

Oracle® Communications Network Integrity

MIB-II SNMP Cartridge Guide



Release 8.0
G34178-01
October 2025

ORACLE®

Copyright © 2016, 2025, Oracle and/or its affiliates.

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, software documentation, data (as defined in the Federal Acquisition Regulation), 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 embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. 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®, Java, MySQL, and NetSuite are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside 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, Epyc, and the AMD 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.

Contents

About This Content

1 Overview

About the MIB-II SNMP Cartridge	1
About Cartridge Dependencies	1
Run-time Dependencies	1
Design Studio Dependencies	2
Opening the Cartridge Files in Design Studio	2
Building and Deploying the Cartridge	2

2 About the Cartridge Components

Discover MIB-II SNMP Action	1
MIB-II Properties Initializer Processor	2
MIB-II SNMP Collector Processor	2
MIB-II SNMP Modeler Processor	2

3 About Poll Lists

About the MIB-II SNMP Collector Poll List	1
About the IF MIB Poll List	1
About the IP MIB Poll List	2

4 About Cartridge Modeling

MIB-II SNMP Cartridge UML Representation	1
Hierarchy Mapping	2
About Oracle Communications Information Model	2
Specifications	2
Logical Device: deviceGeneric	2
Device Interface: interfaceGeneric	3
Device Interface: I2vlan	6
Other Characteristics	7

Specification Lineage	8
Sample Logical Device Hierarchy	8
Field Mappings	8
Logical Device Mapping	9
Media Interface Mapping	9
Device Interface Configuration Mapping (IPv4)	11
Device Interface Configuration (IPv6)	11
Mapping Table	12

5 About Design Studio Construction

Model Collection	1
Specification Lineage	1
Discovery Action	1
Discovery Processors	2
Property File Sample	3

6 About Design Studio Extensions

Updating Property Files	1
-------------------------	---

About This Content

This guide describes the functionality and design of the Oracle Communications Network Integrity Management Information Base (MIB) II SNMP cartridge.

Audience

This guide is intended for network administrators who want to understand the design and functionality of this cartridge and for Network Integrity developers who want either to build or extend similar cartridges.

You should have a good working knowledge of SNMP and SNMP operations, specifications, and the use of Oracle Communications Design Studio for Network Integrity.

You should be familiar with the following documents, included with this release:

- *Network Integrity Concepts*
- *Network Integrity Developer's Guide*

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.

Conventions

The following text conventions are used in this document.

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1

Overview

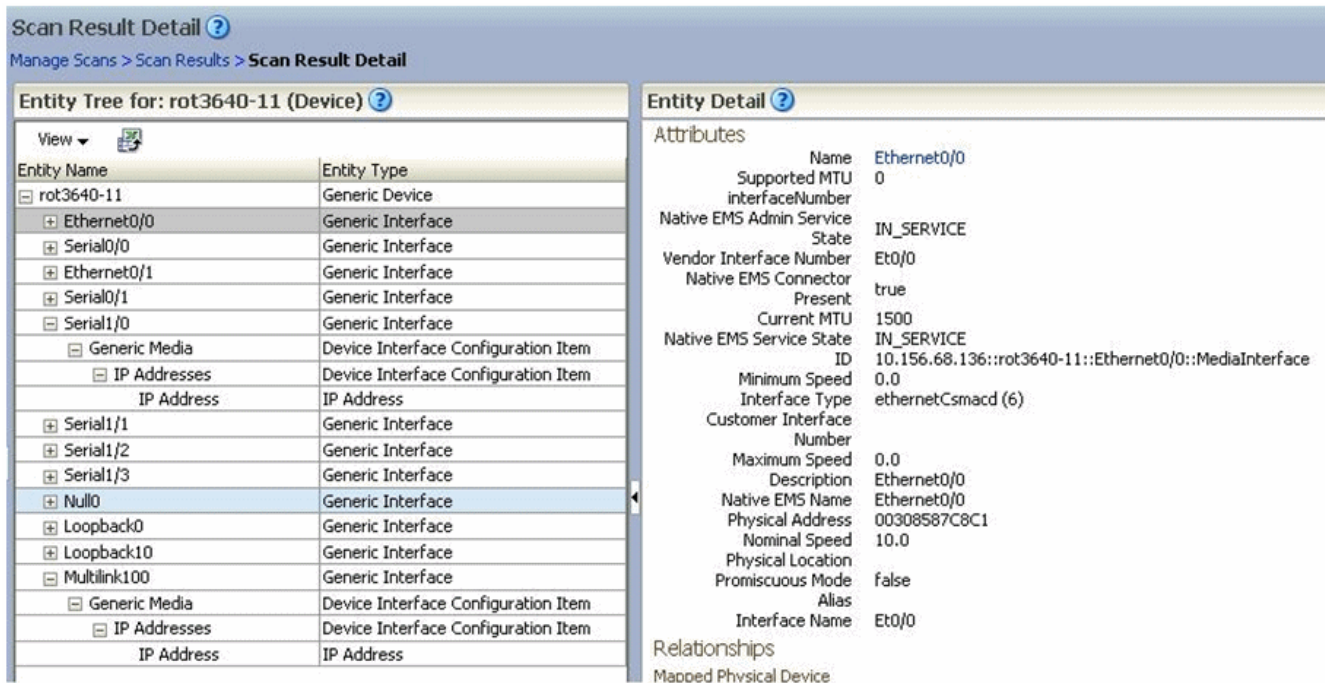
This chapter provides an overview of the Oracle Communications Network Integrity MIB-II SNMP cartridge.

