [“FMM Service Processor Inventory Data Repositories” on page 42](#)

Related Information

- [“Programming API Functions” on page 19](#)
- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming OEM Controls” on page 29](#)

FMM System Board Sensors

ID	Type	Event Category	Read Support
/FMM/DC2DC/T_AMB	TEMPERATURE	SAHPI_EC_THRESHOLD	TRUE

ID	Type	Event Category	Read Support
/FMM/PSU0/T_IN	TEMPERATURE	SAHPI_EC_THRESHOLD	TRUE
/FMM/PSU1/T_IN	TEMPERATURE	SAHPI_EC_THRESHOLD	TRUE
/FMM/T_INLET	TEMPERATURE	SAHPI_EC_THRESHOLD	TRUE
T_AMB	TEMPERATURE	SAHPI_EC_THRESHOLD	TRUE

Related Information

- [“FMM System Board Inventory Data Repositories” on page 40](#)
- [“FMM Service Processor Sensors” on page 41](#)
- [“FMM Service Processor Controls” on page 41](#)
- [“FMM Service Processor Inventory Data Repositories” on page 42](#)

FMM System Board Inventory Data Repositories

ID	Area Type	Field Types
FMM	PRODUCT_INFO	MANUFACTURER
		PRODUCT_NAME
		PART_NUMBER
		SERIAL_NUMBER
		CUSTOM
FMM	BOARD_INFO	MFG_DATETIME
		PRODUCT_NAME
		SERIAL_NUMBER
		PART_NUMBER
		CUSTOM

Related Information

- [“FMM System Board Sensors” on page 39](#)
- [“FMM Service Processor Sensors” on page 41](#)
- [“FMM Service Processor Controls” on page 41](#)
- [“FMM Service Processor Inventory Data Repositories” on page 42](#)

FMM Service Processor Sensors

ID	Type	Event Category	Events	Read Support
/FMM/SP/PRGRS	OEM_SENSOR	SAHPI_EC_SENSOR_SPECIFIC	SAHPI_ES_STATE_00 SAHPI_ES_STATE_01 SAHPI_ES_STATE_02 SAHPI_ES_STATE_03 SAHPI_ES_STATE_04	FALSE

Related Information

- [“FMM System Board Sensors” on page 39](#)
- [“FMM System Board Inventory Data Repositories” on page 40](#)
- [“FMM Service Processor Controls” on page 41](#)
- [“FMM Service Processor Inventory Data Repositories” on page 42](#)

FMM Service Processor Controls

ID	Type	Output Type	Default Mode
/FMM/IP	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_IP_ADDRESS	SAHPI_CTRL_MODE_AUTO
/FMM/LOC	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_OEM	SAHPI_CTRL_MODE_AUTO
/FMM/P2P	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_OEM	SAHPI_CTRL_MODE_AUTO

Related Information

- [“FMM System Board Sensors” on page 39](#)
- [“FMM System Board Inventory Data Repositories” on page 40](#)
- [“FMM Service Processor Sensors” on page 41](#)
- [“FMM Service Processor Inventory Data Repositories” on page 42](#)

FMM Service Processor Inventory Data Repositories

ID	Area Type	Field Types
FMM/SP	PRODUCT_INFO	MANUFACTURER
		PRODUCT_NAME
		PRODUCT_VERSION
	CHASSIS_INFO	PART_NUMBER
		SERIAL_NUMBER

Related Information

- [“FMM System Board Sensors” on page 39](#)
- [“FMM System Board Inventory Data Repositories” on page 40](#)
- [“FMM Service Processor Sensors” on page 41](#)
- [“FMM Service Processor Controls” on page 41](#)

Switch Inventory Data Repositories

ID	Area Type	Field Types
SWITCH	PRODUCT_INFO	PRODUCT_VERSION

Related Information

- [“Programming API Functions” on page 19](#)
- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming OEM Controls” on page 29](#)

BMC Inventory Data Repositories

ID	Area Type	Field Types
BMC	PRODUCT_INFO	MANUFACTURER

ID	Area Type	Field Types
		PRODUCT_NAME
		PRODUCT_VERSION
	CHASSIS_INFO	PART_NUMBER
		SERIAL_NUMBER

Related Information

- [“Programming API Functions” on page 19](#)
- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming OEM Controls” on page 29](#)

FBN Resources

- [“FBN Sensors” on page 43](#)
- [“FBN Controls” on page 45](#)
- [“FBN Inventory Data Repositories” on page 45](#)

Related Information

- [“Programming API Functions” on page 19](#)
- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming OEM Controls” on page 29](#)

FBN Sensors

These are a sample representation of the FBN sensors. In a fully populated modular system, you will see similar entries for each FBN. In the following examples, FBNx indicates any [FBN](#) in the modular system.

ID	Type	Event Category	Events	Read Support
/FBNx/FBN_PRNT	ENTITY_PRESENCE	SAHPI_EC_PRESENCE	SAHPI_ES_ABSENT	FALSE
			SAHPI_ES_PRESENT	

ID	Type	Event Category	Events	Read Support
/FBNx/ND_PRSNT	ENTITY_PRESENCE	SAHPI_EC_PRESENCE	SAHPI_ES_ABSENT	FALSE
/FBNx/SP/PRGRS	OEM_SENSOR	SAHPI_EC_SENSOR_SPECIFIC	SAHPI_ES_PRESENT	FALSE
			SAHPI_ES_STATE_00	
			SAHPI_ES_STATE_01	
			SAHPI_ES_STATE_02	
			SAHPI_ES_STATE_03	
/FBNx/ND_FS	44 (IPMI FRU State)	SAHPI_EC_SENSOR_SPECIFIC	SAHPI_ES_STATE_04	FALSE
			SAHPI_ES_STATE_05	
			SAHPI_ES_STATE_00	
			SAHPI_ES_STATE_01	
			SAHPI_ES_STATE_02	
			SAHPI_ES_STATE_03	
			SAHPI_ES_STATE_04	
/FBNx/ND_MM	OEM_SENSOR	SAHPI_EC_SENSOR_SPECIFIC	SAHPI_ES_STATE_05	FALSE
			SAHPI_ES_STATE_06	
			SAHPI_ES_STATE_07	
			SAHPI_ES_STATE_01	
			SAHPI_ES_STATE_02	
			SAHPI_ES_STATE_03	
			SAHPI_ES_STATE_04	
			SAHPI_ES_STATE_05	
SAHPI_ES_STATE_06				
			SAHPI_ES_STATE_07	

Related Information

- [“FBN Controls” on page 45](#)
- [“FBN Inventory Data Repositories” on page 45](#)

FBN Controls

ID	Type	Output Type	Default Mode
/FBNx/OK x = [2..32]	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_LED	SAHPI_CTRL_MODE_AUTO
/FBNx/FAIL x = [2..32]	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_LED	SAHPI_CTRL_MODE_AUTO
/FBNx/HS x = [2..32]	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_OEM	SAHPI_CTRL_MODE_MANUAL
/FBNx/IP x = [2..32]	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_IP_ADDRESS	SAHPI_CTRL_MODE_MANUAL
/FBNx/AP x = [2..32]	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_OEM	SAHPI_CTRL_MODE_MANUAL
/FBNx/LOC x = [2..32]	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_OEM	SAHPI_CTRL_MODE_MANUAL

Related Information

- [“FBN Sensors” on page 43](#)
- [“FBN Inventory Data Repositories” on page 45](#)

FBN Inventory Data Repositories

ID	Area Type	Field Types
FBNx x = [2..32]	BOARD_INFO	MFG_DATETIME
		MANUFACTURER
		PRODUCT_NAME
		SERIAL_NUMBER
		PART_NUMBER
		CUSTOM

Related Information

- [“FBN Sensors” on page 43](#)
- [“FBN Controls” on page 45](#)

Understanding Oracle Server X5-2M Resources

These topics describe the Oracle Server X5-2M resources and [RDRs](#) represented by OpenHPI:

- [“Node Resources” on page 47](#)
- [“Node System Resources” on page 50](#)
- [“Motherboard Resources” on page 52](#)
- [“Node SP Resources” on page 55](#)
- [“Node Processor Resources” on page 57](#)
- [“DIMM Resources” on page 58](#)
- [“Disk Drive Resources” on page 59](#)
- [“Node Power Supply Resources” on page 61](#)
- [“Node PCIe Slot Resources” on page 63](#)
- [“Node Fan Module Resources” on page 63](#)
- [“Node BMC Resources” on page 64](#)

Related Information

- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming API Functions” on page 19](#)

Node Resources

The term “node” applies to both compute nodes and management nodes.

