

# VMS VAXcluster Manual

Order Number: AA-LA27A-TE

**April 1988**

This manual describes the procedures for setting up and managing VAXcluster configurations.

**Revision/Update Information:** This manual supersedes the Version 4.0 *Guide to VAXclusters* and the Version 4.6 *VMS Local Area VAXcluster Manual*.

**Software Version:** VMS Version 5.0

**digital equipment corporation  
maynard, massachusetts**

---

**April 1988**

---

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

The software described in this document is furnished under a license and may be used or copied only in accordance with the terms of such license.

No responsibility is assumed for the use or reliability of software on equipment that is not supplied by Digital Equipment Corporation or its affiliated companies.

---

Copyright ©1988 by Digital Equipment Corporation

All Rights Reserved.  
Printed in U.S.A.

---

The postpaid READER'S COMMENTS form on the last page of this document requests the user's critical evaluation to assist in preparing future documentation.

The following are trademarks of Digital Equipment Corporation:

DEC	DIBOL	UNIBUS
DEC/CMS	EduSystem	VAX
DEC/MMS	IAS	VAXcluster
DECnet	MASSBUS	VMS
DECsystem-10	PDP	VT
DECSYSTEM-20	PDT	
DECUS	RSTS	
DECwriter	RSX	

**digital**™

ZK4477

---

**HOW TO ORDER ADDITIONAL DOCUMENTATION  
DIRECT MAIL ORDERS**

**USA & PUERTO RICO\***

Digital Equipment Corporation  
P.O. Box CS2008  
Nashua, New Hampshire  
03061

**CANADA**

Digital Equipment  
of Canada Ltd.  
100 Herzberg Road  
Kanata, Ontario K2K 2A6  
Attn: Direct Order Desk

**INTERNATIONAL**

Digital Equipment Corporation  
PSG Business Manager  
c/o Digital's local subsidiary  
or approved distributor

In Continental USA, Puerto Rico, Alaska, and Hawaii call 800-DIGITAL.

In Canada call 800-267-6215.

\* Any prepaid order from Puerto Rico must be placed with the local Digital subsidiary (809-754-7575).

Internal orders should be placed through the Software Distribution Center (SDC), Digital Equipment Corporation, Westminister, Massachusetts 01473.

---

---

## Production Note

This book was produced with the VAX DOCUMENT electronic publishing system, a software tool developed and sold by DIGITAL. In this system, writers use an ASCII text editor to create source files containing text and English-like code; this code labels the structural elements of the document, such as chapters, paragraphs, and tables. The VAX DOCUMENT software, which runs on the VMS operating system, interprets the code to format the text, generate a table of contents and index, and paginate the entire document. Writers can print the document on the terminal or line printer, or they can use DIGITAL-supported devices, such as the LN03 laser printer and PostScript<sup>®</sup> printers (PrintServer 40 or LN03R ScriptPrinter), to produce a typeset-quality copy containing integrated graphics.



---

# Contents

<b>PREFACE</b>	<b>xiii</b>
<b>NEW AND CHANGED FEATURES</b>	<b>xv</b>
<hr/>	
<b>CHAPTER 1 INTRODUCTION TO THE VAXCLUSTER ENVIRONMENT</b>	<b>1-1</b>
<b>1.1 CLUSTER HARDWARE</b>	<b>1-2</b>
<b>1.2 CLUSTER SOFTWARE</b>	<b>1-2</b>
<b>1.3 CLUSTER CONFIGURATION TYPES</b>	<b>1-4</b>
<b>1.3.1 CI-Only VAXcluster Configurations</b>	<b>1-4</b>
<b>1.3.2 Local Area VAXcluster Configurations</b>	<b>1-5</b>
<b>1.3.3 Mixed-Interconnect VAXcluster Configurations</b>	<b>1-7</b>
<b>1.3.4 Cluster Security for Local Area and Mixed-Interconnect Configurations</b>	<b>1-8</b>
<b>1.4 DECNET-VAX COMMUNICATIONS</b>	<b>1-9</b>
<b>1.5 CLUSTER CONNECTION MANAGEMENT</b>	<b>1-9</b>
<b>1.5.1 The Quorum Scheme</b>	<b>1-10</b>
<b>1.5.2 Quorum Disk</b>	<b>1-11</b>
<b>1.6 SHARED PROCESSING AND PRINTER RESOURCES</b>	<b>1-12</b>
<b>1.7 SHARED DISK RESOURCES</b>	<b>1-12</b>
<hr/>	
<b>CHAPTER 2 PREPARING THE CLUSTER OPERATING ENVIRONMENT</b>	<b>2-1</b>
<b>2.1 DIRECTORY STRUCTURE ON A COMMON SYSTEM DISK</b>	<b>2-2</b>

## Contents

<b>2.2</b>	<b>INSTALLING THE VMS OPERATING SYSTEM IN THE VAXCLUSTER ENVIRONMENT</b>	<b>2-4</b>
<b>2.3</b>	<b>CONFIGURING THE DECNET-VAX NETWORK</b>	<b>2-6</b>
<b>2.3.1</b>	<b>Copying Remote Node Databases</b>	<b>2-8</b>
<b>2.3.2</b>	<b>Enabling Cluster Alias Operations</b>	<b>2-8</b>
<b>2.4</b>	<b>COORDINATING CLUSTER COMMAND PROCEDURES</b>	<b>2-9</b>
<b>2.4.1</b>	<b>Building Common Command Procedures</b>	<b>2-10</b>
<b>2.4.2</b>	<b>Using Node-Specific System Command Procedures</b>	<b>2-11</b>
<b>2.5</b>	<b>COORDINATING SYSTEM FILES TO DEFINE THE CLUSTER USER ENVIRONMENT</b>	<b>2-11</b>
<b>2.5.1</b>	<b>Coordinating User Accounts</b>	<b>2-12</b>
<b>2.5.2</b>	<b>Preparing the MAIL Database</b>	<b>2-13</b>
<b>2.5.3</b>	<b>Preparing the Rights Database</b>	<b>2-14</b>
<b>2.5.4</b>	<b>Coordinating Shared System Files in Clusters with Multiple Common System Disks</b>	<b>2-14</b>
<b>CHAPTER 3 BUILDING AND MAINTAINING THE CLUSTER</b>		<b>3-1</b>
<b>3.1</b>	<b>PLANNING CONFIGURATION PROCEDURES</b>	<b>3-1</b>
<b>3.1.1</b>	<b>CLUSTER_CONFIG.COM Functions</b>	<b>3-2</b>
<b>3.1.2</b>	<b>Determining Locations and Sizes for Satellite Page and Swap Files</b>	<b>3-3</b>
<b>3.1.3</b>	<b>Selecting Boot Servers for Mixed-Interconnect Clusters</b>	<b>3-3</b>
<b>3.1.4</b>	<b>Specifying Allocation Class Values in Mixed-Interconnect Clusters</b>	<b>3-4</b>
<b>3.2</b>	<b>CONFIGURING THE CLUSTER</b>	<b>3-5</b>
<b>3.2.1</b>	<b>Adding a Node to the Cluster</b>	<b>3-6</b>
<b>3.2.1.1</b>	Updating Network Data after Adding a Satellite • <b>3-11</b>	
<b>3.2.1.2</b>	Restoring a Satellite's Network Data • <b>3-12</b>	
<b>3.2.1.3</b>	Controlling Clusterwide Broadcast Messages on Satellites and Boot Servers • <b>3-12</b>	
<b>3.2.2</b>	<b>Removing a Node from the Cluster</b>	<b>3-13</b>
<b>3.2.3</b>	<b>Changing a Node's Characteristics</b>	<b>3-14</b>
<b>3.2.4</b>	<b>Changing the Cluster Configuration Type</b>	<b>3-19</b>
<b>3.2.4.1</b>	Changing an Existing CI-Only Cluster to a Mixed-Interconnect Configuration • <b>3-19</b>	
<b>3.2.4.2</b>	Changing an Existing Local Area Cluster to a Mixed-Interconnect Configuration • <b>3-20</b>	
<b>3.2.5</b>	<b>Converting a Standalone Node to a Cluster Node</b>	<b>3-21</b>

3.2.6	Creating a Duplicate System Disk _____	3-21
<hr/>		
3.3	<b>RECONFIGURING THE CLUSTER AFTER A MAJOR CHANGE</b>	3-23
3.3.1	Updating MODPARAMS.DAT Files to Adjust Cluster Quorum _____	3-23
3.3.2	Shutting Down the Cluster _____	3-24
3.3.3	Changing Allocation Class Values on HSCs _____	3-24
3.3.4	Rebooting the Cluster _____	3-24
<hr/>		
3.4	<b>MAINTAINING THE CLUSTER</b>	3-24
3.4.1	Running AUTOGEN with the FEEDBACK Option _____	3-25
3.4.2	Recording Configuration Data _____	3-25
3.4.3	Monitoring Ethernet Activity in Local Area and Mixed-Interconnect Clusters _____	3-26
3.4.4	Restoring Cluster Quorum after an Unexpected Node Failure _____	3-26
3.4.5	Selecting Cluster Shutdown Options _____	3-28
3.4.5.1	The REMOVE_NODE Option • 3-28	
3.4.5.2	The CLUSTER_SHUTDOWN Option • 3-28	
3.4.5.3	The REBOOT_CHECK Option • 3-29	
3.4.5.4	The SAVE_FEEDBACK Option • 3-29	
3.4.6	Performing Security Functions in Local Area and Mixed-Interconnect Clusters _____	3-29
3.4.6.1	Maintaining Cluster Security Data • 3-30	
3.4.6.2	Controlling Conversational Bootstrap Operations for Satellites • 3-31	
<hr/>		
<b>CHAPTER 4 SETTING UP AND MANAGING CLUSTER QUEUES</b>		<b>4-1</b>
<hr/>		
4.1	<b>CLUSTERWIDE QUEUES</b>	4-1
<hr/>		
4.2	<b>CLUSTER PRINTER QUEUES</b>	4-1
4.2.1	Setting Up Printer Queues _____	4-2
4.2.2	Setting Up Clusterwide Generic Printer Queues _____	4-3
<hr/>		
4.3	<b>CLUSTER BATCH QUEUES</b>	4-6
4.3.1	Setting Up Executor Batch Queues _____	4-7
4.3.2	Setting Up Generic Batch Queues _____	4-7

# Contents

<b>4.4</b>	<b>COMMAND PROCEDURES FOR ESTABLISHING QUEUES</b>	<b>4-9</b>
<b>4.4.1</b>	<b>Starting Queues Using Node-Specific Command Procedures</b>	<b>4-9</b>
<b>4.4.2</b>	<b>Starting Queues Using a Common Command Procedure</b>	<b>4-12</b>
<b>4.5</b>	<b>SUMMARY OF COMMANDS FOR SETTING UP CLUSTER QUEUES</b>	<b>4-14</b>
<hr/>		
<b>CHAPTER 5</b>	<b>SETTING UP AND MANAGING CLUSTER DISKS</b>	<b>5-1</b>
<b>5.1</b>	<b>CLUSTER-ACCESSIBLE DISKS</b>	<b>5-1</b>
<b>5.1.1</b>	<b>HSC Disks</b>	<b>5-2</b>
<b>5.1.2</b>	<b>MSCP-Served Disks</b>	<b>5-2</b>
<b>5.1.3</b>	<b>Dual-Pathed Disks</b>	<b>5-3</b>
5.1.3.1	Dual-Ported HSC Disks • 5-3	
5.1.3.2	Dual-Ported DSA Disks • 5-4	
5.1.3.3	Dual-Ported MASSBUS Disks • 5-4	
<b>5.2</b>	<b>CLUSTER DEVICE-NAMING CONVENTIONS</b>	<b>5-5</b>
<b>5.2.1</b>	<b>Rules for Specifying Allocation Class Values</b>	<b>5-5</b>
<b>5.2.2</b>	<b>Sample Configurations with Named Devices</b>	<b>5-6</b>
<b>5.3</b>	<b>SHARED DISKS</b>	<b>5-9</b>
<b>5.4</b>	<b>SETTING UP CLUSTER DEVICES</b>	<b>5-10</b>
<b>5.5</b>	<b>VOLUME SHADOWING IN MIXED-INTERCONNECT CLUSTERS</b>	<b>5-10</b>
<b>5.5.1</b>	<b>Mounting Shadow Sets</b>	<b>5-11</b>
<b>5.5.2</b>	<b>Dismounting Shadow Sets</b>	<b>5-11</b>
<b>5.5.3</b>	<b>Using Shadow Sets as Satellite System Disks</b>	<b>5-12</b>
<hr/>		
<b>APPENDIX A</b>	<b>CLUSTER SYSGEN PARAMETERS</b>	<b>A-1</b>

---

<b>APPENDIX B</b>	<b>BUILDING A COMMON SYSUAF.DAT FILE FROM NODE-SPECIFIC FILES</b>	<b>B-1</b>
-------------------	---	------------

---

<b>APPENDIX C</b>	<b>VAXCLUSTER TROUBLESHOOTING INFORMATION</b>	<b>C-1</b>
-------------------	---	------------

---

<b>C.1</b>	<b>DIAGNOSING FAILURES OF NODES TO BOOT OR TO JOIN THE CLUSTER</b>	<b>C-1</b>
<b>C.1.1</b>	<b>Summary of Events for Nodes Booting and Joining the Cluster</b>	<b>C-1</b>
<b>C.1.2</b>	<b>CI-Connected Node Fails to Boot</b>	<b>C-3</b>
<b>C.1.3</b>	<b>Satellite Node Fails to Boot</b>	<b>C-4</b>
<b>C.1.4</b>	<b>Node Fails to Join the Cluster</b>	<b>C-6</b>
<b>C.1.5</b>	<b>Startup Procedures Fail to Complete</b>	<b>C-7</b>

---

<b>C.2</b>	<b>DIAGNOSING CLUSTER HANGS</b>	<b>C-7</b>
<b>C.2.1</b>	<b>Cluster Quorum Is Lost</b>	<b>C-7</b>
<b>C.2.2</b>	<b>A Shared Cluster Resource Is Inaccessible</b>	<b>C-8</b>

---

<b>C.3</b>	<b>DIAGNOSING CLUEXIT BUGCHECKS</b>	<b>C-8</b>
------------	-------------------------------------	------------

---

<b>C.4</b>	<b>DIAGNOSING VAXPORT DEVICE PROBLEMS</b>	<b>C-9</b>
<b>C.4.1</b>	<b>VAXport Communication Mechanisms</b>	<b>C-9</b>
<b>C.4.2</b>	<b>Port Failures</b>	<b>C-10</b>
<b>C.4.2.1</b>	Verifying CI Port Functions • <b>C-11</b>	
<b>C.4.2.2</b>	Verifying CI Cable Connections • <b>C-12</b>	
<b>C.4.2.3</b>	Repairing CI Cables • <b>C-15</b>	
<b>C.4.3</b>	<b>Analyzing Error Log Entries for VAXport Devices</b>	<b>C-16</b>
<b>C.4.3.1</b>	Error Log Entry Formats • <b>C-16</b>	
<b>C.4.3.2</b>	Device-Attention Entries • <b>C-16</b>	
<b>C.4.3.3</b>	Logged-Message Entries • <b>C-19</b>	
<b>C.4.3.4</b>	Error Log Entry Descriptions • <b>C-21</b>	
<b>C.4.4</b>	<b>OPAO Error Messages</b>	<b>C-29</b>

---

**INDEX**

## Contents

---

### EXAMPLES

2-1	Sample Interactive Network Configuration Session _____	2-7
3-1	Sample Interactive CLUSTER_CONFIG.COM Session to Add a CI-Connected Node as a Boot Server _____	3-7
3-2	Sample Interactive CLUSTER_CONFIG.COM Session to Add a Satellite Node with Local Page and Swap Files ____	3-9
3-3	Sample NETNODE_UPDATE.COM File _____	3-12
3-4	Sample Interactive CLUSTER_CONFIG.COM Session to Remove a Satellite Node with Local Page and Swap Files _____	3-13
3-5	Sample Interactive CLUSTER_CONFIG.COM Session to Enable the Local System as a Disk Server _____	3-16
3-6	Sample Interactive CLUSTER_CONFIG.COM Session to Change the Local System's ALLOCLASS Value _____	3-17
3-7	Sample Interactive CLUSTER_CONFIG.COM Session to Enable the Local System as a Boot Server _____	3-17
3-8	Sample Interactive CLUSTER_CONFIG.COM Session to Change a Satellite's Hardware Address _____	3-18
3-9	Sample Interactive CLUSTER_CONFIG.COM Session to Convert a Standalone Node to a Cluster Boot Server ____	3-21
3-10	Sample Interactive CLUSTER_CONFIG.COM CREATE Session _____	3-22
3-11	Sample Interactive SYSMAN CONFIGURATION Session .	3-31
4-1	STARTQ Command Procedure for Node JUPITER _____	4-9
4-2	STARTQ Command Procedure for Node SATURN _____	4-10
4-3	STARTQ Command Procedure for Node URANUS _____	4-11
4-4	Starting Queues Using a Common Command Procedure .	4-13
5-1	Shadow Set as Seen from Boot Server _____	5-10
5-2	Shadow Set as Seen from Satellite _____	5-11
C-1	CI Device-Attention Entry _____	C-17
C-2	Ethernet Device-Attention Entry _____	C-18
C-3	CI Logged-Message Entry _____	C-20

---

### FIGURES

1-1	Typical CI-Only VAXcluster Configuration _____	1-5
1-2	Typical Local Area VAXcluster Configuration _____	1-7
1-3	Typical Mixed-Interconnect VAXcluster Configuration ____	1-8
2-1	Directory Structure on Common System Disk _____	2-2
2-2	File Search Order on Common System Disk _____	2-3

4-1	Sample Printer Configuration _____	4-2
4-2	Printer Queue Configuration _____	4-3
4-3	Cluster Printer Queue Configuration With Clusterwide Generic Printer Queue _____	4-4
4-4	Printer Queue Configuration With Local Generic Queue _____	4-5
4-5	Sample Batch Queue Configuration _____	4-6
4-6	Batch Queue Configuration With Clusterwide Generic Queue _____	4-8
5-1	CI-Only Configuration With Shared Disks _____	5-2
5-2	Configuration with a Dual-Pathed HSC Disk _____	5-7
5-3	Configuration with a Dual-Pathed DSA Disk _____	5-7
5-4	Device Names in a Mixed-Interconnect Cluster _____	5-8
C-1	A Correctly Connected Two-Node CI Cluster _____	C-13
C-2	Crossed CI Cable Pair _____	C-13

---

**TABLES**

1-1	VAXcluster Hardware Components _____	1-2
2-1	Information Requested for CI-Only Configurations _____	2-5
2-2	Information Requested for Local Area and Mixed-Interconnect Configurations _____	2-5
3-1	Data Requested by CLUSTER_CONFIG.COM _____	3-5
3-2	CLUSTER_CONFIG.COM CHANGE Options _____	3-14
3-3	Summary of SYSMAN CONFIGURATION Commands for Cluster Authorization _____	3-30
5-1	Specifying Values for MSCP_LOAD and MSCP_SERVE_ALL Parameters _____	5-3
A-1	Cluster SYSGEN Parameters _____	A-1



---

# Preface

---

## Intended Audience

This document addresses persons responsible for setting up and managing VAXcluster configurations. To use the document as a guide to cluster management, you must have a thorough understanding of VMS system management concepts and procedures, as described in the *Introduction to VMS System Management*, the *Guide to Setting Up a VMS System*, and the *Guide to Maintaining a VMS System*.

---

## Document Structure

The *VMS VAXcluster Manual* contains five chapters and three appendixes.

Chapter 1 describes the VAXcluster environment.

Chapter 2 explains how to prepare the cluster operating environment before building a cluster.

Chapter 3 explains how to build a cluster once the necessary preparations are made, and how to reconfigure and maintain the cluster.

Chapter 4 discusses cluster queue management concepts and procedures.

Chapter 5 discusses cluster disk management concepts and procedures.

Appendix A lists and defines cluster SYSGEN parameters.

Appendix B provides guidelines for building a cluster common user authorization file.

Appendix C provides VAXcluster troubleshooting information.

---

## Associated Documents

This document is not a one-volume reference manual. The VMS utilities and commands discussed are described in detail in separate VMS Utility Reference Manuals and in the *VMS DCL Dictionary*.

For additional information on the topics covered in this manual, refer to the following documents:

- *Introduction to VMS System Management*
- *Guide to Setting Up a VMS System*
- *Guide to Maintaining a VMS System*
- *Guide to VMS File Applications*
- *VMS Networking Manual*
- *VAX Volume Shadowing Manual*
- *VMS Utility Reference Manuals*

# Preface

---

## Conventions

Convention	Meaning
<code>RET</code>	In examples, a key name (usually abbreviated) shown within a box indicates that you press a key on the keyboard; in text, a key name is not enclosed in a box. In this example, the key is the RETURN key. (Note that the RETURN key is not usually shown in syntax statements or in all examples; however, assume that you must press the RETURN key after entering a command or responding to a prompt.)
<code>CTRL/C</code>	A key combination, shown in uppercase with a slash separating two key names, indicates that you hold down the first key while you press the second key. For example, the key combination CTRL/C indicates that you hold down the key labeled CTRL while you press the key labeled C. In examples, a key combination is enclosed in a box.
<code>\$ SHOW TIME</code> <code>05-JUN-1988 11:55:22</code>	In examples, system output (what the system displays) is shown in black. User input (what you enter) is shown in red.
<code>\$ TYPE MYFILE.DAT</code> <code>.</code> <code>.</code> <code>.</code>	In examples, a vertical series of periods, or ellipsis, means either that not all the data that the system would display in response to a command is shown or that not all the data a user would enter is shown.
<code>input-file, . . .</code>	In examples, a horizontal ellipsis indicates that additional parameters, values, or other information can be entered, that preceding items can be repeated one or more times, or that optional arguments in a statement have been omitted.
<code>[logical-name]</code>	Brackets indicate that the enclosed item is optional. (Brackets are not, however, optional in the syntax of a directory name in a file specification or in the syntax of a substring specification in an assignment statement.)
quotation marks apostrophes	The term quotation marks is used to refer to double quotation marks ("). The term apostrophe (') is used to refer to a single quotation mark.

---

---

## New and Changed Features

New VAXcluster software features for VMS Version 5.0 include the following:

- Support for MicroVAX class processors as VAXcluster members in *mixed-interconnect* cluster configurations. These systems can boot into a mixed-interconnect cluster over the Ethernet.
- Support for an increased number of cluster nodes.
- Enhanced Mass Storage Protocol (MSCP) Server functions. New server functions enable a disk-serving system to serve all suitable disks to the cluster early in the boot sequence, so that the disks become cluster accessible with minimal interruption whenever the serving system reboots. In addition, the server automatically serves any suitable disks that are added to the system later.
- Failover support for DSA disks using UDA/KDA/BDA controllers.
- A revised quorum disk scheme.
- A new command procedure, `SY$MANAGER:CLUSTER_CONFIG.COM`, which you execute to perform cluster configuration functions. This procedure replaces the following VMS Version 4.0 and 4.6 procedures:
  - `MAKEROOT.COM`
  - `BOOT_CONFIG.COM`
  - `SATELLITE_CONFIG.COM`

Note that the configuration information presented in this document is subject to change. For definitive information on supported VAXcluster configurations, refer to the current VAXcluster Software Product Description (SPD) document.



# 1

---

## Introduction to the VAXcluster Environment

A VAXcluster environment is a highly integrated organization of VAX or MicroVAX systems or a combination of these systems. As members of a cluster, the systems can share processing resources, queues, and disk storage under a single VMS security and management domain, and they can boot or fail independently.

Using procedures described in Chapter 2, system managers can tailor the cluster operating environment to create a *common-environment* or a *multiple-environment* cluster.

- In a common-environment cluster, the same resources are available on all nodes. User accounts are identical, the same known images are installed, the same logical names are defined, and mass storage devices and queues are shared.
- In a multiple-environment cluster, a group of nodes may share one set of resources, while another group shares a different set. Or an individual node may perform a specialized function using restricted resources, while other nodes are used for general time-sharing work.

Although most cluster resources may be shared, user processes and system memory are node specific. When a process is created on a cluster node, the process must complete on that node, using memory local to the node. If the node should fail before the process completes, the process is terminated. However, users can recover from such a failure more quickly than on a standalone system, because they need not wait until the system is rebooted. Typically, they can log in on another cluster node to create a new process and continue working—provided that the resources required by the process (such as images and global sections) are available on that node.

This chapter describes the key components and distinctive features of the VAXcluster environment. Topics include the following:

- Cluster hardware and software components
- Cluster configuration types
- DECnet-VAX communications
- Cluster connection management
- Shared cluster resources

Be sure you understand these topics before you attempt to perform the cluster setup operations described in Chapters 2 and 3.

# Introduction to the VAXcluster Environment

## 1.1 Cluster Hardware

---

### 1.1 Cluster Hardware

Basic VAXcluster hardware components are described in Table 1-1.

**Table 1-1 VAXcluster Hardware Components**

Component	Function
VAX processor	A VAX or MicroVAX class processor running the VMS operating system. Any VAX processor in the cluster is considered an <i>active node</i> .
Computer Interconnect (CI)	The CI is a high-speed, dual-path bus that connects VAX processor nodes and intelligent I/O subsystems (HSCs) in a computer room environment.
CI Port Controller	<p>A microcoded, intelligent controller that connects VAX processors to the CI. Each interface connects to the CI bus, which consists of two transmitter and two receiver cables.</p> <p>Under normal operating conditions, both sets of cables are available to meet traffic demands. If one path becomes inoperative, then all traffic uses the remaining path. The VMS operating system periodically tests a failed path. As soon as a failed path becomes available, it will automatically be used for normal traffic.</p>
Star Coupler	<p>The Star Coupler is the common connection point for all nodes connected to a CI. As with the CI bus, the Star Coupler is dual pathed and contains separate components for each path.</p> <p>The star coupler connects all CI cables from the individual nodes, creating a radial or "star" arrangement that has a maximum radius of 45 meters. It supports the physical connection or disconnection of nodes during normal cluster operations, without affecting the rest of the cluster.</p>
Hierarchical Storage Controller (HSC)	The HSC is a self-contained, intelligent, mass storage subsystem that enables cluster nodes to share DIGITAL Standard Architecture (DSA) disks. Because the HSC is an intelligent controller, it optimizes physical disk operations. The HSC is considered a <i>passive node</i> .
Ethernet	The Ethernet is a bus that uses digital baseband signaling. The Ethernet is used both for DECnet-VAX transmissions, and, in some cluster configurations, for interprocessor System Communication Services (SCS). In the VAXcluster environment, the Ethernet and its circuit devices must be configured according to requirements specified in the VAXcluster Software Product Description (SPD) document.

---

### 1.2 Cluster Software

The software components used to implement VAXcluster functions are as follows:

- System Communication Services (SCS)
- VAXport drivers
- Connection Manager
- Distributed File System and VMS Record Management Services (RMS)

# Introduction to the VAXcluster Environment

## 1.2 Cluster Software

- Distributed Lock Manager
- Distributed Job Controller
- Mass Storage Control Protocol (MSCP) Server and disk class driver(s)

These components are always present on each cluster member, so that if one member fails, the cluster continues to function, because all the remaining members possess the necessary software components.

The *System Communication Services* (SCS) software implements internode communication, according to DIGITAL's System Communication Architecture (SCA).

The *VAXport drivers* (for example, PADRIVER and PEDRIVER) control the communication paths between local and remote ports.

The *Connection Manager* dynamically defines and coordinates the cluster. The Connection Manager uses the system communication services and provides an acknowledged message delivery service for higher VMS software layers. The Connection Manager also maintains cluster integrity when nodes join or leave the cluster—that is, when cluster *state transitions* occur.

The *Distributed File System* allows all processors to share disk mass storage, whether the disk is connected to an HSC or to a processor. A local disk may be made available to the entire cluster. All cluster-accessible disks appear as if they are local to every processor.

The distributed file system and VMS Record Management Services (VMS RMS) provide the same access to disks and files clusterwide that is provided on a standalone system. VMS RMS files may be shared clusterwide to the record level.

The *Distributed Lock Manager* is used for synchronization functions by the distributed file system, job controller, device allocation, and other cluster facilities. It is available to users to develop cluster applications. The Distributed Lock Manager implements the \$ENQ and \$DEQ system services to provide clusterwide synchronization of access to resources by allowing the locking and unlocking of resource names. (For detailed information on system services, refer to the *VMS System Services Volume*.) It also provides a queueing mechanism so that processes can be put into a wait state until a particular resource is available. As a result, cooperating processes can synchronize their access to shared objects such as files or records.

If a processor in the cluster fails, all locks it holds are released. This mechanism allows processing to continue on the remaining processors. The Distributed Lock Manager also supports clusterwide deadlock detection.

The *Distributed Job Controller* makes queues available clusterwide. A cluster operates with a common set of batch and print queues. Users can submit jobs to any queue within the cluster, provided that the necessary mass storage volumes and peripheral devices are accessible to the system on which the job executes. System managers can also set up generic batch queues that distribute batch processing workloads among nodes.

The *Mass Storage Control Protocol (MSCP) Server* implements the MSCP protocol, which is used to communicate with a controller for local MASSBUS or UNIBUS disks, or for Digital Standard Architecture (DSA) disks, such as RA series disks. In conjunction with one or both of the disk class drivers (DUDRIVER, DSDRIVER), the MSCP Server implements this protocol on a processor, allowing the processor to function as a storage controller. The

# Introduction to the VAXcluster Environment

## 1.2 Cluster Software

processor submits I/O requests to locally accessed disks, such as UNIBUS, MASSBUS, and Unibus Disk Adapter (UDA) disks, and accepts the I/O requests from any node in the cluster. In this way, the MSCP Server makes locally connected disks available to all nodes in the cluster. The MSCP Server can also make HSC disks accessible over the Ethernet.

---

## 1.3 Cluster Configuration Types

While site-specific processing needs and available hardware resources must determine how you configure your cluster, you always start with one of the following configuration types:

- CI-only VAXcluster configuration
- Local Area VAXcluster configuration
- Mixed-interconnect VAXcluster configuration

These configuration types are distinguished by the interconnect devices (CI, Ethernet, or both) used for SCS interprocessor communications.

Sections 1.3.1 through 1.3.3 describe each type of configuration. For complete information on currently supported configurations, including the type and number of nodes supported in each configuration type, and configuration requirements, refer to the VAXcluster Software Product Description (SPD) document.

Depending on the type of configuration you plan to set up, one or more processor nodes may be required to perform specific functions. For example, in all local area and mixed-interconnect configurations, at least one node must perform both *boot serving* and *disk serving* functions. These functions are described in Section 1.3.2.

Once you have determined which type of configuration best meets your needs, you can set up your cluster using the procedures described in Chapters 2 and 3.

---

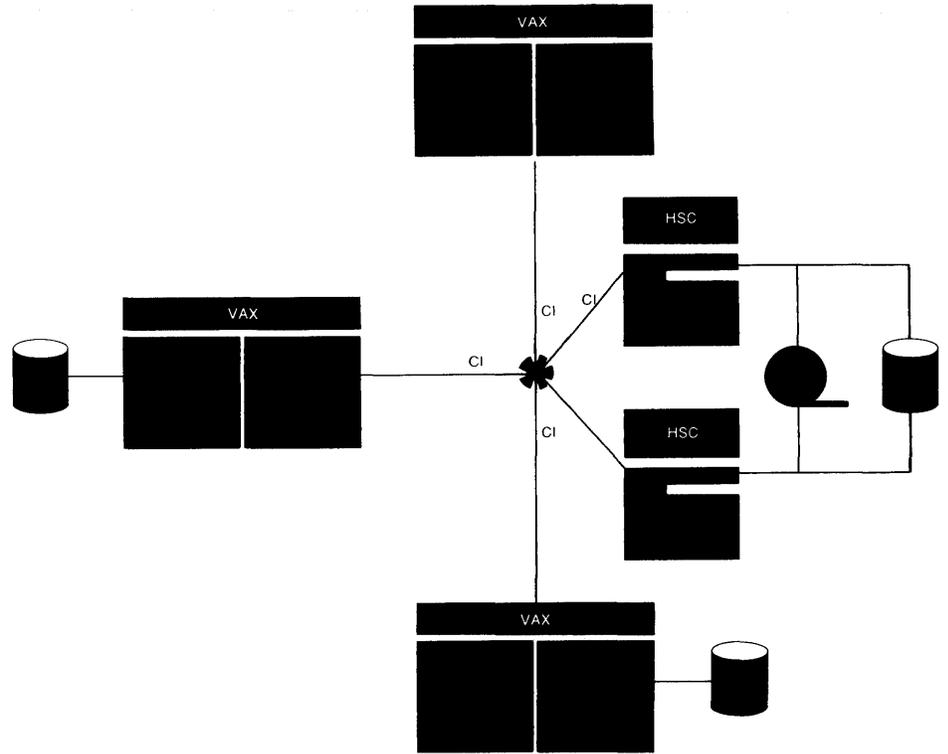
### 1.3.1 CI-Only VAXcluster Configurations

A CI-only cluster uses the CI for interprocessor communication, with the Star Coupler as the common connection point for all cluster nodes (VAX processors and HSCs). Cluster nodes may be any VAX processors specified in the VAXcluster SPD, or they may be HSCs. Figure 1-1 shows how the components are typically configured. Note that any CI-only cluster may later be converted to a mixed-interconnect configuration. Refer to Section 3.2.4 for instructions.

# Introduction to the VAXcluster Environment

## 1.3 Cluster Configuration Types

Figure 1-1 Typical CI-Only VAXcluster Configuration



ZK-1640-84

### 1.3.2 Local Area VAXcluster Configurations

In a local area cluster, interprocessor communication is carried out over the Ethernet by a VAXport driver that emulates certain CI port functions. A cluster node may be any VAX or MicroVAX processor specified in the VAXcluster SPD document. Because HSCs require CI connections, local area clusters do not include HSCs.

A single Ethernet may support multiple local area clusters, each identified and secured by a unique *group number* and a *cluster password*. (For information on cluster security, see Section 1.3.4.)

A local area cluster includes *boot servers* (*boot nodes*) and *satellite nodes*.

A *boot server* is both a management center for the cluster and a major resource provider. Its system disk contains the cluster common files for startup, authorization, and queue setup, as well as the directory roots from which the satellite nodes are booted. (The system manager creates these directory roots—one for each satellite—using the CLUSTER\_CONFIG.COM command procedure, described in Chapter 3.)

A boot server makes available to the cluster such resources as user and application data disks, printers, and distributed batch processing facilities.

# Introduction to the VAXcluster Environment

## 1.3 Cluster Configuration Types

Using DECnet Maintenance Operation Protocol (MOP), a boot server responds to downline load requests from satellites. When a satellite requests an operating system load, the boot server responds to the request and sends an image to the satellite that allows the satellite to load the VMS operating system and join the cluster.

Note that because a boot server *must* serve its system disk to the cluster (and usually its data disks as well), a boot server is, by definition, always a *disk server*. The MSCP Server is therefore always loaded on a boot server, so that the node can serve its disks to the cluster.

Boot servers should be the most powerful machines in the cluster. They should also use the highest bandwidth Ethernet adapters available.

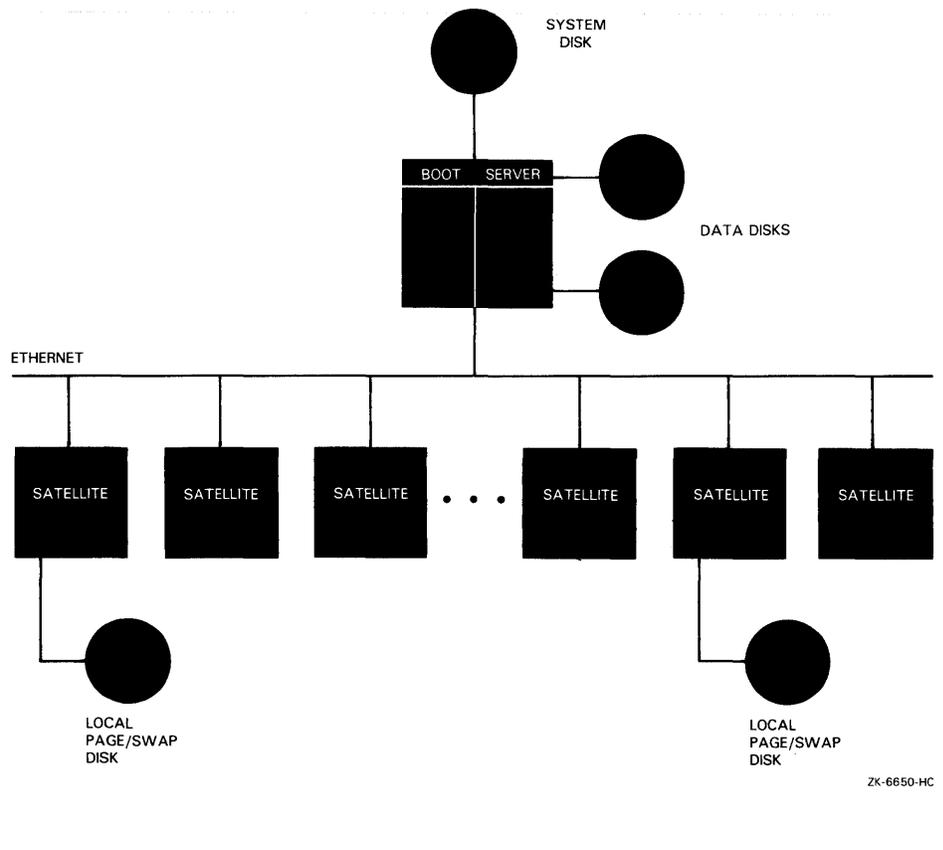
The *satellite nodes* are booted remotely from a boot server's system disk. Generally, these nodes are consumers of cluster resources, though they may also sometimes provide disk serving and batch processing resources. If satellite nodes are equipped with RD series disks, they may, for enhanced performance, use such local disks for paging and swapping.

Figure 1-2 shows a typical local area cluster configuration. Note that any local area cluster may later be converted to a mixed-interconnect configuration. Refer to Section 3.2.4 for instructions.

# Introduction to the VAXcluster Environment

## 1.3 Cluster Configuration Types

Figure 1–2 Typical Local Area VAXcluster Configuration



### 1.3.3 Mixed-Interconnect VAXcluster Configurations

Clusters with both CI and Ethernet interconnects are available for the first time with VMS Version 5.0. A mixed-interconnect cluster may include VAX processors, HSCs, and MicroVAX satellites. Because the MSCP Server and disk class drivers allow VAX processors to serve HSC disks to the cluster, satellites can access the large amounts of storage available through HSC controllers.

