

**Oracle[®] VM Server for SPARC 3.6
Management Information Base User's
Guide**

ORACLE[®]

Part No: E93618
August 2018

Part No: E93618

Copyright © 2007, 2018, 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.

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: E93618

Copyright © 2007, 2018, 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, accorder de licence, 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 livré sous licence au Gouvernement des Etats-Unis, ou à quiconque qui aurait souscrit la licence de ce logiciel 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 un risque de dommages corporels. Si vous utilisez ce logiciel ou ce 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 des applications dangereuses.

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 de 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.

Accès aux services de support Oracle

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

- 1 Oracle VM Server for SPARC Management Information Base Overview** 11
 - Related Products and Features 11
 - Software Components 12
 - Oracle Solaris SNMP Agent 13
 - Logical Domains Manager and the Oracle VM Server for SPARC MIB 14
 - Parsing the XML-Based Control Interface 14
 - Providing SNMP Traps 14
 - Providing Fault and Recovery Information 15
 - Oracle VM Server for SPARC MIB Object Tree 15

- 2 Installing and Configuring the Oracle VM Server for SPARC MIB Software** 17
 - Installing and Configuring the Oracle VM Server for SPARC MIB Software 17
 - ▼ How to Install the Oracle VM Server for SPARC MIB Software Package 18
 - ▼ How to Load the Oracle VM Server for SPARC MIB Module Into the Oracle Solaris SNMP Agent 19
 - ▼ How to Remove the Oracle VM Server for SPARC MIB Software Package 19

- 3 Monitoring Domains** 21
 - Setting Environment Variables 21
 - Querying the Oracle VM Server for SPARC MIB 21
 - Retrieving Oracle VM Server for SPARC MIB Information 24
 - Domain Table (ldomTable) 24
 - Environment Variables Table (ldomEnvVarsTable) 26

Domain Policy Table (ldomPolicyTable)	27
Service Processor Configuration Table (ldomSPConfigTable)	28
Domain Resource Pool and Scalar Variables	28
Virtual CPU Table (ldomVcpuTable)	30
Virtual Memory Tables	31
Virtual Disk Tables	33
SCSI HBA Tables	36
Virtual Network Tables	39
Virtual Console Tables	41
Cryptographic Units Table (ldomCryptoTable)	45
I/O Bus Table (ldomIOBusTable)	45
CMI Table (ldomCMITable)	45
Core Table (ldomCoreTable)	46
Scalar Variables for Domain Version Information	46
4 Using SNMP Traps	49
Using Oracle VM Server for SPARC MIB Module Traps	49
▼ How to Send Traps	49
▼ How to Receive Traps	50
Oracle VM Server for SPARC MIB Trap Descriptions	51
Domain Creation (ldomCreate)	51
Domain Destroy (ldomDestroy)	51
Domain State Change (ldomStateChange)	52
Virtual CPU Change (ldomVCpuChange)	52
Virtual Memory Change (ldomVMemChange)	52
Virtual Disk Service Change (ldomVdsChange)	53
Virtual Disk Change (ldomVdiskChange)	54
SCSI HBA Change (ldomHbaChange)	54
Virtual SCSI HBA Change (ldomVhbaChange)	55
Virtual SAN Change (ldomVsanChange)	55
Virtual Switch Change (ldomVswChange)	56
Virtual Network Change (ldomVnetChange)	56
Virtual Console Concentrator Change (ldomVccChange)	57
Virtual Console Group Change (ldomVconsChange)	57
CMI Resource Change (ldomCMICChange)	58

5 Starting and Stopping Domains	59
Starting and Stopping Logical Domains	59
▼ How to Start a Domain	59
▼ How to Stop a Domain	61

Using This Documentation

- **Overview** – Provides information about using the Oracle VM Server for SPARC 3.6 Management Information Base software.
- **Audience** – System administrators who manage and monitor Oracle VM Server for SPARC configurations
- **Required knowledge** – System administrators on these servers must have a working knowledge of the Oracle Solaris operating system (Oracle Solaris OS) and the Oracle VM Server for SPARC 3.6 software

Product Documentation Library

Documentation and resources for this product and related products are available at <http://www.oracle.com/technetwork/documentation/vm-sparc-194287.html>.

Feedback

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

◆◆◆ CHAPTER 1

Oracle VM Server for SPARC Management Information Base Overview

The Oracle VM Server for SPARC Management Information Base (MIB) enables third-party system management applications to perform remote monitoring of domains, and to start and stop logical domains (domains) by using the Simple Network Management Protocol (SNMP).

You can run only one instance of the Oracle VM Server for SPARC MIB software on the control domain. The control domain should run at least the Solaris 10 11/06 OS and at least the Oracle VM Server for SPARC 2.2 software.

This chapter covers the following topics:

- [“Related Products and Features” on page 11](#)
- [“Software Components” on page 12](#)
- [“Oracle Solaris SNMP Agent” on page 13](#)
- [“Logical Domains Manager and the Oracle VM Server for SPARC MIB” on page 14](#)
- [“Oracle VM Server for SPARC MIB Object Tree” on page 15](#)

Related Products and Features

To successfully use the Oracle VM Server for SPARC MIB, you must understand how to use the following software products and features:

- Oracle Solaris OS
- Oracle VM Server for SPARC software
- Simple Network Management Protocol (SNMP)
- SNMP Management Information Base (MIB)
- Oracle Solaris SNMP Agents
- SNMP version 1 (SNMPv1), SNMP version 2 (SNMPv2c), and SNMP version 3 (SNMPv3) protocols

- Structure of Management Information (SMI) version 1 and version 2
- Management Information Base (MIB) structure
- Abstract Syntax Notation (ASN.1)

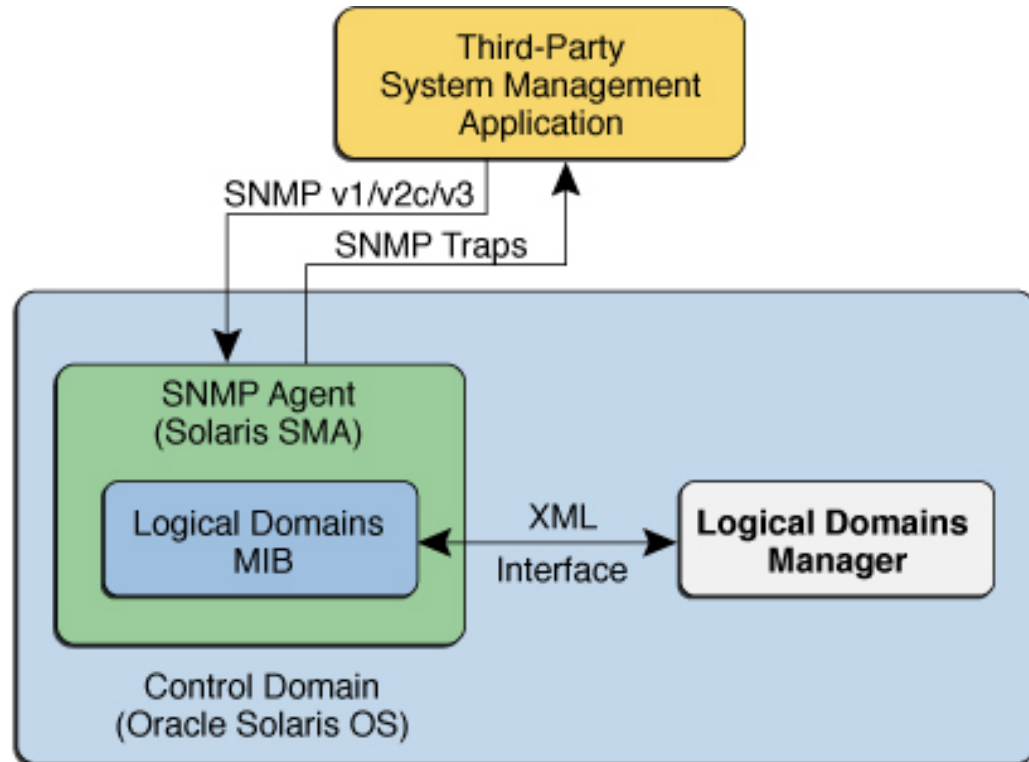
Software Components

The Oracle VM Server for SPARC MIB package, `SUNWldmib.v`, contains the following software components:

- `SUN-LDOM-MIB.mib` is an SNMP MIB in the form of a text file. This file defines the objects in the Oracle VM Server for SPARC MIB.
- `ldomMIB.so` is a Oracle Solaris SNMP agent (net-SNMP) extension module in the form of a shared library. This module enables the Oracle Solaris SNMP agent to respond to requests for information that are specified in the Oracle VM Server for SPARC MIB and to generate traps.

The following figure shows the interaction between the Oracle VM Server for SPARC MIB, the Oracle Solaris SNMP agent, the Logical Domains Manager, and a third-party system management application. The interaction shown in this figure is described in [“Oracle Solaris SNMP Agent” on page 13](#) and [“Logical Domains Manager and the Oracle VM Server for SPARC MIB” on page 14](#).

FIGURE 1 Oracle VM Server for SPARC MIB Interaction With Oracle Solaris SNMP Agent, Logical Domains Manager, and a Third-Party System Management Application



Oracle Solaris SNMP Agent

The Oracle Solaris SNMP agent (net-SNMP) performs the following functions:

- Listens for requests from a third-party system management application to get or set data offered by the Oracle VM Server for SPARC MIB. The agent listens on the standard SNMP port, 161.
- Issues traps to the configured system management application by using the standard port for SNMP notifications, 162.

The Oracle VM Server for SPARC MIB is exported by the Oracle Solaris OS default Oracle Solaris SNMP agent on the control domain.

The Oracle Solaris SNMP agent supports the get, set, and trap functions of SNMP versions v1, v2c, and v3. Most Oracle VM Server for SPARC MIB objects are read-only for monitoring purposes. However, to start or stop a domain, you must write a value to the `ldomAdminState` property of the `ldomTable` table. See [Table 2, “Domain Table \(ldomTable\),” on page 24](#).

Logical Domains Manager and the Oracle VM Server for SPARC MIB

A *domain* is a container that consists of a set of virtual resources for a guest operating system. The Logical Domains Manager provides the command-line interface (CLI) for creating, configuring, and managing the domains. The Logical Domains Manager and the Oracle VM Server for SPARC MIB support the following virtual resources:

- CPUs
- Memory
- Disk, network, and console I/O
- Cryptographic units

Parsing the XML-Based Control Interface

The Logical Domains Manager exports an XML-based control interface to the Oracle VM Server for SPARC MIB. The Oracle VM Server for SPARC MIB parses the XML interface and populates the MIB. The Oracle VM Server for SPARC MIB only provides support for the control domain.

Providing SNMP Traps

The Oracle VM Server for SPARC MIB polls the Logical Domains Manager periodically for updates or status changes, and then issues SNMP traps to the system management applications.

Providing Fault and Recovery Information

If the Oracle VM Server for SPARC MIB can no longer allocate a needed resource, the MIB returns a general error to the system management application through the Oracle Solaris SNMP agent (net-SNMP). The SNMP trap-delivery mechanism does not confirm the error. No specific state or checkpointing is implemented in the Oracle VM Server for SPARC MIB. The Oracle Solaris SNMP agent with the Oracle VM Server for SPARC MIB is started and monitored by the `init` process and the Service Management Facility (SMF). If the Oracle Solaris SNMP agent fails and exits, SMF restarts the process automatically, and then the new process dynamically restarts the Oracle VM Server for SPARC MIB module.

