Media Processing Server System Operator's Guide

(Software Release 2.1)

Avaya Business Communications Manager
Release 6.0

Document Status: **Standard**

Document Number: **P0602484**

Document Version: **02.07**

Date: **June 2010**

Notices
While reasonable efforts have been made to ensure that the information in this document is complete and accurate at the time of printing, Avaya assumes no liability for any errors. Avaya reserves the right to make changes and corrections to the information in this document without the obligation to notify any person or organization of such changes.

Documentation disclaimer
Avaya shall not be responsible for any modifications, additions, or deletions to the original published version of this documentation unless such modifications, additions, or deletions were performed by Avaya. End User agree to indemnify and hold harmless Avaya, Avaya's agents, servants and employees against all claims, lawsuits, demands and judgments arising out of, or in connection with, subsequent modifications, additions or deletions to this documentation, to the extent made by End User.

Link disclaimer
Avaya is not responsible for the contents or reliability of any linked Web sites referenced within this site or documentation(s) provided by Avaya. Avaya is not responsible for the accuracy of any information, statement or content provided on these sites and does not necessarily endorse the products, services, or information described or offered within them. Avaya does not guarantee that these links will work all the time and has no control over the availability of the linked pages.

Warranty
Avaya provides a limited warranty on this product. Refer to your sales agreement to establish the terms of the limited warranty. In addition, Avaya's standard warranty language, as well as information regarding support for this product, while under warranty, is available to Avaya customers and other parties through the Avaya Support Web site: http://www.avaya.com/support
Please note that if you acquired the product from an authorized reseller, the warranty is provided to you by said reseller and not by Avaya.

Licenses
THE SOFTWARE LICENSE TERMS AVAILABLE ON THE AVAYA WEBSITE, HTTP://SUPPORT.AVAYA.COM/LICENSEINFO/ ARE APPLICABLE TO ANYONE WHO DOWNLOADS, USES AND/OR INSTALLS AVAYA SOFTWARE, PURCHASED FROM AVAYA INC., ANY AVAYA AFFILIATE, OR AN AUTHORIZED AVAYA RESELLER (AS APPLICABLE) UNDER A COMMERCIAL AGREEMENT WITH AVAYA OR AN AUTHORIZED AVAYA RESELLER. UNLESS OTHERWISE AGREED TO BY AVAYA IN WRITING, AVAYA DOES NOT EXTEND THIS LICENSE IF THE SOFTWARE WAS OBTAINED FROM ANYONE OTHER THAN AVAYA, AN AVAYA AFFILIATE OR AN AVAYA AUTHORIZED RESELLER, AND AVAYA RESERVES THE RIGHT TO TAKE LEGAL ACTION AGAINST YOU AND ANYONE ELSE USING OR SELLING THE SOFTWARE WITHOUT A LICENSE. BY INSTALLING, DOWNLOADING OR USING THE SOFTWARE, OR AUTHORIZING OTHERS TO DO SO, YOU, ON BEHALF OF YOURSELF AND THE ENTITY FOR WHOM YOU ARE INSTALLING, DOWNLOADING OR USING THE SOFTWARE (HEREINAFTER REFERRED TO INTERCHANGEABLY AS "YOU" AND "END USER"), AGREE TO THESE TERMS AND CONDITIONS AND CREATE A BINDING CONTRACT BETWEEN YOU AND AVAYA INC. OR THE APPLICABLE AVAYA AFFILIATE ("AVAYA").

Copyright
Except where expressly stated otherwise, no use should be made of the Documentation(s) and Product(s) provided by Avaya. All content in this documentation(s) and the product(s) provided by Avaya including the selection, arrangement and design of the content is owned either by Avaya or its licensors and is protected by copyright and other intellectual property laws including the sui generis rights relating to the protection of databases. You may not modify, copy, reproduce, republish, upload, post, transmit or distribute in any way any content, in whole or in part, including any code and software. Unauthorized reproduction, transmission, dissemination, storage, and or use without the express written consent of Avaya can be a criminal, as well as a civil offense under the applicable law.

Third Party Components
Certain software programs or portions thereof included in the Product may contain software distributed under third party agreements ("Third Party Components"), which may contain terms that expand or limit rights to use certain portions of the Product ("Third Party Terms"). Information regarding distributed Linux OS source code (for those Products that have distributed the Linux OS source code), and identifying the copyright holders of the Third Party Components and the Third Party Terms that apply to them is available on the Avaya Support Web site: http://support.avaya.com/Copyright.

Trademarks
The trademarks, logos and service marks ("Marks") displayed in this site, the documentation(s) and product(s) provided by Avaya are the registered or unregistered Marks of Avaya, its affiliates, or other third parties. Users are not permitted to use such Marks without prior written consent from Avaya or such third party which may own the Mark. Nothing contained in this site, the documentation(s) and product(s) should be construed as granting, by implication, estoppel, or otherwise, any license or right in and to the Marks without the express written permission of Avaya or the applicable third party. Avaya is a registered trademark of Avaya Inc. All non-Avaya trademarks are the property of their respective owners.

Downloading documents
For the most current versions of documentation, see the Avaya Support. Web site: http://www.avaya.com/support

Contact Avaya Support
Avaya provides a telephone number for you to use to report problems or to ask questions about your product. The support telephone number is 1-800-242-2121 in the United States. For additional support telephone numbers, see the Avaya Web site: http://www.avaya.com/support

Table of Contents

Preface

Scope

The Avaya Media Processing Server (MPS) Series System Operator's Guide provides information and procedures related to the daily operation, monitoring, and maintenance of the Media Processing Server Series system.

Intended Audience

The audience for this manual consists of people responsible for the monitoring and maintenance of the Media Processing Server Series system. A user of this class is referred to as a *system operator.* System operators must be familiar with telecommunications and computer equipment, their functions, and associated terminology. In addition, they must be familiar with site-specific telephone systems, power systems, computer systems, and peripheral components such as terminals, modems, printers, etc.

This guide assumes that the Media Processing Server Series system operators have completed an on-site system briefing program as part of the initial system installation. In addition, they should be familiar with other site-specific operating procedures relating to the Media Processing Server Series that are due to specific application functions performed by that platform and with any other equipment to which the Media Processing Server Series is connected. This may include a PBX/ACD and/or a host computer system, etc.

It is further assumed that the reader is already familiar with the following publications: *PeriView Reference Manual*, *PeriReporter User's Guide*, *PeriStudio User's Guide*, and *Media Processing Server Developer User's Guide*.

In addition, the following publications should be consulted, as needed: *MPS System Reference Manual*, *MPS Telephony Reference Manual*, and *MPS COMMGR Reference Manual*.

How to Use This Manual

This manual uses many standard terms relating to computer system and software application functions. However, it contains some terminology that can only be explained in the context of the Media Processing Server system. Refer to the *Glossary of Media Processing Server Series Terminology* for definitions of product- specific terms.

It is recommended that initially, this manual be read at least once, from start to finish. Later, use the ***Table of Contents*** and ***Index*** to locate topics of interest for reference and review.

If reading the online version of this document, use cross-reference links to quickly locate related topics. <LEFT> click once with the mouse while the pointer is positioned over a cross-reference link to move to the target page. Click any Table of Contents entry to move to that topic. Click on the page number of any Index entry to access that page.

To become familiar with various specialized textual references within the manual - see ***Conventions Used in This Manual*** on page 12.

Periphonics is part of Avaya. The name Periphonics, and variations thereof, appear in this manual only where it is referred to in a product (As examples, a MPS Developer application, the `PERImps` package, the **`perirev`** command, etc.).

Organization of This Manual

This manual contains separate chapters that describe one or more related tasks for the operation, monitoring, and basic maintenance of the Media Processing Server Series system. References are also provided to outside sources of more detailed information as applicable.

Chapter 1 - Introduction

This chapter discusses the general characteristics of the Media Processing Server Series system and outlines general activities performed by that system's operator. See ***Introduction*** on page 16

Chapter 2 - System Startup, Monitoring and Shutdown

This chapter describes startup operation, system-level monitoring capabilities, and shutdown commands. Included are an overview of system hardware and PeriView activities relevant to such activities. See ***System Startup, Monitoring, and Shutdown*** on page 24

Chapter 3 - Command Line Functions

This chapter describes interaction with the system from a command line perspective. It also discusses the basics of using the PeriView VSH Tool and the on-line documentation system. See ***Command Line Functions*** on page 46.

Chapter 4 - Status Monitoring

This chapter discusses monitoring of individual aspects of the Media Processing Server Series system. Included are instructions on using and interpreting the results of the PeriView Alarm Viewer and PeriReporter statistics tool. See ***Status Monitoring*** on page 54

Chapter 5 - Voice Processing Applications

This chapter discusses application related prerequisites, assigning and starting applications, activating vocabularies, working with MultiMedia Format (MMF) files, and using Caller Message Recording (CMR). See ***Voice Processing Applications*** on page 84

Chapter 6 - Backing up and Restoring Files

This chapter describes commands and procedures used for routine backup and restore operations. See ***Backing up and Restoring Files*** on page 122.

Chapter 7 - Troubleshooting

This chapter discusses how to use various monitoring functions to identify problems with system operation, their possible causes, and suggested remedial actions. See ***Troubleshooting*** on page 152.

Chapter 8 - Printer and Modem Setup

This appendix covers setup and changes to the peripheral components of the Media Processing Server Series system. See ***Printer and Modem Setup*** on page 164.

Chapter 9 - MPS Directory structure

This appendix shows the Media Processing Server Series file hierarchy and explains the intended usage and functions of various directories and associated files. See ***MPS Directory Structure*** on page 168.

Conventions Used in This Manual

This manual uses different fonts and symbols to differentiate between document elements and types of information. These conventions are summarized in the following table.

Conventions Used in This Manual

Notation	Description
Normal text	Normal text font is used for most of the document.
important term	The Italics font is used to introduce new terms, to highlight meaningful words or phrases, or to distinguish specific terms from nearby text.
`system command`	This font indicates a system command and/or its arguments. Such keywords are to be entered exactly as shown (i.e., users are not to fill in their own values).
`command`, **`condition`** and **`alarm`**	Command, Condition and Alarm references appear on the screen in magenta text and reference the *Command Reference Manual*, the *Media Processing Server Developer User's Guide*, or the *Alarm Reference Manual*, respectively. Refer to these documents for detailed information about **`Commands`**, **`Conditions`**, and **`Alarms`**.
`file name / directory`	This font is used for highlighting the names of disk directories, files, and extensions for file names. It is also used to show displays on text-based screens (e.g., to show the contents of a file.)
on-screen field	This font is used for field labels, on-screen menu buttons, and action buttons.
<KEY NAME>	A term that appears within angled brackets denotes a terminal keyboard key, a telephone keypad button, or a system mouse button.
Book Reference	This font indicates the names of other publications referenced within the document.
cross reference	A cross reference appears on the screen in blue text. Click on the cross reference to access the referenced location. A cross reference that refers to a section name accesses the first page of that section.
	The Note icon identifies notes, important facts, and other keys to understanding.
	The Caution icon identifies procedures or events that require special attention. The icon indicates a warning that serious problems may arise if the stated instructions are improperly followed.
	The flying Window icon identifies procedures or events that apply to the Windows 2000 operating system only.[a]
	The Solaris icon identifies procedures or events that apply to the Solaris operating system only.[b]

a. Windows 2000 and the flying Window logo are trademarks or registered trademarks of the Microsoft Corporation.

b. Solaris is a trademark or registered trademark of Sun Microsystems, Inc. in the United States and other countries.

Solaris and Windows 2000 Conventions

This manual depicts examples (command line syntax, configuration files, and screen shots) in Solaris format. In certain instances, Windows 2000-specific commands, procedures, or screen shots are shown where required. The following table lists examples of general operating system conventions to keep in mind when using this manual with either the Solaris or Windows 2000 operating system.

	Solaris	**Windows 2000**
Environment	`$MPSHOME`	`%MPSHOME%`
Paths	`$MPSHOME/common/etc`	`%MPSHOME%\common\etc`
Command	**`<command> &`**	**`start /b <command>`**

Trademark Conventions

The following trademark information is presented here and applies throughout for third-party products discussed within this manual. Trademarking information is not repeated hereafter.

Solaris is a trademark or registered trademark of Sun Microsystems, Inc. in the United States and other countries.

Microsoft, Windows, Windows 2000, Internet Explorer, and the Flying Windows logo are either trademarks or registered trademarks of Microsoft Corporation.

Netscape® and the Netscape N® and Ship's Wheel® logos are registered trademarks of Netscape Communications Corporation in the U.S. and other countries. Netscape Navigator is also a trademark of Netscape Communications Corporation and may be registered outside the U.S.

Common Desktop Environment (CDE) is a copyright © of Sun Microsystems, Inc., Hewlett-Packard Company, International Business Machines Corp., and Novell, Inc.

Copyright Information

HylaFAX Facsimile Software

1

Introduction

This chapter covers:

1. **Introduction**
2. **System Operator Tasks**
3. **System Terminology**
4. **System Configurations**

Introduction

The Avaya Media Processing Server (MPS) Series product lines consist of hardware and software for performing Interactive Voice Response (IVR) and media processing functions in call processing environments. More precisely, MPS systems integrate call processing components with speech, telephony, data communications, and transaction processing functions.

The heart of the MPS system is the Avaya Telephony Media Server (TMS), which provides high phone port densities with user flexibility, scalability, and extensibility. The basic TMS assembly provides resources for telephony media management including switching/bridging, digital signal processing, voice and data memory, and network interfaces. A variety of IVR applications are accommodated easily, from simple information delivery services to complex multimedia (voice/fax/data/web) call processing implementations with local databases, multiple services, and considerable transaction processing functions.

The MPS supports a wide variety of telephony and host computer interfaces, and easily integrates with equipment already in place at any data-processing/ communications environment. MPS systems employ industry standards and distributed processing in an open architecture, allowing plug-in integration of future technological developments. In addition, the MPS supports multiple LAN/WAN interfaces, providing an environment ready for distributed computing.

The MPS is available in the following configurations:

- Media Processing Server 100
  A single rack-mounted unit for use with the Windows 2000 platform. The typical system contains two spans, but can support up to eight. It has a single Digital Communications Controller (DCC) card. Bridging outside of the TMS is not supported.

- Media Processing Server 500
  The MPS 500 consists of one Telephony Media Server (TMS) which supports 8 T1/E1 spans (up to 240 ports) or comparable capacity VoIP channels. The MPS 500 is capable of providing full IVR functionality including fax, caller message recording, conferencing, speech recognition and text-to-speech (using OSCAR resource servers), SQL database access, host screen scrape access, CTI integration, and supports building and running of an application to control a given call

- Media Processing Server 1000
  A cabinet-enclosed, networked configuration, available only on the Solaris platform. It supports four TMS units per chassis, with up to four chassis per cabinet. The MPS can support up to ten thousand ports with the ability to bridge between any two ports regardless of their location.

The MPS also includes a set of easy-to-use, objected-oriented, GUI tools. These tools are used for:

- Application and vocabulary development
- System configuration, control, and monitoring
- Collecting and reporting of statistical data
- Access to online documentation

The Media Processing Server Developer application development environment provides a complete graphical medium for the entire application life cycle. It also allows telephony-based applications to be ported over to Internet-based Web usage.

System Operator Tasks

The primary task of the MPS system operator is to ensure the smooth daily operation of the system. System operator tasks include:

- Keeping records and monitoring system usage and status
- Backing up and restoring system, application, and MultiMedia Format (MMF) vocabulary/fax files
- Manipulating applications
- Occasional shutting down and restarting of the system (in cases of power failure or other system maintenance needs) if it fails to recover automatically
- Performing speech/vocabulary updates (as required)
- Performing initial problem analysis and service requests
- Working with Certified Avaya service personnel or others during hardware upgrade and maintenance

To accomplish these tasks, an understanding of the fundamentals of the MPS is required. This includes knowing the terminology associated with the system and the various configurations a system can have.

Certified Avaya field service engineers normally perform the initial site preparation and system installation. The installation includes configuring appropriate system and application-execution parameters.

System Terminology

The following is a basic presentation of the terminology associated with MPS systems. For additional information about terminology, see the *Glossary of Media Processing Server Series Terminology.*

Component	A unit within the system that performs a particular function (see ***System Configurations*** on page 19).
Media Processing Server (MPS)	An Interactive Voice Response system comprised of various components that provides telephony and call processing services between a call center and the calling community.
Node	A physical workstation in a Media Processing Server Series network.
Telephony Media Server (TMS)	The main component within a MPS system. It provides most essential functions for telephony and media management. It replaces the CPS-1000 used in the previous generation VPS and VPS/is systems.
Application Services Environment (ASE)	The group of software processes used to develop and execute voice and media applications.
Call Control Manager (CCM)	The primary interface between applications and the VOS services. It is the gateway for communication with the telephony hardware platform.
MultiMedia Format (MMF)	A proprietary format for storing and retrieving audio elements and/or fax data in data files. A single MMF file is physically stored as two files. The data file (`.mmd` extension) contains raw audio recordings, such data for voice, fax, TDD tones, etc. The index file (`.mmi` extension) contains the attributes and pointers that differentiate the elements in the data file.
PeriView	The suite of tools designed to administer MPS network activity in a GUI environment (see ***Toolkit*** on page 21)..
Startup and Recovery Process (SRP)	The software process that spawns (i.e., starts up) and monitors all other MPS processes. SRP can also be used to stop and restart these processes if they've been abnormally terminated.
Voice Memory Manager (VMM)	A VOS software subcomponent that provides audio and media management services within the VOS.
Voice Operating Software (VOS)	The main software processes that provide the lower-level operating functions of the MPS system.
Voice Processing Series (VPS)	The predecessor product to the MPS. By design, a number of the MPS path names, software processes, environment variables, and file fields use the term VPS. This terminology is functionally equivalent to MPS nomenclature. In other words, wherever file or software names show the term VPS, it is relevant to the MPS.
$MPSHOME	The home directory environment variable, set to `/opt/vps` by default. This directory contains the operating system software, subdirectories for the various installed software packages subdirectory, symbolic links to these packages, and other software necessary for MPS operations. For more information, (see ***System software home directory $MPSHOME (/opt/vps)*** on page 172).

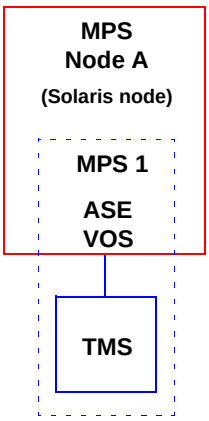
System Configurations

A MPS system setup can be a single MPS or contain multiple networked MPS systems. PeriView Workstations can be added, as needed, to allow monitoring of the system from remote locations. The MPS also supports functions between systems with different operating system releases.

Single Media Processing Server

A Media Processing Server node is the basic building block of a MPS network. A single MPS component consists of the TMS hardware, and ASE and VOS software groups running on a Solaris node. (A node is a physical workstation in the MPS network.)

The single MPS configuration is intended for environments with low to moderately high call volumes. This system is monitored directly by PeriView. Generated statistics are available via PeriReporter.



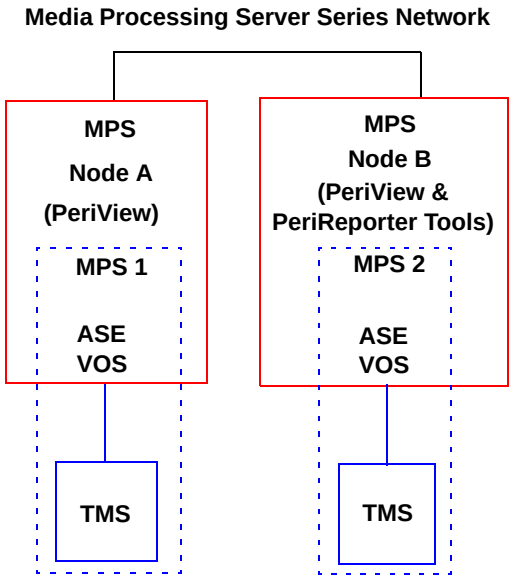
Each workstation in a MPS network is identified by a node name. In the above diagram, it is shown as “Node A.” Each MPS component is identified by a unique component number. In the diagram, it is shown as “MPS 1.” A component number must be unique across all nodes. The convention for naming MPS network components is *component_type.component_number/node_name*. Thus, “mps.1/A” is the designation of the MPS in this diagram,

Multiple Media Processing Servers

In a Media Processing Server Series network, each MPS can process calls only on its own phone lines. Networking the systems increases the total number of available phone lines, permits flexibility in assigning different types of applications to particular segments of the network, and allows component monitoring from multiple nodes.

The following diagram illustrates the basic concepts of a MPS *network*. Although networks can have a different components in various configurations, for illustration purposes, this example shows a two-node Media Processing Server Series network.

PeriView tools reside on every node. However, PeriReporter tools resides *only* on the statistics node (shown as node "B" in this diagram). The nodes are networked (either on a LAN or WAN) thus allowing statistics collected from node "A" to be consolidated on node "B" (along with the statistics from node "B"). Also, either node can be monitored through the other node's PeriView tools. (For more information about PeriReporter Tools, see the *PeriReporter User's Guide*.)

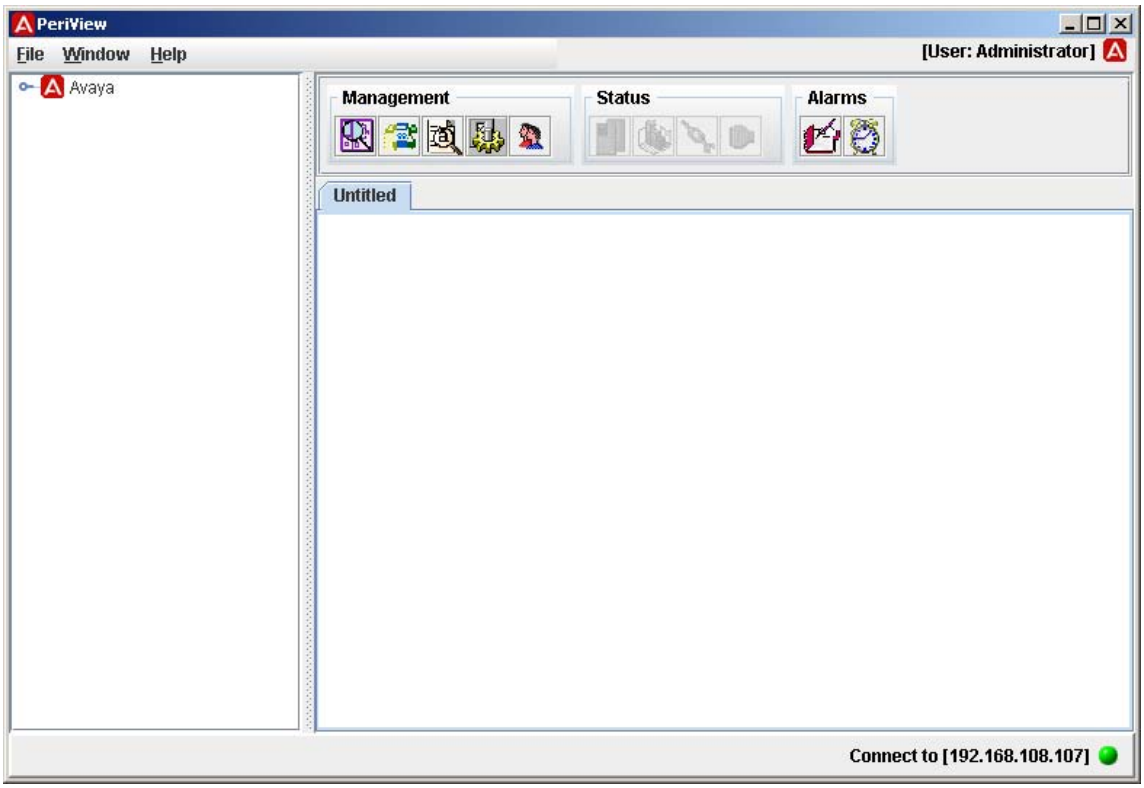


The example shows each MPS node identified by a node name ("A" and "B") and each MPS component identified by a unique number ("1" and "2"). Thus, "mps.1/A" and "mps.2/B" are the designations of the MPS systems in this network.

PeriView

Toolkit

PeriView is a suite of GUI tools designed to administer Media Processing Server Series network activity.



PeriView performs the following various functions:

- Defines the Media Processing Server Series network as a hierarchy of entities and designate network perspectives from which to manipulate and view network activity.
- Launch PeriView tools to configure user accounts and privileges, manage applications, monitor phone lines, view and log alarms, and monitor system activity.
- Customize operations by selecting and specifying PeriView properties.

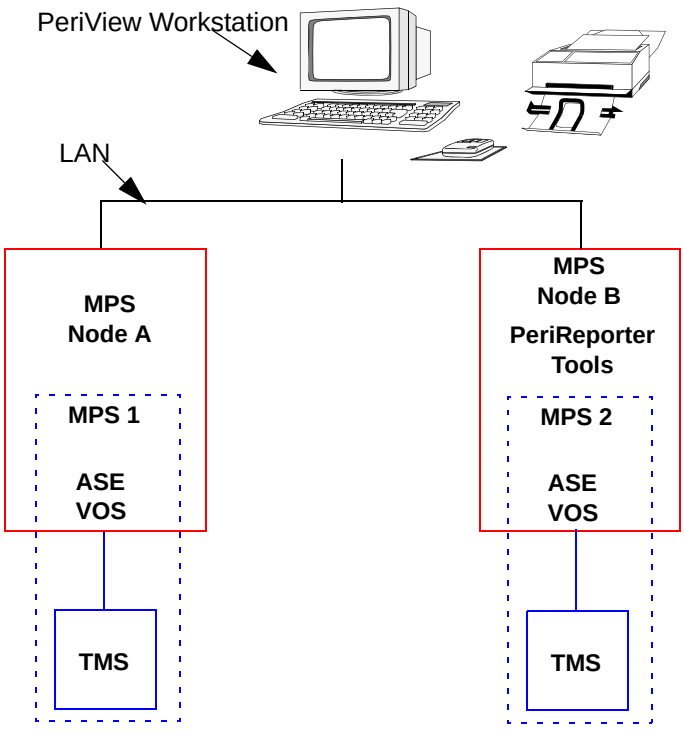
See the *PeriView Reference Manual* for detailed information about PeriView.

Workstation Node

A PeriView Workstation node is a management resource for remote monitoring of the

MPS systems in the network. A PeriView node does not have MPS components. It contains only the software necessary to run the PeriView GUI tools, but none of the software or hardware needed for telephony functions. A PeriView Workstation can be physically located anywhere on a network with component-based nodes.

Media Processing Server Series Network — with Media Processing Server and PeriView Workstation



In this example configuration, PeriView can be launched from any node in the network. PeriReporter tools, however, can only be used only on node "B." The PeriView Workstation can be used to monitor both nodes, but statistics reports can only be generated through PeriReporter on node "B." If desired, the PeriView Workstation could also be designated as the statistics node.

The PeriView Workstation node does not normally appear in the on-screen PeriView workspace. To obtain a visual representation of this node, it has to be added to the `.xtrahostsrc` file. For information about this file, see the *PeriView Reference Manual*.

2

System Startup, Monitoring and Shutdown

This chapter covers:

1. **TMS Startup/Shutdown**
2. **System Monitoring**
3. **Logging off a Node**
4. **System Shutdown**

System Startup, Monitoring, and Shutdown

Occasionally, the Avaya Media Processing Server Series has to be shut down and restarted for maintenance or to recover from a power failure. Also, the TMS might occasionally require recycling, which can be done without having to stop the entire system. The following sections contain details on these procedures, as well as those for overall system monitoring. There are three ways to monitor the system: indicators and controls, the Alarm Viewer, and PeriReporter.

TMS Startup/Shutdown

Power to the TMS can be applied or removed either manually via a switch on the unit or by issuing system commands. Refer to the following table to determine the best method to use in a given situation. (see ***Alarm Viewer*** on page 39).,for information on using the Alarm Viewer as cited below and refer to the *Alarm Reference Manual* for detailed description of the Alarm conditions.

Command/Function	Use to...	Why?
TEST / ON / OFF switch located on the Front Control Panel (FCP, see page 18) or Variable Resource Chassis (VRC) Rear Panel (see page 19).	Manually turn power to the TMS on or off	Use when completely shutting down system or performing service on the TMS (see (see ***System Shutdown*** on page 41). and to reapply power when finished with these steps.
`S20vps.startup start` *(preferred command)*	Start SRP (including VOS) and TMS hardware "gracefully" (that is, through the most efficient means as determined by Avaya)	Use after the stop option (see below) has been invoked, or to restart the TMS if its startup terminated abnormally (use the Alarm Viewer to check for causes)
`S20vps.startup stop` *(preferred command)*	Stop SRP (including VOS) and TMS hardware "gracefully" (that is, through the most efficient means as determined by Avaya)	Use when performing maintenance on the system that requires all Media Processing Server Series processes or the TMS to be down (for example, modifying the Media Processing Server Series network), or to stop Media Processing Server Series processes and the TMS if their startup terminated abnormally (use the Alarm Viewer to check for causes)
`srp vos.# -stop`	Stop VOS processes and TMS	Use if VOS processes have crashed or recycled during an attempt to load the TMS. Use the Alarm Viewer to check for causes.[a]
`srp vos.# -start`	Start VOS processes and TMS	Use to start the VOS and TMS after having issued the **`-stop`** option (see above).[1]
`srp vos.# -restart`	Stop and then restart the VOS and TMS. Can be used as an alternative to the **`-stop`** and **`-start`** commands	Use for same reasons as the previous two commands when there is no need to see the relevant alarms.[1]

a. In these examples, # represents the Media Processing Server with which the TMS is associated.

Automatic Shutdown and Restart

If the TMS Configuration and Alarm Daemon (TCAD) detects a CONFIG or SHUTDOWN state when TMS is started, it shuts down and restarts the TMS hardware. All VOS processes return to the stage in the startup sequence where attempts are made to establish connections to the hardware ports.

State	Indication	Cause	Status
CONFIG	Another VOS is loading the TMS	Severe configuration error	Contact Certified Avaya support personnel immediately
	VOS crashed/ recycled while attempting a prior configuration	Interruption of normal loading processes due to an unexpected condition within the system (i.e. software or hardware fault, or loss of power)	TMS is in an unknown configuration since prior commands may not have completed successfully
SHUTDOWN	TMS was stopped/ shut down manually	Operator issued a command line order	Hardware is detected in an unexpected state by TCAD when system is still up and running

Alarms are generated after TCAD successful corrects error conditions to signify that such actions have occurred. Use the PeriView Alarm Viewer to examine these alarms (see ***Alarm Viewer*** on page 39).) and refer to the *Alarm Reference Manual* for detailed description of the Alarm conditions.

Routine Media Processing Server Startup

Media Processing Server configuration files automate the routine startup of each system in the Media Processing Server Series network. During on-site installation, Avaya staff customize these configuration files for each system. These configuration files are used to:

- activate application programs
- activate MultiMedia Format (MMF) (vocabulary) files
- configure host interaction
- configure the default startup environment

How to ... Perform a Media Processing Server Startup

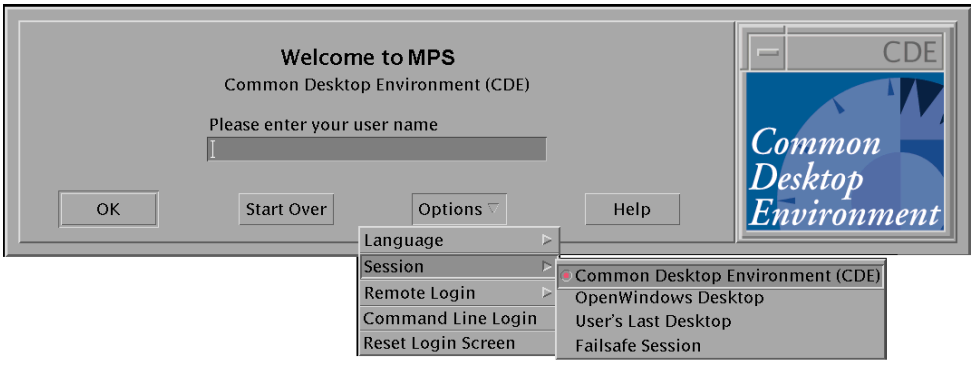
1. **On each Media Processing Server, turn the main power switch on**.

2. **Power up each TMS.**
   For locations of their power switches, see (see ***Front Control Panel*** on page 35). and (see ***VRC Rear Panel*** on page 36)..

3. **Return to the node that acts as the PeriView Workstation.**
   The system automatically starts a monitoring Alarm Viewer in which it displays startup messages.

4. **Enter the login name and select Session from the Options menu button. Select either Common Desktop Environment (CDE) or OpenWindows Desktop from the pulldown menu.**

   This is the desktop environment. At the next time log in, the desktop defaults to the previous selection (as indicated by the red dot at User's Last Desktop). The default login name is **`peri`**. The name of the node to log into (in this example MPS) appears in the "Welcome" prompt.



5. **Click the OK button (or press <RETURN>), then enter the password and repeat**.

   The default password is **`peri;`**). The selected desktop environment should now be displayed.

   The initial Alarm Viewer becomes iconified and a `cmdtool` window opens. The menu buttons on this startup Alarm Viewer are inactive.

   To view alarms after startup, launch a separate Alarm Viewer (see ***Alarm Viewer*** on page 39)..

6. **Wait for each Media Processing Server to boot up.** Use the Alarm Viewer to watch the system load vocabularies, start processes, and initialize the TMS on each one. After several minutes, each unit generates a “Media Processing Server UP” alarm.

7. **Start PeriView**. (see ***Starting/Restarting PeriView*** on page 30)..

Configuration files load all the necessary software. After startup, the Media Processing Server is ready for call processing.

Starting/Restarting PeriView

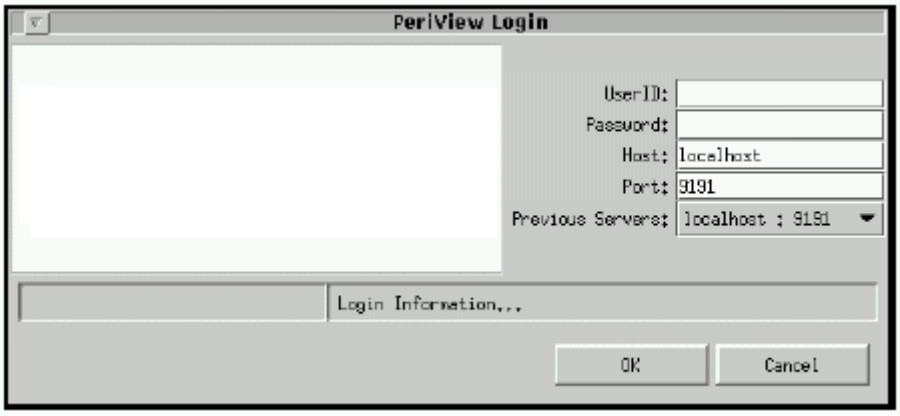
PeriView is used to complete system startup and interact with the Media Processing Server.

If PeriView is exited for any reason, it can be restarted without rebooting the system.

How to ... **Start/Restart PeriView**

1. **<RIGHT> click the desktop to display the Workspace menu**.
2. **<RIGHT> click *Avaya* in the Workspace pop-up menu**.
   This menu is user-defined and its contents may vary. To define the items on this menu, and which ones can otherwise be launched from a command line, edit the `.openwin-menu-`avaya file for the OpenWindows Desktop or the `.dt/dtwmrc` file for CDE, using any text-based editor. These files are typically found in a user's `/home` directory.
3. **From the Avaya menu, select PeriView...**.

The system displays the PeriView Login screen.



PeriView can also be started by entering `periview &` at any command line. A `cmdtool` window opens upon initialization of the system (see ***Routine Media Processing Server Startup*** on page 28).).