Mixed-interconnect clusters combine the advantages of both CI-only and local area cluster configurations:

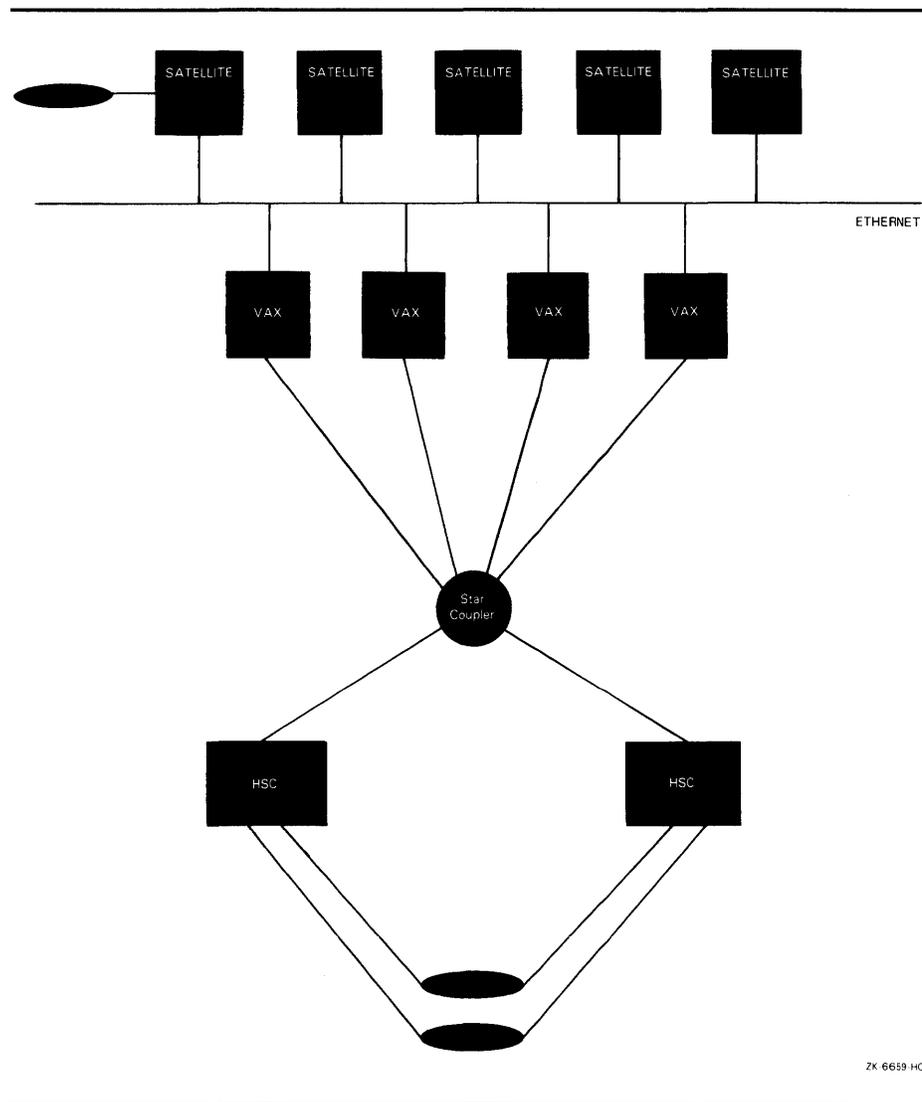
- Use of HSCs for mass storage
- Support for MicroVAX class processors as cluster members
- High availability of system resources
- Centralized cluster management

# Introduction to the VAXcluster Environment

## 1.3 Cluster Configuration Types

Figure 1-3 shows a typical mixed-interconnect configuration.

**Figure 1-3 Typical Mixed-Interconnect VAXcluster Configuration**



### 1.3.4 Cluster Security for Local Area and Mixed-Interconnect Configurations

Local area and mixed-interconnect clusters use a *group number* and a *cluster password* to allow multiple independent clusters to coexist on the same Ethernet and to prevent access to a cluster by unauthorized nodes.

- The *group number* uniquely identifies each mixed-interconnect and local area cluster on a single Ethernet. This number must be in the range from 1 to 4095 or from 61440 to 65535. Note that if you plan to have more than one of these clusters at your site, you must coordinate the assignment of group numbers among cluster system managers.

# Introduction to the VAXcluster Environment

## 1.3 Cluster Configuration Types

- The *cluster password* serves as an additional check to ensure the integrity of individual clusters on the same Ethernet that accidentally use identical group numbers. (Provided that each cluster's password is unique, the clusters will form independently.) The password also prevents an intruder who discovers the group number from joining the cluster. The password must be from 1 to 31 alphanumeric characters in length and may include dollar signs and underscores.

Security data is maintained in the cluster authorization file, SYS\$COMMON:[SYSEXE]CLUSTER\_AUTHORIZE.DAT. This file is created during installation of the VMS operating system, if you indicate that you want to set up a local area or mixed-interconnect cluster. The installation procedure then prompts you for the cluster group number and password. Cluster security functions are described in detail in Chapter 3. (If you convert a CI-only cluster to a mixed-interconnect configuration, the file is created when you execute the CLUSTER\_CONFIG.COM command procedure described in Chapter 3.)

---

## 1.4 DECnet–VAX Communications

In any cluster configuration, DECnet–VAX communications are required for all processor nodes. Use of DECnet–VAX facilities ensures that system managers can access each node in the cluster from a single terminal, even if terminal-switching facilities are not available.

In local area and mixed-interconnect clusters, DECnet is required both for system management functions and interprocessor communication. For example, DECnet is used for remote booting operations (downline loading of satellite nodes).

In these configurations, DECnet and System Communication Services coexist on the same Ethernet. They share the same data link and physical link protocols, which are implemented by the Ethernet data link drivers, the Ethernet adapters, and the Ethernet itself.

---

## 1.5 Cluster Connection Management

Cluster integrity is controlled by a software component called the *Connection Manager*, which determines and coordinates cluster membership. The Connection Manager creates a cluster when the first active nodes are booted, and then reconfigures the cluster when nodes join or leave it.

Cluster members can share various data and system resources, such as disk volumes. To achieve the coordination necessary to maintain resource integrity, the cluster nodes must share a clear sense of cluster membership. This sense of cluster membership is maintained by the Connection Manager.

The integrity of shared resources, however, cannot be guaranteed unless their use is carefully coordinated in the cluster. In the unlikely event that a pair of nodes that are not members of the same cluster share some resource, *cluster partitioning* occurs. Partitioning is undesirable, because resource sharing between two clusters is not coordinated, and the integrity of the shared resource cannot be ensured. To prevent partitioning, the Connection Manager uses a scheme called *quorum*.

# Introduction to the VAXcluster Environment

## 1.5 Cluster Connection Management

---

### 1.5.1 The Quorum Scheme

The quorum scheme is based on the arithmetic principle that the whole cannot be divided into multiple parts in such a way that more than one part is greater than half of the whole.

The quorum scheme functions as follows:

- Each node in the cluster contributes a fixed number of votes towards quorum. The votes value is specified by the SYSGEN parameter VOTES. On satellites, the value is always set to zero by default.
- Each active node in the cluster (including satellites) indirectly specifies an initial quorum value using the SYSGEN parameter EXPECTED\_VOTES. This parameter is the sum of all VOTES held by potential cluster members. It is used to derive an estimate of the correct quorum value for the cluster, according to the following formula:

$$\text{estimated quorum} = (\text{EXPECTED\_VOTES} + 2) / 2$$

- During certain cluster state transitions, the system dynamically computes the cluster quorum to be the *maximum* of the following:
  - The current cluster quorum value
  - The largest of the values calculated from the following formula, where EV is the EXPECTED\_VOTES value specified by each node:

$$(EV+2)/2$$

- The value calculated from the following formula, where V is the total of VOTES held by all cluster members:

$$(V+2)/2$$

The cluster state transitions that cause cluster quorum to be recalculated occur when a node joins the cluster and when the cluster recognizes a quorum disk (see Section 1.5.2).

- If the current number of votes ever drops below the quorum (because of nodes leaving the cluster), the cluster members suspend all process activity and all I/O operations to cluster-accessible disks until sufficient votes are added (nodes joining the cluster) to bring the total number of votes to a value greater than or equal to quorum.
- As the cluster changes, the system only raises the cluster quorum value; it never lowers the value. (However, system managers can lower the value; for details, see Section 3.4.4.)

For example, consider a cluster consisting of three nodes, each node having its VOTES parameter set to 1 and its EXPECTED\_VOTES parameter set to 3. The Connection Manager dynamically computes the cluster quorum value to be 2. In this example, any two of the three nodes constitute a quorum and may run in the absence of the third node. No single node can constitute a quorum by itself. Therefore, there is no way the three cluster nodes can be partitioned and run as two independent clusters.

# Introduction to the VAXcluster Environment

## 1.5 Cluster Connection Management

---

### 1.5.2 Quorum Disk

A quorum disk acts as a virtual node, adding to the cluster votes total. By establishing a quorum disk in configurations with a small number of voting members, you can increase the availability of the cluster.

Such configurations can tolerate the failure either of the quorum disk or of a processor node.

To use a quorum disk, one or more nodes must have a direct (non-MSCP-served) connection to the disk. Such nodes are known as *quorum disk watchers*. Nodes that cannot access the disk directly rely on the quorum disk watchers for information about the status of votes contributed by the quorum disk.

You should enable as quorum disk watchers any nodes that have an active direct connection to the quorum disk, or that have the potential for a direct connection. To enable a node as a quorum disk watcher, you use the CLUSTER\_CONFIG.COM CHANGE function described in Section 3.2.3. The procedure prompts for the name of the quorum disk and specifies that name as a value for the SYSGEN parameter DISK\_QUORUM in MODPARAMS.DAT. The procedure also sets an appropriate value for the QDKSVOTES parameter. The number of votes contributed by the quorum disk is equal to the smallest value of the SYSGEN parameter QDKSVOTES on any quorum disk watcher.

**Note:** You can also enable the first installed cluster node as a quorum disk watcher by answering YES when the VMS installation procedure asks if the cluster will contain a quorum disk.

For the quorum disk's votes to be counted in the cluster votes total, the following conditions must be met:

- On one or more nodes capable of becoming watchers, you must specify the same device name as a value for DISK\_QUORUM. The remaining nodes (nodes with a blank value for DISK\_QUORUM) recognize the name specified by the first watcher node with which they communicate.
- At least one watcher node must have a direct, active connection to the quorum disk. Thus, the quorum disk may be a dual-ported DSA disk, which has an active direct connection to only one node at a time.
- The disk must contain a valid format file named QUORUM.DAT in the master file directory (MFD). The QUORUM.DAT file is created automatically after a system specifying a quorum disk has booted into the cluster. This file will be used on subsequent reboots. If no quorum disk is enabled when a node boots, the file will not be created on that node.
- To permit recovery from failure conditions, the quorum disk must be mounted by all disk watchers.

# Introduction to the VAXcluster Environment

## 1.6 Shared Processing and Printer Resources

---

### 1.6 Shared Processing and Printer Resources

In any cluster configuration, nodes can share processing and printer resources. The ability to share resources allows for better workload balancing, because batch and print job processing can be distributed across the cluster.

System managers control how jobs share batch processing and printer resources by setting up and maintaining clusterwide generic queues. The strategy used to set up and manage these queues will determine how well workloads are matched to available resources. Managers establish and maintain the queues with the same commands used to manage queues on a single-node system.

All clusterwide queues are controlled by a single, cluster common job controller queue file (JBCSYSQUE.DAT), which must be accessible to the nodes participating in the clusterwide queue scheme. This file makes queues available across the cluster and enables jobs to execute on any queue from any node—provided that the necessary mass storage volumes can be accessed by the node on which the job executes.

Procedures for setting up and managing cluster queues are described in Chapter 4.

---

### 1.7 Shared Disk Resources

A major advantage of cluster configurations is the ability to make disk resources accessible to all cluster nodes. A *cluster-accessible* disk can be used by any active node in the cluster that successfully mounts it. A disk that is not cluster accessible can be accessed only by the local node.

Cluster-accessible disks offer the following advantages:

- More efficient use of mass storage, because more than one node can use the same disk.
- Access by users to their default work disks when logging in to any node on which the disks are accessible.
- Clusterwide file sharing. Because nodes can share common versions of files, updates to a file are made only once to a single copy of the file.
- Implementation of clusterwide job controller queues. Batch and print jobs can be processed on any node that has access to the disks.

Procedures for setting up and managing cluster disks are described in Chapter 5.

# 2

---

## Preparing the Cluster Operating Environment

You must prepare the cluster operating environment on the first installed node before configuring other nodes in the cluster. You may prepare either a *common-environment* or a *multiple-environment* cluster. The operating environment you choose depends mainly on the processing needs of your site.

In a *common-environment cluster*, the operating environment is identical on each member node, because the nodes are run from the same system files. The nodes are set up with identical user accounts, the same known images are installed, the same logical names are defined, and mass storage devices and queues are shared. In effect, users in a common-environment cluster can log in to any node and work in the same operating environment.

In a *multiple-environment cluster*, the environment varies from node to node, and users can work in environments that are specific to the node they are logged in to. A multiple-environment cluster is effective when you want to share data among member nodes, but when you want certain nodes to serve specialized needs. For example, you might want to set up a three-node cluster, in which the time-sharing environments on two nodes are the same, while the third node is set up exclusively for batch processing of large inventory jobs. In this case, the time-sharing nodes are set up with a common environment, sharing users, queues, and access to mass storage devices, while the third node runs in its own restricted environment.

This chapter concentrates on the steps necessary to prepare a common-environment cluster. Approaches for preparing a multiple-environment cluster are also described, but are presented as general guidelines.

Topics include the following:

- Directory structure on a common system disk
- Installing the VMS operating system in the VAXcluster environment
- Configuring the DECnet-VAX network
- Coordinating cluster command procedures
- Coordinating system files to define the cluster user environment

Once you have prepared the cluster operating environment on the first cluster node, you can build the cluster using the procedures described in Chapter 3.

# Preparing the Cluster Operating Environment

## 2.1 Directory Structure on a Common System Disk

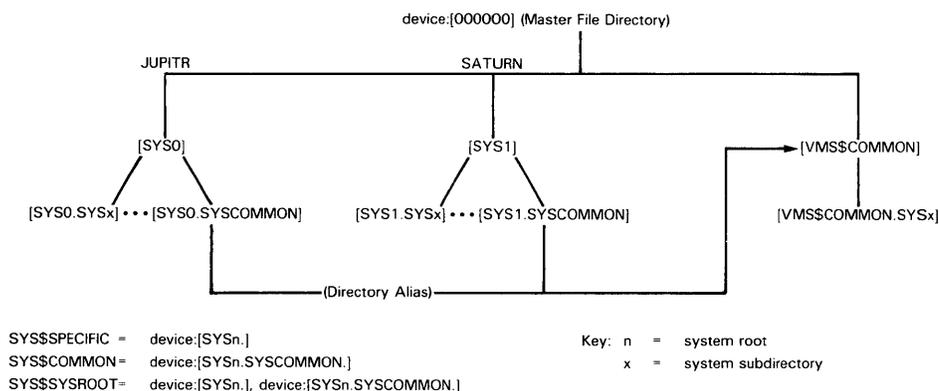
### 2.1 Directory Structure on a Common System Disk

The VMS installation or upgrade procedure generates a *common system disk*, on which most operating system and optional product files are stored in a common root directory. The entire directory structure—that is, the common root plus each node's local root—is stored on the same disk. After the installation or upgrade completes, you use the CLUSTER\_CONFIG.COM command procedure described in Chapter 3 to create a local root for each new cluster node and boot it into the cluster.

Each local root contains, in addition to the usual system directories, a [SYSx.SYSCOMMON] directory that is an alias for [VMS\$COMMON], the cluster common root directory in which cluster common files actually reside. When you add a node to the cluster, CLUSTER\_CONFIG.COM sets up the alias.

Figure 2-1 illustrates the directory structure set up for nodes JUPITR and SATURN, which are run from a common system disk. The disk's master file directory (MFD) contains the local roots (SYS0 for JUPITR, SYS1 for SATURN) and the cluster common root directory, [VMS\$COMMON].

**Figure 2-1 Directory Structure on Common System Disk**

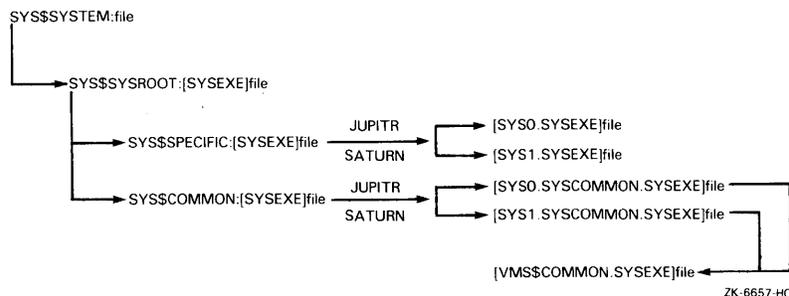


The logical name SYSS\$SYSROOT is defined as a search list that points to a local root first (SYSS\$SPECIFIC) and then to the common root (SYSS\$COMMON). Thus, the logical names for the system directories (SYSS\$SYSTEM, SYSS\$LIBRARY, SYSS\$MANAGER, and so forth) point to two directories: a local root (for example, SYSS\$SPECIFIC:[SYSEXE]) and a common root (for example, SYSS\$COMMON:[SYSEXE]). Figure 2-2 shows how directories on a common system disk are searched when the logical name SYSS\$SYSTEM is used in file specifications.

# Preparing the Cluster Operating Environment

## 2.1 Directory Structure on a Common System Disk

**Figure 2–2 File Search Order on Common System Disk**



It is important to keep this search order in mind when manipulating system files on a common system disk. Node-specific files must always reside and be updated in the appropriate node's system subdirectory. For example, MODPARAMS.DAT must reside in SYS\$SPECIFIC:[SYSEXE], which is [SYS0.SYSEXE] on JUPITR, and [SYS1.SYSEXE] on SATURN. Thus, to create a new MODPARAMS.DAT for JUPITR when logged in on JUPITR, you would enter the following command:

```
$ EDIT SYS$SPECIFIC:[SYSEXE]MODPARAMS.DAT
```

Once the file is created, you could use the following command to modify it:

```
$ EDIT SYS$SYSTEM:MODPARAMS.DAT
```

However, to modify JUPITR's MODPARAMS.DAT when logged in on any other cluster node that boots from the same common system disk, you must enter the following command:

```
$ EDIT [SYS0.SYSEXE]MODPARAMS.DAT
```

If you want to modify records in the cluster common system authorization file in a cluster with a single cluster common system disk, you could enter the following commands on any cluster node:

```
$ SET DEFAULT SYS$COMMON:[SYSEXE]
$ RUN AUTHORIZE
```

But if, for example, you have set up a node-specific system authorization file (SYSUAF.DAT) for node JUPITR and you want to modify records in that file when logged in on another cluster node that boots from the same cluster common system disk, you must, before invoking AUTHORIZE, set your default directory to JUPITR's node-specific [SYSEXE] directory. For example:

```
$ SET DEFAULT [SYS0.SYSEXE]
$ RUN AUTHORIZE
```

# Preparing the Cluster Operating Environment

## 2.2 Installing the VMS Operating System in the VAXcluster Environment

### 2.2 Installing the VMS Operating System in the VAXcluster Environment

You must perform the installation or upgrade once for each system disk in the cluster. Because, however, several nodes normally run from the same cluster common system disk, you need not perform the installation or upgrade on each cluster node.

You may want to set up a cluster that has a combination of one or more common system disks and one or more individual system disks. Again, you must do the installation or upgrade once for each system disk. For example, if your cluster consists of ten nodes, four of which share one common system disk, four of which share a second common system disk, and each of the other two has its own system disk, you would do the installation or upgrade four times. Note that if your cluster includes multiple common system disks, *you must later coordinate system files to define the cluster operating environment, as described in Section 2.5.4.*

To perform the installation, follow instructions in the installation and operations guide for your processor. However, before you start the installation, be sure you have determined which *cluster configuration type* you want to create (CI-only, local area, or mixed-interconnect), because the installation procedure will request configuration-specific information. (Configuration types are described in Section 1.3.)

Table 2-1 lists the information requested for CI-only configurations; Table 2-2 lists the information requested for local area and mixed-interconnect configurations. Typical responses are explained in the tables. Note that initial questions are the same for all configuration types.

If your system disk is on an HSC, you must obtain the HSC's *disk allocation class* value before starting the installation, because the installation procedure will request that information. (Allocation classes are discussed in detail in Section 5.2.) To obtain the value, enter a command sequence like the following at the HSC console. The information displayed will include the allocation class value.

```
CTRL/C
HSC> SHOW SYS
15-Apr-1988 14:31:43.41  Boot:   13-Apr-1988 11:31:11.41  Up:   51:00
.
.
.
DISK allocation class = 1          TAPE allocation class = 0
Start command file m Disabled
SETSHO - Program Exit
```

If you later want to change the allocation class value, follow the instructions in Section 3.3.

**Note:** While rebooting at the end of the installation procedure, the system will display messages warning that you must install required licenses. Be sure to install these licenses, as well as the DECnet-VAX license, as soon as the system is available. Procedures for installing the licenses are described in the release notes distributed with the software kit.

# Preparing the Cluster Operating Environment

## 2.2 Installing the VMS Operating System in the VAXcluster Environment

**Table 2–1 Information Requested for CI-Only Configurations**

Item	Response
Will this node be a cluster member (Y/N)?	Enter Y.
What is the node's DECnet node name?	Enter DECnet node name—for example, JUPITR. The DECnet node name may be from 1 to 6 alphanumeric characters in length and may not include dollar signs or underscores.
What is the node's DECnet node address?	Enter DECnet node address—for example, 2.2
Will the Ethernet be used for cluster communications (Y/N)?	Enter N. The Ethernet is not used for cluster (SCS internode) communications in CI-only configurations.
Will JUPITR be a disk server (Y/N)?	Enter Y or N, depending on your configuration requirements. Refer to Section 1.3.3 and Chapter 5 for information on served cluster disks.
Enter a value for JUPITR's ALLOCLASS parameter:	If the system is connected to a dual-ported disk, enter a value from 1-255 that will be used on both sides. Otherwise, enter 0.
Does this cluster contain a quorum disk [N]?	Enter Y or N, depending on your configuration. If you enter Y, the procedure prompts for the name of the quorum disk. Enter the device name of the quorum disk.

**Table 2–2 Information Requested for Local Area and Mixed-Interconnect Configurations**

Item	Response
Will this node be a cluster member (Y/N)?	Enter Y.
What is the node's DECnet node name?	Enter DECnet node name—for example, JUPITR. The DECnet node name may be from 1 to 6 alphanumeric characters in length and may not include dollar signs or underscores.
What is the node's DECnet node address?	Enter DECnet node address—for example, 2.2
Will the Ethernet be used for cluster communications (Y/N)?	Enter Y. The Ethernet is required for cluster (SCS internode) communications in local area and mixed-interconnect configurations.
Enter this cluster's group number:	Enter a number in the range from 1-4095 or 61440-65535.
Enter this cluster's password:	Enter the cluster password. The password must be from 1 to 31 alphanumeric characters in length and may include dollar signs and underscores.

# Preparing the Cluster Operating Environment

## 2.2 Installing the VMS Operating System in the VAXcluster Environment

**Table 2–2 (Cont.) Information Requested for Local Area and Mixed-Interconnect Configurations**

Item	Response
Re-enter this cluster's password for verification:	Re-enter the password.
Will JUPITR be a disk server (Y/N)?	Enter Y. In local area and mixed-interconnect configurations, the system disk is always served to the cluster. Refer to Section 1.3.3 and Chapter 5 for information on served cluster disks.
Will JUPITR serve HSC disks (Y/N)?	Enter a response appropriate for your configuration.
Enter a value for JUPITR's ALLOCLASS parameter:	If the system will serve HSC disks, enter the HSC's allocation class value. If the system is connected to a dual-ported disk, enter a value from 1-255 that will be used on both sides. Otherwise, enter 0.
Does this cluster contain a quorum disk [N]?	Enter Y or N, depending on your configuration. If you enter Y, the procedure prompts for the name of the quorum disk. Enter the device name of the quorum disk.

### 2.3 Configuring the DECnet–VAX Network

After you have installed the operating system and required licenses, you configure, tailor, and start the DECnet–VAX network. This process typically entails several operations:

- Executing the `SY$MANAGER:NETCONFIG.COM` command procedure.
- Making remote node data available clusterwide.
- Optionally defining an alias node identifier for the cluster. You establish an alias using NCP commands like those shown in step 4 for alias SOLAR. (For more information on alias node identifiers, refer to the *VMS Networking Manual*.) Note that if you plan to define an alias node identifier, you must specify that one cluster node operate as a *router* node when you execute `NETCONFIG.COM`. Note further that you must later enable alias operations for other cluster nodes, as described in Section 2.3.2.
- Starting the network.

To perform these operations, proceed as follows:

- 1 Log in as system manager.
- 2 Execute the command procedure `NETCONFIG.COM`, entering information about your node when prompted, and responding YES when the procedure asks whether you want to configure the network ("want these commands to be executed").

**Note:** When the procedure asks whether you want the network started, answer *NO* if you first want to define a cluster alias.

# Preparing the Cluster Operating Environment

## 2.3 Configuring the DECnet-VAX Network

Example 2-1 shows typical responses for a cluster network configuration session using NETCONFIG.COM.

### Example 2-1 Sample Interactive Network Configuration Session

---

```
$ @NETCONFIG.COM

      DECnet-VAX network configuration procedure

This procedure will help you define the parameters needed to get DECnet
running on this machine.  You will be shown the changes before they are
executed, in case you want to perform them manually.

What do you want your DECnet node name to be?  [JUPITR]: 
What do you want your DECnet address to be?    [2.2]: 
Do you want to operate as a router?  [NO (nonrouting)]: YES
Do you want a default DECnet account?          [YES]: 

      Here are the commands necessary to set up your system.
-----
      .
      .
-----
Do you want these commands to be executed?      [YES]: 
      .
      .

The changes have been made.
If you have not already registered the DECnet-VAX key, then do so now.
After the key has been registered, you should invoke the procedure
SYS$MANAGER:STARTNET.COM to start up DECnet-VAX with these changes.

(If the key is already registered) Do you want DECnet started? [YES] NO
$
```

---

- 3 NETCONFIG.COM creates, in the SYS\$SPECIFIC:[SYSEXE] directory, the permanent remote node database file NETNODE\_REMOTE.DAT, in which remote node data is maintained. To make this data available clusterwide, you must rename the file to the SYS\$COMMON:[SYSEXE] directory:

```
$ RENAMESYS$SPECIFIC:[SYSEXE]NETNODE_REMOTE.DAT -
_ $ SYS$COMMON:[SYSEXE]NETNODE_REMOTE.DAT
```

- 4 If you want to define an alias node identifier for the cluster, invoke the Network Control Program (NCP) Utility to do so. For example:

```
$ RUN SYS$SYSTEM:NCP
NCP> DEFINE NODE 2.1 NAME SOLAR
NCP> DEFINE EXECUTOR ALIAS NODE SOLAR
NCP> EXIT
$
```

The information you specify using these commands is entered in the DECnet-VAX permanent executor database and takes effect when you start the network.

- 5 Start the network:

```
$ @SYS$MANAGER:STARTNET.COM
```

# Preparing the Cluster Operating Environment

## 2.3 Configuring the DECnet–VAX Network

- 6 To ensure that the network is started each time the system boots, add the following line to your site-specific startup command file (for example, SYS\$MANAGER:SYSTARTUP\_V5.COM):

```
$ @SYS$MANAGER:STARTNET.COM
```

For more detailed information on DECnet–VAX configuration issues and procedures, refer to the *VMS Networking Manual*.

### 2.3.1 Copying Remote Node Databases

Some sites with large networks maintain remote node data in a central database file. If this is the case at your site, and if you want to make the data available clusterwide, you can, *after starting the network*, copy remote node database entries from that central file. For example, if the file resides on node SATURN, you could enter the following NCP commands to copy entries from the permanent database on SATURN to the permanent database on your system disk, and then to update your volatile database:

```
NCP> COPY KNOWN NODES FROM SATURN USING PERMANENT TO PERMANENT
NCP> SET KNOWN NODES ALL
```

Note that only node names and addresses are copied. See the *VMS Networking Manual* for more information on copying node databases.

### 2.3.2 Enabling Cluster Alias Operations

If you have defined an alias node identifier for your cluster as described in Section 2.3, you can enable alias operations for other cluster nodes after the nodes have joined the cluster. To enable such operations (that is, to allow a node to accept incoming connect requests directed toward the cluster alias node identifier), follow these steps:

- 1 Log in as system manager and invoke the SYSMAN Utility:

```
$ RUN SYS$SYSTEM:SYSMAN
```

- 2 At the SYSMAN> prompt, enter the following commands:

```
SYSMAN> SET ENVIRONMENT/CLUSTER
%SYSMAN-I-ENV, current command environment:
Clusterwide on local cluster
Username LAZRUS will be used on nonlocal nodes
SYSMAN> SET PROFILE/PRIVILEGES=(OPER,SYSPRV)
SYSMAN> DO MCR NCP SET EXECUTOR STATE OFF
%SYSMAN-I-OUTPUT, command execution on node X...
.
.
SYSMAN> DO MCR NCP DEFINE EXECUTOR ALIAS INCOMING ENABLED
%SYSMAN-I-OUTPUT, command execution on node X...
.
.
SYSMAN> DO @SYS$MANAGER:STARTNET.COM
%SYSMAN-I-OUTPUT, command execution on node X...
.
.
```

# Preparing the Cluster Operating Environment

## 2.4 Coordinating Cluster Command Procedures

---

### 2.4 Coordinating Cluster Command Procedures

You must coordinate your site-specific startup command procedures according to the type of cluster operating environment you want to prepare. For a common-environment cluster, these procedures should perform the same system startup and login functions for each cluster node. For a multiple-environment cluster, you may want some startup commands to remain specific to certain nodes.

Once you have created the common site-specific startup command procedures (for example SYSTARTUP\_V5.COM and SYLOGIN.COM), you can set up each of them as a common file on a cluster-accessible disk or as separate duplicate files.

Using either approach, you can include a command in the node-specific startup file that will invoke the common startup procedure. In a common-environment cluster, the node-specific startup file for each node invokes a common startup procedure, named for example, SYSTARTUP\_COMMON.COM. Thus, each startup procedure on each node would include a command similar to the following:

```
$ @device: [SYSMGR]SYSTARTUP_COMMON.COM
```

Certain startup functions, even in a common-environment cluster, are node specific. Therefore, you should include commands in the node-specific startup procedure on each node to do the following:

- Set up dual-ported and local disks
- Load device drivers
- Set up terminals
- Invoke the common startup command procedure

If the common startup procedure is on a local disk, the node-specific procedure must set up the local disk as a cluster-accessible disk before invoking the common procedure. If the procedure is not on the system disk, the disk on which it resides must be mounted before the procedure can be invoked.

Alternatively, you could set up duplicate copies of the common procedure on a separate volume on each cluster node. To set up a common SYLOGIN procedure, define the logical name SYS\$SYLOGIN on each cluster node to be the full file specification of the procedure. If the common SYLOGIN file is on a cluster-accessible disk, you can include the command that defines SYS\$SYLOGIN in the common startup procedure. If the cluster nodes use separate duplicate copies of SYLOGIN, you should include the definition in the node-specific startup procedure for each node.

For example, the following command defines SYS\$SYLOGIN to be the common file [SYSMGR]SYLOGIN on the cluster-accessible disk WORK5:

```
$ DEFINE/SYSTEM/EXEC SYS$SYLOGIN WORK5: [SYSMGR]SYLOGIN
```

Sections 2.4.1 and 2.4.2 present guidelines for using common and node-specific command procedures to build a cluster environment.

# Preparing the Cluster Operating Environment

## 2.4 Coordinating Cluster Command Procedures

### 2.4.1 Building Common Command Procedures

The first step in preparing a common-environment cluster is to build cluster common startup and login command procedures. In a common-environment cluster, each cluster node executes the common procedures at startup time to set up the same operating environment on each cluster node. Because each node is set up using the common procedures, users can work in the same operating environment no matter which member node they are logged into.

To build these procedures for a cluster in which existing nodes are to be merged, you should compare both the node-specific SYSTARTUP and SYLOGIN command procedures on each node and make any adjustments required. For example, you can compare the procedures from each node and include commands that define the same logical names in the common startup command procedure.

An easy method of comparing the existing procedures and creating common versions is to log into each cluster node (in the single-system environment) and print the existing SYSTARTUP and SYLOGIN command procedure files. You can then use the file listings to compare the procedures. After you have chosen which commands to make common, you can build the common procedures on one of the cluster nodes.

The strategy for clusters being formed from newly installed VMS systems is basically the same as that used for clusters that are to include previously installed systems—that is, include common elements in a common command procedure file. With newly installed systems, however, the SYSTARTUP and SYLOGIN command procedure files are empty. You must therefore build the common procedures from scratch.

For example, you could build a common startup command procedure named SYSTARTUP\_COMMON.COM, and include the commands that you want to be common to all nodes. You must decide which of the following elements you want to include in the common procedure:

- Commands that install images.
- Commands that define logical names; for example, the logical name that refers to the location of SYLOGIN.COM.
- Commands that set up queues. (See Chapter 4 for information on setting up cluster queues.)
- Commands that set up and mount physically accessible mass storage devices. (See Chapter 5 for information on setting up cluster disks.)
- Commands that perform any other common site-specific startup functions. See the *Guide to Setting Up a VMS System* for more information on startup command procedures.

In a common startup command procedure, the execution of commands that set up queues and mount cluster-accessible devices is node dependent. Therefore, you must include conditional DCL commands to control how these commands are executed.

You can include commands that set up queues and mount cluster-accessible devices as part of the common startup procedure or as separate command procedures, such as STARTQ\_COMMON.COM or MOUNT\_COMMON.COM that are invoked by the common procedure. Sample procedures for setting up queues and mounting cluster-accessible volumes are described in Chapter 4 and Chapter 5, respectively.

# Preparing the Cluster Operating Environment

## 2.4 Coordinating Cluster Command Procedures

**Note:** The job-controller queue file, JBCSYSQUE.DAT, must be set up as a common file on a cluster-accessible disk, accessible to all the nodes sharing queues. If you intend to set up common procedures such as SYSTARTUP\_COMMON.COM or STARTQ\_COMMON.COM as common files on a cluster-accessible disk volume, it is a good idea to locate these files on the same cluster-accessible volume containing JBCSYSQUE.DAT.

To build a common SYLOGIN.COM command procedure, include in a common SYLOGIN command file commands that define symbols or that perform other site-specific functions.

---

### 2.4.2 Using Node-Specific System Command Procedures

In a multiple-environment cluster, include elements that you want to remain unique to a node, such as commands to define node-specific logical names, in the node-specific versions of the SYSTARTUP and SYLOGIN files for that node. These files must be placed in the SYS\$SPECIFIC root on each node.

For example, consider a three-node cluster consisting of nodes JUPITR, SATURN, and URANUS. The time-sharing environments on nodes JUPITR and SATURN are the same. URANUS is set up for specific turn key accounts. In this case, you could create common SYSTARTUP and SYLOGIN command procedures for nodes JUPITR and SATURN that set up identical environments on these nodes. The command procedures for node URANUS, however, would be different, set up specifically for URANUS's turn key environment.

---

## 2.5 Coordinating System Files to Define the Cluster User Environment

To prepare the cluster user environment, you must coordinate the following system files:

- SYSUAF.DAT
- NETPROXY.DAT
- RIGHTSLIST.DAT
- VMSMAIL\_PROFILE.DATA
- JBCSYSQUE.DAT
- NETNODE\_REMOTE.DAT<sup>1</sup>

These files, which are part of the VMS operating system, contain information that controls such functions as user logins, proxy login access, mail, and access to files and job queues. By coordinating these files, you can define either a common-environment or a multiple-environment cluster.

To define a common-environment cluster, you use a common version of each system file and place the files in the SYS\$COMMON:[SYSEXEC] directory on a common system disk.

**Note:** If you want to set up a common-environment cluster with more than one common system disk (for example, in local area or mixed-interconnect

---

<sup>1</sup> Depending on the network environment you have set up at your site, you may need to coordinate other network files. For detailed information on coordinating network files in the VAXcluster environment, see the *VMS Networking Manual*.

# Preparing the Cluster Operating Environment

## 2.5 Coordinating System Files to Define the Cluster User Environment

configurations), you must coordinate files on each disk and ensure that the disks are mounted with each cluster reboot. Refer to Section 2.5.4 for instructions.

To define a multiple-environment cluster, you use node-specific versions of one or more system files. For example, if you want to allow only a certain group of users to log in to node URANUS, you would create a node-specific version of SYSUAF.DAT and place that file in URANUS's SYS\$SPECIFIC:[SYSEXE] directory. That directory may be located in URANUS's root on a common system disk ([SYSB.SYSEXE] on JUPITR for instance), or on an individual system disk that you have set up on URANUS.

Sections 2.5.1 through 2.5.3 describe the procedures for building a common version of system files. For information on individual system files, refer to the *Guide to Setting Up a VMS System*.

---

### 2.5.1 Coordinating User Accounts

In a common-environment cluster, you must coordinate the user accounts from each node and build common versions of the following files:

- SYSUAF.DAT
- NETPROXY.DAT

If you are setting up a common-environment cluster that consists of newly installed systems, you can follow instructions in the *Guide to Setting Up a VMS System* to build common SYSUAF.DAT and NETPROXY.DAT files. Because the SYSUAF.DAT file on new VMS systems is empty except for the four DIGITAL-supplied accounts, very little coordination is necessary.

If, however, the cluster is to include one or more systems that have been running with node-specific SYSUAF.DAT and NETPROXY.DAT files, you must create common versions of the files. Procedures for building a common SYSUAF.DAT file from node-specific files are described in Appendix B.

The procedure for creating a common NETPROXY.DAT file is basically the same as that for creating a common SYSUAF.DAT. The main difference is that less coordination is needed when merging the individual NETPROXY.DAT files. For example, UICs are not used in the NETPROXY records, and therefore need not be coordinated.

You should decide which existing proxy login records you want to keep on the cluster and include these records in the common NETPROXY.DAT file. As with the SYSUAF.DAT files, you can use the Convert Utility to merge the NETPROXY.DAT file from each node to create a common file.

Once you have created individual SYSUAF.DAT and NETPROXY.DAT files, you can set up each of them as either a common file on a cluster-accessible disk or as separate duplicate files. Note, however, that if you elect to use duplicate files, you must update all copies whenever you make changes.

If your cluster is running from one common system disk, make sure that SYSUAF.DAT and NETPROXY.DAT are included in SYS\$COMMON:[SYSEXE].

If your cluster is running from any other system disk configuration, you must decide where to locate SYSUAF.DAT and NETPROXY.DAT. Once you have placed these two files in a directory, you must define clusterwide logical names to point to them.

# Preparing the Cluster Operating Environment

## 2.5 Coordinating System Files to Define the Cluster User Environment

Assume that the disk WORK5: is a volume shared by all nodes in the cluster and that it contains cluster common SYSUAF.DAT and NETPROXY.DAT files. The following commands define system logical names that point to the location of the common files:

```
$ DEFINE/SYSTEM/EXEC SYSUAF WORK5:[SYSEXE]SYSUAF
$ DEFINE/SYSTEM/EXEC NETPROXY WORK5:[SYSEXE]NETPROXY
```

You must add the DEFINE commands to the common site-specific startup command file. After you have copied the files to the appropriate directory on the cluster-accessible disk volume, you should delete these files from the system disk.

### 2.5.2 Preparing the MAIL Database

In a common-environment cluster, you may want to prepare a common mail database to allow users to use the Mail Utility (MAIL) to send and read their MAIL messages from any node in the cluster.

Each time MAIL executes in a single-system environment, it accesses a database file named SYS\$SYSTEM:VMSMAIL\_PROFILE.DATA. To set up VMSMAIL\_PROFILE.DATA as a common file, define the logical name VMSMAIL\_PROFILE to be the complete file specification of the common file by specifying the DEFINE command in the following format:

```
$ DEFINE/SYSTEM/EXEC VMSMAIL_PROFILE file-spec
```

You must make sure that you define the logical name before you invoke MAIL for the first time. When invoked for the first time, MAIL creates the database file, VMSMAIL\_PROFILE.DATA, in SYS\$SYSTEM by default. By defining VMSMAIL\_PROFILE to be the location of a common file on a cluster-accessible disk, you cause MAIL to create and use that file.

If your cluster is running from one common system disk, define VMSMAIL\_PROFILE to be SYS\$COMMON:[SYSEXE]VMSMAIL\_PROFILE and invoke the Mail Utility, by entering the following two commands:

```
$ DEFINE/SYSTEM/EXEC VMSMAIL_PROFILE SYS$COMMON:[SYSEXE]VMSMAIL_PROFILE
$ MAIL
```

VMSMAIL\_PROFILE.DATA will be created in the common system directory. You will no longer need to use the logical name, or make changes to the site-specific startup command file.