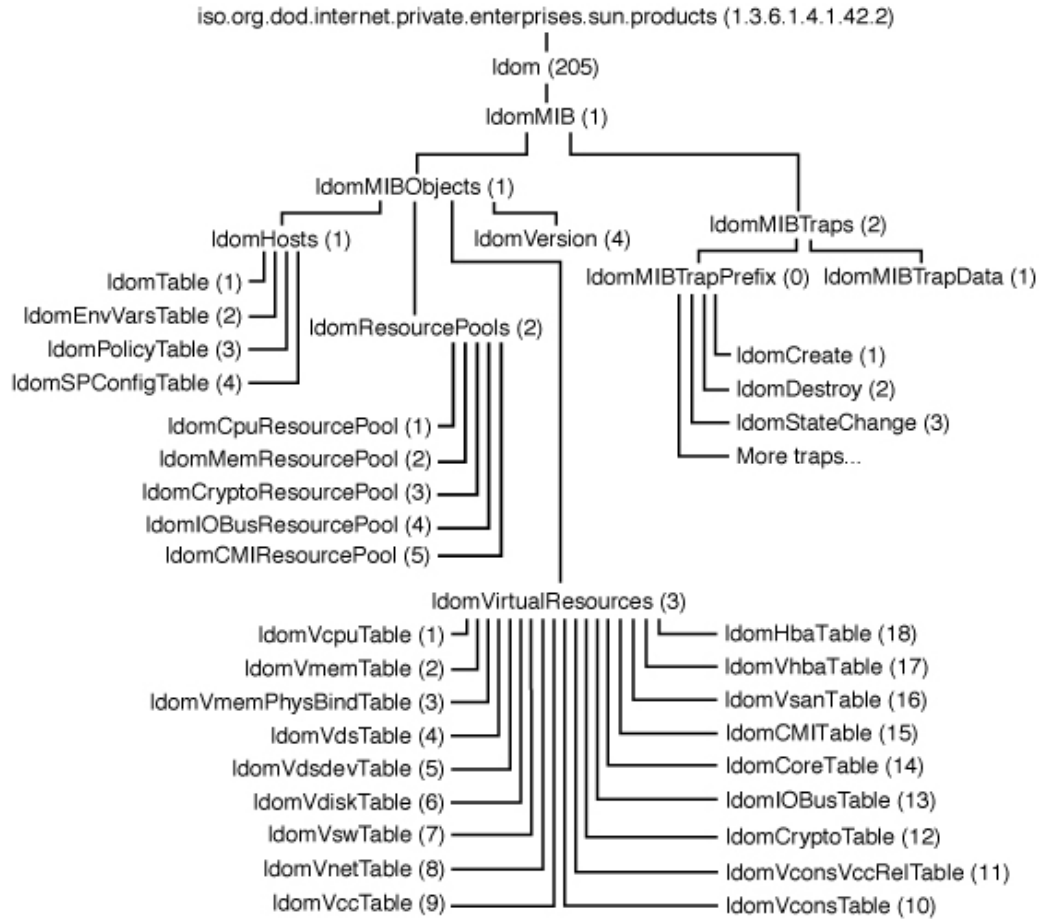
Oracle VM Server for SPARC MIB Object Tree

SNMP-managed objects are organized into a tree-like hierarchy. An object identifier (OID) consists of a series of integers based on the nodes in the tree, separated by dots. Each managed object has a numerical OID and an associated textual name. The Oracle VM Server for SPARC MIB is registered as the `ldom (205)` branch in this part of the object tree:

```
iso(1).org(3).dod(6).internet(1).private(4).enterprises(1).sun(42).products(2)
```

The following figure shows the major subtrees under the Oracle VM Server for SPARC MIB.

FIGURE 2 Oracle VM Server for SPARC MIB Tree



◆◆◆ CHAPTER 2

Installing and Configuring the Oracle VM Server for SPARC MIB Software

This chapter covers the installation and configuration of the Oracle VM Server for SPARC MIB software. For information about administering SNMP, see the `snmpd.conf(5)` man page on the open source community web site <http://www.net-snmp.org/docs/man/>.

After you install and configure the Oracle VM Server for SPARC MIB software, you must create Simple Network Management Protocol (SNMP) version 3 (v3) users to provide secure access to the Oracle Solaris SNMP agent. For SNMP version 1 (v1) and version 2 (v2c), the access control mechanism is the *community string*, which defines the relationship between an SNMP server and its clients. This string controls the client access to the server similar to a password controlling a user's access to a system.

Note - Creating `snmpv3` users enables you to use the Oracle Solaris SNMP agent in SNMP with the Oracle VM Server for SPARC MIB. This type of user does not interact with or conflict with users that you might have configured by using the rights feature of Oracle Solaris for the Logical Domains Manager. For information about creating this user, see the documentation on the [Net-SNMP open source community web site \(http://www.net-snmp.org/\)](http://www.net-snmp.org/).

Installing and Configuring the Oracle VM Server for SPARC MIB Software

The following table lists the tasks that you can use to install and configure the Oracle VM Server for SPARC MIB software.

TABLE 1 Task Map: Installing and Configuring the Oracle VM Server for SPARC MIB Software

Task	Description	For Instructions
Install the Oracle VM Server for SPARC MIB software package on the primary domain.	Use the <code>pkg install</code> command to install the <code>system/ldoms/mib</code> package.	“How to Install the Oracle VM Server for SPARC MIB Software Package” on page 18
Load the Oracle VM Server for SPARC MIB module into the Oracle Solaris SNMP agent to query the Oracle VM Server for SPARC MIB.	Modify the SNMP configuration file to load the <code>ldomMIB.so</code> module.	“How to Load the Oracle VM Server for SPARC MIB Module Into the Oracle Solaris SNMP Agent” on page 19
Remove the Oracle VM Server for SPARC MIB software package from the primary domain.	Use the <code>pkg remove</code> command to remove the <code>system/ldoms/mib</code> package.	“How to Remove the Oracle VM Server for SPARC MIB Software Package” on page 19

▼ How to Install the Oracle VM Server for SPARC MIB Software Package

This procedure describes how to install the Oracle VM Server for SPARC MIB software package. The Oracle VM Server for SPARC MIB software package is included as part of the Oracle VM Server for SPARC 3.6 software.

The Oracle VM Server for SPARC MIB package includes the following files:

- `/opt/SUNWldmib/lib/mibs/SUN-LDOM-MIB.mib`
- `/opt/SUNWldmib/lib/ldomMIB.so`

Before You Begin Install the Oracle VM Server for SPARC 3.6 software. See [Chapter 2, “Installing the Software” in *Oracle VM Server for SPARC 3.6 Installation Guide*](#).

- **Install the Oracle VM Server for SPARC MIB software package, `system/ldoms/mib`.**

```
primary# pkg install -v pkg:/system/ldoms/mib
```

Next Steps After you install this package, you can configure your system to dynamically load the Oracle VM Server for SPARC MIB module. See [“How to Load the Oracle VM Server for SPARC MIB Module Into the Oracle Solaris SNMP Agent” on page 19](#).

▼ How to Load the Oracle VM Server for SPARC MIB Module Into the Oracle Solaris SNMP Agent

The Oracle VM Server for SPARC MIB module, `ldomMIB.so`, must be loaded into the Oracle Solaris SNMP agent to query the Oracle VM Server for SPARC MIB. The Oracle VM Server for SPARC MIB module is dynamically loaded so that the module is included within the Oracle Solaris SNMP agent (`net-SNMP`) without requiring you to recompile and relink the agent binary.

This procedure describes how to configure your system to dynamically load the Oracle VM Server for SPARC MIB module. For information about the Oracle Solaris SNMP agent (`net-SNMP`), install the `pkg:/system/management/snmp/net-snmp/documentation` package, which contains the Oracle Solaris SNMP agent (`net-SNMP`) agent man pages and HTML documentation files.

1. **Update the SNMP configuration file.**

Append the following line to the `/etc/net-snmp/snmp/snmpd.conf` configuration file:

```
dldmod ldomMIB /opt/SUNWldmib/lib/ldomMIB.so
```

2. **Restart the SMF service.**

```
primary# svcadm restart svc:/application/management/net-snmp:default
```

▼ How to Remove the Oracle VM Server for SPARC MIB Software Package

This procedure describes how to remove the Oracle VM Server for SPARC MIB software package and unload the Oracle VM Server for SPARC MIB module.

1. **Stop the SMF service.**

```
primary# svcadm disable svc:/application/management/net-snmp:default
```

2. **Update the SNMP configuration file.**

Remove the line that you added to the `/etc/net-snmp/snmp/snmpd.conf` file during installation.

```
dldmod ldomMIB /opt/SUNWldmib/lib/ldomMIB.so
```

3. Restart the SMF service.

```
primary# svcadm restart svc:/application/management/net-snmp:default
```

4. Remove the Oracle VM Server for SPARC MIB software package from the primary domain.

```
primary# pkg uninstall system/ldoms/mib
```

◆◆◆ CHAPTER 3

Monitoring Domains

This chapter describes how to monitor logical domains (domains) by querying the Oracle VM Server for SPARC MIB. This section also provides descriptions of the various types of MIB output.

This chapter covers the following topics:

- [“Setting Environment Variables” on page 21](#)
- [“Querying the Oracle VM Server for SPARC MIB” on page 21](#)
- [“Retrieving Oracle VM Server for SPARC MIB Information” on page 24](#)

Setting Environment Variables

Before you can query the Oracle VM Server for SPARC MIB, you must set the PATH, MIBDIRS, and MIBS environment variables.

```
$ PATH=/usr/bin:$PATH; export PATH
$ MIBDIRS=/opt/SUNWldmib/lib/mibs:/etc/net-snmp/snmp/mibs; export MIBDIRS
$ MIBS+=SUN-LDOM-MIB; export MIBS
```

Querying the Oracle VM Server for SPARC MIB

When a system has large number of domains, the Oracle Solaris SNMP agent (net-SNMP) might time out before being able to respond to an SNMP request. To increase the timeout value, use the `-t` option to specify a longer timeout value. For example, the following `snmpwalk` command sets the timeout value to 20 seconds:

```
# snmpwalk -t 20 -v1 -c public localhost SUN-LDOM-MIB::ldomTable
```

You can also use the `-t` option to specify the timeout value for the `snmpget` and `snmptable` commands.

- To retrieve a single MIB object:

```
# snmpget -v version -c community-string host MIB-object
```

- To retrieve an array of MIB objects:

Use the `snmpwalk` or `snmptable` command.

```
# snmpwalk -v version -c community-string host MIB-object
# snmptable -v version -c community-string host MIB-object
```

Note - You receive empty SNMP tables if you query the Oracle VM Server for SPARC MIB 2.1 software using the `snmptable` command with the `-v2c` or `-v3` option. The `snmptable` command with the `-v1` option works as expected.

To work around this issue, use the `-CB` option to use only GETNEXT, not GETBULK, requests to retrieve data. See [“Querying the Oracle VM Server for SPARC MIB” on page 21](#).

EXAMPLE 1 Retrieving a Single Oracle VM Server for SPARC MIB Object (`snmpget`)

The following `snmpget` command queries the value of the `ldomVersionMajor` object. The command specifies `snmpv1` (`-v1`) and a community string (`-c public`) for the `localhost` host.

```
# snmpget -v1 -c public localhost SUN-LDOM-MIB::ldomVersionMajor.0
SUN-LDOM-MIB::ldomVersionMajor.0 = INTEGER: 1
```

EXAMPLE 2 Retrieving Object Values From `ldomTable` (`snmpwalk`)

The following examples show how to use the `snmpwalk` command to retrieve object values from `ldomTable`.

