# DEC Distributed Queuing Service for OpenVMS Systems

# Documentation Set

Part Number: AA–Q5FVA–TE

This manual consists of guides for the user, the system manager, and the software installer. These guides describe how to use, configure and manage, and install the DEC Distributed Queuing Service for OpenVMS Systems software on OpenVMS AXP and OpenVMS VAX systems.

**Revision/Update Information:** This manual supersedes the *VAX Distributed Queuing Service User's Guide* (AA–KM86B–TE), the *VAX Distributed Queuing Service Management Guide* (AA–KM87B–TE), the *VAX Distributed Queuing Service Installation Guide* (AA–KM94B–TE), and the *DEC Distributed Queuing Service for OpenVMS AXP Installation Guide* (AA–PW1SA–TE).

**Operating System and Version:** OpenVMS AXP Version 6.1 or higher. OpenVMS VAX Version 5.5-2 or higher. Future maintenance releases may require higher versions.

**Software and Version:** DEC Distributed Queuing Service Version 1.3 for OpenVMS Systems.

**Digital Equipment Corporation**
**Maynard, Massachusetts**

**First Printing, March 1994**

Digital Equipment Corporation makes no representations that the use of its products in the manner described in this publication will not infringe on existing or future patent rights, nor do the descriptions contained in this publication imply the granting of licenses to make, use, or sell equipment or software in accordance with the description.

Possession, use, or copying of the software described in this publication is authorized only pursuant to a valid written license from Digital or an authorized sublicensor.

© Digital Equipment Corporation 1994. All Rights Reserved.

The postpaid Reader's Comments forms at the end of this document request your critical evaluation to assist in preparing future documentation.

The following are trademarks of Digital Equipment Corporation: Alpha AXP, AXP, DEC, DECnet, DECwindows, Digital, OpenVMS, OpenVMS AXP, TK, VAX, VAXcluster, VAX DOCUMENT, VMScluster, and the DIGITAL logo.

Motif is a registered trademark of Open Software Foundation, Inc.

All other trademarks and registered trademarks are the property of their respective holders.

S2439

This document is available on CD-ROM.

This document was prepared using VAX DOCUMENT Version 2.1.

# Contents

**Preface** . . . . . xi

**Part I User's Guide**

**1 Using the DQS Software**

Understanding the Software . . . . . 1–2
Getting Help With DQS . . . . . 1–4

**2 Printing Files Using DQS Software**

Printing in Network Environments . . . . . 2–2
Understanding Print Job Messages . . . . . 2–6
Printing with Forms . . . . . 2–7
Using PRINT Command Qualifiers . . . . . 2–8
Understanding PRINT Command Error Messages . . . . . 2–11

**3 Viewing the Status of Print Jobs**

The QSHOW Command . . . . . 3–2
Checking the Status of a Print Job . . . . . 3–3
Using QSHOW Command Qualifiers . . . . . 3–5
Understanding QSHOW Error Messages . . . . . 3–7

**4 Deleting Print Jobs From a Queue**

The QDELETE/ENTRY Command . . . . . 4–2
Deleting a Print Job . . . . . 4–3
Understanding QDELETE Error Messages . . . . . 4–4

## 5 Modifying the Characteristics of a Job


The QSET/ENTRY Command . . . . . 5–2
Modifying a Print Job . . . . . 5–3
Using QSET Command Qualifiers . . . . . 5–4
Understanding QSET Error Messages . . . . . 5–6


## 6 Spooling Files to a Client Queue

## Part II System Manager's Guide

## 7 Understanding the DEC Distributed Queuing Service


Overview of the DQS Software . . . . . 7–2
Primary Functions of the DQS Software . . . . . 7–5
Functions of the DQS Commands . . . . . 7–7
Functions of DQS Software Components . . . . . 7–11
The DQS Notification Service . . . . . 7–13
The DQS Symbionts . . . . . 7–16
Defining Forms on Client and Server Nodes . . . . . 7–17
The DQS Server Account . . . . . 7–18
The Significance of Parameter 8 in DQS Printing . . . . . 7–19
Configuring DQS Software . . . . . 7–20
Using Daisy-Chained Queues . . . . . 7–22
DQS Error Conditions . . . . . 7–25
DQS Security Issues and Access Rights . . . . . 7–29
Processes Created by DQS Software . . . . . 7–30
Devices Supported by DQS Software . . . . . 7–31
Batch Jobs and DQS Printing . . . . . 7–32


## 8 Configuring and Managing a DQS Server


Overview of Server Management . . . . . 8–2
Defining Server System Logical Names . . . . . 8–3
Modifying Definitions of Certain Logical Names . . . . . 8–5
Before You Set Up Remote Queues . . . . . 8–6
Understanding How to Set Up Remote Queues . . . . . 8–7
Understanding the Configuration File . . . . . 8–9
Setting Up Remote Queues . . . . . 8–11
Enabling Network Access . . . . . 8–12
Denying Client Node Authorization to the DQS Server . . . . . 8–13
Granting Client Node Authorization to the DQS Server . . . . . 8–14

Making Your Changes Take Effect . . . . . 8–16
Determining How Clients are Accessing a Server . . . . . 8–17
Controlling the Note on the Job Banner Page . . . . . 8–22
Enabling Status Messages for Queues . . . . . 8–24
Enabling the DQS Print Symbiont To Control Remote Queues . . . . . 8–26
Specifying the Scanning Interval for Server Notification . . . . . 8–28
Specifying a Directory for PrintServer Log Files . . . . . 8–30
Specifying the Priority of the Server Process . . . . . 8–31
Specifying the Maximum Priority of DQS Print Jobs . . . . . 8–32
Specifying the Duration of Links to Client Nodes . . . . . 8–33
Move the Server Account Directory to Another Device . . . . . 8–35
Advanced VMScluster DQS Configurations . . . . . 8–36
Specifying Unique Notifier Lock Names in a VMScluster Environment . . . . . 8–37
Redefining DQS$SERVER_CONFIG_DAT_FILE and DQS$SERVER_CONFIG_TXT_FILE . . . . . 8–38
Specifying Remote Queues on a VMScluster System . . . . . 8–40
Configuring Daisy-Chained Queues . . . . . 8–41
Summary of Server Tasks . . . . . 8–42

## 9 Configuring and Managing a DQS Client

Overview of Client Management . . . . . 9–2
Managing Client Processes . . . . . 9–3
Creating Client Queues with DQS$IMPORT.COM . . . . . 9–4
Creating Client Queues with More Advanced Methods . . . . . 9–6
Temporarily Changing the Definition of a Client Queue . . . . . 9–12
Deleting Client Queues . . . . . 9–14
Configuring Client Nodes in a VMScluster Environment . . . . . 9–15
Synchronizing Client and Server Forms Definitions . . . . . 9–16
Spooling to a Client Queue . . . . . 9–18
Controlling the Note on the Job Banner Page . . . . . 9–20
Regulating Forms Checking . . . . . 9–22

## 10 Troubleshooting

General Troubleshooting Guidelines . . . . . 10–2
Running the IVP . . . . . 10–3
Problem: Remote Notification Is Not Working for Any Jobs . . . . . 10–6
Problem: Notification of Print Job Completion Is Not Occuring for Some Jobs . . . . . 10–7
Problem: Jobs Remain in the Remote Queue With Completion Status . . 10–8
Problem: A Job Is Not Printing . . . . . 10–9

Problem: Many DQS Client Queues Slow Down the Network . . . . . 10–17

## Part III Installation Guide

## 11 What You Need to Know

Required Hardware and Software . . . . . 11–3
Licensing . . . . . 11–5
Processes Created . . . . . 11–6
First-time Installation Considerations . . . . . 11–7
Upgrade Considerations . . . . . 11–8
Time Required to Install . . . . . 11–10

## 12 Do This First

Inspect the Distribution Kit . . . . . 12–2
Review DQS Installation . . . . . 12–4
Prepare for VMScluster Installation . . . . . 12–5
Log In to a Privileged Account . . . . . 12–6
Access Online Release Notes . . . . . 12–8
Determine Process Account Quotas . . . . . 12–9
Determine Disk Space . . . . . 12–11
Set Up an Alternate Working Device . . . . . 12–12
Register Your License . . . . . 12–14
Perform System Maintenance Procedures . . . . . 12–15
Perform First-Time Installation Activities . . . . . 12–18
Back Up Your System Disk . . . . . 12–20

## 13 Using the PCSI Utility

Accessing Release Notes . . . . . 13–2
Starting the Installation: DCL Interface . . . . . 13–3
Obtaining DCL Help . . . . . 13–6
Determining Installation Defaults: DCL Interface . . . . . 13–7
Verifying Preinstallation Tasks: DCL Interface . . . . . 13–8
Stopping and Restarting: DCL Interface . . . . . 13–10
Responding to Questions: DCL Interface . . . . . 13–11
Determining Installation Progress: DCL Interface . . . . . 13–13
Starting the Installation: Motif Interface . . . . . 13–14
Obtaining Motif Help . . . . . 13–18
Selecting Product Options: Motif Interface . . . . . 13–19
Stopping and Restarting: Motif Interface . . . . . 13–20

Recovering from Errors . . . . . 13–21

## 14 Using the VMSINSTAL Utility

Accessing Release Notes with VMSINSTAL . . . . . 14–2
Starting the Installation Procedure . . . . . 14–4
Stopping and Restarting the VMSINSTAL Procedure . . . . . 14–6
Responding to Installation Questions . . . . . 14–7
Following Installation Progress . . . . . 14–11

## 15 After Installing DQS Software

Postinstallation Tasks: New Server Installation . . . . . 15–2
Postinstallation Tasks: New Client Installation . . . . . 15–3
Postinstallation Tasks for All Systems: New Installation . . . . . 15–4
Postinstallation Tasks: Upgrade Server Installation . . . . . 15–7
Postinstallation Tasks: Upgrade Client Installation . . . . . 15–10
Postinstallation Tasks for All Systems: Upgrade Installation . . . . . 15–11
Postinstallation Tasks: VMScluster Nodes . . . . . 15–13

# Part IV Appendixes

## A DQS User Messages

Message Section . . . . . A–2

## B DQS System Manager Messages

DQS Message Format . . . . . B–2
Submitting a Software Performance Report . . . . . B–3
Message Section . . . . . B–4

## C Server and Client System Logical Names

DQS Product Logical Names . . . . . C–2
DQS Server Logical Names . . . . . C–3
Client Logical Names . . . . . C–6

## D Using DQS With Various Symbionts


How Various Symbionts Function . . . . . D–2
Modifying Your Symbiont To Output Client Job Information . . . . . D–3


## E Files Installed or Modified on the System


Files Installed on Server Nodes . . . . . E–2
Files Installed on Client-Only Nodes . . . . . E–6
Files Modified by the Installation . . . . . E–8
Accounts and Processes Added . . . . . E–9
Installed Images . . . . . E–10


## F Sample Installations


Sample PCSI Installation on an OpenVMS System . . . . . F–2
Sample VMSINSTAL Installation on an OpenVMS AXP System . . . . . F–6


## G DQS Installation Messages


Installation Error Messages . . . . . G–2
IVP Error Messages . . . . . G–3
Error Reporting . . . . . G–4


## Glossary

## Index

## Examples


7–1 A DQS Error That Prevents a Print Job From Transferring . . . . . 7–26
7–2 A DQS Error That Stops a Client Queue . . . . . 7–28
8–1 Viewing NETSERVER.LOG Information . . . . . 8–17
8–2 Logging Client Access to a Server . . . . . 8–18
8–3 Looking at DQS$SERVER.NODES . . . . . 8–19
8–4 Enabling Client Accounting Information . . . . . 8–21
8–5 Defining a Status Message for a Queue . . . . . 8–25
8–6 Enabling the DQS Print Symbiont . . . . . 8–27
8–7 Specifying the Scanning Interval for DQS$NOTIFIER . . . . . 8–29

| | | |
|---|---|---|
| 8–8 | Specifying a Directory for PrintServer Log Files | 8–30 |
| 8–9 | Specifying the Priority of the Server Process | 8–31 |
| 8–10 | Specifying the Maximum Priority for Queuing Print Jobs | 8–32 |
| 8–11 | Specifying the Duration of Links to Inactive Clients | 8–34 |
| 8–12 | Specifying Unique Notifier Lock Names in a Nonhomogeneous VMScluster | 8–37 |
| 9–1 | Creating a Client Queue with DQS$IMPORT.COM | 9–5 |
| 9–2 | Creating Client Queues With DQS$DEFINE.COM | 9–7 |
| 9–3 | Defining Generic Client Queues | 9–10 |
| 9–4 | Defining Logical Client Queues | 9–11 |
| 9–5 | Temporarily Changing the Remote Queue | 9–13 |
| 9–6 | Setting Up a LAT Port Device | 9–18 |
| 9–7 | Changing Client Queue Spooled to LTA999: Device | 9–19 |
| 9–8 | Spooling to a LAT Port Device | 9–19 |
| 9–9 | Regulating Forms Checking on Many Client Systems | 9–24 |
| F–1 | Sample PCSI Client Installation on an OpenVMS System | F–2 |
| F–2 | Sample PCSI Server Installation on OpenVMS VAX System | F–4 |
| F–3 | Sample VMSINSTAL Client Installation on an OpenVMS AXP System | F–6 |
| F–4 | Sample VMSINSTAL Server Installation on OpenVMS VAX System | F–10 |


## Figures


| | | |
|---|---|---|
| 2–1 | Basic DQS Configuration | 2–3 |
| 2–2 | Multinode DQS Configuration | 2–5 |
| 7–1 | Overview: Primary Function of the DQS Software | 7–5 |
| 7–2 | The DQS PRINT Command Process | 7–8 |
| 7–3 | DQS QSHOW, QSET/ENTRY, and QDELETE/ENTRY Commands Processes | 7–10 |
| 7–4 | Function of the DQS Notification Service | 7–14 |
| 7–5 | Daisy-Chained Queues | 7–23 |
| 8–1 | Checking a Client Node's Authorization | 8–8 |
| 13–1 | PCSI Main Window | 13–15 |
| 13–2 | PCSI Installation Window | 13–17 |

## Tables


| | | |
|---|---|---|
| 1 | Additional Information | xvi |
| 2 | Conventions | xvii |
| 2–1 | PRINT Command Qualifiers That Function Differently with DQS Printing | 2–8 |
| 5–1 | QSET/ENTRY Command Qualifiers That Function Differently with DQS Printing | 5–4 |
| 7–1 | Summary of DQS Commands | 7–6 |
| 7–2 | Functions of the DQS Software Components | 7–11 |
| 8–1 | Summary of Server Node Management Tasks | 8–42 |
| 11–1 | Summary: What You Need to Know | 11–1 |
| 11–2 | Prerequisite Software | 11–4 |
| 12–1 | Preinstallation Task Summary | 12–1 |
| 12–2 | Process Quotas for the Installation Account | 12–9 |
| 12–3 | Disk Space Requirements | 12–11 |
| 12–4 | Minimum Required System Parameter Values | 12–15 |
| 13–1 | PCSI Installation Task Summary | 13–1 |
| 13–2 | Features You Can Request When Starting an Installation | 13–5 |
| 14–1 | VMSINSTAL Installation Task Summary | 14–1 |
| 15–1 | Task Summary | 15–1 |
| E–1 | Files Installed on DQS Server Nodes | E–2 |
| E–2 | Files Installed on DQS Client-Only Nodes | E–6 |

# Preface

## About the Software

The DEC Distributed Queuing Service (DQS) for OpenVMS Systems software enables you to print files on devices that are attached to remote DECnet systems on your network. DQS software also allows you to display the status of your jobs at any time during their processing and to delete or modify your print jobs.

The DEC Distributed Queuing Service Version 1.3 for OpenVMS systems software product replaces the following products:

- VAX Distributed Queuing Service Version 1.2 software
- DEC Distributed Queuing Service for OpenVMS AXP Version 1.2 software

This product is based on a client/server design, where two components of software (client and server) cooperate to make the print service function.

## Parts of the Manual

This manual contains:

- A user's guide that describes how to use the DQS software
- A system manager's guide that describes the concepts of the DQS software and how to configure and manage DQS client and server systems
- An installation guide that describes how to install the DQS software on OpenVMS AXP or OpenVMS VAX systems using the VMSINSTAL utility or the POLYCENTER Software Installation utility

At the back of this document are two Reader's Comments forms. If you have any comments about this document, fill out the form and mail it; your comments will be considered for any future revision of the guide.

## Audience

This manual is for:

- Anyone who wants to print files to printers on remote OpenVMS systems.

  A knowledge of the Digital Command Language (DCL) PRINT, SHOW/QUEUE, DELETE/ENTRY, and SET QUEUE /ENTRY commands is assumed.

- DECnet network or system managers who are responsible for installing, configuring, and managing the DQS software in their network.

## Document Structure

### User's Guide

Part I, the user guide, has the following chapters:

| Chapter | Describes |
|---|---|
| Chapter 1 | The DQS software and its capabilities, in brief |
| Chapter 2 | Printing files on remote nodes, including PRINT command qualifiers and possible error conditions |
| Chapter 3 | Displaying the status of print jobs |
| Chapter 4 | Deleting a print job |
| Chapter 5 | Modifying the characteristics of a print job |
| Chapter 6 | Spooling files to a client queue |

### System Manager's Guide

Part II, the system manager's guide, has the following chapters:

| Chapter | Describes |
|---|---|
| Chapter 7 | The DEC Distributed Queuing Service for OpenVMS Systems and its functions, including how the software can be configured and how the software handles error conditions |
| Chapter 8 | Configuring and managing a DQS server system |
| Chapter 9 | Configuring and managing a DQS client system |
| Chapter 10 | Quidelines for troubleshooting problems with the DQS software |

## Installation Guide

Part III, the installation guide, has the following chapters:

| Chapter | Describes |
|---|---|
| Chapter 11 | Requirements for installing the DQS software |
| Chapter 12 | Preinstallation requirements |
| Chapter 13 | Installing the DQS software using the PCSI utility |
| Chapter 14 | Installing the DQS software using the VMSINSTAL utility |
| Chapter 15 | Postinstallation tasks |

## Appendixes

Part IV has the following appendixes:

| Appendix | Describes |
|---|---|
| Appendix A | DQS messages users may see |
| Appendix B | DQS messages system managers may see |
| Appendix C | DQS server and client system logical names |
| Appendix D | How the DQS software functions with various print symbionts |
| Appendix E | Files installed and modified on your system |
| Appendix F | Sample installation logs |
| Appendix G | DQS messages installers may see |

## Associated Documentation

### DQS Information

The DEC Distributed Queuing Service for OpenVMS Systems information set consists of:

- *DEC Distributed Queuing Service for OpenVMS Systems Documentation Set*
- DQS, PRINT, QDELETE, QSET, and QSHOW entries in the OpenVMS HELP facility
- Online Command Reference Card in SYS$HELP:DQS_COMMAND_REF.PS
- Online Release Notes in SYS$HELP:DQS013.RELEASE_NOTES
- *DEC Distributed Queuing Service for OpenVMS Systems Cover Letter*
- *DEC Distributed Queuing Service for OpenVMS Systems Software Product Description (SPD)*

## Other Resources

Resources listed in Table 1 may also be useful.

**Table 1 Additional Information**

| For information on . . . | See the following . . . |
|---|---|
| The OpenVMS Digital Command Language (DCL) commands | *OpenVMS DCL Dictionary* |
| Extensions to the DCL language for PostScript printers | *DECprint Supervisor for OpenVMS User's Guide* |
| Understanding OpenVMS system error messages | *OpenVMS System Messages and Recovery Procedures Reference Manual* or the online HELP/MESSAGE facility |
| OpenVMS system management, in general | *OpenVMS System Manager's Manual* and *OpenVMS System Manager's Utilities Reference Manual* |
| The POLYCENTER Software Installation utility | *Software Integrator User's Guide* |
| How to license a product | *OpenVMS License Management Utility Manual* |

## Conventions

Table 2 lists the conventions used in this guide.

**Table 2 Conventions**

| Convention | Meaning |
|---|---|
| OpenVMS system | Means both the OpenVMS AXP operating system and the OpenVMS VAX operating system. |
| UPPERCASE TEXT | Indicates the name of a command, a file, a parameter, a procedure, or utility. |
| $ **PRODUCT** | In interactive examples, prompts or displayed text appears in a monospace font. User input appears in a bold monospace font. |
| *lowercase italics* | Indicates variables in command syntax or examples for which the user supplies a value. |
| Ctrl/x | In interactive examples, a sequence such as Ctrl/x indicates that you must hold down the Ctrl key while you press another key or a pointing device button; for example, Ctrl/C or Ctrl/Z. |
| Return | Indicates the Return key. |
| [ ] | In command formats, square brackets encloses optional values. (Do not type the brackets.)<br><br>In installation prompts, square brackets enclose a default value.<br><br>In a file specification, square brackets serve as delimiters for a directory name. |

# Part I

## User's Guide

This part describes how to use the DQS software on OpenVMS AXP and OpenVMS VAX systems. The User's Guide is intended for anyone who wants to print, track, delete, or modify jobs on a remote printer in the network.

# 1
# Using the DQS Software

## About This Chapter

This chapter:

- Provides an overview of the DEC Distributed Queuing Service for OpenVMS Systems software
- Explains how to get help when using the DQS commands

## Understanding the Software

### What DQS Can Do for You

The DEC Distributed Queuing Service for OpenVMS Systems is a software product that enables you to:

- Print files on printers connected to remote systems in your network.
- Show the status of your print jobs on remote systems.
- Delete your print jobs from remote systems.
- Set the characteristics of your print jobs on remote systems.

Using DQS software, you can:

- Print files on any printer in your network.
- Take advantage of special printers that might not be physically connected to your local system.
- Print files even if your system has no printers connected.
- Distribute reports efficiently to remote sites within your company or organization through both local and wide area networks.

### What You Need to Know

If you know how to print files on OpenVMS systems, you know how to use most DQS features. The DQS commands (PRINT, QSHOW, QSET, and QDELETE) look and function the same as the Digital Command Language (DCL) PRINT, SHOW, SET, and DELETE commands.

To use DQS software, you do not need to log in to a remote system, copy files across a network, or have proxy access or special privileges. You do not need to consider whether a print queue is local or remote.

## How DQS Works

This is what happens when you print using the DQS software:

| Stage | Description |
|---|---|
| 1 | You issue a PRINT command and queue the job to a remote system. The remote system does not have to be reachable at the time. |
| 2 | The DQS software keeps trying until it establishes a DECnet link to the remote system and transfers the print job to the remote system. |
| 3 | After job transfer, you can request status information, modify the print request, or delete the print job. |
| 4 | DQS software establishes a DECnet link to the remote system and executes your request. The remote system must be available for DQS software to complete the request. |
| 5 | The job prints on the remote printer. DQS software notifies you that the job has printed, if you include the /NOTIFY qualifier on the PRINT command. |

## Some DQS Terms

Some terms used in this guide in connection with the DQS software are:

- Client — the system at which a you reside
- Server — the system to which the printer is connected
- Client queue — the queue you use locally
- Remote queue — the queue on the server that feeds the printer

DQS software moves print jobs from client queues (on clients) to corresponding remote queues (on servers).

## For More Information

For more information on the differences between DQS commands and corresponding DCL commands, use this manual or the OpenVMS HELP facility.

For more information on how to print files, refer to the descriptions of the PRINT, SHOW QUEUE, DELETE/ENTRY, and SET QUEUE /ENTRY commands in the *OpenVMS DCL Dictionary*.

## Getting Help With DQS

### The HELP Command

To get help with the DQS commands, you can use the standard OpenVMS HELP facility. Enter:

```
$ HELP command
```

Where:

*command* Is DQS or the PRINT, QSHOW, QDELETE, or QSET command verb.

### Example

For example, if you enter the following command, text displays on your terminal that explains how to use the DQS QSHOW command.

```
$ HELP QSHOW Return
QSHOW

    Displays information about queues and jobs  that  are  currently  in
    queues.  If the queue is a DQS client queue, QSHOW also displays the
    remote queues associated with the client queue.

    o  Display characteristic names and numbers that  are  available  on
       queues (see /CHARACTERISTICS).

    o  Display form names and numbers that are available on queues  (see
       /FORM).

    Format:

      QSHOW  [queue-name]

  Additional information available:

  Parameters Command_Qualifiers
  /ALL       /BATCH     /BRIEF     /DEVICE    /FILES     /FULL      /OUTPUT
  /CHARACTERISTICS      /FORM

QSHOW Subtopic?
```

# 2
# Printing Files Using DQS Software

## About This Chapter

This chapter:

- Explains the DQS printing process in basic and multinode networks
- Shows examples of messages you receive when you enter a DQS print job
- Explains requirements for printing with forms
- Lists differences between DQS PRINT qualifiers and DCL PRINT qualifiers
- Lists some status and error messages you may see

## For More Information

If you need more information on how to use the Digital Command Language (DCL) PRINT command and its qualifiers, refer to the *OpenVMS DCL Dictionary*.

## Printing in Network Environments

### Basic Network Example

Assume that your site has a network of two systems, APPLE and BANANA:

- Node APPLE has an attached PrintServer 17/600 printer.
- The PrintServer 17/600 queue on APPLE is named Q_CHICAGO.
- Node BANANA has no attached printer.

### Without DQS Software

Without DQS software, from node APPLE, you can print a file named TRIP_REPORT.TXT on your own system by entering this PRINT command line at the DCL prompt ($):

```
$ PRINT TRIP_REPORT.TXT/QUEUE=Q_CHICAGO Return
```

Node BANANA users cannot use the printer on node APPLE.

### With DQS Software

If DQS software is installed on both nodes, node APPLE can function as a server; that is, APPLE's printer can serve users of both systems. Node BANANA is then the DQS client.

Assume that your system manager defined a client DQS queue for node BANANA that corresponds to the queue Q_CHICAGO on APPLE. From node BANANA, to queue your file to print on node APPLE, you enter the identical command line:

```
$ PRINT TRIP_REPORT.TXT/QUEUE=Q_CHICAGO Return
```

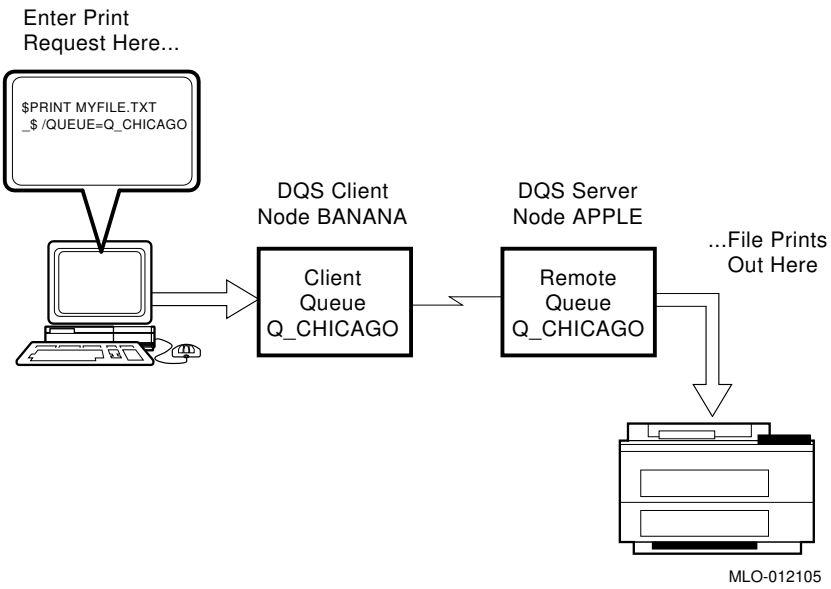
### Note

If the generic queue SYS$PRINT is defined as Q_CHICAGO, users can print files by entering PRINT *file-name*.

## Basic Configuration

Figure 2–1 illustrates how the system manager configures DQS software in this basic network.

**Figure 2–1 Basic DQS Configuration**




## Multinode Network Example

DQS software can be installed on networks of any size. Figure 2–2 illustrates a slightly more complex DQS configuration consisting of three nodes: APPLE, BANANA, and CHERRY:

- Node APPLE has an attached PrintServer 17/600 printer named Q_CHICAGO.
- Node CHERRY has an attached LN06 printer named Q_PARIS.
- Node BANANA has no attached printer.

DQS can be installed on all three nodes. Nodes APPLE and CHERRY can function as servers; that is, their printers can serve users of other systems. All three nodes can be clients; that is, they can use the printers of the server systems.

## Printing in Multinode Network

Your system manager can define client DQS queues for all three systems that send print jobs to the remote queues Q_CHICAGO and Q_PARIS. For example, your system manager can define the following (see Figure 2–2):

| Define client queues: | On client node: | To send print jobs to remote queues: | On server nodes: |
|---|---|---|---|
| CQ_EUROPE | APPLE | Q_PARIS | CHERRY |
| CQ_FRANCE | BANANA | Q_PARIS | CHERRY |
| CQ_ILLINOIS | BANANA | Q_CHICAGO | APPLE |
| CQ_USA | CHERRY | Q_CHICAGO | APPLE |

Then, from node APPLE, for example, you can queue your file to print on the LN06 printer on node CHERRY by entering:

```
$ PRINT TRIP_REPORT.TXT/QUEUE=CQ_EUROPE [Return]
```

From node BANANA you can queue your file to print on the LN06 printer on node CHERRY by entering:

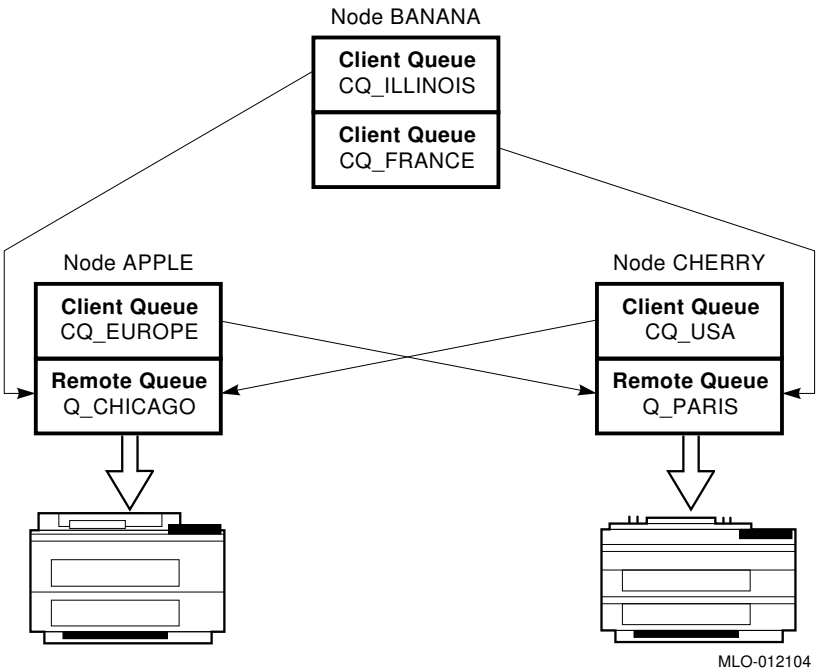
```
$ PRINT TRIP_REPORT.TXT/QUEUE=CQ_FRANCE [Return]
```

In both examples, the print job, TRIP_REPORT.TXT, is sent to the remote queue Q_PARIS on the server node CHERRY for printing.

## Multinode Configuration

Figure 2–2 illustrates how the system manager configures DQS software in this multinode network.

**Figure 2–2 Multinode DQS Configuration**




## Advantages of DQS Printing

With DQS software, you can choose to print files on the printer that is most convenient to your location. You can take advantage of different types of printers as necessary. You can also direct output to printers you know are idle.

## Understanding Print Job Messages

### Status Messages

When you queue a file to be printed, you receive one of two acknowledgment messages:

- If the client queue is empty, you see a message that indicates that the file is being transferred to the server node for printing. For example:

```
Job MARCH-INVOICE (queue CLIENT, entry 806) started on CLIENT
```

- If the client queue is not empty, you see a message like this:

```
Job MARCH-INVOICE (queue CLIENT, entry 810) pending
     pending status caused by queue busy state
```

The messages list the client queue, in this case CLIENT, and the client job number.

As with the DCL software, these messages are sent by default. You can suppress default messages by using the /NOIDENTIFY qualifier with the PRINT command.

### Print Job Entry Numbers

When you queue a file to print, the DQS software displays the job entry number of the client (that is, your) system. Job numbers printed on the header, burst, or trailer pages are client job numbers, if the DQS print symbiont or the DECprint Supervisor for OpenVMS software runs the server's printer. Otherwise, the job numbers are server job numbers.

## Printing with Forms

### Requirements

Before the DQS software transfers a print job, it checks on the form being used.

- If the form exists on both the client and the server, the file is printed.
- If the form does not exist on the client, the job cannot be queued.
- If the form does not exist on the server, the job terminates with an error and remains in the queue until you delete or requeue it.

If you queue a job with a form that is not mounted on the client, the queue manager mounts the form before the software transfers the job. This can take several minutes.

If you queue a job with a form that is not mounted on the server, the job does not print until the operator mounts the form.

## Using PRINT Command Qualifiers

### DQS Qualifier Differences

Most of the qualifiers to the PRINT command function the same with DQS printing as they do with standard OpenVMS printing. Qualifiers that function differently with DQS printing are described in Table 2–1.

**Table 2–1 PRINT Command Qualifiers That Function Differently with DQS Printing**

| Qualifier | Function Without DQS | Function With DQS |
|---|---|---|
| /AFTER=*time* | Prints job after specified time. | Transfers job to the server after specified time. |
| /DELETE | Deletes file after printing. | Deletes file on the client after transfer to the server (files are always deleted from the server's disk after printing). |
| /FORM=*type* | Uses form *type*. | Uses form *type*.<br>• The form name must be defined on both the DQS client and server.<br>• If the form is not mounted on the client, the queue manager software mounts it (this may take several minutes).<br>• If the form is not defined on the server, the job does not transfer, but remains in the queue with error status %JBC-F-NOSUCHFORM.<br>• If the form is defined differently on server and client, the job is printed with the server form. |
| /HEADER | Specifies whether header page prints with file. | Specifies whether header page prints with file. The header page includes server node and device in the file specification; the version number of the file specification may differ. |

(continued on next page)

**Table 2–1 (Cont.) PRINT Command Qualifiers That Function Differently with DQS Printing**

| Qualifier | Function Without DQS | Function With DQS |
|---|---|---|
| /HOLD | Holds job until released (by SET /ENTRY/RELEASE command). | Does not transfer job to server until released (by QSET/ENTRY/RELEASE command). |
| /NOTIFY | Broadcasts message to you after job prints. | Broadcasts messages to you after job transfers and after job prints. |
| /OPERATOR=*string* | Sends *string* to operator when job begins to print. | Sends *string* to operators on both client and server when job begins to print. |
| /PARAMETERS=(*parameter*[,...]) | Passes from 1 to 8 optional parameters to job. | Passes from 1 to 7 optional parameters to job. DQS software reserves parameter 8 for its own use. If you specify parameter 8, DQS software generates an error, and leaves the job in the client queue with an error status of DQS-F-P8RESERVED. |
| /PRIORITY=*n* | Specifies priority of print job. | Specifies priority of print job. Priority limits are configured on the server for all client users. You can request any print priority, but the server queues your job with no more than the specified maximum priority. |
| /SETUP=*module*[,...] | Extracts specified modules from device control library. | Extracts specified modules from device control library on the server; the client system's device control libraries need not be defined. |

## Examples

For example, to transfer a file to the printer queue on node CHERRY from the client queue CQ_EUROPE on node APPLE after 11:30 in the morning, use the /AFTER qualifier:

```
$ PRINT CHANGES.PS/QUEUE=CQ_EUROPE/AFTER=11:30 Return
Job CHANGES (queue CQ_EUROPE, entry 324) holding until 30-JUL-1994 11:30
```

If you want to be notified after your print job is transferred and completes, use the /NOTIFY qualifier. For example:

```
$ PRINT CHANGES.PS/QUEUE=CQ_EUROPE/NOTIFY Return
Job CHANGES (queue CQ_EUROPE, entry 548) started on queue CQ_EUROPE
Job CHANGES (queue EUROPE, entry 548) completed

Print job CHANGES (queue CQ_EUROPE, entry 548) completed at 30-JUL-1994 10:24
$
```

Each "completed" message means something different.

- The first indicates that the file transfer is complete.
- The second indicates that the job printed on the server node.

The messages may not appear immediately; the actual timing depends on the system load and backlog in the queues.

## Understanding PRINT Command Error Messages

### Error Messages

If a DQS print job in a client queue fails to transfer to the server, the job remains in the client queue with the message "terminated with error status" and an associated error status. To examine the error status, use the QSHOW command (see Chapter 3).

If you specify the /NOTIFY qualifier, you receive an error message when your print job fails to transfer. For example:

```
Job TRIP_REPORT (queue CQ_ILLINOIS, entry 341) terminated with error status
%status, message-text
```

You must either delete the job or release it for another try. The QDELETE/ENTRY command (for deleting jobs from a queue) is described in Chapter 4.

### Common DQS Error Messages

Following are some common DQS error messages with an explanation and a recovery action.

- %DQS-F-BADQUEUE—The remote queue on the server system is not accessible to DQS clients. Ask the system manager to modify the remote queue on the server system. (The DQS$QUEUE_*remote-queue* logical name needs to be defined.)
- %JBC-F-NOSUCHFORM—You specified a form that is not defined on the server. Enter the job again, using a different form or specifying a queue where the form is defined.
- %DQS-F-P8RESERVED—You used the /PARAMETERS qualifier and specified all eight parameters. Reenter the print request with less than eight parameters.

Refer to Appendix A for a complete list of messages.

# 3

# Viewing the Status of Print Jobs

## About This Chapter

This chapter:

- Compares the DQS QSHOW command with the DCL SHOW QUEUE command
- Describes how to check the status of a print job using the QSHOW command and how to interpret the associated output
- Compares QSHOW qualifiers with DCL SHOW QUEUE qualifiers
- Explains QSHOW error messages

## The QSHOW Command

### Comparison with DCL Command

| This command . . . | Enables you to . . . |
|---|---|
| DCL SHOW QUEUE | Display the status of queues on your local system. |
| DQS QSHOW | Display the status of both local and remote print queues. |

Using the DCL SHOW QUEUE command to view the status of a DQS client queue works, but it does not show the status of the associated remote queue. For that information, use the QSHOW command. You can use the QSHOW command to examine both client (local) and server (remote) queues.

## Checking the Status of a Print Job

### QSHOW Command

To check the status of a print job, enter the following command at the DCL prompt on the client node:

```
$ QSHOW client-queue-name Return
```

### Sample Output

The following examples use the same node names and queues used in the DQS client/server examples from Chapter 2.

For example, you are working on node APPLE, and you queued the file TRIP_REPORT.PS to the client queue CQ_EUROPE to print on the printer on node CHERRY. To check the status of your print job, enter:

```
$ QSHOW CQ_EUROPE Return
```

The command output may look like this:

```
** Remote queue - [ CQ_EUROPE => Q_PARIS, on CHERRY:: ]
Printer queue Q_PARIS, on CHERRY::

  Jobname         Username     Entry       Blocks   Status
  -------         --------     -----       ------   ------
    (5 intervening jobs containing 2019 blocks)
  TRIP_REPORT     JONES        1229            98   Pending

============================to be transferred=========================
Server queue CQ_EUROPE, on APPLE::, mounted form LN06_DEFAULT (stock=DEFAULT)
```

The information in this example tells you:

- The name of the client node is APPLE. The name of the server node is CHERRY.
- The queue shown as the "server queue" is the client queue CQ_EUROPE on node APPLE. ("Server" refers to the OpenVMS concept of server queues, not the DQS concept of server nodes.)
- The queue shown as "remote queue" is the remote queue Q_PARIS on node CHERRY.
- Five print jobs are currently ahead of yours in the remote queue. They total 2019 blocks of data. You can use this information to estimate how long before your job prints.

- Your print job is in the remote queue on node CHERRY awaiting printing (status "Pending"). If your job is printing, the status is "Printing."
- The form on the client queue is named LN06_DEFAULT. You are informed if the client queue is stopped or stalled.
- You have no jobs awaiting transfer to the server. Like the DCL SHOW QUEUE command, the DQS QSHOW command normally shows only your jobs, unless you are a privileged user and you specify the /ALL qualifier. In any event, you cannot see other users' jobs in the client queue, unless the queue is world readable.

If you queue another job and check the status, you may get results like this:

```
$ QSHOW CQ_EUROPE Return

** Remote queue - [ CQ_EUROPE => Q_PARIS, on CHERRY:: ]
Printer queue Q_PARIS, on CHERRY::

  Jobname         Username     Entry       Blocks   Status
  -------         --------     -----       ------   ------


============================to be transferred==========================
Server queue CQ_EUROPE, on APPLE::, mounted form LN06_DEFAULT (stock=DEFAULT)
  TRIP_REPORT     JONES        1234            98   Processing
```

This example indicates that your job is in the process of being transferred to the server node CHERRY. If jobs are ahead of yours in the client queue on node APPLE, the status is "Pending."

## Using QSHOW Command Qualifiers

### DQS QSHOW Qualifiers

The DQS QSHOW command uses the same qualifiers as the DCL SHOW QUEUE command, plus one more.

QSHOW also uses the qualifier /QUEUE=*queue-name* to support the /FORM and /CHARACTERISTICS qualifiers.

### /FORM Qualifier

- To show the forms available on a local system, enter this DCL command line:

  $ **SHOW QUEUE/FORM** Return

- To show the forms available on the DQS client and server systems, enter:

  $ **QSHOW /QUEUE=*queue-name* /FORM** Return

### /FORM Example

In this example, CQ_EUROPE is used as the *queue-name*:

```
$ QSHOW /QUEUE=CQ_EUROPE /FORM Return
```

You see output like the following:

```
Remote node CHERRY::
Form name                           Number   Description
---------                           ------   -----------
A4_12 (stock=PLAIN_PAPER)             1129   A4 12-Pitch with margins
DCPS$DEFAULT (stock=DEFAULT)          1115   DCPS default

Local node APPLE::
Form name                           Number   Description
---------                           ------   -----------
A4_12 (stock=PLAIN_PAPER)             1129   A4 12-Pitch with margins
DCPS$DEFAULT (stock=DEFAULT)          1115   DCPS default
DEFAULT                                  0   System-defined default
KPS$DEFAULT (stock=DEFAULT)           1183   KPS default
LANDSCAPE (stock=DEFAULT)               10   132 by 66 (landscape)
LASERWRITER (stock=DEFAULT)             30   APPLE LASERWRITER
LCG01$FORM (stock=DEFAULT)             455   LCG01
LN01S (stock=DEFAULT)                   60   64000 by 66 (LN0S )
LPS$$FORM                             1105   PostScript Default
LTR_10 (stock=PLAIN_PAPER)            1124   10-Pitch, no margins
LTR_12 (stock=PLAIN_PAPER)            1127   12-Pitch with margins
PORTRAIT (stock=DEFAULT)                20    80 by 60 (portrait)
PSBLASER (stock=DEFAULT)                47   postscript laser
PS_PLAIN (stock=PLAIN_PAPER)          1132   PostScript (Macintosh)
WPSPLUS (stock=DEFAULT)               1101   WPS-PLUS/VMS
$
```

The QSHOW command lists the forms mounted on both the client and the server nodes, referred to as "local" and "remote" nodes, respectively, in the command output.