If your cluster is running from any other system disk configuration, you must decide where to locate the common VMSMAIL\_PROFILE.DATA file. (Typically, you would place this file in the same directory in which SYSUAF.DAT and NETPROXY.DAT reside—for example, WORK5:[SYSEXE].) You then define a logical name for the file and invoke the Mail Utility:

```
$ DEFINE/SYSTEM/EXEC VMSMAIL_PROFILE WORK5:[SYSEXE]VMSMAIL_PROFILE
$ MAIL
```

The DEFINE command defines VMSMAIL\_PROFILE.DATA to be a file located in [SYSEXE] on the cluster-accessible disk volume WORK5. The first time MAIL is invoked, VMSMAIL\_PROFILE.DATA is created in WORK5:[SYSEXE]. Subsequently, MAIL uses this file as the database. You must also add the DEFINE command to the common site-specific startup command file.

# Preparing the Cluster Operating Environment

## 2.5 Coordinating System Files to Define the Cluster User Environment

### 2.5.3 Preparing the Rights Database

In a common-environment cluster, you can create a common version of the rights database. The rights database is a file that associates users of the system or cluster with special names called identifiers. The rights database file, RIGHTSLIST.DAT, is the basis of the ACL-based protection scheme. For more information on ACLs, see the description in the *Guide to VMS System Security*.

The cluster or security manager maintains the rights database, adding and removing identifiers as needs change. By allowing groups of users to hold identifiers, the manager has now created a different kind of group designation than the one used with the user's UIC. This alternative grouping allows the holders of the identifier to make more efficient use of resources. It also permits each user to be a member of multiple overlapping groups.

For information on how the rights database is set up at the local node level, see the *VMS Authorize Utility Manual*.

If your cluster is running from one common system disk, the installation or upgrade procedure will place the RIGHTSLIST.DAT file in SYS\$COMMON:[SYSEXEXE]. No further action is required on your part.

If your cluster is running from any other system disk configuration, copy SYS\$SYSTEM:RIGHTSLIST.DAT to the directory in which you placed the SYSUAF, NETPROXY, and VMSMAIL\_PROFILE system files. Then define a clusterwide logical name for the RIGHTSLIST.DAT file. For example:

```
$ DEFINE/SYSTEM/EXEC RIGHTSLIST WORK5:[SYSEXEXE]RIGHTSLIST
```

You must also add this DEFINE command to the common site-specific startup command file.

### 2.5.4 Coordinating Shared System Files in Clusters with Multiple Common System Disks

To prepare a common user environment for any cluster configuration that includes more than one common system disk, you must coordinate the system files listed in Section 2.5. In local area and mixed-interconnect clusters, you must also coordinate the file SYS\$MANAGER:NETNODE\_UPDATE.COM.

Proceed as follows:

- 1 Edit the file [VMS\$COMMON.SYSMGR]SYLOGICALS.COM on each system disk and define logical names that specify the location of the cluster common files. For example, if the files are to be located on \$1\$DJA16, you could define logical names like the following:

```
$ DEFINE/SYSTEM/EXEC SYSUAF -
    $1$DJA16:[VMS$COMMON.SYSEXEXE]SYSUAF.DAT
$ DEFINE/SYSTEM/EXEC NETPROXY -
    $1$DJA16:[VMS$COMMON.SYSEXEXE]NETPROXY.DAT
$ DEFINE/SYSTEM/EXEC RIGHTSLIST -
    $1$DJA16:[VMS$COMMON.SYSEXEXE]RIGHTSLIST.DAT
$ DEFINE/SYSTEM/EXEC VMSMAIL_PROFILE -
    $1$DJA16:[VMS$COMMON.SYSEXEXE]VMSMAIL_PROFILE.DAT
$ DEFINE/SYSTEM/EXEC NETNODE_REMOTE -
    $1$DJA16:[VMS$COMMON.SYSEXEXE]NETNODE_REMOTE.DAT
$ DEFINE/SYSTEM/EXEC NETNODE_UPDATE -
    $1$DJA16:[VMS$COMMON.SYSMGR]NETNODE_UPDATE.COM
```

# Preparing the Cluster Operating Environment

## 2.5 Coordinating System Files to Define the Cluster User Environment

- 2 To ensure that the system disks are correctly mounted with each reboot, follow these steps:
  - a. Copy the file SYS\$EXAMPLES:CLU\_MOUNT\_DISK.COM to the directory [VMS\$COMMON.SYSMGR].
  - b. Edit SYLOGICALS.COM and include commands to mount the system disks with appropriate volume labels. For example, if the system disks are \$1\$DJA16 and \$1\$DJA17, you would include commands like these:
- 3 In the site-specific file used for queue setup, specify the location of the job controller queue file (JBCSYSQUE.DAT), using a command like the following:

```
$ @SYS$SYSDEVICE: [VMS$COMMON.SYSMGR] CLU_MOUNT_DISK.COM -  
    $1$DJA16: volume-label  
$ @SYS$SYSDEVICE: [VMS$COMMON.SYSMGR] CLU_MOUNT_DISK.COM -  
    $1$DJA17: volume-label
```

```
$ START/QUEUE/MANAGER $1$DJA16: [VMS$COMMON.SYSEXE] JBCSYSQUE.DAT
```

When you execute CLUSTER\_CONFIG.COM to add nodes to a cluster with more than one common system disk, a different device name must be used for each system disk on which nodes are added. For this reason, CLUSTER\_CONFIG.COM supplies as a default device name the logical volume name (for example, DISK\$MARS\_SYS1) of SYS\$SYSDEVICE: on the local system.

Different device names ensure that each node added will have a unique root directory specification, even if the system disks contain roots with the same name—for example, DISK\$MARS\_SYS1:[SYS10] and DISK\$MARS\_SYS2:[SYS10].



# 3

---

## Building and Maintaining the Cluster

---

After you have prepared the cluster operating environment as described in Chapter 2, you are ready to set up your site-specific configuration. This chapter provides information to help you build and maintain your cluster. Topics include the following:

- Planning configuration procedures
- Configuring the cluster
- Reconfiguring the cluster after a major change
- Maintaining the cluster

Before you attempt to configure your cluster, be sure you understand the discussions in Chapters 1 and 2.

---

### 3.1 Planning Configuration Procedures

The planning needed to configure a cluster depends on several factors:

- The configuration type (CI-only, local area, or mixed interconnect)
- The components to be included in the cluster
- The configuration function you want to execute

Because you must execute the command procedure `SYS$MANAGER:CLUSTER_CONFIG.COM` to perform all basic configuration functions, it is important that you understand the operations that the procedure can perform. These are described in Section 3.1.1.

If you intend to set up a local area or mixed-interconnect cluster, you must, before executing `CLUSTER_CONFIG.COM`, do the following:

- Determine locations and sizes for satellite page and swap files
- Select cluster boot servers
- Specify allocation classes for cluster nodes and disks (also applicable for CI-only configurations)

Guidelines are provided in Sections 3.1.2, 3.1.3, and 3.1.4.

Note that some configuration functions, such as adding or removing a voting cluster node, require one or more additional operations. Refer to Section 3.3 for instructions.

# Building and Maintaining the Cluster

## 3.1 Planning Configuration Procedures

### 3.1.1 CLUSTER\_CONFIG.COM Functions

When you invoke CLUSTER\_CONFIG.COM, the procedure displays a menu of configuration options. By selecting the appropriate option, you can configure the cluster easily and reliably, *without invoking VMS utilities directly*. You use CLUSTER\_CONFIG.COM to perform these functions:

- Add a node to the cluster.
- Remove a node from the cluster.
- Change a cluster node's characteristics.
- Create a duplicate system disk.

Following is a summary of the operations that CLUSTER\_CONFIG.COM performs for each configuration option:

- ADD** Establish the new node's root directory on a cluster common system disk and generate the node's system parameter files (VAXVMSSYS.PAR and MODPARAMS.DAT) in its SYS\$SPECIFIC:[SYSEXE] directory.
- Update the permanent and volatile remote node network databases for the system on which CLUSTER\_CONFIG.COM is executed (local system) to add the new node. If the new node is a satellite, update SYS\$MANAGER:NETNODE\_UPDATE.COM on the local system.
- Generate the new node's page and swap files (PAGEFILE.SYS and SWAPFILE.SYS).
- Optionally set up a cluster quorum disk.
- Set allocation class (ALLOCLASS) value for the new node, if the node is being added as a disk server.
- Generate an initial (temporary) startup procedure for the new node. This initial procedure runs NETCONFIG.COM to configure the network, runs AUTOGEN to set appropriate SYSGEN parameter values for the node, and reboots the node with normal startup procedures.
- REMOVE** Delete another node's root directory and its contents from the local system's system disk. If the node being removed is a satellite, update SYS\$MANAGER:NETNODE\_UPDATE.COM on the local system.
- Update the permanent and volatile remote node network databases on the local system.
- CHANGE** Enable or disable the local system as a disk server; enable or disable the local system as a boot server; enable or disable the Ethernet for cluster communications on the local system; enable or disable a quorum disk on the local system; change the local system's ALLOCLASS value; change a satellite's Ethernet hardware address. *Procedure displays CHANGE menu and prompts for appropriate information.*
- CREATE** Duplicate the local system's system disk and remove all system roots from the new disk.

# Building and Maintaining the Cluster

## 3.1 Planning Configuration Procedures

### 3.1.2 Determining Locations and Sizes for Satellite Page and Swap Files

When you add a node to the cluster, CLUSTER\_CONFIG.COM prompts for the sizes and location of the node's page and swap files. (The default sizes supplied by the procedure are minimums.) Depending on the configuration of your system disk and your network, you may realize a performance improvement in local area and mixed-interconnect configurations by locating page and swap files for satellites on a satellite's local RD series disk, if such a disk is available.

To set up page and swap files on a satellite's local disk, CLUSTER\_CONFIG.COM creates (in the satellite's [SYSx.SYSEX] directory on the boot server's system disk) the command procedure SATELLITE\_PAGE.COM. This procedure executes when AUTOGEN reboots the satellite at the end of CLUSTER\_CONFIG.COM, and it performs the following functions:

- Mounts the satellite's local disk with a volume label in the format 'node'\_SCSSYSTEMID.
- Installs the page and swap files on the local disk.

If you want to alter the volume label, follow these steps after the satellite has been added to the cluster:

1 Enter a DCL command in the following format:

```
$ SET VOLUME/LABEL=volume-label device-spec[:]
```

Note that the SET VOLUME command requires write access (W) to the index file on the volume. If you are not the volume's owner, you must have either a system UIC or the SYSPRV privilege.

2 Update SATELLITE\_PAGE.COM to reflect the new label.

To relocate the satellite's page and swap files (for example, from the satellite's local disk to the boot server's system disk, or the reverse), or to change file sizes, the easiest way is to remove the satellite from the cluster and then add it again, using CLUSTER\_CONFIG.COM.

### 3.1.3 Selecting Boot Servers for Mixed-Interconnect Clusters

While every mixed-interconnect cluster must have at least one boot server, multiple servers offer the following advantages:

- Higher availability—satellites can access served disks and boot, even if one of the boot servers is temporarily unavailable.
- Better workload balancing—the task of serving HSC disks to satellites can place a significant load on a boot server. With multiple boot servers, this workload is distributed across more processors and Ethernet adapters.

Use as boot servers the most powerful machines you have available. Processors with the power of a VAX 8530 or greater have sufficient CPU power to perform disk-serving functions without serious degradation in response time. Less powerful machines can become overloaded when serving many busy satellites, or when many satellites boot simultaneously.

# Building and Maintaining the Cluster

## 3.1 Planning Configuration Procedures

Note, however, that two or more lower-powered boot servers provide better performance than a single high-powered server. Multiple servers give better availability, and they distribute the workload across more Ethernet adapters. If, for example, you have 5 VAX processors available—a VAX 8800, a VAX 8350, two VAX-11/785s, and a VAX-11/750—use all the machines as boot servers except the VAX-11/750.

If you have several processors of roughly comparable power, it is reasonable to use them all as boot servers. This arrangement gives optimal load balancing. And if one machine fails or is shut down, others remain available to serve satellites.

After CPU power, the second most important factor in selecting a boot server is the speed of its Ethernet adapter. Boot servers should be equipped with the highest-bandwidth Ethernet adapters you have available for the machines.

### 3.1.4 **Specifying Allocation Class Values in Mixed-Interconnect Clusters**

Before setting up any mixed-interconnect cluster, you must determine allocation class values for the boot server(s) and HSCs. It is easiest to use the same value for all HSCs and all boot servers—you can arbitrarily choose a number between 1 and 255. Note, however, that to change the allocation class value on any CI-connected VAX processor or HSC, you must shut down and reboot the entire cluster. (See Section 3.3.)

Every device allocation class name (name of the form \$1\$ddcu) must be unique across all boot servers and HSCs. For RA series disks, make sure that all the removable unit plugs on all disks of that allocation class are unique. As long as you have no more than 256 such disks, this is easy to accomplish.

Assume, for instance, that 10 disks are dual pathed between the HSCs VOYGR1 and VOYGR2, and 10 others are dual pathed between the HSCs VIKNG1 and VIKNG2. Provided that all 20 disks have unique unit numbers, you can assign all four HSCs the same allocation class value.

If you have more than 256 HSC-connected disks, you must, to ensure unique disk names, use two or more allocation classes for the HSCs. You must also configure one or more nodes to serve HSC disks and assign allocation class values accordingly. To perform those operations, you can execute the CLUSTER\_CONFIG.COM CHANGE function, described in Section 3.2.3.

Additionally, you must make sure that all locally connected disks have unique allocation class names. Consider the following example: if nodes SATURN and URANUS each have one BDA disk controller with a single-pathed RA81 disk connected to it, and if both controllers have an allocation class value of 1, the RA81 connected to SATURN with unit plug 0 will receive the device name \$1\$DUA0. Likewise, the RA81 connected to URANUS with unit plug 0 will be \$1\$DUA0. Because both disks have the same name, they appear to VMS software to be the same disk, and confusion or even corruption could result. You can avoid this potential problem by switching one disk's unit plug.

Note that because fewer unit numbers are available for MASSBUS or UNIBUS disks, fewer unique disk names are possible. To ensure that disk names remain unique in your cluster, you may have to relocate such disks or disqualify a node as a disk server.

# Building and Maintaining the Cluster

## 3.2 Configuring the Cluster

### 3.2 Configuring the Cluster

To perform configuration functions, you execute CLUSTER\_CONFIG.COM. Before invoking the procedure, be sure to verify the following:

- You are logged in to the system manager's account on an appropriate node. If you are building a new local area or mixed-interconnect cluster, you must be logged in on a node that you want to set up as a boot server. If you are adding a satellite node, you must be logged in on a boot server. Note that the process privileges SYSPRV, OPER, CMKRNL, BYPASS, and NETMBX are required, because the procedure performs sensitive system operations.
- The DECnet-VAX network is up and running.
- You have at hand the data listed in Table 3-1. Note that some items are configuration specific.
- *If your configuration has two or more system disks, you have coordinated cluster common files, as described in Section 2.5.4.*

Sections 3.2.1 through 3.2.6 provide examples of typical interactive CLUSTER\_CONFIG.COM sessions. Section 3.3 describes tasks you must perform after executing CLUSTER\_CONFIG.COM to make major configuration changes.

**Caution:** You may not initiate concurrent CLUSTER\_CONFIG.COM sessions.

**Table 3-1 Data Requested by CLUSTER\_CONFIG.COM**

Item	How To Specify Or Obtain
Device name of cluster system disk on which root directories will be created.	System manager specifies. Default is logical volume name of SYS\$SYSDEVICE: (for example, DISK\$VAXVMSRL5:).
Node's root directory name on cluster system disk.	System manager specifies. Name must be of the form SYSx. For CI-connected nodes, x is a hexadecimal digit in the range 1 through 9 or A through D (for example, SYS1 or SYSA). For satellites, x must be in the range from 10 through FFFF. Procedure supplies valid default.
Node's DECnet node name.	Network manager supplies. Name must be from 1 to 6 alphanumeric characters and <i>may not</i> include dollar signs or underscores.
Node's DECnet node address.	Network manager supplies.
Cluster group number and password if CHANGE is run to enable cluster communications over the Ethernet.	System manager specifies.

# Building and Maintaining the Cluster

## 3.2 Configuring the Cluster

**Table 3–1 (Cont.) Data Requested by CLUSTER\_CONFIG.COM**

Item	How To Specify Or Obtain
If node is a satellite, satellite's Ethernet hardware address. Address has the form xx-xx-xx-xx-xx-xx. Note that you must include the dashes when you specify a hardware address.	<p>When DECnet-VAX network is running on boot server, proceed as follows:</p> <ul style="list-style-type: none"><li>For MicroVAX II and VAXstation II satellites, enter the following commands at satellite's console: <pre>&gt;&gt;&gt; B/100 XQ Bootfile: READ_ADDR</pre></li><li>For MicroVAX 2000 and VAXstation 2000 satellites, enter the following commands at successive console-mode prompts:<sup>1</sup> <pre>&gt;&gt;&gt; T 53 2 ?&gt;&gt;&gt; 3 &gt;&gt;&gt; B/100 ES Bootfile: READ_ADDR</pre></li><li>For MicroVAX 3xxx series satellites, enter the following command at satellite's console: <pre>&gt;&gt;&gt; SHOW ETHERNET</pre></li><li>For VAXstation 8000 satellites, enter commands as shown in the following example, and then construct the Ethernet hardware address from the values displayed by the system. <pre>&gt;&gt;&gt; E/P/1 2000C218 87654321 &gt;&gt;&gt; E/P/1 2000C21C 0000BC9A</pre></li></ul> <p>In this example, the address is 21-43-65-87-9A-BC.</p>
Workstation windowing system.	System manager specifies. Workstation software must be installed before workstation satellites are added. If it is not, the procedure indicates that fact.
Location and sizes of page and swap files.	System manager specifies.
Value for local system's allocation class (ALLOCLASS) parameter.	System manager specifies.
Device name of quorum disk.	System manager specifies.

<sup>1</sup>If the second prompt appears as 3 ?> > > , press RETURN.

### 3.2.1 Adding a Node to the Cluster

Once you have made the necessary preparations, you can execute CLUSTER\_CONFIG.COM to add a new node to the cluster.

- If you are setting up a CI-only cluster, invoke CLUSTER\_CONFIG.COM on an active cluster system and select the ADD function.

# Building and Maintaining the Cluster

## 3.2 Configuring the Cluster

- If you are setting up a new local area or mixed-interconnect cluster, follow these steps:
  - 1 Invoke `CLUSTER_CONFIG.COM` and execute the `CHANGE` function described in Section 3.2.3 to enable the local system as a boot server.
  - 2 After the `CHANGE` function completes, execute the `ADD` function to add either CI-connected nodes or satellites to the cluster. *To add satellites, you must be logged in on a cluster boot server.*

While adding nodes, you may want to disable broadcast messages to your terminal—the `ADD` function generates many such messages. To disable the messages, you can enter the `DCL` command `REPLY/DISABLE=(NETWORK, CLUSTER)`.

Whenever you add a voting (non-satellite) member to the cluster, you must, after the `ADD` function completes, reconfigure the cluster, following instructions in Section 3.3. In addition, if you add a CI-connected node that boots from a cluster common disk, you must create a new default bootstrap command procedure for the node before booting it into the cluster. For instructions, refer to your processor-specific installation and operations guide.

Examples 3-1 and 3-2 illustrate the use of `CLUSTER_CONFIG.COM` on node `JUPITR` to add, respectively, CI-connected node `SATURN` and satellite node `EUROPA` to the cluster.

**Caution:** If either the local system or the new node should fail before the `ADD` function completes, you must, after normal conditions are restored, perform the `REMOVE` function to erase any invalid data, and then restart the `ADD` function.

### Example 3-1 Sample Interactive `CLUSTER_CONFIG.COM` Session to Add a CI-Connected Node as a Boot Server

---

```
$ @CLUSTER_CONFIG.COM
```

#### Cluster Configuration Procedure

Use `CLUSTER_CONFIG.COM` to set up or change a VAXcluster configuration. To ensure that you have the required privileges, invoke this procedure from the system manager's account.

Enter ? for help at any prompt.

1. `ADD` a node to the cluster.
2. `REMOVE` a node from the cluster.
3. `CHANGE` a cluster node's characteristics.
4. `CREATE` a second system disk for `JUPITR`.

Enter choice [1]:

The `ADD` function adds a new node to the cluster.

If the node being added is a voting member, `EXPECTED_VOTES` in all other cluster members' `MODPARAMS.DAT` must be adjusted, and the cluster must be rebooted.

If the new node is a satellite, the network databases on `JUPITR` are updated. The network databases on all other cluster members must be updated.

---

Example 3-1 Cont'd. on next page

# Building and Maintaining the Cluster

## 3.2 Configuring the Cluster

### Example 3-1 (Cont.) Sample Interactive CLUSTER\_CONFIG.COM Session to Add a CI-Connected Node as a Boot Server

---

For instructions, see the VMS VAXcluster Manual.

What is the node's DECnet node name? SATURN

What is the node's DECnet address? 2.3

Will SATURN be a satellite [Y]? N

Will SATURN be a boot server [Y]?  RET

This procedure will now ask you for the device name of SATURN's system root.

The default device name (DISK\$VAXVMSRL5:) is the logical volume name of

SYS\$SYSDEVICE:.

What is the device name for SATURN's system root [DISK\$VAXVMSRL5:]?  RET

What is the name of the new system root [SYSA]?  RET

Creating directory tree SYSA...

%CREATE-I-CREATED, \$1\$DJA11:<SYSA> created

%CREATE-I-CREATED, \$1\$DJA11:<SYSA.SYSEXEXE> created

System root SYSA created.

Enter a value for SATURN's ALLOCLASS parameter: 1

Does this cluster contain a quorum disk [N]? Y

What is the device name of the quorum disk? \$1\$DJA12

Updating network database...

Size of page file for SATURN [10000 blocks]? 50000

Size of swap file for SATURN [8000 blocks]? 20000

Will a local (non-HSC) disk on SATURN be used for paging and swapping? N

If you specify a device other than DISK\$VAXVMSRL5: for SATURN's page and swap files, this procedure will create PAGEFILE\_SATURN.SYS and SWAPFILE\_SATURN.SYS in the <SYSEXEXE> directory on the device you specify.

What is the device name for the page and swap files [DISK\$VAXVMSRL5:]?  RET

%SYSGEN-I-CREATED, \$1\$DJA11:<SYSA.SYSEXEXE>PAGEFILE.SYS;1 created

%SYSGEN-I-CREATED, \$1\$DJA11:<SYSA.SYSEXEXE>SWAPFILE.SYS;1 created

The configuration procedure has completed successfully.

SATURN has been configured to join the cluster.

Before booting SATURN, you must create a new default bootstrap command procedure for SATURN. See your processor-specific installation and operations guide for instructions.

The first time SATURN boots, NETCONFIG.COM and AUTOGEN.COM will run automatically.

The following parameters have been set for SATURN:

```
VOTES = 1
EXPECTED_VOTES = 2
QDSKVOTES = 1
```

After SATURN has booted into the cluster, you must increment the value for EXPECTED\_VOTES in every cluster member's MODPARAMS.DAT. You must then reconfigure the cluster, using the procedure described in the VMS VAXcluster Manual.

---

# Building and Maintaining the Cluster

## 3.2 Configuring the Cluster

### Example 3-2 Sample Interactive CLUSTER\_CONFIG.COM Session to Add a Satellite Node with Local Page and Swap Files

---

\$ @CLUSTER\_CONFIG.COM

#### Cluster Configuration Procedure

Use CLUSTER\_CONFIG.COM to set up or change a VAXcluster configuration. To ensure that you have the required privileges, invoke this procedure from the system manager's account.

Enter ? for help at any prompt.

1. ADD a node to the cluster.
2. REMOVE a node from the cluster.
3. CHANGE a cluster node's characteristics.
4. CREATE a second system disk for JUPITR.

Enter choice [1]:

The ADD function adds a new node to the cluster.

If the node being added is a voting member, EXPECTED\_VOTES in all other cluster members' MODPARAMS.DAT must be adjusted, and the cluster must be rebooted.

If the new node is a satellite, the network databases on JUPITR are updated. The network databases on all other cluster members must be updated.

For instructions, see the VMS VAXcluster Manual.

What is the node's DECnet node name? EUROPA

What is the node's DECnet address? 2.21

Will EUROPA be a satellite [Y]?

Verifying circuits in network database...

This procedure will now ask you for the device name of EUROPA's system root. The default device name (DISK\$VAXVMSRL5:) is the logical volume name of SYS\$SYSDEVICE:.

What is the device name for EUROPA'S system root [DISK\$VAXVMSRL5:]?

What is the name of the new system root [SYS10]?

Allow conversational bootstraps on EUROPA [NO]?

The following workstation windowing options are available:

1. No workstation software
2. VWS Workstation Software

Enter choice [1]: 2

---

Example 3-2 Cont'd. on next page

# Building and Maintaining the Cluster

## 3.2 Configuring the Cluster

### Example 3-2 (Cont.) Sample Interactive CLUSTER\_CONFIG.COM Session to Add a Satellite Node with Local Page and Swap Files

---

```
Creating directory tree SYS10...
%CREATE-I-CREATED, $1$DJA11:<SYS10> created
%CREATE-I-CREATED, $1$DJA11:<SYS10.SYSEXEXE> created
.
.
.
System root SYS10 created.
Will EUROPA be a disk server [N]?  RET
What is EUROPA's Ethernet hardware address? 08-00-2B-03-51-75
Updating network database...
Size of pagefile for EUROPA [10000 blocks]? 20000
Size of swap file for EUROPA [8000 blocks]? 12000
Will a local disk on EUROPA be used for paging and swapping? YES
Creating temporary page file in order to boot EUROPA for the first time...
%SYSGEN-I-CREATED, $1$DJA11:<SYS10.SYSEXEXE>PAGEFILE.SYS;1 created

    This procedure will now wait until EUROPA joins the cluster.

    Once EUROPA joins the cluster, this procedure will ask you
    to specify a local disk on EUROPA for paging and swapping.

    Please boot EUROPA now.

Waiting for EUROPA to boot...
.
.
.
(User enters boot command at satellite's console-mode prompt (>>>)).
For MicroVAX II, VAXstation II, and MicroVAX 3xxx series satellites, user enters B XQ.
For MicroVAX 2000 and VAXstation 2000 satellites, user enters B ES.
For VAXstation 8000 satellites, user enters B ET60)
.
.
.
The local disks on EUROPA are:

Device          Device      Error      Volume      Free      Trans      Mnt
Name            Status      Count      Label       Blocks   Count     Cnt
EUROPA$DUA0:    Online      0          Label       23600    23600    0
EUROPA$DUA1:    Online      0          Label       23600    23600    0

Which disk can be used for paging and swapping? EUROPA$DUA0:
May this procedure INITIALIZE EUROPA$DUA0: [YES]? NO
Mounting EUROPA$DUA0:...
PAGEFILE.SYS already exists on EUROPA$DUA0:

*****
Directory EUROPA$DUA0:[SYS0.SYSEXEXE]
PAGEFILE.SYS;1      23600/23600
Total of 1 file, 23600/23600 blocks.
*****
```

---

Example 3-2 Cont'd. on next page

# Building and Maintaining the Cluster

## 3.2 Configuring the Cluster

### Example 3–2 (Cont.) Sample Interactive CLUSTER\_CONFIG.COM Session to Add a Satellite Node with Local Page and Swap Files

---

```
What is the file specification for the page file on
EUROPA$DUA0: [ <SYSO.SYSEXE>PAGEFILE.SYS ]? 
%CREATE-I-EXISTS, EUROPA$DUA0:<SYSO.SYSEXE> already exists
This procedure will use the existing pagefile,
EUROPA$DUA0:<SYSO.SYSEXE>PAGEFILE.SYS;.

SWAPFILE.SYS already exists on EUROPA$DUA0:

*****

Directory EUROPA$DUA0: [SYSO.SYSEXE]
SWAPFILE.SYS;1      12000/12000

Total of 1 file, 12000/12000 blocks.

*****

What is the file specification for the swap file on
EUROPA$DUA0: [ <SYSO.SYSEXE>SWAPFILE.SYS ]? 
This procedure will use the existing swapfile,
EUROPA$DUA0:<SYSO.SYSEXE>SWAPFILE.SYS;.

    AUTOGEN will now reconfigure and reboot EUROPA automatically.
    These operations will complete in a few minutes, and a
    completion message will be displayed at your terminal.

The configuration procedure has completed successfully.
```

---

#### 3.2.1.1 Updating Network Data after Adding a Satellite

Whenever you add a satellite, CLUSTER\_CONFIG.COM updates both the permanent and volatile remote node network databases on the boot server. However, the volatile databases on other cluster members are not automatically updated. To share the new data throughout the cluster, you must update the volatile databases on all other cluster members. Log in as system manager, invoke the SYSMAN Utility, and enter the following commands at the SYSMAN> prompt:

```
SYSMAN> SET ENVIRONMENT/CLUSTER
%SYSMAN-I-ENV, current command environment:
  Clusterwide on local cluster
  Username LAZRUS      will be used on nonlocal nodes
SYSMAN> SET PROFILE/PRIVILEGES=(OPER,SYSPRV)
SYSMAN> DO MCR NCP SET KNOWN NODES ALL
%SYSMAN-I-OUTPUT, command execution on node X...
.
.
.
SYSMAN> EXIT
$
```

# Building and Maintaining the Cluster

## 3.2 Configuring the Cluster

---

### 3.2.1.2 Restoring a Satellite's Network Data

The first time you execute CLUSTER\_CONFIG.COM to add a satellite, the procedure creates the file NETNODE\_UPDATE.COM in the boot server's SYS\$SPECIFIC:[SYSMGR] directory.<sup>1</sup> This file, which is updated each time you add or remove a satellite, or change its Ethernet hardware address, contains all essential network configuration data for the satellite. If an unexpected condition at your site should cause configuration data to be lost, you can use NETNODE\_UPDATE.COM to restore it. You can also read the file when you need to obtain data about individual satellites. Note that you may want to edit the file occasionally to remove obsolete entries.

Example 3-3 shows the contents of the file after satellite nodes EUROPA and GANYMD have been added to the cluster.

#### Example 3-3 Sample NETNODE\_UPDATE.COM File

---

```
$ run sys$system:ncp
  define node EUROPA address 2.21
  define node EUROPA hardware address 08-00-2B-03-51-75
  define node EUROPA load assist agent sys$share:niscs_laa.exe
  define node EUROPA load assist parameter $1$DJA11:<SYS10.>
  define node EUROPA tertiary loader sys$system:tertiary_vmb.exe
  define node GANYMD address 2.22
  define node GANYMD hardware address 08-00-2B-03-58-14
  define node GANYMD load assist agent sys$share:niscs_laa.exe
  define node GANYMD load assist parameter $1$DJA11:<SYS11.>
  define node GANYMD tertiary loader sys$system:tertiary_vmb.exe
```

---

---

### 3.2.1.3 Controlling Clusterwide Broadcast Messages on Satellites and Boot Servers

When a satellite node joins the cluster, broadcasts for all message classes are initially enabled for the satellite by default. Users can disable such broadcasts selectively by including a form of the DCL command SET BROADCAST in their LOGIN.COM files. For example, the following command would disable OPCOM and SHUTDOWN messages:

```
$ SET BROADCAST=(NOOPCOM, NOSHUTDOWN)
```

Note that broadcasts to the operator console terminal (OPA0:) on satellite workstation nodes are disabled by default and should remain disabled at all times. Users who want to receive broadcast messages can create a terminal window, and then enter the DCL command REPLY/ENABLE. (This command requires OPER privilege.) For more detailed information on workstation operations, refer to the documentation supplied with the workstation software.

In large clusters, state transitions (nodes joining or leaving the cluster) will generate many multi-line OPCOM messages on a boot server's console device. You can abbreviate such messages by including the DCL command REPLY/DISABLE=CLUSTER in the appropriate site-specific startup command file, or by entering the command interactively from the system manager's account.

---

<sup>1</sup> For a common-environment cluster, you must rename this file to SYS\$COMMON:[SYSMGR]:NETNODE\_UPDATE.COM, as described in Section 2.5.4.

# Building and Maintaining the Cluster

## 3.2 Configuring the Cluster

### 3.2.2 Removing a Node from the Cluster

Before you can remove a node from the cluster, you must shut down the node. If possible, use the command procedure `SYSS$SYSTEM:SHUTDOWN.COM` to perform an orderly shutdown. Otherwise, halt the machine.

Note that because the REMOVE function deletes the node's entire root directory tree, it generates VMS RMS error messages while deleting directory files. You can ignore these messages.

Whenever you remove a voting member from the cluster, you must, after the REMOVE function completes, reconfigure the cluster, following instructions in Section 3.3.

Example 3-4 illustrates the use of `CLUSTER_CONFIG.COM` on node JUPITR to remove satellite node EUROPA from the cluster.

**Note:** If the page and swap files for the node being removed do not reside on the same disk as the node's root directory tree, the REMOVE function does not delete these files. It displays a message warning that the files will not be deleted, as in Example 3-4. If you want to delete the files, you must do so after the REMOVE function completes.

#### Example 3-4 Sample Interactive `CLUSTER_CONFIG.COM` Session to Remove a Satellite Node with Local Page and Swap Files

---

```
$ @CLUSTER_CONFIG.COM
```

```
Cluster Configuration Procedure
```

```
Use CLUSTER_CONFIG.COM to set up or change a VAXcluster configuration.
To ensure that you have the required privileges, invoke this procedure
from the system manager's account.
```

```
Enter ? for help at any prompt.
```

1. ADD a node to the cluster.
2. REMOVE a node from the cluster.
3. CHANGE a cluster node's characteristics.
4. CREATE a second system disk for JUPITR.

```
Enter choice [1]: 2
```

```
The REMOVE function disables a node as a cluster member.
```

- o It deletes the node's root directory tree.
- o It removes the node's network information from the network database.

```
If the node being removed is a voting member, you must adjust
EXPECTED_VOTES in each remaining cluster member's MODPARAMS.DAT.
You must then reconfigure the cluster, using the procedure described
in the VMS VAXcluster Manual.
```

```
What is the node's DECnet node name? EUROPA
Verifying network database...
Verifying that SYS10 is EUROPA's root...
```

```
WARNING - EUROPA's page and swap files will not be deleted.
They do not reside on $1$DJA11:.
```

---

Example 3-4 Cont'd. on next page

# Building and Maintaining the Cluster

## 3.2 Configuring the Cluster

### Example 3–4 (Cont.) Sample Interactive CLUSTER\_CONFIG.COM Session to Remove a Satellite Node with Local Page and Swap Files

---

```
Deleting directory tree SYS10...
%DELETE-I-FILDEL, $1$DJA11:<SYS10>SYSCBI.DIR;1 deleted (1 block)
%DELETE-I-FILDEL, $1$DJA11:<SYS10>SYSERR.DIR;1 deleted (1 block)
.
.
.
System root SYS10 deleted.
Updating network database...
The configuration procedure has completed successfully.
```

---

### 3.2.3 Changing a Node's Characteristics

You select the CHANGE function when you want to accomplish any of the operations described in Table 3–2. When you select this function, CLUSTER\_CONFIG.COM displays a menu of CHANGE options. Note that all operations except changing a satellite's Ethernet hardware address must be executed on the system whose characteristics you want to change (local system).

If you plan to set up a new local area or mixed-interconnect cluster, you *must*, before adding nodes, execute the CHANGE function to enable the first installed node as a boot server (see Example 3–7).

**Caution:** Whenever you enable or disable disk serving functions, you must run AUTOGEN with the REBOOT option to reboot the local system. For all other CHANGE operations (except changing a satellite's hardware address), you must reconfigure the cluster, following instructions in Section 3.3.

Table 3–2 CLUSTER\_CONFIG.COM CHANGE Options

Option	Operation Performed
Enable the local system as a disk server.	Load the MSCP Server by setting, in MODPARAMS.DAT, the value of the MSCP_LOAD parameter to 1, and setting an appropriate value for the MSCP_SERVE_ALL parameter.
Disable the local system as a disk server.	Set MSCP_LOAD to 0.
Enable the local system as a boot server.	If you are setting up a local area or mixed-interconnect cluster, you must execute this operation once before you attempt to add nodes to the cluster. You thereby enable DECnet MOP service for the Ethernet adapter circuit that the node will use to service downline load requests from satellites. When you enable the node as a boot server, it automatically becomes a disk server (if it is not one already), because it must serve its system disk to satellites.
Disable the local system as a boot server.	Disable DECnet MOP service for the node's Ethernet adapter circuit.

# Building and Maintaining the Cluster

## 3.2 Configuring the Cluster

**Table 3–2 (Cont.) CLUSTER\_CONFIG.COM CHANGE Options**

Option	Operation Performed
Enable the Ethernet for cluster communications on the local system.	Load the VAXport driver PEDRIVER by setting the value of the NISCS_LOAD_PEA0 parameter to 1 in MODPARAMS.DAT. Create the cluster security database file, SYS\$SYSTEM:[SYSEXE]CLUSTER_AUTHORIZE.DAT on the local system's system disk.
Disable the Ethernet for cluster communications on the local system.	Set NISCS_LOAD_PEA0 to 0.
Enable a quorum disk on the local system.	Set, in MODPARAMS.DAT, an appropriate value for the SYSGEN parameter DISK_QUORUM; set the value of QDSKVOTES to 1 (default value).
Disable a quorum disk on the local system.	Set, in MODPARAMS.DAT, a blank value for the SYSGEN parameter DISK_QUORUM; set the value of QDSKVOTES to 1.
Change the local system's allocation class value.	Set a value for the node's ALLOCLASS parameter in MODPARAMS.DAT.
Change a satellite's Ethernet hardware address.	Change a satellite's hardware address, in the event that its Ethernet device should need replacement. Both the permanent and volatile network databases, and NETNODE_UPDATE.COM, are updated on the local system. <i>You must execute this operation on any node enabled as a boot server for the satellite.</i>

**Note:** When CLUSTER\_CONFIG.COM sets or changes values in MODPARAMS.DAT, the new values are always appended at the end of the file, so that they override earlier values. You may want to edit the file occasionally and delete lines that specify earlier values.

Examples 3–5 through 3–8 show the use of CLUSTER\_CONFIG.COM to perform the following operations:

- Enable node URANUS as a disk server
- Change node URANUS's ALLOCLASS value
- Enable node URANUS as a boot server
- Specify a new hardware address for satellite node ARIEL, which boots from URANUS's system disk.

# Building and Maintaining the Cluster

## 3.2 Configuring the Cluster

### Example 3-5 Sample Interactive CLUSTER\_CONFIG.COM Session to Enable the Local System as a Disk Server

---

\$ @CLUSTER\_CONFIG.COM

#### Cluster Configuration Procedure

Use CLUSTER\_CONFIG.COM to set up or change a VAXcluster configuration. To ensure that you have the required privileges, invoke this procedure from the system manager's account.

Enter ? for help at any prompt.

1. ADD a node to the cluster.
2. REMOVE a node from the cluster.
3. CHANGE a cluster node's characteristics.
4. CREATE a second system disk for URANUS.

Enter choice [1]: 3

#### CHANGE Menu

1. Enable URANUS as a disk server.
2. Disable URANUS as a disk server.
3. Enable URANUS as a boot server.
4. Disable URANUS as a boot server.
5. Enable Ethernet for cluster communications on URANUS.
6. Disable Ethernet for cluster communications on URANUS.
7. Enable a quorum disk on URANUS.
8. Disable a quorum disk on URANUS.
9. Change URANUS's ALLOCLASS value.
10. Change a satellite's hardware address.

Enter choice [1]:

Will URANUS serve HSC disks [Y]?

Enter a value for URANUS's ALLOCLASS parameter: 2

The configuration procedure has completed successfully.

URANUS has been enabled as a disk server. MSCP\_LOAD has been set to 1 in MODPARAMS.DAT. Please run AUTOGEN to reboot URANUS:

\$ @SYS\$UPDATE:AUTOGEN GETDATA REBOOT

If you have changed URANUS's ALLOCLASS value, you must reconfigure the cluster, using the procedure described in the VMS VAXcluster Manual.

---

# Building and Maintaining the Cluster

## 3.2 Configuring the Cluster

### Example 3–6 Sample Interactive CLUSTER\_CONFIG.COM Session to Change the Local System's ALLOCLASS Value

---

\$ @CLUSTER\_CONFIG.COM

#### Cluster Configuration Procedure

Use CLUSTER\_CONFIG.COM to set up or change a VAXcluster configuration. To ensure that you have the required privileges, invoke this procedure from the system manager's account.