- The following `snmpwalk -v1` command returns the values for all objects in the `ldomTable` table.

```
# snmpwalk -v1 -c public localhost SUN-LDOM-MIB::ldomTable
SUN-LDOM-MIB::ldomName.1 = STRING: primary
SUN-LDOM-MIB::ldomName.2 = STRING: LdomMibTest_1
SUN-LDOM-MIB::ldomAdminState.1 = INTEGER: 0
SUN-LDOM-MIB::ldomAdminState.2 = INTEGER: 0
SUN-LDOM-MIB::ldomOperState.1 = INTEGER: active(1)
```

```

SUN-LDOM-MIB::ldomOperState.2 = INTEGER: bound(6)
SUN-LDOM-MIB::ldomNumVCpu.1 = INTEGER: 8
SUN-LDOM-MIB::ldomNumVCpu.2 = INTEGER: 4
SUN-LDOM-MIB::ldomMemSize.1 = INTEGER: 3360
SUN-LDOM-MIB::ldomMemSize.2 = INTEGER: 256
SUN-LDOM-MIB::ldomMemUnit.1 = INTEGER: megabytes(2)
SUN-LDOM-MIB::ldomMemUnit.2 = INTEGER: megabytes(2)
SUN-LDOM-MIB::ldomNumCrypto.1 = INTEGER: 1
SUN-LDOM-MIB::ldomNumCrypto.2 = INTEGER: 0
SUN-LDOM-MIB::ldomNumIOBus.1 = INTEGER: 2
SUN-LDOM-MIB::ldomNumIOBus.2 = INTEGER: 0
SUN-LDOM-MIB::ldomUUID.1 = STRING: 5f8817d4-5d2e-6f7d-c4af-91b5b34b5723
SUN-LDOM-MIB::ldomUUID.2 = STRING: 11284146-87ca-4877-8d80-cd0f60d5ec26
SUN-LDOM-MIB::ldomMacAddress.1 = STRING: 00:14:4f:46:47:d6
SUN-LDOM-MIB::ldomMacAddress.2 = STRING: 00:14:4f:f8:d5:6c
SUN-LDOM-MIB::ldomHostID.1 = STRING: 0x844647d6
SUN-LDOM-MIB::ldomHostID.2 = STRING: 0x84f8d56c
SUN-LDOM-MIB::ldomFailurePolicy.1 = STRING: ignore
SUN-LDOM-MIB::ldomFailurePolicy.2 = STRING: ignore
SUN-LDOM-MIB::ldomMaster.1 = STRING:
SUN-LDOM-MIB::ldomMaster.2 = STRING:
SUN-LDOM-MIB::ldomExtMapInSpace.1 = STRING: off
SUN-LDOM-MIB::ldomExtMapInSpace.2 = STRING: off
SUN-LDOM-MIB::ldomWholeCore.1 = INTEGER: 0
SUN-LDOM-MIB::ldomWholeCore.2 = INTEGER: 0
SUN-LDOM-MIB::ldomCpuArch.1 = STRING: native
SUN-LDOM-MIB::ldomCpuArch.2 = STRING: native
SUN-LDOM-MIB::ldomShutdownGroup.1 = INTEGER: 0
SUN-LDOM-MIB::ldomShutdownGroup.2 = INTEGER: 15
SUN-LDOM-MIB::ldomPerfCounters.1 = STRING: htstrand
SUN-LDOM-MIB::ldomPerfCounters.2 = STRING: global,htstrand
SUN-LDOM-MIB::ldomNumCMI.1 = INTEGER: 0
SUN-LDOM-MIB::ldomNumCMI.2 = INTEGER: 0
SUN-LDOM-MIB::ldomBootPolicy.1 = STRING: n/a
SUN-LDOM-MIB::ldomBootPolicy.2 = STRING: warning

```

- The following snmpwalk commands use snmpv2c and snmpv3 to retrieve the contents of ldomTable:

```

# snmpwalk -v2c -c public localhost SUN-LDOM-MIB::ldomTable
# snmpwalk -v 3 -u test -l authNoPriv -a MD5 -A testpassword localhost \
SUN-LDOMMIB::ldomTable

```

EXAMPLE 3 Retrieving Object Values From `ldomTable` in Tabular Form (`snmptable`)

The following examples show how to use the `snmptable` command to retrieve object values from `ldomTable` in tabular form.

- The following `snmptable -v1` command shows the contents of `ldomTable` in tabular form.

```
# snmptable -v1 -c public localhost SUN-LDOM-MIB::ldomTable
```

- The following `snmptable` command shows the contents of `ldomTable` in tabular form by using `snmpv2c`.

Note that for the `v2c` or `v3` `snmptable` command, use the `-CB` option to specify only `GETNEXT`, not `GETBULK`, requests to retrieve data.

```
# snmptable -v2c -CB -c public localhost SUN-LDOM-MIB::ldomTable
```

Retrieving Oracle VM Server for SPARC MIB Information

This section describes the information that you can retrieve from the Oracle VM Server for SPARC MIB in the form of tables or scalar objects.

Domain Table (`ldomTable`)

`ldomTable` is used to represent each domain in the system. Information includes resource constraints for virtual CPUs, memory, cryptographic units, and I/O buses. The table also includes other domain information, such as the universally unique identifier (UUID), MAC address, host ID, failure policy, and master domain.

TABLE 2 Domain Table (`ldomTable`)

Name	Data Type	Access	Description
<code>ldomIndex</code>	Integer	Not accessible	Integer that is used as an index of this table
<code>ldomName</code>	Display string	Read-only	Name of the domain
<code>ldomAdminState</code>	Integer	Read/Write	Starts or stops the domain for active management: <ul style="list-style-type: none"> ■ Value of 1 starts the domain ■ Value of 2 stops the domain
<code>ldomOperState</code>	Integer	Read-only	Current state of the domain, which can be one of the following values:

Name	Data Type	Access	Description
			<ul style="list-style-type: none"> ■ 1 is the Active state ■ 2 is the Stopping state ■ 3 is the Inactive state ■ 4 is the Binding state ■ 5 is the Unbinding state ■ 6 is the Bound state ■ 7 is the Starting state
ldomNumVCPU	Integer	Read-only	Number of virtual CPUs used. If the domain is in an inactive state, this value is the requested number of virtual CPUs.
ldomMemSize	Integer	Read-only	Amount of virtual memory used. If the domain is in an inactive state, this value is the requested memory size.
ldomMemUnit	Integer	Read-only	One of the following memory units: <ul style="list-style-type: none"> ■ 1 is KB ■ 2 is MB ■ 3 is GB ■ 4 is bytes <p>If not specified, the unit value is bytes.</p>
ldomNumCrypto	Integer	Read-only	Number of cryptographic units used. If the domain is in an inactive state, this value is the requested number of cryptographic units.
ldomNumIOBus	Integer	Read-only	Number of physical I/O devices used
ldomUUID	Display string	Read-only	UUID of the domain
ldomMacAddress	Display string	Read-only	MAC address of the domain
ldomHostID	Display string	Read-only	Host ID of the domain
ldomFailurePolicy	Display string	Read-only	Master domain's failure policy, which can be one of <code>ignore</code> , <code>panic</code> , <code>reset</code> , or <code>stop</code>
ldomMaster	Display string	Read-only	Name of up to four master domains for a slave domain
ldomExtMapinSpace	Display string	Read-only	Extended mapin space for a domain. The extended mapin space refers to the additional LDC shared memory space. This memory space is required to support a large number of virtual I/O devices that use direct-mapped shared memory. This space is also used by virtual network devices to improve performance and scalability. The default value is <code>off</code> .
ldomWholeCore	Integer	Read-only	Constrains the domain to use whole-cores only. If the whole-core constraint is not enabled, the value is <code>0</code> . Otherwise, the value shows the number of <code>max-cores</code> .

Name	Data Type	Access	Description
<code>ldomCpuArch</code>	Display string	Read-only	<p>CPU architecture for a domain. The CPU architecture specifies whether the domain can be migrated to another sun4v CPU architecture. Valid values are:</p> <ul style="list-style-type: none"> ■ <code>native</code>, which means that the domain is permitted to be migrated only to platforms of the same sun4v CPU architecture (default value) ■ <code>generic</code>, which means that the domain is permitted to be migrated to all compatible sun4v CPU architectures
<code>ldomShutdownGroup</code>	Integer	Read-only	<p>Shutdown-group number for a guest domain. On a Fujitsu M10 server, an SP-initiated ordered shutdown will shut down domains in descending order of their shutdown-group numbers, from 15 to 0. The default value is 15.</p>
<code>ldomPerfCounters</code>	String	Read-only	<p>Performance register access information for a guest domain. Values can be <code>global</code> (on one domain at a time only) and optionally one of the following: <code>htstrand</code> or <code>strand</code>. The default value is <code>htstrand</code>.</p>
<code>ldomNumCMI</code>	Integer	Read-only	<p>Number of CMI resources used. If the domain is in an inactive state, this value is the requested number of CMI resources.</p>
<code>ldomBootPolicy</code>	Display string	Read-only	<p>Verified boot policy for a guest domain, which is one of the following policies:</p> <ul style="list-style-type: none"> ■ <code>none</code> ■ <code>warning</code> ■ <code>enforce</code> <p>Use the Oracle ILOM to set the verified boot policy for a host domain.</p>

Environment Variables Table (`ldomEnvVarsTable`)

`ldomEnvVarsTable` describes the OpenBoot PROM environment variables that all domains use.

TABLE 3 Environment Variables Table (`ldomEnvVarsTable`)

Name	Data Type	Access	Description
<code>ldomEnvVarsLdomIndex</code>	Integer	Read-only	<p>Integer that is used as an index into the <code>ldomTable</code> that represents the domain that</p>

Name	Data Type	Access	Description
			contains the OpenBoot PROM environment variables
ldomEnvVarsIndex	Integer	Read-only	Integer that is used to index the OpenBoot PROM environment variables in this table
ldomEnvVarsName	Display string	Read-only	Name of the OpenBoot PROM variable
ldomEnvVarsValue	Display string	Read-only	Value of the OpenBoot PROM variable

Domain Policy Table (ldomPolicyTable)

ldomPolicyTable describes the dynamic resource management (DRM) policies that apply to all domains.

TABLE 4 Domain Policy Table (ldomPolicyTable)

Name	Data Type	Access	Description
ldomPolicyLdomIndex	Integer	Read-only	Integer that is used as an index into the ldomTable that represents the domain that contains the DRM policy
ldomPolicyIndex	Integer	Not accessible	Integer that is used to index the DRM policy in this table
ldomPolicyName	Display string	Read-only	Policy name
ldomPolicyStatus	Display string	Read-only	Policy status
ldomPolicyPriority	Integer	Read-only	Priority that is used to determine which DRM policy is selected when policies overlap
ldomPolicyVcpuMin	Integer	Read-only	Minimum number of virtual CPUs for a domain
ldomPolicyVcpuMax	Integer	Read-only	Maximum number of virtual CPUs for a domain. A value of unlimited uses the maximum integer value of 2147483647.
ldomPolicyUtilLower	Integer	Read-only	Lower utilization level at which policy analysis is triggered
ldomPolicyUtilUpper	Integer	Read-only	Upper utilization level at which policy analysis is triggered
ldomPolicyTodBegin	Display string	Read-only	Effective start time of a policy with a format of <i>hh:mm:ss</i>
ldomPolicyTodEnd	Display string	Read-only	Effective stop time of a policy with a format of <i>hh:mm:ss</i>
ldomPolicySampleRate	Integer	Read-only	Resource cycle time in seconds
ldomPolicyElasticMargin	Integer	Read-only	Amount of buffer between util-lower property (ldomPolicyUtilLower) and the number of free virtual CPUs to avoid oscillations at low virtual CPU counts

Name	Data Type	Access	Description
ldomPolicyAttack	Integer	Read-only	Maximum amount of a resource to be added during any one resource-control cycle. A value of unlimited uses the maximum integer value of 2147483647.
ldomPolicyDecay	Integer	Read-only	Maximum amount of a resource to be removed during any one resource-control cycle

Service Processor Configuration Table (ldomSPConfigTable)

ldomSPConfigTable describes the service processor (SP) configurations for all domains.