## Understanding QSHOW Error Messages

### QSHOW Error Messages

The DQS QSHOW command uses the same error messages as the DCL SHOW QUEUE command. In addition, the QSHOW command has its own messages for handling problems unique to the DQS software.

### Server Unreachable

For example, if the server node is not currently reachable, the DQS software cannot check the status of the remote queue. You receive an explanation of the problem, like the following:

```
$ QSHOW CQ_EUROPE Return

***** Remote queue CHERRY::Q_PARIS not accessible *****
    %SYSTEM-F-UNREACHABLE, remote node is not currently reachable

==============================to be transferred==================
Server queue CQ_EUROPE, stalled, mounted form DEFAULT

  Jobname         Username     Entry  Blocks  Status
  -------         --------     -----  ------  ------
  TRIP_REPORT     JONES         1366      98  Processing
  Q2-FINANCIALS   JONES         1367       1  Pending
```

### Invalid Remote Queue

Another possible error is that the remote queue is defined incorrectly. DQS software displays the following messages:

```
$ QSHOW CQ_EUROPE Return
%DQS-W-MSG_REMOTE, Remote system error:
-DQS-F-BADQUE, Specified queue not valid for DQS access.

==============================to be transferred==================
Server queue CQ_EUROPE, stopped, mounted form DEFAULT
 Remote system error:  Specified queue not valid for DQS access.

  Jobname         Username     Entry  Blocks  Status
  -------         --------     -----  ------  ------
  Q2-FINANCIALS   JONES         1467       1  Pending
```

In this case, either the system manager on the server node must redefine the remote queue or the system manager on the client node must redirect the client queue to a valid remote queue.

### For More Information

Refer to Appendix A for a complete list of messages.

# 4

# Deleting Print Jobs From a Queue

## About This Chapter

This chapter:

- Compares the DQS QDELETE/ENTRY command with the DCL DELETE/ENTRY command
- Describes how to delete a print job using the QDELETE /ENTRY command
- Lists some QDELETE/ENTRY error messages

## The QDELETE/ENTRY Command

### Comparison with DCL Command

| This command . . . | Enables you to . . . |
|---|---|
| DCL DELETE/ENTRY | Delete a job from a local queue. |
| DQS QDELETE/ENTRY | Delete either locally or remotely queued print jobs. |

The QDELETE/ENTRY command functions exactly like the DELETE/ENTRY command. (These commands do not delete files; they only remove jobs from a print queue.) You can use the DQS QDELETE/ENTRY command instead of the DCL DELETE/ENTRY command for all your print jobs.

## Deleting a Print Job

### You Need To Know

To delete a print job, you must know:

- The name of the queue from which you want to delete the job
- The job's entry number in the queue

The PRINT command displays this information by default when you queue a file for printing. You can also obtain this information by using the QSHOW command (see Chapter 3).

### Example

For example, if you queued the file TRIP_REPORT.PS to client queue CQ_EUROPE to print on node CHERRY, and the job is entry 1024, delete the print job by entering:

```
$ QDELETE/ENTRY=1024 CQ_EUROPE Return
```

The DQS software removes the job from the client queue. If the job has already transferred to the remote queue, the DQS software removes the job from the remote queue.

### /NOTIFY Qualifier

If you specify the /NOTIFY qualifier when you queue the job for printing, the DQS software notifies you when the job is deleted.

```
$ QDELETE/ENTRY=341 CQ_EUROPE /NOTIFY Return
Print job TRIP_REPORT (queue CQ_EUROPE, entry 341) deleted
```

## Understanding QDELETE Error Messages

### QDELETE Error Messages

The DQS QDELETE command uses the same error messages as the DCL DELETE command. In addition, the QDELETE command has its own messages for handling problems unique to the DQS software.

### Server Unreachable

If the server node is not currently reachable, the QDELETE /ENTRY command fails and you receive an explanation, like the following:

```
$ QDELETE/ENTRY=1067 CQ_EUROPE Return
%DQS-I-NOJOBDELETED, No jobs deleted
%DQS-I-INACCESSIBLE, Remote queue CHERRY::Q_PARIS
    is inaccessible
```

### Invalid Remote Queue

If you use the QDELETE/ENTRY command to delete a job from a queue on a remote system that is not defined as a valid DQS remote queue, DQS software displays messages like the following:

```
$ QDELETE/ENTRY=1194 CQ_CANADA Return
%DQS-W-MSG_REMOTE, Remote system error:
-DQS-F-BADQUE, Specified queue not valid for DQS access.
```

In this case, either the system manager on the server node must redefine the remote queue or the system manager on the client node must redirect the client queue to a valid remote queue.

### For More Information

Refer to Appendix A for a complete list of messages.

# 5

# Modifying the Characteristics of a Job

## About This Chapter

This chapter:

- Compares the DQS QSET/ENTRY command with the DCL SET/ENTRY command
- Describes how to modify a print job using the QSET/ENTRY command
- Describes the DQS command qualifiers with modified behavior
- Lists some QSET error messages

## The QSET/ENTRY Command

### Comparison with DCL Command

| This command . . . | Enables you to . . . |
|---|---|
| DCL SET/ENTRY | Change the current status or attributes of a job that is not currently executing in a queue. |
| DQS QSET/ENTRY | Changes the current status for either locally or remotely queued print jobs not executing in a queue. |

You can use the DQS QSET/ENTRY command for modifying all your print jobs. You must specify a queue name with the QSET/ENTRY command.

## Modifying a Print Job

### You Need to Know

Before you can modify a print job, you need to know:

- The name of the queue that contains the job you want to modify
- The job's entry number in that queue

The DQS PRINT command displays this information by default when you queue a file for printing. You can also obtain this information by using the QSHOW command (described in Chapter 3).

### QSET/ENTRY Example

For example, you queued the file CHANGES.PS to client queue CQ_EUROPE to print on node CHERRY and requested two copies, as follows:

```
$ PRINT CHANGES.PS/QUEUE=CQ_EUROPE/COPIES=2 Return
Job CHANGES (queue CQ_EUROPE, entry 1324), pending
```

Now you change your mind and want three copies. You can modify the job by entering:

```
$ QSET/ENTRY=1324 CQ_EUROPE/COPIES=3 Return
```

The DQS software modifies the job in the client queue. If the job already transferred to the remote queue, the DQS software modifies the job in the remote queue. However, DQS software cannot modify a job that is moving from a client queue to a remote queue or a job that is actually printing.

## Using QSET Command Qualifiers

### QSET Qualifier Differences

Most of the qualifiers to the QSET/ENTRY command act the same as they do with the standard DCL SET/ENTRY command. Those qualifiers that have modified functions when used with the QSET/ENTRY command are described in Table 5–1.

**Table 5–1 QSET/ENTRY Command Qualifiers That Function Differently with DQS Printing**

| Qualifier | Function Without DQS | Function With DQS |
|---|---|---|
| /AFTER=*time* | Prints job after specified time. | Transfers job to the server after specified time. If already transferred, prints job after specified time. |
| /FORM=*type* | Uses form *type*. | Uses form *type*; form name or number must be defined on both client and server. |
| /HEADER | Specifies whether header page is printed with file. | Specifies whether header page is printed with file. Page displays server node and device in the file specification; file version number may differ. |
| /HOLD | Holds job until released (by SET /ENTRY /RELEASE command). | Does not transfer job to server until released. If already transferred, does not print until released (by QSET /ENTRY/RELEASE command). |
| /NODELETE | Cancels file deletion for job queued with /DELETE qualifier. | While job is still in client queue, this qualifier works correctly; after the job is transferred to the remote queue, command fails and returns an error. |

(continued on next page)

**Table 5–1 (Cont.) QSET/ENTRY Command Qualifiers That Function Differently with DQS Printing**

| Qualifier | Function Without DQS | Function With DQS |
|---|---|---|
| /PARAMETERS =(*parameter[,...]*) | Passes from 1 to 8 optional parameters to job. | Passes from 1 to 7 parameters to job. The DQS software reserves parameter 8 for its own use. If you specify parameter 8, the command fails and returns an error. |
| /PRIORITY=*n* | Specifies priority of print job. | Specifies priority of print job. Priority limits are configured on server nodes for all client users. You can request any print priority, but the server queues your job with no more than the allowed maximum priority. |
| /REQUEUE= *queue-name*[:] | Queues job to another printer queue. | If the job has not yet transferred to the server node, you can requeue it to any other queue. If the job already transferred (but has not printed), you can requeue it to another queue on the same server. |

## Understanding QSET Error Messages

### QSET Error Messages

If the QSET/ENTRY command fails, the job's specifications are unchanged and the DQS software returns an error message. Some common messages are:

- If the server node is not currently reachable, the QSET /ENTRY command fails and returns the following messages:

  ```
  %DQS-I-NOJOBSET, no jobs set
  %DQS-I-INACCESSIBLE, remote queue queue is inaccessible
  ```

- If you specify a queue that points to an invalid remote queue, the command fails and returns the following messages:

  ```
  %DQS-W-MSG_REMOTE, remote system error:
  -DQS-F-BADQUEUE, specified queue not valid for DQS access
  ```

  The server node system manager must modify the remote queue.

- If you try to modify a job while it is being transferred from the client queue to the remote queue, the request fails with an error status:

  ```
  %JBC-E-EXECUTING
  ```

- If you specify a form that is not defined on the client, the command fails and returns the error status:

  ```
  %JBC-E-NOSUCHFORM
  ```

- If you attempt to modify the status of a print job with the QSET /ENTRY/NODELETE command after the job has transferred to the remote queue, the command fails and returns the following messages:

```
%DQS-W-MSG_REMOTE, remote system error:
-DQS-E-NODELETE, /NODELETE is allowed only on local jobs
%DQS-I-NOJOBSET, no jobs set
```

- If you use the /PARAMETERS qualifier and specify eight parameters for a job that already transferred, the command fails and returns the following messages:

```
%DQS-W-MSG-REMOTE, remote system error:
-DQS-F-P8RESERVED, parameter 8 is reserved for DQS on remote systems
%DQS-I-NOJOBSET, no jobs set
```

**For More Information**

Refer to Appendix A for a complete list of messages.

# 6

# Spooling Files to a Client Queue

## Spooling Files

The OpenVMS operating system allows you to assign a logical device in a source program to a physical device. To spool files to a client queue:

1. The system manager creates a LAT port and sets the LAT port spooled to the DQS client queue.
2. You specify the LAT port in a source program as the physical device.

DQS software places the print output in a temporary file, enters the file in the client queue, then transfers the file to the corresponding remote queue.

## Example

In a FORTRAN program, the statement PRINT directs output to the terminal unless otherwise directed to a logical device. If the LAT port is spooled to a client queue, you can direct program output to a remote printer.

For example, if the LAT port LTA999: is spooled to client queue CQ_USA (that is, the system manager used the command line SET DEVICE LTA999/SPOOLED=CQ_USA), you can spool output to the remote printer by entering:

```
$ DEFINE FOR$PRINT LTA999: Return
```

When you execute your FORTRAN program, its output is spooled from LTA999 to the client queue CQ_USA.

## Colon Required

The colon following LTA999: is mandatory. If omitted, the file, LTA999.LIS is created in the current directory.

# Part II

## System Manager's Guide

This part describes how to configure and manage the DQS software on OpenVMS AXP and OpenVMS VAX systems. The System Manager's Guide is intended for the network or system manager.

# 7

# Understanding the DEC Distributed Queuing Service

## About This Chapter

This chapter describes:

- An overview of the design of the DEC Distributed Queuing Service for OpenVMS Systems (DQS) product
- The primary functions of the DQS software
- The functions of the DQS commands: PRINT, QSHOW, QSET/ENTRY, and QDELETE/ENTRY
- The functions of DQS software components
- The DQS notification service and how it works
- The DQS symbionts
- The use of forms with the DQS software
- The DQS server account
- Some configuration issues
- Types of errors that may occur
- How the DQS software handles security issues
- The processes created by the software
- How the DQS software supports devices
- The use of a batch job with DQS software

## Overview of the DQS Software

### What Is DQS Software?

The standard OpenVMS queue system enables you to use the OpenVMS PRINT command to print files on output devices that are connected to your local OpenVMS node.

The DQS software is an OpenVMS layered product that extends the standard OpenVMS queue system to a distributed system environment. The DQS software enables you to use the OpenVMS PRINT command to print files on output devices that are connected to remote OpenVMS nodes in your network. You can also delete, and obtain or change the status of your DQS print jobs.

### Why Use DQS Software?

Installing and configuring the DQS software in your network can provide:

- Printing services to workstation users (and others) who do not have attached printers
- The sharing of expensive or unique printing devices among the users in your distributed system
- The ability for your users to distribute reports efficiently to remote sites within your company or organization, because DQS software works in both wide area and local area networks

### Understanding DQS Clients and Servers

The DQS software is based on a client/server design, which consists of two cooperating software components (client and server).

The following terms are associated with DQS servers:

| | |
|---|---|
| Server nodes | Nodes in your network that have connections to output devices (for example, printers) that you want to make available to the users on other nodes (client nodes) in your network. |

| | |
|---|---|
| Remote queue | DQS queue on the server node that directs print jobs from client nodes to the connected printer. (Standard OpenVMS print queues are queues that direct print jobs to printers connected to the local node.) |
| Server software | DQS software that you must install and configure on server nodes. |

The following terms are associated with DQS clients:

| | |
|---|---|
| Client nodes | Nodes whose users are able to print files on printing devices that are connected to remote DQS server nodes. |
| Client queue | DQS queue on the client node that directs print jobs to the associated remote queue on the DQS server node. |
| Client software | DQS software that you must install and configure on client nodes. |

A node can be only a client, or both a client and a server. Each client can be configured to send print jobs to any server. Similarly, each server can be configured to accept print jobs from any client. Chapters 8 and 9 describe the procedures for configuring servers and clients, respectively.

## Using DQS Software in a Cluster

In a VMScluster environment, a server is potentially every node in a cluster. Similarly, a client is potentially every node in a cluster. Therefore, if a user logs on to different cluster members at different times, the DQS software is always available.

To use the DQS software in a cluster environment:

1. Install the DQS software.
2. License each node in the cluster to be a DQS node.
3. Invoke DQS$STARTUP.COM from each cluster member node.

While DQS software can be used on selected nodes of a cluster, doing so does not take advantage of the flexibility of a cluster.

## Network Requirements for DQS Software

DQS software uses the DECnet network to communicate between clients and servers and to perform its functions. The DQS software requires:

- The client and server to communicate for the print request to be processed by the server.

  However, users can queue print jobs at any time, even if the client system is not in communication with the desired server system. Similarly, a server can continue to process print jobs, even if the client from which the job was received is not in communication with the server.

- A connection between a client and a server remains long enough to transfer the complete file (or files) to be printed.

  If the connection is broken before the entire print job is transferred to the server, the DQS software attempts to transfer the entire print job after the connection is reestablished.

- Communication must exists between the client and the desired server to examine a remote queue or to delete or modify a job in a remote queue.

- A connection must be established from the server to the user's client system for a user to receive job notification messages (described in the section titled The DQS Notification Service).

## Phase IV and Phase V Networks

The DQS software can be used in both DECnet Phase IV and DECnet/OSI Phase V networks. DQS software also operates in properly configured networks that use both networking products. DECnet/OSI systems can be set up to use synonyms to communicate with Phase IV systems.

DECnet Phase IV networks limit node names to six characters. DECnet/OSI Phase V networks allow the use of an expanded node name (full names). DQS clients and servers can handle node names with up to 150 characters with no embedded blank spaces.

## Primary Functions of the DQS Software

### Overview

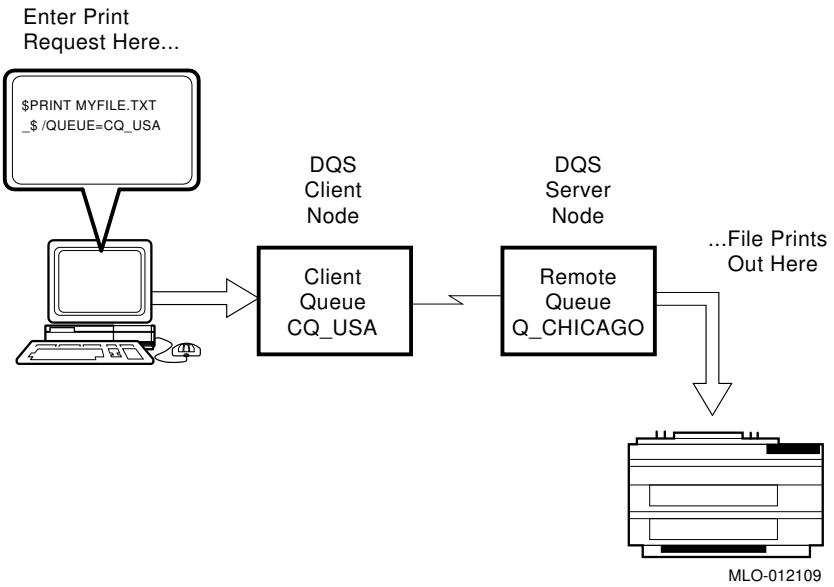
The major functions of the DQS software are initiated by the PRINT, QSHOW, QSET/ENTRY, and QDELETE/ENTRY commands. DQS software uses:

- The standard OpenVMS PRINT command to initiate DQS print requests
- Modified versions of the OpenVMS SHOW, SET, and DELETE commands to show, modify, and delete DQS print requests

### Primary Functions

Figure 7–1 shows the primary functions of the DQS software.

**Figure 7–1 Overview: Primary Function of the DQS Software**



## The Process

The stages shown in Figure 7–1 are summarized as follows:

| Stage | Description |
|---|---|
| 1 | A user enters a PRINT command on a client node specifying a DQS client queue. |
| 2 | DQS software copies the file to be printed from the client queue to the server node, and enters the print job in the remote queue on the server. |
| 3 | The remote queue sends the file to the server's output device and the file is printed. |

## Summary of DQS Commands

Most of the standard OpenVMS command qualifiers are fully supported for the DQS versions of the PRINT, SHOW, SET, and DELETE commands. Table 7–1 summarizes the DQS commands.

**Table 7–1 Summary of DQS Commands**

| Command | Function |
|---|---|
| PRINT | The OpenVMS PRINT command places a print job in a client queue on a client system. The DQS symbiont transfers the print request to the associated server system, where it is placed in the associated remote queue and directed to an output device for printing. |
| QSHOW | Displays the status of a DQS print job that is in either a client or a remote queue. |
| QDELETE /ENTRY | Prevents a print job from being processed or terminates its processing. |
| QSET/ENTRY | Modifies the parameters of a print job after the job has been queued but before it has been processed. |

## For More Information

Refer to Part I for more information on the DQS commands and their qualifiers.

## Functions of the DQS Commands

### PRINT Command Functions

Figure 7–2 shows the functions of the PRINT command using a DQS queue.

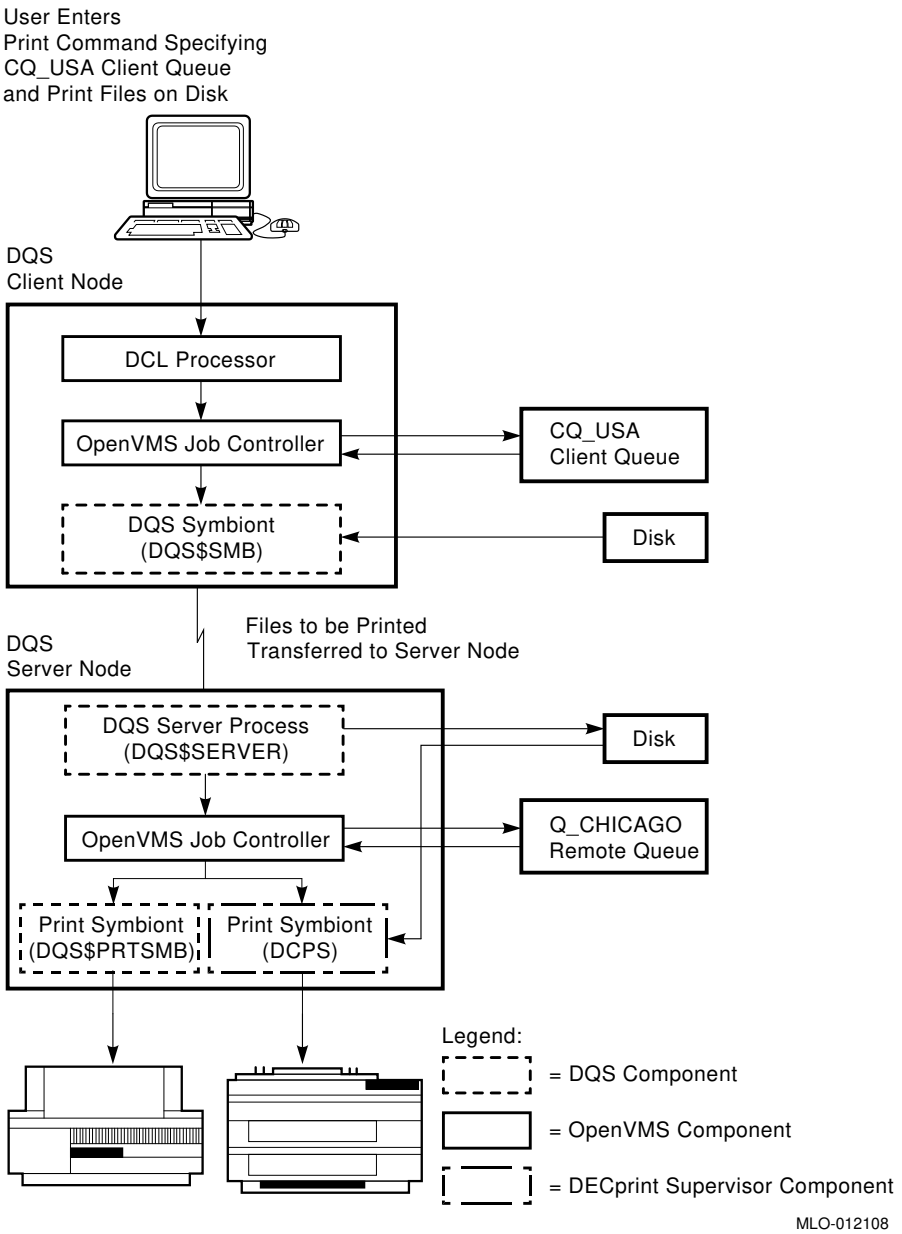
### The Process

The stages in the PRINT process are summarized as follows:

| Stage | Description |
|---|---|
| 1 | A user issues a PRINT command specifying the DQS client queue CQ_USA. |
| 2 | The standard OpenVMS DCL command language intepreter interprets the PRINT command. |
| 3 | The OpenVMS job controller in the client queue on the client node queues the print job to the DQS client queue. |
| 4 | The client symbiont, DQS$SMB, takes the print job from the client queue and transfers the job to the DQS$SERVER process on the server node. |
| 5 | The DQS$SERVER process receives the print job from the client symbiont, writes the file to a disk on the server node, and places the print job in the remote queue Q_CHICAGO. |
| 6 | The print symbiont on the server retrieves the file when the file is ready to be printed and directs the file to the output device for printing. |

After the printer prints the file, the DQS$NOTIFIER process on a server deletes the file from the DQS server account (see the section titled The DQS Server Account). If a user requests notification of job completion, the DQS$NOTIFIER attempts to notify the user before it deletes the files associated with the print job.

**Figure 7–2 The DQS PRINT Command Process**



## Other DQS Commands

Figure 7–3 shows the functions of the QSHOW, QSET/ENTRY, and QDELETE/ENTRY commands.

## The Process

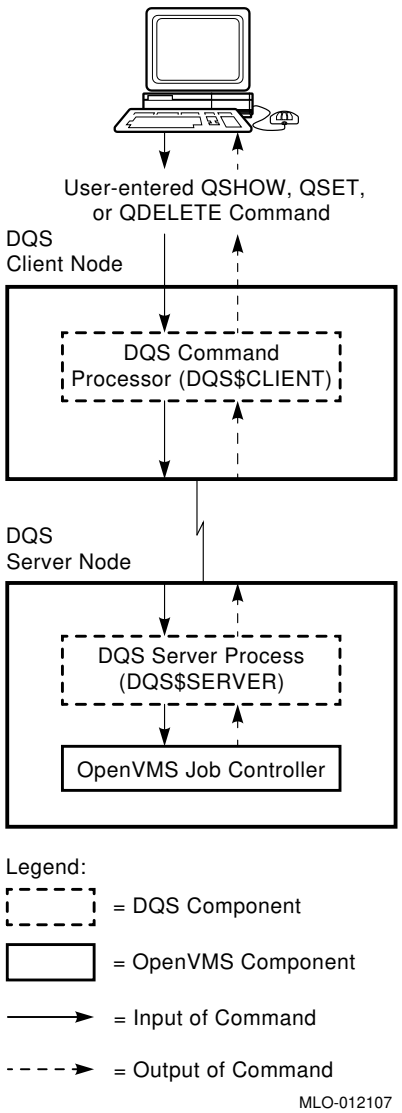
The stages in the DQS QSHOW, QSET/ENTRY, and QDELETE /ENTRY command processes are summarized as follows.

| Stage | Description |
|---|---|
| 1 | The DQS command interpreter DQS$CLIENT interprets the command and transfers the request to the DQS$SERVER process on the server node. |
| 2 | The DQS$SERVER process receives the command request from the client node and passes the request to the OpenVMS job controller. |
| 3 | The OpenVMS job controller performs the command request and returns the appropriate information to the DQS$SERVER process. |
| 4 | DQS$SERVER then transfers this information to the DQS$CLIENT process on the client node. |
| 5 | DQS$CLIENT formats the information and directs the command output to the user who initiated the command. |

## Commands with OpenVMS Queues

The DQS QSHOW, QSET/ENTRY, and QDELETE/ENTRY commands also work on standard OpenVMS print queues. For an entry in a standard OpenVMS queue, QSHOW, QSET/ENTRY, and QDELETE/ENTRY do not send information to a server.

**Figure 7–3 DQS QSHOW, QSET/ENTRY, and QDELETE/ENTRY Commands Processes**



## Functions of DQS Software Components

### Summary of Components

Table 7–2 summarizes the major software components of the DQS software and describes their functions.

**Table 7–2 Functions of the DQS Software Components**

| Component | Name | Description |
|---|---|---|
| DQS command interpreter | DQS$CLIENT | The DQS command interpreter is located on clients and interprets the DQS QSHOW, QSET/ENTRY, and QDELETE /ENTRY commands.<br>The standard OpenVMS command language interpreter interprets the PRINT command. The standard OpenVMS batch/print system enters the print job into a client queue. |
| DQS symbiont | DQS$SMB | The DQS symbiont runs on clients. Each client queue is serviced by an OpenVMS process that runs the DQS$SMB image. The DQS symbiont process takes print jobs from a client queue and transfers the jobs to a process on the associated server. |
| DQS server program | DQS$SERVER | The DQS server program is located on servers and receives print jobs from the DQS symbiont DQS$SMB on clients. The server program writes the file(s) to be printed to the server disk and places the print request in the associated remote queue.<br>DQS$SERVER also processes QSHOW, QSET/ENTRY, and QDELETE/ENTRY command requests received from clients.<br>The DQS server process is also located on clients to receive job completion messages from the DQS$NOTIFIER process on servers. DQS$SERVER on a client broadcasts job completion messages sent by the DQS$NOTIFIER to each user who owns a particular print job that has completed. |

(continued on next page)

**Table 7–2 (Cont.) Functions of the DQS Software Components**

| Component | Name | Description |
|---|---|---|
| Notification program | DQS$NOTIFIER | The DQS notification program is located on servers and periodically scans DQS print jobs in the remote queues to determine which print jobs have completed. After a print job completes, DQS$NOTIFIER informs the DQS$SERVER process on the client from which the print job originated if /NOTIFY is specified. |
| Modified OpenVMS print symbiont | DQS$PRTSMB | The DQS print symbiont, DQS$PRTSMB, is located on servers and is a modified version of the standard OpenVMS print symbiont, PRTSMB. The modified DQS symbiont should replace the standard OpenVMS print symbiont so that client-specific information is printed on the flag, burst, and trailer pages of the print output. See The DQS Symbionts for more information on the DQS symbiont. |

## The DQS Notification Service

### Sending Messages to Users

If you specify /NOTIFY qualifier with the PRINT command, the notification service sends job completion messages when print jobs have completed.

DQS software returns two notification messages:

- The first when the user's print job has transferred to the server.
- The second after the job has printed on the server.

### Deleting Jobs from Remote Queue

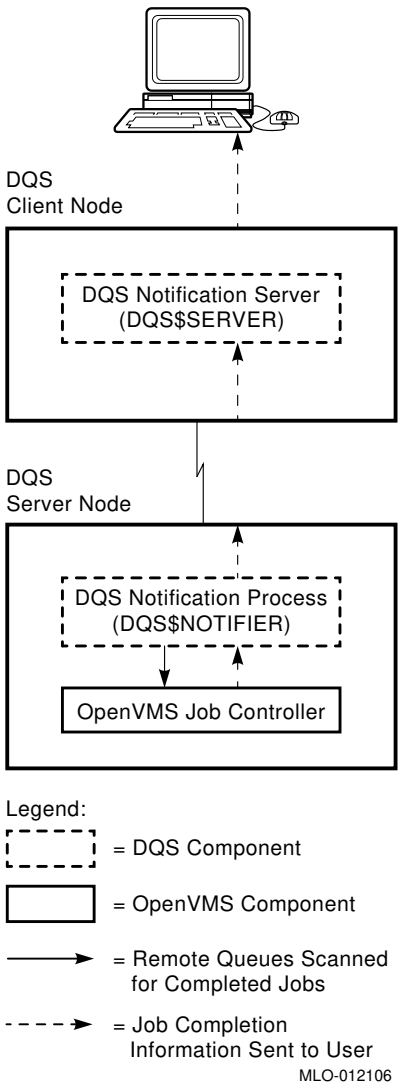
The notification process on the server node, DQS$NOTIFIER, also has the task of deleting completed print jobs from the remote queue. The DQS$SERVER program submits all DQS print jobs with the /RETAIN=ALWAYS qualifier. DQS print jobs remain in DQS queues after the job is printed until the DQS$NOTIFIER deletes them.

If a user requests notification of job completion, DQS$NOTIFIER attempts to notify the user before it deletes the files associated with the user's print job.

### How It Works

Figure 7–4 shows how the DQS notification service works.

**Figure 7–4 Function of the DQS Notification Service**




## The Notification Process

The DQS notification process shown in Figure 7–4 works like this:

| Stage | Description |
|---|---|
| 1 | The DQS notification process on the server, DQS$NOTIFIER, periodically scans DQS jobs in the remote queues to determine which print jobs have completed. |
| 2 | When a print job completes, the DQS$NOTIFIER checks if the user specified /NOTIFY and sends a completion message to the DQS$SERVER process on the client node. |
| 3 | The DQS$SERVER on the client receives the job completion message and a message is sent to the user who owns the job that has completed. |

## For More Information

For more information on using the /NOTIFY qualifier with the PRINT command, refer to Chapter 2 and the *OpenVMS DCL Dictionary*.

## The DQS Symbionts

### Overview

The DQS software contains two symbionts DQS$SMB and DQS$PRTSMB, which are described in this section.

### DQS$SMB

DQS$SMB runs on client systems. The symbiont takes DQS print jobs from client queues and transfers them to the associated server node.

### DQS$PRTSMB

DQS software provides its own print symbiont, DQS$PRTSMB, which is a modified version of the standard OpenVMS print symbiont, PRTSMB. (The DQS$PRTSMB symbiont is upwardly compatible with the OpenVMS PRTSMB symbiont.) DQS$PRTSMB causes the client job number, the client node name, and the time a print job was queued on the client to be printed on the flag, burst, and trailer pages of DQS print output. With the DQS print symbiont controlling each remote queue on a server, the users of that server see the same job information on their print output that they saw displayed when they entered the print request on the client node.

If the standard OpenVMS print symbiont does not handle banner pages to your satifiaction, replace it with the DQS print symbiont, DQS$PRTSMB. If DQS$PRTSMB does not control a remote queue on a server, the standard OpenVMS print symbiont prints the job number of the print job in the remote queue, as well as the time that the print job was queued on the server on the DQS print output.

DQS$PRTSMB can only replace the OpenVMS print symbiont.

### For More Information

For a description of the procedures for enabling the DQS print symbiont to control remote queues, see Chapter 8.

Refer to Appendix D for information on how DQS$PRTSMB relates to other symbionts.

## Defining Forms on Client and Server Nodes

### Overview

You must define all forms that you intend to specify on both the client and its associated servers. To synchronize form definitions on your clients and servers, you can run the DQS$CLIENT program on the client system (see Chapter 9).

### No Form on Server

If you attempt to print using a form that is not defined on the associated server, the client queue accepts the job and the system prompt returns ($). However, your print job remains in the client queue with the error status %DQS-F-NOSUCHFORM. A QSHOW command specifying the client queue where you queued the job shows the retained job and the error message.

### Different Values

Because DQS software transfers only a form's name from a client to its server, a form may have different values on different DQS systems. When the same form is defined differently throughout your network, the server's form definition is used to print a job.

Digital strongly recommends that you give forms with the same name the same definitions throughout your network. However, you may want to have different form definitions because it enables users on client systems to maintain their local form definitions to use when printing locally.

### For More Information

Chapter 9 describes the procedures for synchronizing form definitions.

## The DQS Server Account

### Overview

The DQS installation procedure creates a DQS account, DQS$SERVER, and its associated directory [DQS$SERVER] on all DQS client and server nodes.

### On a Server

A DQS server process is run from the DQS$SERVER account when a connection request is received from a client node. The directory associated with this account [DQS$SERVER] stores files received from client nodes until the files are printed on the server node's output device. After a print job completes, the files associated with the job are deleted from the server account directory.

### On a Client

The DQS$SERVER account is located on clients to receive job completion messages from the DQS$NOTIFIER process on the server. DQS$SERVER on a client broadcasts job completion messages to each user who owns a completed print job, if the user specified /NOTIFY on the PRINT command.

## The Significance of Parameter 8 in DQS Printing

### Overview

The OpenVMS PRINT command allows you to pass from one to eight optional parameters with a print job. DQS software uses parameter 8 to store print job information. Place your print parameter values in parameter 1 to parameter 7.

### Use in DQS Printing

DQS software uses the information in parameter 8 to:

- Validate a user's request
- Delete a print job
- Modify a print job

The information stored in parameter 8 includes:

- Client job number
- Job owner; on OpenVMS systems this usually consists of:
  - Client node name
  - User's name on the client
- Time the job was queued on the client

### Error Messages

DQS software issues an error (%DQS-E-P8RESERVED) if you attempt to specify parameter 8 with either a PRINT or a QSET /ENTRY command.

### For More Information

For more information about:

- The %DQS-E-P8RESERVED error message, see Appendix B
- The format of the information stored in parameter 8, see Appendix D
- The use of the /PARAMETERS qualifier, see Chapter 2

## Configuring DQS Software

### Client and Server Installation

When you run the DQS installation procedure, you are asked whether you want to install DQS server software. You are given this option because not all DQS nodes in your network require server software (for example, those nodes without connections to output devices). However, most or all nodes in your network require client software so that their users can have access to remote printing resources. Therefore, client software is always installed when you install DQS software.

### Client and Server Configuration

A client must be configured to work with its corresponding servers (and vice versa) in order for DQS software to function. You can configure a server to serve as many clients as you want. Similarly, you can configure a client to use the services of any number of servers. There are no restrictions on the number of DQS servers or clients that you can have in your network.

### Restrictions to DQS Configurations

The DQS software is suitable for all OpenVMS nodes in DECnet networks. DQS software can be used in small networks consisting of two nodes and in large networks consisting of many nodes. DQS software operates in either local area networks or wide area networks.

DQS software is limited only by the ability of DQS servers to accept and maintain a large number of simultaneous network connections (that is, logical links) to DQS clients. The maximum number of logical links that a server can support is a parameter that you can define using the Network Control Program (NCP) on DECnet Phase IV systems and the Network Control Language (NCL) on DECnet/OSI Phase V systems. The value of this parameter depends upon many factors, such as the server's processor speed and its typical work load.

Each logical link from a client creates one process on a server. A server must be configured to allow for enough processes. (See the section entitled Processes Created by DQS Software, which describes the processes created by the DQS software on both clients and servers.)

In addition to handling its other network applications, a server should be able to accept simultaneous connections from all clients that it serves. However, all clients trying to access the server at the same time is unlikely. Any client that attempts to print on the server after the maximum number of logical links is reached, automatically attempts to reconnect to the server after a short wait. The QSHOW, QSET/ENTRY, and QDELETE/ENTRY command functions do not attempt this automatic reconnection.

## Using Daisy-Chained Queues

**Purpose**

You can configure the DQS software in your network so that a client queue corresponds to a remote queue, which in turn, corresponds to another remote queue on another server. This configuration is called daisy-chained queues. You can daisy-chain queues across as many servers as you want.

**Benefits**

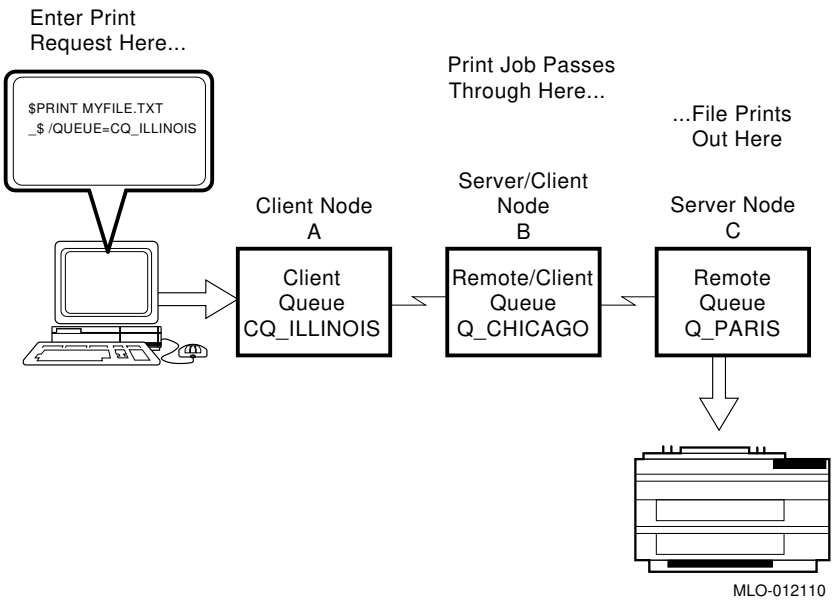
Daisy-chained queues are useful when a printer on a server becomes temporarily unavailable. Users who want to send jobs to the disabled server can still do so if the server's system manager sets up the remote queue to redirect their print output to another server's output device. No change is made to the user's client system, and the users may not even know that the printer was unavailable.

Daisy-chained queues are also useful when a network link from a client to its server is only occasionally available. With daisy-chained queues you can effect a "store and forward" function, where a job is transferred to a reachable, intermediate server that stores the job until the desired server is reachable.

**Example**

Figure 7–5 depicts a daisy-chained queue configuration.

**Figure 7–5 Daisy-Chained Queues**




## Process

The daisy-chain process shown in Figure 7–5 works as follows:

| Stage | Description |
|---|---|
| 1 | A user enters a print job in the client queue CQ_ILLINOIS on client node A. |
| 2 | DQS software transfers the job to node B and enters it in the remote queue Q_CHICAGO. |
| 3 | DQS software then transfers the job from node B to node C, and enters it in the remote queue Q_PARIS on node C. |
| 4 | The job is printed on node C's printer. |

## Avoiding Loops

Avoid creating a loop when configuring daisy-chained queues. If you configure a daisy-chained queue to direct a job back to the system where the job was originally queued, the job terminates on that system with the error, %DQS-E-LOOP.

## For More Information

Refer to Part III for more information on the installation of DQS software.

Chapters 8 and 9, respectively, describe the procedures for configuring servers and clients after installation of the software.

Chapter 8 describes the procedures for setting up daisy-chained queues.

## DQS Error Conditions

### Error Detection

When DQS software detects an error condition, the error is sent to the OpenVMS operator communication process (OPCOM). If an error is detected in a server process, it is also written to a NETSERVER.LOG log file on the server.

### Types of Errors

There are three basic types of errors that can occur in DQS printing.

- A print job fails because a network link to the server is not available or because there is insufficient disk space on the server.

  The DQS symbiont handles this type of error by automatically trying to reconnect to the server node after a five-minute wait (this value is not changeable).

- A print job cannot transfer to a server.

  The DQS symbiont handles this type of error by keeping the print job in the client queue with the error status "Retained on error." A QSHOW command specifying the client queue in question displays the specific reason for the error. In this case, you must requeue, release, or delete the print job.

  This type of error is depicted in Example 7–1.

- A client queue is stopped by the DQS symbiont.

  This type of error occurs when a user attempts to print to an invalid remote queue or to a server node whose name is unknown to the client node. A QSHOW command specifying the client queue in question displays the specific reason for the error.

  This type of error is depicted in Example 7–2.

## Examples

Example 7–1 shows an error detected by the client symbiont, DQS$SMB. The error does not affect the operation of the symbiont but causes the job responsible for the error to terminate with error status. The affected job remains in the client queue with the error status "Retained on error." In this case, you must either requeue or delete the print job. In Example 7–1, the client node has a form defined that is not defined on the server. This prevents the print job from transferring. A QSHOW command specifying the client queue where the job was queued shows the retained job and the specific reason for the error.