Enter ? for help at any prompt.

1. ADD a node to the cluster.
2. REMOVE a node from the cluster.
3. CHANGE a cluster node's characteristics.
4. CREATE a second system disk for URANUS.

Enter choice [1]: 3

#### CHANGE Menu

1. Enable URANUS as a disk server.
2. Disable URANUS as a disk server.
3. Enable URANUS as a boot server.
4. Disable URANUS as a boot server.
5. Enable Ethernet for cluster communications on URANUS.
6. Disable Ethernet for cluster communications on URANUS.
7. Enable a quorum disk on URANUS.
8. Disable a quorum disk on URANUS.
9. Change URANUS's ALLOCLASS value.
10. Change a satellite's hardware address.

Enter choice [1]: 9

Enter a value for URANUS's ALLOCLASS parameter [2]: 1  
The configuration procedure has completed successfully

If you have changed URANUS's ALLOCLASS value, you must reconfigure the cluster, using the procedure described in the VMS VAXcluster Manual.

---

### Example 3–7 Sample Interactive CLUSTER\_CONFIG.COM Session to Enable the Local System as a Boot Server

---

\$ @CLUSTER\_CONFIG.COM

#### Cluster Configuration Procedure

Use CLUSTER\_CONFIG.COM to set up or change a VAXcluster configuration. To ensure that you have the required privileges, invoke this procedure from the system manager's account.

Enter ? for help at any prompt.

1. ADD a node to the cluster.
2. REMOVE a node from the cluster.
3. CHANGE a cluster node's characteristics.
4. CREATE a second system disk for URANUS.

Enter choice [1]: 3

---

**Example 3–7 Cont'd. on next page**

# Building and Maintaining the Cluster

## 3.2 Configuring the Cluster

### Example 3-7 (Cont.) Sample Interactive CLUSTER\_CONFIG.COM Session to Enable the Local System as a Boot Server

---

```
CHANGE Menu

1. Enable URANUS as a disk server.
2. Disable URANUS as a disk server.
3. Enable URANUS as a boot server.
4. Disable URANUS as a boot server.
5. Enable Ethernet for cluster communications on URANUS.
6. Disable Ethernet for cluster communications on URANUS.
7. Enable a quorum disk on URANUS.
8. Disable a quorum disk on URANUS.
9. Change URANUS's ALLOCLASS value.
10. Change a satellite's hardware address.

Enter choice [1]: 3

Verifying circuits in network database...
Updating permanent network database...

In order to enable or disable DECnet MOP service in the volatile
network database, DECnet traffic must be interrupted temporarily.

Do you want to proceed [Y]? 

Enter a value for URANUS's ALLOCLASS parameter [1]: 
The configuration procedure has completed successfully.

URANUS has been enabled as a boot server. Disk serving and
Ethernet capabilities are enabled automatically. If URANUS was
not previously set up as a disk server, please run AUTOGEN to
reboot URANUS:

$ @SYS$UPDATE:AUTOGEN GETDATA REBOOT

If you have changed URANUS's ALLOCLASS value, you must reconfigure the
cluster, using the procedure described in the VMS VAXcluster Manual.
```

---

### Example 3-8 Sample Interactive CLUSTER\_CONFIG.COM Session to Change a Satellite's Hardware Address

---

```
$ @CLUSTER_CONFIG.COM

Cluster Configuration Procedure

Use CLUSTER_CONFIG.COM to set up or change a VAXcluster configuration.
To ensure that you have the required privileges, invoke this procedure
from the system manager's account.

Enter ? for help at any prompt.

1. ADD a node to the cluster.
2. REMOVE a node from the cluster.
3. CHANGE a cluster node's characteristics.
4. CREATE a second system disk for URANUS.

Enter choice [1]: 3
```

---

Example 3-8 Cont'd. on next page

# Building and Maintaining the Cluster

## 3.2 Configuring the Cluster

### Example 3–8 (Cont.) Sample Interactive CLUSTER\_CONFIG.COM Session to Change a Satellite's Hardware Address

---

CHANGE Menu

1. Enable URANUS as a disk server.
2. Disable URANUS as a disk server.
3. Enable URANUS as a boot server.
4. Disable URANUS as a boot server.
5. Enable Ethernet for cluster communications on URANUS.
6. Disable Ethernet for cluster communications on URANUS.
7. Enable a quorum disk on URANUS.
8. Disable a quorum disk on URANUS.
9. Change URANUS's ALLOCLASS value.
10. Change a satellite's hardware address.

Enter choice [1]: 10

What is the node's DECnet node name? ARIEL

What is the new hardware address [08-00-2B-06-81-44]? 08-00-3B-05-37-78

Updating network database...

The configuration procedure has completed successfully.

---

### 3.2.4 Changing the Cluster Configuration Type

As your processing needs change, you may want to add satellites to an existing CI-only cluster, or you may want to add CI-connected processors or HSCs to an existing local area cluster. In either case, you can use CLUSTER\_CONFIG.COM to convert your existing cluster to a mixed-interconnect configuration.

#### 3.2.4.1 Changing an Existing CI-Only Cluster to a Mixed-Interconnect Configuration

If you want to convert an existing CI-only cluster to a mixed-interconnect configuration, you must enable cluster communications over the Ethernet on all VAX processors, and you must enable one or more processors as boot servers. Proceed as follows:

- 1 Log in as system manager on each VAX processor, invoke CLUSTER\_CONFIG.COM, and execute the CHANGE function to enable the Ethernet for cluster communications. *You must perform this operation on all VAX processors.*
- 2 Execute the CHANGE function to enable one or more processors as boot servers.
- 3 Shut down and reboot the cluster, following instructions in Section 3.3.

# Building and Maintaining the Cluster

## 3.2 Configuring the Cluster

---

### 3.2.4.2 Changing an Existing Local Area Cluster to a Mixed-Interconnect Configuration

Before performing the operations described in this section, be sure that the VAX processors and HSCs you intend to include in your new mixed-interconnect configuration are correctly installed and checked for proper operation.

The method you use to convert an existing local area cluster to a mixed-interconnect configuration depends on whether your current boot server is a CI-capable VAX processor. Note that the following procedures assume that the system disk containing satellite node roots will reside on an HSC.

If the boot server is a CI-capable processor, proceed as follows:

- 1 Log in as system manager on the boot server and perform an image backup operation to back up the current system disk to a disk on an HSC. (For complete information on backup operations, refer to the *VMS Backup Utility Manual*.)
- 2 Modify the system's default bootstrap command procedure to boot the system from the HSC disk, following instructions in the appropriate processor-specific installation and operations guide.
- 3 Shut down the cluster. Shut down the satellites first, then shut down the boot server.
- 4 Boot the boot server from the newly created system disk on the HSC.
- 5 Reboot the satellites.

If your current boot server is not a CI-capable processor, proceed as follows:

- 1 Shut down the old local area cluster. Shut down the satellites first, then shut down the boot server.
- 2 Install the VMS operating system on the new CI-connected VAX processor's HSC system disk. When the installation procedure asks if you want to enable the Ethernet for cluster communications, answer YES.
- 3 When the installation completes, log in as system manager and configure and start the DECnet-VAX network, as described in Chapter 2.
- 4 Execute the CLUSTER\_CONFIG.COM CHANGE function to enable the node as a boot server.
- 5 Log in as system manager on the newly added CI-connected node and execute CLUSTER\_CONFIG.COM's ADD function to add the former local area cluster members (including the former boot server) as satellites on the new HSC system disk.

# Building and Maintaining the Cluster

## 3.2 Configuring the Cluster

### 3.2.5 Converting a Standalone Node to a Cluster Node

You execute CLUSTER\_CONFIG.COM on a standalone node to perform the following operations:

- Add the standalone node to an existing cluster.
- Set up the standalone node to form a new cluster, if the node was not set up as a cluster node during installation of the VMS operating system.

Example 3-9 illustrates the use of CLUSTER\_CONFIG.COM on standalone node PLUTO to convert PLUTO to a cluster boot server.

#### Example 3-9 Sample Interactive CLUSTER\_CONFIG.COM Session to Convert a Standalone Node to a Cluster Boot Server

---

```
$ @CLUSTER_CONFIG.COM

Cluster Configuration Procedure

This procedure sets up this standalone node to join an existing
cluster or to form a new cluster.

What is the node's DECnet node name? PLUTO
What is the node's DECnet address? 2.5
Will the Ethernet be used for cluster communications (Y/N)? Y
Enter this cluster's group number: 3378
Enter this cluster's password:
Re-enter this cluster's password for verification:
Will PLUTO be a boot server [Y]? 
Verifying circuits in network database...
Enter a value for PLUTO's ALLOCLASS parameter: 1
Does this cluster contain a quorum disk [N]? 

AUTOGEN computes the SYSGEN parameters for your configuration
and then reboots the system with the new parameters.
```

---

### 3.2.6 Creating a Duplicate System Disk

To duplicate a cluster system disk, proceed as follows, *after* you have coordinated cluster common files, as described in Section 2.5.4.

- 1 Log in as system manager.
- 2 Place a blank disk in an appropriate drive and spin up the disk.
- 3 Invoke CLUSTER\_CONFIG.COM and select the CREATE function. The procedure will prompt you for the device names of the current and new system disks. It will then back up the current system disk to the new one, delete all directory roots from the new disk, and mount that disk clusterwide. Note that you will see VMS RMS error messages while the procedure deletes directory files. You can ignore these messages.

Example 3-10 shows a typical interactive CREATE session on node JUPITR.

# Building and Maintaining the Cluster

## 3.2 Configuring the Cluster

### Example 3-10 Sample Interactive CLUSTER\_CONFIG.COM CREATE Session

---

\$ @CLUSTER\_CONFIG.COM

#### Cluster Configuration Procedure

Use CLUSTER\_CONFIG.COM to set up or change a VAXcluster configuration. To ensure that you have the required privileges, invoke this procedure from the system manager's account.

Enter ? for help at any prompt.

1. ADD a node to the cluster.
2. REMOVE a node from the cluster.
3. CHANGE a cluster node's characteristics.
4. CREATE a second system disk for JUPITR.

Enter choice [1]: 4

The CREATE function generates a duplicate system disk.

- o It backs up the current system disk to the new system disk.
- o It then removes from the new system disk all system roots.

WARNING - Do not proceed unless you have defined appropriate logical names for cluster common files in your site-specific startup procedures. For instructions, see the VMS VAXcluster Manual.

Do you want to continue [N]? YES

This procedure will now ask you for the device name of JUPITR's system root. The default device name (DISK\$VAXVMSRL5:) is the logical volume name of SYS\$SYSDEVICE:.

What is the device name of the current system disk [DISK\$VAXVMSRL5:]?

What is the device name for the new system disk? \$1\$DJA16:

%DCL-I-ALLOC, \_\$1\$DJA16: allocated

%MOUNT-I-MOUNTED, SCRATCH mounted on \_\$1\$DJA16:

What is the unique label for the new system disk [JUPITR\_SYS2]?

Backing up the current system disk to the new system disk...

Deleting all system roots...

Deleting directory tree SYS1...

%DELETE-I-FILDEL, \$1\$DJA16:<SYSO>DECNET.DIR;1 deleted (2 blocks)

.  
.  
.

System root SYS1 deleted.

Deleting directory tree SYS2...

%DELETE-I-FILDEL, \$1\$DJA16:<SYS1>DECNET.DIR;1 deleted (2 blocks)

.  
.  
.

System root SYS2 deleted.

All the roots have been deleted.

%MOUNT-I-MOUNTED, JUPITR\_SYS2 mounted on \_\$1\$DJA16:

The second system disk has been created and mounted clusterwide. Satellites can now be added.

---

# Building and Maintaining the Cluster

## 3.2 Configuring the Cluster

---

### 3.3 Reconfiguring the Cluster after a Major Change

Because the following operations affect the integrity of the entire cluster, you must reconfigure the cluster after executing any of them.

- Adding or removing a voting cluster member
- Enabling or disabling the Ethernet for cluster communications
- Enabling or disabling a quorum disk
- Changing allocation class values
- Changing the cluster group number or password (see Section 3.4.6)

In all cases, you must shut down and reboot the entire cluster. Note that if you add or remove a voting member, or if you enable or disable a quorum disk, you must update MODPARAMS.DAT files before shutting down the cluster. To perform these reconfiguration tasks, follow instructions in Sections 3.3.1 through 3.3.4.

---

#### 3.3.1 Updating MODPARAMS.DAT Files to Adjust Cluster Quorum

Whenever you add or remove a voting cluster node, or whenever you enable or disable a quorum disk, you must edit MODPARAMS.DAT in all other cluster members' [SYSn.SYSEXE] directories and adjust the value for the SYSGEN parameter EXPECTED\_VOTES appropriately. For example, if you add a voting node, or if you enable a quorum disk, you must increment the value by the number of votes assigned to the new member (usually 1). If you add a voting node with 1 vote and enable a quorum disk with 1 vote on that node, you must increment the value by 2.

You must then prepare to shut down and reboot the entire cluster. To ensure that the new values take effect when you reboot, log in on each node as system manager and run AUTOGEN to propagate the values to the node's VAXVMSSYS.PAR file. Enter the following command:

```
$ @SYS$UPDATE:AUTOGEN GETDATA SETPARAMS
```

Be sure *not* to specify the SHUTDOWN or REBOOT options.

**Caution:** Do not perform this operation until you are ready to shut down and reboot the entire cluster. If a node should fail or crash, and then reboot with the new parameters, normal cluster operations can be seriously compromised.

# Building and Maintaining the Cluster

## 3.3 Reconfiguring the Cluster after a Major Change

---

### 3.3.2 Shutting Down the Cluster

After you have run AUTOGEN to set parameter values correctly, you must shut down the entire cluster. Log in as system manager on each node *locally* and enter the following command to perform an orderly shutdown:

```
§ @SYS$SYSTEM:SHUTDOWN
```

When you are prompted for the shutdown options, specify CLUSTER for cluster shutdown. Note that you must run the shutdown procedure and specify this option on each node. You cannot shut down the entire cluster from one node.

---

### 3.3.3 Changing Allocation Class Values on HSCs

If it is necessary to change allocation class values on any HSC controller, you must do so while the entire cluster is shut down. Enter a command sequence like the following at the appropriate HSC consoles:

```
CTRL/C  
HSC> RUN SETSHO  
SETSHO> SET ALLOCATE DISK 1  
SETSHO> EXIT  
SETSHO-Q Rebooting HSC; Y to continue, CTRL/Y to abort:? Y
```

---

### 3.3.4 Rebooting the Cluster

After all HSCs have been set and rebooted, reboot each cluster node. Watch the console listings for unusual messages or warnings.

**Caution:** In local area and mixed-interconnect clusters, you must reboot boot servers before rebooting satellites.

Note that several new messages may appear. For example, if you have used the CLUSTER\_CONFIG.COM CHANGE function to enable cluster communications over the Ethernet, one message will report that the Local Area VAXcluster security database is being loaded. Then, for every disk-serving node, you will see a message reporting that the MSCP Server is being loaded, followed by a list of all the disks being served by that node. You should verify that all disks are being served in the manner that you specified when you designed the configuration.

---

## 3.4 Maintaining the Cluster

Once your cluster is up and running, you can implement routine site-specific maintenance operations—for example, backing up disks or adding user accounts. And you should plan to run AUTOGEN with the FEEDBACK option on a regular basis, as described in Section 3.4.1.

You should also maintain records of current configuration data, especially any changes to hardware or software components. Section 3.4.2 lists items that should be included in your records.

If you are managing a local area or mixed-interconnect cluster, it is important to monitor Ethernet activity. Section 3.4.3 provides information to help you set up a monitoring procedure.

# Building and Maintaining the Cluster

## 3.4 Maintaining the Cluster

From time to time conditions may occur that require the following special maintenance operations:

- Restoring cluster quorum after an unexpected node failure
- Executing conditional shutdown operations
- Performing security functions in local area and mixed-interconnect clusters

These operations are discussed in Sections 3.4.4, 3.4.5, and 3.4.6.

### 3.4.1 Running AUTOGEN with the FEEDBACK Option

In VMS Version 5.0, AUTOGEN has been enhanced with a mechanism called *feedback*. This new mechanism examines data collected during normal system operation, and it adjusts system parameters on the basis of the collected data whenever you run AUTOGEN with the FEEDBACK option.

DIGITAL strongly recommends that you use the new feedback mechanism. Without feedback, it is difficult for AUTOGEN to anticipate patterns of resource usage, particularly in complex configurations. Factors such as the number of nodes and disks in the cluster, and the types of applications being run, require adjustment of system parameters for optimal performance.

You should therefore run AUTOGEN with feedback frequently. As a cluster grows, settings for many parameters must be adjusted. The settings AUTOGEN chooses for a cluster with 3 CI-connected VAX processors and 5 satellites will no longer be appropriate when you add more processors or satellites. In summary, you should rerun AUTOGEN whenever you make significant changes in your configuration. For detailed information on AUTOGEN, refer to the *Guide to Setting Up a VMS System*.

### 3.4.2 Recording Configuration Data

Effective maintenance of a VAXcluster configuration requires that you keep accurate records on the current status of all hardware and software components and on any changes made to those components. Changes to cluster components can have a significant effect on the operation of the entire cluster. And if a failure should occur, you will need to consult your records when diagnosing problems.

At a minimum, your configuration records should include the following:

- SCSNODE and SYSSYSTEMID parameter values for all cluster nodes.
- DECnet names and addresses for all cluster nodes.
- Current values for cluster-related SYSGEN parameters, especially ALLOCLASS values for HSCs and VAX processors. (Cluster SYSGEN parameters are described in Appendix A.)
- Default bootstrap command procedures for all CI-connected nodes.
- Names of Ethernet adapter circuits.
- Names of cluster disk and tape devices.
- In local area and mixed-interconnect clusters, Ethernet hardware addresses for satellite nodes.

# Building and Maintaining the Cluster

## 3.4 Maintaining the Cluster

- Serial numbers of all hardware components.
- Changes to any hardware or software components (including site-specific command procedures) along with dates and times when changes were made.

Maintaining current records for your configuration is necessary both for routine operations and for eventual troubleshooting activities.

---

### 3.4.3 Monitoring Ethernet Activity in Local Area and Mixed-Interconnect Clusters

In local area and mixed-interconnect clusters it is important that you monitor Ethernet activity on a regular basis. Using NCP commands like those shown in the accompanying example, (where BNA-0 is the line-id of the Ethernet line), you can set up a convenient monitoring procedure to report activity for each 12-hour period. Note that DECnet event logging for event 0.2 (automatic line counters) must be enabled. (For detailed information on DECnet-VAX event logging, refer to the *VMS Network Control Program Manual*.)

```
NCP> DEFINE LINE BNA-0 COUNTER TIMER 43200  
NCP> SET LINE BNA-0 COUNTER TIMER 43200
```

Every timer interval (in this case 12 hours) DECnet will create an event that sends counter data to the DECnet event log. If you experience a performance degradation in your cluster, check the event log for increases in counter values that exceed normal variations for your cluster. If all nodes show the same increase, there may be a general problem with your Ethernet configuration. If, on the other hand, only one node shows a deviation from usual values, there is probably a problem with that node or its Ethernet interface device.

---

### 3.4.4 Restoring Cluster Quorum after an Unexpected Node Failure

During the life of a cluster, nodes join and leave the cluster. For example, you may need to add more processors to the cluster to extend the cluster's processing capabilities, or a node may shut down unexpectedly as the result of a hardware or fatal software error. The connection management software coordinates these cluster transitions and controls cluster operation.

When a cluster node shuts down unexpectedly, the remaining nodes, with the help of the Connection Manager, reconfigure the cluster, excluding the node that shut down. The cluster will survive the failure of the node and continue to process, as long as the cluster votes total is greater than the cluster quorum value. If the cluster votes total falls below the cluster quorum value, the cluster suspends the execution of all processes.

For process execution to resume, the cluster votes total must be restored to a value greater than or equal to the cluster quorum value. Often, the required votes are added as nodes join or rejoin the cluster. However, waiting for a node to join the cluster and raising the votes value is not always a simple or convenient remedy. An alternative solution, for example, might be to shut down and reboot all the nodes with a lower quorum value. In any case, it is important to be aware of cluster state changes in order to prevent potential problems.

# Building and Maintaining the Cluster

## 3.4 Maintaining the Cluster

Following the failure of a node, you may want to run the Show Cluster Utility and examine values for the VOTES, EXPECTED\_VOTES, CL\_VOTES, and CL\_QUORUM fields. (See the *VMS Show Cluster Utility Manual* for a complete description of these fields.) The VOTES and EXPECTED\_VOTES fields show the settings for each cluster member; the CL\_VOTES and CL\_QUORUM fields show the cluster votes total and the current cluster quorum value.

To examine these values, enter the following commands:

```
$ SHOW CLUSTER/CONTINUOUS  
COMMAND> ADD VOTES,EXPECTED_VOTES,CL_VOTES,CL_QUORUM
```

**Note:** If you want to enter **SHOW CLUSTER** commands interactively, you must specify the **/CONTINUOUS** qualifier as part of the **SHOW CLUSTER** command string. If you do not specify this qualifier, **SHOW CLUSTER** will display cluster status information returned by the DCL command **SHOW CLUSTER** and will return you to the DCL command level.

If the display from the Show Cluster Utility shows the CL\_VOTES value equal to the CL\_QUORUM value, the cluster will not survive the failure of any remaining voting node. If one of these nodes shuts down, all process activity in the cluster will stop.

To prevent the disruption of cluster process activity, you can lower the cluster quorum value. You can use the DCL command **SET CLUSTER/EXPECTED\_VOTES** to adjust the cluster quorum to a value you specify. If you do not specify a value, the system calculates an appropriate value for you. You need enter the command on only one node to propagate the new value throughout the cluster. When you enter the command, the system reports the new value.

Note that you normally use the **SET CLUSTER/EXPECTED\_VOTES** command only when a node is leaving the cluster for an extended period. (For more information on this command, see the *VMS DCL Dictionary*.)

If, for example, you want to change expected votes to set the cluster quorum to 2, enter the following command:

```
$ SET CLUSTER/EXPECTED_VOTES=3
```

The resulting value is  $(3 + 2)/2 = 2$ .

Note that no matter what value you specify for the **SET CLUSTER/EXPECTED\_VOTES** command, you cannot increase quorum to a value that is greater than the number of the votes present, nor can you reduce quorum to a value that is half or fewer of the votes present.

To make the new value active clusterwide, you must adjust the **SYSGEN** parameter **EXPECTED\_VOTES** in **MODPARAMS.DAT** files on each cluster node, and then reconfigure the cluster, following instructions in Section 3.3.

When a node that was previously a cluster member is ready to rejoin, you must reset the **SYSGEN** parameter **EXPECTED\_VOTES** to its original value in **MODPARAMS.DAT** on all nodes and then reconfigure the cluster, following instructions in Section 3.3. You do not need to use the **SET CLUSTER/EXPECTED\_VOTES** command to increase cluster quorum, because the quorum value will be increased automatically when the node rejoins the cluster.

You can also reduce cluster quorum by selecting one of the cluster-related shutdown options described in Section 3.4.5.

# Building and Maintaining the Cluster

## 3.4 Maintaining the Cluster

### 3.4.5 Selecting Cluster Shutdown Options

The VMS operating system provides four options for shutting down cluster nodes:

- REMOVE\_NODE
- CLUSTER\_SHUTDOWN
- REBOOT\_CHECK
- SAVE\_FEEDBACK

Sections 3.4.5.1 through 3.4.5.4 explain these options.

If you do not select any option (if you select the default SHUTDOWN option NONE) the SHUTDOWN procedure will default to the normal behavior for shutting down a standalone system. If you want to shut down a node that you expect to rejoin the cluster shortly, you can select the default option. In that case, cluster quorum will not be adjusted, because it is assumed that the node will soon rejoin the cluster.

#### 3.4.5.1 The REMOVE\_NODE Option

If you want to shut down a cluster node that you expect will not be rejoining the cluster for an extended period, select the REMOVE\_NODE option. For example, a node may be waiting for new hardware, or you may decide that you want to use a node standalone indefinitely.

When you use the REMOVE\_NODE option, the active quorum in the remainder of the cluster will be adjusted downward to reflect the fact that the removed node's votes will no longer be contributing to the quorum value. The SHUTDOWN procedure readjusts the quorum by issuing the SET CLUSTER/EXPECTED\_VOTES command, which is subject to the usual constraints described in Section 5.4.

Note that it is still the responsibility of the system manager to change the SYSGEN parameter EXPECTED\_VOTES on the remaining nodes, to reflect the new configuration.

#### 3.4.5.2 The CLUSTER\_SHUTDOWN Option

If you want to shut down the entire cluster, select the CLUSTER\_SHUTDOWN option. When you select this option, the node will suspend activity, just short of shutting down completely, until all nodes in the cluster have reached the same point in the SHUTDOWN procedure. When this condition occurs, all nodes shut down together.

Note that when you select the CLUSTER\_SHUTDOWN option to perform a clusterwide shutdown operation, you must still shut down **each** node in the cluster by invoking the SHUTDOWN.COM procedure at each node's console. If any one node in the cluster is not completely shut down, clusterwide shutdown cannot occur. Instead, operations on all other nodes in the cluster are suspended.

# Building and Maintaining the Cluster

## 3.4 Maintaining the Cluster

---

### 3.4.5.3 The REBOOT\_CHECK Option

When you select the REBOOT\_CHECK option, the SHUTDOWN procedure checks for the existence of basic system files that are needed to reboot the system successfully and notifies you if any files are missing. You should replace such files before proceeding. If all files are present, the following success message appears:

```
%SHUTDOWN-I-CHECKOK, Basic reboot consistency check completed.
```

Note that you can select the REBOOT\_CHECK option separately or in conjunction with either the REMOVE\_NODE or CLUSTER\_SHUTDOWN option. If you select REBOOT\_CHECK with one of the other options, be sure to separate the option list with a comma.

---

### 3.4.5.4 The SAVE\_FEEDBACK Option

You select the SAVE\_FEEDBACK option to enable AUTOGEN feedback operation. Note that you should select this option only when your system has been running long enough to reflect your typical workload. For detailed information on AUTOGEN feedback, see the *Guide to Setting Up a VMS System*.

---

## 3.4.6 Performing Security Functions in Local Area and Mixed-Interconnect Clusters

Because multiple local area and mixed-interconnect clusters may coexist on a single Ethernet, mechanisms are provided to ensure the integrity of individual clusters and to prevent access to a cluster (accidental or deliberate) by an unauthorized node.

Cluster security mechanisms prevent problems that could otherwise occur under circumstances like the following:

- When setting up a new cluster, the system manager specifies a group number identical to that of an existing cluster on the same Ethernet. (This condition is not as unlikely as it may at first appear, because system managers will probably not assign group numbers randomly.) However, provided each cluster's password is unique, the new cluster will form independently.
- A satellite node user with access to a local system disk tries to join a cluster by executing a conversational SYSBOOT operation at the satellite's console.

The following mechanisms are designed to help system managers perform security functions:

- A cluster authorization file (SYS\$COMMON:[SYSEXE]CLUSTER\_AUTHORIZE.DAT), initialized during installation of the VMS operating system or during execution of the CLUSTER\_CONFIG.COM CHANGE function. The file is maintained with the SYSMAN Utility.
- Control of conversational bootstrap operations on satellite nodes.

These mechanisms are discussed in Sections 3.4.6.1 and 3.4.6.2.

# Building and Maintaining the Cluster

## 3.4 Maintaining the Cluster

### 3.4.6.1 Maintaining Cluster Security Data

Security data is maintained in the cluster authorization file, `SYS$COMMON:[SYSEXE]CLUSTER_AUTHORIZE.DAT`, which contains the cluster group number and (in encrypted form) the cluster password. The file is accessible only to users with the `SYS$PRV` privilege.

Under normal conditions, you need not alter records in the `CLUSTER_AUTHORIZE.DAT` file interactively. If, however, you suspect a security breach, you may want to change the cluster password. In that case, you use the `SYS$MAN` Utility to make the change.

Note that if your configuration has multiple system disks, each disk must have a copy of `CLUSTER_AUTHORIZE.DAT`. You must run the utility to update all copies.

**Caution:** If you change either the group number or password, you must reboot the entire cluster. For instructions, see Section 3.3.

To invoke the `SYS$MAN` Utility, log in as system manager on a boot server and enter the following command:

```
$ RUN SYS$SYSTEM:SYS$MAN
SYS$MAN>
```

When the utility responds with the `SYS$MAN>` prompt, you can enter any of the `CONFIGURATION` commands listed in Table 3-3.

**Table 3-3 Summary of `SYS$MAN CONFIGURATION` Commands for Cluster Authorization**

Command	Qualifiers	Function
<code>HELP CONFIGURATION SET CLUSTER_AUTHORIZATION</code>	None	Explains the command's functions.
<code>CONFIGURATION SET CLUSTER_AUTHORIZATION</code>	<code>/GROUP_NUMBER</code>  <code>/PASSWORD</code>	Updates the cluster authorization file, <code>SYS\$COMMON:[SYSEXE]CLUSTER_AUTHORIZE.DAT</code> . (The <code>SET</code> command will create this file if it does not already exist.)  Specifies a cluster group number. Group number must be in the range from 1 to 4095 or 61440 to 65535.  Specifies a cluster password. Password may be from 1 to 31 characters in length and may include alphanumeric characters, dollar signs, and underscores.
<code>CONFIGURATION SHOW CLUSTER_AUTHORIZATION</code>	None	Displays the cluster group number.

Example 3-11 illustrates the use of the `SYS$MAN` Utility to change the cluster password.

# Building and Maintaining the Cluster

## 3.4 Maintaining the Cluster

### Example 3–11 Sample Interactive SYSMAN CONFIGURATION Session

---

```
$ RUN SYS$SYSTEM:SYSMAN
SYSMAN> SET ENVIRONMENT/CLUSTER
%SYSMAN-I-ENV, current command environment:
Clusterwide on local cluster
Username LAZRUS      will be used on nonlocal nodes
SYSMAN> SET PROFILE/PRIVILEGES=SYSPRV
SYSMAN> CONFIGURATION SET CLUSTER_AUTHORIZATION/PASSWORD=newpassword
%SYSMAN-I-CAFOLDGROUP, existing group will not be changed
%SYSMAN-I-CAFREBOOT, cluster authorization file updated
The entire cluster should be rebooted.
SYSMAN> EXIT
$
```

---

#### 3.4.6.2 Controlling Conversational Bootstrap Operations for Satellites

When you add a satellite node to the cluster using CLUSTER\_CONFIG.COM, the procedure asks whether you want to allow conversational bootstrap operations for the satellite (default is NO). If you press RETURN, SYSGEN parameter NISCS\_CONV\_BOOT in the satellite's SYSGEN parameter file remains set to 0 to disable such operations. The parameter file, VAXVMSSYS.PAR, resides in the satellite's root directory on a boot node's system disk (device:[SYSx.SYSEXE]). You may later enable conversational bootstrap operations for a given satellite at any time by setting this parameter to 1.

For example, to enable such operations for a satellite booted from root 10 on device \$1\$DJA11, you would proceed as follows:

- 1 Log in as system manager on the boot server.
- 2 Invoke the System Generation Utility (SYSGEN) and enter the following commands:

```
$ RUN SYS$SYSTEM:SYSGEN
SYSGEN> USE $1$DJA11:[SYS10.SYSEXE]VAXVMSSYS.PAR
SYSGEN> SET NISCS_CONV_BOOT 1
SYSGEN> WRITE $1$DJA11:[SYS10.SYSEXE]VAXVMSSYS.PAR
SYSGEN> EXIT
$
```



---

## 4 Setting Up and Managing Cluster Queues

On a standalone system, print and batch job processing is limited to a single processor and local devices. In VAXcluster configurations, however, nodes can share device and processing resources. This ability to share resources allows for better workload balancing because batch and print job processing can be distributed across the cluster.

You control how jobs share device and processing resources in a cluster by setting up and maintaining cluster queues. The strategy you use to set up and manage these queues will determine how well you match workloads to your cluster's device and processor resources.

You establish and control cluster queues with the same commands you use to manage queues on a standalone VMS system. These commands are described in the *VMS DCL Dictionary*. The sections that follow describe how to set up cluster queues. The chapter assumes some knowledge of queue management on a standalone system, as described in the *Guide to Setting Up a VMS System*.

---

### 4.1 Clusterwide Queues

Clusterwide queues are controlled by a clusterwide job controller queue file. This file makes queues available across the cluster and enables jobs to execute on any queue from any node, provided that the necessary mass storage volumes can be accessed by the node on which the job executes.

There can be only one job controller queue file on a cluster. If there is such a queue file, it must be on a disk that is accessible to the nodes participating in the clusterwide queue scheme.

You control which nodes in the cluster share clusterwide queues by specifying the location of the job controller queue file, JBCSYSQUE.DAT, with the DCL command START/QUEUE/MANAGER. You could use the following command string, for example, to set up a clusterwide queue:

```
$ START/QUEUE/MANAGER SYS$COMMON:[SYSEXE] JBCSYSQUE.DAT
```

All nodes using queues must specify the same queue file in the START/QUEUE/MANAGER command.

---

### 4.2 Cluster Printer Queues

To establish printer queues, you should first decide on the type of queue configuration that will best suit your system. On a cluster, you have several alternatives that depend on the number and type of print devices you have on each node, and how you want print jobs to be processed. For example, make these decisions:

- Whether to set up generic printer queues that are local to each node
- Which printer queues should be assigned to any local generic queues

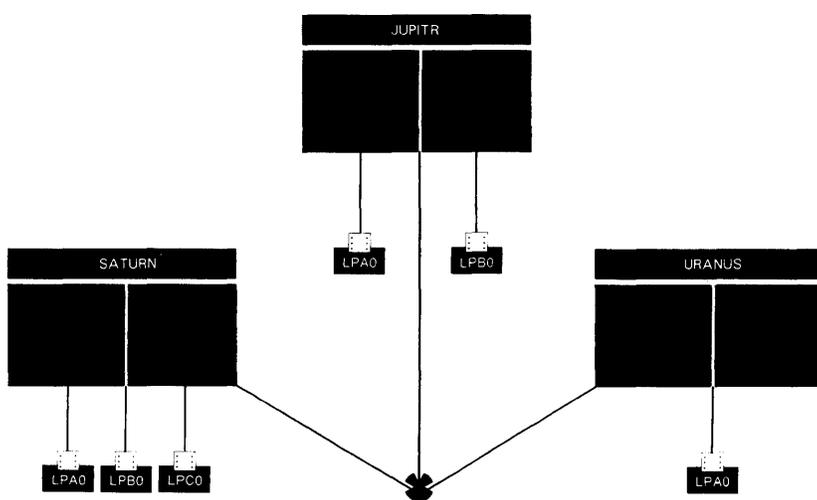
# Setting Up and Managing Cluster Queues

## 4.2 Cluster Printer Queues

- Whether to set up any clusterwide generic queues that will distribute print job processing across the cluster

Once you determine the strategy for your system, you can create a command procedure that will set up your queues. Figure 4-1 shows the printer configuration for a cluster consisting of the active nodes JUPITR, SATURN, and URANUS. The sections that follow will use this example configuration to illustrate various methods for establishing and naming cluster printer queues. Sample command procedures are also included in Section 4.4 to serve as a guide to setting up queues.

**Figure 4-1 Sample Printer Configuration**



ZK-1631-84

### 4.2.1 Setting Up Printer Queues

You should set up printer queues using the same procedures that you would use for a single-node system (see the *Guide to Setting Up a VMS System*). However, since each local node is part of the cluster system, you must provide a unique name for each queue you create in a cluster.

You assign a unique name to a printer queue by specifying the DCL command INITIALIZE/QUEUE in the following format:

```
INITIALIZE/QUEUE/ON=node::device queue-name
```

The /ON qualifier specifies the node and printer that the queue is assigned to.

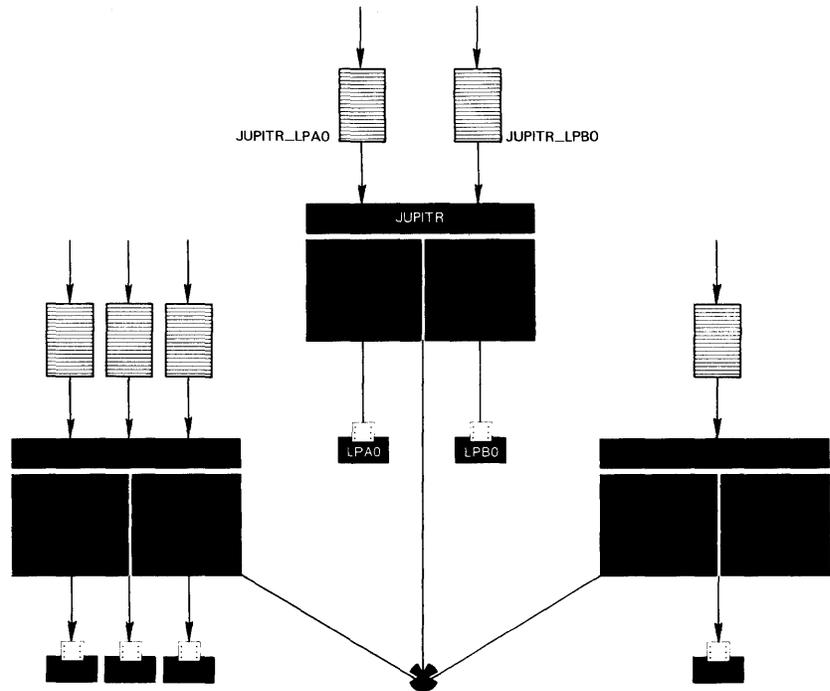
The commands in the following example make local printer queue assignments for the cluster node JUPITR shown in Figure 4-2:

```
$ INITIALIZE/QUEUE/ON=JUPITR::LPA0/START JUPITR_LPA0  
$ INITIALIZE/QUEUE/ON=JUPITR::LPB0/START JUPITR_LPBO
```

# Setting Up and Managing Cluster Queues

## 4.2 Cluster Printer Queues

Figure 4–2 Printer Queue Configuration



ZK-1632-84

### 4.2.2 Setting Up Clusterwide Generic Printer Queues

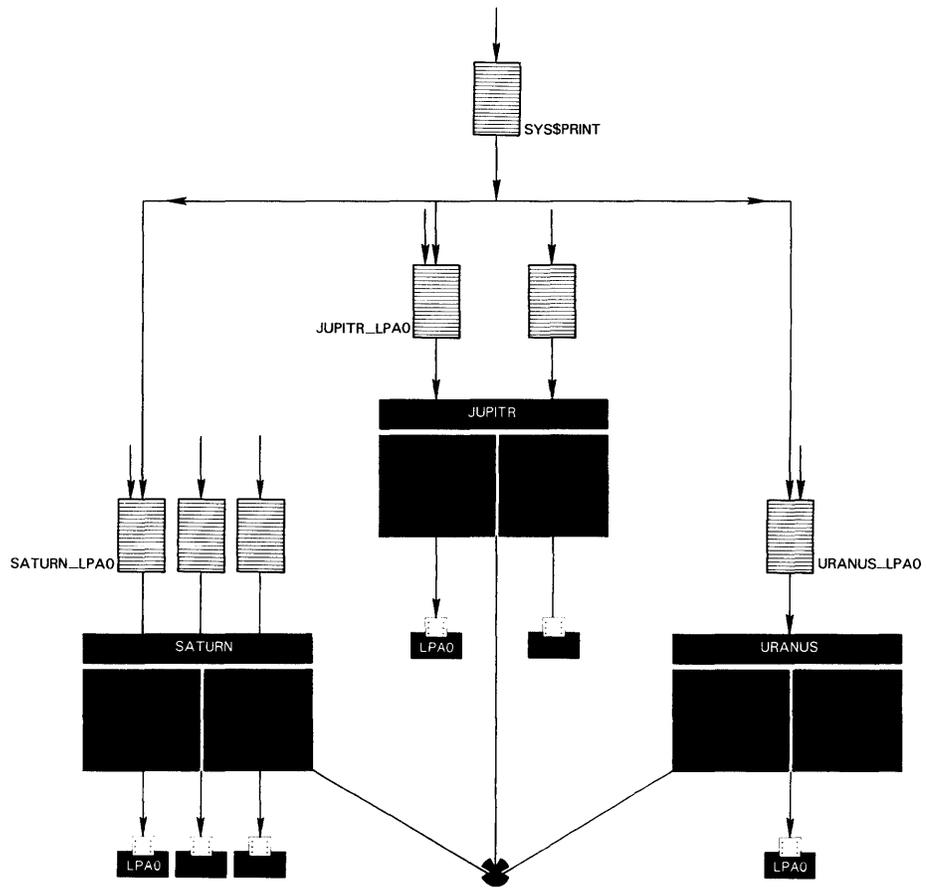
The clusterwide job controller queue file enables you to establish generic queues that function throughout the cluster. Jobs queued to clusterwide generic queues are placed in any assigned printer queue that is available, regardless of its location in the cluster. However, the file queued for printing must be accessible to the node to which the printer is connected.

