Dialogic®
Making Innovation Thrive™

Dialogic® DSI Protocol Stacks

SNMP User Manual

July 2010 U04DPK

www.dialogic.com

Copyright and Legal Notice

All contents of this document are furnished for informational use only and are subject to change without notice and do not represent a commitment on the part of Dialogic Corporation or its subsidiaries ("Dialogic"). Reasonable effort is made to ensure the accuracy of the information contained in the document. However, Dialogic does not warrant the accuracy of this information and cannot accept responsibility for errors, inaccuracies or omissions that may be contained in this document.

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH DIALOGIC® PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN A SIGNED AGREEMENT BETWEEN YOU AND DIALOGIC, DIALOGIC ASSUMES NO LIABILITY WHATSOEVER, AND DIALOGIC DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF DIALOGIC PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY INTELLECTUAL PROPERTY RIGHT OF A THIRD PARTY.

Dialogic products are not intended for use in medical, life saving, life sustaining, critical control or safety systems, or in nuclear facility applications.

Due to differing national regulations and approval requirements, certain Dialogic products may be suitable for use only in specific countries, and thus may not function properly in other countries. You are responsible for ensuring that your use of such products occurs only in the countries where such use is suitable. For information on specific products, contact Dialogic Corporation at the address indicated below or on the web at *www.dialogic.com*.

It is possible that the use or implementation of any one of the concepts, applications, or ideas described in this document, in marketing collateral produced by or on web pages maintained by Dialogic may infringe one or more patents or other intellectual property rights owned by third parties. Dialogic does not provide any intellectual property licenses with the sale of Dialogic products other than a license to use such product in accordance with intellectual property owned or validly licensed by Dialogic and no such licenses are provided except pursuant to a signed agreement with Dialogic. More detailed information about such intellectual property is available from Dialogic's legal department at 9800 Cavendish Blvd., 5th Floor, Montreal, Quebec, Canada H4M 2V9. **Dialogic encourages all users of its products to procure all necessary intellectual property licenses required to implement any concepts or applications and does not condone or encourage any intellectual property infringement and disclaims any responsibility related thereto. These intellectual property licenses may differ from country to country and it is the responsibility of those who develop the concepts or applications to be aware of and comply with different national license requirements.**

This document discusses one or more open source products, systems and/or releases. Dialogic is not responsible for your decision to use open source in connection with Dialogic products (including without limitation those referred to herein), nor is Dialogic responsible for any present or future effects such usage might have, including without limitation effects on your products, your business, or your intellectual property rights.

Any use case(s) shown and/or described herein represent one or more examples of the various ways, scenarios or environments in which Dialogic® products can be used. Such use case(s) are non-limiting and do not represent recommendations of Dialogic as to whether or how to use Dialogic products.

Publication Date: July 2010

Document Number: U04DPK, Issue 2

Contents

Figures

Revision History

Issue	Date	Description
1	May 2010	Manual created.
2	July 2010	Addition of support for Solaris.

Note: The current version of this guide can be found at:
http://www.dialogic.com/support/helpweb/signaling

1 Introduction

1.1 Overview

This document describes the operation and capabilities of Simple Network Management Protocol (SNMP) support for Dialogic® Distributed Signaling Interface (DSI) Components, including Dialogic® DSI SS7 Boards and Dialogic® DSI Protocol Stacks.

The SNMP functionality described here is collectively referred to as Dialogic® Distributed Structured Management Information (DSMI) SNMP.

DSMI SNMP incorporates SNMP Management Information Base (MIB) files, DSI Component software, DSMI SNMP Sub-Agent (DSA) software, and third-party SNMP Agent software.

Information provided by DSMI SNMP is classified into the following object groups:

- Boards
- SS7
- SIGTRAN

Each of these object groups comprises one or more objects. These objects, together with the object groups, are defined in separate SNMP Management Information Base (MIB) definition files.

Dialogic® DSMI SNMP can optionally provide monitoring capabilities to the following DSI Components:

- Dialogic® DSI SS7 Boards
  - SPCI, SS7HD, SS7MD
- Dialogic® DSI Protocol Stacks
  - MTP3, M3UA

This implementation makes use of a third party SNMP agent, which is available free from Net-SNMP and which the user will need to install as described later in this manual.

Net-SNMP supports SNMP versions 1 (RFC1157), 2c (RFC1901), and 3 (RFC2571).

The DSMI SNMP Sub-Agent software binary (DSA), distributed within the Dialogic® DSI Development Package, the DSA binary is subject to a third party software license agreement; for details refer to section 6 of this manual.

DSMI SNMP provides status information about various aspects of DSI component behavior. This implementation supports SNMP ‘read’ (SNMP GET) requests from SNMP managers.

DSMI SNMP also implements SNMP TRAP/NOTIFY events, alerting SNMP manager software to various conditions that the agent has detected. Up to 32 SNMP managers can be defined to receive TRAP notifications. These managers can be configured to receive TRAP notifications for supported DSMI objects.

1.2 Applicability

This document is applicable to the following:

- Dialogic® DSI SS7 Development Package
- Dialogic® DSI SS7 Boards
- Dialogic® DSI SS7 Protocol Stacks
- Dialogic® DSI SIGTRAN Protocol Stacks

This manual is not applicable to the Dialogic® DSI SS7G3x Signaling Servers; users of such products should refer to the *Dialogic® Signaling Server SNMP User Manual*.

1.3 Related Documentation

- *Dialogic® DSI SS7 Protocols MTP3 Programmer's Manual*
- *Dialogic® DSI Signaling Protocols M3UA Programmer's Manual*
- *Dialogic® DSI SS7HD Network Interface Boards Programmer's Manual*
- *Dialogic® DSI SPCI Network Interface Boards Programmer's Manual*
- *Dialogic® DSI SS7MD Network Interface Boards Programmer's Manual*

Current software and documentation supporting Dialogic® DSI components is available at:

http://www.dialogic.com/support/helpweb/signaling

Net-SNMP software and documentation is available at:

http://www.net-snmp.org

2 Installation

2.1 Overview

Providing an SNMP solution using DSMI SNMP requires the following software to be installed:

- Dialogic® DSI Development Package
- Net-SNMP

2.2 Development Package Installation

The DSMI SNMP Sub-Agent binary (DSA) is delivered as part of the standard Development Package installation.

The development package contains the binaries which support DSMI SNMP for the DSI SS7 Board products. DSI SS7 Protocol modules are available online as independent software downloads.

Instructions for installing the Development Package are provided in the manuals of supported DSI Components, including Dialogic® DSI SS7 Boards and the DSI SIGTRAN Protocol Stack. This information is available online - see section 1.3 Related Documentation.

2.3 Software Installation for Linux

Net-SNMP is a prerequisite of DSMI SNMP. The Net-SNMP software is available for Linux under BSD-style licenses. See section 3 for License information.

2.3.1 Net-SNMP Agent Software

Net-SNMP software packages are commonly available as part of Linux distribution software repositories.

The following versions of Net-SNMP are supported by DSMI SNMP for Linux:

- Net-SNMP: version 5.1.4 and later.

It is recommended that the user should, if available, make use of supported versions of Net-SNMP available through a distributions software repository. These can be installed using a supported package management tool.

If the minimum version requirements are not met by a distributions software repository offering, the software can be installed from source software packages available online, see section 1.3 Related Documentation.

Installation Using a RPM-based Tool

For RPM-based Linux distributions, it is recommended that the user installs Net-SNMP through a RedHat Package Management tool such as Yum.

Full documentation for the use of the ‘yum’ tool is available from:

yum.baseurl.org

Example installation routine using the 'yum' installation tool:

Note: The package names referred to in this section are examples; package names may differ depending on OS Distributions and Package Versions. The package search command 'yum search snmp' can be used to identify the relevant packages.

The installation must be performed by a user with Administrator privileges.

1. Installation of the Net-SNMP package:

   ```
   $yum install net-snmp
   ```

2. Optional installation of Net-SNMP utilities to manage Net-SNMP:

   ```
   $yum install net-snmp-utils
   ```

3. Check Net-SNMP installation:

   ```
   $/usr/sbin/snmpd –v
   ```

Note: The above command assumes the default Net-SNMP package installation location. Binaries may be installed to alternative locations depending on the Operating system. Please refer to package documentation.

Example output:

```
$/usr/sbin/snmpd -v
NET-SNMP version:  5.5
Web:               http://www.net-snmp.org/
Email:             net-snmp-coders@lists.sourceforge.net
```

Source Installation

The required Net-SNMP software packages are available directly from Net-SNMP websites in the form of compressed source code packages.

Documentation and supporting manuals for Net-SNMP are available from the Net-SNMP website - see section 1.3 Related Documentation.