### Example 7–1 A DQS Error That Prevents a Print Job From Transferring

```
$ QSHOW QUEUE1 Return

** Remote queue - [ QUEUE1 => SYS$PRINT, on NODEB:: ]
Terminal queue SYS$PRINT, on NODEB::

==============================to be transferred==============================
Server queue QUEUE1, on NODEA::, mounted form DEFAULT

  Jobname         Username     Entry  Blocks  Status
  -------         --------     -----  ------  ------
  JOB             PAUL           925       1  Retained on error
    %JBC-E-NOSUCHFORM, no such form
```

Example 7–2 shows an error detected by the client symbiont, DQS$SMB, that caused the client queue to stop. The DQS$STATUS_*queue-name* logical name is defined to be the error status. This action makes it easy to see the reason for the error by issuing a QSHOW command for the client queue. In Example 7–2, the server node name is not known on the client. Therefore, the symbiont cannot process any jobs for that queue and stops the queue.

**Note**

Remember to deassign the DQS$STATUS_*queue-name* logical name when you restart the queue.

```
$ DEASSIGN/SYSTEM/USER DQS$STATUS_queue-name Return
```

### Example 7–2 A DQS Error That Stops a Client Queue

```
$ QSHOW QUEUE1 Return

***** Remote queue NODEB::QUEUE1 not accessible *****
    %SYSTEM-F-NOSUCHNODE, remote node is unknown

==============================to be transferred==============================
Server queue QUEUE1, stopped, on NODEA::, mounted form DEFAULT
    remote system error: remote node is unknown

  Jobname         Username     Entry  Blocks  Status
  -------         --------     -----  ------  ------
  LOGIN           YOUNG          571       2  Pending
```

**For More Information**

Refer to Chapter 10 for guidelines on how to troubleshoot error conditions that involve DQS software.

## DQS Security Issues and Access Rights

**Restricting Access**

By default, all client nodes have access to a server.

You can restrict access to a server node only by client system, not by individual client users.

**For More Information**

Chapter 8 describes the procedures for granting or denying client access to a server.

## Processes Created by DQS Software

### On a Client

On a client node, every client queue that you create (that is not a generic or logical queue) results in the creation of one process. Also, while a user is being notified of print job completion, a process is temporarily created.

### On a Server

For server nodes, every incoming connection from a client node results in the creation of one process. Also, the notification process on a server uses one permanent process.

## Devices Supported by DQS Software

### Supported Devices

DQS software does not run devices; rather, it places print jobs in queues for processing. DQS software supports devices by placing jobs in queues, if the device

- Is driven by an OpenVMS print symbiont
- Does not use parameter 8 in job requests

DQS software does not support symbionts that require job information outside of that stored or maintained by the OpenVMS queue system.

## Batch Jobs and DQS Printing

DQS software supports the queuing of print jobs for processing on remote nodes in your network. It does not support the queuing of batch jobs.

Do not configure batch queues to be DQS printer queues. DQS software does not prevent you from configuring batch queues, but the security of the system decreases if you do so.

# 8

# Configuring and Managing a DQS Server

## About This Chapter

This chapter presents an overview of the management tasks required by a DQS server and describes how to do the following:

- Modify definitions of special logical names associated with the DQS server
- Set up DQS remote queues
- Control DQS client access to servers and their remote queues
- Determine how DQS clients are accessing a server
- Control the note on the banner page of a print job
- Enable DQS status messages for queues
- Enable the DQS print symbiont to control remote queues
- Specify the scanning interval for the DQS server notification process
- Specify a directory for PrintServer log files
- Specify the priority of the server process
- Specify the maximum priority of DQS print jobs
- Specify the duration of links to client nodes
- Move the server account directory to a new device
- Configure server software for a VMScluster environment
- Configure daisy-chain queues

The last section contains a summary of the server management tasks.

## Overview of Server Management

### Main Tasks

Configuring and managing a DQS server primarily involves:

- Defining server system logical names in the SYS$MANAGER:DQS$SYSTARTUP.COM file

  All site-specific definitions that affect the behavior of the software belong in this file. If this file does not exist on your system, copy the DQS$SYSTARTUP.TEMPLATE to DQS$SYSTARTUP.COM and edit it to include the desired server logical name definitions.

- Setting up access to remote queues in the SYS$MANAGER:DQS$SERVER_CONFIG.TXT file

The SYS$STARTUP:DQS$STARTUP.COM procedure

- Invokes the SYS$MANAGER:DQS$SYSTARTUP.COM procedure to make changes permanent
- Processes the SYS$MANAGER:DQS$SERVER_CONFIG.TXT file to set up DQS client node access to remote queues on DQS server systems

Do not edit SYS$STARTUP:DQS$STARTUP.COM.

### Using SYSMAN

When directed to restart DQS software in a VMScluster, you can use one of the following methods:

- Invoke SYS$STARTUP:DQS$STARTUP.COM on each node in the cluster.
- Execute the SYSMAN utility.

## Defining Server System Logical Names

### Procedure

To make permanent changes to your server configuration in the DQS$SYSTARTUP.COM file, do the following:

| Step | Action |
|---|---|
| 1 | Edit the SYS$MANAGER:DQS$SYSTARTUP.COM file. |
| 2 | Add or change the definitions of the desired server logical names. Make sure that you define the logical name systemwide and in executive mode:<br>`$ DEFINE/SYSTEM/EXECUTIVE_MODE`<br>`_$ logical-name values` Return<br>For example, to set the server process to run at a priority of 5, enter:<br>`$ DEFINE/SYSTEM/EXECUTIVE_MODE DQS$PRIORITY 5` Return |
| 3 | Exit from the file SYS$MANAGER:DQS$SYSTARTUP.COM. |
| 4 | If the change is required immediately, do one of the following:<br>• Also enter the command at the DCL prompt.<br>• Execute the DQS startup command procedure.<br>`$ @SYS$STARTUP:DQS$STARTUP` Return<br>Otherwise, the change takes effect the next time the SYS$STARTUP:DQS$STARTUP.COM is executed or the node is rebooted. |

Adding the logical name definitions to the SYS$MANAGER:DQS$SYSTARTUP.COM file ensures that the logical name is defined each time your server reboots or each time the SYS$STARTUP:DQS$STARTUP.COM procedure is executed on a cluster member.

### Precaution

Do not execute the DQS$STARTUP.COM procedure unless it is necessary. Executing DQS$STARTUP.COM terminates any DQS network operations in progress. This may cause files transferred from a client to be left on the server with no association to the client job. When this happens, the server directory clean-up process, DQS$CLEANSRV.COM, detects and deletes the stray files left on the server disk. (DQS$STARTUP.COM submits the DQS$CLEANSRV.COM procedure at start-up time.)

### Changes at DCL Prompt

You can also define some DQS server logical names by entering the definition at the DCL prompt. These changes to server configuration:

- Take effect the next time the client establishes a connection to the server.
- Remain in effect until the server is rebooted or the DQS$STARTUP.COM is executed again.

## Modifying Definitions of Certain Logical Names

### Logical Names

For changes to the following logical name definitions to take effect, you must make your changes in the site-specific startup file DQS$SYSTARTUP.COM and restart all server processes:

DQS$LOG_ACCESS
DQS$ACCOUNTING_BY_SYSTEM
DQS$NOTIFY_CYCLE_TIME
DQS$PRIORITY
DQS$MAX_PRIORITY
DQS$IDLE_TIME

### Restarting Server Processes

To restart the server processes, execute the DQS$STARTUP.COM command procedure.

```
$ @SYS$STARTUP:DQS$STARTUP Return
```

Executing DQS$STARTUP.COM terminates any DQS network operations in progress.

## Before You Set Up Remote Queues

### Ensure Queues Exist

A queue must exist before you can set up the queue as a DQS remote queue. You can use existing queues or create (or initialize) new queues. (Refer to the *OpenVMS System Manager's Manual* for information on how to initialize a queue.)

### For New Installations

If you are installing this product for the first time, the DQS installation provides the following server configuration file to which you need to add your queue configurations:

```
SYS$MANAGER:DQS$SERVER_CONFIG.TXT
```

### For Product Upgrades

If you are upgrading from a version previous to Version 1.3, the installation procedure:

- Creates the SYS$MANAGER:DQS$SERVER_CONFIG.TXT file
- Attempts to translate your current queue definitions to the format of this file
- Provides you with messages in the SYS$UPDATE:DQS$CONVERT_SECURITY.LOG file
- Saves your queue definitions from the previous version by renaming the old startup file to DQS$STARTUP.COM_OLD

Compare the queue definitions in the DQS$SERVER_CONFIG.TXT file to those in DQS$STARTUP.COM_OLD. You may need to edit DQS$SERVER_CONFIG.TXT to ensure that your queue definitions are as you want them.

## Understanding How to Set Up Remote Queues

### Keywords

You enable network access to queues and control authorization to the server node and remote queues by using the following keywords in the DQS$SERVER_CONFIG.TXT file:

**QUEUE:** Declares a queue as a DQS network accessible queue. You must specify this keyword to allow network printing.

You also use the QUEUE keyword to specify which authorized client nodes have access to the queue.

**DENY_NODE:** Denies a client node authorization to the DQS servers on the server node. This keyword takes precedence over all other keywords.

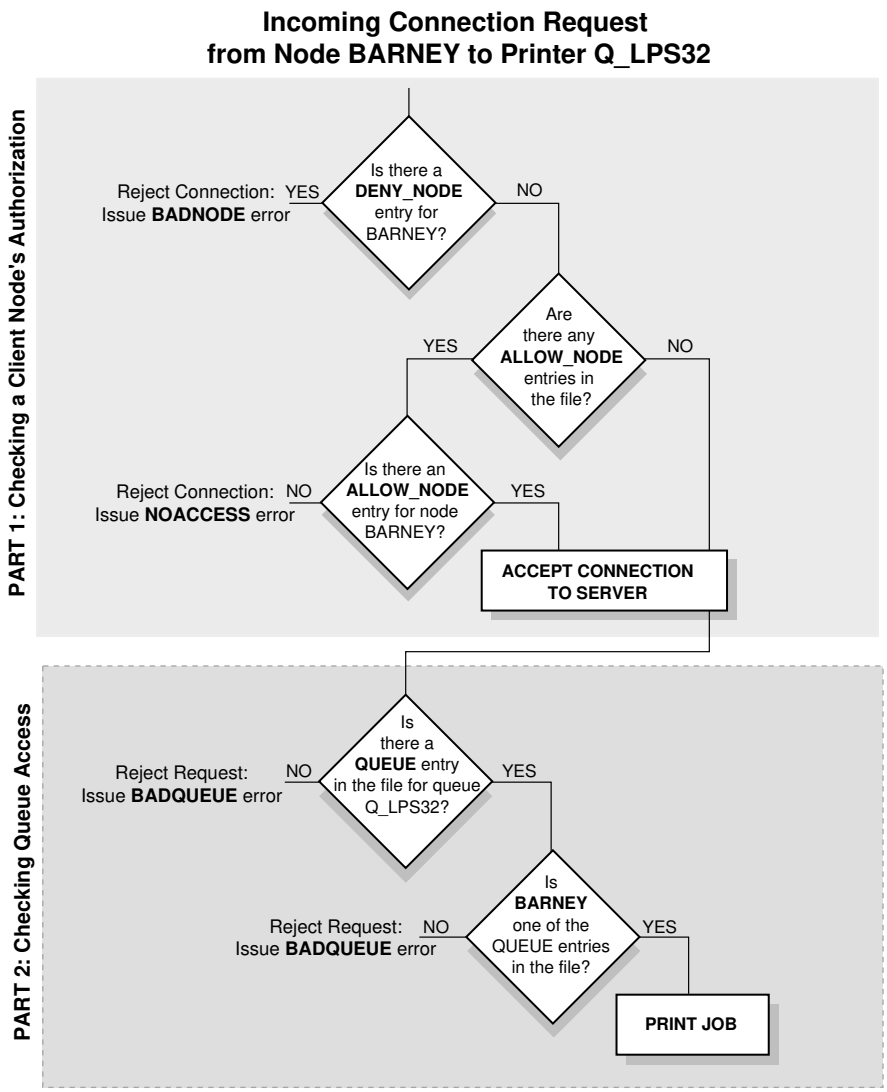
**ALLOW_NODE:** Grants only the client nodes you specify authorization to the DQS servers on the server node. All other nodes are denied authorization.

The QUEUE keyword differs from the other keywords. QUEUE controls access to **queues**, whereas ALLOW_NODE and DENY_NODE authorize access to the **DQS servers** on the server node.

### Client Authorization Process

Figure 8–1 shows how the configuration file checks that the client node is authorized to access the server node and remote queue.

**Figure 8–1 Checking a Client Node's Authorization**



## Understanding the Configuration File

### Features of the File

When you edit the DQS$SERVER_CONFIG.TXT file, notice the following:

- The file is a simple text file.
- Entries are not case-sensitive.
- Entries are specific keywords and their arguments.
- Valid keywords and their arguments are as follows:

```
Keyword      Argument 1              Argument 2
-------      ------------------      ------------------
QUEUE        queue_name              [client_node_name]
ALLOW_NODE   client_node_name
DENY_NODE    [client_node_name]
```

  You can the use wildcard character ( * ) as an argument shown in brackets ( [ ] ).

- Keywords are the first non-whitespace character on a line.
- Keywords and arguments are separated by at least one space.
- Arguments are either simple names or complex names.
  - Simple names contain no spaces and do not span multiple lines. Simple names can contain any printable character, except the space character, the exclamation point ( ! ), and the double quotation mark ( " ).
  - Complex names have double quotation marks ( " ) as delimiters at the beginning and the end of the name. Complex names can contain any printable character (except the double quotation mark ( " )) and can span multiple lines in the file.
- The exclamation point ( ! ) marks a comment. The DQS software treats lines beginning with an exclamation point as comments and ignores them. Likewise, an exclamation point on a line that is not part of a complex name causes the rest of the line to be treated as a comment and ignored.

- The asterisk character ( * ) is the wildcard character. An argument that consists of only the wildcard character implies that all client nodes are affected by the operation. For example:

```
DENY_NODE *        ! means deny access to ALL nodes
QUEUE Q_PARIS *    ! means allow ALL nodes access to Q_PARIS
```

## Setting Up Remote Queues

### Overview

Setting up remote queues involves editing DQS$SERVER_CONFIG.TXT and:

1. Enabling network access to the queues with the QUEUE keyword. (This step is required.)
2. Optionally denying specific client nodes the authorization to access the DQS servers on the server node.
3. Optionally granting only specified client nodes the authorization to access the DQS servers on the server node.
4. Making the changes take effect.

The following sections explain how to do these tasks.

## Enabling Network Access

### Procedure

Enable network access to the queues with the QUEUE keyword. (This step is required.)

Specify one QUEUE entry for each remote queue on your system. This allows the remote queue to be DQS network accessible to authorized client nodes.

By default, all client nodes in your network are authorized to access the DQS servers on your node.

### Unlimited Queue Access

You can use the asterisk (*) as a wildcard to allow all authorized client nodes access to the print queues.

### Example

For example, the following entry allows all authorized client nodes access to the print queue Q_CHICAGO:

```
QUEUE Q_CHICAGO *
```

### Limited Queue Access

To allow only certain authorized client nodes access to your queue, enter a QUEUE entry for each node that you want to allow access to that particular queue.

### Example

For example, the following entries allows authorized client nodes APPLE, BANANA, and CHERRY access to Q_PARIS:

```
QUEUE Q_PARIS APPLE          ! Allow authorized client nodes APPLE,
QUEUE Q_PARIS BANANA         ! BANANA, and CHERRY access to
QUEUE Q_PARIS CHERRY         ! my queue Q_PARIS
```

### Where to Go from Here

Enabling network queue access is the only required step. If you want to grant or deny certain client nodes authorization to DQS servers on your server node, continue to the next sections. Otherwise, see the section titled ‘‘Making Your Changes Take Effect.’’

## Denying Client Node Authorization to the DQS Server

### Purpose

Denying client nodes authorization to the DQS servers is primarily intended to shut out clients that inappropriately and repeatedly establish connections to a server. Denied authorization takes precedence over any other form of client access control.

### Procedure

To deny a client node authorization to a DQS server on the server node, specify the name of the client node as the first argument to the DENY_NODE keyword.

### Example

For example, the following entry denies client node APPLE from accessing DQS servers on the server node:

```
DENY_NODE APPLE
```

### Denying All Clients

To deny all client nodes authorization to the DQS servers, add an entry with the asterisk ( * ) as the first argument to the keyword:

```
DENY_NODE *         ! Deny authorization to all nodes
```

This restricts all client nodes access to your DQS servers.

## Granting Client Node Authorization to the DQS Server

**Purpose**

You can grant certain client nodes authorization to DQS servers on a server node. This method is useful when you want to allow a subset of client nodes access to the DQS servers on the server node.

**Reminder**

If you use ALLOW_NODE, keep in mind:

- You turn off the default that authorizes all client nodes access to the DQS server.
- You must specify every client node that you want to authorize access to the DQS server.
- You cannot use an asterisk (*) with the ALLOW_NODE keyword.

**Procedure**

Add the names of the client nodes to which you want to grant authorization to the DQS server as the argument to the ALLOW_NODE keyword.

**Example**

For example, the following entries allow client nodes APPLE and CHERRY access to the DQS server:

```
ALLOW_NODE APPLE
ALLOW_NODE CHERRY
```

**Using ALLOW_NODE with QUEUE**

You can use the ALLOW_NODE and QUEUE keywords together as a filter in restricting access to the DQS server and queues. First you specify which clients nodes have access to the queue, then you specify which client nodes can access the DQS server.

**Procedure**

Use the following procedure to set up restricted access:

1. Use the QUEUE keyword to enable network access to the queues.

For example, the following entries allow all authorized client nodes access to queues Q_OHIO, Q_KANSAS, and Q_SHANNON:

```
QUEUE Q_OHIO *        ! Allow all authorized client nodes
QUEUE Q_KANSAS *      ! access to queues Q_OHIO, Q_KANSAS,
QUEUE Q_SHANNON *     ! and Q_SHANNON
```

2. Then, use ALLOW_NODE to authorize access to the DQS servers. For example, the following entries authorize client nodes APPLE and GRAPE to access the DQS server:

```
ALLOW_NODE APPLE            ! Grant authorization only to
ALLOW_NODE GRAPE            ! client nodes APPLE and GRAPE
```

**Result**

The combination of the ALLOW_NODE and QUEUE entries allows only a subset of client nodes (APPLE and GRAPE) access to queues Q_OHIO, Q_KANSAS, and Q_SHANNON.

**Note**

When using controlled access between two DQS server nodes that also act as client nodes, make sure each DQS server node is authorized access to the other node. Otherwise, notification messages may not be delivered.

## Making Your Changes Take Effect

### Procedure

When you are finished adding keywords to the configuration file, exit from the file and execute DQS$SERVER_UPDATE_CONFIG.COM to establish your changes:

```
$ @SYS$MANAGER:DQS$UPDATE_SERVER_CONFIG Return
```

Changes to the DQS$SERVER_CONFIG.TXT data file take effect the next time the client establishes a new connection to the server.

## Determining How Clients are Accessing a Server

### Methods

You can determine how and when clients access a particular server by doing any of the following:

- You can check the NETSERVER.LOG log files, which are created in the server account's directory. These files are periodically purged.
- If you enabled a server to log client access with the DQS$LOG_ACCESS logical name, you can check the file created in the [DQS$SERVER.NODES] subdirectory of the server account to see how many times and the last time each client established a connection to the server.
- If OpenVMS accounting is enabled, you can check the process and image accounting records as described in the *OpenVMS System Manager's Utilities Reference Manual*.

  With OpenVMS accounting enabled and the DQS$ACCOUNTING_BY_SYSTEM logical name defined, the server maintains records of print jobs received.

### Checking for Client Access

You can check the NETSERVER.LOG log files in the server account directory for records of client node access. However, these files are periodically purged. Example 8–1 shows how to view this file.

**Example 8–1 Viewing NETSERVER.LOG Information**

```
$ TYPE SYS$COMMON:[DQS$SERVER]NETSERVER.LOG [Return]
```

If you moved the directory, the log files might not be in SYS$COMMON.

## Logging Client Access to a Server

DQS software counts each occurrence of a client connecting to a server if you define the DQS$LOG_ACCESS logical name on the server. With DQS$LOG_ACCESS defined, the server process creates a file in the [DQS$SERVER.NODES] subdirectory of the server account. This file has the name of the client. If the [DQS$SERVER.NODES] subdirectory does not already exist, the server process creates it.

## Procedure

To define the DQS$LOG_ACCESS logical:

| Step | Action |
|---|---|
| 1 | Edit the SYS$MANAGER:DQS$SYSTARTUP.COM file. |
| 2 | Add the definition of the DQS$LOG_ACCESS logical name with an arbitrary value. Its value is not important; only its presence on the server is important. |
| 3 | Save the file, SYS$MANAGER:DQS$SYSTARTUP.COM and exit from the editor. |
| 4 | Restart all server processes by executing the SYS$STARTUP:DQS$STARTUP.COM procedure. |

## Examples

Example 8–2 defines the DQS$LOG_ACCESS logical name to be the arbitrary value ENABLED.

### Example 8–2 Logging Client Access to a Server

```
$ DEFINE/SYSTEM/EXECUTIVE_MODE DQS$LOG_ACCESS ENABLED Return
```

Example 8–3 shows a sample directory listing of the [DQS$SERVER.NODES] subdirectory that contains the names of the clients that accessed the server.

**Example 8–3 Looking at DQS$SERVER.NODES**

```
$ Directory SYS$COMMON:[DQS$SERVER.NODES] Return

Directory SYS$COMMON:[DQS$SERVER.NODES]

SMEDLY.;78     BUGSBY.;1407        JAZZY.;3     RAZBRY.;2
STENCL.;397         STIKNY.;24          TOKYO.;25

Total of 7 files.
```

Because there is a one-version limit for the [DQS$SERVER.NODES] subdirectory, only one file per client is created. The version number of the file corresponds to the number of logical-link connections received from the client.

Node names displayed in this file are trimmed to 8 characters. Some characters may be converted to underscores.

## Enabling Client Accounting Information

With the DQS$ACCOUNTING_BY_SYSTEM logical name defined on the server, the DQS software includes the client's node name as the Account field in the server's accounting record.

User node names written to this record are trimmed to 8 characters. Some characters may be converted to underscores.

## Example with Logical Defined

For example:

```
$ RUN AUTHORIZE Return
UAF> SHOW MACOMBER Return

Username: MACOMBER                      Owner:  Ted Macomber
Account:  DEVELOPMENT                   UIC:    [100,1] ([MACOMBER])
 . . .
UAF> EXIT Return

$ ACCOUNTING/TYPE=PRINT/FULL Return
```

```
Username:          MACOMBER          UIC:               [DQS$SERVER]
Account:           SMEDLY            Finish time:       14-FEB-1994 11:49:29.88
Process ID:        00000D9A          Start time:        14-FEB-1994 11:49:24.59
Owner ID:                            Elapsed time:              0 00:00:05.29
Terminal name:                       Processor time:            0 00:00:00.00
Remote node addr:                    Priority:          100
Remote node name:                    Privilege <31-00>: 00000000
Remote ID:                           Privilege <63-32>: 00000000
Remote full name:
Queue entry:       721               Final status code: 00040001
Queue name:        SERVER
Job name:          ACCOUNTING
Final status text: %JBC-S-NORMAL, normal successful completion

GETs from source:        180
QIOs to printer:           5
Pages printed:             5
```

## Example with Logical Undefined

If you do not define this logical, the DQS client software uses the Account field specified in the client's User Authorization File (UAF) on the client for the user who issued the PRINT command. For example:

```
$ ACCOUNTING/TYPE=PRINT/FULL  Return

PRINT Job Termination
---------------------
Username:          MACOMBER          UIC:               [DQS$SERVER]
Account:           DEVELOPMENT       Finish time:       14-FEB-1994 11:49:07.65
Process ID:        00000D9A          Start time:        14-FEB-1994 11:49:02.36
Owner ID:                            Elapsed time:              0 00:00:05.29
Terminal name:                       Processor time:            0 00:00:00.00
Remote node addr:                    Priority:          100
Remote node name:                    Privilege <31-00>: 00000000
Remote ID:                           Privilege <63-32>: 00000000
Remote full name:
Queue entry:       717               Final status code: 00040001
Queue name:        SERVER
Job name:          ACCOUNTING
Final status text: %JBC-S-NORMAL, normal successful completion

GETs from source:        180
QIOs to printer:           5
Pages printed:             5
```

## Procedure

To define the DQS$ACCOUNTING_BY_SYSTEM logical name:

| Step | Action |
|---|---|
| 1 | Edit the SYS$MANAGER:DQS$SYSTARTUP.COM file. |
| 2 | Add the definition of the DQS$ACCOUNTING_BY_ SYSTEM logical name with an arbitrary value.<br>$ **DEFINE/SYSTEM/EXECUTIVE_MODE -** Return<br>_$ **DQS$ACCOUNTING_BY_SYSTEM *ENABLE*** Return |
| 3 | Exit from the file, SYS$MANAGER:DQS$SYSTARTUP.COM. |
| 4 | Restart all server processes by executing the SYS$STARTUP:DQS$STARTUP.COM procedure. |

## Example

Example 8–4 defines the DQS$ACCOUNTING_BY_SYSTEM logical name to be the arbitrary value ENABLED.

### Example 8–4 Enabling Client Accounting Information

```
$ DEFINE/SYSTEM/EXECUTIVE_MODE DQS$ACCOUNTING_BY_SYSTEM ENABLED Return
```

## Controlling the Note on the Job Banner Page

### Banner Page Note

The DQS software makes use of a print attribute to place a note on the banner page of a print job when logical names are defined in the DQS$SYSTARTUP.COM file. On the server, this logical name definition provides the default text ''Print Job processed by DQS V*n.n*''. The note defined by this logical name prints on the banner page of a print job if:

- The /NOTE qualifier is not specified by the client node user on the PRINT command
- The logical name definition for the note is not specified for the client job

The note that prints on the banner page of a print job follows this order of precedence:

- Note specified by the user with PRINT/NOTE=*note*
- Note specified by a client logical name definition (DQS$CLIENT_DEFAULT_JOB_NOTE)
- Note specified by a server logical name definition (DQS$SERVER_DEFAULT_JOB_NOTE)

### Options

You have the following options:

- Use the default note provided by the DQS software for the banner page
- Turn off the DQS logical definitions for the banner page note
- Define your own note for the banner page

### To Turn Off the Default

To turn off the DQS default note on the server system, do the following:

| Step | Action |
|---|---|
| 1 | Edit the SYS$MANAGER:DQS$SYSTARTUP.COM file. |
| 2 | Search for the following command and remove the exclamation mark ( ! ) comment character:<br>`$! $ DEASSIGN/SYSTEM/EXEC DQS$SERVER_DEFAULT_JOB_NOTE` |
| 3 | Save SYS$MANAGER:DQS$SYSTARTUP.COM and exit from the editor. |

The change takes effect the next time the SYS$STARTUP:DQS$STARTUP.COM is executed.

### To Change Note Text

To change the text of the note that prints on the banner page, do the following:

| Step | Action |
|---|---|
| 1 | Edit the SYS$MANAGER:DQS$SYSTARTUP.COM file. |
| 2 | Search for the following command:<br>`$! $ DEFINE/SYSTEM/EXEC DQS$SERVER_DEFAULT_JOB_NOTE -`<br>`$!  "Place your text here"`<br>Substitute your note for the text ‘‘Place your text here’’ and remove the dollar sign and exclamation mark ( $! ) at the beginning of each line. |
| 3 | Save the file SYS$MANAGER:DQS$SYSTARTUP.COM and exit from the editor. |

The next time the SYS$STARTUP:DQS$STARTUP.COM is executed and the note is not specified by the user or the client, your customized note prints on the job banner page.

## Enabling Status Messages for Queues

### Purpose

You can define the DQS$STATUS_*queue-name* logical name so that a status message is displayed each time a user enters a QSHOW command for the specified queue (or any DQS queue that points to that queue). This function is useful for displaying printer status messages, such as "Q_PRINTER is down until 1PM for repair."

### Procedure

To define the DQS$STATUS_*queue-name* logical name:

| Step | Action |
|---|---|
| 1 | Define the DQS$STATUS_*queue-name* logical name by entering the following at the system prompt:<br>`$ DEFINE/SYSTEM/EXECUTIVE_MODE -`<br>`_$ DQS$STATUS_queue-name "Message"` Return<br>When defining the DQS$STATUS_*queue-name* logical make sure to enclose the message in quotation marks ("").<br>You can define this logical name for both remote and client queues. |

When you change the value of the DQS$STATUS_*queue-name* logical name, the next display of the queue shows the new message.

## Example

Example 8–5 specifies that the status message "Down for repairs" displays each time a user enters a QSHOW command for the remote queue Q_PARIS.

### Example 8–5 Defining a Status Message for a Queue

```
$ DEFINE/SYSTEM/EXECUTIVE_MODE DQS$STATUS_Q_PARIS "Down for repairs" Return
```

When you want to remove the message for the queue, enter one of the following commands:

```
$ DEASSIGN/SYSTEM/EXECUTIVE_MODE DQS$STATUS_Q_PARIS Return
$ DEASSIGN/SYSTEM/USER DQS$STATUS_Q_PARIS Return
```

## Enabling the DQS Print Symbiont To Control Remote Queues

### Why Use the DQS Symbiont?

DQS software supplies its own print symbiont, DQS$PRTSMB, which you can use to replace the standard OpenVMS print symbiont on a server unless you use the DECprint Supervisor for OpenVMS software. The DQS print symbiont writes client-specific job information on the burst, flag, and trailer pages of DQS print output. If you do not like the way the standard OpenVMS symbiont handles banner pages, enable the DQS print symbiont DQS$PRTSMB.

### Procedure to Enable

To enable the DQS print symbiont to control a remote queue, do the following:

| Step | Action |
|---|---|
| 1 | Stop the queue by entering STOP/QUEUE/RESET *remote-queue* at the DCL prompt. |
| 2 | Associate the DQS$PRTSMB process with the queue by entering INIT/QUEUE/PROC=DQS$PRTSMB *remote-queue* at the DCL prompt. |
| 3 | Restart the queue by entering START/QUEUE *remote-queue* at the DCL prompt |
| 4 | Digital recommends that you also enter the SET QUEUE/RETAIN=ERROR *queue-name* command at the DCL prompt to hold jobs in the queue that complete unsuccessfully. With the /RETAIN=ERROR qualifier set, QSHOW commands display the error. |

## Example

Example 8–6 shows how to enable the DQS print symbiont DQS$PRTSMB to control the remote queue Q_PARIS.

**Example 8–6 Enabling the DQS Print Symbiont**

```
$ STOP/QUEUE/RESET Q_PARIS Return
$ INIT/QUEUE/PROC=DQS$PRTSMB Q_PARIS Return
$ START/QUEUE Q_PARIS Return
$ SET QUEUE/RETAIN=ERROR Q_PARIS Return
```

## For More Information

Refer to Chapter 7 for more information on the DQS$PRTSMB symbiont.

Appendix D describes how the DQS software works with other print symbionts.

## Specifying the Scanning Interval for Server Notification

### About the Notifier

The DQS software supplies a process on server nodes, DQS$NOTIFIER, that can send notification messages to users after their print jobs complete. DQS$NOTIFIER runs as a detached process on a server and is started by the SYS$STARTUP:DQS$STARTUP.COM procedure at system start-up time. This process periodically scans remote queues to determine which print jobs are complete.

### Procedure

You can specify the interval at which the server notification process scans the remote queue by defining the DQS$NOTIFY_CYCLE_TIME logical name to be a time in OpenVMS delta time format. By default, the notify cycle time is five minutes.

To specify the DQS$NOTIFY_CYCLE_TIME logical name:

| Step | Action |
|---|---|
| 1 | Edit the SYS$MANAGER:DQS$SYSTARTUP.COM file. |
| 2 | Add the definition of the DQS$NOTIFY_CYCLE_TIME logical name with the desired time in OpenVMS delta time format. |
| 3 | Exit from the file, SYS$MANAGER:DQS$SYSTARTUP.COM. |
| 4 | Stop the DQS$NOTIFIER process. |
| 5 | Restart the DQS$NOTIFIER process by executing the SYS$STARTUP:DQS$STARTUP.COM procedure. |

If you change the default time, consider the following:

- If you define the cycle time to be less than five minutes, the server's job controller and notifier process use more of the server's CPU cycles.
- If the cycle time you define is much greater than the default, users may not be notified of their job completion in a timely manner.

## Example

Example 8–7 defines the interval at which DQS$NOTIFIER scans the remote queues to be every 10 minutes. The DQS$NOTIFIER process is stopped and restarted to effect this change.

**Example 8–7 Specifying the Scanning Interval for DQS$NOTIFIER**

Add to DQS$SYSTARTUP.COM:

```
$ DEFINE/SYSTEM/EXECUTIVE_MODE DQS$NOTIFY_CYCLE_TIME "0 00:10:00.0" Return
```

Then, enter at the DCL prompt:

```
$ @SYS$STARTUP:DQS$STARTUP.COM Return
```

## Specifying a Directory for PrintServer Log Files

### Overview

If your PrintServer software supports the return of log files from DQS print jobs, you can specify a directory into which the log files can be placed.

You specify a directory for the PrintServer log files by defining the DQS$LOG_AREA logical name to be the name of the directory.

### Procedure

To specify a directory for PrintServer software log files:

| Step | Action |
|---|---|
| 1 | Define the DQS$LOG_AREA logical name to be the name of the directory by entering the following at the DCL prompt:<br>$ **DEFINE/SYSTEM/EXECUTIVE_MODE**<br>_$ **DQS$LOG_AREA *device:[directory]*** Return |

If you want this logical name to be permanent, define the logical name in the SYS$MANAGER:DQS$SYSTARTUP.COM.

If this logical name is not defined, the PrintServer software does not write log files for DQS print jobs.

### Example

Example 8–8 specifies the directory [PRINTSERVER] on device $1$DUA0: on the server as the directory for PrintServer log files.

**Example 8–8 Specifying a Directory for PrintServer Log Files**

```
$ DEFINE/SYSTEM/EXECUTIVE_MODE DQS$LOG_AREA $1$DUA0:[PRINTSERVER] Return
```

### For More Information

For more information on logging options for the PrintServer Software, refer to the *PrintServer Software for OpenVMS Management Guide*.

## Specifying the Priority of the Server Process

### Procedure

You can specify the priority at which the server process runs by defining the DQS$PRIORITY logical name to be the desired priority. Be aware that redefining the DQS$PRIORITY logical name affects scheduling algorithms on the OpenVMS system and changes overall system performance.

| Step | Action |
|---|---|
| 1 | Edit the SYS$MANAGER:DQS$SYSTARTUP.COM file. |
| 2 | Add the definition of the DQS$PRIORITY logical name with the desired priority.<br>If you do not define this logical name, the server process runs at priority 4 by default. |
| 3 | Save the file SYS$MANAGER:DQS$SYSTARTUP.COM and exit from the editor. |
| 4 | Restart all server processes by executing the SYS$STARTUP:DQS$STARTUP.COM procedure. |

### Example

Example 8–9 shows the command to add to the SYS$MANAGER:DQS$SYSTARTUP.COM file to set the server process to run at priority 5.

**Example 8–9 Specifying the Priority of the Server Process**

```
$ DEFINE/SYSTEM/EXECUTIVE_MODE DQS$PRIORITY 5 Return
```

The next time a server process is started, the server process runs at the specified priority.

## Specifying the Maximum Priority of DQS Print Jobs

### Procedure

You can specify the maximum priority at which a print job can be queued or set by a user, by defining the DQS$MAX_PRIORITY logical name.

| Step | Action |
|---|---|
| 1 | Edit the SYS$MANAGER:DQS$SYSTARTUP.COM file. |
| 2 | Add the definition of the DQS$MAX_PRIORITY logical name with the desired maximum priority.<br>If you do not define this logical name, the value of the SYSGEN parameter DEFQUEPRI is used by default. |
| 3 | Save the file SYS$MANAGER:DQS$SYSTARTUP.COM and exit from the editor. |
| 4 | Restart all server processes by executing the SYS$STARTUP:DQS$STARTUP.COM procedure. |

### Example

Example 8–10 specifies that print jobs can be queued at no higher than priority 42.

**Example 8–10 Specifying the Maximum Priority for Queuing Print Jobs**

```
$ DEFINE/SYSTEM/EXECUTIVE_MODE DQS$MAX_PRIORITY 42 Return
```

## Specifying the Duration of Links to Client Nodes

### Overview

You can specify the duration that a server maintains logical links to inactive clients. A client is considered inactive if it is not sending print jobs to a server.

### Procedure

You specify the duration of logical links by defining the DQS$IDLE_TIME logical name to be the time duration in minutes in OpenVMS delta time format.

| Step | Action |
|---|---|
| 1 | Edit the file, SYS$MANAGER:DQS$SYSTARTUP.COM. |
| 2 | Add the definition of the DQS$IDLE_TIME logical name with the desired duration time in minutes in OpenVMS delta time format.<br>If you do not define this logical name, an idle time of 15 minutes is used by default. |
| 3 | Save the file SYS$MANAGER:DQS$SYSTARTUP.COM and exit from the editor. |
| 4 | Restart all server processes by executing the SYS$STARTUP:DQS$STARTUP.COM procedure. |

When you redefine the DQS$IDLE_TIME logical name, consider the following:

- If you define the idle time to be less than 15 minutes, processes may be created on your server more often than you would like.
- If you define the idle time to be greater than 15 minutes, too many processes may remain for too long on your server.

## Example

Example 8–11 specifies that the server maintains logical links to idle clients for at least 45 minutes.

**Example 8–11 Specifying the Duration of Links to Inactive Clients**

```
$ DEFINE/SYSTEM/EXECUTIVE_MODE DQS$IDLE_TIME "0 00:45:00.0" Return
```

## Idle Time Exceeded

The DQS software responds as follows:

| When . . . | Then . . . |
|---|---|
| Idle time specified by the DQS$IDLE_TIME logical name is exceeded | The server process disconnects the logical link to the client. |
| The client has a print job to transfer to the server | The client reestablishes the logical link to the server. |

## Move the Server Account Directory to Another Device

### Storage Space

The amount of storage space required for the server account directory varies depending on the following:

- Number of queues defined
- Expected number and size of the queued print jobs
- Rate at which printers can print jobs (so that files are deleted)

The DQS server installation sets the default device for the server account to SYS$COMMON:. You can use the DQS$SERVER_CHANGE_DEFAULT_DEVICE.COM procedure to change the default to another disk or back to SYS$COMMON:.

### Procedure

If you want to move the server account directory to a device other than the system disk, execute the DQS$SERVER_CHANGE_DEFAULT_DEVICE.COM procedure by entering the following command at the DCL prompt:

```
$ @SYS$MANAGER:DQS$SERVER_CHANGE_DEFAULT_DEVICE Return
```

DQS$SERVER_CHANGE_DEFAULT_DEVICE.COM asks you for the device for the account storage.

```
Enter device for the DQS$SERVER account:
```

Enter the name of the device to use for the server account. For example, if your server account is to be on device USERDISK, enter USERDISK:.

```
Enter device for the DQS$SERVER account:USERDISK: Return
```

After you enter the device name, you see a series of status messages, such as the following:

```
User record(s) updated
DQS$SERVER directory is now located on device USERDISK:

To start DQS, enter following:

        $ @SYS$STARTUP:DQS$STARTUP.COM

In VMScluster environment, you need to start DQS on each node
in a cluster.

$
```

## Advanced VMScluster DQS Configurations

### Overview

In an OpenVMScluster environment, you can configure:

- A set of queues to be shared by all nodes in the cluster
- A set of queue to be shared by some nodes and a different set of queues to be shared by other nodes.

When you configure different sets of queues, you need to:

1. Redefine the DQS$NOTIFIER_LOCK logical
2. Specify two separate DQS$SERVER_CONFIG.TXT files.

The following sections explain how to configure different sets of queues.

## Specifying Unique Notifier Lock Names in a VMScluster Environment

**Requirement**

When you configure different sets of queues for different groups of nodes, you must define the same notifier lock name on all cluster members that share a particular set of queues.

**Procedure**

To ensure that each set of queues is associated with a common and unique notifier lock, define the DQS$NOTIFY_LOCK logical name to be the same, unique lock name on all cluster nodes that share the same set of queues.

By default, the DQS installation names the notifier lock to be DQS$NOTIFIER.

**Example**

For example:

- A VMScluster system consists of nine nodes with four members that share one set of queues and five members that share another set of queues.
- Four members that share the same set of queues use the default notifier lock name DQS$NOTIFIER.
- Five cluster members that share the other set of queues require that you define a unique notifier lock name on each of them. The lock name DQS$NOTIFIER_2 can be defined on each of these five cluster members as shown in Example 8–12.

**Example 8–12 Specifying Unique Notifier Lock Names in a Nonhomogeneous VMScluster**

```
$ DEFINE/SYSTEM/EXECUTIVE_MODE DQS$NOTIFY_LOCK DQS$NOTIFIER_2 Return
```

## Redefining DQS$SERVER_CONFIG_DAT_FILE and DQS$SERVER_CONFIG_TXT_FILE

### Recommended Configuration

Digital recommends that you use the same DQS server configuration for all nodes within your cluster. The DQS installation sets up the DQS software on your system with this recommended configuration in mind. As a result, the default values for the locations of the DQS server configuration files are both specifically located in the SYS$COMMON areas of your clusters file system.

The DQS software uses VMS logical names to locate the server configuration files.

### Location of Configuration Data

The location of the server configuration text is controlled by the logical DQS$SERVER_CONFIG_TXT_FILE. The default value for this logical is SYS$COMMON:[SYSMGR]DQS$SERVER_CONFIG.TXT.

The location of the encoded file that the DQS$SERVER.EXE program uses to validate a client's request for access to a particular server queue is controlled by the logical DQS$SERVER_CONFIG_DAT_FILE. The default value for this logical is SYS$COMMON:[SYSEXE]DQS$SERVER_CONFIG.DAT.

### Specifying Multiple Server Configuration Files

DQS can support multiple server configuration files within a cluster. If your cluster cannot use the default configuration, you can use the DQS$SERVER_CONFIG_TXT_FILE and DQS$SERVER_CONFIG_DAT_FILE logical names to provide multiple, separate, DQS server configurations within the same cluster.