TABLE 5 Service Processor Configuration Table (ldomSPConfigTable)

Name	Data Type	Access	Description
ldomSPConfigIndex	Integer	Not accessible	Integer that is used to index an SP configuration in this table
ldomSPConfigName	Display string	Read-only	SP configuration name
ldomSPConfigStatus	Display string	Read-only	SP configuration status
ldomSPConfigAuxStatus	Display string	Read-only	SP configuration auxiliary status

Domain Resource Pool and Scalar Variables

The following resources can be assigned to domains:

- Virtual CPU (vcpu)
- Memory (mem)
- Cryptographic unit (mau)
- Virtual switch (vsw)
- Virtual network (vnet)
- Virtual disk server (vds)
- Virtual disk server device (vdsdev)
- Virtual disk (vdisk)
- Virtual console concentrator (vcc)

- Virtual console (vcons)
- Physical I/O device (io)
- CMI resources (cmi)

The following scalar MIB variables are used to represent resource pools and their properties.

TABLE 6 Scalar Variables for CPU Resource Pool

Name	Data Type	Access	Description
ldomCpuRpCapacity	Integer	Read-only	Maximum reservation allowed by the resource pool in ldomCpuRpCapacityUnits
ldomCpuRpReserved	Integer	Read-only	Accumulated processor clock speed of the CPU, in MHz, that is currently reserved from the resource pool
ldomCpuRpCapacityUnit and ldomCpuRpReservedUnit	Integer	Read-only	One of the following CPU allocation units: <ul style="list-style-type: none"> ■ 1 is MHz ■ 2 is GHz <p>The default value is MHz.</p>

TABLE 7 Scalar Variables for Memory Resource Pool

Name	Data Type	Access	Description
ldomMemRpCapacity	Integer	Read-only	Maximum reservation allowed by the resource pool in MemRpCapacityUnits
ldomMemRpReserved	Integer	Read-only	Amount of memory, in MemRpReservedUnits, that is currently reserved from the resource pool
ldomMemRpCapacityUnit and ldomMemRpReservedUnit	Integer	Read-only	One of the following memory allocation units: <ul style="list-style-type: none"> ■ 1 is KB ■ 2 is MB ■ 3 is GB ■ 4 is bytes <p>If not specified, the unit value is bytes.</p>

TABLE 8 Scalar Variables for Cryptographic Resource Pool

Name	Data Type	Access	Description
ldomCryptoRpCapacity	Integer	Read-only	Maximum reservation allowed by the resource pool
ldomCryptoRpReserved	Integer	Read-only	Number of cryptographic units that is currently reserved from the resource pool

TABLE 9 Scalar Variables for I/O Bus Resource Pool

Name	Data Type	Access	Description
ldomIOBusRpCapacity	Integer	Read-only	Maximum reservation allowed by the pool
ldomIOBusRpReserved	Integer	Read-only	Number of I/O buses that is currently reserved from the resource pool

TABLE 10 Scalar Variables for CMI Resource Pool

Name	Data Type	Access	Description
ldomCMIRpCapacity	Integer	Read-only	Maximum reservation allowed by the pool
ldomCMIRpReserved	Integer	Read-only	Number of CMI resources that are currently reserved from the resource pool

Virtual CPU Table (ldomVcpuTable)

ldomVcpuTable describes the virtual CPUs that all domains use.

TABLE 11 Virtual CPU Table (ldomVcpuTable)

Name	Data Type	Access	Description
ldomVcpuLdomIndex	Integer	Read-only	Integer that is used as an index into the ldomTable that represents the domain that contains the virtual CPU
ldomVcpuIndex	Integer	Not accessible	Integer that is used to index the virtual CPU in this table
ldomVcpuDeviceID	Display string	Read-only	Identifier of the virtual CPU (VID)
ldomVcpuOperationalStatus	Integer	Read-only	One of the following CPU statuses: 1=Unknown 2=Other 3=OK 4=Degraded 5=Stressed 6=Predictive failure 7=Error

Name	Data Type	Access	Description
			8=Nonrecoverable error
			9=Starting
			10=Stopping
			11=Stopped
			12=In service
			13=No contact
			14=Lost communication
			15=Aborted
			16=Dormant
			17=Supporting entity in error
			18=Completed
			19=Power mode
			The default value is 1 (Unknown) because the Logical Domains Manager does not provide the CPU state.
ldomVcpuPhysBind	Display string	Read-only	Physical binding (PID). Contains the identifier of a hardware thread (strand) that is assigned to this virtual CPU. This identifier uniquely identifies the core and the chip.
ldomVcpuPhysBindUsage	Integer	Read-only	Indicates how much (in MHz) of the total capacity of the thread is used by this virtual CPU. For example, assume a thread can run at a maximum of one GHz. If only half of that capacity is allocated to this virtual CPU (50% of the thread), the property value is 500.
ldomVcpuCoreID	Display string	Read-only	Identifier of the core (core ID).
ldomVcpuUtilPercent	Display string	Read-only	Indicates the utilization percentage of the virtual CPU.

Virtual Memory Tables

A domain's memory space is referred to as *real memory*, that is, *virtual memory*. Host platform memory space that is detected by the hypervisor is referred to as *physical memory*. The hypervisor maps blocks of physical memory to form a block of real memory that is used by a domain.

The following example shows that the requested memory size can be split between two memory blocks instead of being assigned to a single large memory block. Assume that a domain requests 521 Mbytes of real memory. The memory can be assigned two 256-Mbyte blocks on the host system as physical memory by using the *{physical-address, real-address, size}* format.

```
{0x1000000, 0x1000000, 256}, {0x2000000, 0x2000000, 256}
```

A domain can have up to 64 physical memory segments assigned to a guest domain. Therefore, an auxiliary table is used to hold each memory segment instead of a display string. A display string has a 255-character limit.

Virtual Memory Table (`ldomVmemTable`)

`ldomVmemTable` describes the properties of virtual memory that domains use.

TABLE 12 Virtual Memory Table (`ldomVmemTable`)

Name	Data Type	Access	Description
<code>ldomVmemLdomIndex</code>	Integer	Read-only	Integer that is used as an index into the <code>ldomTable</code> that represents the domain that contains the virtual memory
<code>ldomVmemIndex</code>	Integer	Not accessible	Integer that is used to index the virtual memory in this table
<code>ldomVmemNumberOfBlocks</code>	Integer	Read-only	Number of blocks of virtual memory

Virtual Memory Physical Binding Table (`ldomVmemPhysBindTable`)

`ldomVmemPhysBindTable` is an auxiliary table that contains physical memory segments for all domains.

TABLE 13 Virtual Memory Physical Binding Table (`ldomVmemPhysBindTable`)

Name	Data Type	Access	Description
<code>ldomVmemPhysBindLdomIndex</code>	Integer	Read-only	Integer that is used as an index into the <code>ldomTable</code> that represents the domain that contains the physical memory segments
<code>ldomVmemPhysBind</code>	Display string	Read-only	List of physical memory that is mapped to this virtual memory block in the following format: <i>{physical-address, real-address, size}</i>

Virtual Disk Tables

A virtual disk service (`vds`) and the physical device to which it maps (`vdsdev`) provide the virtual disk capability to the Oracle VM Server for SPARC technology. A virtual disk service exports a number of local volumes (physical disks or file systems). When a virtual disk service is specified, the following are included:

- Complete `/dev` path of the backing device (`vdsdev`)
- Unique name (volume name) for the device being added to the service

One or more disks, disk slices, and file systems can be bound to a single disk service. Each disk has a unique name and volume name. The volume name is used when the disk is bound to the service. The Logical Domains Manager creates virtual disk clients (`vdisk`) from the virtual disk service and its logical volumes.

Virtual Disk Service Table (`ldomVdsTable`)

`ldomVdsTable` describes the virtual disk services for all domains.

TABLE 14 Virtual Disk Service Table (`ldomVdsTable`)

Name	Data Type	Access	Description
<code>ldomVdsLdomIndex</code>	Integer	Read-only	Integer that is used as an index into the <code>ldomTable</code> that represents the domain that contains the virtual disk service
<code>ldomVdsIndex</code>	Integer	Not accessible	Integer that is used to index the virtual disk service in this table
<code>ldomVdsServiceName</code>	Display string	Read-only	Service name for the virtual disk service. The property value is the <i>service-name</i> specified by the <code>ldm add-vds</code> command.
<code>ldomVdsNumofAvailVolume</code>	Integer	Read-only	Number of logical volumes exported by this virtual disk service
<code>ldomVdsNumofUsedVolume</code>	Integer	Read-only	Number of logical volumes used (bound) to this virtual disk service

Virtual Disk Service Device Table (`ldomVdsdevTable`)

`ldomVdsdevTable` describes the virtual disk service devices that all virtual disk services use.

TABLE 15 Virtual Disk Service Device Table (`ldomVdsdevTable`)

Name	Data Type	Access	Description
<code>ldomVdsdevVdsIndex</code>	Integer	Read-only	Integer that is used to index into the <code>ldomVdsTable</code> that represents the virtual disk service that contains the virtual disk device
<code>ldomVdsdevIndex</code>	Integer	Not accessible	Integer that is used to index the virtual disk service device in this table
<code>ldomVdsdevVolumeName</code>	Display string	Read-only	Volume name for the virtual disk service device. This property specifies a unique name for the device that is being added to the virtual disk service. This name is exported by the virtual disk service to the clients for the purpose of adding this device. The property value is the <i>volume-name</i> specified by the <code>ldm add -vdsdev</code> command.
<code>ldomVdsdevDevPath</code>	Display string	Read-only	Path name of the physical disk device. The property value is the <i>backend</i> specified by the <code>ldm add -vdsdev</code> command.
<code>ldomVdsdevOptions</code>	Display string	Read-only	One or more of the options for the disk device, which are <code>ro</code> , <code>slice</code> , or <code>excl</code>
<code>ldomVdsdevMPGroup</code>	Display string	Read-only	Multipath group name for the disk device

Virtual Disk Table (`ldomVdiskTable`)

`ldomVdiskTable` describes the virtual disks for all domains.

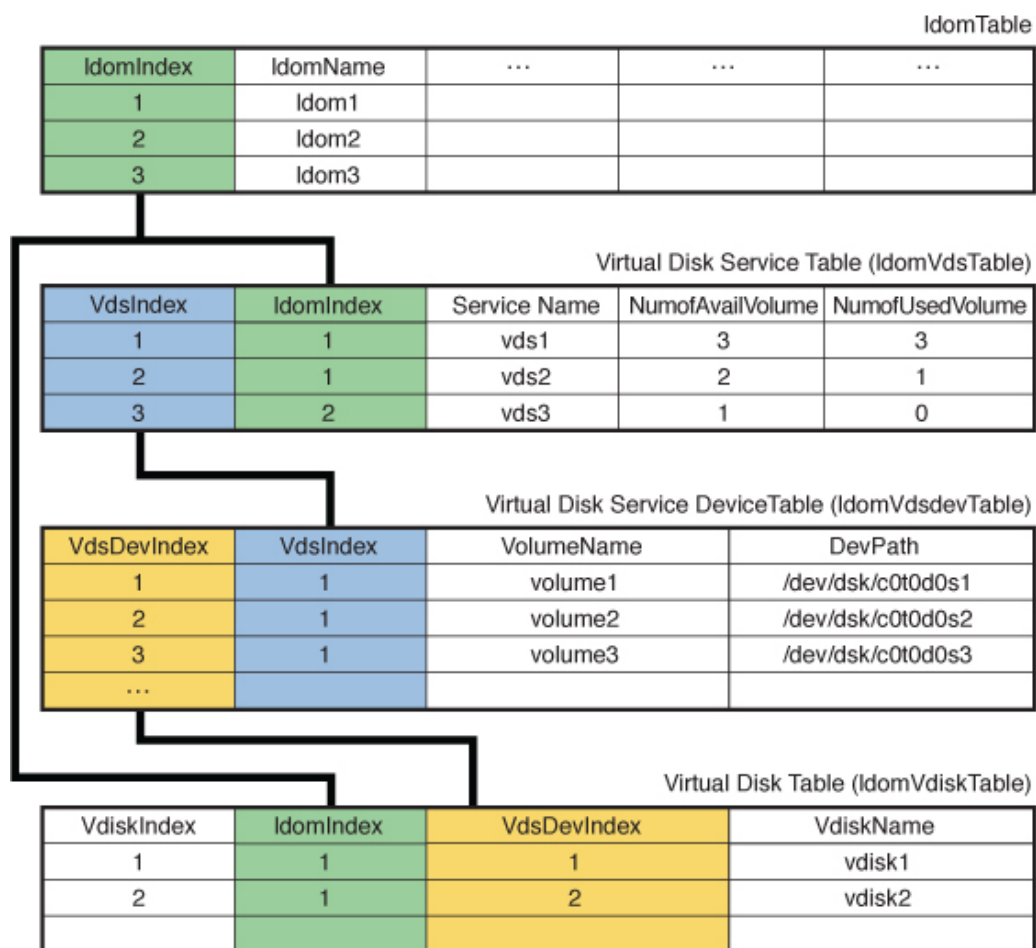
TABLE 16 Virtual Disk Table (`ldomVdiskTable`)