These tables provide details about the node resource tags, entity paths, and capabilities:

- [Table 1, “Resources SYS Through P0/D11,” on page 48](#), Resources SYS Through P0/D11
- [Table 2, “Resources P1/D0 Through P1/D11,” on page 48](#), Resources P1/D0 Through P1/D11
- [Table 3, “Resources DBP Through PS1,” on page 49](#), Resources DBP Through PS1

- [Table 4, “Resources MB/RISER1 Through BMC,” on page 49](#), Resources MB/RISER1 Through BMC

TABLE 1 Resources SYS Through P0/D11

Tag	Entity Path	Capabilities
SYS	{SYSTEM_CHASSIS,0}	CONTROL, FRU, INVENTORY, RDR, RESOURCE, SENSOR
MB	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}	FRU, INVENTORY, POWER, RDR, RESET, RESOURCE, SENSOR
MB/BIOS	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{BIOS,0}	FRU, INVENTORY, RDR, RESOURCE
/SYS/SP	{SYSTEM_CHASSIS,0}{SYS_MGMT_MODULE,0}	CONTROL, FRU, INVENTORY, RDR, RESOURCE
P0	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{PROCESSOR,0}	CONTROL, FRU, INVENTORY, RDR, RESOURCE, SENSOR
P1	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{PROCESSOR,1}	CONTROL, FRU, INVENTORY, RDR, RESOURCE, SENSOR
P0/D0	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{PROCESSOR,0}{MEMORY_DEVICE,0}	CONTROL, FRU, INVENTORY, RDR, RESOURCE, SENSOR
P0/D1	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{PROCESSOR,0}{MEMORY_DEVICE,1}	CONTROL, FRU, INVENTORY, RDR, RESOURCE, SENSOR
P0/D2	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{PROCESSOR,0}{MEMORY_DEVICE,2}	CONTROL, FRU, INVENTORY, RDR, RESOURCE, SENSOR
P0/D3	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{PROCESSOR,0}{MEMORY_DEVICE,3}	CONTROL, FRU, INVENTORY, RDR, RESOURCE, SENSOR
P0/D4	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{PROCESSOR,0}{MEMORY_DEVICE,4}	CONTROL, FRU, INVENTORY, RDR, RESOURCE, SENSOR
P0/D5	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{PROCESSOR,0}{MEMORY_DEVICE,5}	CONTROL, FRU, INVENTORY, RDR, RESOURCE, SENSOR
P0/D6	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{PROCESSOR,0}{MEMORY_DEVICE,6}	CONTROL, FRU, INVENTORY, RDR, RESOURCE, SENSOR
P0/D7	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{PROCESSOR,0}{MEMORY_DEVICE,7}	CONTROL, FRU, INVENTORY, RDR, RESOURCE, SENSOR
P0/D8	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{PROCESSOR,0}{MEMORY_DEVICE,8}	CONTROL, FRU, INVENTORY, RDR, RESOURCE, SENSOR
P0/D9	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{PROCESSOR,0}{MEMORY_DEVICE,9}	CONTROL, FRU, INVENTORY, RDR, RESOURCE, SENSOR
P0/D10	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{PROCESSOR,0}{MEMORY_DEVICE,10}	CONTROL, FRU, INVENTORY, RDR, RESOURCE, SENSOR
P0/D11	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{PROCESSOR,0}{MEMORY_DEVICE,11}	CONTROL, FRU, INVENTORY, RDR, RESOURCE, SENSOR

TABLE 2 Resources P1/D0 Through P1/D11

Tag	Entity Path	Capabilities
P1/D0	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{PROCESSOR,1}{MEMORY_DEVICE,12}	CONTROL, FRU, INVENTORY, RDR, RESOURCE, SENSOR

Tag	Entity Path	Capabilities
P1/D1	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{PROCESSOR,1}{MEMORY_DEVICE,13}	CONTROL, FRU, INVENTORY, RDR, RESOURCE, SENSOR
P1/D2	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{PROCESSOR,1}{MEMORY_DEVICE,14}	CONTROL, FRU, INVENTORY, RDR, RESOURCE, SENSOR
P1/D3	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{PROCESSOR,1}{MEMORY_DEVICE,15}	CONTROL, FRU, INVENTORY, RDR, RESOURCE, SENSOR
P1/D4	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{PROCESSOR,1}{MEMORY_DEVICE,16}	CONTROL, FRU, INVENTORY, RDR, RESOURCE, SENSOR
P1/D5	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{PROCESSOR,1}{MEMORY_DEVICE,17}	CONTROL, FRU, INVENTORY, RDR, RESOURCE, SENSOR
P1/D6	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{PROCESSOR,1}{MEMORY_DEVICE,18}	CONTROL, FRU, INVENTORY, RDR, RESOURCE, SENSOR
P1/D7	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{PROCESSOR,1}{MEMORY_DEVICE,19}	CONTROL, FRU, INVENTORY, RDR, RESOURCE, SENSOR
P1/D8	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{PROCESSOR,1}{MEMORY_DEVICE,20}	CONTROL, FRU, INVENTORY, RDR, RESOURCE, SENSOR
P1/D9	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{PROCESSOR,1}{MEMORY_DEVICE,21}	CONTROL, FRU, INVENTORY, RDR, RESOURCE, SENSOR
P1/D10	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{PROCESSOR,1}{MEMORY_DEVICE,22}	CONTROL, FRU, INVENTORY, RDR, RESOURCE, SENSOR
P1/D11	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{PROCESSOR,1}{MEMORY_DEVICE,23}	CONTROL, FRU, INVENTORY, RDR, RESOURCE, SENSOR

TABLE 3 Resources DBP Through PS1

Tag	Entity Path	Capabilities
DBP	{SYSTEM_CHASSIS,0}{DRIVE_BACKPLANE,0}	FRU, INVENTORY, RDR, RESOURCE
HDD0	{SYSTEM_CHASSIS,0}{DRIVE_BACKPLANE,0}{DISK_BAY,10}	CONTROL, FRU, RDR, RESOURCE, SENSOR
HDD1	{SYSTEM_CHASSIS,0}{DRIVE_BACKPLANE,0}{DISK_BAY,11}	CONTROL, FRU, RDR, RESOURCE, SENSOR
HDD2	{SYSTEM_CHASSIS,0}{DRIVE_BACKPLANE,0}{DISK_BAY,12}	CONTROL, FRU, RDR, RESOURCE, SENSOR
HDD3	{SYSTEM_CHASSIS,0}{DRIVE_BACKPLANE,0}{DISK_BAY,13}	CONTROL, FRU, RDR, RESOURCE, SENSOR
HDD4	{SYSTEM_CHASSIS,0}{DRIVE_BACKPLANE,1}{DISK_BAY,14}	CONTROL, FRU, RDR, RESOURCE, SENSOR
HDD5	{SYSTEM_CHASSIS,0}{DRIVE_BACKPLANE,1}{DISK_BAY,15}	CONTROL, FRU, RDR, RESOURCE, SENSOR
HDD6	{SYSTEM_CHASSIS,0}{DRIVE_BACKPLANE,1}{DISK_BAY,16}	CONTROL, FRU, RDR, RESOURCE, SENSOR
HDD7	{SYSTEM_CHASSIS,0}{DRIVE_BACKPLANE,1}{DISK_BAY,17}	CONTROL, FRU, RDR, RESOURCE, SENSOR
PS0	{SYSTEM_CHASSIS,0}{POWER_SUPPLY,0}	FRU, INVENTORY, RDR, RESOURCE, SENSOR
PS1	{SYSTEM_CHASSIS,0}{POWER_SUPPLY,1}	FRU, INVENTORY, RDR, RESOURCE, SENSOR