## Procedure

To modify the logicals, use the following procedure:

1. Edit the site-specific DQS startup file SYS$STARTUP:DQS$SYSTARTUP.COM.
2. Change SYS$COMMON: to SYS$SYSROOT: in following lines:

```
$ DEFINE/SYS/EXEC DQS$SERVER_CONFIG_TXT_FILE -
     SYS$COMMON:[SYSMGR]DQS$SERVER_CONFIG.TXT

$ DEFINE/SYS/EXEC DQS$SERVER_CONFIG_DAT_FILE -
     SYS$COMMON:[SYSMGR]DQS$SERVER_CONFIG.DAT
```

3. Restart DQS to effect the change by executing SYS$STARTUP:DQS$STARTUP.COM.

## Specifying Remote Queues on a VMScluster System

### Execute the Start-up Procedure

If you specify remote queues on any node in a VMScluster system, you must execute the SYS$STARTUP:DQS$STARTUP.COM procedure on each cluster member. If you do not, DQS connections to the cluster fail randomly because not all nodes in the cluster are configured as DQS servers.

The cluster should have an incoming cluster alias defined so that clients can direct their print jobs to that cluster alias.

## Configuring Daisy-Chained Queues

### Benefits

Daisy-chained queues are useful when a server's output device becomes unavailable. In this case, you can define the remote queue for the unavailable device also to be a client queue that directs print jobs to yet another remote queue on another DQS server node.

### Defining

To define a remote queue as a daisy-chained queue, define the remote queue to be a DQS client queue, using any of the methods described in Chapter 9:

- Use SYS$MANAGER:DQS$DEFINE.COM if you want the definition to be temporary.
- Add the definition to DQS$SYSTARTUP.COM if you want a permanent defintion.

### Beware of Loops

The DQS software does not prevent you from creating a loop in your daisy-chained queues. Be sure not to set up daisy-chained queues where print jobs ultimately are directed back to the queue from which they were forwarded.

### Fixing Looping Errors

Perform the following procedure to eliminate a loop:

1. Stop any queue in the loop, then stop the other queues in the loop.
2. Edit the configuration file (DQS$SYSTARTUP.COM) and change the client and/or server queue definitions to eliminate the loop.
3. Restart DQS and the queues on each node. DQS$NOTIFIER deletes any copies of jobs that were created in the loop.
4. Enter the QSHOW command to verify that the loop has been eliminated.
5. Manually delete any print jobs that were replicated in the loop.

## Summary of Server Tasks

Table 8–1 lists the management tasks and how the task is performed for the DQS server.

**Table 8–1 Summary of Server Node Management Tasks**

| For this task . . . | You do this . . . |
|---|---|
| Editing site-specific startup file to define logical names | Edit the SYS$MANAGER:DQS$SYSTARTUP.COM procedure so that it defines each DQS logical name definition you specify. |
| Setting up remote queues | Add entries to the SYS$MANAGER:DQS$SERVER_CONFIG.TXT file. For each remote queue on the server node, include a QUEUE keyword followed by the name of the queue as the first argument and the name of the client node that is permitted to send jobs to the remote queue as the second argument. |
| Granting clients controlled access to a server | Add entries to the SYS$MANAGER:DQS$SERVER_CONFIG.TXT file. For each client node allowed access to remote queues on the server system, include the ALLOW_ NODE keyword with the name of the client node as the first and only argument. Client nodes must also be allowed access through the QUEUE entries in the server configuration file. |
| Denying clients access to a server | Add entries to the SYS$MANAGER:DQS$SERVER_CONFIG.TXT file. For each client node denied access to remote queues on the server system, include the DENY_NODE keyword with the name of the client node not permitted to access the server as the first and only argument. |
| Logging client access to a server | Define the DQS$LOG_ACCESS logical name so that the server creates a file in the server account that has the name of a client and counts each time the client establishes a connection to the server. |

(continued on next page)

**Table 8–1 (Cont.) Summary of Server Node Management Tasks**

| For this task . . . | You do this . . . |
|---|---|
| Enabling client accounting information | Define the DQS$ACCOUNTING_BY_SYSTEM logical name so that a client's node name (in the format *node-name*) is used in the server's accounting record. |
| Enabling status messages for a queue | Define the DQS$STATUS_*queue-name* logical name to be a status message that is displayed for each queue that has this logical name defined when a QSHOW command is issued. |
| Enabling the DQS print symbiont for a remote queue | Issue OpenVMS commands to stop the remote queue, initialize the queue specifying the DQS$PRTSMB process, and restart the queue. |
| Specifying the interval at which the server notification process scans the remote queues | Define the DQS$NOTIFY_CYCLE_TIME logical name to be the interval at which the DQS server notification process (DQS$NOTIFIER) scans the remote queues to determine when a print job has completed |
| Specifying a directory for PrintServer log files | Define the DQS$LOG_AREA logical name to be the directory into which PrintServer log files are placed. |
| Specifying the priority of the server process | Define the DQS$PRIORITY logical name to be the priority at which you want the server process to run. |
| Specifying the maximum priority of DQS print jobs | Define the DQS$MAX_PRIORITY logical name to be the maximum priority at which a print job can be queued on the server. |
| Specifying the time for maintaining links to client nodes | Define the DQS$IDLE_TIME logical name to be the duration of time the server maintains a logical link to a client symbiont that is idle (that is, a symbiont that is not transferring print jobs). |

(continued on next page)

**Table 8–1 (Cont.) Summary of Server Node Management Tasks**

| For this task . . . | You do this . . . |
|---|---|
| Moving the DQS server default directory. | Run the DQS$SERVER_CHANGE_DEFAULT_DEVICE.COM procedure. |
| Configuring different sets of queues in a VMScluster system. | Define the same DQS$NOTIFY_LOCK logical name on all nodes in a nonhomogeneous VMScluster that share the same queue file. By default, the DQS installation names the notifier lock to be DQS$NOTIFIER.<br>Redefine the logicals DQS$SERVER_CONFIG_DAT_FILE and DQS$SERVER_CONFIG_TXT_FILE. |
| Configuring server software in a VMScluster | Execute the DQS$STARTUP.COM command file on all cluster members in order to define remote queues throughout the cluster. |
| Setting up daisy-chained remote queues | Define a remote queue to also be a client queue that directs jobs to yet another remote queue. |

# 9

# Configuring and Managing a DQS Client

## About This Chapter

This chapter presents an overview of the management tasks required by a DQS client and explains how to:

- Use the DQS startup procedure in client management
- How to manage processes created by client queues
- Create client queues with basic and advanced methods
- Temporarily change the definition of a client queue
- Delete client queues
- Configure client nodes in a VMScluster environment
- Synchronize client and server forms definitions
- Spool to a client queue
- Modify the default note on the banner page of a print job
- Regulate forms checking on client nodes

## Overview of Client Management

### Management Tasks

Configuring and managing a DQS client primarily involves defining DQS client queues to direct print jobs to associated remote queues on server systems. You can also modify and delete DQS client queue definitions.

DQS software provides the following methods of defining and managing client queues:

- A basic method using the DQS$IMPORT.COM procedure
- More complex methods:
  - Adding definitions as parameters to the DQS$DEFINE.COM procedure in the SYS$MANAGER:DQS$SYSTARTUP.COM file
  - Defining generic client queues
  - Defining logical client queues

### Executing the DQS Startup Procedure

To start or restart client queues, you must use the SYS$STARTUP:DQS$STARTUP.COM procedure. This procedure invokes the required procedures to define client queues on the system.

## Managing Client Processes

### Overview

On a client node, every client queue that you create (that is not a generic or a logical queue) results in the creation of one process.

### One Client; One Process

If you do not anticipate a problem with the number of processes on your client node, then you can use either of the following procedures to create client queues:

- DQS$IMPORT.COM (described in the section titled Creating Client Queues with DQS$IMPORT.COM)
- DQS$DEFINE.COM (described in the section titled Creating Client Queues with More Advanced Methods)

Both result in the creation of a process for each client queue.

### Reducing Processes

If you do not want to create as many processes as client queues, then you can create either of the following types of client queues:

- Generic client queues (described in the section titled Method 2: Defining Generic Client Queues
- Logical client queues (as described in the section titled Method 3: Defining Logical Client Queues

Generic and logical queues direct jobs to other client execution queues that actually process the job. Therefore, fewer processes are created on your client node.

## Creating Client Queues with DQS$IMPORT.COM

### A Basic Method

The most basic way to create DQS client queues is to invoke the DQS$IMPORT.COM command procedure.

The DQS$IMPORT.COM procedure provides a way to permanently define client queues. The queue definition is automatically added to the DQS$SYSTARTUP.COM command file. This ensures that the client queue starts with the proper definition each time the client system is rebooted.

### Before You Start

Before you can define a client queue with the DQS$IMPORT command procedure:

- You must have the following privileges: SYSPRV, OPER, SYSNAM, NETMBX, and TMPMBX.
- The associated server node must be reachable.
- The associated remote queue must be initialized and defined as a DQS remote queue.

### Procedure

Create a client queue with the DQS$IMPORT.COM procedure as follows:

| Step | Action |
|---|---|
| 1 | Invoke the DQS$IMPORT.COM command file for each client queue you want to create.<br>$ **@SYS$MANAGER:DQS$IMPORT *server-node*::*remote-queue*** Return<br>For each client queue, supply the associated remote queue and server node in the format *server-node*::*remote-queue* as a parameter to the procedure. |

DQS$IMPORT.COM then does the following:

- Creates the client queue, giving it the name of its associated remote queue.
- Adds the client queue definition to the DQS$SYSTARTUP.COM file so that the queue definition is persistent.

## Example

Example 9–1 show how to use DQS$IMPORT.COM to create a client queue Q_CHICAGO that uses the name of the corresponding remote queue Q_CHICAGO on the server node APPLES.

**Example 9–1 Creating a Client Queue with DQS$IMPORT.COM**

```
$ @SYS$MANAGER:DQS$IMPORT APPLES::Q_CHICAGO Return
```

DQS$IMPORT.COM adds this client queue definition to the DQS$SYSTARTUP.COM file.

## Note

DQS$IMPORT.COM adds an invocation of the DQS$DEFINE.COM procedure to the SYS$MANAGER:DQS$SYSTARTUP.COM file for each client queue and executes the DQS$DEFINE.COM procedure to actually create the queue. When you use DQS$IMPORT.COM to create a client queue, the client queue and the remote queue have the same name.

## Creating Client Queues with More Advanced Methods

### Advanced Methods

More advanced methods of creating client queues are as follows:

- Adding a definition for a client queue directly to the DQS$SYSTARTUP.COM file using the DQS$DEFINE.COM procedure and parameters
- Defining generic client queues
- Defining logical client queues

### Method 1: Using DQS$DEFINE

Creating a client queue with the DQS$DEFINE.COM procedure is useful when you want the name of a client queue to be different from its associated remote queue name.

### Procedure

Create client queues with the DQS$DEFINE.COM procedure as follows:

| Step | Action |
|---|---|
| 1 | Edit the SYS$MANAGER:DQS$SYSTARTUP.COM file and search for the statement, "DQS Client Queues Definitions". The statement appears in the file as follows:<br>`$!-----------------------------------------`<br>`$! DQS Client Queues Definitions`<br>`$!-----------------------------------------.` |

| Step | Action |
|---|---|
| 2 | On a new line following the "DQS Client Queue..." statement, insert a DCL command that invokes the SYS$MANAGER:DQS$DEFINE.COM procedure (see Example 9–2).<br>For each invocation of DQS$DEFINE.COM that you insert, supply the following parameters in the following order (the last parameter is optional):<br>a. Client queue name<br>b. Its corresponding server node name<br>c. Its corresponding remote queue name<br>d. A zero (0)<br>e. Qualifiers with which to start the client queue<br>Separate each parameter by at least one space. Spaces are not permitted in the parameters.<br>Example 9–2 shows how to edit this information properly into the SYS$MANAGER:DQS$SYSTARTUP.COM file. |

DQS$DEFINE.COM is the procedure that actually creates a DQS client queue. When you edit the SYS$MANAGER:DQS$SYSTARTUP.COM file, you should insert your invocation of DQS$DEFINE.COM exactly as shown in Example 9–2.

## Example

In Example 9–2, SYS$MANAGER:DQS$SYSTARTUP.COM is edited to contain an invocation of the DQS$DEFINE.COM procedure that associates the client queue CQ_ILLINOIS with the remote queue Q_CHICAGO on server node APPLE.

### Example 9–2 Creating Client Queues With DQS$DEFINE.COM

```
$ EDIT/EDT SYS$MANAGER:DQS$SYSTARTUP.COM Return
    1       $! IDENT=V1.3
*FIND 'DQS Client Queues Definitions' Return
```

(continued on next page)

**Example 9–2 (Cont.) Creating Client Queues With DQS$DEFINE.COM**

```
  113       $! DQS Client Queues Definitions
  $!
*INSERT Return
  $ @SYS$MANAGER:DQS$DEFINE CQ_ILLINOIS -    ! Queue being defined
                APPLE -                      ! Server node
                Q_CHICAGO -                  ! Queue on server node
                0 -                          ! Unit number of pseudo-device
                /NOENABLE_GENERIC            ! Optional qualifiers on local queue
Ctrl/Z
  113       $! DQS Client Queue Definitions
*EXIT
SYS$COMMON:[SYSMGR]DQS$SYSTARTUP.COM;6 119 lines
$
```

## Method 2: Defining Generic Client Queues

You can define client queues to be generic queues, which direct print jobs to execution queues on the client for processing. Defining generic queues gives you many independently targeted queues processed by a single symbiont. In this way, you trade off parallel job transfers for fewer processes on your client node.

## Procedure

To define generic client queues, perform the following steps:

| Step | Action |
|---|---|
| 1 | Edit the SYS$MANAGER:DQS$SYSTARTUP.COM.<br>You should include all generic queue definitions in the SYS$MANAGER:DQS$SYSTARTUP.COM file so that they are permanent. |
| 2 | Add a definition of the execution queue. Your command should have the following format:<br>`$ @SYS$MANAGER:DQS$DEFINE execution-queue -`<br>`_$ server-node remote-queue`<br>`_$ 0 /ENABLE_GENERIC[/qualifiers]` |
| 3 | Add the definition of the DQS$REMOTE_*client-queue* logical name for the associated server node and remote queue, in the form, *server-node*::*remote-queue*:<br>`$ DEFINE/SYSTEM/EXECUTIVE_MODE -`<br>`_$ DQS$REMOTE_client-queue server-node::remote-queue` |

| Step | Action |
|---|---|
| 4 | Add the command to initialize and start the queue specifying the /GENERIC qualifier. The parameter to the /GENERIC qualifier must be the name of the execution queue on the client system. Your command should have the following format:<br>`$ INITIALIZE/QUEUE/START/GENERIC=`***`execution-queue`*** `-`<br>`_$` ***`client-queue`*** |
| 5 | Save the DQS$SYSTARTUP.COM file and exit from the editor. |
| 6 | Execute the DQS$STARTUP.COM procedure. |

If the DQS$REMOTE_*client-queue* logical name is not defined for a generic queue, the logical name definition for the execution queue is used to establish the associated server node and remote queue.

## Example

Example 9–3 shows how to define a generic client queue.

### Example 9–3 Defining Generic Client Queues

```
$ @SYS$MANAGER:DQS$DEFINE D NODEC QUEUE4 0 /ENABLE_GENERIC Return

$ DEFINE/SYSTEM/EXECUTIVE_MODE DQS$REMOTE_A NODEA::QUEUE1 Return
$ INITIALIZE/QUEUE/START/GENERIC=D A Return

$ DEFINE/SYSTEM/EXECUTIVE_MODE DQS$REMOTE_B NODEB::QUEUE2 Return
$ INITIALIZE/QUEUE/START/GENERIC=D B Return

$ DEFINE/SYSTEM/EXECUTIVE_MODE DQS$REMOTE_C NODEA::QUEUE3 Return
$ INITIALIZE/QUEUE/START/GENERIC=D C Return
```

Example 9–3 defines the generic queues A, B, and C, which direct print jobs to the client execution queue D. There is a single symbiont running for queue D.

- If you print to queue A, your job transfers to server node NODEA, remote queue QUEUE1.
- If you print to queue B, your job transfers to server node NODEB, remote queue QUEUE2.
- If you print to queue C, your job transfers to server node NODEA, remote queue QUEUE3.
- If you print to queue D, your job transfers to server node NODEC, remote queue QUEUE4.

## Efficiency Suggestion

An execution queue connects to the associated server for each of its generic queues. However, defining generic queues with the same associated server to direct jobs to the same execution queue is most efficient.

## For More Information

Refer to the *OpenVMS System Manager's Manual* for more information on generic queues.

## Method 3: Defining Logical Client Queues

Logical queues are similar to generic queues in that each can have its own remote queue, independent of the execution queue. The difference is that you use the ASSIGN/QUEUE command to associate the logical queue with its execution queue.

The DQS software determines the associated server node and remote queue by looking for a DQS$REMOTE_*client-queue* logical name for each logical queue.

## Procedure

To define logical client queues, perform the following steps:

| Step | Action |
|---|---|
| 1 | Stop the client queue with the following command:<br>`$ STOP/QUEUE/RESET client-queue` Return |
| 2 | Define the DQS$REMOTE_*client-queue* logical name for the associated server node and remote queue, in the form, *server-node*::*remote-queue*. Your command should have the following format:<br>`$ DEFINE/SYSTEM/EXECUTIVE_MODE -`<br>`_$ DQS$REMOTE_client-queue server-node::remote-queue` Return |
| 3 | Associate the logical client queue with a client execution queue.<br>`$ ASSIGN/QUEUE execution-queue logical-queue` Return |
| 4 | Start the logical queue.<br>`$ START/QUEUE logical-queue` Return |

If DQS$REMOTE_*client-queue* is not defined for the logical queue, the DQS software uses the logical name definition for the execution queue.

## Example

Example 9–4 shows how to define a logical queue.

### Example 9–4 Defining Logical Client Queues

```
$ STOP/QUEUE/RESET QUEUE_A Return
$ DEFINE/SYSTEM/EXECUTIVE_MODE DQS$REMOTE_QUEUE_A NODEX::QUEUE_Y Return
$ ASSIGN/QUEUE QUEUE_D QUEUE_A Return
$ START/QUEUE QUEUE_A Return
```

Example 9–4 assigns the logical client queue QUEUE_A to direct print jobs to the execution queue QUEUE_D, which is a DQS queue. QUEUE_D processes jobs queued to QUEUE_A and directs them to the server NODEX, remote queue QUEUE_Y.

## For More Information

Refer to the *OpenVMS System Manager's Manual* for more information on logical queues.

## Temporarily Changing the Definition of a Client Queue

### Overview

The DQS software determines a client queue's associated server node and remote queue by translating the DQS$REMOTE_*client-queue* logical name to be the associated server and remote queue. The DQS$REMOTE_*client-queue* logical name is defined for a client queue by both the DQS$IMPORT.COM and the DQS$DEFINE.COM procedures.

### Procedure

To temporarily change the target of a client queue:

| Step | Action |
|---|---|
| 1 | Stop the client queue by entering the following command at the DCL prompt:<br>$ **STOP/QUEUE/RESET *client-queue*** Return |
| 2 | Redefine the DQS$REMOTE_*client-queue* logical name at the DCL prompt by entering:<br>$ **DEFINE/SYSTEM/EXECUTIVE_MODE DQS$REMOTE_*client-queue server-node::remote-queue*** Return |
| 3 | Restart the queue at the DCL prompt by entering:<br>$ **START/QUEUE *queue-name*** Return |

This temporary definition of the client queue is lost the next time the client system reboots because the logical name definition was not added to the SYS$MANAGER:DQS$SYSTARTUP.COM file.

## Example

Example 9–5 shows how to change the definition of a client queue temporarily. This example stops the client queue CQ_ILLINOIS, redefines its remote node to be server node CHERRY, remote queue Q_OHIO, and restarts the client queue.

### Example 9–5 Temporarily Changing the Remote Queue

```
$ STOP/QUEUE/RESET CQ_ILLINOIS Return
$ DEFINE/SYSTEM/EXECUTIVE_MODE DQS$REMOTE_CQ_ILLINOIS CHERRY::Q_OHIO Return
$ START/QUEUE CQ_ILLINOIS Return
```

## Deleting Client Queues

### Procedure

To delete a client queue, perform the following steps:

| Step | Action |
|---|---|
| 1 | Edit the SYS$MANAGER:DQS$SYSTARTUP.COM file and delete the lines that define the client queue. |
| 2 | Stop the queue by entering the following command at the DCL prompt:<br>$ **STOP/QUEUE/RESET *client-queue*** Return |
| 3 | Delete any jobs in the queue.<br>$ **QDELETE/ENTRY=*job-entry-numbers client-queue*** Return |
| 4 | Delete the queue by entering the following command at the DCL prompt:<br>$ **DELETE/QUEUE *client-queue*** Return |
| 5 | Deassign the queue logical name:<br>$ **DEASSIGN/SYSTEM/USER DQS$REMOTE_*queue-name*** Return<br>Where *queue-name* is the name of the client queue that you deleted. |

## Configuring Client Nodes in a VMScluster Environment

**Execute Startup File**

You must execute the SYS$STARTUP:DQS$STARTUP.COM command file on each cluster member to define the required DQS logical names on each cluster member. This action makes available the DQS software to users in the cluster regardless of the cluster node.

**Cluster Alias**

You should use an outgoing cluster alias as the name of your client system. This makes the cluster appear to be one system to its server(s). If you do not define and use this cluster alias, users may switch cluster members and may not be able to show, modify, or delete print jobs that they own.

## Synchronizing Client and Server Forms Definitions

### About the DQS$CLIENT Program

You should synchronize form definitions between your client and server systems. Do this by running the DQS client program SYS$SYSTEM:DQS$CLIENT.EXE. This requires OPER, TMPMBX, and NETMBX privileges.

The DQS$CLIENT program looks at the queues on a client. For each queue, it queries the associated server(s) to determine what forms are defined there. DQS$CLIENT then defines all the server forms on the client system.

### Procedure

To synchronize forms on your DQS client and server systems:

| Step | Action |
|---|---|
| 1 | Run the DQS$CLIENT program by issuing the following command:<br>$ **RUN SYS$SYSTEM:DQS$CLIENT** Return |

DQS$CLIENT synchronizes client forms with the server forms in alphabetical order of the client queue names. Each server is queried only once, regardless of the number of client queues associated with remote queues on that server.

### Conflicting Server Forms Definitions

If DQS$CLIENT finds conflicting form definitions on different servers, it is not able to define both forms on a DQS client. For example, if one DQS server defines form 3 as X and another defines 3 as Y, form 3 can be defined as either X or Y at a DQS client. No practical way exists to determine which case of form 3, X or Y, occurs at the client.

If DQS$CLIENT encounters more than one form number for a given form name, you may not get the results you desire.

## Form Definition Override

For DQS$CLIENT to override the local form definition with its value on the server, you must define the DQS$FORM_OVERRIDE logical name on the client system to be an arbitrary value.

If you do not define DQS$FORM_OVERRIDE on your client node, your local form definitions are maintained.

When you run DQS$CLIENT, you are notified that your local form definition is superseded by the server's definition (with DQS$FORM_OVERRIDE defined) or remains the same (DQS$FORM_OVERRIDE not defined).

## Procedure

To allow the server form definition to supersede the client definition:

| Step | Action |
|---|---|
| 1 | Edit SYS$MANAGER:DQS$SYSTARTUP.COM file and search for the following statement:<br>`$! Form Synchronization:` |
| 2 | Uncomment the DQS$FORM_OVERRIDE logical name with an arbitrary value, for example, TRUE.<br>`$ DEFINE/SYSTEM/EXECUTIVE_MODE DQS$FORM_OVERRIDE`<br>`_$ TRUE` Return |
| 3 | Exit from the DQS$SYSTARTUP.COM file. |

## Spooling to a Client Queue

### Procedure

To spool to a DQS client queue, you must associate a LAT port with the DQS client queue. To do so, perform the following steps:

| Step | Action |
|---|---|
| 1 | Run the LATCP utility. |
| 2 | Create the LAT port LTA*nnn*:.<br>The *nnn* represents a number that is not already in use on your system, for example, LTA999:. |
| 3 | Exit from LATCP utility. |
| 4 | Set that LAT port device spooled. |

A user-written program can then output to the specified LAT port device. The program output is written to a disk and queued to the specified output queue when the file is closed.

### Note

If the LAT software is not started, enter the following command at the OpenVMS DCL prompt:

```
$ @SYS$STARTUP:LAT$STARTUP.COM Return
```

### Example

Example 9–6 shows how to set up a LAT port device.

#### Example 9–6 Setting Up a LAT Port Device

```
$ MCR LATCP Return
LATCP> CREATE PORT LTA999: Return
LATCP> SHOW PORT Return
Port Name  Port Type    Status         Remote Target (Node/Port/Service)
---------  ----------  -------------  ----------------------------------------
_LTA999:   Application  Inactive       //

LATCP> EXIT Return
$ SET DEVICE LTA999:/SPOOLED=QUEUE_Z Return
```

You can spool this LTA999 device to only one DQS client queue at a time. Example 9–7 shows how to change the DQS client queue that is spooled to the LTA9999: device, by entering the following DCL commands where *new_queue_name* is the name of the new DQS client queue:

### Example 9–7 Changing Client Queue Spooled to LTA999: Device

```
$ SET DEVICE LTA999/NOSPOOLED Return
$ SET DEVICE LTA999/SPOOLED=new_queue_name Return
```

Example 9–8 shows how to use the LAT port with a FORTRAN program. In this example, the FOR$PRINT logical is associated with the LTA999: device. In this case, the LTA999: device is spooled to a DQS client queue QUEUE_Z (see Example 9–6). When the program runs, the output is directed through the FOR$PRINT logical to the LTA999: device, and from there to the spooled queue, QUEUE_Z.

### Example 9–8 Spooling to a LAT Port Device

```
$ DEFINE FOR$PRINT LTA999: Return
$ SHOW QUEUE QUEUE_Z Return
  Server queue QUEUE_Z, stopped, on YOUNG::, mounted from DEFAULT
$ TYPE EXAMPLE.FOR Return
      print *,'This is a test.'
      end
$ RUN EXAMPLE Return
$ SHOW QUEUE QUEUE_Z Return
  Server queue QUEUE_Z, stopped, on YOUNG::, mounted from DEFAULT

   Jobname         Username     Entry  Blocks  Status
   -------         --------     -----  ------  ------
   FORPRINT        YOUNG          573       1  Pending
   FORPRINT        YOUNG          573       1  Pending
$
```

This example shows QUEUE_Z as stopped to show the submitted job in the DQS server queue.

## Controlling the Note on the Job Banner Page

### How the Note is Printed

The DQS software makes use of a print attribute to place a note on the banner page of a print job when logical names are defined in the DQS$SYSTARTUP.COM file. On the client node, this logical name definition provides the default text ''Print Job transported by DQS V*n.n*''. The note defined by this logical name prints on the banner page of a print job if the /NOTE qualifier is not specified by the client node user on the PRINT command.

The note that prints on the banner page of a print job follows this order of precedence:

- Note specified by the user with PRINT/NOTE
- Note specified by the client logical name definition (DQS$CLIENT_DEFAULT_JOB_NOTE)
- Note specified by the server logical name definition (DQS$SERVER_DEFAULT_JOB_NOTE)

### Options

You have the following options:

- Use the default note provided by the DQS software for the banner page.
- Turn off the DQS logical definitions for the banner page note.
- Define your own note for the banner page.

### To Turn Off the Default

To turn off the DQS default note on the client system, do the following:

| Step | Action |
|---|---|
| 1 | Edit the SYS$MANAGER:DQS$SYSTARTUP.COM file. |
| 2 | Search for the following command and remove the exclamation mark ( ! ) comment character:<br>`$! $ DEASSIGN/SYSTEM/EXEC DQS$CLIENT_DEFAULT_JOB_NOTE` |
| 3 | Save the file SYS$MANAGER:DQS$SYSTARTUP.COM and exit from the editor. |

The change takes efect the next time the SYS$STARTUP:DQS$STARTUP.COM is executed.

### To Change Note Text

To change the text of the note that prints on the banner page, do the following:

| Step | Action |
|---|---|
| 1 | Edit the SYS$MANAGER:DQS$SYSTARTUP.COM file. |
| 2 | Search for the following command:<br>`$! $ DEFINE/SYSTEM/EXEC DQS$CLIENT_DEFAULT_JOB_NOTE -`<br>`$!  "Place your text here"`<br>Substitute your note for the text ‘‘Place your text here’’ and remove the dollar sign and exclamation mark ( $! ) at the beginning of each line. |
| 3 | Save the file SYS$MANAGER:DQS$SYSTARTUP.COM and exit from the editor. |

The next time the SYS$STARTUP:DQS$STARTUP.COM is executed and the note is not specified by the user, your customized note prints on the job banner page.

## Regulating Forms Checking

### The Forms Change Timer

A part of the DQS software known as the forms change timer periodically checks DQS client queues to see if an operator or the software, itself, has properly changed forms.

The DQS software requires a match of form types between the one mounted on a DQS client queue and the one specified for each of the queue's jobs.

### Logical Names

Regulating the forms change timer component requires editing the SYS$MANAGER:DQS$SYSTARTUP.COM file and adding logical name definitions:

| Logical Name | Parameter | Action |
|---|---|---|
| DQS$NO_FORMS | TRUE | Turns off the forms timer on all DQS queues.<br>If a DQS client node submits a job to one of its queues and the job requires a form that is not mounted on the queue, no data transfers, no errors occur, and the job waits indefinitely. |
| DQS$FORM_*queue_name* | TRUE | Names a particular DQS client queue, *queue_name*. You can repeat the logical definition for any number of DQS client queues.<br>If a DQS client node submits a job to the named queue with a form that is not mounted on the queue, the data transfers and no errors occur. The result is similar to having the form mounted on the queue. |

### Procedure

To regulate forms checking, do the following:

| Step | Action |
|---|---|
| 1 | Edit the SYS$MANAGER:DQS$SYSTARTUP.COM file to add the logical names.<br>Look for the following statement:<br>`$! Forms Change Timer:` |
| 2 | Uncomment the command that defines the logical name DQS$NO_FORMS to be TRUE for all DQS client queues:<br>`$ DEFINE/SYSTEM/EXECUTIVE_MODE DQS$NO_FORMS "TRUE"` |
| 3 | Uncomment the command that defines a logical name DQS$FORM_*queue_name* to be TRUE, where *queue_name* is a particular DQS client queue.<br>`$ DEFINE/SYSTEM/EXECUTIVE_MODE DQS$FORM_queue_name "TRUE"`<br>This command applies only to the named DQS client queue. Repeat this command for any number of DQS client queues. |
| 4 | Save the file, SYS$MANAGER:DQS$SYSTARTUP.COM and exit from the editor. |
| 5 | Execute the SYS$STARTUP:DQS$STARTUP.COM procedure:<br>`$ @SYS$STARTUP:DQS$STARTUP.COM` Return |

## Example

Example 9–9 shows how to add logical names to regulate forms checking on VMScluster systems with 50 clients or more. This examples turns off the forms change timer for all client queues and allows client queues LNO6_ROOM23A, LPS17_NUM016, and LPS17_600_BLUE to accept print jobs regardless of the forms mounted on them.

**Example 9–9 Regulating Forms Checking on Many Client Systems**

```
$!
$! Forms Change Timer:
$!
$!   Uncomment the following logical if you desire to turn off the forms
$!   change timer. Digital recommends that you do this on VMScluster
$!   systems with 50 or more DQS client queues.
$!
! DEFINE/SYSTEM/EXECUTIVE_MODE DQS$NO_FORMS TRUE
$!
$! Ignoring forms definitions on specific client queues:
$!
$!   You can disable certain client queues from rejecting print jobs when
$!   the form mounted on the client queue differs from  the form mounted
$!   on the remote queue.  Define a logical of the form DQS$FORM_queue_name .
$!   This allows the client queue, 'queue_name' ,  to accept and process jobs,
$!   regardless of the form mounted on the remote queue. For example:
$!
$!
$ DEFINE/SYSTEM/EXECUTIVE_MODE DQS$FORM_LNO6_ROOM23A "TRUE"
$ DEFINE/SYSTEM/EXECUTIVE_MODE DQS$FORM_LPS17_NUM016 "TRUE"
$ DEFINE/SYSTEM/EXECUTIVE_MODE DQS$FORM_LPS17_600_BLUE "TRUE"
```

After adding these logical name definitions to the site-specific startup file DQS$SYSMANAGER:DQS$SYSTARTUP.COM and executing the DQS startup procedure SYS$STARTUP:DQS$STARTUP, the DQS software no longer checks for forms mismatches between each named queue and its jobs.

# 10
# Troubleshooting

## About This Chapter

This chapter provides some general guidelines for troubleshooting apparent problems with the DQS software and suggests recovery procedures when:

- Remote notification is not working
- Jobs remain in the remote queue with completion status
- A job is not printing
- Large numbers of DQS client queues slow down the network

The guidelines are not meant to cover all possible error conditions, but are intended to help you understand some common situations.

## General Troubleshooting Guidelines

### What To Do First

If something fails to work with the DQS software, check these first:

- Determine whether your DQS configuration has changed recently.
- Run the DQS Installation Verification Procedure (IVP) located in SYS$TEST:DQS$IVP.COM. It may confirm that the DQS software is functioning properly, or it may detect the error condition.
- Check that the DQS$STARTUP.COM procedure is invoked in the system start-up file,
  - SYS$STARTUP:SYSTARTUP_V5.COM for OpenVMS VAX V5.*n* systems
  - SYS$STARTUP:SYSTARTUP_VMS.COM for OpenVMS VAX V6.*n* systems and OpenVMS AXP systems

  DQS$STARTUP.COM starts the DQS software and invokes procedures that define the required DQS logical names. If the DQS$VERSION system logical name is not defined, then this may be the cause of your error condition.
- On client nodes, check that queues are defined in the SYS$MANAGER:DQS$SYSTARTUP.COM file.
- On server nodes, check that remote queues are set up in the SYS$MANAGER:DQS$SERVER_CONFIG.TXT file.

## Running the IVP

### Command

You can execute the Installation Verification Procedure (IVP) to check if the DQS software is running properly. To run the IVP, enter:

```
$ @SYS$TEST:DQS$IVP.COM Return
```

### IVP on a Client Node

Following is a sample of an IVP run on a node with the DQS client software configured.

```
 DQS V1.3 Installation Verification Procedure

  Restricted Rights: Use, duplication, or disclosure by the U.S. Government
  is subject to restrictions as set forth in subparagraph (c)(1)(ii) of
  DFARS 252.227-7013, or in FAR 52.227-19, or in FAR 52.227-14 Alt. III, as
  applicable.

  This software is proprietary to and embodies the confidential technology
  of Digital Equipment Corporation. Possession, use, or copying of this
  software and media is authorized only pursuant to a valid written license
  from Digital or an authorized sublicensor.
```

For a client-only node, the following prompt is displayed:

```
It is necessary to test the DQS client with a remote DQS server node.
Enter remote server NODE::QUEUE:
```

Enter the name of a server node and a remote queue on that node. For example, to test the proper operation of server queue Q_CHICAGO on node APPLE, enter the following:

```
Enter remote server NODE::QUEUE APPLES::Q_CHICAGO RETURN
```

For the IVP to succeed on a client-only installation, the following must exist:

- A remote queue on the server node that is accessible from the client system (see the chapter on configuring a DQS server in the Chapter 8 for information on making server queues accessible).

- The DQS$IVP_TEST_FORM on the server node. If this form no longer exists on the server node, enter the following DCL commands on the server node:

```
$ DEFINE/FORM DQS$IVP_TEST_FORM 1110 -
        /STOCK = DQS_IVP_TEST_STOCK -
        /DESCRIPTION = "DQS test form, do not delete" Return
```

Then, if the DQS software is working properly. the IVP displays a series of messages (this may take several minutes), such as:

```
%DQS-I-CRESRVQUE Creating remote queue
%DQS-I-CRECLIQUE Creating client queue
%DQS-I-CHKCLIQUE Checking client queue
%DQS-I-ENTJOB Entering test job in client queue
%DQS-I-CHECKJOB Checking test job in client queue
%DQS-I-SNDJOB Sending job to remote queue
%DQS-I-CHKSRVQUE Checking test job in remote queue
%DQS-I-DELJOB Deleting job from remote queue
%DQS-I-CHKDEL Checking that job is gone from remote queue
%DQS-I-CLEANUP Cleaning up, deleting IVP queues
%DQS-I-QUESUCCESS IVP test successful
```

## IVP on a Server Node

Following is a sample of an IVP run on a node with the DQS server software configured.

```
DQS V1.3 Installation Verification Procedure

 Restricted Rights: Use, duplication, or disclosure by the U.S. Government
 is subject to restrictions as set forth in subparagraph (c)(1)(ii) of
 DFARS 252.227-7013, or in FAR 52.227-19, or in FAR 52.227-14 Alt. III, as
 applicable.

 This software is proprietary to and embodies the confidential technology
 of Digital Equipment Corporation. Possession, use, or copying of this
 software and media is authorized only pursuant to a valid written license
 from Digital or an authorized sublicensor.
```

Then, if the DQS software is working properly. the IVP displays a series of messages (this may take several minutes), such as:

```
%DQS-I-CRESRVQUE Creating remote queue
%DQS-I-CRECLIQUE Creating client queue
%DQS-I-CHKCLIQUE Checking client queue
%DQS-I-ENTJOB Entering test job in client queue
%DQS-I-CHECKJOB Checking test job in client queue
%DQS-I-SNDJOB Sending job to remote queue
%DQS-I-CHKSRVQUE Checking test job in remote queue
%DQS-I-DELJOB Deleting job from remote queue
%DQS-I-CHKDEL Checking that job is gone from remote queue
%DQS-I-CLEANUP Cleaning up, deleting IVP queues
%DQS-I-QUESUCCESS IVP test successful
```

### For Information on Error Messages

If the DQS IVP encountered problems, the procedure displays error messages. Refer to Appendix B, DQS System Manager Messages for an explanation of specific error messages.

## Problem: Remote Notification Is Not Working for Any Jobs

**Action**

If remote notification is not working check for the following:

- The DQS$NOTIFIER process is running on the server.
- The network object DQS is defined on the client.
- The client node is defined in the server node's NCP database for DECnet Phase IV systems or the server node's NCL database for DECnet/OSI Phase V systems.
- If the client node is in a cluster, make sure that the DQS software is properly configured on all nodes of the cluster.

## Problem: Notification of Print Job Completion Is Not Occuring for Some Jobs

### Action

The DQS$NOTIFIER searches the node for all known DQS server queues when it starts. If you define a new queue as a valid DQS server queue, you must stop and restart the DQS$NOTIFIER process to enable print job completion messages of jobs submitted to the new queue. The simpliest way to do this is to restart DQS.

If the DQS$NOTIFIER is not restarted:

- Print jobs remain in the DQS server queue with the retained attribute set.
- The temporary files are not deleted.
- The user is not notified when the job is completed.

## Problem: Jobs Remain in the Remote Queue With Completion Status

**Action**

- Make sure that the DQS$NOTIFIER process is running on the server.
- If the server is a member of a VMScluster system in which the queues are not common across the cluster members, check the definition of the DQS$NOTIFY_LOCK logical name as described in Chapter 8.

## Problem: A Job Is Not Printing

### Information You Need

To determine why a print job does not appear to be printing, you need to determine the following for the job:

- The status; is it in the process of printing on the server node? Is it stopped or stalled?
- The location; is it in the client queue or the remote queue?

### Using the QSHOW Command

To obtain this information, issue a QSHOW command at the client system of the client queue to which the job was queued for printing. The output of the QSHOW command provides information on both the status and the location of the print job.

### Examining the QSHOW Display

The output of the QSHOW command is designed to provide all the information you need to determine the status and location of a print job. The command output usually appears as follows, showing the location of the job on either the client or the server system:

```
** Remote queue - [ CQ_EUROPE => Q_PARIS, on CHERRY:: ]
Terminal queue Q_PARIS

============================to be transferred==================
Server queue CQ_EUROPE, mounted form DEFAULT

  Jobname         Username     Entry  Blocks  Status
  -------         --------     -----  ------  ------
  JOB             GUENTHER      2737       1  Printing
```

### Understanding Location Information

Note that the client queue and remote queue are referred to in the command output as the "Server queue" and the "Remote queue," respectively. This terminology is tied to the OpenVMS concept of served queues. For the DQS QSHOW command,

- Server queue refers to the DQS client queue
- Remote queue refers to the DQS remote queue

When the QSHOW command output contains information for both the client and remote queues (as shown in the preceding display) you should first decide whether the job is in the client queue, in the remote queue, or neither.

### Location: Neither Node

If the job is not in either the client or the remote queue, then the following possibilities exist:

- The job printed and is lost.
- The user printed to a different queue or printed a different job.
- The actual printer was different from the expected one. The system manager or the DQS software may have rerouted the request to another available printer.

If you suspect a DQS problem, examine the NETSERVER.LOG files in the DQS server account's directory or, if OpenVMS accounting is enabled, check the accounting files.

### Location: Server Node

If the job is located on the server node, then the DQS software has functioned properly. If the job still is not printing, then the following possibilities exist:

- It may be pending because of form or characteristic problems.
- It may be held or waiting to print after a specified time.

Check that the printer is on line, powered on, and has paper in it.

### Location: Client Node

If the job is still located on the client node, then examine the status of the client queue in the QSHOW output. The queue status is either stopped or stalled.

### Understanding Status Information

You see the following status if there is a problem:

- **Stopped:** If the queue is stopped, the QSHOW command output contains the reason for stopping the queue if the DQS software stopped the queue. The queue may be stopped because the remote system is unknown or because the client node is denied access to the server.

  Check the OPCOM output or the OPERATOR.LOG file for messages regarding why the queue stopped.

- **Stalled:** Stalled means that the DQS symbiont has not sent a checkpoint message to the job controller within the timeout limit of the stall timer. This could mean that the server is running slowly. It could also mean that the DQS

symbiont failed to connect to the server and is in the process of retrying the connect.

Check for the OPCOM message "Remote system off line."

Disk problems on the server can also cause a stalled state.

The stalled state is transient and requires no special attention. The DQS software automatically tries the job every five minutes. Generally, this action alone corrects the stalled state.

Any of the following indicate that the client queue is operational and running:

- **Printing:** Indicates that the job is transferring to the server node. This may signify that the user was impatient, that the server system was overloaded, or that the transfer is slow. This status is shown as follows:

```
** Remote queue - [ CQ_EUROPE => Q_PARIS, on CHERRY:: ]
Terminal queue Q_PARIS

==============================to be transferred==================
Server queue CQ_EUROPE, mounted form DEFAULT

  Jobname         Username     Entry  Blocks  Status
  -------         --------     -----  ------  ------
  JOB             MACOMBER      2738       1  Printing
```

- **Pending:** Indicates that the job cannot be transferred for some reason local to the client system. You can determine the reasons for this pending state by entering:

  $ **QSHOW/FULL *queue-name*** Return

  Possible reasons for a pending status are the following:

  - A characteristic was used to print the job that is not enabled for the queue.