About the MIB-II SNMP Cartridge

The MIB-II SNMP cartridge enables you to discover devices in your network using the SNMP protocol. The cartridge produces a logical device hierarchy that represents the discovered device and includes a logical device instance and child interface(s) or subInterface instance(s) (collectively called interfaces).

[Figure 1-1](#) shows a discovered logical device.

Figure 1-1 Sample Discovered Logical Device



About Cartridge Dependencies

This section provides information on dependencies that the MIB-II SNMP cartridge has on other cartridges.

Run-time Dependencies

For the MIB-II SNMP cartridge to work at run time, the Address_Handlers cartridge must be deployed to Network Integrity.

Design Studio Dependencies

The MIB-II SNMP cartridge has the following dependencies:

- Address_Handlers
- NetworkIntegritySDK
- ora_ni_uim_device
- ora_uim_model

Opening the Cartridge Files in Design Studio

To review and extend the Network Integrity MIB-II SNMP cartridge, you must first download the Oracle Communications Network Integrity MIB-II SNMP Cartridge software from the Oracle software delivery website:

<https://edelivery.oracle.com>

The software contains the MIB-II SNMP cartridge ZIP file, which has the following structure:

- \UIM_Cartridge_Projects\
- \Network_Integrity_Cartridge_Projects\
- \SNMP_MIBs\
- Address_Handlers-R7.jar
- MIB_II_SNMP_Cartridge-R7.jar

The \UIM_Cartridge_Projects\ora_ni_uim_device\ project contains the model project used by the MIB-II SNMP cartridge.

The \Network_Integrity_Cartridge_Projects\MIB_II_SNMP_Cartridge\ project contains the extensible Design Studio files.

Note

When importing MIB files into Design Studio, ensure that the MIB files do not have any filename extension. For example, rename the **FileMIB1.txt** file to **FileMIB1** before importing it into Design Studio.

See Design Studio Help and Network Integrity Developer's Guide for information about opening files in Design Studio. For guidelines and best practices for extending cartridges, see Network Integrity Concepts.

Building and Deploying the Cartridge

The MIB-II SNMP cartridge does not include any extendable cartridges.

See SCD Design Studio Modeling Network Integrity for information about building and deploying cartridges.

2

About the Cartridge Components

This chapter provides information about the components of the Oracle Communications Network Integrity MIB-II SNMP cartridge.

The MIB-II SNMP cartridge contains the following actions:

- [Discover MIB-II SNMP Action](#)

Discover MIB-II SNMP Action

The Discover MIB-II SNMP action discovers a device and creates its hierarchical model.

The MIB-II SNMP cartridge is designed to be used on a standalone basis to display the logical device hierarchy in Network Integrity. The SNMP cartridge provides no integration with other products but can be extended.

The Discover MIB-II SNMP action contains the following processors run in the following order:

1. [MIB-II Properties Initializer Processor](#)
2. [MIB-II SNMP Collector Processor](#)
3. [MIB-II SNMP Modeler Processor](#)

[Figure 2-1](#) illustrates the processor workflow of the Discover MIB-II SNMP action.

Figure 2-1 Discover MIB-II SNMP Action Processor Workflow



MIB-II Properties Initializer Processor

The MIB-II Properties Initializer processor produces the following data sets:

- `snmpVendorNameMap`: Contains a snapshot of industry enterprise numbers to help identify devices in the network.
- `snmpIfTypeMap`: Contains a snapshot of ifTypes to help identify interface types in the network.

[Table 2-1](#) shows a fragment of each data set output from the MIB-II Properties Initializer.

Table 2-1 MIB-II Properties Initializer Fragment

Sample <code>snmpIfTypeMap</code>	Sample <code>snmpVendorNameMap</code>
1: other (1)	0 = Reserved
2: regular1822 (2)	1 = NxNetworks
3: hdh1822 (3)	2 = IBM
4: ddX25 (4)	3 = Carnegie Mellon
5: rfc877x25 (5)	4 = UNIX
6: ethernetCsmacd (6)	5 = ACC
7: iso88023Csmacd (7)	6 = TWG
8: iso88024TokenBus (8)	7 = CAYMAN
9: iso88025TokenRing (9)	8 = PSI
10: iso88026Man (10)	9 = ciscoSystems
251: vdsl2 (251)	34730 = FRANCILIENNE D'INGENIERIE ET DE SERVICES INFORMATIQUES SAS

The content of these files may change from time to time, so they are maintained as part of cartridge revisions. SDK extensions to this cartridge can update the content of the property files. See "[About Design Studio Extensions](#)" for more information.