Name	Data Type	Access	Description
<code>ldomVdiskLdomIndex</code>	Integer	Read-only	Integer that is used as an index into the <code>ldomTable</code> that represents the domain that contains the virtual disk device
<code>ldomVdiskVdsDevIndex</code>	Integer	Read-only	Integer that is used to index into the <code>ldomVdsdevTable</code> that represents the virtual disk service device
<code>ldomVdiskIndex</code>	Integer	Not accessible	Integer that is used to index the virtual disk in this table
<code>ldomVdiskName</code>	Display string	Read-only	Name of the virtual disk. The property value is the <i>disk-name</i> specified by the <code>ldm add -vdisk</code> command.
<code>ldomVdiskTimeout</code>	Integer	Read-only	Timeout, in seconds, for establishing a connection between a virtual disk client and a virtual disk server
<code>ldomVdiskID</code>	Display string	Read-only	Identifier of the virtual disk

The following figure shows how indexes are used to define relationships among the virtual disk tables and the domain table. The indexes are used as follows:

- ldomIndex in ldomVdsTable and ldomVdiskTable points to ldomTable.
- VdsIndex in ldomVdsdevTable points to ldomVdsTable.
- VdsDevIndex in ldomVdiskTable points to ldomVdsdevTable.

FIGURE 3 Relationship Among Virtual Disk Tables and the Domain Table



SCSI HBA Tables

These tables cover information about the physical SCSI HBA, the virtual SCSI HBA, and the virtual SAN.

Physical SCSI HBA Table (`ldomHbaTable`)

`ldomHbaTable` describes the physical SCSI HBA services for all domains.

TABLE 17 Physical SCSI HBA Table (`ldomHbaTable`)

Name	Data Type	Access	Description
<code>ldomHbaLdomIndex</code>	Integer	Read-only	Integer that is used as an index into the <code>ldomTable</code> that represents the domain that contains the physical SCSI HBA.
<code>ldomHbaIPortName</code>	Display string	Read-only	Initiator port name of the physical SCSI HBA.
<code>ldomHbaLunCount</code>	Integer	Read-only	Number of LUNs that are associated with the physical SCSI HBA.

Virtual SCSI HBA Device Table (`ldomVhbaTable`)

`ldomVhbaTable` describes the virtual SCSI HBA devices.

TABLE 18 Virtual SCSI HBA Device Table (`ldomVhbaTable`)

Name	Data Type	Access	Description
<code>ldomVhbaLdomIndex</code>	Integer	Read-only	Integer that is used to index into the <code>ldomHbaTable</code> that represents the virtual disk service that contains the virtual disk device.
<code>ldomVhbaName</code>	Display string	Read-only	Name of the virtual SCSI HBA device.
<code>ldomVhbaVsanIndex</code>	Integer	Read-only	Integer that is used as an index into <code>ldomVsanTable</code> .
<code>ldomVhbaTimeout</code>	Integer	Read-only	Timeout, in seconds, for establishing a connection between a virtual SCSI HBA and a virtual SAN.
<code>ldomVhbaId</code>	Integer	Read-only	Identifier of the virtual SCSI HBA device.

Virtual SAN Device Table (`ldomVsanTable`)

`ldomVsanTable` describes the virtual SAN devices for all domains.

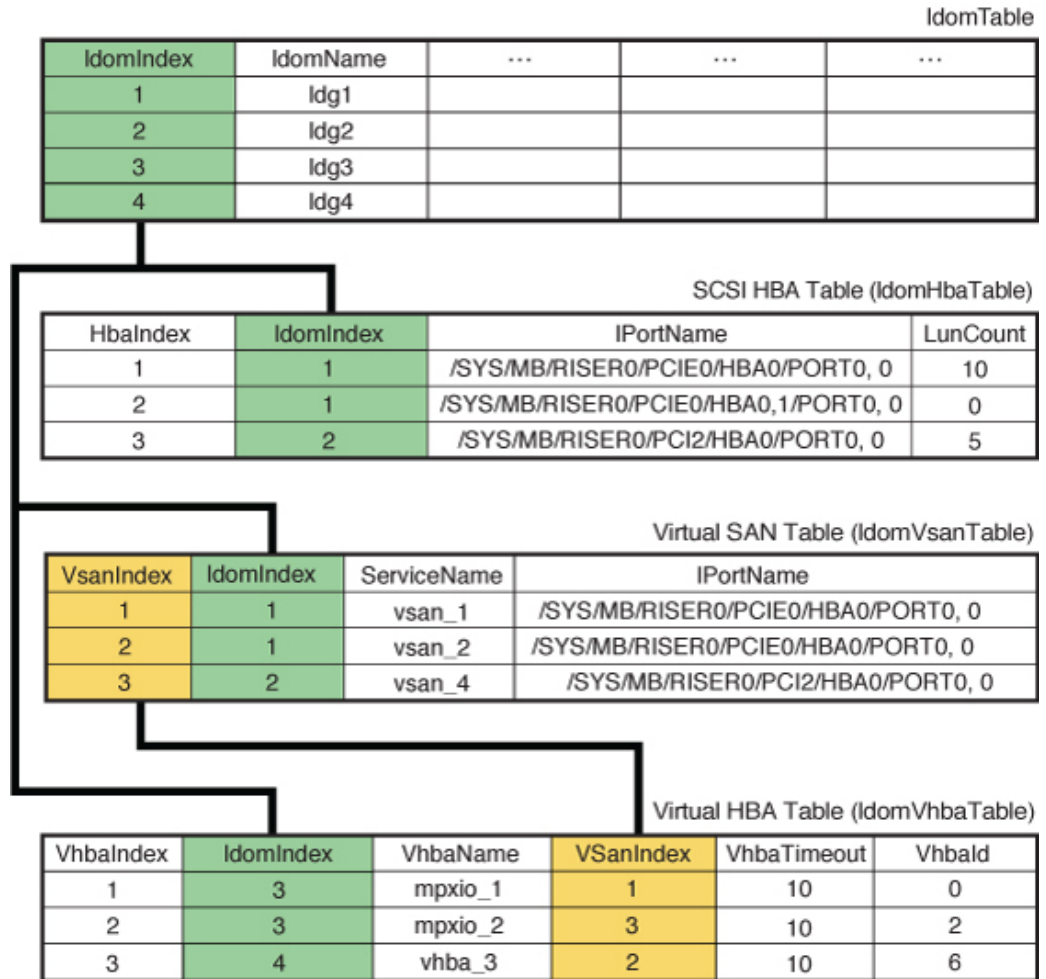
TABLE 19 Virtual SAN Device Table (`ldomVsanTable`)

Name	Data Type	Access	Description
<code>ldomVsanLdomIndex</code>	Integer	Read-only	Integer that is used as an index into the <code>ldomTable</code> that represents the domain that contains the virtual SAN device.
<code>ldomVsanServiceName</code>	Display string	Read-only	Name of the virtual SAN service.
<code>ldomVsanIPortName</code>	Display string	Read-only	Initiator port name of the virtual SAN device.

The following figure shows how indexes are used to define relationships among the SCSI HBA tables and the domain table. The indexes are used as follows:

- `ldomIndex` in `ldomHbaTable`, `ldomVsanTable`, and `ldomVhbaTable` points to `ldomTable`.
- `ldomVsanIndex` in `ldomVhbaTable` points to `VsanIndex` in `ldomVsanTable`.

FIGURE 4 Relationship Among SCSI HBA Tables and the Domain Table



Virtual Network Tables

Oracle VM Server for SPARC virtual network support enables guest domains to communicate with each other and with external hosts through a physical Ethernet device. The virtual network contains the following main components:

- Virtual switch (vsw)
- Virtual network device (vnet)

After you create a virtual switch on a service domain, you can bind a physical network device to the virtual switch. After that, you can create a virtual network device for a domain that uses the virtual switch service for communication. The virtual switch service communicates with other domains by connecting to the same virtual switch. The virtual switch service communicates with external hosts if a physical device is bound to the virtual switch.

Virtual Switch Service Table (ldomVswTable)

ldomVswTable describes the virtual switch services for all domains.

TABLE 20 Virtual Switch Service Table (ldomVswTable)

Name	Data Type	Access	Description
ldomVswLdomIndex	Integer	Read-only	Integer that is used as an index into the ldomTable that represents the domain that contains the virtual switch service
ldomVswIndex	Integer	Not accessible	Integer that is used to index the virtual switch device in this table
ldomVswServiceName	Display string	Read-only	Virtual switch service name
ldomVswMacAddress	Display string	Read-only	MAC address used by the virtual switch
ldomVswPhysDevPath	Display string	Read-only	Physical device path for the virtual network switch. The property value is null when no physical device is bound to the virtual switch.
ldomVswMode	Display string	Read-only	Value is mode=sc for running cluster nodes
ldomVswDefaultVlanID	Display string	Read-only	Default VLAN ID for the virtual switch
ldomVswPortVlanID	Display string	Read-only	Port VLAN ID for the virtual switch
ldomVswVlanID	Display string	Read-only	VLAN ID for the virtual switch
ldomVswLinkprop	Display string	Read-only	Value is linkprop=phys-state to report the link status based on the physical network device
ldomVswMtu	Integer	Read-only	Maximum transmission unit (MTU) for a virtual switch device

Name	Data Type	Access	Description
ldomVswID	Display string	Read-only	Identifier of the virtual switch device
ldomVswInterVnetLink	Display string	Read-only	State of LDC channel assignment for inter-vnet communications. Value is on, off, on/auto, or off/auto.

Virtual Network Device Table (ldomVnetTable)

ldomVnetTable describes the virtual network devices for all domains.

TABLE 21 Virtual Network Device Table (ldomVnetTable)

Name	Data Type	Access	Description
ldomVnetLdomIndex	Integer	Read-only	Integer that is used as an index into the ldomTable that represents the domain that contains the virtual network device
ldomVnetVswIndex	Integer	Read-only	Integer that is used to index into the virtual switch service table
ldomVnetIndex	Integer	Not accessible	Integer that is used to index the virtual network device in this table
ldomVnetDevName	Display string	Read-only	Virtual network device name. The property value is the net-dev property specified by the ldm add-vnet command.
ldomVnetDevMacAddress	Display string	Read-only	MAC address for this network device. The property value is the mac-addr property specified by the ldm add-vnet command.
ldomVnetPortVlanID	Display string	Read-only	Port VLAN ID for the virtual network device
ldomVnetVlanID	Display string	Read-only	Comma-separated list of VLAN IDs for the virtual network device
ldomVnetLinkprop	Display string	Read-only	Value is linkprop=phys-state to report the link status based on the physical network device
ldomVnetMtu	Integer	Read-only	MTU for a virtual network device
ldomVnetID	Display string	Read-only	Identifier of the virtual network device
ldomVnetPvlan	Display string	Read-only	Private VLAN (PVLAN) for the virtual network device
ldomVnetProtection	Display string	Read-only	Types of protection in the form of a bitwise OR of the protection types (mac_nospoof, ip_nospoof, dhcp_nospoof, restricted); default is no protection
ldomVnetAllowedIPs	Display string	Read-only	Comma-separated list of IP addresses
ldomVnetAllowedDhcpCIDs	Display string	Read-only	Comma-separated list of MAC addresses or host names

Name	Data Type	Access	Description
ldomVnetPriority	Display string	Read-only	Relative priority of the link, which is used for packet processing scheduling within the system (high, medium, and low)
ldomVnetCos	Display string	Read-only	Class of service priority that is associated with outbound packets on the link. Values are 0-7 and the default value is 0
ldomVnetCustom	Display string	Read-only	Enabled or disabled custom settings for the maximum number of VLANs and MAC addresses that can be assigned to a virtual network device from a trusted host. The default value is disable
ldomVnetMaxMacAddrs	Display string	Read-only	Maximum number of MAC addresses that can be assigned to a virtual network device from a trusted host. The default value is 4096
ldomVnetMaxVlans	Display string	Read-only	Maximum number of VLANs that can be assigned to a virtual network device from a trusted host. The default value is 4096
ldomVnetMaxBw	Display string	Read-only	Maximum bandwidth limit for the specified port in megabits per second
ldomVnetAltMacAddrs	Display string	Read-only	Comma-separated list of alternate MAC addresses

Virtual Console Tables

The Oracle VM Server for SPARC service domain provides a virtual network terminal service (vNTS). vNTS provides a virtual console service, called a virtual console concentrator (vcc), with a range of port numbers. Each virtual console concentrator has multiple console groups (vcons), and each group is assigned a port number. Each group can contain multiple domains.

