ORACLE
INSURANCE

Docucorp Queue Systems

Windows and UNIX

Version 10.2

Revised 04/20/10

ORACLE

Docucorp Queue Systems

Version 10.2

April 2010

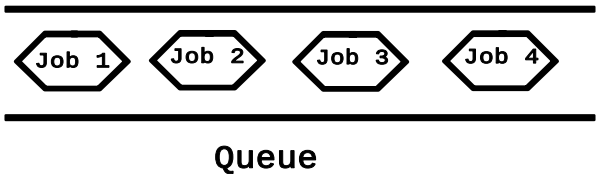
Contents

Introduction

Queue Overview

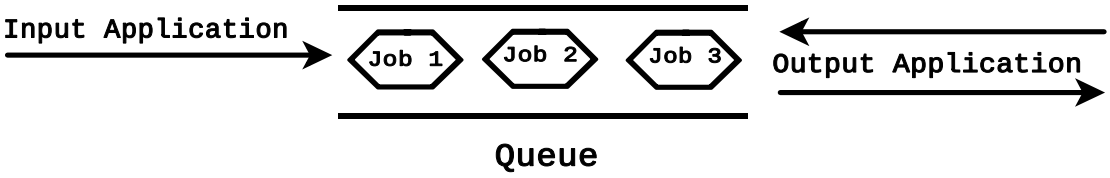
Dedicated Local Area Network (LAN) queues and servers improve the throughput processing time for several Docucorp® products. The improvement comes from distributing jobs across multiple computers available to the network. A queue is simply a holding area for jobs awaiting processing.

You can think of a queue as the equivalent of standing in line at the grocery store. You line up to be checked out, and the next person in line moves up when the previous person is finished. This form of first-come, first-served processing is known as FIFO (First-in, First-out). All the Docucorp queues use FIFO processing.



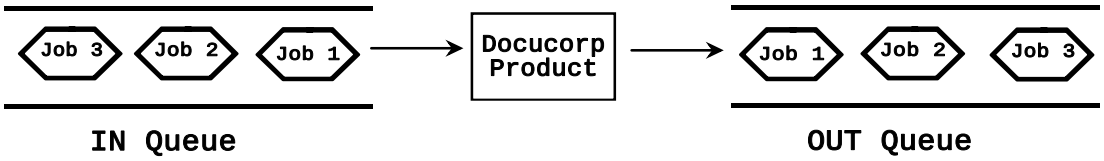
Using Queues

If applicable, you can configure your Docucorp application to use a queue on a file server. Using a very simple example, the application sending the job places the job in the queue. The application receiving the job retrieves the job from the queue. While jobs reside in the queue, the user can change the order of the applications in the queue.



When you're using multiple queues, on the other hand, the input application sends a job to the file server for placement in the IN queue. While the job resides in the IN queue, the output application periodically checks it for jobs. If the IN queue isn't empty, the output application removes the earliest job for processing. After the output application has processed the job, it then places the job in the OUT queue for additional processing.

With multiple applications, the OUT queue may not be the final destination. It could become the IN queue for yet another application. You can easily see how the OUT queue is only an OUT queue if no other applications retrieve jobs from it for further processing.



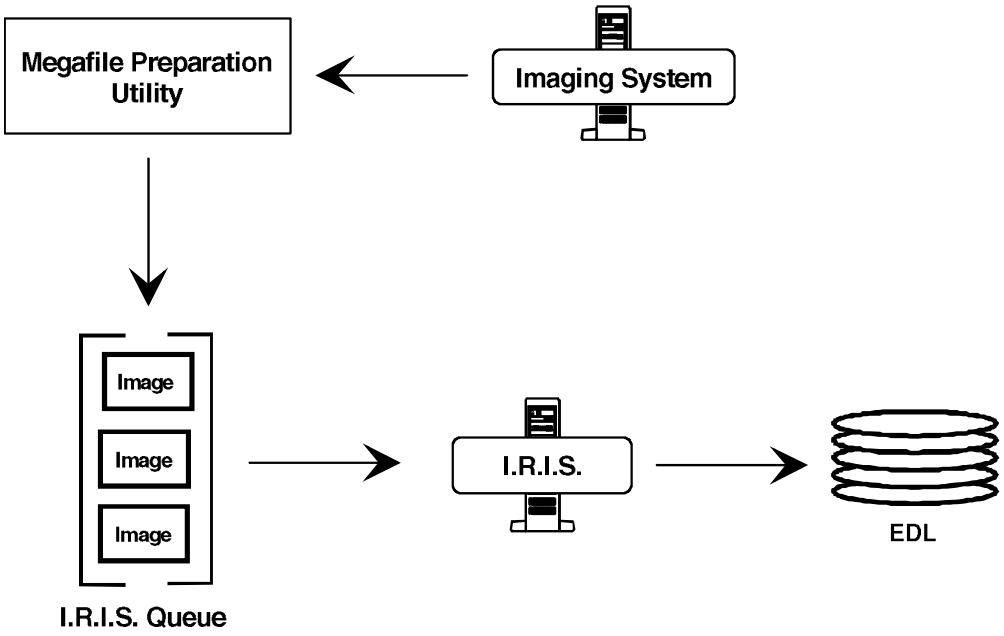
Features & Benefits

Queues lend a definite advantage in a business environment where one of the requirements is heavy and time-consuming document processing. Queues let you set aside several completed jobs for later processing, thus freeing up the individual workstations for additional data entry or document construction.