    In this case, add the appropriate characteristics to the queue or remove them from the print job.

  - The form type mounted for the queue may not match the form type specified for the job.

    In this case, the DQS software automatically changes forms.

This status is shown as follows:

```
** Remote queue - [ CQ_EUROPE => Q_PARIS, on CHERRY:: ]
Terminal queue Q_PARIS

============================to be transferred==================
Server queue CQ_EUROPE, mounted form DEFAULT

  Jobname         Username     Entry  Blocks  Status
  -------         --------     -----  ------  ------
  JOB             DAVIES        2739       1  Pending
```

- **Holding:** Indicates that the job cannot be transferred because the user specified that the job should be held. This status is shown as follows:

```
** Remote queue - [ CQ_EUROPE => Q_PARIS, on CHERRY:: ]
Terminal queue Q_PARIS

============================to be transferred==================
Server queue CQ_EUROPE, mounted form DEFAULT

  Jobname         Username     Entry  Blocks  Status
  -------         --------     -----  ------  ------
  JOB             WOODWARD      2740       1  Holding
```

- **Holding until:** Indicates that the job cannot be transferred because the user specified that the job should be held until after the specified time. This status is shown as follows:

```
** Remote queue - [ CQ_EUROPE => Q_PARIS, on CHERRY:: ]
Terminal queue Q_PARIS

============================to be transferred==================
Server queue CQ_EUROPE, mounted form DEFAULT

  Jobname         Username     Entry  Blocks  Status
  -------         --------     -----  ------  ------
  JOB             SMITH         2741       1  Holding until
                                                 1-Jan-1994 01:00
```

- **Retained on completion:** Indicates that the job has transferred to the server system. This status is shown as follows:

```
** Remote queue - [ CQ_EUROPE => Q_PARIS, on CHERRY:: ]
Terminal queue Q_PARIS

============================to be transferred==================
Server queue CQ_EUROPE, mounted form DEFAULT
```

```
   Jobname         Username     Entry  Blocks  Status
   -------         --------     -----  ------  ------
   JOB             SCHULLMAN     2742       1  Retained
                                                    on completion
```

- **Retained on error:** Indicates that the job failed to transfer to the server system because of the error shown with the job.

  The user should correct the problem and retry the request. This status is shown as follows:

```
** Remote queue - [ CQ_EUROPE => Q_PARIS, on CHERRY:: ]
Terminal queue Q_PARIS

==============================to be transferred==================
Server queue CQ_EUROPE, mounted form DEFAULT

  Jobname         Username     Entry  Blocks  Status
  -------         --------     -----  ------  ------
  JOB             TIMLEGE       2743       1  Retained on error
    %JBC-E-NOSUCHFORM, no such form
```

## The Remote Queue Display

Issuing a QSHOW command for a client queue sometimes does not show job status information because either the remote queue or the server node is inaccessible. In these cases, you see the following in the QSHOW command output:

- No remote queue information
- A message explaining why the remote queue is not reachable

## Missing Remote Queue Information

When the remote queue information does not display when you issue a QSHOW command, it means that the DQS software does not view the client queue as a valid DQS queue. The command output would appear as follows:

```
 Server queue CQ_EUROPE, mounted form DEFAULT

   Jobname         Username     Entry  Blocks  Status
   -------         --------     -----  ------  ------
   JOB             CHO           2744       1  Pending
```

In this case, you should issue a QSHOW/FULL command for the queue. If the command output contains the information /PROCESSOR=DQS$SMB, invoke the DQS$STARTUP.COM file to define the queue as a valid DQS queue.

You can also check the DQS$REMOTE_*client-queue* logical name definition for the client queue. It should be defined system wide and in executive mode to be the corresponding server node and remote queue.

## Error Messages in the Remote Queue Display

In some cases, the QSHOW command output contains an error message that cites a reason why the remote queue is not accessible. The command output would appear as follows:

```
***** Remote queue CHERRY::Q_PARIS not accessible *****
    %SYSTEM-F-UNREACHABLE, remote node is not currently reachable

==============================to be transferred=======================
Server queue CQ_EUROPE, mounted form DEFAULT
```

The following is a list of error messages that might be displayed when a remote queue is inaccessible:

- -DQS-F-BADNODE, specified node has been denied access to this server.

```
%DQS-W-MSG_REMOTE, remote system error:
-DQS-F-BADNODE, specified node has been denied access to this server.
%SYSTEM-F-LINKDISCON, network partner disconnected logical link
```

**Explanation:** The specified queue is not accessible with the DQS software because the client system has been specifically denied access to the server.

**User Action:** Check with the server system manager to remove the client's node name from deny status in the DQS$SERVER_CONFIG.TXT file.

- -DQS-F-BADQUE, specified queue not valid for DQS access.

```
%DQS-W-MSG_REMOTE, remote system error:
-DQS-F-BADQUE, specified queue not valid for DQS
access.

==============================to be transferred=======================
Server queue CQ_EUROPE, mounted form DEFAULT
```

**Explanation:** The specified queue is not accessible with the DQS software. Either the client's node name was not added to the access list or the remote queue does not exist.

**User Action:** Check with the server system manager to determine the reason for the error.

- %SYSTEM-F-INVLOGIN, login information invalid at remote node

```
***** Remote queue CHERRY::Q_PARIS not accessible *****
    %SYSTEM-F-INVLOGIN, login information invalid at remote node

==============================to be transferred=======================
Server queue CQ_EUROPE, mounted form DEFAULT
```

**Explanation:** The server account information may not be defined properly.

**User Action:** Make sure the server's password in the network object database on the server matches the password in the UAF. Also, make sure the owner and UIC of the server's directory is the server's account.

- -DQS-E-NOTSTARTED, DQS has not been started yet

```
%DQS-W-MSG_REMOTE, Remote system error:
-DQS-E-NOTSTARTED, DQS has not been started yet

 ==============================to be transferred==================
 Server queue CQ_EUROPE, mounted form DEFAULT
```

**Explanation:** The DQS software was not started on the server.

**User Action:** Ask the server system manager to start the DQS software. After the DQS software is started, there may be a delay of up to five minutes before jobs are transferred.

- %SYSTEM-F-NOSUCHNODE, remote node is unknown

```
 ***** Remote queue CHERRY::Q_PARIS not accessible *****
     %SYSTEM-F-NOSUCHNODE, remote node is unknown

 ==============================to be transferred==================
 Server queue CQ_EUROPE, mounted form DEFAULT
```

**Explanation:** The server node is not known to your client node.

**User Action:** Either add the server node to your DECnet node database or change the node name to a known node.

- %SYSTEM-F-NOSUCHOBJ, network object is unknown at remote node

```
***** Remote queue CHERRY::Q_PARIS not accessible *****
    %SYSTEM-F-NOSUCHOBJ, network object is unknown at remote node

==============================to be transferred==================
Server queue CQ_EUROPE, mounted form DEFAULT
```

**Explanation:** Either the specified node is not a DQS server node or the DQS server has not been started.

**User Action:** Change the node to a DQS server or start the DQS server on the server node.

- %SYSTEM-F-UNREACHABLE, remote node is not currently reachable

```
***** Remote queue CHERRY::Q_PARIS not accessible *****
    %SYSTEM-F-UNREACHABLE, remote node is not currently reachable

==============================to be transferred==================
Server queue CQ_EUROPE, mounted form DEFAULT
```

**Explanation:** The server is not on the network. If the job is in the queue, it is transferred to the server soon after the server becomes reachable.

**User Action:** Wait until the server becomes reachable or queue your print job to a remote queue on another server node.

## Problem: Many DQS Client Queues Slow Down the Network

### About the Problem

A part of the DQS software known as the forms change timer periodically checks DQS client queues to see if an operator or the software has properly changed forms. The DQS software requires a match of form types between the one mounted on a DQS client queue and the one specified for each of the queue's jobs. This checking can slow down the overall performance of a VMScluster system with 50 or more DQS client queues.

### Action

To increase performance of systems with many DQS client queues, you can turn off the forms change timer component of the DQS software. As a result, the software cuts forms-checking operations to a minimum and performance is improved. Digital recommends that you do this on VMScluster systems with 50 or more DQS client queues.

Regulating the forms change timer component requires editing the SYS$MANAGER:DQS$SYSTARTUP.COM file and adding logical name definitions DQS$NO_FORMS and DQS$FORM_*queue_name*. For information, refer to the section titled Regulating Forms Checking in Chapter 9.

# Part III

## Installation Guide

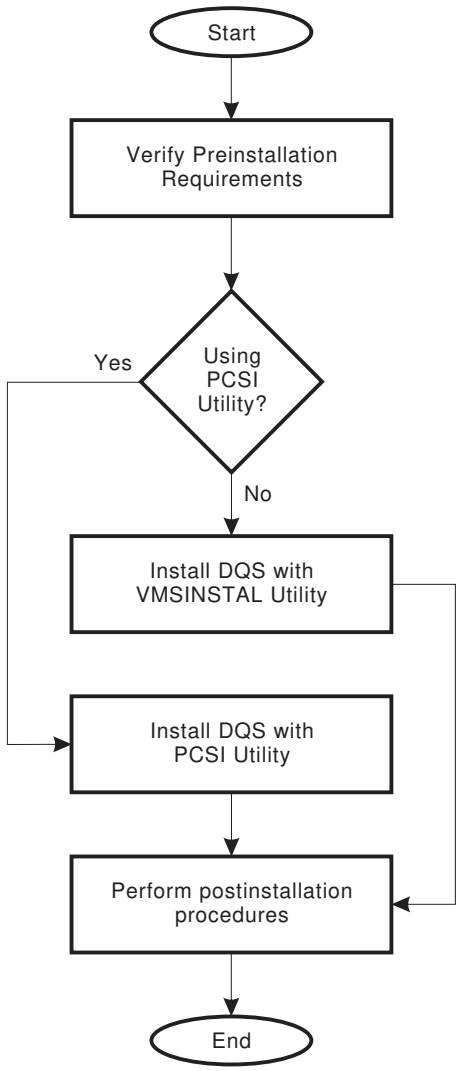
### Overview

This part describes how to install the DQS software on OpenVMS AXP and OpenVMS VAX systems using either the POLYCENTER Software Installation (PCSI) utility or the VMSINSTAL installation utility. The Installation Guide is intended for the network or system manager who is responsible for installing the software.

### Using the Installation Guide

You can use either the PCSI utility or the VMSINSTAL utility to install the software.

Refer to the following flowchart to determine which sections of this guide are relevant to the installation utility that you are using.




**Step 1:** (See Chapter 11 and Chapter 12)

**Step 2:** (See Chapter 14)

(See Chapter 13)

**Step 3:** (See Chapter 15)

MLO-012651

# 11
# What You Need to Know

## About This Chapter

This chapter describes what you need to know before installing the DEC Distributed Queuing Service for OpenVMS Systems (DQS) software on an OpenVMS AXP or OpenVMS VAX system. Table 11–1 lists the topics included.

**Table 11–1 Summary: What You Need to Know**

| Topic | Title |
|---|---|
| 1 | Required hardware and software |
| 2 | Licensing |
| 3 | Processes created |
| 4 | First-time installation considerations |
| 5 | Upgrade considerations |
| 6 | Time required to install |

## Related Information

Chapter 12 describes what you need to do before you install the DQS software.

Chapter 13 describes how to install the DQS client and server software with the PCSI utility. Chapter 14 describes how to install the DQS client and server software with the VMSINSTAL utility.

Refer to Part II, the System Manager's Guide, for information on how to set up and manage DQS software on client and server nodes after installation.

## For Information on Terms

This guide refers to DQS server nodes, client nodes, remote queues, and client queues. For a definition of these terms and concepts, refer to Chapter 7 in Part II, the System Manager's Guide.

## Required Hardware and Software

### Hardware Required

DQS software can be installed on any Alpha AXP or VAX processor.

### Operating System Versions

Installation of DEC Distributed Queuing Service Version 1.3 for OpenVMS systems software requires one of the following:

- OpenVMS VAX Operating System, Version 5.5-2 or higher
- OpenVMS AXP Operating System, Version 6.1 or higher

Future DEC Distributed Queuing Service for OpenVMS Systems releases may require higher versions of the operating system, as described in the online release notes or the product cover letter.

### OpenVMS Tailoring Classes

The OpenVMS operating system comes with a variety of support options, or tailoring classes. Tailoring classes include such features as networking and RMS journaling. For full functionality of this layered product, the following tailoring classes are required:

- OpenVMS Required Saveset
- Network Support
- Secure User's Environment

## Prerequisite Products

Table 11–2 provides information on the software you must have installed on your system before installing DQS software.

**Table 11–2 Prerequisite Software**

| Prerequisite Products | Purpose |
|---|---|
| DECnet for OpenVMS Systems, V5.5 (Phase IV) or DECnet/OSI for OpenVMS Systems, V5.6 (Phase V) | Networking software; Required for DQS client to server and server to client communication |

## For More Information

For a complete list of the supported hardware configurations, required operating system components and software, see the *DQS Software Product Description* (SPD) included with your distribution kit. In addition, the SPD contains a complete list of prerequisite and optional software and their required version numbers.

## Licensing

### Requirements

The DQS product supports the OpenVMS License Management Facility (LMF) provided by OpenVMS systems. Before you can start the DQS software, LMF requires that you register a License Product Authorization Key (License PAK) for the product. The License PAK is shipped with the kit if you ordered a license and media together; otherwise, it is shipped separately to a location specified by your license order. Use the LMF utility to register and load the information from your License PAK into your system's license database.

Traditionally, each node that executes the DQS software requires a license.

### For More Information

Refer to the *DEC Distributed Queuing Service for OpenVMS Systems Software Product Description (SPD)* for more information on DQS licensing. For information on how to license DQS software after it is installed, see the *OpenVMS License Management Utility Manual*.

## Processes Created

### Client Nodes

On a client node, every client queue you create that is not a generic or logical queue, creates one process. Also, the system creates a temporary process when the DQS software notifies the user of a print job completion.

### Server Nodes

On server nodes, every incoming connection from a client node results in the creation of a process. Also, the notification process on a server node uses one permanent process.

## First-time Installation Considerations

### Registration of Products with PCSI

If you are installing the DQS product with the PCSI utility, then any software product required for DQS installation must also be registered in the PCSI product database. Products installed with installation procedures other than the PCSI utility, such as the VMSINSTAL utility, cannot be registered in the PCSI product database.

### For More Information

For information on how to determine if products are registered in the PCSI database and how to register products in the database, see the section titled Perform First-Time Installation Activities in Chapter 12.

## Upgrade Considerations

### New Startup File

Version 1.3 of the DQS software provides a new DQS$STARTUP.COM command procedure. Your older version of DQS$STARTUP.COM file is copied to DQS$STARTUP.COM_OLD. The new startup procedure does not contain DQS queue definitions or definitions for logical names that you can change. Instead an editable file, SYS$MANAGER:DQS$SYSTARTUP.TEMPLATE, is provided where you can make changes to DQS logical name definitions and define client queues. You copy this file by changing .TEMPLATE to .COM and making your changes.

Whenever, you execute the DQS startup procedure SYS$STARTUP:DQS$STARTUP.COM, it invokes the site-specific startup procedure SYS$MANAGER:DQS$SYSTARTUP.COM.

### New Security and Queue Access System

Version 1.3 of the DQS software provides a new method to define a set of accessible server queues and to control access to these queues.

During the installation procedure, the DQS software attempts to use information from the currently running system to convert your server queue definitions into the new security and queue access system. This conversion may not be correct even though it is successful. After the installation you need to validate that your server queue definitions are correct.

### Note

Make sure that the DQS software is running on the server node when you are installing server software. Conversion of your current server queue definitions into the new security and queue access system requires this. If DQS software is not running you have the following options:

- Stopping the installation, executing the startup procedure, and restarting the installation
- Manually entering your server queue definitions and client access information after the installation is complete.

## For More Information

| For more information on . . . | Refer to . . . |
|---|---|
| Validating the conversion of server queue definitions | Chapter 15 |
| Adding or changing DQS logical name definitions | Part II, System Manager's Guide, Chapter 8, Defining Server System Logical Names |
| Setting up client node access to remote queues on server nodes | Part II, System Manager's Guide, Chapter 8, Setting Up Remote Queues |

## Time Required to Install

Depending on your type of media and your system configuration, DQS installation takes:

- About 10 minutes, if you are installing using the VMSINSTAL utility
- About 5 minutes, if you are installing using the PCSI utility

# 12
# Do This First

## About This Chapter

This chapter describes what you need to do before installing the DEC Distributed Queuing Service for OpenVMS Systems (DQS) software on an OpenVMS AXP or OpenVMS VAX system. Table 12–1 summarizes the preinstallation tasks described in this chapter.

**Table 12–1 Preinstallation Task Summary**

| Task | Description |
|---|---|
| 1 | Inspect the distribution kit |
| 2 | Review the DQS installation |
| 3 | Prepare for VMScluster installation |
| 4 | Log in to a privileged account |
| 5 | Access the online release notes |
| 6 | Determine process account quotas |
| 7 | Determine disk space |
| 8 | Set up an alternate working device |
| 9 | Register your license |
| 10 | Perform system maintenance procedures |
| 11 | Perform first-time installation activities |
| 12 | Back up your system disk |

## Inspect the Distribution Kit

### DQS for OpenVMS AXP Kit

For OpenVMS AXP systems, the distribution kit contains:

- DQS software on a layered-product compact disc (CD-ROM).
- DQS documentation in text and PostScript format, provided with the software on the layered product CD-ROM.
  - Software Product Description (SPD)
  - DEC Distributed Queuing Service for OpenVMS Systems Cover Letter
  - *DEC Distributed Queuing Service for OpenVMS Systems Documentation Set*
  - Software Performance Report (SPR)
- DQS documentation in an online format, provided on the software layered products documentation CD-ROM. (Hardcopy documentation can be ordered separately.)

## DQS for OpenVMS VAX Kit

For OpenVMS VAX systems, the distribution kit contains:

- Bill of materials (BOM) and indented bills report (BIL)
- Media containing the DQS client and server software
  - TK50 streaming tape
  - 9-track 1600 BPI magnetic tape
- Hardcopy Documentation
  - Software Product Description (SPD)
  - DEC Distributed Queuing Service for OpenVMS Systems Cover Letter
  - *DEC Distributed Queuing Service for OpenVMS Systems Documentation Set*
  - Software Performance Report (SPR)

Contact your Digital representative if any of the components of your kit are missing.

The DQS client and server software and documentation are also available online on the Consolidated Distribution CD-ROM.

## Review DQS Installation

### Before Starting

Before you begin the installation, review the installation procedure.

- See Chapter 13 for the PCSI procedure.
- See Chapter 14 for the VMSINSTAL procedure.

### Options

You can select to set up your system as:

- A DQS server and client
- A DQS client

A DQS server system can accept print requests from DQS client systems. DQS client systems initiate remote print requests.

When you select the DQS server option, the installation procedure also provides client support.

## Prepare for VMScluster Installation

### Overview

The DQS software operates in VMScluster environments. Printing within the VMScluster system does not require the DQS software. However, printing between VMScluster systems requires that you install the DQS software on each VMScluster.

### VMScluster Considerations

When installing the DQS software on a VMScluster system, plan for the following:

- All DQS software files are placed in the clusterwide system directories by the installation procedure.
- The DQS software startup file SYS$STARTUP:DQS$STARTUP.COM must be executed during system startup on each node within the VMScluster system where you want DQS software to run.
- The DQS installation procedure defines the DQS network object on the cluster member that you are logged into when installing the DQS software. The definition of the network object enables the cluster member to:
  - Receive notification messages from the DQS$NOTIFIER indicating the completion status of a particular print request.
  - Act as a DQS server if you choose the DQS server option.

## Log In to a Privileged Account

**Privileges Needed**

To install DQS software, you must be logged in to an account that has SETPRV or at least the following privileges:

- CMKRNL
- WORLD
- SYSPRV
- SYSNAM
- OPER
- NETMBX
- TMPMBX

Digital recommends that you use the system manager account, SYSTEM. This account provides the privileges necessary to register your Product Authorization Key (PAK) and install the DQS software.

## Determine Account Privileges

Determine your account privileges as follows:

| Step | Action |
|---|---|
| 1 | Check the default privileges of the installation account. Log in and enter this DCL command:<br><br>`$ SHOW PROCESS/PRIVILEGES` Return<br><br>If the account lacks the SETPRV privilege, proceed to step 2. |
| 2 | Do either of the following:<br><br>• Ask your system manager to use AUTHORIZE to modify the default privileges of the account to include the SETPRV privilege.<br><br>• Run AUTHORIZE and make the changes yourself, if your account has the SYSPRV privilege:<br><br>`$ SET DEFAULT SYS$SYSTEM` Return<br>`$ RUN AUTHORIZE` Return<br>`UAF> MODIFY account-name/PRIVILEGES=(SETPRV)` Return<br>`UAF> EXIT` Return |
| 3 | To activate the change in privileges, log out and then log in again. |

## Access Online Release Notes

### Review the Notes

The DQS product provides online release notes. You should review the release notes because they describe new features and differences among multiple versions of the DQS software, as well as changes in the installation procedure.

### For More Information

For information on how to access the release notes before or after a PCSI installation, see Chapter 13.

For information on how to access the release notes during or after a VMSINSTAL installation, see Chapter 14.

## Determine Process Account Quotas

### Minimum Requirements

Table 12–2 lists the minimum quotas that the installation utility requires for the installation account.

**Table 12–2 Process Quotas for the Installation Account**

| | |
|---|---|
| ASTLM | 24 |
| BIOLM | 18 |
| BYTLM | 18,000 |
| DIOLM | 18 |
| ENQLM | 30 |
| FILLM | 20 |

### To Verify and Change Values

Use the OpenVMS Authorize utility to verify and change process quotas for the installation account in the user authorization file (SYSUAF.DAT). (Some sites may restrict the use of the OpenVMS Authorize utility to certain accounts or people.)

## Example

To verify and then change the BYTLM quota for the *account-name* installation account, you might enter the following command sequence:

| To . . . | Enter . . . |
|---|---|
| Invoke the Authorize utility | $ **RUN SYS$SYSTEM:AUTHORIZE** Return |
| Show the account quotas | UAF> **SHOW *account-name*** Return |
| Modify the BYTLM quota | UAF> **MODIFY *account-name* /BYTLM = 18000** Return |
| Exit the Authorize utility | UAF> **EXIT** Return |
| Log out | $ **LOGOUT** Return |
| Log in to activate the new quotas | Perform the login procedure for your system. |

## For More Information

For more information on modifying account quotas, see the description of the Authorize utility in the OpenVMS System Management documentation subkit.

## Determine Disk Space

### Requirements

DQS requirements for free disk storage space are different during installation than after installation. Table 12–3 summarizes the storage requirements. The numbers provided are approximate. Installation and use of the DQS software should not require more than the number of blocks listed in Table 12–3.

**Table 12–3 Disk Space Requirements**

| Operating System | Blocks During Installation | Blocks After Installation |
|---|---|---|
| OpenVMS AXP | 4500 | 2300 |
| OpenVMS VAX | 2700 | 1300 |

### Extra for Server Account Directory

In addition, the disk where the server account directory is located requires extra space for temporary storage of files queued for printing. Disk space for storing files to be printed does not have to be on the system disk. See Chapter 8 for information on how to move the server account directory to another device.

### Procedure

To determine the number of free disk blocks on the current system disk, enter the following command at the DCL prompt:

```
$ SHOW DEVICE SYS$SYSDEVICE Return
```

### For More Information

If you need to create free disk space, refer to the the *OpenVMS Upgrade and Installation Manual* or the *OpenVMS DECwindows Motif Installation Guide*.

If you need to move the server account to another disk, refer to Chapter 15, the section entitled Move the Server Account Directory to Another Device.

## Set Up an Alternate Working Device

### Reason

The DQS software installation requires fewer blocks of free disk space if you create a temporary working directory on an alternate device.

### Before You Begin

You have the option of creating the working directory on the alternate device before you begin the installation. For example, to create the working directory [INSTALL] on the alternate device DUA2, enter:

```
$ CREATE/DIRECTORY DUA2:[INSTALL] Return
```

### Options

Consider the following when deciding whether to specify a temporary working directory when installing the DQS software.

| If you . . . | Then the installation utility . . . |
|---|---|
| Specify a directory | Creates a working subdirectory for the product and then deletes the working subdirectory after the installation completes. |
| Do not specify a directory | Creates the working subdirectory on the specified device with a directory specification of [000000.*facvvu*], where *facvvu* represents the facility name and version number of the product. The installation utility deletes the working subdirectory after the installation completes. |
| Specify a nonexistent directory | Creates the directory that you specify and the working subdirectory. After the installation completes, the installation utility deletes the working subdirectory but does not delete the directory. |

## Examples

You can enter the following command to install the DQS software using an alternate working directory:

- With PCSI utility

  ```
  $ PRODUCT INSTALL DQS/VERSION=1.3 /WORK=DUA2:[INSTALL]  Return
  ```

- With VMSINSTAL utility

  ```
  $ @SYS$UPDATE:VMSINSTAL DQS013 MUA0: OPTIONS AWD=DUA2:[INSTALL] Return
  ```

| Where . . . | Is . . . |
|---|---|
| DUA2: | The alternate working device name, for this example. |
| [INSTALL] | For this example, the name of the directory on the alternate working device in which the installation utility creates the product subdirectory. |

## Using a DFS-mounted Disk

You can specify a DFS-mounted disk as your alternate working device. However, the DFS client (installation procedure) is unable to determine available disk space on the DFS-mounted disk and incorrectly reports a total of -1 free blocks.

Before using a DFS-mounted disk, determine its free disk space. Log in to the system on which the disk is locally mounted and enter the command:

```
$ SHOW DEVICE  Return
```

Refer to Table 12–3 for the amount of free disk space that must be on the DFS-mounted disk to complete the DQS installation.

## Register Your License

### Requirement

The DQS product supports the License Management Facility (LMF) provided by the OpenVMS systems. Before you can run the DQS IVP and start the DQS software, LMF requires that you register a License Product Authorization Key (License PAK) for the product. License PAK information is provided with the DQS product. Register and load this information into your system's license database using the LMF utility.

### Procedure

To register your license:

| Step | Action |
|---|---|
| 1 | Log in to the system manager's account, SYSTEM. |
| 2 | You then have a choice of two ways to perform the registration:<br>• Invoke the SYS$UPDATE:VMSLICENSE.COM procedure. When it prompts you for information, respond with the data from your License PAK.<br>`$ @SYS$UPDATE:VMSLICENSE.COM` Return<br>• At the DCL prompt, enter the LICENSE REGISTER command with the appropriate qualifiers that correspond to the License PAK information.<br>`$ LICENSE REGISTER` Return<br>Add information carefully. The utility may return a checksum error message if you omit or incorrectly enter any license data. Check characters typed on each line, not just the checksum string. |

### For More Information

For complete information on using the LMF utility, see the *OpenVMS License Management Utility Manual*.

## Perform System Maintenance Procedures

### Overview

This section explains how to use the System Generation utility (SYSGEN) and the AUTOGEN utility to prepare your system for installation of the DQS software. AUTOGEN uses system configuration data to automatically set system parameters.

### Required Values

Table 12–4 lists the minimum required system parameters for the installation. These values do not represent the number required to run your system and other software.

**Table 12–4 Minimum Required System Parameter Values**

| System Parameters | OpenVMS AXP Values | OpenVMS VAX Values |
|---|---|---|
| GBLPAGES | 1500 | 600 |
| GBLSECTIONS | 10 | 5 |

If your your system does not have the necessary global pages and global section SYSGEN parameters, the installation fails.

### Check Global Pages and Sections Parameters

To install and run DQS software, you must have sufficient free global pages and global sections. First find out how many free global pages and sections you have on your system. Then use AUTOGEN if you need to increase the global pages and global sections system parameters.

To check the values of the global pages and global sections parameters, use the DCL WRITE command with the F$GETSYI lexical function as follows:

```
$ WRITE SYS$OUTPUT F$GETSYI("CONTIG_GBLPAGES")
7578

$ WRITE SYS$OUTPUT F$GETSYI("FREE_GBLSECTS")
143
```

If the SYSGEN values displayed by the system are greater than the values listed in Table 12–4, you do not need to increase them. If the values of the global pages and global sections parameters displayed by the system are less than the values listed in Table 12–4, you must increase their values. Proceed to the next section.

## Change Global Pages and Sections Parameters

Use the AUTOGEN command procedure to change system parameters. AUTOGEN automatically adjusts values for parameters that are associated with the values you reset manually.

| Step | Action |
|---|---|
| 1 | Edit the SYS$SYSTEM:MODPARAMS.DAT file. |
| 2 | Modify incremental parameters, such as these, by typing ADD_*parameter* in the file, where *parameter* is GBLPAGES or GBLSECTIONS.<br>For example, to increase the global pages parameter by 2000, add this line in the MODPARAMS.DAT file:<br>**ADD_GBLPAGES = 2000** |
| 3 | After making your changes, exit from the editor. |
| 4 | Execute the AUTOGEN procedure to recalculate your system parameters. The following command recalculates your system parameters and reboots the system:<br>$ **@SYS$UPDATE:AUTOGEN GETDATA REBOOT** Return<br>When you specify REBOOT, AUTOGEN performs an automatic system shutdown and then reboots the system. Any users logged on to the system are immediately disconnected during the shutdown. The automatic reboot puts the new parameter values into effect. |

The AUTOGEN utility automatically adjusts some of the SYSGEN parameters based on the consumption of resources since the last reboot. If you do not want to take advantage of this automatic adjustment, include the NOFEEDBACK qualifier on the AUTOGEN command line. For example,

```
$ @SYS$UPDATE:AUTOGEN GETDATA REBOOT NOFEEDBACK Return
```

## For More Information

For more information about using the AUTOGEN utility, see the *OpenVMS System Manager's Utilities Reference Manual*.

## Perform First-Time Installation Activities

### PCSI Utility

These activities apply if you are installing the DQS software with the PCSI utility.

### Register Products in PCSI Database

The PCSI utility checks for software products required for installation and use of the DQS software. These products need to be registered in the PCSI product database.

For example, DQS installation requires either of the following:

- OpenVMS VAX operating system, Version 5.5-2 or higher
- OpenVMS AXP operating system, Version 6.1 or higher

To determine if the required operating system software is registered and to register the product, if required, do the following:

### Using the Motif Interface

Perform the following steps to check the operating system software on Motif:

| Step | Action |
|---|---|
| 1 | Enter the PRODUCT command at the DCL prompt.<br>$ **PRODUCT** Return<br>The PCSI Main menu displays the Show Product List window by default. |
| 2 | Scroll through the list of installed products. Search for DEC AXPVMS VMS V6.1 or higher or DEC VAXVMS VMS V5.5-2 or higher. If your operating system (or other required software) is not in the list, you must register the product. See Step 3. |
| 3 | Pull down the Mode menu and select the REGISTER PRODUCT command. Enter the product name and version of the product to register. |

## Using the DCL Command Line Interface

Perform the following steps to check the operating system software from DCL:

| Step | Action |
|---|---|
| 1 | Enter the PRODUCT SHOW PRODUCT *product-name* command at the DCL prompt.<br><br>$ **PRODUCT SHOW PRODUCT *VMS** Return |
| 2 | Search the list for DEC AXPVMS VMS V6.1 or higher or DEC VAXVMS VMS V5.5-2 or higher.<br><br>If the required operating system (or other required software) is not in the SHOW PRODUCT list, then you need to register the product. See Step 3. |
| 3 | Enter the PRODUCT REGISTER PRODUCT *product-name* command at the DCL prompt. For example, to register OpenVMS VAX Version 6.0, enter:<br><br>$ **PRODUCT REGISTER PRODUCT -**<br>_$ **VMS/VERSION=6.0 /SOURCE=SYS$UPDATE:** Return |

## Back Up Your System Disk

**Overview**

Near the beginning of the installation, the installation utility asks if you completed the pre-installation tasks. Digital recommends that you do a system disk backup before installing any software. Use the backup procedures that are established at your site.

**For More Information**

For details on performing a system disk backup, see the section on the Backup utility in the OpenVMS System Management documentation subkit.

**To Proceed**

Before proceeding with the installation of the software, refer to the flowchart in Part III.

# 13
# Using the PCSI Utility

## About This Chapter

This chapter provides information about the POLYCENTER Software Installation utility (PCSI) and the procedure for installing the DEC Distributed Queuing Service for OpenVMS Systems software. Table 13–1 lists the tasks described in this chapter.

**Table 13–1 PCSI Installation Task Summary**

| Task | Description |
|---|---|
| 1 | Accessing release notes |
| 2 | Starting the installation: DCL interface |
| 3 | Obtaining DCL Help |
| 4 | Determining installation defaults: DCL interface |
| 5 | Verifying preinstallation tasks: DCL interface |
| 6 | Stopping and restarting: DCL interface |
| 7 | Responding to questions: DCL interface |
| 8 | Determining installation progress: DCL interface |
| 9 | Starting the installation: Motif interface |
| 10 | Obtaining Motif Help |
| 11 | Selecting product options: Motif interface |
| 12 | Stopping and restarting: Motif interface |
| 13 | Recovering from errors |

## Accessing Release Notes

### Procedure

To extract a copy of the release notes for the DQS product to an online disk, enter:

```
$ PRODUCT EXTRACT RELEASE_NOTES DQS /VERSION=Vn.n /FILE=output-file-name Return
```

Where:

- The *Vn.n* represents the version number. For example, V1.3 stands for Version 1.3.
- The *output-file-name* is the name you provide for the release notes file.

By default the PCSI utility searches the location defined by PCSI$SOURCE. If the software kit is in another location, this location must be specified with the /SOURCE qualifier on the command line.

```
$ PRODUCT EXTRACT RELEASE_NOTES DQS /VERSION=Vn.n /SOURCE=location /FILE=output-file-name Return
```

After extracting the release notes to a file, you can print or view the file. You do not need to install the software before you extract the release notes.

### After the Installation

After you install DQS software, type or print the release notes file from:

SYS$HELP:DQS*nnn*.RELEASE_NOTES

The *nnn* represents the version number. For example, 013 stands for Version 1.3.

## Starting the Installation: DCL Interface

### Command To Use

To start the installation of the DQS software, enter the PRODUCT INSTALL command at the DCL prompt.

If this is the first time you are installing the DQS software, include the /HELP qualifier on the command line. /HELP provides detailed explanations about the product and product options. When you perform subsequent installations or upgrades, you can choose to bypass the explanations to save time.

The following command installs Version 1.3 of the DQS software with detailed explanations:

```
$ PRODUCT INSTALL DQS/VERSION=1.3 /HELP Return
```

The installation utility displays messages like the following and asks if you want to continue:

```
$ PRODUCT INSTALL DQS/VERSION=1.3 /HELP Return
The following product has been selected:
DEC [AXP|VAX]VMS DQS Vn.n     [Available]

Do you want to continue [YES] Return

*** DEC [AXP][VAX]VMS DQS Vn.n: DEC [AXP][VAX]VMS DQS Vn.n for OpenVMS

    Requires from nnn to nnn disk blocks depending upon configuration.

    The DEC Distributed Queuing Service for OpenVMS Systems uses the
    DECnet networking system to extend the standard OpenVMS queue
    system to enable users to print jobs connected to other systems
    than their own, and to show the status of, cancel, and change the
    specifications of their jobs on those systems.

    Copyright © 1994 Digital Equipment Corporation

    All Rights Reserved. Unpublished rights reserved under the copyright
    laws of the United States.
    .
    .
    .
```

## Specifying Kit Location

You can specify the location (source) of the DQS kit in one of two ways:

- If you install your product kits from a location defined by the PCSI$SOURCE logical name, make sure that the DQS kit is in this location. By default, the PCSI utility searches the location defined by the PCSI$SOURCE logical name.

  For example, this command defines DISK$:[LOCAL_KITS] as the device and directory that contains products to be installed:

  ```
  $ DEFINE PCSI$SOURCE DISK$:[LOCAL_KITS] Return
  ```

  For this example, the DQS kit should reside in DISK$:[LOCAL_KITS] to use the default location. You do not need to include the location on the command line:

  ```
  $ PRODUCT INSTALL DQS/VERSION=n.n Return
  ```

- If the DQS kit is not in the default location, include the /SOURCE qualifier on the PRODUCT INSTALL command line:

  ```
  $ PRODUCT INSTALL/SOURCE=MUA0: DQS/VERSION=1.3 Return
  ```

  For this example, MYDISK$:[NEW_PRODUCTS] contains the DQS kit for installation.

## Command Line Features

You can also request features like purging files and running the installation verification procedure (IVP) by adding appropriate qualifiers to the PRODUCT INSTALL command line. Table 13–2 lists features you can control with command qualifiers.

**Table 13–2 Features You Can Request When Starting an Installation**

| Feature | Qualifier |
|---|---|
| Display full descriptions of all product installation options and information | /HELP |
| Display log messages on your terminal | /LOG |
| Save previous versions of product files during an upgrade | /KEEP |
| Include a remark in the product database | /REMARK |
| Specify where the distribution kit is located | /SOURCE=*dev:[dir]* |
| Choose not to run the installation verification procedure (IVP) | /NOTEST |
| Specify a work area for temporary files | /WORK=*device* |

## For More Information

For more detailed information on all features that you can request when starting an installation, refer to DCL help for the PRODUCT INSTALL command.

## Obtaining DCL Help

### DCL HELP Options

The following DCL help options are available:

| Option | Action |
|---|---|
| 1 | Include the /HELP qualifier on the PRODUCT INSTALL command line. |
| 2 | During the installation, press the Help key or enter a question mark in response to a question from the PCSI utility. |
| 3 | Type the DCL command HELP PRODUCT, followed by the name of a PCSI command (for example, HELP PRODUCT INSTALL). |

## Determining Installation Defaults: DCL Interface

### Overview

During installation, the PCSI utility allows you to accept or change the defaults. Default values are enclosed in brackets ([ ]).

### DQS Default Values

The DQS software offers you the option to install or not install the server software on the system. The client software is always installed. The initial installation default is to install client software only.

```
The DQS client software will be installed automatically on this node

Do you want to install the DQS server software on this node? [NO]
```

If you are installing the DQS software on a system that has a printer that you want to make available to clients on other systems, then you enter YES and press the Return key.

This action changes the default value for this question to YES. The next time you install the DQS software, the question is:

```
The DQS client software will be installed automatically on this node

Do you want to install the DQS server software on this node? [YES]
```

## Verifying Preinstallation Tasks: DCL Interface

### Verify Tasks

During the installation procedure, you are asked if you completed the preinstallation tasks for the DQS software.

```
You need to complete the pre-installation tasks to install the DQS software.
Do you want to continue [YES]
```

### Tasks Checked by PCSI Utility

The PCSI utility might verify that any or all of the following is true, depending on your product:

- You are logged in to a privileged account.

  It is good practice to install software from the system manager's account with your default device and directory set to SYS$UPDATE.

- Disk space for installation is adequate.

  If sufficient disk space is not found, the installation procedure provides the following options:

  - You can restart the installation and specify another disk to use as an alternate working device for the temporary working directory. See Chapter 12 for information about how to use an alternate working device.
  - You can abort the installation and then create more disk space on the system disk. The *OpenVMS Upgrade and Installation Manual* and *OpenVMS DECwindows Motif Installation Guide* describe how to use tailoring utilities so that you can delete files and create more disk space.

  If you lack sufficient free disk space for installation or have no other disk to use as an alternate working device, the installation procedure aborts.

- OpenVMS AXP Version 6.1 or higher is installed.
- OpenVMS VAX Version 5.5-2 or higher is installed.

When the PCSI utility detects a problem in any of these areas, it asks if you want to terminate the installation. If you want to stop the installation, press the Return key. If you want to continue, enter NO.

### Not Checked by PCSI Utilty

The PCSI utility does not check process quotas or system parameters. You need to make sure that these values are sufficient.

You must also make sure that the DQS software is running if you are upgrading from a version of the software before Version 1.3 and you are installing server software.

## Stopping and Restarting: DCL Interface

To stop and restart the installation:

| Step | Action |
|---|---|
| 1 | To stop the procedure at any time press Ctrl/Y or if you are asked if you want to continue, answer NO. |
| 2 | To retry the installation procedure, enter the PRODUCT INSTALL command as described in the section titled Starting the Installation: DCL Interface. |

## Responding to Questions: DCL Interface

### Valid Responses

The following are valid responses to questions asked during an installation:

- Press the Help key or enter a question mark to request additional information provided by the product.
- Enter your own value and press the Return key.
- Press the Return key to accept a default value.

### Questions You Are Asked

While installing the DQS software with the DCL interface, you might be asked the following questions. (Refer to Appendix F to see sample installation dialogs.)

- If you want to continue:
  - Verify the product(s) being installed

```
The following product has been selected:
DEC VAXVMS DQS V1.3     [Available]

Do you want to continue? [YES]
```

  - Verify that you completed all preinstallation tasks

```
You need to complete the pre-installation tasks to install the DQS software.
Do you want to continue? [YES]
```

- If you want to accept the default values for the product:

```
Do you want all the default values for this product? [YES]
```

- If you want to install the DQS server software (that is, does this node have an attached printer that you want to set up to accept print jobs from users on nodes in other systems):

```
Do you want to install the DQS server software on this node? [NO]
```

  Client software is automatically installed.

- If you want to view your choices:

```
Do you want to view the values? [NO]
```

If you want to view your answers to the quesions asked during the installation, answer YES and press the Return key.

```
Do you want to view the values [NO] YES Return
DEC VAXVMS DQS V1.3: DEC VAXVMS DQS V1.3 for OpenVMS     [Available]
    DEC VAXVMS VMS V6.0     [Available]
    Do you want to install the DQS server software on this node: YES

    Are you satisfied with the values [YES]
```

If you are satisfied with your selections, answer YES and press the Return key. The installation of the DQS software continues.

If you are not satisfied with your selections, answer NO to this question. You now have the opportunity to make new selections or to end the installation.

```
    Are you satisfied with the values [YES] NO Return
    Do you want to re-enter the values [YES] YES Return
        *** DEC VAXVMS VMS V6.0     [Available]
    Did you complete the pre-installation tasks for DQS
    Do you want to continue [YES]
   .
   .
   .
```

## Determining Installation Progress: DCL Interface

### Progress

As an installation procedure progresses, the system displays a percentage to indicate how much of the installation is done. For example:

```
Portion Done: 10%...20%...30%...40%...50%...70%...80%...90%...100%
The following product has been installed:
DEC VAXVMS DQS V1.3
   .
   .
   .
```

If you started the installation using the /LOG qualifier, the system displays details of the installation.