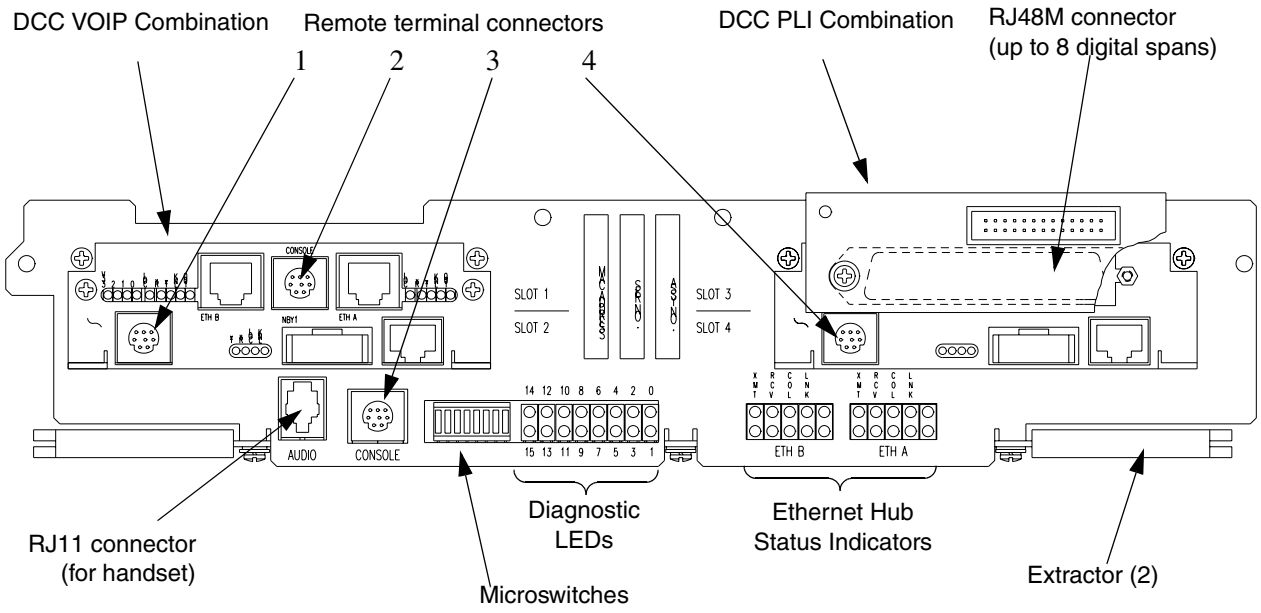
System Monitoring

MPS 500 Indicators and Controls

Front Panel
The chassis front panel contains no controls, connectors or visible indicators.

TMS Front Panel
In addition to front panel indicators, the system components have internal indicators

useful for system maintenance and fault isolation. Located on the front panel of the TMS are connections to a diagnostic console for troubleshooting, LED indicators for TMS health checks, and the span input/output connector.



TMS Front Panel Connectors and Indicators

Item	Description
AUDIO connector	RJ11 jack for connecting a telephone handset to audibly monitor an MPS 500 line.
CONSOLE connectors	RS232 (9-pin DIN) port for connecting a debug terminal. Used for boot ROM and diagnostics.
LEDs **0** - **15**	Provide diagnostic indications on startup.
Microswitches	Reserved for future enhancement.
HUB A LEDs	**EXT** Indicates condition on external port of HUB A. **SL2** Indicates condition on slot 2 port of HUB A. **SL1** Indicates condition on slot 1 port of HUB A. **MP** Indicates condition on microprocessor port of HUB A. **COL** Indicates errors or collisions in HUB A.
HUB B LEDs	**EXT** Indicates condition on external port of HUB B. **SL4** Indicates condition on slot 4 port of HUB B. **SL3** Indicates condition on slot 3 port of HUB B. **MP** Indicates condition on microprocessor port of HUB B. **COL** Indicates errors or collisions in HUB B.

DCC Front Panel

Item	Description
Span connector	RJ48M 50-pin connector connects up to eight digital spans (T1 or E1) to PSTN.
Remote terminal connector	RS232 (9-pin DIN) port for connecting a debug terminal. Used for boot ROM and diagnostics.
Other connectors and indicators	Reserved for future enhancement.

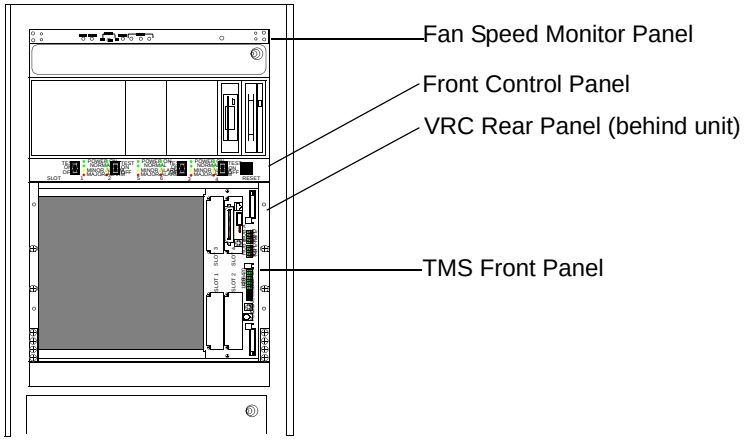
Ethernet Hub Indicators

The dual **(green** and **re**d) LED indicators for **HUB A** and **HUB B** on the front of the TMS can display five separate conditions of the item to which they apply. The LEDs will be on steady, or blinking in the following combinations to indicate the respective condition:

Indicator/Control	Function
Steady **green**	Receive link pulse FIFO error
Blinking **green** (@~ 4Hz)	Transmit packet (N/A)
Steady **red**	Reversed polarity Collision
Blinking **red** (@~ 4Hz)	Receive packet (N/A)
Alternating **red** and **green** (@~ 5Hz) P	artitioned out (N/A)

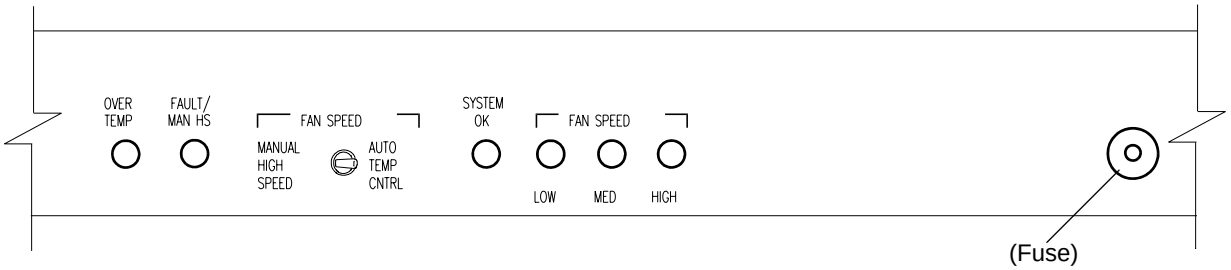
MPS 1000 Indicators and Controls

The Media Processing Server Series systems provide a number of indicators on various panels. The indicators give a visual status of system operation.



Fan Speed Monitor Panel

This panel provides information and control for overall cabinet cooling. Normal cooling is maintained automatically, but this can be overridden in the event of an over temperature (overtemp) condition (i.e., a state in which a preset temperature limit is exceeded).



Fan Speed Monitor Panel

Indicator/Control	Function
OVER TEMP indicator	Illuminates if the cabinet exceeds the predetermined safe operating temperature of 35°C (95°F).
FAULT/MAN HS indicator	Illuminates when a cooling system fault has been detected and the fan has been switched to MANUAL HIGH SPEED (see next).
FAN SPEED switch	Two-position switch used to set cooling fan speed for cabinet airflow.
MANUAL HIGH SPEED	Use this setting when an overtemp condition is present and continued maximum cooling effectiveness is required. Placing the FAN SPEED switch to this setting also causes the FAULT/MAN HS indicator to illuminate. **Caution should be exercised and service immediately performed to alleviate the overtemp condition.**
AUTO TEMP CNTRL	This setting is normally used to cool the cabinet enclosure. Fan speed is automatically adjusted for the optimum cooling capacity. The actual fan speed when in this setting is shown by the FAN SPEED indicators.
SYSTEM OK indicator	Illuminates when all predetermined cooling and temperature parameters have been met. An OVER TEMP or FAULT/MAN HS condition causes this indicator to go out.
FAN SPEED indicators	Illuminates in conjunction with the FAN SPEED AUTO TEMP CNTRL setting to indicate the actual fan speed. Indicators are self-explanatory.
Fuse	Replaceable fuse for cooling system over voltage protection. Replace with a 3 amp fast 5x20mm cartridge fuse only. Check system for cause if fuse is blown or call Avaya service department.

Front Control Panel

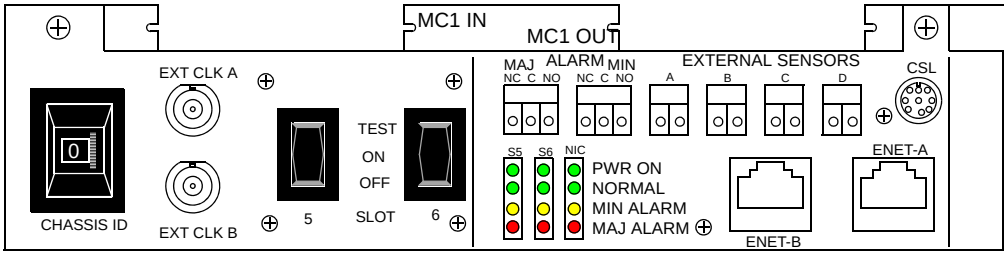
The cabinet front panel contains a standard 3.5 inch floppy disk drive, a 4mm DAT tape drive, and an FCP (Front Control Panel) for each VRC in the system. The FCP provides separate power controls and status indicators for each TMS (by chassis slot).

Front Control Panel

Indicator/Control	Function
TEST / **ON** / **OFF** switch (slots 1, 2, 3, and 4)	Three-position rocker switch used to turn power to the TMS on or off, or to test the associated power supply. (The **TEST** / **ON** / **OFF** switches for slots 5 and 6 are on the VRC rear panel. (see ***VRC Rear Panel*** on page 36)..)
POWER ON indicator (all slots)	Illuminates **green** when power is applied to TMS.
NORMAL indicator (all slots)	Illuminates **green** when TMS is operating properly and there are no alarms or system faults.
MINOR ALARM indicator (all slots)	Flashes **yellow** when a minor alarm is active, accompanied by an audible indication (beep). To reset, press the RESET switch. Use as a troubleshooting aid by viewing alarms in the Alarm Viewer (see ***Alarm Viewer*** on page 39)..
MAJOR ALARM indicator (all slots)	Flashes **red** when a major alarm is active, accompanied by an audible indication (beep). To reset, press the RESET switch. Use as a troubleshooting aid by viewing alarms in the Alarm Viewer (see ***Alarm Viewer*** on page 39).).
RESET switch	Push-button switch for resetting visual and audible alarm indications.

VRC Rear Panel

The rear panel of the VRC contains indicators, switches, and connectors for maintenance, configuration, and connection to other system components. The following illustration shows the VRC rear panel.

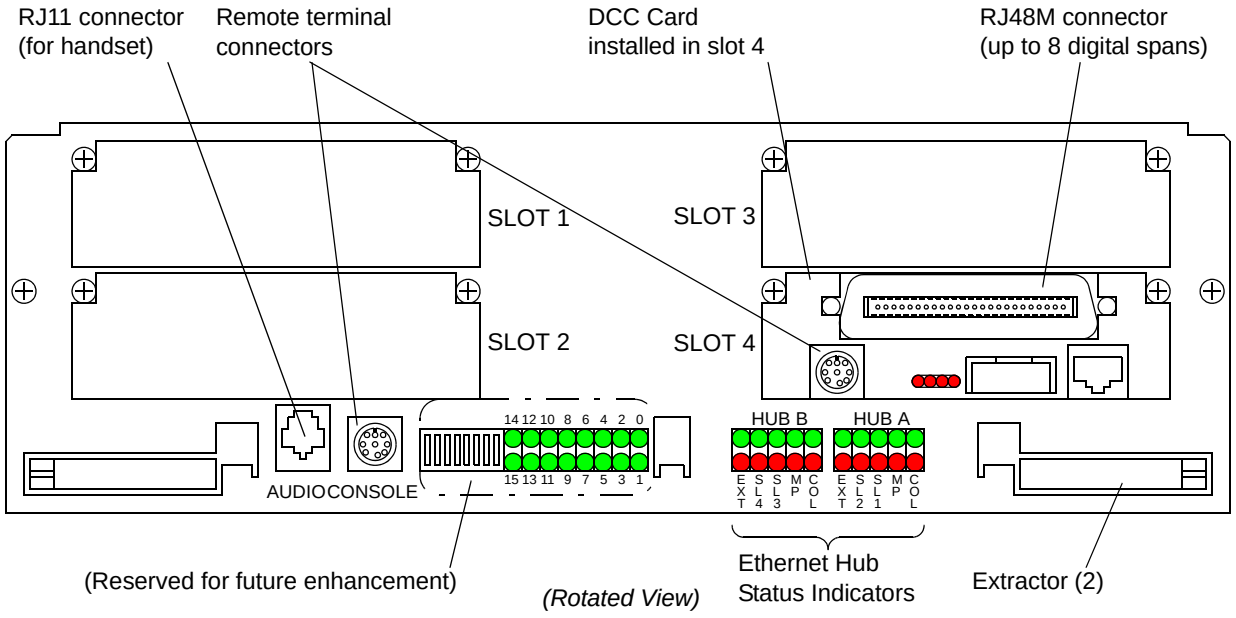


VRC Rear Panel

Item	Function
CHASSIS ID selector switch	Used to assign the chassis number. Each chassis in the system is assigned a unique number, starting at 0 and running consecutively.
EXT CLK A and **EXT CLK B** connectors	BNC connectors used to obtain synchronization reference clocks for the CT bus from an external source. The CT bus is the means by which voice and audio data is transmitted throughout the TMS.
TEST / **ON** / **OFF** switches (slots 5 and 6)	Three position rocker switch used to turn power to the slot on or off, or to test the associated power supply.
ALARM relays	Used to connect to external indicators for major and minor alarms. Common (C), normally closed (NC), and normally open (NO) contact connections are provided.
EXTERNAL SENSORS relays	Can be used to connect to external sensors for system indication of up to four separate events (A,B,C,D). **CAUTION: Voltage and current limitations. Do not connect to line level signals.**
PWR ON indicators (S5, S6, and NIC)	Illuminates **green** when power is applied to slot or NIC.
NORMAL indicators (S5, S6, and NIC)	Illuminates **green** when the module in the associated slot or NIC is operating properly and there are no alarms or system faults.
MINOR ALARM indicators (S5, S6, and NIC)	Flashes **yellow** when a minor alarm is active for the slot or NIC. Accompanied by audible indication (beep). To reset, press the RESET switch on the FCP. Use as a troubleshooting aid - alarms are promulgated to the Alarm Viewer (see ***Alarm Viewer*** on page 39).
MAJOR ALARM indicators (S5, S6, and NIC)	Flashes **red** when a major alarm is active for the slot or NIC. Accompanied by audible indication (beep). To reset, press the RESET switch on the FCP. Use as a troubleshooting aid - alarms are promulgated to the Alarm Viewer (see ***Alarm Viewer*** on page 39).
CSL connector	Console connector, reserved for future enhancement.
ENET A and **ENET B** connectors	RJ45 connectors used to connect the dual rail chassis Ethernet to the LAN.

TMS Front Panel

In addition to front panel indicators, the system components have internal indicators useful for system maintenance and fault isolation. Located on the front panel of the TMS are connections to a diagnostic console for troubleshooting, LED indicators for TMS health checks, and the span input/output connector.



TMS Front Panel Connectors and Indicators

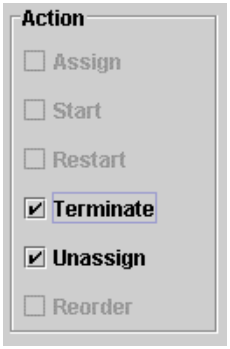
Item		Description
AUDIO jack		RJ11 jack for connecting a telephone handset for monitoring a phone line. (see ***AUDIO Monitor Jack*** on page 38).
CONSOLE jack		RS232 (9-pin DIN) port for connecting a debug terminal. Used for boot ROM and diagnostics (procedures which are typically handled by certified Avaya Field Engineers).
LEDs **0** - **15** and microswitches		Reserved for future enhancement.
HUB A/B LEDs (see ***Ethernet Hub Indicators*** on page 38).	**EXT**	Indicates condition on external port of hub A or B.
	SL1/2	Indicates condition on slot 1 or 2 port of hub A.
	SL3/4	Indicates condition on slot 3 or 4 port of hub B.
	MP	Indicates condition on microprocessor port of hub A or B.
	COL	Indicates errors or collisions in hub A or B.

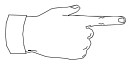
TMS Front Panel Connectors and Indicators (Continued):

Item	Description
DCC Front Panel	
Span connector	RJ48M 50-pin connector connects up to eight digital spans (T1 or E1) to PSTN.
Remote terminal connector	RS232 (9-pin DIN) port for connecting a debug terminal. Used for boot ROM and diagnostics.
Other connectors and indicators	Reserved for future enhancement.

AUDIO Monitor Jack

The AUDIO monitor jack is used to monitor a single phone line. A telephone handset is connected and the desired line can be monitored using the **`ccm listen`** command (see the *Avaya Media Processing Server Series System Reference Manual* for more information). Only one side of the conversation can be monitored at a time.

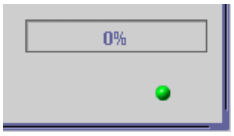




Use call monitoring (listening to actual calls) only to confirm that caller/Media Processing Server interaction is going smoothly and that the Media Processing Server is operating properly. Check applicable laws relating to privacy of conversations before listening to calls.

Ethernet Hub Indicators

The dual **green** and **red** LED indicators for **HUB A** and **HUB B** on the front of the TMS can display five separate conditions. The LEDs can either be lit continuously or blinking in the following combinations to indicate a particular condition.



Ethernet Hub Status Indicators

Indication	Conditions	
	EXT, SL*x*, MP	**COL**
Steady **green**	Receive link pulse	FIFO error
Blinking **green** (@~ 4Hz)	Transmit packet	(N/A)
Steady **red**	Reversed polarity	Collision
Blinking **red** (@~ 4Hz)	Receive packet	(N/A)
Alternating **red** and **green** (@~ 5Hz)	Partitioned out	(N/A)

Alarm Viewer

Media Processing Server Series software, applications, and components generate text alarm messages to indicate their status or to alert users to possible problems. Each component in the network stores its alarms in its default alarm log file. This allows viewing of “live” alarms on any component as the system generates them and viewing the alarm log file of any component.

Use the Alarm Viewer tool to monitor alarms. Filtering and Properties of alarm log files can be applied using the corresponding buttons at the top of the Alarm Viewer window or via the new daemon level filtering functions. Multiple alarm viewers, in any form (live or logged), can be launched at one time. This allows comparisons amongst components to be made.

For information on launching and using the Alarm Viewer (see ***Alarm Viewer*** on page 39). See the *PeriView Reference Manual* for a complete description of the Alarm Viewer. See the *Avaya Media Processing Server Series System Reference Manual* for information on the alarm daemons and corresponding filtering capabilities and refer to the *Alarm Reference Manual* for detailed description of the Alarm conditions.

Statistics Data Collection and Reporting

Typically, a single node in a Media Processing Server Series network is designated to collect and report statistical data. This statistical data can be used to monitor system and application level functions and to interpret various performance parameters and milestones (such as when and how calls are taken and handled or whether certain systems have experienced down time). PeriReporter is installed and resides on the node designated for statistics consolidation. Use PeriReporter to view and generate statistics reports.

System statistics are collected from each Media Processing Server by its VSTAT process. One VSTAT process exists for each Media Processing Server, and resides in that component’s VOS subcomponent. System variables (used in defining system reports) are predefined.

Application statistics are generated during an application’s execution cycle and collected by each node’s VSUPD process. One VSUPD process exists for each node, and resides in the GEN subcomponent of the common component on the node.

For details on how to configure report parameters, see the *PeriReporter User’s Guide* and (see ***System Statistics Reports*** on page 78).

Logging off a Node

One can log off a Solaris node by exiting the X-Windows environment. After log off, software operations cannot be changed, however, the Media Processing Server Series network *continues to run* without any manual input.

How to ... Log Off/Back Onto a Node

1. **Close any tools that are running**.
2. **Click <MENU> on the desktop** to pull up the Workspace menu.
3. **For OpenWindows systems, select** Exit...
   For CDE systems, select Log out...
4. The system prompts to confirm the exit. **Select Exit on OpenWindows systems, OK on CDE systems.**

The system returns to the desktop login screen.

To log back onto the Solaris node X-Windows environment:

5. **Select the desired desktop environment from the Options menu button** of the desktop login screen, or click OK to return to the previous environment setup. (For more information, see the related subordinate steps on page 13).

Avaya ships Solaris Media Processing Server Series systems with a *Sun* station as the node. *OpenWindows* and *CDE* are trademarked products presently used by Sun to simulate an X-Windows environment. If the environment is not a Sun workstation, events may vary from the above descriptions.

System Shutdown

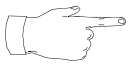
After the Certified Avaya personnel starts the system for the first time, there is rarely a need to shut down or restart the system. Most changes to the applications and MultiMedia files can be made while the system is in normal operation. However, the MPS may have to be shut down to upgrade the hardware or software, or relocate the system.

Use PeriView to stop the applications, then stop the TMS and Central Processing Unit (CPU).

How to ... Shut down the Media Processing Server

1. **Select all the Media Processing Servers**. Select the domain, then select the Application Manager (APPMAN).

2. **Select Terminate from the Action list.** Select the Terminate (soft terminate by default) and/or Unassign checkbox(es) in the Action item list. This allows any calls in progress to terminate normally or continue up until the value of the kill timer has expired.



Failure to terminate the applications softly can cause the IVR to hang up on callers currently dialed into the system.

3. **Select the APPLY button.** Wait for the Terminate/Unassign Lines Progress window to reach 100%.

4. **Close all tools that are running.**

5. **Shut down power to the TMS systems.** For locations of their power switches, ***Front Control Panel*** on page 35 and ***VRC Rear Panel*** on page 36

6. **In any command tool, log in as "root."**
   a. From the command line enter **`su`**.
   b. Enter the root password (_____).

7. **Enter `halt`**. This command has been modified by Avaya to perform a controlled shutdown, taking down system processes and functions in the proper sequence and timing.

8. **Wait for the ROM prompt (`ok>`)**.

If the **`halt`** command has been executed and the system does not respond, execute the **`halt.orig`** command instead.

9. **Toggle off the main power switches**.

Before powering off the system, if it is desired to restart it, enter **`boot`** at the ROM prompt and press <ENTER>. Proceed with the login instructions outlined at ***Logging off a Node*** on page 40. Remember to power up the TMS units if they've been shut down.

To restart the system after power down, see ***Routine Media Processing Server Startup*** on page 28. Normally and by default the Media Processing Server Series system is designed to automatically boot when powered up. If the system does not perform in this manner and stops at the ROM prompt:

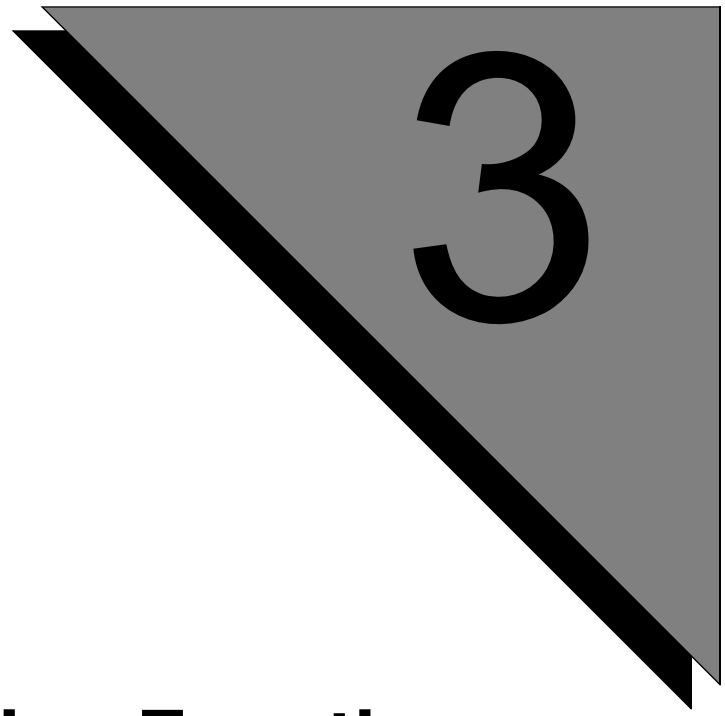
- log in as "root."
- enter **`su`**.
- enter the root password (_____).
- enter the **`reboot`** command.

Avaya has altered the **`reboot`** command to first perform a controlled shutdown, then bring the system up gracefully. A message will be displayed that the original Solaris **`reboot`** command has been renamed to **`reboot.orig`**.

The system memory can be "flushed" before rebooting by entering the **`reset`** command from the ROM prompt. This ensures that there are no processes still in memory prior to the system coming back up.

If there is a fire or some other emergency, *and time allows*, damage to the Media Processing Server can be limited by following ***step 5.*** through ***step 9.*** above and unplugging the unit. If the situation does not allow time for all the above (it can take up to five minutes), the system can simply be unplugged. If the emergency persists, it is wise to leave the area.

This page has been intentionally left blank.



Command Line Functions

This chapter covers:

1. **Command Line Interaction**
2. **Online Documentation**

Command Line Functions

This chapter provides the information needed to interact with the Avaya Media Processing Server, use the documentation package, and locate reference material.

Although most interaction with the system is through GUI tools like PeriView or MPS Developer, command line interaction is needed to issue certain status and call monitoring commands and to interact with the operating system. This section describes these command tool operations.

Command Line Interaction

A command line is the system command interpreter. It reads commands and interprets them as requests to execute tasks. Commands can be entered in the Media Processing Server VSH command line. Access a VSH command line by opening a command/shell tool on the Solaris desktop. The *PeriView Reference Manual* contains details about using the VSH Command Line.

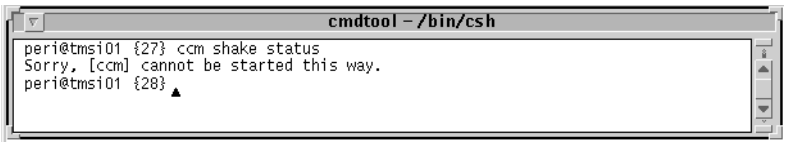
Access is available to several command lines. The two most important are:

Type	Default Prompt	Description	Use for:
Media Processing Server shell	`vsh#<component_type>.<component_number>/<node_name> {N} ->`	Avaya command shell	• all Solaris commands to the local node • all component/ Media Processing Server Series commands
Command Tool	`<username>@node_name {N}` (or, if `$HOME` for user is not set...) `node_name%`	Solaris command shell (`csh`)	• all Solaris commands to the local node • starting component tools

See the Media Processing Server Series *Command Reference Manual* for a detailed explanation of the MPS command shell, and the Solaris system documentation for a detailed explanation of the Solaris command shell.

It is possible to move between command windows at any time. For information see ***Command Control*** on page 50.

If a Media Processing Server Series command is entered in a Solaris command tool, the system sends an error message and returns to the prompt. For example, if **`ccm shake status`** is entered at a Solaris command shell, the following results:



Command Line Editing

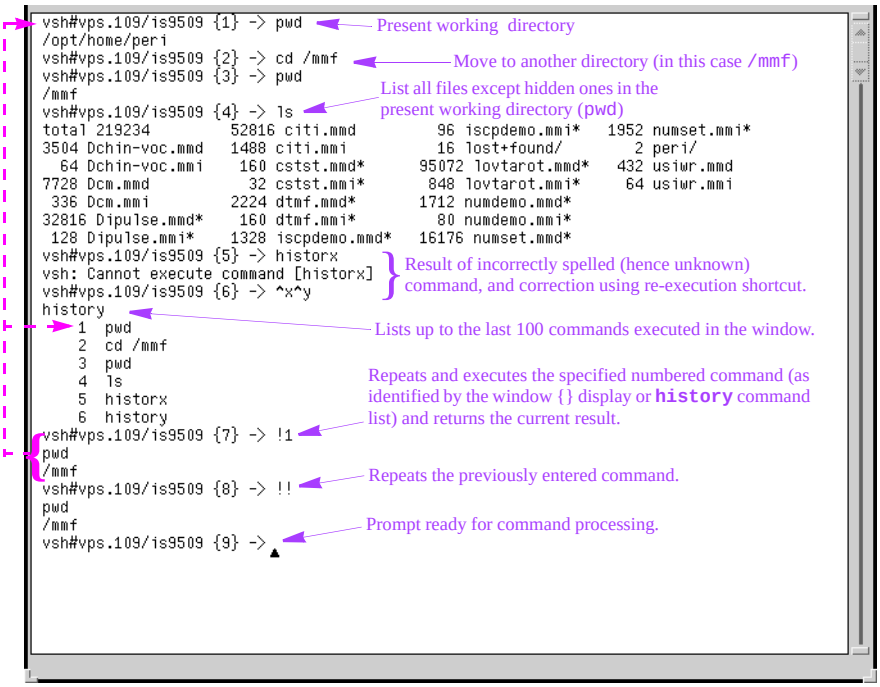
The Media Processing Server Series uses standard C-shell (csh) conventions for command line editing. This section is a quick look at some common command line editing techniques. See a Solaris guide for a comprehensive discussion of these conventions.

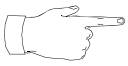
In the default configuration mode, the Media Processing Server shell prompt is: vsh#<component_type>.<component_number>/<nodename>{N}-> where <component_type> is the current component type, <component_number> is the current component number, <nodename> is the local node, and N is an up to three-digit number the system increments for each command entered, if no prompt is visible, press <RETURN>. This should cause the Media Processing Server to display a prompt (unless the system is already performing some other action and is not available for immediate command processing).

The following information pertains to command line editing:

- The system only receives and executes console commands after the <RETURN> key is used.
- The system maintains a command history buffer that contains the last 100 commands executed. Display this buffer to the console by entering **history** at any command line.
- Commands can be executed from the history buffer by using the number of the command preceded by an "!". The command number is in the prompt within the "{ }" braces. For example, **!95** repeats command number 95.
- The command number increments for each new command entered at the console. After more than 100 commands have been entered, the system removes the oldest command from the history buffer.
- Repeat the last command executed by entering **!!**.
- Change any character string in a previous command to a different string by entering **^oldstring^newstring** at the command prompt. If the last character in newstring is a space, use a concluding ^. For example, **^24^25** changes the first occurrence of string 24 in the last command to 25 and re-executes the command with the new value. Also, if the command **cp<oldname> <newname>** is entered (note the missing space), enter **^cp^cp ^** to correct the command.

The following is a sample console interaction that uses the basic command editing functions. If viewing this manual on-line, click on a description below to find out more about the corresponding command.





Commands can be entered in the Media Processing Server shell using the VSH Command Line. For information about using the VSH Command Line, see the *PeriView Reference Manual*.

Command Control

Individual control can be exercised over each component in the network, as well as over how each component executes the commands entered. Components are outlined at see ***System Configurations*** on page 19..

Issuing Commands to a Component

Issue commands to any component in the network through the command line for that component. The Media Processing Server shell command line may be accessed in one of two ways: through the VSH Command Line.

To access the Media Processing Server shell from a Solaris shell, **`rsh`** to any component node, then enter **`vsh`** and press <RETURN>. If this is done at a Media Processing Server node, the shell prompt of the lowest numbered component displays. Select the desired component to elicit its shell prompt.

The default Media Processing Server shell prompt indicates the current component type and component number (that is, the component that is local to the node) as well as the node from which the tool was launched. If more than one component is configured for the node, the initial prompt displays the component with the lowest number. For example, if the prompt is `vsh#mps.109/is9509 {1} ->`, then `mps.109` is the current component and `is9509` is the name of the node.

The **`comp`** command identifies the currently configured components along with their status. "`Local`" indicates the component is connected to this node. "`Remote`" indicates the component is connected to another node in the network. See the *Media Processing Server Series System Reference Manual* for a detailed discussion of the Media Processing Server Series network.

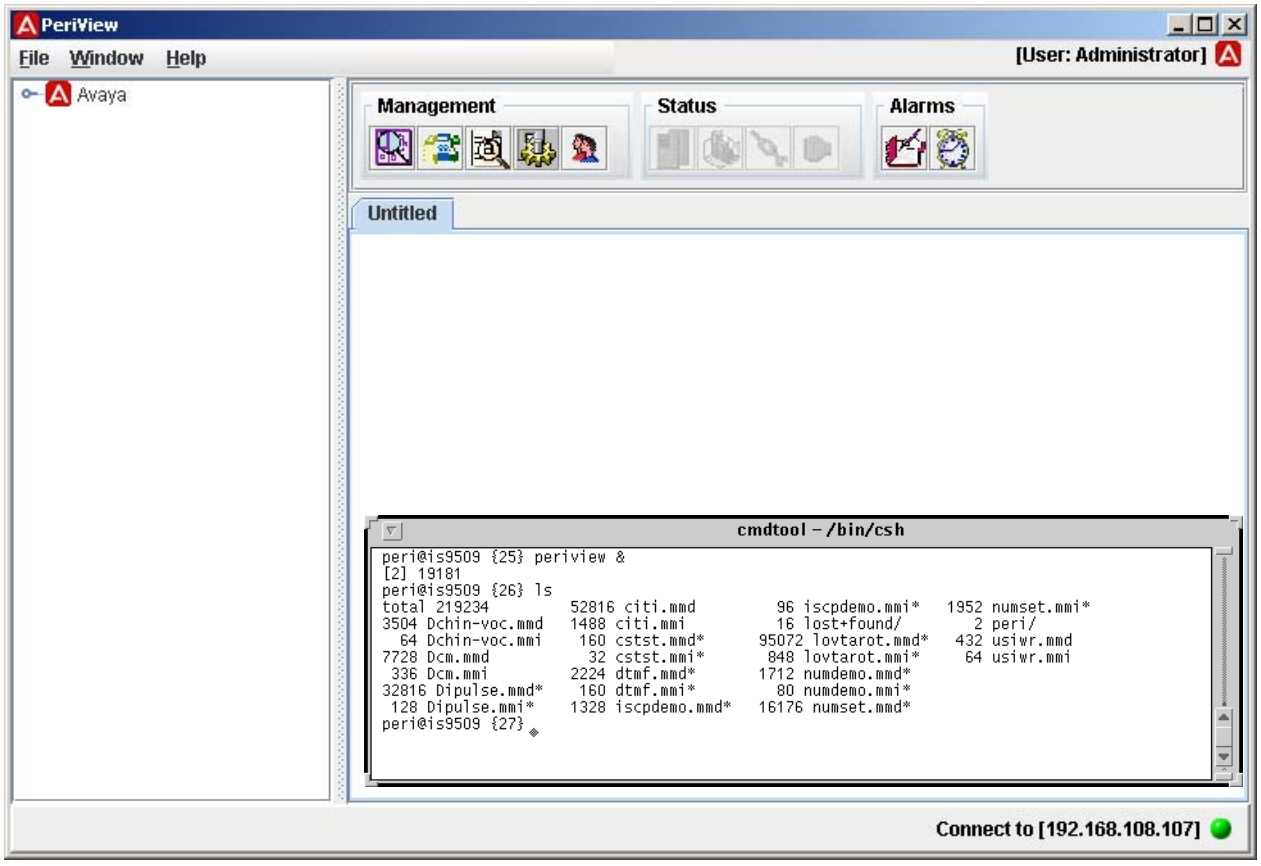
```
vsh#mps.109/is9509 {1} -> comp
Configured components are:

        [1]     #common.0/is9509                <local/up>
        [2]     #common.0/grouse                <remote/up>
        [3]     #mps.109/is9509                 <local/up>
        [4]     #vas.109/grouse                 <remote/up>
        [5]     #tmscomm.109/tms1000            <local/up>

Enter number (displayed in []) of component desired ->
```

All Solaris commands entered in a Media Processing Server shell are issued to the local node *regardless of the current component*. For example, if the current component is `vas.1` and `grouse` is the name of the current node, but the VSH Command Line was launched on node `is9509`, **`ls`** lists the files in the directory on `is9509`, not on `grouse`. To identify the local node when connected to a component remote to that node, enter the **`hostname`** command at the prompt.