TABLE 4 Resources MB/RISER1 Through BMC

Tag	Entity Path	Capabilities
MB/RISER1	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{SYS_EXPANSION_BOARD,13}	FRU, RDR, RESOURCE, SENSOR

Tag	Entity Path	Capabilities
MB/RISER2	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{SYS_EXPANSION_BOARD,14}	FRU, RDR, RESOURCE, SENSOR
MB/RISER3	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{SYS_EXPANSION_BOARD,15}	FRU, RDR, RESOURCE, SENSOR
R1/PCIE1	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{SYS_EXPANSION_BOARD,13} {PCI_EXPRESS_BUS,1}	FRU, RDR, RESOURCE, SENSOR
R2/PCIE2	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{SYS_EXPANSION_BOARD,14} {PCI_EXPRESS_BUS,2}	FRU, RDR, RESOURCE, SENSOR
R3/PCIE3	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{SYS_EXPANSION_BOARD,15} {PCI_EXPRESS_BUS,3}	FRU, RDR, RESOURCE, SENSOR
R3/PCIE4	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}{SYS_EXPANSION_BOARD,15} {PCI_EXPRESS_BUS,4}	FRU, RDR, RESOURCE, SENSOR
FM0	{SYSTEM_CHASSIS,0}{COOLING_UNIT,30}	CONTROL, FRU, RDR, RESOURCE, SENSOR
FM1	{SYSTEM_CHASSIS,0}{COOLING_UNIT,31}	CONTROL, FRU, RDR, RESOURCE, SENSOR
FM2	{SYSTEM_CHASSIS,0}{COOLING_UNIT,32}	CONTROL, FRU, RDR, RESOURCE, SENSOR
FM3	{SYSTEM_CHASSIS,0}{COOLING_UNIT,33}	CONTROL, FRU, RDR, RESOURCE, SENSOR
BMC	{SYSTEM_CHASSIS,0}{MC_FIRMWARE,0}	EVENT_LOG, FRU, FUMI, INVENTORY, RDR, RESOURCE, WATCHDOG

Related Information

- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming API Functions” on page 19](#)

Node System Resources

- [“System Sensors” on page 51](#)
- [“System Controls” on page 51](#)
- [“System Inventory Data Repositories” on page 52](#)

Related Information

- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming API Functions” on page 19](#)

System Sensors

ID	Type	Event Category	Events	Read Support
PWRBS	POWER_SUPPLY	SAHPI_EC_STATE	SAHPI_ES_STATE_DEASSERTED SAHPI_ES_STATE_ASSERTED	FALSE
ACPI	SYSTEM_ACPI_POWER_STATE	SAHPI_EC_SENSOR_SPECIFIC	SAHPI_ES_STATE_00 SAHPI_ES_STATE_05	FALSE
INTSW	PHYSICAL_SECURITY	SAHPI_STATE	SAHPI_ES_STATE_DEASSERTED SAHPI_ES_STATE_ASSERTED	FALSE
T_AMB	TEMPERATURE	SAHPI_THRESHOLD	SAHPI_ES_UPPER_MINOR SAHPI_ES_UPPER_MAJOR SAHPI_ES_UPPER_CRIT	TRUE
VPS_CPUS	POWER_UNIT	SAHPI_THRESHOLD		TRUE
VPS_MEMORY	POWER_UNIT	SAHPI_THRESHOLD		TRUE
VPS_FANS	POWER_UNIT	SAHPI_THRESHOLD		TRUE
HOST_ERR	OEM_SENSOR	SAHPI_STATE	SAHPI_ES_STATE_DEASSERTED SAHPI_ES_STATE_ASSERTED	FALSE
HOST/STATUS	OS_BOOT	SENSOR_SPECIFIC	SAHPI_ES_STATE_06	FALSE
/SYS/PWR_BTN	BUTTON	SAHPI_EC_SENSOR_SPECIFIC		FALSE
/SYS/VPS	POWER_UNIT	SAHPI_THRESHOLD	SAHPI_ES_UPPER_MINOR	TRUE

Related Information

- [“System Controls” on page 51](#)
- [“System Inventory Data Repositories” on page 52](#)

System Controls

ID	Type	Output Type	Default Mode
OK	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_LED	SAHPI_CTRL_MODE_AUTO
SERVICE	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_LED	SAHPI_CTRL_MODE_AUTO
LOCATE	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_LED	SAHPI_CTRL_MODE_AUTO

ID	Type	Output Type	Default Mode
PS_FAULT	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_LED	SAHPI_CTRL_MODE_AUTO
TEMP_FAULT	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_LED	SAHPI_CTRL_MODE_AUTO
FAN_FAULT	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_LED	SAHPI_CTRL_MODE_AUTO

System Inventory Data Repositories

ID	Area Type	Field Types
SYS	PRODUCT_INFO	MANUFACTURER
		PRODUCT_NAME
		PART_NUMBER
		SERIAL_NUMBER
UUID	PRODUCT_INFO	CUSTOM_FIELD
TOP_LEVEL_PROD	PRODUCT_INFO	MANUFACTURER
		PRODUCT_NAME
		PART_NUMBER
		SERIAL_NUMBER

Related Information

- [“System Sensors” on page 51](#)
- [“System Controls” on page 51](#)

Motherboard Resources

- [“Motherboard Sensors” on page 53](#)
- [“Motherboard Inventory Data Repositories” on page 53](#)
- [“Motherboard Riser Card Sensors” on page 55](#)
- [“Motherboard/BIOS Inventory Data Repositories” on page 55](#)

Related Information

- [“Understanding the Interfaces and Software” on page 11](#)

- [“Programming API Functions” on page 19](#)

Motherboard Sensors

ID	Type	Event Category	Events	Read Support
T_CORE_NET01	TEMPERATURE	SAHPI_EC_THRESHOLD	SAHPI_ES_UPPER_MINOR SAHPI_ES_UPPER_MAJOR SAHPI_ES_UPPER_CRIT	TRUE
T_CORE_NET23	TEMPERATURE	SAHPI_EC_THRESHOLD	SAHPI_ES_UPPER_MINOR SAHPI_ES_UPPER_MAJOR SAHPI_ES_UPPER_CRIT	TRUE
T_IN_PS	TEMPERATURE	SAHPI_EC_THRESHOLD		TRUE
T_IN_PS1	TEMPERATURE	SAHPI_EC_THRESHOLD		TRUE
T_IN_SLOT1	TEMPERATURE	SAHPI_EC_THRESHOLD		TRUE
T_OUT_SLOT1	TEMPERATURE	SAHPI_EC_THRESHOLD		TRUE
T_IN_SLOT2	TEMPERATURE	SAHPI_EC_THRESHOLD		TRUE
T_OUT_SLOT2	TEMPERATURE	SAHPI_EC_THRESHOLD		TRUE
T_IN_SLOT3	TEMPERATURE	SAHPI_EC_THRESHOLD		TRUE
T_OUT_SLOT3	TEMPERATURE	SAHPI_EC_THRESHOLD		TRUE

Related Information

- [“Motherboard Inventory Data Repositories” on page 53](#)
- [“Motherboard Riser Card Sensors” on page 55](#)
- [“Motherboard/BIOS Inventory Data Repositories” on page 55](#)