Figure 4–3 illustrates a clusterwide generic printer queue, in which the queues for all LPA0 printers in the cluster are assigned to a clusterwide generic queue named SYS\$PRINT.

# Setting Up and Managing Cluster Queues

## 4.2 Cluster Printer Queues

Figure 4–3 Cluster Printer Queue Configuration With Clusterwide Generic Printer Queue



ZK-1634-84

The following command initializes and starts the clusterwide generic queue SYS\$PRINT:

```
$ INITIALIZE/QUEUE/GENERIC=(JUPITR_LPA0,SATURN_LPA0,-  
URANUS_LPA0)/START SYS$PRINT
```

Jobs queued to SYS\$PRINT are placed in whichever assigned printer queue is available. Thus, in this example, a print job from node JUPITR that is queued to SYS\$PRINT may in fact be queued to JUPITR\_LPA0, SATURN\_LPA0, or URANUS\_LPA0.

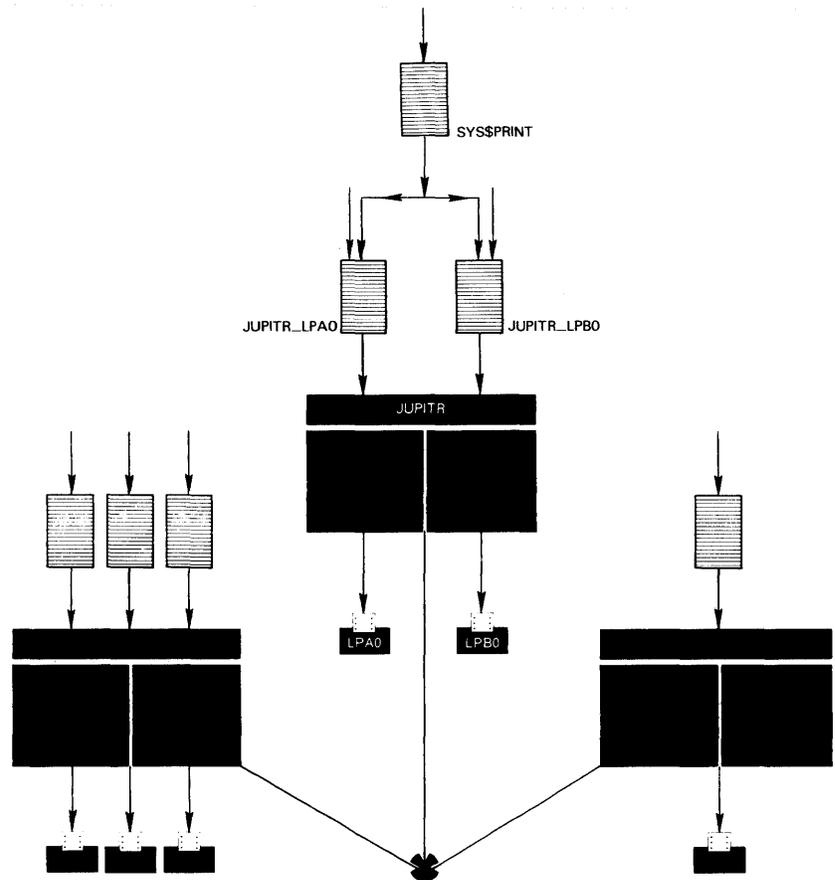
In addition to creating a queue for each local printer, you may want to establish at least one local generic queue for similar devices on the local node. The following commands set up the local generic queue for node JUPITR shown in Figure 4–4.

```
$ INITIALIZE/QUEUE/GENERIC=(JUPITR_LPA0,JUPITR_LPBO)/START JUPITR_PRINT  
$ DEFINE/SYSTEM SYS$PRINT JUPITR_PRINT
```

# Setting Up and Managing Cluster Queues

## 4.2 Cluster Printer Queues

Figure 4–4 Printer Queue Configuration With Local Generic Queue



ZK-1633-84

In Figure 4–4 the generic printer queue JUPITR\_PRINT is set up and explicitly assigned the printer queues JUPITR\_LPA0 and JUPITR\_LPBO.

In a single-node environment, you would name the generic queue SYS\$PRINT, because print jobs are queued to SYS\$PRINT by default. In a cluster, however, the separate nodes cannot have independent queues with the same name; therefore, you cannot create multiple generic queues named SYS\$PRINT. To get around this problem, you can create a generic queue, assign it a unique queue name, and then establish a systemwide logical name equating SYS\$PRINT to the generic queue name. This logical name assignment is systemwide on the local node, affecting operations on that node. Thus, only print jobs from users on JUPITR are queued to JUPITR\_PRINT by default.

Because print jobs on each cluster node are queued to SYS\$PRINT by default, you might want to establish SYS\$PRINT as a clusterwide generic printer queue that distributes print job processing throughout the cluster.

# Setting Up and Managing Cluster Queues

## 4.3 Cluster Batch Queues

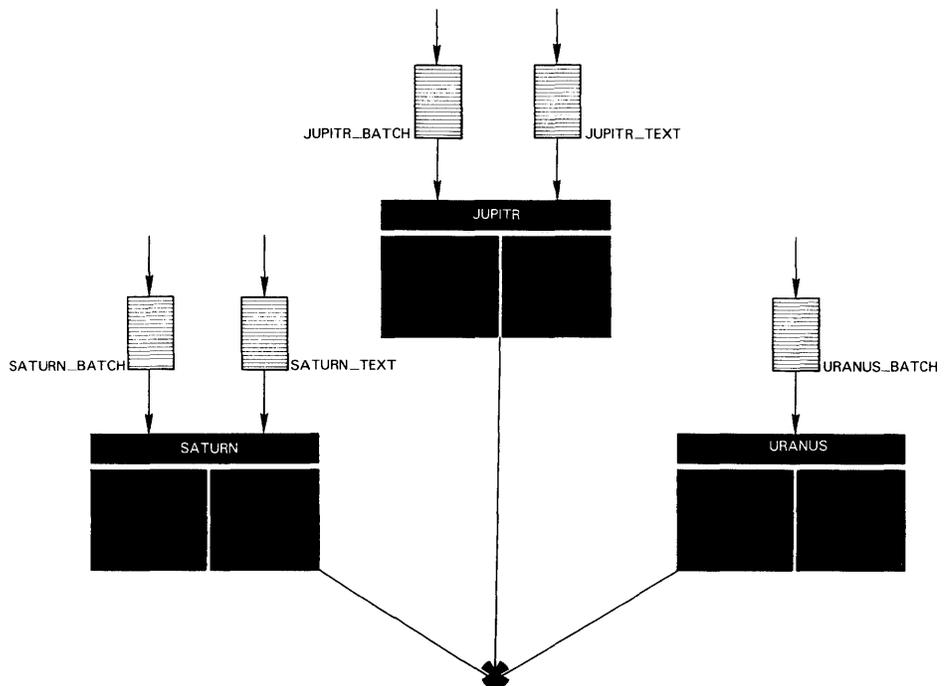
### 4.3 Cluster Batch Queues

Before you establish batch queues, you should first decide on the type of queue configuration that will best suit your cluster. As system manager, you are responsible for setting up batch queues to maintain efficient batch job processing on the cluster. For example, you should do the following:

- Determine what type of processing will be performed on each node
- Set up local batch queues that conform to these processing needs
- Decide whether to set up any clusterwide generic queues that will distribute batch job processing across the cluster

Once you determine the strategy that best suits your system needs, you can create a command procedure that will set up your queues. Figure 4–5 shows the batch queue configuration for a cluster consisting of the active nodes JUPITER, SATURN, and URANUS. The sections that follow will use this example configuration to illustrate various methods for establishing and naming cluster batch queues. Sample command procedures for this configuration are also included in Section 4.4 to serve as a guide to setting up queues.

**Figure 4–5 Sample Batch Queue Configuration**



ZK-1635-84

# Setting Up and Managing Cluster Queues

## 4.3 Cluster Batch Queues

### 4.3.1 Setting Up Executor Batch Queues

Generally, you set up executor batch queues on each cluster node using the same procedures you use for a single-node system. For more detailed information on how this is done, see the *Guide to Setting Up a VMS System*.

You assign a unique name to a batch queue by specifying the DCL command INITIALIZE/QUEUE in the following format:

```
INITIALIZE/QUEUE/ON=node:: queue-name
```

The /ON qualifier specifies the node on which the batch queue runs.

The commands in the following example make local batch queue assignments for the cluster node JUPITR shown in Figure 4-5:

```
$ INITIALIZE/QUEUE/BATCH/ON=JUPITR::/START JUPITR_BATCH  
$ INITIALIZE/QUEUE/BATCH/ON=JUPITR::/START JUPITR_TEXT
```

In a single-node environment, you would name one batch queue SYS\$BATCH, because batch jobs are queued to SYS\$BATCH by default. You may decide to follow this convention for each node in the cluster. In a cluster, however, the separate nodes cannot have independent queues with the same name; therefore you cannot create a queue named SYS\$BATCH for each node in the cluster. To get around this problem, you can create a queue, assign it a unique queue name, and then establish a systemwide logical name equating SYS\$BATCH to the queue name as follows:

```
$ INITIALIZE/QUEUE/BATCH/ON=JUPITR::/START JUPITR_BATCH  
$ DEFINE/SYSTEM SYS$BATCH JUPITR_BATCH
```

This logical name definition is systemwide on the local node, affecting only operations on that node. Thus, only batch jobs from users on JUPITR are queued to JUPITR\_BATCH by default.

Because batch jobs on each cluster node are queued to SYS\$BATCH by default, you should consider establishing SYS\$BATCH as a clusterwide generic batch queue that distributes batch job processing throughout the cluster. Note, however, that you should do this only if you have a common-environment cluster. Guidelines for establishing clusterwide generic batch queues are presented in the following section.

### 4.3.2 Setting Up Generic Batch Queues

Unlike a printer queue, a batch queue can be set up to allow more than one job to execute simultaneously. For this reason it is often not necessary on a single-node system to create multiple batch queues of the same type and assign them to a generic batch queue.

On a cluster, however, where you have multiple processors, you may want to distribute batch processing across the nodes to balance the use of processing resources. You can achieve this workload distribution by assigning local batch queues to one or more clusterwide generic batch queues. These generic batch queues control batch processing over the cluster by placing batch jobs in assigned batch queues that are available.



# Setting Up and Managing Cluster Queues

## 4.4 Command Procedures for Establishing Queues

---

### 4.4 Command Procedures for Establishing Queues

To configure queues on a cluster properly, you must coordinate, among cluster nodes, commands in procedures that initialize and start queues. Each active node in a cluster must initialize its local queues as well as the queues of other cluster nodes, so that when new nodes join the cluster, queues are recognized by all the nodes. However, because cluster nodes boot separately rather than simultaneously, a booting node must start only its own local queues.

As a rule, the startup command procedure for each active cluster node must initialize every queue in the cluster, but start only its local queues and any clusterwide generic queues.

You should include commands to establish queues in the SYSTARTUP procedure or in a separate command procedure file named, for example, STARTQ.COM that is invoked by your SYSTARTUP procedure. DIGITAL suggests that you set up your STARTQ command procedure(s) as a common file on a shared disk. In this case, the common STARTQ.COM file may reside on the same disk as the job controller queue file.

---

#### 4.4.1 Starting Queues Using Node-Specific Command Procedures

For each node in the cluster, either add node-specific queue commands to the node-specific SYSTARTUP procedure or create a STARTQ command procedure that is invoked by the node-specific SYSTARTUP procedure. Examples 4-1 through Example 4-3 illustrate the use of separate node-specific command procedures to initialize and start the printer configuration shown in Figure 4-1 and the batch configuration shown in Figure 4-5.

##### Example 4-1 STARTQ Command Procedure for Node JUPITR

---

```
$ SET NOON
$ !
$ ! STARTQ Command Procedure for Node JUPITR
$ !
$ ! Start job queue manager.
$ !
$ START/QUEUE/MANAGER WORK1:[CLUSMAN]
$ !
$ ! Initialize and start local printer queues.
$ !
$ INITIALIZE/QUEUE/ON=JUPITR::LPAO/START JUPITR_LPAO
$ INITIALIZE/QUEUE/ON=JUPITR::LPBO/START JUPITR_LPBO
$ !
$ ! Initialize remote printer queues.
$ !
$ INITIALIZE/QUEUE/ON=SATURN::LPAO SATURN_LPAO
$ INITIALIZE/QUEUE/ON=SATURN::LPBO SATURN_LPBO
$ INITIALIZE/QUEUE/ON=SATURN::LPCO SATURN_LPCO
$ INITIALIZE/QUEUE/ON=URANUS::URANUS_LPAO URANUS_LPAO
$ !
$ ! Initialize and start clusterwide generic printer queue.
$ !
$ INITIALIZE/QUEUE/GENERIC=(JUPITR_LPAO,SATURN_LPAO,URANUS_LPAO -
/START SYS$PRINT
```

---

Example 4-1 Cont'd. on next page

# Setting Up and Managing Cluster Queues

## 4.4 Command Procedures for Establishing Queues

### Example 4-1 (Cont.) STARTQ Command Procedure for Node JUPITR

---

```
$ !  
$ ! Initialize batch queues on local node.  
$ !  
$ INITIALIZE/QUEUE/BATCH/ON=JUPITR::/START JUPITR_BATCH  
$ INITIALIZE/QUEUE/BATCH/ON=JUPITR::/START JUPITR_TEXT  
$ !  
$ ! Initialize queues from other nodes.  
$ !  
$ INITIALIZE/QUEUE/BATCH/ON=SATURN:: SATURN_BATCH  
$ INITIALIZE/QUEUE/BATCH/ON=SATURN:: SATURN_TEXT  
$ INITIALIZE/QUEUE/BATCH/ON=URANUS:: URANUS_BATCH  
$ !  
$ ! Initialize clusterwide generic batch queue.  
$ !  
$ INITIALIZE/QUEUE/BATCH/GENERIC=(JUPITR_BATCH,SATURN_BATCH,-  
URANUS_BATCH)/START SYS$BATCH
```

---

### Example 4-2 STARTQ Command Procedure for Node SATURN

---

```
$ SET NOON  
$ !  
$ ! STARTQ Command Procedure for Node SATURN  
$ !  
$ ! Start job queue manager.  
$ !  
$ START/QUEUE/MANAGER WORK1:[CLUSMAN]  
$ !  
$ ! Initialize and start local printer queues.  
$ !  
$ INITIALIZE/QUEUE/ON=SATURN::LPAO/START SATURN_LPAO  
$ INITIALIZE/QUEUE/ON=SATURN::LPBO/START SATURN_LPBO  
$ INITIALIZE/QUEUE/ON=SATURN::LPCO/START SATURN_LPCO  
$ !  
$ ! Initialize remaining printer queues.  
$ !  
$ INITIALIZE/QUEUE/ON=JUPITR::LPAO JUPITR_LPAO  
$ INITIALIZE/QUEUE/ON=JUPITR::LPBO JUPITR_LPBO  
$ INITIALIZE/QUEUE/ON=URANUS::URANUS_LPAO URANUS_LPAO
```

---

Example 4-2 Cont'd. on next page

# Setting Up and Managing Cluster Queues

## 4.4 Command Procedures for Establishing Queues

### Example 4–2 (Cont.) STARTQ Command Procedure for Node SATURN

---

```
$ ! Initialize and start clusterwide generic printer queue.
$ !
$ INITIALIZE/QUEUE/GENERIC=(JUPITR_LPAO,SATURN_LPAO,-
  URANUS_LPAO)/START SYS$PRINT
$ !
$ ! Initialize batch queues on local node.
$ !
$ INITIALIZE/QUEUE/BATCH/ON=SATURN::/START SATURN_BATCH
$ INITIALIZE/QUEUE/BATCH/ON=SATURN::/START SATURN_TEXT
$ !
$ ! Initialize queues from other nodes.
$ !
$ INITIALIZE/QUEUE/BATCH/ON=JUPITR:: JUPITR_BATCH
$ INITIALIZE/QUEUE/BATCH/ON=JUPITR:: JUPITR_TEXT
$ INITIALIZE/QUEUE/BATCH/ON=URANUS:: URANUS_BATCH
$ !
$ ! Initialize clusterwide generic batch queue.
$ !
$ INITIALIZE/QUEUE/BATCH/GENERIC=(JUPITR_BATCH,SATURN_BATCH,-
  URANUS_BATCH) SYS$BATCH
```

---

### Example 4–3 STARTQ Command Procedure for Node URANUS

---

```
$ SET NOON
$ !
$ ! STARTQ Command Procedure for Node URANUS
$ !
$ ! Start job queue manager.
$ !
$ START/QUEUE/MANAGER WORK1:[CLUSMAN]
$ !
$ ! Initialize and start local printer queue.
$ !
$ INITIALIZE/QUEUE/ON=URANUS::LPAO/START URANUS_PRINT
$ !
$ ! Initialize remaining printer queues.
$ !
$ INITIALIZE/QUEUE/ON=JUPITR::LPAO JUPITR_LPAO
$ INITIALIZE/QUEUE/ON=JUPITR::LPBO JUPITR_LPBO
$ INITIALIZE/QUEUE/ON=SATURN::LPAO SATURN_LPAO
$ INITIALIZE/QUEUE/ON=SATURN::LPBO SATURN_LPBO
$ INITIALIZE/QUEUE/ON=SATURN::LPCO SATURN_LPCO
$ !
$ ! Initialize and start clusterwide generic printer queue.
$ !
$ INITIALIZE/QUEUE/GENERIC=(JUPITR_LPAO,SATURN_LPAO,-
  URANUS_LPAO)/START SYS$PRINT
$ !
$ ! Initialize batch queues on local node.
$ !
$ INITIALIZE/QUEUE/BATCH/ON=URANUS::/START URANUS_BATCH
```

---

Example 4–3 Cont'd. on next page

# Setting Up and Managing Cluster Queues

## 4.4 Command Procedures for Establishing Queues

### Example 4–3 (Cont.) STARTQ Command Procedure for Node URANUS

---

```
$ !
$ ! Initialize queues from other nodes.
$ !
$ INITIALIZE/QUEUE/BATCH/ON=JUPITR:: JUPITR_BATCH
$ INITIALIZE/QUEUE/BATCH/ON=JUPITR:: JUPITR_TEXT
$ INITIALIZE/QUEUE/BATCH/ON=SATURN:: SATURN_BATCH
$ INITIALIZE/QUEUE/BATCH/ON=SATURN:: SATURN_TEXT
$ !
$ ! Initialize clusterwide generic batch queue.
$ !
$ INITIALIZE/QUEUE/BATCH/GENERIC=(JUPITR_BATCH,SATURN_BATCH,-
URANUS_BATCH) SYS$BATCH
```

---

In Examples 4–1 through 4–3, each command procedure performs the following operations for the specific node:

- Starts the system job queue manager
- Specifies the location of the job controller queue file
- Initializes and starts each local queue on the local node
- Initializes all other queues from other nodes
- Initializes and starts the clusterwide generic printer queue SYS\$PRINT
- Initializes and starts the clusterwide generic batch queue SYS\$BATCH

---

### 4.4.2 Starting Queues Using a Common Command Procedure

You can create a common command procedure, named for example, STARTQ.COM, and store it on a shared disk. Using this method, each node can share the same copy of the common STARTQ.COM procedure. Each node invokes the common STARTQ.COM procedure from the common version of SYSTARTUP. You can also include the commands to set up queues in the common SYSTARTUP file instead of in a separate STARTQ.COM file.

Example 4–4 illustrates the use of a common STARTQ command procedure on a shared disk to initialize and start the printer queues shown in Figure 4–1.

# Setting Up and Managing Cluster Queues

## 4.4 Command Procedures for Establishing Queues

### Example 4-4 Starting Queues Using a Common Command Procedure

---

```
$ !
$ ! Compute the name of the executing node.
$ !
$ NODE = F$GETSYI("NODENAME")
$ !
$ JUPITR_START = "/NOSTART"
$ SATURN_START = "/NOSTART"
$ URANUS_START = "/NOSTART"
$ !
$ ! Redefine one of the previous symbols.
$ !
$ 'NODE'_START = "/START"
$ !
$ SET NOON
$ !
$ ! Start up the job controller.
$ !
$ START/QUEUE/MANAGER WORK1:[CLUSMAN]
$ !
$ ! Set up printer queues.
$ ! Initialize all nodes. Start local node only.
$ !
$ INITIALIZE/QUEUE/ON=JUPITR::LPAO 'JUPITR_START' JUPITR_LPAO
$ INITIALIZE/QUEUE/ON=JUPITR::LPBO 'JUPITR_START' JUPITR_LPBO
$ !
$ INITIALIZE/QUEUE/ON=SATURN::LPAO 'SATURN_START' SATURN_LPAO
$ INITIALIZE/QUEUE/ON=SATURN::LPBO 'SATURN_START' SATURN_LPBO
$ INITIALIZE/QUEUE/ON=SATURN::LPCO 'SATURN_START' SATURN_LPCO
$ !
$ INITIALIZE/QUEUE/ON=URANUS::LPAO 'URANUS_START' -
URANUS_PRINT
$ !
$ ! Set up main batch queues.
$ !
$ INITIALIZE/QUEUE/BATCH/ON=JUPITR::/JOB=6/WSEXTENT=500 -
'JUPITR_START' JUPITR_BATCH
$ !
$ INITIALIZE/QUEUE/BATCH/ON=SATURN::/JOB=5/WSEXTENT=600 -
'SATURN_START' SATURN_BATCH
$ !
$ INITIALIZE/QUEUE/BATCH/ON=URANUS/JOB=6/WSEXTENT=600 -
'URANUS_START' URANUS_BATCH
$ !
$ ! Set up batch processing queues.
$ !
$ INITIALIZE/QUEUE/BATCH/ON=JUPITR::/JOB=2/WSEXTENT=1500 -
'JUPITR_START' JUPITR_TEXT
$ !
$ INITIALIZE/QUEUE/BATCH/ON=SATURN::/JOB=2/WSEXTENT=1500 -
'SATURN_START' SATURN_TEXT
$ !
$ ! Set up clusterwide generic batch processing queue.
$ !
$ INITIALIZE/QUEUE/BATCH/GENERIC=(JUPITR_BATCH,SATURN_BATCH,-
URANUS_BATCH) SYS$BATCH
```

---

# Setting Up and Managing Cluster Queues

## 4.4 Command Procedures for Establishing Queues

The command procedure in Example 4-4 performs the same queue setup operations as the command procedures shown in Examples 4-1 through 4-3. However, the common STARTQ file in this example executes a common set of commands that function according to the node executing them. A set of conditional symbols are assigned to control whether queues are started. In this way, each node initializes all the queues in the cluster but starts only its own.

## 4.5

---

### Summary of Commands for Setting Up Cluster Queues

Following is a summary of commands used to set up cluster queues.

- **Start the system job queue manager**

```
$ START/QUEUE/MANAGER file-spec
```

- **Set up printer queues**

```
$ INITIALIZE/QUEUE/ON=node::device queue-name  
$ INITIALIZE/QUEUE/ON=node::device/START queue-name
```

- **Set up generic printer queues**

```
$ INITIALIZE/QUEUE/GENERIC=(queue1,queue2...)/START queue-name
```

- **Set up batch queues**

```
$ INITIALIZE/QUEUE/BATCH/ON=node:: queue-name  
$ INITIALIZE/QUEUE/BATCH/ON=node::/START queue-name
```

- **Set up generic batch queues**

```
$ INITIALIZE/QUEUE/BATCH/GENERIC=(queue1,queue2...)/START queue-name
```

# 5

---

## Setting Up and Managing Cluster Disks

In any VAXcluster configuration, there are two types of disk and tape devices:

- Restricted-access devices, which are accessible only by the local node or nodes to which they are directly connected.
- Cluster-accessible devices, which are accessible by any node in the cluster.

A disk or magnetic tape device connected to an HSC is by design a cluster-accessible device. Any other disk device, such as a MASSBUS, UNIBUS, or BI disk, is a restricted-access device, unless you explicitly set it up as a cluster-accessible device.

As system manager, you are responsible for planning, organizing, and setting up the proper cluster device configuration for your site. You must decide which disk devices should have access restricted to the local node, and which should be accessible to the cluster. For example, you may want to restrict access to a particular disk to the users on the node directly connected to the device. Or, you may decide to set up a disk as a cluster-accessible device, so that any user on any cluster node can allocate and use it.

Once you have planned your configuration strategy, you can use the procedures outlined in this chapter to set up and manage cluster disks. Topics include the following:

- Cluster-accessible disks
- Cluster device-naming conventions
- Shared disk volumes
- Setting up cluster devices
- Volume shadowing in mixed-interconnect clusters

---

### 5.1 Cluster-Accessible Disks

A *cluster-accessible disk* is a disk that every node in the cluster can recognize and access. The following types of disks are cluster accessible:

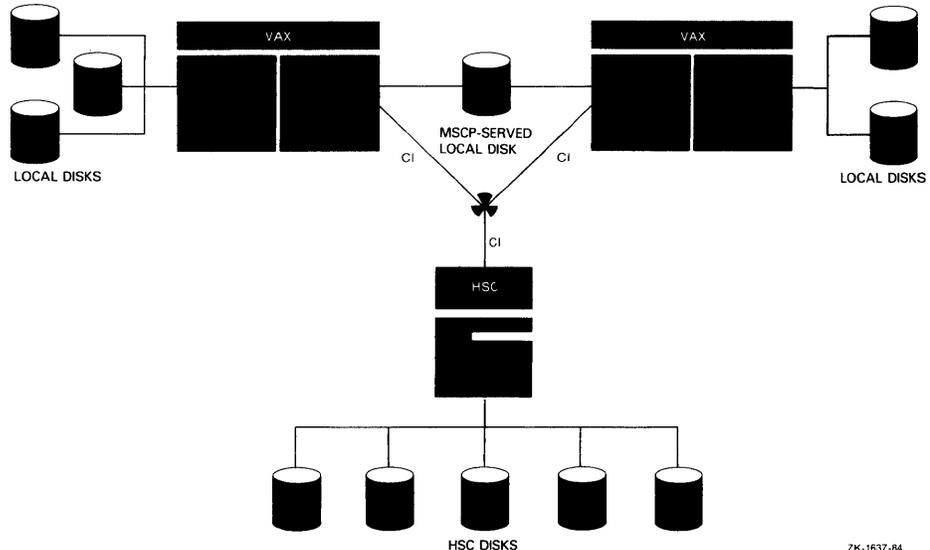
- HSC disks
- MSCP-served disks
- Dual-pathed disks

Figure 5-1 illustrates how disks might be configured in a typical CI-only cluster. The HSC disks and the dual-ported MSCP-served local disk are considered cluster accessible.

# Setting Up and Managing Cluster Disks

## 5.1 Cluster-Accessible Disks

Figure 5-1 CI-Only Configuration With Shared Disks



### 5.1.1 HSC Disks

An HSC disk is a DIGITAL Storage Architecture (DSA) disk that is connected to an HSC.

If an HSC is connected in a cluster, its disks are automatically accessible by any node in the cluster. You can also set up HSC disks to be dual pathed between two HSCs. Dual-pathed disks are described in Section 5.1.3.

### 5.1.2 MSCP-Served Disks

MSCP is the protocol used to communicate between a VAX host and a DSA controller. The MSCP Server enables a VAX processor to make locally connected disks such as MASSBUS, UNIBUS, or BI disks available to all other cluster members.

Unlike HSC devices, controllers for locally connected disks are not automatically cluster accessible. Access to these devices is restricted to the local node unless you explicitly set them up as cluster accessible, using the MSCP Server.

To make a disk accessible to all cluster nodes, the MSCP Server must be loaded on the local node, and it must be instructed to make the disk available clusterwide. These functions are enabled with the SYSGEN parameters `MSCP_LOAD` and `MSCP_SERVE_ALL`. By specifying appropriate values for these parameters in a node's `MODPARAMS.DAT` file, and then running `AUTOGEN` to reboot the node, you enable the node to serve all suitable disks to the cluster early in the boot sequence. (You can also use the `CLUSTER_CONFIG.COM CHANGE` function to perform these operations.) The served disks thus become accessible with minimal interruption whenever the serving node reboots. Further, the MSCP Server automatically serves any suitable disk that is added to the system later. For example, if new drives are attached

# Setting Up and Managing Cluster Disks

## 5.1 Cluster-Accessible Disks

to an HSC controller, the disks become available within seconds after the cables are connected.

Table 5–1 shows the values you can specify for the parameters to configure the MSCP Server. Initial values are determined by your responses when you execute the VMS installation or upgrade procedure, or when you execute the CLUSTER\_CONFIG.COM command procedure described in Chapter 3 to set up your configuration. Note that if you later change the values, you must reboot the system on which the values are changed, before the new values can take effect (see Section 3.2.3).

**Table 5–1 Specifying Values for MSCP\_LOAD and MSCP\_SERVE\_ALL Parameters**

Parameter	Value	Function
MSCP_LOAD	0	Do not load the MSCP Server (default value).
	1	Load the MSCP Server with attributes specified by MSCP_SERVE_ALL parameter.
MSCP_SERVE_ALL	0	Do not serve any disks (default value).
	1	Serve all available disks.
	2	Serve only locally-connected (non-HSC) disks.

### 5.1.3 Dual-Pathed Disks

A *dual-pathed disk* is a dual-ported disk that is accessible to all the nodes in the cluster, not just to the nodes that are physically connected to the disk. Dual-pathed disks can be any of the following:

- Dual-ported HSC disks
- Dual-ported DSA disks using UDA/KDA/BDA controllers
- Dual-ported MASSBUS disks

The term *dual-pathed* refers to the two paths through which cluster nodes can access a disk to which they are not directly connected. If one path fails, the disk is accessed over the other path. (Note that with a dual-ported MASSBUS disk, a node directly connected to the disk always accesses it locally.)

#### 5.1.3.1 Dual-Ported HSC Disks

By design, HSC disks are cluster accessible. Therefore, if they are dual ported, they are automatically dual pathed. CI-connected cluster nodes can access a dual-pathed HSC disk by way of a path through either HSC connected to the device.

For each dual-ported HSC disk, you can control failover to a specific port using the port select buttons on the front of each drive. By pressing either port select button (A or B) on a particular drive, you can cause the device to fail over to the specified port.

With the port select buttons, you can select alternate ports to balance the disk controller workload between two HSCs. For example, you could set half of your disks to use Port A and set the other half to use Port B.

The port select buttons also enable you to fail over all the disks to an alternate port manually when you anticipate the shutdown of one of the HSCs.

# Setting Up and Managing Cluster Disks

## 5.1 Cluster-Accessible Disks

---

### 5.1.3.2 Dual-Ported DSA Disks

A dual-ported DSA disk be failed over between the two VAX systems that serve it to the cluster. However, because a DSA disk can be online to only one controller at a time, only one of the systems can use its local connection to the disk. The second system accesses the disk through the MSCP Server. If the system that is currently serving the disk fails, the other system detects the failure and fails the disk over to its local connection. The disk is thereby made available to the cluster once more.

---

### 5.1.3.3 Dual-Ported MASSBUS Disks

In clusters with only two active nodes, a dual-ported MASSBUS disk is considered cluster accessible if it is connected between the two nodes, and if it has the same device name on both nodes. The Distributed File System synchronizes access to files on the disk.

To set up a dual-ported MASSBUS disk in a two-node cluster, enter the DCL command SET DEVICE in the following format before mounting the disk:

```
§ SET DEVICE/DUAL_PORT device-name
```

**Note:** A MASSBUS disk may be used either as a dual-ported disk or as a system disk, but not both.

In clusters with more than two active nodes, you can set up a dual-ported MASSBUS disk to be cluster accessible through the MSCP Server on either or both nodes to which the disk is connected. Be sure, however, *not* to use the SYSGEN commands AUTOCONFIGURE or CONFIGURE to configure a dual-ported MASSBUS disk that is already available on the system through the MSCP Server. Establishing a local connection to the disk when a remote path is already known creates two uncoordinated paths to the same disk. Use of these two paths can corrupt files and data on any disk mounted on the drive.

If the local path to the disk is not found during the system bootstrap procedure, the MSCP Server path from the remote node is the only available access to the drive. The local path is not found during a boot if any of the following conditions exist:

- The port select switch for the drive is not enabled for the local node.
- The disk, cable, or adapter hardware for the local path is broken.
- There is sufficient activity on the other port to "mask" the existence of the port.
- The system is booted in such a way that the SYSGEN command AUTOCONFIGURE ALL in the site-independent startup procedure (SYS\$SYSTEM:STARTUP.COM) was not executed.

Use of the disk is still possible through the MSCP Server path.

**Caution:** Under these conditions, do not attempt to add the local path back into the system I/O database using the SYSGEN commands AUTOCONFIGURE or CONFIGURE. SYSGEN is currently unable to detect the presence of the disk's MSCP path and would incorrectly build a second set of data structures to describe it. Subsequent events could lead to incompatible and uncoordinated file operations, which might corrupt the volume.

To recover the local path to the disk, you must reboot the system connected to that local path.

# Setting Up and Managing Cluster Disks

## 5.1 Cluster-Accessible Disks

Note that if the disk is not dual ported or is never MSCP served on the remote host, this restriction does not apply.

---

## 5.2 Cluster Device-Naming Conventions

To manage cluster devices properly, you must understand the conventions used to identify them. Every cluster device is identified by a unique name, which provides a reliable way to access it in the cluster.

Devices that are local to a cluster node can be accessed by that node through the traditional device name (for example, DJA1) or through a cluster device name in the format `node$device` (for example, JUPITR\$DJA1).

However, a device that is dual pathed between two nodes must be identified by a unique, path-independent name that includes an *allocation class*. The allocation class is a numeric value from 0 to 255 that is used to create a device name in the following format:

`$allocation-class$device-name`

For example, the allocation class device name `$1$DJA16` identifies a disk that is dual ported between two nodes (VAX or HSC) that both have an allocation class value of 1.

Each time a node that is not directly connected to such a disk tries to access the disk, the choice of which path to take is made arbitrarily, because no path to the disk is ever guaranteed. Because the access path is chosen without regard to the names of the nodes (VAX or HSC) serving the disk, an allocation class device name is required to identify the disk uniquely.

---

### 5.2.1 Rules for Specifying Allocation Class Values

Allocation classes play an important role in determining strategies for configuring and naming disks. In fact, the VMS operating system uses allocation class values above all other available information when determining the configuration of cluster devices.

The following rules apply for specifying allocation class values:

- VAX or HSC nodes connecting a dual-pathed disk must have the same non-zero allocation class value.
- All cluster-accessible disks on nodes with a non-zero allocation class value must have unique names. For example, if two VAX nodes have the same allocation class value, it is invalid for both nodes to have a disk named DJA0. This restriction also applies to HSCs.
- Single-ported disks with an allocation class value of zero can have the same unit number on different cluster nodes.

# Setting Up and Managing Cluster Disks

## 5.2 Cluster Device-Naming Conventions

Note that 0 is the default allocation class value. Any node in a CI-only cluster that is not connected to a dual-pathed disk should be assigned this value. *In a mixed-interconnect cluster, however, all of the following must have a non-zero allocation class value:*

- HSCs
- Systems serving HSC disks
- Systems connected to dual-pathed disks

Failure to set allocation class values correctly may cause both disk corruption and locking conflicts that can suspend normal cluster operations.

To assign an allocation class value to a VAX node that supports dual-pathed devices, specify the value with the SYSGEN parameter ALLOCLASS. To assign an allocation class for an HSC, specify the value using the HSC console to enter a command in the following format, where n is the allocation class value.

```
SET ALLOCATE DISK n
```

For complete information on HSC console commands, refer to the HSC hardware documentation.

---

### 5.2.2 Sample Configurations with Named Devices

Figures 5-2 and 5-3 show how cluster device names are specified for the following:

- Dual-pathed HSC disks
- Dual-pathed DSA disks

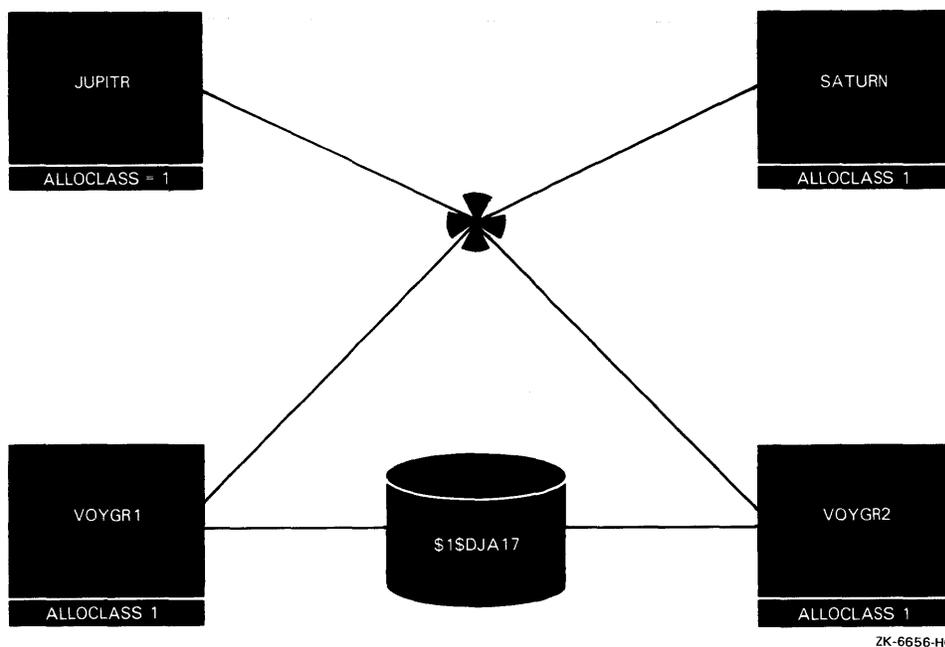
Figure 5-4 shows how device names are typically specified in a mixed-interconnect cluster. This figure also shows relevant SYSGEN parameter settings in MODPARAMS.DAT.

A typical configuration with a dual-pathed HSC disk is illustrated in Figure 5-2. Note that the allocation class value (1) is the same on all nodes, and that the disk's device name (\$1\$DJA17) is constructed using that value. VAX nodes JUPITR and SATURN can access the disk through either of the HSCs VOYGR1 or VOYGR2.

# Setting Up and Managing Cluster Disks

## 5.2 Cluster Device-Naming Conventions

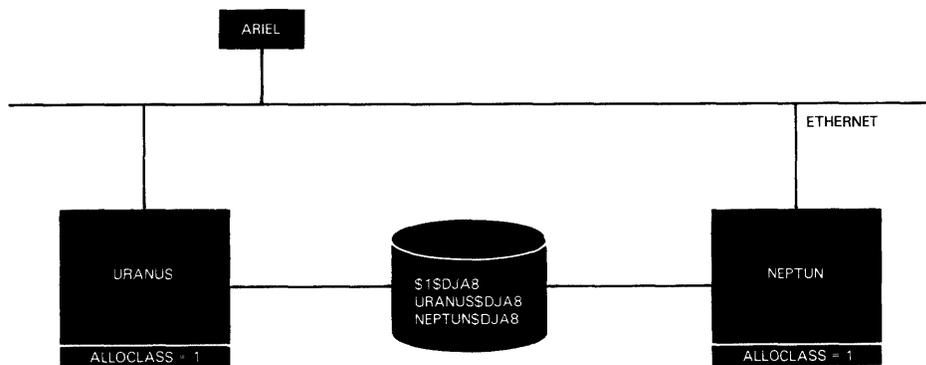
Figure 5–2 Configuration with a Dual-Pathed HSC Disk



ZK-6656-HC

Figure 5–3 shows a configuration with a dual-pathed DSA disk.

