

---

# BSR 64000 SNMP MIB Reference Guide



**MOTOROLA**  
*intelligence everywhere™*

---

---

## Notice

Copyright © 2002  
Motorola, Inc.  
All rights reserved

No part of this publication may be reproduced in any form or by any means or used to make any derivative work (such as translation, transformation or adaptation) without written permission from Motorola, Inc.

Motorola reserves the right to revise this publication and to make changes in content from time to time without obligation on the part of Motorola to provide notification of such revision or change. Motorola provides this guide without warranty of any kind, either implied or expressed, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Motorola may make improvements or changes in the product(s) described in this manual at any time.

Motorola, the stylized M logo, and Intelligence Everywhere are registered trademarks of Motorola, Inc. Broadband Services Router, BSR, BSR 64000, RiverDelta, SmartFlow are trademarks of RiverDelta Networks, Inc. All other trademarks and registered trademarks are the property of their respective owners.

---

# Contents

## **Preface**

Scope	v
Audience	v
Documentation Set	v
Conventions	vi
Notes, Cautions, Warnings	vii
Contacting Support	viii

## **1 Simple Network Management Protocol Overview**

Introduction	1-1
About SNMP MIBs	1-1
SNMP Commands	1-2
How a MIB is Organized	1-2
Defining MIB Objects	1-3

## **2 Standard and Proprietary MIB Support**

Introduction	9
MIB Support	9
MIB-II	10

Transmission	12
SNMP	12
Experimental	13
BSR 64000 Proprietary MIBs	13

### **3 Using a MIB Browser to Access MIB Variables**

Introduction	3-1
About MIB Browsers	3-1
Contacting the Device	3-2
Browsing the MIB Tree	3-2
SNMP Get/Get Next	3-4
SNMP Set	3-5
Compiling and Importing New SNMP MIBs	3-5

### **4 Sample Tasks**

Introduction	4-1
Accessing MIB Variable Information	4-1
Sample Task	4-1
Changing MIB Variables	4-5
Sample Task	4-5

---

# Figures

Figure 2-1	BSR 64000 Proprietary MIB Structure	14
Figure 3-1	The Top Level of the MIB Tree	3-3
Figure 3-2	The Path to the Internet Branch of the MIB Tree	3-3
Figure 3-3	The MIB Tree Expanded to Reveal an Object	3-4
Figure 4-1	The mib-2 Hierarchy	4-2
Figure 4-2	The mib-2 and system Folders	4-3
Figure 4-3	Selecting the sysUpTime Object	4-4
Figure 4-4	The rdn MIB Hierarchy	4-6
Figure 4-5	The rdn and syslog Folders	4-7
Figure 4-6	Selecting the rdnSyslogClear Object	4-8



---

# Tables

Table 1-1	SNMP Commands	1-2
Table 2-1	Supported MIB-II Groups	10
Table 2-2	Supported Transmission Groups	12
Table 2-3	Supported SNMP Groups	12
Table 2-4	Supported Experimental MIBs	13
Table 2-5	Proprietary MIB Support	13



---

# Preface

## Scope

This document provides descriptions of the standard SNMP and proprietary MIBs supported by the Motorola™ Broadband Services Router™ 64000 (BSR 64000™). It also provides a tutorial on how to “walk” the MIBs using an SNMP MIB browser and provides example tasks that can be performed using the SNMP or proprietary MIBs.

## Audience

This document is for use by those persons who will configure and manage the BSR 64000™ product. Only trained service personnel should configure and manage the BSR 64000.

## Documentation Set

The following documents comprise the BSR 64000 documentation set:

- *BSR 64000 Command Reference Guide*  
This document contains the Command Line Interface (CLI) commands for managing, configuring, and maintaining the BSR 64000.
- *BSR 64000 Configuration and Management Guide*  
This document provides the instructions and procedures for configuring and managing the BSR 64000.
- *BSR 64000 Installation Guide*  
This document describes how to install the BSR 64000 product.

- *BSR 64000 Release Notes*

These documents provide information about features not described or incorrectly documented in the main documentation set; known problems and anomalies; product limitations; and problem resolutions.

- *BSR 64000 SNMP MIB Reference Guide*

This document describes the Simple Network Management Protocol (SNMP) MIBs; provides information that describes standard and proprietary MIB support; describes how to walk the MIBs and how to compile and load the SNMP MIBs. It also provides task examples.

- *BSR Troubleshooting Guide*

This document provides instructions and procedures for troubleshooting the BSR product.

- *BSR 64000 Quick Start Guide*

This document provides basic tasks used to get the BSR 64000™ out of the box, running, connected to the network, and operational.

## Conventions

This document uses the conventions in the following table:

Convention	Example	Explanation
angle brackets < >	<b>ping</b> <i>&lt;ip-address&gt;</i> <b>ping 54.89.145.71</b>	Arguments in italic and enclosed by angle brackets must be replaced by the text the argument represents. In the example, <b>54.89.145.71</b> replaces <i>&lt;ip-address&gt;</i> . When entering the argument, do not type the angle brackets.
bar brackets [ ]	<b>disable</b> [ <i>level</i> ]	Bar brackets enclose optional arguments. The example indicates you can use the <b>disable</b> command with or without specifying a <i>level</i> . Some commands accept more than one optional argument. When entering the argument, do not type the bar brackets.