Foreground and Background Commands

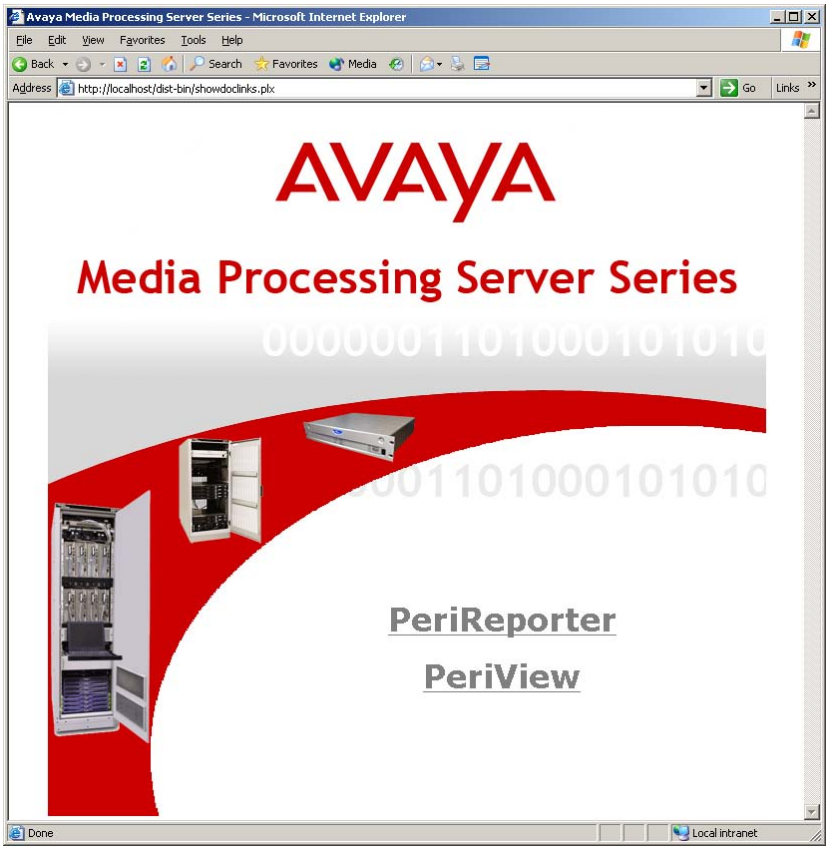
The Media Processing Server handles command execution like a standard operating system. That is, the system executes commands in the order they are entered. When entering standard commands, the Media Processing Server runs each command to completion before it displays a prompt for the next command.

Commands can also be executed in the background by following the command with "**&**". In this mode, the Media Processing Server takes the input and begins processing, but does not wait for the command to complete. Instead, it immediately displays a prompt. For example, PeriView can be launched in the background and then another command can be immediately issued.



Online Documentation

Avaya documentation is available on-line in PDF format. This documentation is accessed via a Web browser by entering the address, `http://localhost` and clicking on the Documentation hyperlink.



4

Status Monitoring

This chapter covers:

1. **The Alarm Viewer**
2. **Status Monitoring**
3. **System Statistics Reports**

Status Monitoring

This chapter provides the information needed to:

- Audit overall system performance
- Monitor component functionality
- Monitor phone line events
- Report system statistics

Although most interaction with the system is through GUI tools like PeriView or MPS Developer, command line entries are needed to issue some status and call monitoring commands. This section also describes these operations. For specifics concerning interaction with the system as a whole, see ***Command Line Interaction*** on page 47..

The Alarm Viewer

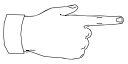
The system software processes of the Avaya Media Processing Server Series generate text alarm messages that indicate their status or alert users to possible problems. Each component in the network stores its alarms in its default Alarm Log File. This allows:

- viewing of alarms on any component as the system generates them (live alarms), and
- viewing of the alarm log file of any component.

The following steps explain how to use the Alarm Viewer tool to monitor alarms.

How to ...View Alarms

1. **Start PeriView.**



Before launching the Alarm Viewer, use the Alarm Filter to determine which alarms to view.

2. **Start the Alarm Filter.**
   a. <LEFT> click on the Alarm Filter icon.

Define the alarms that actually display in the Alarm Viewer by setting the filtering properties. Alternatively, load a previously saved viewer configuration file (contains filter definitions). See the *PeriView Reference Manual* for a complete description of the Alarm Filter.

b. Select Choose Log in the Alarm Filter window to select user defined Historical Alarm Log files.

Specific alarm log files must be selected before they can be displayed in the Alarm Viewer.

3. Start the Alarm Viewer.

a. <LEFT> click on the Alarm Viewer icon.

Live alarms display as the generated alarms are received by the Alarm Viewer from alarm daemons.

Default alarm log files display in the order in which they are selected in the Logged Alarms Filter. See the *PeriView Reference Manual* for a complete description of the Alarm Viewer.

Alarms are generated by the **`alarmd`** daemon. If alarm filtering has been instituted at this level or that of **`alarmf`** (an intermediate filter daemon), the Alarm Viewer *only* receives those alarms that pass the filtering criteria. For detailed information see the *Media Processing Server Series System Reference Manual*.

see ***To Monitor Components*** on page 59.

The log file for each component can also be accessed from the Media Processing Server shell. Both the default log file and its backup are in the directory `$MPSHOME/common/log`, where the files for the specific components are identified in the format `alarm.component_type.component_number.log`. The default log is `.log` and its backup is `.log.bak`. The `.log` file is created when the system boots. The `.log.bak` file is created when the preconfigured file size is exceeded. Use standard commands or editors to view and/or edit these files.

Refer to the *Alarm Reference Manual* for detailed description of the Alarm conditions, error causes, and remedial actions.

The alarms that the Media Processing Server generates can be customized by modifying records in the alarm database, located in `$MPSHOME/common/etc`. Additional information concerning the alarm database can also be found in the *PeriView Reference Manual*.

Status Monitoring

The Media Processing Server has many built-in functions for monitoring system and line status. With these functions, the status of components, spans, external host(s), phone lines, MMF files, applications, the Call Control Manager (CCM), the Voice Memory Manager (VMM), and the Startup and Recovery Process (SRP) can be easily monitored. Standard commands can also be used to display and/or modify the system date and time.

System Date/Time

To check the system date and time, enter **`date`** at any Solaris or Media Processing Server shell command line. The super-user can use **`date`** to modify the system date and time. See the Solaris system documentation for more information.

Although it is not necessary to alter the system date for daylight savings time, users are often unaware of this. In this regard, it is not recommended to change the system time by issuing the **`date`** command directly. If the time is set back by issuing the **`date`** command, **`cron`** will have difficulty performing scheduled tasks, all of which will become queued up until **`cron`** itself terminates. Evidence of this can usually be detected in the `/var/cron/log` file. If the system time is adjusted to a point prior to the start time of **`cron`**, its jobs are not rescheduled correctly. The correct method to alter the system date is to stop **`cron`**, issue the **`date`** command, and restart **`cron`** as shown below:

```
/etc/rc2.d/S75cron stop
date <new_date/time>
/etc/rc2.d/S75cron start
```

If the time or date on the system was recently changed, be sure that the clocks on all the communicating systems are synchronized. Non-synchronized system times can adversely affect statistics collection.

Solaris Management Console

Solaris Management (SMC) Console provides a scheduler GUI. This allows to set up **cron** jobs through the use of a GUI instead of editing the **crontabs** file manually.

Follow the below steps to edit a list of existing **crons**:

- On the console, (as any valid user such as peri or root) open a command window and execute: **`/usr/sadm/bin/smc`**
- Allow several minutes for the SMC to initialize and configure itself.
- At the welcome screen, expand "`this computer`" icon.
- Select the "`services`" option.
- Enter the user in the login prompt (this is the cron user you plan to manage—peri or root)
- The existing **crons** will be listed and you can add/delete/modify the jobs as

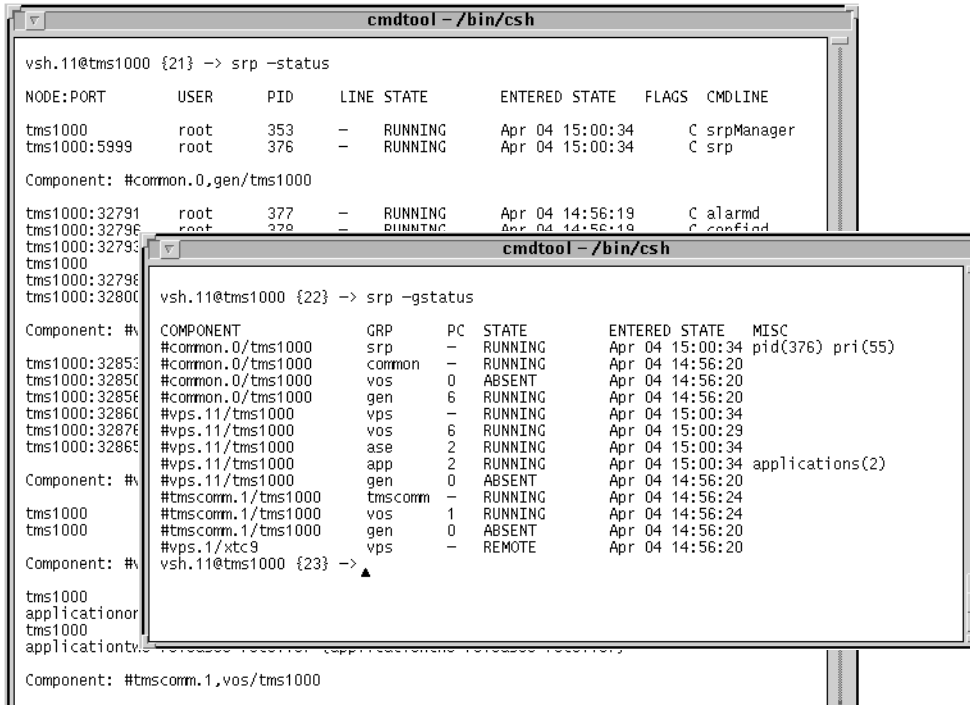
desired.

SRP Status

The Startup and Recovery Process (SRP) functions as the parent of all MPS system processes. SRP spawns the relevant programs at system startup, performs watchdog duties for each TMS and VOS process, acts as a topology database server; spawns and terminates VENGINE applications, and restarts abnormally terminated programs.

The status of SRP can be obtained (as well as that of other processes and applications) or process groups and applications at a VSH command line (on a particular node) with the following commands:

Command syntax	Report
srp -status	SRP status for the (specified) node
srp -gstatus	SRP group status for the specified node



The *PeriView Reference Manual* contains details about using the VSH Command Line. See the *Media Processing Server Series System Reference Manual* for detailed descriptions and examples.

Component Status

The status of components can be considered the highest level of overall system monitoring. A component status of “down” can be indicative of more serious problems somewhere within the system. These problems can then be located using

more specific means (see the paragraphs that follow as well as the *PeriView Reference Manual*.

How to ... Monitor Component Status

From the Command Line

The most generalized means of checking overall component status is through use of the VSH command line.

- **`rsh`** to the node that contains the component(s) to be monitored (use the **`vsh`** command to access the VSH command line).

- Enter **`comp`** and press <RETURN>. This command identifies the currently configured components along with their status.

Using PeriView

Components supported by PeriView include MPS, OSCAR, MTS, CCSS, CSS, VAPI, RTR, SNTRY, DCS, CTXCOMM, and TMSCOMM (for an overview of these components see the *PeriView Reference Manual*). The means of monitoring each varies, however.

Basic information that applies to monitoring of all components in PeriView includes interpretation of colors resident on component icons in relation to component state. State is a representation of a PeriView tool attempting to contact some Media Processing Server Series process. In the case of tree objects, PeriView tools (Application Manager and Status tools) attempt to contact SRP (Startup and Recovery Process). Until contact is confirmed, the state of the component is defined as **Unreachable**, as no information is available. **Unreachable** components display as **black** objects in the workspace (assuming the tree is expanded to display its components). A component will be considered **Unreachable** for the window of time that it takes to return a response to the tool requesting either *initial* contact or any contact after contact is lost.

Components acknowledge contact with the tool on an individual basis. A component is no longer considered **Unreachable** when it returns information to the tool about its state, which will be either **Up** or **Down** according to SRP. Components in an **Up** state (identified by SRP as **RUNNING**) display as **green** objects. Components in a **Down** state (identified by SRP as anything other than RUNNING) display as **red** objects.

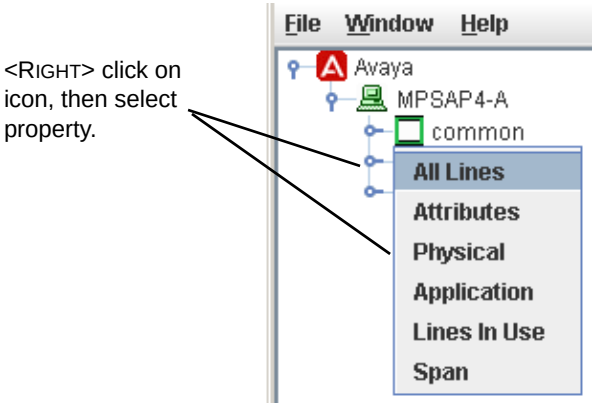
Components are represented by icons at the tree level. The following are examples of these icons:

MPS	tmscomm	OSCAR
mps310 mps311 mps312 mps313	tmscomm1	oscar12

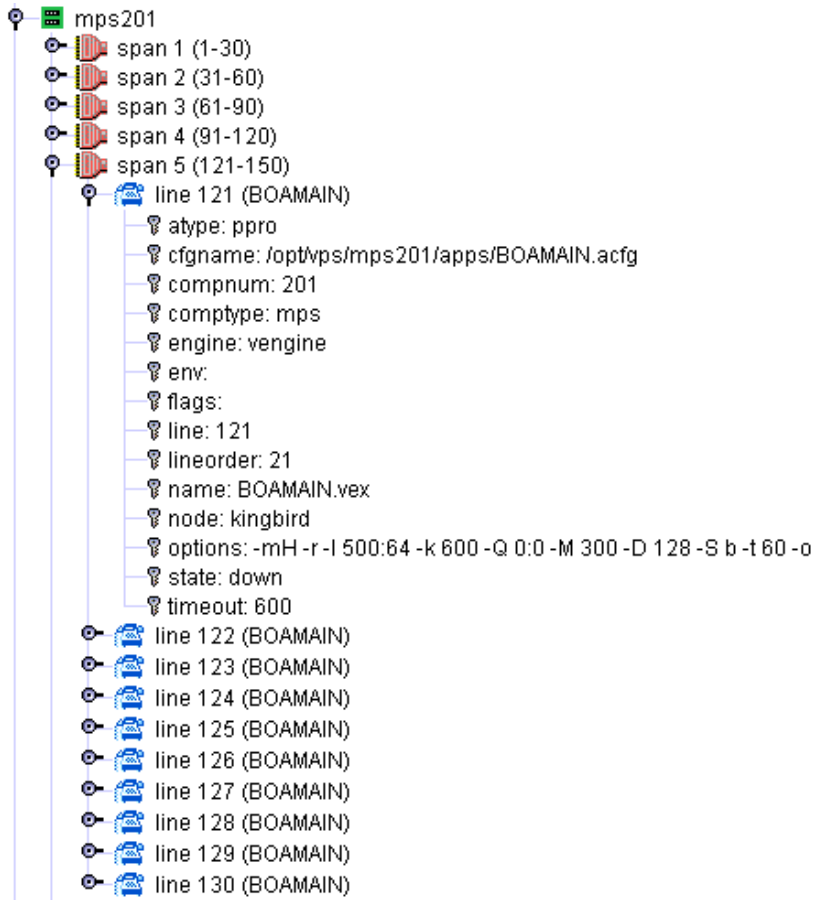
For complete information on how colors, icons, and states apply to PeriView, and for further information regarding SRP, see the *PeriView Reference Manual*.

To Monitor Components

1. **Select the Desired Component.** <RIGHT> click on the icon next to a component to display various properties. Properties are different depending on the component type selected. The following represents a list of properties for an MPS.



Multiple pieces of information about a component can also be displayed. For example, selecting a components span, then the spans lines in use, then the lines attributes produces the following output.



For complete information on this and other items concerning PeriView, see the *PeriView Reference Manual*.

Phone Line Status

The status of the phone lines can be displayed with the Line Status tool.

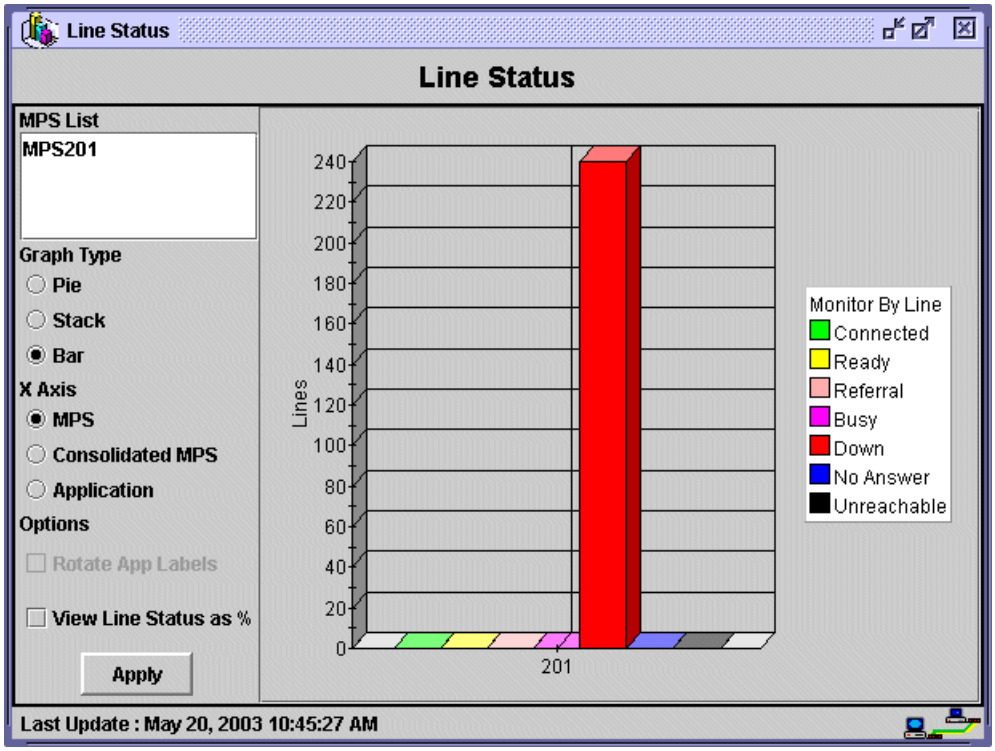
How to ... Monitor Phone Line Status

Using PeriView

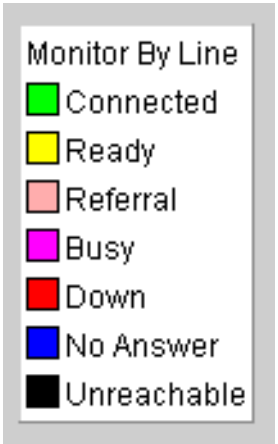
Launch the Line Status tool. Select a node or component from the PeriView navigation pane, then select the Line Status tool.

The Line Status tool, shown below, displays the physical state of phone lines for a

specific period of time. Phone line status indicates what is actually happening on the line during a time period. The current phone line status for MPSs and applications can be monitored.



Phone line status is defined by seven color coded activity states, which display along the y-axis of each graph.



The Line Status tool quantifies and displays each the following phone line states: **Connected**, **Ready**, **Referral**, **Busy**, **Down**, **No Answer**, and **Unreachable**. These six phone line states are defined in the following table.

Table 1:

Phone Line State	Color	Description
Connected	Green	The monitor detects ongoing interaction between the or MPS's phone lines and callers, indicating calls are in progress on these lines.
Ready	Yellow	The monitor detects the or MPS's phone lines are in service and ready to take calls.
Referral	Orange	Detected activity is dependent upon the type of referral method that has been instituted—either a hook flash or digital referral method.
Hook Flash Referral		The monitor detects the MPS's phone lines are in the process of referring the call to an operator.
Digital Referral		Activity can be in one of the two following states: • The lines are in the process of referring to operators. • The callers are interacting with operators
Busy	Purple	The monitor detects the MPS's phone lines are in an in-service-busy state and not able to take calls.
Down	Red	The monitor is able to contact the or MPS's phone lines, but detects the lines are not able to receive calls, indicating that these lines can be faulty.
No Answer	Blue	The line is Idle and the state is set to no answer.
Unreachable	Black	Phone line status has been queried for its state and no message has been received by the Monitor (basically, a missed response). Also, it could mean that SRP did not respond to information about the phone line from the Monitor. The Monitor needs to query SRP about how it should contact the CCM (Call Control Manager) process, before it can actually contact CCM.

Span Status

Span status can be monitored from the command line. Monitoring can also be accomplished through the PeriView Span Status tool, which provides graphical representations of various aspects of a digital span.

How to ... Monitor Span Status

From the Command Line

To determine span status from the command line, first switch to the node on which the component that contains the span(s) to be checked is located (see ***Issuing Commands to a Component*** on page 50.). Next, enter the command **`dlt`** and select the component that contains the span(s) from the list of configured components (this list only displays if there is more than one local component other than `common` or `tmscomm` configured on the node). After the node has connected to both `tcad` and `trip` (two processes needed to develop the information), enter **`spanstatus <n>`**, **`<n-n>`**, or **`<all>`**, where **`<n>`** is the number of the span in question, **`<n-n>`** is a series of spans, and **`<all>`** is all spans configured on that node. The span status report displays as follows:

```
vsh.20tmsi02 {2} -> dlt
Configured components are:

       [1]    #common.0/tmsi02       <local/up>
       [2]    #vps.1/tmsi02          <local/up>
       [3]    #vps.2/tmsi02          <local/up>
       [4]    #tmscomm.1/tmsi02      <local/up>

Enter number (displayed in []) of component desired -> 3

Default component set to [#vps.2/tmsi02]

dlt#vps.2/tmsi02 {1} ->

Attempting reconnect to trip

dlt#vps.2/tmsi02 {1} ->

Attempting reconnect to tcad

dlt#vps.2/tmsi02 {1} ->

Connected to tcad
dlt#vps.2/tmsi02 {1} ->

Connected to trip

dlt#vps.2/tmsi02 {1} -> spanstatus 1
dlt#vps.2/tmsi02 {2} ->

Span Type:          T1 SF CAS
Span Status:          Out of service due to network errors
Alarm State:          RED
Alarms Received:      RAI
Alarms Transmitted:   RAI
Span Conditions:      LOS LOF RAI UAS FE
Loopback:             Disabled
Source of Timing:     No
Number of Channels:   24
Channels Online:      ALL

dlt#vps.2/tmsi02 {2} -> spanstatus 1-5
dlt#vps.2/tmsi02 {3} ->
-------------------------------------------------------------------------------
    Alarm           Time  Span  Alarms     Alarms     Span
Sp  State  Status   Src   Type  Received   Xmitted    Conditions
-------------------------------------------------------------------------------
1   RED    OutOfServ  No   T1    RAI        RAI        LOS LOF RAI UAS FE
2   GREEN  InService  No   T1    (None)     (None)     (None)
3   GREEN  InService  No   T1    (None)     (None)     (None)
4   GREEN  InService  No   T1    (None)     (None)     (None)
5   GREEN  InService  No   T1    (None)     (None)     (None)

dlt#vps.2/tmsi02 {3} -> spanstatus all
dlt#vps.2/tmsi02 {4} ->
-------------------------------------------------------------------------------
    Alarm           Time  Span  Alarms     Alarms     Span
Sp  State  Status   Src   Type  Received   Xmitted    Conditions
-------------------------------------------------------------------------------
1   RED    OutOfServ  No   T1    RAI        RAI        LOS LOF RAI UAS FE
5   GREEN  InService  No   T1    (None)     (None)     (None)
2   GREEN  InService  No   T1    (None)     (None)     (None)
3   GREEN  InService  No   T1    (None)     (None)     (None)
4   GREEN  InService  No   T1    (None)     (None)     (None)
6   GREEN  InService  No   T1    (None)     (None)     (None)
7   GREEN  InService  No   T1    (None)     (None)     (None)
8   GREEN  InService  No   T1    (None)     (None)     (None)

dlt#vps.2/tmsi02 {4} ->
```

Using PeriView

Span Status is available for each digital Media Processing Server defined for the domain in the Activity Monitor's workspace. A Span Status tool can be launched for a single Media Processing Server or for multiple Media Processing Servers; a window

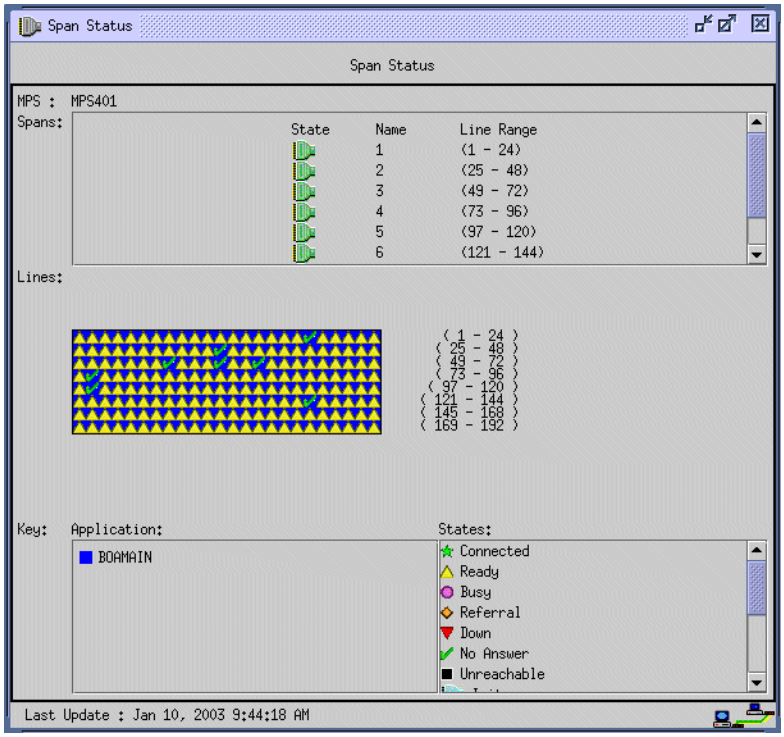
opens for each selected.

The information applies to digital phone lines only and is available for Media Processing Servers with either all digital lines, or a combination of digital and analog lines. However, the information does not apply to the analog phone lines.

To launch the Span Status tool:

Launch the Span Status tool. Select a component from the PeriView navigation pane, then select the Span Status tool.

Use the Span Status tool to display the state of digital spans (physical phone lines), graphically, on a per component basis. Span Status is available for each component defined for the domain. This is a display tool only.



Span status is updated, by default, at 15 second intervals. This interval is specified by the PeriView Data Provider software.

Span status displays only for (digital) physical phone (call processing) lines, which are capable of processing calls. Span status does not display for logical (administrative) phone lines, which are unable to process calls and to which administrative applications are, typically, assigned. Therefore, administrative applications are not

listed in the Applications scrolling window, unless they are assigned to physical lines, which are digital.

The Span Status displays the component name and state; the name of the spans (`A`, `B`, `C`, etc.) and their states; and the phone lines and applications that are associated with each span.

Span dividers are lines that identify the boundaries between individual spans in the Lines display area. When you position the cursor over a line, the phone line number displays under the Lines display area.

The date and time of the last update of information to this window displays in the lower left corner of the Component Span Status window.

The Component Span Status tool displays the following information.

- Component:
  The node name and component number of the component used to launch the Component Span Status tool displays in the format `nodename:comp_number`. In this example, `is9501:101` (MPS `101` on node `is9501`) was used to launch the tool. The current state of the component (**Up**, **Down**, **Unreachable**) is identified by the color coded symbols defined in Key area of the window, labeled States.

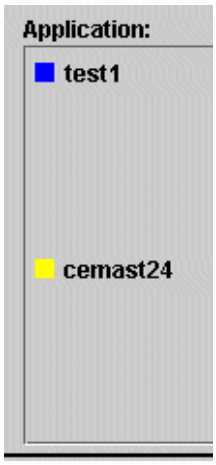
  When you launch Component Spans status with a component configured only with analog phone lines, the Span Status window indicates some information about the component. However, the message in the Spans display area indicates that no spans have been detected. In the following illustration, the Span Status tool was launched with an analog MPS (`is7501:1`).

- Spans:
  Span information indicates the span state, span name, and phone line range associated with the span. The current state of the span (Red Alarm, Purple Alarm, Blue Alarm, Yellow Alarm, Unreachable, Up, Disabled, Unknown) is identified by the color coded symbols defined in Key area of the window, labeled States.

- Lines:
  The Lines display area identifies information about the spans that are associated with phone lines. Span dividers are the black lines that define the boundaries between the individual spans. When the cursor is over an line, the line number displays under the Lines display area.

  The current phone line state is indicated by the color coded symbols defined in Key area of the window and labeled States.

The applications associated with a line (if any) are also identified by color coded symbols defined in the Key area of the window, labeled Applications.



- Key:
  The Key display area is used to associate the color coded symbols for Applications and States with the Span Status display.

  - Applications:
    The Applications display scrolling window identifies all applications (if any) that have been assigned to the component. Each application is associated with a specific color. Each line to which an application is assigned displays with a colored background, which matches one of the application colors displayed in the Applications scrolling window. The application names and the phone lines to which they are assigned can be identified by matching the color sets.

    Because span status displays only for (digital) physical phone lines, administrative applications are not listed in the Applications scrolling window, unless they are assigned to (digital) physical lines.

- States:
  The States display scrolling window identifies the range of possible component, Span, and Phone Line states. These symbols are used to indicate the current state of the component, each span, and each of the digital, physical phone lines associated with the spans in the Lines display area. Phone Line, Span, and component states are defined in the following table.

Table 2:

Component Span Status States *Sheet 1 of 2*

Symbol	State	Applies To	Description
	Connected	Phone Line	Phone line state in which interaction is detected between the MPS's phone lines and callers, indicating that calls are in process on these lines.
	No Answer	Phone Line	The line is Idle and the state is set to no answer.
	Referral	Phone Line	Detected activity is dependent upon the type of referral method that has been instituted (hook flash or digital).
	Busy	Phone Line	Phone line state in which the MPS's phone lines are in an in-service-busy state and not able to take calls.
	Down	MPS	One or more component-specific processes have either been interrupted or are not yet fully operational.
		Phone Line	Phone line state in which the MPS's or MPS's phone lines are not able to receive calls.
	Unreachable	Span	The state of the span is unknown because the span is unreachable.
		MPS	The state of the MPS is unknown. No information is available. Once information is available, the state of the MPS is either **Down** or **Up**.
		Phone Line	Phone line state in which the MPS cannot be contacted therefore, phone line state cannot be determined.
	Ready	Phone Line	Phone line state in which the MPS's phone lines are in service and ready to take calls.

Table 2:

Component Span Status States *Sheet 2 of 2*

Symbol	State	Applies To	Description
	Up	Span MPS	The Span is in service. All MPS-specific processes are fully operational.
	Disabled	Span	The span has been intentionally disabled from the MPS.
	Yellow Alarm	Span	The span is in a yellow alarm state. The span is receiving a yellow alarm from the remote end. This indicates that the far end is not able to receive your signal but you are receiving a signal from the remote end. The span requires attention.
	Red Alarm	Span	The span is in a red alarm state, which indicates that the span is *out of sync*. A red alarm indicates the span has lost the ability to synchronize with the remote equipment. The span has, effectively, lost the incoming signal. The span requires attention.
	Blue Alarm	Span	The span is in a blue alarm state. The span is receiving a blue alarm from the remote end. This means you are receiving a pattern of all ones. Typically, a blue alarm indicates that the remote end is in a maintenance mode.
	Purple Alarm	Span	The span is in a purple alarm state. The span is receiving a purple alarm from the remote end. This means you are receiving a pattern of all ones on channel 16). Typically, a purple alarm indicates that the remote end is in a maintenance mode.
	Unknown	Span	The state of the span is unrecognized. It does not comply with the definition of any other State listed in this scrolling window.

CCM Status

The Call Control Manager (CCM) is the primary interface between applications and VOS services and is the gateway for communication with the telephony hardware. A status report for the system, a line, or a range of lines on a Media Processing Server VSH command line can be obtained using the following commands:

Command syntax	Report
`ccm status` **`ccm st`**	Overall CCM status for the system, including process startup, work thread statuses, and the state of each line CCM knows about.
`ccm {phone:<line#>} status` **`c<line#> st`**	CCM status for the specified line, including line states, CCM related configuration parameters, and queue sizes.
`ccm {phone:<line#-line#>} status` **`c<line#-line#> st`**	Replicates CCM line status reports as above for each of the specified range of lines

These commands can be entered in the Media Processing Server shell using the VSH Command Line. For information about using the VSH Command Line, see the *PeriView Reference Manual*.

The version of CCM that is running on the system can be obtained by entering the command **`ccm version`** in a Media Processing Server VSH command line.

Application Status

Use the APPMAN (Application Management) Status tool to check the status of any application assigned to a Media Processing Server. For information about working with applications, see ***Voice Processing Applications*** on page 84..

How to ... Check the Status of Applications

From the Command Line

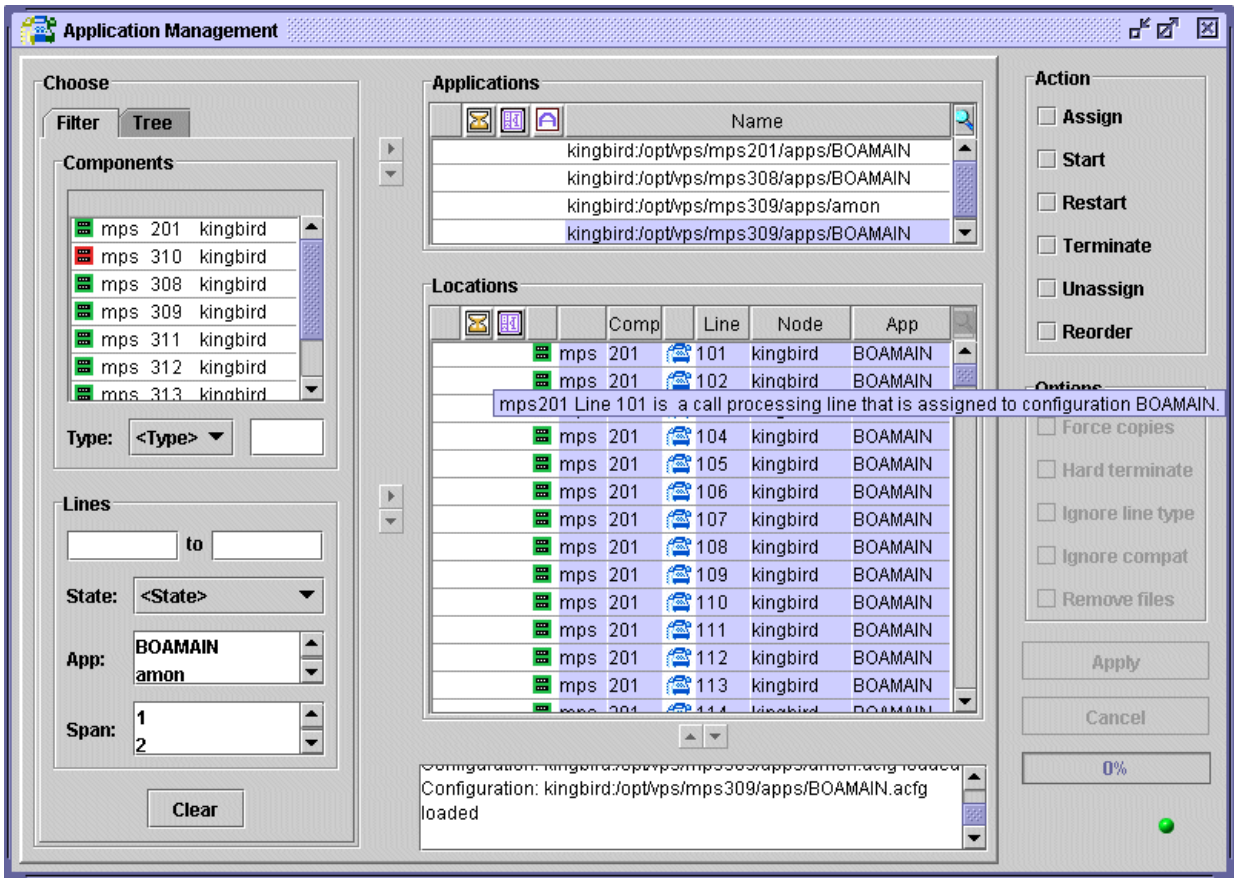
The status of applications is obtained by entering the command **`vmm appstatus`** at a VSH command line.

Using PeriView

1. **Launch the Application Management Tool**.

Select a component from the PeriView navigation pane, then select the Application Management tool.

Icons displayed in the Applications and Locations scrolling windows are cues to application, phone line, and component information. These icons symbolize: administrative applications that are configured to execute completely before other applications are started; linked applications; applications configured not to start at reboot; coded components; and, coded call processing and administrative phone lines. The color of component and application icons indicate their state.