Instructions:

These instructions are provided with the assumption that the target build machine has 'gcc' and 'make' software packages installed. The installation must be performed by a user with Administrator privileges.

1. Download a supported source package from the Net-SNMP website:

   e.g., `net-snmp-5.5.tar.gz`

2. Uncompress the package:

   ```
   $tar –zxvf <filename>.tar.gz
   ```

3. Read the INSTALL file for additional instructions and custom compilation options.

4. Configure the build environment.

   Default configuration:

   ```
   $./configure
   ```

5. Build the package:

   ```
   $make
   ```

6. Install the package:

```
$make install
```

The Net-SNMP software can be checked by running the snmpd agent binary with the –v switch option.

Controlling Net-SNMP agent

File locations described here are based on default locations for Net-SNMP installations. File locations may vary depending on the version of Net-SNMP installed and/or compile time options.

The Net-SNMP daemon binary (snmpd) must be running before the DSMI SNMP Sub-Agent or DSI Component software is started. The software may be started using the service control mechanism:

```
/sbin/service snmpd [start|stop|restart|status]
```

The binary is typically installed to the following location:

```
/usr/sbin/snmpd
```

The default Agent configuration file location:

```
/etc/snmp/snmpd.conf
or
~/.snmp/snmpd.conf
```

To enable automatic run at startup:

```
/sbin/chkconfig –-level 345 snmpd on
```

To disabled automatic run at startup:

```
/sbin/chkconfig –-level 345 snmpd off
```

Alternative installations of Net-SNMP may require the manual calling of the snmpd binary. In this scenario, issue the –C switch command to prevent additional configuration files from being read, and the –c switch command to specify a specific configuration file. No other switch options are required for DSMI SNMP operation.

Example:

```
$/home/user/bin/snmpd –C –c /etc/snmp/snmpd.conf
```

2.4 Software Installation for Solaris

Net-SNMP is a prerequisite of DSMI SNMP. The Net-SNMP software is available for Solaris under BSD-style licenses. See section 6 for License information.

Solaris installations may feature additional SNMP agent software. The Sun Solstice Enterprise Master Agent (snmpdx) and/or the Net-SNMP based System Management Agent (snmpd) may be installed. In order to use DSMI SNMP, a separate installation of Net-SNMP is required.

This installation may be configured to work with or replace the aforementioned SNMP agent software.

2.4.1 Net-SNMP Agent Software

The following versions of Net-SNMP are supported by DSMI SNMP for Solaris:

- Net-SNMP: version 5.1.4 and later.

Source Installation

The required Net-SNMP software packages are available directly from the Net-SNMP website, http://www.net-snmp.org, in the form of compressed source code packages. Documentation for Net-SNMP software is also available from this website.

Instructions:

The following instructions assume that the target build machine has the ‘gcc’ and ‘make’ packages installed. The installation must be performed by a user with Administrator privileges.

1. Download a source package from Net-SNMP website:

   e.g., `net-snmp-5.5.tar.gz`

2. Uncompress the gzip package:

   ```
   $gzip -dc <filename>.tar.gz | tar xvf -
   ```

3. Read the INSTALL and README.solaris files for additional instructions and custom compilation options.

4. Ensure that the PATH environment contains required binary locations for your build environment:

   e.g., `PATH=/usr/sbin:/usr/local/bin:/usr/ccs/bin:/usr/bin:`

5. Configure the build environment. The default configuration is:

   ```
   $./configure –with-cc=gcc
   ```

6. Build the package:

   ```
   $make
   ```

7. Install the package:

   ```
   $make install
   ```

Binary Package Installation

Pre-built binary packages of Net-SNMP for both Solaris x86 and Solaris SPARC platforms are available through the following third-party website:

www.sunfreeware.com

The installation must be performed by a user with Administrator privileges.

The process for Installation of Sun packages is as follows:

1. Download the package. For example:

   ```
   netsnmp-5.4.2.1-sol10-sparc-local.gz
   ```

2. Check the prerequisites detailed for the selected package, and install them if required.

3. Uncompress the package:

   ```
   $gunzip <package_name>
   ```

4. Install the package:

   ```
   $pkgadd –d <package_name>
   ```

Controlling SNMP Agent Software

To disable the Sun Solstice Enterprise Master Agent:

```
$svcadm disable svc:/application/management/snmpdx:default
```

To disable the Sun System Management Agent:

```
$svcadm disable svc:/application/management/sma:default
```

Default Net-SNMP binary location:

```
/usr/local/sbin/snmpd
```

Default Net-SNMP configuration file:

```
/usr/local/share/snmp/snmpd.conf
```

Launching Net-SNMP:

```
$/usr/local/sbin/snmpd –C –c <configuration_file>
```

The –C option prevents the Net-SNMP Agent from reading additional configuration files located on the system. The –c option is used to specify a specific configuration file.

Note: Two SNMP agents cannot listen on the same TCP/UDP port. Therefore, if operating in conjunction with either the Solstice Enterprise Agent or the System Management Agent, the Net-SNMP agent must be configured on an alternative port. Refer to section 3.2 for more information.

2.5 Software Installation for Windows®

Net-SNMP is prerequisite of DSMI SNMP Sub-Agent. This software is available for Windows® platforms under BSD-style licenses. These are packaged into a single installation binary provided by Net. See section 6 for License information.

2.5.1 Net-SNMP Agent Software

The prerequisites are available as precompiled binaries and a source for compilation. It is recommended that the user installs Net-SNMP from the precompiled binary installers provided by Net-SNMP.

The following versions of Net-SNMP are supported by DSMI SNMP for Windows:

- Net-SNMP: version 5.1.4 and later.

Installation Using Precompiled Binaries

The installation must be performed by a user with Administrator privileges. Before performing the installation, close all other applications.

1. Obtain Net-SNMP package:

   Software packages are available online - see section 1.3 Related Documentation.

   For both 32bit and 64bit operating systems, select the 32bit package, example:

   e.g., net-snmp.5.x.x-x.x86.exe

2. Installation of the Net-SNMP package:

   Run the downloaded installation binary. The installation will take the user through the license agreement and provide a component choice dialog.

   Select the following components for installation:

   - Base Component
   - Net-SNMP Agent Service
     - Standard Agent
   - Net-SNMP Trap Services

   For SNMPv3 support, select the following additional component:

   - Encryption support

Controlling Net-SNMP agent

Default Net-SNMP binary location:

C:\usr\bin\snmpd

Default Configuration Location:

C:\usr\etc\snmp\snmpd.conf

Net-SNMP Control:

Manage the Net-SNMP Agent through the Service control:

Control Panel -> Administrative Tools -> Services

From the Services panel, start and stop the 'Net-SNMP Agent' software.

3 DSMI SNMP Configuration

3.1 Overview

DSMI SNMP is based on a number of DSMI enabled DSI Component modules associated with DSI Signaling Boards and Protocol Stacks.

The DSMI enabled components provide status and event information to the DSMI SNMP Sub-Agent software (DSA).

The DSA module implements an SNMP network interface through the third party Net-SNMP software package, providing GET and TRAP notification support to external SNMP managers.

The diagram below illustrates the high level architecture of the DSMI SNMP solution.

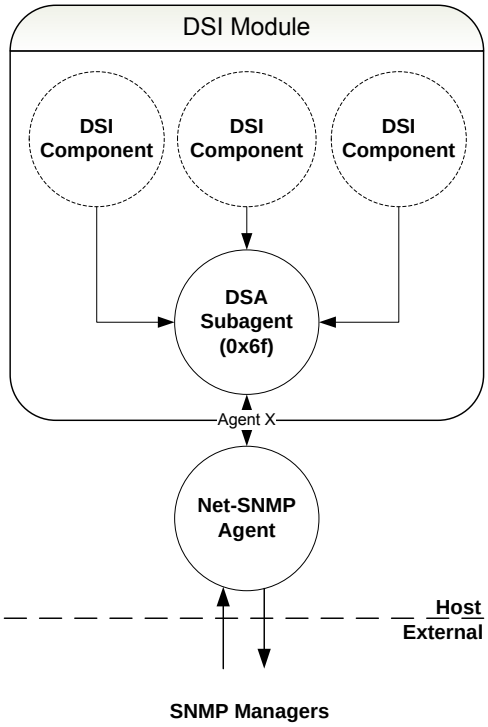


Figure 1 - High Level DSMI SNMP Architecture

The following sections describe configuration of the following components:

- Net-SNMP Agent (snmpd)
- DSMI SNMP Sub-Agent (dsa)
- DSI Components (e.g., ssds, mtp, m3ua)

3.2 Net-SNMP Configuration

This section describes the process of configuring Net-SNMP for DSMI SNMP and provides an example of supporting an additional third party SNMP Agent.