MIB-II SNMP Collector Processor

The MIB-II SNMP Collector processor collects SNMP variables from a device. See "[About Poll Lists](#)" for more information.

MIB-II SNMP Modeler Processor

The MIB-II SNMP Modeler processor models the data that is collected by the MIB-II SNMP Collector processor. Modeling includes building the hierarchical relationship of logical device and child interfaces.

3

About Poll Lists

This chapter provides a poll list for the MIB-II SNMP Collector processor that collects SNMP variables from a device.

About the MIB-II SNMP Collector Poll List

The following list shows the MIB-II SNMP Collector poll list:

- MIB-II:
 - * RFC1213-MIB.mgmt.mib-2.system.sysObjectID
 - * RFC1213-MIB.mgmt.mib-2.system.sysDescr
 - * RFC1213-MIB.mgmt.mib-2.system.sysName
 - * RFC1213-MIB.mgmt.mib-2.system.sysLocation
 - * RFC1213-MIB.mgmt.mib-2.interfaces.ifNumber
 - * RFC1213-MIB.mgmt.mib-2.interfaces.ifTable.ifEntry.ifIndex
 - * RFC1213-MIB.mgmt.mib-2.interfaces.ifTable.ifEntry.ifDescr
 - * RFC1213-MIB.mgmt.mib-2.interfaces.ifTable.ifEntry.ifMtu
 - * RFC1213-MIB.mgmt.mib-2.interfaces.ifTable.ifEntry.ifSpeed
 - * RFC1213-MIB.mgmt.mib-2.interfaces.ifTable.ifEntry.ifAdminStatus
 - * RFC1213-MIB.mgmt.mib-2.interfaces.ifTable.ifEntry.ifPhysAddress
 - * RFC1213-MIB.mgmt.mib-2.interfaces.ifTable.ifEntry.ifLastChange
 - * RFC1213-MIB.mgmt.mib-2.ip.ipAddrTable.ipAddrEntry.ipAdEntAddr
 - * RFC1213-MIB.mgmt.mib-2.ip.ipAddrTable.ipAddrEntry.ipAdEntIfIndex
 - * RFC1213-MIB.mgmt.mib-2.ip.ipAddrTable.ipAddrEntry.ipAdEntNetMask
 - * RFC1213-MIB.mgmt.mib-2.ip.ipAddrTable.ipAddrEntry.ipAdEntBroadcastAddr

About the IF MIB Poll List

The following list shows the IF MIB poll list:

- IF MIB:
 - IF-MIB.mgmt.mib-2.ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName
 - IF-MIB.mgmt.mib-2.ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifHighSpeed
 - IF-MIB.mgmt.mib-2.ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifPromiscuousMode
 - IF-MIB.mgmt.mib-2.ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifConnectorPresent
 - IF-MIB.mgmt.mib-2.ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifAlias
 - IF-MIB.mgmt.mib-2.ifMIB.ifMIBObjects.ifStackTable.ifStackEntry.ifStackStatus

- IF-MIB.mgmt.mib-2.interfaces.ifTable.ifEntry.ifType
- IF-MIB.mgmt.mib-2.interfaces.ifTable.ifEntry.ifOperStatus

About the IP MIB Poll List

The following list shows the IP MIB poll list:

- IP MIB:
 - IP-MIB.mgmt.mib-2.ip.ipAddressTable.ipAddressEntry.ipAddressPrefix
 - IP-MIB.mgmt.mib-2.ip.ipAddressTable.ipAddressEntry.ipAddressType
 - IP-MIB.mgmt.mib-2.ip.ipAddressTable.ipAddressEntry.ipAddressIfIndex