Convention	Example	Explanation
<b>bold text</b>	<b>cable relay-agent-option</b>	Boldface text must be typed exactly as it appears.
brace brackets {}	<b>page {on   off}</b>	Brace brackets enclose required text. The example indicates you must enter either <b>on</b> or <b>off</b> after <b>page</b> . The system accepts the command with only one of the parameters. When entering the text, do not type the brace brackets.
<i>italic text</i>	<b>boot system</b> <filename>	Italic type indicates variables for which you supply values in command syntax descriptions. It also indicates file names, directory names, document titles, or emphasized text.
screen display	Wed May 6 17:01:03 2000	This font indicates system output.
vertical bar	<b>page {on   off}</b>	A vertical bar separates the choices when a parameter is required. The example indicates you can enter either command: <b>page on</b> or <b>page off</b> When entering the parameter, do not type the vertical bar or the brace brackets.

## Notes, Cautions, Warnings

The following icons and associated text may appear in this document.



**Note:** A note contains tips, suggestions, and other helpful information, such as references to material not contained in the document, that can help you complete a task or understand the subject matter.



**Caution:** The exclamation point, within an equilateral triangle, is intended to alert the user to the presence of important installation, servicing, and operating instructions in the documents accompanying the equipment.



**Warning:** This symbol indicates that dangerous voltages levels are present within the equipment. These voltages are not insulated and may be of sufficient strength to cause serious bodily injury when touched. The symbol may also appear on schematics.

## Contacting Support

Use the following information to contact Support:

U.S.	1-888-944-HELP 1-888-944-4357
International	+215-323-0044
WWW	<a href="http://www.gi.com/BUSAREA/CUSACC/websupport.html">http://www.gi.com/BUSAREA/CUSACC/websupport.html</a>
Email	<a href="mailto:cmtssupport@motorola.com">cmtssupport@motorola.com</a>

---

# 1

## **Simple Network Management Protocol Overview**

---



## Introduction

This chapter provides a brief description of the Simple Network Management Protocol (SNMP). This includes the following sections:

- An overview of SNMP MIBs
- SNMP Commands
- The organization of a MIB
- How MIB objects are defined

## About SNMP MIBs

The Structure of Management Information (SMI), defined in RFC 1155, is a framework that describes what kinds of information can be manipulated using SNMP. Using SMI, objects are placed as nodes in an object tree. The object tree groups logically related objects into subtrees. Each of these subtrees is called a MIB (Management Information Base).

A MIB is a database maintained by the device that stores all of its known management information. Each individual element of information in the MIB is called an Object. A MIB is analogous to an information warehouse with specific floors, aisles and racks stocked with inventory. Request for Comments (RFC) MIBs are written by the Internet Engineering Task Force (IETF) and are available for public use. Other MIBs have been developed by private organizations for private use.

An information database isolated at the device is useless if there is no means of communicating that information between the device and remote management. This communication is accomplished by means of an SNMP *agent*, which understands how to retrieve information from and store information to a MIB. SNMP management follows this model:

- MIBs located on the SNMP network management station describe the information that is relayed from the SNMP agents.
- The SNMP network management station or application can send queries to each SNMP agent.
- Each SNMP agent collects information about its device and provides that information to the SNMP network management station. The agent process acts as a server in a typical client-server model.

## SNMP Commands

The SNMP agent software on the BSR 64000 supports the SNMP commands **Get**, **GetNext**, and **Set** as defined in RFC 1157. [Table 1-1](#) describes these commands.

**Table 1-1 SNMP Commands**

Command	Description
Get	Retrieves the value of a specific object from one of the supported MIBs.
GetNext	Traverses the MIB tree to retrieve the next object's management information.
Set	Modifies the values of MIB objects. The BSR 64000 proprietary MIBs and several objects in the standard MIBs allow you to configure the BSR 64000 from an SNMP management station on the network.

## How a MIB is Organized

Because networking devices made by a variety of manufacturers must all be able to communicate with one another, the Internet standards organization requires that each network device organize its management information according to a pre-defined *tree* format. This tree structure branches out from the Internet layer into several subtrees, with each subtree organized into *branches* (groups of related information) and *leaves* (the individual pieces of information, or objects). Among these subtrees is an Enterprises subtree, in which private vendors can apply to the Internet Assigned Numbers Authority for a branch in which to store management information (or objects) specific to their products.

Each layer of this tree is numerically encoded, so that each group and object is identified by a unique number known as an Object Identifier (OID). This identifier is a path to the information stored as the object's value, and provides the means by which the SNMP agent is able to locate the object in a device's MIB. An ASCII name is also assigned to each branch or OID, for convenience in identifying a management object. For example, the MIB II object *ipForwarding* is identified as follows:

**Numeric OID:** 1. 3. 6. 1. 2. 1. 4. 1

**ASCII string:** iso/org /dod /internet /mgmt /mib-2 /ip /ipForwarding

Some objects may occur multiple times for a single device. Objects of this type are called tabular objects and are organized in tables. Each occurrence of a tabular object is called an instance, and each instance also numerically encoded. For example, if you were querying a two-port bridge's interface table, there would be two instances (values) returned for each object in the table (one for each port), and each instance would have a unique instance value appended to the object's OID.

## Defining MIB Objects

MIB objects represent data that the device can retrieve or configuration information that can be modified.

- RFC 1155 (*Structure and Identification of Management Information for TCP/IP-based Internets*) describes the layout and encoding of exchanged data objects. The SMI uses the ISO standard ASN.1 (Abstract Syntax Notation One) to define a method for describing a hierarchical name space for managed information.
- RFC 1212 (*Concise MIB Definitions*) is an easier-to-read form used in most standard MIBs today. It defines the private enterprise MIB.