There are alternative architectural deployment possibilities with the Net-SNMP software which are beyond the scope of this document. The user can refer to the Net-SNMP documentation - see section 1.3 Related Documentation.

3.2.1 Description

Net-SNMP is a third party software package used to implement SNMP v1, SNMP v2c and SNMP v3 using IPv4. Net-SNMP is required to provide an external SNMP interface to the DSMI SNMP Sub-Agent, and associated DSI components.

Net-SNMP can be configured to provide an interface exclusively to DSMI SNMP Sub-Agent, or used in conjunction with other SNMP Agents and/or Sub-Agents.

3.2.2 Configuration Files

Configuration of the Net-SNMP agent is achieved through text-based configuration files. Entries (text lines) within the configuration files are referred to here as ‘commands’, Net-SNMP documentation may refer to these entries as ‘directives’.

The **snmpd.conf** configuration file provides the primary configuration information for the Net-SNMP Agent. Other configuration files, snmp.conf and snmptrap.conf are provided with the Net-SNMP agent software, but are specific to operating as an SNMP manager and are beyond the scope of this document.

3.2.3 Net-SNMP Configuration (snmpd.conf)

Overview

This section describes the configuration commands available through the snmpd.conf configuration file.

- Basic Configuration Process:
  - Agent Behavior
  - AgentX
  - Trap Notifications
- Optional Configuration:
  - User Defined Information

Note: A list of recognized commands for this configuration file can be obtained by running the command `snmpd –H`.

Agent Behavior

Listening address

The agent must specify an address to listen on for SNMP requests (GET/SET). This is the interface to which SNMP managers will query the system.

Syntax:

```
agentaddress <transport-specifier>:<transport-address>[:port]
```

Note: The default port of 161 is assumed if the [port] parameter is not passed. In the event that multiple SNMP agents are operating on a single system, these must operate on unique ports.

Example:

```
agentaddress  tcp:localhost:1161
```

SNMP Version and Access Control

DSMI SNMP is capable of SNMPv1, SNMPv2 and SNMPv3 modes of operation.

The version of SNMP operation defines the access control methods available to the user. Note that this version of DSMI SNMP only supports SNMP GET/READ requests; therefore, read-only access would be appropriate.

SNMPv1 and SNMPv2

Specify a read-only community command to enable SNMP GET and GETNEXT requests to the agent. By default, this configuration will apply to all SNMP data from this agent, unless an OID range is specified.

The user can limit this access definition to the DSMI SNMP data set by defining the top level OID of the DSMI SNMP MIB.

Syntax:

```
rocommunity <community_name> [<source> [<oid>] ]
```

Where*:*

`community_name`	User defined index for access group
`source`	System name or address
`oid`	The SNMP Object ID to be restricted / allowed

Example:

```
rocommunity my_community localhost .1.3.6.1.4.1.3028.6.2
```

Note: The OID value provided in the above example is the dlgDSMI root object.

SNMPv3 Overview

Version 3 of SNMP implements enhanced security features and introduces optional authentication, authorization and message encryption for SNMP GET and GETNEXT requests.

SNMPv3 can operate using a User-based Security (USM) user profiles or use native user credentials for authentication through Secure Shell (SSH) or Datagram Transport Layer Security (DTLS).

Here we describe v3 support using the USM-based method. Alternative methods are described in the Net-SNMP documentation.

SNMPv3 Users

Use the 'createUser' command to establish a USM user. Users can be configured to use either MD5 or SHA as authentication encryption methods. DES and AES are available as privacy protocols for data exchange.

Note: SHA, DES and AES algorithms are provided by the OpenSSL prerequisite package.

Syntax:

```
createUser <username> (MD5/SHA) <authpassphrase> [DES/AES]
[privpassphrase]
```

Where:

`username`	User defined username
`authpassphrase`	User defined password (min 8 characters)
`privpassphrase`	Optional, default is to assume the same value as the authpassphrase

Example:

```
createUser myusername MD5 mypassword
```

SNMPv3 Access Control

This version of DSMI SNMP only supports read only functionality.

Define a read only user using the `rouser' command.

Syntax:

```
rouser [-s secmodel] <user> [noauth|auth|priv [oid]]
```

Where:

`secmodel`	Method of security, values: usm (default) tsm (SSH) ksm (Kerberos)
`noauth`	No authentication required
`auth`	Authentication required
`priv`	Enforce use of encryption
`oid`	Restrict policy to OID object

Example:

```
rouser –s usm myusername auth .1.3.6.1.4.1.3028.6.2
```

The above example sets an access control policy for a USM-based user called myusername, stipulates that authentication is required and restricts policy to the DSMI SNMP OID tree.

AgentX Sub-Agent

The DSMI SNMP Sub-Agent operates in conjunction with the Net-SNMP Agent and requires both components to be configured to interface using the AgentX protocol.

The Net-SNMP software is to be configured as a SNMP Master Agent and requires a socket configuration for the DSMI SNMP Sub-Agent to subsequently connect.

The snmpd.conf configuration specifies AgentXSocket at a parameter and defines the socket value. It is recommended that the user configures a default tcp socket of 705. This value must match the socket configuration of the DSMI SNMP Sub-Agent.

Example, local tcp socket support on port 705:

master *agentx*
agentxSocket *tcp:localhost:705*

It is possible for a binary installation of Net-SNMP to be compiled without the support for AgentX tcp/udp communication. In this scenario, on a UNIX-based system a UNIX Domain Socket may be used for communication.

Example, local UNIX Domain Socket:

master *agentx*
agentxSocket */tmp/agentx*

When using a UNIX Domain Socket for process communication, the user must ensure that the UID/GID running the DSA subagent has permission to access the specified UNIX Socket.

TRAP Configuration

To enable trap generation from the DSMI SNMP Sub-Agent the applicable commands are required in snmpd.conf. Traps from DSMI SNMP will be generated in accordance with the configuration of the DSA module (see section 3.3).

If Net-SNMP traps are enabled, the Agent will generate a SNMPv2 MIB::coldStart trap on startup of the Net-SNMP Agent, and a UCS-SNMP-MIB::ucdShutDown when shutting down the Net-SNMP Agent.

A community string is required as a unique reference for the traps to be generated:

```
trapcommunity <string>
```

SNMPv1 Traps:

To enable SNMPv1 traps, use the following command:

```
trapsink <host>[:<port>] [<community>]
```

SNMPv2 Traps:

To enable SNMPv2 traps, use the following command:

```
trap2sink <host>[:<port>] [<community>]
```

Inform notifications:

To enable inform notifications (acknowledged traps), use the following command:

```
informsink <host>[:<port>] [<community>]
```

Additional Traps:

To enable traps on authentication failures, use the following command:

```
authtrapenable <1|2>
```

Where:

1 – enabled
2 – disabled (default)

User Defined Information

This optional configuration section defines SNMP information without the use of a Sub-Agent. This allows the user to manually define system information and to specify other optional monitoring information.

System Information:

The following commands can be added to snmpd.conf to provide string-based system information:

```
sysLocation   <string>
sysContact    <string>
sysName       <string>
sysServices   <string>
sysDescr      <string>
sysObjectID   <string>
```

3.2.4 Example Net-SNMP Configuration (snmpd.conf)

SNMPv2c Example

```
########################################
# snmpd.conf – Example Configuration (SNMPv2c)
########################################

# Access Control – Allow all networks/all OID read access
# A SNMPv1/SNMPv2c read-only access name
# rocommunity <community> [<network> <oid>]
rocommunity public

# Agent Behavior – Run as master agent
# master [yes|no]
master yes

# AgentX Sub-Agent Connection – DSA connectivity: tcp port 705
# agenxSocket <type>:<host>:<port>
agentxSocket tcp:localhost:705

# Traps – Sends v1 traps to two managers
# trapsink <host>:<port> <community>
trapsink 192.168.0.10:162  public
trapsink 192.168.0.11:162  public

########################################
```

3.2.5 Operating With Other Agents

Net-SNMP can be configured to pass selected SNMP GET/SET requests (based on OID range) to an alternative SNMP agent, acting independently of Net-SNMP. The alternative agent can be running on the same system or on an external system.

The Net-SNMP software is able to act as a “proxy” for the alternative agent, and delegate the appropriate GET/SET requests and responses. TRAP notifications are not handled through the proxy mechanism; these are typically pushed directly from each agent towards configured SNMP managers.

External Agents

The Net-SNMP software can forward requests to an alternative system, which will allow alternative SNMP agents to operate on a common port to the Net-SNMP software.

The diagram below shows the high level interaction between systems in this configuration. Here, the Net-SNMP agent and a second agent (here referred to as *alternative* agent) are able to operate on port 161.