## Starting the Installation: Motif Interface

### Command to Use

To start the installation of the DQS software, enter the SET DISPLAY and PRODUCT commands at the DCL prompt.

```
$ SET DISPLAY/CREATE/NODE=mynode
$ PRODUCT Return
```

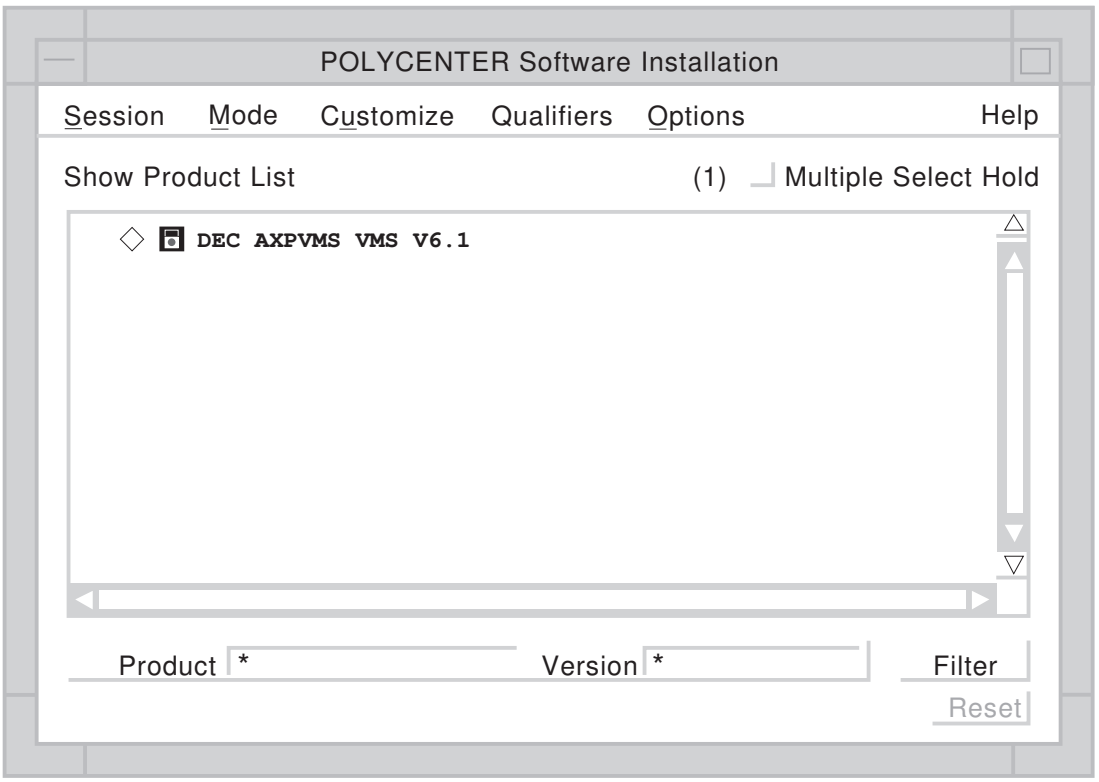
### PCSI Main Window

The PCSI main window opens (see Figure 13–1).

The items in this window include the following:

| This Item... | Displays... |
|---|---|
| Title bar | The name of the window |
| Menu bar | The menu names for the PCSI utility |
| The Product List box | The list of software products available for installation |
| The Product and Version fields | An asterisk (*) wildcard, which indicates all versions of all products are displayed in the Product List box |

**Figure 13–1 PCSI Main Window**




MLO-012661

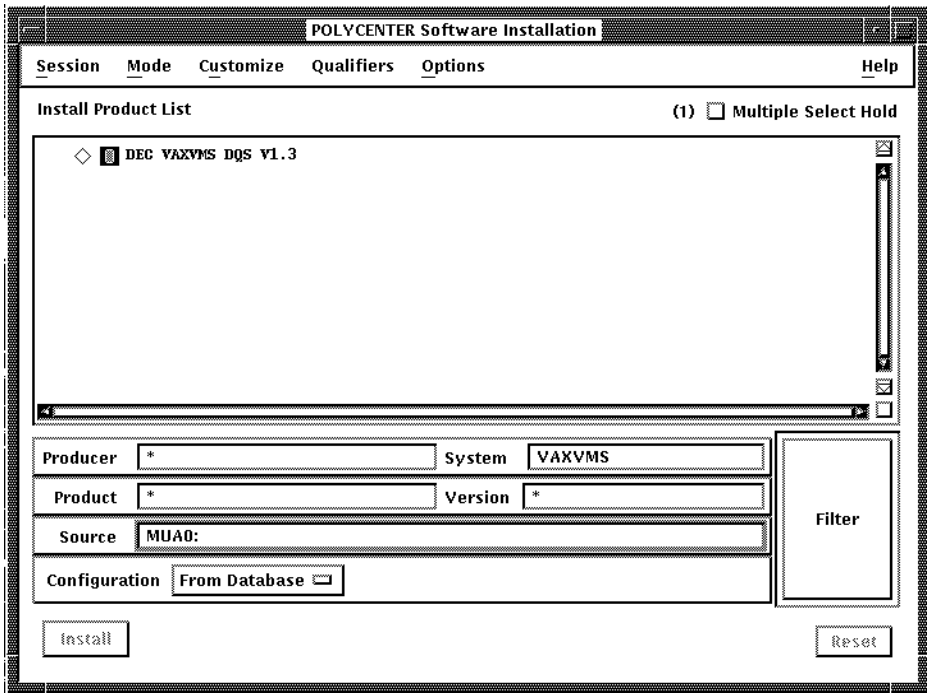
## Procedure to Install

Use the following procedure to install the DQS software using the Motif interface (see Figure 13–2):

1. From the Main Menu, pull down the **Mode** menu and click on **Installation** The software installation window displays.
2. Double-click on the button to the left of DEC AXPVMS DQS V*n.n* (for OpenVMS AXP systems) or DEC VAXVMS DQS V*n.n* (for OpenVMS VAX systems). (You may have to scroll to find the product name.)

   The Selected Product list box then displays either DEC AXPVMS DQS V*n.n* or DEC VAXVMS DQS V*n.n* to be installed.
3. If the DQS distribution kit is located in a place other than the default (that is, other than a directory pointed to by the PCSI$SOURCE logical name), enter the device and directory in the Source field and click on the Filter button.
4. Click on the **Install** box at the bottom of the screen to install the DQS software.
5. When the installation is complete, you can return to the Install Product List screen by clicking on the **Start New Operation** box in the Work in Progress window.

**Figure 13–2 PCSI Installation Window**



## Obtaining Motif Help

### Motif Help Options

The following Motif help options are available:

| Option | Action |
|---|---|
| 1 | Choose On C ontext from the H elp menu.<br>For a description of objects and labels in the main window, move the resulting ? to a screen object or label and click MB1. |
| 2 | Choose O n Window from the H elp menu.<br>For a description of how fonts are used to convey different types of information, double click on the Display Conventions topic. Double click on other help topics for information on how to perform a specific task. |
| 3 | Double click on option names that are preceded by a right arrow ⟹ in the Selected Products to be Installed list box.<br>For a full description of suboptions (if provided), continue clicking on the resulting right arrow ⟹ that precedes each suboption. |
| 4 | Choose On H elp from the H elp menu to learn more about using the online help system. |

### Motif help topics

Motif help describes how to get started performing a task from the Motif interface.

| Step | Help Topic |
|---|---|
| 1 | Select the task. |
| 2 | Select one or more products. |
| 3 | Select product options. |
| 4 | Start the task. |

### For More Information

For information about how you can customize menus by adding and removing menu items, refer to the Customize help topic for each menu.

## Selecting Product Options: Motif Interface

### How to Select

After you select the product, you can select and display information about product options by clicking on icons or by choosing items from the Options menu. You can also double-click on the options name to get a full description of the option.

### Available Options

The DQS product offers these options:

- Install the client software only
- Install the client and server software

The client software is always installed, but does not need to be configured for a server node. Choose to install the server software if your node has a printer that you want to make available to clients on other systems in the network.

## Stopping and Restarting: Motif Interface

### Procedure

To stop and restart the installation:

| Step | Action |
|---|---|
| 1 | To stop the procedure at any time press Ctrl/Y or click on the Cancel button. |
| 2 | To retry the installation procedure, select Installation from the PCSI utility Mode pull-down menu as described in the section titled Starting the Installation: Motif Interface. |

## Recovering from Errors

### How to Recover

This section describes how to recover from errors during product installation.

If the installation procedure fails for any reason, the following message is displayed:

```
%PCSI-E-OPFAILED, operation failed
  Terminating is strongly recommended.
  Do you want to terminate? [YES]
```

An error during the installation can occur if one or more of the following conditions exist:

- The operating system version is incorrect.
- Disk space is not sufficient.
- The OpenVMS Help library is currently in use.
- System parameters are not set to the correct values.

If you receive any error message beginning with %PCSI-, refer to *OpenVMS System Messages and Recovery Procedures Reference Manual* for an explanation and possible solution to the problem.

Most DEC Distributed Queuing Service for OpenVMS Systems software messages have the same format as standard OpenVMS system messages: %DQS-E-ident, text.

If the installation fails, you must restart the installation procedure. If the installation fails due to an IVP failure, contact a Digital Services representative.

# 14

# Using the VMSINSTAL Utility

## About This Chapter

This chapter provides information about the VMSINSTAL installation utility and the procedure for installing the DEC Distributed Queuing Service for OpenVMS Systems software. Table 14–1 lists the tasks described in this chapter.

**Table 14–1 VMSINSTAL Installation Task Summary**

| Task | Description |
|---|---|
| 1 | Accessing release notes |
| 2 | Starting the installation procedure |
| 3 | Stopping and restarting the installation procedure |
| 4 | Responding to installation questions |
| 5 | Following installation progress |

## Accessing Release Notes with VMSINSTAL

### During the Installation

You can read or print the DQS release notes before installating the DQS software. To do this, you must include OPTIONS N on the command line when you invoke the VMSINSTAL procedure:

```
$ @SYS$UPDATE:VMSINSTAL DQS013 MUA0: OPTIONS N Return
```

OPTIONS N allows you to select release notes options:

```
Release notes included with this kit are always copied to SYS$HELP.

   Additional Release Notes Options:

        1.  Display release notes
        2.  Print release notes
        3.  Both 1 and 2
        4.  None of the above

* Select option [3]: Return
```

| Enter . . . | VMSINSTAL . . . |
|---|---|
| 1 or 3 | Displays the release notes at your terminal. |
| 2 or 3 | Prompts you for a print queue name for printing the release notes, as follows:<br><br>`Queue name [SYS$PRINT]: Return`<br><br>Enter a queue name or press the Return key. Remember, to print the release notes, you must have a printer attached to the system on which you are installing the DQS software. (You cannot use the DQS software to print the release notes until after the software is installed.) VMSINSTAL queues a copy of the release notes to print and displays a message such as the following:<br><br>`Job DQS013 (queue queue, entry nnn) started on device` |
| 4 | Copies the product release notes to SYS$HELP. Release notes are neither displayed or printed. |

Digital recommends that you read the release notes before installing the DQS software for the first time.

After you have answered the questions regarding the release notes, this message is displayed:

```
* Do you want to continue the installation [N]? YES Return
```

Enter YES or NO, as appropriate. If you enter YES and press the Return key, the procedure continues with this message:

```
%VMSINSTAL-I-RELMOVED, The products release notes have been successfully moved to SYS$HELP.
```

## After the Installation

After the installation procedure completes, release notes are available from the following location:

SYS$HELP:DQS*nnn*.RELEASE_NOTES

The *nnn* represents the version number. For example, 013 stands for Verison 1.3.

## Starting the Installation Procedure

### Before You Start

Before you start the installation of the DQS software, log in to the system manager account. The system manager account has the necessary privileges to run the VMSINSTAL utility.

### Logging Your Installation Procedure

If you want to save a copy of your VMSINSTAL session for future reference, you can enter the SET HOST 0 /LOG command at the system prompt before starting the installation of the software. This creates the log file SETHOST.LOG in your current default directory.

### Command Line

To start the installation of the DQS software, enter the following command at the DCL prompt:

```
$ @SYS$UPDATE:VMSINSTAL DQSnnn dev: [OPTIONS N,AWD=dev:[dir]] Return
```

@SYS$UPDATE:VMSINSTAL invokes the VMSINSTAL procedure. In addition, you can supply the following information in the command line:

| | |
|---|---|
| DQS*nnn* | Refers to the DEC Distributed Queuing Service for OpenVMS Systems saveset. The *nnn* represents the version number. For example, 013, stands for Version 1.3. |
| *dev:* | Is the device where your distribution media is mounted. |
| OPTIONS N | Causes VMSINSTAL to ask you about displaying or printing the release notes.<br>Digital recommends that you review the release notes before you proceed with the installation of the software in case they contain information about the installation.<br>If you are restarting the installation and have already reviewed the release notes, you do not need to specify OPTIONS N.<br>Release notes are installed on the system whether or not you specify OPTIONS N. |

| | |
|---|---|
| OPTIONS AWD=*dev:[dir]* | Lets you specify an alternate working directory for the temporary working directory used by VMSINSTAL during installation. With this option, fewer free blocks are required on the system disk.<br><br>The *dev:[dir]* names the alternate working device and directory under which VMSINSTAL creates its working directory. Specifying a directory name is optional.<br><br>If you do not specify this option, VMSINSTAL creates the following temporary working directory:<br><br>SYS$SPECIFIC:[SYSUPD.DQS$*nnn*] |

In response to the VMSINSTAL command, a banner like the following may be displayed on the screen:

```
        OpenVMS VAX Software Product Installation Procedure Vx.x

It is 11-JUN-1994 at 15:47.
Enter a question mark (?) at any time for help.
```

## Stopping and Restarting the VMSINSTAL Procedure

### Procedure

To stop the installation procedure at any time, press Ctrl/Y. The installation procedure deletes all files it has created up to that point and exits.

If you want to retry the installation procedure after you press Ctrl/Y, you must invoke VMSINSTAL again, as shown in the section titled Starting the Installation Procedure.

## Responding to Installation Questions

### Valid Responses

The VMSINSTAL utility asks a series of questions to which you must provide answers. The following are valid responses to questions asked during an installation:

- Press the Return key to accept a default value.

  In most cases, a default response appears in square brackets ([]) at the end of the question or prompt.

- Enter your own value and press the Return key.
- Enter a question mark (?) to request additional information provided by the product.

### Questions You Are Asked

During the installation of the DQS software with the VMSINSTAL utility, you are asked the following questions:

- If you want to continue:
  - After warnings about the environment:

```
%VMSINSTAL-W-NOTSYSTEM, You are not logged in to the SYSTEM account.
%VMSINSTAL-W-ACTIVE, The following processes are still active:
        DECW$SERVER_0
        _FTA13:
        VUE$DAVIES_3
        DECW$TE_005C
* Do you want to continue anyway [NO]?
```

    This display is informational. The default is NO. If you answer NO or only press the Return key, the installation terminates and you are returned to the system command level. If you answer YES and press the Return key, the installation continues.

  - After selecting a release notes option:

```
* Do you want to continue the installation [NO]?
```

    Enter YES or NO, as appropriate. If you enter YES and press the Return key, the procedure continues with this message:

```
%VMSINSTAL-I-RELMOVED, The products release notes have been successfully moved to SYS$HELP.
```

- If you are satisfied with the backup of your system disk:

  ```
  * Are you satisfied with the backup of your system disk [YES]?
  ```

  You are reminded that you should back up your system disk before you install the software.

- Where the distribution volumes are mounted:

  ```
  * Where will the distribution volumes be mounted:
  ```

  If you did not indicate the device where the distribution volumes are mounted when you invoked VMSINSTAL, you are asked this question. Enter the name of the device where you loaded the media.

- To select a release notes option, if you entered OPTIONS N on the command line:

  ```
  Release notes included with this kit are always copied to SYS$HELP.

     Additional Release Notes Options:

          1.  Display release notes
          2.  Print release notes
          3.  Both 1 and 2
          4.  None of the above

  * Select option [3]:
  ```

  Digital recommends that you read the release notes before installing the software for the first time.

- To enter a print queue name for printing the release notes, if you selected release notes options 2 or 3:

  ```
  Queue name [SYS$PRINT]:
  ```

  Enter a queue name or press the Return key. Remember, to print the release notes, you must have a printer attached to the system on which you are installing the DQS software. (You cannot use the DQS software to print the release notes until after the software is installed.)

- If you want to install the server software:

```
        The DQS client will be installed automatically on this node.

* Do you want to install the DQS server software on this node [NO]?
```

  The DQS software offers you the option to install or not install the server software on the system. The client software is always installed. The installation default is to install client software only.

  If you are installing the DQS software on a system that has a printer that you want to make available to clients on other systems, then you enter YES and press the Return key.

- If you want to proceed (after selecting to install the server software):

```
                    *** PLEASE NOTICE ***

        If you are upgrading DQS server software from an earlier version
        to DQS V1.3, the earlier version of the DQS server software must
        be running. This allows the DQS V1.3 installation procedure to read
        the existing DQS server configuration information and create a
        new DQS V1.3 server configuration file. Otherwise, after the
        installation, you must manually update your DQS V1.3 server
        configuration file.

        The directory and name of the DQS V1.3 configuration file is:

                  SYS$MANAGER:DQS$SERVER_CONFIG.TXT

        Digital recommends that you review the new DQS V1.3 server
        configuration file after this installation. Refer to the section
        on server management in the DQS System Manager's Guide for more
        information.

        The DQS client configuration is not automatically updated. Refer
        to the section on client management in the DQS System Manager's
        Guide for more information.

* Do you want to proceed [YES]?
```

- If you want to run the Installation Verification Procedure (IVP):

```
* Do you want to run the IVP after the installation [YES]?
```

  Digital recommends that you answer YES to this question.

- If you want to purge files replaced during the installation process:

```
* Do you want to purge files replaced by this installation [YES]?
```

If you want to retain the old copies, answer NO and press the Return key. Otherwise, answer YES or press the Return key. (Remember, once the files are purged, they are erased and cannot be accessed.)

- If you have loaded and registered an authorization key:

```
        Product:      DQS
        Producer:     DEC
        Version:      1.3
        Release Date: 07-MAR-1994

* Does this product have an authorization key registered and loaded?
```

The DQS product supports the License Management Facility (LMF). Before starting the DQS software on an OpenVMS system, LMF requires that you register a License Product Authorization Key (License PAK) for the DQS product.

You must load the DQS License PAK on each node in an OpenVMS cluster. For information on registering and loading your License PAK, see the *OpenVMS License Management Utility Manual*.

## Getting Help on VMSINSTAL Questions

To get help on any of the VMSINSTAL questions, enter a question mark ( ? ) in response to the question.

## Following Installation Progress

### No More Questions

After validating that you have loaded the DQS License PAK, VMSINSTAL asks no more questions.

```
No further questions.
```

### Status Messages

The DQS installation procedure then creates the DQS$SERVER account and directory, installs the client software, installs the server software if selected, and creates the DQS network object. VMSINSTAL displays informational messages about the status of the installation, like the following:

```
%VMSINSTAL-I-RESTORE, Restoring product saveset B...
%VMSINSTAL-I-ACCOUNT, This installation creates an ACCOUNT named DQS$SERVER.
%UAF-I-ADDMSG, user record successfully added
%UAF-I-RDBADDMSGU, identifier DQS$SERVER value: [000300,000311] added to rights
database

        DQS$SERVER account is created.

%VMSINSTAL-I-ACCOUNT, This installation updates an ACCOUNT named DQS$SERVER.
%UAF-I-MDFYMSG, user record(s) updated
%VMSINSTAL-I-SYSDIR, This product creates system disk directory
                                                SYS$COMMON:[DQS$SERVER].

        [DQS$SERVER] directory is created in SYS$COMMON: area.

        Installing DQS SERVER software . . .

        Installing DQS CLIENT software . . .

        DQS network object is created.
```

## Informational Messages

VMSINSTAL displays the following informational messages that concern system parameter requirements and postinstallation procedures:

```
      ****************************************************************
      *           Instructions for completing DQS installation       *
      ****************************************************************

        o Please follow the post installation instructions which have
          been outlined in the DQS V1.3 Installation Guide.

        o Please be sure to edit your OpenVMS System Startup Procedure
          and add the following line to invoke the DQS Startup Procedure:

                $ @SYS$STARTUP:DQS$STARTUP.COM

        o Once you have completed the DQS post installation procedures
          you can invoke the DEC DQS installation verification procedure
          to test your customizations. To invoke this file start DQS
          and type:

                $ @SYS$TEST:DQS$IVP.COM

        o This installation has updated the DCL Tables. Therefore
          users must log out and then back in, in order to use the
          QSHOW, QDELETE, and QSET commands for displaying, deleting,
          and modifying jobs on the remote server queues.

        o Please read the DQS V1.3 release notes located in the file
          SYS$HELP:DQS013.RELEASE_NOTES .

%VMSINSTAL-I-MOVEFILES, Files will now be moved to their target directories...
```

## Verification Messages

If you indicated that you want to run the IVP as part of the installation, the following messages are displayed; otherwise, this step is skipped:

```
Beginning of the DQS V1.3 Installation Verification Procedure.

  Copyright Digital Equipment Corporation 1994. All rights reserved.

  Restricted Rights: Use, duplication, or disclosure by the U.S. Government
  is subject to restrictions as set forth in subparagraph (c)(1)(ii) of
  DFARS 252.227-7013, or in FAR 52.227-19, or in FAR 52.227-14 Alt. III, as
  applicable.

  This software is proprietary to and embodies the confidential technology
  of Digital Equipment Corporation. Possession, use, or copying of this
  software and media is authorized only pursuant to a valid written license
  from Digital or an authorized sublicensor.

%DQS-I-QUESUCCESS IVP test successful
End of the DEC DQS for OpenVMS V1.3 Installation Verification Procedure
```

The IVP checks to see if files are installed in the correct locations. Then, the IVP displays the following message:

```
%DQS-I-QUESUCCESS IVP test successful
```

## Procedure Ending Messages

Once installation is completed, this message appears:

```
Installation of DQS Vn.n completed at hh:mm

VMSINSTAL procedure done at hh:mm
```

The message from VMSINSTAL means that the installation and verification (if selected) of the DQS software has completed successfully. If VMSINSTAL reports a failure, review the installation prerequisites.

You can run the IVP at any time to check the DQS software.

## For More Information

Refer to Chapter 15 for more information on postinstallation procedures, including:

- Editing the system startup file to include DQS startup command
- Running the IVP independent of the installation
- Setting up DQS software to run in a VMScluster environment

For details on setting up the DQS queues, see Part II, System Manager's Guide.

# 15

# After Installing DQS Software

## About This Chapter

This chapter describes tasks that you need to do after installing the DQS software (see Table 15–1).

**Table 15–1 Task Summary**

| Task | Description |
|---|---|
| 1 | Postinstallation Tasks: New Server Installation |
| 2 | Postinstallation Tasks: New Client Installation |
| 3 | Postinstallation Tasks for All Systems: New Installation |
| 4 | Postinstallation Tasks: Upgrade Server Installation |
| 5 | Postinstallation Tasks: Upgrade Client Installation |
| 6 | Postinstallation Tasks for All Systems: Upgrade Installation |
| 7 | Postinstallation Tasks: VMScluster Nodes |

## Note

If you find software problems while performing the postinstallation procedures, report the problem to Digital. Refer to Appendix G for instructions on how to report problems.

## Postinstallation Tasks: New Server Installation

### Overview

This section describes the tasks you may need to perform if this is a new installation of the DQS software on a DQS server node and tells you where to look for more information, if needed.

### Move DQS Server Account Directory

The DQS server uses a directory to temporarily store copies of the files that it is printing for DQS clients. These print files vary in size. Depending on the requirements of your environment, you may need to move this directory to an alternate device. DQS software provides a command procedure that enables you to change or move this directory to a different device.

If needed, move the DQS server account default directory to another device.

### Reference

See the section titled Move the Server Account Directory to Another Device in Chapter 8 of the System Manager's Guide for details.

### Define Server Queues

Before client nodes can access remote queues on your server node using the DQS software, define the set of queues on a DQS server node that may be made accessible to remote queues.

### Reference

See the section titled Setting Up Remote Queues in Chapter 8 of the System Manager's Guide for details.

### Customize Server Software

If you need to change DQS server logical definitions that relate to the behavior of the server software, do the following:

a. Copy SYS$COMMON:[SYSMGR]DQS$SYSTARTUP.TEMPLATE to SYS$COMMON:[SYSMGR]DQS$SYSTARTUP.COM.

b. Edit SYS$COMMON:[SYSMGR]DQS$SYSTARTUP.COM and make the required changes for your environment.

### Reference

See Chapter 8 in the System Manager's Guide for details on managing a DQS server.

## Postinstallation Tasks: New Client Installation

### Overview

The next sections describe the tasks you may need to perform if this is a new installation of the DQS software on a DQS client node and tells you where to look for more information, if needed.

### Customize Client Software

Customize the DQS client software for your specific environment.

If you need to change DQS client logical definitions that relate to the behavior of the client software, do the following:

a. If you have not already done so, copy SYS$COMMON:[SYSMGR]DQS$SYSTARTUP.TEMPLATE to SYS$COMMON:[SYSMGR]DQS$SYSTARTUP.COM.

b. Edit SYS$COMMON:[SYSMGR]DQS$SYSTARTUP.COM and make the required changes for your environment.

### Reference

See Chapter 9 in the System Manager's Guide for details on managing a DQS client.

## Postinstallation Tasks for All Systems: New Installation

### Overview

This section describes the tasks you may need to perform on all sytems if this is a new installation of the DQS software.

### Edit System Startup File

Edit the system startup file and add the command line that starts the DQS software.

| Operating System | Name of Startup File |
|---|---|
| OpenVMS AXP and OpenVMS VAX Version 6.*n* | SYSTARTUP_VMS.COM |
| OpenVMS VAX Version 5.*n* | SYSTARTUP_V5.COM |

#### DECnet Phase IV systems

With DECnet Phase IV networking software, the DQS software must start after the network has started. You must position this new command line after the line that invokes the network startup command procedure. The following is a sample portion of a system startup file that has been edited to start the DQS software:

```
$ @SYS$MANAGER:STARTNET.COM    ! Start DECnet software
       .
       .
       .
$ @SYS$STARTUP:DQS$STARTUP.COM ! Then start DQS software
```

Some installations use a batch job to start the network. With DECnet Phase IV networking software, the job that starts the network must complete before the DQS software can be started. The following portion of a system startup file shows one way to ensure that the network starts before the DQS software:

```
$ SUBMIT -
  SYS$MANAGER:STARTNET.COM, -  ! Start DECnet software, first
       .
       .
       .
  SYS$STARTUP:DQS$STARTUP.COM  ! Then, start DQS software
```

### DECnet Phase V systems

These examples do not apply to the networks running the DECnet/OSI Phase V networking software. DECnet Phase V software is started automatically. Check system startup requirements for any restrictions on the placement of the DQS startup command.

## Start DQS Software

The installation procedure does not start the DQS software. On all nodes that you want to run the DQS software, execute this command:

```
$ @SYS$STARTUP:DQS$STARTUP.COM Return
```

You can also use the SYSMAN utility to start DQS software in a VMScluster environment.

## Running IVP

To run the IVP independently of the installation procedure, enter the following command at the DCL prompt:

```
$ @SYS$TEST:DQS$IVP.COM Return
```

The DQS IVP is a command procedure that verifies that the DQS files are installed correctly. The installation procedure places the IVP command file, DQS$IVP.COM, in the SYS$TEST directory.

To troubleshoot a failed IVP, you can verify that each directory contains its proper file names by entering the following command at the DCL prompt:

```
$ DIRECTORY SYS$COMMON:[SYS*]*DQS* Return
```

## Activate DCL Command Table

The installation procedure modifies the DCL command table so that the DQS commands are recognized and processed. Users who are logged on during the installation process can use the DQS PRINT command immediately. However, other DQS commands: QSHOW, QDELETE/ENTRY, QSET/ENTRY are not available.

Notify all logged-in users who want to use the QSHOW, QDELETE/ENTRY, QSET/ENTRY commands that they need to do either of the following:

- Log out and log back in again
- Enter the following DCL command:

  ```
  $ SET COMMAND /TABLES=SYS$LIBRARY:DCLTABLES
  ```

  You can also use the SYSMAN utility to activate the DCL tables in a VMScluster environment.

## Postinstallation Tasks: Upgrade Server Installation

### Overview

This section describes the tasks you may need to perform if this is an upgrade installation of the DQS software on a DQS server node and tells you where to look for more information, if needed.

### Move DQS Server Account Directory

The DQS server uses a directory to temporarily store copies of the files that it is printing for DQS clients. These print files vary in size. Depending on the requirements of your environment, you may need to move this directory to an alternate device. DQS software provides a command procedure that enables you to change or move this directory to a different device.

### Reference

See the section titled Move the Server Account Directory to Another Device in Chapter 8 of the System Manager's Guide for details.

### Verify Queue Access Definitions

If your DQS server system is running the DQS software during the installation procedure, the procedure converts the Version 1.*n* server queue access definitions to the new format provided with DQS Version 1.3 software. The installation procedure creates the following file, which should contain the queue access definitions for your environment:

```
SYS$MANAGER:DQS$SERVER_CONFIG.TXT
```

The queue access definitions are specified in your DQS$STARTUP.COM file. The installation procedure renames your previous DQS$STARTUP.COM to DQS$STARTUP.COM_OLD.

Please review your DQS$STARTUP.COM_OLD file and compare it to the queue access definitions in the new .TXT file

Check the following log file for any issues encountered during the conversion of your queue definitions to the new security and access format:

```
$ TYPE SYS$UPDATE:DQS$CONVERT_SECURITY.LOG Return
```

## Remove Rightslist IDs

After DQS Version 1.3 software has been running on your system for a few weeks and you are satisfied with the operation of the new queue access mechanism, you may want to remove all the DQS$NODE_*xxxx* rightslist identifiers from your rightslist database. Do not do this until you are sure that you do not want to go back to a DQS Version 1.2 environment.

Use the OpenVMS Authorize utility to remove the rightslist identifiers that were used prior to DQS version 1.3.

The DQS V1.3 rightslist identifiers have the form DQS$NODE_ *nodename*, where *nodename* is a 6-character DECnet node name.

## Procedure

Use the following procedure to remove a rightslist identifier:

```
$ MCR AUTHORIZE
UAF> REMOVE/IDENTIFIER DQS$NODE_nodename
UAF> EXIT
```

## Example

The following example removes DQS$NODE_SMEDLY and DQS$NODE_JONJON from the rights list database:

```
$  MCR AUTHORIZE
UAF> REMOVE/IDENTIFIER DQS$NODE_SMEDLY
Identifier DQS$NODE_SMEDLY value: %X800100A2 removed from rights data base
UAF> REMOVE/IDENTIFIER DQS$NODE_JONJON
Identifier DQS$NODE_JONJON value: %X800100A3 removed from rights data base
UAF> EXIT
```

## Customize Server Software

With DQS Version 1.3, the DQS startup procedure is divided into two files:

- SYS$STARTUP:DQS$STARTUP.COM: Used to invoke SYS$MANAGER:DQS$SYSTARTUP.COM.
- SYS$MANAGER:DQS$SYSTARTUP.COM: Used to customize DQS logical names.

DQS$STARTUP.COM invokes SYS$MANAGER:DQS$SYSTARTUP.COM, which contains all the customizations.

A site-specific startup template that you can customize is provided your convenience:

```
SYS$COMMON:[SYSMGR]DQS$SYSTARTUP.TEMPLATE
```

## Procedure

To change server logical definitions, perform the following:

a. Copy SYS$COMMON:[SYSMGR]DQS$SYSTARTUP.TEMPLATE to SYS$COMMON:[SYSMGR]DQS$SYSTARTUP.COM.

b. Edit SYS$COMMON:[SYSMGR]DQS$SYSTARTUP.COM and add the server related changes that were previously defined in DQS$STARTUP.COM. These can be found in DQS$STARTUP.COM_OLD.

DQS server queue access is no longer defined by adding logical definitions to the startup procedure. Do not copy the following logicals to your DQS$SYSTARTUP.COM file:

- DQS$CONTROLLED_ACCESS
- DQS$DENY_ACCESS
- DQS$QUEUE_*xxxx* (where *xxxx* is a queue-name)

## Reference

See Chapter 8 in the System Manager's Guide for details on customizing a DQS server.

## Postinstallation Tasks: Upgrade Client Installation

### Overview

This section describes the tasks you may need to perform if this is an upgrade installation of the DQS software on a DQS client node and tells you where to look for more information, if needed.

### Customize Client Software

Customize the DQS client software for your specific environment.

With DQS Version 1.3, the startup procedure is divided into two files:

- SYS$STARTUP:DQS$STARTUP.COM: Used to invoke SYS$MANAGER:DQS$SYSTARTUP.COM.
- SYS$MANAGER:DQS$SYSTARTUP.COM: Used to customize the client logical names.

DQS$STARTUP.COM invokes SYS$MANAGER:DQS$SYSTARTUP.COM, which contains all the definitions.

A site specific startup template that you can customize is provided your convenience:

```
SYS$COMMON:[SYSMGR]DQS$SYSTARTUP.TEMPLATE
```

To change client logical definitions perform the following:

a. If you have not already done so, copy SYS$COMMON:[SYSMGR]DQS$SYSTARTUP.TEMPLATE to SYS$COMMON:[SYSMGR]DQS$SYSTARTUP.COM.

b. Edit SYS$COMMON:[SYSMGR]DQS$SYSTARTUP.COM and make the client-related changes that were previously defined in DQS$STARTUP.COM. These can be found in DQS$STARTUP.COM_OLD.

### Reference

See Chapter 9 in the System Manager's Guide for details on customizing a client.

## Postinstallation Tasks for All Systems: Upgrade Installation

### Overview

This section describes the tasks you may need to perform on all sytems if this is an upgrade installation of the DQS software.

### Edit System Startup File

Edit the system startup file to check that the DQS software is started correctly.

| Operating System | Name of Startup File |
|---|---|
| OpenVMS AXP and OpenVMS VAX Version 6.*n* | SYSTARTUP_VMS.COM |
| OpenVMS VAX Version 5.*n* | SYSTARTUP_V5.COM |

Look in the system startup procedure for the one of the following lines:

- @SYS$MANAGER:DQS$STARTUP.COM
- @SYS$STARTUP:DQS$STARTUP.COM

DQS Version 1.3 software requires that the DQS software start from the SYS$STARTUP directory. Change the command to the following if it was not already changed:

```
@SYS$STARTUP:DQS$STARTUP.COM
```

Do this on each node of a cluster that you want to run the DQS software. DQS software then starts automatically when the node reboots.

### Start DQS Software

The installation procedure does not start the DQS software. On all nodes that you want to run the DQS software, execute this command:

```
$ @SYS$STARTUP:DQS$STARTUP.COM Return
```

You can also use the SYSMAN utility to start DQS software in a VMScluster environment.

### Run IVP

To run the IVP independently of the any installation procedure, enter the following command at the DCL prompt:

```
$ @SYS$TEST:DQS$IVP.COM Return
```

The DQS IVP is a command procedure that verifies that the DQS files are installed correctly. The installation procedure places the IVP command file, DQS$IVP.COM, in the SYS$TEST directory.

To troubleshoot a failed IVP, you can verify that each directory contains its proper file names by entering the following command at the DCL prompt:

```
$ DIRECTORY SYS$COMMON:[SYS*]*DQS* Return
```

### Update Server Queues

After the DQS software is running, if you installed the DQS software on a server node, execute the following command procedure:

```
$ @SYS$MANAGER:DQS$SERVER_UPDATE_QUEUE_DEFNS Return
```

DQS Version 1.3 software does not require DQS queues to have the /RETAIN queue attribute. This command procedure allows you to change the definitions of your DQS queues to eliminate these attributes.

### Activate DCL Command Table

The installation procedure modifies the DCL command table so that the DQS commands are recognized and processed. Users who are logged on during the installation process can use the DQS PRINT command immediately. However, other DQS commands: QSHOW, QDELETE/ENTRY, QSET/ENTRY are not available.

Notify all logged-in users who want to use the QSHOW, QDELETE/ENTRY, QSET/ENTRY commands that they need to do either of the following:

- Log out and log back in again
- Enter the following DCL command:

  ```
  $ SET COMMAND /TABLES=SYS$LIBRARY:DCLTABLES
  ```

  You can also use the SYSMAN utility to activate DCL tables a VMScluster environment.

## Postinstallation Tasks: VMScluster Nodes

### Overview

This section describes the tasks you may need to perform on a VAXcluster system.

### Edit System Startup File

Edit the system startup file and add the following line to start the DQS software for each cluster member where the DQS software is to execute:

```
$ @SYS$STARTUP:DQS$STARTUP
```

### Replace DCL Tables

If you install DQS client software on a VMScluster member, the DCL tables are updated to include the DQS QSHOW, QDELETE /ENTRY, and QSET/ENTRY commands on that member only. To include these commands in the DCL tables of other cluster members, either execute the SYSMAN utility or issue the following commands from the system manager's account on each cluster member:

```
$ INSTALL Return
INSTALL> REPLACE SYS$SHARE:DCLTABLES Return
INSTALL> EXIT Return
$
```

### Define Network Object

The DQS network object must be defined on each node within the VMScluster where you want to run DQS software.

On DECnet Phase IV systems, use the Network Control Program. On DECnet/OSI Phase V systems, use the Network Control Language (NCL).

**NCP Commands**

To define the DQS object using NCP, execute the SYSMAN utility or follow these steps:

a. Obtain the randomly generated password, created during the installation procedure, for the DQS client/server account. From a privileged account on the node where you installed the software, enter the following commands:

```
$ RUN SYS$SYSTEM:NCP Return
NCP> SHOW OBJECT DQS Return
```

The system returns the password for the DQS client/server account in the Password field. In the following example, TESTING123 is the randomly generated client/server account password.

```
Object Volatile Summary as of  5-JAN-1994 10:42:14

  Object  Number  File/PID          User Id       Password

 DQS          66  DQS$SERVER.EXE    DQS$SERVER    TESTING123
NCP>
```

b. Enter the following NCP commands on each of the other nodes in the VMScluster environment to define the DQS object (using the returned password for *password* in the commands):

```
$ RUN SYS$SYSTEM:NCP [Return]
NCP> DEFINE OBJECT DQS NUMBER 66 FILE DQS$SERVER.EXE - [Return]
_ USER DQS$SERVER PASSWORD password [Return]
NCP> SET OBJECT DQS NUMBER 66 FILE DQS$SERVER.EXE - [Return]
_ USER DQS$SERVER PASSWORD password [Return]
NCP> EXIT [Return]
$
```

### NCL Commands

To define the DQS object using NCL, execute the SYSMAN utility or enter the following NCL commands on each of the other nodes in the VMScluster environment:

```
s>(NCL>) CREATE SESSION CONTROL APPLICATION DQS
NCL> SET SESSION CONTROL APPLICATION DQS, -
_NCL> ADDRESS = {NUMBER=66}, IMAGE NAME = DQS$SERVER.EXE, -
_NCL> INCOMING PROXY = FALSE, OUTGOING PROXY=FALSE, -
_NCL> NODE SYNONYM=FALSE, USER NAME = "DQS$SERVER"

Node 0 Session Control Application dqs
at 1994-03-16-13:34:58.193-05:00I1.614

Characteristics

    Addresses                       =
       {
          number = 66
       }
    Image Name                      = DQS$SERVER.EXE
    Incoming Proxy                  = False
    Outgoing Proxy                  = False
    Node Synonym                    = False
    User Name                       = "DQS$SERVER"

NCL> EXIT
```

# Part IV
## Appendixes

**Contents**

This part contains additional information for the DQS user, system manager, and the installer.

| | |
|---|---|
| For the user: | Description of relevant DQS messages. |
| For the system manager: | Description of relevant DQS messages, summary of DQS server and client logical names, and a description of how other symbionts interact with the DQS software. |
| For the installer: | List of files added to the system, sample installation scripts and configuration files, and a description of relevant DQS messages. |

# A

# DQS User Messages

## About This Appendix

This appendix lists error and diagnostic messages issued by the DEC Distributed Queuing Service for OpenVMS Systems (DQS) software. Each message includes a description and, where appropriate, corrective actions that you can take.

## DQS Message Format

Most DQS messages follow the same format as standard OpenVMS system messages and appear as follows:

```
%DQS-l-ident, text
```

Where:

| | |
|---|---|
| *l* | Is the severity level of the message, such as -W-, for a warning message. |
| *ident* | Is the message identification, such as NOACCESS. |
| *text* | Is the message text, such as "specified node does not have access to this server." |

## Severity Codes

DQS messages use the same severity codes as OpenVMS system messages: S, I, W, E, and F.

## For More Information

DQS messages that concern the management and configuration of the DQS software are documented in Appendix B.

Refer to the *OpenVMS System Messages and Recovery Procedures Reference Manual* for an explanation of the format and a definition of severity codes for OpenVMS system messages.

## Message Section

### Notification Messages

The DQS software can return the following notification messages (these messages do not follow the standard OpenVMS message format).

Print job *file-name* (queue *queue-name*, entry *job-number*) terminated with error status at *date/time*

**Explanation:** Your print job has not successfully printed on the server node. A secondary error message is displayed describing the reason why the error occurred.

**User Action:** Rectify the problem based on the reason given in the secondary error message.

Print job *file-name* (queue *queue-name*, entry *job-number*) completed at *date/time*

**Explanation:** None, this message is informational.

**User Action:** Retrieve your print job.

### DQS Message Definitions

Because all DQS messages have the same prefix %DQS-*l*-, the messages in this section are listed alphabetically according to their message *ident*.

BADNODE, specified node has been denied access to this server.

**Explanation:** The remote node for your queue is not accessible from your client.

**User Action:** Contact your system manager.

BADQUE, specified queue not valid for DQS access.

**Explanation:** The remote queue on the server is not accessible by DQS clients.

**User Action:** Contact your system manager.

BAD_REMOTE_NAME, *logical-name* must be defined for this queue.

**Explanation:** The remote system name has been specified incorrectly.

**User Action:** Contact your system manager.

CHECKPOINTED, job was checkpointed.

**Explanation:** Your print job was checkpointed and DQS could not remove the checkpoint.

**User Action:** Delete your job from the queue and resubmit it.

ENTRY_REQUIRED, /entry=number required.