The shape of phone line icons indicates phone line type. The color of phone line icons indicate the state of the application. This information is summarized in the following tables.

Table 3:

Shape	Description
	Call Processing (Physical) Line; capable of processing calls. Call processing applications are assigned to these lines.
	Administrative (Logical) Line; incapable of processing calls. Administrative applications are assigned to these lines.

Table 4:

Color	Description
Green	An application is assigned to a phone line and is in the process of executing (running).
Blue	An application was assigned to a phone line but it is not currently executing (not running). The application: • was assigned but not started, • has completed its execution cycle, or • has terminated but has not been un-assigned from the phone line.
Yellow	The phone line has been selected, but no application has been assigned to the phone line.

The Locations window lists all configuration options for the lines associated with the selected component(s). Placing the cursor over a particular location, indicates the status of the parameters for the application. In addition, the following icons may appear to indicate configuration selections.

Table 5:

Shape	Description
	Administration applications that are configured to Wait for Exit display with an initialization icon. Wait for Exit is an Execution option that is configured by the Configure Application tool.
	Applications that are configured not to restart during reboot display with a blue (no restart) application icon. Start on Reboot is an Execution option configured by the Configure Application tool.

See the *PeriView Reference Manual* for specific details regarding the Application Management and Application Configuration tools.

VMM Status

The Voice Memory Manager (VMM) is a process that resides in the VOS subcomponent of the Media Processing Server component and provides media

management services for the VOS. The VMM status report indicates initialization and connection states and static and dynamic thread configurations, if any. This report can be obtained from a VSH command line with the following command:

Command syntax	Report
`vmm status or vmm st`	Overall VMM status for the system

The version of VMM running on the system can be obtained by entering the command **`vmm version`** at a VSH command line.

MultiMedia Format (MMF) File Status

The Media Processing Server uses MMF files to store audio elements and fax data. The audio elements can be:

- Vocabularies (i.e., spoken prompts in applications)
- Items recorded by an application (i.e., the Caller Message Recording [CMR] feature - see ***Caller Message Recording*** on page 117.)

How to ... Check the Status of Audio Elements

From the Command Line

Enter the following commands in a Media Processing Server VSH Command Line to display a variety of status reports for MMF files.

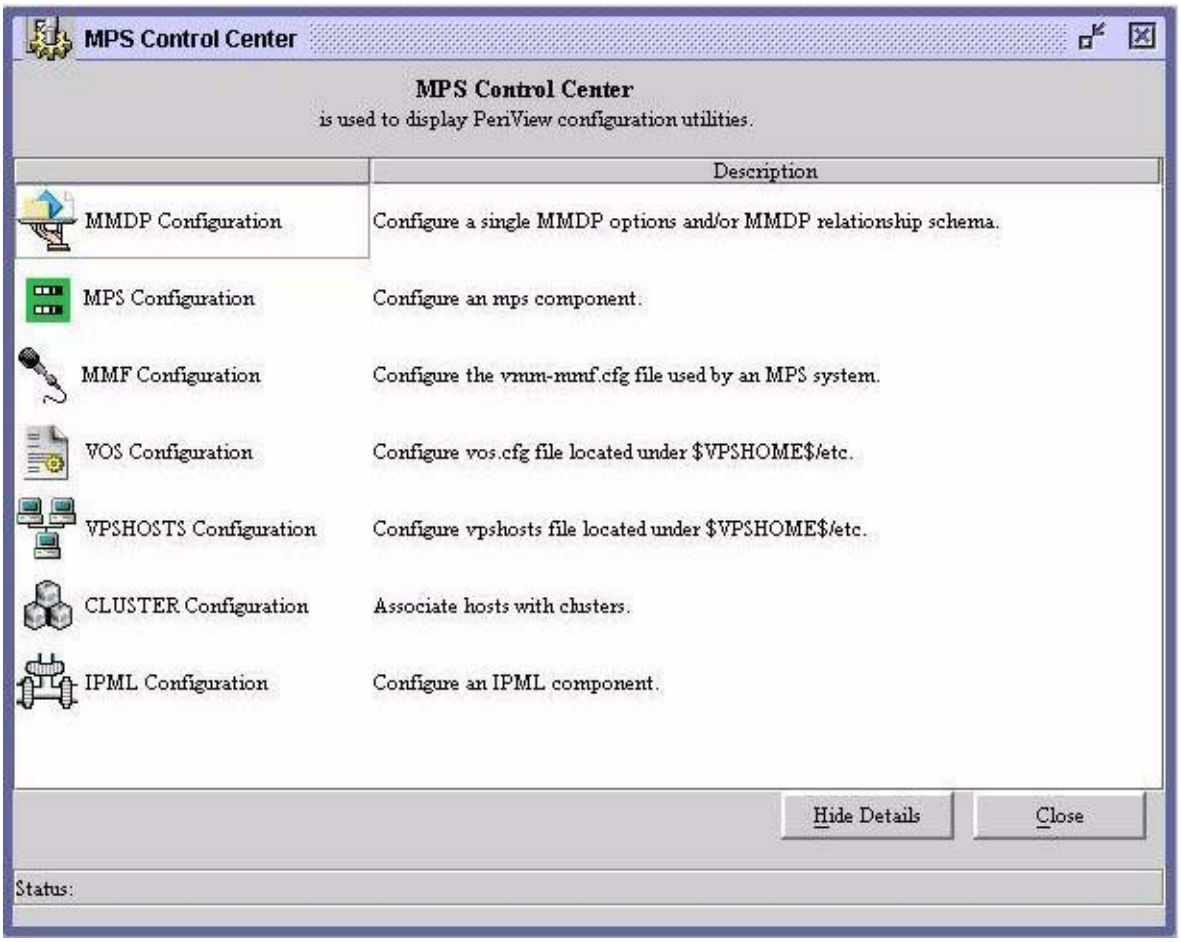
Command syntax	Report
`vmm mmfstatus`	Status of all activated (loaded) MMF files, including the system-wide record file
`vmm refstatus <mmf_file>`	Element play statistics for **`<mmf_file>`**, including outstanding play/record requests

The *PeriView Reference Manual* contains details about using the VSH Command Line.

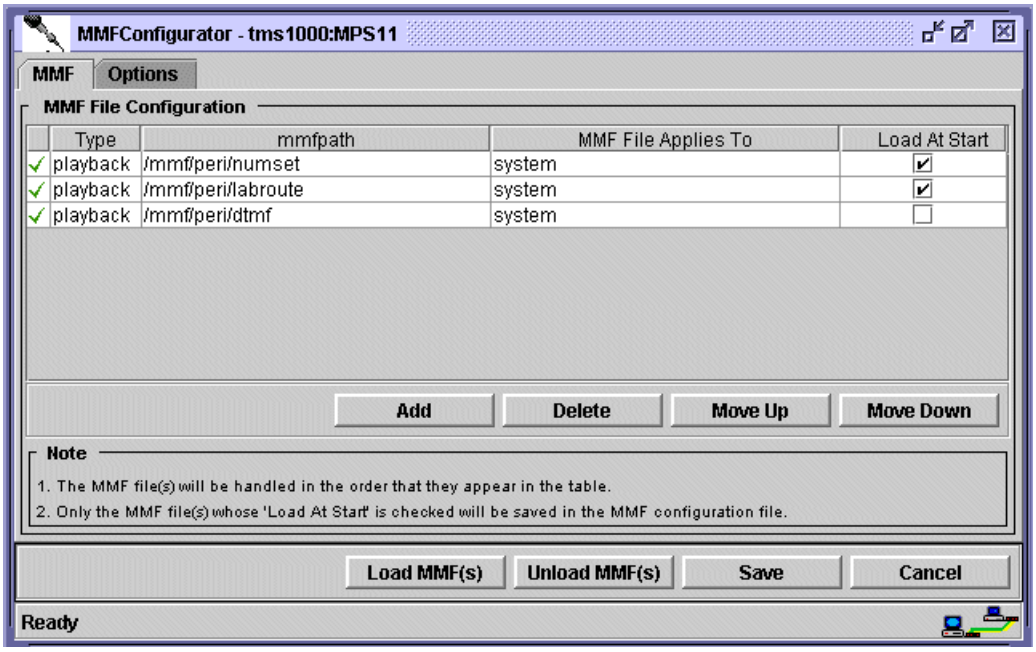
Using PeriView

1. **Launch the MPS Control Center Tool**.

2. **Select the MMF Configuration Tool.** Double <LEFT> click on MMF Configuration and select the desired Node and Component to view the MMF status.



The status of the MMFs are displayed. For more information about the MMF Configuration tool, see the *PeriView Reference Manual*.



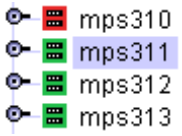
Host Status

The Host Status tool can be used to display the state of any communication link to a host. Information about the state of phone lines configured to use a host's Virtual Terminals, as well as the terminals themselves is also available.

How to ... Monitor Host Status

Using PeriView

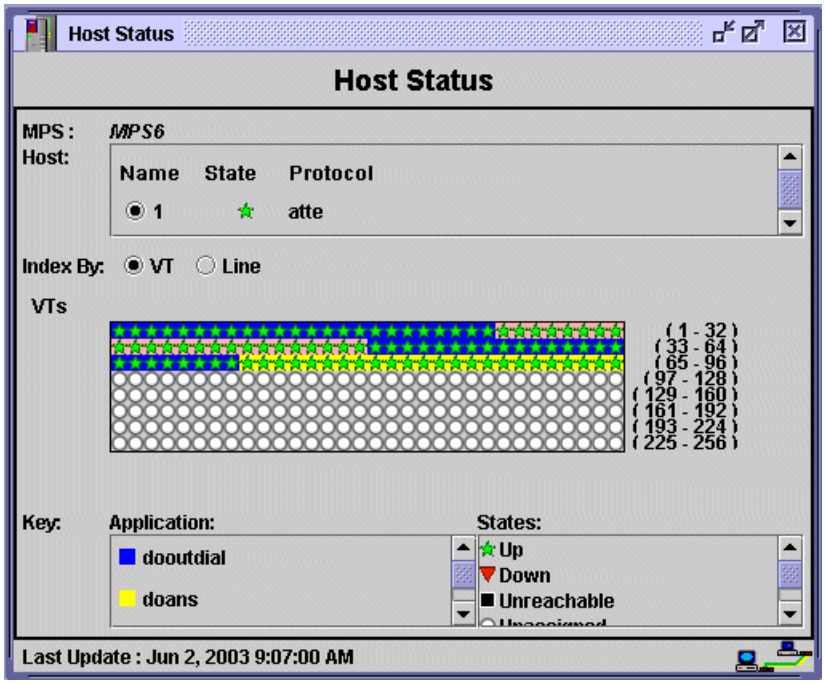
1. **Select a Component.** Select a component from the PeriView navigation pane.



2. **Select the Host Status tool.**

The state of the hosts, as well as their (numerical) "name" and protocol being used, can be determined in the `Hosts:` field of the tool. State is a representation of a PeriView tool attempting to contact some Media Processing Server Series process: the Host Status Tool attempts to contact the **`commgr`** (Communications Manager)

process.


Host Status
Host Status
MPS :
MPS6
Host:
Name
State
Protocol
1
atte
Index By:
VT
Line
VTs
(1 - 32)
(33 - 64)
(65 - 96)
(97 - 128)
(129 - 160)
(161 - 192)
(193 - 224)
(225 - 256)
Key:
Application:
dooutdial
doans
States:
Up
Down
Unreachable
Last Update : Jun 2, 2003 9:07:00 AM

Host states can be one of the following as indicated by a colored icon, shown in the table below and which appears in the State field shown above. VT states are also indicated by a colored icon: in this case they are shown in the VTs: field when the Host Status tool Index By: selection is set to VT.

Table 6:

Component Host Status States

Symbol	State	Applies To	Description
	Up	Host	Media Processing Server Series protocol software has determined that the remote host is available for communication.
		VT	Media Processing Server Series protocol software has determined that a given VT can be used to establish a session with the remote host.
	Down	Host	Media Processing Server Series protocol software has determined that the remote host is not available for communication.
		VT	Media Processing Server Series protocol software has determined that a given VT can not be used to establish a session with the remote host.
	Unreachable	Host	The state of the host is not known, as it is unreachable.
		VT	The state of the VT is not known, as it is unreachable.
	Unassigned VT	VT	The phone line is not currently associated with a VT. However, at some other point in time it can be associated with a VT as a result of pooling. For information about pooling VTs, see the *Media Processing Server Series COMMGR Reference Manual*.
	Unknown	Host	The state of the host is unrecognized. It does not comply with the definition of any other state listed in this scrolling window.
		VT	The state of the VT is unrecognized. It does not comply with the definition of any other state listed in this scrolling window.

System Statistics Reports

The Media Processing Server automatically gathers system statistics for monitoring activity on a single MPS or across the entire MPS network.

Use PeriReporter Tools to display system statistics reports (see ***Generate System Statistics Reports*** on page 78.)

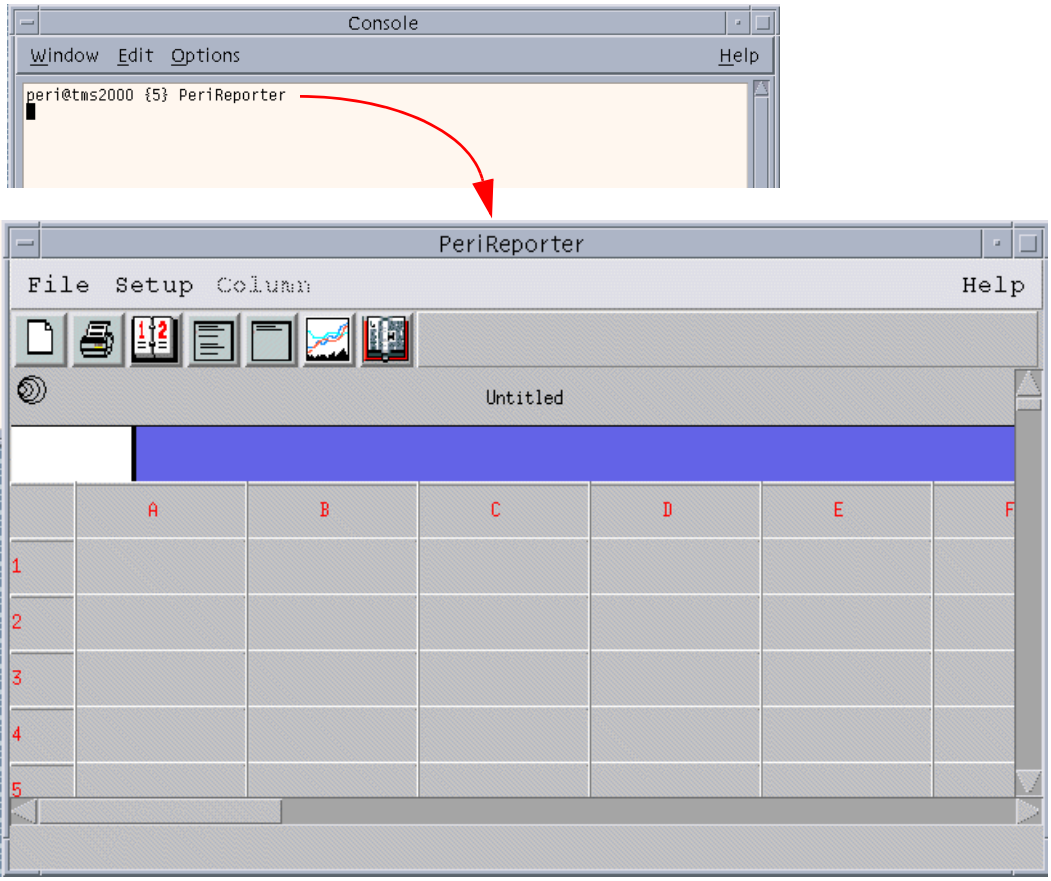
PeriReporter Tools run on a single node within a network, but can collect and report statistics network-wide.

How to ... Generate System Statistics Reports

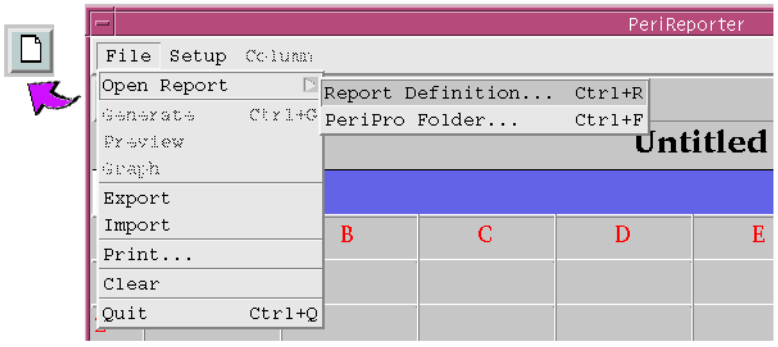
Statistics reports can be generated and viewed on demand. This report can also be printed as a hard copy or printed to a file.

To generate the report:

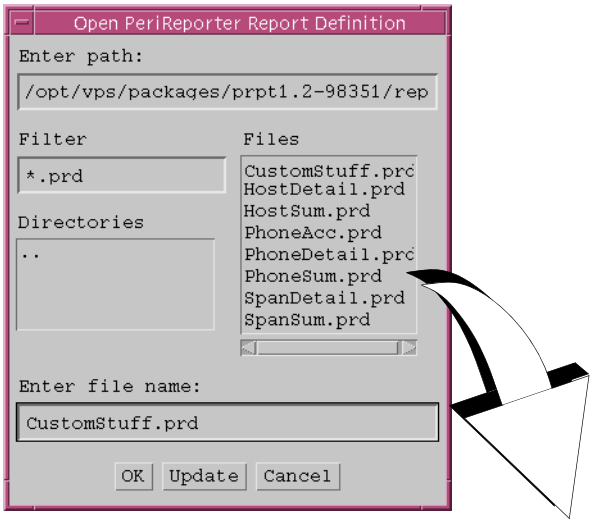
1. **Launch PeriReporter.** Select PeriReporter from the Start menu on a Windows 2000 platform or by entering `PeriReporter` on a UNIX or Windows 2000 command line.



2. **From the PeriReporter window, open a file.** Use the File Open icon or the menu path File—Open Report (use the Report Definition... option to select predefined reports). The Report Definition window displays.

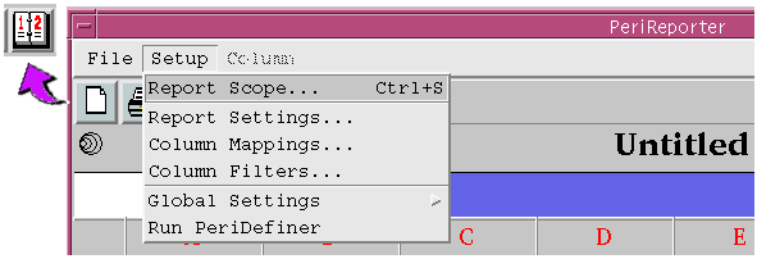


3. **From the Report Definition window, select the desired report.** This can be one of the seven predefined system reports or a report that has been defined and saved by someone else. Click OK.

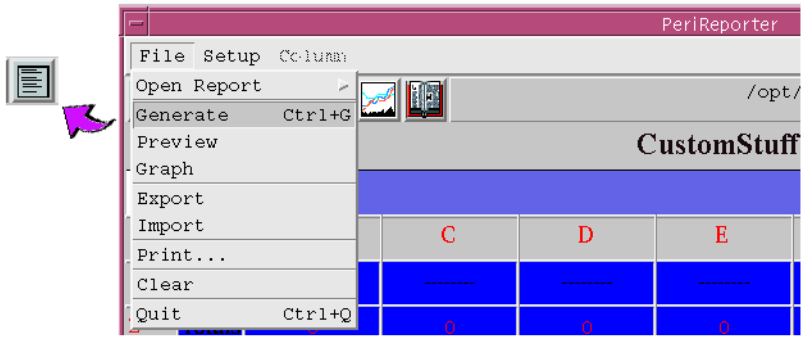


Report (File) Name	Definition
PhoneDetail.prd	Detail Phone Line Statistics Report. Phone line usage is reported for selected or all Media Processing Servers and phone lines, for a specified date range. Totals reflect usage per individual Media Processing Server.
PhoneSum.prd	Summary Phone Line Statistics Report. Data is collectively summarized for selected or all phone lines, then reported for selected or all Media Processing Servers, for a specified date range. Data for each Media Processing Server is summarized on a single line. Totals reflect usage inclusive of all Media Processing Servers.
PhoneAcc.prd	Accumulated Summary Phone Line Statistics Report. Data is summarized collectively for the time period specified by the user for selected or all phone lines, then reported for selected or all Media Processing Servers for the specified date–time range.
SpanDetail.prd	Detail Span Statistics Report. Data is reported for each span on selected or all Media Processing Servers, for a specified range.
SpanSum.prd	Summary Span Statistics Report. Data is collectively summarized for all spans, for selected or all Media Processing Servers, then reported for a specified date range. Data for each Media Processing Server is summarized on a single line. Totals reflect span activity inclusive of all Media Processing Servers.
HostDetail.prd	Detail System Statistics Report. Data is reported for selected or all Media Processing Servers and external hosts for a specified date range.
HostSum.prd	Summary System Statistics Report. Data is cumulatively summarized for each external host for selected or all Media Processing Servers, then reported for a specified date range. Data for each Media Processing Server is summarized on a single line. Totals reflect system activity inclusive of all Media Processing Servers.

4. **Set the scope of the report.** Use the Report Scope icon or the menu path Setup—Report Scope... . Enter the requirements in the Report Scope window.



5. **Generate the report.** Use the Generate icon or the menu path File—Generate.



This page has been intentionally left blank.

5

Voice Processing Applications

This chapter covers:

1. **Application Memory Requirements**
2. **Activating Playback MMF Files**
3. **Assigning and Starting Application Programs**
4. **Terminating and Unassigning Application Programs**
5. **Deactivating MMF Files**
6. **Updating or Replacing**

Voice Processing Applications

This chapter provides the information needed to:

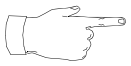
- Start and stop applications
- Work with vocabularies
- Set up Caller Message Recording (CMR)

Although most interaction with the system is through GUI tools like PeriView or MPS Developer, a command line is needed to issue some commands and to perform system-related application maintenance. This section also describes these operations. For specifics concerning interaction with the system as a whole, see ***Command Line Interaction*** on page 47.

For information on monitoring application and MultiMedia Format (MMF) file status, see ***Status Monitoring*** on page 56.

Application Memory Requirements

An important consideration to make before assigning and running applications is their impact on overall system performance. One of the greater aspects of this impact is the amount of memory an application requires during its execution. The following formula accurately calculates an application's memory requirements.



This formula *only* addresses application overhead and does not include other system components or custom features that are part of the native operating system (OS) software.

The total memory requirements to run *N* instances of an application, which uses Indexed Sequential Access Method (ISAM) databases (ISAM significantly reduces memory requirements), is calculated as follows:

(150K + *WS*) * *N* + (*APP*)

where *WS* is the total length in Kbytes of all Working Storage items in the application, *N* is the Number of phone lines running the application, and *APP* is the size of the APPlication `.vex` file in Kbytes.

If the application does not use ISAM databases, the memory requirements formula changes as follows:

(200K + *WS*) * *N* + (*APP*) + 500K

To determine the amount of memory needed for an application's Working Storage (WS), use the following **`vexdump`** command:

```
vexdump -v appname | grep Values
```

Additional Memory Considerations

If an application uses Shared Memory, the above memory requirements increase by 150K + total length of all shared memory items.

If an application uses Hashed Sequential Access Method (HSAM) databases, the actual file size in bytes of each HSAM file must be added to the total size of the `.vex` file. Adding memory to HSAM applications improves overall performance.

If an application executes the LINK command, the total length of all linked-to `.vex` files which can be in memory simultaneously, must be added to the total size of the `.vex` file.

Example:

An application with a 300K `.vex` file, running on 48 lines, **not** using ISAM and linking to two different 100K size applications, with a total of 20K working storage requires:

(200 + 20) * 48 + (300 + 100 + 100) + 500 = 12M

Use this worksheet as a guide to computing application memory requirements:

	(If ISAM)				(If HSAM)		(If no ISAM)
(	150K ~~~~+WS) 200K	*	#_of_lines_running_app	+ (.vex_file_size	+ @ HSAM_files'_size ~~~~ + @ linked-to_files'_size	)	+ 500K
	(If no ISAM)				(If LINK)		

Activating Playback MMF Files

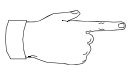
A playback MMF file can be activated either for a particular application or for all applications in the system. Normally, commands in the VMM-MMF configuration file (`$MPSHOME/mpsN/etc/vmm-mmf.cfg`) activate the MMF files when the system boots. MMF files can also be activated from the MPS command shell. This is useful for testing new applications and configurations. MMF files activated from a command shell are not reactivated if the system is restarted.

See the *Media Processing Server Series System Reference Manual* for configuration details and an explanation of MMF files and hash tables.

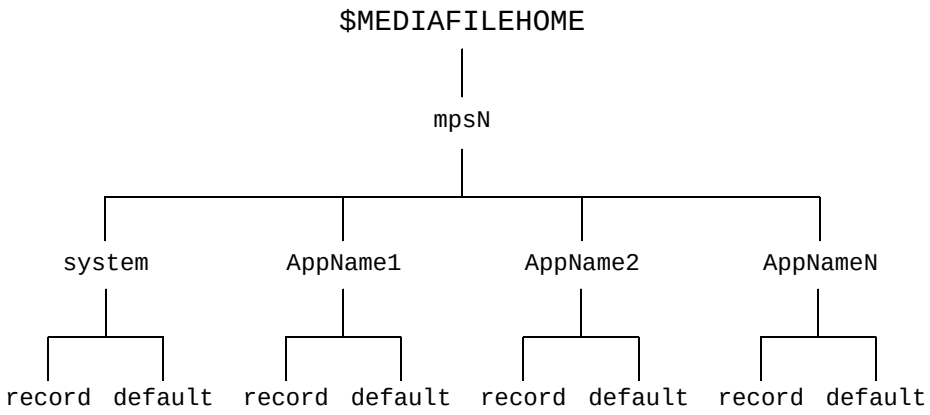
There are two methods for activating MMF files:

- For backward compatibility, the activation of MMF files during system initialization is done by processing the `vmm-mmf.cfg` configuration file and activating each listed MMF file in the specified manner (record, system-wide, application-specific, etc). This activation method has several drawbacks. When running with a secondary node backing up several non-homogeneous nodes, VMM has to activate the same MMF files that were active on the primary system. Keeping these files up to date on the secondary node is difficult due to the fact that, the files can be located anywhere on the primary system. Another drawback is that anytime a new MMF file is added, a configuration file must be modified to ensure that on restart the MMF will be loaded.

- The newer method is introduced in MPS software release 2.1. It requires all MMF files to be in the `$MEDIAFILEHOME` directory (or one of its subdirectories—see the diagram on the next page). Upon initialization, VMM activates all MMF files in the `$MEDIAFILEHOME/mpsN` directory. Each `mpsN` directory contains a `system` subdirectory, which stores the MMF files to be activated system-wide. There is also a subdirectory for each application for application-specific MMF files. Both the `system` and application directories have two subdirectories named `record` and `default`. The `record` subdirectory stores the MMF file into which data is recorded by default. (This should be the only file in that subdirectory.) The `default` subdirectory contains the default MMF (for playing) if one is needed. This eliminates the need for the `vmm-mmf.cfg` configuration file and provides a well-defined location from which a secondary node can periodically update its MMF files. To ensure backward compatibility, VMM still attempts to process the `vmm-mmf.cfg` file after activating any files in `$MEDIAFILEHOME`. The processing of the `vmm-mmf.cfg` file is not supported for systems with secondary nodes.



The Nx1 redundancy configuration does not support the use of `vmm-mmf.cfg`.



VMM loads all files in `$MEDIAFILEHOME/mpsN/` and its subdirectories. In the case that there is more than one file in the `record/default` directory, the last file activated in the directory will be designated as the default record/play MMF. After activating all files, VMM attempts to process the `vmm-mmf.cfg` file (if one is present). VMM is informed about which MPS it is running on by TRIP, so that when a secondary node is taking over for a primary, the correct file is loaded.

When adding a new file inside of `$MEDIAFILEHOME`, the file does not become active until the system is restarted. Use the **`vmm mmfload`** command to activate the file immediately without restarting.

System Wide Playback MMF Files

A playback MMF file can be activated such that all applications on the system can access it.

How to ...Activate System-Wide Playback MultiMedia Files

1. **Open a shell on the Media Processing Server that will be running the applications.** see ***Command Control*** on page 50..
2. **Assign the applications to the desired lines and start them.** see ***Assigning and Starting Application Programs*** on page 93.
3. **Change location (`cd`) to the directory where the desired MMF file exists** (see ***Working with Directories*** on page 122.). Alternatively, the path to this location can be supplied in the **`vmm mmfload`** command shown in the next step.
4. **Allow the applications to access the MMF file using the `vmm mmfload` command.** Use the syntax **`vmm mmfload <mmfname>[,system]`**, where **`<mmfname>`** is either the name of the file to be loaded (if in its directory) or the path to the file. Special permissions or privileges are not needed to use this command. (It is

not necessary to specify the "**`system`**" parameter as this is the default.)

Application-Specific Playback MMF Files

An MMF file can be activated such that only a specific application can access it. Included in this procedure is the creation of a hash table. A hash table is a data structure which resides in system memory (as opposed to voice data memory). When the MMF file is activated, VMM enters the names of all vocabulary elements in that file into a hash table. When an application references an element by name (instead of referencing the element by index number), VMM uses the hashing scheme to locate the recorded contents of that element. A hash table can function for the entire system or just one application.

It takes several steps to activate a playback MMF file for a particular application:

- create the hash table for the application (that is, initialize the application)
- activate the MMF file for the application
- assign the application to phone lines and start it (using APPMAN)

How to ...Activate Application-Specific Playback MultiMedia Files

1. **Place the MMF file in a directory under** `$MEDIAFILEHOME`**.**
2. **Open a shell on the Media Processing Server that will be running the application.** see ***Command Control*** on page 50.
3. **Create the application-specific hash table using the** **`vmm appinit <appname>`** **command.** See the *Media Processing Server Series System Reference Manual* for a detailed description of hash tables.
   a. To have the application look in the application-specific hash table before the system hash table, use the syntax **`vmm hashfirst <appname>,local`**
   b. To have the application look in the system hash table before the application-specific hash table, use the syntax **`vmm hashfirst <appname>,system`**

System hash tables are used in relation to applications to contain such common and frequently accessed elements as DTMF tones (`dtmf`) and numeric elements (`numset`), and therefore reduce the size of the application hash table (large tables can negatively impact system performance).

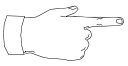
4. **Allow the application to access the MMF file by using the** **`vmm mmfload`** **command.** Use **`vmm mmfload <mmfname>,<appname>`** to activate an MMF file in the application-specific hash table. Use **`vmm mmfload <mmfname>,system`** to activate an MMF file in the system hash table.

When loading an MMF for a specific application, if the application has not already been initialized, VMM will initialize it using the default setting for hashfirst (which is

local).

5. **Assign the application to the desired lines and start it.** see ***Assigning and Starting Application Programs*** on page 93.

The **`vmm hashfirst`** command can be used to change the hashfirst sequence after an MMF file is activated (that is, without deactivating the MMF file). For example, if the command **`vmm hashfirst <appname>,local`** was used, the command **`vmm hashfirst <appname>,system`** can be used to change the hashfirst sequence. For all systems, **`hashfirst`** commands must be placed in the `vmm-mmf.cfg` file.



The following information is for backward compatibility on non Nx1 systems:

The command **`vmm mmfload <mmfname>`** loads the specified MMF file into the system hash table. To load the MMF into an application hash table, specify the application name in the command **`vmm mmfload <mmfname>,<appname>`**. In a development environment, the application can be started, and then the command **`vmm mmfload <mmfname>`** can be applied. In a production environment, the **`mmfload`** commands should be placed in the file `$MPSHOME/mpsN/etc/vmm-mmf.cfg`.

Do not **place any `mmfload` commands in the VMM configuration file `$MPSHOME/mpsN/etc/vmm.cfg`. Be sure they are placed into the proper file (VMM-MMF configuration file) as outlined above!**

For additional information, see the *Media Processing Server Series System Reference Manual*.

Assigning and Starting Application Programs

To run an application program, first assign it to a phone line (i.e., bind the application) and then start the application. In most circumstances, use the APPMAN (Application Management) tool to assign and start applications. APPMAN allows setting up the applications and saving the current configuration. If the system shuts down, the applications return to this configuration.

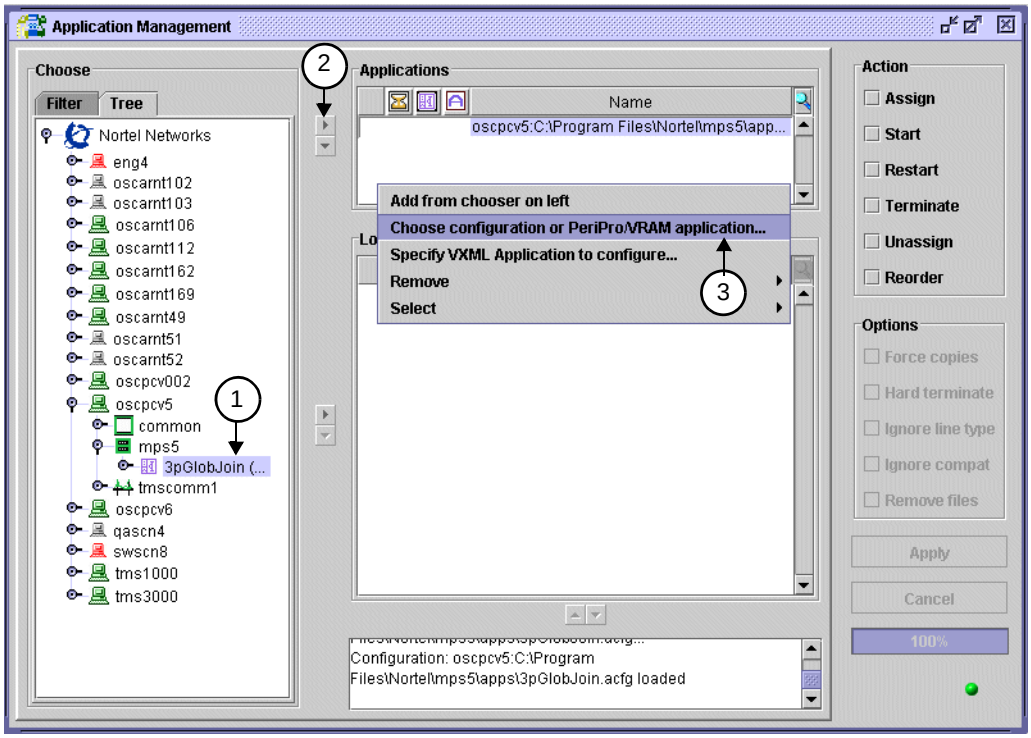
Applications can be assigned to MPS components. For information about components, see ***System Configurations*** on page 19.

APPMAN automatically creates a default configuration for each application. The applications run-time environment can be customized using the Application Configuration tool.

See the *PeriView Reference Manual* for information about creating and using custom configurations.