A MIB object is defined and identified by the following:

OBJECT_TYPE	The object's ASN.1 (Abstract Syntax Notation.1) defined name and type.
SYNTAX	How the data represented by this object is structured: Integer, Octet String, Object Identifier, Null, Sequence, Sequence of, IpAddress, NetworkAddress, Counter, Gauge, TimeTicks, Opaque, or some other user-defined data type.

ACCESS	The level of management access available for this specific object: read-only (instances of the object may be read, but not set), read-write (instances of the object may be read or set), write-only (instances of the object may be set but not read), or not-accessible (instances of the object may not be read or set). Note that this does not designate the level of access provided by the community name you used to contact the device, but the maximum level of access available for the object by definition.
:	Indicates whether a managed device is required to implement this object in its MIB. A status of mandatory indicates that the device must use this object (by convention, all MIBs adhering to the Internet-standard contain only mandatory objects); a status of optional indicates that the device may implement this object (by Internet-standard MIB convention, this is used to denote objects within an experimental MIB); and a status of obsoleted indicates that the device need no longer implement this object, as it is no longer meaningful in an Internet-standard MIB.
DESCRIPTION	A brief description of the management information provided by this object.
::=	The variable name of the object, in the form OBJECT TYPE OBJECT IDENTIFIER fragment.

The following is an example of the MIB object *ifNumber* which is part of the Interfaces group of RFC 1213 (*Management Information Base for Network Management of TCP/IP-based internets:MIB-II.*)

```
ifNumber OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "The number of network interfaces (regardless of
        their current state) present on this system."
    ::= { interfaces 1 }
```



---

# 2

## **Standard and Proprietary MIB Support**

---



## Introduction

The BSR 64000 supports SNMP V3, V2, and V1. This chapter provides information about standard MIB support and proprietary MIBs for the BSR 64000. This information is useful when using a MIB browser or other SNMP-management software.

- For standard MIBs, this chapter provides a list of the standard MIBs supported.
- For the BSR 64000 proprietary MIBs, this chapter provides MIB titles and MIB structure.

## MIB Support

The BSR 64000 supports the following standard MIBs.

[Table 2-1](#) provides a list of MIB-II groups, their corresponding RFCs, and a description of each group.

[Table 2-2](#) provides a list of Transmission groups, their corresponding RFCs, and a description of each group.

[Table 2-3](#) provides a list of SNMP groups, their corresponding RFCs, and a description of each group.

[Table 2-4](#) provides a list of Experimental groups, their corresponding RFCs, and a description of each group.

[Table 2-5](#) provides a list of the BSR 64000 proprietary MIBs

## MIB-II

**Table 2-1 Supported MIB-II Groups**

<b>Group</b>	<b>RFC</b>	<b>Description</b>
System	RFC 1213 RFC 1907	General system-type MIB objects such as the system description and system name.
Interfaces	RFC 1213	Software and hardware interface statistics (Interfaces).
AT	RFC 1213	Address translation (AT) statistics.
IP	RFC 1213 RFC 2011	System Internet Protocol (IP) related statistics.
ICMP	RFC 1213	Internet Control Message Protocol (ICMP) statistics.
TCP	RFC 1213 RFC 2012	Transmission Control Protocol (TCP) statistics.
UDP	RFC 1213 RFC 2013	User Datagram Protocol (UDP) statistics.
Ether-like	RFC 1643	System Ethernet type statistics (dot3 or Ether-like).
SNMP	RFC 1213 RFC 1907	Basic SNMP statistics such as packet in/out rates, errors, and bad community names (SNMP).
OSPF	RFC 1850	Open Shortest Path First (OSPF) statistics.
BGP	draft-bgp4.MIB	Border Gateway Protocol (BGP) statistics.
RIP2	RFC 1724	Routing Information Protocol (RIP2) statistics.
IF Extensions	RFC 2233	Interface group using the structure of management information in SNMP v2 (SMIv2)

**Table 2-1 Supported MIB-II Groups**

<b>Group</b>	<b>RFC</b>	<b>Description</b>
Entity	RFC 2737	MIB for managing multiple logical and physical entities managed by the SNMP Agent.
IPCDN	RFC 2669	DOCSIS cable device MIB
	RFC 2670	DOCSIS interface-related (docsIf) statistics, such as upstream and downstream interface tables.
	draft-ietf-ipcdn-mcns-bpi-mib-04.txt	DOCSIS baseline privacy configuration and statistics tables.
	docs-cable-device-trap.mib	DOCSIS traps
	docs-if-ext.mib	RFC 2670 extensions
	draft-ietf-ipcdn-qos-mib-04.mib	DOCSIS quality of service
VRRP	draft-vrrp-09.mib	Virtual Router Redundancy Protocol (VRRP).
IGMP	RFC 2933	Internet Group Management Protocol (IGMP).
DVMRP	draft-dvmrp-11.mib	Distance Vector Multicast Routing Protocol (DVMRP).

## Transmission

**Table 2-2 Supported Transmission Groups**

<b>Group</b>	<b>RFC</b>	<b>Description</b>
Ether-like (dot3)	RFC 2665	System Ethernet-like type statistics.
PPP-IPCP	RFC 1473	IP Network Control Protocol of the Point-to-Point Protocol.
PPP-LCP	RFC 1471	Link Control Protocol of the Point-to-Point Protocol.
SONET	RFC 2558	SONET/SDH Interface Type.