**Explanation:** You did not specify the required /ENTRY=*number* qualifier in your QSET or QDELETE command.

**User Action:** Reenter the command using the /ENTRY=*number* qualifier, specifying the appropriate job entry number.

ERROROPENING, error opening *file-name* as input.

**Explanation:** The client's print symbiont could not open the specified file for the reason cited in the secondary error message that accompanies this message. This error message will be printed in the print output on the server. Usually, this message occurs when the file to be printed is deleted before the file is transferred by the client symbiont to the server.

**User Action:** Rectify the problem based on the reason given in the secondary error message.

ERRORREADING, error reading *file-name*.

**Explanation:** The client's print symbiont could not read the specified file for the reason cited in the secondary error message that accompanies this message.

**User Action:** Rectify the problem based on the reason given in the secondary error message.

INACCESSIBLE, remote queue *queue-name* is inaccessible.

**Explanation:** The server you are attempting to print to is not accessible.

**User Action:** The job will not be processed until the server is reachable. However, for QSHOW, QSET/ENTRY, or QDELETE/ENTRY command functions, you must wait until the server is available.

INCOMPPROT, incompatible DQS Server and Client protocol versions.

**Explanation:** The version of DQS software on the client is incompatible with the version of DQS software on the server.

**User Action:** Contact your system manager.

INVLOGFIL, invalid log file.

**Explanation:** You specified an invalid log file in a QSET /ENTRY command. The reason for the error in your file specification is cited in the secondary error message that accompanies this message.

**User Action:** Correct the file specification and reenter the QSET/ENTRY command.

INVQUAVAL, value '*value*' invalid for */qualifier-name* qualifier.

**Explanation:** You specified an invalid value for the qualifier */qualifier-name*.

**User Action:** Specify the proper value for the qualifier.

INVVER, versions of protocol do not match.

**Explanation:** The version of DQS software on the client is incompatible with the version of DQS software on the server to which you are attempting to queue your print job.

**User Action:** Contact your system manager.

JOBDELETED, *number* job(s) deleted.

**Explanation:** The QDELETE/ENTRY command you specified did not delete all of the jobs you specified, only the number of jobs specified by *number*.

**User Action:** Check the job number(s) for the job(s) you attempted to delete, as well as your access privileges for deleting those jobs, and reenter the command.

LONGNODENAME, Node name size problem due to DECnet /OSI fullname.

**Explanation:** The DQS symbiont received a node name (possibly from a daisy-chained queue) that is too long for the

version of the OpenVMS operating system or the DECnet networking software to handle.

**User Action:** Notify your system manager of the problem. Systems need upgrading or DQS queues need to be reconfigured or both.

MSG_PE, protocol error.

**Explanation:** A protocol error indication was received. This error usually occurs when incompatible versions of DQS software reside on server and client systems.

**User Action:** Contact your system manager.

MSG_REMOTE, remote system error:

**Explanation:** An error occurred on the remote system and is described in the secondary error message that accompanies this message.

**User Action:** Rectify the problem based on the reason given in the secondary error message.

NOACCESS, specified node does not have access to this server.

**Explanation:** The client is not authorized to use the queue(s) on the server.

**User Action:** Contact your system manager.

NODELETE, /NODELETE is allowed only on local jobs.

**Explanation:** You cannot use the /NODELETE qualifier for a job that is not on your local node; that is, a job that has transferred to the server.

**User Action:** Do not use the /NODELETE qualifier for jobs that have transferred to the server.

NOJOBDELETED, no jobs deleted.

**Explanation:** The QDELETE/ENTRY command you specified did not delete any jobs.

**User Action:** Check the job number of the job you are attempting to delete, as well as your access privileges to delete the job, and try again.

NOJOBSET, no jobs set.

**Explanation:** The QSET/ENTRY command you specified did not set any jobs.

**User Action:** Check the job number of the job you are attempting to modify, as well as your access privileges to modify the job, and try again.

NOSUCHQUE, no such remote queue *queue-name*.

**Explanation:** The remote queue specified by *queue-name* does not currently exist on the server.

**User Action:** Contact your system manager.

NOTOWN, not owner of the job.

**Explanation:** You are not the owner of the job that you are attempting to delete or modify. The DQS software determines job ownership by the node name and user name with which the job was initiated.

**User Action:** Ensure that you have specified the correct job number and that you are working from the same account on the same client from which you initiated the job.

NO_ACCESS_CONTROL, no access control strings allowed in server node name.

**Explanation:** The remote system name has been specified incorrectly.

**User Action:** Contact your system manager.

NO_NODE_SPECIFIED, printer queue name does not specify a node.

**Explanation:** The remote system name has been specified incorrectly.

**User Action:** Contact your system manager.

NO_SET_LISTS, list of job numbers for SET not supported.

**Explanation:** You cannot specify a sequence of job numbers in a QSET/ENTRY command, such as /ENTRY=(1,2,3).

**User Action:** Issue a separate QSET/ENTRY command for each job.

P8RESERVED, parameter 8 (P8) is reserved for DQS on remote systems.

**Explanation:** You attempted to specify P8 when issuing a PRINT command to a DQS queue. Parameter 8 is reserved for DQS information.

**User Action:** Do not specify P8 when using DQS queues. Instead add your P8 arguments to parameters 1 through 7.

REC_MSG, received message: "*message*".

**Explanation:** The server is running an old version of DQS software.

**User Action:** Contact your system manager.

REINSTALL, install new DQS software.

**Explanation:** The version of DQS software on the client is incompatible with the version of DQS software on the server.

**User Action:** Contact your system manager.

REQNOEXIST, requeue remote does not exist on remote system.

**Explanation:** The queue you specified with a QSET/ENTRY /REQUEUE=*queue-name* command does not exist. You can only requeue to a queue that is on the same remote system as the print job.

**User Action:** Requeue your job to another remote queue, or delete the job and submit a new job to a valid remote queue.

# B

# DQS System Manager Messages

## About This Appendix

This appendix alphabetically lists and defines all error and diagnostic messages issued by DQS. Where appropriate, it describes any corrective actions that you can take.

## DQS Message Format

DQS messages have the same format as standard OpenVMS system messages and appear as follows:

`%DQS-`*`l`*`-`*`ident`*`, `*`text`*

Where

| | |
|---|---|
| *l* | Is the severity level of the message, such as -W-, for a warning message. |
| *ident* | Is the message identification, such as NOACCESS. |
| *text* | Is the message text, such as "specified node does not have access to this server." |

### Severity Levels of DQS Messages

DQS messages use the same severity codes as OpenVMS system messages: S, I, W, E, and F.

### For More Information

DQS messages that concern printing with the DQS software are documented in Appendix A for the user and in this appendix for the system manager.

Refer to the *OpenVMS System Messages and Recovery Procedures Reference Manual* for an explanation of the format and a definition of severity codes for OpenVMS system messages.

## Submitting a Software Performance Report

### When to Submit

When you receive a DQS internal software error, you should submit a Software Performance Report (SPR) to Digital.

Refer to the Appendix G for information on how to submit an SPR.

## Message Section

### DQS Notification Messages

The DQS software can return the following notification messages (these messages do not follow the standard OpenVMS message format).

Print job *job* (queue *queue*, entry *job-number*) completed at *date time*

**Explanation:** Your print job has printed successfully on the server.

**User Action:** None. This is an informational message.

Print job *job* (queue *queue*, entry *job-number*) terminated with error status at *date time*

**Explanation:** Your print job has not printed successfully on the server node. A secondary error message is displayed that describes the reason why the error occurred.

**User Action:** Rectify the problem based on the information in the secondary message.

### DQS Message Definitions

Because all DQS messages have the same prefix %DQS-*l*-, the messages in this section are listed alphabetically according to their message *ident*.

BADFILE, UL total syntax errors detected in file: *filename errors*.

**Explanation:** The DQS$SERVER_UPDATE_CONFIG.EXE program detected errors when parsing the server configuration text file. The name of the file and the total number of syntax errors in the file are displayed.

**User Action:** Edit the file and fix the syntax errors. Invoke the SYS$MANAGER:DQS$SERVER_UPDATE_CONFIG.COM command procedure to effect the change.

BADFORM, INTERNAL SOFTWARE ERROR - Translate form failed for form *form-name*.

**Explanation:** This is an internal DQS error.

**User Action:** Submit an SPR.

BADNAM, INTERNAL SOFTWARE ERROR - process name does not contain underscore, and must.

**Explanation:** This is an internal DQS error.

**User Action:** Submit an SPR.

BADNODE, specified node has been denied access to this server.

**Explanation:** Your client is denied access to the server to which you are attempting to queue your print job.

**User Action:** Consult the server system manager and request that your client node have access to the server. The server system manager needs to modify the DQS$SERVER_CONFIG.TXT file to allow this access.

BADQUE, specified queue not valid for DQS access.

**Explanation:** The remote queue you specified is not defined on the server as accessible to DQS clients. This means one of the following conditions exists:

- The SYS$MANAGER:DQS$SYSTARTUP.COM command file on the server is not set up correctly.
- The SYS$STARTUP:DQS$STARTUP.COM file has not been executed.
- The client is not permitted to use the remote queue.

**User Action:** Consult the server system manager and request that your node be granted access to the remote queue. The system manager may need to modify the DQS$SERVER_CONGIG.TXT file on the server node.

BAD_REMOTE_NAME, *logical-name* must be defined for this queue.

**Explanation:** The client queue you specified has not been defined as a DQS client queue. For a queue to be defined as a client queue, a logical name of the form DQS$REMOTE_*client-queue* must be defined for the queue.

**User Action:** Check the SYS$MANAGER:DQS$SYSTARTUP.COM command file to see whether the client queue has been defined; if so, reexecute the SYS$STARTUP:DQS$STARTUP.COM command file. If not, add a command line to the SYS$MANAGER:DQS$SYSTARTUP.COM command file that invokes the SYS$MANAGER:DQS$DEFINE.COM

command file and defines the queue as a DQS client queue. Then reexecute SYS$STARTUP:DQS$STARTUP.COM.

If the queue is not intended to be a DQS client queue, you can initialize the queue with a symbiont other than the DQS symbiont.

If these actions fail to solve the problem, submit an SPR.

BADTMPFILE, bad temporary file.

**Explanation:** A temporary file created during the IVP could not be deleted.

**User Action:** The IVP may not have failed. Delete the file manually.

BLKCNTEXC, INTERNAL SOFTWARE ERROR - Block count in message header too large.

**Explanation:** This is an internal DQS error.

**User Action:** Submit an SPR.

CANTPRINT, cannot print job.

**Explanation:** The IVP could not execute the PRINT command and has failed.

**User Action:** Reinstall the DQS software.

CHANGEFORMFAILED, INTERNAL SOFTWARE ERROR - Change form failed for queue *queue-name*.

**Explanation:** This is an internal DQS error.

**User Action:** Submit an SPR.

CHECKPOINTED, job was checkpointed.

**Explanation:** Your print job was checkpointed and the DQS software cannot remove the checkpoint.

**User Action:** Delete your job from the queue and resubmit it.

DISKFULL, remote server disk is full.

**Explanation:** Your print job cannot be copied to the server node. Either there is not enough disk space on the server to store your file(s) for printing, or the quota for the DQS server account [DQS$SERVER] was exceeded.

**User Action:** If this error persists, either ask the server system manager to free up some disk space on the server, or queue your job to another server.

EMPTYFILE, DQS server configuration file: *filename* is empty.

**Explanation:** Either the DQS$SERVER_UPDATE_CONFIG.EXE program or the DQS$SERVER program encountered an empty file when trying to read the indicated file.

**User Action:** Check that the file exists, is unprotected, and is not corrupted. The file's contents may have somehow become corrupted. If file name is DQS$SERVER_CONFIG.DAT, you can regenerate the file by deleting DQS$SERVER_CONFIG.DAT and then invoking the SYS$MANAGER:DQS$SERVER_UPDATE_CONFIG.COM command procedure to recreate the file.

ENTRY_REQUIRED, entry=number required.

**Explanation:** You did not specify the required /ENTRY=*number* qualifier in your QSET or QDELETE command.

**User Action:** Reenter the command using the /ENTRY=*number* qualifier, specifying the appropriate job entry number.

ERROROPENING, error opening *file-name* as input.

**Explanation:** The DQS client symbiont cannot open the specified file for the reason cited in the secondary error message that accompanies this message. The secondary error message is printed in the print output on the server. Usually, this error message occurs when the file to be printed is deleted before the file is transferred by the client symbiont to the server.

**User Action:** Rectify the problem based on the information in the secondary message.

ERRORREADING, error reading *file-name*.

**Explanation:** The DQS client symbiont could not read the specified file for the reason cited in the secondary error message that accompanies this message.

**User Action:** Rectify the problem based on the information in the secondary message.

INACCESSIBLE, remote queue *queue-name* is inaccessible.

**Explanation:** A connection cannot be established to the server with which you are attempting to perform a QSHOW, QSET/ENTRY, or QDELETE/ENTRY function.

**User Action:** Remember that if you submit a print request to the server, it is not processed until the server is reachable. However, for QSHOW, QSET/ENTRY, or QDELETE/ENTRY command functions, the server must be reachable when the command is issued.

INCOMPPROT, incompatible DQS Server and Client protocol versions.

**Explanation:** The version of DQS software on the client is incompatible with the version of DQS software on the server.

**User Action:** Make sure that compatible versions of DQS software are installed on both the client and server systems.

INVALIDSERVER, invalid server node or queue name.

**Explanation:** During a client IVP, you entered either a node name or a queue name that is not valid. You are prompted to provide another node and queue name.

**User Action:** Enter a valid node and queue name in response to the prompt:

```
Enter remote NODE::QUEUE
```

INVBLKTYP, INTERNAL SOFTWARE ERROR - Invalid block type *number* in message.

**Explanation:** This is an internal DQS error.

**User Action:** Submit an SPR.

INVLINE, invalid line encountered in file: *filename*, at line number *line*.

**Explanation:** The DQS$SERVER_UPDATE_CONFIG.EXE program detected a syntax error when parsing the server configuration file. The name of the file and the line number where the error was detected are displayed.

**User Action:** Edit the file and fix the syntax error. Invoke the SYS$MANAGER:DQS$SERVER_UPDATE_CONFIG.COM command procedure to effect the change.

INVLOGFIL, invalid log file.

**Explanation:** You specified a log file incorrectly in a QSET /ENTRY command. The reason for the error is cited in the secondary error message that accompanies this message.

**User Action:** Correct the file specification and reenter the QSET/ENTRY command.

INVMSGLEN, INTERNAL SOFTWARE ERROR - Invalid message length in header.

**Explanation:** This is an internal DQS error.

**User Action:** Submit an SPR.

INVMSGTYP, INTERNAL SOFTWARE ERROR - Invalid message type *number* in header.

**Explanation:** This is an internal DQS error.

**User Action:** Submit an SPR.

INVMSG, INTERNAL SOFTWARE ERROR -invalid message in current state.

**Explanation:** This is an internal DQS error.

**User Action:** Submit an SPR.

INVQUAVAL, value '*value*' invalid for */qualifier-name* qualifier.

**Explanation:** The value you specified for the qualifier */qualifier-name* is invalid.

**User Action:** Consult the documentation if necessary and specify the proper value for the qualifier.

INVREQCOD, invalid request code, code = *code*.

**Explanation:** The job controller issued an invalid request code.

**User Action:** Submit an SPR.

INVVER, versions of protocol do not match.

**Explanation:** The software version of DQS on your client is incompatible with the software version of DQS on the server to which you are attempting to queue your print job.

**User Action:** Make sure that the proper versions of DQS software are installed both on your client and on your client's server(s).

JOBDELETED, *number* job(s) deleted.

**Explanation:** The QDELETE/ENTRY command you issued did not delete all of the jobs you specified, only the number of jobs specified by *number*.

**User Action:** Check the job number(s) for the job(s) you attempted to delete as well as your access privileges for deleting those jobs.

LONGNODENAME, Node name size problem due to DECnet /OSI fullname.

**Explanation:** The DQS symbiont received a node name (possibly from a daisy-chained queue) that is too long for the version of the OpenVMS operating system or the DECnet networking software to handle.

**User Action:** Upgrade systems to DECnet/OSI Phase V software or reconfigure queues or both.

MSG_PE, protocol error.

**Explanation:** A protocol error indication was received. This error is usually incurred by having incompatible DQS software versions on server and client systems.

**User Action:** Make sure that your client and server systems have compatible versions of DQS software installed on them.

MSG_REMOTE, remote system error.

**Explanation:** An error occurred on the remote system. The error is described in the secondary error message that accompanies this message.

**User Action:** Rectify the problem based on the information in the secondary message.

MSG_TYPE_RANGE, INTERNAL SOFTWARE ERROR - Message type received '*message-value*' out of range.

**Explanation:** This is an internal DQS error.

**User Action:** Submit an SPR.

MSG_TYPE_UNEXPECTED, INTERNAL SOFTWARE ERROR - Message type received '*message-value*' unexpected.

**Explanation:** This is an internal DQS error.

**User Action:** Submit an SPR.

MSG_UNK, INTERNAL SOFTWARE ERROR - Unknown error message class: *class*.

**Explanation:** This is an internal DQS error.

**User Action:** Submit an SPR.

NOACCESS, specified node does not have access to this server.

**Explanation:** Your client is not authorized to use the queue(s) on the server.

**User Action:** If you want your client to use the remote queue(s) on the server, you must have the server system manager grant your client system access to the server.

NODELETE, job cannot be deleted from queue.

**Explanation:** The IVP could not delete the test job from the remote queue and failed.

**User Action:** Delete the job manually, check the network link, and rerun the IVP.

NODELETE, /NODELETE is allowed only on local jobs.

**Explanation:** You cannot use the /NODELETE qualifier for a job that is not on your local node (that is, a job that has transferred to the server).

**User Action:** Do not use the /NODELETE qualifier for jobs that have transferred to the server.

NODEOFF, node *node-name* offline.

**Explanation:** The specified server is not currently reachable and a job is waiting to transfer to it.

**User Action:** Check the status of the server or ask the server system manager to bring the server back on line.

NOJID, INTERNAL SOFTWARE ERROR - Job number, job name, or * required.

**Explanation:** This is an internal DQS error.

**User Action:** Submit an SPR.

NOJOBDELETED, no jobs deleted.

**Explanation:** The QDELETE/ENTRY command you issued did not delete any jobs.

**User Action:** Check the job number of the job you are attempting to delete, as well your access privileges to delete the job.

NOJOBSET, no jobs set.

**Explanation:** The QSET/ENTRY command you issued did not set any jobs.

**User Action:** Check the job number of the job you are attempting to modify, as well your access privileges to modify the job.

NOPRIV, need SYSPRV, SYSNAM, OPER, NETMBX, TMBMBX privileges.

**Explanation:** Your account does not have the correct privileges, or they have not been enabled.

**User Action:** Change your account registration (or enable your privileges) and rerun the IVP.

NOREMOTEQUE, remote queue does not exist or is inaccessible.

**Explanation:** The remote queue you specified during the installation does not exist, the network link is down, or the client does not have access to the remote queue.

**User Action:** For a server IVP, reinstall the DQS software For a client IVP, check the target queue name, the network link, and that the DQS$QUEUE_*queue-name* logical name on the server allows the client access. Then reinstall the server software and rerun the IVP.

NOSUCHQUE, no such remote queue *queue-name*.

**Explanation:** The remote queue specified by *queue-name* does not currently exist on the server, even though it is defined as a valid DQS remote queue.

**User Action:** Initialize the specified remote queue on the server.

NOTOWN, not owner of the job.

**Explanation:** You are not the owner of the job that you are attempting to delete or modify. The DQS software determines job ownership by the node name and user name with which the job was initiated.

**User Action:** Check to see that you have specified the correct job number. If so, make sure that you are working from the same account on the same client as when you initiated the job.

NOTSTARTED, The DQS software has not been started yet.

**Explanation:** The DQS software has not been started on the server yet.

**User Action:** Ask the server system manager to start the DQS software. Print jobs transfer up to 5 minutes after the software starts.

NO_ACCESS_CONTROL, no access control strings allowed in server node name.

**Explanation:** A DQS$REMOTE_*client-queue* logical name, which specifies a client queue's associated server and remote queue, includes access control information in its definition.

**User Action:** Redefine the logical name using the format DQS$REMOTE_*client-queue server-node*::*remote-queue*. Also, check the definition of the DQS$REMOTE logical name for the client queue in the DQS$STARTUP.COM command file.

NO_NODE_SPECIFIED, printer queue name does not specify a node.

**Explanation:** A DQS$REMOTE_*client-queue* logical name does not specify a node name in its definition.

**User Action:** Redefine the logical name using the format DQS$REMOTE_*client-node server-node*::*remote-queue*. Also, check the definition of the DQS$REMOTE logical name for the client queue in the DQS$STARTUP.COM command file.

NO_SET_LISTS, list of job numbers for SET not supported.

**Explanation:** You cannot specify a sequence of job numbers in a QSET/ENTRY command, such as /ENTRY=(1,2,3).

**User Action:** Issue a separate QSET/ENTRY command for each job entry.

OLDIVP, old IVP job exists in queue, deleting.

**Explanation:** An IVP test print job already exists in the queue. It is deleted.

**User Action:** No user action. This is an informational message.

OPENFILE, error opening DQS server configuration file: *filename*.

**Explanation:** Either the DQS$SERVER_UPDATE_CONFIG.EXE program or the DQS$SERVER program encountered an error when trying to open the indicated file.

**User Action:** Check that the file exists and is unprotected. Make sure there is sufficient disk space to create a new file.

P8RESERVED, parameter P8 is reserved for the DQS software on remote systems.

**Explanation:** You specified parameter 8 when issuing a PRINT command to a DQS queue. Parameter 8 is reserved for DQS information.

**User Action:** Do not specify parameter 8 when using the DQS software. Combine parameter 8 arguments with parameter 1 to parameter 7 arguments.

PE_MSG_NAME, INTERNAL SOFTWARE ERROR - Protocol error, message_type = *type*.

**Explanation:** This is an internal DQS error.

**User Action:** Submit an SPR.

PE_MSG_NUMBER, INTERNAL SOFTWARE ERROR - Protocol error, unknown message_type = #*number*.

**Explanation:** This is an internal DQS error.

**User Action:** Submit an SPR.

PROTOCOL_ERROR, INTERNAL SOFTWARE ERROR - Protocol error; invalid message received - *number*.

**Explanation:** This is an internal DQS error.

**User Action:** Submit an SPR.

READFILE, error reading DQS server configuration file: *filename*.

**Explanation:** Either the DQS$SERVER_UPDATE_CONFIG.EXE program or the DQS$SERVER program encountered an error when trying to read the indicated file.

**User Action:** Check that the file exists, is unprotected, and is not corrupted. The file's contents may have somehow become corrupted. If file name is DQS$SERVER_CONFIG.DAT, you can regenerate the file by deleting DQS$SERVER_CONFIG.DAT and then invoking the SYS$MANAGER:DQS$SERVER_UPDATE_CONFIG.COM command procedure to recreate the file.

REC_MSG, received message: "*message*".

**Explanation:** One of the previous messages came from a server that is running obsolete DQS software and that does not return a full error code.

**User Action:** Install the current version of DQS software on the server.

REINSTALL, install new DQS software.

**Explanation:** The version of DQS software on the client is incompatible with the version of DQS software on the server.

**User Action:** Make sure that your client and server systems have compatible versions of DQS software installed.

REPORT, reporting the following message to the client.

**Explanation:** The subsequent error messages are being reported to the client system.

**User Action:** None.

REQNOEXIST, requeue, remote queue does not exist on remote system.

**Explanation:** The queue you specified with a QSET/ENTRY /REQUEUE=*queue-name* command does not exist. You can only requeue to a queue that is on the same remote system as the print job.

**User Action:** Requeue your job to another remote queue, or delete the job and submit a new job to a valid remote queue.

SHUTDOWN, shutting down queue *queue-name* because of error.

**Explanation:** An error occurred while processing a print job on the queue *queue-name*. The queue is stopped because it cannot process any jobs.

**User Action:** Issue a QSHOW command of the specified queue to determine the reason for the error. Correct the error and restart the queue.

TOOMANYERRORS, too many errors talking to the server.

**Explanation:** Too many errors have occurred while attempting to transfer a print job to a server. The client symbiont stops until the problem is fixed.

**User Action:** Check the reported errors, rectify the problem, and restart the client queue.

UNEXPECTEDSTATE, INTERNAL SOFTWARE ERROR - Unexpected state: UL, encountered while parsing arguments.

**Explanation:** The DQS$SERVER_UPDATE_CONFIG.EXE program detected an inconsistency in its internal state while parsing the user input file.

**User Action:** Submit an SPR. If necessary, use the previous version of DQS$SERVER_CONFIG_.TXT file and rerun the program. [En

UNKNOWN_FUNCTION, INTERNAL SOFTWARE ERROR - function: *function-name* unknown.

**Explanation:** This is an internal DQS error.

**User Action:** Submit an SPR.

VALID_MESSAGES, INTERNAL SOFTWARE ERROR - Valid messages mask = *number*.

**Explanation:** Indicates which messages this server expected.

**User Action:** Submit an SPR.

WAIT, waiting for file to transfer to remote queue.

**Explanation:** The test file is being transferred or will be transferred.

**User Action:** No user action. This is an informational message.

WRITEFILE, error writing DQS server configuration file: *filename*.

**Explanation:** The DQS$SERVER_UPDATE_CONFIG.EXE program encountered an error when trying to write to the indicated file.

**User Action:** Check that the file exists and is unprotected. Make sure sufficient disk space exists to extend the file. When the problem is fixed, invoke the

SYS$MANAGER:DQS$SERVER_UPDATE_CONFIG.COM command procedure to effect the change.

XFERFAILED, Failed to transfer job to remote queue.

**Explanation:** The file could not be transferred to the remote queue.

**User Action:** An additional status message accompanies this error message to explain why the transfer failed. Possible reasons for the transfer failure include the following:

- DQS$IVP_TEST_FORM may no longer exist on the server node.

  ```
  %JBC-E-NOSUCHFORM\ No such form
  ```

  To define the DQS$IVP_TEST_FORM, enter the following DCL command on the server system:

  ```
  $ DEFINE/FORM DQS$IVP_TEST_FORM 1110 -
           /STOCK = DQS_IVP_TEST_STOCK -
           /DESCRIPTION = "DQS test form, do not delete"
  ```

  Define this form on the server and start the IVP again.

- Not enough disk space is available, the DQS software was not started, or some other problem exists.

  ```
  %DQS-W-MSG-REMOTE, reason
  ```

  The *reason* defines the problem. Examine the reason, correct the problem, and try again.

XFERFAILED, transfer to server failed with error.

**Explanation:** Your current print job failed to transfer to the server node for the reason specified in the secondary error message. The DQS software attempts to retransfer the job after a wait.

**User Action:** If the problem persists, ask the server system manager to rectify the problem on the server node. The problem may be transient, such as unavailability of a temporary resource.

XFERTIME, timed out trying to transfer file to remote queue.

**Explanation:** The file transfer could not complete in 15 minutes.

**User Action:** Check the network link and rerun the IVP.

# C

# Server and Client System Logical Names

## About This Appendix

This appendix alphabetically summarizes the DQS server and client system logical names. Each logical name is described briefly, along with its value, function, default value (if any), and image.

## DQS Product Logical Names

**Important** The following logical names are used by the DQS software and should NOT be modified. These logical name definitions may affect the DQS client software, the DQS server software, or both.

| Product Logical Name | Description, Value, Default, and Image |
|---|---|
| DQS$CONFIGURATION | **Description:** Type of DQS software installed.<br>**Value:** Value is either CLIENT or SERVER.<br>**Default:** None.<br>**Image:** DQS$SERVER, DQS$SMB |
| DQS$CONTROLLED_ ACCESS | **Description:** Set by the DQS software. Indicates on the server whether or not to check the security database to grant clients access to the server.<br>**Default:** None. If this logical name is not defined on a server, the DQS software does not check on whether a client is a valid client of a server.<br>**Image:** DQS$SERVER |
| DQS$DENY_ACCESS | **Description:** Set by the DQS software. Indicates on the server whether to check the security database to determine if clients have been denied access to the server.<br>**Default:** None. All client nodes are valid.<br>**Image:** DQS$SERVER |
| DQS$QUEUE_*remote-queue* | **Description:** Set by the DQS software. Identifies a queue as a DQS network-accessible queue.<br>**Default:** None. If this logical name is not defined for a queue, the queue is not available as a DQS remote queue.<br>**Image:** DQS$SERVER |
| DQS$VERSION | **Description:** A string that names the version number of the installed DQS software.<br>**Value:** V*n.n*. The *n.n* is a number that stands for the currently installed version; for example, V1.3.<br>**Default:** Version number of the installed product. |

## DQS Server Logical Names

### Modifying Names

DQS software provides the following logical name definitions for the DQS server software. The system manager can modify the definitions in the site-specific startup procedure SYS$MANAGER:DQS$SYSTARTUP.COM.

| Server Logical Name | Description, Value, Default and Image |
|---|---|
| DQS$ACCOUNTING_BY_SYSTEM | **Description:** Enables client node names to be used in the account field in a server's accounting record.<br>**Value:** Arbitrary.<br>**Default:** If this logical name is not defined, accounting by client system is not enabled.<br>**Image:** DQS$SERVER |
| DQS$IDLE_TIME | **Description:** The delta time that a DQS$SERVER process maintains a connection to an inactive client that is not sending print jobs.<br>**Value:** The delta time.<br>**Default:** "0 00:15:00.0" (15 minutes)<br>**Image:** DQS$SERVER |
| DQS$LOG_ACCESS | **Description:** Creates a file in the [DQS$SERVER.NODES] subdirectory of the DQS server account each time a client establishes a connection to the server.<br>**Value:** Arbitrary.<br>**Default:** If this logical name is not defined, a file is not created in the [DQS$SERVER.NODES] directory.<br>**Image:** DQS$SERVER |
| DQS$LOG_AREA | **Description:** Specifies the directory in which PrintServer log files are placed.<br>**Value:** The name of the directory.<br>**Default:** The DQS server account directory, [DQS$SERVER].<br>**Image:** DQS$NOTIFIER |

| Server Logical Name | Description, Value, Default and Image |
|---|---|
| DQS$MAX_PRIORITY | **Description:** Specifies the maximum processing priority with which a user can have a job queued on a server.<br>**Value:** The maximum priority.<br>**Default:** The value of the SYSGEN parameter DEFQUEPRI, which is usually 100.<br>**Image:** DQS$SERVER |
| DQS$NOTIFY_CYCLE_ TIME | **Description:** Interval at which the server process scans remote queues to determine when print jobs are complete.<br>**Value:** Time in OpenVMS delta time.<br>**Default:** "0 00:05:00.0" (5 minutes)<br>**Image:** DQS$NOTIFIER |
| DQS$NOTIFY_LOCK | **Description:** Prevents multiple DQS$NOTIFIER processes from running on a homogeneous VMScluster system.<br>**Value:** The name of the lock.<br>**Default:** DQS$NOTIFIER<br>**Image:** DQS$NOTIFIER |
| DQS$PRIORITY | **Description:** Priority at which the server process runs.<br>**Value:** Form decimal number between 0 and 16.<br>**Default:** Priority 4.<br>**Image:** DQS$SERVER |

| Server Logical Name | Description, Value, Default and Image |
|---|---|
| DQS$SERVER_CONFIG_ TXT_FILE | **Description:** Specifies the location of the DQS$SERVER_ CONFIG.TXT file.<br>**Value:** File specification<br>**Default:** SYS$COMMON:[SYSMGR]DQS$SERVER_ CONFIG.TXT<br>**Image:** DQS$SERVER_UPDATE_CONFIG.EXE |
| DQS$SERVER_CONFIG_ DAT_FILE | **Description:** Specifies the location of the DQS$SERVER_ CONFIG.DAT file.<br>**Value:** File specification<br>**Default:** SYS$COMMON:[SYSMGR]DQS$SERVER_ CONFIG.DAT<br>**Image:** DQS$SERVER_UPDATE_CONFIG.EXE, DQS$SERVER |
| DQS$SERVER_ DEFAULT_JOB_NOTE | **Description:** Places a note on the banner page of DQS jobs to indicate that the DQS software was used to print the job.<br>**Value:** Text string that is the message to print on the banner page of print jobs.<br>**Default:** Text string "Print Job processed by DQS V1.3."<br>**Image:** DQS$SERVER |
| DQS$STATUS_*queue-name* | **Description:** Specifies that a status message for a queue is displayed when a QSHOW command of the queue is issued.<br>**Value:** Arbitrary text string that is the status message to be displayed.<br>**Default:** If this logical name is not defined, no status message is displayed.<br>**Image:** DQS$CLIENT, DQS$SMB, DQS$SERVER |

## Client Logical Names

### Modifying Names

DQS software provides the following logical name definitions for the DQS client software. These definitions can be modified by the system manager in the site-specific startup procedure SYS$MANAGER:DQS$SYSTARTUP.COM.

| Client Logical Name | Description, Value, Default, and Image |
|---|---|
| DQS$CLIENT_ DEFAULT_JOB_NOTE | **Description:** Places a note on the banner page of DQS jobs to indicate that the DQS software was used to print the job.<br>**Value:** Text string that is the message to print on the banner page of print jobs.<br>**Default:** Text string "Print Job transported by DQS V1.3."<br>**Image:** DQS$SMB |
| DQS$FORM_OVERRIDE | **Description:** Overrides a client's form definition with its server's definition.<br>**Value:** Arbitrary.<br>**Default:** None.<br>**Image:** DQS$CLIENT |
| DQS$REMOTE_*client-queue* | **Description:** Associates the client queue with a corresponding server system (or with multiple servers) and with a remote queue on each server.<br>**Value:** The server node name and remote queue name in the format *server-node*::*remote-queue*.<br>**Default:** None. This logical name must be defined in order for the queue to function as a DQS client queue.<br>**Image:** DQS$SMB, DQS$CLIENT |
| DQS$STATUS_*queue-name* | **Description:** Specifies that a status message for a queue is displayed when a QSHOW command of the queue is issued.<br>**Value:** Arbitrary text string that is the status message to be displayed.<br>**Default:** If this logical name is not defined, no status message is displayed.<br>**Image:** DQS$CLIENT, DQS$SMB, DQS$SERVER |

# D

# Using DQS With Various Symbionts

## About This Appendix

This appendix describes how jobs print when print symbionts other than the DQS print symbiont (DQS$PRTSMB) are used to process DQS print jobs.

Any symbiont that is built according to the rules specified in the OpenVMS documentation should be able to process DQS print jobs.

## How Various Symbionts Function

### Replacing the OpenVMS Symbiont

Only the OpenVMS print symbiont PRTSMB can be replaced with the DQS print symbiont DQS$PRTSMB. Replacing PRTSMB with the DQS print symbiont causes client-specific job information to be printed on the flag, burst, and trailer pages of DQS print output. This is recommended because the job information includes the client node job number, the client node name, and the time the print job is queued on the client node.

Other symbionts cannot be replaced with the DQS print symbiont. In these cases, the server job number and the time that the print job was queued on the server are printed on the output.

### List of Print Symbionts

The following is a list of various print symbionts:

- **Standard OpenVMS print symbiont (PRTSMB)**. The DQS print symbiont should replace the OpenVMS symbiont.
- **PrintServer 40 symbiont (LPS$SMB)**. The LPS$SMB symbiont cannot be replaced by the DQS print symbiont.
- **LAT printer symbiont (LATSYM)**. The LATSYM symbiont cannot be replaced by the DQS print symbiont.
- **User-modified symbionts**. User-modified symbionts cannot be replaced by the DQS print symbiont. However, users can modify their symbionts to print client-specific information, as described in the section titled Modifying Your Symbiont To Output Client Job Information.
- **User-written symbionts**. User-written symbionts cannot be replaced with the DQS print symbiont. However, users can modify their symbionts to print client-specific information, as described in the section titled Modifying Your Symbiont To Output Client Job Information.

## Modifying Your Symbiont To Output Client Job Information

### Format

DQS remote user information is stored in parameter 8. You can modify your symbiont to use this client-specific print information. The user information has the following format:

```
DQS: nnnnxjob-owner dd-mmm-yyyy hh:mm
```

Where:

| | |
|---|---|
| *nnnn* | Is the job number on the client node. This requires a minimum of 4 characters and a maximum of 10. If the number is less than 4 characters, the job number contains leading spaces. |
| *x* | Is a single nonnumeric code that delimits the job number. |
| *job-owner* | Is the client's node name (which may be trimmed) a double colon ( :: ), and the client's user name. |
| *dd-mmm-yyyy hh:mm* | Is the date and time the job was queued on the client. |

### Example

In this example of a remote user name, each circumflex character ( ^ ) denotes a single space.

```
DQS:^^^12xAPPLES::SHANNON^^^^05-FEB-1994^04:55
```

### Important Parsing Information

Additional considerations for using parameter 8 include the following:

- If a parameter 8 string does not parse according to the format as specified, it is not processed as a P8 string.
- When interpreting the *job-number* field, if the value of the *job-number* is zero, the symbiont uses the local job number associated with the print job by the batch/print system. (This works because the value 0 is currently an illegal job number.)
- When processing the *job-owner* field, a symbiont should treat the entire *job-owner* as a string. The string is terminated by the first space. On OpenVMS systems, the *job-owner* for DQS purposes has the form NODE::USERNAME.

Because of restrictions with trimmed full node names, a symbiont should not rely on the NODE-NAME::USERNAME type syntax, nor should a symbiont interpret the contents of the *job-owner* field.

- The P8 string has a maximum length of 254 characters (not including the null terminator).
- Digital reserves the right to add new information to the end of the P8 string. Print symbionts that parse this string should not depend on the string length nor should they interpret undocumented information that may appear in a particular P8 string.

## Differences from Previous Versions

| Item | In Version 1.2 | In Version 1.3 |
|---|---|---|
| Job Number field | Always 4 characters long. | May be more than 4 characters long. |
| Job Owner field | Number of spaces after the field varies. | Always one space after the field. |

# E

# Files Installed or Modified on the System

## About This Appendix

This appendix describes the files produced or changed during the DQS installation. This information is organized as follows:

| Section | Topic |
|---|---|
| 1 | Files installed on server nodes |
| 2 | Files installed on client-only nodes |
| 3 | Existing files modified by the installation |
| 4 | Accounts and processes added by the installation |
| 5 | Installed images |

Unless otherwise specified, all files are copied from the DQS savesets.

## For More Information

For more information on using and modifying DQS files, refer to the Part II, the System Manager's Guide.

## Files Installed on Server Nodes

Table E–1 lists the files that are installed on server nodes.

**Table E–1 Files Installed on DQS Server Nodes**

| File Name | Description |
|---|---|
| **Added to SYS$MANAGER:** | |
| DQS$CLEANSRV.COM | Deletes lost files in the DQS server account. There are a few conditions where files may be ''lost'' in the directory. |
| DQS$DEFINE.COM | Defines DQS client queues. Invoked by the DQS$IMPORT, DQS$IVP, and the DQS$SYSTARTUP command files. |
| DQS$IMPORT.COM | Allows you to Add another client queue to the current DQS configuration on a client node. |
| DQS$SERVER_CHANGE_ DEFAULT_DEVICE.COM | Allows you to move the default server account directory to a device other than the system device. |
| DQS$SERVER_ CONFIG.TEMPLATE | Allows you to define a set of server queues accessible over the network and to control access to these queues. |
| DQS$SERVER_CONFIG.TXT | Provides information about client node access to server queues. |
| DQS$SERVER_UPDATE_ CONFIG.COM | Allows you to enable changes to the DQS$SERVER_CONFIG.TXT file. |
| DQS$SERVER_UPDATE_ QUEUE_DEFNS.COM | Allows you to update server queues by removing the /RETAIN qualifier for DQS Version 1.3. |
| DQS$SYSTARTUP. TEMPLATE | Allows you to modify server site-specific logical definitions that affect the behavior of the DQS software. |

(continued on next page)

**Table E–1 (Cont.) Files Installed on DQS Server Nodes**

| File Name | Description |
|---|---|
| **Added to SYS$STARTUP:** | |
| DQS$STARTUP.COM | Defines DQS defaults and logical names and invokes the site-specific DQS$SYSTARTUP.COM. This new startup procedure is provided by the installation. Do not edit this file. |
| **Added to SYS$MESSAGE:** | |
| DQS$MSG.EXE | Provides the text for DQS-specific error messages. |

(continued on next page)

**Table E–1 (Cont.) Files Installed on DQS Server Nodes**

| File Name | Description |
|---|---|
| **Added to SYS$SYSTEM:** | |
| DQS$CLIENT.EXE | The user command execution program. Executes the QSHOW, QSET, and QDELETE commands. |
| DQS$NOTIFIER.EXE | A detached process that runs on the server node. Sends print job notification to client users. |
| DQS$PRTSMB.EXE | Replaces the standard print symbiont on the server node. Places DQS-specific information on the FLAG, BURST, and TRAILER pages of print output. Use this symbiont if you do not like how the standard print symbiont handles information on the banner pages. |
| DQS$SERVER.EXE | The DQS server program. Receives incoming requests from client nodes and processes them. Processes all DQS commands (PRINT, QSHOW, QDELETE, and QSET). If notification is requested, also performs notification on the client node. |
| DQS$SERVER_ CONGFIG.DAT | Private DQS server configuration data file. This file is created by the DQS$SERVER_UPDATE_ CONFIG.COM procedure. Do not edit this file. |
| DQS$SERVER_UPDATE_ CONFIG.EXE | Server configuration update program. |
| DQS$SMB.EXE | The DQS symbiont. Transfers print jobs and files from the client system to the server. |
| **Added to SYS$TEST:** | |
| DQS$IVP.COM | Verifies that the DQS software installation completed successfully. |

(continued on next page)

**Table E–1 (Cont.) Files Installed on DQS Server Nodes**

| File Name | Description |
|---|---|
| **Added to SYS$HELP:** | |
| DQS*nnn*.RELEASE_NOTES | The release notes for the DQS product. The *nnn* represents the version number; for example, 013 for Version 1.3. |

## Files Installed on Client-Only Nodes

Table E–2 lists the files installed on a DQS client node.