Motherboard Inventory Data Repositories

ID	Area Type	Field Types
MB	BOARD_INFO	MFG_DATETIME MANUFACTURER

ID	Area Type	Field Types
		PRODUCT_NAME
		SERIAL_NUMBER
		PART_NUMBER
		CUSTOM
MB/NET0	PRODUCT_INFO	MANUFACTURER
		PRODUCT_NAME
		SERIAL_NUMBER
		PART_NUMBER
		CUSTOM
MB/NET1	PRODUCT_INFO	MANUFACTURER
		PRODUCT_NAME
		SERIAL_NUMBER
		PART_NUMBER
		CUSTOM
MB/NET2	PRODUCT_INFO	MANUFACTURER
		PRODUCT_NAME
		SERIAL_NUMBER
		PART_NUMBER
		CUSTOM
MB/NET3	PRODUCT_INFO	MANUFACTURER
		PRODUCT_NAME
		SERIAL_NUMBER
		PART_NUMBER
		CUSTOM
MB/CPLD	PRODUCT_INFO	MANUFACTURER
		PRODUCT_NAME
		PRODUCT_VERSION

Related Information

- [“Motherboard Sensors” on page 53](#)
- [“Motherboard Riser Card Sensors” on page 55](#)

- [“Motherboard/BIOS Inventory Data Repositories” on page 55](#)

Motherboard Riser Card Sensors

ID	Type	Event Category	Events	Read Support
MB/RISERx/PRSNT ,x = [1..3]	ENTITY_PRESENCE	SAHPI_EC_PRESENCE	SAHPI_ES_ABSENT	FALSE
			SAHPI_ES_PRESENT	

Related Information

- [“Motherboard Sensors” on page 53](#)
- [“Motherboard Inventory Data Repositories” on page 53](#)
- [“Motherboard/BIOS Inventory Data Repositories” on page 55](#)

Motherboard/BIOS Inventory Data Repositories

ID	Area Type	Field Types
MB/BIOS	PRODUCT_INFO	MANUFACTURER
		PRODUCT_NAME
		PART_NUMBER
		PRODUCT_VERSION

Related Information

- [“Motherboard Sensors” on page 53](#)
- [“Motherboard Inventory Data Repositories” on page 53](#)
- [“Motherboard Riser Card Sensors” on page 55](#)

Node SP Resources

- [“SP Controls” on page 56](#)
- [“SP Inventory Data Repositories” on page 56](#)

Related Information

- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming API Functions” on page 19](#)

SP Controls

ID	Type	Output Type	Default Mode
SP/OK	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_LED	SAHPI_CTRL_MODE_AUTO

Related Information

- [“SP Inventory Data Repositories” on page 56](#)

SP Inventory Data Repositories

ID	Area Type	Field Types
/SYS/SP	PRODUCT_INFO	MANUFACTURER
		PRODUCT_NAME
		PRODUCT_VERSION
SP/NET0	CHASSIS_INFO	PART_NUMBER
	PRODUCT_INFO	MANUFACTURER
SP/NET1	PRODUCT_INFO	PRODUCT_NAME
		PART_NUMBER
		SERIAL_NUMBER
		CUSTOM
		MANUFACTURER
		PRODUCT_NAME
		PART_NUMBER
		SERIAL_NUMBER
		CUSTOM

Related Information

- [“SP Controls” on page 56](#)

Node Processor Resources

- [“Processor Sensors” on page 57](#)
- [“Processor Controls” on page 57](#)
- [“Processor Inventory Data Repositories” on page 58](#)

Related Information

- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming API Functions” on page 19](#)

Processor Sensors

ID	Type	Event Category	Events	Read Support
MB/Px/PRSNT $x = [0..1]$	ENTITY_PRESENCE	SAHPI_SENSOR_SPECIFIC	SAHPI_ES_STATE_00 SAHPI_ES_STATE_01	FALSE
Px/V_DIMM $x = [0..1]$	VOLTAGE	SAHPI_EC_THRESHOLD		TRUE

Related Information

- [“Processor Controls” on page 57](#)
- [“Processor Inventory Data Repositories” on page 58](#)

Processor Controls

ID	Type	Output Type	Default Mode
Px/SERVICE $x = [0..1]$	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_LED	SAHPI_CTRL_MODE_AUTO

Related Information

- [“Processor Sensors” on page 57](#)
- [“Processor Inventory Data Repositories” on page 58](#)

Processor Inventory Data Repositories

ID	Area Type	Field Types
MB/Px x = [0..1]	PRODUCT_INFO	PRODUCT_NAME
		PART_NUMBER
		PRODUCT_VERSION

Related Information

- [“Processor Sensors” on page 57](#)
- [“Processor Controls” on page 57](#)

DIMM Resources

- [“DIMM Sensors” on page 58](#)
- [“DIMM Controls” on page 59](#)
- [“DIMM Inventory Data Repositories” on page 59](#)

Related Information

- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming API Functions” on page 19](#)

DIMM Sensors

ID	Type	Event Category	Read Support
/Px/Dy/PRSNT x = [0..1] y = [0..11]	ENTITY_PRESENCE	SAHPI_EC_PRESENCE	FALSE

Related Information

- [“DIMM Controls” on page 59](#)
- [“DIMM Inventory Data Repositories” on page 59](#)

DIMM Controls

ID	Type	Output Type	Default Mode
Px/Dy/SERV x = [0..1] y = [0..11]	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_LED	SAHPI_CTRL_MODE_AUTO

Related Information

- [“DIMM Sensors” on page 58](#)
- [“DIMM Controls” on page 59](#)

DIMM Inventory Data Repositories

ID	Area Type	Field Types
MB/Px/Dy x = [0..1] y = [0..11]	PRODUCT_INFO	MANUFACTURER PRODUCT_NAME PART_NUMBER SERIAL_NUMBER CUSTOM

Related Information

- [“DIMM Sensors” on page 58](#)
- [“DIMM Controls” on page 59](#)

Disk Drive Resources

- [“Disk Drive Sensors” on page 60](#)

- [“Disk Drive Controls” on page 60](#)
- [“Disk Drive Backplane Inventory Data Repositories” on page 61](#)

Related Information

- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming API Functions” on page 19](#)

Disk Drive Sensors

ID	Type	Event Category	Events	Read Support
HDDx/PRSNT x = [0..7]	ENTITY_PRESENCE	SAHPI_EC_PRESENCE	SAHPI_ES_ABSENT SAHPI_ES_PRESENT	FALSE
HDDx/STATE x = [0..7]	SLOT_CONNECTOR	SAHPI_EC_SENSOR_SPECIFIC		FALSE

Related Information

- [“Disk Drive Controls” on page 60](#)
- [“Disk Drive Backplane Inventory Data Repositories” on page 61](#)

Disk Drive Controls

ID	Type	Output Type	Default Mode
HDDx/SVC x = [0..7]	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_LED	SAHPI_CTRL_MODE_AUTO
HDDx/OK2RM x = [0..7]	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_LED	SAHPI_CTRL_MODE_AUTO

Related Information

- [“Disk Drive Sensors” on page 60](#)
- [“Disk Drive Backplane Inventory Data Repositories” on page 61](#)

Disk Drive Backplane Inventory Data Repositories

ID	Area Type	Field Types
DBP	BOARD_INFO	MFG_DATE_TIME MANUFACTURER PRODUCT_NAME PART_NUMBER SERIAL_NUMBER CUSTOM

Related Information

- [“Disk Drive Sensors” on page 60](#)
- [“Disk Drive Controls” on page 60](#)

Node Power Supply Resources

- [“Power Supply Sensors” on page 61](#)
- [“Power Supply Inventory Data Repositories” on page 62](#)

Related Information

- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming API Functions” on page 19](#)

Power Supply Sensors

ID	Type	Event Category	Events	Read Support
PSx/P_IN x = [0..1]	POWER_UNIT	SAHPI_EC_THRESHOLD		TRUE
PSx/P_OUT x = [0..1]	POWER_UNIT	SAHPI_EC_THRESHOLD		TRUE