Figure 5–3 Configuration with a Dual-Pathed DSA Disk



ZK-6655-HC

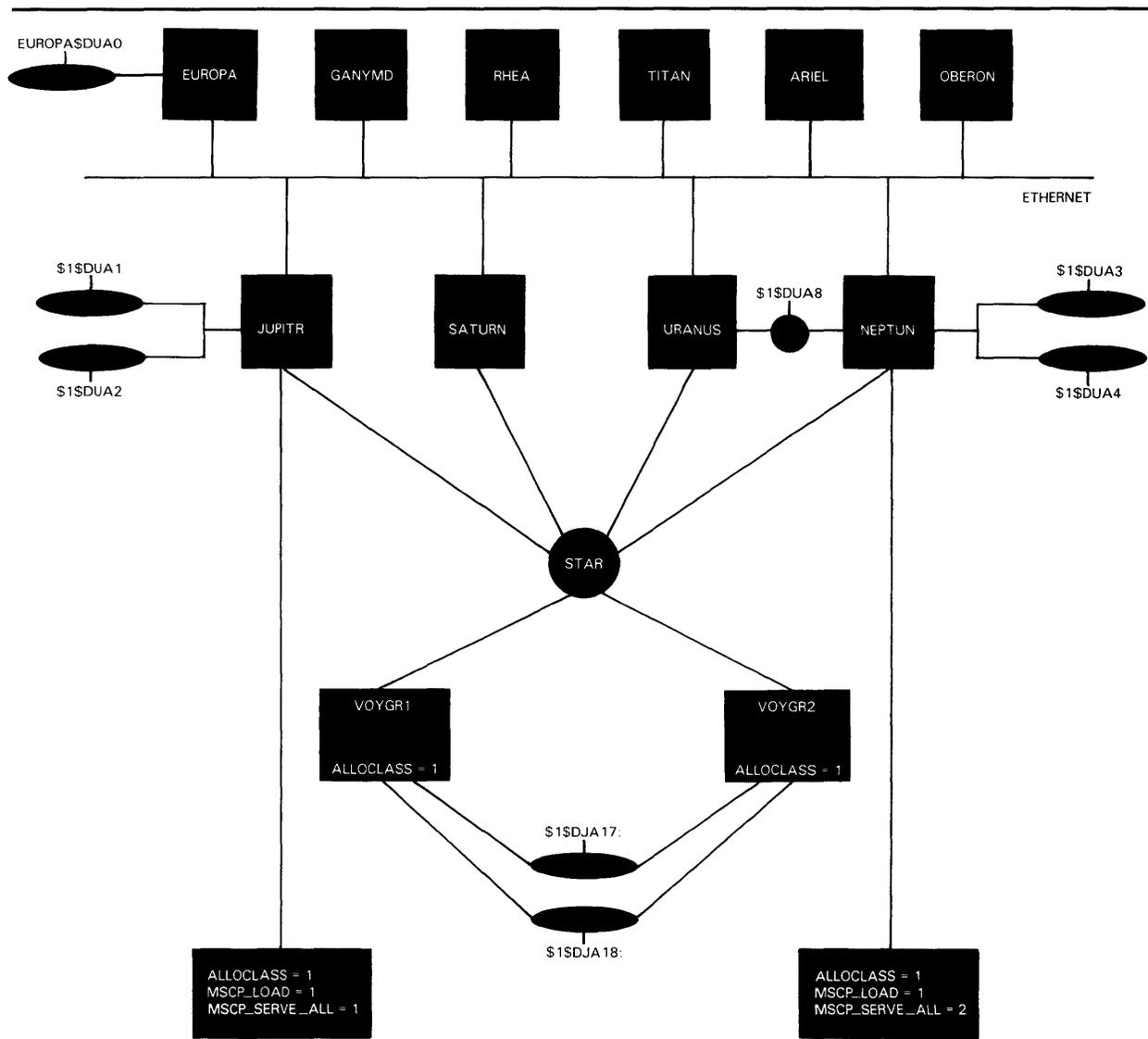
Nodes URANUS and NEPTUN can access the disk either locally or through the other node's MSCP Server. When satellite node ARIEL accesses the disk, however, it arbitrarily chooses a path through either URANUS or NEPTUN. If ARIEL tries to access the disk by using the node-specific device name URANUSSDJA8, and this disk is not currently accessible through URANUS, access will fail. But if ARIEL uses the allocation class device name \$1SDJA8, it can access the disk through NEPTUN. As a general rule, you should always use a path-independent, allocation class device name to identify dual-pathed cluster disks.

# Setting Up and Managing Cluster Disks

## 5.2 Cluster Device-Naming Conventions

Figure 5-4 illustrates the use of device names in a mixed-interconnect cluster.

**Figure 5-4 Device Names in a Mixed-Interconnect Cluster**



ZK-6660-HC

In this configuration, a set of disks is dual-pathed to the HSC controllers named VOYGR1 and VOYGR2, and these controllers are connected to VAX processor JUPITR. Because ALLOCLASS is set to the same value (1) on JUPITR and on both HSCs, JUPITR can serve the disks on VOYGR1 and VOYGR2 to all satellite nodes in the cluster.

Disks on the HSCs have allocation class names of the form \$1\$ddcu. For example, the disk DUA17 is named \$1\$DUA17. On CI-connected nodes, VMS software would also recognize the disk as JUPITR\$DUA17 and as either VOYGR1\$DUA17 or VOYGR2\$DUA17. On satellites, it would recognize the disk as JUPITR\$DUA17 or as \$1\$DUA17. This example shows why you

# Setting Up and Managing Cluster Disks

## 5.2 Cluster Device-Naming Conventions

should always use an allocation class name like \$1\$DUA17 when configuring cluster devices: the allocation class name is the only name that all cluster nodes recognize at all times.

Note that, for optimal availability, two or more CI-connected VAX processors should serve HSC disks to the cluster. For example, because MSCP\_SERVE\_ALL is set to 1 on nodes JUPITR, SATURN, and URANUS, and because ALLOCLASS is set to the same value on those nodes and on the HSCs, JUPITR, SATURN, and URANUS can serve disks on the HSCs. But because MSCP\_SERVE\_ALL is set to 2 on node NEPTUN, that node can serve only its local disks.

---

### 5.3 Shared Disks

A *shared disk* is a disk that is mounted on a cluster-accessible device by one or more nodes in the cluster. Shared disks play a key role in common-environment clusters, because when you place system files or command procedures on a shared disk, cluster nodes can share a single copy of each common file (see Chapter 2). Note, however, that a shared disk is a single point of failure for data access by the nodes sharing the disk.

To mount cluster-accessible disks that are to be shared among all cluster nodes, specify the same MOUNT command on each node or specify the MOUNT command with the /CLUSTER qualifier on one node. When you execute MOUNT/CLUSTER on one node, the disk is mounted on every node in the cluster at the time the command executes. Note that only system or group disks can be mounted clusterwide. Thus, if you specify MOUNT/CLUSTER without the /SYSTEM or /GROUP qualifier, /SYSTEM is assumed. Also note that each cluster disk mounted with the /SYSTEM, /GROUP, or /SHARED qualifiers must have a unique volume label.

If you want to mount a shared disk on some but not all the nodes in the cluster, execute the same MOUNT command (without the /CLUSTER qualifier) on each node sharing the disk.

For example, suppose you want all the nodes in a three-node cluster to share a disk named COMPANYDOCS. To share the disk, each of the three nodes could execute identical MOUNT commands, or one of the three nodes could mount COMPANYDOCS using the MOUNT/CLUSTER command, as follows:

```
$ MOUNT/SYSTEM/CLUSTER/NOASSIST $1$DUA4: COMPANYDOCS
```

If you want just two of the three nodes to share the disk, those two nodes must both mount the disk with the same MOUNT command. For example:

```
$ MOUNT/SYSTEM/NOASSIST $1$DUA4: COMPANYDOCS
```

To mount the disk at startup time, include the mount command either in a common command procedure that is invoked at startup time, or in the node-specific startup command procedure.

# Setting Up and Managing Cluster Disks

## 5.4 Setting Up Cluster Devices

---

### 5.4 Setting Up Cluster Devices

To implement your plans for configuring cluster disks, you can create command procedures to set up and mount them. You may want to include commands that set up and mount cluster disks in a separate command procedure file that is invoked by a site-specific SYSTARTUP procedure. Depending on your cluster environment, you can set up your command procedure in either of the following ways:

- As a separate file specific to each node in the cluster
- As a common node-independent file

You can set up the common procedure as a shared file on a shared disk, or you can make duplicate copies of the common procedure and store them as separate files. With either method, each node can invoke the common procedure from the site-specific SYSTARTUP procedure.

The MSCPMOUNT.COM example in the SYS\$EXAMPLES directory on your system shows a sample common command procedure used to mount cluster disks.

---

### 5.5 Volume Shadowing in Mixed-Interconnect Clusters

If shadowing is to be used anywhere in a mixed-interconnect cluster, all CI-connected VAX nodes *must* have the SYSGEN parameter SHADOWING set to 1. This setting causes them to use the shadowing driver, DSDRIVER. The MSCP Server serves the shadow set virtual unit to the satellites.

Example 5-1 shows how the shadow set appears when you enter the DCL command SHOW DEVICES D on a boot server.

#### Example 5-1 Shadow Set as Seen from Boot Server

---

Device Name	Device	Device Status	Error Count	Volume Label	Free Blocks	Trans Count	Mnt Cnt
\$1\$DUA111:	(VOYGR1)	ShadowSetMember	0	(member of \$1\$DUS111:)			
\$1\$DUA151:	(VOYGR1)	ShadowSetMember	0	(member of \$1\$DUS111:)			
\$1\$DUS111:	(VOYGR1)	Mounted	0	VMS08JUL	244688	118	21

---

Satellites must have the SHADOWING parameter set to 0. This setting causes them to use the non-shadowing driver, DUDRIVER. Satellites access the shadow set by mounting the virtual unit, and they can see the virtual unit through the MSCP Server. The shadow set appears to have the same characteristics as any other disk device, as shown in Example 5-2. However, while satellites can see shadow set member units, they cannot access them individually.

# Setting Up and Managing Cluster Disks

## 5.5 Volume Shadowing in Mixed-Interconnect Clusters

### Example 5–2 Shadow Set as Seen from Satellite

---

Device Name	Device Status	Error Count	Volume Label	Free Blocks	Trans Count	Mnt Cnt
\$1\$DUA111:	(SATURN) Online	0	(remote shadow member)			
\$1\$DUA151:	(SATURN) Online	0	(remote shadow member)			
\$1\$DUS111:	(SATURN) Mounted	0	VMS08JUL	244688	121	21

---

In mixed-interconnect clusters it is recommended that at least two boot servers should serve the shadow set, so that if one server should fail, another is available to keep the shadow set intact. For complete information on volume shadowing, see the *VAX Volume Shadowing Manual*.

### 5.5.1 Mounting Shadow Sets

Satellites have no knowledge of shadow set configuration, and they cannot issue any shadow set maintenance commands using the /SHADOW qualifier. All commands that create, modify, and dissolve shadow sets must be entered on a CI-connected node. For example, you must enter a command like the following on a CI-connected node:

```
$ MOUNT/SYSTEM $1$DUS111:/SHADOW=($1$DUA111,$1$DUA151) VMS08JUL
```

When a shadow set virtual unit is created by a MOUNT command on a CI-connected node, the MSCP Server automatically serves the virtual unit to other CI-connected nodes. A MOUNT/SYSTEM command entered on a CI-connected node forms the shadow set on the CI-connected node. Once the shadow set is formed, you can use the MOUNT/CLUSTER command to mount it on all CI-connected nodes and satellites.

For example, to mount clusterwide the shadow set shown in Example 5–1, you must enter two commands. First, enter the following command on any CI-connected node:

```
$ MOUNT/SYSTEM $1$DUS111:/SHADOW=($1$DUA111,$1$DUA151) VMS08JUL
```

This command creates the virtual unit, forms the shadow set, and mounts it on the CI-connected node. The virtual unit is automatically served after it is created.

Next, enter the following command:

```
$ MOUNT/CLUSTER $1$DUS111: /SHADOW=VMS08JUL
```

This command mounts the shadow set on the remaining CI-connected nodes and on satellites.

### 5.5.2 Dismounting Shadow Sets

Be careful when dismounting shadow sets. The shadow set virtual unit must always be dismounted on all satellites before being dismounted (and possibly dissolved) on the CI-connected VAX nodes. If these nodes dismount the shadow set before satellites do, the shadow set will be dissolved. The satellites will then have the virtual unit mounted, but will have no path (through a CI-connected node) to the member units. The satellites will therefore place the virtual unit in mount verification. This condition can result in suspended operations, and require a cluster reboot, because satellites

# Setting Up and Managing Cluster Disks

## 5.5 Volume Shadowing in Mixed-Interconnect Clusters

may hold locks that must be released before the CI-connected node can rebuild the shadow set.

If this condition occurs, you can remount the shadow set on a CI-connected serving node. When that node reforms the shadow set, the satellites can once again access the volume—provided that the CI-connected node has been able to rebuild the shadow set.

In general, you should use the command DISMOUNT/SYSTEM, rather than DISMOUNT/CLUSTER, to dismount shadow sets in mixed-interconnect clusters.

### 5.5.3 Using Shadow Sets as Satellite System Disks

A satellite system disk can be a shadow set. The system device parameter in the DECnet database for satellites must be the device name of the shadow set virtual unit (for example, \$1\$DUS111). No description of shadow set member units is needed.

# A

## Cluster SYSGEN Parameters

For systems to boot properly into a cluster, certain system parameters must be set on each cluster node. Table A-1 lists SYSGEN parameters used in cluster configurations.

**Table A-1 Cluster SYSGEN Parameters**

Parameter	Description
ALLOCLASS	Specifies a numeric value from 0 to 255 to be assigned as the allocation class for the node. The default value is 0.
DISK_QUORUM	The name, in ASCII, of an optional quorum disk. ASCII spaces indicate that no quorum disk is being used. DISK_QUORUM must be defined on one or more cluster nodes capable of having a direct (non-MSCP served connection to the disk). These nodes are called <i>quorum disk watchers</i> . The remaining nodes (nodes with a blank value for DISK_QUORUM) recognize the name defined by the first watcher node which which they communicate.
EXPECTED_VOTES	Specifies a setting that is used to derive the initial quorum value. This setting is the sum of all VOTES held by potential cluster members.  By default, the value is 1. The connection manager sets a quorum value to a number that will prevent cluster partitioning (see Section 1.5). To calculate quorum, the system uses the following formula:  $\text{estimated quorum} = (\text{EXPECTED\_VOTES} + 2)/2$
MSCP_LOAD	Controls whether the MSCP Server is loaded. Specify 1 to load the server. By default, the value is set to zero, and the server is not loaded.
MSCP_SERVE_ALL	Specifies MSCP disk serving functions when the MSCP Server is loaded. The default value of zero specifies that no disks are served. A value of 1 specifies that all available disks are served. A value of 2 specifies that only locally-connected (non-HSC) disks are served.
NISCS_CONV_BOOT	Specifies whether conversational bootstraps are enabled on the node. The default value of zero specifies that conversational bootstraps are disabled. A value of 1 enables conversational bootstraps.
NISCS_LOAD_PEA0	Specifies whether the VAXport driver PEDRIVER is to be loaded to enable cluster communications over the Ethernet. The default value of zero specifies that the driver is not loaded. A value of 1 specifies that that driver is loaded.
NISCS_PORT_SERV	Specifies whether data checking is enabled for the node. The default value of zero specifies that data checking is disabled.
QDSKVOTES	Specifies the number of votes contributed to the cluster votes total by a quorum disk. The maximum is 127, the minimum is 0, and the default is 1. This parameter is used only when DISK_QUORUM is defined.
QDSKINTERVAL	Specifies the disk quorum polling interval, in seconds. The maximum value is 32767, the minimum value is 1, and the default is 10. Lower values trade increased overhead cost for greater responsiveness.  DIGITAL recommends that this parameter be set to the same value on each cluster node.

# Cluster SYSGEN Parameters

**Table A–1 (Cont.) Cluster SYSGEN Parameters**

Parameter	Description
RECNXINTERVAL	<p>Specifies, in seconds, the interval during which the connection manager attempts to reconnect a broken connection to another VMS system. If a new connection cannot be established during this period, the connection is declared irrevocably broken, and either this system or the other must leave the cluster. This parameter trades faster response to certain types of system failures against the ability to survive transient faults of increasing duration.</p> <p>DIGITAL recommends that this parameter be set to the same value on each cluster node.</p>
VAXCLUSTER	<p>Controls whether the system should join or form a cluster. This parameter accepts the following three values:</p> <ul style="list-style-type: none"> <li>• 0—Specifies that the system will not participate in a cluster.</li> <li>• 1—Specifies that the system should participate in a cluster if hardware supporting SCS is present (CI, UDA, HSC50).</li> <li>• 2—Specifies that the system should participate in a cluster</li> </ul> <p>You should always set this parameter to 2 on systems intended to run in a cluster, 0 on systems that boot from a UDA and are not intended to be part of a cluster, and 1 (the default) otherwise.</p>
VOTES	<p>Specifies the number of votes towards a quorum to be contributed by the node. By default, the value is 1.</p>
<p><b>SCS Parameters</b></p>	
PANUMPOLL	<p>Specifies the number of ports to poll at each interval. DIGITAL recommends that this parameter be set to the same value on each cluster node.</p>
PASTIMOUT	<p>Specifies the interval at which the CI port driver performs time-based bookkeeping operations. This interval is also the period after which a start handshake datagram is assumed to have timed out.</p> <p>Normally the default value is adequate. DIGITAL recommends that this parameter be set to the same value on each cluster node.</p>
PASTDGBUF	<p>Specifies the number of datagram receive buffers to queue for the CI port driver's configuration poller; that is, the maximum number of start handshakes that can be in progress simultaneously.</p> <p>Normally the default value is adequate. DIGITAL recommends that this parameter be set to the same value on each cluster node.</p>
PAMAXPORT	<p>Specifies the maximum number of CI ports the CI port driver polls for a broken port-to-port virtual circuit, or a failed remote node.</p> <p>You can decrease this parameter in order to reduce polling activity if the hardware configuration has fewer than 16 ports. For example, if the configuration has a total of five ports assigned port numbers 0-4, then you should set PAMAXPORT to 4.</p> <p>The default for this parameter is 15 (poll for all possible ports 0 through 15). DIGITAL recommends that this parameter be set to the same value on each cluster node.</p>

# Cluster SYSGEN Parameters

**Table A-1 (Cont.) Cluster SYSGEN Parameters**

Parameter	Description
PANOPOLL	<p>Disables CI polling for ports if set to 1. (The default is 0.) When PANOPOLL is set, a system will not discover that another system has shut down or powered down promptly and will not discover a new system that has booted. This parameter is useful when you want to bring up a system detached from the rest of the cluster for checkout purposes. It is roughly equivalent to uncabing the system from the star coupler.</p> <p>PANOPOLL = 0 is the normal setting and is required if you are booting from an HSC.</p>
PAPOLLINTERVAL	<p>Specifies in seconds, the polling interval the computer interconnect (CI) port driver uses to poll for a newly booted system, a broken port-to-port virtual circuit, or a failed remote node.</p> <p>This parameter trades polling overhead against quick response to virtual circuit failures. DIGITAL recommends that you use default value for this parameter.</p> <p>DIGITAL recommends that this parameter be set to the same value on each cluster node.</p>
PAPOOLINTERVAL	<p>Specifies in seconds, the interval at which the PA port driver checks for available nonpaged pool after a failure to allocate.</p> <p>Normally the default value is adequate.</p>
PASANITY	<p>Controls whether the port sanity timer is enabled to permit remote systems to detect a system that has been halted or retained at IPL 7 for a prolonged period. This parameter is normally set to 1 and should only be set to 0 when debugging with XDELTA.</p> <p>PASANITY is a dynamic parameter (altered the next time the port is initialized) and has a default value of 1.</p>
PRCPOLINTERVAL	<p>Specifies, in seconds, the polling interval used to look for SCS applications, such as the connection manager and MSCP disks, on other nodes. Each node is polled, at most, once each interval.</p> <p>This parameter trades polling overhead against quick recognition of new systems or servers as they appear. DIGITAL recommends that you set this parameter to 15, which is the default.</p>
SCSBUFFCNT	<p>Specifies the number of computer interconnect (CI) buffer descriptors configured for all CI ports on the system.</p>
SCSCONNCNT	<p>Specifies the total number of SCS connections that are configured for use by all system applications.</p> <p>Normally, the default value is adequate.</p>
SCSMAXMSG	<p>Specifies the SCS maximum sequenced message size.</p> <p>Normally, the default value is adequate.</p>
SCSMAXDG	<p>Specifies the maximum number of bytes of application data in one datagram.</p> <p>Normally the default value is adequate.</p>

# Cluster SYSGEN Parameters

**Table A-1 (Cont.) Cluster SYSGEN Parameters**

<b>Parameter</b>	<b>Description</b>
SCSFLOWCUSH	Specifies the lower limit for receive buffers at which point SCS starts to notify the remote SCS of new receive buffers. For each connection, SCS tracks the number of receive buffers available. SCS communicates this number to the SCS at the remote end of the connection. However, SCS does not need to do this for each new receive buffer added. Instead, SCS notifies the remote SCS of new receive buffers if the number of receive buffers falls as low as the SCSFLOWCUSH value. Normally the default value is adequate.
SCSSYSTEMID	Specifies the lower-order 32 bits of the 48-bit system identification number. This parameter is not dynamic and must be the same as the DECnet node number ( $1024 * \langle \text{DECnet area} \rangle + \text{DECnet node number}$ ).
SCSSYSTEMIDH	Specifies the high-order 16 bits of the 48 bit system identification number. This parameter must be set to 0. It is reserved by DIGITAL for future use.
SCSNODE	Specifies the SCS system name. This parameter is not dynamic. You should use a name that is the same as the DECnet node name (limited to six characters) since the name must be unique among all systems in the cluster.  Note that once a node has been recognized by another node in the cluster, you cannot change the SCSSYSTEMID or SCSNODE parameter without changing both.
SCSRESPCNT	Specifies the total number of response descriptor table entries configured for use by all system applications.

# B

## Building a Common SYSUAF.DAT File from Node-Specific Files

This appendix provides guidelines for building a common user authorization file from node-specific files. For more detailed information on how to set up a node-specific authorization file, see the descriptions in the *VMS Authorize Utility Manual* and in the *Guide to Setting Up a VMS System*.

To build a common SYSUAF.DAT file, proceed as follows: steps.

- 1 Print a listing of SYSUAF.DAT on each node. To print this listing, invoke AUTHORIZE and specify the AUTHORIZE command LIST as follows:

```
§ SET DEF SYS$SYSTEM
§ RUN AUTHORIZE
UAF> LIST/FULL [*,*]
```

- 2 Use the listings to compare the accounts from each node. On the listings, mark down any necessary changes.

One such change is to delete any accounts that you no longer need. You should also make sure that each user account in the cluster has a unique UIC.

For example, node VENUS of the cluster may have a user account JONES that has the same UIC as user account SMITH on node MARS. When nodes VENUS and MARS are joined to form a cluster, accounts JONES and SMITH will exist in the cluster environment with the same UIC. If the UICs of these accounts are not differentiated, each user will have the same access rights to various objects in the cluster. In this case you should assign each account a unique UIC.

Make sure that accounts that perform the same type of work have the same group UIC. Accounts in a single-system environment probably follow this convention. However, there may be groups of users on each node that will perform the same work in the cluster but have group UICs unique to their local node. As a rule, the group UIC for any given work category should be the same on each node in the cluster. For example, data entry accounts on node VENUS should have the same group UIC as data entry accounts from node MARS and node RED.

Note that if you change the UIC for a particular user, you should also change the owner UICs for that user's existing files and directories. You can use the DCL commands SET FILE and SET DIRECTORY to make these changes. These commands are described in detail in the *VMS DCL Dictionary*.

- 3 Choose the SYSUAF.DAT from one of the nodes to be a master SYSUAF.DAT.
- 4 Merge the SYSUAF.DAT files from the other nodes to the master SYSUAF.DAT by running the Convert Utility (CONVERT) on the node that owns the master SYSUAF.DAT. (See the *VMS Convert and Convert/Reclaim Utility Manual* for a description of CONVERT.) To use CONVERT to merge the files, each SYSUAF.DAT file must be accessible to the node that is running CONVERT.

## Building a Common SYSUAF.DAT File from Node-Specific Files

To merge the UAFs into the master SYSUAF.DAT file, specify the CONVERT command in the following format:

```
§ CONVERT SYSUAF1,SYSUAF2,...SYSUAFn MASTER_SYSUAF
```

Note that if a given username appears in more than one source file, only the first occurrence of that name will appear in the merged file.

The command in the following example adds the SYSUAF.DAT file from two cluster nodes to the master SYSUAF.DAT in the current default directory:

```
§ SET DEFAULT SYS$SYSTEM
§ CONVERT [SYS1.SYSEXE]SYSUAF.DAT, [SYS2.SYSEXE]SYSUAF.DAT SYSUAF.DAT
```

The CONVERT command in this example adds the records from the files [SYS1.SYSEXE]SYSUAF.DAT and [SYS2.SYSEXE]SYSUAF.DAT to the file SYSUAF.DAT on the local node.

After you run CONVERT, you are left with a master SYSUAF.DAT that contains records from the other SYSUAF.DAT files.

- 5 Use AUTHORIZE to modify the accounts in the master SYSUAF.DAT according to the changes you marked on the initial listings of the SYSUAF.DAT files from each node.

---

# C VAXcluster Troubleshooting Information

This appendix contains information to help you perform troubleshooting operations for the following:

- Failures of nodes to boot or to join the cluster
- Cluster hangs
- CLUEXIT bugchecks
- VAXport device problems

---

## C.1 Diagnosing Failures of Nodes to Boot or to Join the Cluster

Before you initiate diagnostic procedures, be sure to verify that these conditions are met:

- All cluster hardware components are correctly connected and checked for proper operation.
- Cluster nodes and mass storage devices are configured according to requirements specified in the VAXcluster Software Product Description (SPD) document.

When attempting to add a new or recently repaired CI-connected node to the cluster, you must verify that the CI cables are correctly connected, as described in Section C.4.2.2.

When attempting to add a satellite node to a local area or mixed-interconnect cluster, you must verify that the Ethernet is configured according to requirements specified in the VAXcluster SPD document, and that the machine's memory resources and Ethernet adapter device meet the requirements specified in that document. You must also verify that you have correctly configured and started the DECnet-VAX network, following the procedures described in Section 2.3.

If after performing preliminary checks and taking appropriate corrective action, you find that a node still fails to boot or to join the cluster, you can follow the procedures in Sections C.1.2 through C.1.4 to attempt recovery.

---

### C.1.1 Summary of Events for Nodes Booting and Joining the Cluster

To perform diagnostic and recovery procedures effectively, you must understand the events that occur when a node boots and attempts to join the cluster. This section outlines those events and shows typical messages displayed at the console.

Note that events vary, depending on whether a node is the first node to boot in a new cluster or whether it is booting in an active cluster. Note further that some events (such as loading the cluster security database) occur only in local area and mixed-interconnect clusters.

# VAXcluster Troubleshooting Information

## C.1 Diagnosing Failures of Nodes to Boot or to Join the Cluster

The normal sequence of events is as follows:

- 1 The node boots. If the node is a satellite, a message like the following shows the name and Ethernet address of the boot server that has downline loaded the satellite:

```
%VAXcluster-I-SYSLOAD, system loaded from node X... (XX-XX-XX-XX-XX-XX)
```

For any booting node, the VMS "banner message" is displayed in the following format:

```
VAX/VMS Version n.n DD-MMM-YYYY hh:mm:ss
```

- 2 The node attempts to form or join the cluster, and the following message appears:

```
waiting to form or join a VAXcluster system
```

If the node is a member of a local area or mixed-interconnect cluster, the cluster security database is loaded. Optionally, the MSCP Server may be loaded:

```
%VAXcluster-I-LOADSECDB, loading the cluster security database  
%MSCPLOAD-I-LOADMSCP, loading the MSCP disk server
```

- 3 If the node discovers a cluster, the node attempts to join. If a cluster is found, the Connection Manager displays one or more messages in the following format:

```
%CNXMAN, Sending VAXcluster membership request to system X...
```

Otherwise, the Connection Manager forms the cluster when it has enough votes to establish quorum (that is, when enough voting nodes have booted).

- 4 As the booting node joins the cluster, the Connection Manager displays a message in the following format:

```
%CNXMAN, now a VAXcluster member -- system X...
```

Note that if quorum is lost while the node is booting, or if a node is unable to join the cluster within two minutes of booting, the Connection Manager displays messages like the following:

```
%CNXMAN, Discovered system X...  
%CNXMAN, Deleting CSB for system X...  
%CNXMAN, Established "connection" to quorum disk  
%CNXMAN, Have connection to system X...  
%CNXMAN, Have "connection" to quorum disk
```

The last two messages show any connections that have already been formed.

If the cluster includes a quorum disk, you may also see messages like the following:

```
%CNXMAN, Using remote access method for quorum disk  
%CNXMAN, Using local access method for quorum disk
```

The first message indicates that the Connection Manager is unable to access the quorum disk directly, either because the disk is unavailable, or because it is accessed through the MSCP Server. Another node in the cluster that can access the disk directly must verify that a reliable connection to the disk exists.

# VAXcluster Troubleshooting Information

## C.1 Diagnosing Failures of Nodes to Boot or to Join the Cluster

The second message indicates that the Connection Manager can access the quorum disk directly and can supply information about the status of the disk to nodes that cannot access the disk directly.

Note that the Connection Manager may not see the quorum disk initially, because the disk may not yet be configured. In that case, the Connection Manager first uses remote access, then switches to local access.

- 5 Once the node has joined the cluster, normal startup procedures execute. One of the first functions is to start the OPCOM process:

```
%%%%%%%%%% OPCOM 15-APR-1988 16:33:55.33 %%%%%%%%%%%  
Logfile has been initialized by operator _X...$OPAO:  
Logfile is SYS$SYSROOT:[SYSMGR]OPERATOR.LOG;17
```

```
%%%%%%%%%% OPCOM 15-APR-1988 16:33:56.43 %%%%%%%%%%%  
16:32:32.93 Node X... (csid 0002000E) is now a VAXcluster member
```

When other nodes join the cluster, OPCOM displays messages like the following:

```
%%%%%%%%%% OPCOM 15-APR-1988 16:34:25.23 %%%%%%%%%%% (from node X... at 16:34:25.23)  
16:34:24.42 Node X... (csid 000100F3) received VAXcluster membership request from node X...
```

As startup procedures continue, various messages report startup events.

**Note:** For troubleshooting purposes, you may want to include in your site-specific startup procedures messages announcing each phase of the the startup process—for example, mounting disks or starting queues.

### C.1.2 CI-Connected Node Fails to Boot

If a CI-connected node fails to boot, perform the following checks:

- Verify that the node's SCSNODE and SYSSYSTEMID parameters are unique in the cluster. If they are not, you must either alter *both* values or reboot all other nodes.
- Verify that you are using the correct bootstrap command file. This file must specify the internal bus node number (if applicable), the HSC node number, and the HSC disk from which the node is to boot. Refer to your processor-specific installation and operations guide for information on setting values in default bootstrap command procedures.
- Verify that the SYSGEN parameter PAMAXPORT is set to a value greater than or equal to the largest CI port number.
- Verify that the HSC is ONLINE. The ONLINE switch on the HSC Operator Control Panel should be depressed.
- Verify that the disk is available. The correct port switches on the disk's operator control panel should be depressed.
- Verify that the node has access to the HSC. The SHOW HOSTS command of the HSC SETSHO Utility displays status for all VAX nodes (hosts) in the cluster. (For complete information on the SETSHO Utility, consult the HSC hardware documentation.) If the node in question appears in the display as DISABLED, use the SETSHO Utility to set the node to the ENABLED state.

# VAXcluster Troubleshooting Information

## C.1 Diagnosing Failures of Nodes to Boot or to Join the Cluster

- Verify that the HSC allows access to the boot disk. Invoke the SETSHO Utility to ensure that the boot disk is available to the HSC. The utility's SHOW DISKS command displays the current state of all disks visible to the HSC and displays all disks in the no-host-access table. If the boot disk appears in the no-host-access table, use the SETSHO Utility to set the boot disk to host-access. If the boot disk is AVAILABLE or MOUNTED and host-access ENABLED, but does not appear in the no-host-access table, contact your Field Service representative and explain both the problem and the steps you have taken.

### C.1.3 Satellite Node Fails to Boot

To boot successfully, a satellite must communicate with a boot server over the Ethernet. You can use DECnet event logging to verify this communication. Proceed as follows:

- 1 Log in as system manager on the boot server.
- 2 If event logging for management layer events is not already enabled, enter the following NCP commands to enable it:

```
NCP> SET LOGGING MONITOR EVENT 0.*  
NCP> SET LOGGING MONITOR STATE ON
```

- 3 Enter the following DCL command:

```
§ REPLY/ENABLE=NETWORK
```

This command enables the terminal to receive DECnet messages reporting downline load events.

- 4 Boot the satellite. If the satellite and the boot server can communicate, and if all boot parameters are correctly set, messages like the following are displayed at the boot server's terminal:

```
DECnet event 0.3, automatic line service  
From node 2.4 (URANUS), 15-APR-1988 09:42:15.12  
Circuit QNA-0, Load, Requested, Node = 2.42 (OBERON)  
File = SYS$SYSDEVICE:<SYS10.>, Operating system  
Ethernet address = 08-00-2B-07-AC-03
```

```
DECnet event 0.3, automatic line service  
From node 2.4 (URANUS), 15-APR-1988 09:42:16.76  
Circuit QNA-0, Load, Successful, Node = 2.42 (ARIEL)  
File = SYS$SYSDEVICE:<SYS11.>, Operating system  
Ethernet address = 08-00-2B-07-AC-13
```

If the satellite cannot communicate with the boot server, no message for that satellite appears. There may be a problem with an Ethernet cable connection or adapter service.

If the satellite's data in the DECnet database is incorrectly specified (for example, if the hardware address is incorrect), a message like the following displays the correct address and indicates that a load was requested:

```
DECnet event 0.7, aborted service request  
From node 2.4 (URANUS), 15-APR-1988 09:42:09.67  
Circuit QNA-0, Line open error, Ethernet address = 08-00-2B-03-29-99
```

Note the absence of the node name, node address, and system root.

# VAXcluster Troubleshooting Information

## C.1 Diagnosing Failures of Nodes to Boot or to Join the Cluster

If a satellite fails to boot, perform the following checks:

- Verify that the boot device is available. This check is particularly important for local area and mixed-interconnect clusters in which satellites boot from multiple system disks.
- Verify that the satellite's SCSNODE and SCSSYSTEMID values and its DECnet node name and address are unique in the cluster.
- Verify that the DECnet-VAX network is up and running.
- Verify that circuit service is enabled for the boot server's Ethernet adapter device. Invoke the NCP Utility and enter an NCP command in the following format, where circuit-id is the name of the Ethernet adapter circuit that the boot server uses to service downline load requests from satellites:

```
NCP> SHOW CIRCUIT circuit-id
```

If service is not enabled, you can enter NCP commands like the following to enable it:

```
NCP> SET CIRCUIT circuit-id STATE OFF
NCP> DEFINE CIRCUIT circuit-id SERVICE ENABLED
NCP> SET CIRCUIT circuit-id SERVICE ENABLED STATE ON
```

The DEFINE command updates the permanent database and ensures that service is enabled the next time you start the network. Note that DECnet traffic will be interrupted while the circuit is off.

- Verify that you have specified the correct Ethernet hardware address for the satellite. Proceed as follows:

- 1 Enter an NCP command in the following format on the boot server, specifying the satellite's node name:

```
NCP> SHOW NODE X... CHARACTERISTICS
```

The system displays data like the following:

```
Node Volatile Characteristics as of 15-APR-1988 13:15:28
```

```
Remote node = 2.41 (ARIEL)
```

```
Hardware address      = 08-00-2B-03-27-95
Tertiary loader       = SYS$SYSTEM:TERTIARY_VMB.EXE
Load Assist Agent     = SYS$SHARE:NISCS_LAA.EXE
Load Assist Parameter = DISK$VAXVMSRL5:<SYS12.>
```

- 2 At the satellite's console prompt (> > > ), enter the commands shown in Table 3-1 to display the satellite's current Ethernet hardware address.
- 3 Compare the hardware address values displayed by NCP and at the satellite's console. The values should be identical and should also match the value shown in the file SYS\$MANAGER:NETNODE\_UPDATE.COM. If the values do not match, you must make appropriate adjustments. For example, if you have recently replaced the satellite's Ethernet adapter device, you must execute CLUSTER\_CONFIG's CHANGE function to update the network database and NETNODE\_UPDATE.COM on the appropriate boot server.

# VAXcluster Troubleshooting Information

## C.1 Diagnosing Failures of Nodes to Boot or to Join the Cluster

- Verify that the satellite's load assist parameter specifies the correct device and root directory name and that the satellite's root is unique in the cluster. If changes are needed, you can use CLUSTER\_CONFIG.COM to remove the satellite and then add it again with correct values.

### C.1.4 Node Fails to Join the Cluster

If a node boots but fails to join the cluster, proceed as follows:

- Verify that VAXcluster software has been loaded. Look for Connection Manager (%CNXMAN) messages like those shown in Section C.1.1. If no such messages are displayed, it is likely that VAXcluster software was not loaded at boot time. Reboot the node in conversational mode. At the SYSBOOT> prompt, set the VAXCLUSTER parameter to 2. (In local area or mixed-interconnect clusters, you must also set NISCS\_LOAD\_PEA0 to 1.) Note that these parameters should also be set in the node's MODPARAMS.DAT file. For more information on booting a node in conversational mode, consult your processor-specific installation and operations guide.

In local area and mixed-interconnect clusters, verify that the cluster security database file (SYS\$COMMON:CLUSTER\_AUTHORIZE.DAT) exists and that you have specified the correct group number for this cluster.

- Verify that the node has booted from the correct disk and system root. If %CNXMAN messages are displayed, and if after the conversational reboot the node still does not join the cluster, check the console output on all active cluster nodes and look for messages indicating that one or more nodes found a remote system that conflicted with a known or local system. Such messages suggest that two nodes have booted from the same system root.

Review the boot command files for all CI-connected nodes and ensure that all are booting from the correct disks and from unique system roots. If you find it necessary to modify the node's bootstrap command procedure (console media), you may be able to do so on another processor that is already running in the cluster. Replace the running processor's console media with the media to be modified, and use the Exchange Utility and a text editor to make the required changes. Consult the appropriate processor-specific installation and operations guide for information on examining and editing boot command files.

- Verify that the node's SCSNODE and SCSSYSTEMID parameters are unique in the cluster. To be eligible to join a cluster, a node must have unique SCSNODE and SYSSYSTEMID parameter values. Check that the current values do not duplicate any values set for existing cluster nodes. Note that if you discover that one or the other value is not unique, you must alter *both* values or reboot all other cluster nodes. To check or modify values, you can perform a conversational bootstrap operation. However, for reliable future bootstrap operations, you must specify appropriate values for these parameters in the node's MODPARAMS.DAT file.

# VAXcluster Troubleshooting Information

## C.1 Diagnosing Failures of Nodes to Boot or to Join the Cluster

---

### C.1.5 Startup Procedures Fail to Complete

If a node boots and joins the cluster but appears to hang before startup procedures complete—that is, before you are able to log in to the system, be sure that you have allowed sufficient time for the startup procedures to execute.