Virtual Console Concentrator Table (ldomVccTable)

ldomVccTable describes the virtual console concentrators for all domains.

TABLE 22 Virtual Console Concentrator Table (ldomVccTable)

Name	Data Type	Access	Description
ldomVccLdomIndex	Integer	Read-only	Integer that is used as an index into the ldomTable that represents the domain that contains the virtual console service

Name	Data Type	Access	Description
<code>ldomVccIndex</code>	Integer	Not accessible	Integer that is used to index the virtual console concentrator in this table
<code>ldomVccName</code>	Display string	Read-only	Virtual console concentrator name. The property value is the <i>vcc-name</i> specified by the <code>ldm add-vcc</code> command.
<code>ldomVccPortRangeLow</code>	Integer	Read-only	Low number for the range of TCP ports to be used by the virtual console concentrator. The property value is the <i>x</i> part of the <i>port-range</i> specified by the <code>ldm add-vcc</code> command.
<code>ldomVccPortRangeHigh</code>	Integer	Read-only	High number for the range of TCP ports to be used by the virtual console concentrator. The property value is the <i>y</i> part of the <i>port-range</i> specified by the <code>ldm add-vcc</code> command.

Virtual Console Group Table (`ldomVconsTable`)

`ldomVconsTable` describes the virtual console groups for all virtual console services. This table also shows whether console logging is enabled or disabled on each domain.

TABLE 23 Virtual Console Group Table (`ldomVconsTable`)

Name	Data Type	Access	Description
<code>ldomVconsIndex</code>	Integer	Not accessible	Integer that is used to index a virtual group in this table
<code>ldomVconsGroupName</code>	Display string	Read-only	Group name to which to attach the virtual console. The property value is the <i>group</i> specified by the <code>ldm set-vcons</code> command.
<code>ldomVconsLog</code>	Display string	Read-only	Console logging status. The property value is the string <i>on</i> or <i>off</i> as specified by the <code>ldm set-vcons</code> command. When a group contains more than one domain, this property shows the console logging status of the domain that has most recently been modified by the <code>ldm set-vcons</code> command.
<code>ldomVconsPortNumber</code>	Integer	Read-only	Port number assigned to this group. The property value is the <i>port</i> specified by the <code>ldm set-vcons</code> command.

Virtual Console Relationship Table (`ldomVconsVccRelTable`)

`ldomVconsVccRelTable` contains index values to show the inter-table relationships among a domain, a virtual console concentrator, and console groups.

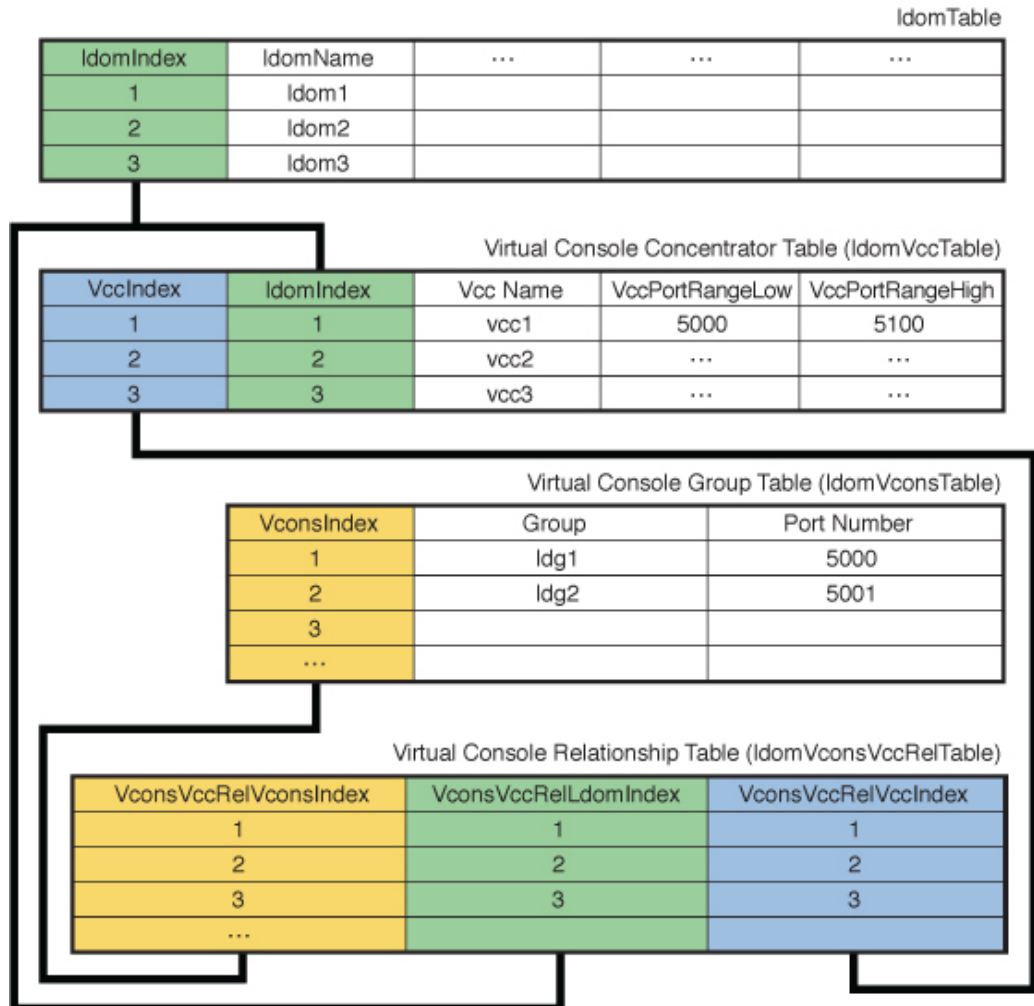
TABLE 24 Virtual Console Relationship Table (`ldomVconsVccRelTable`)

Name	Data Type	Access	Description
<code>ldomVconsVccRelVconsIndex</code>	Integer	Read-only	Value of <code>ldomVconsIndex</code> in <code>ldomVconsTable</code>
<code>ldomVconsVccRelLdomIndex</code>	Integer	Read-only	Value of <code>ldomIndex</code> in <code>ldomTable</code>
<code>ldomVconsVccRelVccIndex</code>	Integer	Read-only	Value of <code>ldomVccIndex</code> in <code>ldomVccTable</code>

The following figure shows how indexes are used to define relationships among the virtual console tables and the domain table. The indexes are used as follows:

- `ldomIndex` in `ldomVccTable` and `ldomVconsVccRelTable` points to `ldomTable`.
- `VccIndex` in `ldomVconsVccRelTable` points to `ldomVccTable`.
- `VconsIndex` in `ldomVconsVccRelTable` points to `ldomVconsTable`.

FIGURE 5 Relationship Among Virtual Console Tables and the Domain Table



Cryptographic Units Table (`ldomCryptoTable`)

`ldomCryptoTable` describes the cryptographic units that all domains use. A cryptographic unit is sometimes referred to as a modular arithmetic unit (MAU).

TABLE 25 Cryptographic Units Table (`ldomCryptoTable`)

Name	Data Type	Access	Description
<code>ldomCryptoLdomIndex</code>	Integer	Read-only	Integer that is used as an index into the <code>ldomTable</code> that represents the domain that contains the cryptographic unit
<code>ldomCryptoIndex</code>	Integer	Not accessible	Integer that is used to index the cryptographic unit in this table
<code>ldomCryptoCpuSet</code>	Display string	Read-only	List of CPUs that is mapped to MAU-unit <code>cpuset</code> . For example, {0, 1, 2, 3}.

I/O Bus Table (`ldomIOBusTable`)

`ldomIOBusTable` describes the physical I/O devices and PCI buses that all domains use.

TABLE 26 I/O Bus Table (`ldomIOBusTable`)

Name	Data Type	Access	Description
<code>ldomIOBusIndex</code>	Integer	Not accessible	Integer that is used to index the I/O bus in this table
<code>ldomIOBusName</code>	Display string	Read-only	Physical I/O device name
<code>ldomIOBusPath</code>	Display string	Read-only	Physical I/O device path
<code>ldomIOBusStatus</code>	Display string	Read-only	Physical I/O device status such as IOV, EMP, OCC and so on
<code>ldomIOBusDomain</code>	Display string	Read-only	Domain name for the I/O device
<code>ldomIOBusType</code>	Display string	Read-only	I/O device type such as BUS, NIU, PF, VF and so on
<code>ldomIOBusAlias</code>	Display string	Read-only	Alias for the I/O device
<code>ldomIOBusBus</code>	Display string	Read-only	Bus for the I/O device

CMI Table (`ldomCMITable`)

`ldomCMITable` describes the CMI resource information for all domains.

TABLE 27 CMI Table (ldomCMITable)

Name	Data Type	Access	Description
ldomCMIldomIndex	Integer	Read-only	Integer that is used as an index into the ldomTable that represents the domain that contains the CMI resource
ldomCMIIndex	Integer	Not accessible	Integer that is used to index the CMI resource in this table
ldomCMIID	Display string	Read-only	Identifier of a CMI resource (CMI ID)
ldomCMICpuSet	Display string	Read-only	List of CPUs that are mapped to the CMI resource
ldomCMICores	Display string	Read-only	List of cores that are mapped to the CMI resource

Core Table (ldomCoreTable)

ldomCoreTable describes the core information, such as core-id and cpuset, for all domains.

TABLE 28 Core Table (ldomCoreTable)

Name	Data Type	Access	Description
ldomCoreLdomIndex	Integer	Read-only	Integer that is used as an index into the ldomTable that represents the domain that contains the core
ldomCoreIndex	Integer	Not accessible	Integer that is used to index a core in this table
ldomCoreID	Display string	Read-only	Identifier of a core (core ID)
ldomCoreCpuSet	Display string	Read-only	List of CPUs that is mapped to the core cpuset

Scalar Variables for Domain Version Information

The Logical Domains Manager protocol supports domain versions, which consists of a major number and a minor number. The Oracle VM Server for SPARC MIB has scalar variables to describe the domain version information.