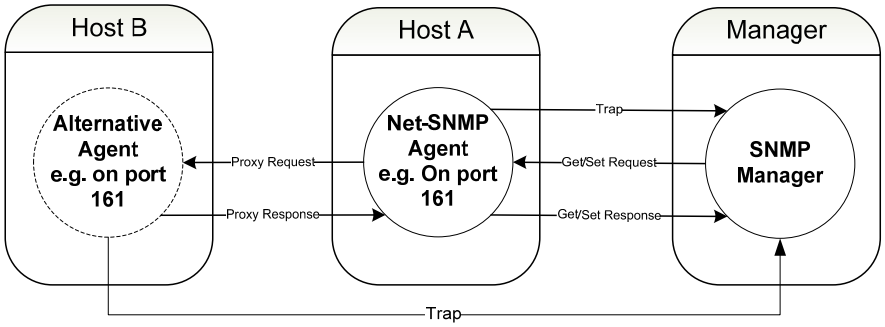


Figure 2 - Multiple SNMP Agents - External Proxy Agent

Local Agents

Enabling two SNMP agents to operation on a single host requires that separate listening ports be allocated to each agent.

For example, the Net-SNMP agent may be configured to listen on the default port of 161.

To enable a second agent (here referred to as the *alternative* agent) to run on the system, the agent may be configured on a different port (e.g., 1161).

In the example presented here, the user may query the alternative agent either directly through port 1161 (as shown in Figure 3), or via the Net-SNMP agent on port 161 (as shown in Figure 4).

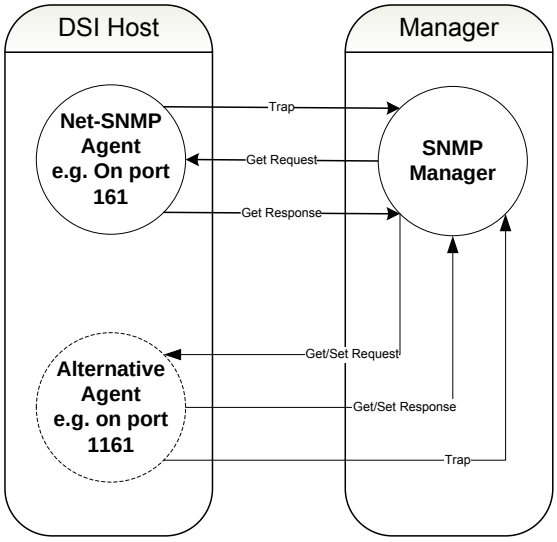


Figure 3 - Multiple SNMP Agents - Direct Requests

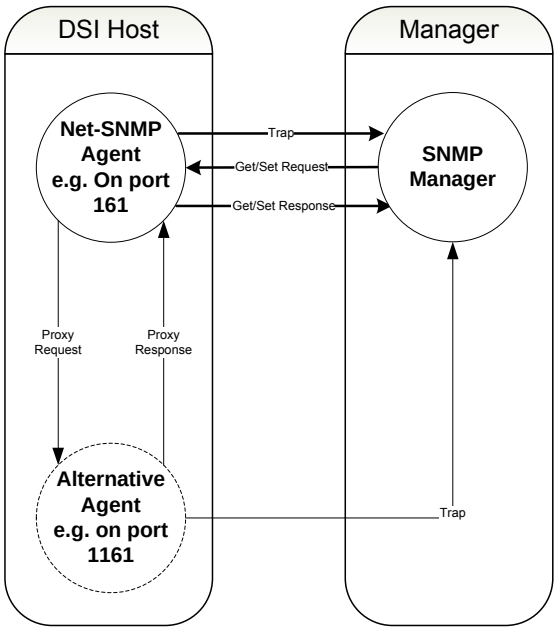


Figure 4 - Multiple SNMP Agents - Proxy Requests

Proxy Configuration

The following commands can be appended to the snmpd.conf configuration file to allow requests to be passed to an alternative agent.

Setting Security Context

The com2sec command provides the ability to specify a community string for a proxy device/system. A unique community string must be used for each device which is handled through the Net-SNMP software.

Syntax:

```
com2sec [NAME] [SOURCE] [COMMUNITY_STRING]
```

The com2sec command requires an arbitrary security NAME value to define the context for which the community string shall map.

The SOURCE value defines the address range for which the request is permitted. This can be global (“default”) or limited to a specific hostname, address, or subnet.

The community string is the value which will be used externally by an SNMP manager, regardless of the alternative agent’s true community string.

Example:

```
com2sec readonly default public
```

The above example uses the “public” community string globally (default) to refer to the “readonly” security context.

Defining Proxy

The proxy command allows requests for the OID value (including subordinates) to be redirected to the HOST specified. Agent query commands can be passed via the SNMPDCMD_ARGS parameter; this is the equivalent of commands passed to an agent through a direct query. For example, the SNMP request version and target community string can be specified.

Syntax:

```
proxy [-Cn CONTEXTNAME] [SNMPCMD_ARGS] HOST OID [REMOTEOID]
```

Example:

```
proxy –v2c -c public 192.168.0.101:161 .1.3.6.1.4.1.20.1.4
```

Note: The OID example given above is an example of a non-Dialogic OID.

The proxy command can be setup to use SNMPv3 authentication if required. Please refer to the Net-SNMP documentation for alternative proxy deployment options - see section 1.3 Related Documentation.

3.2.6 Operating With Other Agents – HMP SNMP

This example illustrates the usage of DSMI SNMP Sub-Agent in conjunction with the Dialogic® Host Media Processing (HMP) SNMP Agent.

It makes use of the Net-SNMP Agent software to “proxy” requests and responses for the HMP SNMP agent.

SNMP GET requests are passed through the Net-SNMP proxy, while TRAPs are generated from the individual agents.

In this example, v2c SNMP packets are used to interface to the HMP SNMP Agent.

Example snmpd.conf entry:

```
com2sec readonly default publicproxy –v2c -c Craftsperson
localhost:1161 .1.3.6.1.4.1.3028.6.3
```

In the above example, all requests for the HMP root OID (.1.3.6.1.4.1.3028.3) will be redirected to the localhost (same host as master agent) on the alternative port 1161. The Net-SNMP Agent also passes the SNMP commands to operate as a version 2c request, using the HMP read only community string “Craftsperson”.

Please refer to the HMP documentation for additional HMP configuration options - see section 1.3 Related Documentation.

3.3 DSMI SNMP Sub-Agent (DSA) Operation

3.3.1 Description

The DSMI SNMP Sub-Agent (DSA) handles all SNMP requests for the DSI-specific OID range and provides information based on registered DSI component modules.

DSMI SNMP utilizes a “push” update model, where the DSA module retains the current information set for all registered DSI components.

DSI components register with the DSA module and manage various SNMP data objects. Upon a state change or event within the DSI module, information is pushed to the DSA module. The information may then be published in the form of TRAP notifications, or queried with SNMP GET requests.

The DSA module can be configured to control TRAP notification generation.

3.3.2 Command Line Options

Example output:

DSA SNMP Subagent

```
-v              :  Display version (without running)
-m<module_id>   :  The id the module will run as. (default 0x6f)
-n<trap level>  :  Notification/Trap Level.
                     1 – Trap on all events, i.e., Trap on creation, deletion
                         and status events.
                     2 – No Trap.
                     3 – Trap on create only.
                     4 – Trap on change of state only (default).
                     5 – Trap on deletion only.
-p<port id>     :  Agent-x port id (default:705).
```

Module Id

The default DSA module id is 0x6f. All DSI component modules are statically configured to report all SNMP updates to the 0x6f module address. Therefore, it is recommended that this parameter remains at the default value.

Trap Level

The DSA module controls the generation of TRAP notifications. Traps are issued for a given SNMP object stored within DSA. There are three reasons for a TRAP notification to be generated:

1. An object has been created (e.g., a PCM has been configured)
2. An object event has occurred (e.g., a PCM alarm has been detected)
3. An object has been deleted (e.g., a PCM has been disabled)

Each object also has an internal state (see section 5.3). It is possible for events to occur without affecting the state of an object. Therefore, the configuration option is provided to only trap on event changing events.

Agent-X Port Id

DSA communicates with the Net-SNMP software through the use of the Agent-X protocol. This parameter specifies the socket to be used for communication. It is the user's responsibility to choose a port which is not assigned and in use by another process. It is recommended for the default tcp port of 705 to be used.

Example Usage, alternative port:

```
$./dsa –n1 –p1161 –m0x6f
```

The DSA Agent also supports UNIX Domain Sockets for AgentX communication. This can be used if the version of Net-SNMP on the target machine does not have udp/tcp support enabled.

Example, UNIX Domain Socket:

```
$./dsa –n2 –p/tmp/agentx
```

3.4 DSI Module Configuration Reference for SNMP

DSI modules with support for SNMP can be configured to enable SNMP through the s7mgt protocol configuration utility, or directly through GCT configuration messages.