4

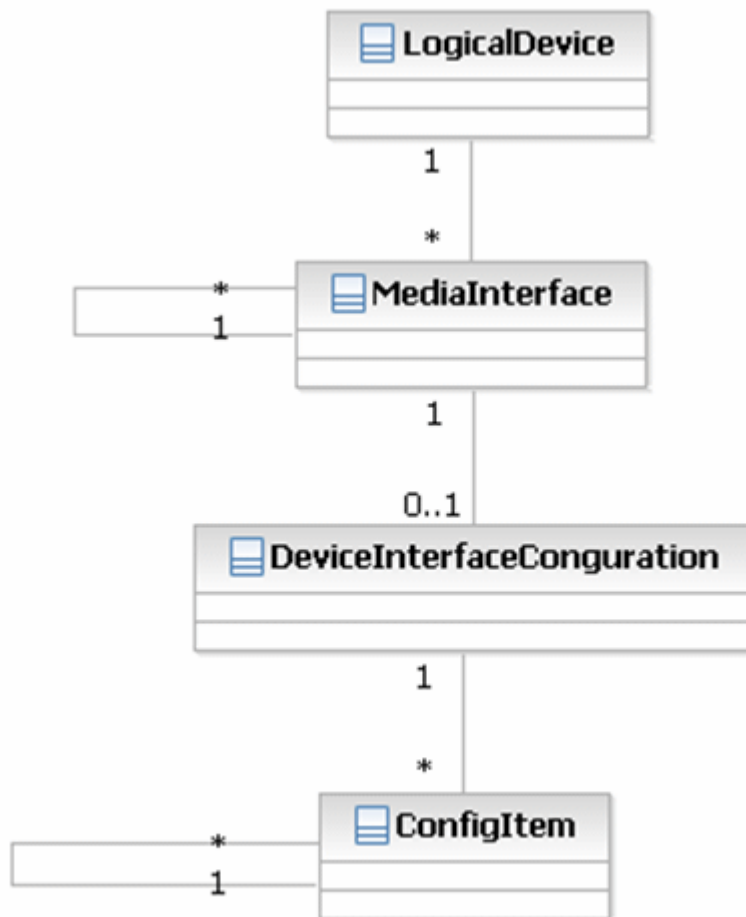
About Cartridge Modeling

This chapter provides information on modeling the Oracle Communications Network Integrity MIB-II SNMP cartridge.

MIB-II SNMP Cartridge UML Representation

[Figure 4-1](#) displays a unified modeling language (UML) diagram depicting the object relationship being rendered.

Figure 4-1 MIB-II SNMP Cartridge UML Representation



Hierarchy Mapping

The logical device object is established and seeded with data sourced by RFC1213-MIB.mgmt.mib-2.system tables.

As a device is discovered, the media interface encapsulates common information about that interface. The device interface configuration captures the media type information that decorates the interface with media-specific parameters that define its behavior (that is; Generic, ATM, Frame Relay or VLAN). This cartridge supports only Generic.

Interfaces are established and seeded with data sourced by:

- RFC1213-MIB.mgmt.mib-2.system
- RFC1213-MIB.mgmt.mib-2.interfaces.ifTable
- IF-MIB.mgmt.mib-2.ifMIB.ifMIBObjects.ifXTable.ifXEntry
- IP-MIB.mgmt.mib-2.ip.ipAddressTable.ipAddressEntry

The interface hierarchy is established by:

- F-MIB.mgmt.mib-2.ifMIB.ifMIBObjects.ifStackTable.ifStackEntry.ifStackStatus

Device interface configuration is established and seeded with data sourced by:

- RFC1213-MIB.mgmt.mib-2.ip.ipAddrTable.ipAddrEntry
- IP-MIB.mgmt.mib-2.ip.ipAddressTable.ipAddressEntry

About Oracle Communications Information Model

The logical device and media interface objects are Oracle Communications Information Model 1.0 compliant for static fields. The dynamic fields (sometimes referred to as characteristics) are considered application-specific.

The device interface configuration is considered a mechanism to customize the media interface to show application specific data. See [Figure 4-1](#).

This cartridge supports only generic media. Other cartridges support other configurations:

- AtmMedia
- FrameRelayMedia
- VLAN Membership

For a listing of the Information Model fields, see "[Field Mappings](#)".

Specifications

This section lists the specifications included in the Network Integrity MIB-II SNMP cartridge.

Logical Device: deviceGeneric

This specification models a generic Logical Device entity.

This specification is used to model a logical device discovered by Network Integrity.

[Table 4-1](#) shows the characteristics applied to the generic device specification.

Table 4-1 Characteristics Applied to the deviceGeneric Specification

Characteristics	Field Type	Field Content	Intended Usage/ Notes
mgmtIpAddress	String	Text	Management IP address of discovered device
nativeEmsAdminServiceState	Drop Down box	Enum with the following values: <ul style="list-style-type: none"> UNKNOWN IN_SERVICE OUT_OF_SERVICE TESTING IN_MAINTENANCE 	Device administered state
nativeEmsServiceState	Drop Down box	Enum with the following values: <ul style="list-style-type: none"> UNKNOWN IN_SERVICE OUT_OF_SERVICE TESTING IN_MAINTENANCE 	Device actual state
nativeEmsName	String	Text	Device actual name
sysObjectId	String	Text	The vendor's authoritative identification of the network management subsystem contained in the entity.

Device Interface: interfaceGeneric

This specification models a generic interface entity.

This specification is used to model an interface discovered by Network Integrity. It can also be used to model new devices and details discovered for routing protocols such as LLDP, Static routing, BGP and OSPF.

[Table 4-2](#) shows the characteristics applied to the interfaceGeneric specification.