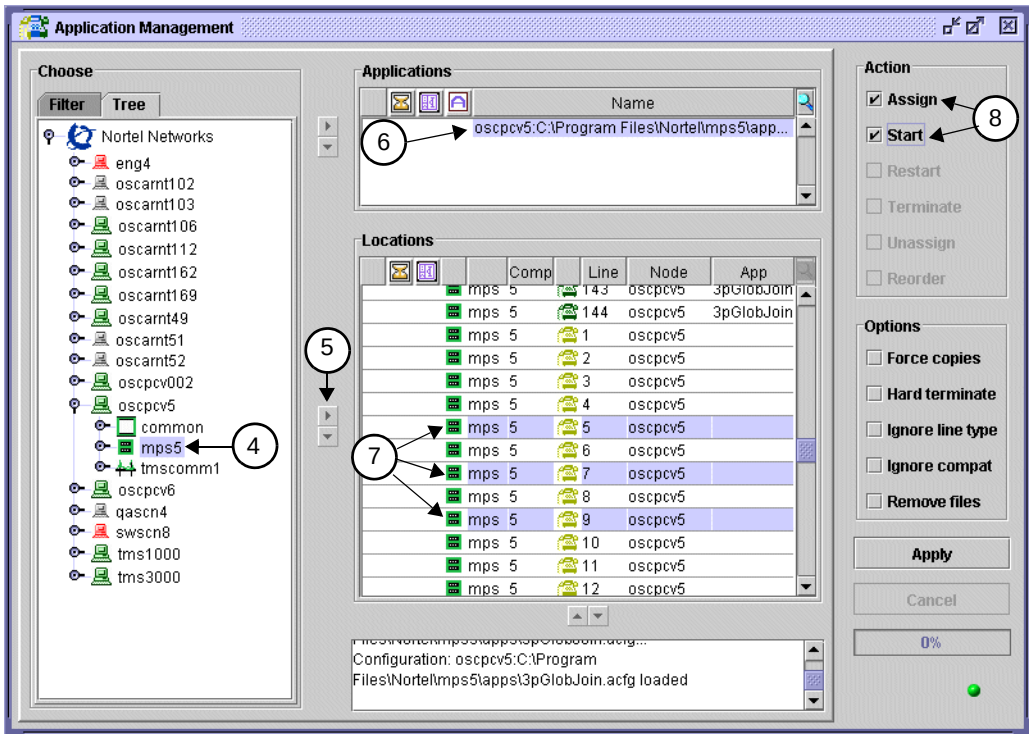
How to ... Assign and Start Application Programs

1. **Launch the Application Management Tool.**
2. **Select the Application.** Either select an existing application (#1 and #2) from the Tree Choose list or add a new application (#3) to the Applications window of the APPMAN tool.



3. **Select the Location(s).** Select individual or multiple node(s) and/or component(s)
(#4) and add them (#5) to the Locations window.

4. **Assign and Start the Application.** Select the application (#6), the location(s) (#7) and the Assign and Start checkboxes (#8). Select the Apply button.



After the system assigns and starts the application, the phone line icons turn **green**.

See the *PeriView Reference Manual* for a complete description of PeriView and the Application Management tool.

Applications can also be assigned and started from a MPS command shell. See the *Command Reference Manual*, for a detailed command descriptions.

Start and Restart Line Order

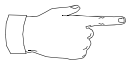
Phone lines are assigned, started, and restarted, in the order listed under Locations in the Assign/(Re)Start Lines window. This list can be reordered, if necessary, by using the up and down arrows below the Locations window.

How to ... Reorder Selected Application Locations

1. **Select one or more locations in the Locations window.** <LEFT> click to select an entry. <RIGHT> click the Select—None menu path to clear all selected entries before selecting individual locations.

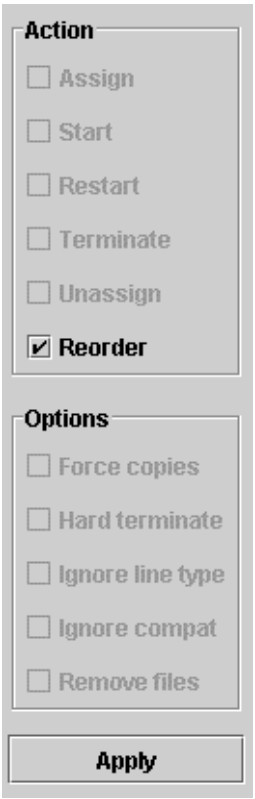
2. **Use the up and down arrows to reorder selected listings.** Items move one position with each <LEFT> mouse click. Multiple selections move up or down together.

Repeat the previous steps for any items that have to be moved within the Locations window.



Do **not** perform step #3 to modify the line start order for the current session only. To define a default line start order for future system bootups, continue with step #3.

3. **Select the Reorder checkbox and click Apply.** This applies the reorder changes for future system bootups.

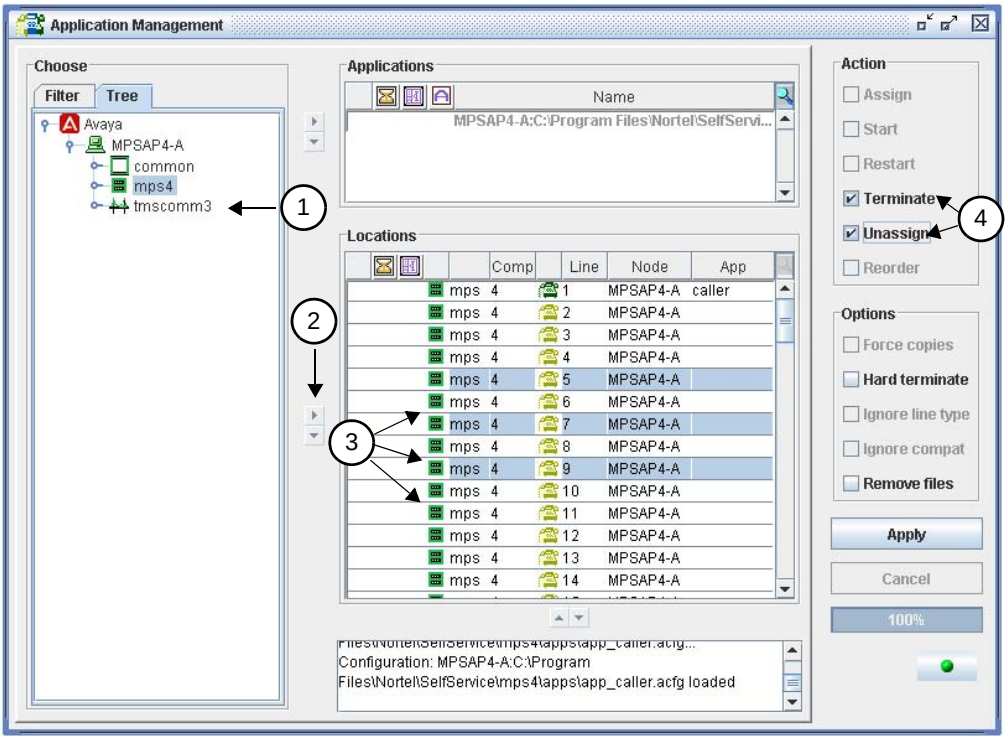


Terminating and Unassigning Application Programs

In most circumstances, use APPMAN to terminate and unassign applications. However, applications can also be terminated from a Media Processing Server shell. See **`tappman`** in the *Command Reference Manual*, for command description.

How to ... Terminate and Unassign Application Programs

1. **Launch the Application Management Tool**.
2. **Select the Location(s).** Select individual or multiple node(s) and/or component(s)
   (#1) and add them (#2) to the Locations window.



3. **Choose the Lines to Terminate.** Select the locations for termination in the Locations window (#3).
4. **Terminate and Unassign.** Select the Terminate and Unassign checkboxes (#4) and click the Apply button.

By default, the Media Processing Server uses a soft termination. A soft termination waits for callers to hang up before terminating each application.

See the *PeriView Reference Manual* for a complete description of PeriView, the Application Management Tool, and Terminating/Un-Assigning Lines.

Deactivating MMF Files

When activated, MPS MMF files (both playback and record) take up space in Voice Data Memory (VDM). Since this space is limited, applications should be deactivated when they are no longer required. VDM status can be obtained from the VMM Cache Status Report, which is displayed by entering the command **`vmm cachestatus`** either in an MPS shell or the VSH Command Line.

The **`vmm repconfig`** command can be used to display configuration information for VDM (as well as how much VDM is in the system) however, it does not indicate usage. The **`vmm cachestatus`** command is preferred because it shows how much VDM is actually in use (as well as other useful information).

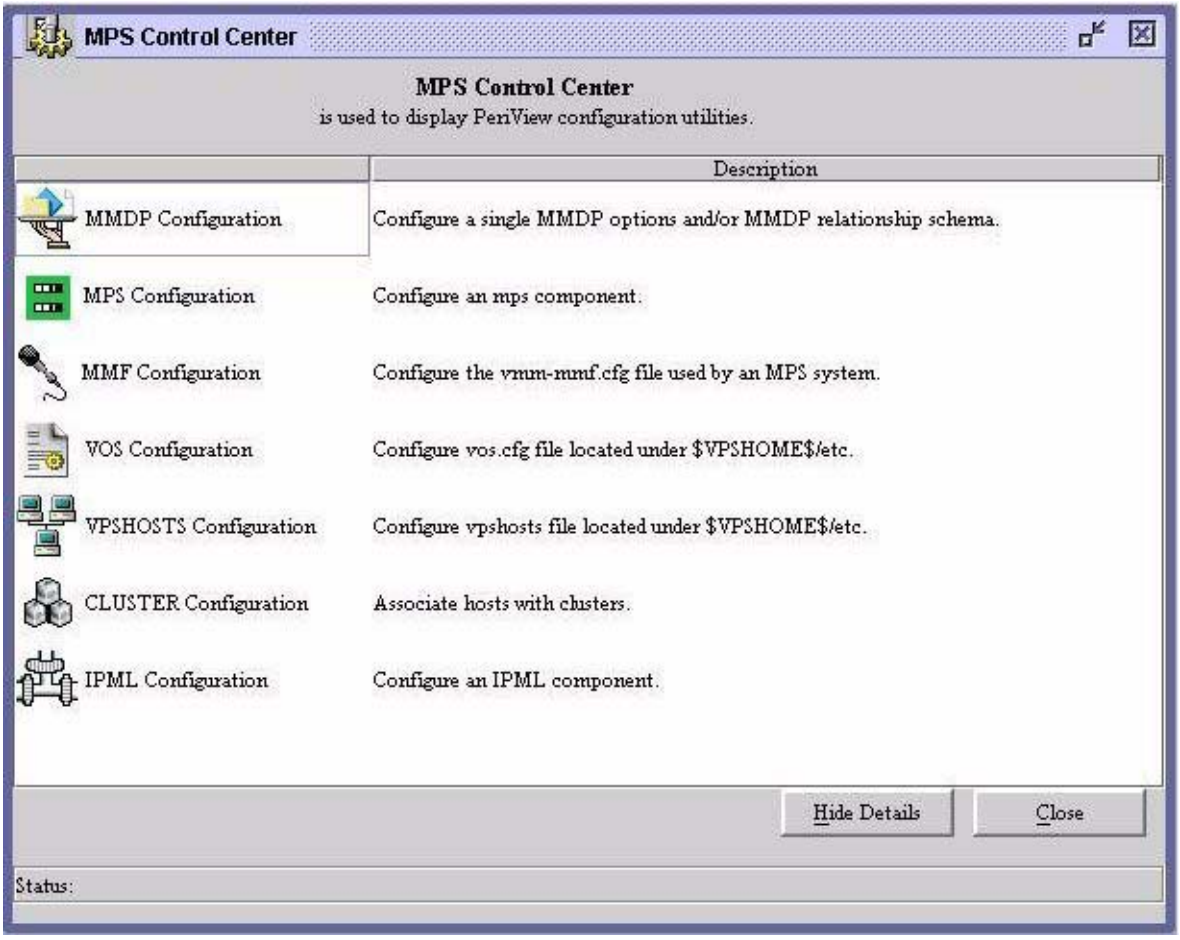
If an attempt is made to deactivate a file currently being accessed by one or more applications, VMM does not deactivate the file immediately. Instead, VMM places the file in a deactivation queue. If any application tries to make any *new* references to an MMF file in the deactivation queue, the reference fails and VMM generates an alarm. When all prior references to elements in the file have completed, VMM deactivates the file.

While outstanding references to an MMF file are being satisfied, it is possible to deactivate another MMF file.

How to ...Deactivate MultiMedia Format Files using the MMF Configurator Tool

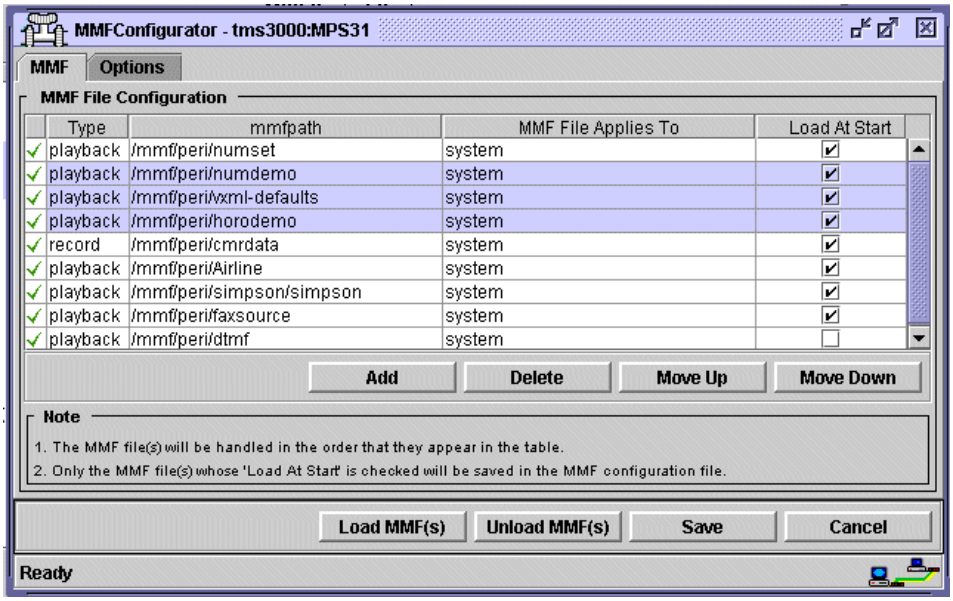
1. **Launch the MPS Control Center.**

2. **Select the Node and Component.** Double <LEFT> click on MMF Configuration in the MPS Control Center.



In the Component Selector window, select the Node: and the Component: containing the MMF file(s) to Unload. Select OK to start the MMF Configurator Tool.

3. **Select the MMF file(s) to Unload.** Select the file(s) and <LEFT> click on Unload MMF(s). To prevent them from loading on startup, uncheck the box Load At Start.



How to ...Deactivate MultiMedia Format Files using the Command Line

1. **Open a Media Processing Server shell on the Media Processing Server that contains the unneeded MMF file,** see ***Command Control*** on page 50.

2. **Enter the `vmm mmfunload` command**.
   - To unload an MMF file that was available to all applications in the system-wide hash table, use the syntax **`vmm mmfunload <mmfname>,system`** (It is not necessary to specify the "**`system`**" parameter as this is the default.)
   - To unload the MMF file from a particular application (application-specific hash table), use the syntax **`vmm mmfunload <mmfname>,<appname>`**
   - To unload *all* MMF files available system wide or to a specific application, use the applicable prior command and substitute the option **`all`** for **`<mmfname>`**

3. Move the file out of `$MEDIAFILEHOME` or it will be activated next time the system is (re)started. Also, remove its name from the `vmm-mmf.cfg` file (if this file is used).

When unloading an MMF, if only the MMF name is specified, the MMF will be unloaded from the system hash table by default.

To determine if an MMF has been deployed system-wide or for a particular application, issue the **`vmm mmfstatus`** command in an MPS VSH command line.

The last two columns indicate system (Sys) or application (App) usage.

```
vsh.11@tms1000 {1} -> vmm mmfstatus
=================================================================
                                VMM: MMF Status Report
=================================================================
System-Wide Default MMF:    None
System-Wide Record MMF:     /mmf/peri/cmrdata

                   SCSI   EAP      EAPs   Space
MMF Path    PERMS  ID  Capacity  Loaded   Used  HWM   LWM   Sys  App
/mmf/peri/dtmf
            RDWR    3     48       23     58%  90%   70%   Y    0
/mmf/peri/cmrdata
            RDWR    3     2048     0      99%  90%   70%   N    3
```

A “Y” under Sys denotes that the MMF is available system wide. An “N” indicates it is not available system wide. A number under App indicates the number of applications using that MMF. A zero (0) indicates it has not been loaded on an application-specific basis.

If the MMF file is available for several applications and/or system wide, it must be deactivated for each application and/or the entire system. That is, the file has to be deactivated for each **mmfload**, **mmfrec** or **mmfdefault** command issued.

Use the **vmm appremove <appname>** command to remove an application from the application hash table (this negates the effect of the **vmm appinit** command). Any open application-specific MMF files are deactivated when the application is removed. This command cannot be used in the vmm.cfg file.

See the *Media Processing Server Series System Reference Manual* for a detailed description of hash tables. The *PeriView Reference Manual* contains details about using the VSH Command Line.

Updating or Replacing Playback MMF Files

During system initialization, MPS configuration files set up the MMF files used by applications to produce speech. MMF files have to be updated in the following circumstances:

- a new vocabulary file on disk becomes available.
- a new vocabulary file is created with PeriStudio.
- one or more elements in the vocabulary have been updated (for example, a message of the day or weather update application).

The entire MMF file can be updated (see ***Updating/Replacing a Playback MMF Playback File*** on page 102.) or individual elements updated without interrupting service (see ***Updating Elements in an Active Playback MMF File*** on page 107.) Also, any or all identical MMFs across a LAN or WAN can be updated via the *Zero Administration for Prompts* (ZAP) utility (see ***Synchronizing MMF Files Across Nodes (ZAP Utility)*** on page 110.). This automated synchronization facility provides a means of administering updates to and maintaining consistency among all active instances of a particular MMF residing on different nodes of a network.

Updating/Replacing a Playback MMF Playback File

There are three methods for replacing MMF elements and files:

- The preferred on-line method ensures continuous customer service. This is used to replace speech elements in the MMF. For additional information, see ***Replace a Playback MultiMedia File (On-line) (Preferred)*** on page 103..
- The alternative on-line method also ensures continuous customer service, but involves changing the VMM-MMF configuration file (if used). The system may page from disk (access data from the hard drive) some elements that were previously in Voice Data Memory. For additional information, see ***Replace a Playback MultiMedia File (On-line)*** on page 104.
- The off-line method can cause applications to temporarily construct fragmented phrases (i.e., for a brief time the application will not speak prompts located in the file being updated), but does not involve any changes to the VMM-MMF configuration files. For additional information, see ***Replace a Playback MultiMedia File (Off-line)*** on page 106.

How to ...Replace a Playback MultiMedia File (On-line) (Preferred)

This procedure updates existing speech elements in an MMF. Any elements that exist in the original MMF, but are not in the new MMF, will still be present after the update.

1. **Open a shell on the MPS running the application.** For additional information, see ***Command Control*** on page 50..
2. **Copy the new MMF file to the Media Processing Server.** The new MMF file *must have a different name than the one being replaced.*
   - If the file is on a disk, use the **`cp`** command to copy *both* the `.mmi` and the `.mmd` files to the MMF files directory.
   - If the file was created in PeriStudio, make sure it is in the MMF files directory.
3. **Create a Vgen incremental file which represents the MMF file.**
   - Make sure there is enough disk space in the MultiMedia file directory.
   - Use the syntax **`mmf2def1 -v<vgen_inc_name> <mmf_name>`** where `<mmf_name>` is the name of the MMF file copied in step 2. This generates the file `<vgen_inc_name>.def1`.
   - If no Vgen incremental filename is specified, one is created with the same filename and the `.def1` extension.
4. **Update the existing MMF (vocabulary) file with the temporary Vgen incremental file created in** step 3.
   - From the Media Processing Server shell, use the syntax **`vmm mmfupdate <mmf_name>,<vgen_inc_name.def1>`** where `mmf_name` is the active MMF file and `vgen_inc_name.def1` is the Vgen incremental file created in step 3.

How to ... Replace a Playback MultiMedia File (On-line)

1. **Open a Media Processing Server shell on the Media Processing Server running the application.** For additional information, see ***Command Control*** on page 50.
2. **Copy the new MMF file into the** `$MEDIAFILEHOME` **directory on the Media Processing Server**. The new MMF file must have a different name than the one being replaced.
   - If the file is on a disk, use the **`cp`** command to copy *both* the `.mmi` and the `.mmd` files to the MMF files directory.
   - If the file was created in PeriStudio, make sure it is in the MMF files directory.
3. **Activate the new MMF file**.
   - To add the new MMF file to an application specific hash table, use the syntax **`vmm mmfload <newvocab>,<appname>`**
   - To allow all applications on the Media Processing Server to use the MMF file, use the syntax **`vmm mmfload <newvocab>,system`**
4. **Deactivate the original MMF file**.
   - To unload an MMF file that was available to all applications on the system (system wide hash table), use the syntax **`vmm mmfunload <originalvocab>,system`**
   - To unload the MMF file from a particular application (application specific hash table), use the syntax **`vmm mmfunload <originalvocab>,<appname>`**

   If the MMF file is available for several applications and/or system wide, it must be deactivated for each application and/or the entire system. That is, the file has to be deactivated for each **`mmfload`**, **`mmfrec`** or **`mmfdefault`** command issued.

   Use the VSH command line to issue the **`vmm mmfload`** and **`mmfunload`** commands. For more information, see ***Activating Playback MMF Files*** on page 87. and see ***Deactivating MMF Files*** on page 98. respectively.
5. **Move the original file out of the** `$MEDIAFILEHOME` **directory.**
6. **If the VMM-MMF configuration file is being used, then edit the file to reflect the new vocabulary name**. Use a text editor to modify and save `$MPSHOME/mpsN/etc/vmm-mmf.cfg`. Replace the line containing the original MMF file with a line for the new MMF file.

If the MMF files have elements with the same name, the application(s) continue to access elements without corrupting concatenated phrases. However if the voice data memory is limited, activating `<newvocab>` *before* deactivating

`<originalvocab>` may cause the system to page some elements from disk. This can happen even when the original was entirely in voice data memory.

How to ... Replace a Playback MultiMedia File (Off-line)

1. **Open a Media Processing Server shell on the Media Processing Server running the application.** For additional information, see ***Command Control*** on page 50.

2. **Deactivate the original MMF file**.
   - To unload an MMF file that was available to all applications on the system (system wide hash table), use the syntax **`vmm mmfunload <originalvocab>,system`**
   - To unload the MMF file from a particular application (application specific hash table), use the syntax **`vmm mmfunload <originalvocab>,<appname>`**

   If the MMF file is available for several applications and/or system wide, it must be deactivated for each application and/or the entire system. That is, the file has to be deactivated for each **`mmfload`**, **`mmfrec`** or **`mmfdefault`** command issued.

3. **Copy the new MMF file over the original MMF file**. *Use the same name as the original MMF file.*
   - If the file is on a disk, use the **`cp`** command to copy *both* the `.mmi` and the `.mmd` files to the MMF files directory.
   - If the file was created in PeriStudio, make sure it is in the MMF files directory.

4. **Activate the new MMF file (now named `<originalvocab>`)**.
   - To add the new MMF file to an application specific hash table, use the syntax **`vmm mmfload <originalvocab>,<appname>`**
   - To allow all the applications on the Media Processing Server to use the MMF file, use the syntax **`vmm mmfload <originalvocab>,system`**

With this method, applications that access the vocabulary generate alarms and may speak corrupted phrases until the new vocabulary is activated.

The original MMF must be deactivated before the new file is activated. This frees space in voice data memory and allows the system to use it for the new MMF.

Use the VSH command line to issue the **`vmm mmfload`** and **`mmfunload`** commands. For further information, see ***Activating Playback MMF Files*** on page 87. and see ***Deactivating MMF Files*** on page 98. respectively.

Updating Elements in an Active Playback MMF File

Use PeriStudio to add or modify specific elements in a vocabulary file without interrupting customer service.

Use this procedure to:

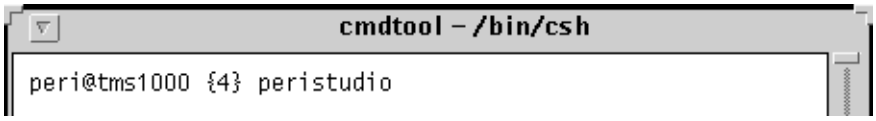
- update the audio data of the elements on-line, and
- modify the elements data within the MMF file

Activities such as digitizing elements are processing-intensive; therefore, it is suggested that such procedures be performed during times of low system activity.

The ZAP utility provides an automated means of updating and rectifying multiple active copies of identical MMF files across a network without putting an undue burden on network facilities (see ***ZAP and MMF files on the MPS*** on page 111.).

How to ... Update Elements in a Vocabulary

1. **Launch PeriStudio**. Start PeriStudio from the command line or from the Start menu on a Windows 2000 platform.



2. **Open the playback MMF file to be modified**. Since this vocabulary is active, the file is Read-only.
3. **Select the elements to be modified.**
4. **Export the elements to a temporary file (**for the purposes of this example, name it `temp1`**).** Specify MMF (PeriStudio) for the file format.
5. **Unload the current (active) file from PeriStudio.**
6. **Open the previously created MMF file** (`temp1` created in step 4.)
7. **Modify the elements**. New elements can be added or previous recordings updated as needed.
8. **Rename and save the file** (e.g., `temp2`).
9. **Export all of the elements that have been modified to a temporary file.** Specify VGEN Incremental for the file format and the appropriate encoding for the type of telephone interface (for a digital system, select u-law encoding: for an analog system, select adpcm encoding). The name chosen for step 4. (`temp1`) can be used or choose a new name (recommended) (e.g., `temp3`) and append the VGEN Incremental filename extension (`.def1`, resulting in `temp3.def1` for

example).

10. **Exit from or iconify PeriStudio.**

11. **Enter the following command from the Media Processing Server shell:**

```
vmm mmfupdate <MMF_filename>,<export_file.def1>
```

where:

• **<MMF_filename>** specifies the base name of the loaded (active) MMF file to be updated, and

• **<export_file.def1>** specifies the name of the file that has the elements to be incorporated into **<MMF_filename>**. This was the last file exported from PeriStudio (in step 9.). The path name to the file has to be included if it is not in the present working directory. In addition, the VGEN Incremental filename extension (`.def1`) is required.

The process is now complete. All subsequent use of the file by applications will access the updated recordings.

For an offline file, simply load it into PeriStudio and process it normally.

For details about using PeriStudio, see the *PeriStudio User's Guide*.

Synchronizing MMF Files Across Nodes (ZAP Utility)

In some configurations, applications across multiple nodes use the same MMF files. The *Zero Administration for Prompts (ZAP)* utility automates the process of administering updates to and maintaining consistency between all activated instances of an MMF file across the network. It determines if a set of files contains identical elements and provides the capability to synchronize the files. ZAP can modify MMF files on all nodes list in a `vpshosts` file or in an alternate file; individual nodes or Media Processing Servers; groups of Media Processing Servers; or specific elements within an MMF file.

ZAP also generates reports illustrating the differences between the source and target MMF files, and the results of modifications made to the target MMF files. By default these log files are created whenever ZAP is used, and are never overwritten. While administration of these files is left to the discretion of the user, this can eventually lead to disk saturation if files are not off-loaded or deleted. To reduce this need for manual intervention, use the **`-C`** option to combine these files into a corresponding consolidated log file.

A synchronization distribution log file is generated by the node originating the synchronization request. An update results log file is generated by remote (target) nodes. Each of these nodes also contains a synchronization status log file. These files are stored in the `$MPSHOME/common/log` directory of the reference node. The synchronization distribution log file is named in the format `zap.distribute.refnode.mmf_name.selected_elements.MMDDCCYY`, where `refnode` is the name of the node originating the synchronization request, `mmf_name` indicates the base name of the reference MMF file, `selected_elements` is the name or EAP number(s) that have been selected for updates, and `MMDDCCYY` indicates the date the file was generated. This log file contains information regarding the distribution and completion status for all MMF file synchronization requests, information on which nodes were *not* notified of the updates (and the reason thereof), and errors encountered during the synchronization process.

If all elements within an MMF have been selected for updating, the `selected_elements` portion of the log file name appears as `ALL_ELEMENTS`.

The update results log file is named in the format `zap.results.target_node.mps#.mmf_name.selected_elements`, where `target_node` is the name of the remote node where the synchronization has occurred, `mps#` is the number of the Media Processing Server on which the target MMF file is located, `mmf_name` indicates the base name of the reference MMF file, and `selected_elements` is the name or EAP number(s) that have been selected for updates. The file contains information on modifications made to the MMF file, as well as errors encountered due to inconsistencies between the reference and target MMFs.
If an MMF contains multiple elements with duplicate names, ZAP uses only the first duplicate element from the source MMF file to update the target MMF file; therefore, only this first element needs to be updated and maintained. The element which appears first in the target MMF file (i.e. with the lowest EAP number) is updated; however, none of the remaining duplicated elements is. The update results log file

indicates that multiple elements with the same encoding are present in the MMF file.

The synchronization status log file is named in the format `zap.status.refnode.mmf_name.selected_elements.MMDDCCYY`, where `refnode` is the name of the reference node where synchronization was initiated, `mmf_name` indicates the base name of the reference MMF file, `selected_elements` is the name or EAP number(s) that have been selected for updates, and `MMDDCCYY` indicates the date the file was generated. The state of the synchronization process for each target node is contained in this file.

Upon completion of the **`zap`** process, all synchronized MMF files contain identical elements and data, even though the elements may be stored at different positions within the file. This utility presently updates only active instances of the MMF file identified by the reference file.

ZAP and MMF files on the MPS

In an MPS system, when ZAP updates any MMF file, it is required that there exists a copy of that MMF file for each component in the system. The directory should be created for each of the MPS components on the MMF partition and all the files, that ZAP operates on, be duplicated under these directories. Make sure that the `/opt/vps/mpsN/etc/vmm-mmf.cfg` files on the system are updated to reflect the change in the file locations.

For example:

On an MPS 500 (*with components mps1 and mps2*), the MMF "`myPrompts`" needs to be updated periodically by ZAP. Hence, the following directories must be created:

```
/mmf/mps1
/mmf/mps2
```

The MMF "`myPrompts`" must be copied into each of these directories. The files `$MPSHOME/mpsN/etc/vmm-mmf.cfg` must have the following line added:
`mmfload /mmf/mpsN/myPrompts`

Ensure that any previous references to the MMF in vmm-mmf.cfg file are removed

ZAP by Proxy (Using Groups)

By default, ZAP connects from a local (reference) node to all remote (target) nodes. Where multiple LANs exist, which in turn contain multiple nodes that need to be updated by ZAP, network traffic can be further reduced and performance improved by having ZAP function on a proxy basis. In this case ZAP updates *one* Media Processing Server for a particular node in a group (LAN): each of the other Media Processing Servers on this node, and one Media Processing Server on each of the other nodes in the group, are updated remotely from this "locally updated" (proxy) Media Processing Server. This functionality requires the presence of a user-defined `zap.networks.cfg` file.

The `zap.networks.cfg` file *must* contain every node in the network since this file is used to determine the topography of the network. Typically, each LAN is defined as a group. In all cases, the following syntax rules must be followed:

- Groups are defined by using the term `[GROUP]` on its own line. All nodes that follow will be construed as belonging to that group until ZAP encounters another `[GROUP]` tag or the end of the file.
- Only one node can be listed per line, and each node can belong to only one group.
- There can be no empty groups, and no node may appear ahead of the first group.
- A pound sign ("#") precedes commented data. This symbol must appear at the beginning of a line (comments entire line) or have at least one space before it.
- Blank lines are ignored.

The `zap.networks.cfg` file must be placed into the `$MPSHOME/common/etc` directory. By default, only the Media Processing Servers listed in the `vpshosts` file on the reference node are addressed. If any node in any group contains Media Processing Servers that are not in this file, those components are not updated.

If all target nodes have not been updated to use the latest ZAP release or have security in place that does not allow remote ZAP sessions to complete correctly, the **`-L`** option must be used to ensure compatibility. This command line option forces all applicable components on all nodes to be updated directly from the local (reference) node, and prevents any remote ZAP processes from occurring, thereby overriding any `zap.network.cfg` files that have been defined

Updating a Specific Element

By default ZAP compares each target MMF with the designated MMF on the reference node and transmits to each one those elements which are different. In instances where the element that has changed is known, ZAP can be directed to update *only* that element and ignore any other comparison of the file. This can increase significantly to the speed at which ZAP functions.

The **-e** option is used to update a specific element. If specifying an element name that contains spaces, it *must* be enclosed in quotes. To combine multiple element updates in one command, define a plain text file list of elements, with one element per line and no quotes, and use it with the **-E** option instead. However, *never* use the **-e** and **-E** options together.

See the *Media Processing Server Series System Reference Manual* for details on as well as advanced uses of ZAP.

How to ... **zap a MultiMedia File**

1. **Designate a master MMF file as a reference file.** This file can exist on any node in the network.

2. **Make additions, deletions, and modifications to this designated file *only***, preferably using PeriStudio (see ***Updating/Replacing a Playback MMF Playback File*** on page 102., see ***Updating Elements in an Active Playback MMF File*** on page 107.), and/or the *PeriStudio User's Guide*.

All changes affecting the particular MMF file ***must*** be made to the designated reference file in order for the ZAP utility to be effective. ZAP also requires the `/etc/vpsrc.sh` file on every node that will be synchronized. This file is usually present as part of the standard Media Processing Server Series installation.

3. **To ZAP all nodes** in the `$MPSHOME/common/etc/vpshosts` file on the host node, **use the command line syntax `zap <mmf_name>`** from the command line of the host node on which the reference file resides, and where `<mmf_name>` indicates the path and name of the reference file noted in step 1.

To ZAP all Media Processing Servers in the `$MPSHOME/common/etc/vpshosts` file **of a specific node**, **use the command line syntax `zap -n <node_name> <mmf_name>`** from the command line of the host node on which the reference file resides, and where `<node_name>` indicates the target node.

To ZAP a specific Media Processing Server, **use the command line syntax `zap -v <mps#> <mmf_name>`** from the command line of the host node on which the reference file resides, and where `<mps#>` indicates the desired Media Processing Server.

To ZAP all Media Processing Servers in an alternate `vpshosts` file, use the command line syntax `zap -f <filename> <mmf_name>` from the command line of the host node on which the reference file resides, and where `<filename>` indicates the name (if located in the present working directory) or path and name of the alternate file.

The alternate file *must* be in the same format of the actual `vpshosts` file.

To ZAP MMF files using a pre-existing MAC file (file created from master MMF and used as a reference throughout synchronization), **use the command line syntax `zap -m <filename>`** from the command line of the host node on which the reference file resides, and where `<filename>` indicates the name or path and name of the MAC file.

The option/argument pairs above and that follow may be used in combination in a

single command. They may also be used in conjunction with the following:

Option	Description
`-d <delay>`	The delay interval (in minutes) for each retry attempt. The minimum allowable value is 10 minutes (default is 30)
`-r <retries>`	Number of times to again attempt a synchronization if a remote node fails to respond or MAC file cannot be transmitted (default is 3)
`-t <timespec>` Examples: **`-t 11pm`** **`-t 5am Wednesday`** **`-t 11:59pm march11`**	Specifies the date and/or time that the sync should be executed. The time is specified as one, two, or four digit numbers (where one and two digit numbers are hours and four digit numbers are hour and minute, separated by a colon). The am/pm identifier can be appended to the time, otherwise 24 hour time is assumed (i.e. 6:00 is 6am, 18:00 is 6pm). The date is optional and is specified as either the month name and day number or the day of the week. The current date is assumed if no date is specified
`-A`	Generate alarms upon completion (whether successfully or not) of synchronization of each Media Processing Server.

How to ... zap MultiMedia Files on a Proxy Basis

1. **Designate a master MMF file as a reference file.** This file can exist on any node in the network.
2. **Make additions, deletions, and modifications to this designated file *only***, preferably using PeriStudio (see ***Updating/Replacing a Playback MMF Playback File*** on page 102., see ***Updating Elements in an Active Playback MMF File*** on page 107.), and/or the *PeriStudio User's Guide*.
3. **Define a `zap.networks.cfg` file** (see ***ZAP by Proxy (Using Groups)*** on page 112.).
4. **Issue the `zap <mmf_name>` command** from the command line of the host node on which the reference file resides, and where `<mmf_name>` indicates the path and name of the reference file noted in step 1.

To ZAP all nodes in a specific group, **use the command line syntax `zap -G <group_number> <mmf_name>`** from the command line of the host node on which the reference file resides, and where `<group_number>` indicates the group as defined in the `zap.networks.cfg` file and identified in ascending sequential order starting at 1.

How to ... **zap a Specific Element**

1. **Designate a master MMF file as a reference file.** This file can exist on any node in the network.
2. **Make additions, deletions, and modifications to this designated file *only***, preferably using PeriStudio (see ***Updating/Replacing a Playback MMF Playback File*** on page 102., see ***Updating Elements in an Active Playback MMF File*** on page 107.), and/or the *PeriStudio User's Guide*.
3. **Determine the element name or number** that needs to be updated.
4. **Issue the `zap -e {@ <EAP_number> | <"Element Name">} <mmf_name>` command** from the command line of the host node on which the reference file resides, and where `<mmf_name>` indicates the path and name of the reference file noted in step 1. If specifying an element name that contains spaces, it *must* be enclosed in quotes.