This section will describe the command lines required when using the s7_mgt configuration utility, and provides summarized details of the message parameters required to enable SNMP.

The configuration descriptions are grouped in the following categories:

- Physical Interface SNMP Configuration Commands
- MTP SNMP Configuration Commands
- SIGTRAN SNMP Configuration Commands

3.4.1 Physical Interface SNMP Command Configuration

Board Object

SNMP can be enabled for the physical interfaces on a per-board basis. Information provided through this configuration includes board specific data, and all Line Interface Units subsequently configured.

SNMP for the board and all PCMs can be enabled by setting bit 16 of the flags field to 1.

Example, SNMP enabled:

Syntax:

```
SS7_BOARD <board_id> <board_type> <flags> <code_file> <run_mode>
```

Example:

```
SS7_BOARD 0 SPCI4 0x00010000 ss7.dc3 MTP2
```

3.4.2 MTP SNMP Command Configuration

Enable MTP Link Object

SNMP can be enabled for individual MTP links through the MTP_LINK command line in a config.txt file. SNMP is enabled by setting bit 30 of the flags field to 1.

Enable MTP Linkset Object

SNMP can be enabled for individual MTP Linksets through the MTP_LINKSET command in a config.txt file. SNMP is enabled by setting bit 4 of the flags field to 1.

Enable MTP Route Object

SNMP can be enabled for individual MTP Routes through the MTP_ROUTE command in a config.txt file. SNMP is enabled by setting bit 6 of the flags field to 1.

3.4.3 SIGTRAN SNMP Configuration Commands

Enable SIGTRAN Signaling Links

SNMP can be enabled for individual SIGTRAN Links through the SNSLI:SNLINK command in a config.txt file. SNMP is enabled by passing the 'SNMP=Y' parameter.

SIGTRAN Remote Application Servers

SNMP can be enabled for individual SIGTRAN Remote Application Servers through the SNRAI:RAS command in a config.txt file. SNMP is enabled by passing the 'SNMP=Y' parameter.

SIGTRAN Routes

SNMP can be enabled for individual SIGTRAN Routes through the SNRTI:SNRT command in a config.txt file. SNMP is enabled by passing the 'SNMP=Y' parameter.

3.4.4 SNMP Command Configuration Example

The example config.txt below shows SNMP enabled with a single Dialogic® DSI SPCI Network Interface Board, MTP Link, MTP Linkset and MTP Route.

```
**************************
* config.txt – SNMP Example
**************************

*SPCI Board Configuration – SNMP enabled (includes LIUs)
*Syntax: SS7_BOARD
* <board_id> <board_type> <flags> <code_file> <run_mode>
SS7_BOARD 0 SPCI4 0x00010042 ss7.dc3 MTP *SNMP enabled

*Line Interface configuration
*Syntax: LIU_CONFIG
* <board_id> <liu_id> <liu_type> <line_code> <frame_format>
* <crc_mode>[<build_out>]
LIU_CONFIG  0  0  5  1  1  1
LIU_CONFIG  0  1  5  1  1  1

*MTP Configuration
*Syntax: MTP_CONFIG <reserved> <reserved> <options>
MTP_CONFIG  0  0  0x00040000

*MTP Linksets – SNMP enabled for Linkset 0
*Syntax: MTP_LINKSET <linkset_id> <adjacent_spc> <num_links>
* <flags> <local_spc> <ssf>
MTP_LINKSET  0  1  1  0x0010  2  0x0008  *SNMP enabled
MTP_LINKSET  1  2  1  0x0000  1  0x0008  *SNMP disabled

*MTP Links – SNMP enabled for Link 0
*Syntax: MTP_LINK <link_id> <linkset_id> <link_ref> <slc>
* <board_id> <blink> <stream> <timeslot> <flags>
MTP_LINK  0  0  0  0  0  0  0  16  0x40000006  *SNMP enabled
MTP_LINK  1  1  0  0  0  1  1  16  0x00000006  *SNMP disabled
```

```
*MTP Routes – SNMP enabled for Route  DPC 1
*Syntax: MTP_ROUTE <dpc> <linkset_id> <user_part_mask>
MTP_ROUTE  1  0  0x0020  0x0040  0  *SNMP enabled
MTP_ROUTE  2  1  0x0020  0x0000  0  *SNMP disabled
**************************
```

3.5 Running DSMI SNMP

DSMI SNMP requires that software binaries be initiated in the following order:

1. Start Net-SNMP Agent (snmpd) with administrative privileges.
2. Start GCT environment (gctload).
3. Start DSMI Sub-Agent Module (dsa) and DSI component modules (e.g., mtp3).

3.5.1 Example – Running on Linux

Using the configuration examples in this document, the next section provides an example of launching the complete environment to support SNMP for a Linux system. Please refer to the relevant sections of this document for details of operating system differences.

1. Starting the SNMP Agent (with administrative privileges):

   ```
   $/usr/sbin/snmpd -C –c /etc/snmp/snmpd.conf –L
   ```

   Switch options in example:
   -C Discard default configuration.
   -c User defined configuration file.
   -L Debug output to terminal.

Note: The user will require sufficient privileges to listen on the tcp/udp/unix socket defined in snmpd.conf.

Expected Output:

```
Turning on AgentX master support.

NET-SNMP version 5.x.x
```

2. Starting the GCT environment

   The DSA Sub-Agent and DSI component modules operate within the DSI GCT messaging environment.

   Call gctload binary with appropriate system.txt:

   ```
   $./gctload –csystem.txt
   ```

3. Starting the DSA module

   The DSA module can be launched from gctload in a similar fashion to other DSI modules, or manually on the command line:

   ```
   $./dsa –n1
   ```

4. Configuring DSI component modules

   Start the s7_mgt protocol configuration utility with the appropriate config.txt (see section 3.4.4).

   ```
   $./s7_mgt –kconfig.txt
   ```

   Upon completion of configuration, the system is ready for external query from an SNMP manager.

3.5.2 Example - System Environment

The example system.txt below provides the module address defines and binary launch commands for SNMP support with the SPCI board.

```
**************************
* system.txt SNMP example
**************************
*Essential Host Modules

LOCAL             0x20          * Board interface task – ssds
LOCAL             0x00          * Timer task – tim_lnx

*Optional Host Modules

LOCAL             0xcf          * Management Task – s7_mgt
LOCAL             0xef          * Logging – s7_log
LOCAL             0x6f          * DSMI SNMP Sub-Agent – dsa

*Optional Modules running on board

REDIRECT   0x71   0x20   *MTP2 module
REDIRECT   0x10   0x20   *Ct bus/Clocking control
REDIRECT   0x8e   0x20   *On-board management
REDIRECT   0x22   0x20   *On-board MTP3

*Redirect status indications

REDIRECT   0xdf   0xef   * LIU/MTP2 status message -> s7_log

*Dimensioning the Message Passing Environment

NUM_MSGS   10000        * Number of standard messages.

*Tasks to start – OS Specific Parts (uncomment)

*Linux binaries

*FORK_PROCESS ./ssds
*FORK_PROCESS ./tim_lnx
*FORK_PROCESS ./tick_lnx
*FORK_PROCESS ./s7_mgt
*FORK_PROCESS ./s7_log
*FORK_PROCESS ./dsa –n1 –p705 –m0x6f

*Solaris binaries

*FORK_PROCESS ./ssds
*FORK_PROCESS ./tim_sol
*FORK_PROCESS ./tick_sol
*FORK_PROCESS ./s7_mgt
*FORK_PROCESS ./s7_log
*FORK_PROCESS ./dsa –n1 –p705 –m0x6f

*Windows binaries

*FORK_PROCESS ssds.exe
*FORK_PROCESS tim_nt.exe
*FORK_PROCESS tick_nt.exe
*FORK_PROCESS s7_mgt.exe
*FORK_PROCESS s7_log.exe
*FORK_PROCESS dsa.exe –n1 –p705 –m0x6f

******************
```

4 DSMI MIB Structure

4.1 Product Family MIB Structure

A MIB is a specification containing definitions of management information so that networked systems can be remotely monitored, configured, and controlled.

The information objects defined across MIB's are organized hierarchically. An OID value can be used to reference a particular object within the hierarchical data structure.

Dialogic® SNMP MIBS are organized within the standard hierarchical SNMP structure under the private *enterprises(1)* branch:

.1(iso)
.1.3(org)
.1.3.6(dod)
.1.3.6.1(internet)
.1.3.6.1.4(private)
.1.3.6.1.4.1(enterprises)

The Dialogic-specific information is held within the *3028(dialogic)* branch.