Table 4-2 Characteristics Applied to the interfaceGeneric Specification

Characteristics	Field Type	Field Content	Intended Usage/ Notes
nativeEmsName	String	Text	Alternate name this interface could be known by
nativeEmsAdminServiceState	String	Enum with the following values: <ul style="list-style-type: none"> UNKNOWN IN_SERVICE OUT_OF_SERVICE TESTING IN_MAINTENANCE 	Interface administered state
nativeEmsServiceState	Drop Down box	Enum with the following values: <ul style="list-style-type: none"> UNKNOWN IN_SERVICE OUT_OF_SERVICE TESTING IN_MAINTENANCE 	Interface actual state

Table 4-2 (Cont.) Characteristics Applied to the interfaceGeneric Specification

Characteristics	Field Type	Field Content	Intended Usage/ Notes
ifType	Drop Down box	Text	IANA iftype
mtuSupported	String	Text (numeric)	Max allowed MTU
mtuCurrent	String	Text (numeric)	Current Value of MTU
physicalAddress	String	Text	Interface's address at its protocol sub-layer
physicalLocation	String	Text	Location of interface
minSpeed	String	Text (numeric)	Minimum speed of interface
maxSpeed	String	Text (numeric)	Max speed of interface
nominalSpeed	String	Text (numeric)	Nominal speed of interface
ifAlias	String	Text	Alternate name this interface could be known by
ifName	String	Text	Alternate name this interface could be known by
ifSpeed	String	Text (numeric)	ifSpeed of interface
rateCode	String	Text	rateCode of interface based on ifSpeed value
highSpeed	String	Text (numeric)	highSpeed of interface

The following table provides information of characteristics included in the interfaceGeneric Specification for BGP details.

Table 4-3 Characteristics of BGP Details Within interfaceGeneric Specification

Char	Field Type	Field Content	Description
bgpPeerAdminStatus	String	Text	Administrative state of the BGP peer
bgpPeerLastError	String	Text	Last error recorded with the BGP peer
bgpPeerLocalAddr	String	Text	Local IP address used in the BGP session
bgpPeerRemoteAddr	String	Text	Remote IP address of the BGP peer

The following table provides information of characteristics included in the interfaceGeneric specification for static routing details.

Table 4-4 Characteristics of Static Routing Details Included in interfaceGeneric Specification

Char	Field Type	Field Content	Description
ipCidrRouteDest	String	Text	Destination address for a route
ipCidrRouteMask	String	Text	Subnet mask associated with a route

Table 4-4 (Cont.) Characteristics of Static Routing Details Included in interfaceGeneric Specification

Char	Field Type	Field Content	Description
ipCidrRouteNextHop	String	Text	Next-hop IP address for a route

The following table provides information of characteristics included in the interfaceGeneric specification for LLDP details.

Table 4-5 Characteristics of LLDP Details Within interfaceGeneric Specification

Char	Field Type	Field Content	Description
A-End Interface Description	String	Text	Description of the local LLDP port
A-End Interface ID	String	Text	Identifier of the local LLDP port
A-End Interface Subtype	String	Text	Subtype of the local port ID
A-End Device Description	String	Text	LLDP Local System Description
A-End Device Name	String	Text	LLDP Local System Name
Z-End Interface Description	String	Text	Description of the remote LLDP port
Z-End Interface Id	String	Text	Identifier of the remote port
Z-End Interface Subtype	String	Text	Subtype of the remote port ID
Z-End Device Description	String	Text	Description of the remote system
Z-End Device Name	String	Text	Name of the remote system
A-End Interface Ip Address	String	Text	Local (A-side) endpoint IP from LLDP data
Z-End Interface Ip Address	String	Text	Remote (Z-side) endpoint IP from LLDP data

The following table provides information of characteristics included in the interfaceGeneric specification for MPLS VPN details.

Table 4-6 Characteristics of MPLS VPN Details Within interfaceGeneric Specification

Char	Field Type	Field Content	Description
mplsVpnVrfDescription	String	Text	Description of the VPN Routing and Forwarding (VRF)
mplsVpnVrfRouteDistinguisher	String	Text	Route distinguisher for identifying unique VPNs

Table 4-6 (Cont.) Characteristics of MPLS VPN Details Within interfaceGeneric Specification

Char	Field Type	Field Content	Description
rtBothValue	String	Text	Route targets for both import and export
rtExportValue	String	Text	Route target(s) used for route export
rtImportValue	String	Text	Route target(s) used for route import

The following table provides information of characteristics included in the interfaceGeneric specification for OSPF details.

Table 4-7 Characteristics of OSPF Details Within interfaceGeneric Specification

Char	Field Type	Field Content	Description
ospflfAdminState	String	Text	Administrative state of the OSPF interface
ospflfAreaId	String	Text	OSPF area identifier for the interface
ospflfIpAddress	String	Text	IP address associated with the OSPF interface
ospflfLsaCount	String	Text	Count of LSAs seen or generated on interface
ospflfState	String	Text	Operational state of OSPF (e.g., Down, DR, BDR)
ospflfType	String	Text	Interface type (broadcast, point-to-point, etc.)
ospfNbrIpAddr	String	Text	IP address of the adjacent OSPF neighbor

Device Interface: I2vlan

This specification models an interface entity of ifType "I2vlan" discovered by Network Integrity.

Interfaces with this specification are not reconciled to UIM. The I2vlan interfaces are Ethernet flow points and created as flow interfaces when a link is discovered using the Assimilate IP Links scan available in the IP Assimilation cartridge.

[Table 4-8](#) shows the characteristics applied to the I2vlan specification.