If the startup procedures fail to complete after a period that is normal for your site, try to access the procedures from another cluster node and make appropriate adjustments. For example, verify that all required devices are configured and available.

One potential cause of such a failure is the lack of some system resource such as NPAGEDYN or page file space. If you suspect that the value for the NPAGEDYN parameter is set too low, you can perform a conversational bootstrap operation to increase it. Use SYSBOOT to check the current value, and then double the value. If this procedure is unsuccessful, double the value once more.

If you suspect a shortage of page file space, and if another cluster node is available, you can log in on that node and use the System Generation Utility (SYSGEN) to provide adequate page file space for the problem node. (Note that insufficient page file space on the booting node may cause other nodes to hang.) If the node still cannot complete the startup procedures, contact your DIGITAL Field Service Representative.

---

## C.2 Diagnosing Cluster Hangs

Conditions like the following can cause a VAXcluster member system to suspend process or system activity—that is, to hang:

- Cluster quorum is lost.
- A shared cluster resource is inaccessible.

Sections C.2.1 and C.2.2 discuss these conditions.

---

### C.2.1 Cluster Quorum Is Lost

The VAXcluster quorum scheme coordinates activity among cluster member systems and ensures the integrity of shared cluster resources. (The quorum scheme is described fully in Section 1.5.1.) Quorum is checked after any change to the cluster configuration—for example, when a voting node leaves or joins the cluster. If quorum is lost, process creation and I/O activity on all nodes in the cluster are blocked.

Information about the loss of quorum and clusterwide events that cause loss of quorum are sent to the OPCOM process, which broadcasts messages to designated operator terminals. The information is also broadcast to each cluster node's operator console (OPA0), unless broadcast activity is explicitly disabled on that terminal. Because, however, quorum may be lost before OPCOM has been able to inform the operator terminals, the messages sent to OPA0 are the most reliable source of information about events that may cause loss of quorum.

If quorum is lost, you can follow instructions in Section 3.4.4 to recover.

# VAXcluster Troubleshooting Information

## C.2 Diagnosing Cluster Hangs

---

### C.2.2 A Shared Cluster Resource Is Inaccessible

Access to shared cluster resources is coordinated by the Distributed Lock Manager. If a particular process is granted a lock on a resource (for example, a shared data file), other processes in the cluster that request incompatible locks on that resource must wait until the original lock is released. If the original process retains its lock for an extended period, other processes waiting for the lock to be released may appear to hang.

Occasionally a system activity must acquire a restrictive lock on a resource for an extended period. For example, to perform a volume rebuild, system software takes out an exclusive lock on the volume being rebuilt. While this lock is held, no processes can allocate space on the disk volume. If they attempt to do so, they may appear to hang.

Access to files that contain data necessary for the operation of the system itself is coordinated by the Distributed Lock Manager. For this reason, a process that acquires a lock on one of these resources and is then unable to proceed may cause the cluster to appear to hang.

For example, this condition may occur if a process locks a portion of the system authorization file (SYS\$SYSTEM:SYSUAF.DAT) for write access. Any activity that requires access to that portion of the file, such as logging into an account with the same or similar username or sending mail to that username, will be blocked until the original lock is released. Normally this lock would be released quickly, and users would not notice the locking operation.

However, if the process holding the lock is itself unable to proceed, other processes could enter a wait state. Because the authorization file is used during login and for most process creation operations (for example, batch and network jobs) blocked processes could rapidly accumulate in the cluster. Because the Distributed Lock Manager is functioning normally under these conditions, users are not notified by broadcast messages or other means that a problem has occurred.

---

## C.3 Diagnosing CLUEXIT Bugchecks

The VMS operating system performs *bugcheck* operations only when it detects conditions that could compromise normal system activity or endanger data integrity. A *CLUEXIT bugcheck* is a type of bugcheck initiated by the Connection Manager, the VAXcluster software component that manages the interaction of cooperating VAXcluster member systems. Most such bugchecks are triggered by conditions resulting from hardware failures (particularly failures in communications paths), configuration errors, or system management errors.

The conditions that most commonly result in CLUEXIT bugchecks are as follows:

- The cluster connection between two nodes is broken for longer than RECNXINTERVAL seconds. Thereafter, the connection is declared irrevocably broken. If the connection is later reestablished, either or both of the nodes shut down with a CLUEXIT bugcheck.

This condition can occur upon power failure recovery with battery backup, after the repair of an SCS communication link, or after the node was halted for a period longer than RECNXINTERVAL seconds, and was restarted with a CONTINUE command entered at the operator console. You must determine the cause of the interrupted connection and

# VAXcluster Troubleshooting Information

## C.3 Diagnosing CLUEXIT Bugchecks

correct the problem. For example, if powerfail recovery is longer than RECNXINTERVAL seconds, you may want to increase the value of the RECNXINTERVAL parameter on all nodes.

- Cluster partitioning occurs. A member of a cluster discovers or establishes connection to a member of another cluster, or a foreign cluster is detected in the quorum file. In this case, you must review the setting of EXPECTED\_VOTES on all nodes.
- The value specified for the SYSGEN parameter SCSMAXMSG on a node is too small. Verify that the value of SCSMAXMSG on all cluster nodes is set to a value that is at the least the default value.

---

## C.4 Diagnosing VAXport Device Problems

The following sections present information on the CI and Ethernet VAXport devices. Information is also provided on entries in the system error log and on corrective actions to take when errors occur. Topics include the following:

- VAXport communication mechanisms
- Port failures
- VAXcluster error log entries
- OPA0 error messages

---

### C.4.1 VAXport Communication Mechanisms

This section describes CI and Ethernet port communication mechanisms and System Communications Services (SCS) connections.

#### Port Polling

Shortly after a CI-connected system boots, the CI port driver (PADRIVER) begins configuration polling to discover other active ports on the CI. Normally the poller runs every five seconds (the default value of the SYSGEN parameter PAPOLLINT). In the first polling pass, all addresses are probed over cable path A; on the second pass all addresses are probed over path B; on the third pass path A is probed again, and so on.

The poller probes by sending request id (REQID) packets to all possible port numbers, including itself. Active ports receiving the REQIDs return id packets (IDREC) to the port issuing the REQID. A port may respond to a REQID even if the system attached to the port is not running.

In any CI-only, local area, or mixed-interconnect cluster, the port drivers perform a start handshake when a pair of ports and port drivers has successfully exchanged id packets. The port drivers exchange datagrams containing information about the systems, such as the type of CPU and the operating system version. If this exchange is successful, each system declares a virtual circuit open. An open virtual circuit is prerequisite to all other activity.

# VAXcluster Troubleshooting Information

## C.4 Diagnosing VAXport Device Problems

### Ethernet Communications

In local area and mixed-interconnect clusters, a multicast scheme is used to locate cluster nodes on the Ethernet. Every three seconds the Port Emulator driver (PEDRIVER) sends HELLO messages to a cluster-specific multicast address that is derived from the cluster group number. The driver also enables the reception of these messages from other nodes. When the driver receives a HELLO message from a node with which it does not currently share an open virtual circuit, it attempts to create a circuit. HELLO messages received from a node with a currently open virtual circuit indicate that the remote node is operational.

A standard three-message exchange handshake is used to create a virtual circuit. The handshake messages contain information about the transmitting node and its record of the cluster password. These parameters are verified at the receiving system, which continues the handshake only if its verification is successful. Thus, each node authenticates the other. After the final message, the virtual circuit is opened for use by both nodes.

### System Communications Services (SCS) Connections

System services such as the disk class driver, the VAXcluster Connection Manager, and the MSCP Server communicate between nodes with a protocol called System Communications Services (SCS). Primarily, SCS is responsible for the formation and breaking of intersystem process connections and for flow control of message traffic over those connections. In VMS Version 5.0, SCS is implemented in the VAXport driver (for example, PADRIVER, PBDRIVER, PEDRIVER), and in a loadable piece of the system called SCSLOA.EXE (loaded automatically during system initialization).

When a virtual circuit has been opened, a VMS system periodically probes a remote node for system services that the remote system may be offering. The SCS directory service, which makes known services that a node is offering, is always present on both VMS and HSC systems. As system services discover their counterparts on other systems, they establish SCS connections to each other. These connections are full duplex and are associated with a particular virtual circuit. Multiple connections are typically associated with a virtual circuit.

---

### C.4.2 Port Failures

Taken together, SCS, the VAXport drivers, and the port itself support a hierarchy of communications paths. Working up from the most fundamental level, these are as follows:

- The physical wires. The Ethernet is a single coaxial cable. The CI has two pairs of transmit and receive cables (Path A transmit and receive and Path B transmit and receive). For the CI, VMS software normally sends traffic in automatic path select mode. The port chooses the free path or, if both are free, an arbitrary path (implemented in the cables and Star Coupler, and managed by the port).
- The virtual circuit (implemented partly in the CI port or Ethernet Port Emulator driver (PEDRIVER) and partly in SCS software).
- The SCS connections (implemented in system software).

# VAXcluster Troubleshooting Information

## C.4 Diagnosing VAXport Device Problems

Failures can occur at each communications level and in each component. Failures at one level translate into failures at other levels as follows:

- **Wires.** If the Ethernet fails or is disconnected, Ethernet traffic stops or is interrupted, depending on the nature of the failure. For the CI, either Path A or B can fail while the virtual circuit remains intact. All traffic is directed over the remaining good path. When the wire is repaired, the repair is detected automatically by port polling, and normal operations resume on all ports.
- **Virtual circuit.** If no path works between a pair of ports, the virtual circuit fails and is closed. A path failure is discovered as follows:
  - For the CI, when polling fails, or when attempts are made to send normal traffic, and the port reports that neither path yielded transmit success.
  - For the Ethernet, when no multicast HELLO message or incoming traffic is received from another node.

When a virtual circuit fails, every SCS connection on it fails. The software automatically reestablishes connections when the virtual circuit is reestablished. Normally, reestablishing a virtual circuit takes several seconds after the problem is corrected.

- **CI port.** If a port fails, all virtual circuits to that port fail, and all SCS connections on those virtual circuits fail. If the port is successfully reinitialized, virtual circuits and connections are reestablished automatically. Normally, port reinitialization and reestablishment of connections take several seconds.
- **Ethernet adapter.** If an Ethernet adapter device fails, attempts are made to restart it. If repeated attempts fail, all virtual circuits time out, and their connections are broken.
- **SCS connection.** When the software protocols fail or, in some instances, when the software detects a hardware malfunction, a connection is terminated. Other connections are normally unaffected, as is the virtual circuit. Breaking of connections is also used under certain conditions as an error recovery mechanism—most commonly when there is insufficient nonpaged pool available on the system.
- **System.** If a system fails because of operator shutdown, bugcheck, or halt and reboot, all other systems in the cluster record the failure as failures of their virtual circuits to the port on the failed system.

---

### C.4.2.1 Verifying CI Port Functions

Before you boot in a cluster a CI-connected system that is new, just repaired, or suspected of having a problem, you should have DIGITAL Field Service verify that the system runs correctly on its own.

To diagnose communication problems, you can invoke the Show Cluster Utility and tailor the SHOW CLUSTER report by entering the SHOW CLUSTER command ADD CIRCUIT CABLE\_ST. This command adds a class of information about all the virtual circuits as seen from the system on which you are running SHOW CLUSTER. Primarily, you are checking whether there is a virtual circuit in the OPEN state to the failing system. Common causes of failure to open a virtual circuit and keep it open are the following:

# VAXcluster Troubleshooting Information

## C.4 Diagnosing VAXport Device Problems

- Port errors on one side or the other
- Cabling errors
- A port set off line because of software problems
- Insufficient nonpaged pool available on both sides
- Failure to set correct values for the SYSGEN parameters SCSNODE, SCSSYSTEMID, PAMAXPORT, PANOPOLL, PASTIMOUT, and PAPOLLINT.

Run SHOW CLUSTER from each active system in the cluster to verify whether each system's view of the failing system is consistent with every other system's view. If all the active systems have a consistent view of the failing system, the problem may be in the failing system. If, on the other hand, only one of several active systems detects that the newcomer is failing, that particular system may be experiencing a problem.

If no virtual circuit is open to the failing system, check the bottom of the SHOW CLUSTER display for information on circuits to the port of the failing system. Virtual circuits in partially open states are shown at the bottom of the display. If the circuit is shown in a state other than OPEN, communications between the local and remote ports are taking place, and the failure is probably at a higher level than in port or cable hardware. Next, check that both Paths A and B are good to the failing port. The loss of one path should not prevent a system from participating in a cluster.

---

### C.4.2.2 Verifying CI Cable Connections

Whenever the configuration poller finds that no virtual circuits are open and that no handshake procedures are currently opening virtual circuits, the poller analyzes its environment. It does so by using the send-loopback-datagram facility of the CI port.

The send-loopback-datagram facility tests the connections between the CI port and the Star Coupler by routing messages across them. The messages are called loopback datagrams. (The port processes other self-directed messages without using the Star Coupler or external cables.)

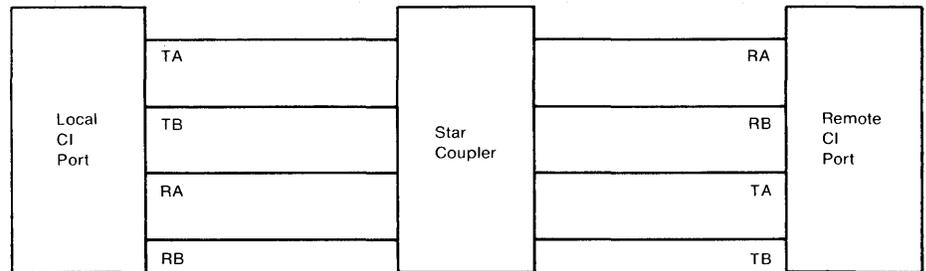
The configuration poller makes entries in the error log whenever it detects a change in the state of a circuit. Note, however, that it is possible for two changed-to-failed-state messages to be entered in the log without an intervening changed-to-succeeded-state message. Such a series of entries means that the circuit state continues to be faulty.

The following paragraphs discuss various incorrect CI cabling configurations and the entries made in the error log when these configurations exist. Figure C-1 shows a two-node configuration with all cables correctly connected. Figure C-2 shows a CI cluster with a pair of crossed cables.

# VAXcluster Troubleshooting Information

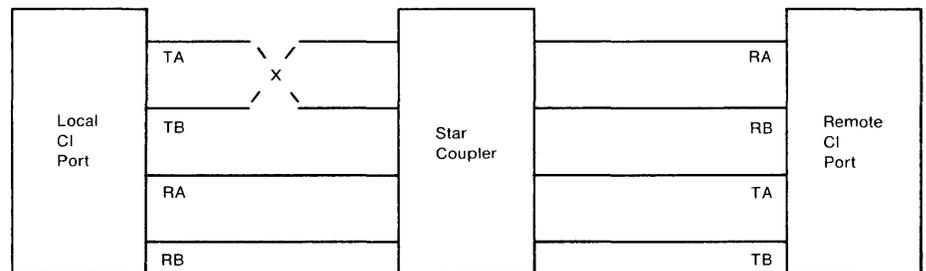
## C.4 Diagnosing VAXport Device Problems

**Figure C-1 A Correctly Connected Two-Node CI Cluster**



ZK-1924-84

**Figure C-2 Crossed CI Cable Pair**



ZK-1925-84

If a pair of transmitting cables or a pair of receiving cables is crossed, a message sent on TA is received on RB, and a message sent on TB is received on RA. This is a hardware error condition from which the port cannot recover. An entry is made in the error log to say that a single pair of crossed cables exists. The entry contains the following lines:

```
DATA CABLE(S) CHANGE OF STATE
PATH 1. LOOPBACK HAS GONE FROM GOOD TO BAD
```

If this situation exists, you can correct it by reconnecting the cables properly. The cables could be misconnected in several places. The coaxial cables that connect the port boards to the bulkhead cable connectors can be crossed, or the Ethernet cables can be misconnected to the bulkhead or the Star Coupler.

The information in Figure C-2 can be represented more simply. Configuration 1 shows the cables positioned as in Figure C-2, but it does not show the star coupler or the nodes. The letters LOC and REM indicate the pairs of transmitting (T) and receiving (R) cables on the local and remote nodes, respectively.

# VAXcluster Troubleshooting Information

## C.4 Diagnosing VAXport Device Problems

### Configuration 1

```
T x   = R
R =   = T
LOC   REM
```

The pair of crossed cables causes loopback datagrams to fail on the local node, but succeed on the remote node. Crossed pairs of transmitting cables and crossed pairs of receiving cables cause the same behavior.

Note that only an odd number of crossed-cable pairs causes these problems. If an even number of cable pairs is crossed, communications succeed. An error log entry is made in some cases, however, and the contents of the entry depends on which pairs of cables are crossed.

Configuration 2 shows two-node clusters with the combinations of two pairs of crossed-cable pairs. These crossed pairs cause the following entry to be made in the error log of the node that has the cables crossed:

```
DATA CABLE(S) CHANGE OF STATE
CABLES HAVE GONE FROM UNCROSSED TO CROSSED
```

Loopback datagrams succeed on both nodes, and communications are possible.

### Configuration 2

```
T x   = R      T =   x R
R x   = T      R =   x T
LOC   REM      LOC   REM
```

Configuration 3 shows the possible combinations of two pairs of crossed cables that cause loopback datagrams to fail on both nodes in the cluster. Communications can still take place between the nodes. An entry stating that cables are crossed is made in the error log of each node.

### Configuration 3

```
T x   = R      T =   x R
R =   x T      R x  = T
LOC   REM      LOC   REM
```

Configuration 4 shows the possible combinations of two pairs of crossed cables that cause loopback datagrams to fail on both nodes in the cluster, but allow communications. No entry stating that cables are crossed is made in the error log of either node.

### Configuration 4

```
T x   x R      T =   = R
R =   = T      R x  x T
LOC   REM      LOC   REM
```

# VAXcluster Troubleshooting Information

## C.4 Diagnosing VAXport Device Problems

Configuration 5 shows the possible combinations of three pairs of crossed cables. In each case, loopback datagrams fail on the node that has only one crossed pair of cables. Loopback datagrams succeed on the node with both pairs crossed. No communications are possible.

### Configuration 5

T x	x R	T x	= R	T =	x R	T x	x R
R x	= T	R x	x T	R x	x T	R =	x T
LOC	REM	LOC	REM	LOC	REM	LOC	REM

If all four cable pairs between two nodes are crossed, communications succeed, loopback datagrams succeed, and no crossed-cable message entries are made in the error log. Such a condition might be detected by noting error log entries made by a third system in the cluster, but only if the third node has one of the crossed-cable cases described.

---

### C.4.2.3 Repairing CI Cables

This section describes some ways in which DIGITAL Field Service can make repairs on a running system. This information is provided to aid system managers in scheduling repairs.

For cluster software to survive cable-checking activities or cable-replacement activities, you must be sure that either Path A or Path B is intact at all times between each port and between every other port in the cluster.

You can, for example, remove Path A and Path B in turn from a particular port to the Star Coupler. To make sure that the configuration poller finds a path that was previously faulty but is now operational, follow these steps:

- 1 Remove Path B.
- 2 After the poller has discovered that Path B is faulty, reconnect Path B.
- 3 Wait two poller intervals, and then enter the DCL command `SHOW CLUSTER` to make sure that the poller has reestablished Path B. Or, enter the DCL command `SHOW CLUSTER/CONTINUOUS` followed by the `SHOW CLUSTER` command `ADD CIRCUITS, CABLE_ST`. Wait until `SHOW CLUSTER` tells you that Path B has been reestablished.
- 4 Remove Path A.
- 5 After the poller has discovered that Path A is faulty, reconnect Path A.
- 6 Wait two poller intervals to make sure that the poller has reestablished Path A.

If both paths are lost at the same time, the virtual circuits are lost between the port with the broken cables and all other ports in the cluster. This condition will in turn result in loss of SCS connections over the broken virtual circuits. However, recovery from this situation is automatic after an interruption in service on the affected node. The length of the interruption varies, but it is usually approximately two poller intervals (or 10 seconds) at the default `SYSGEN` parameter settings.

# VAXcluster Troubleshooting Information

## C.4 Diagnosing VAXport Device Problems

---

### C.4.3 Analyzing Error Log Entries for VAXport Devices

To anticipate and avoid potential problems, you must monitor events recorded in the error log. From the total error count, displayed by a DCL command in the format `SHOW DEVICE device-name`, you can determine whether errors are increasing. If so, you should examine the error log.

The DCL command `ANALYZE/ERROR_LOG` invokes the Error Log Utility to report the contents of an error log file. (For more information on the Error Log Utility, see the *VMS Error Log Utility Manual*.)

Note that some error log entries are informational only, and require no action. For example, if you shut down a system in the cluster, all other active systems that have open virtual circuits between themselves and the system that has been shut down make entries in their error logs. Such systems record up to three errors for the event: Path A received no response; Path B received no response; the virtual circuit is being closed. These messages are normal and reflect the change of state in the circuits to the system that has been shut down.

On the other hand, some error log entries are made for problems that degrade operation, or for nonfatal hardware problems. The VMS operating system might continue to run satisfactorily under these conditions. The purpose of detecting these problems early is to prevent nonfatal problems (such as loss of a single CI path) from becoming serious problems (such as loss of both paths).

---

#### C.4.3.1 Error Log Entry Formats

Errors and other events on the CI or Ethernet cause VAXport drivers to enter information in the system error log. The two formats used for error log entries are the *device-attention* format and the *logged-message* format. Sections C.4.3.2 and C.4.3.3 describe those formats.

Device-attention entries for the CI record events that, in general, are indicated by the setting of a bit in a hardware register. For the Ethernet, device-attention entries typically record errors on an Ethernet adapter device. Logged-message entries record the receipt of a message packet that contains erroneous data or that signals an error condition.

---

#### C.4.3.2 Device-Attention Entries

Examples C-1 and C-2 show device-attention entries for the CI and Ethernet, respectively. The left column gives the name of a device register or a memory location. The center column gives the value contained in that register or location, and the right column gives an interpretation of that value.

# VAXcluster Troubleshooting Information

## C.4 Diagnosing VAXport Device Problems

### Example C-1 CI Device-Attention Entry

```
***** ENTRY      83. ***** ①
ERROR SEQUENCE 10.          LOGGED ON:      SID 0150400A
DATE/TIME 15-APR-1988 11:45:27.61          SYS_TYPE 01010000 ②
DEVICE ATTENTION  KA780                    ③
                SCS NODE: MARS

CI SUB-SYSTEM, MARS$PAA0: - PORT POWER DOWN ④

    CNFGR          00800038                ADAPTER IS CI
                                           ADAPTER POWER-DOWN

    PMCSR          000000CE                MAINTENANCE TIMER DISABLE
                                           MAINTENANCE INTERRUPT ENABLE
                                           MAINTENANCE INTERRUPT FLAG
                                           PROGRAMMABLE STARTING ADDRESS
                                           UNINITIALIZED STATE

    PSR            80000001                RESPONSE QUEUE AVAILABLE
                                           MAINTENANCE ERROR

    PFAR           00000000
    PESR           00000000
    PPR            03F80001

    UCB$B_ERTCNT   32                      50. RETRIES REMAINING ⑤
    UCB$B_ERTMAX   32                      50. RETRIES ALLOWABLE ⑥
    UCB$L_CHAR     0C450000                SHAREABLE
                                           AVAILABLE
                                           ERROR LOGGING
                                           CAPABLE OF INPUT
                                           CAPABLE OF OUTPUT

    UCB$W_STS      0010                    ONLINE
    UCB$W_ERRCNT   000B                    11. ERRORS THIS UNIT ⑦
```

- ① The first two lines are the entry heading. These lines contain the number of the entry in this error log file, the sequence number of this error, and the identification number (SID) of this system's CPU. Each entry in the log file contains such a heading.
- ② The next line contains the date and time, and the system type.
- ③ The next two lines contain the entry type, the processor type (KA780), and the system's SCS node name.
- ④ The line CI SUB-SYSTEM, MARS\$PAA0: - PORT POWER DOWN contains the name of the subsystem and the device that caused the entry, and the reason for the entry. The CI subsystem's device PAA0 on node MARS was powered down.

The next 15 lines contain the names of hardware registers in the port, their contents, and interpretations of those contents. See the appropriate CI hardware manual for a description of all the CI port registers.

# VAXcluster Troubleshooting Information

## C.4 Diagnosing VAXport Device Problems

The CI port can recover from many errors, but not all. When an error occurs from which the CI cannot recover, the port notifies the port driver. The port driver logs the error and attempts to reinitialize the port. If the port fails after 50 such initialization attempts, the driver takes it off line, unless the system disk is connected to the failing port or this system is supposed to be a cluster member. If the CI port is required for system disk access or cluster participation and all 50 reinitialization attempts have been used, then the system bugchecks with a CIPORT-type bugcheck. Once a CI port is off line, you can put the port back on line only by rebooting the system.

- ⑤ The UCB\$\_ERTCNT field contains the number of reinitializations that the port driver can still attempt. The difference between this value and UCB\$\_ERTMAX is the number of reinitializations already attempted.
- ⑥ The UCB\$\_ERTMAX field contains the maximum number of times the port can be reinitialized by the port driver.
- ⑦ The UCB\$\_ERRCNT field contains the total number of errors that have occurred on this port since it was booted. This total includes both errors that caused reinitialization of the port and errors that did not.

### Example C-2 Ethernet Device-Attention Entry

```
***** ENTRY      80. ***** ①
ERROR SEQUENCE 26.          LOGGED ON:      SID 08000000
DATE/TIME 15-APR-1988 11:30:53.07          SYS_TYPE 01010000 ②
DEVICE ATTENTION KA630 ③
                SCS NODE: PHOBOS
NI-SCS SUB-SYSTEM, PHOBOS$PEAO: ④
    FATAL ERROR DETECTED BY DATALINK ⑤
        STATUS1      0000002C ⑥
        STATUS2      00000000
        DATALINK UNIT 0001 ⑦
        DATALINK NAME 41515803 ⑧
                        00000000
                        00000000
                        00000000
                                DATALINK NAME = XQA1:
        REMOTE NODE   00000000 ⑨
                        00000000
                        00000000
                        00000000
        REMOTE ADDR   00000000 ⑩
                        0000
        LOCAL ADDR    000400AA ⑪
                        4C07
                                ETHERNET ADDR = AA-00-04-00-07-4C
        ERROR CNT     0001 ⑫
                                1. ERROR OCCURRENCES THIS ENTRY
        UCB$_ERRCNT   0007
                                7. ERRORS THIS UNIT
```

- ① The first two lines are the entry heading. These lines contain the number of the entry in this error log file, the sequence number of this error, and the identification number (SID) of this system's CPU. Each entry in the log file contains such a heading.
- ② The next line contains the date and time, and the system type.

# VAXcluster Troubleshooting Information

## C.4 Diagnosing VAXport Device Problems

- ③ The next two lines contain the entry type, the processor type (KA630), and the system's SCS node name.
- ④ This line shows the name of the subsystem and component that caused the entry.
- ⑤ This line shows the reason for the entry. The Ethernet driver has shut down the datalink because of a fatal error. The datalink will be restarted automatically if possible.
- ⑥ STATUS1 and STATUS 2 show the I/O completion status returned by the Ethernet driver. If a message transmit was involved, the status applies to that transmit.
- ⑦ DATALINK UNIT shows the unit number of the Ethernet device on which the error occurred.
- ⑧ DATALINK NAME is the name of the Ethernet device on which the error occurred.
- ⑨ REMOTE NODE is the name of the remote node to which the packet was being sent. If zero, no remote node was available or no packet was associated with the error.
- ⑩ REMOTE ADDR is the Ethernet address of the remote node to which the packet was being sent. If zero, no packet was associated with the error.
- ⑪ LOCAL ADDR is the Ethernet address of the local node.
- ⑫ ERROR CNT—Because some errors can occur at extremely high rates, some error log entries represent more than one occurrence of an error. This field indicates how many. The errors counted occurred in the 3 seconds preceding the time stamp on the entry.

---

### C.4.3.3 Logged-Message Entries

Logged-message entries are made when the CI or Ethernet port receives a response that contains either data that the port driver cannot interpret or an error code in the status field of the response.

Example C-3 shows a CI logged-message entry with an error code in the status field PPD\$B\_STATUS.

# VAXcluster Troubleshooting Information

## C.4 Diagnosing VAXport Device Problems

### Example C-3 CI Logged-Message Entry

```
***** ENTRY      3. ***** ①
ERROR SEQUENCE 3.                               LOGGED ON SID 01188542
ERL$LOGMESSAGE, 15-APR-1988 13:40:25.13        ②
          KA780 REV #3. SERIAL #1346.   MFG PLANT 15. ③
CI SUB-SYSTEM, MARS$PAA0:                      ④
DATA CABLE(S) STATE CHANGE - PATH #0. WENT FROM GOOD TO BAD ⑤
      LOCAL STATION ADDRESS, 000000000002 (HEX) ⑥
      LOCAL SYSTEM ID, 000000000001 (HEX)       ⑦
      REMOTE STATION ADDRESS, 000000000004 (HEX) ⑧
      REMOTE SYSTEM ID, 0000000000A9 (HEX)      ⑨
UCB$B_ERTCNT      32                            ⑩
UCB$B_ERTMAX      32          50. RETRIES REMAINING
UCB$W_ERRCNT      0001          50. RETRIES ALLOWABLE
PPD$B_PORT        04          1. ERRORS THIS UNIT
PPD$B_STATUS      A5          REMOTE NODE #4.    ⑪
          FAIL
          PATH #0., NO RESPONSE
          PATH #1., "ACK" OR NOT USED
          NO PATH
PPD$B_OPC         05                            ⑬
PPD$B_FLAGS       03          IDREQ             ⑭
          RESPONSE QUEUE BIT
          SELECT PATH #0.
"CI" MESSAGE                                       ⑮
          00000000
          00000000
          80000004
          0000FE15
          4F503000
          00000507
          00000000
          00000000
          00000000
          00000000
          00000000
          00000000
          00000000
          00000000
          00000000
          00000000
          00000000
          00000000
```

- ① The first two lines are the entry heading. These lines contain the number of the entry in this error log file, the sequence number of the error, and the identification number (SID) of the system's CPU. Each entry in the log file contains a heading.
- ② The next line contains the entry type, the date, and time.

# VAXcluster Troubleshooting Information

## C.4 Diagnosing VAXport Device Problems

- ③ The next line contains the processor type (KA780), the hardware revision number of the CPU (REV #3), the serial number of the CPU (SERIAL #1346), and the plant number (15).
- ④ The line CI SUB-SYSTEM, MARS\$PAA0: contains the name of the subsystem and the device that caused the entry.
- ⑤ The next line gives the reason for the entry (one or more data cables have changed state), and a more detailed reason for the entry. Path 0, which the port used successfully before, cannot be used now.

**Note:** ANALYZE/ERROR\_LOG uses the notation path 0 and path 1; cable labels use the notation path A (=0) and path B (=1).

- ⑥ The local ⑥ and remote ⑧ station addresses are the port numbers (range 0–15) of the local and remote ports. The port numbers are set in hardware switches by field service. The local ⑦ and remote ⑨ system IDs are the SCS system IDs set by the SYSGEN parameter SCSSYSTEMID for the local and remote VAX systems. For HSCs, the system ID is set with the HSC console.
- ⑩ The rest of the entry, which consists of the entry fields that begin with UCB\$, gives information on the contents of the unit control block (UCB) for this CI device.

The following fields, which begin with PPD\$, are fields in the message packet that the local port has received.

- ⑪ PPD\$\_PORT contains the station address of the remote port. In a loopback datagram, however, this field contains the local station address.
- ⑫ The PPD\$\_STATUS field contains information on the nature of the failure that occurred during the current operation. When the operation completes without error, ERF prints the word NORMAL beside this field; otherwise, ERF decodes the error information contained in PPD\$\_STATUS. Here a NO PATH error occurred because of a lack of response on path 0, the selected path.
- ⑬ The PPD\$\_OPC field contains the code for the operation that the port was attempting when the error occurred. The port was trying to send a request-for-id message.
- ⑭ The PPD\$\_FLAGS field contains bits that indicate, among other things, the path that was selected for the operation.
- ⑮ The "CI" MESSAGE is a hexadecimal listing of bytes 16 through 83 (decimal) of the response (message or datagram). Since responses are of variable length depending upon the port opcode, bytes 16 through 83 may contain either more or fewer bytes than actually belong to the message. Here the request-for-id contains no information in bytes 16 through 83.

---

### C.4.3.4 Error Log Entry Descriptions

This section describes error log entries for the CI and Ethernet ports. Each entry shown is followed by a brief description of what the associated VAXport driver (PADRIVER, PBDRIVER, PEDRIVER) does, and the suggested action a system manager should take. In cases where Software Performance Reports with crash dumps are requested, it is important to capture the crash dumps as soon as possible after the error. For CI entries, note that path A and path 0 are the same path, and that path B and path 1 are the same path.

# VAXcluster Troubleshooting Information

## C.4 Diagnosing VAXport Device Problems

### BIIC FAILURE

**Explanation:** The VAXport driver attempts to reinitialize the port; after 50 failing attempts, it marks the device offline.

**User Action:** Call DIGITAL Field Service.

### CI PORT TIMEOUT

**Explanation:** The VAXport driver attempts to reinitialize the port; after 50 failing attempts, it marks the device offline.

**User Action:** First, increase the SYSGEN parameter PAPOLLINTERVAL. If the problem disappears and you are not running privileged user-written software, submit an SPR. Otherwise, call DIGITAL Field Service.

### 11/750 CPU MICROCODE NOT ADEQUATE FOR PORT

**Explanation:** The VAXport driver sets the port off line with no retries attempted. In addition, if this port is needed because the system is booted from an HSC or is participating in a cluster, the system bugchecks with a UCODEREV code bugcheck.

**User Action:** Read the appropriate section in the current VAXcluster SPD for information on required CPU microcode revisions. Call Field Service if necessary.

### PORT MICROCODE REV NOT CURRENT, BUT SUPPORTED

**Explanation:** The VAXport driver detected that the microcode is not at the current level, but will continue normally. This error is logged as a warning only.

**User Action:** Contact Field Service when convenient to have the microcode updated.

### PORT MICROCODE REV NOT SUPPORTED

**Explanation:** The VAXport driver sets the port off line without attempting any retries.

**User Action:** Read the VAXcluster SPD for information on the required CI port microcode revisions. Contact Field Service if necessary.

### DATA CABLE(S) STATE CHANGE CABLES HAVE GONE FROM CROSSED TO UNCROSSED

**Explanation:** The VAXport driver logs this event.

**User Action:** No action needed.

### DATA CABLE(S) STATE CHANGE CABLES HAVE GONE FROM UNCROSSED TO CROSSED

**Explanation:** The VAXport driver logs this event.

**User Action:** Check for crossed-cable pairs. See Section C.4.2.2.

# VAXcluster Troubleshooting Information

## C.4 Diagnosing VAXport Device Problems

DATA CABLE(S) STATE CHANGE  
PATH 0. WENT FROM BAD TO GOOD

**Explanation:** The VAXport driver logs this event.

**User Action:** No action needed.

DATA CABLE(S) STATE CHANGE  
PATH 0. WENT FROM GOOD TO BAD

**Explanation:** The VAXport driver logs this event.

**User Action:** Check path A cables to see that they are not broken or improperly connected.

DATA CABLE(S) STATE CHANGE  
PATH 0. LOOPBACK IS NOW GOOD, UNCROSSED

**Explanation:** The VAXport driver logs this event.

**User Action:** No action needed.

DATA CABLE(S) STATE CHANGE  
PATH 0. LOOPBACK WENT FROM GOOD TO BAD

**Explanation:** The VAXport driver logs this event.

**User Action:** Check for crossed-cable pairs or faulty CI hardware. See Sections C.4.2.1 and C.4.2.2.

DATA CABLE(S) STATE CHANGE  
PATH 1. WENT FROM BAD TO GOOD

**Explanation:** The VAXport driver logs this event.

**User Action:** No action needed.

DATA CABLE(S) STATE CHANGE  
PATH 1. WENT FROM GOOD TO BAD

**Explanation:** The VAXport driver logs this event.

**User Action:** Check path B cables to see that they are not broken or improperly connected.

DATA CABLE(S) STATE CHANGE  
PATH 1. LOOPBACK IS NOW GOOD, UNCROSSED

**Explanation:** The VAXport driver logs this event.

**User Action:** No action needed.

DATA CABLE(S) STATE CHANGE  
PATH 1. LOOPBACK WENT FROM GOOD TO BAD

**Explanation:** The VAXport driver logs this event.

**User Action:** Check for crossed-cable pairs or faulty CI hardware. See Sections C.4.2.1 and C.4.2.2.

# VAXcluster Troubleshooting Information

## C.4 Diagnosing VAXport Device Problems

### DATAGRAM FREE QUEUE INSERT FAILURE

**Explanation:** The VAXport driver attempts to reinitialize the port; after 50 failing attempts, it marks the device off line.

**User Action:** Call DIGITAL Field Service. This error is caused by a failure to obtain access to an interlocked queue. Possible sources of the problem are CI hardware failures or memory, SBI (11/780), CMI (11/750), or BI (8200, 8300, and 8800) contention.

### DATAGRAM FREE QUEUE REMOVE FAILURE

**Explanation:** The VAXport driver attempts to reinitialize the port; after 50 failing attempts, it marks the device off line.

**User Action:** Call DIGITAL Field Service. This error is caused by a failure to obtain access to an interlocked queue. Possible sources of the problem are CI hardware failures, or memory, SBI (11/780), CMI (11/750), or BI (8200, 8300, and 8800) contention.

### FAILED TO LOCATE PORT MICRO-CODE IMAGE

**Explanation:** The VAXport driver marks device off line and makes no retries.

**User Action:** Make sure console volume contains the microcode file CI780.BIN (for the CI780, CI750, or CIBCI) or the microcode file CIBCA.BIN for the CIBCA-AA, then reboot the system.

### HIGH PRIORITY COMMAND QUEUE INSERT FAILURE

**Explanation:** The VAXport driver attempts to reinitialize the port; after 50 failing attempts, it marks the device off line.

**User Action:** Call DIGITAL Field Service. This error is caused by a failure to obtain access to an interlocked queue. Possible sources of the problem are CI hardware failures or memory, SBI (11/780), CMI (11/750), or BI (8200, 8300, 8800) contention.

### MSCP ERROR LOGGING DATAGRAM RECEIVED

**Explanation:** On receipt of an error message from the HSC, the VAXport driver logs the error and takes no other action. It is recommended that you disable the sending of HSC informational error log datagrams with the appropriate HSC console command. Informational error log datagrams take considerable space in the error log data file.

**User Action:** They are useful to read only if they are not captured on the HSC console for some reason (for example, the HSC console ran out of paper.) This logged information is a duplicate of the messages logged on the HSC console.

### INAPPROPRIATE "SCA" CONTROL MESSAGE

**Explanation:** The VAXport driver closes the port-to-port virtual circuit to the remote port.

**User Action:** Submit a Software Performance Report to DIGITAL including the error logs and the crash dumps from the local and remote systems.

# VAXcluster Troubleshooting Information

## C.4 Diagnosing VAXport Device Problems

### INSUFFICIENT NON-PAGED POOL FOR INITIALIZATION

**Explanation:** The VAXport driver marks device off line and makes no retries.

**User Action:** Reboot the system with a larger value for NPAGEDYN or NPAGEVIR.

### LOW PRIORITY CMD QUEUE INSERT FAILURE

**Explanation:** The VAXport driver attempts to reinitialize the port; after 50 failing attempts, it marks the device off line.

**User Action:** Call DIGITAL Field Service. This error is caused by a failure to obtain access to an interlocked queue. Possible sources of the problem are CI hardware failures or memory, SBI (11/780), CMI (11/750), or BI (8200, 8300, and 8800) contention.

### MESSAGE FREE QUEUE INSERT FAILURE

**Explanation:** The VAXport driver attempts to reinitialize the port; after 50 failing attempts, it marks the device off line.