**Table E–2 Files Installed on DQS Client-Only Nodes**

| File Name | Purpose |
|---|---|
| **Added to SYS$MANAGER:** | |
| DQS$DEFINE.COM | Defines DQS client queues. Invoked by the DQS$IMPORT, DQS$IVP, and the DQS$SYSTARTUP command files. |
| DQS$IMPORT.COM | Allows you to Add another client queue to the current DQS configuration on a client node. |
| DQS$SYSTARTUP. TEMPLATE | Allows you to modify client site-specific logical definitions that affect the behavior of the DQS software. |
| **Added to SYS$STARTUP:** | |
| DQS$STARTUP.COM | Starts the DQS client queues. Invokes the site-specific DQS$SYSTARTUP.COM. Do not edit this file. |
| **Added to SYS$MESSAGE:** | |
| DQS$MSG.EXE | Provides the text for DQS-specific error messages. |

(continued on next page)

**Table E–2 (Cont.) Files Installed on DQS Client-Only Nodes**

| File Name | Purpose |
|---|---|
| **Added to SYS$SYSTEM:** | |
| DQS$CLIENT.EXE | The user command execution program. Executes the QSHOW, QSET, and QDELETE commands. |
| DQS$SERVER.EXE | If print job notification is requested, performs notification on the client node. |
| DQS$SMB.EXE | The DQS symbiont. Transfers print jobs and files from the client system to the server. |
| **Added to SYS$TEST:** | |
| DQS$IVP.COM | Verifies the DQS software installation completed successfully. |
| **Added to SYS$HELP:** | |
| DQS*nnn*.RELEASE_NOTES | The release notes for the DQS product. The *nnn* represents the version number, for example, 013 for Version 1.3. |

## Files Modified by the Installation

### Modified Files

The DQS software modifies the following files during installation:

- SYS$HELP:HELPLIB.HLB — Help library
- SYS$LIBRARY:DCLTABLES.EXE — DCL command definition tables
- DQS$STARTUP.COM_OLD — Older version of the startup command file renamed by the Version 1.3 installation. Defines DQS defaults and all DQS logical names for previous version software.

## Accounts and Processes Added

The DQS installation adds the following accounts and processes:

**Accounts:**

A DQS client/server account, DQS$SERVER, is added to both server and client nodes.

The DQS$SERVER account is given the following OpenVMS privilges:

- TMPMBX
- NETMBX

**Proxies:**

No proxies are added by the installation.

**DECnet objects:**

The following are created on both server and client nodes:

- Object: DQS
- Number: 66
- File/PID: DQS$SERVER.EXE
- User ID: DQS$SERVER
- Proxy access: none

**Note**

The default user ID is DQS$SERVER; the password is randomly generated during the installation procedure.

**Processes:**

On clients, each newly created client queue (that is not a generic or logical queue) results in a process being created. Also, while print job notification is returned to a client user, a temporary process is created.

On servers, each logical link to a client node creates a process. Also, the notification process on a server uses one permanent process.

## Installed Images

### Executable files

The following executables are installed as known images when DQS is started:

| Image | Privileges |
|---|---|
| DQS$NOTIFIER | CMKRNL, ALTPRI, OPER, TMPMBX, NETMBX, WORLD |
| DQS$SERVER | SYSPRV, CMKRNL, ALTPRI, OPER, TMPMBX, NETMBX, WORLD |
| DQS$MSG | |
| DQS$SMB | |

# F

# Sample Installations

## About This Appendix

This appendix contains sample installation dialogs of the DEC Distributed Queuing Service for OpenVMS Systems.

| Section | Topic |
|---|---|
| 1 | Sample PCSI installations on OpenVMS VAX systems |
| 2 | Sample VMSINSTAL installations on OpenVMS VAX and AXP systems |

# Sample PCSI Installation on an OpenVMS System

## Client Installation

Example F–1 shows a sample DQS PCSI client installation on an OpenVMS VAX system. The DQS software resides in a directory pointed to by the PCSI$SOURCE logical name.

### Example F–1 Sample PCSI Client Installation on an OpenVMS System

```
$ DEFINE PCSI$SOURCE NODEA$MKA500: Return
$ PRODUCT INSTALL DQS Return

The following product has been selected:
DEC VAXVMS DQS V1.3

Do you want to continue? [YES] Return

*** DEC VAXVMS DQS V1.3: DEC VAXVMS DQS V1.3 for OpenVMS

    Copyright © 1994 Digital Equipment Corporation
    Digital Equipment Corporation

    This product uses the PAKs:  DEC DQS V1.3.

Do you want all the default values for this product? [YES]  Return

Do you want to view the values? [NO]  Return

Execution phase starting
The following product will be installed:
DEC VAXVMS DQS V1.3
Estimated space information for volume DISK$OPENVMS060
710 required; 64060 available; 63350 net
Portion Done: 0%...10%...20%...40%...60%...70%...90%...100%
The following product has been installed:
DEC VAXVMS DQS V1.3
Start of test procedure
End of test procedure; completed with no errors

*** DEC VAXVMS DQS V1.3: DEC VAXVMS DQS V1.3 for OpenVMS

    [DQS$SERVER] directory is created in SYS$COMMON: area.
    DQS$SERVER account is created.

    This product requires the following SYSGEN parameters:
        GBLPAGES minimum 600
    This product requires the following SYSGEN parameters:
        GBLSECTIONS minimum 5
```

(continued on next page)

**Example F–1 (Cont.) Sample PCSI Client Installation on an OpenVMS System**

```
    The client installation is complete.
    DQS network object is created.
    Perform post-installation before using DQS.
$
```

## PCSI Installation of Server Software

Example F–2 shows a sample DQS server installation on an OpenVMS VAX system. The DQS software resides in a directory pointed to by the PCSI$SOURCE logical name.

### Example F–2 Sample PCSI Server Installation on OpenVMS VAX System

```
$ DEFINE PCSI$SOURCE NODEA$MKA500: Return
$ PRODUCT INSTALL DQS Return

The following product has been selected:
DEC VAXVMS DQS V1.3     [Available]

Do you want to continue? [YES] Return

*** DEC VAXVMS DQS V1.3: DEC VAXVMS DQS V1.3 for OpenVMS

    Copyright © 1994 Digital Equipment Corporation

    Digital Equipment Corporation

    This product uses the PAKs:  DEC DQS V1.3.

    Do you want all the default values for this product? [YES] NO

        *** DEC VAXVMS VMS V6.1     [Available]

    You need to complete the pre-installation tasks to install DQS software.
    Do you want to continue? [YES] Return

The DQS client software will be installed automatically on this node.

Do you want to install the DQS server software on this node [NO] YES Return

Do you want to view the values? [NO]  Return

Execution phase starting
The following product will be installed:
DEC VAXVMS DQS V1.3
Estimated space information for volume DISK$OPENVMS060
1018 required; 64248 available; 63230 net
Portion Done: 0%...10%...20%...30%...40%...50%...70%...80%...90%...100%
The following product has been installed:
DEC VAXVMS DQS V1.3
Start of test procedure
End of test procedure; completed with no errors

*** DEC VAXVMS DQS V1.3: DEC VAXVMS DQS V1.3 for OpenVMS

    [DQS$SERVER] directory is created in SYS$COMMON: area.

    DQS$SERVER account is created.

    This product requires the following SYSGEN parameters:
        GBLPAGES minimum 600

    This product requires the following SYSGEN parameters:
        GBLSECTIONS minimum 5
```

(continued on next page)

**Example F–2 (Cont.) Sample PCSI Server Installation on OpenVMS VAX System**

```
    The client installation is complete.
    The server installation is complete.
    DQS network object is created.
    Perform post-installation before using DQS.
$
```

# Sample VMSINSTAL Installation on an OpenVMS AXP System

## Client Installation

Example F–3 shows a sample installation of DQS client software on an OpenVMS AXP system, using the VMSINSTAL utility.

### Example F–3 Sample VMSINSTAL Client Installation on an OpenVMS AXP System

```
Username: SYSTEM [Return]
Password: [Return]

$ @SYS$UPDATE:VMSINSTAL DQS013 MKA500: [Return]

OpenVMS AXP Software Product Installation Procedure V6.1

It is 22-MAR-1994 at 13:57.

Enter a question mark (?) at any time for help.

%VMSINSTAL-W-ACTIVE, The following processes are still active:
        DQS$NOTIFIER

* Do you want to continue anyway [NO]? YES [Return]
* Are you satisfied with the backup of your system disk [YES]? [Return]

Please mount the first volume of the set on  MKA500:.
* Are you ready? YES [Return]
%MOUNT-I-MOUNTED, DQS mounted on _AXPPD$MKA500:

The following products will be processed:

  DQS V1.3

        Beginning installation of DQS V1.3 at 14:06

%VMSINSTAL-I-RESTORE, Restoring product save set A ...
%VMSINSTAL-I-RELMOVED, Product's release notes have been moved to SYS$HELP.

        The DQS client will be installed automatically on this node.

* Do you want to install the DQS server software on this node [NO]? [Return]
* Do you want to run the IVP after the installation [YES]? [Return]
* Do you want to purge files replaced by this installation [YES]? [Return]

        Product:      DQS
        Producer:     DEC
        Version:      1.3
        Release Date: 7-MAR-1994

* Does this product have an authorization key registered and loaded? YES [Return]
```

(continued on next page)

**Example F–3 (Cont.) Sample VMSINSTAL Client Installation on an OpenVMS AXP System**

```
        No further questions.

%VMSINSTAL-I-RESTORE, Restoring product save set B ...
%VMSINSTAL-I-ACCOUNT, This installation updates an ACCOUNT named DQS$SERVER.
%UAF-I-MDFYMSG, user record(s) updated
%VMSINSTAL-I-ACCOUNT, This installation updates an ACCOUNT named DQS$SERVER.
%UAF-I-MDFYMSG, user record(s) updated
%VMSINSTAL-I-ACCOUNT, This installation updates an ACCOUNT named DQS$SERVER.
%UAF-I-MDFYMSG, user record(s) updated

        Installing DQS CLIENT software . . .

     Checksum file updated last by WEBER on 22-MAR-1994 09:40:50.30

%NET$CONFIGURE-I-VERCHECKSUM, verifying checksums
%NET$CONFIGURE-I-MODCHECKSUM, checksumming NCL management scripts modified by
NET$CONFIGURE
%NET$CONFIGURE-I-CONFIGCOMPLETED, DECnet/OSI for OpenVMS configuration completed
     Checksum file updated last by SYSTEM on 22-MAR-1994 14:08:08.73

%NET$CONFIGURE-I-VERCHECKSUM, verifying checksums
%NET$CONFIGURE-I-MAKEACCOUNT, this procedure creates user account DQS$SERVER
$NCL @SYS$SYSROOT:[SYSMGR]NET$APPLICATION_STARTUP.NCL-TEMP;

Node 0 Session Control Application DQS
at 1994-03-22-14:08:16.150-05:00I0.649

Characteristics

    Addresses                       =
       {
          number = 66
       }
    Client                          = <Default value>
    Incoming Alias                  = False
    Incoming Proxy                  = False
    Outgoing Alias                  = False
    Outgoing Proxy                  = False
    Node Synonym                    = False
    Image Name                      = SYS$SYSTEM:DQS$SERVER.EXE
    User Name                       = "DQS$SERVER"
    Incoming OSI TSEL               = <Default value>

$vef = F$verify(vef)
%NET$CONFIGURE-I-MODCHECKSUM, checksumming NCL management scripts modified by
NET$CONFIGURE
%NET$CONFIGURE-I-CONFIGCOMPLETED, DECnet/OSI for OpenVMS configuration completed
%VMSINSTAL-I-ACCOUNT, This installation updates an ACCOUNT named DQS$SERVER.
%UAF-I-MDFYMSG, user record(s) updated

        DQS network object is created.
```

(continued on next page)

## Example F–3 (Cont.) Sample VMSINSTAL Client Installation on an OpenVMS AXP System

```
        ****************************************************************
        *           Instructions for completing DQS installation       *
        ****************************************************************

        o Please follow the post installation instructions which have
          been outlined in the DQS V1.3 Installation Guide.

        o Please be sure to edit your OpenVMS System Startup Procedure
          and add the following line to invoke the DQS Startup Procedure:

                $ @SYS$STARTUP:DQS$STARTUP.COM

        o Once you have completed the DQS post installation procedures
          you can invoke the DEC DQS installation verification procedure
          to test your customizations. To invoke this file start DQS
          and type:

                $ @SYS$TEST:DQS$IVP.COM

        o This installation has updated the DCL Tables. Therefore
          users must log out and then back in, in order to use the
          QSHOW, QDELETE, and QSET commands for displaying, deleting,
          and modifying jobs on the remote server queues.

        o Please read the DQS V1.3 release notes located in the file
          SYS$HELP:DQS013.RELEASE_NOTES .

%VMSINSTAL-I-MOVEFILES, Files will now be moved to their target directories...

Beginning of the DQS V1.3-0 Installation Verification Procedure.

  Copyright Digital Equipment Corporation 1994. All rights reserved.

  Restricted Rights: Use, duplication, or disclosure by the U.S. Government
  is subject to restrictions as set forth in subparagraph (c)(1)(ii) of
  DFARS 252.227-7013, or in FAR 52.227-19, or in FAR 52.227-14 Alt. III, as
  applicable.

  This software is proprietary to and embodies the confidential technology
  of Digital Equipment Corporation. Possession, use, or copying of this
  software and media is authorized only pursuant to a valid written license
  from Digital or an authorized sublicensor.

%DQS-I-QUESUCCESS IVP test successful
End of the DEC DQS for OpenVMS V1.3-0 Installation Verification Procedure

        Installation of DQS V1.3 completed at 14:08

    Adding history entry in VMI$ROOT:[SYSUPD]VMSINSTAL.HISTORY

    Creating installation data file: VMI$ROOT:[SYSUPD]DQS013.VMI_DATA
```

(continued on next page)

**Example F–3 (Cont.) Sample VMSINSTAL Client Installation on an OpenVMS AXP System**

```
        VMSINSTAL procedure done at 14:08
$
```

## Server Installation

Example F–4 shows a sample installation of a DQS server on an OpenVMS VAX system using the VMSINSTAL installation procedure.

### Example F–4 Sample VMSINSTAL Server Installation on OpenVMS VAX System

```
Username: SYSTEM [Return]
Password: [Return]

$ @SYS$UPDATE:VMSINSTAL DQS013 MUA0: [Return]

        OpenVMS VAX Software Product Installation Procedure T6.1-5Y3

It is 18-MAR-1994 at 09:51.

Enter a question mark (?) at any time for help.

%VMSINSTAL-W-ACTIVE, The following processes are still active:
        _FTA13:
        DAVIES
        _FTA6:
* Do you want to continue anyway [NO]? YES [Return]
* Are you satisfied with the backup of your system disk [YES]? [Return]

The following products will be processed:

  DQS V1.3


        Beginning installation of DQS V1.3 at 9:53

%VMSINSTAL-I-RESTORE, Restoring product save set A ...

%VMSINSTAL-I-RELMOVED, Product's release notes have been moved to SYS$HELP.

        The DQS client will be installed automatically on this node.

* Do you want to install the DQS server software on this node [NO]? YES

                        *** PLEASE NOTICE ***

        If you are upgrading DQS server software from an earlier version
        to DQS V1.3, the earlier version of the DQS server software must
        be running. This allows the DQS V1.3 installation procedure to read
        the existing DQS server configuration information and create a
        new DQS V1.3 server configuration file.  Otherwise, after
        installation, you must manually update your DQS V1.3 server
        configuration file.

        The directory and name of the DQS V1.3 configuration file is:

                SYS$MANAGER:DQS$SERVER_CONFIG.TXT

        Digital recommends that you review the new DQS V1.3 server
        configuration file after this installation. Refer to the section
        on server management in the DQS System Manager's Guide for more
        information.
```

(continued on next page)

## Example F–4 (Cont.) Sample VMSINSTAL Server Installation on OpenVMS VAX System

```
        The DQS client configuration is not automatically updated. Refer
        to the section on client management in the DQS System Manager's
        Guide for more information.

* Do you want to proceed [YES]? RETURN
* Do you want to run the IVP after the installation [YES]? RETURN
* Do you want to purge files replaced by this installation [YES]? RETURN

    Product:      DQS
        Producer:     DEC
        Version:      1.3
        Release Date: 7-MAR-1994
* Does this product have an authorization key registered and loaded? YES

        No further questions.

%VMSINSTAL-I-RESTORE, Restoring product save set B ...
%VMSINSTAL-I-ACCOUNT, This installation creates an ACCOUNT named DQS$SERVER.
%UAF-I-ADDMSG, user record successfully added
%UAF-I-RDBADDMSGU, identifier DQS$SERVER value [000300,000311] added to rights d
atabase

        DQS$SERVER account is created.

%VMSINSTAL-I-ACCOUNT, This installation updates an ACCOUNT named DQS$SERVER.
%UAF-I-MDFYMSG, user record(s) updated
%VMSINSTAL-I-ACCOUNT, This installation updates an ACCOUNT named DQS$SERVER.
%UAF-I-MDFYMSG, user record(s) updated
%VMSINSTAL-I-ACCOUNT, This installation updates an ACCOUNT named DQS$SERVER.
%UAF-I-MDFYMSG, user record(s) updated
%VMSINSTAL-I-SYSDIR, This product creates system disk directory  SYS$COMMON:[DQS
$SERVER].

        [DQS$SERVER] directory is created in SYS$COMMON: area.

        Installing DQS SERVER software . . .


        Installing DQS CLIENT software . . .

     Checksum file updated last by CHO on 17-MAR-1994 09:48:25.61

%NET$CONFIGURE-I-VERCHECKSUM, verifying checksums
%NET$CONFIGURE-I-MODCHECKSUM, checksumming NCL management scripts modified by NE
T$CONFIGURE
%NET$CONFIGURE-I-CONFIGCOMPLETED, DECnet/OSI for OpenVMS configuration completed
     Checksum file updated last by CHO on 18-MAR-1994 09:56:48.11
%NET$CONFIGURE-I-VERCHECKSUM, verifying checksums
%NET$CONFIGURE-I-MAKEACCOUNT, this procedure creates user account DQS$SERVER
$NCL @SYS$SYSROOT:[SYSMGR]NET$APPLICATION_STARTUP.NCL-TEMP;
```

(continued on next page)

**Example F–4 (Cont.) Sample VMSINSTAL Server Installation on OpenVMS VAX System**

```
Node 0 Session Control Application DQS
at 1994-03-18-09:57:00.327-05:00I0.289


Node 0 Session Control Application DQS
at 1994-03-18-09:57:00.947-05:00I0.289

Characteristics
    Addresses                       =
       {
          number = 66
       }
    Client                          = <Default value>
    Incoming Alias                  = False
    Incoming Proxy                  = False
    Outgoing Alias                  = False
    Outgoing Proxy                  = False
    Node Synonym                    = False
    Image Name                      = SYS$SYSTEM:DQS$SERVER.EXE
    User Name                       = "DQS$SERVER"
    Incoming OSI TSEL               = <Default value>

$vef = F$verify(vef)
%NET$CONFIGURE-I-MODCHECKSUM, checksumming NCL management scripts modified by NE
T$CONFIGURE
%NET$CONFIGURE-I-CONFIGCOMPLETED, DECnet/OSI for OpenVMS configuration completed
%VMSINSTAL-I-ACCOUNT, This installation updates an ACCOUNT named DQS$SERVER.
%UAF-I-MDFYMSG, user record(s) updated

        DQS network object is created.

      ****************************************************************
      *          Instructions for completing DQS installation        *
      ****************************************************************


        o Please follow the post installation instructions which have
          been outlined in the DQS V1.3 Installation Guide.

        o Please be sure to edit your OpenVMS System Startup Procedure
          and add the following line to invoke the DQS Startup Procedure:

                $ @SYS$STARTUP:DQS$STARTUP.COM

        o Once you have completed the DQS post installation procedures
          you can invoke the DEC DQS installation verification procedure
          to test your customizations. To invoke this file start DQS
          and type:

                $ @SYS$TEST:DQS$IVP.COM
```

(continued on next page)

**Example F–4 (Cont.) Sample VMSINSTAL Server Installation on OpenVMS VAX System**

```
        o This installation has updated the DCL Tables. Therefore
          users must log out and then back in, in order to use the
          QSHOW, QDELETE, and QSET commands for displaying, deleting,
          and modifying jobs on the remote server queues.

        o Please read the DQS V1.3 release notes located in the file
          SYS$HELP:DQS013.RELEASE_NOTES .

%VMSINSTAL-I-MOVEFILES, Files will now be moved to their target directories...

Beginning of the DQS V1.3 Installation Verification Procedure.

  Copyright Digital Equipment Corporation 1994. All rights reserved.

  Restricted Rights: Use, duplication, or disclosure by the U.S. Government
  is subject to restrictions as set forth in subparagraph (c)(1)(ii) of
  DFARS 252.227-7013, or in FAR 52.227-19, or in FAR 52.227-14 Alt. III, as
  applicable.

  This software is proprietary to and embodies the confidential technology
  of Digital Equipment Corporation. Possession, use, or copying of this
  software and media is authorized only pursuant to a valid written license
  from Digital or an authorized sublicensor.


%DQS-I-QUESUCCESS IVP test successful
End of the DEC DQS for OpenVMS V1.3 Installation Verification Procedure

        Installation of DQS V1.3 completed at 09:57


        VMSINSTAL procedure done at 09:57

$
```

# G

# DQS Installation Messages

## About This Appendix

This appendix describes error messages that you might encounter when you install or verify the installation of DQS software. Suggestions for error recovery are included, as well as instructions for reporting problems to Digital for irrecoverable errors. Refer to the following sections for information on these topics:

| Section | Topic |
|---|---|
| 1 | Installation error messages |
| 2 | IVP error messages |
| 3 | Error reporting |

## Message Format

All DQS error messages begin with this prefix:

`%DQS-`*`l`*`-`*`ident`*`,`

The messages in this appendix are listed alphabetically by their message *ident* (or identification).

## For More Information

Refer to the *OpenVMS System Messages and Recovery Procedures Reference Manual* for an explanation of the following:

- The format of OpenVMS system messages and their severity levels
- Error messages beginning with %PCSI and possible solutions to these problems

For a description of any errors returned by the License Management Facility (LMF), refer to the *OpenVMS License Management Utility Manual*.

## Installation Error Messages

This section describes error messages that you might encounter while you are installing DQS software.

### Message Section

NOSPACE, there is not enough free space on the system disk to install DQS V*n.n*

**Explanation:** There is not enough free space on the system disk for the installation.

**User Action:** Delete some files and try again.

VERSION, DQS V*n.n* requires OpenVMS Version V*n.n* or later to install

**Explanation:** The installed version of the OpenVMS operating system is not the prerequisite one.

**User Action:** Install or upgrade to the appropriate version of the OpenVMS operating system and reinstall the DQS software.

## IVP Error Messages

This section describes error messages that you might encounter when you run the DQS Installation Verification Procedure (IVP).

### IVP Messages

FILENOTFOUND *file-name* is not found

**Explanation:** The IVP could not find the file *file-name*. The *file-name* may be one of the following:

- SYS$MANAGER:DQS$CLIENT.EXE
- SYS$MANAGER:DQS$SMB.EXE
- SYS$MANAGER:DQS$SERVER.EXE
- SYS$MANAGER:DQS$MSG.EXE

Refer to Appendix E for information about these files.

**User Action:** Reinstall the DQS software. If you still get this message, report your problem to Digital.

## Error Reporting

### Reporting

If an error occurs while you are using the DQS software and you have followed the suggestions in this section and the troubleshooting section in the Part II and still have reason to believe that the error is a problem with the DQS sofware, report the problem to Digital.

### Actions

Depending on the nature of the problem and the type of support you have, you can take one of the following actions:

- Call Digital if your software contract or warranty agreement entitles you to telephone support.
- Determine whether the problem is a result of a defective media. If so, contact your Digital Services representative for a replacement copy. If the requirements for installing the DQS software have been met, and the media is not defiective, then submit a Software Performance Report (SPR). When you prepare to submit an SPR, please do the following:
  - Provide as complete a description of the problem as possible, trying not to overlook any details. Include in the description the version numbers of the DQS software and operating system (both server and client).
  - Reduce the problem to as simple a case as possible, preferably to a single command.
  - Include listings of all output from the DQS software (if any) and any relevant SYS$MANAGER:OPERATOR.LOG entries and NETSERVER.LOG files. The DQS$NETSERVER.LOG files are located in the DQS server account default directory.
  - Provide the following:
    * A SHOW LOGICAL listing of all the DQS logical definitions on both the server and client nodes
    * A SHOW QUEUE/FULL/ALL listing of the queue on both the server and client nodes
    * A SHOW PROCESS/ALL listing of the server DQS process

- Fill out and submit a Reader's Comments form if the problem has to do with the DQS documentation. There are Reader's Comments forms at the back of each manual. Use the form from the manual in which you found the error. Include the section and page number.

Review the current Software Product Description (SPD) and Warranty Addendum. If you encounter a problem during the warranty period, report the problem as indicated above or follow alternate instructions provided by Digital for reporting SPD nonconformance problems.

# Glossary

### burst page

A page printed between print jobs or copies of print files, if requested. Useful for separating print jobs manually. The burst page contains client-specific information if the remote queue on the server is controlled by the DQS software. Otherwise, the burst page contains server-specific information.

### client node

Nodes whose users can print files on printing devices that are attached to remote server nodes.

### client queue

DQS queue on the client node that directs print jobs to the associated remote queue on the server system.

### client software

DQS software that is installed on all nodes, and must be configured if the node is to act as a client node.

### DECnet

Digital's networking protocol.

### DECnet long name

Long node names allowed by the DECnet/OSI for OpenVMS, Version 5.6 software (Phase V).

### device control library

A library that contains a series of text modules that can be sent to the device associated with a queue, to affect the behavior of that device. DQS printing uses the device control libraries of the server node.

### DQS

Acronym for Distributed Queuing Service.

### execution queue

The print queue associated with a specific physical printing output device.

### flag page

An optional page that precedes each file or job that is printed. The flag page contains information about the printer and the print job. If DQS software controls the remote queue on the server, then the flag page contains client-specific information. Otherwise, the flag page contains server-specific information.

### full name

For DQS purposes, the DECnet client's or server's remote node name comprised of 1 to 150 characters. The DECnet full name has the following format:

```
NamespaceNickname:.DirectoryPath.NodeObject
```

The node full name must begin with the namespace nickname and a colon (:) and the directory path must begin with a dot (.). The full name can be any combination of letters, digits, and certain punctuation characters from the OSI Latin-1 character set. Some other characters are allowed as long as they are enclosed in quotation marks. For a list of all allowable characters, refer to *DECnet/OSI DECdns Management, Appendix A*.

### generic queue

The logical printer queue that accepts the print job from the user and directs it to the appropriate execution queue.

### logical queue

*See* **generic queue**.

### PCF

Acronym for product configuration file.

### PCSI

Acronym for POLYCENTER Software Installation utility.

### print job

A print request of one or more files and associated print requirements.

### print queue

The queue in which a print job waits to be printed.

### server node

Nodes in your network that have an attached output device (for examples, printers) that you can make available to the users on remote client nodes in your network.

### server software

DQS software that must be installed and configured on a node for the node to act as a server node.

### remote queue

DQS queue on the server node that directs print jobs from the remote client to the attached printer.

### trailer page

An optional print page that follows each file printed. The trailer page is often printed even if the print job fails, and contains messages to the user explaining the reason for the print error. The trailer page contains client-specific information if the DQS software controls the remote queue on the server. Otherwise, the trailer page contains server-specific information.

# Index


## A

Access rights, 7–29
Accounting, 8–17
  for client access to servers, 8–18
ALLOW_NODE keyword, 8–14
ASSIGN/QUEUE command
  for defining logical queues, 9–10
AUTOGEN utility, 12–15

## B

Banner page note, 8–22, 9–20
Batch jobs
  and printing, 7–32

## C

Client
  as a VMScluster system, 7–3
    connectivity requirements, 7–4
Client access
  checking, 8–17
Client logical names, C–6
  defining, 9–10, 15–3
Client node
  configuring, 7–3, 7–20
  configuring in a VMScluster environment, 9–15
  definition, 1–3, 2–2, 7–3, 12–4
Client queue, 1–3
  advanced methods of creating, 9–6
  changing definition temporarily, 9–12
  creating with DQS$DEFINE.COM file, 9–6
Client queue (cont'd)
  creating with DQS$IMPORT.COM file, 9–4
  definition, 7–3
  deleting, 9–14
  finding status, 10–10
  setting up a LAT port, 9–18
  spooling to, 9–18
Client software
  definition, 7–3
  upgrading, 15–10
Command interpreter
  for DQS commands, 7–11
Commands
  enabling after installation, 15–6, 15–12
  PRINT, 2–2
  QDELETE, 4–2
  QSET, 5–2
  QSHOW, 3–2
  summary, 7–6
Configuration
  for client nodes, 7–3, 7–20
  for server nodes, 7–3, 7–20
  restrictions, 7–20
Connectivity requirements
  for DQS, 7–4

## D

Daisy-chained queues
  benefits, 7–22
  creating loops, 7–24
  defining, 8–41
  definition, 7–22
  function, 7–23

Daisy-chained queues (cont'd)
setting up, 8–44
DECnet Phase IV
installation requirements, 11–4
postinstallation steps, 15–4
use with the DQS software, 7–4
DECnet/OSI Phase V
installation requirements, 11–4
postinstallation steps, 15–5
use with the DQS software, 7–4
DENY_NODE keyword, 8–13
Disk space requirements, 12–11
Distribution kit
OpenVMS AXP systems, 12–2
OpenVMS VAX systems, 12–3
DQS messages, A–1
DQS$ACCOUNTING_BY_SYSTEM logical name, 8–5, 8–17, 8–43
function, 8–19
DQS$CLEANSRV.COM file
function in DQS printing, 8–4
DQS$CLIENT program
for synchronizing forms, 9–16
function, 7–11
DQS$DEFINE.COM file, 8–41, 9–6
creating client queues with, 9–6
parameters for invoking, 9–7
DQS$FORM_OVERRIDE logical name
function in DQS environment, 9–17
DQS$IDLE_TIME logical name, 8–5, 8–43
defining, 8–33
function, 8–33
DQS$IMPORT.COM file
connectivity requirements, 9–4
for creating client queues, 9–4
using, 9–4
DQS$LOG_ACCESS logical name, 8–5, 8–17, 8–42
function, 8–18
DQS$LOG_AREA logical name, 8–30, 8–43
function, 8–30
DQS$MAX_PRIORITY logical name, 8–5, 8–43
function, 8–32
DQS$MOVE_DEFDIR.COM file
moving server account, 8–35
DQS$NOTIFIER program
defining the scanning interval, 8–28
function, 7–13, 7–15
DQS$NOTIFY_CYCLE_TIME logical name, 8–5, 8–43
defining, 8–28
function, 8–28
DQS$NOTIFY_LOCK logical name, 8–37, 8–44
defining, 8–37
DQS$PRIORITY logical name, 8–5, 8–43
defining, 8–31
function, 8–31
DQS$PRTSMB symbiont, 8–43
enabling, 8–26
function, 7–12, 7–16
function with other symbionts, D–2
replacing PRTSMB, 7–16
DQS$REMOTE logical name
changing definition, 9–12
function in generic queues, 9–9
function in logical queues, 9–10
DQS$SERVER account
setting up during installation, 14–11
DQS$SERVER program
function, 7–11, 7–12
function on client, 7–18
function on server, 7–18
[DQS$SERVER.NODES] subdirectory
for accounting, 8–18
of server account, 8–18
DQS$SERVER_CHANGE_DEFAULT_DEVICE.COM file, 8–35
DQS$SERVER_CONFIG.TXT file, 8–2
denying client access, 8–42
format, 8–9
granting client access, 8–42
setting up remote queues, 8–42
DQS$SERVER_UPDATE_QUEUE_DEFNS.COM file, 15–12
DQS$SMB symbiont
function, 7–11, 7–16

DQS$STARTUP.COM file, 11–8
editing to synchronize forms, 9–17
for defining client logical names, 9–2
for starting client queues, 9–2
invoking, 15–5, 15–11
on a VMScluster system, 7–3
DQS$STARTUP.COM_OLD file, 11–8
DQS$STATUS logical name, 8–43
function, 8–24
in error reporting, 7–27
DQS$SYSTARTUP.COM file
editing, 8–2
for defining server logical names, 8–3, 8–42

## E

Error messages *see also* Messages
Error conditions, 7–25
function of DQS$STATUS logical name, 7–27
in DQS printing, 7–25
printing, 7–25
Error messages
*see also* Messages
during installation, G–2
during installation verification, G–3
for daisy-chained queues, 7–24
for QDELETE command, 4–4
for QSET command, 5–6
PRINT command, 2–11
QSHOW command, 3–7, 10–14
with parameter 8, 7–19
Error reporting
procedures, G–4

## F

Files
installed on DQS client nodes, E–6
installed on DQS server nodes, E–2
modified by the installation procedure, E–8
purging, 14–9
Forms
displaying, 3–5
function, 7–17
requirements for printing with, 2–7
Forms synchronization
editing DQS$SYSTARTUP.COM file, 9–17
for DQS jobs, 7–17
in DQS environments, 9–16
performing, 9–16

## G

Generic queues
configuring, 9–10
defining, 9–8
definition, 9–8
Global pages
changing system values, 12–16
checking system values, 12–15
Global sections
changing system values, 12–16
checking system values, 12–15

## H

Hardware requirements
HELP command, 1–4

## I

Installation Verification Procedure *see* IVP
Installation procedure
accounts added by, E–9
checking progress, 13–13
errors, 13–21
files added on client nodes, E–6
files added on server nodes, E–2
files modified, E–8
first-time considerations, 11–7
following progress of VMSINSTAL, 14–11
getting help from DCL, 13–6
getting help from MOTIF, 13–18
invoking from DCL, 13–3
invoking VMSINSTAL, 14–4

Installation procedure (cont'd)
on a VMScluster system, 7–3, 12–5
OpenVMS tailoring classes for, 11–3
planning, 12–4
processes added, E–9
purging files replaced by, 14–9
sample dialogs, F–1
starting from MOTIF, 13–14
stopping from DCL, 13–10
stopping from MOTIF, 13–20
stopping with VMSINSTAL, 14–6
system management procedures, 12–15
upgrade considerations, 11–8
what you need to know, 11–1
with PCSI, 13–3
with VMSINSTAL, 14–1
Installation requirements, 11–1
accounts, 12–6
disk space, 12–11
hardware, 11–3
licensing, 11–5
operating system, 11–3
PCSI, 11–7
prerequisite products, 11–4
privileges, 12–6
process account quotas, 12–9
software, 11–3
time, 11–10
IVP
for troubleshooting, 10–2
running separately, 15–5
troubleshooting, 15–5

## L

LAT port, 9–18
License Management Facility
See LMF
License PAK, 11–5, 12–14
License Product Authorizatin Key
See License PAK
Licensing requirements, 11–5
LMF, 11–5, 12–14
Log files, 8–30
specifying the location, 8–30
Logical device, 6–1
Logical link
specifying duration in DQS printing, 8–33
Logical links
setting, 7–20
Logical name
summary
for servers, 8–42
Logical queues
defining, 9–11
definition, 9–10

## M

Messages, B–4
format with DQS software, A–1, B–2
from notification service, A–2, B–4
severity levels, A–1, B–2
showing print status, 2–6
Moving server account, 8–35

## N

NETSERVER.LOG file, 7–25
checking client access, 8–17
for troubleshooting, 10–10
Network connections
setting, 7–20
Network connectivity
requirements for DQS$IMPORT.COM file, 9–4
Networking software
installation requirements, 11–4
/NOTE
DQS default note, 8–22, 9–20
Notification service
connectivity requirements, 7–4
DQS$SERVER program, 7–15, 7–18
function, 7–13, 7–15
messages, 7–13, 7–15, A–2, B–4
/NOTIFY qualifier, 7–13
troubleshooting, 10–6

/NOTIFY qualifier, 7–13

## O

Operator Communication Process *see* OPCOM
OPCOM
  for troubleshooting, 10–10
  function, 7–25
OpenVMS tailoring classes, 11–3
Operating system requirements, 11–3
OPERATOR.LOG file
  for troubleshooting, 10–10

## P

PAK
  *See* License PAK
Parameter 8, 7–19
  format of information, 7–19, D–3
  modifying, D–3
Parameters
  for DQS$DEFINE.COM invocation, 9–7
PCSI utility
  DCL help, 13–6
  DCL interface, 13–3
  installing server software, 13–7
  invoking from MOTIF, 13–14
  MOTIF help, 13–18
  MOTIF interface, 13–14
  registering products, 11–7, 12–18
  stopping from DCL, 13–10
  stopping from MOTIF, 13–20
Postinstallation
  customizing client software, 15–3, 15–10
  customizing server software, 15–2, 15–9
  defining server queues, 15–2
  editing system startup file, 15–4, 15–11
  for OpenVMS clusters members, 15–13
  moving server account directory, 15–2, 15–7
  running the IVP separately, 15–5, 15–12
  starting DQS software, 15–5, 15–11
  updating server queues, 15–12
  upgrading client software, 15–10, 15–11
Postinstallation (cont'd)
  upgrading DQS server, 15–7
  upgrading server software, 15–11
  verifying queue definitions, 15–7
PRINT command, 2–2, 7–5
  /AFTER qualifier, 2–8
  /DELETE qualifier, 2–8
  error messages, 2–11
  /FORM qualifier, 2–8
  function, 7–6, 7–7
  /HEADER qualifier, 2–8
  /HOLD qualifier, 2–9
  network connectivity requirements, 1–3
  /NOIDENTIFY qualifier, 2–6
  notification completion, 1–3
  notification of completion, 7–13
  /NOTIFY qualifier, 1–3, 2–9, 2–10, 4–3, 7–13
  /OPERATOR qualifier, 2–9
  /PARAMETERS qualifier, 2–9
  /PRIORITY qualifier, 2–9
  qualifiers, 2–8, 7–6
  /SETUP qualifier, 2–9
Print job
  deleting from a queue, 4–2
  displaying entry number, 5–3
  entry number, 2–6
    displaying, 4–3
  modifying, 5–3
  specifying maximum priority, 8–32
  status messages, A–2, B–4
  troubleshooting, 10–9
Print Symbiont
  function, 8–26
Printing process, 7–7
Privileges
  for installing DQS software, 12–6
  for running DQS$CLIENT program, 9–16
  for using DQS$IMPORT.COM file, 9–4
  modifying, 12–7
Process account quotas
  modifying, 12–10
  required for installation, 12–9

Processes
client requirements, 7–30, 9–3
created in server account, 7–18
DQS$NOTIFIER
scanning interval, 8–28
priority for server process, 8–31
reducing amount on client node, 9–3, 9–8, 9–10
required by DQS software, 11–6
restarting server processes, 8–5
server requirements, 7–20, 7–30
Product Authorization Key
*See* License PAK
PRODUCT INSTALL command, 13–3
qualifiers, 13–5
PRTSMB print symbiont, 7–12

## Q

QDELETE command, 4–2
error messages, 4–4
QDELETE/ENTRY command
function, 7–6, 7–9
qualifiers, 7–6
QSET command, 5–2
/AFTER qualifier, 5–4
error messages, 5–6
/FORM qualifier, 5–4
/HEADER qualifier, 5–4
/HOLD qualifier, 5–4
/NODELETE qualifier, 5–4
/PARAMETERS qualifier, 5–5
/PRIORITY qualifier, 5–5
/REQUEUE qualifier, 5–5
QSET/ENTRY command
function, 7–6, 7–9
qualifiers, 7–6
QSHOW command, 3–2, 10–9
enabling status messages, 8–24
error messages, 3–7, 10–14
examining output, 10–9
finding status, 10–10, 10–11, 10–13
for troubleshooting, 10–9
function, 7–6, 7–9
qualifiers, 7–6
QSHOW command (cont'd)
/QUEUE qualifier, 3–5

## R

Release notes
accessing with PCSI utility, 13–2
accessing with VMSINSTAL, 14–2
Remote queue, 1–3
definition, 7–3
/RETAIN=ALL characteristic, 7–13
/RETAIN=ALL characteristic, 7–13

## S

Software Performance Report>*see* SPR
Software requirements
Security requirements, 7–29
Server
as a VMScluster system, 7–3
connectivity requirements, 7–4
logical link requirements, 7–20
Server account
definition, 7–18
directory function, 7–18
[DQS$SERVER.NODES] subdirectory, 8–18
file deletion, 7–18
stray files, 8–4
Server account directory
moving to a new device, 8–35
moving to new device, 15–2, 15–7
Server logical names, C–3
defining, 8–3, 8–42, 15–2
Server node
allowing access to specific clients, 8–14
configuring, 7–3, 7–20
definition, 1–3, 2–2, 7–2, 12–4
enabling, 13–7
installation requirements, 11–8
Server process, 7–18
error conditions, 7–25
setting priority, 8–31

Server processes
restarting, 8–5
Server queues
defining, 8–6, 11–8, 15–2
updating, 15–12
Server software
customizing, 15–2, 15–9
definition, 7–3
installing, 13–7
on a VMScluser node, 8–36
on a VMScluster system, 8–44
upgrading, 15–7
Spooling
to a client queue, 6–1
SPR
instructions for submitting, G–4
Status messages
for DQS queues, 8–24
Symbionts
DQS$PRTSMB, 7–12, 7–16, 8–43
DQS$SMB, 7–11, 7–16
functions with DQS software, D–2
OpenVMS PRTSMB, 7–12
SYSGEN utility, 12–15
SYSTARTUP_V5.COM file, 15–5
System generation utility
See SYSGEN utility
System management procedures, for
installation, 12–15
System parameters
setting, 12–15
System startup file
editing, 15–4, 15–11

## T

Tailoring classes, 11–3
Troubleshooting
error messages in QSHOW display, 10–14
guidelines, 10–2
in VMScluster environments, 10–8
NETSERVER.LOG files, 10–10
notification service, 10–6
OPERATOR.LOG file, 10–10
Troubleshooting (cont'd)
print jobs, 10–9

## V

VMScluster system
as a DQS client, 7–3
as a DQS server, 7–3
configuring client software, 9–15
configuring server software, 8–36, 8–44
defining the DQS object, 15–13
DQS$STARTUP.COM procedure, 7–3
installing DQS software, 12–5
licensing of nodes, 7–3
nonhomogeneous clusters, 8–37, 8–44
outgoing cluster alias for clients, 9–15
postinstallation tasks, 15–13
troubleshooting, 10–8
VMSINSTAL command, 14–4
VMSINSTAL procedure
aborting, 14–6
following progress, 14–11
help, 14–10
installing server software, 14–9
invoking, 14–4
logging, 14–4
printing release notes, 14–2, 14–8
responding to questions, 14–7
running the IVP, 14–9
setting up DQS client/server account, 14–11
VMSLICENSE.COM file, 12–14

## W

Working directory
creating, 12–12
using a DFS-mounted disk, 12–13