Table 4-8 Characteristics Applied to the I2vlan Specification

Characteristics	Field Type	Field Content	Intended Usage
nativeEmsName	String	Text	Alternate name this interface could be known by

Table 4-8 (Cont.) Characteristics Applied to the I2vlan Specification

Characteristics	Field Type	Field Content	Intended Usage
nativeEmsAdminServiceState	String	Enum with the following values: <ul style="list-style-type: none"> UNKNOWN IN_SERVICE OUT_OF_SERVICE TESTING IN_MAINTENANCE 	Interface administered state
nativeEmsServiceState	Drop Down list	Enum with the following values: <ul style="list-style-type: none"> UNKNOWN IN_SERVICE OUT_OF_SERVICE TESTING 	Interface actual state
ifType	Drop-down list	Text	IANA iftype
mtuSupported	String	Text (numeric)	Maximum allowed MTU
mtuCurrent	String	Text (numeric)	Current Value of MTU
physicalAddress	String	Text	Interface's address at its protocol sub-layer
physicalLocation	String	Text	Location of interface
minSpeed	String	Text (numeric)	Minimum speed of interface
maxSpeed	String	Text (numeric)	Maximum speed of interface
nominalSpeed	String	Text (numeric)	Nominal speed of interface
ifAlias	String	Text	Alternate name this interface could be known by
ifName	String	Text	Alternate name this interface could be known by
ifSpeed	String	Text (numeric)	ifSpeed of interface
rateCode	String	Text	rateCode of interface based on ifSpeed value
highSpeed	String	Text (numeric)	highSpeed value of interface

Other Characteristics

This MIB-II SNMP Cartridge additionally models the following characteristics. These characteristics are used by the Network Integrity Generic SNMP cartridge which extends the sample technology pack.

[Table 4-9](#) shows the characteristics used by Network Integrity Generic SNMP.

Table 4-9 Characteristics used by ora_ni_uim_device

Characteristics	Field Type	Field Content	Intended Usage/ Notes
discoveredModelNumber	String	Text	All Physical Equipment
discoveredPartNumber	String	Text	All Physical Equipment
discoveredVendorName	String	Text	All Physical Equipment

Table 4-9 (Cont.) Characteristics used by ora_ni_uim_device

Characteristics	Field Type	Field Content	Intended Usage/ Notes
hardwareRev	String	Text	All Physical Equipment
softwareRev	String	Text	All Physical Equipment
modelName	String	Text	All Physical Equipment

Specification Lineage

```

deviceGeneric
  [0.100000] interfaceGeneric
    [0.100000] interfaceGeneric

```

Sample Logical Device Hierarchy

[Figure 4-2](#) shows a logical device hierarchy.

Figure 4-2 Sample Logical Device Hierarchy

The screenshot displays the 'Scan Result Detail' window for 'router0 (Device)'. The left pane shows the 'Entity Tree' with a hierarchy: router0 (deviceGeneric) -> Ethernet0/0 (interfaceGeneric) -> Generic Media (GenericMedia) -> IP Addresses (IpAddresses) -> IP Address (IpAddress). The right pane shows the 'Entity Detail' for 'Ethernet0/2', listing various attributes such as Name, Z-End Interface Subtype, Z-End Interface Ip Address, Z-End Interface Id, Z-End Interface Description, Z-End Device Name, Z-End Device Description, Vendor Interface Number, Supported MTU, RI Import Value, RI Export Value, RI Both Value, Rate Code, Physical Location, Physical Address, Owner, Ospl Neighbour Ip Address, Ospl If Type, Ospl If State, Ospl If Lsa Count, Ospl If Ip Address, Ospl If Area Id, Ospl If Admin State, Nominal Speed, Native EMS Service State, Native EMS Name, Native EMS Connector Present, Native EMS Admin Service State, Mpls Vpn Vrf Route Distinguisher, Mpls Vpn Vrf Description, Minimum Speed, Maximum Speed, Static Route Next Hop IP Address, Static Route IP Mask, Static Route Destination IP Address, Interface Type, Interface Number, If Speed, Promiscuous Mode, Interface Name, Alias, ID, High Speed, Description, Customer Interface Number, Current MTU, Bgp Peer Remote Address, Bgp Peer Local Address, Bgp Peer Last Error, Bgp Peer Admin Status, A-End Interface Subtype, A-End Interface Ip Address, A-End Interface Id, A-End Interface Description, A-End Device Name, and A-End Device Description.

Field Mappings

This section provide information about field mappings used in the cartridge:

- *Text*: Implies Text [255].
- *static*: The Information Model 1.0 defines this field to be static on the entity specification. The specification provides getters/setters for this field.

- *dynamic*: This is a dynamic field where the entity specification treats the field as a name/value pair. The specification does not provide getter/setters but generically has a get/setCharacteristics method holding a HashSet of entries.

The following tables describe field mappings used in the cartridge:

- [Table 4-10](#)
- [Table 4-11](#)
- [Table 4-12](#)
- [Table 4-13](#)
- [Table 4-14](#)

Logical Device Mapping

[Table 4-10](#) shows the logical device mapping for the MIB-II SNMP cartridge.