To ZAP multiple specific elements in one command, issue `zap -E <filename> <mmf_name>` from the command line of the host node on which the reference file resides, and where `<filename>` indicates the name (if located in the present working directory) or path and name of the plain text element list file.

Elements can be listed by EAP number preceded by the @ sign or by name. Do not use quotes, even if there is a space in the name. Only one element must be listed per line.

Do not use the upper case `-E` option with the lower case `-e` option: these two must not be combined.

Caller Message Recording

Caller Message Recording (CMR) allows an application to record messages from the caller and save them in an MMF file. For details about the CMR feature and MMF files, see the *Media Processing Server Series System Reference Manual*.

Before an application can use CMR, an MMF file has to be created (see ***Creating a File for Caller Message Recording*** on page 117.) and activated (see ***Activating Record MMF Files*** on page 119.).

Caller Message Recording (CMR) is a Avaya licensed feature and requires the appropriate licensing permissions prior to its use!

Creating a File for Caller Message Recording

Use a Media Processing Server shell to create an MMF file to use for Caller Message Recording.

How to ... Create a MultiMedia Format File for CMR

1. **Open a Media Processing Server shell on the Media Processing Server that will hold the MMF file**. For additional information, see ***Command Control*** on page 50..
2. **Use the syntax `mkmf <mmf_file>`** where `<mmf_file>` is a descriptive name for the MMF file.

This command creates an expandable MMF file with the system default settings (2048 elements, 2048 data blocks). Alternatively...

3. **Use PeriStudio to create the MMF file**. **To create a new MMF file:**

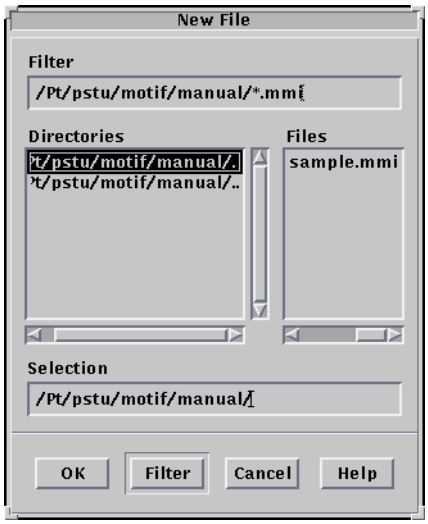
a. Start PeriStudio.

b. Select File—New. This opens the New File window.

c. Use the New file window to navigate to the directory where the new file is to be created.

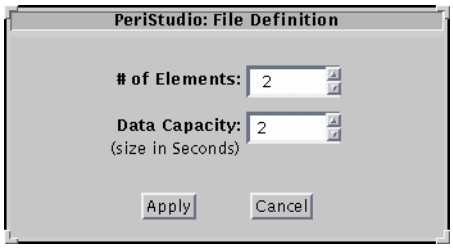
d. Add the name of the new file to the path in the Selection field. The `.mmi` extension is not needed.

e. Click the OK button. The File Definition window appears.



f. Optionally, specify different parameter values for the # of Elements and Data Capacity fields if the defaults are inappropriate.

g. Click the Apply button to proceed with creating the file. At this point, the system physically creates the new MMF file on the system disk in the specified directory. The file is automatically opened for immediate use. Elements may now be added and recorded.



See the Media Processing Server Series *Command Reference Manual*, for details about **mkmf**. Consult the *Media Processing Server Series System Reference Manual* for details about CMR and MMF files. Refer to the *PeriStudio User's Guide* for information on developing and manipulating MMF files in a graphical environment.

Activating Record MMF Files

Use a Media Processing Server shell to activate the record MMF file for a specific application or for all applications on the system.

How to ... Activate Record MultiMedia Files

1. **Open a Media Processing Server shell on the Media Processing Server that holds the MMF file**. For additional information, see ***Command Control*** on page 50..
2. **Activate the record MMF file with the `vmm mmfrec` command**.
   - To allow a specific application to record into an MMF file, use the syntax **`vmm mmfrec <mmf_file>,<app_name>`** where `<mmf_file>` is an empty MMF file and `<app_name>` is the name of the application that will record into `<mmf_file>`. The file should be placed in the appropriate subdirectory under `$MEDIAFILEHOME`. (Only a single MMF file should exist in each record directory.)
   - To allow all applications on the Media Processing Server to use an MMF file for recording, use the syntax **`vmm mmfrec <mmf_file>[,system]`** where `<mmf_file>` is an empty MMF file. (It is not necessary to specify the "**`system`**" parameter as this is the default.)

The **`vmm mmfrec`** command activates the specified MMF file automatically. A separate **`mmfload`** command is not needed.

MMF files can take up a considerable amount of disk space. If it is decided to use a different MMF file for recording, deactivate the unused MMF file. For more information, see ***Deactivating MMF Files*** on page 98.. See the *Media Processing Server Series System Reference Manual* for details about CMR and MMF files.

6

Backing up and Restoring Files

This chapter covers:

1. **Directory and File Manipulation**
2. **Backup Devices and Options**
3. **Guidelines for Backup**
4. **Backup/Restore Methods**
5. **Backing Up and Restoring MultiMedia Elements**
6. **Backing Up and Maintaining a Mirrored System Drive**

Backing up and Restoring Files

System and vocabulary files have to be backed up on a regular basis to ensure uninterrupted service. This section details backup options and procedures on Solaris systems and briefly discusses the Avaya Media Processing Server Series directory structure. (Refer to a standard Solaris guide for more details on Solaris/UNIX topics.) This chapter also discusses the backup and maintenance of systems which are equipped with an optional mirroring configuration.

Directory and File Manipulation

The files on the Media Processing Server Series system are organized into tree-structured directories and subdirectories like the ones on many personal and mini computers. Each directory generally contains files that are related to a common function, though the number of files in a directory varies. The base directory is the *root* directory.

For the layout of the Media Processing Server directory structure, see ***MPS Directory Structure*** on page 168.

Any *Solaris* commands entered in a Media Processing Server shell are issued to the local Solaris node *regardless of the current component*. For example, if the current component is `vas.1` and `grouse` is the name of the current node, but the VSH Command Line was launched on node `is9509`, **`ls`** lists the files in the directory on `is9509`, not on `grouse`.

Working with Directories

Enter commands to move around the directory tree and interact with the Media Processing Server. Think of the logical location on the directory tree as being "in" that directory. This directory is the *present working directory* (*pwd*), also referred to as the *current working directory.*

Enter **`pwd`** at a command line to display the full name of the present directory. The full name of the directory is the *path name*. It includes the directory names leading from the *root* to the *present working directory* (that is, the *path*).

The Media Processing Server Series system uses a slash character (`/`) to separate directory names. The *root* directory is always indicated as `/`.

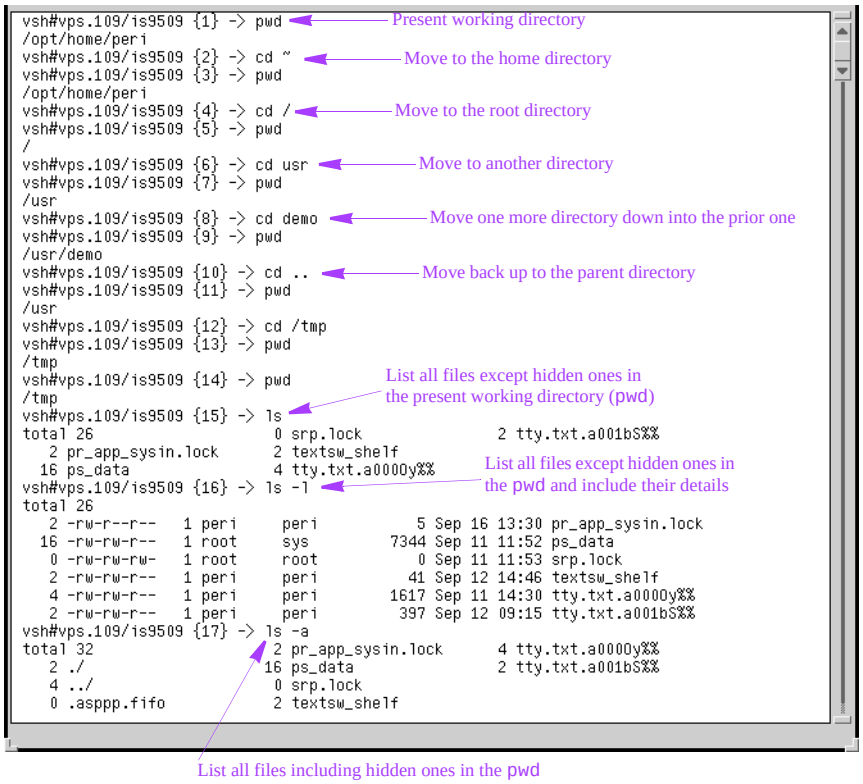
When a leading slash character ("`/`") is used to specify a directory, it implies the full path name. When in the *root* directory, the leading slash can be omitted in the path name when identifying a directory one level lower down in the tree.

Move between directories with a *change directory* (*cd*) command. Use the syntax **`cd <pathname>`** to change from the present working directory to the specified

directory.

- The *pathname* may be either a full *pathname* from the *root* (starts with a leading **/**) or a *relative* name to directories further down on the directory tree (starts without a leading **/**).
- Use the command **cd /** to move to the *root* directory from any other directory.
- To specify the *parent* directory (that is, the directory that contains the *present working directory*), use the **..** indicator. For example, to move up to the *parent* directory, use the command **cd ..** .
- If a pathname is not specified, the system defaults to the *present working directory.*
- Use the tilde (**~**) to return to the *home* directory by typing in **cd ~** and pressing <RETURN>. To determine the value of the *home* directory, enter **echo $HOME** at the command line.

The following illustrates these commands. If viewing this manual on-line, click on an explanation below to find out more about the corresponding command.



File Listing

Use the list command (**ls**) to display a list of files in a directory. Modify the output of **ls** by adding *switches*. Common **ls** commands are:

command	meaning
ls	lists all the files in the working directory, except hidden files
ls -l	lists all the files in the directory with their size, the time they were last modified, and their file permissions
ls -a	lists all the files in the working directory, including hidden files

See the illustration on page 99 for examples of these commands.

If a *pathname* is specified in the **ls** command, the system lists only the files in that directory. In addition, the switches can be combined (i.e., **ls -al**).

File Removal

Use the remove command (**rm**) to remove an unnecessary file. Use the syntax **rm -i <filename>**.

If files are removed, they cannot be recovered unless there is a backup copy.

Other commands can be used to remove (empty) directories. However, all the directories on the Media Processing Server are important to the system.

Do not remove any directories without consulting the local Media Processing Server Series support staff and application developers. Never remove any system-level directories unless instructed to do so during system upgrades.

To remove a directory, use the command **rmdir -i <directory_name>**. To remove a directory and all its files *and* subdirectories, use the command **rm -ir <directory_name>**.

The -i switch shown above causes the system to provide a warning prompt before removal of any directories or files. It is important to use this switch when using the rm and rmdir commands; otherwise, the result may be the removal of such files *without* prompting.

For more information about these options and possible scenarios, refer to the Solaris documentation of these commands.

Backup Devices and Options

To perform routine backups, the Media Processing Server is delivered with a 4mm Digital Audio Tape (DAT) drive, a remote DAT drive and a secondary hard drive.

Each device has a unique logical name:

Device	Name	Capacity	Used For
4mm DAT drive	`/dev/rmt/0`	12 GB (125 meters) - DDS3 (systems with an AXmP application processor) or 20 GB (150 meters) - DDS4 (systems with a SUN Blade, SUN Fire and IBM application processor).	weekly backup of all system and application files.
Remote DAT drive	`/dev/rmt/0`	20 GB (150 meters) - DDS4 (systems with SUN Fire, SUN Netra, IBM's (X335) application processor).	weekly backup of all system and application files
Secondary drive	`/dev/dsk`	varies with system configuration. (systems with Sun Fire, Sun Netra, IBM's (X335) application processor).	daily or weekly backup of all system and application files.

Tapes have a write-protect feature, must be disabled for the media to store additional data. For details, see the instructions included with the media.

Guidelines for Backup

At a minimum, back up the system files weekly to ensure uninterrupted service from the Media Processing Server. If any extensive changes are made in the interim, back that file system up immediately afterward. Perform backups when the system is relatively inactive (off-peak). For ease of use and accountability, Avaya suggests backing up each slice (file system) on its own tape.

In the following passages, the term `vps` is used in the discussions of path names, environment variables, and file fields. This terminology is equivalent to Avaya Media Processing Server nomenclature and directly relevant to MPS systems.

Command Summary

To back up...	to...	change directory to...	use...
system files	tape	`/opt`	**`tar cvf /dev/rmt/0 /opt`** **`ufsdump 0cf /dev/rmt/0 /opt`**[1]
MultiMedia files	tape	`/<mmfdirectory>`	**`tar cvf /dev/rmt/0 /<mmfdirectory>`** **`ufsdump 0cf /dev/rmt/0 <mmfdirectory>`**[a]
MultiMedia (vocabulary) elements	VGEN incremental format		PeriStudio. For additional information, see ***Backing Up and Restoring MultiMedia Elements*** on page 143.
user files	tape	`/opt/home`	**`tar cvf /dev/rmt/0 /opt/home`** **`ufsdump 0cf /dev/rmt/0 /opt/home`**[1]
application files [b]	tape	`/opt/vps/mpsN/apps` `/opt/vps/vasN/apps`	**`tar cvf /dev/rmt/0 /opt/vps/mpsN/apps`** **`tar cvf /dev/rmt/0 /opt/vps/vasN/apps`** or **`ufsdump 0cf /dev/rmt/0 /opt/vps/mpsN/apps`**[1] **`ufsdump 0cf /dev/rmt/0 /opt/vps/vasN/apps`**[1]
operating system [c]	tape	`/`	**`ufsdump 0cf /dev/rmt/0 <file_sys_name>`**[1]
mirrored system drive[d]	tape	`N/A`	**`mirrorbu`**

a. The examples shown are for backup on a local machine. For information about using **`ufsdump`** remotely, see ***Backing up with ufsdump*** on page 136..

b. Systems running database and CMR applications require more frequent backups of these files. CMR and associated database files *must* be backed up together while the applications that use them are **NOT** running.

c. The entire operating system might not fit on one tape if using the **`tar`** command (see ***Backing Up with tar*** on page 134.). If there is a potential space problem, use the **`ufsdump`** command, which can span multiple volumes of media (see ***Backing up with ufsdump*** on page 136.). To determine if a directory will fit on a single tape, see on page 126..

d. Refer to see ***Backing Up and Maintaining a Mirrored System Drive*** on page 144.and the *MPS Disk Mirroring Installation, Configuration and Recovery* manual for more information.

Determining Directory Size

To determine if a backup tape has enough space to hold a specific directory, use the **du** command to display the directory's size. For example, to determine the size of the system files directory `/opt`, execute the following command:

```
du -ks /opt
```

The **-k** option displays the size of the directory in kilobytes; the **-s** option suppresses the output from listing all individual files contained in the directory.

To determine the amount of disk space allocated to a particular file system, use the **df** command. The resulting output provides the total amount of space in kilobytes, the number of kilobytes used and available, and the amount of space used as a percentage of total available space.

Backup Tips

- Keep a supply of sequenced and labeled tapes on hand.
- Store backup media in a safe place. This media needs to be obtainable by service personnel if restoration of data is required.
- Write the date and names of files backed up on the label of each tape.
- Keep a log of backups, including dates and file names.
- Handle backup media carefully. Follow all instructions included with the media.
- Enable write protection on the tapes or disks after the backup. Follow the instructions included with the media.

Backup/Restore Methods

The following are the methods for the backup and restoration of files. (See the see ***Guidelines for Backup*** on page 126..)

Method	Used For	For Information
`disk to disk`	The `clone.pl` script to backup data from one disk to another, on non-mirrored systems where no tape drive is present, and a backup disk is available.	***See see Disk to Disk Backup Procedure*** on page 128.
`tar`	Transferring files to a tape or an internal drive. Use **`tar`** to archive files to a single 'tar' file. The **`tar`** command *does not* perform any file compression or span multiple media.	See see ***Backing Up with tar*** on page 134.
`ufsdump ufsrestore`	Transferring files to a tape, internal drive or disk, or diskette. Whereas the **`ufsdump`** command does not perform file compression, it *does* detect end of media and span multiple copies. This is useful for very large file backups.	See see ***Backing up with ufsdump*** on page 136.
`cp` (copy)	Backing up individual files or file groups to other internal drives or directories.	See see ***Backing Up with cp (Copy)*** on page 143.
PeriStudio	Backing up select elements in MultiMedia files.	See see ***Backing Up and Restoring MultiMedia Elements*** on page 143.
`mirrorbu`	Backup script to create a full backup of the mirrored system drive.	See see ***Backing Up and Maintaining a Mirrored System Drive*** on page 144.

Disk to Disk Backup Procedure

The following procedures describe how to use the `clone.pl` script to backup data from one disk to another, on non-mirrored systems where no tape drive is present, and a backup disk is available. All backups should be conducted when the system is as quiet as possible. It is best to run backups in single-use mode, if possible.

Preparation for Solaris Backup

1. Reconfigure boot to ensure all device files have been created for all connected disks:

This step is not needed on every backup, it is only required when new devices are added to ensure the system sees them.

```
# touch /reconfigure
# reboot
```

-or-
at the ok prompt: **`boot -r`**

2. Run backup script with `-h` option to get information on the drives in the system and all the command options. The following is the output of running **`clone.pl -h`** on a typical system.

`# perl clone.pl -h`
Sample output

```
Usage:
-L # Set DEBUG Logging Level (1-4):
1-INFO 2-WARN 3-ERROR 4-INFO&Console.
Default level: ERROR
-D #Set DAT Level (1-2):
1-detail 2-high level.
Default level: detail
-p 'PARTITIONS' Set partitions to be
backed up or recovered
Delimiters: space ',' ';' and '.'
Each partition specified from root,
i.e. begun w. '/'
Enclose [PARTITIONS] within single
quotes, not the back quote.
-d driveSet drive name for backup or recovery
-s driveSet drive name for system when booted
off CD
-b on/offSet backup on/off
-r on/offSet recovery on/off
Mandatory: recovery ON works only
backup set OFF
Restore to root or /usr requires all
applications to be down
-v #Get the version of this script
and perhaps a reboot.
-f fullSet full disk to be backed up
Mandatory: '-f full' needed for entire
disk backup
-hPrint this usage message
```

3. Select a drive as the target backup drive.

   Find one from the list, in the output of **`perl clone.pl -h`**

   Example:

```
System drives information:
System drive:c0t3d0.
The mount slices:c0t3d0s0 c0t3d0s6 c0t3d0s4 c0t3d0s7
                            c0t3d0s5.
Corresponding partitions:/ /usr /var /mmf /opt.
The swap slice:c0t3d0s1.
```

```
All the potential backup drive(s):c0t0d0 c0t1d0
                                  c0t2d0 c1t0d0 c1t3d0.
The possible backup drive(s) for 'dd' commands:
c0t1d0 c0t2d0
```

4. To perform a **dd** backup perform step 1 under ***Execution*** on page 131.

5. For **tar** backups, format the target backup drive.

   Make sure the backup disk is partitioned equally or bigger than the system disk, use the format command. Partitioning the backup drive is a 'one time' event, but whenever performing the **tar** backup the backup slices should be cleared with the **newfs** command below.
   For each formatted slice, except the swap slice, execute **newfs** to create a mountable ufs.

   Example:
   # newfs /dev/rdsk/c0t0d0s0

   Do not execute **newfs** on the backup swap slice, for example: c0t0d0s1 or the 's2' slice c0t0d0s2.

Execution

There are various command options when running **clone.pl** to backup data from disk to disk. To get help on the **clone.pl** script refer to step 2 under ***Preparation for Solaris Backup*** on page 128.

1. Backup Entire Disk – The '-f' option determines which method to use, either **dd** for drives of the same geometry or **tar** for drives with different geometry.

   Performing an entire disk backup on a system with only one target drive available.

   ```
   #perl clone.pl -f full
   ```

   Performing an entire disk backup on a system with more than two drives. There are two ways to specify the backup drive. Either specify the target drive in the command line or configure it in the clone.cfg file. Below is a sample command line for the first case.

   ```
   #perl clone.pl -f full -d c0t1d0
   ```

   To configure **clone.pl** using the clone.cfg file, see ***Configure*** below.

2. Backup Partitions using **tar**. For any backup not using **dd** complete step 5 under ***Preparation for Solaris Backup*** on page 128 for formatting and creating a file system on partitions.

   Performing a partition backup: the command below shows how to backup partitions (using **tar**) /var /usr on backup drive:c0t0d0 with all the debug options on.

   ```
   #perl clone.pl -p ' /var  /usr ' -L 4 -d c0t0d
   ```

   The command below will backup the entire disk using **tar**. It is an entire disk backup because all partitions on the system drive are chosen. When backing up the root partition '/' the boot block is automatically installed. This method of **tar** backup is the safest for backing up a system still running in multi-user mode (system up), this is due to **tar**'s ability to copy "open files".

   It best to perform this backup process during a "quite" time as to not impact system performance.

   ```
   #perl clone.pl -p '/ /var /opt /usr /mmf' -L 4 -d
   c0t0d
   ```

3. Recover Partitions using **tar**.

   Recover partitions /var and /usr.

   ```
   #perl clone.pl -b off -r on -p '/var /usr' -L 4 -d
   c0t0d0
   ```

4. Recover Files

   Recovery of files using the **clone.pl** script is the same as step 3 under ***Execution*** on page 131, this restores the entire partition to recover the needed files. Otherwise, to obtain individual files mount the backup partition with the files you wish to recover and copy them back to the original location manually.

5. Recover Partitions/Files/Root when booted off CD

   If the system drive is corrupted, boot the system off the CD and do the recovery. If `clone.cfg` is saved, copy `clone.cfg` to `/tmp`. Otherwise, manually mount the slices on the system drive before executing **clone.pl**.

   Recover partitions /var /usr and root, when `clone.cfg` is in `/tmp`.

   ```
   #perl clone.pl -b off -r on -p '/ /var /usr' -L 4
   ```

   Recover partitions /usr and root, when no `clone.cfg` exists.

   ```
   #mount /dev/dsk/c0t0d0s0 /tmp/ c0t0d0s0
   #mount /dev/dsk/c0t0d0s6 /tmp/ c0t0d0s6
   #perl clone.pl -b off -r on -p '/ /usr' -L 4
   ```

Configure

The `clone.cfg` file contains a list of all system and backup drives and all mounted slices information of the most recent backup procedure. The configuration file is created automatically after the successful completion of a **clone.pl** operation. If `$MPSHOME` is set then the `clone.cfg` file is stored in `/opt/vps/common/etc/clone.cfg`. Otherwise, the file is located in the `/tmp` directory. The content of this file can be changed.

```
#
# Example clone.cfg file.
#
# Note the backup drive specified in command line options
has high priority.
#
# TYPEDRIVE
#
systemc0t0d0
#
backupc0t1d0
#
# Mounted slices information
#
# TYPEMOUNTED ONFILESYSTEM
mountedfs/c0t0d0s0
mountedfs/usrc0t0d0s6
mountedfs/varc0t0d0s4
mountedfs/optc0t0d0s5
```

```
mountedfs/mmf2c0t1d0s5
#
# End of clone.cfg file.
```

Once the system is successfully backed up, save the `clone.cfg` file in case the system drive was totally corrupted.

Logging

1. Clone.dat

   Data file `clone.dat` is created during the backup or recovery operation, recording every clone operation. If `$MPSHOME` is set then the clone data file is stored in `/opt/vps/common/log/clone.dat`. Otherwise, the file is located in the `/tmp` directory. A sample `clone.dat` file is shown below.

```
Mon Mar 15 14:54:08 2004: ---------Start procedure logging ---------
Mon Mar 15 14:54:10 2004: OPTION -D DAT logging level set to HIGH
Mon Mar 15 14:56:53 2004: ---------Start procedure logging ---------
Mon Mar 15 14:56:55 2004: OPTION -D DAT logging level set to DETAIL
Mon Mar 15 14:56:55 2004: Current directory:/var
Mon Mar 15 14:56:55 2004: tar cvf - . 2>/opt/vps/common/log/tarc.out|(cd
/mnt/c0t1d0s4; tar xvfBp - >/opt/vps/common/log/tarx.out)
Mon Mar 15 15:00:54 2004: a ./ 0K
Mon Mar 15 15:00:54 2004: a ./lost+found/ 0K
Mon Mar 15 15:00:54 2004: a ./sadm/ 0K
.
.
.
Mon Mar 15 15:01:06 2004: x ./crash, 0 bytes, 0 tape blocks
Mon Mar 15 15:01:06 2004: x ./crash/tb020, 0 bytes, 0 tape blocks
Mon Mar 15 15:01:06 2004: Successfully tared /var to <c0t1d0s4>
```

   There are two levels for this type of logging which can be specified using the `-D #` option on the command line. To set the logging level, refer to step 2 under ***Preparation for Solaris Backup*** on page 128.

2. Clone.log

   The clone.log file, records debug information. The logging file is created during the backup or recovery operation. If `$MPSHOME` is set the `clone.log` file is stored in `/opt/vps/common/log/clone.log`. Otherwise, the file is located in the `/tmp` directory. A sample `clone.log` file is shown below.

```
Mon Mar 15 14:56:55 2004: add_to_partitions: Partition </var> will be
added to the following list
Mon Mar 15 14:56:55 2004: add_to_partitions: Partition list <>
Mon Mar 15 14:56:55 2004: add_to_partitions: Partition list updated as
</var>
Mon Mar 15 14:56:55 2004: grep_dir: The direct directory under root is
</var> for partition </var>
Mon Mar 15 14:56:55 2004: add_to_partitions: Partition </var> will be
added to the following list
```

```
Mon Mar 15 14:56:55 2004: add_to_partitions: Partition list <>
Mon Mar 15 14:56:55 2004: add_to_partitions: Partition list updated as
</var>
Mon Mar 15 14:56:55 2004: filter_partitions_input: Existing partitions
specified </var>!
Mon Mar 15 14:56:55 2004: filter_partitions_input: Valid partitions
under root </var>
Mon Mar 15 14:56:55 2004: grep_dir: The direct directory under root is
</var> for partition </var>
Mon Mar 15 14:56:55 2004: get_mount_sliceID: Slice <#4> found for
partition </var>
Mon Mar 15 14:56:55 2004: tar_nonrt_sli: Backup </var> on <c0t1d0s4>
Mon Mar 15 14:56:55 2004: do_mount: Successfully mounted c0t1d0s4 to /
mnt/c0t1d0s4
Mon Mar 15 14:56:55 2004: tar_nonrt_sli: Current directory:/var
Mon Mar 15 14:56:55 2004: tar_nonrt_sli: This will take a little bit
while to finish
Mon Mar 15 14:56:55 2004: crt_nonroot_arch: tar cvf - . 2>/opt/vps/
common/log/tarc.out|(cd /mnt/c0t1d0s4; tar xvfBp - >/opt/vps/common/
log/tarx.out)
Mon Mar 15 15:01:00 2004: wfiledat: Successfully saved operation file</
opt/vps/common/log/tarc.out> into /opt/vps/common/log/clone.dat.
Mon Mar 15 15:01:06 2004: wfiledat: Successfully saved operation file</
opt/vps/common/log/tarx.out> into /opt/vps/common/log/clone.dat.
Mon Mar 15 15:01:06 2004: crt_nonroot_arch: Successfully tared /var to
<c0t1d0s4>
Mon Mar 15 15:01:23 2004: do_umount: Successfully umount file system:/
mnt/c0t1d0s4
Mon Mar 15 15:01:23 2004: tar_nonrt_sli: Taring is done for </var> on
<c0t1d0s4>
Mon Mar 15 15:01:23 2004: main: Successfully backuped file system:c0t0d0
to c0t1d0. Done!
Mon Mar 15 15:01:23 2004: write_cfg: Created /opt/vps/common/etc/
clone.cfg containing backup/system drives and partitions's info.
```

There are four levels for this type of logging which can be specified using the `-L #` option on the command line. To set the logging level, refer to step 2 under ***Preparation for Solaris Backup*** on page 128.

3. Saving Old Log Files

   When the size of a log file exceeds 1000 KB, the **`clone.pl`** script saves the file to `file.date`. If there is a file called `file.date`, then the current `file.date` is renamed as `file.date.old` and the new file is saved as `file.date`. New logging files are created for future use.

Backing Up with `tar`

The **`tar`** command combines and saves files to a tape or internal drive.

`tar` does not support multi-volume backups. If there is a need to make backups beyond the capacity of the media, split the job into several backups (that is, back up individual partitions) or use the **`ufsdump`** command. For information on using this utility that *does* span multiple media (and which is especially useful for large backups), see see ***Backing up with ufsdump*** on page 136. To determine if a directory will fit on a single tape or disk, see see on page 126..

Do not backup to a medium that contains useful files. The `tar` command overwrites existing files.

How to ... Back Up with `tar`

1. **To make a tape-based backup**:
   a. Label the tape.
   b. Insert the tape into the drive.

2. **Log in as `root`.**
   a. From any command line enter **`su`**.
   b. Enter the `root` password (_____).

3. **Move to the directory where the files are located (the source directory).** See see ***Working with Directories*** on page 122..

4. **Enter the `tar` command**.
   - To backup to a tape, use the syntax **`tar cvf /dev/rmt/0 <sourcefiles>`**.
   - To backup to the *present working directory*, use the syntax **`tar cvf <archivefile.tar> <sourcefiles>`**. Use **`cp`** to place the tar file in an alternate directory or drive.

Use a descriptive file name with the `.tar` extension for <`archive-file.tar`> (for example, `oldvocabs.tar`). This command combines and saves the source file(s) in the archive file <`archivefile.tar`>.

Specify either individual files or entire directories to be backed up. Backups can also be made to an alternate directory or internal drive. See the Solaris system documentation or details about the **`tar`** and **`cp`** commands.

Displaying the Contents of a `tar` Backup

Display the table of contents for each backup to ensure that it is readable and complete.

How to ... Display the Contents of a `tar` Backup

1. **To display a tape backup, insert the tape into the drive**.

2. **Log in as `root`.**
   a. From any command line enter **`su`**.

b. Enter the `root` password (_____).

3. **Enter the `tar` command**.
   - To verify the contents of a tape, enter **`tar tvf /dev/rmt/0`**.
   - To verify the contents of an internal drive, move to the directory that contains the tar file and use the syntax **`tar tvf <archivefile.tar>`** where <`archivefile.tar`> is the tar file.

This command verifies each file stored in the archive file and prints its name and size to the screen.

The contents of a tar file in an alternate directory can also be verified. See the Solaris system documentation for details about the **`tar`** command.

Restoring with `tar`

Use the **`tar`** command to restore selected files or the entire backup.

How to ... Restore Files with `tar`

1. **If restoring a tape backup, insert the tape into the drive**.
2. **Log in as `root`.**
   a. From any command line enter **`su`**.
   b. Enter the `root` password (_____).
3. **Move to the directory where the files will be located (destination directory)**.
4. **Enter the `tar` command**.
   - To restore files from a tape, enter **`tar xvf /dev/rmt/0`**.
   - To restore files from an internal drive, use **`cp`** to place the tar file in the *destination directory* and use the syntax **`tar xvf <archivefile.tar>`** where <`archivefile.tar`> is the tar file.

See the Solaris system documentation for details about the **`tar`** command, and for details and examples on using **`cp`**.

Backing up with `ufsdump`

Use **`ufsdump`** to transfer files to a tape, internal drive or disk, or diskette. Whereas the **`ufsdump`** command does not perform file compression, it *does* detect end of media and span multiple volumes. This is useful for very large file backups. The systems provides prompting when a media source is full and an additional volume is needed.

If backing up a mirrored system drive, see
see ***Backing Up and Maintaining a Mirrored System Drive*** on page 144.

When running `ufsdump`, the file system *must* be inactive, otherwise, the output of `ufsdump` may be inconsistent and restoring files correctly may be impossible. A file system is considered inactive when it is unmounted or the system is in single-user mode. To access this mode, follow *step 4.* on page 24 **through *step 8.*** on page 25**, then enter the `boot -s` command at the ROM prompt.**

A file system is not considered inactive if one tree of the file system is dormant while another has files or directories under modification.

See the Solaris **`ufsdump`** documentation for important information concerning overlapping partitions if dumping to disk.

How to ... Back Up with `ufsdump` (locally)

1. **If backing up to a tape**:
   a. Label the tape.
   b. Insert the tape into the appropriate drive.
2. **Access a Solaris command shell**.
   a. Click <MENU> on the desktop.
   b. In the Workspace pop-up menu, click <MENU> on Programs.
   c. From the Programs menu, select Command Tool... .
3. **Log in as `root`.**
   a. From any command line enter **`su`**.
   b. Enter the `root` password (_____).
4. **Move to the directory where the files are located (source directory)** (see see ***Working with Directories*** on page 122.) Files must be contained in the same file system that is local to the system where **`ufsdump`** is being run.
5. **Enter the `ufsdump` command**. Typically, the syntax used is:

```
ufsdump <options> <arguments> <files_to_dump>
```

where **`<options>`** is a single string of one-letter options, **`<arguments>`** may be multiple strings associated with the options as determined by order, and **`<files_to_dump>`** specifies either the files, directories, or an entire file system to back up. For further information. see the table entry see ***Option <argument>*** on page 138.

How to ...Back Up with `ufsdump` (across a network)

Performing a ufsdump across a network may cause network degradation. Check with your network administrator first prior to attempting the backup.

Files can be backed from one machine in a network to a devices elsewhere in network.

1. **Follow *step 1.* through *step 4.*** Place the media into the drive of the machine that will store the backups. Access the Command Tool on the machine containing the files to be backed up.
2. **Enter the `ufsdump` command**, using the following syntax:

```
ufsdump <options>f <machine:device> <files_to_dump>
```

where **`<options>`** is a single string of one-letter options (of which **`f`** *must* be used for network backup), **`<machine:device>`** represents the name of the machine and the device ID on that machine where the files are to be dumped, and **`<files_to_dump>`** specifies either the files, directories, or an entire file system to back up.

Since **`ufsdump`** is normally run by `root`, the name of the local machine *must* appear in the `/.rhosts` file of the remote machine. The **`f`** argument can also be specified as **`user@machine:device`**, in which case **`ufsdump`** attempts to execute as the specified user on the remote machine. In this case, the specified user *must* have a `.rhosts` file on the remote machine, which allows the user invoking the command from the local machine to access the remote machine.

There are several important options that should be considered when using **`ufsdump`**. The following are not the only options, but can be considered the most important from the standpoint of backing up Media Processing Server Series-related information:

Option <argument>	Definition
`0-9`	Dump level of files. This can be used for comparative purposes during multiple or incremental dumps. A level **`0`** dump copies the entire file system to the dump file destination media.
`a <filename>`	Archive file. Creates a dump table of contents in the backup (archive) file, the name of which must be specified. This table of contents can later be used by **`ufsrestore`** to determine whether or not a file or directory exists on the backup. If this option is not used, **`ufsrestore`** can still read the contents of the backup directly from the media rather than the archive file. It is highly suggested that only Field Engineers use **`ufsrestore`**.

Option <argument>	Definition
c	Cartridge. Sets the defaults for cartridge tapes instead of the standard half-inch reel. ***This option should always be used when backing up to a Media Processing Server Series tape drive!***
D	Diskette. Dump to diskette. ***This option is case sensitive and should always be used when backing up to a diskette!***
f <filename>	Dump file. Specifies a file to dump to, instead of `/dev/rmt/0`: the file name *must* be specified. If the file is specified as "-", the backup is dumped to the terminal. If the name of the file is of the form `machine:device`, the dump is performed at the specified machine over the network. If the file is specified as `user@machine:device`, **`ufsdump`** attempts to execute as the specified user on the remote machine. See ***step 2.*** on page 110, for important information regarding `.rhosts` files and network backups.
S	Size estimate. Determines the amount of space in bytes needed for the backup without actually performing the transfer. Allows the operator to determine how many volumes of media will be needed for the dump. ***This option is case sensitive!***
v	Verify. After each tape or diskette is written, the contents of the media are verified against the source file system. If any discrepancies are found, the system will prompt for a new media and the dump/verification process will be repeated.

If no options are provided, the default definition of **`ufsdump`** is **`9uf / dev/rmt/0 <files_to_dump>`**.