## SNMP

**Table 2-3 Supported SNMP Groups**

<b>Group</b>	<b>RFC</b>	<b>Description</b>
Community MIB	RFC 2576	SNMP community MIB
SNMP_Framework	RFC 2571	Basic SNMP structure, such as engine statistics (SNMP_Framework).
MPD	RFC 2572	SNMP Message Processing and Dispatching (MPD).
USM	RFC 2574	SNMP User Security Control Model (USM).
VACM	RFC 2575	SNMP View Access Control Model (VACM).
Target MIB	RFC 2573	SNMP Target MIB.
Notification MIB	RFC 2573	SNMP Notification MIB.

## Experimental

**Table 2-4 Supported Experimental MIBs**

Group	RFC	Description
PIM	RFC 2934	Protocol Independent Multicast MIB for IPv4.
docs Sub Management	draft-submgt-02.mib	Submanagement MIB

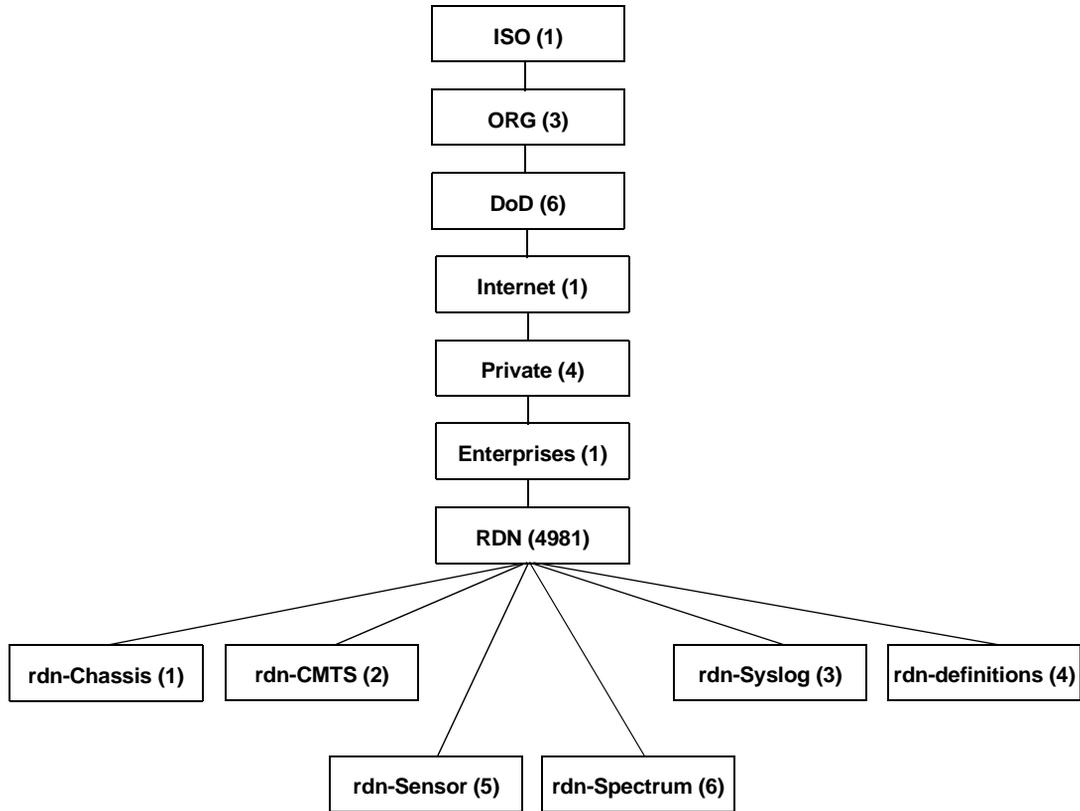
## BSR 64000 Proprietary MIBs

The BSR 64000 supports the following proprietary MIBs. [Table 2-5](#) lists the MIB name and provides a brief description.

**Table 2-5 Proprietary MIB Support**

Proprietary MIB Name	Description
RDN	MIB module definition for RiverDelta Networks.
rdn-Chassis	MIB module definitions for RiverDelta Networks' products, chassis, ports, modules, etc.
rdn-CMTS	Enterprise MIB Module for RiverDelta Networks' Cable Modem Termination Systems (CMTS)
rdn-Syslog	Enterprise MIB for RiverDelta Networks' system log.
rdn-definitions	MIB module definitions for RiverDelta Networks' products, chassis, ports, modules, etc.
rdn-Cable Spectrum	rdn Cable Spectrum MIB for flap list
rdn-Sensor	Chassis temperature status and thresholds

The structure of the proprietary MIBs is shown in [Figure 2-1](#).



**Figure 2-1 BSR 64000 Proprietary MIB Structure**

---

# 3

## **Using a MIB Browser to Access MIB Variables**

---



## Introduction

This chapter provides a brief overview of using a graphical interface MIB browser including:

- Basic procedures for using the browser
- The SNMP Get, GetNext, and Set commands
- The rationale behind updating a MIB database.



**Note:** The information and procedures provided use a “generic” graphical interface MIB browser application as an example. There are many interface-based MIB browsers available and, while they support similar functionality and presentation, there are vendor-specific differences in procedure.

There are also several command line MIB browsers available which are beyond the scope of this document.

## About MIB Browsers

With a MIB browser, you can examine the MIBs supported by an active device on your network and perform SNMP Get, GetNext, and Set commands on the MIB objects.

To successfully communicate with intelligent devices with SNMP, some MIB browsers rely on their own database of MIB information. The OIDs, ASCII names, and other technical information stored in this database allow you to easily search for and select the objects whose information you want to view or change. You use the MIB browser application to both query and set objects in your intelligent devices.