TABLE 29 Scalar Variables for Domain Version Information

Name	Data Type	Access	Description
ldomVersionMajor	Integer	Read-only	Major version number

Name	Data Type	Access	Description
ldomVersionMinor	Integer	Read-only	Minor version number

The values for `ldomVersionMajor` and `ldomVersionMinor` are equivalent to the version shown by the `ldm list -p` command. For example:

```
$ ldm ls -p
VERSION 1.6
...
```

```
$ snmpget -v1 -c public localhost SUN-LDOM-MIB::ldomVersionMajor.0
SUN-LDOM-MIB::ldomVersionMajor.0 = INTEGER: 1
```

```
$ snmpget -v1 -c public localhost SUN-LDOM-MIB::ldomVersionMinor.0
SUN-LDOM-MIB::ldomVersionMinor.0 = INTEGER: 5
```


◆◆◆ CHAPTER 4

Using SNMP Traps

This chapter describes how to set up your system to send and receive traps. It also describes the traps that you can use to receive change notification for logical domains (domains), as well as other traps that you can use.

The `snmptrapd` daemon does not automatically accept all incoming traps. Instead, the daemon must be configured with authorized SNMP v1 and v2c community strings, with SNMPv3 users, or both. Unauthorized traps or notifications are dropped. See the `snmptrapd.conf(5)` man page on the open source community web site <http://www.net-snmp.org/docs/man/>.

Using Oracle VM Server for SPARC MIB Module Traps

Access control checks are applied to incoming notifications. If `snmptrapd` runs without a suitable configuration file, or with equivalent access control settings, such traps are not processed. See the `snmptrapd.conf(5)` man page on the open source community web site <http://www.net-snmp.org/docs/man/>.

▼ How to Send Traps

1. Configure the trap.

Edit the `/etc/net-snmp/snmp/snmpd.conf` SNMP configuration file to add the directives to define the trap, inform version, and destination.

You must use the `pfedit` command to edit the `/etc/net-snmp/snmp/snmpd.conf` file.

```
trapcommunity string --> define community string to be used when sending traps
trapsink host[community [port]] --> to send v1 traps
trap2sink host[community [port]] --> to send v2c traps
```

```
informsink host[community [port]] --> to send informs
```

For more information, see the `snmpd.conf(5)` man page on the open source community web site <http://www.net-snmp.org/docs/man/>.

For example, the following directives use the `public` string as the community string when sending traps and indicate that the v1 traps are sent to the `localhost` destination:

```
trapcommunity public
trapsink localhost
```

2. Configure access control settings by creating or editing the `/etc/net-snmp/snmp/snmptrapd.conf` SNMP trapd configuration file.

You must use the `pfedit` command to edit the `/etc/net-snmp/snmp/snmpd.conf` file.

The following example shows who is authorized to send traps (`public`) and how incoming traps should be processed (`log,execute,net`). See the `snmptrapd.conf(5)` man page on the open source community web site <http://www.net-snmp.org/docs/man/>.

```
authCommunity log,execute,net public
```

3. To receive SNMP trap messages, start the SNMP trap daemon utility, `snmptrapd`.

Example 4 Sending SNMP v1 and v2c Traps

This example sends send both v1 and v2c traps to the SNMP trap daemon that runs on the same host. Update the `/etc/net-snmp/snmp/snmpd.conf` file with the following directives:

```
trapcommunity public
trapsink localhost
trap2sink localhost
```

▼ How to Receive Traps

● **Start the SNMP trap daemon utility.**

For information about the output format options, see the [snmptrapd\(8\)](#) man page.

The `snmptrapd` utility is an SNMP application that receives and logs SNMP TRAP messages. For example, the following `snmptrapd` command shows that a new domain was created (`ldomTrapDesc = Ldom Created`) with a name of `ldg2` (`ldomName = ldg2`).

```
# /usr/sbin/snmptrapd -f -Le -F \
"TRAP from %B on %m/%L/%y at %h:%j:%k Enterprise=%N Type=%w SubType=%q\n
```

```

with Varbinds: %v\nSecurity info:%P\n\n" localhost:162
NET-SNMP version 5.4.1
TRAP from localhost on 6/27/2012 at 12:13:48
Enterprise=SUN-LDOM-MIB::ldomMIBTraps Type=6 SubType=SUN-LDOM-MIB::ldomCreate
with Varbinds: SUN-LDOM-MIB::ldomIndexNotif = INTEGER: 3
SUN-LDOM-MIB::ldomName = STRING: ldg2 SUN-LDOM-MIB::ldomTrapDesc = STRING:
Ldom Created
Security info:TRAP, SNMP v1, community public

```

Note that the -F option argument string is broken on to two lines for readability purposes.

Oracle VM Server for SPARC MIB Trap Descriptions

This section describes the Oracle VM Server for SPARC MIB traps that you can use.

Domain Creation (ldomCreate)

This trap notifies you when any domains are created.

TABLE 30 Domain Creation Trap (ldomCreate)

Name	Data Type	Description
ldomIndexNotif	Integer	Index into ldomTable
ldomName	Display string	Name of the domain
ldomTrapDesc	Display string	Description of the trap

Domain Destroy (ldomDestroy)

This trap notifies you when any domains are destroyed.

TABLE 31 Domain Destroy Trap (ldomDestroy)

Name	Data Type	Description
ldomIndexNotif	Integer	Index into ldomTable
ldomName	Display string	Name of the domain
ldomTrapDesc	Display string	Description of the trap

Domain State Change (ldomStateChange)

This trap notifies you of any domain operating state changes.

TABLE 32 Domain State Change Trap (ldomStateChange)

Name	Data Type	Description
ldomIndexNotif	Integer	Index into ldomTable
ldomName	Display string	Name of the domain
ldomOperState	Integer	New state of the domain
ldomStatePrev	Integer	Previous state of the domain
ldomTrapDesc	Display string	Description of the trap

Virtual CPU Change (ldomVCpuChange)

This trap notifies you when the number of virtual CPUs in a domain changes.

TABLE 33 Domain Virtual CPU Change Trap (ldomVCpuChange)

Name	Data Type	Description
ldomIndexNotif	Integer	Index into ldomTable
ldomName	Display string	Name of the domain that contains the virtual CPU
ldomNumVCPU	Integer	New number of virtual CPUs for the domain
ldomNumVCPUPrev	Integer	Previous number of virtual CPUs for the domain
ldomTrapDesc	Display string	Description of the trap

Virtual Memory Change (ldomVMemChange)

This trap notifies you when the amount of virtual memory in a domain changes.

TABLE 34 Domain Virtual Memory Change Trap (ldomVMemChange)

Name	Data Type	Description
ldomIndexNotif	Integer	Index into ldomTable

Name	Data Type	Description
ldomName	Display string	Name of the domain that contains the virtual memory
ldomMemSize	Integer	Amount of virtual memory for the domain
ldomMemSizePrev	Integer	Previous amount of virtual memory for the domain
ldomMemUnit	Integer	Memory unit for virtual memory, which is one of the following: <ul style="list-style-type: none"> ■ 1 is KB ■ 2 is MB ■ 3 is GB ■ 4 is bytes <p>If not specified, the unit value is bytes.</p>
ldomMemUnitPrev	Integer	Memory unit for previous virtual memory, which is one of the following: <ul style="list-style-type: none"> ■ 1 is KB ■ 2 is MB ■ 3 is GB ■ 4 is bytes <p>If not specified, the unit value is bytes.</p>
ldomTrapDesc	Display string	Description of the trap

Virtual Disk Service Change (ldomVdsChange)

This trap notifies you when a domain's virtual disk service changes.

TABLE 35 Domain Virtual Disk Service Change Trap (ldomVdsChange)

Name	Data Type	Description
ldomIndexNotif	Integer	Index into ldomTable
ldomName	Display string	Name of the domain that contains the virtual disk service
ldomVdsServiceName	Display string	Name of the virtual disk service that has changed
ldomChangeFlag	Integer	Indicates one of the following changes that occurred to the virtual disk service: <ul style="list-style-type: none"> ■ 1 is Added ■ 2 is Modified ■ 3 is Removed

Name	Data Type	Description
ldomTrapDesc	Display string	Description of the trap

Virtual Disk Change (ldomVdiskChange)

This trap notifies you when a domain's virtual disk changes.

TABLE 36 Virtual Disk Change Trap (ldomVdiskChange)

Name	Data Type	Description
ldomIndexNotif	Integer	Index into ldomTable
ldomName	Display string	Name of the domain that contains the virtual disk device
ldomVdiskName	Display string	Name of the virtual disk device that has changed
ldomChangeFlag	Integer	Indicates one of the following changes that occurred to the virtual disk service: <ul style="list-style-type: none"> ■ 1 is Added ■ 2 is Modified ■ 3 is Removed
ldomTrapDesc	Display string	Description of the trap

SCSI HBA Change (ldomHbaChange)

This trap notifies you when a domain's SCSI HBA changes.

TABLE 37 SCSI HBA Change Trap (ldomHbaChange)

Name	Data Type	Description
ldomIndexNotif	Integer	Index into ldomTable.
ldomName	Display string	Name of the domain that contains the virtual SCSI HBA device.
ldomHbaIPortName	Display string	Name of the SCSI HBA initiator port that has changed.
ldomChangeFlag	Integer	Indicates one of the following changes that occurred to the virtual SCSI HBA: <ul style="list-style-type: none"> ■ 1 is Added

Name	Data Type	Description
ldomTrapDesc	Display string	<ul style="list-style-type: none"> ■ 2 is Removed Description of the trap.

Virtual SCSI HBA Change (ldomVhbaChange)

This trap notifies you when a domain's virtual SCSI HBA changes.

TABLE 38 Virtual SCSI HBA Change Trap (ldomVhbaChange)

Name	Data Type	Description
ldomIndexNotif	Integer	Index into ldomTable.
ldomName	Display string	Name of the domain that contains the virtual SCSI HBA device.
ldomVhbaName	Display string	Name of the virtual SCSI HBA that has changed.
ldomChangeFlag	Integer	Indicates one of the following changes that occurred to the virtual SCSI HBA: <ul style="list-style-type: none"> ■ 1 is Added ■ 2 is Removed
ldomTrapDesc	Display string	Description of the trap.

Virtual SAN Change (ldomVsanChange)

This trap notifies you when a domain's virtual SAN changes.

TABLE 39 Virtual SAN Change Trap (ldomVsanChange)

Name	Data Type	Description
ldomIndexNotif	Integer	Index into ldomTable.
ldomName	Display string	Name of the domain that contains the virtual SCSI HBA device.
ldomVsanServiceName	Display string	Name of the virtual SAN server that has changed.
ldomChangeFlag	Integer	Indicates one of the following changes that occurred to the virtual SCSI HBA: <ul style="list-style-type: none"> ■ 1 is Added

Name	Data Type	Description
ldomTrapDesc	Display string	<ul style="list-style-type: none"> ■ 2 is Removed Description of the trap.

Virtual Switch Change (ldomVswChange)

This trap notifies you when a domain's virtual switch changes.

TABLE 40 Virtual Switch Change Trap (ldomVswChange)

Name	Data Type	Description
ldomIndexNotif	Integer	Index into ldomTable
ldomName	Display string	Name of the domain that contains the virtual switch service
ldomVswServiceName	Display string	Name of the virtual switch service that has changed
ldomChangeFlag	Integer	Indicates one of the following changes that occurred to the virtual switch service: <ul style="list-style-type: none"> ■ 1 is Added ■ 2 is Modified ■ 3 is Removed
ldomTrapDesc	Display string	Description of the trap

Virtual Network Change (ldomVnetChange)

This trap notifies you when a domain's virtual network changes.

TABLE 41 Virtual Network Change Trap (ldomVnetChange)

Name	Data Type	Description
ldomIndexNotif	Integer	Index into ldomTable
ldomName	Display string	Name of the domain that contains the virtual network device
ldomVnetDevName	Display string	Name of the virtual network device for the domain
ldomChangeFlag	Integer	Indicates one of the following changes that occurred to the virtual disk service: <ul style="list-style-type: none"> ■ 1 is Added

Name	Data Type	Description
		<ul style="list-style-type: none"> ■ 2 is Modified ■ 3 is Removed
ldomTrapDesc	Display string	Description of the trap

Virtual Console Concentrator Change (ldomVccChange)

This trap notifies you when a domain's virtual console concentrator changes.