- **To backup to a local tape**, use the following command *as a minimum* (other options may be added at the user's discretion):

```
ufsdump 0cf /dev/rmt/0 <files_to_dump>
```

- **To backup to another local device**, substitute that device name for **`/dev/rmt/0`** in the above example, and change the device options as applicable (that is, for a floppy backup, change option **`c`** to **`D`**).
- **To backup to a tape device on another machine in a network**, use the following command *as a minimum*:

```
ufsdump 0cf <machine:/dev/rmt/0> <files_to_dump>
```

- **To backup to a tape device as a particular user on another machine in a network**, use the following command *as a minimum*:

```
ufsdump 0cf <user@machine:/dev/rmt/0>
<files_to_dump>
```

- **To backup to another remote device**, substitute that device name for **`/dev/rmt/0`** in the above examples, and change the device options as applicable (that is, for a floppy backup, change

option **c** to **D**).

There are important considerations regarding `.rhosts` files when performing a remote backup over a network. See ***step 2.*** on page 110, for additional information.

For complete definitions of these and other options, including pertinent usage and ramifications, please see the Solaris **`ufsdump`** documentation.

Backing up partitions with `ufsdump`

Use **`ufsdump`** to backup all partitions on a non-mirrored system drive.

This is the only backup type that a "Certified Avaya Field Engineer" is authorized to use to restore or replace a damaged drive.

The filesystem must be inactive during ufsdump.

How to ... Back Up Partitions with `ufsdump`

1. **Determine the number of tapes required.** Execute **`df -k`** and add up the usage for all the displayed system disk slices, to determine how may tapes are required. The space allocation listed under the "used" heading are in kbytes when the **`-k`** option is used. Backup all system disk slices shown in the **`df`** command output, with the exception of swap. Swap does not need to be backed up because it contains temporary data. The mmf filesystem(s) may or may not be on the system disk, however, they should also be backed up.
2. **Check for keyboard/mouse.** Ensure that the system has either a console/keyboard/mouse connected or a laptop connected to "ttya" if you are bringing the system down to single user mode.
3. **Boot system.** Boot the system as single user or login as super user and have the system in an inactive state. (no calls or disk I/O taking place.)
4. **Backup filesystems.** Dump the filesystems to one or more tapes as determined
   in ***step 1.*** above. The example below illustrates the commands used to dump the four slices of a standard MPS boot drive on a single tape, your disk configuration may differ: (All the commands below use "ZEROs" not "OHs")

```
# ufsdump 0ucf /dev/rmt/0n /dev/rdsk/c0t3d0s0
```
(Wait for the prompt to return prior to issuing each command below)
```
# ufsdump 0ucf /dev/rmt/0n /dev/rdsk/c0t3d0s4
# ufsdump 0ucf /dev/rmt/0n /dev/rdsk/c0t3d0s5
# ufsdump 0ucf /dev/rmt/0n /dev/rdsk/c0t3d0s6
```

The example above uses the "**n**" option to stop the tape from rewinding. This allows each dump to be stacked on a single tape. To use a separate tape for each dump, remove the "**n**" and at the end of each dump replace the tape.

5. **Rewind the tape.** When all dumping for this particular tape is complete, rewind the tape with:

```
# mt -f /dev/rmt/0 rew
```

6. **Eject the tape and label the cover.** Write the date, hostname, and slice dump order on the label, and store the tape and partition table information in a safe and accessible location.

7. **Additional backups.** If necessary, continue dumping to additional tapes. The following is an example for dumping an MMFdisk:

```
# ufsdump 0ucf /dev/rmt/0 /dev/rdsk/c0t0d0s1
```

Displaying the Contents of a ufsdump Backup

Display the table of contents for each backup to ensure that it is readable and complete, and that a particular media volume contains the files and/or directories to be extracted.

How to ...Display the Contents of a ufsdump Backup

1. **Insert the backup media into the appropriate drive.**
2. **Log in as root.**
   a. From any command line enter **su**.
   b. Enter the root password (_____).
3. **Enter the ufsrestore command with the following options**:
   - To verify the contents of a tape directly, enter **ufsrestore tf /dev/rmt/0**. By not supplying any file names, the root directory is listed, resulting in a list of all files on the media.
   - To verify the contents of a tape that was backed up with the archive file option, enter **ufsrestore taf <archive_file> /dev/rmt/0**
   - To verify the contents of other media, substitute that media device name for **/dev/rmt/0** in the above examples.

These commands verify each file stored in the archive file and prints its name to the screen.

Restoring with `ufsrestore`

The **`ufsrestore`** command loads files, that were previously backed up with the **`ufsdump`** command, from tape into the current directory. The directory must be a stable filesystem. The following is an example of restoring an entire ufsdump file from a dump tape (`/dev/rmt/0`) to `/opt`.

How to ... Restore with `ufsrestore`

1. **Insert the backup media into the appropriate drive.**
2. **Log in as `root`.**
   a. From any command line enter **`su`**.
   b. Enter the `root` password (_____).
3. **Change to the directory where the ufsdump file should be restored to, in this example, `/opt`.**

```
cd /opt
```

If the system was booted from a CD, then mount the disk slice to restore to, for example:

```
mount /dev/dsk/c0t3d0s5/mnt
cd /mnt
```

4. **Make sure the tape is positioned at the beginning, so it can skip to the desired location.**

```
mt -f /dev/rmt/0 rew
```

5. **To determine if `/opt` is in the 3rd dump file on the tape, list the table of contents on the tape to see if it resembles, `/opt`.**

```
ufsrestore tfs /dev/rmt/0n 3
```

The norewind flag is required on the tape device.

6. **Rewind to the beginning of the tape.**

```
mt -f /dev/rmt/0 rew
```

7. **Run `ufsrestore`, starting at the 3rd file (assuming `/opt` was the 3rd file on the tape).**

```
ufsrestore rfs /dev/rmt/0n 3
```

The norewind flag is required on the tape device.

8. **When the restore is complete, remove the following file:**

```
rm restoresymtable
```

9. **Rewind and eject the tape.**

```
mt -f /dev/rmt/0 rew
```

Backing Up with cp (Copy)

The Solaris **cp** command copies individual files or groups of files. Use this command to back up crucial files to alternate directories and to copy files between directories.

See the Solaris system documentation for details and examples about using **cp**.

Backing Up and Restoring MultiMedia Elements

Use PeriStudio to back up crucial elements in MultiMedia (vocabulary) files. This procedure is only efficient if one or two elements in an MMF file are extensive and/or critical. If more than a few elements have to be backed up, back up the entire file with **cp** or **tar**.

How to ... Back up MultiMedia Elements

1. **Launch PeriStudio**. From the command line enter: **peristudio &**
2. **Open the MultiMedia file with the elements to be backed up.**
3. **Select the elements.**
4. **Export the elements to a Vgen incremental backup file.**
   a. Select Vgen incremental format.
   b. For a digital system, select u-law encoding. For an analog system, select adpcm encoding.
5. **Use any of the backup methods described earlier to back up (store) the Vgen incremental file.**

For details about using PeriStudio, see the *PeriStudio User's Guide*.

If selected elements are backed up to a Vgen incremental file, these elements can be restored in the Media Processing Server shell.

How to ... Restore MultiMedia Elements

1. **Open a Media Processing Server 1000 shell on the system that contains the vocabulary file**. For additional information, see see ***Command Control*** on page 50.
2. **Use the restore procedure appropriate for the backup method used on the elements to restore the Vgen incremental backup file.**
3. **Update the existing MultiMedia file with the Vgen incremental backup file**. Use the syntax **vmm mmfupdate <existingvocab>,<backfile.def1>** where <existingvocab> is the existing vocabulary file and <backfile.def1> is the Vgen incremental backup file.

Backing Up and Maintaining a Mirrored System Drive

The **mirrorbu** script backs up mirrored systems. The following is an overview of mirrorbu execution. For complete details, refer to the *MPS Disk Mirroring Installation, Configuration and Recovery* manual.

- Mirrorbu can only be run on systems that have active, synchronized mirrors.
- Each filesystem (except root) is write-locked momentarily while one of its mirrors is taken offline. Anything that attempts to write to the locked filesystem hangs until it is unlocked, therefore, run mirrorbu during a time when the system is least active. Once the mirror is offline, the filesystem is unlocked (write allowable). The filesystem is **not** mirrored while offline.
- ufsdump is executed to dump the offline mirror to tape. Filesystems are dumped consecutively to the tape. If the tape becomes full, a prompt appears to insert another tape. To place each filesystem dump on a separate tape, specify the **-o** option on the `mirrorbu` command line. A prompt appears for a new tape for every filesystem when the **-o** option is specified.
- The mirror is brought back online when the ufsdump is complete, and mirror resynchronization occurs automatically.
- This procedure takes between 1 hour to several hours, depending on the amount of data being backed up.
- Mirrorbu dumps the mirrored system drive only. Backup the mmf filesystem(s) separately.
- If mirrorbu is invoked by a cron job, check the cron log in `/var/log` and the mirrorbu log in `/etc/mirrorBUdata`. Also check mail for the root user. The backup will not complete if it requires a tape change when invoked via cron.

Preparation

Perform the following steps prior to executing mirrorbu. Some information must be recorded for later restoration.

1. Login as root. (Execute su -).
2. Choose a backup time when the system is least active. Since the mirror is taken offline and dumped, it is not necessary to terminate applications. However, there is a time when the filesystem is locked that can cause an application write to hang. The filesystem is locked while the mirror is taken offline, then immediately unlocked. The offline mirror is then dumped and the mirror is taken online. Extensive writes to the unmirrored primary slice could generate a small performance degradation when the mirror is taken online and forced to synchronize.
3. Determine the number of tapes required. Execute **df -k** and add up the usage for all the displayed system disk slices, to determine the number of tapes required. The space allocation listed under the "used" heading is in kbytes when the **-k** option is used. All system disk slices shown in the **df** command output should be backed up, with the exception of swap. Swap does not need to be backed up because it contains temporary data.

4. Check the state of disk mirrors using the following script.

   ```
   # /etc/mirrorcheck
   ```

   Proceed with one of the following bullets, based on the mirrorcheck results:
   - If the output shows “mirrors appear ok”, proceed to mirrorbu execution.
   - If the output indicates states not ok, then do not continue with the backup. Have Certified Avaya Personnel correct the states prior to attempting to running mirrorbu.

Execution

Perform the following steps to execute mirrorbu.

1. Change to the `etc` directory:

   ```
   # cd /etc
   ```

2. Execute mirrorbu, specifying command line options when necessary.

   The mirrorbu command line options are as follows:
   `-o`

   one slice per tape. If this option is specified, a prompt occurs to insert a new backup tape for each slice. If this option is not specified, dump files are stacked on the same tape.

   `-t` <*tape-device*>

   tape device. Do not include a trailing n for norewind; this is handled by the script. If this option is not specified, `/dev/rmt/0` is used.

   Example:
   # **`./mirrorbu`**(uses the default `/dev/rmt/0`)
   or
   # **`./mirrorbu -t`** `/dev/rmt/1`(indicates the tape is on `/dev/rmt/1`)

3. A message is received with the log name. Watch the screen output and check the log for problems or errors when the backup is complete.

Sample Mirrorbu Screen Output

```
# mirrorbu
================================================================================
The log for this backup is /etc/mirrorBUdata/mirrorbu.out041002-
17:04:41
================================================================================
This script will backup filesystems by taking the mirror offline and
dumping it.
Redundancy will be lost for the duration of the backup and regained
when the mirror resynchs - after the backup.

note the size of /dev/md/dsk filesystems, you may require more than one
tape.
   Filesystems will be dumped in this order:
Filesystem       kbytes   used    avail    capacity  Mounted on
/dev/md/dsk/d0   481067   51671   381290   12%       /
/dev/md/dsk/d4   770187   9018    722660   2%        /var
/dev/md/dsk/d5   3843668  528061  3123424  15%       /opt
/dev/md/dsk/d6   1488162  723320  705316   51%       /usr
taking / /dev/md/dsk/d20 offline from /dev/md/dsk/d0
Dumping /  /dev/md/rdsk/d20.......please wait......
  DUMP: Writing 63 Kilobyte records
  DUMP: Date of this level 0 dump: Wed Apr 10 17:04:48 2002
  DUMP: Date of last level 0 dump: the epoch
  DUMP: Dumping /dev/md/rdsk/d20 to /dev/rmt/0n.
  DUMP: Mapping (Pass I) [regular files]
  DUMP: Mapping (Pass II) [directories]
  DUMP: Estimated 109936 blocks (53.68MB).
  DUMP: Dumping (Pass III) [directories]
  DUMP: Dumping (Pass IV) [regular files]
  DUMP: 109870 blocks (53.65MB) on 1 volume at 731 KB/sec
  DUMP: DUMP IS DONE
bringing  / /dev/md/dsk/d20 back online to /dev/md/dsk/d0
Locking /var
taking /var /dev/md/dsk/d24 offline from /dev/md/dsk/d4
unlocking unmirrored /var
Dumping /var  /dev/md/rdsk/d24.......please wait......
  DUMP: Writing 63 Kilobyte records
  DUMP: Date of this level 0 dump: Wed Apr 10 17:06:09 2002
  DUMP: Date of last level 0 dump: the epoch
  DUMP: Dumping /dev/md/rdsk/d24 to /dev/rmt/0n.
  DUMP: Mapping (Pass I) [regular files]
  DUMP: Mapping (Pass II) [directories]
  DUMP: Estimated 25432 blocks (12.42MB).
  DUMP: Dumping (Pass III) [directories]
  DUMP: Dumping (Pass IV) [regular files]
  DUMP: 25324 blocks (12.37MB) on 1 volume at 913 KB/sec
  DUMP: DUMP IS DONE
bringing  /var /dev/md/dsk/d24 back online to /dev/md/dsk/d4
Locking /opt
taking /opt /dev/md/dsk/d25 offline from /dev/md/dsk/d5
unlocking unmirrored /opt
Dumping /opt  /dev/md/rdsk/d25.......please wait......
  DUMP: Writing 63 Kilobyte records
  DUMP: Date of this level 0 dump: Wed Apr 10 17:06:30 2002
  DUMP: Date of last level 0 dump: the epoch
```

```
  DUMP: Dumping /dev/md/rdsk/d25 to /dev/rmt/0n.
  DUMP: Mapping (Pass I) [regular files]
  DUMP: Mapping (Pass II) [directories]
  DUMP: Estimated 1057522 blocks (516.37MB).
  DUMP: Dumping (Pass III) [directories]
  DUMP: Dumping (Pass IV) [regular files]
  DUMP: 82.28% done, finished in 0:02
  DUMP: 1057390 blocks (516.30MB) on 1 volume at 747 KB/sec
  DUMP: DUMP IS DONE
bringing  /opt /dev/md/dsk/d25 back online to /dev/md/dsk/d5
Locking /usr
taking /usr /dev/md/dsk/d26 offline from /dev/md/dsk/d6
unlocking unmirrored /usr
Dumping /usr  /dev/md/rdsk/d26.......please wait......
  DUMP: Writing 63 Kilobyte records
  DUMP: Date of this level 0 dump: Wed Apr 10 17:18:32 2002
  DUMP: Date of last level 0 dump: the epoch
  DUMP: Dumping /dev/md/rdsk/d26 to /dev/rmt/0n.
  DUMP: Mapping (Pass I) [regular files]
  DUMP: Mapping (Pass II) [directories]
  DUMP: Estimated 1497140 blocks (731.03MB).
  DUMP: Dumping (Pass III) [directories]
  DUMP: Dumping (Pass IV) [regular files]
  DUMP: 51.17% done, finished in 0:09
  DUMP: 1497130 blocks (731.02MB) on 1 volume at 637 KB/sec
  DUMP: DUMP IS DONE
bringing  /usr /dev/md/dsk/d26 back online to /dev/md/dsk/d6
Backup done....rewinding.....
```

Sample Log Output

```
taking / /dev/md/dsk/d20 offline from /dev/md/dsk/d0
d0: submirror d20 is offlined
Did /usr/opt/SUNWmd/sbin/metaoffline /dev/md/dsk/d0 /dev/md/dsk/d20
Dumping /  /dev/md/rdsk/d20.......please wait......
Did /usr/sbin/ufsdump  0cf /dev/rmt/0n /dev/md/rdsk/d20
bringing  / /dev/md/dsk/d20 back online to /dev/md/dsk/d0
d0: submirror d20 is onlined
Did /usr/opt/SUNWmd/sbin/metaonline /dev/md/dsk/d0 /dev/md/dsk/d20
Locking /var
Did /usr/sbin/lockfs -w /var
taking /var /dev/md/dsk/d24 offline from /dev/md/dsk/d4
d4: submirror d24 is offlined
Did /usr/opt/SUNWmd/sbin/metaoffline /dev/md/dsk/d4 /dev/md/dsk/d24
unlocking unmirrored /var
Did /usr/sbin/lockfs -u /var
Dumping /var  /dev/md/rdsk/d24.......please wait......
Did /usr/sbin/ufsdump  0cf /dev/rmt/0n /dev/md/rdsk/d24
bringing  /var /dev/md/dsk/d24 back online to /dev/md/dsk/d4
d4: submirror d24 is onlined
Did /usr/opt/SUNWmd/sbin/metaonline /dev/md/dsk/d4 /dev/md/dsk/d24
Locking /opt
Did /usr/sbin/lockfs -w /opt
taking /opt /dev/md/dsk/d25 offline from /dev/md/dsk/d5
d5: submirror d25 is offlined
Did /usr/opt/SUNWmd/sbin/metaoffline /dev/md/dsk/d5 /dev/md/dsk/d25
unlocking unmirrored /opt
Did /usr/sbin/lockfs -u /opt
Dumping /opt  /dev/md/rdsk/d25.......please wait......
```

```
Did /usr/sbin/ufsdump  0cf /dev/rmt/0n /dev/md/rdsk/d25
bringing  /opt /dev/md/dsk/d25 back online to /dev/md/dsk/d5
d5: submirror d25 is onlined
Did /usr/opt/SUNWmd/sbin/metaonline /dev/md/dsk/d5 /dev/md/dsk/d25
Locking /usr
Did /usr/sbin/lockfs -w /usr
taking /usr /dev/md/dsk/d26 offline from /dev/md/dsk/d6
d6: submirror d26 is offlined
Did /usr/opt/SUNWmd/sbin/metaoffline /dev/md/dsk/d6 /dev/md/dsk/d26
unlocking unmirrored /usr
Did /usr/sbin/lockfs -u /usr
Dumping /usr  /dev/md/rdsk/d26.......please wait......
Did /usr/sbin/ufsdump  0cf /dev/rmt/0n /dev/md/rdsk/d26
bringing  /usr /dev/md/dsk/d26 back online to /dev/md/dsk/d6
d6: submirror d26 is onlined
Did /usr/opt/SUNWmd/sbin/metaonline /dev/md/dsk/d6 /dev/md/dsk/d26
Backup done....rewinding.....
```

4. If the following message is received, then the current tape is full. Eject the tape (and label according to the next step), insert a new tape and then respond **yes** to the following prompt:

   ```
   DUMP: NEEDS ATTENTION: Is the new volume (#2) mounted on
   `SYSTEMNAME:/dev/rmt/0n' and ready to go?: ("yes" or "no") yes
   ```

5. After the tape rewinds, eject it and write the date, sequence number, and number of tapes in this backup set, on the label. For example, write:

   - [today's date] mirrorbu 1 of 1 (when everything fits on a single tape).
     or
   - [today's date] mirrorbu 1 of 2 (on first tape) and
     [today's date] mirrorbu 2 of 2 (on second tape).
   - Write the filesystem order on the external tape label. This helps identify the correct filenumber when restoring. The mirrorbu screen output, as well as the `mirrorbu.out` file corresponding to this dump, shows the order of filesystems on the dumptape.

6. The system drive backup is now complete, however, it may take some time to complete synchronizing the re-attached mirrors. Do not reboot the system and do not perform another system backup until the synchronization is complete. This can be checked with the command:

   ```
   # /usr/opt/SUNWmd/sbin/metastat | grep sync
   ```

   Sample output:

   ```
   State: Resyncing
   Resync in progress: 33 % done
   ```

   Intermittently execute the above **metastat** command until grep finds no match for sync.

7. Remember to backup any non-mirrored drive such as the MMF drive on a separate tape, since this is not a part of the mirrorbu script. For more information on backing up non-mirrored drives, see see ***Backup/Restore Methods*** on page 128.

This page has been intentionally left blank.

7

Troubleshooting

This chapter covers:

1. **Fault Categories**
2. **Isolating and Resolving Specific Problems**
3. **SRP Log File**

Troubleshooting

This section discusses categories of faults that may occur, some common fault conditions, causes, and suggested actions. Follow these suggestions before contacting Avaya technical support.

Fault Categories

Faults on the Media Processing Server fit in the following general categories:

- Vocabulary files
- File synchronization
- Host communication
- Telephone lines
- Application software related

Perform the following checklist actions to isolate problems. If necessary, supplement this procedure with site-specific actions to verify proper operation of other connecting telephone and computer systems. Use these guidelines only in the proper context of overall system-level problem determination.

Use this procedure after Avaya has completed the initial/upgrade installation and the system has been in an operational state with application software.

1. **Verify the operation of the system** with the following actions:

To verify that...	Perform this action...
System console is active and able to accept commands	Move the mouse pointer into a command window and press <RETURN> on the keyboard. The system should respond with the command prompt.
System answers calls and performs normally	Call the system and make sure that it processes calls normally. Make both normal and error-laden calls to verify all the functions.
System is processing calls in a normal manner and is not generating any alarms due to errors	Check the Alarm Viewer for alarm messages generated by incoming calls. Also examine the alarm log file to check for recent messages. see ***The Alarm Viewer*** on page 54.

To verify that...	Perform this action...
Status of startup and recovery processes, phone lines, voice memory, application programs, host lines, and vocabularies are within acceptable limits	Issue the following status commands and check for normal/abnormal results: • **`srp status`** • **`ccm status`** • **`vmm status`** • **`vmm MMF filestatus`** For additional information, see see ***Status Monitoring*** on page 56..
Operating parameters for the system are normal	Check the front and rear panel indicators on the cabinet, TMS, and VRC. For locations and functions, see see ***MPS 500 Indicators and Controls*** on page 30..

2. **If it is possible, isolate the problems into one of the Categories of Faults**. Refer to the common symptoms, cause, and actions for that category.
3. **If the problem cannot be isolated or repaired, or if it continues to occur, contact Certified Avaya Support Personnel**. Have the following information available:

- A description of the problem, including any error messages.
- The number of lines taking calls when the problem occurred.
- The system's serial number.
- The remote access telephone number.

Isolating and Resolving Specific Problems

This guide discusses commonly occurring problems across a variety of systems and configurations. Therefore, it may or may not address problems at a specific site.

Try to isolate the problem into one of the Categories of Faults. Referring directly to that section. Always check the Alarm Viewer and Alarm Log files for pertinent messages when using these sections (see see ***The Alarm Viewer*** on page 54.).

Analyze the causes in the order they appear in the tables.

MMF (Vocabulary) File Troubleshooting

Refer to the *PeriStudio User's Guide* for a detailed description of creating and troubleshooting vocabularies. The following addresses problems activating, deactivating, monitoring, and updating vocabularies.

Some problems that appear to be related to vocabulary files may be related to the application or system configuration.

Vocabulary Files Troubleshooting: Sheet 1 of 2

Condition	Cause	Action
An MMF file (vocabulary) file cannot be activated	Invalid MMF file name.	Verify the file name.
	File is corrupted.	Run **`mfck -f`** or MMF should be restored from the backup or copied from another MPS with a valid copy
An MMF file (vocabulary) file cannot be deactivated	Invalid application or MMF file name.	Verify the application and file names.
There is no output for the status of a vocabulary file	Invalid MMF file name.	Verify the file name.
	MMF file is not activated.	Activate the MMF file (see see ***Activating Playback MMF Files*** on page 87.).
The application speaks incorrect vocabulary items	Hash table contains elements with the same name.	Edit the vocabulary file or create an application-specific hash table (see see ***Activating Playback MMF Files*** on page 87.).
Online updates cannot be performed	The MMF file was opened read-only.	First, deactivate the MultiMedia (vocabulary) file (see see ***Deactivating MMF Files*** on page 98.). Make sure it is not open in MPS Developer or PeriStudio. Then, activate the file (see see ***Activating Playback MMF Files*** on page 87.).
New elements cannot be recorded	The MMF file is static and does not contain enough available space.	Delete unused elements from the MMF file. See the *PeriStudio User's Guide*.
	The MMF file is dynamic, but the file system is 98% full.	Delete unnecessary files from the file system where the MMF file resides (see see ***File Removal*** on page 124.).

Vocabulary Files Troubleshooting: Sheet 2 of 2

Condition	Cause	Action
An MMF file (vocabulary) file cannot be expanded	The MMF file is static.	A static MMF file cannot be expanded.
	There is not enough disk space on the file system. After expansion, the file system must have at least 2% available space.	Delete unnecessary files from the file system where the MMF file resides (see see ***File Removal*** on page 124.).
The application does not speak the elements	The elements have not been loaded (i.e., activated).	Activate the MMF file (see see ***Activating Playback MMF Files*** on page 87.).
All the elements do not fit into VDM when the MMF file is loaded (i.e., (activated)	There is not enough voice memory.	• Remove any elements not used by applications. See the *PeriStudio User's Guide*. • Set the **`vmm loadall`** command to **`off`**. This allows only elements with a lock flag set to be loaded into VDM (limits total number of elements loaded). • Set the **`vmm preload`** option to accommodate the **`loadall`** command (see previous bullet). If **`loadall`** is **`off`**, set **`preload`** to **`all`**. If **`loadall`** is **`on`**, the number of seconds to preload into VDM should be kept small if this condition occurs. The **`preload`** and **`loadall`** commands are set in the `$MPSHOME/mpsN/etc/vmm-mmf.cfg file` (see the note under see ***SRP Log File*** on page 161.). If this file is modified, VMM must be stopped and restarted for the changes to take effect (see see ***TMS Startup/ Shutdown*** on page 25.). • Adjust system voice memory size. Use the **`vmm pagesize <n_kbytes>`** command to set the size of a single segment of VDM. The smaller the size, the more efficient use of VDM, but the greater the use of system memory. • Lock the most commonly spoken elements into voice memory and page the remaining elements from cache. • Use the **`vmm vdmmaxlock <n%>`** command to stipulate the maximum percentage of VDM to use for locked elements. *The default (50%) is highly recommended.*

File Synchronization Troubleshooting

Automated MMF file synchronization is provided for all active, designated files through use of the Zero Administration for Prompts (ZAP) utility (see see ***Synchronizing MMF Files Across Nodes (ZAP Utility)*** on page 110.). This utility generates faults and exception conditions as part of its function. While some causes can be general in nature and not directly attributable to the utility itself, understanding the situation surrounding them can help alleviate possible repetition of the condition.

File Synchronization Troubleshooting (Sheet 1 of 2)

Condition	Cause	Action
Cannot connect to VMM	VMM is not running on the target node.	• Be sure VMM is up and running on subject node (see see ***VMM Status*** on page 71.). • Synchronization is rescheduled based on preconfigured delay interval.
Multiple instances of VMM have the MMF file activated	ZAP running on MMF files that are shared between MPS systems	• zap should not be run on MMF files that are shared between MPS systems. • If zap is to run on an MMF, each MPS must have it's own copy of the MMF
Update of an element failed	The target MMF file was not on-line when the update was attempted.	Make sure target MMF file is active for update (see see ***MultiMedia Format (MMF) File Status*** on page 72.).
	VMM has the MMF file opened in a read-only state.	Make sure the target MMF file has appropriate write permissions.
	See ***Multiple instances of VMM have the MMF file activated***.	Don't activate the same copy of the MMF file for multiple instances of VMM. Use different copies of the MMF file.
Deletion of an element failed	The target MMF file was not on-line when the deletion was attempted.	Make sure target MMF file is active (see see ***MultiMedia Format (MMF) File Status*** on page 72.).
	VMM has the MMF file opened in a read-only state.	Make sure the target MMF file has write permission available.
	See ***Multiple instances of VMM have the MMF file activated***	Don't activate the same copy of the MMF file for multiple instances of VMM. Use different copies of the MMF file.
Remote node is down	Remote node is not active or is unreachable.	• Synchronization fails after prescribed number of retries. • Make sure node is properly configured and running (see the *PeriView Reference Manual*). • Synchronization is rescheduled based on preconfigured delay interval.
Network connection is broken	Connectivity between the host and remote nodes has been lost.	• Resolve network connectivity issue. • File transfers previously interrupted are retransmitted once connection is re-established. • If transfer times-out, synchronization is rescheduled based on preconfigured delay interval.

File Synchronization Troubleshooting (Sheet 2 of 2)

Condition	Cause	Action
Target node reboots or crashes while processing the synchronization request	An exceptional condition that causes crash or necessitates a reboot occurs during synchronization request.	• Resolve node issues to get back online. • Synchronization process is rescheduled based on preconfigured delay interval. • Elements already processed are either updated again or, if previously deleted, have delete requests ignored.
Reference node reboots or crashes while processing synchronization request	An exceptional condition that causes crash or necessitates a reboot occurs during synchronization request.	• Resolve node issues to get back on-line. • If the reference node crashes/reboots while synchronization facility is creating the MAC file, process has to be manually restarted. • If the reference node crashes/reboots while the synchronization facility is distributing the MAC file, the distribution is automatically restarted after the system is back up based on the preconfigured delay interval.
No space on target file system to copy update package	Not enough file system or disk space left on target node.	• Free up space by deleting unneeded/unused files, especially vocabulary and log files (see see ***File Removal*** on page 124.). • Synchronization process is rescheduled based on preconfigured delay interval. • Process fails after the prescribed number of retries.
No space on reference node file system to create update package	Not enough file system or disk space left on reference node.	• Free up space by deleting needed/unused files (especially vocabulary files and log files that are no longer used) (see see ***File Removal*** on page 124.). • Synchronization process must be reinstated manually.
Synchronization failed due to retry limit being exceeded.	Any of the above.	Selectively synchronize the target node.

Host Communication Troubleshooting

Since each application communicates directly with the host, problems that arise may be logical (i.e., software or protocol related) rather than absolute (i.e., the physical interface).

Host Communication Troubleshooting

Condition	Cause	Action
MPS is not communicating with the host	The host link is physically disconnected.	Check all physical connections. Make sure they are secure.
	The host is down.	Wait for the host to start up.
	The host is up, but all terminals (LUs) are down.	Bring up all appropriate host applications.
	The protocol software is not functioning properly.	Call for maintenance service.
Host went down and came up but the MPS is not responding	The application does not handle the host-up condition properly.	Verify that the application properly handles host up/ down status messages.
Host error message (COMMGR) occurred during application run time	The host is sending delayed responses that the MPS is discarding.	The MPS discards messages sent in 24-byte header mode. This is normal.
	The application code contains incorrect host access commands.	Verify and correct the application host access sequences.
	There is not enough delay time between the host send and host receive calls.	Two send requests without an intervening receive request causes an error.
Excessive missed polls or retry sequences	The modem and clocking options are not correct.	Synchronize the MPS and modem clocking options.

For additional information on working with applications, see the *MPS Developer User's Guide*. For additional host-specific issues, refer to the *Media Processing Server Series COMMGR Reference Manual*.

Telephone Line Troubleshooting

The Media Processing Server telephone lines are either analog or digital. Some of the conditions in the following table do not apply to both types of lines. In some cases, application program logic may terminate calls in progress.

For additional information pertaining to telephone lines, see the *Media Processing Server Series Telephony Reference Manual*.

Telephone Line Troubleshooting

Condition	Cause	Action
A phone line or group of phone lines are down	A CCM software module has failed.	It is possible (but unlikely) that one CCM process exists for each phone line. In this case, the failure of an individual CCM leads to the same for its corresponding line. The CCM module should be restarted automatically by SRP. If it isn't, restart the MPS. (See see ***Routine Media Processing Server Startup*** on page 28..)
	A span is not working properly.	Contact the local Media Processing Server Series support staff.
Line does not answer incoming calls	The phone line is not connected.	Check the phone line connection.
	There is no application assigned to the line.	Assign and start an application on the line. (See see ***Assigning and Starting Application Programs*** on page 93..)
	The signalling protocol is improperly configured (digital system only).	Check the configuration of the signaling protocol.
Incoming line stays in a busy state	The active application is terminated (or not started).	Start the application. (See see ***Assigning and Starting Application Programs*** on page 93.***)***
	There is no application assigned to the line. The default state is busy.	Assign and start an application on the line. (see see ***Assigning and Starting Application Programs*** on page 93..
	The application keeps the state of the physical phone line in the not-in-service condition.	Check the coding of the application to determine how it handles the condition of the line when not in service. Otherwise, try stopping and restarting the application (see see ***Terminating and Unassigning Application Programs*** on page 97. and see ***Assigning and Starting Application Programs*** on page 93.).
	The line is not working.	Replace the physical component containing the line.
System does not display logging	There is no active call on the line.	Call in to the line and restart logging.
Call cannot be monitored	There are no active calls on the lines.	Call in to the line (wait for an active call).
	The telephone handset is faulty.	Use a different telephone handset.
	The **`ccm listen`** command was not issued correctly.	Issue the **`ccm listen in/out`** command. See the *Media Processing Server Series System Reference Manual* for details.

Application Software Troubleshooting

The application software on Media Processing Server Series systems is diverse. Often, problems are specific to a site or a particular program.

Application Software Related Troubleshooting

Condition	Cause	Action
Application does not answer incoming calls	The application is not started.	Start the application. See see ***Assigning and Starting Application Programs*** on page 93..
	The application is not attached to a physical phone line.	Attach call-processing applications to physical phone lines. See see ***Assigning and Starting Application Programs*** on page 93..
	The application does not contain coding to answer calls.	Add proper code to application. (See the *MPS Developer User's Guide*.)
Caller is disconnected prematurely either at the start or during the call	The MMF file is not loaded into Voice Data Memory.	See see ***Activating Playback MMF Files*** on page 87..
	Some vocabulary elements are corrupted or do not exist.	See see ***Updating or Replacing Playback MMF Files*** on page 102..
Application speaks the wrong vocabulary elements	The application code references the wrong items.	Review the application code. See the *MPS Developer User's Guide*.
	There are two like-named MMF files in the application or system hash table.	• Edit the MMF files so that all elements in the same hash table have unique names. (See see ***Updating or Replacing Playback MMF Files*** on page 102.. For a detailed description of hash tables, see the *Media Processing Server Series System Reference Manual*.) • Deactivate the MMF file with the duplicate elements. See see ***Deactivating MMF Files*** on page 98..
An MPS 2.0 application does not execute properly on an MPS 1.0 system	Some commands and functions (e.g., those pertaining to **`plm`**) from previous software releases are not supported in the MPS environment.	The application has to be recoded to accommodate new comparable commands or functions, or to remove those no longer supported. (For more information, see the *Media Processing Server Series Transition Guide* and *MPS Developer User's Guide*.)

SRP Log File

The Startup and Recovery Process (SRP) maintains a continuous record of processes monitored by SRP in the file `$MPSHOME/common/log/srp.log`.

This log file includes dates and times for process spawning, fatal alarms, and start/stop application requests

The `srp_state.log` file, located in the same directory, maintains a history of process state changes, including the date and time the change in state occurred.

For the complete Solaris Media Processing Server directory tree, see see ***MPS Directory Structure*** on page 168..

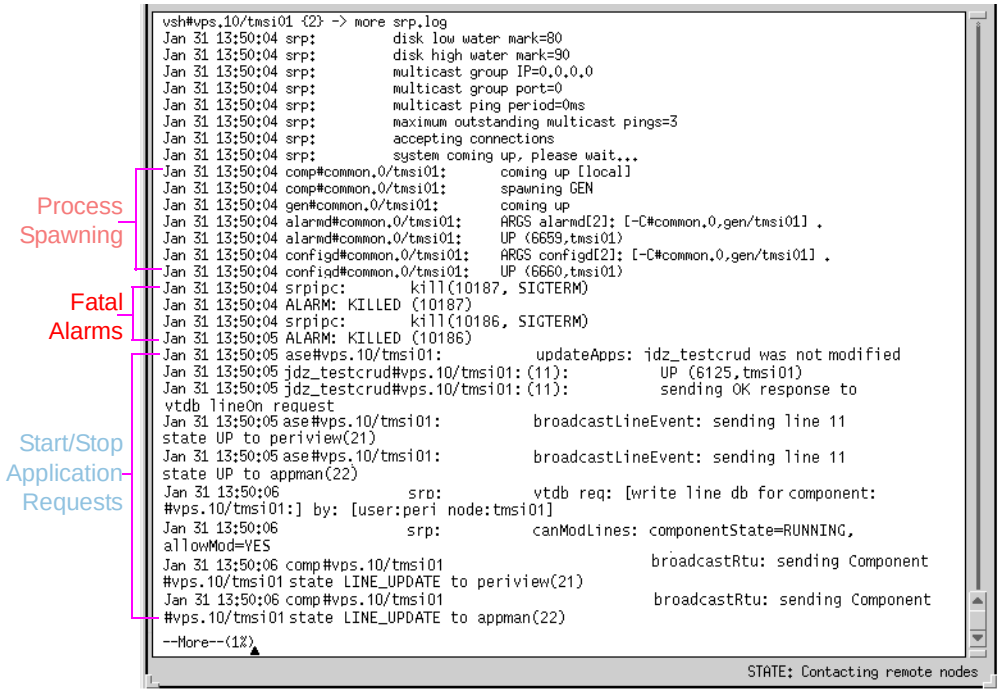
In the following passages, the term `vps` is used in discussions of path names, environment variables, and file fields. This terminology is equivalent to MPS nomenclature. That is, mentions of “VPS” are applicable to “MPS.”