.1.3.6.1.4.1.3028(dialogic)

MIBS specific to Dialogic® products are located within the *dlgProducts(6)* branch:

.1.3.6.1.4.1.3028.6(dlgProducts)

MIBS described within this document and specific to the Dialogic® DSI component product range are located within the Dialogic *dlgDSMI(2)* branch:

.1.3.6.1.4.1.3028.6.2(dlgDSMI)

Within the DSI specific dlgDSMI object, there are two branches: One branch defines DSI product information, *dsmiObjects(1);* the second branch defines Event Notifications (TRAPS) and Textual conventions, *dsmiModules(2)*:

.1.3.6.1.4.1.3028.6.2.1(dsmiObjects)

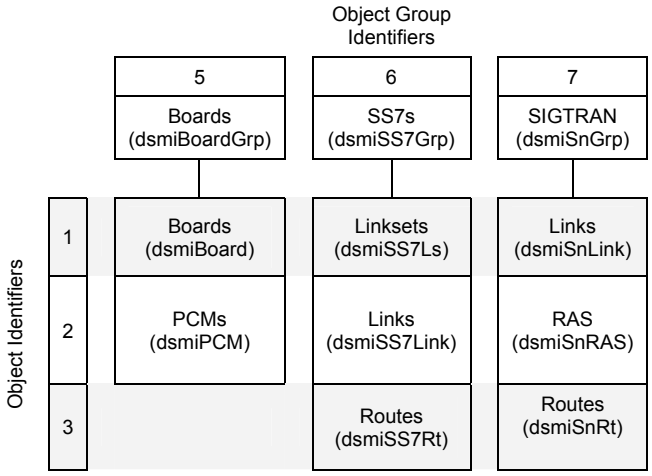
.1.3.6.1.4.1.3028.6.2.2(dsmiModules)

The following sections describe these branches and associated data.

4.2 The DSMI Object Groups and Objects

The following diagram represents the structure and location of the *dsmiObjects,* which represent DSI components:

Figure 5 - Structure and Location of the Dialogic® DSI SNMP MIB Groups and their Component Objects

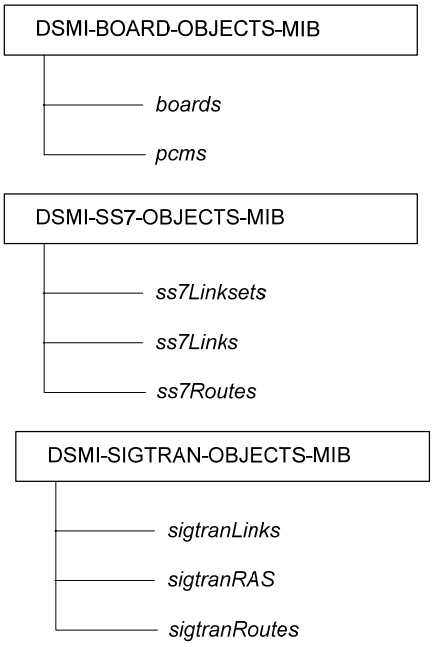


An object is referenced by specifying its object group identifier followed by its own identifier. For example, the PCMs object (2.in the Boards group) is referenced as 5.2. Its fully qualified OID, therefore, is .1.3.6.1.4.1.3028.6.2.1.**5.2**.

4.3 The MIB Files

There is one MIB definition file per DSMI object group as well as additional MIB definition files, which define the location of the DSMI objects within the SNMP object hierarchy (DSMI-SMI), the textual conventions used in defining the objects (DSMI-TC) and the notifications generated by the DSMI SNMP agent (DSMI-EVENTS).

The MIB definition files give the object groups and objects more user-friendly names. The following diagrams depict the user-friendly names of the object groups and their respective objects.



It is required that the user load the DSMI-SMI, DSMI-TC and DSMI-DSI-EVENTS and DLGC-GLOBAL-REG MIBs, as well as all relevant object group MIBs, into their SNMP manager to correctly interpret information.

4.4 Components of an Object

4.4.1 Introduction

Each object comprises two tables. The first table, or the **object table**, holds status data about the object, whereas the second table, or the **administration table**, details the number of rows in the table as well as the current TRAP configuration settings for the object. The object table resides at OID x.y.1 (where x is the object group identifier and y is the object identifier). The administration table resides at OID x.y.2. For example, the sigtranLinks object table is found at .1.3.6.1.4.1.3028.6.2.1.7.1.**1**, and the administration table is located at .1.3.6.1.4.1.3028.6.2.1.7.1.**2**. The administration and objects tables are now described. Whereas the administration table serves a common object-independent function across all objects, the object table has contains attributes which may have object-dependent meaning.

4.4.2 The Object Table

The object table consists of a common set of columns that are to be found in every object. There may be one or more rows in the object table. This will depend on the functionality being represented by the object. In addition to these columns, some objects have object-specific columns that provide extra information pertaining to the object in question. The common columns are as follows:

Column Name	Data Type	Description
dsmiHeadIndex	Unsigned32	The internal row index
dsmiHeadRowStatus	RowStatus	Used for row maintenance
dsmiHeadTimeInState	TimeTicks	The duration of time that the object has been in the current state
dsmiHeadIdVal	Unsigned32	A unique identifier for the row
dsmiHeadIdDescription	DisplayString	A string that holds object-specific information
dsmiHeadState	DSMI-OBJSTATE	The current state of the object
dsmiHeadOwnerId	OCTET STRING	The internal owner of the object.

The following section describes each object in greater detail with reference to the common header columns and, if relevant, additional columns that are associated with the object. If the common header fields in an object have object-specific behavior, the details are given. If no details are given for one of the common header fields, it can be assumed that the general purpose description given above applies.

4.4.3 The Administration Table

The administration table provides information in relation to the tabular data object sibling within the OID hierarchy. This table object holds information including counters and TRAP administration information.

Column Name	Data Type	Description
dsmiAdminIndex	Unsigned32	The admin table index
dsmiUpCount	Unsigned32	The number of rows in the “Up” state
dsmiDownCount	Unsigned32	The number of rows in the “Down” state
dsmiInactiveCount	Unsigned32	The number of rows in the “Inactive” state
dsmiImpairedCount	Unsigned32	The number of rows in the “Impaired” state
dsmiRestartCount	Unsigned32	The number of rows in the “Restart” state
dsmiQuiescingCount	Unsigned32	The number of rows in the “Quiescing” state
dsmiWarningCount	Unsigned32	The number of rows in the “Warning” state
dsmiTotalRowCount	Unsigned32	The total number of rows in the Object Table
dsmiUpTrapCfg	DSMI-TRAPCONFIG	TRAP/INFORM generation on transition to “Up” state
DsmiDownTrapCfg	DSMI-TRAPCONFIG	TRAP/INFORM generation on transition to “Down” state
DsmiInactiveTrapCfg	DSMI-TRAPCONFIG	TRAP/INFORM generation on transition to “Inactive” state
DsmiImpairedTrapCfg	DSMI-TRAPCONFIG	TRAP/INFORM generation on transition to “Impaired” state
DsmiRestartTrapCfg	DSMI-TRAPCONFIG	TRAP/INFORM generation on transition to “Restart” state
DsmiQuiescingTrapCfg	DSMI-TRAPCONFIG	TRAP/INFORM generation on transition to “Quiescing” state
DsmiWarningTrapCfg	DSMI-TRAPCONFIG	TRAP/INFORM generation on transition to “Warning” state

5 The DSMI Object Groups and Objects

5.1 DSMI-BOARD-OBJECTS-MIB (The Board Object Group)

This object group represents the signaling hardware interfaces within the Signaling Server.

5.1.1 dsmiBoardObjectTable

This object holds data relating to signaling boards installed in the system.

Column Name	Column Description
dsmiBoardHeadIndex	-
dsmiBoardHeadRowStatus	-
dsmiBoardHeadTimeInState	The period of time since the last state change for the board.
dsmiBoardHeadIdVal	DSMI object identifier for a board
dsmiBoardHeadIdDescription	Additional information used to identify the board
dsmiBoardHeadState	The current state of the board. Possible states a board may be in are: ● up ● Down ● inactive
dsmiBoardHeadOwnerId	-

5.1.2 dsmiPCMObjectTable

This object holds data relating to the PCMs installed in a system.