ID	Type	Event Category	Events	Read Support
PSx/V_IN x = [0..1]	VOLTAGE	SAHPI_EC_THRESHOLD		TRUE
PSx/V_12V x = [0..1]	VOLTAGE	SAHPI_EC_THRESHOLD		TRUE
PSx/V_12V_STBY x = [0..1]	VOLTAGE	SAHPI_EC_THRESHOLD		TRUE
PSx/T_OUT x = [0..1]	TEMPERATURE		SAHPI_ES_UPPER_MINOR	TRUE
			SAHPI_ES_UPPER_CRIT	
PSx/STATE x = [0..1]	POWER_SUPPLY	SAHPI_EC_SENSOR_SPECIFIC	SAHPI_ES_STATE_00	FALSE
			SAHPI_ES_STATE_01	
			SAHPI_ES_STATE_02	
			SAHPI_ES_STATE_03	
			SAHPI_ES_STATE_04	
			SAHPI_ES_STATE_05	
			SAHPI_ES_STATE_06	
PSx/PRSNT x = [0..1]	ENTITY_PRESENCE	SAHPI_EC_PRESENCE	SAHPI_ES_ABSENT	FALSE
			SAHPI_ES_PRESENT	

Related Information

- [“Power Supply Inventory Data Repositories” on page 62](#)

Power Supply Inventory Data Repositories

ID	Area Type	Field Types
PSx x = [0..1]	BOARD_INFO	MFG_DATE_TIME
		MANUFACTURER
		PRODUCT_NAME
		PART_NUMBER
		SERIAL_NUMBER

Related Information

- [“Power Supply Sensors” on page 61](#)

Node PCIe Slot Resources

Only the following sensors are supported with this product.

ID	Type	Event Category	Events	Read Support
R1/PCIE1/PRSNT	ENTITY_PRESENCE	SAHPI_EC_PRESENCE	SAHPI_ES_ABSENT	FALSE
			SAHPI_ES_PRESENT	
R2/PCIE2/PRSNT	ENTITY_PRESENCE	SAHPI_EC_PRESENCE	SAHPI_ES_ABSENT	FALSE
			SAHPI_ES_PRESENT	
R3/PCIE3/PRSNT	ENTITY_PRESENCE	SAHPI_EC_PRESENCE	SAHPI_ES_ABSENT	FALSE
			SAHPI_ES_PRESENT	
R3/PCIE4/PRSNT	ENTITY_PRESENCE	SAHPI_EC_PRESENCE	SAHPI_ES_ABSENT	FALSE
			SAHPI_ES_PRESENT	

Related Information

- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming API Functions” on page 19](#)

Node Fan Module Resources

- [“Fan Module Sensors” on page 64](#)
- [“Fan Module Controls” on page 64](#)

Related Information

- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming API Functions” on page 19](#)

Fan Module Sensors

ID	Type	Event Category	Events	Read Support
FMx/PRSNT x = [0..3]	ENTITY_PRESENCE	SAHPI_EC_PRESENCE	SAHPI_ES_ABSENT SAHPI_ES_PRESENT	FALSE
FMx/Fy/TACH x = [0..3] y = [0..3]	FAN	SAHPI_EC_THRESHOLD	SAHPI_ES_LOWER_MINOR SAHPI_ES_LOWER_MAJOR SAHPI_ES_LOWER_CRIT	TRUE

Related Information

- [“Fan Module Controls” on page 64](#)

Fan Module Controls

ID	Type	Output Type	Default Mode
FMx/SERVICE x = [0..3]	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_LED	SAHPI_CTRL_MODE_AUTO

Related Information

- [“Fan Module Sensors” on page 64](#)

Node BMC Resources

Only the following data inventory repository is supported.

ID	Area Type	Field Types
BMC	CHASSIS_INFO PART_INFO	MANUFACTURER PRODUCT_NAME SERIAL_NUMBER

Related Information

- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming API Functions” on page 19](#)

Understanding Switch Resources

These topics describe the resources for both the Oracle Switch ES2-64 and the Oracle Switch ES2-72:

- [“Switch Resources” on page 67](#)
- [“Switch System Resources” on page 68](#)
- [“Switch Power Supply Resources” on page 73](#)
- [“Switch Fan Module Resources” on page 74](#)
- [“Switch SP Resources” on page 75](#)
- [“Switch BMC Inventory Data Repositories Resources” on page 77](#)
- [“Switch Chassis Resources” on page 78](#)

Related Information

- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming API Functions” on page 19](#)
- [“Understanding Netra Modular System Resources” on page 37](#)

Switch Resources

Tag	Entity Path	Capabilities
TOP_LEVEL_PROD	{SYSTEM_CHASSIS,0}	FRU, RDR, RESOURCE, SENSOR
SYS	{SYSTEM_CHASSIS,0}{SYSTEM_BOARD,0}	CONTROL, FRU, INVENTORY, POWER, RDR, RESET, RESOURCE, SENSOR
SP	{SYSTEM_CHASSIS,0}{SYS_MGMNT_MODULE,0}	CONTROL, FRU, INVENTORY, RDR, RESOURCE, SENSOR
PS0	{SYSTEM_CHASSIS,0}{POWER_SUPPLY,0}	FRU, INVENTORY, RDR, RESOURCE, SENSOR
PS1	{SYSTEM_CHASSIS,0}{POWER_SUPPLY,1}	FRU, INVENTORY, RDR, RESOURCE, SENSOR
FM0	{SYSTEM_CHASSIS,0}{COOLING_UNIT,0}	CONTROL, FRU, RDR, RESOURCE, SENSOR
FM0/FAN0	{SYSTEM_CHASSIS,0}{COOLING_UNIT,0}	FRU, RDR, RESOURCE, SENSOR

Tag	Entity Path	Capabilities
FM0/FAN1	{SYSTEM_CHASSIS,1}{COOLING_UNIT,1}	FRU, RDR, RESOURCE, SENSOR
FM1	{SYSTEM_CHASSIS,0}{COOLING_UNIT,1}	CONTROL, FRU, RDR, RESOURCE, SENSOR
FM1/FAN0	{SYSTEM_CHASSIS,0}{COOLING_DEVICE,0}	FRU, RDR, RESOURCE, SENSOR
FM1/FAN1	{SYSTEM_CHASSIS,0}{COOLING_DEVICE,1}	FRU, RDR, RESOURCE, SENSOR
FM2	{SYSTEM_CHASSIS,0}{COOLING_UNIT,2}	CONTROL, FRU, RDR, RESOURCE, SENSOR
FM2/FAN0	{SYSTEM_CHASSIS,0}{COOLING_DEVICE,4}	FRU, RDR, RESOURCE, SENSOR
FM2/FAN1	{SYSTEM_CHASSIS,0}{COOLING_DEVICE,5}	FRU, RDR, RESOURCE, SENSOR
FM3	{SYSTEM_CHASSIS,0}{COOLING_UNIT,3}	CONTROL, FRU, RDR, RESOURCE, SENSOR
FM3/FAN0	{SYSTEM_CHASSIS,0}{COOLING_DEVICE,6}	FRU, RDR, RESOURCE, SENSOR
FM3/FAN1	{SYSTEM_CHASSIS,0}{COOLING_DEVICE,7}	FRU, RDR, RESOURCE, SENSOR
FM4	{SYSTEM_CHASSIS,0}{COOLING_UNIT,4}	CONTROL, FRU, RDR, RESOURCE, SENSOR
FM4/FAN0	{SYSTEM_CHASSIS,0}{COOLING_DEVICE,8}	FRU, RDR, RESOURCE, SENSOR
FM4/FAN1	{SYSTEM_CHASSIS,0}{COOLING_DEVICE,9}	FRU, RDR, RESOURCE, SENSOR
BMC	{SYSTEM_CHASSIS,0}{MC_FIRMWARE,0}	EVENT_LOG, FRU, FUMI, INVENTORY, RDR, RESOURCE, WATCHDOG