How to ... View the SRP Log Files

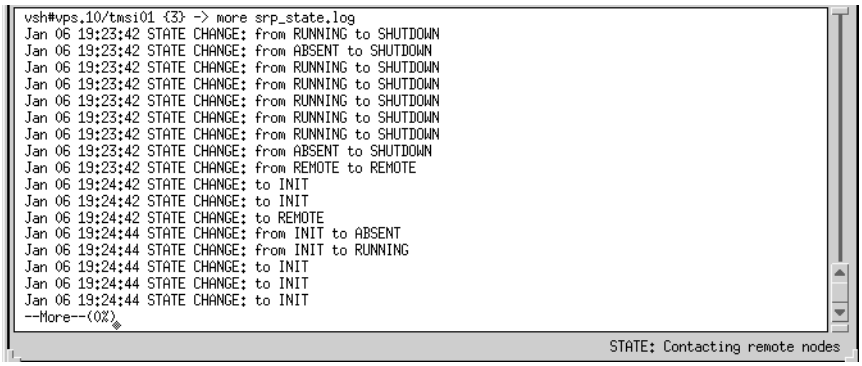
1. **Open a command shell on the component to be investigated**. For additional information, see see ***Command Control*** on page 50..
2. **Change to the directory holding the SRP log file.** Enter **`cd $MPSHOME/common/log`**.
3. **Enter the `more` or `page` command** followed by either **`srp.log`** or **`srp_state.log`** as applicable (an underscore must be used between the words **`srp`** and **`state`**). A portion of the file equal to the size of the command window will display, followed by a prompt indicating how much of the file has already been viewed.
4. **Press the space bar to scroll the screen**.
5. **Optionally, use the `tail` command.** By default, this command displays the last 10 lines of a file: thus, it becomes useful when only the most recent input to the `srp` files is of interest (the data in these files is sorted in ascending order).

See the Solaris system documentation for a detailed description of the **`more`**, **`page`**, and **`tail`** commands.

The following is an example of the SRP log file:



The following is an example of an SRP state log file:



For more general information about these two files, see the *PeriView Reference Manual*.

Printer and Modem Setup

This chapter covers:

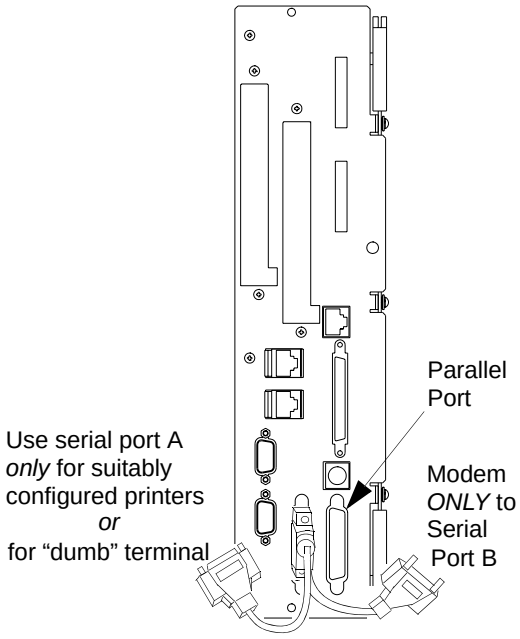
1. **Printer Setup**
2. **Modem Setup**
3. **The Administration Tool**
4. **Alternate Console**

Printer and Modem Setup

This appendix describes the hookup and default configuration of the printer, the modem, and an alternate console. Certified Avaya Field Engineers initially set up the Media Processing Server and peripherals. Refer to this appendix if there is a need to alter the initial setup.

Printer Setup

Attach the printer to the parallel port on the back of the SPARC station mounted in the MPS. Alternatively, a serial splitter cable is included for modem use. If the printer is capable of using a serial connection, it can be attached to serial port A of the cable. Otherwise, use the parallel port as usual (the end connectors of the splitter cable are labeled A and B). Access these ports through the removable back panel. Power the printer by plugging it in to the system power bar.



Rear view of application processor showing serial splitter cable (typical)

The MPS comes preconfigured to support a laser printer. Configure the printer, if necessary, by using the printer management settings in accordance with the instructions at see ***The Administration Tool*** on page 165..

Modem Setup

The Media Processing Server is equipped with an external modem to support remote access, software updates, and fault isolation. The modem comes preconfigured to 28.8 Kbps on TTYB. A serial splitter cable supports the modem connection. *This connection must be made to port B only!* For an illustrated example of this set-up.

Configure the modem, if necessary, by using the serial ports settings described under see ***The Administration Tool*** on page 165.. By default, the modem is set up with the following parameters:

- **Detail:** Expert
- **Template:** Modem-Bidirectional
- **Baud Rate:** 38400EH

`BAUD=38400` `PARITY=N` `WORDLEN=8`

`DIAL=PULSE ON HOOK` (for units with 57 S-registers (V6.0))

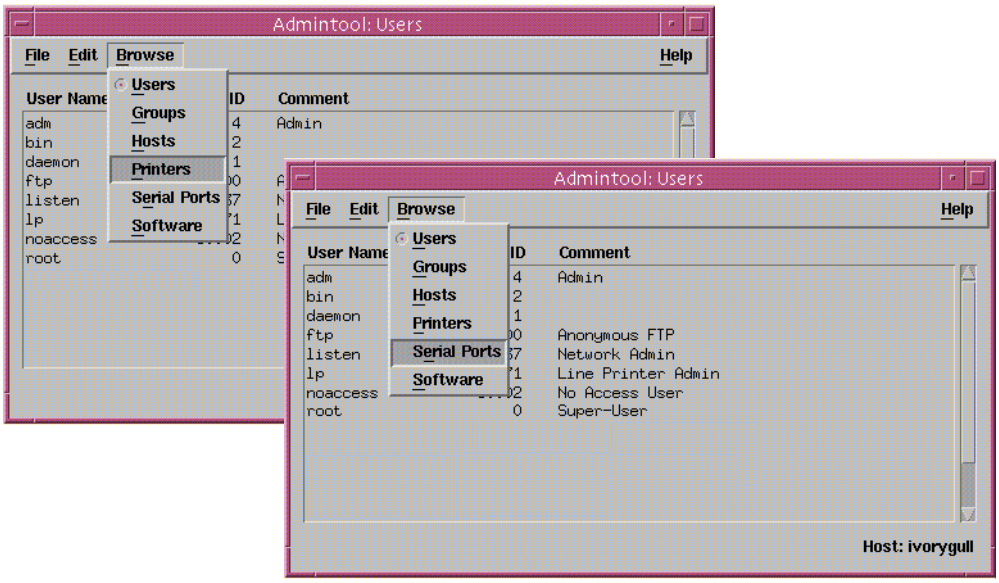
`DIAL=HUNT ON HOOK` (for units with 38 S-registers (V1.1))

The Administration Tool

Use the Solaris Administration Tool to configure printer and modem setups. To start the Administration Tool, enter **`admintool &`** at any command line for the node to which the item is connected.

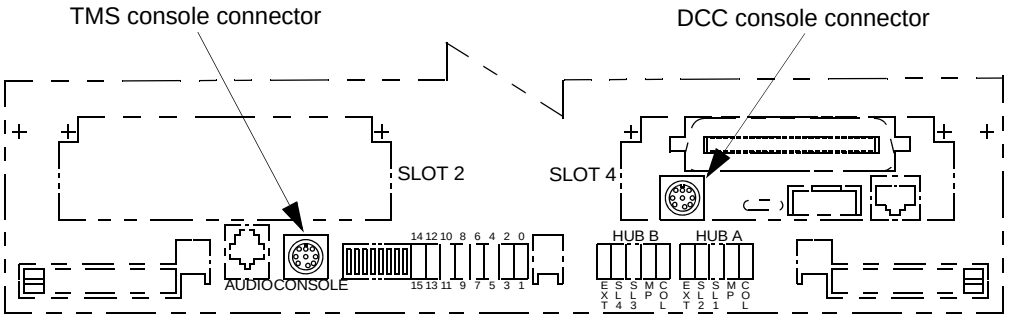
To make changes with these tools, super-user (`root`) level permissions are required.

Access the printer settings by following the Browse—Printers menu path to display the `Admintool:Printers` window. To access modem settings, follow the Browse—Serial Ports path to display the `Admintool:Serial Ports` window. Make any necessary changes in these subsets of screens.

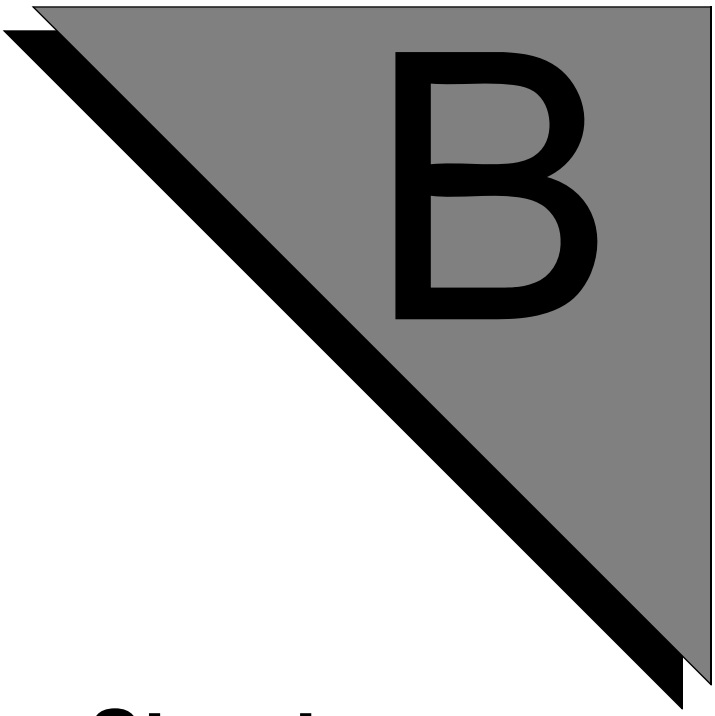


Alternate Console

If the system does not have a console, an alternate console ("dumb" terminal) can be used to monitor the Solaris system only, or to monitor the TMS or DCC. To monitor the Solaris portion of the system, simply attach a PC or ASCII terminal to TTYA with a NULL modem cable. To monitor the TMS or DCC, attach a serial port adapter cable (a cable with a `tip` connector on one end and a NULL modem connector on the other) or DB25 to 8-pin DIN cable with NULL modem adapter from the terminal to the console connector of the TMS or DCC (shown below).



A PC or laptop computer can also be connected using these cables. For details concerning the use, configuration, and types of monitoring that can be accomplished with these set-ups, see the *Media Processing Server Series System Reference Manual*.



MPS Directory Structure

This chapter covers:

1. **System software home directory $MPSHOME (/opt/vps)**
2. **User-Specified Distribution Directory (usually /opt/vps)**

MPS Directory Structure

All operating system software for the Avaya Media Processing Server is installed in or under the directory `/opt/vps`. This directory is referred to as the *system software home directory.* During the installation process, the `/opt/vps` directory is created (if it doesn't already exist) and the environment variable `$MPSHOME` is set to reference this directory.

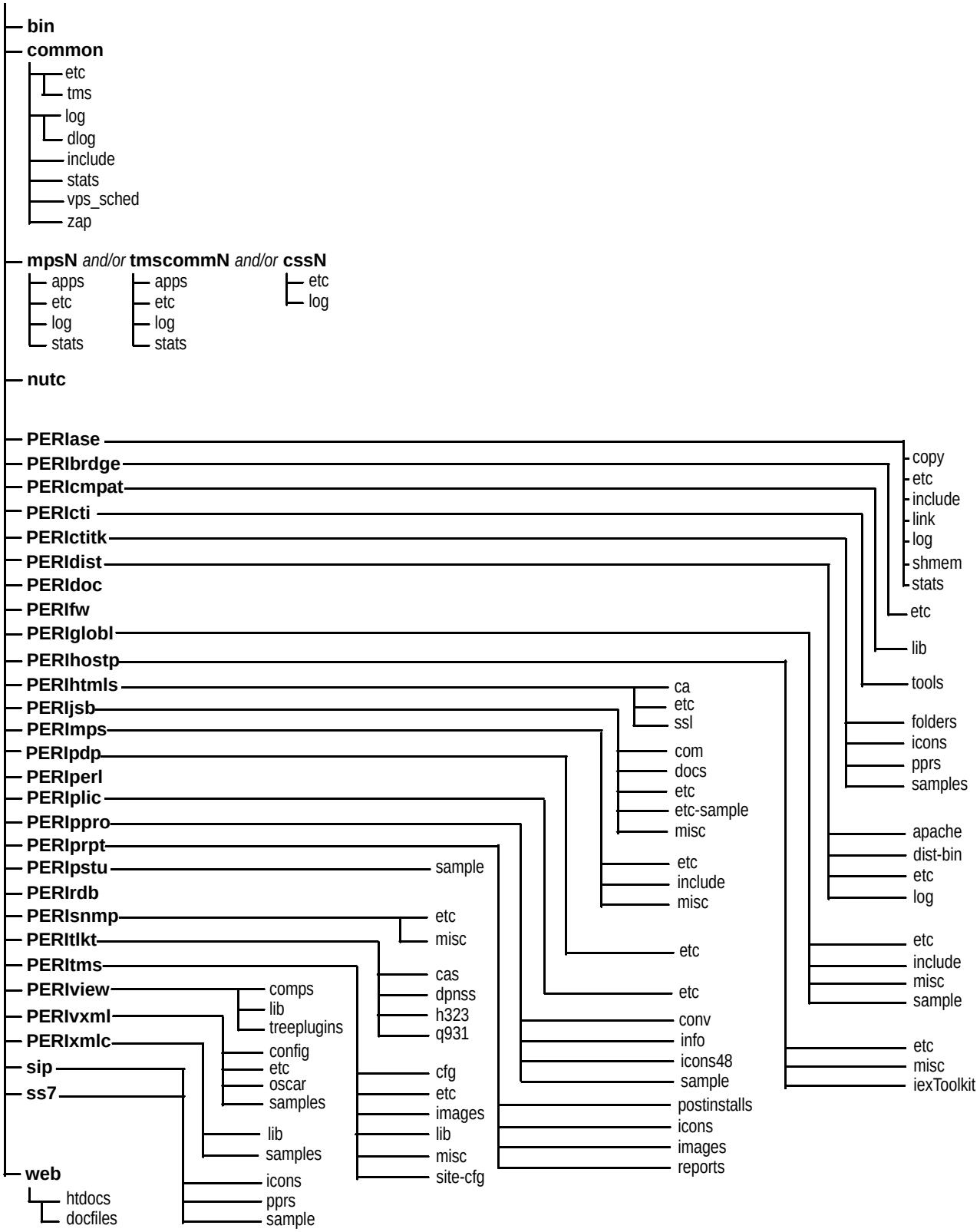
Each Avaya software package for major system functions (e.g., PERIstudio, PERIview, licensing files, etc.) is stored in its own subdirectory under the `packages` directory. During software installation, the option is presented to specify the location of the `packages` directory. The directory under which the `packages` directory is installed is referred to as the *user-specified distribution directory.* By default, the distribution directory is set to the same path as the system directory, `/opt/vps`.

When the Avaya software packages are installed, symbolic links to their locations are set up in `/opt/vps`. That is, if a directory other than `/opt/vps` is chosen to be the user-specified distribution directory, symbolic links in `/opt/vps` identify the location of the various packages.

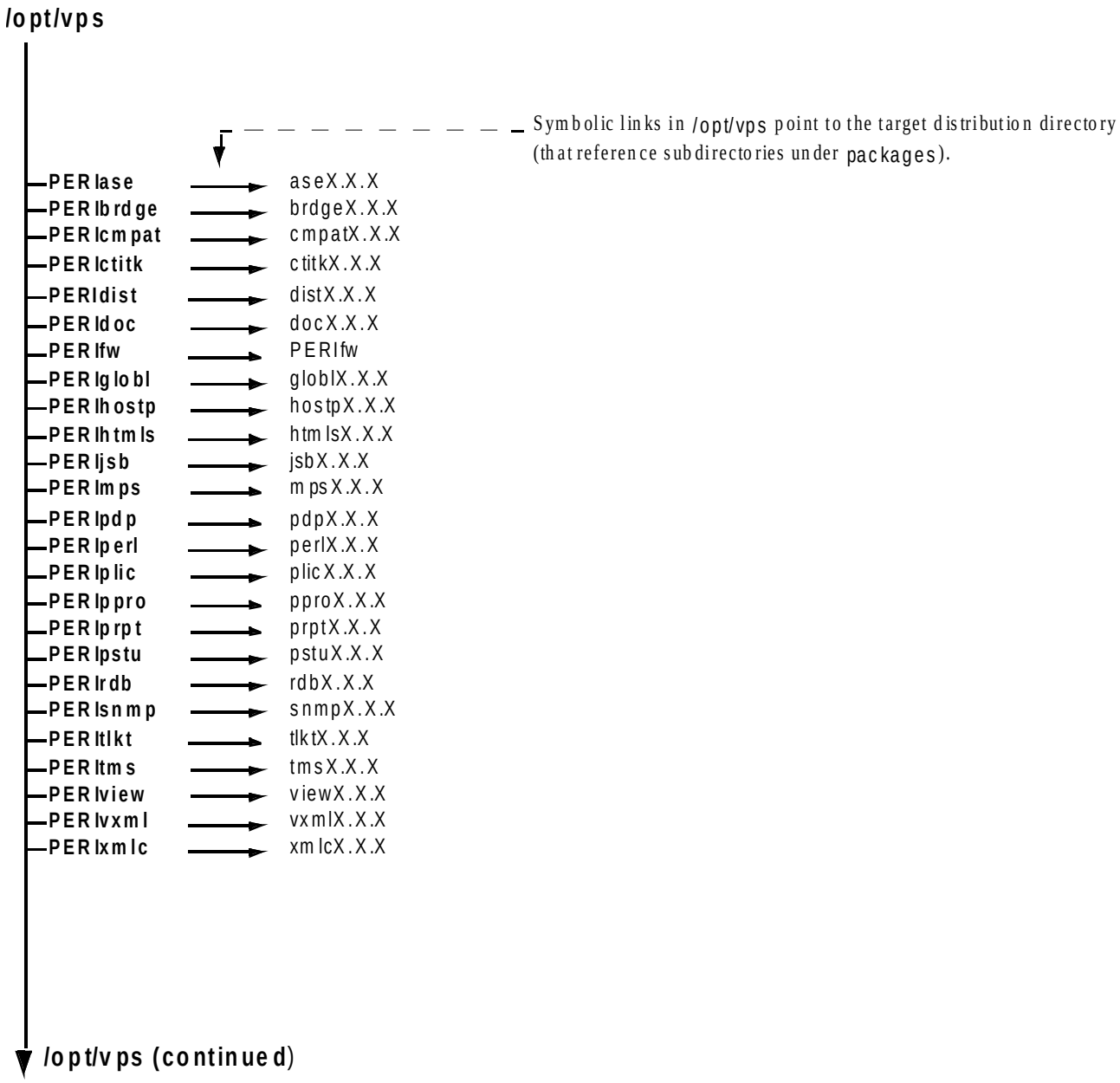
The following is important information about the naming conventions for installation directories:

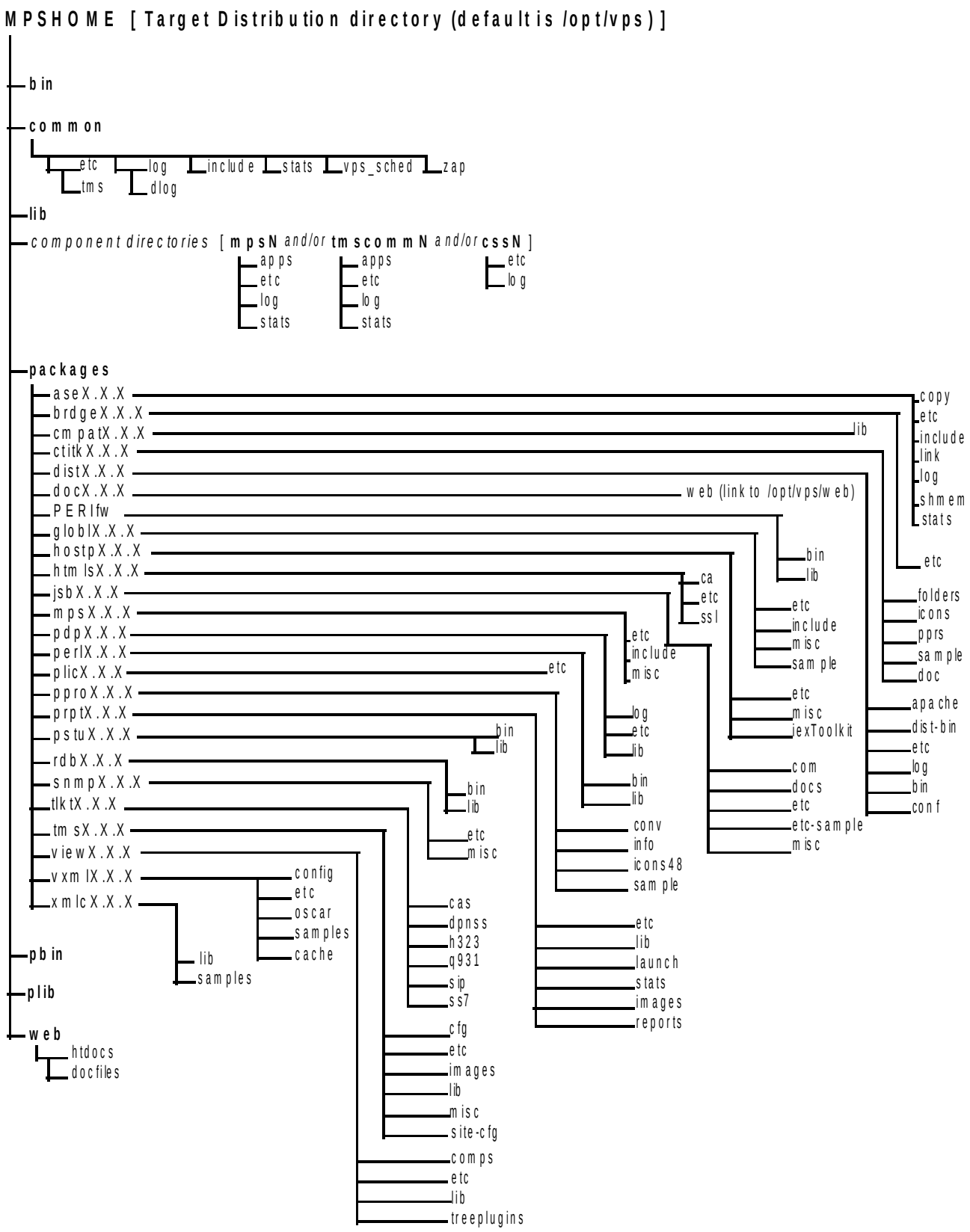
- The value of `$MPSHOME` should never be changed by any user or system administrator.
- It is highly recommended that the user-specified distribution directory remain set to `/opt/vps`, unless there is a specific reason to change it. This can make external troubleshooting easier, should it ever become necessary.
- The term `vps` is used frequently in the discussions of path and file names, and environment variables. This terminology is equivalent to Media Processing Server nomenclature. That is, `$MPSHOME` is the environment variable for referencing the home directory, `/opt/vps`, for system software on MPS systems.
- The command **`echo $<VARIABLE>`** displays the value of a specified environment variable. For example, to see the value of the system software home environment variable, enter **`echo $MPSHOME`** at a VSH command prompt.
- Individual software packages are stored in directory names of the form `<name>X.X.X` or `<name>X.XX`. This naming convention is used to identify particular packages and their version numbers.
- This appendix provides documentation only of the essential MPS directory structure. Depending on site requirements, there may be additional or fewer directories than are presented here. Also, the order in which the directories are shown might be different than on the site's installed MPS system(s).

%MPSHOME% (C:\Program Files\Avaya) - Windows Directory Structure



Solaris Directory Structure




MPSHOME [Target Distribution directory (default is /opt/vps)]
bin
common
etc
tms
log
dlog
include
stats
vps_sched
zap
lib
component directories [mpsN and/or tmscommN and/or cssN]
apps
etc
log
stats
apps
etc
log
stats
etc
log
packages
aseX.X.X
brdgeX.X.X
cmpatX.X.X
ctitkX.X.X
distX.X.X
docX.X.X
web (link to /opt/vps/web)
PERLfw
globlX.X.X
hostpX.X.X
htmlsX.X.X
jsbX.X.X
mpsX.X.X
pdpX.X.X
perlX.X.X
plicX.X.X
pproX.X.X
prptX.X.X
pstuX.X.X
rdbX.X.X
snmpX.X.X
tlktX.X.X
tmsX.X.X
viewX.X.X
vxmlX.X.X
xmlcX.X.X
config
etc
oscar
samples
cache
lib
samples
pbin
plib
web
htdocs
docfiles
copy
etc
include
link
log
shmem
stats
lib
etc
folders
icons
pprs
sample
doc
bin
lib
ca
etc
ssl
etc
include
misc
sample
etc
misc
jexToolkit
apache
dist-bin
etc
log
bin
conf
com
docs
etc
etc-sample
misc
etc
include
misc
etc
misc
log
etc
lib
bin
lib
bin
lib
bin
lib
etc
misc
conv
info
icons48
sample
cas
dpnss
h323
q931
sip
ss7
etc
lib
launch
stats
images
reports
cfg
etc
images
lib
misc
site-cfg
comps
etc
lib
treeplugins

All system-level software is installed in the directories identified below. The directory structure must not be altered, or the system might not be able to find necessary files, as needed.

System software home directory $MPSHOME (/opt/vps)

$MPSHOME (/opt/vps)	System directory containing subdirectories for the operating system files and symbolic links to the various packages installed on the system.	
bin	Contains the executable files for the installed software packages.	
common	Files available for use by all MPS systems.	
	etc	Configuration, administration, and alarm database files.
	log	SRP, file transfer, alarm, and ZAP log files.
	include	Header files for compiling user call functions.
	stats	Binary files for statistics collection and data files created by statistics-gathering scripts (including logs and status files).
	vps_sched	Scheduling files.
	zap	Temporary files used during ZAP execution.
fmIcons	Information for file manager icons.	
lib	Shared library files.	
mpsN	Files specific to a single MPS. The value of N indicates the MPS system number. One mpsN directory exists on each MPS system. The number zero (0) cannot be used.	
	apps	Working directory for application files. When an application is started, copies of its VENGINE executable (*.vex) and configuration (*.acfg) files are placed in this directory. SRP starts applications from this directory. This directory also contains subdirectories for shared user libraries.
	etc	Configuration and administration files.
	log	Log files.
	stats	Statistics files and shell scripts.
pbin	Diagnostics executables.	
plib	Diagnostic libraries.	
tmscommN	Files for bridging between and within MPS components. The value of N indicates the tmscomm component number. The number zero (0) cannot be used.	
	apps	Library subdirectory. Not used at the present time.
	etc	Contains the vos.cfg configuration file, which starts the Network Interface Controller Daemon (NCD) at system startup.
	log	Log files.
	stats	Statistics files and shell scripts.
PERI<name>	Symbolic links to the subdirectories under the packages directory.	

User-Specified Distribution Directory (usually /opt/vps)

The particular path used as the distribution directory is set up when the system-level software is installed. By default, it is set to `/opt/vps`. The distribution directory contains a lower-level subdirectory named `packages`. All Avaya software packages are stored in their own subdirectories under `packages`. When a particular package is installed, a symbolic link in `$MPSHOME` is set up by the system to reference that package.

The following is important information about the subdirectories and files in the distribution directory:

- The information provided here discusses only the essentials of the distribution subdirectory structure. Depending on site requirements, there may be additional or fewer directories than are shown here.

- Some of the distribution subdirectories contain a lower-level `log` subdirectory. Log files are used by the system internally and should never be modified with command line functions or text editors.

- If a particular software package has been upgraded since its initial installation, its subdirectory might contain a `patches` subdirectory and `patchlog` file. The `patches` subdirectory contains lower-level subdirectories and files necessary for implementing the patch(es), and the `patchlog` file lists the results of the installation of the patch(es). Patches and patch logs are not discussed further in this appendix.

- Some of the distribution subdirectories contain lower-level subdirectories with the names `pimages`, `pbin`, and `plib`. These files are used by Certified Avaya personnel for diagnostics and troubleshooting, and should not be modified by users or system administrators unless directed to do so by Avaya staff.

User-Specified Packages Directory (usually /opt/vps/packages)

packages	Home directory for all installed Avaya Media Processing Server software packages.	
aseX.X.X	Files for the Application Services Environment (ASE) software. ASE is the runtime environment for MPS Developer.	
	copy	Stubs for statistics folders (*src, VSUPD documents).
	etc	Sample configuration and administration files.
	include	Files for user CALL functions (*.h).
	link	Sample .vex and .ppr files.
	log	PeriReporter log files containing information about applications and collected system statistics. **These files should never be modified by the user.**
	shmem	Shared memory files (for file-based shared memory, if any).
	stats	Application statistics files.
brdgeX.X.X	Software for the tmscomm component. It supports bridging functions of the Media Processing Server system.	
	etc	Original version of the vos.cfg file. This is copied to tmscommN/etc during installation.
cmpatX.X.X	Shared libraries for PeriView operations that ensure compatibility between previous generation VPS/is software and current MPS systems. This package is required on all MPS systems.	
	lib	Shared library files.
distX.X.X	Directories and files for the Apache Web server.	
	apache	MPS configured Web server files.
	bin	Default executable files.
	conf	Default configuration files.
	dist-bin	Perl scripts.
	etc	Shell scripts.
	log	Web server log files.
docX.X.X	Directories and files that support online documentation for the MPS.	
	web	Location of documentation support files.
	htdocs	HTML intro pages to access the library of manuals or to start PeriView.
	docfiles	Location of manuals in pdf format.
fftX.X.X	Field factory test scripts that test the MPS and OSCAR components.	
	fftests	Directory of field factory test scripts.

User-Specified Packages Directory (usually /opt/vps/packages)

packages	Home directory for all installed Avaya Media Processing Server software packages.	
PERIfw	A system library that enables platform-independent process execution.	
	bin	Executable file for PeriFrameWork.
	lib	Libraries for PeriFrameWork.
globlX.X.X	Globally accessed directories and files, including libraries and binaries used by all other packages.	
	etc	Configuration files that are copied to $MPSHOME/common.
	include	Files for user CALL functions (*.h).
	misc	Alarm database files and SRP environment variable scripts.
	sample	Sample alarm files created through alarm API.
hostpX.X.X	Directories, files, and protocol software for communicating with host computers.	
	etc	Host protocol configuration files and character sets.
	misc	Reserved for future use.
	iexToolkit	IEX MPS Developer tool kit files.
htmlsX.X.X	Directories and files for the HTML Service daemon, which are used to retrieve files from the Web and serve them to applications.	
	ca	Contains security certificates.
	etc	HTMLS Daemon configuration files.
	ssl	Contains Secure Socket Layer (SSL) certificates and keys.
jsbX.X.X	Directories and files for the Java Services Bridge, which allows applications written in MPS Developer to access Java services.	
	com	Sub-directories containing Java class files.
	docs	Java Services Bridge manuals.
	etc	Java Services Bridge configuration files.
	etc-sample	Additional configuration files.
	misc	Sample applications utilizing the Java Services Bridge.
mpsX.X.X	Directories and files used by MPS processes and utilities.	
	etc	Configuration and administration files.
	images	Current software images not assigned to PeriView.
	misc	Startup and miscellaneous files required for MPS functions and administration.
	pimages	Diagnostic images.

User-Specified Packages Directory (usually /opt/vps/packages)

packages	Home directory for all installed Avaya Media Processing Server software packages.	
pdpX.X.X	Directories and files used by the PeriView Data Provider, which collects statistics and alarm data from nodes/components in a PeriView network.	
	etc	Configuration files for PERIpdp.
	lib	Libraries for PERIpdp.
perlX.X.X	Directories and files used to integrate the Perl programming language into the Avaya software suite.	
	bin	Perl runtime environment executable.
	lib	Libraries and scripts for PERIperl.
plicX.X.X	Directories containing files for licensing MPS software packages.	
	etc	Default location for the license file. Also contains the license package log file.
pproX.X.X	Tool kit and configuration files, and directories for MPS Developer processes and utilities.	
	conv	Files used to convert apps from different versions of MPS Developer.
	info	User interface object help text files.
	icons48	Application-specific icons.
	sample	Sample MPS Developer application files.
prptX.X.X	Directories and files used by PeriReporter, which is the statistics reporting and management interface.	
	etc	Configuration and administration files.
	images	Graphics files used in tools.
	lib	PeriReporter shared libraries.
	reports	Standard system report files.
	stats	Statistics-gathering scripts and log files.
pstuX.X.X	Directories and files used by PeriStudio processes and utilities. PeriStudio is the GUI tool used to create, manage, and edit speech vocabularies stored in MultiMedia Format (MMF) files.	
	bin	Solaris executable binaries and files.
	lib	PeriStudio shared libraries.
	bin	Solaris executable binaries and files.
	etc	Shell scripts used to set environment variables and paths.
	lib	Shared libraries used by this package.
	log	Web transaction logs.
	misc	Relay server configuration directories and files.
rdbX.X.X	Directories and files used by PeriRDB, which is the Relational Data Base manager.	

User-Specified Packages Directory (usually /opt/vps/packages)

packages	Home directory for all installed Avaya Media Processing Server software packages.	
	bin	Solaris executable binaries and files.
	lib	Libraries used for individual databases.
snmpX.X.X	Directories and files used by PeriSNMP processes and utilities. PeriSNMP allows the control and monitoring of MPS systems through a standard protocol, development of custom applications, and integration of MPS products into existing TCP/IP networks controlled by third-party management tools.	
	etc	Stores MIB and configuration files during installation.
	misc	Stores SNMP agent startup files during installation.
tlktX.X.X	Contains directories and files for Protocol Tool kits used by MPS Developer applications.	
	cas	Contains files for the CAS protocol tool kit and sample applications.
	dpnss	Contains files for the DPNSS protocol tool kit and sample applications.
	h323	Contains files for the h323 protocol tool kit and sample applications.
	q931	Contains files for the q931 protocol tool kit and sample applications.
	sip	Contains files for the SIP protocol tool kit and sample applications.
	ss7	Contains files for the SS7 protocol tool kit and sample applications.
tmsX.X.X	Directories and files used by the Telephony Media Server (TMS) processes and utilities.	
	bin	Solaris executable binaries and files, and image files.
	cfg	Protocol configuration and image definition files.
	etc	TMS data files.
	lib	TMS shared libraries.
	misc	Keyboard macro scripts and other files used by Certified Avaya personnel for testing purposes.
	site-cfg	Protocol configuration files and TMS system configuration files copied to the $MPSHOME/common/etc/tms directory.
viewX.X.X	Directories and files used by PeriView, for running the graphical tools used for system administration, operation, and control.	
	comps	Component JAR (Java ARchive) files.
	etc	Configuration files and subdirectory containing current software tools images.
	lib	PeriView shared libraries.
	treeplugins	Plugin files for PeriView tools.
xmlcX.X.X	Directories and files used by the Extensible Markup Language (XML) call function, which provides access to XML and HTML documents.	
	lib	Contains libraries for the PERIxmlc package.

User-Specified Packages Directory (usually /opt/vps/packages)

packages	Home directory for all installed Avaya Media Processing Server software packages.	
	samples	Contains sample applications.

This page has been intentionally left blank.

Index

D

E

F

G

H

W

X

Z