Column Name	Column Description
dsmiPCMHeadIndex	-
dsmiPCMHeadRowStatus	-
dsmiPCMHeadTimeInState	The period of time since the last state change for the PCM
dsmiPCMHeadIdVal	DSMI object identifier for a PCM
dsmiPCMHeadIdDescription	Additional information used to identify the PCM
dsmiPCMHeadState	The current state of the PCM. Possible states a PCM may be in are: ● up ● down ● inactive ● impaired
dsmiPCMOwnerId	-
dsmiPCMBoard	The PCM board's identifier

Column Name	Column Description
dsmiPCMLiu	The PCM's LIU identifier
dsmiPCMLineStatus	This field is an octet containing the alarm status for the PCM. bit 0 unused bit 1 ber10minus5 - The PCM is encountering a Bit Error Rate (BER) exceeding 10^3 bit 2 ber10minus3 - The PCM is encountering a Bit Error Rate (BER) exceeding 10^3 bit 3 remotealarm - The remote end indicates that is it is OK, but also indicates that it is detecting an error condition bit 4 syncloss - Loss of frame alignment since no frame synchronization has been received bit 5 ais - Alarm indication signal. The remote side sends all ones indicating that there is an error condition, or it is not initialized bit 6 Pcmloss - No signal sensed on the PCM input bit 7 Mismatch - The PCMTYPE setting is inconsistent with the hardware settings on the board If no errors are present on the line, this value will read as zero. Note: SNMP "BITS" fields are defined in reverse order. Bit 0 resides at the MSB and bit 7 at the LSB.

5.2 DSMI-SS7-OBJECTS-MIB (The SS7 Object Group)

This group represents the SS7 Links, Linksets and Routes that have been configured.

5.2.1 dsmiSS7LsObjectTable

This object holds data regarding the SS7 linksets configured in a system.

Column Name	Column Description
dsmiSS7LsHeadIndex	-
dsmiSS7LsHeadRowStatus	-
dsmiSS7LsHeadTimeInState	The period of time since the last state change for the SS7 linkset
dsmiSS7LsHeadIdVal	DSMI object identifier for a SS7 linkset
dsmiSS7LsHeadIdDescription	Additional information used to identify the SS7 linkset
dsmiSS7LsHeadState	The current state of the SS7 linkset. Possible states a SS7 linkset may be in are: ● up ● down ● inactive
dsmiSS7LsOwnerId	-

5.2.2 dsmiSS7LinkObjectTable

This object holds data regarding the SS7 links configured in a system.

Column Name	Column Description
dsmiSS7LinkHeadIndex	-
dsmiSS7LinkHeadRowStatus	-
dsmiSS7LinkHeadTimeInState	The period of time since the last state change for the SS7 link
dsmiSS7LinkHeadIdVal	DSMI object identifier for a SS7 link
dsmiSS7LinkHeadIdDescription	Additional information used to identify the SS7 link
dsmiSS7LinkHeadState	The current state of the SS7 link. Possible states a SS7 link may be in are: ● up ● down ● inactive
dsmiSS7LinkHeadOwnerId	-

5.2.3 dsmiSS7RtObjectTable

This object holds data regarding the SS7 routes configured in a system.

Column Name	Column Description
dsmiSS7RtHeadIndex	-
dsmiSS7RtHeadRowStatus	-
dsmiSS7RtHeadTimeInState	The period of time since the last state change for the SS7 Route
dsmiSS7RtHeadIdVal	DSMI object identifier for a SS7 route.
dsmiSS7RtHeadIdDescription	Additional information used to identify the SS7 route.
dsmiSS7RtHeadState	The current state of the SS7 route. Possible states a SS7 route may be in are: • up • down
dsmiSS7RtHeadOwnerId	-

5.3 DSMI-SIGTRAN-OBJECTS-MIB (The SIGTRAN Object Group)

This group represents the SIGTRAN Links, Remote Application Servers and Routes that have been configured.

5.3.1 dsmiSnLinkObjectTable

This object holds data regarding the SIGTRAN links configured in a system.

Column Name	Column Description
dsmiSnLinkHeadIndex	-
dsmiSnLinkHeadRowStatus	-
dsmiSnLinkHeadTimeInState	The period of time since the last state change for the SIGTRAN link
dsmiSnLinkHeadIdVal	DSMI object identifier for a SIGTRAN links
dsmiSnLinkHeadIdDescription	Additional information used to identify the SIGNTRAN link
dsmiSnLinkHeadState	The current state of the SIGTRAN link. Possible states a SIGTRAN link may be in are: ● up ● down ● inactive
dsmiSnLinkHeadOwnerId	-

5.3.2 dsmiSnRASObjectTable

This object holds data regarding the SIGTRAN Remote Application Servers (RASs) configured in a system.

Column Name	Column Description
dsmiSnRASHeadIndex	-
dsmiSnRASHeadRowStatus	-
dsmiSnRASHeadTimeInState	The period of time since the last state change for a SIGTRAN RAS
dsmiSnRASHeadIdVal	DSMI object identifier for a SIGTRAN RAS
dsmiSnRASHeadIdDescription	Additional information used to identify the SIGTRAN RAS
dsmiSnRASHeadState	The current state of the SIGTRAN RAS. Possible states a SIGTRAN RAS may be in are: ● up ● down ● inactive
dsmiSnRASHeadOwnerId	-

5.3.3 dsmiSnRtObjectTable

This object holds data regarding the SIGTRAN routes configured in a system.

Column Name	Column Description
dsmiSnRtHeadIndex	-
dsmiSnRtHeadRowStatus	-
dsmiSnRtHeadTimeInState	The period of time since the last state change for the SIGTRAN link
dsmiSnRtHeadIdVal	DSMI object identifier for a SIGTRAN Route
dsmiSnRtHeadIdDescription	Additional information used to identify the SIGTRAN Route
dsmiSnRtHeadState	The current state of the SIGTRAN route. Possible states a SIGTRAN route may be in are: ● up ● down ● inactive
dsmiSnRtHeadOwnerId	-

5.4 DSI Event TRAP Notification Fields

TRAP's provide a mechanism of reporting events for each registered object from a DSI component. For each SNMP enabled DSI Component, events are reported to the DSMI SNMP Sub-Agent. The agent is then able to generate TRAPs to subscribed SNMP managers according to the user's configuration (as described in section 3.2.

The DSMI SNMP system uses a single TRAP definition for passing event notifications. The TRAP format is described in the DSMI-DSI-EVENT.mib file.

TRAP notification's contains the following fields:

Field	Description
dsmiDsiEventOid	The Object Identifier (OID) of the object against which the TRAP was generated. This can be used to subsequently query the object in question.
dsmiDsiEventOidIndex	The internal row index within the object for which the TRAP was generated.
dsmiDsiEventOidIdVal	The identifier for the object against which the TRAP was generated.
dsmiDsiEventOidIdDescription	A description of the object against which the TRAP was generated.
dsmiDsiEventEvId	The identifier for a particular event that occurred. (see below)
dsmiDsiEventSeverity	The perceived severity of the TRAP. (see below)
dsmiDsiEventString	A description of the event and the associated object for which the TRAP was generated.
dsmiDsiEventState	The current state of the associated object for which the TRAP has been generated. (refer to per object states)

The EventId indicates the type of event that has occoured, the field can carry one of the following values:

Event	ID	Description
boardresetreq	1	A request has been received to reset a Signaling Board.
boardresetcomplete	2	A Signaling Board has successfully been reset.
boardfailure	3	A fault has been detected with a Signaling Board and it has stopped operating. Note: Signaling Board licensing issues are reported as board faults and the boards will become unresponsive until reset.
pcmlosonset	4	Loss of signal at PCM input port.
pcmloscleared	5	A signal is detected at PCM input port.
pcmsynclossonset	6	Loss of frame alignment since no frame synchronization has been received
pcmsynclosscleared	7	Restoration of frame alignment on PCM port.
pcmaisonset	8	PCM input port contains the Alarm Indication Signal (all ones on all timeslots).

Event	ID	Description
pcmaiscleared	9	PCM input port no longer contains the Alarm Indication Signal (all ones on all timeslots).
pcmraionset	10	PCM port is receiving a Remote Alarm Indication.
pcmraicleared	11	PCM port is no longer receiving a Remote Alarm Indication.
pcmber3onset	12	The input PCM signal contains a Bit Error Rate (BER) in excess of 1 in 1000 as measured on the frame alignment pattern.
pcmber3cleared	13	The input PCM signal no longer contains a Bit Error Rate (BER) in excess of 1 in 1000 as measured on the frame alignment pattern.
pcmber5onset	14	The input PCM signal contains a Bit Error Rate (BER) in excess of 1 in 100,000 as measured on the frame alignment pattern.
pcmber5cleared	15	The input PCM signal no longer contains a Bit Error Rate (BER) in excess of 1 in 100,000 as measured on the frame alignment pattern.
mtplinksetactivation	16	A request has been received to activate an MTP Link set.
mtplinksetdeactivation	17	A request has been received to deactivate an MTP Link set.
mtplinksetfailure	18	All signaling links in an SS7 signaling link set have failed.
mtplinksetrecovery	19	1 or more signaling links in an SS7 signaling link set are available.
mtpdestinationinaccessible	20	MTP route destination is inaccessible.
mtpdestinationaccessible	21	MTP route destination is accessible
mtplinkactivation	22	Request has been received to activate an MTP Link
mtplinkdeactivation	23	Request has been received to deactivate an MTP Link
mtplinkunavailable	24	An SS7 signaling link has failed.
mtplinkavailable	25	An SS7 signaling link has become available.
m3uaserverunavailable	26	An M3UA server has become unavailable
m3uaserveravailable	27	An M3UA server has become available
m3uadestinationinaccessable	28	An M3UA destination has become inaccessible
m3uadestinationaccessable	29	An M3UA destination has become accessible
m3ualinkactivation	30	An M3UA link has been activated
m3ualinkdeactivation	31	An M3UA link has been deactivated
m3ualinkunavailable	32	An M3UA link has become unavailable
m3ualinkavailable	33	An M3UA link has become available
configurationinitiate	200	An SNMP object has been created upon initial configuration.