The following sections describe using a MIB browser to:

- Contact a device
- Navigate through its supported MIBs
- Query the device for MIB values
- Set a new value for a MIB object at the device

## Contacting the Device

Before you can access MIB information for a device through a MIB browser, you must establish contact with the device. You contact a device through the MIB browser interface by specifying the device's IP address and the community name.

SNMP community names are used to control management access to a device. The level of control that a managing entity has over the device corresponds to the access level provided by the community name used in a management request: Read Only, Read-Write, or Superuser.

For SNMP communication to take place, the SNMP agent must be running on the device and the specified community name must match one of the device's community names. SNMP runs as a daemon on Solaris and as a service on NT.

To determine if the SNMP agent is running on the device, do the following:

1. Telnet to the device
2. use the **ps** command and look for an entry for the `snmpd` process. For example, this entry may read:

```
407 0 s0 7e8d0 103c 7f7f4 12 12 0 18 0:00.017 ? snmpd
```

## Browsing the MIB Tree

A graphical MIB browser provides an interface that lets you query any SNMP device on your network and examine its supported MIBs. The MIB information is visually structured into a tree format, similar to a directory tree. Using the mouse, you can scroll through the tree and click on folder icons to open the tree layer of interest. Like a file directory tree, the MIB Tree is represented by a series of collapsible and expandable folders, with the individual MIB Objects contained in each folder.

Once you have successfully contacted a device, you will be able to browse its supported MIBs through a MIB tree display in the browser window, and query the device for its MIB object values. Like a file directory tree, the MIB tree is represented by a series of collapsible and expandable folders, containing the individual MIB objects.

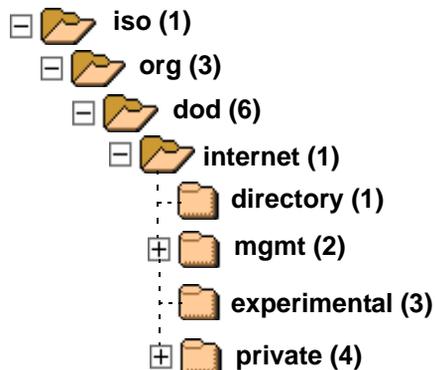
Figure 3-1 shows the top level of the MIB tree is shown. When you click on a level of a tree, it is highlighted to show that it is selected. Each branch of the tree is indicated by a folder.



**Figure 3-1 The Top Level of the MIB Tree**

Figure 3-2 shows that the **iso** -> **org** -> **dod** -> **internet** branch has been opened. Each folder icon changes and is marked with a minus sign (-) to denote that it has been expanded. The **mgmt** and **private** folders are marked with a plus sign (+), indicating that they are expandable to reveal more information, but have not yet been opened.

A folder without a plus sign (+) or minus sign (-) indicates an empty tree branch that is not expandable.



**Figure 3-2 The Path to the Internet Branch of the MIB Tree**

Figure 3-3 shows The MIB tree further expanded to reveal a management object (leaf), specifically the **iso -> org -> dod -> internet -> mgmt -> mib-2 -> system -> sysDescr** object

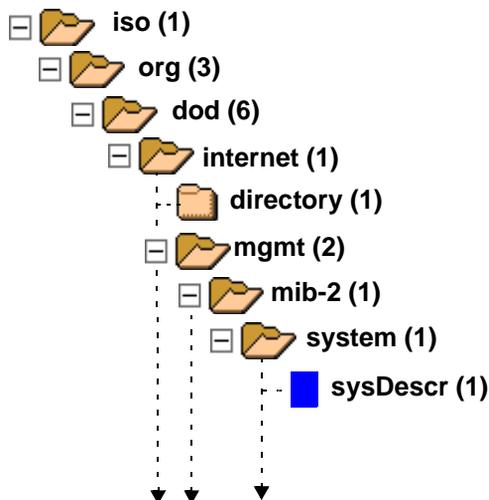


Figure 3-3 The MIB Tree Expanded to Reveal an Object

## SNMP Get/Get Next

To find the current value set at the selected device for a specific MIB object or set of objects, you must *query* the device for the information. A query consists of an SNMP Get (to retrieve a specific piece of management information) or SNMP GetNext (to traverse a portion of the tree and retrieve all its information) command issued by the browser to the contacted device. When you query a device, it returns a value for every instance of the specified object(s).

## SNMP Set

With a MIB browser, you can change the value of a writable object in a device's MIB. This is accomplished with an SNMP Set command. If you attempt to change the value of an object and a message similar to "Selected item is not write enabled." appears. This may occur because:

- You do not have permission to write to the device
- The object itself has been designated read-only.
- Objects which are present in the device have not been added to the current MIB database.

Before you begin, you can determine if an object is writable by performing a query on the object and reading the data that is returned.

## Compiling and Importing New SNMP MIBs

There are several reasons that you may want to import additional or updated MIBs into your database, or create custom MIB databases:

- You want to manage third-party devices not already supported, you will need to import their vendors' proprietary MIBs (which will not be included in your default MIB database).
- MIBs — especially proprietary MIBs — are frequently updated, and you may need to import a new version of a MIB.
- Some of your newer networking devices may still be under heavy development, and new firmware may include functionality provided by entirely new proprietary MIBs.

By importing new and updated MIBs to create custom MIB databases, you can ensure that your MIB databases remain up-to-date and contain all the MIB information you need to successfully monitor and manage your network devices.



