Dialogic

Dialogic® Converged Services Platform - SwitchKit® Development Environment

Simple Network Management Protocol User's Guide

February 2008

www.dialogic.com

Copyright and Legal Disclaimer

All contents of this document are subject to change without notice and do not represent a commitment on the part of Dialogic Corporation or its subsidiaries. Reasonable effort is made to ensure the accuracy of the information contained in the document. However, due to ongoing product improvements and revisions, Dialogic Corporation and its subsidiaries do not warrant the accuracy of this information and cannot accept responsibility for errors 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 EXPLICITLY SET FORTH BELOW OR 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.

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 Corporation or its subsidiaries may infringe one or more patents or other intellectual property rights owned by third parties. Dialogic Corporation or its subsidiaries do 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 Corporation or its subsidiaries. More detailed information about such intellectual property is available from Dialogic Corporation's legal department at 9800 Cavendish Blvd., 5th Floor, Montreal, Quebec, Canada H4M 2V9. The software referred to in this document is provided under a Software License Agreement. Refer to the Software License Agreement for complete details governing the use of the software.

Dialogic Corporation 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.

Dialogic, Dialogic Pro, Brooktrout, Cantata, SnowShore, Eicon, Eicon Networks, Eiconcard, Diva, SIPcontrol, Diva ISDN, TruFax, Realblocs, Realcomm 100, NetAccess, Instant ISDN, TRXStream, Exnet, Exnet Connect, EXS, ExchangePlus VSE, Switchkit, N20, Powering The Service-Ready

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.

Dialogic Product Line Warranty

Unless otherwise stated in an applicable product purchase agreement between the Customer and Dialogic, Dialogic warrants that during the Warranty Period, products will operate in substantial conformance with Dialogic's standard published documentation accompanying the product. If a product does not operate in accordance therewith during the Warranty Period, the Customer must promptly notify Dialogic. Dialogic, at its option, will either repair or replace the product without charge. The Customer has the right, as their exclusive remedy, to return the product for a refund of purchase price or license fee if Dialogic is unable to repair or replace it.

Warranty Period

In the event that you have no signed agreement setting out a warranty period, the Warranty Period shall be the standard warranty period set out on www.dialogic.com on the date of your purchase of the product.

The Warranty Period begins on the date of shipment of any products or software by Dialogic.

The Warranty Period for repaired, replaced or corrected products and software shall be coterminous to the Warranty Provided for the original products or software purchased.

To report warranty claims, Customer may contact Dialogic via email at techsupport@cantata.com or call (781) 433-9600.

Warranty Provisions

A. During the Warranty Period, Dialogic warrants to Customer only that:

(i) Products manufactured by Dialogic (including those manufactured for Dialogic by an original equipment manufacturer) will be free from defects in material and workmanship and will substantially conform to specifications for such products;

(ii) software developed by Dialogic will be free from defects which materially affect performance in accordance with the specifications for such software. With respect to products or software or partial assembly of products furnished by Dialogic but not manufactured by Dialogic, Dialogic hereby assigns to Customer, to the extent permitted, the warranties given to Dialogic by its vendors of such items.

B. If, under normal and proper use, a defect or non conformity appears in warranted products or software during the applicable Warranty Period and Customer promptly notifies Dialogic in writing during the applicable warranty period of such defect or non conformance, and follows Dialogic's instructions regarding return of such defective or non conforming Product or Software, then Dialogic will, at no charge to Customer, either:

(i) repair, replace or correct the same at its manufacturing or repair facility or

(ii) if Dialogic determines that it is unable or impractical to repair, replace or correct the product or software, provide a refund or credit not to exceed the original purchase price or license fee.

C. No product or software will be accepted for repair or replacement without the written authorization of and in accordance with instructions from Dialogic. Removal and reinstallation expenses as well as transportation expenses associated with returning such product or software to Dialogic shall be borne by Customer. Dialogic shall pay the costs of transportation of the repaired or replaced product or software to the destination designated in the original Order. If Dialogic determines that any returned product or software is not defective, Customer shall pay Dialogic's costs of handling, inspecting, testing and transportation. In repairing or replacing any product, part of product, or software medium under this warranty, Dialogic may use new, remanufactured, reconditioned, refurbished or functionally equivalent products, parts or software media. Replaced products or parts shall become Dialogic's property.

D. Dialogic makes no warranty with respect to defective conditions or non conformities resulting from any of the following: Customer's modifications, misuse, neglect, accident or abuse; improper wiring, repairing, splicing, alteration, installation, storage or maintenance performed in a manner not in accordance with Dialogic's or its vendor's specifications, or operating instructions; failure of Customer to apply Dialogic's previously applicable modifications or corrections; or items not manufactured by Dialogic or purchased by Dialogic pursuant to its procurement specifications. Dialogic makes no warranty with respect to products which have had their serial numbers removed or altered; with respect to expendable items, including, without limitation, fuses, light bulbs, motor brushes and the like; or with respect to defects related to Customer's data base errors. Improper packaging of product for repair will not be covered under this warranty agreement. No warranty is made that software will run uninterrupted or error free.

E. Warranty does not include:

a) Dialogic's assistance in diagnostic efforts;

b) access to Dialogic's Technical Support web sites, databases or tools;

c) product integration testing;

d) on-site assistance; or

e) product documentation updates.

These services are available either during or after warranty at Dialogic's published prices.

F. THE FOREGOING WARRANTIES ARE EXCLUSIVE & ARE GRANTED IN LIEU OF ALL OTHER EXPRESS & IMPLIED WARRANTIES (WHETHER WRITTEN, ORAL, STATUTORY OR OTHERWISE), INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. CUSTOMER'S SOLE AND EXCLUSIVE REMEDY AND DIALOGIC'S SOLE OBLIGATION HEREUNDER, SHALL BE TO REPAIR, REPLACE, CREDIT OR REFUND AS SET FORTH ABOVE.

G. IN NO EVENT SHALL DIALOGIC, ITS DIRECTORS, OFFICERS, EMPLOYEES, AGENTS OR AFFILIATES, BE LIABLE FOR ANY COSTS OR DAMAGES ARISING DIRECTLY OR INDIRECTLY FROM YOUR USE OF ANY PRODUCT INCLUDING ANY INDIRECT,

INCIDENTAL, SPECIAL, EXEMPLARY, MULTIPLE, PUNITIVE OR CONSEQUENTIAL DAMAGES, INCLUDING LOST PROFITS, WHETHER BASED ON CONTRACT, TORT (INCLUDING NEGLEGENCE), STRICT LIABILITY OR OTHER LEGAL THEORY, EVEN IF DIALOGIC, OR ANY OF ITS DIRECTORS, OFFICERS, EMPLOYEES, AGENTS OR AFFILIATES HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. IN ANY EVENT, DIALOGIC'S CUMULATIVE LIABILITY TO YOU FOR ANY AND ALL CLAIMS RELATING TO THE USE OF ANY PRODUCT SHALL NOT EXCEED THE TOTAL AMOUNT OF THE PURCHASE PRICE OR LICENSE FEES PAID TO DIALOGIC FOR SUCH PRODUCT.

H. CUSTOMER AND DIALOGIC HEREBY WAIVE THEIR RIGHT TO TRIAL BY JURY TO THE FULLEST EXTENT PERMITTED BY LAW IN CONNECTION WITH ALL CLAIMS ARISING OUT OF OR RELATED TO THIS WARRANTY, THE PRODUCTS COVERED HEREBY OR THE PERFORMANCE OF ANY PARTY HEREUNDER.

I. THIS WARRANTY SHALL BE CONSTRUED UNDER AND GOVERNED BY THE LAWS OF THE COMMONWEALTH OF MASSACHUSETTS WITHOUT GIVING EFFECT TO ANY CHOICE OR CONFLICT OF LAW PROVISION OR RULE (WHETHER OF THE COMMONWEALTH OF MASSACHUSETTS OR ANY OTHER JURISDICTION) THAT WOULD CAUSE THE APPLICATION OF THE LAWS OF ANY JURISDICTION OTHER THAN THE COMMONWEALTH OF MASSACHUSETTS. CUSTOMER SPECIFICALLY AND IRREVOCABLY CONSENTS TO THE PERSONAL AND SUBJECT MATTER JURISDICTION AND VENUE OF THE FEDERAL AND STATE COURTS OF THE COMMONWEALTH OF MASSACHUSETTS AND SUCH COURTS SHALL HAVE EXCLUSIVE JURISDICTION WITH RESPECT TO ALL MATTERS CONCERNING THIS WARRANTY OR THE ENFORCEMENT OF ANY OF THE FOREGOING.

J. THIS WARRANTY GIVES YOU SPECIFIC LEGAL RIGHTS. YOU MAY ALSO HAVE OTHER RIGHTS WHICH VARY FROM JURISDICTION TO JURISDICTION.

About this Publication

Purpose

This publication provides guidelines for using the Dialogic® CSP.

Safety Labels

The following Safety labels may appear in this information product to alert customers to avoidable hazards. The following are in the order of priority:

DANGER

Danger indicates the presence of a hazard that will cause death or severe personal injury if the hazard is not avoided.

WARNING

Warning indicates the presence of a hazard that can cause death or severe personal injury if the hazard is not avoided.

CAUTION

Caution indicates the presence of a hazard that will or can cause minor personal injury or property damage if the hazard is not avoided. Caution can also indicate the possibility of data loss, loss of service, or that an application will fail.

Conventions used

This information product uses the text conventions explained below. In addition, hexadecimal numbers are preceded by a zero and small “x.” For example, the decimal number 15 is represented in hexadecimal as 0x0F.

Convention	Description
. . .	A horizontal ellipsis in an API message indicates fields of variable length.
:	A vertical ellipsis in an API message indicates that a block of information is repeated or is variable.
n	The letter *n* is a generic placeholder for a number.
`Sans serif mono space`	Indicates a command name, option, input, output, non-GUI error, and system messages.
`Sans serif monospace italic`	Indicates a parameter name in an input message. Example: `move *.dot a: c:` *`-s`* The *`-s`* is the parameter.
Serif italic	Indicates the name of a book, chapter, path, file, or API message. Example: *`UserDirectory/Config.exe`*
Boldface	Indicates keyboard keys, key combinations, and command buttons Example: **Ctrl+Alt+Del**
Sans serif boldface	Identifies text that is part of a graphical user interface (GUI). Example: Go to the **Configuration** menu and select **Card**->**Span Configuration**

Contents

1 SNMP Introduction

Purpose This user guide describes how to install the Simple Network Management Protocol (SNMP) Agent for use with EXS SwitchKit. It also describes notifications specific to the CSP that are sent from the CSP via SNMP to the destination host. The CSP SNMP Agent was developed using the SNMP Research Emanate® Subagent Development Kit, a product of SNMP Research International, Incorporated (SRI).

> **Important!** In some instances, output from SRI tools refers to notifications as ***traps***.

SNMP Agent Overview

Overview

The SNMP User's Guide describes the SNMP Agent for use with the Low-Level Communicator (LLC) of EXS SwitchKit. Throughout this document, the software is referred to as the SNMP Agent. The SNMP User's Guide provides a brief introduction to SNMP, but it assumes that you have a basic knowledge of SNMP and network management.

SNMP

The SNMP provides a way to control and monitor a variety of equipment using one network management protocol. To do this, SNMP uses a number of common Management Information Bases (MIBs) and some company-specific MIBs to allow vendors to provide specific information about the equipment being managed. A MIB is a collection of objects that can be accessed via SNMP. Dialogic includes a private MIB on your installation CD that provides information about the CSP. After you install the SNMP software, you can find the csp.mib in *srconf/agt/csp.mib*.

SNMP History

In May 1990, the Internet Engineering Task Force (IETF) published SNMPv1 in RFC1157. The security administration framework of SNMPv1 is based on the community model. This algorithm allows the agent and Network Management Station (NMS) to agree on a community string that is passed along for authentication, but this string is not encrypted or in any way protected. Following its publication, the IETF immediately formed a new working group to update SNMP to offer a more robust security administration framework. The working group was unable to settle on one standard, so two different standards emerged: SNMPv2c and SNMPv2. But neither of these standards addressed the security issue.

The NOTIFICATION-TYPE replaced the TRAP-TYPE from SNMPv1 and a new type of notification named INFORM was introduced to implement reliable event announcement between Network Management Stations. In 1998, the IETF published SNMPv3, which offers the features of SNMPv2c and user-based security. This security algorithm attempted to incorporate the best ideas of the SNMPv2c variants that had emerged in the mid-1990s.

Dialogic's SNMP Compliance

The Dialogic SNMP MIB is defined according to RFC 2578 (Structure of Management Information (SMI) for Version 2 of SNMP), while the Dialogic SNMP Traps are defined in both RFC 1215 (version 1) format and RFC 1902 (version 2: see NOTIFICATION-TYPE).

The current release of the Dialogic SNMP Agent supports the ability to receive events and alarms from the CSP, and then create and distribute these as SNMPv1 notifications or SNMPv2c notifications. By default, SNMPv2c notifications are sent. You can edit the snmpd.cfg file so that SNMPv1 notifications can be sent. See *System Customization (4-1)*. Part of the installation is the configuration file, snmpd.cnf, which defines the community strings and passwords that are currently valid for this agent. By default, any NMS running SNMPv1, SNMPv2c, or SNMPv3 can perform get or get-next queries using the community string `public`. By default, any NMS running SNMPv1, SNMPv2c, or SNMPv3 can perform set queries using the community string `CSPAdmin`.

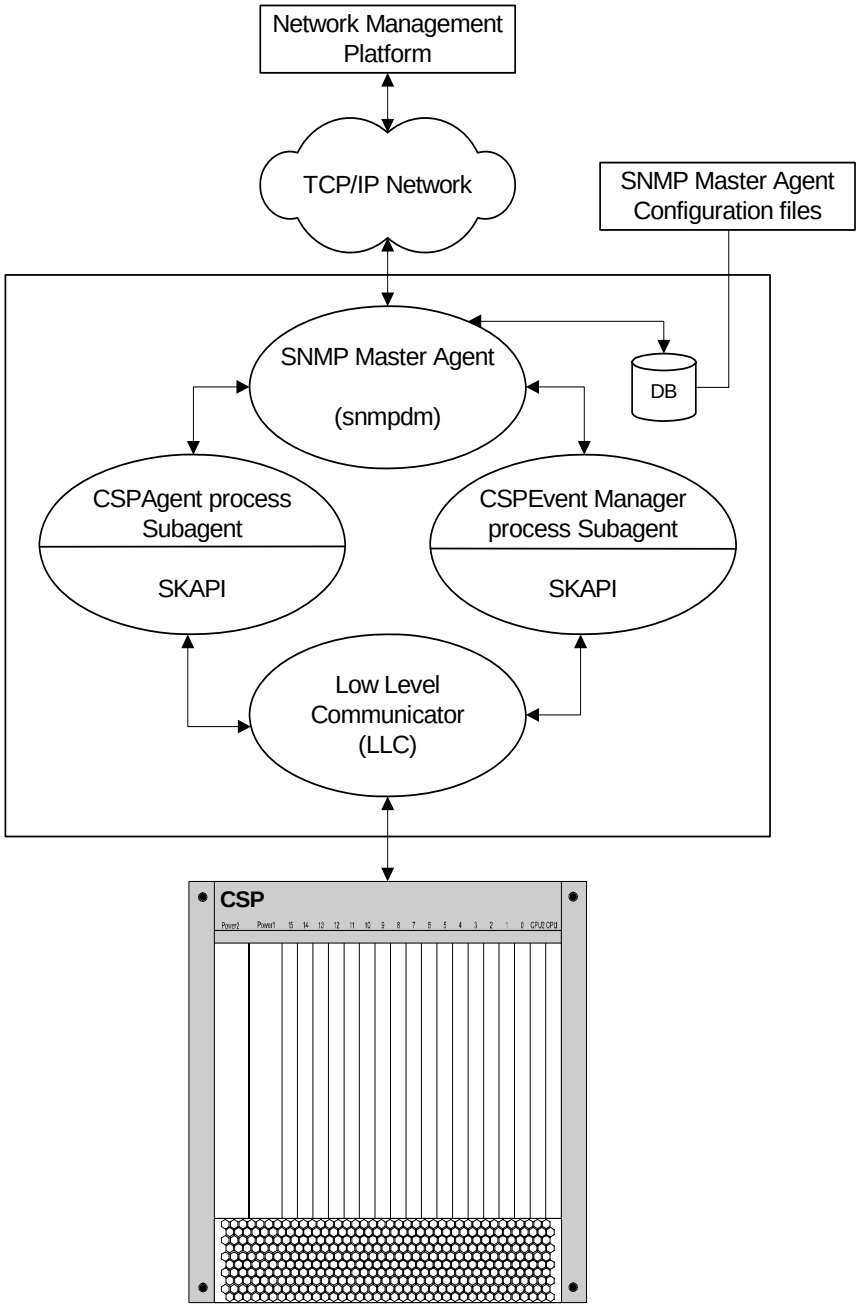
CAUTION

Do not run the Network Management System (NMS) and the SNMP Agent software on the same system for any reason. As this diagram shows, the NMS and the Host system are separate physical systems connected by the network.

Do not run the SNMP Agent on a Windows® system that has or ever had HP OpenView installed.

The following shows the flow of information between a Network Management System and devices being managed:

Figure 1-1 SNMP Setup Diagram



SNMP_Architecture_051401.vsd

Initial requests for data are generated by the Network Management System (HP OpenView in this example). SNMP requests travel to the destination system (the host system in this example) where the SNMP

master agent (snmpdm) receives them. The master agent sends the SNMP requests to the appropriate subagent (either CSPAgent process or CSPEventManager process) where the SNMP message processing occurs. Based on the information contained in the request, the subagent process either processes the request itself or requests information from the Low Level Communications (LLC).

Completed SNMP responses, successful or not, are routed back through the SNMP master agent (snmpdm) which is responsible for sending the response message back to the originating system.

When an event occurs on either the switch or host system that warrants outside notification, the CSPEventManager process subagent creates an SNMP notification and sends it to the master agent. The master agent then forwards the SNMP notification to the configured destination(s).

Supported Operating Environments

EXS SwitchKit Run-time Environment

The table that follows shows the operating systems and versions supported by EXS SwitchKit in a run-time environment.

Table 1-1 Operating System Requirements

Operating System	EXS SwitchKit is supported on...
Solaris	Version 8.0 for the SPARC platform
Windows NT®	Version 4.0, service pack 5 or greater
Windows® 2000	Windows® 2000

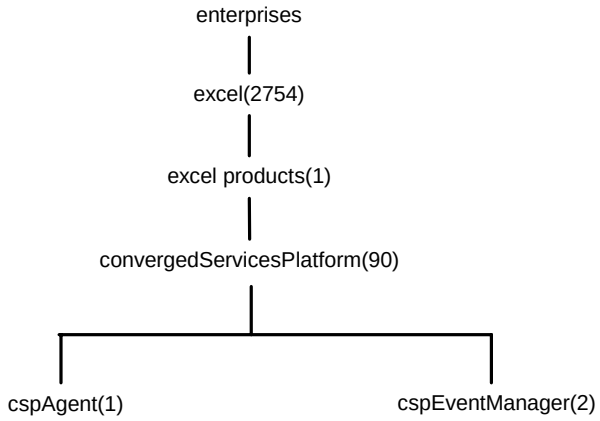
Hardware Support

All of the cards included with this CSP release are recognized by the SNMP Agent and have alarms associated with them. When you do a “get” of cspCardType, you generate a response, for example ds3 (31). See the file csp.mib. Common card groups within cspHardware contain objects representing the card. Highly specialized cards, such as: the CSP Matrix Series 3 Card, EXNET-ONE, T-ONE, E-ONE, DS3, SS7, ISDN and VDAC; contain specific groups within the cspProtocol group.

CSP SNMP Agent MIB

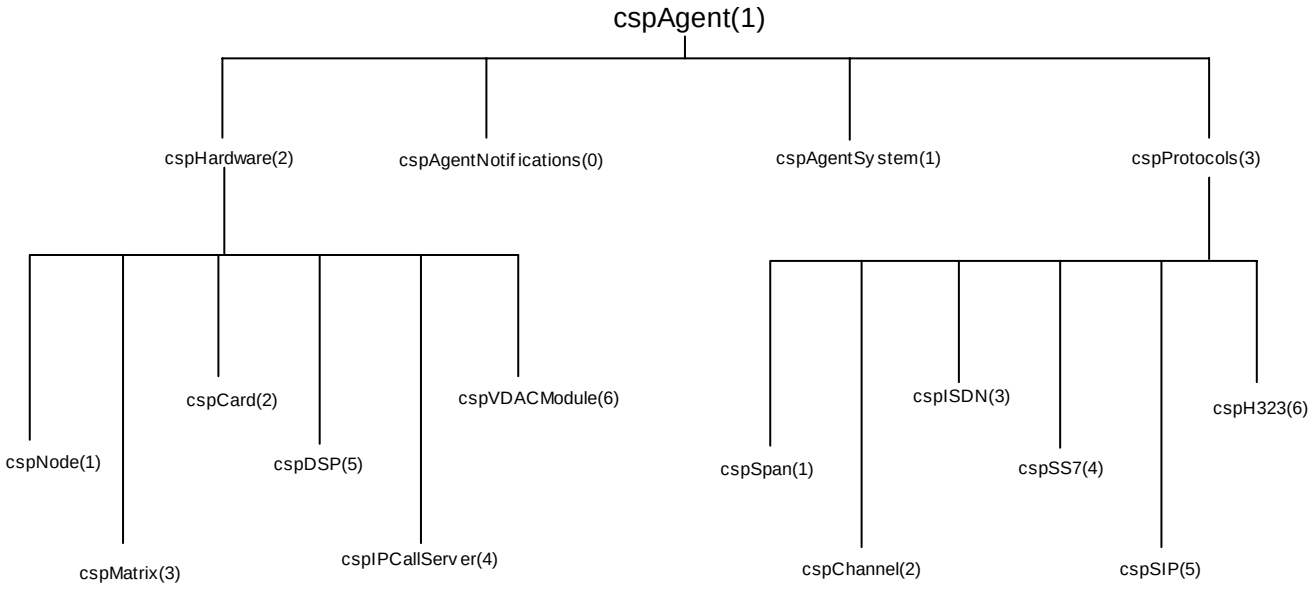
Overview For the Management Information Bases (MIBs), all objects are organized into two major groups underneath the node, Converged Services Platform (excelProducts 90). The cspAgent group contains all of the notifications, objects, and tables associated with the CSPAgent process. The cspEventManager group contains all of the notifications, objects, and tables associated with the CSPEventManager process.

Important! In this user guide, references to CSPAgent or CSPEventManager (note the capitalization) indicate the executable files. References to cspAgent or cspEventManager indicate the ASN.1 group names in the csp.mib file. In some instances, output from SNMP tools refers to notifications as ***traps***.



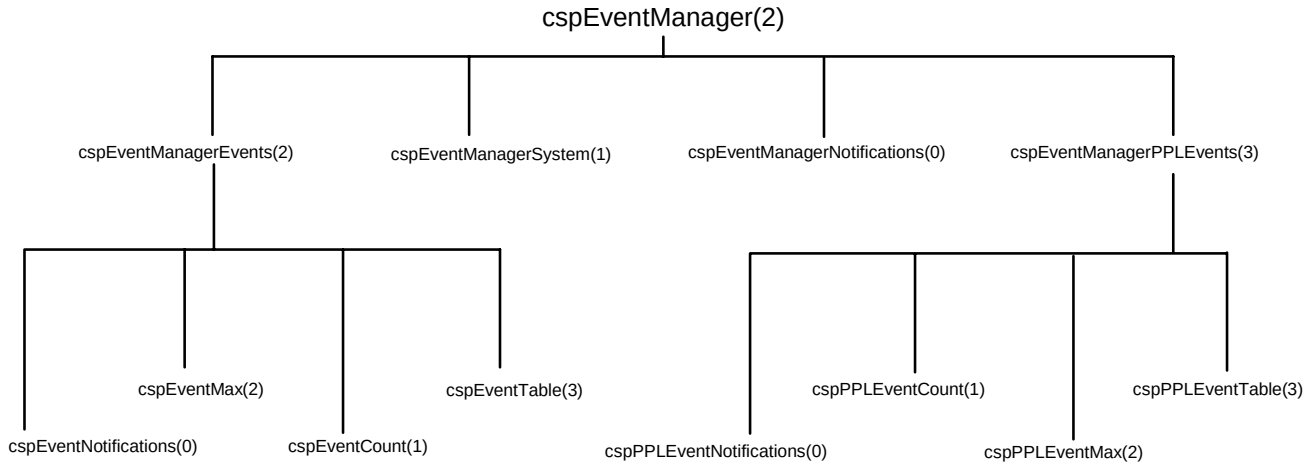
cspAgent This branch is divided into four groups:

Group	Description
cspAgentNotifications(0)	For all notifications sent by the CSPAgent process
cspAgentSystem(1)	For agent configuration, the only object currently present is cspAgentSystemVerbose
cspHardware(2)	Objects describing proprietary CSP hardware (includes chassis, cards, dsps, and matrix cards.)
cspProtocols(3)	Objects describing public/standard protocol stacks on CSP hardware. (such as, T1, ISDN, and SS7.)



cspEventManager This branch is divided into four groups:

Group	Description
cspEventManagerNotifications(0)	For the connection notifications sent by the CSPEventManager process.
cspEventManagerSystem(1)	For event manager configuration. Currently, the only object present is cspEventManagerSystemVerbose.
cspEventManagerEvents (2)	For notifications and cspeventTable used to process most switch-initiated events.
cspPPLEventManagerEvents(3)	For notifications and pplEventTable used to process the *PPL Event Indication* message



One-based Addresses Zero-based indexing of tables is not allowed because of the constraint of the ASN.1 syntax in which all SNMP MIBs are defined. All addressing on the CSP is zero-based. All addressing information represented as an index in the MIB is one-based, meaning that it has been converted from the zero-based representation on the switch to the one-based MIB representation by adding a one. All index objects in the cspAgent group are represented this way. In other words, if you would like to find out card information about the card in slot 0 on node 5, you would submit a command similar to the following, with the specific indices 6.1:

```
[c:\]getone -v2c 135.119.36.107 public cspCardType.6.1
cspCardType.6.1 = dsp_one(21)
```

In the cspEventManager group, addressing of switch entities is handled mostly as variables bound to notifications. As such, these addresses are zero-based because they are not table indices. When the card in slot 0 on node 5 comes into service, the cspCardUp notification would contain the following information:

```
Received SNMPv2c Trap:
Community: public
From: 127.0.0.1
sysUpTime.0 = 30773
snmpTrapOID.0 = cspCardUp
cspEventNode = 5
cspEventSeverity = informative(1)
cspEventAddressType = slot(3)
cspEventAddressData1 = 0
cspEventStatus = cardInService(5)
```

Event Handling

The SNMP uses User Datagram Protocol (UDP) as its transport protocol. UDP is not a connection-based, guaranteed delivery network protocol. Therefore, notifications are not guaranteed delivery. So notifications are not always the most reliable way to find out about extraordinary events occurring on a device. Because of this, all switch events are logged and archived by the CSPEventManager process. All switch events are logged to a file named, CSPEventManager.log. The file is created in the directory: $(SK_LOG_DIR), if defined. If SK_LOG_DIR is not defined, this log is created in the directory from which the CSPEventManager process is executed. This log is managed in a similar way to SwitchKit logs. All switch events are archived to either the cspEventTable or the cspPPLEventTable. The cspPPLEventTable archives all occurrences of PPL Event Indication messages. The cspEventTable archives all other switch-initiated messages that are defined as notifications under cspEventNotifications (cspEventManagerEvents 0). Both tables have associated with them a count object and a max object. The count object indicates how many events have been archived in the table at a given time. Count objects are ideal for event polling. The max object indicates the maximum number of events that are archived before the table index rolls over. The max object is a maximum of 500, but is read/writable, and you can set it to as low as ten. You can set this object by issuing the following command at the command line:

```
setany –v2c localhost cspAdmin cspEventMax.0 –i  300
```

Dynamic Addressing Scheme

Addressing information on the switch is handled in the EXS API by a dynamic addressing scheme using Address Information Blocks (AIBs). When event-driven messages arrive from the CSP, entities involved in

the event are addressed by the AIB. The AddressType textual-convention is an attempt at representing this dynamic scheme in the static environment of ASN.1. AddressType is defined in csp.mib. The AddressType enumerates all of the possible entities, including:

- Some hardware
- Some software that is addressed by alarms
- PPL Event Indications
- Card Status Reports
- Server Status Change messages

An AddressType object is bound to notifications and is archived in the event tables. The address type object is always accompanied by three data objects. The value of these data objects depends on the value of the AddressType Object. For example, in the cspCardUp notification above, cspEventAddressType equals slot(3). The definition of AddressType in csp.mib contains a table that explains the meaning of each data object. For slots, cspEventAddressData1 represents the zero-based slot number.

In the following example, the cspPPLEventAddressType is ss7Link(9). In the AddressType table, it implies that cspPPLEventAddressData1 is the SS7 Stack and cspPPLEventAddressData2 is the SS7Link. Because SS7 Links need only two data objects to represent the address, the third data object is unused and initialized to (-1).

```
Received SNMPv2c Trap:
Community: public
From: 127.0.0.1
sysUpTime.0 = 191067
snmpTrapOID.0 = cspPPLEvent
cspPPLEventNode = 5
cspPPLEventComponentID = ss7mtp3_lsac(46)
cspPPLEventID = 1
cspPPLEventDescription = PPLEventIndication Evt=0x 1 on
   LinkID = 1, StackID=0, Comp=0x2E: MTP3 LSAC: Link
   Activation Failure
cspPPLEventAddressType = ss7Link(9)
cspPPLEventAddressData1 = 0
cspPPLEventAddressData2 = 1
cspPPLEventAddressData3 = -1
```

To obtain a full list of the CSP MIB objects, you can use your Network Management Station to do a MIB walk.

SNMP Components

Overview The SNMP Agent enables you to monitor the CSP using SNMP. You can distribute events and monitor card status across the enterprise.

SNMP Components The following tables show the executables, tools, and files required to provide the management and monitor functions.

CAUTION

Do not install the SNMP Agent on a system that has or ever had HP OpenView Network Node Manager installed.

MIB File	Description	Default Directory
csp.mib	Converged Services Platform private MIB.	winnt:snmp\srconf\agt Unix:/etc/srconf/agt

SNMP Module	Description	Default Directory
CSPAgent.exe	Dialogic-developed process. Responds to requests for objects under the cspAgent branch in the csp.mib. Generates notifications defined under the cspAgent branch.	Windows®: snmp\srconf\bin Unix: /etc/srconf/exe
CSPEventManager.exe	Dialogic-developed process. Responds to requests for objects under the cspEventManager branch. Generates notifications defined under the cspEventManager branch.	Windows®: *snmp\srconf\bin* Unix: */etc/srconf/exe*
snmpdm.exe	SNMP master agent purchased from SNMP Research. This process receives SNMP requests from an end user (possibly an NMS) and routes the request to the appropriate subagent process (in this case either CSPAgent or CSPEventManager). It routes SNMP responses and SNMP notifications to the appropriate destinations.	Windows: *snmp\srconf\bin* Unix: */etc/srconf/exe*
msnsaagt.exe	Third party purchased process. Responsible for processing all MIB-II queries in Windows®.	Windows®: *snmp\srconf\bin*

SNMP Module	Description	Default Directory
mib2agt.exe	Third party purchased process. Responsible for processing all MIB-II queries in Solaris SPARC.	Unix: */etc/srconf/exe*

SNMP Tools	Description	Default Directory
getmany.exe	Third party purchased component. Tool used by customers to perform a mib-walk on either sub-agent.	Windows®: *snmp\srconf\bin\exe* Unix: */etc/srconf/dir*
getnext.exe	Third party purchased component. Tool used by customers to send a get-next SNMP-PDU to either sub-agent.Specifying a node number larger than any present in the system may result in a query timeout. This can be avoided by specifying a valid node number, or using the timeout parameter (-timeout 30) when executing getnext.exe.	Windows®: snmp\srconf\bin\exe unix: /etc/srconf/dir
getone.exe	Third party purchased component. Tool used by customers to send a get SNMP-PDU to either sub-agent.	Windows®: snmp\srconf\bin\exe unix: /etc/srconf/dir
setany.exe	Third party purchased component. Tool used by customers to set SNMP information accessible from CSPAgent.	Windows®: *snmp\srconf\bin\exe* Unix: */etc/srconf/dir*
traprcv.exe	Third party purchased component. This process is used to receive SNMP notifications generated by the CSPAgent and CSPEventManager on a host. It is typically used to check the system and make sure notifications can be received. It would be used only if an NMS was unavailable to receive notifications.	Windows®: *snmp\srconf\bin\exe* Unix: */etc/srconf/dir*

Configuration Files	Description	Default Directory
snmpd.cnf	Configuration file used by the SNMP master agent. Contains SNMP notification destination information and user security information. Refer to the *System Customization (4-1)* for detailed information on how to update this file.	Windows:snmp\srconf\agt Unix:/etc/srconf/agt

Configuration Files	Description	Default Directory
snmpinfo.dat	Read-only configuration file used by the SNMP master agent. Contains information on available MIB objects. ***Do not change this file!***	Windows:snmp\srconf\mgr Unix:/etc/srconf/mgr
mgr.cnf	snmpdm configuration file. ***Do not change this file.***	Windows:snmp\srconf\mgr Unix:/etc/srconf/mgr
hpov__csp.conf	A configuration file that configures alarms in HP OpenView Network Node Manager. ***Do not change this file!***	Windows:snmp\srconf\mgr Unix:/etc/srconf/mgr

Valid Command Line Arguments

The SNMP Agent supports one command line argument which prints out version information:

```
-v
```

Assumptions

Because the SNMP components are EXS SwitchKit applications, they require a connection to the LLC to receive information, events, and alarms from the CSP.

The SNMP master agent (snmpdm.exe) and the SNMP subagent processes must reside on the same physical system. There is no such requirement for the SNMP subagent components and the LLC.

Do not run the Network Management System and the SNMP software on the same system for any reason. The NMS and the Host system are separate physical systems connected by the network.

Viewing CSP Alarms in HP OpenView

Purpose

If you are using HP OpenView Network Node Manager (NNM) to manage your CSP, this procedure will help you format the alarms.

By default, NNM does not print out notifications to the alarm viewer log window. If you want to enable this printout, you must update the *trapd.conf* file that is part of NNM. You can do this by changing the configuration through the **Event Configuration** dialog box for each notification. Dialogic has provided a configuration file that can be installed in the NNM so you do not need to configure each notification individually.

Before you begin

Start NNM. Make sure that the bin directory of the NNM installation is in your path.

Steps for Viewing and Formatting Alarms

The steps below describe how to load a MIB and install a configuration file so that you can view alarms.

1 From the **Options** menu, go to **Load** → **Unload MIBs: SNMP**.

2 Click **Load**.

3 Open *your installation/snmp/srconf/agt/csp.mib.*

4 Get to a command line prompt.

5 Change directory to *your installation./snmp/srconf/mgr.*

Enter the following commands:

```
xnmevents -replace hpov_csp.conf
xnmevents -event
```

END OF STEPS

2 Installation

Purpose This chapter contains the procedures for the installation of the SNMP Agent on Windows® and Solaris SPARC systems.

Installing the SNMP Agent on Windows®

Purpose

This procedure describes the installation of the SNMP Agent on a Windows® system.

Before you begin

SwitchKit® must be installed before you start the SNMP installation. Please refer to the installation instructions in the *SwitchKit Development Environment Installation and Maintenance Guide*.

If you have a previous version of the SNMP Agent installed on your system, you must first remove that installation with the Add/Remove Programs utility under the Control Panel.

Installing SNMP on Windows®

To install the SNMP Agent, follow the steps below:

CAUTION

You must reboot your system after the installation in order to run all SNMP Agent components properly.

1 Log on to your target machine with an administrative account.

2 Insert the EXS SwitchKit Installation CD. In the folder: */snmp/windowsnt;* double-click the file *csp*S*nmp.exe* to start the installation. During the installation, the following environment variables are created on your system:

- SR_MGR_CONF_DIR
- SR_AGT_CONF_DIR
- SR_BIN_CONF_DIR

3 Refer to the verification procedures after the installation is finished.

4 Reboot your machine.

END OF STEPS

Verifying the SNMP Installation on Windows®

Purpose

This procedure describes the verification of your SNMP installation on a Windows® system.

Before you begin

SwitchKit® must be installed and the LLC needs to run. Refer to the *SwitchKit Development Environment Installation and Maintenance Guide* for further information. In some instances, output from SNMP tools refers to notifications as ***traps***.

Installation Verification on Windows®

To verify your installation follow the steps below:

1 To verify that the ENANATE® Master Agent installed successfully, open the *snmpdm.out* file with a text editor. The file is located in your installation directory, which by default is *C:\Program Files\Cantata \SNMP\srconf\bin*.

The first entry in the file should read as follows:

```
# SNMP EMANATE Master Agent service installed
   successfully.
```

If you don't see this entry, refer to the troubleshooting section and see *LongPathToShortPath Error (3-4)*

2 Start the Trap Receive program by doing one of the following:

- Click **Start→Programs→ SNMP→Trap Receive**
- Open Windows® Explorer and go to your installation directory, which by default is: *C:\Program Files\Cantata\SNMP\srconf\bin*. Double-click *traprcv.exe*. A new window opens with the following message:

```
# Waiting for traps.
```

If you don't see this message, refer to the troubleshooting section.

3 Start the cspAgent by doing one of the following:

- Click **Start→Programs→ SNMP→ CSPAgent**

- From the same directory, double-click *cspAgent.exe*. If the process is started successfully, the following notifications appear in the **traprcv** window:

```
Received SNMPv2c Trap:
Community: public
From: 127.0.0.1
sysUpTime.0 = 336668
snmpTrapOID.0 = cspAgentMasterAgentConnect

Received SNMPv2c Trap:
Community: public
From: 127.0.0.1
sysUpTime.0 = 336673
snmpTrapOID.0 = cspAgentLLCConnect
```

4 Start the cspEventManager by doing one of the following:

- Click **Start→Programs→SNMP→ CSPEventManager**
- From your installation directory, by default *C:\Program Files\Cantata\SNMP\srconf\bin*, double-click *CSPEventManager.exe*. If the process is successfully started, the following notifications appear in the **traprcv** window:

```
Received SNMPv2c Trap:
Community: public
From: 127.0.0.1
sysUpTime.0 = 356816
snmpTrapOID.0 = cspEventManagerLLCConnect

Received SNMPv2c Trap:
Community: public
From: 127.0.0.1
sysUpTime.0 = 356820
snmpTrapOID.0 = cspEventManagerMasterAgentConnect
```

5 Open a **Command Prompt** window and type the following:

```
# cd C:\Program Files\Cantata
   \SNMP\srconf\bin
# getmany -v2c localhost public cspCardType
```

This indicates that you can communicate with the CSP and see information pass back from the CSP. The output from this command should be as follows:

```
cspCardType.6.1 = dsp_one(21)
cspCardType.6.2 = t_one16_span(24)
cspCardType.6.4 = mfdsp(3)
```

```
cspCardType.6.5 = e_one16_span(25)
cspCardType.6.7 = j_one16_span(30)
cspCardType.6.9 = ss7Series3(113)
cspCardType.6.10 = ss7Series3(113)
cspCardType.6.16 = exnet_one(84)
cspCardType.6.17 = lowerFanTray(248)
cspCardType.6.19 = powerSupply(240)
cspCardType.6.20 = midplane(250)
cspCardType.6.21 = ex_cpuIO(243)
cspCardType.6.22 = ex_cpuIO(243)
cspCardType.6.33 = matrixControllerSeries3(115)
cspCardType.6.53 = e_one120ohmStandbyIO(229)
cspCardType.6.54 = e_one120ohmRedundantIO(226)
cspCardType.6.57 = ccsIOSeries3(216)
cspCardType.6.58 = ccsIOSeries3(216)
```

END OF STEPS

Reference You can now change the destination of the notifications, the Structure of Management Information (SMI) version of the notification sent or change the community string, see *System Customization (4-1)*.

Installing the SNMP Agent on a Solaris SPARC

Purpose

This procedure describes the installation of the SNMP Agent on a Solaris SPARC system.

Before you begin

If an SNMP Agent is already installed, you must first uninstall it before continuing. Please refer to the procedure *Uninstalling the SNMP Agent on Solaris SPARC*. If you are not sure if there is a previous version installed, check for one with the following command:

```
> /etc/srconf/exe/CSPAgent -v
```

If this command works, then there is an SNMP agent already installed. If snmpdx process or other SNMP processes are running, then stop the processes and uninstall them.

EXS SwitchKit must be installed and the LLC needs to run. Refer to the *EXS SwitchKit Installation and Maintenance Guide* for further information.

Installing SNMP on Solaris SPARC

To install the SNMP Agent, follow the steps below:

1 Log in as user root.

2 Run the installation script with the command:

#./install.sh

3 If you try to run the script and the *tar* file is not found, the following message is displayed:

```
# Installation media not found
```

4 The following shows the screen output of InstallAnywhere. When you are asked to press enter to continue, do so:

```
-bash-3.00# ./install.sh
Preparing to install...
Extracting the JRE from the installer archive...
Unpacking the JRE...
Extracting the installation resources from the installer archive...
Configuring the installer for this system's environment...

Launching installer...

Preparing CONSOLE Mode Installation...

===============================================================
                         (created with InstallAnywhere by Zero G)
-------------------------------------------------------------------------------




===============================================================================
Introduction
------------

InstallAnywhere will guide you through the installation of cspSNMP.

It is strongly recommended that you quit all programs before continuing with
this installation.

Respond to each prompt to proceed to the next step in the installation.  If you
want to change something on a previous step, type 'back'.

You may cancel this installation at any time by typing 'quit'.

PRESS <ENTER> TO CONTINUE:
```

```
==========================================
Pre-Installation Summary
------------------------

Please Review the Following Before Continuing:

Product Name:
   cspSNMP

Install Folder
   /etc/srconf

Disk Space Information (for Installation Target):
   Required:  94,014,803 bytes
   Available: 4,263,697,408 bytes

PRESS <ENTER> TO CONTINUE:




================================================
Installing...
-------------

[==================|==================|============
[------------------|------------------|------------------]



================================================
Installation Complete
---------------------

Congratulations. cspSNMP has been successfully installed to:

  /etc/srconf

PRESS <ENTER> TO EXIT THE INSTALLER:
-bash-3.00#
```

5 When the installation is complete, **snmpdm** and **mib2agt** will have executed. You can verify this with the following commands:

```
# ps -ef | grep snmp
# ps -ef | grep mib2
```

Verifying the SNMP Installation on a Solaris SPARC

Purpose

This procedure describes the verification of your SNMP installation on a Solaris SPARC system.

Before you begin

EXS SwitchKit must be installed and the LLC and SwitchManager need to be running, with the CSP configured. Refer to the *EXS SwitchKit Installation and Maintenance Guide* for further information. In some instances, output from SNMP tools refers to notifications as ***traps***.

Installation Verification on Solaris SPARC

To verify your installation, follow the steps below:

1 Verify that mib2agt and snmpdm are running. You can verify this with the following commands:

```
# ps -ef | grep snmp
# ps -ef | grep mib2
```

If mib2agt and snmpdm are not running, go to /etc/srconf/exe and use these commands:

```
# ./snmpdm&
# ./mib2agt&
```

2 Change directories and start the **traprcv** process with the following command:

```
# cd /etc/srconf/exe
# ./traprcv&
```

A new window appears with the following message:

```
# Waiting for traps
```

3 Verify that the environmental variables required for these processes to execute properly have been defined. Type the following command:

```
# env | grep SR
```

The command should produce the following output:

```
SR_MGR_CONF_DIR=/etc/srconf/mgr
SR_AGT_CONF_DIR=/etc/srconf/agt
```

4 If this command returns nothing, then the environmental variables are not defined. Add the following to the */etc/profile* file at the end of the file:

```
SR_MGR_CONF_DIR=/etc/srconf/mgr
export SR_MGR_CONF_DIR
SR_AGT_CONF_DIR=/etc/srconf/agt
export SR_AGT_CONF_DIR
```

5 Then, log in as **root** again, and verify using:

```
env/grep/sr
```

6 From the same directory, */etc/srconf/exe*, start the CSPAgent daemon with the following command:

```
# ./CSPAgent&
```

This command starts the process using SNMPv2c notifications as the communications protocol. If the process is successfully started, the following notifications appear in the **traprcv** window:

```
Received SNMPv2c Trap:
Community: public
From: 127.0.0.1
sysUpTime.0 = 599440
snmpTrapOID.0 = cspAgentMasterAgentConnect

Received SNMPv2c Trap:
Community: public
From: 127.0.0.1
sysUpTime.0 = 599444
snmpTrapOID.0 = cspAgentLLCConnect
```

7 From the same directory, */etc/srconf/exe*, start the CSPEventManager daemon with the command:

```
# ./CSPEventManager&
```

This command starts the process using SNMPv2c notifications as the communications protocol. If the process is successfully started, the following notifications appear in the **traprcv** window:

```
Received SNMPv2c Trap:
Community: public
From: 127.0.0.1
sysUpTime.0 = 606971
```

```
snmpTrapOID.0 = cspEventManagerLLCConnect

Received SNMPv2c Trap:
Community: public
From: 127.0.0.1
sysUpTime.0 = 606975
snmpTrapOID.0 = cspEventManagerMasterAgentConnect
```

8 At the command line, type the following:

```
# cd /etc/srconf/exe
# ./getmany -v1 localhost public cspCardType
```

The response to this command should be similar to the following:

```
cspCardType.6.1 = dsp_one(21)
cspCardType.6.2 = t_one16_span(24)
cspCardType.6.4 = mfdsp(3)
cspCardType.6.5 = e_one16_span(25)
cspCardType.6.7 = j_one16_span(30)
cspCardType.6.9 = ss7Series3(113)
cspCardType.6.10 = ss7Series3(113)
cspCardType.6.16 = exnet_one(84)
cspCardType.6.17 = lowerFanTray(248)
cspCardType.6.19 = powerSupply(240)
cspCardType.6.20 = midplane(250)
cspCardType.6.21 = ex_cpuIO(243)
cspCardType.6.22 = ex_cpuIO(243)
cspCardType.6.33 = matrixControllerSeries3(115)
cspCardType.6.53 = e_one120ohmStandbyIO(229)
cspCardType.6.54 = e_one120ohmRedundantIO(226)
cspCardType.6.57 = ccsIOSeries3(216)
cspCardType.6.58 = ccsIOSeries3(216)
```

9 To stop the processes that were started up, type at the command line:

```
# ps-ef | grep CSP
```

This gives you the CSPAgent and CSPEventManager PIDs, then type:

```
# kill -9 <CSPAgent PID> <CSPEventManager PID>
```

This kills the CSPAgent and CSPEventManager processes.

END OF STEPS

Reference You can now change the destination of the notifications, the Structure of Management Information (SMI) version of the notification sent or change the community string, see *System Customization (4-1)*.

Uninstalling the SNMP Agent on Solaris SPARC

Purpose

This procedure describes how to uninstall the SNMP Agent on a Solaris SPARC system.

Before you begin

Make sure that none of the SNMP Agent processes are currently running.

Uninstalling the SNMP Agent on Solaris SPARC

To uninstall the SNMP Agent, follow the steps below:

1 Log on as user root.

2 Change directory to:

```
/etc/srconf
```

3 Run the install script by typing the following:

```
./solarisDeinstall.sh
```

If you try to run the script and you are not logged on as root, the following message is displayed:

```
# Need to be root to execute this script
```

If this message appears, log on as root and try the command again.

If you get an output similar to the following example, the uninstall was successful:

```
User = root
Stopping emanate snmp deamon
Stopping emanate mib2agt deamon
CSPAgent not running
CSPEventManager not running
Uninstall Complete
```

END OF STEPS

Uninstalling the SNMP Agent on Windows®

Purpose

This procedure guides you through uninstalling SNMP on Windows®.

Before you begin

None of the SNMP components should be running on the system when you uninstall SNMP (CSPAgent, CSPEventManager, and Trap Receive).

If you previously saved customized files in your installation directory such as snmpd.cnf, move the files to a temporary location of your choice.

Uninstalling SNMP

Follow the steps below to uninstall SNMP on Windows® systems.

1 Log on using an administrative account.

2 If the SNMP Emanate Adapter and SNMP Emanate Master Agent are running as services, they must be stopped prior to starting the uninstall. Go to **Control Panel→Administrative Tools→Services**.

3 To remove your current *SNMP Network Management* installation from your system, use the **Add**/**Remove Programs** feature under the **Control Panel**.

END OF STEPS

SNMP Network Management is now uninstalled.

SNMP Configuration: Valid Queries

Valid Queries

The following types of getmany/setany queries are valid with the current snmpd.cnf configuration. The file *snmpd.cnf* is found in this directory: */etc/srconf\agt*.

Community strings used in SNMPv1 and SNMPv2c, and user names, used in SNMPv3 are defined in agt/snmpd.cnf. This is a configuration file, and its format is defined by SNMP Research International (SRI) to be used by the master agent. By default, queries in SNMPv1 and SNMPv2c can be made using the community string `public`. A query in SNMPv3 must be associated with the user name cspAdmin. Sets in all three versions must be associated with the user name cspAdmin.

getmany

```
getmany -v1 localhost public cspCardType
getmany -v2c localhost public cspCardType
getmany -v3 localhost cspAdmin cspCardType
```

setany

```
setany -v1 localhost cspAdmin cspAgentSystemVerbose.0-i1
setany -v2c localhost cspAdmin cspAgentSystemVerbose.0-i1
setany -v3 localhost cspAdmin cspAgentSystemVerbose.0-i1
```

3 Troubleshooting

Purpose This chapter contains possible errors and resolutions for the SNMP Agent on Windows® and Solaris SPARC systems.

Troubleshooting the SNMP Agent on Windows®

Overview

The following is a collection of errors and solutions for the SNMP Agent on a Windows® operating system.

> **Important!** In some instances, output from SNMP tools refers to notifications as ***traps***.

Error at Start-up of CSPAgent and CSPEventManager

You get error messages at start-up of *CSPAgent.exe* or *CSPEventManager.exe*. Refer to the following example output.

```
InitSubagent() connect successful
InsertIOData:bad parameter (at line 245 in file common.c)
fd=-1, buf=00341a40, n=292 (at line 246 in file common.c)
Connect trap successfully sent
InsertIOData:bad parameter (at line 245 in file common.c)
fd=-1, but=00345900, n=376 (at line 246 in file common.c)
```

Resolution

Go to **Control Panel→Administrative Tools→Services** and start SNMP EMANATE Master Agent and SNMP EMANATE Adapter. These two services should start automatically at reboot. If Windows® cannot locate them, you must ensure the following:

- The files are in your installation directory, which by default is *C:\Program Files\Cantata\SNMP\srconf\bin*.
- The location indicated in the registry under *System/CurrentControlSet/Services/msnsa* matches the path of the *msnaagt.exe* file.
- The location indicated in the registry under *System/CurrentControlSet/Services/SNMPDM* matches the path of the *snmpdm.exe* file.

If the files are not listed, try reinstalling the SNMP Agent.

Error at Start-up of traprcv.exe

If you receive these errors, it is an indicator that another process may be trying to read the SNMP notification port. Refer to the following example output.

```
traprcv:  bind: No error
traprcv:  bind: No error
traprcv:  bind: No error
traprcv:  bind: No error
traprcv:  bind: No error
Waiting for traps.
```

```
ParseType, past end of packet.
   at line 797 in file prse_pkt.c
process_trap: Error parsing packet
   at line 351 in file traprcv.c
ParseType, past end of packet.
```

Resolution

- Stop any process running, such as SNMPc, HP OpenView, or SNMP Trap Service.
- To start, *Traprcv.exe* needs the files *mrg.cnf* and *snmpinfo.dat* in your installation directory, which by default is *C:\Program Files\Cantata\SNMP\srconf\mgr*. Make sure that they are installed correctly.
- The installation should have set the environment variable SR_MGR_CONF_DIR to the installation directory. Make sure that the variable is listed.

No Traps at Start-up of Agent

If you do not receive any notifications after starting up the *CSP Agent.exe*, the reasons could be as follows:

- You did not reboot your system after you installed the SNMP Agent.
- The environment variables are not set correctly.

Resolution

Try to reboot your system.

Check that the environment variables are set correctly and point to your installation directory.

```
SR_MGR_CONF_DIR
SR_AGT_CONF_DIR
```

Check the file *c:\temp\snmpd.log*. The file contains warnings or errors from a routine called ProcessConfigRecord in the snmpdm process. The warnings or errors from that routine refer to problems in snmpd.cnf. Line numbers map back to snmpd.cnf. See the following example:

From snmpd.log:

```
ProcessConfigRecord: Error, incomplete entry at line 13
   at line 772 in file scanfile.c
ProcessConfigRecord: Error, incomplete entry at line 18
   at line 772 in file scanfile.c
ProcessConfigRecord: Error, incomplete entry at line 23
   at line 772 in file scanfile.c
```

From snmpd.cnf, which is found in the directory: *C:\Program Files\Cantata\SNMP\srconf\agt*:

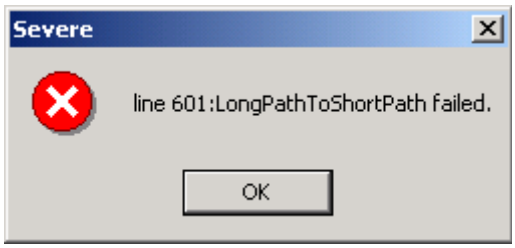
```
# Entry type: sysLocation
# Entry format: octetString
line 13: sysLocation ""
# Entry type: sysContact
# Entry format: octetString
line 18: sysContact ""
# Entry type: sysName
# Entry format: octetString
line 23: sysName ""
```

Important! snmpd.cnf must be a writable file.

To fix errors, put text between quotation marks.

LongPathToShortPath Error

You may encounter the following error during the SNMP installation, in particular, after the **choose destination location** window opens:



Resolution

To resolve this issue, delete the environment variables:

- SR_AGT_CONF_DIR
- SR_BIN_CONF_DIR
- SR_MGR_CONF_DIR

Next, restart the SNMP installation *cspSNMP.exe*.

Troubleshooting the SNMP Agent on a Solaris SPARC

Overview

The following is a collection of errors and solutions for the SNMP Agent on a Solaris SPARC system.

> **Important!** In some instances, output from SNMP tools refers to notifications as ***traps***.

Error at Start-up of CSPAgent process and CSPEventManager process

You get an error message at start-up of *CSPAgent.exe* or *CSPEventManager.exe,* stating that *snmpd.exe* and *msnaagt.exe* cannot start:

```
InitSubagent() connect successful
InsertIOData:bad parameter (at line 245 in file common.c)
fd=-1, buf=00341a40, n=292 (at line 246 in file common.c)
Connect trap successfully sent
InsertIOData:bad parameter (at line 245 in fole common.c)
fd=-1, but=00345900, n=376 (at line 246 in file common.c)
```

Resolution

The snmpdm process stopped or has problems and needs to be restarted. See *Verifying the SNMP Installation on a Solaris SPARC (2-10)*. Log in as root and stop the previous snmpdm processes. Then start snmpdm again. Do the same with the mib2agt process.

If this does not work, try uninstalling using the solarisDeinstall.sh script, and then reinstalling.

Permissions error at Start-up of CSPAgent and CSPEventManager

You get an error message at start up of CSPAgent or CSPEventManager that looks something like this:

```
AgentSocketConnect: connect: Permission denied
                              at line 174 in file uds.c
InitSubagent() connect successful
InsertIOData: bad parameter                              at
   line 247 in file common.c
fd = -1, buf = 009d3560, n = 156
                              at line 248 in file common.c
Connect trap unsuccessfully sent
InsertIOData: bad parameter                              at
   line 247 in file common.c
fd = -1, buf = 009f2d80, n = 156
                              at line 248 in file common.c
AgentSocketConnect: connect: Permission denied
                              at line 174 in file uds.c
```

Resolution

The root user must begin the subagent processes: CSPAgent process and CSPEventManager process. These messages probably indicate that another user has unsuccessfully attempted to start the subagents.

Error at Start-up of traprcv.exe

If you receive these errors, it is an indicator that another process may be trying to read the SNMP notification port. See the following example output:

```
traprcv:  bind: No error
traprcv:  bind: No error
traprcv:  bind: No error
traprcv:  bind: No error
traprcv:  bind: No error
Waiting for traps.
ParseType, past end of packet.
   at line 797 in file prse_pkt.c
process_trap: Error parsing packet
   at line 351 in file traprcv.c
ParseType, past end of packet.
```

Resolution

Stop any processes running. *Traprcv.exe* needs the files *mrg.cnf* and *snmpinfo.dat* in *"your installation directory"/SNMP/srconf/mgr* to start. Make sure they are installed correctly and SR_MGR_CONF_DIR points to that directory.

No Traps at Start-up of Agent

You do not receive any notifications after starting up the *CSP Agent*. The reason could be that the environment variables are not set correctly.

Resolution

Check that the environment variables are set correctly and that they point to your installation directory:

```
SR_MGR_CONF_DIR
SR_AGT_CONF_DIR
```

Check the file */tmp/snmpd.log*. The file contains warnings or errors form a routine called ProcessConfigRecord. The warnings or errors from that routine refer to problems in *snmpd.cnf*. Linenumbers map back to snmpd.cnf. See the example:

From snmpd.log:

```
ProcessConfigRecord: Error, incomplete entry at line 13
   at line 772 in file scanfile.c
```

```
ProcessConfigRecord: Error, incomplete entry at line 18
   at line 772 in file scanfile.c
ProcessConfigRecord: Error, incomplete entry at line 23
   at line 772 in file scanfile.c
```

From snmpd.cnf:

```
# Entry type: sysLocation
# Entry format: octetString
line 13: sysLocation ""
# Entry type: sysContact
# Entry format: octetString
line 18: sysContact ""
# Entry type: sysName
# Entry format: octetString
line 23: sysName ""
```

Important! snmpd.cnf must be a writable file.

4 System Customization

Purpose This chapter contains the procedures for customizing the SNMP Agent on Windows® and Solaris SPARC systems.

Changing Destination of Notifications on Windows®

Purpose

This section describes how to change an existing destination to a new NMS destination. To add a new destination without changing the existing destination, see the procedure *Adding a New Destination for Notifications on Windows*.

Before you begin

To deliver notifications to the correct destination, you must know whether your destination uses SNMPv1 or SNMPv2c. If you do not know which version you are using (v1 or v2), find out before you change the destination of the notifications. If you do not set the destination on a line of the file containing the correct version of SNMP, you will receive no notifications.

Changing Destination of Notifications on Windows®

To change the destination of notifications, do the following:

1 Use a text editor to open the file:
C:\Program Files\Cantata\SNMP\srconf\agt\snmpd.cnf
located in the directory, snmp\srconf\agt.

2 Search for and edit the ***snmpTargetAddrEntry*** object in the *snmpd.cnf* file. The entry will look similar to the following:

```
#Entry type: snmpTargetAddrEntry
#Format: snmpTargetAddrName (text)
#        snmpTargetAddrTDomain (snmpUDPDomain, snmpIPX
          Domain, etc.)
#        snmpTargetAddrTAddress (transport address,
          i.e. 192.147.142.254:0)
#        snmpTargetAddrTimeout (integer)
#        snmpTargetAddrRetryCount (integer)
#        snmpTargetAddrTagList (text)
#        snmpTargetAddrParams (text)
#        snmpTargetAddrStorageType (nonVolatile,
          permanent, readOnly)
#        tgtAddressMask (transport mask,
          i.e. 255.255.255.255:0)
snmpTargetAddrEntry 31 snmpUDPDomain 127.0.0.1:0 100 3
   localhost_v1 v1ExampleParams nonVolatile
   255.255.255.255:0
```

```
snmpTargetAddrEntry 32 snmpUDPDomain 127.0.0.1:0 100 3
   localhost_v2 v2cExampleParams nonVolatile
   255.255.255.255:0
```

The first 10 lines of the entry (all lines beginning with pound (#) signs) describe the format of the ***snmpTargetAddrEntry*** fields. They are informational only.

If your destination uses SNMP v1 protocol, edit the following lines:

```
snmpTargetAddrEntry 31 snmpUDPDomain 127.0.0.1:0 100 3
   localhost_v1 v1ExampleParams nonVolative
   255.255.255.255:0
```

Note the destination: localhost_v1.

If your destination uses SNMP v2 protocol, edit the following lines:

```
snmpTargetAddrEntry 32 snmpUDPDomain 127.0.0.1:0 100 3
   localhost_v2 v2cExampleParams nonVolative
   255.255.255.255:0
```

Note the destination: localhost_v2.

Do not change all these lines, only the ones that apply to the version of the protocol you are using (v1 or v2).

3 Modify the IP address on the appropriate line to the new destination IP address. The IP address of the destination is the ***snmpTargetAddrTAddress*** field. It is the fourth field in the line. The IP address must be followed by **:0**. For example, an IP address might be 192.143.43.22**:0**.

4 If localhost_v1 or localhost_v2 are not acceptable names for the new notification destination, then see the procedure *Adding a New Destination for Notifications on Windows*.

5 Save the *snmpd.cnf* file. Make a copy of this file and store it in another directory. The SNMP Master Agent overwrites modified or added entries.

6 Stop the SNMP Master Agent by opening a Command Prompt window and typing the following:

```
net stop snmpdm
```

After stopping the SNMP Master Agent, the following messages should be displayed:

```
# The SNMP EMANATE Master Agent service is stopping.
# The SNMP EMANATE Master Agent service was stopped
   successfully.
```

7 Restart the SNMP Master Agent by opening a Command Prompt window and typing the following:

```
# net start snmpdm
```

After the restart, the following message should be displayed:

```
# The SNMP EMANATE Master Agent service was started
   successfully.
```

8 You should not need to stop and restart the LLC and *traprcv*. You should now see notifications at the new destination.

END OF STEPS

Outlook Once the system is running, you can direct notifications to a corporate SNMP system such as HP OpenView or Castlerock.

Adding a Destination for Notifications on Windows®

Purpose

This procedure describes how to add a new destination for notifications on Windows®, in addition to the other destinations that are configured.

> **Important!** In some instances, output from SNMP tools refers to notifications as ***traps***.

Before you begin

To deliver notifications to the correct destination, you need to know whether your destination uses SNMPv1 or SNMPv2c. If you do not know which version you are using (v1 or v2), find out before you change the destination of the notifications. If you do not set the destination on a line of the file containing the correct version of SNMP, you will receive no notifications.

Add a New Destination for Notifications on Windows®

To add a new destination for notifications, do the following:

1. Stop the SNMP Master Agent by opening a **Command Prompt** window and typing the following:

   ```
   # net stop snmpdm
   ```

   After stopping the SNMP Master Agent, the following messages should be displayed:

   ```
   # The SNMP EMANATE Master Agent service is stopping.
   # The SNMP EMANATE Master Agent service was stopped
      successfully.
   ```

2. Using a text editor, open the file C:\Program Files*Cantata**etc**srconf**agt**snmpd.cnf* .

3. To add a new destination for notifications, copy and edit lines in the ***snmpTargetAddrEntry*** and the ***snmpNotifyEntry*** objects in this file.

4. Search for ***snmpTargetAddrEntry***. The entry will look similar to the following:

   ```
   #Entry type: snmpTargetAddrEntry
   #Format: snmpTargetAddrName (text)
   ```

```
#        snmpTargetAddrTDomain (snmpUDPDomain, snmpIPX
          Domain, etc.)
#        snmpTargetAddrTAddress (transport address,
          i.e. 192.147.142.254:0)
#        snmpTargetAddrTimeout (integer)
#        snmpTargetAddrRetryCount (integer)
#        snmpTargetAddrTagList (text)
#        snmpTargetAddrParams (text)
#        snmpTargetAddrStorageType (nonVolatile,
          permanent, readOnly)
#        tgtAddressMask (transport mask,
          i.e. 255.255.255.255:0)
snmpTargetAddrEntry 31 snmpUDPDomain 127.0.0.1:0 100 3
   localhost_v1 v1ExampleParams nonVolatile
   255.255.255.255:0
snmpTargetAddrEntry 32 snmpUDPDomain 127.0.0.1:0 100 3
   localhost_v2 v2cExampleParams nonVolatile
   255.255.255.255:0
```

The first 10 lines of the entry (all lines beginning with pound (#) signs), describe the format of the ***snmpTargetAddrEntry*** fields. They are informational only.

The last four lines of the file are the lines that you actually copy and edit. If your destination uses SNMP v1 protocol, copy and edit the following lines:

```
snmpTargetAddrEntry 31 snmpUDPDomain 127.0.0.1:0 100 3
   localhost_v1 v1ExampleParams nonVolative
   255.255.255.255:0
```

Note the destination: localhost_v1.

If your destination uses SNMP v2 protocol, copy and edit the following lines:

```
snmpTargetAddrEntry 32 snmpUDPDomain 127.0.0.1:0 100 3
   localhost_v2 v2cExampleParams nonVolative
   255.255.255.255:0
```

Note the destination: localhost_v2.

5 Copy the lines associated with SNMP v1 or v2 and paste them at the bottom of the ***snmpTargetAddrEntry***.

6 Increment the ***snmpTargetAddrName***, the second field in the line, to 33 (or to the next number in sequence).

7 Modify the IP address of the line to the new destination IP address. The IP address of the destination is the ***snmpTargetAddrTAddress*** field. It is the fourth field in the line. The IP address must be followed by **:0**. For example, an IP address might be 192.143.43.22**:0**.

8 Modify the ***snmpTargetAddrTagList***, the seventh field in the line, to the name of the new target destination. The name must be unique. Use underscores for spaces in the name. The name must be followed by **_v1** or **_v2** depending on the SNMP protocol you are using.

For example, the new name might be: Boston_v2.

9 The entry above the ***snmpTargetAddrEntry*** is the ***snmpNotifyEntry***. Search for ***snmpNotifyEntry*** in the file. It should look similar to the following:

```
#Entry type: snmpNotifyEntry
#Format: snmpNotifyName (text)
#        snmpNotifyTag (text)
#        snmpNotifyType (trap(1), inform(2))
#        snmpNotifyStorageType (nonVolatile, permanent,
          readOnly)
# snmpNotifyEntry  31 localhost_v1 trap nonVolatile
# snmpNotifyEntry 32 localhost_v2 trap nonVolatile
```

The last two lines of this entry are the lines that you copy and edit when you add a new destination for notifications. As above, if your destination uses SNMP v1, copy and edit the line:

```
# snmpNotifyEntry 31 localhost_v1 trap nonVolatile
```

If your destination uses SNMP v2, copy and edit the line:

```
# snmpNotifyEntry 32 localhost_v2 trap nonVolatile
```

10 Copy the line associated with either SNMP v1 or v2 and paste it to the bottom of the ***snmpNotifyEntry***.

11 Increment the ***snmpNotifyName***, the second field in the line, to 33 (or to the next number in sequence).

12 Modify the ***snmpNotifyTag***, the third field in the line, to match name of the new target destination that you included in the ***snmpTargetAddrEntry*** above (see Step 7). These names must match exactly.

13 Save the *snmpd.cnf* file. Save a backup copy of the file. The Master Agent will modify the file if there are any errors.

14 From the same Command Prompt window, start the SNMP Master Agent by opening a Command Prompt window and typing the following:

```
# net start snmpdm
```

After restarting the SNMP Master Agent, the following message should be displayed:

```
# The SNMP EMANATE Master Agent service was started
   successfully.
```

15 If you have stopped the CSPAgent process, restart the process by opening a Command Prompt window and typing:

```
# cd C:\Program Files\Cantata\SNMP\srconf\bin
# cspAgent
```

16 If you have stopped the CSPEventManager process, restart the process by opening a Command Prompt window and typing:

```
# cd C:\Program Files\Cantata\SNMP\srconf\bin
# cspEventManager
```

You should not need to stop and restart the LLC and *traprcv.* You should now see notifications at the new destination.

END OF STEPS

Once the system is running, you can point to a corporate SNMP system, such as HP OpenView or Castlerock.

Changing the Password on Windows®

Purpose

This procedure describes how to change the password for your SNMP Agent on a Windows® system.

Before you begin

Decide what password you want to use. The password should contain a maximum of eight alphanumeric characters.

Changing the Password on Windows®

Follow the steps below to change your SNMP Agent password on Windows®:

1 Stop the SNMP Master Agent by opening a Command Prompt window and typing the following:

```
# net stop snmpdm
```

After stopping the SNMP Master Agent, the following messages should be displayed:

```
# The SNMP EMANATE Master Agent service is stopping.
# The SNMP EMANATE Master Agent service was stopped
   successfully.
```

2 With a text editor, open *snmpd.cnf* which is found in the directory: *C:\Program Files\Cantata\SNMP\srconf\agt*.

3 Using the editor, search for `CSPAdmin` (or the current password).

Replace all instances of `CSPAdmin` (or the password) with the new password. The password should contain a maximum of eight alphanumeric characters.

4 Save the *snmpd.cnf* file. Save a backup copy. The Master Agent will modify the file if there are any errors.

5 From the same Command Prompt window, start the SNMP Master Agent by typing the following:

```
# net start snmpdm
```

After starting the SNMP Master Agent, the following message should be displayed:

```
# The SNMP EMANATE Master Agent service was started
   successfully.
```

You do not need to start and stop CSPEventManager and CSPAgent. Once the password has been changed, this is the password that you must use whenever you want to do a Set command. Changing the password only applies to the local system. If you want to change the password globally, you must do this on each destination.

6 To test the new password, open a Command Prompt window and type the following:

```
# cd c:\Program Files\Cantata\
   SNMP\srconf\bin
```

7 Type the following:

```
# setany -v1 localhost <new password> cspEventMax.0 -i
   300
```

The result of this command should be:

```
# cspEventMax.0 = 300
```

END OF STEPS

Editing Fields in Configuration File on Windows®

Purpose

This procedure describes how to edit values in the SNMP configuration file on a Windows® system.

Before you begin

Open *snmpd.cnf* with a text editor.

Editing Configuration File on Windows®

The following steps explain how to edit values such as the following in a configuration file:

- *sysLocation*
- *sysContact*
- *sysName*

1 Using the text editor, search for the name of each field you want to change. You will see an entry that looks similar to one of the following:

```
# Entry type: sysLocation
# Entry format: octetString
# sysLocation  "Concord NH"
# Entry type: sysContact
# Entry format: octetString
# sysContact  "Joe Smith"
# Entry type: sysName
# Entry format: octetString
# sysName  system_74
```

In the last line of the entry, type the new information that you want to add. Enter character strings inside quotation marks.

2 Save the *snmpd.cnf* file.

3 Restart the SNMP daemon by opening a Command Prompt window and typing the following:

```
# net stop snmpdm
```

After stopping the SNMP daemon, the following messages should be displayed:

```
# The SNMP EMANATE Master Agent service is stopping.
# The SNMP EMANATE Master Agent service was stopped
   successfully.
```

4 From the same Command Prompt window, restart the SNMP daemon by opening a Command Prompt window and typing the following:

```
# net start snmpdm
```

After restarting the SNMP daemon, the following message should be displayed:

```
# The SNMP EMANATE Master Agent service was started
   successfully.
```

You do not need to stop and start CSPEventManager and CSPAgent.

Changing the Notifications Version

Purpose

By default, the *snmpd.cnf* file configures the SNMP Research International (SRI) master agent to send SNMPv2c notifications only. You can change this so that the master agent sends the SNMPv1 notifications. This section describes how to change the notification version.

Before you begin

Open the file: *snmpd.cnf*. This file is found in the following directories:

Windows: *snmp\srconf\agt*

Unix: */etc/srconf/agt*

Changing the Notification Version

Follow the steps below to change the notification version from SNMPv2c to SNMPv1 notifications.

1 Find the line that looks like this:

```
snmpTargetAddrEntry  32 snmpUDPDomain 127.0.0.1:0 100 3
   Console
v2cExampleParams nonVolatile 255.255.255.255:0 2048
```

2 Change the field containing the entry v2cExampleParams to:

```
v1ExampleParams
```

This causes only SNMPv1 notifications to be sent.

3 If you want both versions of notifications, then duplicate the entire line:

```
v1ExampleParams
```

Then on one line, set the first entry to the following:

```
v2cExampleParams
```

On the other line set the first entry to the following:

```
v1ExampleParams
```

On the new line, increment the snmpTargetAddrName.

END OF STEPS

Changing the Destination of Notifications on Solaris SPARC

Purpose

This procedure describes how to change an existing destination to a new NMS destination.

Before you begin

To deliver notifications to the correct destination, you must know whether your destination uses SNMPv1 or SNMPv2c. If you do not know which version you are using (v1 or v2), find out before you change the destination of the notifications. If you do not set the destination on a line of the file containing the correct version of SNMP, you will receive no notifications.

Changing Destination of Notifications on Solaris SPARC

To change the destination of notifications, do the following:

1 Use a text editor to open the */etc/srconf/agt/snmpd.cnf* file.

2 To change the destination of notifications, edit the ***snmpTargetAddrEntry*** object in this file. The entry will look similar to the following:

```
#Entry type: snmpTargetAddrEntry
#Format: snmpTargetAddrName (text)
#        snmpTargetAddrTDomain (snmpUDPDomain, snmpIPX
         Domain, etc.)
#        snmpTargetAddrTAddress (transport address,
         i.e. 192.147.142.254:0)
#        snmpTargetAddrTimeout (integer)
#        snmpTargetAddrRetryCount (integer)
#        snmpTargetAddrTagList (text)
#        snmpTargetAddrParams (text)
#        snmpTargetAddrStorageType (nonVolatile,
         permanent, readOnly)
#        tgtAddressMask (transport mask,
         i.e. 255.255.255.255:0)
snmpTargetAddrEntry 31 snmpUDPDomain 127.0.0.1:0 100 3
   localhost_v1 v1ExampleParams nonVolatile
   255.255.255.255:0
snmpTargetAddrEntry 32 snmpUDPDomain 127.0.0.1:0 100 3
   localhost_v2 v2cExampleParams nonVolatile
   255.255.255.255:0
```

The first 10 lines of the entry (all lines beginning with pound (#) signs) describe the format of the snmpTargetAddrEntry fields. They are informational only.

If your destination uses SNMP v1 protocol, edit the following lines:

```
snmpTargetAddrEntry 31 snmpUDPDomain 127.0.0.1:0 100 3
   localhost_v1 v1ExampleParams nonVolative
   255.255.255.255:0
```

Note the destination: localhost_v1.

If your destination uses SNMP v2 protocol, edit the following lines:

```
snmpTargetAddrEntry 32 snmpUDPDomain 127.0.0.1:0 100 3
   localhost_v2 v2cExampleParams nonVolative
   255.255.255.255:0
```

Note the destination: localhost_v2.

Important! Do not change all these lines, only the ones that apply to the version of the protocol you are using (v1 or v2).

3 Modify the IP address on the appropriate line to the new destination IP address. The IP address of the destination is the ***snmpTargetAddrTAddress*** field. It is the fourth field in the line. The IP address must be followed by **:0**. For example, an IP address might be 192.143.43.22**:0**.

4 If localhost_v1 or localhost_v2 are not acceptable names for the new notification destination, see the procedure *Adding a New Destination for Notifications on Solaris SPARC*.

5 Save the *snmpd.cnf* file.

6 Stop the SNMP by typing the following at the command line:

```
# ps -ef | grep snmpdm
```

A message is displayed indicating the process number of the snmpdm. To kill that process from Solaris, type the following:

```
# kill -9 <process number>
```

7 Restart the SNMP daemon by typing the following at the command line:

```
# /etc/srconf/exe/snmpdm&
```

END OF STEPS

Outlook When the system is running, you can direct notifications to a corporate SNMP system, such as HP OpenView or Castlerock.

Adding a Destination for Notifications on Solaris SPARC

Purpose

This procedure describes how to set a new destination for notifications on a Solaris SPARC system in addition to other destinations that are configured.

> **Important!** In some instances, output from SNMP tools refers to notifications as ***traps***.

Before you begin

To deliver notifications to the correct destination, you need to know whether your destination uses SNMPv1 or SNMPv2c. If you do not know which version you are using (v1 or v2), find out before you change the destination of the notifications. If you do not set the destination on a line of the file containing the correct version of SNMP, you will receive no notifications.

Adding a New Destination for Notifications on Solaris SPARC

To add a new destination for notifications, do the following:

1. Stop the SNMP Master Agent by typing the following at the command line:

   ```
   # ps -ef | grep snmpdm
   ```

   A message is displayed indicating the process number of the snmpdm. To kill that process, type the following:

   ```
   # kill -9 <process number>
   ```

2. Using a text editor, open the */etc/srconf/agt/snmpd.cnf* file.

3. To add a new destination for notifications, copy and edit lines in the ***snmpTargetAddrEntry*** and the ***snmpNotifyEntry*** objects in this file.

4. Search for ***snmpTargetAddrEntry***. The entry will look similar to the following:

   ```
   #Entry type: snmpTargetAddrEntry
   #Format: snmpTargetAddrName (text)
   ```

```
#        snmpTargetAddrTDomain (snmpUDPDomain, snmpIPX
          Domain, etc.)
#        snmpTargetAddrTAddress (transport address,
          i.e. 192.147.142.254:0)
#        snmpTargetAddrTimeout (integer)
#        snmpTargetAddrRetryCount (integer)
#        snmpTargetAddrTagList (text)
#        snmpTargetAddrParams (text)
#        snmpTargetAddrStorageType (nonVolatile,
          permanent, readOnly)
#        tgtAddressMask (transport mask,
          i.e. 255.255.255.255:0)
snmpTargetAddrEntry 31 snmpUDPDomain 127.0.0.1:0 100 3
   localhost_v1 v1ExampleParams nonVolatile
   255.255.255.255:0
snmpTargetAddrEntry 32 snmpUDPDomain 127.0.0.1:0 100 3
   localhost_v2 v2cExampleParams nonVolatile
   255.255.255.255:0
```

The first 10 lines of the entry (all lines beginning with pound (#) signs), describe the format of the ***snmpTargetAddrEntry*** fields. They are informational only.

The last four lines of the file are the lines that you actually copy and edit. If your destination uses SNMP v1 protocol, copy and edit the following lines:

```
snmpTargetAddrEntry 31 snmpUDPDomain 127.0.0.1:0 100 3
   localhost_v1 v1ExampleParams nonVolative
   255.255.255.255:0
```

Note the destination: localhost_v1.

If your destination uses SNMP v2 protocol, copy and edit the following lines:

```
snmpTargetAddrEntry 32 snmpUDPDomain 127.0.0.1:0 100 3
   localhost_v2 v2cExampleParams nonVolative
   255.255.255.255:0
```

Note the destination: localhost_v2.

5 Copy the lines associated with SNMP v1 or v2 and paste them at the bottom of the ***snmpTargetAddrEntry***.

6 Increment the ***snmpTargetAddrName***, the second field in the line, to 33 (or to the next number in sequence).

7 Modify the IP address of the line to the new destination IP address. The IP address of the destination is the ***snmpTargetAddrTAddress*** field. It is the fourth field in the line. The IP address must be followed by **:0**. For example, an IP address might be 192.143.43.22**:0**.

8 Modify the ***snmpTargetAddrTagList***, the seventh field in the line, to the name of the new target destination. The name must be unique. Use underscores for spaces in the name. The name must be followed by **_v1** or **_v2** depending on the SNMP protocol you are using.

For example, the new name might be: Boston_v2.

9 The entry above the ***snmpTargetAddrEntry*** is the ***snmpNotifyEntry***. Search for ***snmpNotifyEntry*** in the file. It should look similar to the following:

```
#Entry type: snmpNotifyEntry
#Format:  snmpNotifyName (text)
#         snmpNotifyTag (text)
#         snmpNotifyType (trap(1), inform(2))
#         snmpNotifyStorageType (nonVolatile, permanent,
   readOnly)
snmpNotifyEntry  31 localhost_v1 trap nonVolatile
snmpNotifyEntry 32 localhost_v2 trap nonVolatile
```

The last two lines of this entry are the lines that you copy and edit when you add a new destination for notifications. As above, if your destination uses SNMP v1, copy and edit the following line:

```
snmpNotifyEntry 31 localhost_v1 trap nonVolatile
```

If your destination uses SNMP v2, copy and edit the following line:

```
snmpNotifyEntry 32 localhost_v2 trap nonVolatile
```

10 Copy the line associated with either SNMP v1 or v2 and paste it to the bottom of the ***snmpNotifyEntry***.

11 Increment the ***snmpNotifyName***, the second field in the line, to 33 (or to the next number in sequence).

12 Modify the ***snmpNotifyTag***, the third field in the line, to match name of the new target destination that you included in the ***snmpTargetAddrEntry*** above (see Step 7). These names must match exactly.

13 Save the *snmpd.cnf* file. Save a backup copy of the file.

14 From the same Command Prompt window, start the SNMP Master Agent by typing the following at the command line:

```
# /etc/srconf/exe/snmpdm&
```

END OF STEPS

Outlook Once the system is running, you can point to a corporate SNMP system, such as HP OpenView or Castlerock.

Changing the Password on Solaris SPARC

Purpose

This procedure describes how to change the password on a Solaris SPARC system.

Before you begin

Open *snmpd.cnf* with a text editor.

Changing the Password on Solaris SPARC

Follow the steps below to change the password on a Solaris SPARC system:

1 Stop the SNMP Master Agent by typing the following on the command line:

```
# ps -ef | grep snmpdm
```

A message is displayed indicating the process number of the snmpdm. To kill that process from Solaris, type the following:

```
# kill -9 <process number>
```

2 Using a text editor, search for `cspAdmin` (or the current password). Replace all instances of `cspAdmin` (or the password) with the new password. The password should contain a maximum of eight alphanumeric characters.

3 Save the *snmpd.cnf* file. Save a backup copy of the file.

4 From the same Command Prompt window, start the SNMP Master Agent by typing the following at the command line:

```
# /etc/srconf/exe/snmpdm&
```

You do not need to start and stop CSPEventManager and CSPAgent.

Once the password has been changed, this is the password that you must use whenever you want to do a Set command.

Changing the password applies to the local system only. If you want to change the password globally, you must do this on each destination.

5 To test the new password, type the following at the command line:

```
# cd /etc/srconf/exe
```

6 Type:

```
# setany -v1 localhost <new password> cspEventMax.0 -i
   300
```

The result of this command should be the following:

```
# cspEventMax.0 = 300
```

END OF STEPS

Editing Fields in Configuration File on Solaris SPARC

Purpose

This procedure describes how to edit fields in the *snmpd.cnf* file.

Before you begin

Open *snmpd.cnf* with a text editor.

Editing Fields on Solaris SPARC

Follow the steps below to edit fields in the configuration file on a Solaris SPARC system.

1 Open *snmpd.cnf* with a text editor.

Using the editor, search for the name of each field you want to change (sysLocation, sysContact, or sysName). You will see an entry that looks similar to one of the following:

```
# Entry type: sysLocation
# Entry format: octetString
# sysLocation  "Concord NH"
# Entry type: sysContact
# Entry format: octetString
# sysContact  "Joe Smith"
# Entry type: sysName
# Entry format: octetString
# sysName  system_74
```

In the last line of the entry, type the new information that you want to add. Enter character strings inside quotation marks.

2 Save the *snmpd.cnf* file.

3 Restart the SNMP daemon by typing the following at the command line:

```
# ps -ef | grep snmpdm
```

A message is displayed indicating the process number of the snmpdm. To kill that process from Solaris, type the following:

```
# kill -9 <process number>
```

4 Restart the SNMP daemon by typing the following at the command line:

```
# /etc/srconf/exe/snmpdm&
```

You do not need to start and stop CSPEventManager and CSPAgent.

END OF STEPS