Related Information

- [“Switch System Resources” on page 68](#)
- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming API Functions” on page 19](#)

Switch System Resources

- [“Switch System Sensors” on page 68](#)
- [“Switch System Controls” on page 72](#)
- [“Switch System Inventory Data Repositories” on page 72](#)

Related Information

- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming API Functions” on page 19](#)

Switch System Sensors

- [Table 5, “Switch System Sensors MB/IOMS0 Through MB/V_+1V0_B,” on page 69](#)

- [Table 6, “Switch System Sensors MB/V_+1V2 Through MB/V_+12V0,” on page 70](#)

TABLE 5 Switch System Sensors MB/IOMS0 Through MB/V_+1V0_B

ID	Type	Event Category	Events	Read Support
MB/IOMS0	ENTITY_PRESENCE	SAHPI_EC_PRESENCE	SAHPI_ES_ABSENT	FALSE
			SAHPI_ES_PRESENT	
MB/IOMS1	ENTITY_PRESENCE	SAHPI_EC_PRESENCE	SAHPI_ES_ABSENT	FALSE
			SAHPI_ES_PRESENT	
MB/IOMT0	ENTITY_PRESENCE	SAHPI_EC_PRESENCE	SAHPI_ES_ABSENT	FALSE
			SAHPI_ES_PRESENT	
MB/IOMT1	ENTITY_PRESENCE	SAHPI_EC_PRESENCE	SAHPI_ES_ABSENT	FALSE
			SAHPI_ES_PRESENT	
PWRSTATE	ENTITY_PRESENCE	SAHPI_EC_STATE	SAHPI_ES_STATE_DEASSERTED	FALSE
			SAHPI_ES_STATE_ASSERTED	
MB/T_SWITCH	TEMPERATURE	SAHPI_EC_THRESHOLD	SAHPI_ES_LOWER_MINOR	TRUE
			SAHPI_ES_LOWER_MAJOR	
			SAHPI_ES_LOWER_CRIT	
			SAHPI_ES_UPPER_MINOR	
			SAHPI_ES_UPPER_MAJOR	
			SAHPI_ES_UPPER_CRIT	
MB/T_AMB	TEMPERATURE	SAHPI_EC_THRESHOLD	SAHPI_ES_LOWER_MINOR	TRUE
			SAHPI_ES_LOWER_MAJOR	
			SAHPI_ES_LOWER_CRIT	
			SAHPI_ES_UPPER_MINOR	
			SAHPI_ES_UPPER_MAJOR	
			SAHPI_ES_UPPER_CRIT	
MB/V_+0V9	VOLTAGE	SAHPI_EC_THRESHOLD	SAHPI_ES_LOWER_MINOR	TRUE
			SAHPI_ES_LOWER_MAJOR	
			SAHPI_ES_LOWER_CRIT	
			SAHPI_ES_UPPER_MINOR	
			SAHPI_ES_UPPER_MAJOR	
MB/V_+0V9_B	VOLTAGE	SAHPI_EC_THRESHOLD	SAHPI_ES_UPPER_CRIT	TRUE
			SAHPI_ES_LOWER_MINOR	

Switch System Resources

ID	Type	Event Category	Events	Read Support
MB/V_+1V0_A	VOLTAGE	SAHPI_EC_THRESHOLD	SAHPI_ES_LOWER_MAJOR	TRUE
			SAHPI_ES_LOWER_CRIT	
			SAHPI_ES_UPPER_MINOR	
			SAHPI_ES_UPPER_MAJOR	
			SAHPI_ES_UPPER_CRIT	
			SAHPI_ES_LOWER_MINOR	
			SAHPI_ES_LOWER_MAJOR	
			SAHPI_ES_LOWER_CRIT	
			SAHPI_ES_UPPER_MINOR	
			SAHPI_ES_UPPER_MAJOR	
MB/V_+1V0_B	VOLTAGE	SAHPI_EC_THRESHOLD	SAHPI_ES_UPPER_CRIT	TRUE
			SAHPI_ES_LOWER_MINOR	
			SAHPI_ES_LOWER_MAJOR	
			SAHPI_ES_LOWER_CRIT	
			SAHPI_ES_UPPER_MINOR	
			SAHPI_ES_UPPER_MAJOR	

TABLE 6 Switch System Sensors MB/V_+1V2 Through MB/V_+12V0

ID	Type	Event Category	Events	Read Support
MB/V_+1V2	VOLTAGE	SAHPI_EC_THRESHOLD	SAHPI_ES_LOWER_MINOR	TRUE
			SAHPI_ES_LOWER_MAJOR	
			SAHPI_ES_LOWER_CRIT	
			SAHPI_ES_UPPER_MINOR	
			SAHPI_ES_UPPER_MAJOR	
MB/V_+1V8	VOLTAGE	SAHPI_EC_THRESHOLD	SAHPI_ES_UPPER_CRIT	TRUE
			SAHPI_ES_LOWER_MINOR	
			SAHPI_ES_LOWER_MAJOR	
			SAHPI_ES_UPPER_MINOR	

ID	Type	Event Category	Events	Read Support
MB/V_+2V5	VOLTAGE	SAHPI_EC_THRESHOLD	SAHPI_ES_UPPER_MAJOR	TRUE
			SAHPI_ES_UPPER_CRIT	
			SAHPI_ES_LOWER_MINOR	
			SAHPI_ES_LOWER_MAJOR	
			SAHPI_ES_LOWER_CRIT	
			SAHPI_ES_UPPER_MINOR	
			SAHPI_ES_UPPER_MAJOR	
MB/V_+3V3	VOLTAGE	SAHPI_EC_THRESHOLD	SAHPI_ES_UPPER_CRIT	TRUE
			SAHPI_ES_LOWER_MINOR	
			SAHPI_ES_LOWER_MAJOR	
			SAHPI_ES_LOWER_CRIT	
			SAHPI_ES_UPPER_MINOR	
			SAHPI_ES_UPPER_MAJOR	
			SAHPI_ES_UPPER_CRIT	
MB/V_+3V3_STDBY	VOLTAGE	SAHPI_EC_THRESHOLD	SAHPI_ES_LOWER_MINOR	TRUE
			SAHPI_ES_LOWER_MAJOR	
			SAHPI_ES_LOWER_CRIT	
			SAHPI_ES_UPPER_MINOR	
			SAHPI_ES_UPPER_MAJOR	
			SAHPI_ES_UPPER_CRIT	
			SAHPI_ES_UPPER_CRIT	
MB/V_+12V0	VOLTAGE	SAHPI_EC_THRESHOLD	SAHPI_ES_LOWER_MINOR	TRUE
			SAHPI_ES_LOWER_MAJOR	
			SAHPI_ES_LOWER_CRIT	
			SAHPI_ES_UPPER_MINOR	
			SAHPI_ES_UPPER_MAJOR	
			SAHPI_ES_UPPER_CRIT	
			SAHPI_ES_UPPER_CRIT	

Related Information

- [“Switch System Controls” on page 72](#)
- [“Switch System Inventory Data Repositories” on page 72](#)

Switch System Controls

ID	Type	Output Type	Default Mode
LOCATE	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_LED	SAHPI_CTRL_MODE_AUTO
OK	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_LED	SAHPI_CTRL_MODE_AUTO
SERVICE	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_LED	SAHPI_CTRL_MODE_AUTO

Related Information

- [“Switch System Sensors” on page 68](#)
- [“Switch System Inventory Data Repositories” on page 72](#)