Table 4-10 Logical Device Mapping

LogicalDevice	Information Model Support	MIB Object	Field Type	Notes
Id	static	N/A	Text	Programmatically generated as MgmtIpAddress::sysName::"LogicalDevice"
Name	static	sysName	Text	N/A
Description	static	sysDescr	Text	N/A
Specification	static	N/A	N/A	Programmatically applies specification "deviceGeneric"
nativeEmsAdminServiceState	static	N/A	Enum: <ul style="list-style-type: none"> UNKNOWN OUT_OF_SERVICE TESTING IN_SERVICE IN_MAINTENANCE 	No source
nativeEmsServiceState	static	N/A	Enum: <ul style="list-style-type: none"> UNKNOWN OUT_OF_SERVICE TESTING IN_SERVICE IN_MAINTENANCE 	No source
nativeEmsName	static	sysName	Text	N/A
mgmtIpAddress	dynamic	N/A	Text	discoveryAddress
sysObjectId	dynamic	sysObjectId	Text	Support legacy systems

Media Interface Mapping

[Table 4-11](#) shows the media interface mapping for the MIB-II SNMP cartridge.

Table 4-11 Media Interface Mapping

Media Interface	Information Model Support	MIB Object	Field Type	Notes
Id	static	N/A	Text	Programmatically generated asMgmtIPAddress:: sysName::ifDesc::"MediaInterface"
Name	static	ifDescr	Text	N/A
Description	static	ifDescr	Text	N/A
Specification	static	N/A	N/A	Programmatically applies specification "interfaceGeneric"
interfaceNumber	static	N/A	Text	No source
customerInterfaceNumber	static	N/A	Text	No source
vendorInterfaceNumber	static	ifName	Text	N/A
nativeEmsName	static	ifDesc	Text	Field must be unique. ifDescr guarantees uniqueness in device.
nativeEmsAdminServiceState	static	ifAdminStatus	Enum: <ul style="list-style-type: none"> UNKNOWN OUT_OF_SERVICE TESTING IN_SERVICE IN_MAINTENANCE 	Mapped. See " Mapping Table " for more information.
nativeEmsServiceState	static	ifOperStatus	Enum: <ul style="list-style-type: none"> UNKNOWN OUT_OF_SERVICE TESTING IN_SERVICE IN_MAINTENANCE 	Mapped. See " Mapping Table " for more information.
ifType	dynamic	ifType	Text	Mapped from IANA MIB using properties file "snmplIfTypeMap". Look up returning null results in value of "n"
mtuSupported	static	N/A	Float	No source. Defaults to 0.0
mtuCurrent	static	ifMtu	Float	N/A
physicalAddress	static	ifPhysAddress	Text	N/A
physicalLocation	static	sysLocation	Text	N/A
minSpeed	static	N/A	Float	No source. Defaults to 0.0
maxSpeed	static	N/A	Float	No source. Defaults to 0.0

Table 4-11 (Cont.) Media Interface Mapping

Media Interface	Information Model Support	MIB Object	Field Type	Notes
nominalSpeed	static	ifSpeed ifHighSpeed	Float	ifHighSpeed overrides ifSpeed when ifHighSpeed is available
ifAlias	dynamic	ifAlias	Text	N/A
ifName	dynamic	ifName	Text	Support legacy systems

Device Interface Configuration Mapping (IPv4)

[Table 4-12](#) shows the device interface configuration mapping for Internet Protocol, Version 4, for the MIB-II SNMP cartridge.

Table 4-12 Device Interface Configuration Mapping (IPv4)

Generic Media	Information Model Support	MIB Object	Field Type	Notes
ipaddress	dynamic	ipAdEntAddr	Text	N/A
prefix	dynamic	ipAdEntNetMask	Text	N/A
ipVersion	dynamic	ipAddressType	Enum {IPV4 IPV6}	Programmatically set to IPV4
Specification	static	N/A	N/A	Programmatically applies specification "GenericMedia"

Device Interface Configuration (IPv6)

[Table 4-13](#) shows the device interface configuration for the Internet Protocol, Version 6 for the MIB-II SNMP cartridge.

Table 4-13 Device Interface Configuration Mapping (IPv6)

Generic Media	Information Model Support	MIB Object	Field Type	Notes
ipAddress	dynamic	ipAddressIfIndex	Text	For IP v6 addresses, the actual IP is derived from the index since ipAddressAddr is not accessible.
prefix	dynamic	ipAddressPrefix	Text	N/A
ipVersion	dynamic	ipAddressType	Enum {IPV4 IPV6}	Programmatically set to IPV6
Specification	static	N/A	N/A	Programmatically applies specification "GenericMedia"

Mapping Table

[Table 4-14](#) shows the mapping of ifOperStatus/ifAdminStatus to nativeEms(Admin)ServiceState.