Queues, whether on a workstation or a network server, let the user:

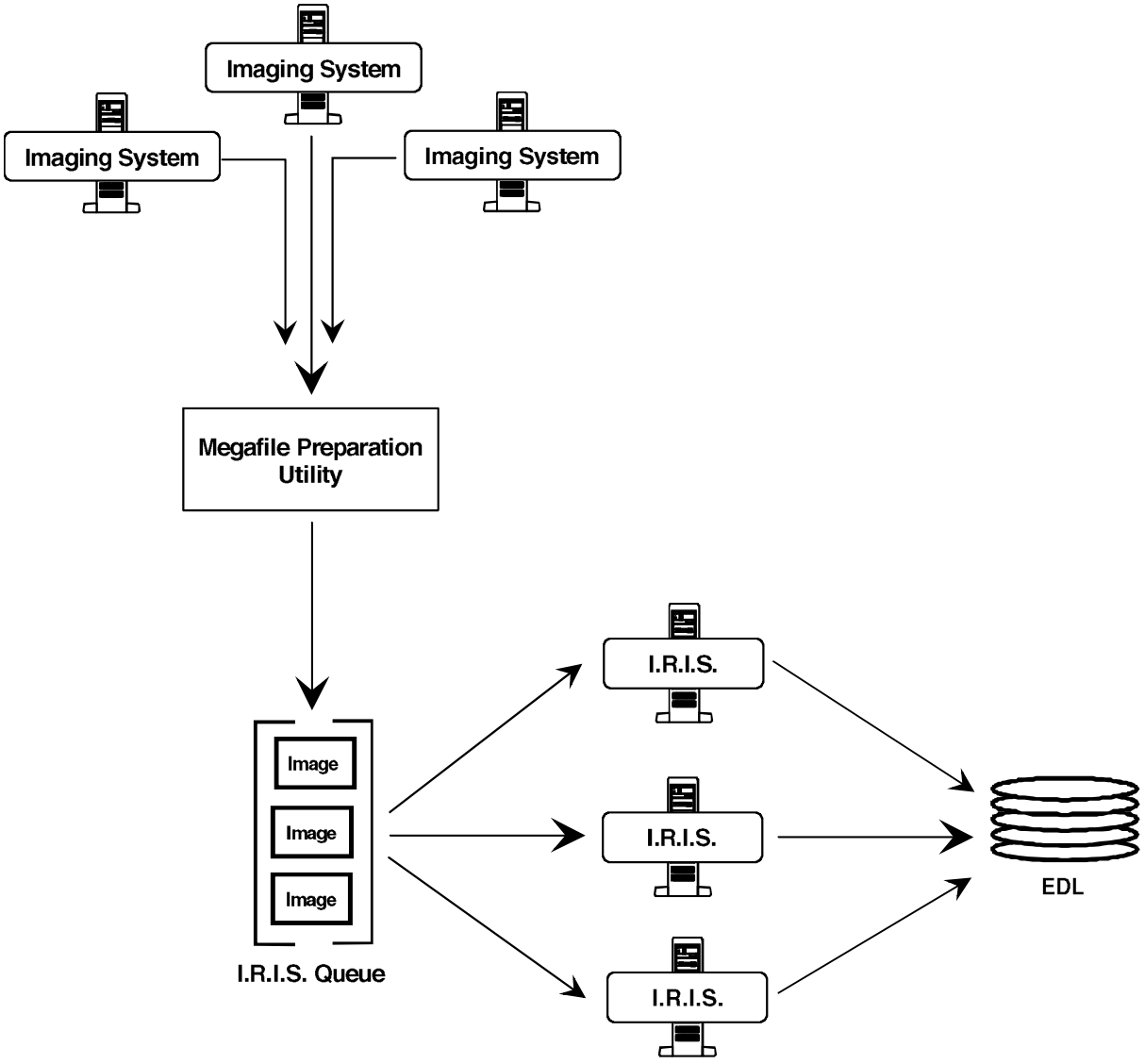
- Establish temporary working areas on the server to store jobs for later processing.
- Increase the throughput processing time of Docucorp products accessing queues.
- Reduce the required input time by storing completed tasks for completion at non-peak times.

The example below illustrates the use of queues as a part of the interaction between Docucorp products. I.R.I.S.™ removes a megafile (job), converts it to an EDL member, and sends the new member to a selected EDL.



A significant advantage to the queue configuration in a network environment is that multiple jobs can be added to and removed from the queue by multiple workstations.

The example below illustrates the use of queues in a network environment, where multiple workstations utilize the same queue.