Switch System Inventory Data Repositories

ID	Area Type	Field Types
SYS	PRODUCT_INFO	MANUFACTURER
		PRODUCT_NAME
		SERIAL_NUMBER
		PART_NUMBER
IO	BOARD_INFO	MFG_DATETIME
		MANUFACTURER
		PRODUCT_NAME
		SERIAL_NUMBER
MB	BOARD_INFO	PART_NUMBER
		CUSTOM
		MFG_DATETIME
		MANUFACTURER
		PRODUCT_NAME
		SERIAL_NUMBER
		PART_NUMBER
		CUSTOM

Related Information

- [“Switch System Sensors” on page 68](#)
- [“Switch System Controls” on page 72](#)

Switch Power Supply Resources

- [“Power Supply Sensors” on page 73](#)
- [“Power Supply Inventory Data Repositories” on page 74](#)

Related Information

- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming API Functions” on page 19](#)

Power Supply Sensors

ID	Type	Event Category	Events	Read Support
PSx/IN_POWER x = [0..1]	POWER_UNIT	SAHPI_EC_THRESHOLD		
PSx/I_IN x = [0..1]	CURRENT	SAHPI_EC_THRESHOLD		
PSx/I_OUT x = [0..1]	CURRENT	SAHPI_EC_THRESHOLD		
PSx/OUTPUT_POWER x = [0..1]	POWER_UNIT	SAHPI_EC_THRESHOLD		
PSx/PRSNT x = [0..1]	ENTITY_PRESENCE	SAHPI_EC_PRESENCE	SAHPI_ES_ABSENT SAHPI_ES_PRESENT	
PSx/V_IN x = [0..1]	VOLTAGE	SAHPI_EC_THRESHOLD		
PSx/V_OUT x = [0..1]	VOLTAGE	SAHPI_EC_THRESHOLD		

Related Information

- [“Power Supply Inventory Data Repositories” on page 74](#)

Power Supply Inventory Data Repositories

ID	Area Type	Field Types
PSx x = [0..1]	BOARD_INFO	MFG_DATETIME MANUFACTURER PRODUCT_NAME SERIAL_NUMBER PART_NUMBER

Related Information

- [“Power Supply Sensors” on page 73](#)

Switch Fan Module Resources

- [“Fan Module Sensors” on page 74](#)
- [“Fan Module Controls” on page 75](#)

Related Information

- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming API Functions” on page 19](#)

Fan Module Sensors

ID	Type	Event Category	Events	Read Support
FMx/PRSNT x = [0..4]	ENTITY_PRESENCE	SAHPI_EC_PRESENCE	SAHPI_ES_ABSENT SAHPI_ES_PRESENT	FALSE
FMx/FANy/TACH x = [0..4] y = [0..1]	FAN	SAHPI_EC_THRESHOLD	SAHPI_ES_LOWER_MINOR SAHPI_ES_LOWER_MAJOR SAHPI_ES_LOWER_CRIT	FALSE

Related Information

- [“Fan Module Controls” on page 75](#)

Fan Module Controls

ID	Type	Output Type	Default Mode
FMx/SERVICE x = [0..3]	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_LED	SAHPI_CTRL_MODE_AUTO

Related Information

- [“Fan Module Sensors” on page 74](#)

Switch SP Resources

- [“SP Sensors” on page 75](#)
- [“SP Controls” on page 76](#)
- [“SP Inventory Data Repositories” on page 77](#)

Related Information

- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming API Functions” on page 19](#)

SP Sensors

ID	Type	Event Category	Events	Read Support
SP/V_+1V0	VOLTAGE	SAHPI_EC_THRESHOLD	SAHPI_ES_LOWER_MINOR SAHPI_ES_LOWER_MAJOR SAHPI_ES_LOWER_CRIT	TRUE

ID	Type	Event Category	Events	Read Support
SP/V_+1V5	VOLTAGE	SAHPI_EC_THRESHOLD	SAHPI_ES_UPPER_MINOR	TRUE
			SAHPI_ES_UPPER_MAJOR	
			SAHPI_ES_UPPER_CRIT	
			SAHPI_ES_LOWER_MINOR	
			SAHPI_ES_LOWER_MAJOR	
			SAHPI_ES_LOWER_CRIT	
			SAHPI_ES_UPPER_MINOR	
SP/V_+1V8	VOLTAGE	SAHPI_EC_THRESHOLD	SAHPI_ES_UPPER_MINOR	TRUE
			SAHPI_ES_UPPER_MAJOR	
			SAHPI_ES_UPPER_CRIT	
			SAHPI_ES_LOWER_MINOR	
			SAHPI_ES_LOWER_MAJOR	
			SAHPI_ES_LOWER_CRIT	
			SAHPI_ES_UPPER_MINOR	
SP/V_+3V3	VOLTAGE	SAHPI_EC_THRESHOLD	SAHPI_ES_UPPER_MINOR	TRUE
			SAHPI_ES_UPPER_MAJOR	
			SAHPI_ES_UPPER_CRIT	
			SAHPI_ES_LOWER_MINOR	
			SAHPI_ES_LOWER_MAJOR	
			SAHPI_ES_LOWER_CRIT	
			SAHPI_ES_UPPER_MINOR	

Related Information

- [“SP Controls” on page 76](#)
- [“SP Inventory Data Repositories” on page 77](#)

SP Controls

ID	Type	Output Type	Default Mode
SP/OK	SAHPI_CTRL_TYPE_OEM	SAHPI_CTRL_LED	SAHPI_CTRL_MODE_AUTO

Related Information

- [“SP Sensors” on page 75](#)
- [“SP Inventory Data Repositories” on page 77](#)

SP Inventory Data Repositories

ID	Area Type	Field Types
SP	PRODUCT_INFO	MANUFACTURER
		PRODUCT_NAME
		PRODUCT_VERSION
	CHASSIS_INFO	PART_NUMBER
		SERIAL_NUMBER
SP/NET0	PRODUCT_INFO	MANUFACTURER
		PRODUCT_NAME
		PART_NUMBER
		SERIAL_NUMBER
		CUSTOM

Related Information

- [“SP Sensors” on page 75](#)
- [“SP Controls” on page 76](#)

Switch BMC Inventory Data Repositories Resources

ID	Area Type	Field Types
BMC	PRODUCT_INFO	MANUFACTURER
		PRODUCT_NAME
		PRODUCT_VERSION
		SERIAL_NUMBER
	CHASSIS_INFO	PART_NUMBER

ID	Area Type	Field Types
		SERIAL_NUMBER

Related Information

- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming API Functions” on page 19](#)

Switch Chassis Resources

These topics provide the switch chassis resources for Oracle's Netra Modular System.

- [“Chassis Sensors” on page 78](#)
- [“Chassis Inventory Data Repositories” on page 78](#)

Related Information

- [“Understanding the Interfaces and Software” on page 11](#)
- [“Programming API Functions” on page 19](#)

Chassis Sensors

ID	Type	Event Category	Events	Read Support
MB/V_VBAT	VOLTAGE	SAHPI_EC_THRESHOLD	SAHPI_ES_LOWER_MINOR	TRUE

Related Information

- [“Chassis Inventory Data Repositories” on page 78](#)

Chassis Inventory Data Repositories

ID	Area Type	Field Types
TOP_LEVEL_PROD	PRODUCT_INFO	MANUFACTURER

ID	Area Type	Field Types
		PRODUCT_NAME
		PART_NUMBER
		SERIAL_NUMBER

Related Information

- [“Chassis Sensors” on page 78](#)

Glossary

A

ARP Address Resolution Protocol.

B

bridge A device that connects multiple network components or segments.

C

CIM Common Information Model.

CLI Command line interface.

F

FBA Frame backplane adaptor (attached to a node).

FBN See [FBS PCB](#).

FBS Frame backplane segment, which is attached to the modular system frame and provides the connector to the node [FBAs](#).

FBS PCB FBS printed circuit board (also known as FBN). See also [FBS](#).

FMM Frame monitoring module. Provides the backplane features for compute nodes, management nodes, networking, and storage devices in the system. The FMM controls the frame backplane nodes and provides a management switch for the management components installed in the system.