**User Action:** Call DIGITAL Field Service. This error is caused by a failure to obtain access to an interlocked queue. Possible sources of the problem are CI hardware failures or memory, SBI (11/780), CMI (11/750), or BI (8200, 8300, and 8800) contention.

### MESSAGE FREE QUEUE REMOVE FAILURE

**Explanation:** The VAXport driver attempts to reinitialize the port; after 50 failing attempts, it marks the device off line.

**User Action:** Call DIGITAL Field Service. This error is caused by a failure to obtain access to an interlocked queue. Possible sources of the problem are CI hardware failures or memory, SBI (11/780), CMI (11/750), or BI (8200, 8300, and 8800) contention.

### MICRO-CODE VERIFICATION ERROR

**Explanation:** The VAXport driver detected an error while reading the microcode that it just loaded into the port. The driver attempts to reinitialize the port; after 50 failing attempts, it marks the device off line.

**User Action:** Call DIGITAL Field Service.

### NO PATH-BLOCK DURING "VIRTUAL CIRCUIT" CLOSE

**Explanation:** The VAXport driver attempts to reinitialize the port; after 50 failing attempts, it marks the device off line.

**User Action:** Submit a Software Performance Report to DIGITAL including the error log and a crash dump from the local system.

### NO TRANSITION FROM UNINITIALIZED TO DISABLED

**Explanation:** The VAXport driver attempts to reinitialize the port; after 50 failing attempts, it marks the device off line.

**User Action:** Call DIGITAL Field Service.

# VAXcluster Troubleshooting Information

## C.4 Diagnosing VAXport Device Problems

### PORT ERROR BIT(S) SET

**Explanation:** The VAXport driver attempts to reinitialize the port; after 50 failing attempts, it marks the device off line.

**User Action:** For CI microcode version 7 or later, a *maintenance timer expiration* bit may mean that the PASTIMOUT SYSGEN parameter is set too low, especially if the local node is running privileged user-written software. For all other bits, call DIGITAL Field Service.

### PORT HAS CLOSED "VIRTUAL CIRCUIT"

**Explanation:** The VAXport driver closes the virtual circuit that the local CI port opened to the remote port.

**User Action:** Check the PPD\$B\_STATUS field of the error log entry for the reason the virtual circuit was closed. This error is normal if the remote system crashed or was shut down.

### PORT POWER DOWN

**Explanation:** The VAXport driver halts port operations, and then waits for power to return to the port hardware.

**User Action:** Restore power to the port hardware.

### PORT POWER UP

**Explanation:** The VAXport driver reinitializes the port and restarts port operations.

**User Action:** No action needed.

### RECEIVED "CONNECT" WITHOUT PATH-BLOCK

**Explanation:** The VAXport driver attempts to reinitialize the port; after 50 failing attempts, it marks the device off line.

**User Action:** Submit a Software Performance Report to DIGITAL including the error log and a crash dump from the local system.

### REMOTE SYSTEM CONFLICTS WITH KNOWN SYSTEM

**Explanation:** The configuration poller discovered a remote system with SCSSYSTEMID and/or SCSNODE equal to that of another system to which a virtual circuit is already open.

**User Action:** Shut the new system down as soon as possible. Reboot it with a unique SCSSYSTEMID and SCSNODE. Do not leave the new system up any longer than necessary. If you are running a cluster and two systems with conflicting identity are polling when any other virtual circuit failure takes place in the cluster, then systems in the cluster may crash with a CLUEXIT bugcheck.

# VAXcluster Troubleshooting Information

## C.4 Diagnosing VAXport Device Problems

### RESPONSE QUEUE REMOVE FAILURE

**Explanation:** The VAXport driver attempts to reinitialize the port; after 50 failing attempts, it marks the device off line.

**User Action:** Call DIGITAL Field Service. This error is caused by a failure to obtain access to an interlocked queue. Possible sources of the problem are CI hardware failures or memory, SBI (11/780), CMI (11/750), or BI (8200, 8300, and 8800) contention.

### SCSSYSTEMID MUST BE SET TO NON-ZERO VALUE

**Explanation:** The VAXport driver sets the port off line without attempting any retries.

**User Action:** Reboot the system with a conversational boot and set the SCSSYSTEMID to the correct value. At the same time, check that SCSNODE has been set to the correct nonblank value.

### SOFTWARE IS CLOSING "VIRTUAL CIRCUIT"

**Explanation:** The VAXport driver closes the virtual circuit to the remote port.

**User Action:** Check error log entries for the cause of the virtual circuit closure. Faulty transmission or reception on both paths, for example, causes this error and may be detected from the one or two previous error log entries noting bad paths to this remote node.

### SOFTWARE SHUTTING DOWN PORT

**Explanation:** The VAXport driver attempts to reinitialize the port; after 50 failing attempts, it marks the device off line.

**User Action:** Check other error log entries for the possible cause of the port reinitialization failure.

### UNEXPECTED INTERRUPT

**Explanation:** The VAXport driver attempts to reinitialize the port; after 50 failing attempts, it marks the device off line.

**User Action:** Call DIGITAL Field Service.

### UNRECOGNIZED "SCA" PACKET

**Explanation:** The VAXport driver closes the virtual circuit to the remote port. If the virtual circuit is already closed, the port driver inhibits datagram reception from the remote port.

**User Action:** Submit a Software Performance Report to DIGITAL, including the error log file that contains this entry and the crash dumps from both the local and remote systems.

# VAXcluster Troubleshooting Information

## C.4 Diagnosing VAXport Device Problems

### VIRTUAL CIRCUIT TIMEOUT

**Explanation:** The VAXport driver closes the virtual circuit that the local CI port opened to the remote port. This closure occurs if the remote node is running CI microcode version 7 or later, and the remote node has failed to respond to any messages sent by the local node.

**User Action:** This error is normal if the remote system halted, crashed, or was shut down. This error may mean that the local node's PASTIMOUT SYSGEN parameter is set too low, especially if the remote node is running privileged user-written software.

### INSUFFICIENT NON-PAGED POOL FOR VIRTUAL CIRCUITS

**Explanation:** The VAXport driver closes virtual circuits because of insufficient pool.

**User Action:** Enter the DCL command SHOW MEMORY to determine pool requirements, and then adjust the appropriate SYSGEN requirements.

**Note:** The following descriptions apply only for Ethernet devices.

### FATAL ERROR DETECTED BY DATALINK

**Completion status:** SS\$\_ABORT (0000002C)

**Explanation:** The Ethernet driver has shut down the device because of a fatal error and is returning all outstanding transmits with SS\$\_OPINCOMPL. The Ethernet device is automatically restarted, and all the aborted transmits are logged in the error log.

**User Action:** Infrequent occurrences of this error are probably not a problem. If they occur frequently, or are accompanied by connections to remote nodes being lost and reestablished, there is probably a hardware problem. Check for the proper Ethernet adapter revision level or call DIGITAL field service.

### TRANSMIT ERROR FROM DATALINK

**Completion status:** SS\$\_OPINCOMPL (000002D4)

**Explanation:** The Ethernet driver is in the process of restarting the datalink, because there was an error that forced the driver to shut down the controller and all users (see FATAL ERROR DETECTED BY DATALINK).

**Completion status:** SS\$\_DEVREQERR (00000334)

**Explanation:** The Ethernet controller tried to transmit the packet 16 times and failed because of defers and/or collisions. This condition indicates that Ethernet traffic is very heavy.

**Completion status:** SS\$\_DISCONNECT (0000204C)

**Explanation:** There was a loss of carrier during or after the transmit.

**User Action:** The Port Emulator automatically recovers from any of these errors, but excessive numbers of them indicate either that the Ethernet controller is faulty or that the Ethernet is overloaded. If you suspect either of these conditions, contact DIGITAL Field Service.

# VAXcluster Troubleshooting Information

## C.4 Diagnosing VAXport Device Problems

### INVALID CLUSTER PASSWORD RECEIVED

**Explanation:** A node is trying to join the cluster using the correct cluster group number for this cluster, but an invalid password. The Port Emulator discards the message. The probable cause is another cluster on the Ethernet using the same cluster group number.

**User Action:** Provide all clusters on the same Ethernet with unique cluster group numbers.

### NISCS PROTOCOL VERSION MISMATCH RECEIVED

**Explanation:** A node is trying to join the cluster using a version of the cluster Ethernet protocol that is incompatible with the one in use on this cluster.

**User Action:** Install a version of the VMS operating system that uses a compatible protocol, or change the cluster group number so that the node joins a different cluster.

---

#### C.4.4 OPA0 Error Messages

VAXport drivers detect certain error conditions and attempt to log them. Under some circumstances, attempts to log the error to the error logging device may fail. Such failures may occur because the error logging device is not accessible when attempts are made to log the error condition. Because of the central role that the VAXport device plays in clusters, the loss of error-logged information in such cases makes it difficult to diagnose and fix problems.

A second, redundant method of error logging captures at least some of the information about VAXport device error conditions that would otherwise be lost. This second method consists of broadcasting selected information about the error condition to OPA0, in addition to the port driver's attempt to log the error condition to the error logging device. The VAXport driver attempts both OPA0 error broadcasting and standard error logging under any of the following circumstances:

- The system disk has not yet been mounted.
- The system disk is undergoing mount verification.
- During mount verification, the system disk drive contains the wrong volume.
- Mount verification for the system disk has timed out.
- The local system is participating in a cluster, and quorum has been lost.

Note the implicit assumption that the system and error logging devices are one and the same.

This second method of reporting errors is also not entirely reliable. Because of the way OPA0 error broadcasting is performed, some error conditions may not be reported. This situation occurs whenever a second error condition is detected before the VAXport driver has been able to broadcast the first error condition to OPA0. In such a case, only the first error condition is reported to OPA0, because that condition is deemed to be the more important one.

Certain error conditions are always broadcast to OPA0, regardless of whether the error logging device is accessible. In general, these are errors that cause the port to shut down either permanently or temporarily.

# VAXcluster Troubleshooting Information

## C.4 Diagnosing VAXport Device Problems

One OPA0 error message for each error condition is always logged. The text of each error message is similar to the text in the summary displayed by formatting the corresponding standard error log entry using the Error Log Utility. (See Section C.4.3.4 for a list of Error Log Utility summary messages and their explanations.)

Many of the OPA0 error messages contain some optional information such as the remote port number, CI packet information (flags, port operation code, response status, and port number fields), or specific CI port registers.

Following is a list of OPA0 error messages, subdivided by error type. See the CI hardware documentation for a detailed description of the CI port registers (CNF = Configuration Register; PMC = Port Maintenance and Control Register; PSR = Port Status Register), which are optionally displayed for certain of the error conditions. The codes, always file accessible, specify whether the message is *always* logged on OPA0 or is logged only when the system device is *inaccessible*.

### Software Errors During Initialization (Always Logged on OPA0)

%Pxxn, Insufficient Non-Paged Pool for Initialization  
%Pxxn, Failed to Locate Port Micro-code Image  
%Pxxn, SCSSYSTEMID has NOT been set to a Non-Zero Value

### Hardware Errors (Always Logged on OPA0)

%Pxxn, BIIC failure - BICSR/BER/CNF xxxxxx/xxxxxx/xxxxxx  
%Pxxn, Micro-code Verification Error  
%Pxxn, Port Transition Failure - CNF/PMC/PSR xxxxxx/xxxxxx/xxxxxx  
%Pxxn, Port Error Bit(s) Set - CNF/PMC/PSR xxxxxx/xxxxxx/xxxxxx  
%Pxxn, Port Power Down  
%Pxxn, Port Power Up  
%Pxxn, Unexpected Interrupt - CNF/PMC/PSR xxxxxx/xxxxxx/xxxxxx  
%Pxxn, CI Port Timeout  
%Pxxn, CI port ucode not at required rev level. RAM/PROM rev is xxxx/xxxx  
%Pxxn, CI port ucode not at current rev level. RAM/PROM rev is xxxx/xxxx  
%Pxxn, CPU ucode not at required rev level for CI activity

### Queue Interlock Failures (Always Logged on OPA0)

%Pxxn, Message Free Queue Remove Failure  
%Pxxn, Datagram Free Queue Remove Failure  
%Pxxn, Response Queue Remove Failure  
%Pxxn, High Priority Command Queue Insert Failure  
%Pxxn, Low Priority Command Queue Insert Failure  
%Pxxn, Message Free Queue Insert Failure  
%Pxxn, Datagram Free Queue Insert Failure

# VAXcluster Troubleshooting Information

## C.4 Diagnosing VAXport Device Problems

### Errors Signaled with a CI Packet

%Pxxxn, Unrecognized SCA Packet - FLAGS/OPC/STATUS/PORT xx/xx/xx/xx  
(ALWAYS)

%Pxxxn, Port has Closed Virtual Circuit - REMOTE PORT xxx  
(ALWAYS)

%Pxxxn, Software Shutting Down Port  
(ALWAYS)

%Pxxxn, Software is Closing Virtual Circuit - REMOTE PORT xxx  
(ALWAYS)

%Pxxxn, Received Connect Without Path-Block - FLAGS/OPC/STATUS/PORT xx/xx/xx/xx  
(ALWAYS)

%Pxxxn, Inappropriate SCA Control Message - FLAGS/OPC/STATUS/PORT xx/xx/xx/xx  
(ALWAYS)

%Pxxxn, No Path-Block During Virtual Circuit Close - REMOTE PORT xxx  
(ALWAYS)

%Pxxxn, HSC Error Logging Datagram Received - REMOTE PORT xxx  
(INACCESSIBLE)

%Pxxxn, Remote System Conflicts with Known System - REMOTE PORT xxx  
(ALWAYS)

%Pxxxn, Virtual Circuit Timeout - REMOTE PORT xxx  
(ALWAYS)

%Pxxxn, Parallel Path is Closing Virtual Circuit - REMOTE PORT xxx  
(ALWAYS)

%Pxxxn, Insufficient Non-paged Pool for Virtual Circuits  
(ALWAYS)

### Cable Change-of-State Notification

%Pxxxn, Path #0. Has gone from GOOD to BAD - REMOTE PORT xxx  
(INACCESSIBLE)

%Pxxxn, Path #1. Has gone from GOOD to BAD - REMOTE PORT xxx  
(INACCESSIBLE)

%Pxxxn, Path #0. Has gone from BAD to GOOD - REMOTE PORT xxx  
(INACCESSIBLE)

%Pxxxn, Path #1. Has gone from BAD to GOOD - REMOTE PORT xxx  
(INACCESSIBLE)

%Pxxxn, Cables have gone from UNCROSSED to CROSSED - REMOTE PORT xxx  
(INACCESSIBLE)

%Pxxxn, Cables have gone from CROSSED to UNCROSSED - REMOTE PORT xxx  
(INACCESSIBLE)

%Pxxxn, Path #0. Loopback has gone from GOOD to BAD - REMOTE PORT xxx  
(ALWAYS)

%Pxxxn, Path #1. Loopback has gone from GOOD to BAD - REMOTE PORT xxx  
(ALWAYS)

%Pxxxn, Path #0. Loopback has gone from BAD to GOOD - REMOTE PORT xxx  
(ALWAYS)

%Pxxxn, Path #1. Loopback has gone from BAD to GOOD - REMOTE PORT xxx  
(ALWAYS)

%Pxxxn, Path #0. Has become working but CROSSED to Path #1. - REMOTE PORT xxx  
(INACCESSIBLE)

# VAXcluster Troubleshooting Information

## C.4 Diagnosing VAXport Device Problems

%Pxxx, Path #1. Has become working but CROSSED to Path #0. - REMOTE PORT xxx  
(INACCESSIBLE)

Note that if the port driver can identify the remote SCS node name of the affected system, the driver replaces the "REMOTE PORT xxx" text with "REMOTE SYSTEM X...", where X... is the value of the SYSGEN parameter SCSNODE on the remote system. If the remote SCS node name is not available, the port driver uses the existing message format.

Two other messages concerning the CI port appear on OPA0. They are as follows:

%Pxxx, CI port is reinitializing (xxx retries left.)

%Pxxx, CI port is going off line.

The first message indicates that a previous error requiring the port to shut down is recoverable, and that the port will be reinitialized. The 'xxx retries left' information specifies how many more reinitializations are allowed before the port must be left permanently off line. Each reinitialization of the port (for reasons other than power fail recovery) causes approximately 2K bytes of nonpaged pool to be lost.

The second message indicates that a previous error is not recoverable, and that the port will be left off line. In this case, the only way to recover the port is to reboot the system.

---

# Index

---

## A

---

- Adding a CI-connected node • 3–7
- Adding a satellite node • 3–7
- Alias node identifier
  - See DECnet–VAX network
- Alias operations
  - See DECnet–VAX network
- Allocation class • 5–5 to 5–9
  - assigning value to HSCs • 5–6
  - assigning value to nodes • 5–6
  - device name • 5–5
  - rules for specifying • 5–5
  - sample configurations • 5–6
- Allocation class identifier • 5–5
- Allocation class value
  - determining in mixed-interconnect VAXcluster configuration • 3–4
- Authorize Utility (AUTHORIZE) • B–1, B–2
- AUTOGEN
  - running with feedback option in mixed-interconnect VAXcluster configuration • 3–25
- AUTOGEN.COM command procedure
  - executed during CLUSTER\_CONFIG.COM ADD phase • 3–2

---

## B

---

- Batch queue • 4–6 to 4–8
  - assigning unique name to • 4–7
  - clusterwide generic • 4–7 to 4–8
  - initializing • 4–7
  - sample configuration • 4–6
  - setting up • 4–7 to 4–8
  - starting • 4–7
  - SY\$\$BATCH • 4–7
- Boot node
  - See Boot server
- Boot server
  - function in Local Area VAXcluster configuration • 1–5
  - functions • 1–5

- BOOT\_CONFIG.COM command procedure
  - sample interactive ADD session • 3–21
- Broadcast messages
  - controlling • 3–12
  - disabling while adding nodes • 3–6
  - OPCOM messages • 3–12
  - shutdown messages • 3–12
- Building a cluster • 3–1 to 3–24

---

## C

---

- CI (computer interconnect)
  - analyzing error log entry • C–16
  - communication path failure • C–11
  - communication path hierarchy • C–10
  - error log entry • C–16, C–21
  - port • C–9
    - loopback datagram facility • C–12
    - polling • C–9
- CI Cable
  - repairing • C–15
- CI-connected node
  - adding • 3–6
- CI Port
  - verifying function • C–11
- CLUEXIT bugcheck
  - diagnosing • C–8
- Cluster-accessible disk • 1–12, 5–1, 5–1 to 5–5
  - and MSCP Server • 5–1, 5–2
  - MASSBUS disk • 5–1, 5–2
  - setting up • 5–1
  - UDA disk • 5–1, 5–2
  - UNIBUS disk • 5–1, 5–2
- Cluster authorization file (CLUSTER\_AUTHORIZE.DAT)
  - See Security functions
  - function in Local Area VAXcluster configuration • 1–9
  - function in mixed-interconnect VAXcluster configuration • 1–9
- Cluster common files • 1–5
- Cluster queues • 1–12
- Cluster SYSGEN parameters • A–1 to A–2
- CLUSTER\_CONFIG.COM command procedure
  - adding nodes • 3–6

## Index

- CLUSTER\_CONFIG.COM command procedure (cont'd.)
  - converting standalone node to cluster node • 3-21
  - functions • 3-2
  - modifying satellite Ethernet hardware address • 3-14
  - preparing to execute • 3-5
  - removing satellite nodes • 3-13
  - required information • 3-5
  - sample interactive CREATE session • 3-21
  - system files created during ADD phase for satellite node • 3-2
- Common command procedures
  - coordinating • 2-9 to 2-11
  - creating • 2-10
  - executing • 2-10
  - on cluster-accessible disks • 2-9
  - setting up • 2-10
  - SYLOGIN.COM • 2-11
- Common-environment cluster • 2-1
  - creating • 2-9
  - preparing environment • 2-10
  - preparing operating environment • 2-1
- Common file
  - coordinating for multiple boot servers • 2-14
  - coordinating for multiple system disks • 2-14
  - job controller • 4-1, 4-9
  - mail database • 2-13
  - NETPROXY.DAT • 2-12
  - rights database • 2-14
  - RIGHTSLIST.DAT • 2-14
  - system • 2-11
  - SYSUAF.DAT • 2-12
  - VMSMAIL\_PROFILE.DATA • 2-13
- Common system disk
  - directory structure • 2-2
- Computer interconnect (CI) • 1-2
- Connection manager
  - restoring quorum after unexpected node failure • 3-26
- Connection Manager • 1-9 to 1-11
- Conversational bootstrap
  - See Security functions
- Convert Utility (CONVERT)
  - and exceptions file • B-2
  - to merge SYSUAF.DAT files • B-1
- Crossed cable • C-12

---

## D

---

- DECnet-VAX network
  - alias node identifier, defining for cluster • 2-6
  - alias operations, enabling for satellite nodes • 2-8
  - circuit service, enabling for cluster boot server • 2-6
  - cluster functions • 1-9
  - configuring using NETCONFIG.COM command procedure • 2-6
  - copying remote node databases in VAXcluster environments • 2-8
  - making databases available clusterwide • 2-7
  - maximum address value, defining for cluster boot server • 2-6
  - modifying satellite Ethernet hardware address • 3-14
  - NETCONFIG.COM command procedure, sample interactive session • 2-6
  - NETNODE\_REMOTE.DAT file, renaming to SYS\$COMMON directory • 2-7
  - Network Control Program (NCP) • 2-7
  - remote node data, making available clusterwide • 2-6
  - restoring satellite configuration data • 3-12
  - restoring satellite network configuration data • 3-12
  - starting the network • 2-7
  - tailoring • 2-6
- Device
  - cluster
    - setting up • 5-10
  - disk
    - managing • 5-1 to 5-12
    - naming conventions • 5-5 to 5-9
- Device driver
  - loading • 2-9
- Device name • 5-5 to 5-9
  - allocation class • 5-5
  - and allocation • 5-5 to 5-9
- Directory structure
  - on common system disk • 2-2
- Disk
  - See also Dual-pathed disk
  - See also Dual-ported disk
  - cluster-accessible • 5-1, 5-1 to 5-5
    - storing common procedures on • 2-9
  - command procedures for setting up • 2-10
  - device naming conventions • 5-5 to 5-9

Disk (cont'd.)

- directory structure on common system disk • 2-2
- HSC • 5-1, 5-6
- managing • 5-1 to 5-12
- MASSBUS • 5-1, 5-2
  - dual-ported • 5-4
- mounting • 5-10
- MSCP-served • 5-1
- paths • 5-5
- quorum • 1-11
- restricted access • 5-1
- setting up • 2-10, 5-10
- UDA • 5-1, 5-2
- UNIBUS • 5-1, 5-2

Disk class driver • 1-3

Disk controller • 1-2

Distributed file system • 1-3

Distributed job controller • 1-3

Distributed Lock Manager • 1-3

Distributed processing • 1-12, 4-1

DSA disk
 

- dual-ported • 5-4
- failover • 5-4

Dual-pathed disk • 5-2, 5-3 to 5-5
 

- DSA • 5-4
- HSC • 5-3, 5-6
- MASSBUS • 5-4

Dual-ported disk • 5-2
 

- MASSBUS • 5-4
- setting up • 2-9

Duplicate cluster system disk
 

- creating • 3-21

---

## E

---

Environment
 

- creating common-environment cluster • 2-1, 2-9
- multiple-environment cluster • 2-1
- user
  - defining • 2-11

Ethernet
 

- error log entry • C-21
- monitoring activity • 3-26
- port • C-10
  - communication • C-10

Ethernet hardware address
 

- See Satellite node

Exceptions file
 

- and CONVERT • B-2
- use of • B-2

---

## F

---

Failover
 

- dual-ported DSA disk • 5-4

Failure of node to boot or join the cluster • C-1

File access
 

- controlling • 2-11

File system
 

- coordinating • 2-11 to 2-12

---

## G

---

Generic queue
 

- clusterwide batch • 4-7 to 4-8
- clusterwide printer • 4-3 to 4-5
- establishing local • 4-3

---

## H

---

Hang condition
 

- diagnosing • C-7

Hardware component
 

- computer interconnect (CI) • 1-2
- Ethernet • 1-2
- hierarchical storage controller • 1-2
- HSC • 1-2
  - optional • 1-2
  - star coupler • 1-2
  - VAXcluster • 1-2
  - VAX processor • 1-2

Hierarchical Storage Controller (HSC)
 

- changing allocation class values • 3-24

HSC disk • 1-2, 5-1, 5-2
 

- dual-pathed • 5-3, 5-6

---

## I

---

INITIALIZE/QUEUE/BATCH command • 4-7

## Index

---

### J

---

#### JBCSYSQUE.DAT

- as common file • 2–10
  - sharing • 2–11
  - specifying location of • 4–1
- Job controller • 1–3
- Job-controller queue file • 1–12, 2–10, 4–1, 4–9
- 

### K

---

#### Known images

- installing • 2–10
- 

### L

---

#### Local Area VAXcluster configuration

- boot server • 1–5
- creating cluster security database • 1–8
- monitoring Ethernet activity • 3–26

#### Local disk

- setting up • 2–9

#### Logical name

- defining • 2–10
- defining for NETPROXY.DAT • 2–12
- defining for SYLOGIN.COM • 2–9
- defining for SYSUAF.DAT • 2–12
- defining for VMSMAIL\_PROFILE.DATA • 2–13

#### Login

- controlling • 2–11
- 

### M

---

#### MAIL Database

- preparing common file • 2–13

#### Mail Utility (MAIL)

- controlling • 2–11
- preparing common database • 2–13

#### MASSBUS disk • 5–1

- as cluster-accessible device • 5–1, 5–2
- dual-pathed • 5–4
- dual-ported • 5–4

#### Mixed-Interconnect VAXcluster configuration • 1–7 to 1–8

#### Mixed-Interconnect VAXcluster configuration (cont'd.)

- changing allocation class values on HSCs • 3–24
- creating cluster security database • 1–8
- determining allocation class value • 3–4
- monitoring Ethernet activity • 3–26
- MSCP-served HSC disk • 1–7
- running AUTOGEN with feedback option • 3–25
- updating MODPARAMS.DAT files • 3–23
- volume shadowing • 5–10 to 5–12

#### MODPARAMS.DAT

- updating in mixed-interconnect VAXcluster configuration • 3–23

#### MODPARAMS.DAT file

- created during CLUSTER\_CONFIG.COM ADD phase • 3–2

#### Mounting disks • 5–10

#### MSCP Server • 1–3

- for cluster-accessible disks • 5–1, 5–2
- initializing • 5–2

#### MSCP\_LOAD parameter • 5–2

#### MSCP\_SERVE\_ALL parameter • 5–2

#### Multiple-environment cluster • 2–1

- creating • 2–9
  - operating environment • 2–1
  - setting up operating environment • 2–11
- 

### N

---

#### Naming devices • 5–5 to 5–9

#### NETCONFIG.COM command procedure

- See DECnet-VAX network

#### NETNODE\_REMOTE.DAT

- sharing • 2–11

#### NETNODE\_UPDATE.COM command procedure

- See DECnet-VAX network

#### NETPROXY.DAT

- building common version • 2–12 to 2–13
- defining logical name for • 2–12
- setting up • 2–12
- sharing • 2–11

#### Network

- See DECnet-VAX network

#### Network Control Program (NCP)

- See DECnet-VAX network

#### Node

- HSC • 1–2
- passive • 1–2

#### Node-specific startup functions • 2–11

---

## O

- OPA0: workstation operator console terminal
  - See Workstation node
- OPCOM messages
  - See Broadcast messages
- Operating system
  - coordinating files • 2-11 to 2-12
  - installing • 2-4
  - upgrading • 2-4

---

## P

- Page file (PAGEFILE.SYS)
  - created during CLUSTER\_CONFIG.COM ADD phase • 3-2, 3-3
- Partitioning of cluster • 1-9, C-9
- Port select button • 5-3
- Preparation
  - of common-environment cluster • 2-1
  - of common MAIL Database • 2-13
  - of common Rights Database • 2-14
  - of multiple-environment cluster • 2-1
- Preparing cluster operating environment • 2-1 to 2-15
- Preparing operating environment
  - multiple-environment • 2-1
- Printer queue • 4-1 to 4-5
  - assigning unique name to • 4-2
  - clusterwide generic • 4-3 to 4-5
  - establishing local generic • 4-3
  - initializing • 4-3
  - sample configuration • 4-2
  - setting up • 4-1 to 4-3
  - starting • 4-3
  - SYSPRINT • 4-5
- Proxy login
  - controlling • 2-11
  - records • 2-12

---

## Q

- Queue
  - batch
    - See Batch queue
  - command procedures • 2-10, 4-9 to 4-14

- Queue (cont'd.)
  - controlling • 1-12, 4-1
  - job controller • 2-10
    - queue file • 1-12
  - job controller queue file • 4-1
  - printer
    - See Printer queue
  - setting up • 2-10
  - sharing • 2-10
  - single-node and cluster • 4-1 to 4-14
- Quorum
  - equation • 1-10
  - loss of quorum causes cluster hang condition • C-7
  - lowering value • 3-27
  - reasons for loss • C-7
- QUORUM.DAT file • 1-11
- Quorum disk • 1-11
- Quorum disk mounting • 1-11
- Quorum disk watcher • 1-11
- Quorum file • 1-11
- Quorum Scheme • 1-10

---

## R

- RD series disk
  - See Satellite node
- Recovering from failure
  - satellite node fails to boot • C-4
- Remote network node data
  - controlling • 2-11
- Remote node databases
  - copying • 2-8
- Removing a satellite node • 3-13
- Resource sharing in cluster • 1-9
- Restricted access disk • 5-1
- Rights Database
  - preparing common file • 2-14
- RIGHTSLIST.DAT
  - preparing common version of • 2-14
  - sharing • 2-11
- RMS
  - VMS RMS distributed file system • 1-3
- Rules
  - for allocation classes • 5-5

# Index

---

## S

---

- Satellite node
  - adding • 3–6
  - disabling conversational bootstrap operations • 3–31
  - functions • 1–6
  - maintaining network configuration data • 3–12
  - modifying Ethernet hardware address • 3–14
  - obtaining Ethernet hardware address • 3–5
  - RD series disk used for local paging and swapping • 1–6
  - removing • 3–13
  - restoring network configuration data • 3–12
  - shutting down before removing from cluster • 3–13
  - system files created during CLUSTER\_CONFIG.COM ADD phase • 3–2
- SCS (System Communications Services) • C–10
- SCS SYSGEN parameters • A–2 to A–4
- Security functions
  - cluster authorization file (CLUSTER\_AUTHORIZE.DAT) • 3–30
  - Cluster\_Authorize Utility (CLUSTER\_AUTHORIZE)
    - sample interactive session • 3–30
  - controlling conversational bootstrap operations on satellite nodes • 3–31
  - overview • 3–29
  - SYSMAN Utility
    - altering cluster security data • 3–30
- SET CLUSTER/EXPECTED\_VOTES command • 3–27
- SET DEVICE/DUAL\_PORT command • 5–4
- Setup procedure
  - coordinating cluster common files for multiple boot servers • 2–14
  - coordinating cluster common files for multiple system disks • 2–14
- SHADOWING parameter
  - setting on CI-connected nodes in mixed-interconnect VAXcluster configuration • 5–10
  - setting on satellite nodes in mixed-interconnect VAXcluster configuration • 5–10
- Shared command procedure files • 2–9
- Shared disk volume • 5–9
  - for job controller queue file • 4–9
  - mounting • 5–9
- Shared file
  - JBCSYSQUE.DAT • 2–11
  - Shared file (cont'd.)
    - NETPROXY.DAT • 2–11, 2–12
    - RIGHTSLIST.DAT • 2–11
    - SYSUAF.DAT • 2–11, 2–12
    - VMSMAIL\_PROFILE.DATA • 2–11
  - Shared queues • 4–1 to 4–14
  - Show Cluster Utility (SHOW CLUSTER) • 3–26
  - Shutdown messages
    - See Broadcast messages
  - Shutting down the cluster • 3–27
  - Site-specific startup command file elements • 2–11
  - Standalone node
    - converting to cluster node • 3–21
  - Star coupler • 1–2
  - START/QUEUE/MANAGER command • 4–1
  - Startup
    - node-specific function • 2–11
  - Startup command file
    - building common version • 2–10
    - coordinating • 2–9 to 2–11
    - creating common version • 2–10
    - site-specific
      - elements • 2–11
  - Swap file (SWAPFILE.SYS)
    - created during CLUSTER\_CONFIG.COM ADD phase • 3–2, 3–3
  - SYLOGIN.COM
    - building common version • 2–11
    - coordinating • 2–9 to 2–11
    - creating common version of • 2–10
    - defining logical name for • 2–9
  - SYS\$BATCH
    - redefining • 4–7
  - SYS\$PRINT
    - redefining for local generic queues • 4–5
  - SYSGEN parameters
    - Cluster parameters • A–1 to A–2
    - SCS parameters • A–2 to A–4
  - SYSMAN Utility
    - See Security functions
  - SYSTARTUP.COM
    - to set up queues • 4–9
  - System command procedures
    - coordinating • 2–9 to 2–11
  - System communications services
    - See SCS
  - System disk
    - directory structure on common system disk • 2–2

## System file

- building common versions • 2-11
- coordinating • 2-11 to 2-12

## SYSUAF.DAT

- building common version • 2-12 to 2-13
- defining logical name for • 2-12
- printing listing of • B-1
- setting up • 2-12
- sharing • 2-11
- using CONVERT to merge • B-1

---

**T**


---

## Terminal

- setting up • 2-9

## Troubleshooting • C-1 to C-32

---

**U**


---

## UDA disk • 5-1

- as cluster-accessible device • 5-1, 5-2

## UNIBUS disk • 5-1

- as cluster-accessible device • 5-1, 5-2

## Upgraded systems • 2-4

## User accounts

- comparing • B-1
- coordinating • 2-12 to 2-13, B-1
- group UIC • B-1

## User environment

- defining • 2-11

## User identification code

- changing for directories • B-1
- changing for files • B-1
- coordinating • B-1
- coordination • B-1

---

**V**


---

## VAXcluster

- boot events • C-1
- building • 3-1 to 3-24
- changing configuration type • 3-19
- changing from CI-only to mixed-interconnect configuration • 3-19
- changing from local area to mixed-interconnect configuration • 3-20

## VAXcluster (cont'd.)

- communication mechanisms • 1-9
- configuration data
  - recording • 3-25
- Connection Manager • 1-3
- devices • 5-1 to 5-12
- diagnosing CLUEXIT bugcheck • C-8
- diagnosing cluster hang condition • C-7
- distributed file system • 1-3
- Distributed Job Controller • 1-3
- Distributed Lock Manager • 1-3
- error log entries for VAXport device • C-16
- failure of node to boot • C-1
- failure of node to join the cluster • C-1, C-6
- hang condition • C-7 to C-8
- overview • 1-1 to 1-12
- planning configuration functions • 3-1
- preparing operating environment • 2-1 to 2-15
- queues • 4-1 to 4-14
- Quorum
  - reasons for loss • C-7
- recording configuration data • 3-25
- recovering from startup procedure failure • C-7
- resource access • 1-3
- resource locking • 1-3
- satellite node boot failure • C-4
- System Communication Services • 1-3
- troubleshooting • C-1 to C-32
- VAXport device error log entries • C-16
- VAXport driver • 1-3

## VAXCluster

- local configuration
  - monitoring Ethernet activity • 3-26
- mixed-interconnect configuration
  - monitoring Ethernet activity • 3-26

## VAXVMSSYS.PAR file

- created during CLUSTER\_CONFIG.COM ADD phase • 3-2

## Virtual circuit • C-9

## VMSMAIL\_PROFILE.DATA

- defining logical name for • 2-13
- preparing common version of • 2-13
- sharing • 2-11

## Volume label

- modifying for satellite's local disk • 3-3

## Volume shadowing

- in mixed-interconnect VAXcluster configuration • 5-10 to 5-12

## Index

---

### W

---

Workload balancing • 1–12, 4–1

Workstation node

controlling broadcasts to operator console  
terminal (OPA0:) • 3–12

# Reader's Comments

VMS VAXcluster Manual  
AA-LA27A-TE

Please use this postage-paid form to comment on this manual. If you require a written reply to a software problem and are eligible to receive one under Software Performance Report (SPR) service, submit your comments on an SPR form.

Thank you for your assistance.

**I rate this manual's:**

	Excellent	Good	Fair	Poor
Accuracy (software works as manual says)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Completeness (enough information)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Clarity (easy to understand)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Organization (structure of subject matter)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Figures (useful)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Examples (useful)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Index (ability to find topic)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Page layout (easy to find information)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

I would like to see more/less \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

What I like best about this manual is \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

What I like least about this manual is \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

I found the following errors in this manual:

Page	Description
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____

Additional comments or suggestions to improve this manual:  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

I am using **Version** \_\_\_\_\_ of the software this manual describes.

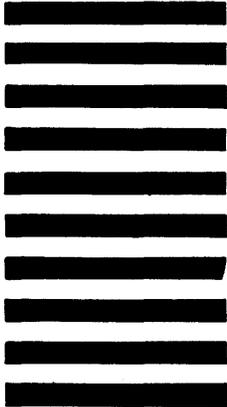
Name/Title \_\_\_\_\_ Dept. \_\_\_\_\_  
Company \_\_\_\_\_ Date \_\_\_\_\_  
Mailing Address \_\_\_\_\_ Phone \_\_\_\_\_

— Do Not Tear - Fold Here and Tape —

**digital**<sup>TM</sup>



No Postage  
Necessary  
if Mailed  
in the  
United States



**BUSINESS REPLY MAIL**  
FIRST CLASS PERMIT NO. 33 MAYNARD MASS.

POSTAGE WILL BE PAID BY ADDRESSEE

DIGITAL EQUIPMENT CORPORATION  
Corporate User Publications—Spit Brook  
ZK01-3/J35 110 SPIT BROOK ROAD  
NASHUA, NH 03062-9987



— Do Not Tear - Fold Here —

# Reader's Comments

VMS VAXcluster Manual  
AA-LA27A-TE

Please use this postage-paid form to comment on this manual. If you require a written reply to a software problem and are eligible to receive one under Software Performance Report (SPR) service, submit your comments on an SPR form.

Thank you for your assistance.

I rate this manual's:	Excellent	Good	Fair	Poor
Accuracy (software works as manual says)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Completeness (enough information)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Clarity (easy to understand)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Organization (structure of subject matter)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Figures (useful)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Examples (useful)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Index (ability to find topic)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Page layout (easy to find information)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

I would like to see more/less \_\_\_\_\_  
\_\_\_\_\_

What I like best about this manual is \_\_\_\_\_  
\_\_\_\_\_

What I like least about this manual is \_\_\_\_\_  
\_\_\_\_\_

I found the following errors in this manual:

Page	Description
_____	_____
_____	_____
_____	_____
_____	_____

Additional comments or suggestions to improve this manual:  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

I am using **Version** \_\_\_\_\_ of the software this manual describes.

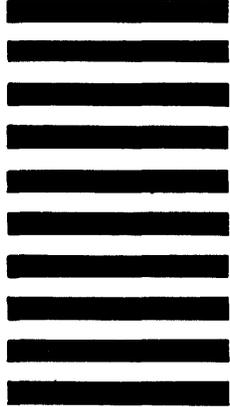
Name/Title \_\_\_\_\_ Dept. \_\_\_\_\_  
Company \_\_\_\_\_ Date \_\_\_\_\_  
Mailing Address \_\_\_\_\_  
Phone \_\_\_\_\_

-- Do Not Tear - Fold Here and Tape --

**digital**<sup>TM</sup>



No Postage  
Necessary  
if Mailed  
in the  
United States



**BUSINESS REPLY MAIL**  
FIRST CLASS PERMIT NO. 33 MAYNARD MASS.

POSTAGE WILL BE PAID BY ADDRESSEE

DIGITAL EQUIPMENT CORPORATION  
Corporate User Publications—Spit Brook  
ZK01-3/J35 110 SPIT BROOK ROAD  
NASHUA, NH 03062-9987



-- Do Not Tear - Fold Here --