---

# 4

## Sample Tasks

---



## Introduction

This chapter provides two sample tasks using a graphical MIB browser:

- Accessing MIB variable information
- Changing a MIB variable

## Accessing MIB Variable Information

To find the current value set at the selected device for a specific MIB object or set of objects, the MIB browser issues an SNMP Get command (to retrieve a specific piece of management information) or SNMP Get Next command (to traverse a portion of the tree and retrieve all its information).

### Sample Task

A network administrator needs to know how long a particular device has been on-line since it was last rebooted. In our example, we will be finding the value of the *sysUpTime* object. The MIB hierarchy that leads to the *sysUpTime* object is:

1.2.6.1.2.1.1.3

iso -> org -> dod-> internet -> mgmt -> mib-2 -> system -> sysUpTime

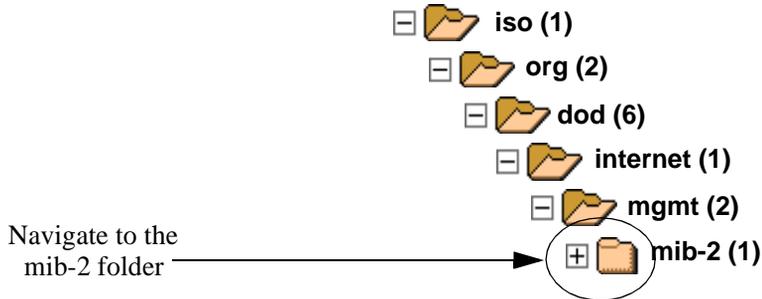


**Note:** Be sure that you have contacted the device and that you are using a community name which provides you with write access to the desired MIB object. The appropriate community name may depend upon which MIB component contains the object you want to set. Refer to [Contacting the Device](#).

To query a device for the value of the *sysUpTime* object, do the following:

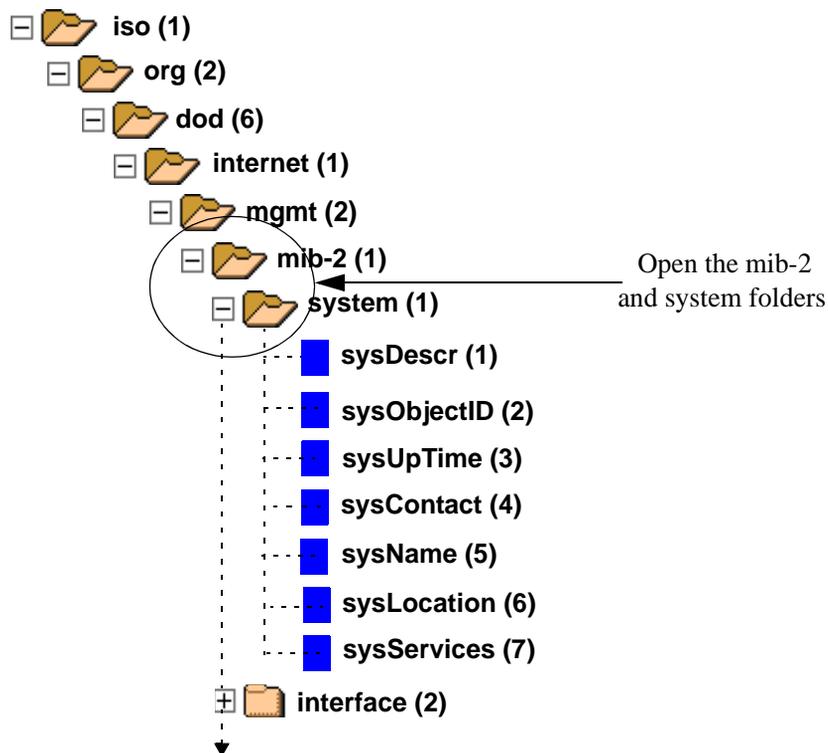
1. Start your MIB browser.

2. Connect to the device of interest by specifying the device's IP address and community name.
3. In the MIB tree display, navigate through the MIB hierarchy until you reach the **mib-2** folder (Figure 4-1).