FSA Frame system agent.

H**HA** High availability.**HPI** Hardware Platform Interface.**HPI user** Any software that uses the HPI. Software includes an operating system, high-availability middleware, application software, and so on.**I****IaaS** Infrastructure as a Service. One of three service models of cloud computing with Platform as a Service (PaaS) and Software as a Service (SaaS).**IB** Inbound.**ICMP** Internet Control Message Protocol.**ILOM** See [Oracle ILOM](#).**IPMI** Intelligent Platform Management Interface.**J****JMX** Java Management Extension API.**M****modular system** Shortened name for Oracle's Netra Modular System.**O****OOB** Out-of-bound.**OpenHPI** Open Hardware Platform Interface.**Oracle ILOM** Oracle Integrated Lights Out Manager.

P

- ping** An echo request message in ICMP.
- PSK** Pre-shared key, used for authentication.
- PVI** Private Virtual Interconnect.

Q

- QOS** Quality of service.

R

- RDR** Resource data records. RDRs are contained in the RDR repository of the [resource](#).
- resource** Represents the management access to a component in the system. Provides management capabilities such as power control and event logging. Additional capabilities might be provided through management instruments associated with a resource. These management instruments include sensors, controls, and inventory records. When applicable, management instruments are described in [RDRs](#).
- RMCP** Remote Management Control Protocol, a protocol for sending [IPMI](#) messages over a LAN.
- RMS** Rackmount server.
- RPT** Resource Presence Table.

S

- SA** Security association.
- SAF** Service Availability Forum.
- SDM** Simplified data model.
- SLB** Server load balancing.
- SNMP** Simple Network Management Protocol.
- SP** Service processor.

T

TCP Transmission Control Protocol.

tenant One of multiple tenants in a hosted cloud computing virtual network configuration. A hosting provider connects multiple tenant networks to their network and offers secure network isolation for each user.

U

UDP User Datagram Protocol.

V

VCPU Virtual CPU.

VIP Virtual IP address.

VM Virtual machine.

VNI Virtual Network Interface.

VPN Virtual private network.

VRID Virtual Router ID.

VRRP Virtual Router Redundancy Protocol.

vSwitch Virtual layer-2 switch in a virtualized network environment.

W

web tier A web that includes components that interact with end users at the outermost tier of application grid infrastructure, typically through HTTP requests and responses.

Index

A

- alarm profile
 - controls, 34
 - get, 35
 - set, 35
- API
 - functions, 19
 - functions, programming, 19

B

- BIOS inventory data repositories, 55

C

- chassis resources, 38
- CLI commands, 12
- controls
 - alarm profile
 - get, 35
 - set, 35
 - DIMM, 59
 - disk drive, 60
 - fan module, 64
 - FBN resources, 45
 - FMM service processor, 41
 - FRU
 - get, 31
 - set, 31
 - IP
 - get, 32
 - set, 33
 - LED
 - get, 30
 - set, 30

node

- processor, 57
- SP, 56
- system, 51
- resource functions, 24
- switch
 - fan module resources, 75
 - SP resources, 76
 - system, 72

D

- DIMM resources, 58
 - controls, 59
 - inventory data repositories, 59
 - sensors, 58
- discovery domain functions, 20
- disk drive resources
 - backplane inventory data repositories, 61
 - controls, 60
 - disk drive, 59
 - sensors, 60
- documentation
 - audience, 9
 - overview, 9
 - required knowledge, 9
- domain functions
 - described, 15
 - discovery, 20
 - domain alarms, 22
 - event
 - functions, 22
 - log functions, 21
 - resource presence functions, 21
 - session management, 20

E

entities, described, 16
event
 functions, 22
 log functions, 21

F

FBN

control resources, 45
inventory data repository resources, 45
resources, 43
sensor resources, 43

FMM

service
 processor controls, 41
 processor sensors, 41
system
 board inventory data repositories, 40
 board resources, 39
 board sensors, 39

FRU controls, 30

get, 31
set, 31

FUMI functions, 25

G

general functions, 19

H

hot-swap functions, 26

I

interface options, 11
inventory data repositories
 BMC, 42
 DIMM resources, 59
 disk drive backplane, 61
 FBN, 45
 FMM
 service processor, 42

 system, 40

motherboard/BIOS, 55

node

 motherboard, 53
 power supply, 62
 processor, 58
 SP, 56
 system, 52

switch, 42

 BMC resources, 77
 chassis resources, 78
 power supply resources, 74
 SP resources, 77
 system, 72

system

 functions, 24
 resource functions, 23
 resources, 39

IP controls, 32

get, 32
set, 33

IPMI overview, 13

L

LED controls, 29

get, 30
set, 30

M

management software overview, 17
mapping management tasks to CLI targets on the
FMM, 12
motherboard
 BIOS inventory data repositories, 55
 inventory data repositories, 53
 riser card sensors, 55
 sensors, 53

N

Netra Modular System resources, 37

node

 BMC resources, 64

- fan
 - module resources, 63
 - module sensors, 64
 - PCIe slot resources, 63
 - power supply
 - inventory data repositories, 62
 - resources, 61
 - sensors, 61
 - processor
 - controls, 57
 - inventory data repositories, 58
 - resources, 57
 - sensors, 57
 - resources, 47
 - fan module controls, 64
 - motherboard, 52, 55
 - motherboard inventory data repositories, 53
 - motherboard riser card sensors, 55
 - motherboard sensors, 53
 - SP controls, 56
 - SP inventory data repositories, 56
 - system, 50
 - system controls, 51
 - system inventory data repositories, 52
 - system sensors, 51
 - SP resources, 55
- O**
- OEM controls, programming, 29
 - OpenHPI overview, 16
 - Oracle ILOM overview, 14
 - Oracle X5-2M resources, 47
- P**
- point-to-point controls, 33
 - power management functions, 27
 - power supply switch resources, 73
 - processor
 - controls, node, 57
 - inventory data repositories, node, 58
 - resources, node, 57
 - sensors, node, 57
 - programming API functions, 19
- R**
- RCMP overview, 13
 - reset management functions, 27
 - resources
 - chassis, 38
 - FBN, 43
 - FMM
 - service processor sensors, 41
 - system board sensors, 39
 - FUMI, 25
 - hot-swap, 26
 - inventory data repositories, 23
 - modular system, 37
 - node
 - motherboard, 53
 - processor, 57
 - system, 50
 - power management, 27
 - presence functions, 21
 - reset management, 27
 - sensors, 23
 - SP node, 55
 - switch, 67
 - system inventory data repository, 24
 - top-level product, 38
 - watchdog timer, 25
- S**
- SAF HPI overview, 15
 - sensors
 - DIMMs, 58
 - disk drive, 60
 - fan module, 64
 - FBN resources, 43
 - FMM
 - service processor, 41
 - system board, 39
 - motherboard
 - node, 53
 - riser card, 55
 - node
 - power supply, 61
 - processor, 57
 - system, 51
 - switch

- fan module resources, 74
- power supply resources, 73
- SP resources, 75
- system, 68
- system functions, 23
- serial console limitation, 14
- session management functions, 20
- sessions, described, 15
- SNMP overview, 13
- software environment, 11
- switch
 - BMC inventory data repositories, resources, 77
 - chassis
 - inventory data repositories, 78
 - resources, 78
 - fan module resources, 74
 - controls, 75
 - sensors, 74
 - inventory data repositories, 42
 - power supply resources, 73
 - inventory data repositories, 74
 - sensors, 73
 - resources, 67, 67
 - SP resources, 75
 - controls, 76
 - inventory data repositories, 77
 - sensors, 75
 - system resources, 68
 - controls, 72
 - inventory data repositories, 72
 - sensors, 68
- system resources, 37

T

- top-level product resources, 38

W

- watchdog timer functions, 25