Table 4-14 Mapping Table

ifOperStatus	ifAdminStatus	nativeEmsServiceState nativeEmsAdminState
1: up	1: up	IN_SERVICE
2: down	2: down	OUT_OF_SERVICE
3: testing	3: testing	TESTING
4: unknown	N/A	UNKNOWN
5: dormant	2: down	OUT_OF_SERVICE
6: notPresent	N/A	UNKNOWN
7: lowerLayerDown	2: down	OUT_OF_SERVICE
N/A	N/A	IN_MAINTENANCE

About Design Studio Construction

This chapter provides information on using Oracle Communications Design Studio to construct for the Oracle Communications Network Integrity MIB-II SNMP cartridge.

Model Collection

[Table 5-1](#) shows the model collection for the MIB-II SNMP cartridge.

Table 5-1 MIB-II Model Collection

Specification	Notes
deviceGeneric	Intended to represent any root object discovered on the network
interfaceGeneric	Intended to represent any interface discovered under deviceGeneric
GenericMedia	Intended to represent IpAddresses that are applied to a interfaceGeneric
IPAddresses	Container of IP addresses
IPAddress	Container of IP address details
I2vlan	Intended to represent any interface discovered under deviceGeneric

Specification Lineage

This section outlines the specification lineage.

```
deviceGeneric
  [0..*] interfaceGeneric
    [0..1] GenericMedia
      [0..1] IP Addresses
        [0..*] IpAddress

        IpAddress (characteristic)

                                P                                refix (characteristic)

        IpVersion (characteristic)
```

Discovery Action

[Table 5-2](#) shows the discovery action for the MIB-II SNMP cartridge.

Table 5-2 Discover MIB-II SNMP Action

Result Category	Address Handler	Scan Parameters	Model	Processors
Device	IPAddressHandler	<ul style="list-style-type: none">version, Enum {v1, v2c, v3}port, StringsnmpReadCommunity, StringsnmpTimeout, StringsnmpRetries, Stringusername, StringcontextName, StringauthProtocol, Enum {MD5, SHA}authPassword, StringprivacyProtocol Enum {DES}privacyPassword, String	MIB-II Model	<ul style="list-style-type: none">MIB-II Properties InitializerMIB-II SNMP CollectorMIB-II SNMP Modeler

[Figure 5-1](#) depicts the Discover MIB-II SNMP action chain.

Figure 5-1 Discover MIB-II SNMP Action Chain



Discovery Processors

[Table 5-3](#) shows the discovery processors for the MIB-II SNMP cartridge.

Table 5-3 Discovery Processors

Processor Name	Variable
MIB-II Properties Initializer	<div>Input: N/A</div> <div>Output:</div> <ul style="list-style-type: none">snmplfTypeMap Property map containing listing of ifTypes to string name.snmpVendorNameMap Property map containing listing of sysObjectId suffixes to vendorName.

Table 5-3 (Cont.) Discovery Processors

Processor Name	Variable
MIB-II SNMP Collector	Input: N/A Output: <ul style="list-style-type: none"> mibiisnmpCollectorResponseDocument (implicit) Polled SNMP data, see "About Poll Lists".
MIB-II SNMP Modeler	Input: mibiisnmpCollectorResponseDocument, snmpIfTypeMap Output: <ul style="list-style-type: none"> deviceInterfaceMap A map that contains interfaces with IfIndex as key. logicalDevice This is the logical device that was created in the MIB-II Modeler.

The MIB-II SNMP Collector polling the SNMP variables implicitly outputs this data in mibiisnmpCollectorResponseDocument. This output field is not seen explicitly in Design Studio. This document is available to the MIB-II SNMP Modeler processor.

Property File Sample

[Table 5-4](#) shows a sample of the property files being loaded by the MIB-II Properties Initializer.

Table 5-4 Sample Property Files Loaded by MIB-II Properties Initializer

Sample snmpIfTypeMap	Sample snmpVendorNameMap
1: other (1)	0 = Reserved
2: regular1822 (2)	1 = NxNetworks
3: hdh1822 (3)	2 = IBM
4: ddnX25 (4)	3 = Carnegie Mellon
5: rfc877x25 (5)	4 = UNIX
6: ethernetCsmacd (6)	5 = ACC
7: iso88023Csmacd (7)	6 = TWG
8: iso88024TokenBus (8)	7 = CAYMAN
9: iso88025TokenRing (9)	8 = PSI
10: iso88026Man (10)	9 = ciscoSystems
251: vds12 (251)	34730 = FRANCILIENNE D'INGENIERIE ET DE SERVICES INFORMATIQUES SAS

Note

The **snmpVendorNameMap** is not used in this cartridge but is initialized for extension cartridges to make use of this map.

6

About Design Studio Extensions

This chapter provides information about Oracle Communications Design Studio extensions to the Oracle Communications Network Integrity MIB-II SNMP cartridge.

Updating Property Files

To update the property files (if new interface types or new vendors are introduced), the user can extend **Discover MIB-II SNMP** and add a new **Post** processor, which takes as input, `snmpIfTypeMap` or `snmpVendorNameMap`.

Source code can then update the map either programmatically or by loading a new property file and then (re)write into the map. For further information on extensibility, see "Using Design Studio to Extend Network Integrity" in *Network Integrity Developer's Guide*.