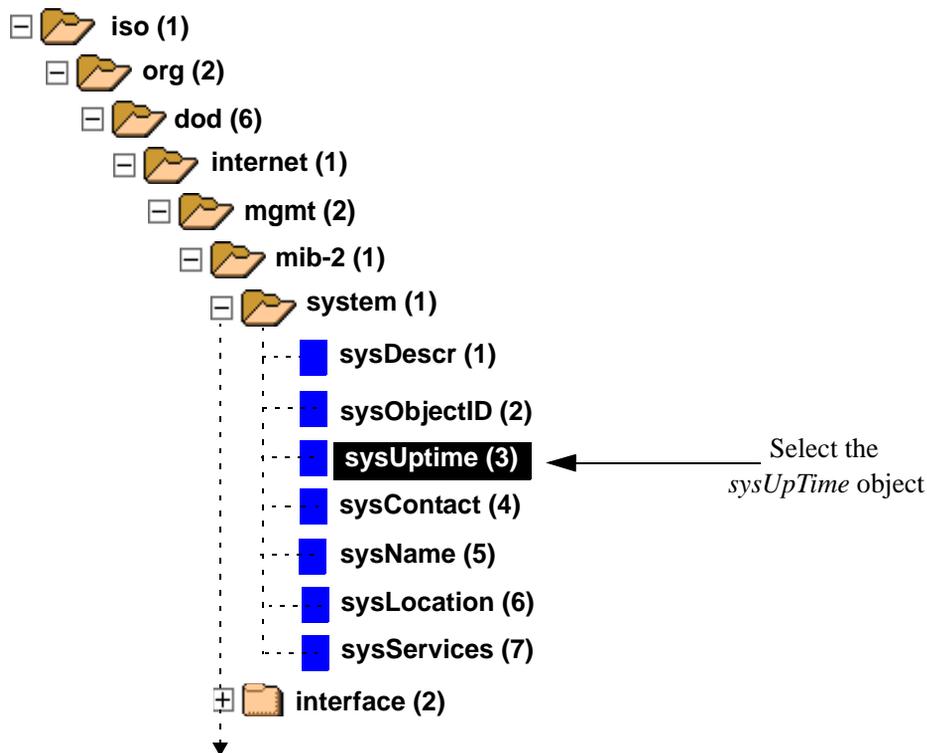
**Figure 4-1 The mib-2 Hierarchy**

4. Open the **mib-2** folder
5. Open the **system** folder to display the objects in the system group (Figure 4-2).



**Figure 4-2** The mib-2 and system Folders

6. Select the *sysUpTime* object (Figure 4-3).



**Figure 4-3 Selecting the *sysUpTime* Object**

All or part of the following should be retrieved and displayed for the *sysUpTime* object:

Object Name	sysUpTime
Object ID	1.2.6.1.2.1.1.3
Type	TimeTicks
Access	read-only
Description	The time (in hundredths of a second) since the network management portion of the system was last re-initialized.
Value	28 days 20:05:41



**Note:** Depending on the particular MIB browser application that you are using, the query information for the *sysUpTime* object may be accessed and displayed differently. Refer to your MIB browser documentation.

## Changing MIB Variables

To change the value of a writable MIB object, the MIB browser issues an SNMP Set command. Before you begin, you can determine if an object is writable by performing a query on the object and reading the data that is returned. See [Accessing MIB Variable Information](#).



**Note:** Be sure that you have contacted the device and that you are using a community name which provides you with write access to the desired MIB object. The appropriate community name may depend upon which MIB component contains the object you want to set. Refer to [Contacting the Device](#).

## Sample Task

A network administrator wants to clear all messages from the syslog files on a remote syslog server device. In our example, we will be changing the value of the *rdnSyslogClear* object. The MIB hierarchy that leads to the *rdnSyslogClear* object is:

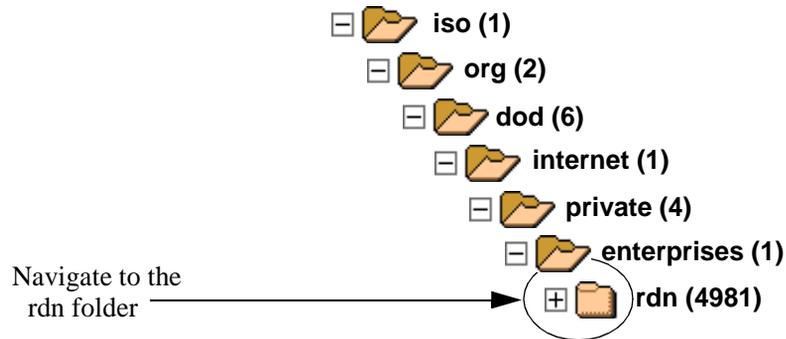
1.2.6.1.4.1.4981.3.7

iso -> org -> dod -> internet -> private -> enterprises -> rdn -> syslog  
->rdnSyslogClear

To change the value of the *rdnSyslogClear* object on a device, do the following:

1. Start your MIB browser.

2. Connect to the device of interest by specifying the device's IP address and community name.
3. In the MIB tree display, navigate through the MIB hierarchy until you reach the **rdn** folder (Figure 4-4).