Event	ID	Description
configurationchange	201	An SNMP object has been recreated upon a subsequent configuration.
configurationend	202	An SNMP object has been deleted upon configuration end.

TRAP notifications contain a value to indicate the perceived severity of an event. The value follows ITU severity convention and is set to one of the following values:

Severity	ID	Description
sevCleared	1	An alarm condition has cleared.
sevIndeterminate	2	Notification of a non erroneous event (e.g., a configuration change).
sevCritical	3	A service-affecting event has occurred and immediate corrective action is required.
sevMajor	4	A service-affecting event has occurred and urgent corrective action is required.
sevMinor	5	A non–service-affecting event has occurred and corrective action is required to prevent the condition from becoming more serious.
sevWarning	6	A potential or impending service-affecting event has been detected but no significant effects have been felt as yet. Action should be taken to further diagnose the problem to prevent the condition from becoming more serious.

6 License

6.1 Introduction

The DSMI SNMP Sub-Agent software described in this document incorporates libraries from the Net-SNMP application suite in accordance with the Net-SNMP license. This Net-SNMP license is reproduced in full in the text below as it appeared at 25-May-10, which was the date of initial publication of this document. A copy of the license, as of that date, can be found via: http://www.net-snmp.org

6.2 License

---- Net-SNMP License -----

Various copyrights apply to this package, listed in 6 separate parts below. Please make sure that you read all the parts. Up until 2001, the project was based at UC Davis, and the first part covers all code written during this time. From 2001 onwards, the project has been based at SourceForge, and Networks Associates Technology, Inc hold the copyright on behalf of the wider Net-SNMP community, covering all derivative work done since then. An additional copyright section has been added as Part 3 below also under a BSD license for the work contributed by Cambridge Broadband Ltd. to the project since 2001. An additional copyright section has been added as Part 4 below also under a BSD license for the work contributed by Sun Microsystems, Inc. to the project since 2003.

Code has been contributed to this project by many people over the years it has been in development, and a full list of contributors can be found in the README file under the THANKS section.

---- Part 1: CMU/UCD copyright notice: (BSD like) -----
Copyright 1989, 1991, 1992 by Carnegie Mellon University

Derivative Work - 1996, 1998-2000
Copyright 1996, 1998-2000 The Regents of the University of California
All Rights Reserved

Permission to use, copy, modify and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appears in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of CMU and The Regents of the University of California not be used in advertising or publicity pertaining to distribution of the software without specific written permission.

CMU AND THE REGENTS OF THE UNIVERSITY OF CALIFORNIA DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL CMU OR THE REGENTS OF THE UNIVERSITY OF CALIFORNIA BE LIABLE FOR ANY SPECIAL, INDIRECT OR ONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM THE LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

---- PART 2: NETWORKS ASSOCIATES TECHNOLOGY, INC COPYRIGHT NOTICE (BSD) -----

COPYRIGHT (C) 2001-2003, NETWORKS ASSOCIATES TECHNOLOGY, INC ALL RIGHTS RESERVED.

REDISTRIBUTION AND USE IN SOURCE AND BINARY FORMS, WITH OR WITHOUT MODIFICATION, ARE PERMITTED PROVIDED THAT THE FOLLOWING CONDITIONS ARE MET:
* REDISTRIBUTIONS OF SOURCE CODE MUST RETAIN THE ABOVE COPYRIGHT NOTICE, THIS LIST OF CONDITIONS AND THE FOLLOWING DISCLAIMER.
* REDISTRIBUTIONS IN BINARY FORM MUST REPRODUCE THE ABOVE COPYRIGHT NOTICE, THIS LIST OF CONDITIONS AND THE FOLLOWING DISCLAIMER IN THE DOCUMENTATION AND/OR OTHER MATERIALS PROVIDED WITH THE DISTRIBUTION.
* NEITHER THE NAME OF THE NETWORKS ASSOCIATES TECHNOLOGY, INC NOR THE NAMES OF ITS CONTRIBUTORS MAY BE USED TO ENDORSE OR PROMOTE PRODUCTS DERIVED FROM THIS SOFTWARE WITHOUT SPECIFIC PRIOR WRITTEN PERMISSION.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

---- PART 3: CAMBRIDGE BROADBAND LTD. COPYRIGHT NOTICE (BSD) -----

PORTIONS OF THIS CODE ARE COPYRIGHT (C) 2001-2003, CAMBRIDGE BROADBAND LTD. ALL RIGHTS RESERVED.

REDISTRIBUTION AND USE IN SOURCE AND BINARY FORMS, WITH OR WITHOUT MODIFICATION, ARE PERMITTED PROVIDED THAT THE FOLLOWING CONDITIONS ARE MET:
* REDISTRIBUTIONS OF SOURCE CODE MUST RETAIN THE ABOVE COPYRIGHT NOTICE, THIS LIST OF CONDITIONS AND THE FOLLOWING DISCLAIMER.
* REDISTRIBUTIONS IN BINARY FORM MUST REPRODUCE THE ABOVE COPYRIGHT NOTICE, THIS LIST OF CONDITIONS AND THE FOLLOWING DISCLAIMER IN THE DOCUMENTATION AND/OR OTHER MATERIALS PROVIDED WITH THE DISTRIBUTION.
* THE NAME OF CAMBRIDGE BROADBAND LTD. MAY NOT BE USED TO ENDORSE OR PROMOTE PRODUCTS DERIVED FROM THIS SOFTWARE WITHOUT SPECIFIC PRIOR WRITTEN PERMISSION.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDER ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

---- PART 4: SUN MICROSYSTEMS, INC. COPYRIGHT NOTICE (BSD) -----

COPYRIGHT © 2003 SUN MICROSYSTEMS, INC., 4150 NETWORK CIRCLE, SANTA CLARA, CALIFORNIA 95054, U.S.A. ALL RIGHTS RESERVED.

USE IS SUBJECT TO LICENSE TERMS BELOW.
THIS DISTRIBUTION MAY INCLUDE MATERIALS DEVELOPED BY THIRD PARTIES.
SUN, SUN MICROSYSTEMS, THE SUN LOGO AND SOLARIS ARE TRADEMARKS OR REGISTERED TRADEMARKS OF SUN MICROSYSTEMS, INC. IN THE U.S. AND OTHER COUNTRIES.
REDISTRIBUTION AND USE IN SOURCE AND BINARY FORMS, WITH OR WITHOUT MODIFICATION, ARE PERMITTED PROVIDED THAT THE FOLLOWING CONDITIONS ARE MET:
* REDISTRIBUTIONS OF SOURCE CODE MUST RETAIN THE ABOVE COPYRIGHT NOTICE, THIS LIST OF CONDITIONS AND THE FOLLOWING DISCLAIMER.
* REDISTRIBUTIONS IN BINARY FORM MUST REPRODUCE THE ABOVE COPYRIGHT NOTICE, THIS LIST OF CONDITIONS AND THE FOLLOWING DISCLAIMER IN THE DOCUMENTATION AND/OR OTHER MATERIALS PROVIDED WITH THE DISTRIBUTION.
* NEITHER THE NAME OF THE SUN MICROSYSTEMS, INC. NOR THE NAMES OF ITS CONTRIBUTORS MAY BE USED TO ENDORSE OR PROMOTE PRODUCTS DERIVED FROM THIS SOFTWARE WITHOUT SPECIFIC PRIOR WRITTEN PERMISSION.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

---- PART 5: SPARTA, INC COPYRIGHT NOTICE (BSD) -----

COPYRIGHT (C) 2003-2005, SPARTA, INC ALL RIGHTS RESERVED.

REDISTRIBUTION AND USE IN SOURCE AND BINARY FORMS, WITH OR WITHOUT MODIFICATION, ARE PERMITTED PROVIDED THAT THE FOLLOWING CONDITIONS ARE MET: