

DIGITAL ServerWORKS™ Manager

Enterprise Management Integration Guide

Part number: ER-ENTMG-IA. A01

Digital Equipment Corporation

September 1998

The information in this document is subject to change without notice and should not be construed as a commitment by Digital Equipment Corporation. Digital Equipment Corporation assumes no responsibility for any errors that might appear in this document.

The software described in this document is furnished under a license and may be used or copied only in accordance with the terms of such license. No responsibility is assumed for use or reliability of software or equipment that is not supplied by Digital Equipment Corporation or its affiliated companies.

Copyright 1995, 1996, 1997, 1998 Digital Equipment Corporation. All rights reserved.

The following are trademarks of Digital Equipment Corporation: DIGITAL, the DIGITAL logo, OpenVMS, ClientWORKS, ServerWORKS Manager, StorageWorks, SWCC, RSM, AlphaServer, and AlphaGeneration.

The following are registered trademarks of Compaq Computer Corporation: Compaq and Compaq Insight Manager.

The following are third-party trademarks:

Hewlett-Packard, HP, and OpenView are registered trademarks of Hewlett-Packard Company.

IBM is a registered trademark and NetView is a trademark of International Business Machines Corporation.

Microsoft, MS-DOS, Windows 95, and Windows NT are trademarks of Microsoft Corporation.

Mylex is a registered trademark and Global Array Manager is a trademark of Mylex Corporation.

NetBIOS is a registered trademark of Micro Computer Systems, Inc.

Novell, NetWare, ManageWise are registered trademarks of Novell, Inc.

PATROL is a registered trademark of BMC Software, Inc.

SCO UNIX is a registered trademark licensed exclusively through Santa Cruz Operation, Inc.

TME 10 is the property of the Tivoli Corporation and International Business Machines Corporation.

Computer Associates, CA and Unicenter are registered trademarks and TNG is a trademark of Computer Associates, Inc.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company, Ltd.

Contents

Preface

Conventions Used in This Guide	x
Keyboard Conventions	xi

1 Integrating ServerWORKS Manager

Integration with Enterprise and Network Management Solutions.....	1-2
Levels of Integration.....	1-4
Maximum Integration.....	1-4
Minimum Integration	1-5
Basic Concepts of ServerWORKS Manager Integration.....	1-5
DIGITAL Servers, Agents, and MIBs	1-6
Alarms and Minimal Health.....	1-7
Intelligent Monitoring	1-7
DIGITAL Device Information	1-8

2 Integrating with HP OpenView

Installing ServerWORKS Manager Components	2-2
Goals of the Installation	2-2
Prerequisites for the Installation.....	2-3
Installing on NT	2-5
Installing on HP-UX	2-10
Learning What You Can Manage	2-13

Discovering and Managing DIGITAL Objects	2-14
Discovering DIGITAL Objects on Demand	2-15
Finding DIGITAL Servers by Attribute.....	2-15
Managing Devices Using the DIGITAL MIB.....	2-17
Creating DIGITAL-Specific Submaps.....	2-20
Displaying Server Information	2-22
DIGITAL Variables	2-25
Configuring Traps and Trap Destinations	2-29
Designating the Trap Destination.....	2-30
Forwarding Traps	2-31
Browsing DIGITAL Server Events	2-32
Viewing DIGITAL Server Events.....	2-32
Filtering DIGITAL Server Events.....	2-33

3 Integrating with Tivoli TME 10 NetView

Installing ServerWORKS Manager Components	3-2
Goals of the Installation	3-2
Prerequisites for the Installation.....	3-3
Installing on NT	3-5
Installing on DIGITAL UNIX	3-10
Learning What You Can Manage	3-13
Discovering and Managing DIGITAL Objects	3-14
Discovering DIGITAL Objects on Demand	3-15
Finding DIGITAL Servers On NT.....	3-15
Finding DIGITAL Servers On DIGITAL UNIX	3-16
Managing Devices Using the DIGITAL MIB.....	3-18

Displaying Server Information	3-20
DIGITAL Variables	3-22
Configuring Traps and Trap Destinations	3-27

4 Integrating with CA Unicenter TNG

Installing ServerWORKS Components	4-2
Goals of the Installation	4-2
Prerequisites for the Installation.....	4-3
Integrating ServerWORKS Components	4-5
Confirming the Installation	4-6
Uninstalling ServerWORKS from an NT System.....	4-7
Learning What You Can Manage	4-8
Managing DIGITAL Objects.....	4-9
Getting Device Status Using Node View.....	4-10
Generating Reports about DIGITAL Devices.....	4-12
Managing Devices Using the DIGITAL MIB.....	4-12
Configuring Traps and Trap Destinations	4-14
Designating the Trap Destination.....	4-14
Forwarding Traps	4-15

5 Integrating with Compaq Insight Manager

Managing DIGITAL Servers from the Insight Manager Console.....	5-2
Adding DIGITAL Servers to Insight Manager Device List.....	5-2
Learning What You Can Manage	5-6
Managing DIGITAL Servers with Compaq Insight Manager XE.....	5-8
Managing Compaq Servers from the ServerWORKS Manager Console.....	5-9
Launching With Context	5-9

A Compiling ServerWORKS DIGITAL MIBs into ManageWise

Compiling MIB Files.....	A-2
--------------------------	-----

B ServerWORKS Alarms and Browser

Setting Alarms	B-1
The Alarm Configuration Window and Toolbar	B-2
Minimal Health Alarms.....	B-4
User-Defined Console Alarms	B-5
Using the System Browser.....	B-6

C Advanced User Functions

Starting Applications from the Command Line.....	C-2
Starting ServerWORKS	C-3
Starting Alarm Configuration	C-4
Starting System Browser.....	C-5
Starting the Launchme Utility	C-6
Initialization File Settings	C-10
Before you Begin	C-11
Overall Settings	C-11
Pager.....	C-12
Printing	C-14
Viewing Cluster Details	C-16
Adding Historical Data Directory	C-17
Event Logger	C-18
Data Collector	C-19
Gathering Information with the Poller	C-20
Sizing the Main Window Alarm Status Pane	C-21
When Using a Novell NetWare Client	C-23

D The DIGITAL Server System MIB

Using the DIGITAL Server System MIB	D-2
Support for MIB Variables.....	D-2
Conventions Used in This Appendix	D-3
SNMP Syntax Extensions.....	D-5
Base System Configuration	D-7
svrSystemFamily.....	D-7
svrSystemModel.....	D-7
svrSystemDescr.....	D-7
svrSystemBoardFruIndex.....	D-8
svrSystemOCPDisplay	D-8
svrSystemBootedOS	D-8
svrSystemBootedOSVersion.....	D-9
svrSystemShutdownReason	D-9
SvrSystemRemoteMgrNum	D-9
svrFirmwareTable	D-9
svrFwSymbolTable	D-10
Central Processor Configuration and Utilization.....	D-11
svrCpuPollInterval	D-11
svrCpuMinPollInterval.....	D-11
svrCpuTable	D-12
svrCpuAvgTable	D-14
svrCpuCacheTable	D-16

System Memory	D-18
svrPhysicalMemorySize	D-18
svrPhysicalMemoryFree	D-19
svrPagingMemorySize	D-19
svrPagingMemoryFree	D-19
svrMemComponentTable	D-19
svrMemElementTable	D-24
System Buses and Slots	D-26
svrBusCount	D-26
SvrBusTable	D-26
svrLogicalSlotTable	D-27
svrSlotFunctionTable	D-29
Peripheral Devices	D-30
svrDeviceCount	D-30
svrDeviceTable	D-30
svrDevInterruptTable	D-32
svrDevMemTable	D-33
svrDevDmaTable	D-34
svrSerialPortCount	D-35
svrSerialPortTable	D-35
svrParallelPortCount	D-36
svrParallelPortTable	D-36
svrConsoleKeyboard	D-37
svrConsoleDisplay	D-37
svrConsolePointDevice	D-39

Physical Configuration	D-40
svrChassisType	D-40
svrChassisFruIndex	D-40
svrFruTable	D-41
Environment	D-44
svrThermalSensorCount.....	D-44
svrThermalSensorTable	D-45
svrFanCount	D-48
svrFanTable.....	D-48
svrPowerRedunEnable	D-50
svrPowerSensorCount.....	D-50
svrPowerSensorTable.....	D-51
svrPowerSupplyCount.....	D-53
svrPowerSupplyTable	D-54

Preface *P*

The *DIGITAL™ ServerWORKS™ Manager Enterprise Management Integration Guide* provides networking and information systems professionals with the information needed to integrate DIGITAL ServerWORKS Manager into an enterprise management solution.

This guide describes

- Installing ServerWORKS integration components within each supported enterprise management environment
- Discovering and managing DIGITAL servers, DIGITAL clusters, and Microsoft clusters
- Gaining access to the DIGITAL Management Information Bases (MIBs)
- Configuring and forwarding traps from DIGITAL devices to the enterprise manager's console
- Using ServerWORKS tools such as System Browser and Alarm Configuration

Use this guide in conjunction with the *DIGITAL ServerWORKS Manager Installation and User Guide*. The installation and user guide provides more detailed information about the Simple Network Management Protocol (SNMP) and the use of ServerWORKS features.

Related Information

To integrate ServerWORKS Manager effectively with an enterprise management solution, you should be familiar with

- Hardware and software prerequisites for the enterprise manager
- The user interface of the enterprise manager's console
- The documentation set for the enterprise manager

In addition, this manual assumes that you are familiar with TCP/IP networking and the Simple Network Management Protocol (SNMP).



Note: Appendixes C and D contain information that advanced users of ServerWORKS Manager or an enterprise manager may find useful. This information should be used with care.

Conventions Used in This Guide

The terms “Select” and “Choose” are used frequently in the procedures described in this guide. Both terms refer to specific mouse pointer or keyboard operations:

- **Select**—Move the mouse pointer to an item (icon, command, name) and single-click the mouse button, or use the specified set of keyboard keys to indicate a choice.
- **Choose**—Move the mouse pointer to an item and double-click the mouse button, or use the specified set of keyboard keys to start an action on the selected item.

The following icons are used in the manual:



Note: Indicates helpful information. The note can be a tip, special technique, shortcut or other information to help you use the product. A note is not a warning or caution of serious consequences.

Keyboard Conventions

To Do This	Press These Keys:
Scroll one window up or down	PAGE UP or PAGE DOWN
Go to the beginning of the list	CTRL+HOME
Go to the end of the list	CTRL+END
Move focus left or right	LEFT or RIGHT ARROW
Move one line up or down	UP or DOWN ARROW
Move to next window	CTRL+TAB
Move to previous window	CTRL+SHIFT+TAB
Go to the next field	DOWN ARROW or TAB
Go to the previous field	P ARROW or SHIFT+TAB
Go to the next group	CTRL+DOWN+ARROW
Go to the previous group	CTRL+UP+ARROW
Move the focus up or down without affecting the state of the previous line (to add or remove lines from a selected set)	SHIFT+UP ARROW or SHIFT+DOWN ARROW
Toggle the state of the focus item	SPACEBAR
Display Help	F1
Display Help (from a console window)	CTRL+ALT+F1

Preface

Integrating ServerWORKS Manager **1**

DIGITAL ServerWORKS Manager provides server administrators with a set of comprehensive tools for operating work group and departmental networks reliably. ServerWORKS expands your existing network management capabilities when it is integrated into your enterprise manager.

This chapter details the benefits of DIGITAL ServerWORKS Manager when it is integrated with an enterprise manager. This chapter addresses these topics:

- Integration with enterprise and network management solutions
- Levels of integration
- Basic concepts of ServerWORKS Manager integration

As a standalone management tool, ServerWORKS Manager discovers, monitors, and enables you to troubleshoot devices that support the Simple Network Management Protocol (SNMP). You can set environmental alarms on servers and you can track their performance over time. Devices you can manage include:

- DIGITAL and non-DIGITAL servers and clusters
- Network devices such as routers, bridges, and hubs

When components of ServerWORKS Manager are integrated with other tools it provides the power of full enterprise management.

Integration with Enterprise and Network Management Solutions

Integrating DIGITAL ServerWORKS Manager with an enterprise or network management tools means you can effectively manage DIGITAL devices in large, heterogeneous computing environments. ServerWORKS Manager interoperates with the enterprise manager, providing critical and exact information to the management console.

You use the familiar interface of the network or enterprise management solution to accomplish the following tasks:

- Discover DIGITAL servers, DIGITAL NT clusters, and Microsoft NT clusters and display them on network maps
- Use DIGITAL MIB applications to display information
- Enable intelligent monitoring of DIGITAL devices
- Run the ServerWORKS Manager System Browser (NT only)
- Configure alarms and traps (NT only)
- Browse DIGITAL device events

ServerWORKS Manager integrates with the enterprise and network management packages described in Table 1-1.

Table 1-1 Enterprise, Network, and Server Manager Console Applications

Management Package	Version and Platform
Hewlett Packard® OpenView™ Network Node Manager	<ul style="list-style-type: none"> • V5.01 on HP-UX® 10.20 • V5.01 on Windows® NT™ 4.0 (X-86 based)
Tivoli TME™10 NetView®	<ul style="list-style-type: none"> • V5.0 on DIGITAL UNIX 4.0 • V5.0 for Windows NT 4.0 (X86-based)
Computer Associates® Unicenter® TNG™	<ul style="list-style-type: none"> • V2.1 on Windows NT 4.0 (X86-based)
Compaq Insight Manager	<ul style="list-style-type: none"> • V4.01 on Windows NT 4.0, SCO UNIX 5.0.4, Netware® 3.0 and 4.11 (X86-based)
Novell® ManageWise®	Netware 3.0 and 4.11

Levels of Integration

The level of ServerWORKS Manager integration that you can achieve depends on the capabilities of the tool that you want to integrate it with. Some tools support maximum integration, while others support a minimal level of integration.

Maximum Integration

Maximum integration means you can

- Discover, display, and manage DIGITAL devices from the enterprise manager's network topology map.
- Run the ServerWORKS Manager System Browser and Alarm Configuration tools (on X86-based NT platforms only).



Note: HP-UX and DIGITAL UNIX users can view system information by running macros from the new DIGITAL pull down menu, which is added to their enterprise manager. These users can view alarms for DIGITAL servers from the Event Browser of their enterprise manager. First they must configure traps and trap destinations.

You can achieve maximum integration with these tools:

- HP® OpenView Network Node Manager
- Tivoli TME10 NetView
- CA® Unicenter TNG

You can also achieve a high level of integration with Compaq Insight Manager, but the implementation differs from what is described in this section. Compaq Insight Manager integration is described in Chapter 5 “Integrating with Compaq Insight Manager.”

Minimum Integration

Minimum integration enables you to:

- Forward SNMP traps to any other SNMP-compliant application
- Define DIGITAL servers as a specific object type or device class within an application
- Compile DIGITAL MIBs into another management application

This is the level of integration that you can achieve with Novell ManageWise.

Basic Concepts of ServerWORKS Manager Integration

ServerWORKS Manager is designed to help you get optimum performance and reliability from work group and departmental networks containing DIGITAL servers and other SNMP-compliant resources. If your goal is to manage large networks or an entire enterprise, you can integrate ServerWORKS Manager with tools designed specifically for that purpose. You may already, in fact, have standardized on such a tool. Integration offers the capabilities of those tools — plus the level of system manageability that ServerWORKS Manager provides.

Integrating ServerWORKS Manager with your enterprise manager gives more detailed information about DIGITAL devices than you would otherwise have. This information is available by:

- Choosing from new DIGITAL menu options
- Selecting new DIGITAL icons
- Using the ServerWORKS Manager System Browser (X86-based NT platforms only).

DIGITAL Servers, Agents, and MIBs

A DIGITAL server is defined as a managed system that is running the full set of DIGITAL agents. Agents are software programs installed on managed systems that collect and measure system data, transmitting the data gathered to the management tool console. Table 1-2 describes the set of DIGITAL agents. This set includes a group of specialized agents that transmit data to the Compaq Insight Manager console.

Table 1-2 DIGITAL Agents

Agent	Monitors...
Server System (svrSys)	Server configuration, status, and environmental settings.
Server Management (svrMgmt)	Alarm threshold settings, reporting alarm conditions to the network management console.
NT Cluster Management (ntcMgt)	DIGITAL NT clusters V1.1, checking the status of the cluster software and the names of the nodes that are members of the cluster.
Server Cluster (svrClu)	DIGITAL cluster products and Microsoft NT clusters.
Server NT Cluster (svrNtc)	DIGITAL cluster products and Microsoft NT clusters.
Compaq Agents for DIGITAL Servers	DIGITAL servers for Compaq Insight Manager, enabling Insight Manager to query and manage DIGITAL servers.

When you install the integration components of ServerWORKS Manager, you automatically load DIGITAL Management Information Bases (MIBs) into the enterprise manager. The enterprise manager can then discover and display DIGITAL servers on a management console's topology map. The console can also issue SNMP requests to agents located on those objects.

Integrating ServerWORKS Manager

A server agent performs these operations:

- **Get** and **Get Next** retrieve data from a managed object and send it to the management console
- **Set** changes the value of a variable associated with a managed object
- **Trap** sends a message to the management console when a change or an error occurs on the object.

For more information about SNMP and DIGITAL MIBs, refer to the *DIGITAL ServerWORKS Manager Installation and User Guide*.

Alarms and Minimal Health

When you install ServerWORKS Manager agents on DIGITAL servers and clusters running NT or Netware, you are providing an early warning system for problem detection. This is the ServerWORKS Manager minimal health feature. The ServerWORKS Manager agents provide out-of-the-box configuration of alarms for server components such as fans and disks. The minimal health agent sets alarms for environmental conditions — power supplies, voltage, fans, and temperature – and for status on processors, disks, and memory components.

For information about installing the ServerWORKS Manager agents, refer to the *DIGITAL ServerWORKS Manager Installation and User Guide*. These agents must have been installed on DIGITAL servers prior to integration.

Intelligent Monitoring

ServerWORKS Manager agents gather server and component information intelligently. Instead of relying on constant polling from the management console, agents that are installed on managed systems interrogate themselves and notify the console when a unit reaches an alarm threshold or alarm state, saving considerable network bandwidth.

DIGITAL Device Information

ServerWORKS Manager integration changes the user interface of your enterprise solution so that you can obtain exact information about DIGITAL devices.

ServerWORKS Manager adds a DIGITAL menu option to the HP OpenView and Tivoli TME10 NetView interfaces. This menu option gives you access to detailed information about DIGITAL devices. Under CA Unicenter TNG, DIGITAL options are available from a pop up menu when you click the right mouse button on a DIGITAL device.

The DIGITAL menu contains the options described in Table 1-3.

Table 1-3 DIGITAL Menu Options

Menu Option	Purpose
Discover	Discovers DIGITAL managed systems.
Alarm Configuration	Sets alarms on DIGITAL servers (available only on X86-based NT systems with DIGITAL agents installed).
System Browser	Provides information about both static and dynamic characteristics of DIGITAL servers (available only on X86-based NT systems with DIGITAL agents installed).
Environment	Displays information about a server's cooling system, power supplies, thermal sensors, and voltage sensors. This macro provides the way for users on Unix platforms to view environment information. For NT users of HP OpenView and Tivoli TME 10 Netview, it is available as an alternative way of looking at information.

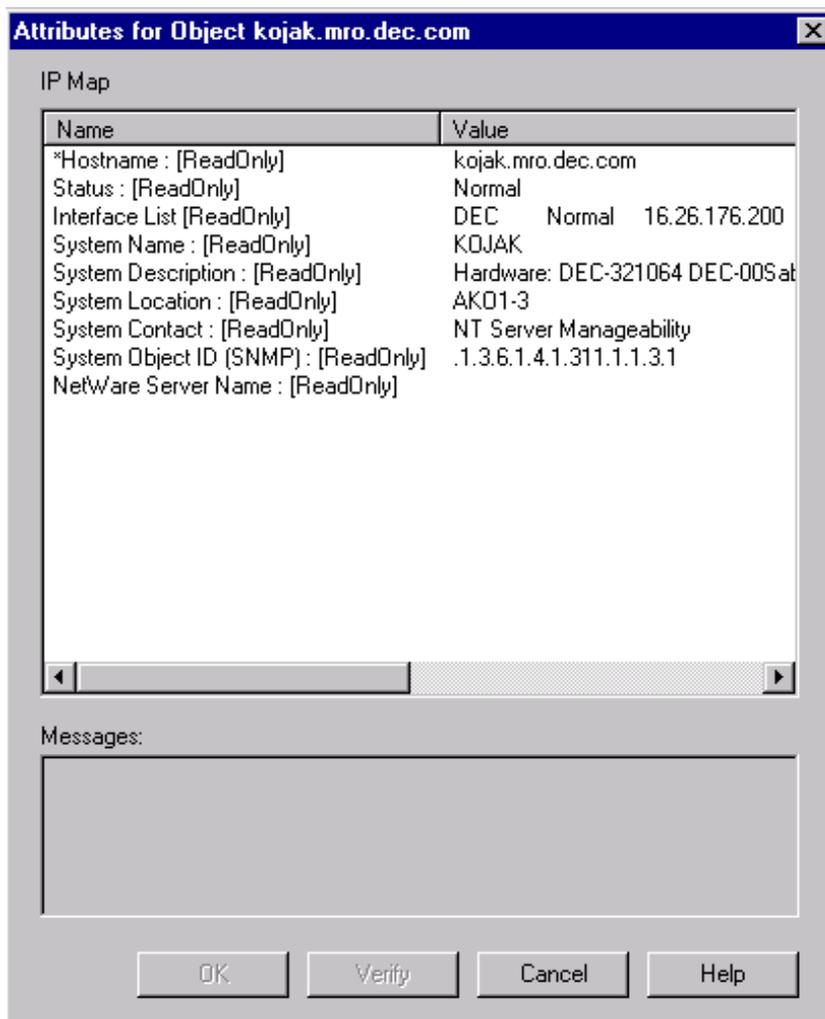
Integrating ServerWORKS Manager

Menu Option	Purpose
Memory	Displays information about an object's memory configuration. This macro provides the way for users on Unix platforms to view memory information. For NT users of HP OpenView and Tivoli TME 10 Netview, it is available as an alternative way of looking at information.
Storage	Displays information about partitions, physical and logical devices, and file systems. This macro provides the way for users on Unix platforms to view storage information. For NT users of HP OpenView and Tivoli TME 10 Netview, it is available as an alternative way of looking at information.
System	Displays information about cluster groups, cluster members, firmware, processors, and field replaceable units (FRU). This macro provides the way for users on Unix platforms to view system information. For NT users of HP OpenView and Tivoli TME 10 Netview, it is available as an alternative way of looking at information.
Graphs	Provides charts of statistical information; for example, you can view charts about voltage, temperature, file system use, processor load, packets discarded, and packet errors.

Some of the information available from the DIGITAL menu is also available from OpenView and NetView menus. The DIGITAL menu, however, presents the information in more detail.

For example, under HP OpenView, if you highlight a DIGITAL server and click Edit → Object Properties → IP Map, you view information as shown in Figure 1-1.

Figure 1-1 DIGITAL Server Information Viewed with HP OpenView

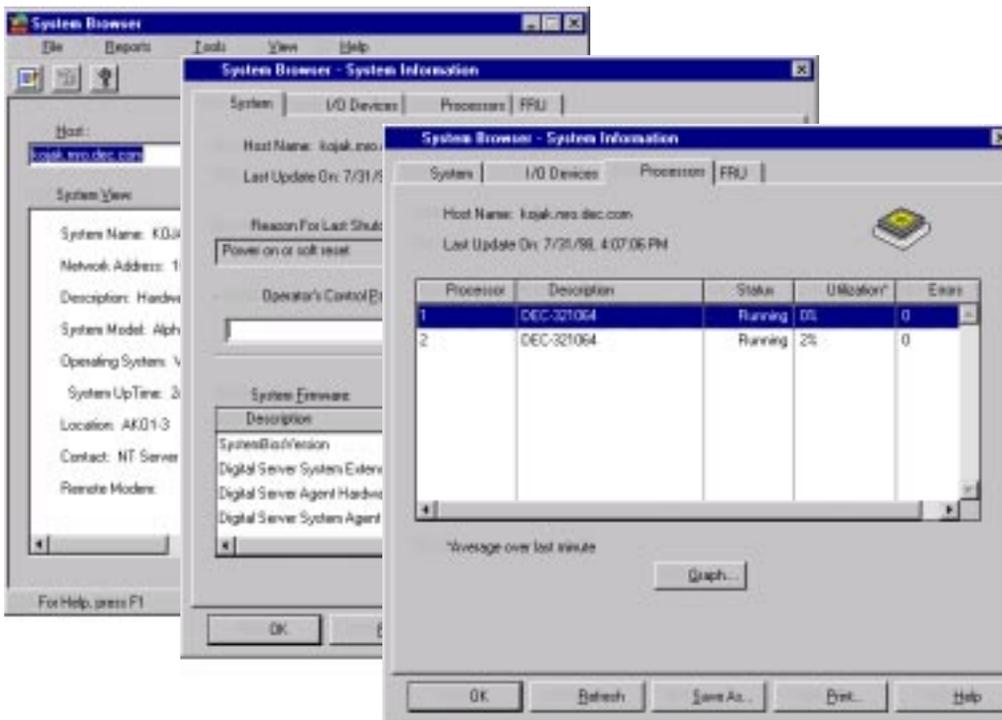


This window displays general information about the server, including the interface and its type.

Integrating ServerWORKS Manager

In contrast, if you select a DIGITAL server and click DIGITAL → System Browser and select any of the tabs shown you view a more detailed description of the network interface, as in Figure 1-2.

Figure 1-2 DIGITAL Server Information Viewed with ServerWORKS Manager



Integrating ServerWORKS Manager

Integrating with HP OpenView **2**

You can integrate DIGITAL ServerWORKS Manager into an environment that uses HP OpenView as its enterprise management solution. Integration gives you the ability to monitor and manage DIGITAL servers and NT clusters through the OpenView Network Node Manager.

To integrate ServerWORKS Manager with OpenView, refer to the topics and tasks described in this chapter:

- Installing ServerWORKS Manager components
- Learning what you can manage
- Discovering and managing DIGITAL objects
- Displaying server information
- Configuring traps and trap destinations
- Browsing DIGITAL server events

This chapter describes integration with OpenView running on X86-based Windows NT consoles and on HP-UX consoles.

Installing ServerWORKS Manager Components

To be sure you are ready to install the ServerWORKS Manager components, you should understand the goals and prerequisites for the installation.



Note: The installation instructions presented in this chapter assume that SNMP is installed on both the network management console and on the DIGITAL devices that you want to manage. For information about installing SNMP, refer to the documentation for your Windows NT or HP-UX operating system.

Goals of the Installation

You run an automated installation script to integrate ServerWORKS into OpenView Network Node Manager. The goals of the installation procedure are to accomplish the following tasks:

- Load DIGITAL MIB files
 - Server System MIB (svrsys.mib) which provides access to basic server configuration information
 - Server Management MIB (svrmgt.mib) which provides access to the instrumentation in the DIGITAL agent, including the ability to monitor and detect thresholds
 - Cluster Management MIB (ntcmgt.mib) which provides the ability to manage DIGITAL clusters
 - Server Cluster MIB (svrclu.mib) which provides information about DIGITAL cluster products and Microsoft NT clusters
 - Server NT Cluster MIB (svrntc.mib) which provides information about DIGITAL cluster products and Microsoft NT clusters



Note: The three cluster MIBs differ in how they are implemented. Use the OpenView MIB Browser tool to determine which MIB or MIBs is relevant to your enterprise.

Integrating with HP OpenView

- Install DIGITAL icons
- Modify the OpenView menu bar by adding a DIGITAL menu item
- Install trap files
- Add MIB applications
- Add DIGITAL Server OIDs to the data collector
- Install the ServerWORKS System Browser (on NT only)
- Install the ServerWORKS Alarm Configuration utility (on NT only)

Prerequisites for the Installation

Before you integrate ServerWORKS 4.0 with HP OpenView Network Node Manager, make sure you meet the software prerequisites.



Note: In addition to the software prerequisites listed in this section, you should be aware of the minimum hardware requirements to run HP OpenView. Refer to your HP OpenView documentation for this information.

You may also want to familiarize yourself with the hardware and software requirements for a complete ServerWORKS installation. Refer to the *DIGITAL ServerWORKS Manager Installation and User Guide*.

On an X86-based NT Console

To integrate ServerWORKS on an NT console requires:

- ServerWORKS CD
- Windows NT 4.0 (on an X86 processor)
- HP OpenView 5.01
- 30MB of disk space
- TCP/IP network protocol
- SNMP service

Integrating with HP OpenView

To verify that TCP/IP and SNMP are installed:

1. Access the OpenView application.
2. Click the SNMP nodes from the subnet map.
3. Choose Fault → Test IP/TCP/SNMP.

Make sure that DIGITAL agents are installed on the devices you want to manage through OpenView. For information about installing DIGITAL agents, refer to the *DIGITAL ServerWORKS Manager Installation and User Guide*.

On an HP-UX Console

To integrate ServerWORKS on an HP-UX console requires:

- HP UX 10.20
- HP OpenView 5.01
- ServerWORKS CD
- TCP/IP network protocol
- SNMP service
- 5 MB of disk space

To verify that networking and SNMP are installed, refer to the HP-UX documentation and:

1. Access OpenView.
2. Verify that Network is available.
3. Verify SNMP on a node known to have SNMP installed.

Make sure that DIGITAL agents are installed on the devices you want to manage through OpenView. For information about installing DIGITAL agents, refer to the *DIGITAL ServerWORKS Manager Installation and User Guide*.

Installing on NT

To install the ServerWORKS components on an X86-based NT console, follow these steps:

1. Log on to the console using the account used to monitor the system.
2. Close all programs that are running, including HP OpenView, mail programs, and the Microsoft Office shortcut bar.
3. Insert the ServerWORKS Manager CD in the CD drive of the console system.

As soon as the CD is engaged, the main installation screen appears. This screen prompts you to select the language you want to use to install ServerWORKS components.

If it does not appear, follow these steps:

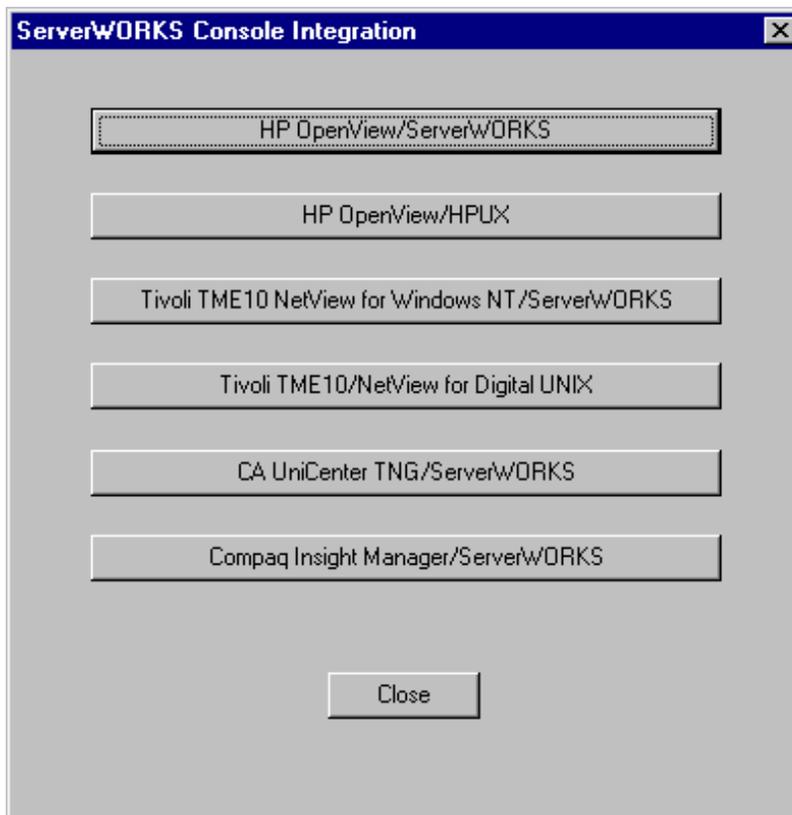
- a. From the desktop, click the Start menu.
- b. Choose Run. Enter the following path:
`<cd-rom drive>:\Autoplay.exe`
4. On the language selection screen, choose English. ServerWORKS integration is not available in other languages.

The ServerWORKS component installation screen appears.

Integrating with HP OpenView

5. Click the ServerWORKS Console Integration button.
The ServerWORKS Console Integration screen appears.
6. Click the HP OpenView/ServerWORKS button.
7. Follow the remaining prompts to complete the installation of the ServerWORKS integration components.

Figure 2-1 ServerWORKS Console Integration Screen



Integrating with HP OpenView

As the installation script runs, it displays messages telling you what is being installed. The installation script creates two log files: SWInstall.log and SWInstallErr.log. When the script finishes, it displays a message telling you whether the installation succeeded. You are prompted to view the log files.



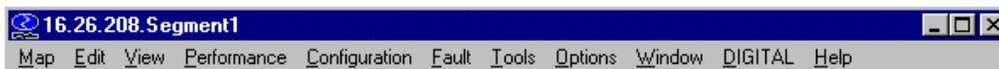
Note: The log files are deleted after you complete the installation. If the installation script encounters an error while it is running, it halts and prompts you to view the log files.

Confirming the Installation

To confirm the installation, follow these steps:

1. Start OpenView at your management console.
2. Verify that the DIGITAL menu is present on the menu bar.

Figure 2-2 DIGITAL Menu Added to the OpenView Menu Bar on NT



3. Select Edit → Find → Object By Attribute.
4. Verify that these attributes appear in the list of Object attributes:
 - DECsvrClusterType
 - DECsvrQuery Type
 - DECsvrSysType

To find DIGITAL servers by attribute, see “Discovering DIGITAL Servers,” a subsequent section of this chapter.

Uninstalling ServerWORKS Manager from an NT System

ServerWORKS Manager is tightly integrated with HP OpenView. As a result, if you need to uninstall ServerWORKS Manager integration components, DIGITAL strongly recommends that you uninstall HP OpenView. Uninstalling HP OpenView removes ServerWORKS Manager integration components. You can then reinstall OpenView.

If you choose not to follow the DIGITAL recommendation, an uninstall of ServerWORKS Manager components requires you to:

- Run an uninstall batch file
- Remove the ServerWORKS Manager database from the Registry
- Remove OID_TO_SYM and OID_TO_TYPE mapping entries

To Run the Uninstall Batch File

In the OpenView subdirectory on the ServerWORKS Manager CD, locate and run uninstal.bat.

The uninstall batch file attempts to restore your system to the state it was before you installed ServerWORKS Manager. However, you must manually remove the database and the mapping entries.

To Remove the ServerWORKS Manager Database from the Registry

1. In the Control Panel, double-click the ODBC icon.
2. Click the System DSN tab.
3. Select the SWMgrDB entry.
4. Click Remove. You are prompted to make sure you want to delete SWMgrDB.
5. Click Yes.

Integrating with HP OpenView

To Remove OID_TO_SYM Entries

1. In OpenView\conf, select the file oid_to_sym.
2. Open the file in a text editor such as Notepad.
3. Find and delete the following lines:

```
1.2.6.1.4.1.311.1.1.3.1.1:Computer:WindowsNT # Windows NT  
Workstation
```

```
1.2.6.1.4.1.311.1.1.3.1.2:Computer:WindowsNT # Windows NT  
Server
```

```
1.2.6.1.4.1.311.1.1.3.1.3:Computer:WindowsNT # Windows NT  
Domain Controller"
```

4. Save the file.

To Remove OID_TO_TYPE entries

1. In OpenView\conf, find the file oid_to_type.
2. Open the file in a text editor such as Notepad.
3. Find the following line:

```
1.3.6.1.4.1.311.1.1.3.1.*::Windows NT:
```

4. Replace it with this line:

```
1.3.6.1.4.1.311.1.1.3.1::Windows NT:
```

5. Save the file.

Installing on HP-UX

The HPUNIX directory or your ServerWORKS CD contains two files that install the integration components of ServerWORKS: `install.sh` and `DECHPOV.tar`.

Before you run the installation script, review the software prerequisites described previously in this chapter.

The install program uses temporary disk space on the HP UX system. When the installation is complete, this space is recovered.

To install the ServerWORKS components on HP-UX, follow these steps:

1. Log on to the console as root. OpenView services startup automatically.
2. If the OpenView Network Node Manager interface is running, shut it down.
3. On the CD access the HPUNIX directory. Copy the `DECHPOV.tar` file from the CD to `/tmp`. Enter the following commands:

```
cd /tmp
cp DECHPOV.tar
```

4. Untar the `DECHPOV.tar` file. Enter the following command:

```
tar -xvf DECHPOV.tar
```

5. Go to the OpenView directory. Enter the following command:

```
cd /tmp/OpenView
```

6. Run the install shell script `./install.sh`. As the installation program runs, you are prompted to verify configuration and setup information.

As the installation script runs, it displays messages telling you what it is installing. The installation script creates two log files: `SWInstall.log` and `SWInstallErr.log`. When the script finishes, you are prompted to view the log files.



Note: The log files are deleted after the installation script finishes running. If the installation script encounters an error while it is running, it halts and prompts you to view the log files.

Confirming the Installation

To confirm the installation, follow these steps:

1. Start OpenView on your management console.
2. Verify that the DIGITAL menu is present on the menu bar.

Figure 2-3 DIGITAL Menu Added to the OpenView Menu Bar on HP-UX



3. Select Locate → Object → By Attribute.
4. Verify that these attributes appear in the list of Object attributes:
 - DECsvrClusterType
 - DECsvrQuery Type
 - DECsvrSysType

To find DIGITAL servers by attribute, see “Discovering DIGITAL Servers,” a subsequent section of this chapter.

Uninstalling ServerWORKS from HP-UX

To uninstall ServerWORKS from HP-UX system, follow these steps:

1. Log on to the console using an account with administrator privileges.
2. If OpenView is running, close it.
3. Insert the ServerWORKS Manager CD into your CD drive.
4. Locate and run `uninstal.bat` in the OpenView subdirectory on the ServerWORKS Manager CD.

The uninstall batch file attempts to restore your system to the state it was before you installed ServerWORKS. It does not, however, remove DIGITAL OIDs from the data collector configuration file.

5. To remove the DIGITAL OIDs, select Options → Data Collection & Thresholds: SNMP. And highlight the following OIDs. Select Suspend to end their use.
 - HrProcessor Load
 - HrStorageUsed
 - SvrPowerSensorReading
 - SvrThSensorReading

Learning What You Can Manage

Integrating DIGITAL ServerWORKS Manager with HP OpenView has these results:



- Now when you work with an OpenView map or submap, you see a DIGITAL menu in the menu bar. With the new menu, you can accomplish these tasks:
 - Discover DIGITAL servers
 - Configure alarms and traps (NT only)
 - Run the ServerWORKS System Browser (NT only)
 - Display server information using DIGITAL macros: Environment, Memory, Network, Storage, System, and Graphs



- Now when you work with the OpenView Event Categories window, you see a button for DIGITAL Server Events. The color of the button reflects the most severe unacknowledged event. When you click on this button, you display events received from a DIGITAL managed system.

For information about the colors of the buttons in the Event Categories window, refer to the HP OpenView *Using Network Node Manager Guide*.

Discovering and Managing DIGITAL Objects

When you start OpenView Network Node Manager, DIGITAL servers and clusters are discovered automatically and are displayed on the network map.

DIGITAL devices are represented on your network map by special icons. You can work with these icons much as you would with other symbols on the map. Table 2-1 lists and describes the DIGITAL icons.

Table 2-1 DIGITAL Icons

Icon	Represents
	A DIGITAL Alpha-based server
	A DIGITAL Server cluster
	A DIGITAL Intel-based server

You can manage DIGITAL devices through the DIGITAL menu. You can also use HP OpenView tools. Examples of what you can do include:

- Discover DIGITAL objects on demand
- Find DIGITAL objects by attribute
- Manage devices using the DIGITAL MIB
- Create DIGITAL-specific submaps

Discovering DIGITAL Objects on Demand

Although DIGITAL objects are discovered automatically when you start OpenView, you may find it useful to discover them on demand, updating your open submap as required. For example, you may want to locate servers that were recently added.

To discover all the DIGITAL servers on your network or subnetwork:

1. Open an existing map or create a new map. Refer to the *HP OpenView Using Network Node Manager Guide* for information about this task.
2. Click DIGITAL → Discover.

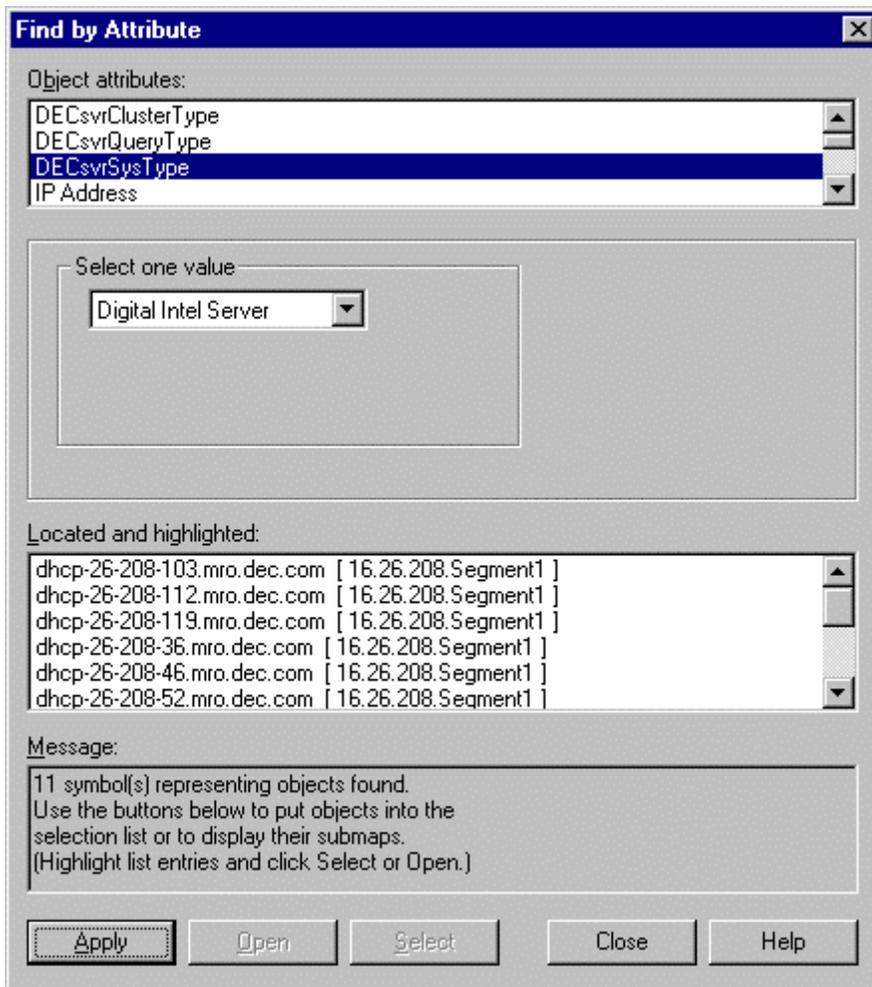
The discovery process uses the conditions set up for OpenView. See the *HP OpenView Using Network Node Manager Guide* for more information.

Finding DIGITAL Servers by Attribute

Searching by attribute offers a quick way to locate a server by a specific attribute such as processor type or network address. To quickly find DIGITAL servers on an OpenView submap, follow these steps:

1. Open an existing map or create a new map. Refer to the HP OpenView manuals for information about this task.
2. On NT, click Edit → Find → Object by Attribute.
On HP-UX, click Locate → Objects → By Attribute.
3. From the Object Attributes list, select DECsvrSysType. (See Figure 2-4).
4. Choose a value from the Select One Value list. For example, choose DIGITAL Intel Server.
5. Click Apply to discover the servers that match the attribute and value.
6. From the Located and Highlighted list, select those servers with submaps that you want to display and click Open.

Figure 2-4 Find by Attribute Window



Managing Devices Using the DIGITAL MIB

SNMP MIBs define data structures that organize and provide access to state information or control points on network devices. The DIGITAL MIBs are designed to integrate the status of DIGITAL devices, or to activate their control logic. The DIGITAL MIBs are accessed through HP OpenView Network Node Manager. From here you can query and set MIB variables on DIGITAL devices.

To use the MIB to manage a DIGITAL device:

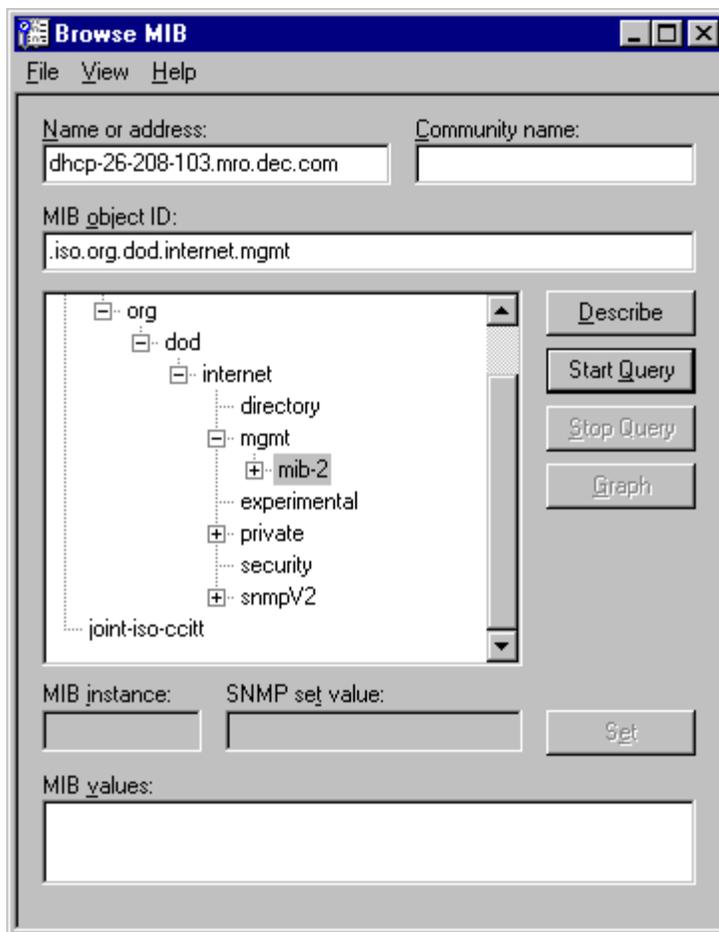
1. Select a DIGITAL device on an OpenView map.
2. On NT, click Tools → SNMP MIB Browser.

On HP-UX, click Misc → SNMP MIB Browser.

Figure 2-5 shows a region of the Browse MIB window on NT.

Integrating with HP OpenView

Figure 2-5 Browsing the DIGITAL MIB in HP OpenView



Integrating with HP OpenView

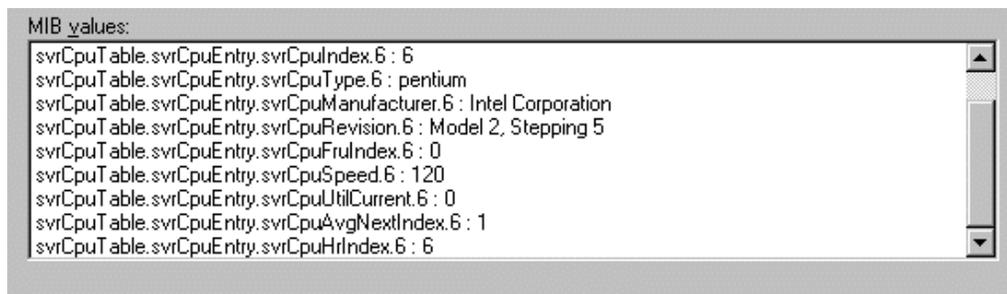
3. On the Browse MIB window, navigate the MIB tree.

Use this path:

```
\iso\org\dod\internet\private\enterprises\dec\ema\  
\mib-extensions-1\svrSystem
```

4. From svrSystem, choose one of these MIBs to browse:
 - svrBaseSystem (base system)
 - svrmgt (server management)
 - ntcmtg (cluster management)
 - svrclu.mib (cluster management)
 - svrntc.mib (cluster management)
5. Follow standard HP OpenView procedures to query and set MIB variables. Figure 2-6 shows the results of a querying the svrCpuTable.

Figure 2-6 Results of Querying svrCpuTable



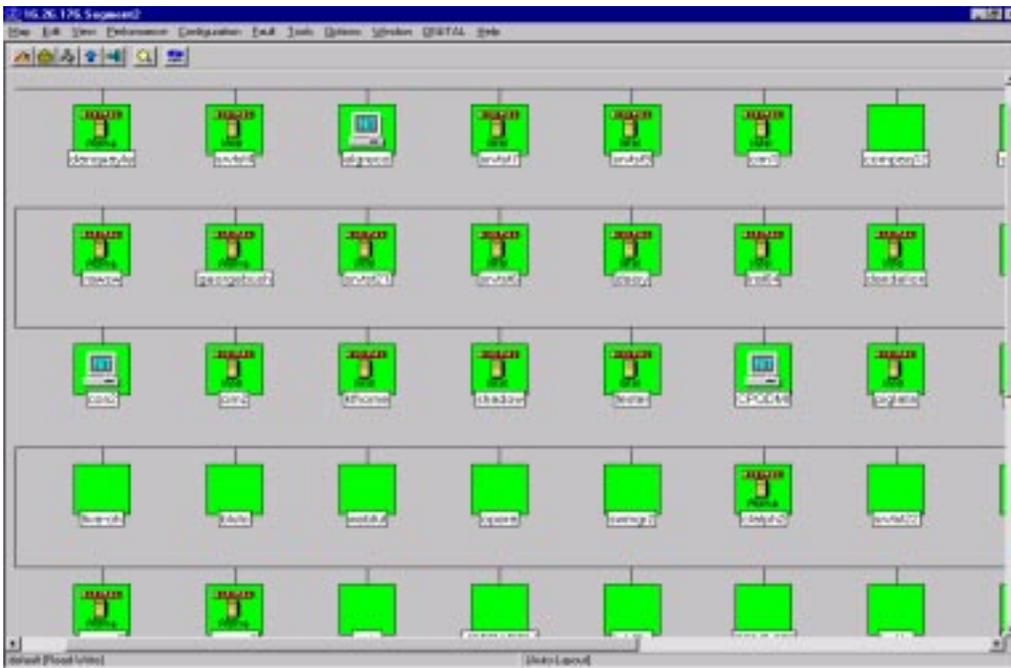
Creating DIGITAL-Specific Submaps

You may find it useful to group DIGITAL devices in their own submap. To create a DIGITAL-specific submap, follow these steps:

1. Select the DIGITAL servers you want to group.

One way to select the servers is to follow the procedure in the section “Finding DIGITAL Servers by Attribute.” Highlight the servers you want to group, then click the Select button. Refer to Figure 2-4.
2. Click Edit → Copy.
3. Click Map → Submap → New.
4. Respond to the prompts presented by the New Submap Wizard.
5. Click Edit → Paste to display the icons on the new submap.

Figure 2-7 DIGITAL-Specific Submap Window



Displaying Server Information

Using the DIGITAL macros, you can display the following server information:

The  DIGITAL object detail icon also accesses the information icon.

Environment



Storage



Network

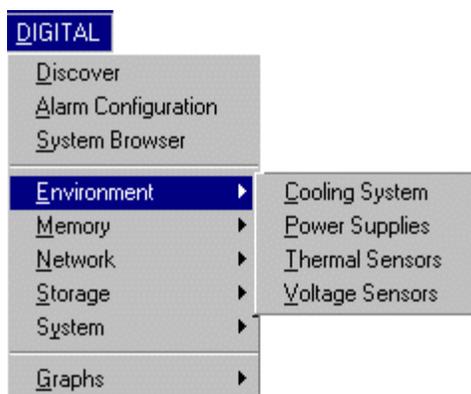


System



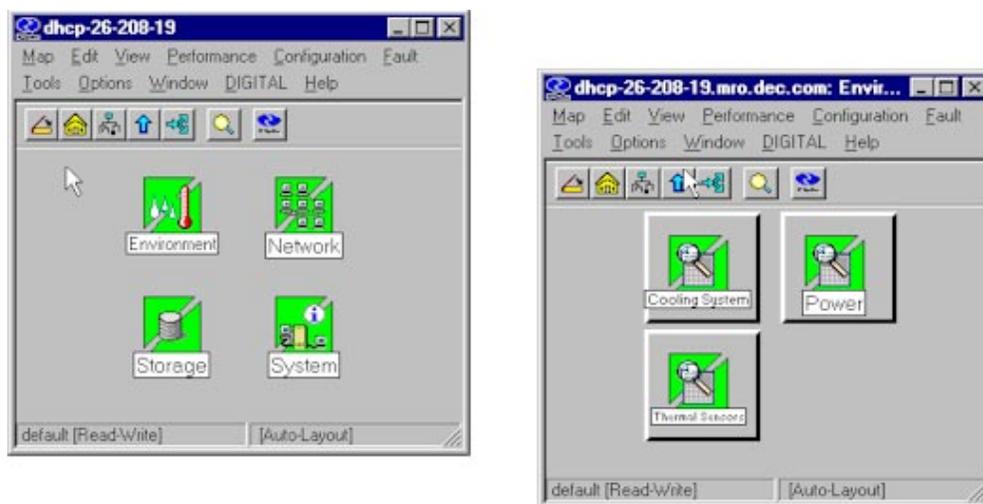
This information is available from the DIGITAL menu, as shown in Figure 2-8.

Figure 2-8 Using the DIGITAL Menu to Display Device Information



You can also obtain DIGITAL device information by double-clicking a DIGITAL device icon, then double-clicking the icon for the variable you want to view, as illustrated in Figure 2-9.

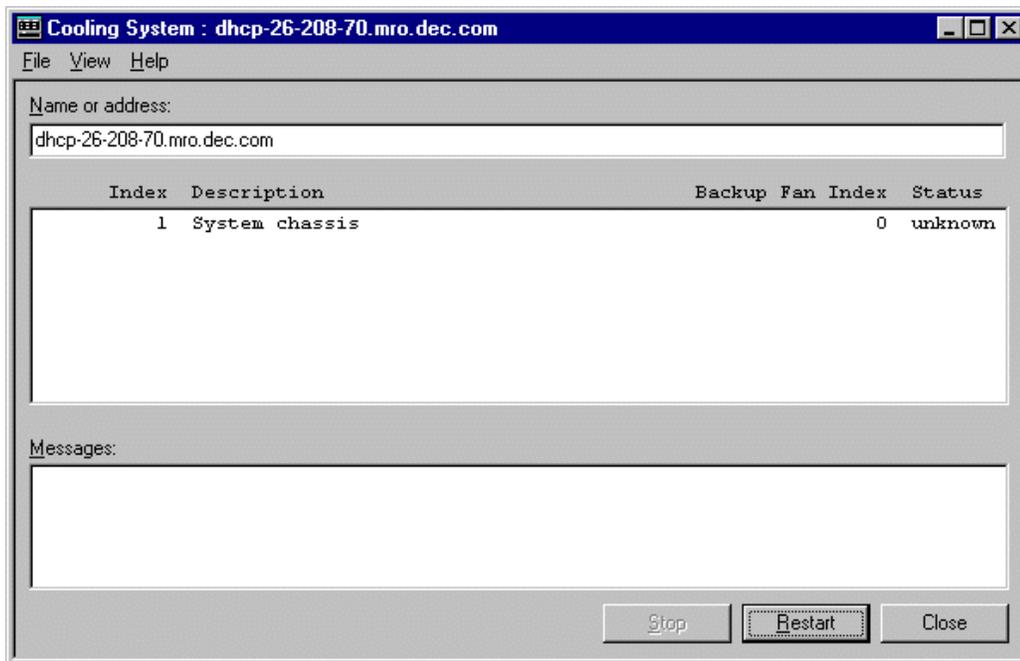
Figure 2-9 Using DIGITAL Icons to Display Device Information



Whether you use the DIGITAL menu or the DIGITAL icons, the information is displayed in an HP OpenView style.

Integrating with HP OpenView

Figure 2-10 DIGITAL Device Information Viewed in HP OpenView



DIGITAL Variables

To display each of the variables:

1. Select a DIGITAL icon.
2. Click DIGITAL and then choose a variable.

The following tables describe each of the parameters available in the variables.

Table 2-2 Environmental Variables

From DIGITAL → Environment, Select...	To Display...
Cooling System	The status of cooling systems.
Power Supplies	The status of power supplies installed on the server.
Thermal Sensors	Values obtained from the server's thermal sensors.
Voltage Sensors	Information about the voltage sensors on a server; for example, the nominal voltage rating and the sensors' present reading of voltage or current.

Table 2-3 Memory Variables

From DIGITAL → Memory, Select...	To Display...
Components	Information about the server's memory components; for example, the type and amount of memory.
Overview	General information about the memory on the server; for example, the amount of total physical memory and the amount of available physical memory.
Slots	The type and number of slots in use and the number of slots available.

Table 2-4 Network Variables

From DIGITAL → Network, Select ...	To Display...
Interface	Information about the network interface components on the server; for example, the interface bandwidth (speed) and the maximum packet size (maximum transmission unit or MTU) that the interface can send or receive.
Statistics	Transmission and reception statistics.

You can also display network information using the DIGITAL → Graphs option. For more information, refer to DIGITAL → Graphs.

Table 2-5 Storage Variables

From DIGITAL → Storage, Select...	To Display...
File System	Information about the file systems in use.
Devices	Information about the physical storage devices on a server; for example the type of device and its capacity.
Disk Partitions	Information about how disks are partitioned on a server.
Storage	Information about logical storage devices on a server; for example, virtual memory.

Table 2-6 System Variables

From DIGITAL → System, Select...	To Display ...
Cluster Groups	If a server belongs to a cluster, the name of the cluster group and other devices that belong to the cluster.
Cluster Members	The server's cluster name.
Devices	The status of devices installed on the server: Unknown; Running; Warning; Testing; Down
Firmware	The version of firmware on a server.
Field Replaceable Units (FRU)	Information about field replaceable units installed on the server, including the device type, its history, part number and serial number.
Miscellaneous	Miscellaneous system information, including the reason for the last system shutdown, if known.
Processors	Information about CPUs installed on the server and their most recent usage rates.

Table 2-7 Graphing Options from the DIGITAL menu

From DIGITAL → Graphs, Select...	To Graph...
Voltage	Voltage sensor readings on the server.
Temperature	Temperature readings on the server.
File System Usage	Percent of storage on the device.
Processor Load	Percent of processor utilized.
Octets	Total octets transmitted and received on each network interface on the server.
Non-Unicast Packets	Total non-unicast packets transmitted and received on each network interface.
Packets Discarded	Total number of packets discarded on all network interfaces on the server.
Packet Errors	The number of packets with error that each network interface has received.
Unknown Protocol Packets	The number of unknown protocol packets that each network interface has received.
Total Packets	The total unicast and non-unicast packets that all network interfaces on a server have received.
Non-Unicast Packets	The number of non-unicast packets that each interface on a server has transmitted.
Transmit Queue Length	The length of the transmit queue on each interface on a server.

Configuring Traps and Trap Destinations

Using SNMP traps, you can ensure that DIGITAL devices notify HP OpenView Network Node Manager when particular events occur.

The DIGITAL svrmgt.mib defines four levels of traps: high, medium, low, and informational. Each trap can indicate a variety of conditions depending on how a particular server agent is configured. Each of these traps has a unique trap ID.

HP OpenView defines five event categories: Error Events, Threshold Events, Status Events, Configuration Events, and Application Alert Events. Within each category, events are assigned one of five levels of severity. Table 2-8 compares DIGITAL trap and HP event levels.

Table 2-8 Trap and Event Levels in OpenView and ServerWORKS

OpenView	ServerWORKS
Critical	High
Major	Medium
Minor	Low
Warning	(no corresponding trap level)
Normal	Informational

To view DIGITAL traps in HP OpenView, you designate a ServerWORKS Manager Console as the trap destination; then configure the ServerWORKS Console to forward the trap to OpenView.

Designating the Trap Destination

To designate a ServerWORKS console as a trap destination:

1. Make sure that SNMP is configured as a service on each DIGITAL device where a ServerWORKS agent is installed.
2. Choose Tools → Trap Control from the ServerWORKS Manager Console or from Alarm Configuration.
3. Click the Remote Destination tab.
4. Enter the Host name (managed device name).
5. Choose one of the following:
 - Accept the default community (for example, Public for Windows NT).
 - Deselect the default option and enter Get Community and Set Community names.
6. Click Get Info to display the current community name and trap destination information for the host.
7. Specify the destination by community name. The destination is included in the trap message.

Forwarding Traps

To configure the ServerWORKS Manager Console to forward the trap to an HP OpenView console:

1. From the ServerWORKS Console menu, choose Tools→Trap Control.
2. Click the Local Forwarding tab and perform the following tasks:
 - a. Enter the community name, if necessary. Public is the NT default community. You can change the community name, but the name you use applies to all forwarding destinations in the list.
 - b. Select Forward alarms as traps.
 - c. Click Get Info to see a list of the forwarding addresses for the selected community.
3. To add a forwarding address, click Add. On the Add dialog box, enter the Address and Port number. Then click OK.
4. Click Apply to enable the changes and Close to close the dialog box.



Note: For complete instructions about configuring SNMP on a managed device and trap forwarding, refer to the *DIGITAL ServerWORKS Manager Installation and User Guide*.

Browsing DIGITAL Server Events

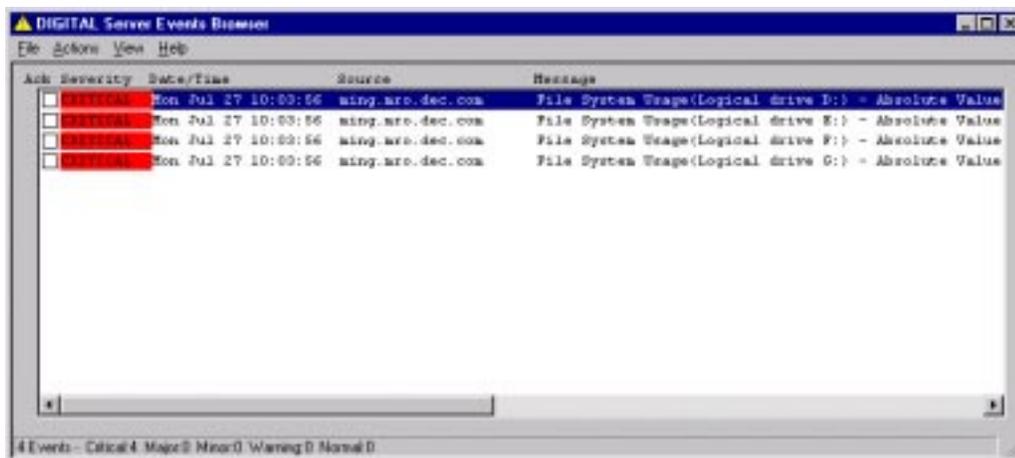
Whenever a ServerWORKS agent generates a trap, you can view information about it from the OpenView Event Categories window.

Viewing DIGITAL Server Events

To view DIGITAL agent traps:

1. Make Event Categories the active window.
2. Click DIGITAL Server Events. The DIGITAL Server Events Browser Window is displayed, as shown in Figure 2-11.

Figure 2-11 DIGITAL Server Events Browser Window



The DIGITAL Server Events Browser window contains a scrolled list of events in chronological order with the most recent event last.

Integrating with HP OpenView

For each event, you see the following information:

- Ack – A checkbox indicating whether the event has been acknowledged. If the box is empty, the event is unacknowledged.
- Severity – The trap level for the event.
- Date/Time – The day, date, and time the event occurred.
- Source – An identifier, such as the server name, for the device where the event occurred.
- Message – A short statement about the event.

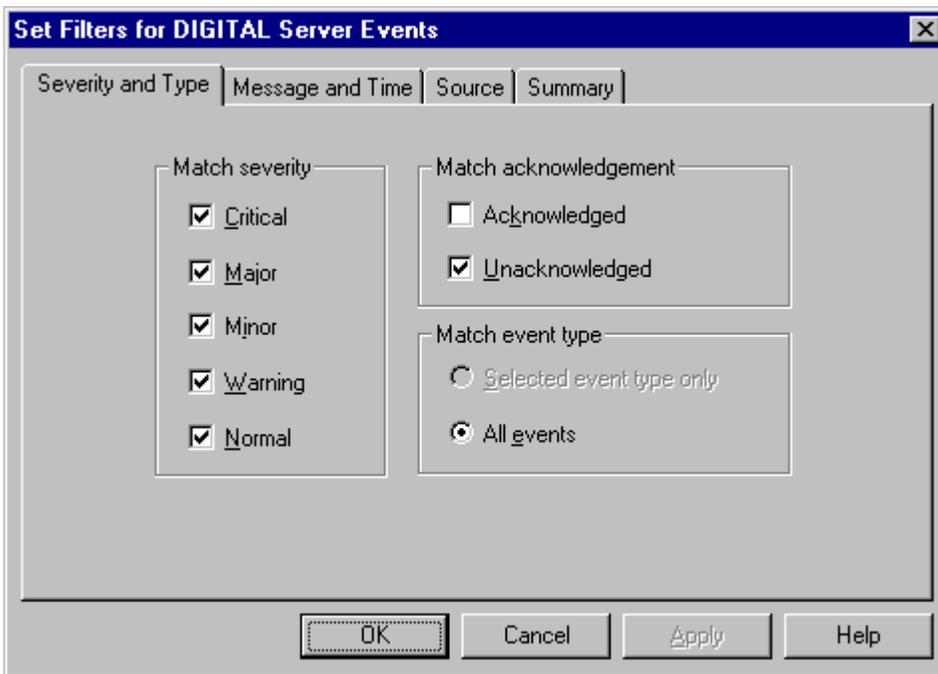
Filtering DIGITAL Server Events

You can filter traps received from DIGITAL devices so that you can focus on just those that are of immediate concern.

To filter events:

1. From the DIGITAL Server Events Browser window, select View → Set Filters.
2. Set the filter to show events that match any combination of these criteria:
 - Severity
 - Acknowledgement
 - Event type
 - Message string
 - After time/date
 - Before time/date
 - Source
3. Click OK to apply the filter.

Figure 2-12 Set Filters for DIGITAL Server Events



For example, to see only unacknowledged events that occurred after a given time and date:

1. In the Match Acknowledgement region of the window, deselect Acknowledged.
2. Click the Message and Time tab
3. In the Match by Time region of the window, check Received on or After.
4. Select the time and date.

Integrating with Tivoli TME 10 NetView **3**

You can integrate DIGITAL ServerWORKS Manager into an environment that uses Tivoli TME 10 NetView as its enterprise management framework. This integration extends your ability to monitor and manage DIGITAL servers and clusters through the NetView Console.

To integrate ServerWORKS Manager with NetView, refer to the topics and tasks described in this chapter:

- Installing ServerWORKS Manager components
- Learning what you can manage
- Discovering and managing DIGITAL objects
- Displaying server information
- Configuring traps and trap destinations

This chapter describes integration with NetView running on X86-based Windows NT and DIGITAL UNIX consoles.

Installing ServerWORKS Manager Components

To be sure you are ready to install the ServerWORKS Manager components, you should understand the goals and prerequisites for the installation.



Note: The installation instructions presented in this chapter assume that SNMP is installed on both the network management console and on the DIGITAL devices that you want to manage. For information about installing SNMP, refer to the documentation for your Windows NT or DIGITAL UNIX operating system.

Goals of the Installation

Run the automated installation script to integrate ServerWORKS into Tivoli TME 10 NetView. The goals of the installation procedure are to accomplish the following tasks:

- Load DIGITAL MIB files
 - Server System MIB (svrsys.mib) which provides access to basic server configuration information
 - Server Management MIB (svrmgt.mib) which provides access to the instrumentation in the DIGITAL agent, including the ability to monitor and detect thresholds
 - Cluster Management MIB (ntcmgt.mib) which provides the ability to manage DIGITAL clusters
 - Server Cluster MIB (svrclu.mib) which provides information about DIGITAL cluster products and Microsoft NT clusters
 - Server NT Cluster MIB (svrntc.mib) which provides information about DIGITAL cluster products and Microsoft NT clusters



Note: The three cluster MIBs differ in how they are implemented. Use the MIB Browser tool to determine which MIB or MIBs is relevant to your enterprise.

Integrating with Tivoli TME 10 NetView

- Install DIGITAL icons
- Modify the NetView menu bar by adding a DIGITAL menu
- Install trap files
- Add MIB applications
- Add DIGITAL Server OIDs to the data collector
- Install the ServerWORKS Manager System Browser (on NT only)
- Install the ServerWORKS Alarm Configuration utility (on NT only)

Prerequisites for the Installation

Before you integrate ServerWORKS 4.0 with Tivoli TME 10 NetView, make sure you comply with the software prerequisites listed in the following sections.



Note: In addition to the software prerequisites listed in this section, you should be aware of the minimum hardware requirements to run Tivoli TME 10 NetView. Refer to your Tivoli TME 10 NetView documentation for this information.

You may also want to familiarize yourself with the hardware and software requirements for a complete ServerWORKS installation. Refer to the *DIGITAL ServerWORKS Manager Installation and User Guide*.

On an X86-based NT Console

Integration of ServerWORKS on an NT console requires:

- Windows NT 4.0 (on an X86 processor)
- Tivoli TME 10 NetView 5.0
- 30 MB free disk space
- TCP/IP network protocol
- SNMP services

Integrating with Tivoli TME 10 NetView

To verify that TCP/IP and SNMP are installed:

1. Access the Tivoli application.
2. Select the SNMP nodes from the subnet map.
3. Choose Test → Connectivity from the main menu to check for SNMP and TCP/IP connectivity.

Make sure that DIGITAL agents are installed on the devices you want to manage through Tivoli TME 10 NetView. For information about installing DIGITAL agents, refer to the *DIGITAL ServerWORKS Manager Installation and User Guide*.

On a DIGITAL UNIX Console

Integration of ServerWORKS on a DIGITAL UNIX console requires:

- DIGITAL UNIX 4.2 or greater
- Tivoli TME 10 NetView 5.0
- TCP/IP network protocol
- SNMP
- 5 MB of disk space

To verify that networking and SNMP are installed, refer to the DIGITAL Unix documentation and:

1. Access the application.
2. Verify that Network is available.
3. Verify SNMP on a node known to have SNMP installed.

Make sure that DIGITAL agents are installed on the devices you want to manage through OpenView. For information about installing DIGITAL agents, refer to the *DIGITAL ServerWORKS Manager Installation and User Guide*.

Installing on NT

To install the ServerWORKS components on an X86-based NT console, follow these steps:

1. Log on to the console using an account with administrator privileges.
2. Close all programs that are running, including Tivoli TME 10 NetView, mail programs, and the Microsoft Office shortcut bar.
3. Insert the ServerWORKS Manager CD in the CD drive of the console system.

As soon as the CD is engaged, the main installation screen appears. This screen prompts you to select the language you want to use to install ServerWORKS components.

If it does not appear, follow these steps:

- a. From the desktop, click the Start menu.
- b. Choose Run. Enter the following path:

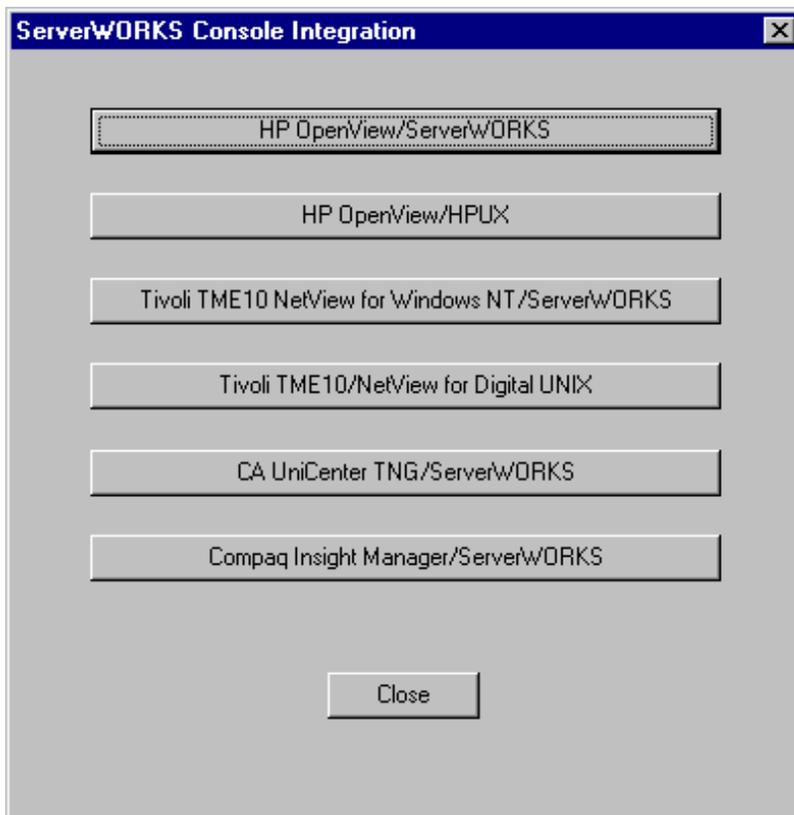
```
<cd-rom drive>:\Autoplay.exe
```
4. On the language selection screen, choose English. ServerWORKS integration is not internationalized.

The ServerWORKS Component Installation screen appears.

Integrating with Tivoli TME 10 NetView

5. Click the ServerWORKS Console Integration button.
The ServerWORKS Console Integration screen appears.
6. Click the Tivoli TME 10 NetView/ServerWORKS button.
7. Follow the remaining prompts to complete the installation of the ServerWORKS integration components.

Figure 3-1 ServerWORKS Console Integration Screen



Integrating with Tivoli TME 10 NetView

As the installation script runs, it displays messages telling you what is being installed. The installation script creates two log files: SWInstall.log and SWInstallErr.log. When the script finishes, it displays a message telling you whether the installation succeeded. You are prompted to view the log files.



Note: The log files are deleted after you complete the installation. If the installation script encounters an error while it is running, it halts and prompts you to view the log files.

Confirming the Installation

To confirm the installation follow these steps:

1. Start Tivoli TME 10 NetView on your management console.
2. Verify that the DIGITAL menu is present on the menu bar.

Figure 3-2 DIGITAL Menu Added to the NetView Menu Bar on NT



3. Select Edit → Find → Object By Attribute.
4. Verify that these attributes appear in the list of Object attributes:
 - DECsvrClusterType
 - DECsvrQuery Type
 - DECsvrSysType

To find DIGITAL servers by attribute, see “Discovering DIGITAL Servers,” a subsequent section of this chapter.

Uninstalling ServerWORKS Manager from an NT System

ServerWORKS Manager is tightly integrated with Tivoli TME 10 NetView. As a result, if you need to uninstall ServerWORKS Manager integration components, DIGITAL strongly recommends that you uninstall Tivoli TME 10 NetView. Uninstalling Tivoli TME 10 NetView removes ServerWORKS Manager integration components. You can then reinstall NetView.

If you choose not to follow the DIGITAL recommendation, an uninstall of ServerWORKS Manager components requires you to:

- Run an uninstall batch file
- Remove the ServerWORKS Manager database from the Registry
- Remove OID_TO_SYM and OID_TO_TYPE mapping entries

To Run the Uninstall Batch File

In the NetView subdirectory on the ServerWORKS Manager CD, locate and run uninstal.bat.

The uninstall batch file attempts to restore your system to the state it was before you installed ServerWORKS Manager. However, you must manually remove the database and the mapping entries.

To Remove the ServerWORKS Manager Database from the Registry

1. In the Control Panel, double-click the ODBC icon.
2. Click the System DSN tab.
3. Select the SWMgrDB entry.
4. Click Remove. You are prompted to make sure you want to delete SWMgrDB.
5. Click Yes.

Integrating with Tivoli TME 10 NetView

To Remove OID_TO_SYM Entries

1. In NetView\config, find the file oid_to_sym.
2. Open the file in a text editor such as Notepad.
3. Find and delete the following lines:

```
1.2.6.1.4.1.311.1.1.3.1.1:Computer:WindowsNT # Windows NT  
Workstation
```

```
1.2.6.1.4.1.311.1.1.3.1.2:Computer:WindowsNT # Windows NT  
Server
```

```
1.2.6.1.4.1.311.1.1.3.1.3:Computer:WindowsNT # Windows NT  
Domain Controller
```

4. Save the file.

To Remove OID_TO_TYPE entries

1. In NetView\config, find the file oid_to_type.
2. Open the file in a text editor such as Notepad.
3. Find the following line:

```
1.3.6.1.4.1.311.1.1.3.1.*::Windows NT:
```

4. Replace it with this line:

```
1.3.6.1.4.1.311.1.1.3.1::Windows NT:
```

5. Save the file.

Installing on DIGITAL UNIX

The DUNIX directory on your ServerWORKS Manager CD contains two files that you need to install the integration components of ServerWORKS: `install.sh` and `DECHPOV.tar`.

Before you run the installation script, review the software prerequisites described previously in this chapter.

The installation program uses temporary disk space on the DIGITAL UNIX system. When the installation is complete, this space is recovered.

To install the ServerWORKS components on DIGITAL UNIX, follow these steps:

1. Log on to the console as *root*.
2. Start NetView services.
3. Verify that there is at least 5 MB of free disk space in the file system where TME 10 NetView is installed.
4. Mount the CD on to the file system.
5. Stop the TME 10 NetView GUI.
6. Close all `ovw` sessions by running the command

```
OV_BIN/ovstop
```

7. Copy the `DECPNV.tar` file from the CD to `/tmp`. Enter the following command:

```
cp DECPNV.tar/tmp
```

8. Go to the `/tmp` directory. Enter the following command:

```
cd /tmp
```

9. Untar the `DECPNV.tar` file. Enter the following command:

```
tar -xvf DECPNV.tar
```

10. Go to the NetView directory. Enter the following command:

```
cd /tmp/NetView
```

11. Run the install shell script `./install.sh`.

Integrating with Tivoli TME 10 NetView

As the installation program runs, you are prompted to verify configuration and setup information.

As the installation script runs, it displays messages telling you what is being installed. The installation script creates two log files: SWInstall.log and SWInstallErr.log. When the script finishes, you are prompted to view the log files.



Note: The log files are deleted after the installation script finishes running. If the installation script encounters an error while it is running, it halts and prompts you to view the log files.

Confirming the Installation

To confirm the installation, follow these steps:

1. Start NetView on your management console.
2. Verify that the DIGITAL menu is present on the menu bar.

Figure 3-3 DIGITAL Menu Added to the NetView Menu Bar on DIGITAL UNIX



3. Select Locate → Object → By Attribute.
4. Verify that these attributes appear in the list of Object attributes:
 - DECsvrClusterType
 - DECsvrQuery Type
 - DECsvrSysType

To find DIGITAL servers by attribute, see “Discovering DIGITAL Servers,” a subsequent section of this chapter.

Uninstalling ServerWORKS from DIGITAL UNIX

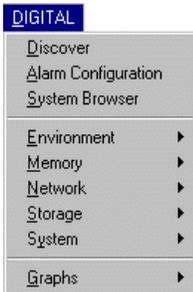
To uninstall ServerWORKS from a DIGITAL UNIX system, follow these steps:

1. Log on to the console in the root account.
2. If NetView is running, close it.
3. Insert the DIGITAL CD into your CD drive.
4. Locate and run `uninstal.bat` in the NetView subdirectory on the ServerWORKS Manager CD.

The uninstall batch file attempts to restore your system to the state it was in before you installed ServerWORKS. It does not, however, remove DIGITAL OIDs from the data collector configuration file.

5. To remove the DIGITAL OIDs, select Options → Data Collection & Thresholds: SNMP.
6. Remove the following OIDs:
 - HrProcessor Load
 - HrStorageUsed
 - SvrPowerSensorReading
 - SvrThSensorReading

Learning What You Can Manage



When you integrate DIGITAL ServerWORKS Manager with Tivoli TME 10 NetView, you gain access to the following DIGITAL tools through the new DIGITAL menu:

- Discovery
- ServerWORKS System Browser (NT X86-based consoles only)
- ServerWORKS Alarm Configuration tool (NT X86-based consoles only)
- DIGITAL macros: Environment, Memory, Network, Storage, System, and Graphs

To quickly access these tools:

1. On a NetView map, select a DIGITAL device.
2. Click DIGITAL from the Main Menu bar and select the tool you want to use. To display the Alarm Configuration tool for example, click DIGITAL → Alarm Configuration.

Discovering and Managing DIGITAL Objects

When you start NetView, objects such as DIGITAL workstations, servers and clusters (objects) are discovered automatically and are displayed on the network map.

DIGITAL devices are represented on your network map by special icons. You can work with these icons much as you would with other symbols on the map. Table 3-1 lists and describes the DIGITAL icons.

Table 3-1 DIGITAL Object Icons

Icon	Represents
	DIGITAL Alpha-based Server running Windows NT or DIGITAL Unix
 Cluster	DIGITAL Server Cluster
 Intel	DIGITAL Intel-based Server running Windows NT or DIGITAL Unix

You can manage DIGITAL devices through the DIGITAL menu. You can also use NetView tools. Examples of what you can do include:

- Discover DIGITAL objects on demand
- Find DIGITAL objects by attribute
- Manage devices using the DIGITAL MIB
- Create DIGITAL-specific submaps

Discovering DIGITAL Objects on Demand

Although DIGITAL objects are discovered automatically when you start NetView, you may find it useful to discover them on demand, updating your open submap as needed.

To discover all DIGITAL servers on your network or subnetwork, follow these steps:

1. Start the NetView Console.

On NT, from the Start menu, click Programs → NetView → NetView Console.
2. On DIGITAL UNIX, at the prompt type

pnv
3. Open an existing map or create a new map. Refer to Tivoli TME 10 NetView documentation for information about this task.
4. Navigate to the submap that is of interest to you.
5. Click DIGITAL → Discover.

Finding DIGITAL Servers On NT

To find DIGITAL servers on NT, follow these steps:

1. Open an existing map or create a new map. Refer to the NetView documentation for information about this process.
2. Click Edit → Find. The Find window appears.
3. Select Find by = Object Type.
4. In the Type region of the window, click the Computer icon.
5. In the Subtype region of the window, click the icon for the type of system you want to find. For example, click the DECsvrIntel icon.
6. Click Apply.

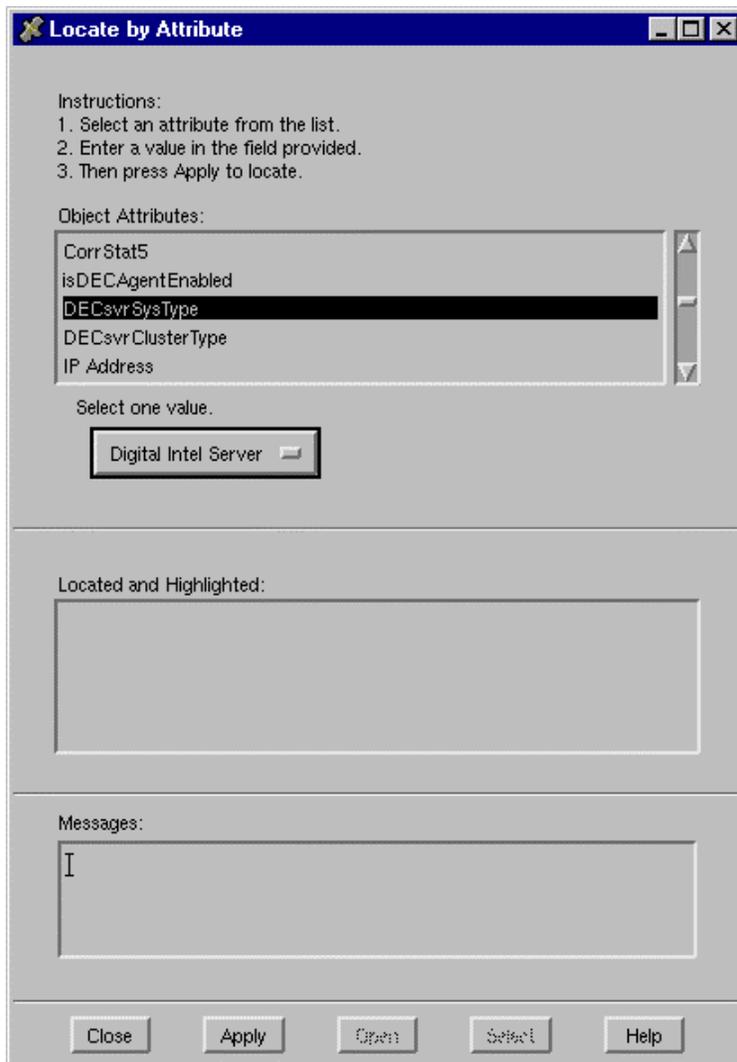
Use this procedure to find DIGITAL managed systems of the type specified on your currently open submap.

Finding DIGITAL Servers On DIGITAL UNIX

To find DIGITAL servers with a specific attribute, follow these steps:

1. Open an existing map or create a new map. Refer to the Tivoli TME NetView manuals for information about this task.
2. Click Locate → Objects → By Attribute.
3. From the Object Attributes list, select DECsvrSysType.
4. Choose a value from the Select One Value list; for example, DIGITAL Intel Server.
5. From the Object Attributes list, select DECsvrSysType.
6. Choose a value from the Select One Value list; for example, DIGITAL Intel Server.
7. Click Apply to discover the servers that match the attribute and value.
8. From the Located and Highlighted list, select those servers with submaps that you want to display and click Open.

Figure 3-4 NetView Locate by Attribute Window on DIGITAL UNIX



Managing Devices Using the DIGITAL MIB

You can access the DIGITAL MIB through the Tivoli NetView Console. You can then query and set MIB variables on DIGITAL devices.

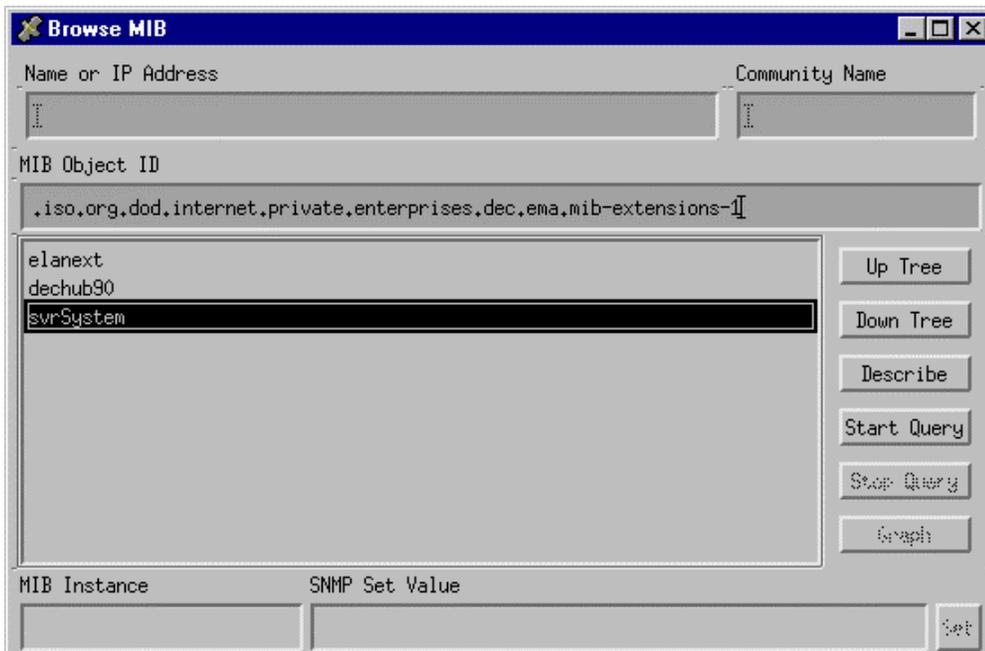
To use the MIB to manage a DIGITAL device:

1. Select a DIGITAL device on a NetView map.
2. On NT, click Tools → MIB → Browser.

On DIGITAL UNIX, click Tools → MIB Browser → SNMP.

Figure 3-5 shows a region of the Browse MIB window on DIGITAL UNIX.

Figure 3-5 Browsing the DIGITAL MIB in NetView



Integrating with Tivoli TME 10 NetView

3. On the MIB Browser (Browse MIB on DIGITAL UNIX) window, navigate the MIB tree.

Use this path:

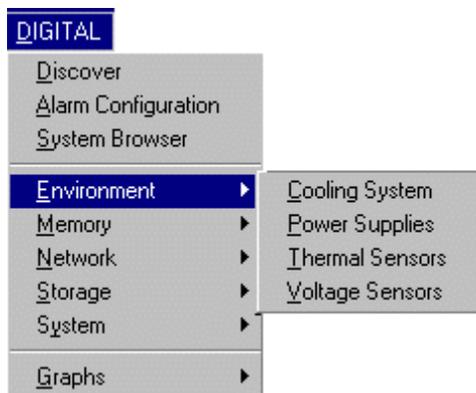
```
\iso\org\dod\internet\private\enterprises\dec\ema\  
mib-extensions-1\svrSystem
```

4. From svrSystem, choose one of these MIBs to browse:
 - svrBaseSystem (base system)
 - svrmgt (server management)
 - ntcmtg (cluster management)
 - svrclu.mib (cluster management)
 - svrntc.mib (cluster management)
5. Follow standard Tivoli TME 10 NetView procedures to query and set MIB variables.

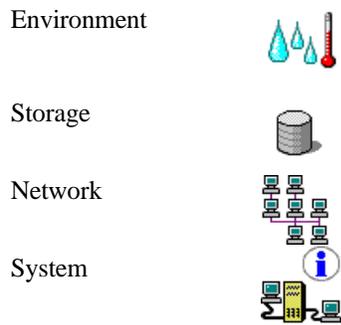
Displaying Server Information

Using the DIGITAL macros, you can display the following server information. The information is available using the DIGITAL pull down menu, as shown in Figure 3-6.

Figure 3-6 Using the DIGITAL Menu to Display Device Information



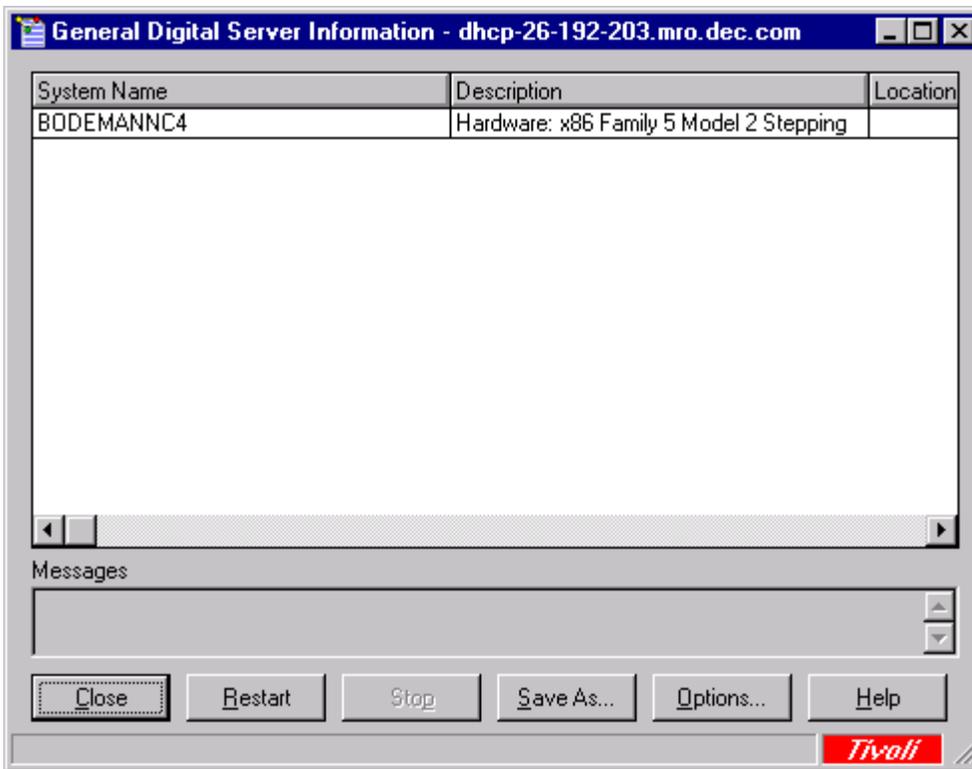
The  DIGITAL object detail icon also accesses the same information when you double-click on it.



Integrating with Tivoli TME 10 NetView

DIGITAL device information is illustrated in Figure 3-7.

Figure 3-7 DIGITAL Device Information Viewed in NetView



DIGITAL Variables

To display each of the variables:

1. Select a DIGITAL icon.
2. Click DIGITAL and then choose a variable.

The following tables describe each of the parameters available in the variables.

Table 3-2 Environmental Variables

From DIGITAL → Environment, Select...	To Display...
Cooling System	The status of the server's cooling systems.
Thermal Sensors	Values obtained from the server's thermal sensors.
Power Supplies	The status of power supplies installed on the server.
Voltage Sensors	Information about the voltage sensors on a server; for example, the nominal voltage rating and the sensors present reading of voltage or current.

Table 3-3 Memory Variables

From DIGITAL → Memory, Select...	To Display...
Components	Information about the server's memory components; for example, the type and amount of memory.
Overview	General information about the memory on the server; for example, the amount of total physical memory and the amount of available physical memory.
Slots	The type and number of slots in use and the number of slots available.

Table 3-4 Network Variables

From DIGITAL → Network, Select ...	To Display...
Interface	Information about the network interface controllers on the server; for example, the interface bandwidth (speed) and the maximum packet size (maximum transmission unit or MTU) that the interface can send or receive.
Statistics	Transmission and reception statistics.

You can also display network information using the DIGITAL → Graphs option.

Table 3-5 Storage Variables

From DIGITAL → Storage, Select...	To Display...
Physical Devices	Information about the physical storage devices on a server; for example the type of device and its capacity.
File System	Information about the file systems in use.
Partitions	Information about how disks are partitioned on a server.
Logical Devices	Information about logical storage devices on a server; for example, virtual memory.

Table 3-6 System Variables

From DIGITAL → System, Select...	To Display ...
Information	Overview information about the selected server.
Cluster Groups	The name of the cluster group and other devices that belong to the cluster.
Cluster Members	The name of the servers in the cluster.
Devices	The status of devices installed on the server: Unknown; Running; Warning; Testing; Down
Firmware	The version of firmware on a server.
Field Replaceable Units (FRU)	Information about field replaceable units installed on the server, including the device type, its history, part number and serial number.
Miscellaneous	Miscellaneous system information, including the reason for the last system shutdown, if known.
Processors	Information about CPUs installed on the server and their most recent usage rates.

Table 3-7 Graphing Options from the DIGITAL menu

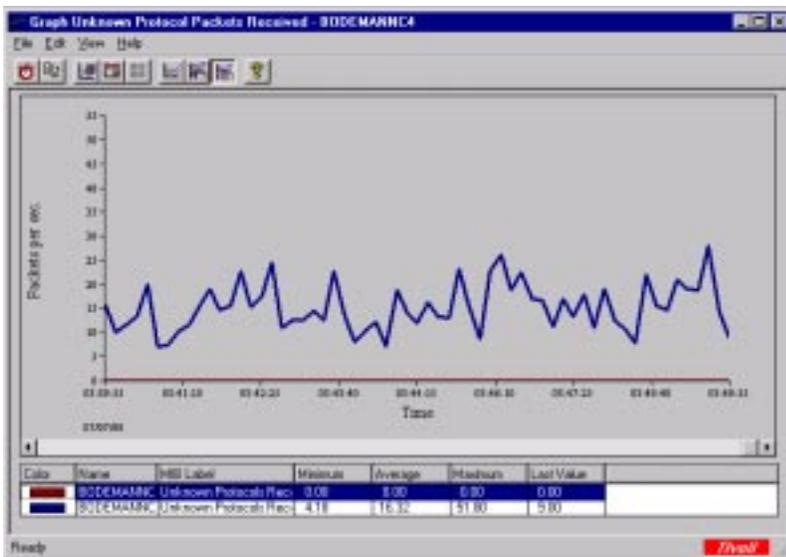
The DIGITAL menu also gives you the option to display graphs of some of this information.

From DIGITAL → Graphs, Select...	To Graph...
Packets Discarded	Total number of packets discarded on all network interfaces on the server.
File System Usage	Percent of storage used on the device.
Octets	Total octets transmitted and received on each network interface on the server.
Non-Unicast Packets	Total non-unicast packets transmitted and received on each network interface.
Packet Errors	The number of packets with errors that each network interface has received.
Processor Load	Percent of processor utilized.
Unknown Protocol Packets	The number of unknown protocol packets that each network interface has received.
Temperature	Temperature readings on the server.
Total Packets	The total unicast and non-unicast packets that all network interfaces on a server have received.
Transmit Queue Length	The length of the transmit queue on each interface on a server.
Unicast Packets	The number of unicast packets that each interface on a server has transmitted.
Voltage	Voltage sensor readings on the server.

Integrating with Tivoli TME 10 NetView

Figure 3-7 shows a sample graph.

Figure 3-7 Sample Graph



Configuring Traps and Trap Destinations

Using SNMP traps, you can ensure that DIGITAL devices notify the Tivoli TME10 NetView when particular events occur.

The DIGITAL svrmgt.mib defines four levels of traps: high, medium, low, and informational. Each trap can indicate a variety of conditions depending on how a particular server agent is configured. Each of these traps has a unique trap ID.

Tivoli defines six event categories: Critical Events, Major Events, Minor Events, Warning Events, and Indeterminate. Within each category, events are assigned a level of severity. Table 3-8 compares DIGITAL trap and NetView event levels.

Table 3-8 Trap and Event Levels in NetView and ServerWORKS

NetView	ServerWORKS
Critical	High
Major	Medium
Minor	Low
Warning	(no corresponding trap level)
Indeterminate	(no corresponding trap level)
Cleared	(no corresponding trap level)

Integrating with Tivoli TME 10 NetView

To send traps to Tivoli TME 10 NetView, follow these steps:

1. Make sure that SNMP is configured as a service on each DIGITAL device where a ServerWORKS agent is installed.
2. Designate a ServerWORKS Manager Console as the trap destination.
3. Configure the ServerWORKS Manager Console to forward the trap to a Tivoli TME 10 NetView console.



Note: For complete instructions on configuring SNMP on a managed device and trap forwarding, refer to the *DIGITAL ServerWORKS Manager Installation and User Guide*.

Integrating with CA Unicenter TNG **4**

You can integrate DIGITAL ServerWORKS Manager into an environment that uses CA Unicenter TNG as its enterprise management solution. Integration facilitates your ability to monitor and manage DIGITAL servers and NT clusters through the Unicenter TNG interface.

To integrate ServerWORKS with CA Unicenter TNG, refer to the topics and tasks described in this chapter:

- Installing ServerWORKS components
- Learning what you can manage
- Managing DIGITAL objects
- Configuring traps and trap destinations

You can integrate ServerWORKS Manager with CA Unicenter TNG on X86-based Windows NT consoles.

Installing ServerWORKS Components

To be sure you are ready to install the ServerWORKS components, you should understand the goals and prerequisites for the installation.



Note: The installation instructions presented in this chapter assume that SNMP is installed on both the network management console and on the DIGITAL devices that you want to manage. For information about installing SNMP, refer to your Windows NT documentation.

Goals of the Installation

You run an automated installation script to integrate ServerWORKS into CA Unicenter TNG. The goals of the installation procedure are to:

- Load DIGITAL MIB files
 - Server System MIB (svrsys.mib) which provides access to basic server configuration information
 - Server Management MIB (svrmgt.mib) which provides access to the instrumentation in the DIGITAL agent, including the ability to monitor and detect thresholds
 - Cluster Management MIB (ntcmgt.mib) which provides the ability to manage DIGITAL clusters
 - Server Cluster MIB (svrclu.mib) which provides information about DIGITAL cluster products and Microsoft NT clusters
 - Server NT Cluster MIB (svrntc.mib) which provides information about DIGITAL cluster products and Microsoft NT clusters



Note: The three cluster MIBs differ in how they are implemented. Use the MIB Browser tool to determine which MIB or MIBs is relevant to your enterprise.

Integrating with CA Unicenter TNG

- Install DIGITAL icons
- Create the classes and objects in the Unicenter TNG database that are needed to define DIGITAL servers
- Enable the identification of DIGITAL servers by adding Distributed State Machine policies and definitions
- Enable DIGITAL server to report status through a Finite State Machine
- Install trap files
- Install the ServerWORKS System Browser
- Install the ServerWORKS Alarm Configuration utility



Note: Installing ServerWORKS integration components requires modifying Unicenter TNG files. The installation process preserves the original files by making back up copies of them.

Prerequisites for the Installation

Before you integrate ServerWORKS 4.0 with CA Unicenter TNG, make sure you meet the software prerequisites listed in this section.



Note: In addition to the software prerequisites listed in this section, you should be aware of the minimum hardware requirements to run CA Unicenter TNG. Refer to your CA documentation for this information.

You may also want to familiarize yourself with the hardware and software requirements for a complete ServerWORKS installation. Refer to the *DIGITAL ServerWORKS Manager Installation and User Guide*. A qualified CA Unicenter TNG administrator should install the ServerWORKS integration components.

Integrating with CA Unicenter TNG

To integrate ServerWORKS with TNG, you need:

- Windows NT 4.0 (on an X86 processor)
- CA Unicenter TNG 2.1
- SQL Server 6.5
- TCP/IP network protocol
- SNMP



Note: You should verify that CA Unicenter TNG 2.1 was installed with agent technology. Agent technology, which is an option in the Unicenter TNG installation, is required to monitor and manage DIGITAL servers.

Integrating ServerWORKS Components

To install the ServerWORKS components on an NT console:

1. Log on to the console using an account with administrator privileges.
2. Close all programs that are running, including CA Unicenter TNG, mail programs, and the Microsoft Office shortcut bar.
3. Insert the ServerWORKS Manager in the CD drive of the console system.
4. From the Start menu, run setup.
5. Respond to the setup program prompts for the following information:
 - a) At the first prompt, enter the path for the CA Unicenter TNG directory on the management console. ServerWORKS installation setup will automatically default to the current version running. If you have both TNG V2.1 and V2.2 on your system enter the path name to TNG V2.1 here.
 - b) Next, enter the path where you want to install the ServerWORKS components. You are prompted to identify the root directory for the installation. The installation procedure automatically creates two directories under the root. These directories are DIGITAL and SWMGR.

The following example shows a typical ServerWORKS path:

```
C:\ProgramFiles\Digital\SWMGR
```

- c) Enter the path to CA Unicenter TNG Common Object Repository (database).
- d) Enter a User ID and Password for the repository.

As the installation script runs, it displays messages telling you what it is installing. The installation script creates two log files: SWInstall.log and SWInstallErr.log.

6. Restart your system.

Integrating with CA Unicenter TNG

Confirming the Installation

To confirm that the installation succeeded:

1. Start CA Unicenter TNG at your management console.
2. In the CA Unicenter TNG program group, click Class Browser.
A dialog box provides the path to the Common Object Repository.
3. Click OK.
The Unicenter TNG Class Browser window is displayed.
4. From TNG Root, click Managed Object → Agent.
5. Verify that DigAgNT appears in the list of agents.
6. From TNG Root, click Managed Object → Host.
7. Verify that Digital Server appears in the list of hosts.

Uninstalling ServerWORKS from an NT System

ServerWORKS Manager is tightly integrated with CA Unicenter TNG. As a result, if you need to uninstall ServerWORKS Manager integration components, DIGITAL strongly recommends that you uninstall Unicenter TNG. Uninstalling Unicenter TNG removes ServerWORKS Manager integration components. You can then reinstall Unicenter TNG.

If you choose not to follow the DIGITAL recommendation, you can run an uninstall script.

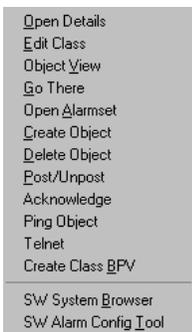
1. Locate and run `uninstal.bat` in the CA Unicenter TNG subdirectory on the ServerWORKS Manager CD.

The uninstall batch file attempts to restore your system to the state it was before you installed ServerWORKS. It does not, however, remove DIGITAL OIDs from the data collector configuration file.

2. Restore Unicenter TNG to its pre-ServerWORKS state:
 - a. Go to the `TNG\bin` directory.
 - b. Locate and run `uninstng.bat`.

This restores the original Unicenter TNG files and rolls back any ServerWORKS changes to the Common Object Repository.

Learning What You Can Manage



When you integrate DIGITAL ServerWORKS Manager with CA Unicenter TNG, you gain access to these tools:

- ServerWORKS System Browser
- ServerWORKS Alarm Configuration utility

To quickly access these tools:

1. On a CA Unicenter TNG map, select a DIGITAL device.
2. Click the right mouse button. System Browser and Alarm Configuration are now choices on the pop up menu that is displayed.

The menu options are displayed as

- SW System Browser
- SW Alarm Config Tool

Managing DIGITAL Objects

When you start CA Unicenter TNG , DIGITAL servers and clusters are discovered automatically and are displayed on the network map. Initially, the DIGITAL devices are represented by NT server icons. As the discovery process continues, however, the devices are reclassified and they are represented by DIGITAL icons.

You can work with the DIGITAL icons much as you would with other symbols on the map. Table 4-1 lists and describes the DIGITAL icons.

Table 4-1 DIGITAL Icons

Icon	Represents
	DIGITAL Intel-based or Alpha-based Server
	DIGITAL Agent

You can manage DIGITAL devices through the DIGITAL menu. You can also use Unicenter TNG tools. Examples of what you can do include:

- Get device status using Node View
- Generate reports about DIGITAL devices
- Manage devices using the DIGITAL MIB

Integrating with CA Unicenter TNG

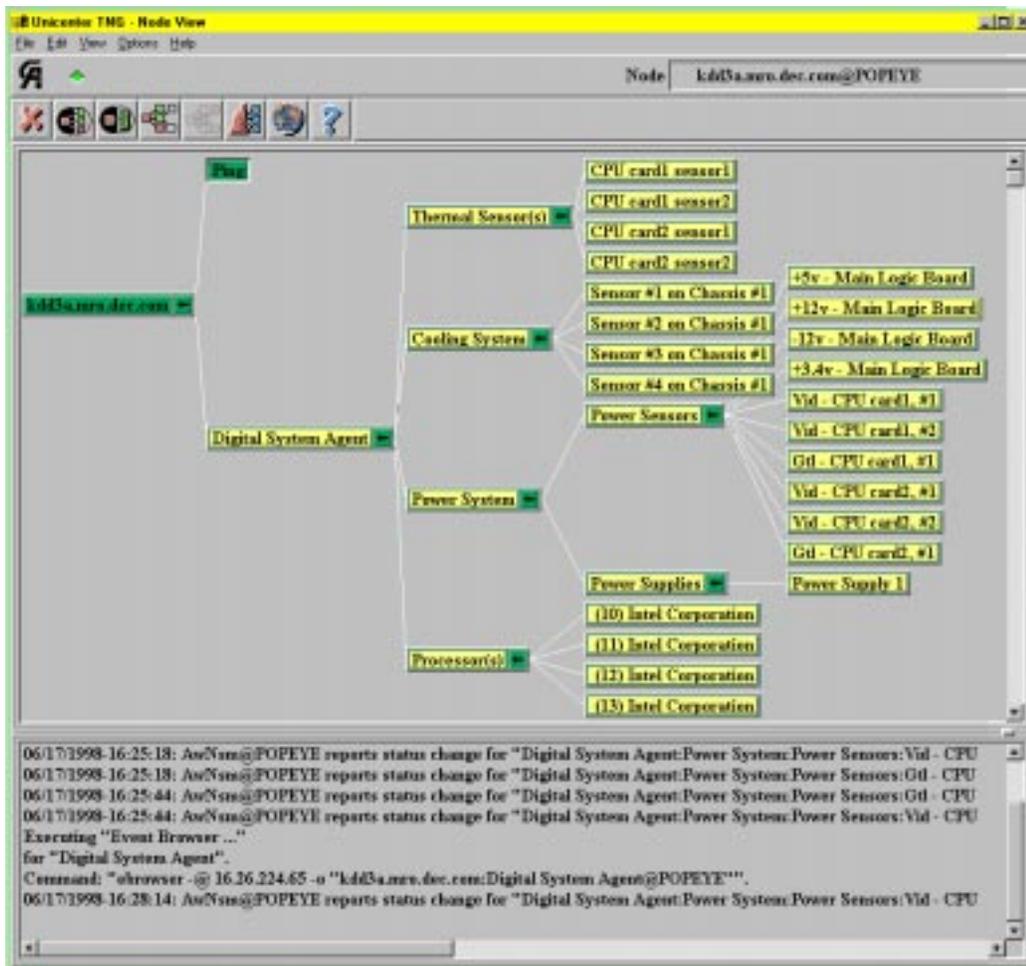
Getting Device Status Using Node View

You can get status information about DIGITAL managed systems through a Unicenter TNG window.

1. Double-click the icon for the DIGITAL managed system you want to view.
2. Using the right mouse button, click the Digital System Agent icon.

Integrating with CA Unicenter TNG

Figure 4-1 Unicenter TNG — Node View Window



Generating Reports about DIGITAL Devices

You can use the Unicenter TNG reporting tool to generate reports about DIGITAL devices. You can display the reports on your console, print the reports, or save them to a file.

To generate a report:

1. Select a DIGITAL device on a Unicenter TNG map.
2. From the CA Unicenter TNG menu bar, click Report → Object by Class.
3. In the Object Report by Class window, go to the Object Classes list.
4. Scroll through the list and select
 - DIGITAL Server to generate a report about servers
 - DigAgtNT to generate a report about DIGITAL agents

Managing Devices Using the DIGITAL MIB

You can access the DIGITAL MIB through Unicenter TNG's Object View. You can then query and set MIB variables on DIGITAL devices.

To use the MIB to manage a DIGITAL device:

1. Select a DIGITAL device on a Unicenter TNG map.
2. Click the right mouse button to display the pop up menu.
3. Click Object View.

The Unicenter TNG Object View window is displayed.

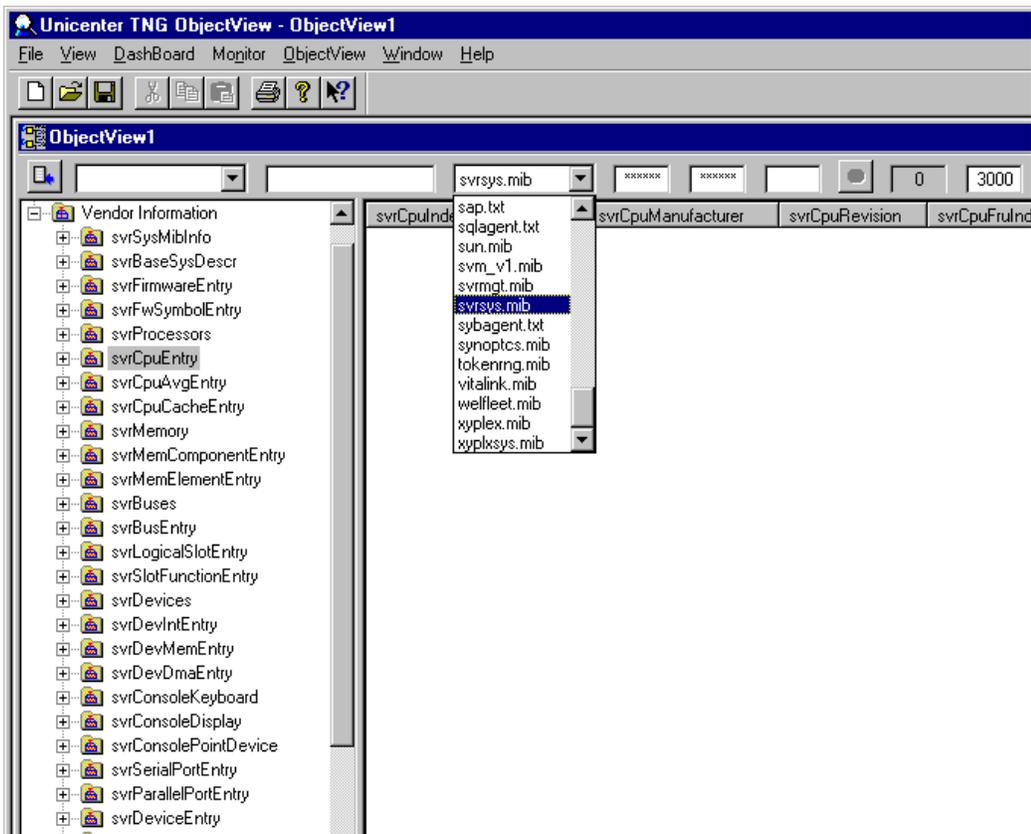
4. In the Object View window, select Vendor Info.

The DIGITAL svrSys (server system) MIB is displayed.

Integrating with CA Unicenter TNG

5. To view other DIGITAL MIBs, from the MIBs list select
 - svrmgt.mib (server management) to view the instrumentation in the DIGITAL agent
 - ntc.mib to view DIGITAL cluster management information
 - svrclu.mib (cluster management)
 - svrntc.mib (cluster management)

Figure 4-2 Viewing the DIGITAL svrSys MIB in CA Unicenter TNG



Configuring Traps and Trap Destinations

Using SNMP traps, you can ensure that DIGITAL devices notify CA Unicenter TNG when particular or exceptional events occur.

The DIGITAL svrmgt.mib defines four levels of traps: high, medium, low, and informational. Each trap can indicate a variety of conditions depending on how a particular server agent is configured. Each of these traps has a unique trap ID.

To view DIGITAL traps in CA Unicenter TNG, you designate a ServerWORKS Manager Console as the trap destination, then configure the ServerWORKS Console to forward the trap to TNG.

Designating the Trap Destination

To designate a ServerWORKS console as a trap destination:

1. Make sure that SNMP is configured as a service on each DIGITAL device where a ServerWORKS agent is installed.
2. Choose Tools → Trap Control from the ServerWORKS Manager Console or from the Alarm Configuration utility.
3. Click the Remote Destination tab.
4. Enter the Host name (managed device name).
5. Choose one of the following:
 - Accept the default community (for example, Public for Windows NT).
 - Deselect the default option and enter Get Community and Set Community names.

Integrating with CA Unicenter TNG

6. Click Get Info to display the current community name and trap destination information for the host.
7. Specify the destination by community name. The destination is included in the trap message.

Forwarding Traps

To configure the ServerWORKS Manager Console to forward the trap to a CA Unicenter TNG console

1. From the ServerWORKS Console menu, choose Tools → Trap Control.
2. Click the Local Forwarding tab and perform the following tasks:
 - a. Enter the community name, if necessary. Public is the NT default community. You can change the community name, but the name you use applies to all forwarding destinations in the list.
 - b. Select Forward alarms as traps.
 - c. Click Get Info to see a list of the forwarding addresses for the selected community.
3. To add a forwarding address, click Add. On the Add dialog box, enter the Address and Port number. Then click OK.
4. Click Apply to enable the changes and Close to close the dialog box.



Note: For complete instructions about configuring SNMP on a managed device and trap forwarding, refer to the *DIGITAL ServerWORKS Manager Installation and User Guide*.

Integrating with CA Unicenter TNG

Integrating with Compaq Insight Manager **5**

You can easily manage a mixed environment of DIGITAL and Compaq servers using either the DIGITAL ServerWORKS Manager Console or the Compaq Insight Manager Console.

Refer to the topics and tasks in this chapter for information about:

- Managing DIGITAL servers from the Insight Manager Console
- Managing DIGITAL servers from a web browser with web-enabled agents
- Managing Compaq servers from the ServerWORKS Manager Console

Use this chapter in conjunction with the *DIGITAL ServerWORKS Manager Installation and User Guide* and the *Compaq Insight Manager User Guide*.

ServerWORKS integration with Insight Manager is available on Windows NT, SCO Unix, and Netware platforms.

Integrating with Compaq Insight Manager

Managing DIGITAL Servers from the Insight Manager Console

A new group of agents, which are called Compaq Management Agents for Digital Servers, enable the Insight Manager Console to query and manage DIGITAL servers.

These agents are installed automatically during a ServerWORKS installation. If you are an Insight Manager user, refer to the Compaq documentation for installation information.

Adding DIGITAL Servers to Insight Manager Device List

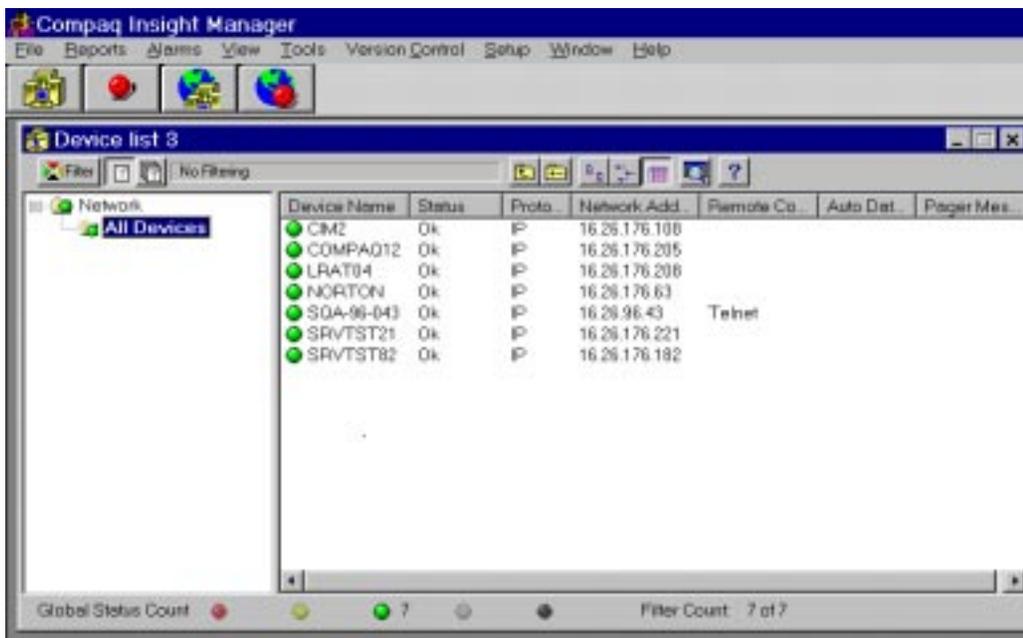
To enable the Insight Manager Console to query a DIGITAL server, you need to add the server to the Insight Manager device list:

1. At your management console, start Insight Manager.

The Device List window is displayed.

Integrating with Compaq Insight Manager

Figure 5-1 Compaq Insight Manager Device List Window

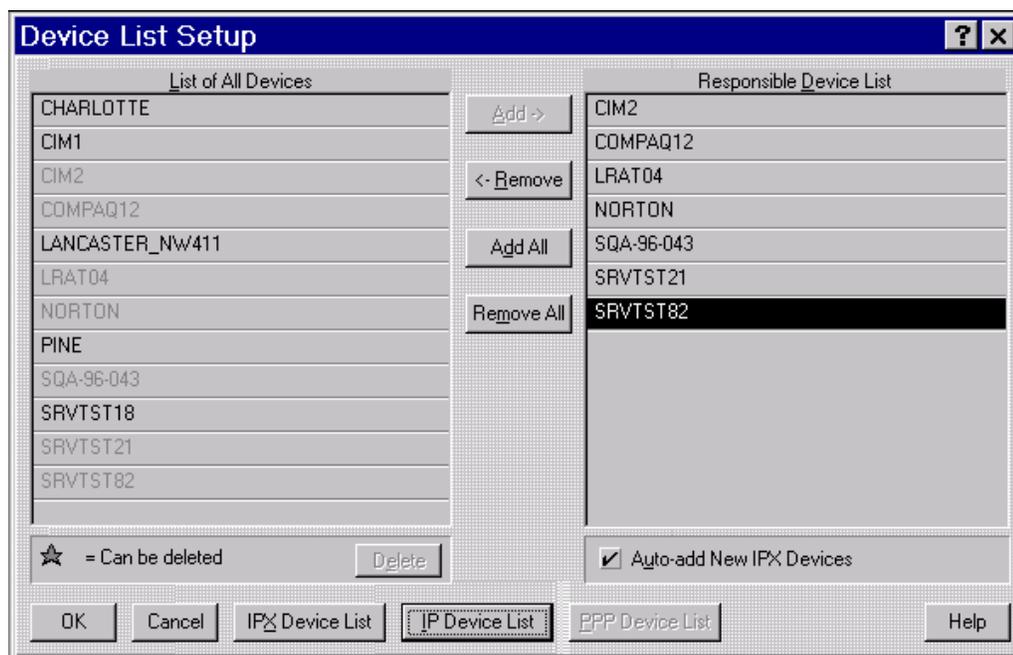


Integrating with Compaq Insight Manager

2. Click Setup → Device List.

The Device List Setup window box displays a list of IP devices (and associated IP addresses) that can be managed by Compaq Insight Manager.

Figure 5-2 Compaq Insight Manager Device List Setup Window



Integrating with Compaq Insight Manager

3. Click IP Device List.
4. In the IP Device List dialog box, click New.
A dialog box is displayed that allows you to add an entry to the IP Device List.
5. Enter the device name and the IP address of the DIGITAL server.
6. Click Add and then click Close.
7. In the Device List Setup window, select the server and click Add to move it to the Responsible Device List.



Note: To obtain the IP address from a WINS server, a DNS server, or from a Hosts file, click WINS/DNS IP Address Resolution. This feature allows Compaq Insight Manager to query a WINS or DNS server (if supported on your network) or to search a Hosts file to obtain the IP address of a given device using the Windows Sockets (WINSOCK) protocol. If you use this feature, you avoid the need to enter static IP addresses for each device.

Learning What You Can Manage

After you add a DIGITAL server to the Insight Manager device list, you can monitor and manage it as you would any Compaq server on the network.

To begin managing a DIGITAL server:

1. In the Device List window, right-click the DIGITAL server and choose View Device Data.

The Device View window is displayed. Notice that a DIGITAL server symbol replaces the Compaq server symbol in the center of the window.

2. From the Device View window, choose one of the seven groups of attributes to manage.

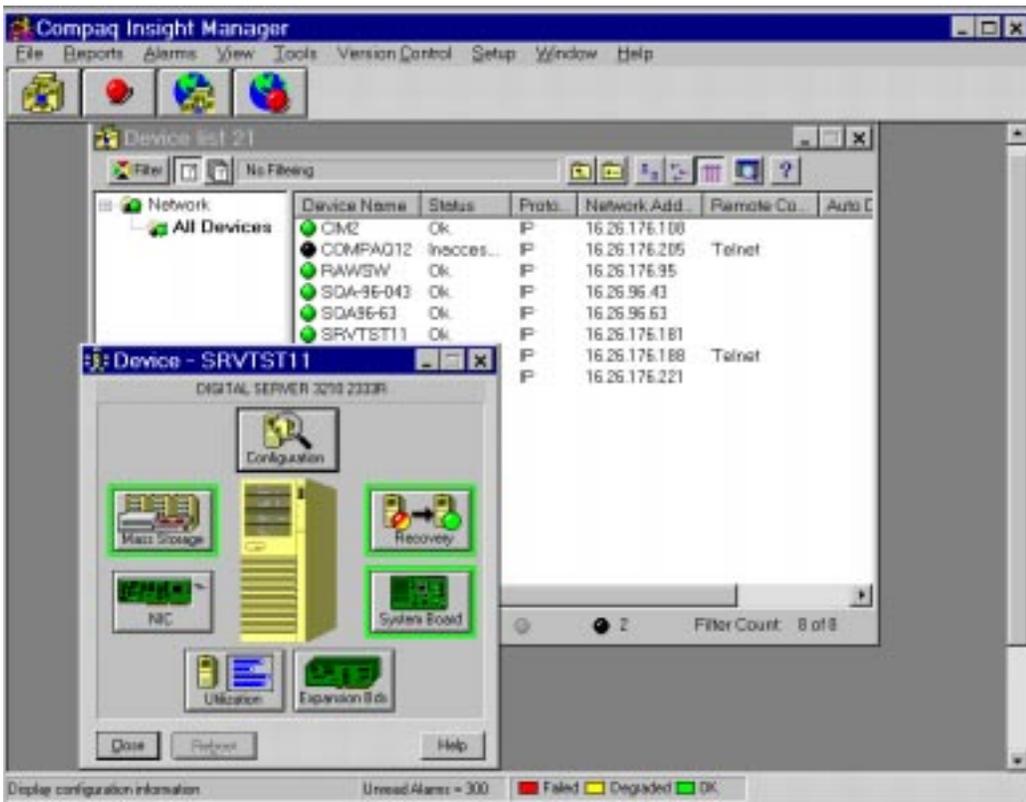
Table 5-1 Compaq Insight Manager Device Groups

Choose ...	To Display...
Configuration	Server attributes such as system name, processor type, system memory, system ports, keyboard, diskette drive, and file system information
Recovery	Environmental information, particularly temperature and fan status
System Board	A description of the board, including system memory, number of CPUs, and I/O devices
Expansion Boards	Descriptions of PCI and EISA cards installed on the server

Integrating with Compaq Insight Manager

Choose ...	To Display...
Utilization	Real time information about CPU usage
NIC	Description and status of each network interface card on the server
Mass Storage	Information about file system usage and hard disk capacity

Figure 5-3 DIGITAL Server in Insight Manager Device View Window



Integrating with Compaq Insight Manager

Managing DIGITAL Servers with Web-Enabled Agents

You can manage DIGITAL servers from a web browser over an intranet or the Internet. The software components necessary for web-based management are included on the ServerWORKS Manager CD.

These components include the DIGITAL agents for a specific Digital server and the Compaq agents for DIGITAL servers, which allows the DIGITAL agents to be viewed by Compaq Insight Manager and the Compaq Insight Manager HTTP Management server.

To install these components, install the DIGITAL server agent. The process of installing this agent automatically invokes the process to install Compaq agents for DIGITAL servers and the Compaq Insight Manager web management server.

Once installed, you can view DIGITAL server system details viewed using Microsoft Internet Explorer version 4.01 or greater. Direct the browser to the following URL:

`<node name>:2301`

Please note the it is important that you first verify that the web browser you intend to use conforms with the requirement as indicated above.

Managing Compaq Servers from the ServerWORKS Manager Console

ServerWORKS Manager gives you the ability to launch an application like Insight Manager *with context*. When you launch Insight Manager with context, the application opens with the data for a Compaq object that you have selected in ServerWORKS.



Note: If you installed Insight Manager before you installed ServerWORKS Manager, you may receive error messages involving SNMP trap services.

If you receive such an error message when you start ServerWORKS, you can ignore it. If you receive an error message when you launch Insight Manager, you need to change how SNMP trap services are configured. Go to the Window NT Control Panel and click Services. In the Services dialog box, change the SNMP Trap Services setting from Automatic to Manual.

The problem occurs because ServerWORKS attempts to use SNMP trap port 162, which Insight Manager has already taken.

Launching With Context

To set up Insight Manager to launch with context, perform the following steps:

- Configure the application to associate it with the object type.
- Specify the application as the default action for the object type.

Integrating with Compaq Insight Manager

Configuring Insight Manager

To configure Insight Manager:

1. From the Console, choose Tools → Application Launch.
2. On the Configure Application Launch dialog box, select or enter the following information:
 - a. In the Object field, select the object type (for example Server.Compaq).
 - b. In the Menu Item Name field, enter the name for the menu option (for example, Insight Manager).
 - c. In the Application Path field, click on the browse button and search for the application. Be sure to use the full path and exact spacing (for example: “c:\Program Files\Compaq\Insight Manager\cim.exe”).
 - d. In the Toolbar button field, choose any 16x16 pixel bitmap.
3. On the Application Command Line Setup group of the dialog box, enter the command that will launch the application. Always use exact spacing and match case sensitivity.
To launch Insight Manager, use the following command line:

```
cim.exe -ObjIPAddress=<Internet Address>
```

You have the option of selecting statements for the command line. The Command Line Parameters selection list lets you choose from commonly used parameters, which are added to the Parameters and switches field. If your command line includes several commands, select Allow multiple objects and enter a delimiter type.
4. Choose Add to place the menu name on the Actions menu and add the bitmap as a toolbar button.
5. Choose Close.



Integrating with Compaq Insight Manager

Specifying Insight Manager as the Default Action

To specify Insight Manager as the default action:

1. Choose Tools → Options.
2. On the Options dialog box, click on the Default Actions page.
3. From the object list, select the object type.
4. From the Action list, select Insight Manager.
5. Choose Close.

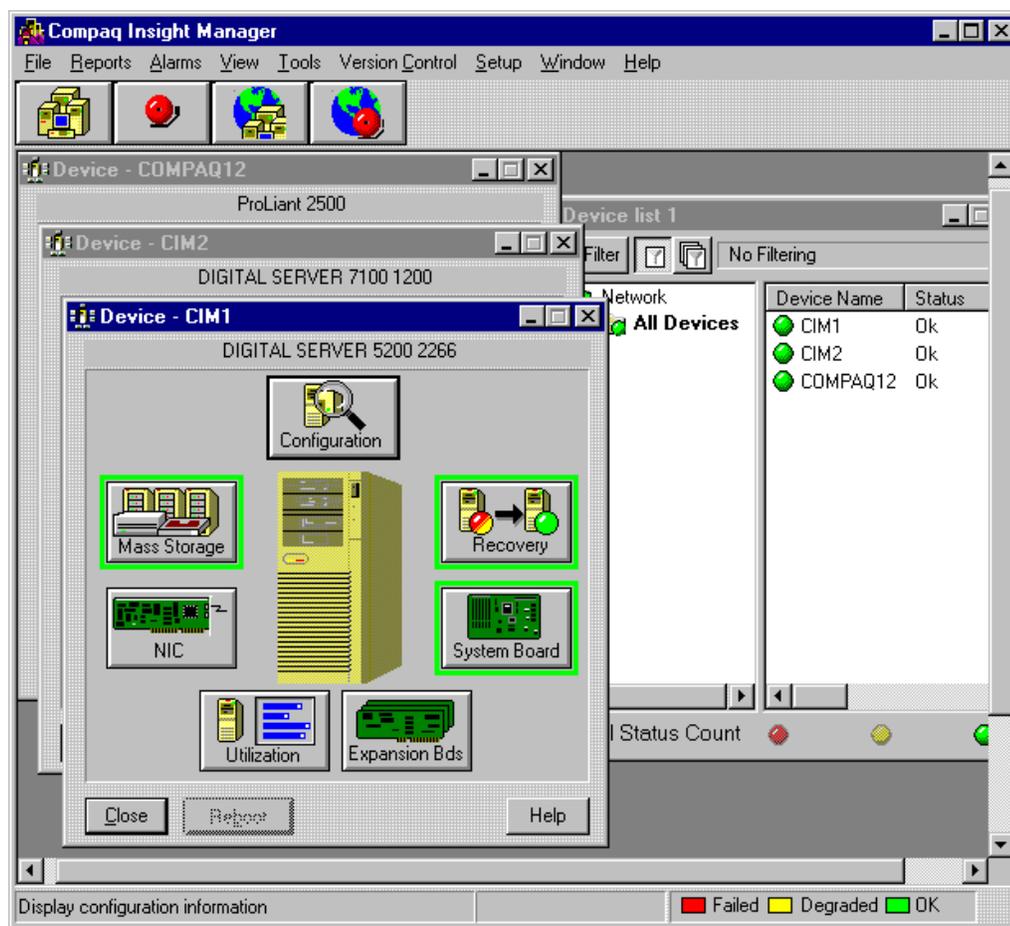
Opening Insight Manager With Context

To open Insight Manager with context:

From a Discovery map or hierarchical view, double-click on a Compaq server. Figure 5-3 illustrates a Compaq server as seen through Insight Manager and launched from a ServerWORKS Discovery map.

Integrating with Compaq Insight Manager

Figure 5-4 Insight Manager launched from ServerWORKS Manager Console



Compiling ServerWORKS MIBS into ManageWise



This appendix explains how to compile the ServerWORKS DIGITAL MIBs into ManageWise. The ServerWORKS CD contains seven MIB files that you can compile into ManageWise:

- bmc.mib – the SNMP MIB of BMC Software
- exide.mib – the SNMP MIB of Exide Electronics
- ntcmtg.mib – DIGITAL's NT cluster management MIB
- raidmgr.mib – DIGITALS's RAID Manager for Windows MIB
- stdupsv1.mib – Internet Standard UPS MIB (based on RFC-1628)
- svrmgt.mib – DIGITAL's server management MIB
- svrsys.mib – DIGITAL's MIB describing a management interface to system information

Compiling ServerWORKS MIBS into ManageWise

Compiling MIB Files

To compile the ServerWORKS MIB files:

1. Locate the files on the ServerWORKS CD. They are in the following directory:

```
unsptmib\digital\unsup\novell
```

2. Copy the files to these ManageWise directories:

```
MW\NMS\SNMPMIBS\CURRENT
```

```
MW\NMS\SNMPMIBS\ALLMIBS
```



Note: Novell recommends keeping a copy of all MIBs in the ALLMIBS directory.

3. Select Tools → SNMP MIB Compiler.



Note: The MIBs supplied on the ServerWORKS CD compile in ManageWise but generate a warning message concerning Annotations. Disregard this message, since the MIBs function properly.

The SNMP MIB Compiler compiles all the MIB text files located in the CURRENT directory into a single binary file:

```
MW\NMS\BIN\SNMPMIBS.BIN.
```

The SNMP MIB Browser and the SNMP protocol decoder use this file. The SNMP MIB Compiler also adds or updates any trap definitions to the alarm type database file, MW\NMS\NMSDB\TRAP.BTV, for use by the ManageWise alarm system.

ServerWORKS Alarms and Browser *B*

This appendix reviews the ServerWORKS Alarm utilities and System Browser. For more detail refer to the *DIGITAL ServerWORKS Manager Installation and User Guide*.

Setting Alarms

If you have integrated ServerWORKS Manager on an X86-based NT console, you can choose to set and view alarms and traps on DIGITAL devices using the Alarm Configuration utility.



Note: If you have integrated ServerWORKS Manager on an HP-UX or DIGITAL UNIX console, use the enterprise manager's Event Browser to view alarms for DIGITAL servers. You must first configure traps and traps destinations for your DIGITAL servers. For more information refer to the chapter in this guide that discusses your enterprise manager.

Alarm Configuration is a ServerWORKS Manager utility that is available from the DIGITAL menu.

ServerWORKS Alarms and Browser

The Alarm Configuration tool provides the mechanism for ServerWORKS agents to intelligently monitor the devices on which they are installed. Instead of relying on constant polling from the management console, ServerWORKS agents that are installed on managed systems interrogate themselves and notify the console when a unit reaches an alarm threshold, saving considerable bandwidth.

The Alarm Configuration Window and Toolbar

Use Alarm Configuration to set alarms on DIGITAL devices. You can manage minimal health alarms and user-defined console alarms.

The Alarm Configuration window displays a list of network hosts (the name assigned to the object) in the left pane and the alarm description and details in the right pane.

The alarm details include the host name, IP address, object type, a description, the alarm severity, the source of the alarm, the category, and its enabled status. You can choose which data is displayed and the order in which it appears. You can sort a column's data alphabetically by clicking on a column label. Figure B-1 illustrates the Alarm Configuration window.

ServerWORKS Alarms and Browser

Figure B-1 Alarm Configuration Window

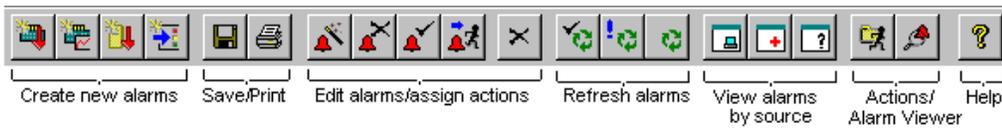
Host	Category	Status	Description	Severity	Action	Source
daisy.mso.dec.	Fan Sensors	Enabled	Fan Sensor(Sensor #5 on Charac...	High		Console
daisy.mso.dec.	Power Suppl...	Enabled	Power Supply Sensor(Power Sen...	High		Console
lexis.mso.de.	Network Dubs...	Enabled	Network Dubsound Error(UP #3D...	Medium		Console
garret.dec.c	Cluster Group...	Enabled	Cluster Group Status(EXT_DRV_DI...	High		Console
garret.dec.c	Cluster Group...	Enabled	Cluster Group Status(EXT_DRV_FI...	High		Minimal Health
garret.dec.c	Cluster Group...	Enabled	Cluster Group Status(EXT_DRV_FI...	High		Minimal Health
garret.dec.c	CPU Usage	Enabled	CPU Usage(CPU #1 DEC-221154)	High		Minimal Health
dhcp-26-289...	Disks	Enabled	Disk(/Device/Harddisk0) - Not Fu...	High	MHHigh	Minimal Health
dhcp-26-289...	Disks	Enabled	Disk(/Device/Harddisk0) - Flaming	Informat...	MHInfo...	Minimal Health
dhcp-26-289...	Disks	Enabled	Disk(/Device/Harddisk0) - Warning	Medium	MHMed...	Minimal Health
dhcp-26-289...	Disks	Enabled	Disk(/Device/Harddisk1) - Not Fu...	High	MHHigh	Minimal Health
dhcp-26-289...	Disks	Enabled	Disk(/Device/Harddisk1) - Flaming	Informat...	MHInfo...	Minimal Health
dhcp-26-289...	Disks	Enabled	Disk(/Device/Harddisk1) - Warning	Medium	MHMed...	Minimal Health
dhcp-26-289...	Processors	Enabled	Processor(CPU#1 x86 Family 6 Mo...	High	MHHigh	Minimal Health
dhcp-26-289...	Processors	Enabled	Processor(CPU#1 x86 Family 6 Mo...	Informat...	MHInfo...	Minimal Health
dhcp-26-289...	Processors	Enabled	Processor(CPU#1 x86 Family 6 Mo...	Medium	MHMed...	Minimal Health

For help, press F1 | 3 Host(s) selected | 1 Alarm(s) selected | 14 Alarm(s) displayed

ServerWORKS Alarms and Browser

You can use the menu options or the companion toolbar to work with alarms. (Figure B-2.)

Figure B-2 Alarm Configuration Toolbar



Minimal Health Alarms

Minimal health is an installation option that provides a set of hardware-specific thresholds on X86-based NT platforms: Minimal health monitors environmental and operating conditions affecting:

- Power supplies
- Fans
- Voltage
- Temperature
- Hard disks
- Processors
- Memory components



Hosts with minimal health enabled are identified by the red cross icon next to the host name in the Alarm Configuration window. The icon is displayed when at least one minimal health alarms is in effect.



Note: For more information about minimal health alarms, refer to the *DIGITAL ServerWORKS Manager Installation and User Guide*.

User-Defined Console Alarms

Using the Alarm Configuration utility, you can create four kinds of user-defined alarms:

- System (interface) status alarms — Report when a system or interface, such as an adapter, changes its status.
- SNMP traps — Send SNMP messages that are triggered by the SNMP agent.
- Component status alarm — Report the operational status of a DIGITAL server or node object.
- Component threshold alarms — Report when a characteristic of a DIGITAL server meets a specified condition. For example, if the temperature exceeds a specified level, a threshold alarm is triggered.



Note: For more information about user-Defined alarms, refer to the *DIGITAL ServerWORKS Manager Installation and User Guide*.

Using the System Browser

If you have integrated ServerWORKS Manager on an X86-based NT console, you can use the DIGITAL System Browser to obtain information about both static and dynamic parameters found in DIGITAL objects.

The System Browser (Figure B-3) organizes this information in a way that makes the most sense for DIGITAL managed devices and presents the information in a user-friendly manner. The information may be accessible using your enterprise management tools, but it is presented generically and may not be as informative.

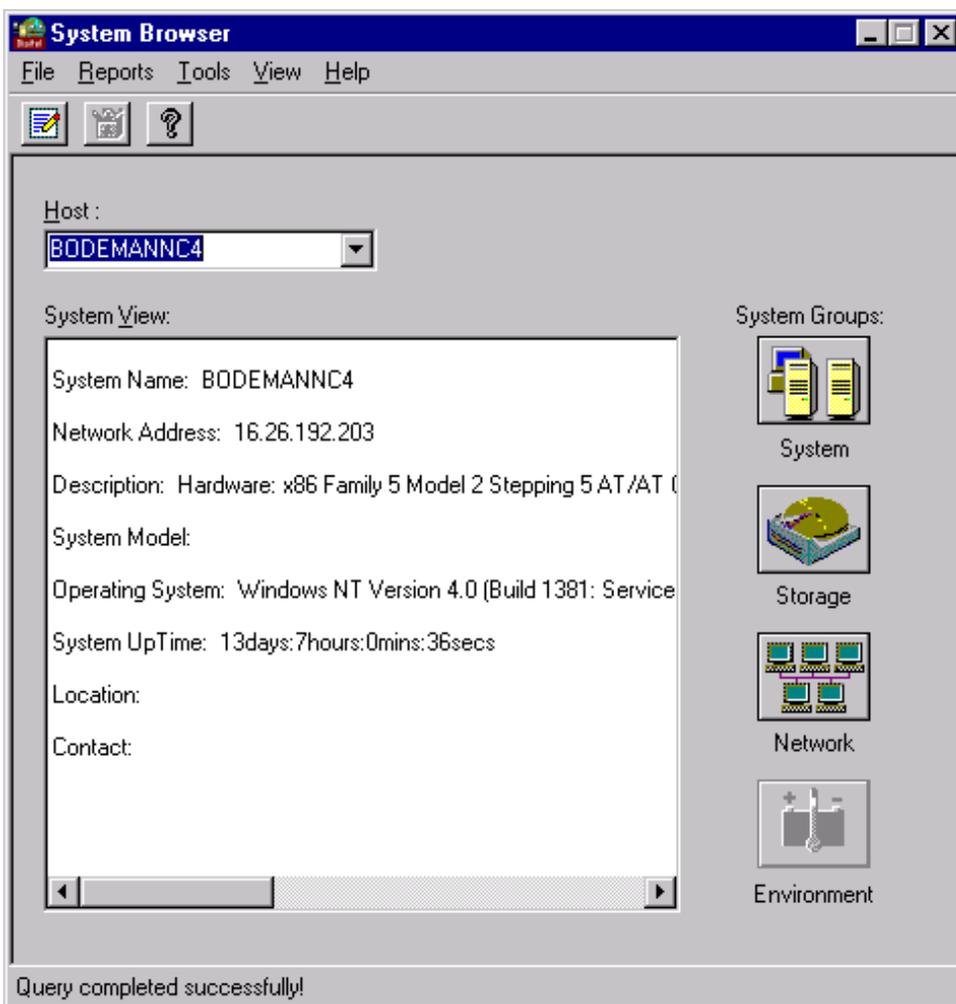
To work with the System Browser, select SW System Browser from the pop up menu or

1. Double-click a DIGITAL icon.
2. Click the DIGITAL System Agent icon.

The System Browser uses information that SNMP agents provide to display:

- System configuration information that generally does not change.
- Current information that is refreshed each time you examine an object from the System Browser. Table B-1 shows the type of information found in each of the System Browser windows.
- Historical information. When current information is not available due to network or system problems, the System Browser displays node information that was previously collected and saved.
- Information about cluster members and cluster resources.
- Dynamic or historical graphed data.

Figure B-3 System Browser Window



ServerWORKS Alarms and Browser

Table B-1 System Browser Information

This Window	Displays This Information
System Browser	Host name Network (IP) address Description Physical location and contact Model and operating system* Length of system has been running (Up Time)
System	General information I/O devices Processor FRU* Cluster (when the server is a cluster member)*
Storage	Disks Disk Partitions File System Storage Memory* Memory Component Slots (SIMMs and DIMMs)*
Network	Interface Statistics
Environment*	Thermal Sensors Voltage Sensors Cooling System (fans)* Power (supply)

*Not always present on all systems

Advanced User Functions C

This appendix explains how to use the command line to launch ServerWORKS Manager applications, and how to make modifications to the ServerWORKS Manager initialization file (swmgr.ini).

Command Line Access

Command line access lets you run specific ServerWORKS Manager applications from the DOS prompt. Modifying the initialization file will alter what information you see and how it is displayed by ServerWORKS Manager.

Command line access is intended for use with management environments other than ServerWORKS Manager. These commands will launch selected ServerWORKS Manager applications. The commands do not replace the graphical interface access to ServerWORKS Manager.

Modifying the Initialization File

The initialization file (swmgr.ini) lets you refine your use of ServerWORKS Manager. This section surveys the available initialization entry's that are useful for daily use and when using pagers, printers, and log files.



WARNING: The information in this appendix is intended for highly advanced users ONLY. Experimenting with any items detailed in this appendix could result in problems using your existing ServerWORKS Manager application.

Starting Applications from the Command Line

The following commands allow other management environments to launch selected parts of ServerWORKS Manager. These commands may fail under some environments. To avoid failure make sure the environment meets the following conditions:

- IP connectivity between the managed system and the ServerWORKS machine
- Agents must be installed on systems you wish to monitor
- The ServerWORKS Manager environment must be win-32 based
- ODBC runtime access should be available. The required Microsoft executables are available on the ServerWORKS Manager CD.

To use the commands as shown in this section, access a DOS box, or access Run from the Start menu. The applications available from the command line are:

- ServerWORKS Manager console
- Alarm configuration
- System browser
- Launchme



WARNING: When starting ServerWORKS Manager applications from the command line use only the command parameters and values documented. Other combinations are not supported.

The following sections detail each command and display any optional switches available in a table after each command description. Use spaces to separate commands, switches and values when using the command line interface.

Starting ServerWORKS

To start the ServerWORKS Manager console from the command line, access the directory level as shown.

```
C:>cd\program files\digital\swmgr  
C:>pwmgmt.exe swmgr.ini
```

This starts the ServerWORKS Manager console.

Starting Alarm Configuration

To start the alarm configuration tool from the command line, access the directory level as shown.

```
C:>cd\program files\digital\swmgr  
C:> alarmcfg
```

The optional switches allow you to point to specific servers or communities. The switch will not be recognized unless these objects have already been discovered by ServerWORKS Manager and are in the ServerWORKS Manager database. If no entries exist in the database, use the -1 switch to create an entry in the database.



WARNING: Server names should not contain dashes (-) in the naming conventions of ServerWORKS Manager. Most applications will parse the dash (-) as a command line switch. If the name does contain a dash, enclose the server name in double quotes. Example: -n <servername>

Advanced User Functions

Example:

```
C:>alarmcfg -n "NiceServer-1" -cg eastcoast
```

Switch	Switch value	Example	Description
-n	-n <server name>	-n servername -n servername.new.com	Displays alarm configuration for the selected server.
-i	-i <IP address>	-i 16.151.168.26	Displays alarm configuration for this IP address.
-cg	-cg <communitygetname> (default is public)	-cg eastcoast	Displays alarm configuration for specified get community.
-cs	- cs <communitysetname> (default is public)	-cs westcoast	Displays alarm configuration for specified set community.
-l		-l -n servername -cs fred	Use with other network management environments. Inserts alarm information into the ServerWORKS Manager database.

Starting System Browser

To start the system browser from the command line, type the following:

```
C:>cd\program files\digital\swmgr
C:> smb
```

Examples:

```
C:> smb -n 122.122.33.44 -c private -g new -h
C:> smb -f1 netview graph.exe -n winter
```

Switch	Example	Description
-n	-n <servername> -n <122.12.14.155>	Starts the browser with this server.
-c	-c <GetCommunityName> (default is public)	Uses the specified SNMP get community.
-d	-d <SetCommunityName> (default is public)	Uses the specified set community.
-f1	-f1 <NetView graph tool dir> graph.exe -n <servername>	Enables the graphing engine for the NetView environment. Disables access to ServerWORKS Manager graphing. (Only supported in a Tivoli NetView environment.)
-f2	-f2 <OpenView graph tool dir> xgraph.exe -n <servername>	Enables the graphing engine for the OpenView environment. Disables access to ServerWORKS Manager graphing. (Only supported in an HP OpenView environment.)
-p	-p -n <servername>	Recognizes information is coming from TNG. Disables access to graphing and alarm configuration. (Only supported in a Unicenter TNG environment.)
-h	-h	Displays all servers with historical data in the initial system browser drop down listbox.

Starting the Launchme Utility

Launchme.exe was created to aid advanced users in testing executable files that interact with ServerWORKS Manager. Users can launch applications from three places within the ServerWORKS Manager interface:

- Tools → Application Launch (creates launchme as a global application)
- Tools → Options → Tools (creates a new Tools menu option)
- Tools → Action Directory Set Up (creates a new Tools menu option in Alarm Configuration)

To verify parameters, substitute launchme for your own application. Launchme verifies that the launch parameters of the application are correct. It displays a dialog box containing the parameters and syntax of the command line passed to it by the ServerWORKS Manager application. This allows you to verify that the command line was passed correctly to your executable.

In the following example, launchme is used to verify parameters passed to an alarm action. There are three sections to this task:

- Set up launchme as a global application.
- Create the alarm and associate an action (launchme) with it.
- Verify the parameters.

To Set Up Launchme

1. Click Tools→Options→Application Launch.



You can also use the command line.

```
C:\<defaultdirectory>\Launchme <parameter> <parameter>
```

The <defaultdirectory> is normally C:\program files\Digital\swmgr.

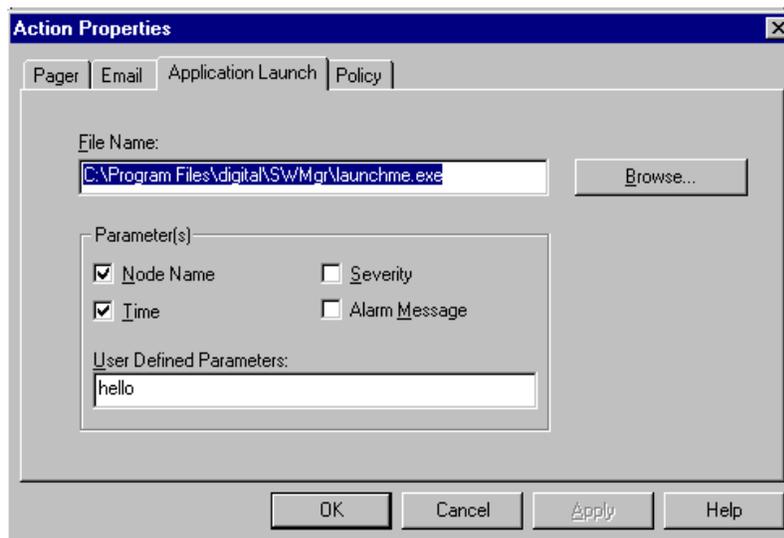
2. Complete the Configure Application Launch dialog box to add launchme to the Actions menu, as shown in the previous diagram.
3. Click Close.

Advanced User Functions

To Create an Alarm and Associate an Action with It

To create an alarm, refer to your *DIGITAL ServerWORKS Manager Installation and User Guide* for information. To associate an action with the alarm, follow these steps:

1. From the ServerWORKS Manager Console, click Tools → Alarm Configuration.
2. Associate an action with a trap or alarm on a host:
 - a. Click the Action Directory button.
 - b. Click Add New → Application Launch.
 - c. In the Action Properties window, enter the launchme file name for the action.
 - d. Choose the parameters that you want to use — you can add a user-defined parameter — and click Apply.

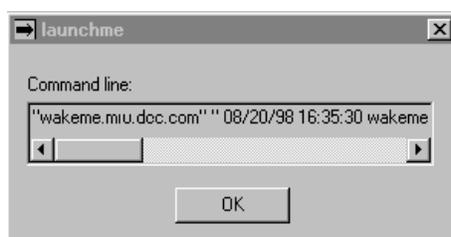


Advanced User Functions

3. Enter the Action name and click Close.

To Verify the Parameters

Wait for the previously created alarm to occur. Because the launchme application is associated with the alarm action, the launchme dialog appears when the alarm is triggered. A typical launchme dialog is shown below.



To verify other parameters, select a host from the map or hierarchical view and choose Actions → Launchme. The dialog box will appear with the parameters as they were set in the Application Launch task.

The launchme display will show you the parameters that were passed. If the correct data was passed, you are ready to substitute the real application for the launchme action in the Action Directory Setup dialog box.

Initialization File Settings

Initialization files let you tailor your use of ServerWORKS Manager, and in some cases troubleshoot problems. This section describes the initialization file settings used by ServerWORKS Manager.

The swmgr.ini file is located in the root directory of your ServerWORKS Manager application: C:\program files\Digital\swmgr. Many of these initialization entries may not exist in the swmgr.ini file. They must be added to the swmgr.ini file in a specific section of the file. If the section does not exist in your swmgr.ini file, add the section and the entry. The examples below show both the section, as [sectionname], followed by an entry.

Before you Begin



WARNING: Before you begin to make any changes, we advise you to copy the `swmgr.ini` file and save it in a local directory for possible future use.

Overall Settings

General information that may be of use to you is included in this section. These ini file settings are located in `swmgr.ini`.

Example:

```
[Setup]
CloseAllAppsOnExit=0
```

Name	Value	Description
[Setup] CloseAllAppsOnExit	0 Leaves all applications open 1 Closes all applications (default)	Leaves open or closes all related applications after exiting from the ServerWORKS Manager Console.
[SNMP Setup] MW_TRAP_SERVICE	<location> (default is snmp-trap)	This trap destination default location may be overwritten by other applications.
[AlarmCfg] LogToOSEventLog	0 Do not log 1 Log: Smart Agents forward alarms, events SNMP traps	Forwards events and alarms to the NT server event log.
TrapDestinationPrompt	1 On, None Off	Adds the prompt to the toolbar.

Advanced User Functions

Pager

Pager settings are especially useful for troubleshooting pager problems and dealing with the limitations of older pagers. You must have a solid understanding of your pager and modem systems before attempting these changes.

Example:

```
[Setup]
PagingMsgSplitFlag=1
```

Name	Value	Description
[Setup] PagerWaitTime	<number of commas> (default is 1)	Number of commas to signify the wait between the dialing numbers. Each comma cause a wait of approximately 1-2 seconds.
PagerRetryCount	<number of retries> (default is 1)	Number of retries to dial pager.
NoPagingPrefix	0 No paging prefix displayed (default) 1 Show paging prefix	Displays on the pager message the host name that generated the alarm.

(continued)

Advanced User Functions

Name	Value	Description
[Setup]		
PagerTextLineCount	<The number of lines to be displayed on the pager> (default is 3, maximum is 20)	Displays the specified number of lines of alarm information. The default displays date and time, system name, and time running.
PagerLogging	0 No log (default) 1 Create log file	Creates a log file in the Temp directory (C:\TEMP as defined in System Properties/Environment tab in NT) that contains the pager activity record for debugging purposes (for example, whether connections succeed or fail). The file name is in the form of pagerNNNNN.log, where NNNNN is replaced by the system clock count.
PagingMsgSplitFlag	0 Do not split (default) 1 Split the lines	Splits alarm information into three 79 character pieces to accommodate older pager displays.

Printing

If printed reports do not appear as desired for alarm configuration or alarm viewer reports, swmgr.ini settings can be added to alter the printouts.

You can choose to alter just one setting, for example the left hand margin only (PrintMarginLeft), or all the settings.

The default settings do not correspond to a specific printer. You will have to try different values to obtain the results you want from your printer. Also, the measurements are taken from the printer that is attached to your system, and do not have a specific type of measure (pica, inches, etc.).

Alarm Configuration

The following code will modify the printout margins for the alarm configuration reports.

Example:

```
[AlarmCfg]
PrintMarginLeft=65
```

Name	Default Setting	Description
[AlarmCfg]		
PrintMarginTop	250	Printed page offset. Increase or decrease value.
PrintMarginBottom	150	Printed page offset. Increase or decrease value.
PrintMarginLeft	40	Printed page offset. Increase or decrease value.
PrintMarginLeftIndent	100	Printed page offset. Increase or decrease value.
ColWidth<nn>	100	Contains the saved column grid widths between sessions.

Advanced User Functions

Alarm Viewer

The following code will modify the printout margins for the alarm viewer reports.

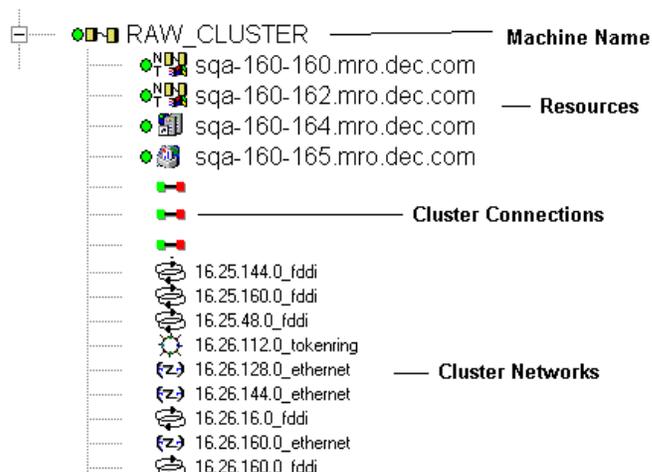
Example:

```
[AlarmViewer]  
PrintMarginTop=65
```

Name	Default Setting	Description
[AlarmViewer]		
PrintMarginTop	250	Printed page offset.
PrintMarginBottom	150	Printed page offset.
PrintMarginLeft	40	Printed page offset.
PrintMarginLeftIndent	100	Printed page offset.
PositionLeft	31	Saved window position (centering).
PositionRight	679	Saved window position (centering).
PositionTop	-9	Saved window position (centering).
PositionBottom	525	Saved window position (centering).
PrintIgnoreFirstDetailLines	0 Ignore the lines	Ignore the first two detail lines on the printed output because this is duplicate data.
	1 Read the lines (default)	

Viewing Cluster Details

Clusters can be seen as single entities in ServerWORKS Manager. The default view shows the basic cluster elements; machine name, resources, and associated applications. Additional details are available by adding the values as shown below.



Example:

```
[SNMP Setup]
ShowClusterConnections=1
```

Name	Value	Description
[SNMP Setup]		
ShowClusterConnections	0	Does not show details (default)
	1	Shows details of cluster
ShowClusterNetworks	0	Does not show details (default)
	1	Shows cluster details

Adding Historical Data Directory

Historical data can be stored and kept in a directory of your choice.

Example:

```
[Historical Data]
Directory=c:\temp\generaldata
```

Name	Value	Description
[Historical Data] Directory	<full path for the root historical data directory>	Creates directory to store historical data.

Advanced User Functions

Event Logger

To start the event logger or ignore certain types of alarms, add the following information to the swmgr.ini file in the section shown.

Example:

```
[eventlogger]  
IgnoreLow=1
```

Name	Value	Description
[eventlogger]		
AutoStartEventLogger	0 Off 1 On (Default)	Starts the event logger on startup of the ServerWORKS Manager Console.
IgnoreInformational	0 Normal (Default) 1 Ignore alarms	Ignores all informational alarms received by the event logger.
IgnoreLow	0 Normal (Default) 1 Ignore alarms	Ignores all low severity alarms received by the event logger.
IgnoreMedium	0 Normal (Default) 1 Ignore alarms	Ignores all medium severity alarms received by the event logger.

Data Collector

The data collector is one of three tools used to gather alarm data. To start data collection immediately upon startup, or ignore some of the informational alarms, add the following information to the swmgr.ini file.

Example:

```
[DataCollector]
AutoStartDataCollector=0
```

Name	Value	Description
[DataCollector]		
AutoStartDataCollector	0 Off 1 On (default)	Starts the data collector on startup of the ServerWORKS Manager Console.
IgnoreInternalAlarms	0 Allow informational alarms 1 Ignore informational alarms (default)	Ignore alarms marked as informational.

Gathering Information with the Poller

The poller requests status information from specified objects. The default settings poll objects at a specified interval. The interval may be modified to slow down or speed up the poller. This may be useful if you are experiencing performance degradation.

Example:

```
[snmppoller]
MaxPollsPerSecond=35
```

Name	Value	Description
[snmppoller]		
AutoStartPoller	0=off 1=on (default)	Starts the poller on startup of the ServerWORKS Manager Console.
Max Polls Per Second	<seconds> default is 10, max is 50	Limits the number of requests being made each second.
Max Events Per Second	<seconds> default is 20, Max is 50	Limits the number of events sent to the event logger each second.
Max Polls Outstanding	<number of items waiting to get> default is 100, 1-100	Limits the number of items waiting for resolution. As items time-out, new items are added to reach this maximum number.

Sizing the Main Window Alarm Status Pane

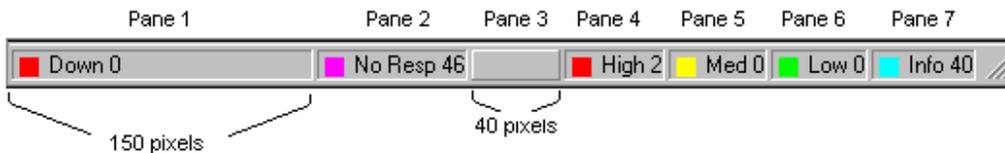
The default sizing of all status panes is done automatically using WIN32 API calls. If you experience sizing problems, you can correct them by modifying the width for any of the seven panes.

The pane measurements are in pixels that are appropriate for your monitor and screen resolution. The default width settings are found in the [Setup] section in the swmgr.ini file after ServerWORKS Manager is run for the first time. Settings may vary. Different screen resolutions produce different values for each pane. In addition, the screen font may require different settings.

In the following diagram, the first pane has been resized to 150 pixels and the third pane, the divider, has been resized to 40 pixels. The settings in the next table are suggested defaults for a monitor with a resolution of 1024x768 pixels.

To change a pane width

1. In the [Setup] section of swmgr.ini, find the UseStatusPanel*n*Size parameter for the number pane you want to modify and change the setting to 1 (Enable).
2. In the corresponding StatusPanel*n*Size parameter, enter the number of pixels for the width you want.



Advanced User Functions

Name	Value	Description
[Setup]		
UseStatusPane1Size	0=Disable 1=Enable	When enabled, will take the value for the StatusPane1Size entry.
StatusPane1Size	0 to 200 (default is 69)	Assigns the number of pixels for the pane.
UseStatusPane2Size	0=Disable 1=Enable	When enabled, will take the value for the StatusPane2Size entry.
StatusPane2Size	0 to 200 (default is 96)	Assigns the number of pixels for the pane.
UseStatusPane3Size	0=Disable 1=Enable	When enabled, will take the value for the StatusPane3Size entry.
StatusPane3Size	0 to 200 (default is 25)	Assigns the number of pixels for the pane.
UseStatusPane4Size	0=Disable 1=Enable	When enabled, will take the value for the StatusPane4Size entry.
StatusPane4Size	0 to 200 (default is 78)	Assigns the number of pixels for the pane.
UseStatusPane5Size	0=Disable 1=Enable	When enabled, will take the value for the StatusPane5Size entry.
StatusPane5Size	0 to 200 (default is 60)	Assigns the number of pixels for the pane.
UseStatusPane6Size	0=Disable 1=Enable	When enabled, will take the value for the StatusPane6Size entry.
StatusPane6Size	0 to 200 (default is 60)	Assigns the number of pixels for the pane.
UseStatusPane7Size	0=Disable 1=Enable	When enabled, will take the value for the StatusPane7Size entry.
StatusPane7Size	0 to 200 (default is 78)	Assigns the number of pixels for the pane.

When Using a Novell NetWare Client

If the NetWare client is installed after ServerWORKS Manager is installed, the entry "NetWare Server Management=NETWARE" in the swmgr.ini file needs to be modified for the NetWare objects to appear in the ServerWORKS Manager Explorer. To modify the entry, locate the [OMM Groups] section. Locate the line:

```
;NetWare Server Management=NETWARE
```

Remove the semicolon (;). You must restart ServerWORKS Manager after you make the change.

Advanced User Functions

The DIGITAL Server System MIB *D*

This appendix documents the DIGITAL Server System management information base (MIB). This MIB provides access to the following information:

- Base system configuration
- Central processor (CPU) configuration and utilization
- System memory
- System buses and slots
- Peripheral devices
- Physical configuration
- Environment



Note: This appendix is intended for users who are knowledgeable about SNMP and MIBs who want to compile the DIGITAL Server System MIB into another network or enterprise manager. Be aware that information contained in the Server System MIB is subject to change. The information in this appendix is accurate at the date of publication.

The Digital Server System MIB

Using the DIGITAL Server System MIB

Before you use the DIGITAL Server System MIB, you should understand:

- Support for MIB variables by hardware platform and operating system
- Conventions used in this appendix
- SNMP syntax extensions in the MIB

Support for MIB Variables

Support for the MIB variables described in this appendix is, to some extent, hardware platform and operating system dependent.

The hardware platforms include:

- X-86-based NT servers
- Alpha NT servers

A variable that returns useful information on one platform may not do so on the other platform. For example, Alpha servers do not return power status.

The operating systems supported are:

- Windows NT
- SCO UNIX
- Netware

The variable for the CPU polling interval returns useful information for NT and SCO but not for Netware.

In addition, some variables return useful information only from DIGITAL servers with advanced manageability features.

Advanced X-86-based Servers

Examples of X-86-based servers with advanced manageability features include:

- X16200
- DIGITAL Server 7100
- DIGITAL Server 5100, 5105, 5200
- DIGITAL Server 3100, 3105, 3200

Advanced Alpha-based Servers

Examples of Alpha-based servers with advanced manageability features include:

- AlphaServer 2100 and 2100A
- DIGITAL Server 7000 Alpha, or AlphaServer 4000 and 4100
- AlphaServer 1000 and 1000A

Conventions Used in This Appendix

In this appendix, each variable is introduced with a statement about hardware platform and operating system support.

For example, the `svrCpuPollInterval` variable is introduced with this statement:

All Platforms/NT and SCO UNIX Operating Systems

This indicates the variable is implemented on all X86-based and Alpha-based hardware platforms and returns useful data under the NT and SCO UNIX operating systems. Tables are introduced only with a statement of operating system support. For example, the `svrFirmwareTable` is introduced with this statement:

The Digital Server System MIB

All Operating Systems

The following table lists the possible components of an introductory statement and gives their meaning.

Statement	Meaning
All Platforms	Implemented on all X86-based and Alpha-based hardware platform.
Advanced X86-Based Servers	Implemented on advanced X86-based servers.
Advanced X86-Based and Advanced Alpha-Based Servers	Implemented on both advanced X86-based servers and advanced Alpha-Based servers.
Advanced X86-Based Servers and All Alpha-Based Servers	Implemented on advanced X86-based servers, in addition to <i>all</i> Alpha-Based servers.
All Operating Systems	Returns meaningful data under the NT, SCO UNIX, and Netware operating systems.
NT and SCO UNIX Operating Systems	Returns meaningful data only under NT and SCO UNIX.
NT and Netware Operating Systems	Returns meaningful data only under NT and Netware.
NT Operating System	Returns meaningful data only under NT.

In this appendix, SNMP syntax types and their extensions are presented in italic type; for example, *DisplayString*.

SNMP Syntax Extensions

The DIGITAL Server System MIB extends the SNMP syntax type *INTEGER* with the following textual conventions. (The DIGITAL Server System MIB also uses the SNMP syntax type *DisplayString*.)

- *Kbytes* — an extension to the *INTEGER* syntax type. The possible values for this syntax type are 0 ... 2147483647.
- *MemoryAddress* — an extension to the *OCTET STRING* syntax type.
- *Boolean* an extension to the *INTEGER* syntax type. The possible values are 1 (true) or 2 (false).
- *SystemStatus* — an extension to the *INTEGER* syntax type. Possible numeric values for this syntax type are 1 (unknown), 2 (okay), 3 (warning).
- *ThermUnits* — an extension to the *INTEGER* syntax type. Possible numeric values for this syntax type are as follows:
 - 1 (unknown)
 - 2 (other)
 - 3 (degrees Fahrenheit)
 - 4 (degrees Celsius)
 - 5 (relative temperature)
- *PowerUnits* — an extension to the *INTEGER* syntax type. Possible numeric values for this syntax type are as follows:
 - 1 (unknown)
 - 2 (other)
 - 3 (millivolts DC)
 - 4 (millivolts AC)
 - 5 (volts DC)
 - 6 (volts AC)

The Digital Server System MIB

- 7 (milliamperes DC)
- 8 (milliamperes AC)
- 9 (amperes DC)
- 10 (amperes AC)
- 11 (relative)

- *BusTypes* — an extension to the *INTEGER* syntax type. Possible numeric values for this syntax type are as follows:

- 1 unknown bus type
- 2 other
- 3 system bus
- 4 ISA
- 5 EISA
- 6 MCA
- 7 turbochannel
- 8 PCI
- 9 VME
- 10 new bus
- 11 PCMCIA
- 12 c bus
- 13 MPI
- 14 MPSA
- 15 USB

Base System Configuration

The base system configuration (svrBaseSysDescr) group of the MIB provides basic information about the hardware platform, the operating system, and the firmware installed on the server.

This section defines the MIB variables in this group and provides technical detail about each variable: the hardware platforms and operating systems that support it.

svrSystemFamily

All Platforms/All Operating Systems

The svrSystemFamily variable identifies the family of hardware platforms to which a server is a member. When queried, it returns one of the following numeric values:

- 1 (unknown)
- 2 (other)
- 3 (X86)
- 4 (Alpha)

svrSystemModel

All Platforms/All Operating Systems

The svrSystemModel variable identifies the specific model in the hardware platform family; for example, DIGITAL Server 5XXX.

The syntax for this variable is *DisplayString*.

svrSystemDescr

All Platforms/All Operating Systems

The svrSystemDescr variable provides a general description of the system type.

The syntax for this variable is *DisplayString*.

The Digital Server System MIB

svrSystemBoardFruIndex

Advanced X86-Based Servers/All Operating Systems

The `svrSystemBoardFruIndex` variable defines the index value of the Field Replaceable Unit (FRU) in the FRU table. The FRU table lists the serial number and other asset information about this board. The value is 0 if the index is unknown. The syntax for this variable is *INTEGER*.

svrSystemOCPDisplay

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The `svrSystemOCPDisplay` variable shows the content of the operator control panel (OCP). The value is null if the content is not known. This MIB variable can be written to on some systems. The syntax is *INTEGER*.

This variable does not return useful information on the X16200, an advanced X86-based server.

svrSystemBootedOS

All Platforms/All Operating Systems

The `svrSystemBootedOS` variable identifies the currently booted operating system. The syntax for this variable is *INTEGER*.

- | | |
|------------------|------------------|
| 1 (unknown) | 2 (other) |
| 3 (Windows NT) | 4 (Netware) |
| 5 (SCO UNIX) | 6 (DIGITAL UNIX) |
| 7 (Open VMS) | 8 (Windows) |
| 9 (SCO UNIXware) | |

svrSystemBootedOSVersion

All Platforms/All Operating Systems

This variable contains the version number of the currently booted operating system. The syntax for this variable is *DisplayString*.

svrSystemShutdownReason

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

This variable identifies the possible cause of a system shutdown. The syntax is *DisplayString*. This variable does not return useful information on the X16200, an advanced X86-based server.

SvrSystemRemoteMgrNum

All Platforms/All Operating Systems

If a server is managed remotely, this variable displays the telephone number of the remote manager. The syntax for this variable is *DisplayString*.

svrFirmwareTable

All Operating Systems

The *svrFirmwareTable* stores information about the firmware present on a system. It may include SRM and ARC console firmware; optionally Fail Safe Loader.

On X86-based systems, this table describes the system BIOS and may also include second levels, such as PALcode and BIOS emulator versions, if they are accessible to the software. On Alpha systems, this table includes the MIB version and driver versions.

The syntax for this variable is *SEQUENCE OF SvrFirmwareEntry*.

The Digital Server System MIB

svrFirmwareIndex

All Platforms/All Operating Systems

This variable in the svrFirmwareTable represents an index value, which is unique to the local server, for a selected entry in the table. As defined in the MIB, the syntax for svrFirmwareIndex is *INTEGER*.

svrFirmwareDescr

All Platforms/All Operating Systems

This variable in the svrFirmwareTable describes the firmware as SRM console, ARC console, system BIOS, and so on. The syntax is *DisplayString*.

svrFirmwareRev

All Platforms/All Operating Systems

This variable in the svrFirmwareTable contains the firmware version number. The syntax for svrFirmwareRev is *DisplayString*.

svrFwSymbolTable

NT Operating System

The svrFwSymbolTable contains information about the soft switches and symbols that the firmware maintains. This information is specific to a server and may also be specific to an operating system or to a particular option.

The syntax for svrFwSymbolTable is *SEQUENCE OF SvrFwSymbolEntry*.

svrFwSymbolName

All Platforms/ NT Operating System

This variable contains the display name for the symbol name in the Graphical User Interface (GUI). The syntax is *DisplayString*.

svrFwSymbolValue

All Platforms/NT Operating System

This variable represents the value of the symbol. It is null if the value is none or unknown. The syntax is *OCTET STRING*.

Central Processor Configuration and Utilization

The central processor configuration and utilization (svrProcessors) group of the MIB provides basic information about the CPUs on the server: configuration, polling, utilization statistics, and caching.

This section defines the MIB variables in this group and provides technical detail about each variable: the hardware platforms and operating systems that support it.

svrCpuPollInterval

All Platforms/NT and SCO UNIX Operating Systems

The interval, given in seconds, between queries of the CPU to determine CPU usage. The value 0 (zero) disables polling. Otherwise values must be equal to or greater than the minimum polling interval. (svrCpuMinPollInterval).

The syntax for svrCpuPollInterval is *INTEGER*.

svrCpuMinPollInterval

All Platforms/NT and SCO UNIX Operating Systems

The minimum allowable time between CPU usage queries on this server. The value must be greater than zero. The syntax is *INTEGER*.

The Digital Server System MIB

svrCpuTable

All Operating Systems

The svrCpuTable contains information about the central processors on this server. It contains their base configuration data. The table is an extension of the hrDeviceTable of the Host Resources MIB, RFC1514.

The syntax for svrCpuPollInterval is *SEQUENCE OF SvrCpuEntry*.

svrCpuIndex

All Platforms/All Operating Systems

The svrCpuIndex variable contains an index value for this CPU entry that is unique to the local server. The syntax is *INTEGER*.

svrCpuType

Advanced X86-Based Servers and All Alpha-Based Servers/All Operating Systems

The svrCpuType variable identifies the processor. When queried, it returns one of the following numeric values:

1 (unknown)	2 (other)
3 (i386)	4 (i486)
5 (Pentium)	6 (Pentium Pro)
7 (Alpha 21064)	8 (Alpha 21164)
9 (Alpha 21164 PC)	10 (Alpha 21264)
11 (Pentium II)	12 (K6)

The syntax for svrCpuType is *INTEGER*.

svrCpuManufacturer

Advanced X86-Based Servers and All Alpha-Based Servers/All Operating Systems

The `svrCpuManufacturer` variable identifies the manufacturer of this processor. The syntax is *DisplayString*.

svrCpuRevision

Advanced X86-Based Servers and All Alpha-Based Servers/All Operating Systems

The `svrCpuRevision` variable contains CPU version information presented in processor-specific format.

svrCpuFruIndex

Advanced X86-Based Servers and All Alpha-Based Servers/All Operating Systems

The `svrCpuFruIndex` variable cross-references the index value for this CPU entry in the FRU table (`svrFruTable`). This provides asset information about the component that contains this processor. The value is zero (0) if the asset information is not known. The syntax is *INTEGER*.

svrCpuSpeed

Advanced X86-Based Servers and All Alpha-Based Servers/All Operating Systems

The `svrCpuSpeed` variable contains the speed this CPU expressed in megahertz (MHz). The value is 0 if the CPU speed is not known. The syntax is *INTEGER*.

svrCpuUtilCurrent

All Platforms/NT and Netware Operating Systems

The `svrCpuUtilCurrent` variable shows the percentage of CPU capacity used since the last poll (`svrCpuPollInterval`). The syntax for `svrCpuUtilCurrent` is *INTEGER*.

The Digital Server System MIB

svrCpuAvgNextIndex

All Platforms/NT Operating System

The `svrCpuAvgNextIndex` variable represents the next available index value for setting an interval to track average CPU usage. The syntax is *INTEGER*.

svrCpuHrIndex

All Platforms/All Operating Systems

The `svrCpuHrIndex` variable cross-references the index value of the `hrDeviceEntry` for this CPU in the Host Resources MIB. The value is zero (0) if the index value is not known. The syntax is *INTEGER*.

svrCpuAvgTable

NT Operating System

The `svrCpuAvgTable` contains information about the average utilization of a CPU. The syntax for `svrCpuAvgTable` is *SEQUENCE OF SvrCpuAvgEntry*. Entries in this table:

- Describe the CPU
- Define the interval over which usage is averaged
- Record the average usage measured at the last interval

svrCpuAvgIndex

All Platforms/NT Operating System

The `svrCpuAvgIndex` variable represents an index value for this entry that is unique to the local server. Depending upon your implementation, there may be a maximum value. The syntax is *INTEGER*.

svrCpuAvgInterval

All Platforms/NT Operating System

The `svrCpuAvgInterval` indicates the interval of time used to measure CPU utilization. The syntax is *INTEGER*.

svrCpuAvgStatus

All Platforms/NT Operating System

The svrCpuAvgStatus variable indicates the status of a row in the svrCpuAvgTable. Possible values are

- 1 (underCreation)
- 2 (rowInvalid)
- 3 (rowEnabled)
- 4 (rowDisabled)
- 5 (rowError)

svrCpuAvgPersist

All Platforms/NT Operating System

The svrCpuAvgPersist variable determines whether data is preserved when an agent restarts.

True – indicates data is preserved.

False – indicates data is not retained.

The syntax for svrCpuAvgPersist is *Boolean*.

svrCpuAvgValue

All Platforms/NT Operating System

The svrCpuAvgValue variable indicates the average load on the CPU during the last measured interval. The syntax for svrCpuAvgValue is *INTEGER*.

svrCpuCacheTable

NT Operating System

The svrCpuCacheTable describes the cache memory that each processor on a server uses. The syntax for is *SEQUENCE OF SvrCpuCacheEntry*.

svrCpuCacheIndex

Advanced X86-Based Servers and All Alpha-Based Servers/NT Operating System

The svrCpuCacheIndex variable represents an index value for this entry that is unique to the local server. The syntax is *INTEGER*.

svrCpuCacheLevel

Advanced X86-Based Servers and All Alpha-Based Servers/NT Operating System

The svrCpuCacheLevel variable indicates the level of cache that this entry uses. When queried, it returns one of the following numeric values:

- 1 (unknown)
- 2 (other)
- 3 (primary)
- 4 (secondary)
- 5 (tertiary)

The syntax for svrCpuCacheLevel is *INTEGER*.

svrCpuCacheType

Advanced X86-Based Servers and All Alpha-Based Servers/NT Operating System

The `svrCpuCacheType` variable defines the cache that this entry uses; that is, whether it is internal or external to the CPU chip. When queried, it returns one of the following numeric values:

- 1 (internal)
- 2 (external)
- 3 (internal instruction)
- 4 (internal data)

The syntax for `svrCpuCacheType` is *INTEGER*.

svrCpuCacheSize

Advanced X86-Based Servers and All Alpha-Based Servers/NT Operating System

The `svrCpuCacheSize` variable indicates the amount of cache that this entry uses. The syntax for `svrCpuCacheSize` is *KBytes*.

svrCpuCacheSpeed

Not Implemented

The `svrCpuCacheSpeed` variable defines the speed at which data stored in cache can be retrieved. The speed is expressed in nanoseconds (ns).

The Digital Server System MIB

SvrCpuCacheStatus

Advanced X86-Based Servers and All Alpha-Based Servers/NT Operating System

The `svrCpuCacheStatus` variable contains the current status of the cache. When queried, it returns one of the following numeric values:

- 1 (unknown)
- 2 (other)
- 3 (enabled)
- 4 (disabled)

This variable does not return useful information on the X16200, an advanced X86-based server. The syntax for `svrCpuCacheStatus` is *INTEGER*.

System Memory

The system memory (`svrMemory`) group of the MIB provides basic information about memory configuration.

This section defines the MIB variables in this group and provides technical detail about each variable: the hardware platforms and operating systems.

svrPhysicalMemorySize

All Platforms/All Operating Systems

The `svrPhysicalMemorySize` variable shows the total amount of physical memory residing on the server that is recognized by the operating system.

svrPhysicalMemoryFree

All Platforms/All Operating Systems

The `svrPhysicalMemoryFree` variable defines the amount of physical memory that is available. The syntax for `svrPhysicalMemoryFree` is *KBytes*.

svrPagingMemorySize

All Platforms/NT and SCO UNIX Operating Systems

The `svrPagingMemorySize` variable indicates the total virtual memory that the operating system can make available through paging. The syntax for `svrPagingMemorySize` is *KBytes*.

svrPagingMemoryFree

All Platforms/NT and SCO UNIX Operating Systems

The `svrPagingMemoryFree` variable represents the amount of virtual memory that is currently available to a process. The syntax for `svrPagingMemoryFree` is *KBytes*.

svrMemComponentTable

All Operating Systems

The `svrMemComponentTable` contains information about how memory components are configured in hardware.

The syntax for `svrMemComponentTable` is *SEQUENCE OF SvrMemComponentEntry*.

svrMemIndex

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The `svrMemIndex` variable contains an index value for this entry that is unique to the local server. The syntax is *INTEGER*.

svrMemType

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The `svrMemType` variable indicates the type of memory that this component represents. When queried, it returns one of the following numeric values:

- 1 (unknown)
- 2 (other)
- 3 (systemMemory)
- 4 (shadowMemory)
- 5 (videoMemory)
- 6 (flashMemory)
- 7 (nvram)
- 8 (expansion RAM)
- 9 (expansion ROM)

The syntax for `svrMemType` is *INTEGER*.

svrMemSize

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The `svrMemSize` variable indicates the length of memory range, expressed in KB.

The syntax for `svrMemSize` is *KBytes* — an extension to the SNMP *INTEGER* syntax type.

svrMemStartAddress

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The `svrMemStartAddress` variable contains the 64-bit starting address of the memory that is contained on or controlled by this component. The syntax for `svrMemStartAddress` is *MemoryAddress*.

svrMemPhysLocation

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The `svrMemPhysLocation` variable describes the location of physical memory components. When queried, it returns one of the following numeric values:

- 1 (unknown)
- 2 (other)
- 3 (system board)
- 4 (memory board)
- 5 (processor board)

The syntax for `svrMemPhysLocation` is *INTEGER*.

svrMemEdcType

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The `svrMemEdcType` variable describes the type of error detection and correction that is provided for memory contained on or controlled this component detects and corrects errors. When queried, it returns one of the following numeric values:

- 1 (unknown)
- 2 (other)
- 3 (none)
- 4 (parity)
- 5 (single-bit error checking and correction)
- 6 (multi-bit ECC)

The syntax for `svrMemEdcType` is *INTEGER*.

svrMemElementSlots

Advanced X86-Based and All Alpha-Based Servers/All Operating Systems

The `svrMemElementSlots` variable shows the total number of slots that are available for removable memory modules (that is, SIMMs and DIMMs). The value is 0 (zero) if the number of slots is unknown. The syntax is *INTEGER*.

svrMemElementSlotsUsed

Advanced X86-Based Servers/All Operating Systems

The `svrMemElementSlotsUsed` variable shows the number of slots that are occupied by removable memory modules. The value is 0 (zero) if the number is unknown. The syntax for `svrMemElementSlotsUsed` is *INTEGER*.

svrMemInterleafFactor

Advanced X86-Based Servers/All Operating Systems

The `svrMemInterleafFactor` variable defines the interleaving factor that a memory component uses. The value is 0 (zero) if the factor is unknown; the value is 1 if there is not an interleaving factor; the value is 2 if the factor is two-way.

- This variable does not return useful information on the X16200, an advanced X86-based server.

The syntax for `svrMemInterleafFactor` is *INTEGER*.

svrMemInterleafElement

Advanced X86-Based Servers/All Operating Systems

The `svrMemInterleafElement` variable defines which element this is in interleaving. The value is 0 (zero) if the element is not interleaved.

This variable does not return useful information on the X16200, an advanced X86-based server.

The syntax for `svrMemInterleafElement` is *INTEGER*.

svrMemFruIndex

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The `svrMemFruIndex` variable cross-references the index value of the FRU entry in the FRU table in which this memory is represented. The value is 0 (zero) if the index is not known.

This variable does not return useful information on these advanced Alpha servers:

- DIGITAL Server 7000 Alpha, or AlphaServer 4000 and 4100
- AlphaServer 1000 and 1000A

The syntax for `svrMemFruIndex` is *INTEGER*.

svrMemElementTable

All Operating Systems

The svrMemElementTable extends the server memory component table (svrMemComponentTable). It contains information that describes actual memory hardware elements, such as a SIMM or a DIMM. If the memory component cannot be decomposed into elements, a generic element may be listed in the table.

The syntax for svrMemElementTable is *SEQUENCE OF SvrMemElementEntry*.

svrMemElementIndex

Advanced X86-Based and All Alpha-Based Servers/All Operating Systems

The svrMemElementIndex contains an index value for an entry that is unique to the local server. The syntax for svrMemElementIndex is *INTEGER*.

svrMemElementType

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The svrMemElementType variable defines the type of a piece of memory. When queried, it returns one of the following numeric values:

- 1 (unknown)
- 2 (other)
- 3 (non-removable)
- 4 (SIMM)
- 5 (DIMM)

This variable does not return useful information on these advanced Alpha servers: DIGITAL Server 7000 Alpha, or AlphaServer 4000 and 4100.

The syntax for svrMemElementType is *INTEGER*.

svrMemElementSlotNo

Advanced X86-Based Servers/All Operating Systems

The `svrMemElementSlotNo` variable defines where a removable memory module resides. The value is 0 (zero) if the slot number is unknown. The syntax for `svrMemElementSlotNo` is *INTEGER*.

svrMemElementWidth

Advanced X86-Based Servers/All Operating Systems

The `svrMemElementWidth` variable defines the width of a memory module. The width is expressed in bytes. The value is 0 (zero) if the width is unknown. The syntax is *INTEGER*.

svrMemElementDepth

Advanced X86-Based Servers/All Operating Systems

The `svrMemElementDepth` variable defines the width of a memory module. The value is 0 (zero) if the width is unknown. This variable is implemented on these X86-based servers:

The syntax for `svrMemElementDepth` is *Kbytes*.

svrMemElementSpeed

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The `svrMemElementSpeed` variable describes the speed of the memory module, as used. The value is expressed in nanoseconds and may be less than the maximum possible speed. The value is 0 (zero) if the speed is unknown.

This variable does not return useful information on the X16200, an advanced X86-based server, or on the AlphaServer 1000 and 1000A.

The syntax for `svrMemElementSpeed` is *INTEGER*.

The Digital Server System MIB

svrMemElementStatus

Advanced X86-Based Servers/All Operating Systems

The `svrMemElementStatus` variable contains the current status of the memory module.

The syntax for `svrMemElementSpeed` is *SystemStatus*.

System Buses and Slots

All Platforms/All Operating Systems

The system buses and slots (`svrBuses`) group of the MIB provides basic information about EISA and PCI buses.

svrBusCount

All Platforms/All Operating Systems

The `svrBusCount` variable contains the number of buses that the operating system sees. The syntax for `svrBusCount` is *INTEGER*.

SvrBusTable

NT Operating System

The server bus table (`svrBusTable`) contains information about each bus that the operating system sees. The syntax for `svrBusTable` is *SEQUENCE OF SvrBusEntry*.

svrBusIndex

All Platforms/NT Operating System

The `svrBusIndex` variable contains an index value that is unique to the local server.

svrBusType

All Platforms/ NT Operating System

The `svrBusType` variable describes the type of bus an entry represents. The syntax for `svrBusType` is *BusTypes* — an extension to the SNMP *INTEGER* syntax type.

svrBusNumber

All Platforms/ NT Operating System

The `svrBusNumber` variable indicates the number of the bus. The value is unique among buses of the same type. The syntax for `svrBusNumber` is *INTEGER*.

SvrBusSlotCount

All Platforms/ NT Operating System

The `svrBusSlotCount` variable shows the number of physical slots on a bus. The value is zero (0) if the number is unknown. The syntax for `svrBusSlotCount` is *INTEGER*.

svrLogicalSlotTable

NT Operating System

The logical slot table (`svrLogicalSlotTable`) contains information about logical bus slots on a server. It is indexed by `busIndex` variable and slot number. A server may have more logical slots than physical slots. An entry in logical slot table does not have to be associated with a physical slot. The syntax for `svrLogicalSlotTable` is *SEQUENCE OF SvrLogicalSlotEntry*.

svrLogicalSlotNumber

All Platforms/ NT Operating System

The `svrLogicalSlotNumber` variable contains a unique, logical slot number for each bus. The syntax for `svrLogicalSlotNumber` is *INTEGER*.

The Digital Server System MIB

svrLogicalSlotDescr

All Platforms/ NT Operating System

The svrLogicalSlotDescr variable contains a description of the device which can be derived from vendor's device identifier or assigned by a network management station. The syntax for svrLogicalSlotDescr is *DisplayString*.

svrLogicalSlotDeviceID

All Platforms/ NT Operating System

The svrLogicalSlotDeviceID variable contains the vendor's device identifier. The syntax for svrLogicalSlotDeviceID is *INTEGER*.

svrLogicalSlotVendor

All Platforms/ NT Operating System

The svrLogicalSlotVendor variable displays the name of the device manufacturer. The value is null if the vendor is unknown. The syntax for svrLogicalSlotVendor is *DisplayString*.

svrLogicalSlotRevision

All Platforms/ NT Operating System

The svrLogicalSlotRevision variable contains vendor-supplied major and minor revision numbers for the device in a slot. The syntax for svrLogicalSlotRevision is *DisplayString*.

svrLogicalSlotFnCount

All Platforms/ NT Operating System

The svrLogicalSlotFnCount variable defines the number of functions available on a multi-function device. For PCI buses, the value is usually 1 (one). The syntax for svrLogicalSlotFnCount is *INTEGER*.

svrSlotFunctionTable

NT Operating System

The slot function table extends the logical slot table (svrLogicalSlotTable). It describes the functions of the device. Frequently, there is only one function. An entry in the table describes how the device in a logical slot functions. The syntax for svrSlotFunctionTable is *SEQUENCE OF SvrSlotFunctionEntry*.

svrSlotFnIndex

All Platforms/ NT Operating System

The svrSlotFnIndex variable provides an index value for an entry on the local server. The syntax for svrSlotFnIndex is *INTEGER*.

svrSlotFnDevType

All Platforms/ NT Operating System

The svrSlotFnDevType variable determines the device type that this function represents. The value is derived from the device identifier. The syntax for svrSlotFnDevType is *DisplayString*.

svrSlotFnRevision

All Platforms/ NT Operating System

The svrSlotFnRevision variable contains the revision number for the device function. The syntax for svrSlotFnRevision is *DisplayString*.

Peripheral Devices

The peripheral devices (svrDevices) group extends the information available in the host hrDevices group.

Except where noted, the tables and variables in this group are used on all X86-based and Alpha servers, but only on the NT operating system.

The peripheral devices group also contains these subgroups:

- Console Keyboard
- Console Display
- Console Pointing Device

svrDeviceCount

All Platforms/NT Operating System

The svrDeviceCount variable contains the number of entries in the server device table (svrDeviceTable). The syntax for svrDeviceCount is *INTEGER*.

svrDeviceTable

NT Operating System

The server device table (svrDeviceTable) contains configuration information about each device or adapter that the host operating system recognizes. An entry in the table represents one device or adapter. The syntax for svrDeviceTable is *SEQUENCE OF SvrDeviceEntry*.

svrDevIndex

All Platforms/NT Operating System

The svrDevIndex variable is an integer value that is unique to the local server that identifies a device or adapter. The syntax for svrDevIndex is *INTEGER*.

svrDevDescr

All Platforms/NT Operating System

The `svrDevDescr` variable contains a character string that describes the device or adapter. The syntax for `svrDevIndex` is *DisplayString*.

svrDevBusInterfaceType

All Platforms/NT Operating System

The `svrDevBusInterfaceType` variable defines the type of bus that a peripheral device or adapter uses to connect to one of the CPUs on a server. The syntax for `svrDevBusInterfaceType` is *BusTypes*.

svrDevBusNumber

All Platforms/NT Operating System

The `svrDevBusNumber` variable contains the index value assigned to the bus. A value of 0 (zero) indicates an internal bus. The syntax for `svrDevBusNumber` is *INTEGER*.

svrDevSlotNumber

All Platforms/NT Operating System

The `svrDevSlotNumber` variable contains the logical slot number of the peripheral device or adapter. The value is 0 (zero) if the slot number is unknown. The syntax for `svrDevSlotNumber` is *INTEGER*.

svrDevFruIndex

Advanced X86-Based and Advanced Alpha-Based Servers/NT Operating System

The `svrDevFruIndex` variable cross-references the entry in the FRU table for this device. The value is 0 (zero) if the index for the FRU table is unknown. The syntax is *INTEGER*.

This variable does not return useful information AlphaServer 1000 and 1000A

The Digital Server System MIB

svrDevCPUAffinity

All Platforms/NT Operating System

The `svrDevCPUAffinity` variable holds a bit map that describes the processors to which interrupts are dispatched. The syntax is *INTEGER*.

svrDevHrIndex

All Platforms/NT Operating System

The `svrDevHrIndex` variable points to the entry for this device in the Host Resources MIB device table. The value is 0 (zero) if the index for the entry in Host Resources MIB device table is unknown. The syntax is *INTEGER*.

svrDevInterruptTable

NT Operating System

The server device interrupt table (`svrDevInterruptTable`) extends the server device table (`svrDeviceTable`). It lists the interrupt configurations for a peripheral device or adapter. An entry in the table describes one interrupt for a peripheral device or adapter. The syntax is *SEQUENCE OF SvrDevInterruptEntry*.

svrDevIntIndex

All Platforms/NT Operating System

The `svrDevIntIndex` variable represents an implementation-specific index value for an entry. The syntax is *INTEGER*.

svrDevIntLevel

All Platforms/NT Operating System

The `svrDevIntLevel` variable contains the interrupt level for which a device is configured. The interrupt level is governed by the operating system. In Windows NT the configuration is IRQL. The syntax is *INTEGER*.

svrDevIntVector

All Platforms/NT Operating System

The `svrDevIntVector` variable defines the interrupt vector for the device. The syntax is *INTEGER*.

svrDevIntShared

All Platforms/NT Operating System

The `svrDevIntShared` variable defines whether the device shares an interrupt with other devices. The syntax is *Boolean*.

svrDevIntTrigger

All Platforms/NT Operating System

The `svrDevIntTrigger` variable determines how an interrupt is triggered. An interrupt is either a level-sensitive event or an edge-triggered event. A level-sensitive event triggers an interrupt condition that persists as long as the input signal of the event. An edge-triggered interrupt occurs with a low-to-high or high-to-low transition of the input signal. A value of 1 (one) indicates a level-sensitive event; a value of 2 (two) an edge-triggered interrupt. The syntax for `svrDevIntTrigger` is *INTEGER*.

svrDevMemTable

NT Operating System

The server device memory table (`svrDevMemTable`) extends the server device table (`svrDeviceTable`). It contains information about the way memory is configured on a peripheral device or adapter. The syntax is *SEQUENCE OF SvrDevMemTable*.

svrDevMemIndex

All Platforms/NT Operating System

The `svrDevMemIndex` variable represents an implementation-specific index value for this entry. The syntax is *INTEGER*.

The Digital Server System MIB

svrDevMemAddress

All Platforms/NT Operating System

The `svrDevMemAddress` variable represents the physical address at the start of the memory range. The syntax is *MemoryAddress*.

svrDevMemLength

All Platforms/NT Operating System

The `svrDevMemLength` variable represents the size of the memory range. The value is expressed in bytes. The syntax is *INTEGER*.

svrDevMemMapping

All Platforms/NT Operating System

The `svrDevMemMapping` variable shows whether the entry represents space that is memory mapped or I/O mapped. When queried, this variable returns one of these numeric values:

- 1 (unknown)
- 2 (other)
- 3 (memory mapped)
- 4 (I/O space mapped)

The syntax for `svrDevMemMapping` is *INTEGER*.

svrDevDmaTable

NT Operating System

The server device DMA table lists the direct memory access configurations for a peripheral device or adapter. The syntax is *SEQUENCE OF SvrDevDmaEntry*.

svrDevDmaIndex

All Platforms/NT Operating System

The `svrDevDmaIndex` variable represents an Implementation-specific index value for the entry. The syntax is *INTEGER*.

svrDevDmaChannel

All Platforms/NT Operating System

The svrDevDma channel indicates which channel on DMA controller the device uses. The syntax is *INTEGER*.

svrSerialPortCount

All Platforms/NT Operating System

The svrSerialPortCount variable maintains the number of entries in the server serial port table (svrSerialPortTable). The syntax is *INTEGER*.

svrSerialPortTable

NT Operating System

The server serial port table (svrSerialPortTable) contains information about the serial ports on a server. The syntax is *SEQUENCE OF SvrSerialPortEntry*.

svrSerialIndex

All Platforms/NT Operating System

The svrSerialIndex variable represents a value that is unique to the local server. The syntax is *INTEGER*.

svrSerialPortDescr

All Platforms/NT Operating System

The svrSerialPortDescr contains a character string that describes this serial port. The syntax is *DisplayString*.

svrSerialHrIndex

All Platforms/NT Operating System

The svrSerialHrIndex cross-references the hrDeviceIndex value of the hrDeviceEntry in the Host Resources MIB that describes this. The value is 0 (zero) if the index value is unknown. The syntax is *INTEGER*.

svrParallelPortCount

All Platforms/NT Operating System

The svrParallelPortCount variable maintains the number of entries in the server parallel port table (svrParallelPortTable). The syntax is *INTEGER*.

svrParallelPortTable

NT Operating System

The server parallel port table contains information about the parallel ports on a device. The syntax is *SEQUENCE OF SvrParallelPortEntry*.

svrParallelIndex

All Platforms/NT Operating System

The svrParallelIndex represents an index value that is unique to the local server. The syntax is *INTEGER*.

svrParallelPortDescr

All Platforms/NT Operating System

The svrParallelPortDescr variable contains a character string that describes this parallel port. The syntax is *DisplayString*.

svrParallelHrIndex

All Platforms/NT Operating System

The svrParallelHrIndex variable cross-references the index value of hrDeviceEntry for this port in the Host Resources MIB. The value is 0 (zero) if the index value is unknown. The syntax is *INTEGER*.

svrConsoleKeyboard

In the DIGITAL server system MIB, `svrConsoleKeyboard` represents a subgrouping under peripheral devices (`svrDevices`).

svrKeybdHrIndex

All Platforms/NT Operating System

The `svrKeybdHrIndex` variable cross-references the index value of the `hrDeviceEntry` in the Host Resources MIB. The value is 0 (zero) if the index value is unknown. The syntax is *INTEGER*.

svrKeybdDescr

All Platforms/NT Operating System

The `svrKeybdDescr` variable contains a character string describing the keyboard. The description is provided by the operating system. The syntax is *DisplayString*.

svrConsoleDisplay

In the DIGITAL server system MIB, `svrConsoleDisplay` represents a subgrouping under peripheral devices (`svrDevices`).

svrVideoHrIndex

All Platforms/NT Operating System

The `svrVideoHrIndex` variable cross-references the index value in the host devices table. The value is 0 (zero) if the index value is unknown. The syntax is *INTEGER*.

svrVideoDescr

All Platforms/NT Operating System

The `svrVideoDescr` variable contains a description of the console display. The description is provided by the operating system. The syntax is *DisplayString*.

svrVideoXRes

All Platforms/NT Operating System

The svrVideoXRes variable describes console resolution on the X-axis. The syntax is *INTEGER*.

svrVideoYRes

All Platforms/NT Operating System

The svrVideoYRes variable describes console resolution on the Y-axis. The syntax is *INTEGER*.

svrVideoNumColor

All Platforms/NT Operating System

The svrVideoNumColor variable describes the number of colors that this console displays. The value is converted, if necessary, from bits-per-pixel. The syntax is *INTEGER*.

svrVideoRefreshRate

All Platforms/NT Operating System

The svrVideoRefreshRate variable maintains the current refresh rate of the console screen. The value is expressed in Hz. The value is 0 (zero) if the refresh rate is unavailable. The value is 1 (one) if it is the hardware default rate. The syntax is *INTEGER*.

svrVideoScanMode

All Platforms/NT Operating System

The svrVideoScanMode variable defines whether the video display is interlaced. Possible values are

- 1 (unknown)
- 2 (interlaced)
- 3 (nonInterlaced)

The syntax for svrVideoScanMode is *INTEGER*.

svrVideoMemory

All Platforms/NT Operating System

The `svrVideoMemory` variable represents the amount of video memory available or in use. The syntax is *KBytes*.

svrConsolePointDevice

In the DIGITAL server system MIB, `svrConsolePointDevice` represents a subgrouping under peripheral devices (`svrDevices`). This subgrouping usually describes a mouse.

svrPointingHrIndex

All Platforms/NT Operating System

The `svrPointingHrIndex` variable cross-references the index value in the host devices table. The value is 0 (zero) if the index value is unknown. The syntax is *INTEGER*.

svrPointingDescr

All Platforms/NT Operating System

The `svrPointingDescr` variable contains a description of the pointing device that is supplied by the operating system. The syntax is *DisplayString*.

svrNumButtons

All Platforms/NT Operating System

The `svrNumButtons` variable shows the number of buttons on a mouse. The syntax is *INTEGER*.

Physical Configuration

The physical configuration (svrPhysicalConfiguration) group of the MIB provides basic information about the physical characteristics of a server.

This section defines the MIB variables in the group and provides technical detail about each variable: the hardware platforms and operating systems that support it.

svrChassisType

All Platforms/NT and Netware Operating Systems

The svrChassisType variable describes the server's chassis type. When queried, it returns one of the following numeric values:

- 1 (unknown)
- 2 (other)
- 3 (desktop)
- 4 (tower)
- 5 (miniTower)
- 6 (rackMount)
- 7 (laptop)

The syntax for svrChassisType is *INTEGER*.

svrChassisFruIndex

Advanced X86-Based and Advanced Alpha-Based Servers/ NT and Netware Operating Systems

The svrChassisFruIndex variable provides the index value of the chassis FRU descriptor in the server FRU table (svrFruTable). The value is 0 (zero) if the index value is unknown.

This variable does not return useful information on the AlphaServer 1000 and 1000A.

The syntax for svrChassisFruIndex is *INTEGER*.

svrFruTable

All Operating Systems

The server FRU table (svrFruTable) maintains a list of field replaceable units (FRU) on a server. It provides inventory and asset control information. This table is referenced from logical hardware descriptions elsewhere in the server system MIB.

Entries in this table do not return useful information on the AlphaServer 1000 and 1000A.

The syntax for svrFruTable is *SEQUENCE OF SvrFruEntry*.

svrFruIndex

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The svrFruIndex variable represents an index value that is unique to the local server. The syntax is *INTEGER*.

svrFruType

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The svrFruType variable describes the general category of FRU to which this entry belongs. When queried, it returns one of the following numeric values:

- 1 (unknown)
- 2 (other)
- 3 (motherBoard)
- 4 (processor)
- 5 (memoryCard)
- 6 (memoryModule)
- 7 (peripheralDevice)
- 8 (systemBusBridge)

The Digital Server System MIB

9 (powerSupply)

10 (chassis)

11 (fan)

12 (ioCard)

The syntax for svrFruType is *INTEGER*.

svrFruDescr

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The svrFruDescr variable describes the FRU type in more, if the description is available. The syntax is *DisplayString*.

svrFruVendor

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The svrFruVendor variable provides the FRU manufacturer's name or ID. The syntax is *DisplayString*.

svrFruPartNumber

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The svrFruPartNumber provides the part number you can use to order the unit. The syntax is *DisplayString*.

svrFruRevision

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The svrFruRevision variable gives the version number of this unit. If artwork level is available, it will appear as 'Artwork: XXX' following FRU version portion. The syntax is *DisplayString*.

SvrFruFirmwareRevision

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The `svrFruFirmwareRevision` variable provides the firmware revision number. This variable is not supported on the XL6200 X86-based server. The syntax is *DisplayString*

svrFruSerialNumber

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The `svrFruSerialNumber` variable provides the serial number. The syntax is *DisplayString*

svrFruAssetNo

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The `svrFruAssetNo` variable provides the asset number for this unit. This variable is not implemented on the XL6200 X86-based server or on any Alpha servers. The syntax is *DisplayString*

svrFruClass

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The `svrFruClass` variable represents the status category of the unit. When queried, it returns one of the following numeric values:

- 1 (unknown)
- 2 (other)
- 3 (current board in slot)
- 4 (prior board in slot)

The Digital Server System MIB

- 5 (parent board)
- 6 (prior parent board)
- 7 (prior parent system)

The syntax for `svrFruClass` is *INTEGER*.

Environment

The environment group (`svrEnvironment`) group describes components that affect environmental conditions on a server. The group is organized into these subgroups:

- Thermal system (`svrThermalSystem`)
- Cooling system (`svrCoolingSystem`)
- Power system (`svrPower System`)

Except where noted, the tables and variables in the environment group are supported on all operating systems:

svrThermalSensorCount

All Platforms/All Operating Systems

The `svrThermalSensorCount` variable is a member of the `svrThermalSystem` subgroup. It maintains a count of the number of thermal sensors that are present and readable on a server. If there are no thermal sensors on a server, the value of the variable is 0 (zero). The syntax is *INTEGER*.

svrThermalSensorTable

All Operating Systems

The thermal sensor table (svrThermalSensorTable) lists the thermal sensors on a server. A thermal sensor may be either a temperature gauge or temperature status indicator.

In addition to the advanced Alpha servers listed at the beginning of this appendix, most entries in this table are implemented on Alpha servers with embedded remote management.

The syntax for svrThermalSensorTable is *SEQUENCE OF SvrThermalSensorEntry*.

svrThSensorIndex

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The svrThSensorIndex variable contains an index value that is unique to a local server. The syntax is *INTEGER*.

svrThSensorLocation

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The svrThSensorLocation variable identifies where the gauge or status indicator is located. The syntax is *DisplayString*.

SvrThSensorReading

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The svrThSensorReading contains the current reading of the temperature gauge or status indicator. The unit in which the reading is expressed is defined by the svrThSensorReadingUnits variable. This variable is not implemented on these Alpha-servers: AlphaServer 2100, 2100A, 1000 and 1000A. The syntax is *INTEGER*.

svrThSensorReadingUnits

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The `svrThSensorReadingUnits` variable defines whether a sensor reading is expressed in degrees Fahrenheit or Celsius, or as a relative value such as High or Low. The syntax is *ThermUnits*.

svrThSensorLowThresh

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The `svrThSensorLowThresh` variable sets the default value for the low temperature threshold. The unit in which the value is expressed is defined by the `svrThSensorThreshUnits` variable. The syntax is *INTEGER*.

svrThSensorHighThresh

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The `svrThSensorHighThresh` variable set the default value for the high temperature threshold. The unit in which the value is expressed is defined by the `svrThSensorThreshUnits` variable. The syntax is *INTEGER*.

svrThSensorShutSoonThresh

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The `svrThSensorShutSoonThresh` variable sets the default value for the Shutdown Soon temperature threshold. The unit in which the value is expressed is defined by the `svrThSensorThreshUnits` variable. The syntax is *INTEGER*.

svrThSensorShutNowThresh

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The `svrThSensorShutNowThresh` variable sets the default value for the Shutdown Now temperature threshold. The unit in which the value is expressed is defined by the `svrThSensorThreshUnits` variable. The syntax is *INTEGER*.

svrThSensorThreshUnits

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The `svrThSensorThreshUnits` variable represents how all the threshold values should be interpreted; that is whether the values are expressed in degrees Fahrenheit or Celsius, or as a relative value such as High or Low.

If the server does not support default threshold, the value of this variable is set at unknown.

The syntax is *ThermUnits*.

svrThSensorStatus

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The `svrThSensorStatus` variable indicates the status of the sensor. The syntax for `svrThSensorStatus` is *INTEGER*. When queried, it returns one of the following numeric values:

- 1 (unknown)
- 2 (other)
- 3 (low)
- 4 (low warning)
- 5 (status okay)
- 6 (high warning)
- 7 (high)

The Digital Server System MIB

svrThSensorFruIndex

Advanced X86-Based Servers/All Operating Systems

The `svrThSensorFruIndex` cross-references the index value in the FRU table in which this sensor is represented. If the index value is unknown, the value of the variable is 0 (zero). This variable is not implemented on Alpha servers. The syntax is *INTEGER*.

svrFanCount

All Platforms/All Operating Systems

The `svrFanCount` variable is a member of the `svrCoolingSystem` subgroup. It maintains a count of the number of fans whose state can be detected. If there is no fan on the server, the value of this variable is 0 (zero). The syntax is *INTEGER*.

svrFanTable

All Operating Systems

The server fan table (`svrFanTable`) lists the fans on a server. The syntax is *SEQUENCE OF SvrFanEntry*.

svrFanIndex

Advanced X86-Based and All Alpha-Based Servers/All Operating Systems

The `svrFanIndex` variable contains an index value that is unique to a local server. The syntax is *INTEGER*.

svrFanLocation

Advanced X86-Based and All Alpha-Based Servers/All Operating Systems

The `svrFanLocation` variable identifies the location of the fan. The syntax is *DisplayString*.

svrFanStatus

Advanced X86-Based and All Alpha-Based Servers/All Operating Systems

The `svrFanStatus` variable indicates the current status of a fan. When queried, it returns one of the following numeric values:

- 1 (unknown)
- 2 (running)
- 3 (backup)
- 4 (failed)

This variable is not implemented on the X86-based X16200 server. On Alpha-based servers it is implemented on the AlphaServer 2100 and 2100A, the DIGITAL Server Alpha 7000, the AlphaServer 4000 and 4100, the AlphaServer 1000 and 1000A, and the AlphaServer 800.

The syntax for `svrFanStatus` is *INTEGER*.

svrFanBackup

Advanced X86-Based and All Alpha-Based Servers/All Operating Systems

The `svrFanBackup` variable contains the index value of the fan for which this fan is the backup. This variable is not implemented on the X86-based X16200 server. The syntax is *INTEGER*.

The Digital Server System MIB

svrFanFruIndex

Advanced X86-Based and All Alpha-Based Servers/All Operating Systems

The `svrFanFruIndex` variable cross-references the index value in the FRU table in which this fan is represented. If the index value is unknown, the value of the variable is 0 (zero). The syntax for is *INTEGER*.

This variable is only available the X86-based NT servers listed in the description of the server fan table (`svrFanTable`). On Alpha-based servers it is only implemented on the AlphaServer 2100 and 2100A, the AlphaServer 4000 and 4100, and the DIGITAL Server 7000 Alpha.

svrPowerRedunEnable

All Alpha Servers/NT Operating System

The `svrPowerRedunEnable` variable is a member of the `svrPowerSystem` subgroup. It indicates whether redundancy is enable in this power supply subsystem. It is supported only on Alpha servers. The syntax is *Boolean*.

svrPowerSensorCount

All Platforms/All Operating System

The `svrPowerSensorCount` variable maintains a count of the number of detectable voltage sensors on the server.

If a power sensor is not present on a server, the value of this variable is 0 (zero).

The syntax for `svrPowerSensorCount` is *INTEGER*.

svrPowerSensorTable

All Operating Systems

The server power sensor table (svrPowerSensorTable) lists information about power sensors on the server. A power sensor may detect either voltage or current.

The syntax for svrPowerSensorTable is *SEQUENCE OF SvrPowerSensorEntry*.

svrPowerSensorIndex

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The svrPowerSensorIndex variables contains an index value that is unique to a local server. The syntax is *INTEGER*.

svrPowerSensorLocation

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The svrPowerSensorLocation variable describes the physical location of the sensor on the server. The syntax is *DisplayString*.

svrPowerSensorRating

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The svrPowerSensorRating variable defines the nominal rating for a sensor. The unit in which the rating is expressed is determined by the svrPowerSensorReadingUnits variable. The syntax is *INTEGER*.

svrPowerSensorReading

Advanced X86-Based Servers/All Operating Systems

The svrPowerSensorReading variable represents the state of voltage or current that this sensor has detected. The unit in which the rating is expressed is determined by the svrPowerSensorReadingUnits variable.. The syntax is *INTEGER*.

svrPowerSensorReadingUnits

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The `svrPowerSensorReadingUnits` variable determines the unit of measure that the sensor uses. Possible values include, volts AC, DC, amperes AC, DC. The syntax is *PowerUnits*.

svrPowerSensorNeedPwrThresh

Advanced X86-Based Servers/All Operating Systems

The `svrPowerSensorNeedPwrThresh` sets the default value for the need more power voltage threshold. The unit in which this is expressed is determined by `svrPowerSensorThreshUnits` variable. The syntax is *INTEGER*.

svrPowerSensorLowThresh

Advanced X86-Based Servers/All Operating Systems

The `svrPowerSensorLowThresh` sets the default value for the low voltage threshold. The unit in which this is expressed is determined by `svrPowerSensorThreshUnits` variable. The syntax is *INTEGER*.

svrPowerSensorHighThresh

Advanced X86-Based Servers/All Operating Systems

The `svrPowerSensorHighThresh` sets the default value for the high voltage threshold. The unit in which this is expressed is determined by `svrPowerSensorThreshUnits` variable. The syntax is *INTEGER*.

svrPowerSensorShutNowThresh

Advanced X86-Based Servers/All Operating Systems

The `svrPowerSensorShutNowThresh` sets the default value for the shut down now voltage threshold. The unit in which this is expressed is determined by `svrPowerSensorThreshUnits` variable. The syntax is *INTEGER*.

svrPowerSensorThreshUnits

Advanced X86-Based Servers/All Operating Systems

The `svrPowerSensorThreshUnits` variable represents the unit of measure used in interpreting all the threshold values. The syntax is *PowerUnits*.

svrPowerSensorStatus

Advanced X86-Based and Advanced Alpha-Based Servers/ All Operating Systems

The `svrPowerSensorStatus` variable indicates the sensor's status. When queried, it returns one of the following numeric values:

- 1 (unknown)
- 2 (other)
- 3 (low)
- 4 (low warning)
- 5 (status okay)
- 6 (high warning)
- 7 (high)

svrPowerSensorFruIndex

Advanced X86-Based Servers/All Operating Systems

The `svrPowerSensorFruIndex` cross-references the index value in the FRU table in which this sensor is represented. This variable is not available on Alpha platforms. The syntax is *INTEGER*.

svrPowerSupplyCount

All Platforms/All Operating Systems

The `svrPowerSupplyCount` variable maintains a count of the number of detectable power supplies on a server. The syntax is *INTEGER*.

svrPowerSupplyTable

All Operating Systems

The server power supply table (svrPowerSupplyTable) contains information about power supplies on the server.

The syntax is *SEQUENCE OF SvrPowerSupplyEntry*.

svrPowerSupplyIndex

Advanced X86-Based and All Alpha-Based Servers/All Operating Systems

The svrPowerSupply Index variable presents an index value that is unique to a the local server. This variable is implemented on all Alpha platforms. The syntax is *INTEGER*.

svrPowerSupplyStatus

Advanced X86-Based and Advanced Alpha-Based Servers/All Operating Systems

The svrPowerSupplyStatus variable shows the current state of the power supply. When queried, it returns one of the following numeric values:

- 1 (unknown)
- 2 (running)
- 3 (backup)
- 4 (warning)
- 5 (failed)

svrPowerSupplyFruIndex

Advanced X86-Based Servers/All Operating Systems

The svrPowerSupplyFruIndex variable cross-references the index value in the FRU table in which this power supply is represented. This variable is not available on Alpha platforms. The syntax is *INTEGER*.

Index

A

Agents

- Compaq for Digital Servers, 1-6, 5-2
- DIGITAL, 1-6
- operations, 1-7

Alarm

- utilities, B-1

Alarm Configuration

- toolbar (*illus.*), B-4
- window (*illus.*), B-3
- window, description, B-2

Alarms. *See Also* SNMP traps

- component status, B-5
- component threshold, B-5
- details, B-2
- minimal health, 1-7, B-4
- overview, 1-7
- setting, B-1
- system status, B-5
- user defined, B-5
- viewing, B-1

B

Browsing

- DIGITAL MIB in CA Unicenter TNG, 4-12
- DIGITAL MIB in HP OpenView (*illus.*), 2-18
- DIGITAL MIB in Tivoli TME 10 NetView, 3-18

C

CA Unicenter TNG

- device status, 4-10
- DIGITAL MIB, 4-12
- integration, 4-1
- reporting, 4-12
- trap forwarding, 4-15
- traps, 4-14
- version, platform supported, 1-3

CA Unicenter TNG installation

- confirming, 4-4, 4-6
- prerequisites, 4-3, 4-4
- uninstalling, 4-7

Index

- Cluster Management MIB (ntcmgt.mib), 2-2, 3-2, 4-2
- Command lines
 - specifying for applications, 5-10
- Community names
 - default, 2-31, 4-15
- Compaq Agents for DIGITAL Servers, 1-6, 5-2
- Compaq Insight Manager
 - configuring, 5-10
 - device groups, 5-6
 - device list, 5-2, 5-6
 - integration, 5-1
 - launched from ServerWORKS Console (*illus.*), 5-12
 - version, platform supported, 1-3
 - Web-enabled agents, 5-8
- Compaq servers
 - managed from ServerWORKS, 5-9
- Component status alarms, B-5
- Component threshold alarms, B-5
- Configuring
 - alarms, B-2
 - Compaq Insight Manager, 5-10
 - SNMP traps on HP OpenView, 2-31, 2-29–2-31
 - SNMP traps on Tivoli TME 10 Netview, 3-27
 - traps, 4-14
 - traps, trap destinations, 4-14

- Context
 - in application, on node, 5-10

D

- Default actions
 - specifying application to launch, 5-11
- DIGITAL agents, 1-6
- DIGITAL icons, 2-14, 3-14, 4-9
- DIGITAL menu options, 1-8
- DIGITAL MIB, 4-12
 - CA Unicenter TNG, 4-12
 - loading, 2-2, 3-2
 - managing devices with, 2-17
 - path, 2-19, 3-19
- DIGITAL server
 - definition, 1-6
- DIGITAL Server in Insight Manager Device View Window (*illus.*), 5-7
- Discover
 - by attribute on OpenView, 2-15
 - DIGITAL device, 2-15
 - DIGITAL servers in OpenView, 2-14
 - DIGITAL Servers on DIGITAL UNIX, 3-16
 - DIGITAL Servers on NT, 3-15
 - on NetView, 3-14
- Displaying server information
 - on HP OpenView, 2-22
 - on NetView, 3-20

E

- Environment Group
 - of System Browser, contents, B-8
- Events Browser, 2-33

F

- Filter
 - Events on OpenView, 2-33
 - Traps, 2-33
- Forwarding
 - traps to OpenView, 2-31

G

- Graphing
 - options on NetView, 3-25
 - options on OpenView, 2-28

H

- Historical information
 - in System Browser, B-6
- HP OpenView
 - displaying server information, 2-22
 - menu bar on HP-UX (*illus*), 2-11
 - menu bar on NT(*illus*), 2-7
 - version, platforms supported, 1-3

- HP OpenView installation
 - confirming on HP-UX, 2-11
 - confirming on NT, 2-7
 - prerequisites on HP-UX, 2-4
 - prerequisites on NT, 2-3
 - procedure on HP-UX, 2-10
 - procedure on NT, 2-5
 - uninstalling from HP-UX, 2-12
 - uninstalling from NT, 2-8, 3-8
- HTTP management server. *See* Compaq Insight Manager XE

I

- Insight Manager. *See* Compaq Insight Manager
- Installation
 - CA Unicenter TNG, 4-2
 - on DIGITAL UNIX, 3-10
 - on DIGITAL UNIX with NetView, 3-4
 - on HP OpenView on NT, 2-5
 - on HP-UX, 2-4, 2-10
 - on NT with NetView, 3-5
 - on OpenView, 2-2
 - on Tivoli TME 10 NetView, 3-2
 - prerequisites on Tivoli TME 10 NetView, 3-3
- Integration
 - CA Unicenter TNG, 4-1, 4-5
 - Compaq Insight Manager, 5-1
 - concepts of, 1-11
 - levels of, 1-5
 - overview, 1-2
 - ServerWORKS on HP OpenView, 2-13
 - with management packages, 1-3

Index

Intelligent monitoring, 1-7
IP address
 obtaining for Insight Manager, 5-5

L

Launching applications
 with context, 5-9, 5-11

M

Macros
 on DIGITAL menu, 1-4
ManageWise. *See* Novell ManageWise
Managing DIGITAL objects
 with CA Unicenter TNG, 4-9
 with HP OpenView, 2-14–2-16
 with Tivoli TME 10 NetView, 3-17
MIB
 Cluster Management (ntcmgt.mib), 2-2,
 3-2, 4-2
 Server Cluster MIB (svrclu.mib), 2-2,
 3-2, 4-2
 Server Management (svrmgt.mib), 2-2,
 3-2, 4-2
 Server NT Cluster MIB (svrntc.mib), 2-2,
 3-2, 4-2
 Server System (svrsys.mib), 2-2,
 3-2, 4-2
MIBs
 compiling in ManageWise, A-1, A-2
 DIGITAL on OpenView, 2-17
 managing devices with, 2-17, 3-18
 SNMP, 2-17

Microsoft NT
 CA Unicenter TNG, 4-5
Minimal health icon, B-4

N

NetView. *See* Tivoli TME 10 NetView
Network Group
 of System Browser, contents, B-8
Novell ManageWise
 compiling ServerWORKS MIBs. *See*
 version, platform supported, 1-3

O

OpenView. *See* HP OpenView

P

Public community, 2-31, 4-15

R

Reporting
 CA Unicenter TNG, 4-12

S

Search by
 attribute, 2-15
Server Cluster MIB (svrclu.mib), 2-2, 3-2, 4-2
Server Management MIB (svrmgt.mib), 2-2,
 3-2, 4-2
Server NT Cluster MIB (svrntc.mib), 2-2, 3-2,
 4-2

Server System MIB. *See Also* DIGITAL MIB
 base system configuration, D-7–D-11
 buses and slots, D-26–D-29
 central processor configuration, utilization,
 D-18–D-26
 environment, D-44–D-54
 loading, 2-2, 3-2, 4-2
 peripheral devices, D-29–D-39
 physical configuration, D-44
 SNMP syntax extensions, D-5, D-6
 system memory, D-18–D-26
 viewing in CA Unicenter TNG, 4-13
 viewing in HP OpenView, 2-19
 viewing in Tivoli TME 10 Netview, 3-19

server, DIGITAL
 definition, 1-6

ServerWORKS
 Integration, 1-2

SNMP trap alarms, B-5

SNMP traps. *See Also* Alarms
 configuring on HP OpenView, 2-29
 defined by Tivoli, 3-27
 designating the destination, 2-30
 destination in CA Unicenter TNG, 4-14
 destination on OpenView, 2-30
 forwarding in CA Unicenter TNG, 4-15
 forwarding to OpenView, 2-31
 in CA Unicenter TNG, 4-14
 on NetView, 3-27
 services and Compaq servers, 5-9

Status
 of DIGITAL device, 4-10

Storage Group
 of system Browser, contents, B-8

Submap

devices, 2-20
 DIGITAL, 2-20

System Browser, B-1
 details available, B-8

System Browser window (*illus.*), B-7

System Group
 of System Browser, contents, B-8

System status alarms, B-5

T

Tivoli TME 10 NetView, 3-1
 Discover DIGITAL devices, 3-14
 managing devices using DIGITAL MIB,
 3-18
 menu bar on DIGITAL UNIX (*illus.*), 3-11
 menu bar on NT (*illus.*), 3-7
 versions, platforms supported, 1-3

Tivoli TME 10 NetView installation
 confirming on DIGITAL UNIX, 3-11
 confirming on NT, 3-7
 prerequisites on DIGITAL UNIX, 3-4
 prerequisites on NT, 3-3
 procedure on DIGITAL UNIX, 3-10
 procedure on NT, 3-5

Tools
 CA Unicenter TNG, 4-8
 ServerWORKS, 4-8

Trap and Event Levels
 HP OpenView and ServerWORKS, 2-29

Traps. *See* SNMP traps

Index

U

Unicenter TNG. *See* CA Unicenter TNG

Uninstall

 from DIGITAL UNIX, 3-12

Uninstalling ServerWORKS from HP-UX, 2-12

V

Variables

 DIGITAL on HP OpenView, 2-25

 DIGITAL on NetView, 3-22

 Displaying, 2-25

 environmental on NetView, 3-22

 environmental on OpenView, 2-25

 memory on NetView, 3-22

 memory on OpenView, 2-25

 network on NetView, 3-23

 network on OpenView, 2-26

 storage on NetView, 3-23

 storage on OpenView, 2-26

 system on NetView, 3-24

 system on OpenView, 2-27

View

 Events on OpenView, 2-32

W

Web-based management. *See* Compaq Insight Manager.