**Figure 4-4 The rdn MIB Hierarchy**

4. Open the **rdn** folder
5. Open the **syslog** folder to display the objects in the syslog group (Figure 4-5).



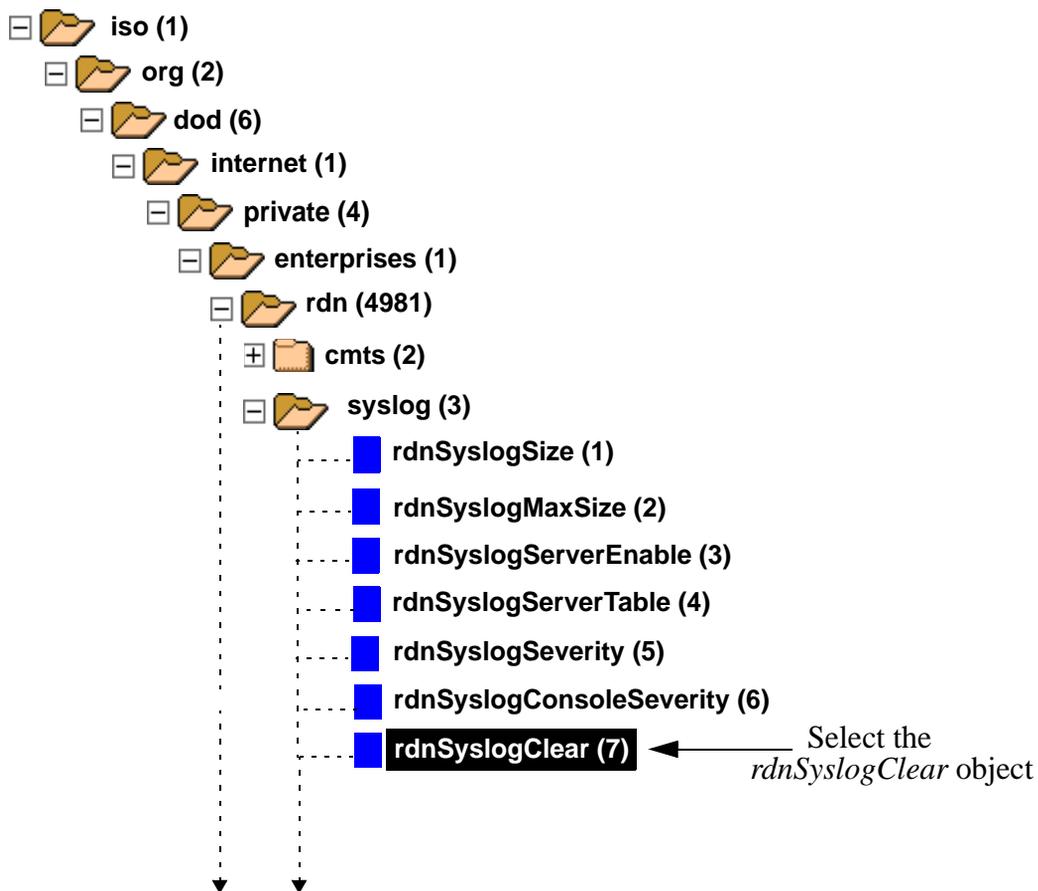


Figure 4-6 Selecting the *rdnSyslogClear* Object

7. Set the *rdnSyslogClear* object to “TRUE”.



**Note:** The exact procedure for changing an object's value through a MIB browser interface depends on the particular MIB browser application that you are using. Refer to your MIB browser documentation.



---

# Index

## A

Abstract Syntax Notation One, 1-3  
address translation statistics, 10  
AT, 10

## B

BGP, 10  
Border Gateway Protocol, 10  
BSR 64000  
    proprietary MIBs, 1-2, 9  
    CMTS, 13  
    RDN, 13  
    Syslog, 13  
SNMP agent software, 1-2  
Syslog, 13

## C

Cable Modem Termination Systems, 13  
Chassis, 13

## D

Distance Vector Multicast Routing Protocol, 11  
docsIf, 11  
DOCSIS, 11  
dot3, 10  
DVMRP, 11

## E

Entity MIB, 11

Ether-like, 10  
Ether-like (dot3), 12  
Ethernet statistics, 10

## F

firmware, 3-5  
folder, 3-2  
    icon, 3-3

## G

Get, 1-2  
GetNext, 1-2

## I

ICMP, 10  
IETF, 1-1  
IF Extensions, 10  
IGMP, 11  
Interfaces, 10  
Internet Assigned Numbers Authority, 1-2  
Internet Control Message Protocol, 10  
Internet Engineering Task Force, 1-1  
Internet Group Management Protocol, 11  
Internet Protocol, 10  
IP, 10  
IP address, 3-2, 4-2, 4-6  
IP Network Control Protocol, 12

IPCDN, 11

## L

Link Control Protocol, 12

## M

Management Information Base, 1-1

MIB, 1-1

- browser, 3-1, 3-2, 3-5
- command line, 3-1
  - graphical, 3-1, 4-1
  - interface, 3-1, 4-9
- compiling, 3-5
- database, 3-1, 3-5
- hierarchy, 4-1, 4-2, 4-5, 4-6
- importing, 3-5
- Object Identifier, 1-2
- OID, 1-2
- organization, 1-2
- private, 1-1
- proprietary, 9, 13, 3-5
- RFC, 1-1
- standard, 9
- supported, 3-1, 3-2
- tree, 3-2, 3-3, 4-2, 4-6
  - browsing, 3-2
- values, 3-1
- variable
  - accessing, 4-1
  - changing, 4-1

MPD, 12

## O

object

- ACCESS, 1-4
- DESCRIPTION, 1-4
- instance, 1-3
- OBJECT\_TYPE, 1-3
- STATUS
  - mandatory, 1-4
  - obsoleted, 1-4
  - optional, 1-4
- SYNTAX, 1-3

- tabular, 1-3
- writable, 3-5, 4-5

Open Shortest Path First, 10

OSPF, 10

## P

Point-to-Point Protocol., 12

PPP-IPCP, 12

PPP-LCP, 12

## Q

query, 3-1, 3-4

## R

Request for Comments, 1-1

RIP2, 10

Routing Information Protocol, 10

## S

Set, 1-2, 3-5

SMI, 1-1, 1-3

SMIv2, 10

SNMP, 1-1

- agent, 1-1, 1-2, 3-2
- commands, 1-2
- community name, 3-2, 4-1, 4-5
- Get, 3-4
- get, 3-1, 4-1
- get next, 4-1
- GetNext, 3-1, 3-4
- management, 1-1
- Message Processing and Dispatching, 12
- network management station, 1-1, 1-2
- Set, 3-1, 3-5, 4-5
- statistics, 10
- structure, 12
- User Security Control Model, 12
- V1, 9
- V2, 9
- V3, 9

---

View Access Control Model, 12  
SNMP Agent, 11  
SNMP v2, 10  
SNMP\_Framework, 12  
SONET, 12  
SONET/SDH, 12  
Structure of Management Information, 1-1  
system, 10

## **T**

TCP, 10  
Transmission Control Protocol, 10  
tree  
    branches, 1-2  
    leaves, 1-2  
    MIB, 3-3  
    structure, 1-2

## **U**

UDP, 10  
User Datagram Protocol, 10  
USM, 12

## **V**

VACM, 12  
Virtual Router Redundancy Protocol, 11  
VRRP, 11