TABLE 42 Virtual Console Concentrator Change Trap (ldomVccChange)

Name	Data Type	Description
ldomIndexNotif	Integer	Index into ldomTable
ldomName	Display string	Name of the domain that contains the virtual console concentrator
ldomVccName	Display string	Name of the virtual console concentrator service that has changed
ldomChangeFlag	Integer	Indicates one of the following changes that occurred to the virtual console concentrator: <ul style="list-style-type: none"> ■ 1 is Added ■ 2 is Modified ■ 3 is Removed
ldomTrapDesc	Display string	Description of the trap

Virtual Console Group Change (ldomVconsChange)

This trap notifies you when a domain's virtual console group changes.

TABLE 43 Virtual Console Group Change Trap (ldomVconsChange)

Name	Data Type	Description
ldomIndexNotif	Integer	Index into ldomTable
ldomName	Display string	Name of the domain that contains the virtual console group
ldomVconsGroupName	Display string	Name of the virtual console group that has changed

Name	Data Type	Description
ldomChangeFlag	Integer	Indicates one of the following changes that occurred to the virtual console group: <ul style="list-style-type: none"> ■ 1 is Added ■ 2 is Modified ■ 3 is Removed
ldomTrapDesc	Display string	Description of the trap

CMI Resource Change (ldomCMChange)

This trap notifies you when the number of CMI resources in a domain changes.

TABLE 44 CMI Resource Change Trap (ldomCMChange)

Name	Data Type	Description
ldomIndexNotif	Integer	Index into ldomTable
ldomName	Display string	Name of the domain that contains the CMI resource
ldomNumCMI	Integer	New number of CMI resources for the domain
ldomNumCMIPrev	Integer	Previous number of CMI resources for the domain
ldomTrapDesc	Display string	Description of the trap

Starting and Stopping Domains

This chapter describes the active management operations that you use to stop and start domains. You can control these active management operations by setting a value for the `ldomAdminState` property of the Domain Table, `ldomTable`. See [Table 2, “Domain Table \(`ldomTable`\),” on page 24](#).

Starting and Stopping Logical Domains

▼ How to Start a Domain

This procedure describes how to start an existing bound domain. If a domain with the specified domain name does not exist or is not already bound, this operation fails.

1. **Verify that the *domain-name* domain exists and is bound.**

```
# ldm list domain-name
```

2. **Identify *domain-name* in `ldomTable`.**

```
# snmpwalk -v1 -c public localhost SUN-LDOM-MIB::ldomTable
SUN-LDOM-MIB::ldomName.1 = STRING: primary
SUN-LDOM-MIB::ldomName.2 = STRING: LdomMibTest_1
SUN-LDOM-MIB::ldomAdminState.1 = INTEGER: 0
SUN-LDOM-MIB::ldomAdminState.2 = INTEGER: 0
SUN-LDOM-MIB::ldomOperState.1 = INTEGER: active(1)
SUN-LDOM-MIB::ldomOperState.2 = INTEGER: bound(6)
SUN-LDOM-MIB::ldomNumVCpu.1 = INTEGER: 8
SUN-LDOM-MIB::ldomNumVCpu.2 = INTEGER: 4
SUN-LDOM-MIB::ldomMemSize.1 = INTEGER: 3360
```

```

SUN-LDOM-MIB::ldomMemSize.2 = INTEGER: 256
SUN-LDOM-MIB::ldomMemUnit.1 = INTEGER: megabytes(2)
SUN-LDOM-MIB::ldomMemUnit.2 = INTEGER: megabytes(2)
SUN-LDOM-MIB::ldomNumCrypto.1 = INTEGER: 1
SUN-LDOM-MIB::ldomNumCrypto.2 = INTEGER: 0
SUN-LDOM-MIB::ldomNumIOBus.1 = INTEGER: 2
SUN-LDOM-MIB::ldomNumIOBus.2 = INTEGER: 0
SUN-LDOM-MIB::ldomUUID.1 = STRING: 5f8817d4-5d2e-6f7d-c4af-91b5b34b5723
SUN-LDOM-MIB::ldomUUID.2 = STRING: 11284146-87ca-4877-8d80-cd0f60d5ec26
SUN-LDOM-MIB::ldomMacAddress.1 = STRING: 00:14:4f:46:47:d6
SUN-LDOM-MIB::ldomMacAddress.2 = STRING: 00:14:4f:f8:d5:6c
SUN-LDOM-MIB::ldomHostID.1 = STRING: 0x844647d6
SUN-LDOM-MIB::ldomHostID.2 = STRING: 0x84f8d56c
SUN-LDOM-MIB::ldomFailurePolicy.1 = STRING: ignore
SUN-LDOM-MIB::ldomFailurePolicy.2 = STRING: ignore
SUN-LDOM-MIB::ldomMaster.1 = STRING:
SUN-LDOM-MIB::ldomMaster.2 = STRING:
SUN-LDOM-MIB::ldomExtMapinSpace.1 = STRING: off
SUN-LDOM-MIB::ldomExtMapinSpace.2 = STRING: off
SUN-LDOM-MIB::ldomWholeCore.1 = INTEGER: 0
SUN-LDOM-MIB::ldomWholeCore.2 = INTEGER: 0
SUN-LDOM-MIB::ldomCpuArch.1 = STRING: native
SUN-LDOM-MIB::ldomCpuArch.2 = STRING: native
SUN-LDOM-MIB::ldomShutdownGroup.1 = INTEGER: 0
SUN-LDOM-MIB::ldomShutdownGroup.2 = INTEGER: 15
SUN-LDOM-MIB::ldomPerfCounters.1 = STRING: htstrand
SUN-LDOM-MIB::ldomPerfCounters.2 = STRING: global,htstrand
SUN-LDOM-MIB::ldomNumCMI.1 = INTEGER: 0
SUN-LDOM-MIB::ldomNumCMI.2 = INTEGER: 0
SUN-LDOM-MIB::ldomBootPolicy.1 = STRING: n/a
SUN-LDOM-MIB::ldomBootPolicy.2 = STRING: warning

```

3. Start the *domain-name* domain.

Use the `snmpset` command to start the domain by setting a value of 1 to the `ldomAdminState` property. *n* specifies the domain to start.

```

# snmpset -v version -c community-string hostname \
SUN-LDOM-MIB::ldomTable.1.ldomAdminState.n = 1

```

4. Verify that the *domain-name* domain is active by using one of the following commands:

- # `ldm list domain-name`
- # `snmpget -v version -c community-string hostname SUN-LDOM-MIB::ldomOperState.n`

Example 5 Starting a Guest Domain

This example verifies that the `LdomMibTest_1` domain exists and is bound before setting the `ldomAdminState` property to 1. Finally, the `ldm list LdomMibTest_1` command verifies that the `LdomMibTest_1` domain is active.

```
# ldm list LdomMibTest_1
# snmpset -v1 -c private localhost SUN-LDOM-MIB::ldomTable.1.ldomAdminState.2 = 1
# ldm list LdomMibTest_1
```

You can also use the `snmpget` command to retrieve the `LdomMibTest_1` domain's state instead of using the `ldm list` command.

```
# snmpget -v1 -c public localhost SUN-LDOM-MIB::ldomOperState.2
```

Note that if the domain is inactive when you use `snmpset` to start the domain, the domain is first bound and then started.

▼ How to Stop a Domain

This procedure describes how to stop a started domain. Any operating system instances that are hosted by the domain are stopped.

1. Identify *domain-name* in `ldomTable`.

```
# snmpwalk -v1 -c public localhost SUN-LDOM-MIB::ldomTable
SUN-LDOM-MIB::ldomName.1 = STRING: primary
SUN-LDOM-MIB::ldomName.2 = STRING: LdomMibTest_1
SUN-LDOM-MIB::ldomAdminState.1 = INTEGER: 0
SUN-LDOM-MIB::ldomAdminState.2 = INTEGER: 0
SUN-LDOM-MIB::ldomOperState.1 = INTEGER: active(1)
SUN-LDOM-MIB::ldomOperState.2 = INTEGER: bound(6)
SUN-LDOM-MIB::ldomNumVCpu.1 = INTEGER: 8
SUN-LDOM-MIB::ldomNumVCpu.2 = INTEGER: 4
SUN-LDOM-MIB::ldomMemSize.1 = INTEGER: 3360
SUN-LDOM-MIB::ldomMemSize.2 = INTEGER: 256
SUN-LDOM-MIB::ldomMemUnit.1 = INTEGER: megabytes(2)
SUN-LDOM-MIB::ldomMemUnit.2 = INTEGER: megabytes(2)
SUN-LDOM-MIB::ldomNumCrypto.1 = INTEGER: 1
SUN-LDOM-MIB::ldomNumCrypto.2 = INTEGER: 0
SUN-LDOM-MIB::ldomNumIOBus.1 = INTEGER: 2
SUN-LDOM-MIB::ldomNumIOBus.2 = INTEGER: 0
SUN-LDOM-MIB::ldomUUID.1 = STRING: 5f8817d4-5d2e-6f7d-c4af-91b5b34b5723
SUN-LDOM-MIB::ldomUUID.2 = STRING: 11284146-87ca-4877-8d80-cd0f60d5ec26
```

```

SUN-LDOM-MIB::ldomMacAddress.1 = STRING: 00:14:4f:46:47:d6
SUN-LDOM-MIB::ldomMacAddress.2 = STRING: 00:14:4f:f8:d5:6c
SUN-LDOM-MIB::ldomHostID.1 = STRING: 0x844647d6
SUN-LDOM-MIB::ldomHostID.2 = STRING: 0x84f8d56c
SUN-LDOM-MIB::ldomFailurePolicy.1 = STRING: ignore
SUN-LDOM-MIB::ldomFailurePolicy.2 = STRING: ignore
SUN-LDOM-MIB::ldomMaster.1 = STRING:
SUN-LDOM-MIB::ldomMaster.2 = STRING:
SUN-LDOM-MIB::ldomExtMapinSpace.1 = STRING: off
SUN-LDOM-MIB::ldomExtMapinSpace.2 = STRING: off
SUN-LDOM-MIB::ldomWholeCore.1 = INTEGER: 0
SUN-LDOM-MIB::ldomWholeCore.2 = INTEGER: 0
SUN-LDOM-MIB::ldomCpuArch.1 = STRING: native
SUN-LDOM-MIB::ldomCpuArch.2 = STRING: native
SUN-LDOM-MIB::ldomShutdownGroup.1 = INTEGER: 0
SUN-LDOM-MIB::ldomShutdownGroup.2 = INTEGER: 15
SUN-LDOM-MIB::ldomPerfCounters.1 = STRING: htstrand
SUN-LDOM-MIB::ldomPerfCounters.2 = STRING: global,htstrand
SUN-LDOM-MIB::ldomNumCMI.1 = INTEGER: 0
SUN-LDOM-MIB::ldomNumCMI.2 = INTEGER: 0
SUN-LDOM-MIB::ldomBootPolicy.1 = STRING: n/a
SUN-LDOM-MIB::ldomBootPolicy.2 = STRING: warning

```

2. Stop the *domain-name* domain.

Use the `snmpset` command to stop the domain by setting a value of 2 to the `ldomAdminState` property. *n* specifies the domain to stop.

```

# snmpset -v version -c community-string hostname \
SUN-LDOM-MIB::ldomTable.1.ldomAdminState.n = 2

```

3. Verify that the *domain-name* domain is bound by using one of the following commands:

- # `ldm list domain-name`
- # `snmpget -v version -c community-string hostname SUN-LDOM-MIB::ldomOperState.n`

Example 6 Stopping a Guest Domain

This example sets the `ldomAdminState` property to 2 to stop the guest domain and then uses the `ldm list LdomMibTest_1` command to verify that the `LdomMibTest_1` domain is bound.

```

# snmpset -v1 -c private localhost SUN-LDOM-MIB::ldomTable.1.ldomAdminState.2 = 2
# ldm list LdomMibTest_1

```