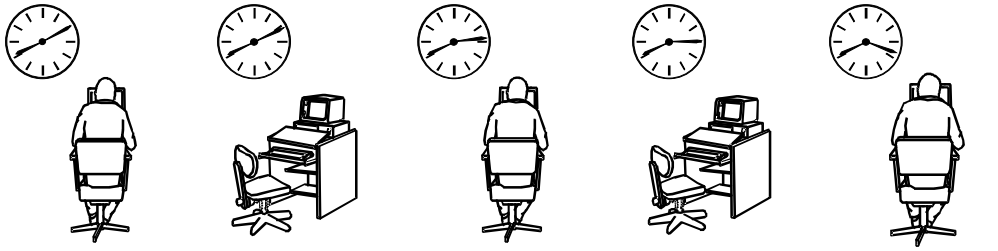


NOTE

The server portion of your queue system can be operated on a different platform than your client portion. For example, the client can operate under Microsoft Windows, whereas the server can reside in either Windows or UNIX.

Queues and Servers

Many Docucorp products operate in multiple stages, with an early stage requiring user input to compile a list of processing to be completed in a later stage. If both early and late stages were performed on one workstation, the situation might be as illustrated below.



Begin early stage, job one. Input from user. One min. duration.

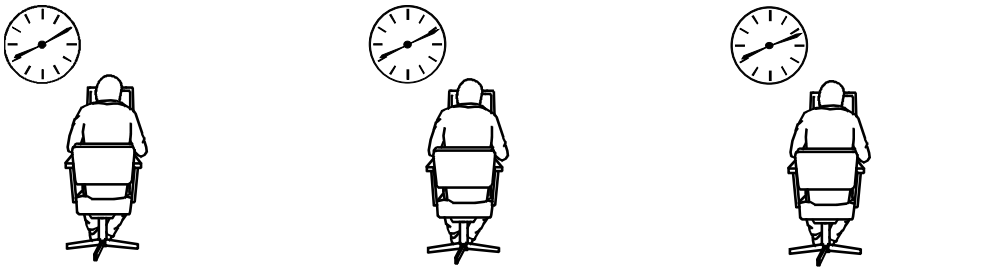
Begin late stage of job one. No user input. Three min. duration.

Early stage, job two. One min. duration.

Late stage, job two. Three min. duration.

Early stage, job three, etc.

Docucorp products use LAN queues and servers to move late processing stages to another machine. Spreading these tasks across machines is referred to as *distributed processing*. The advantage is minimizing the amount of time the user must wait on the machine. The user can complete an early stage, send the remaining work to the product queue, and begin a second task. Contrast the illustration below with the preceding one.



Begin early stage, job one. One min. duration. Send to queue for late stage processing.

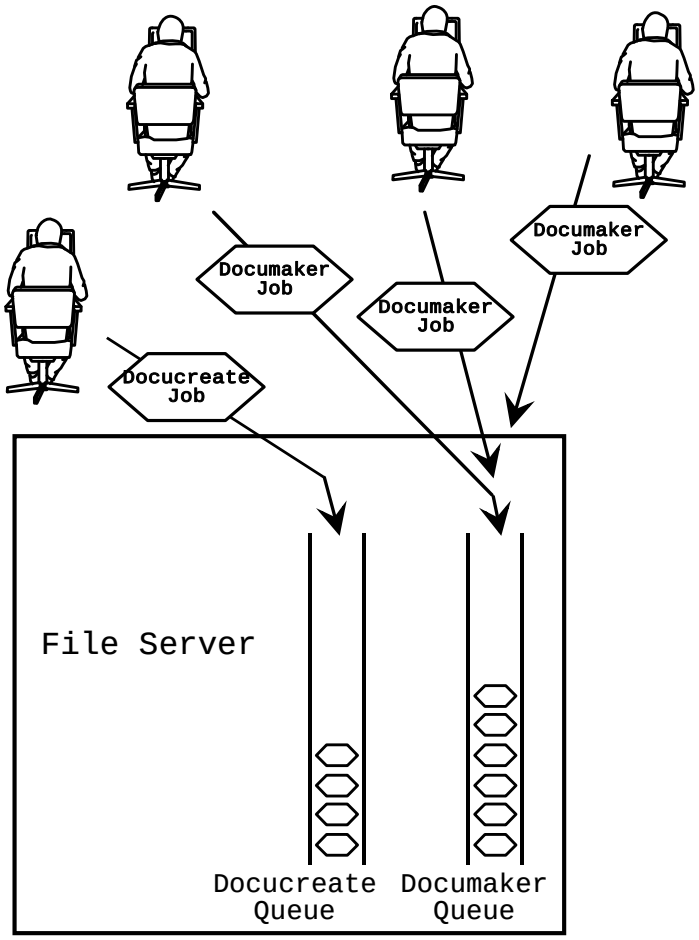
Begin early stage, job two. One min. duration. Send to queue for late stage processing.

Begin early stage, job three, etc.

Network Overview

Product Queues

One avenue in the distributed processing solution is to have several product queues reside on the LAN file server, where the individual workstations perform the processing at later times. Each product queue is dedicated to a single product and is identified by the product it serves.

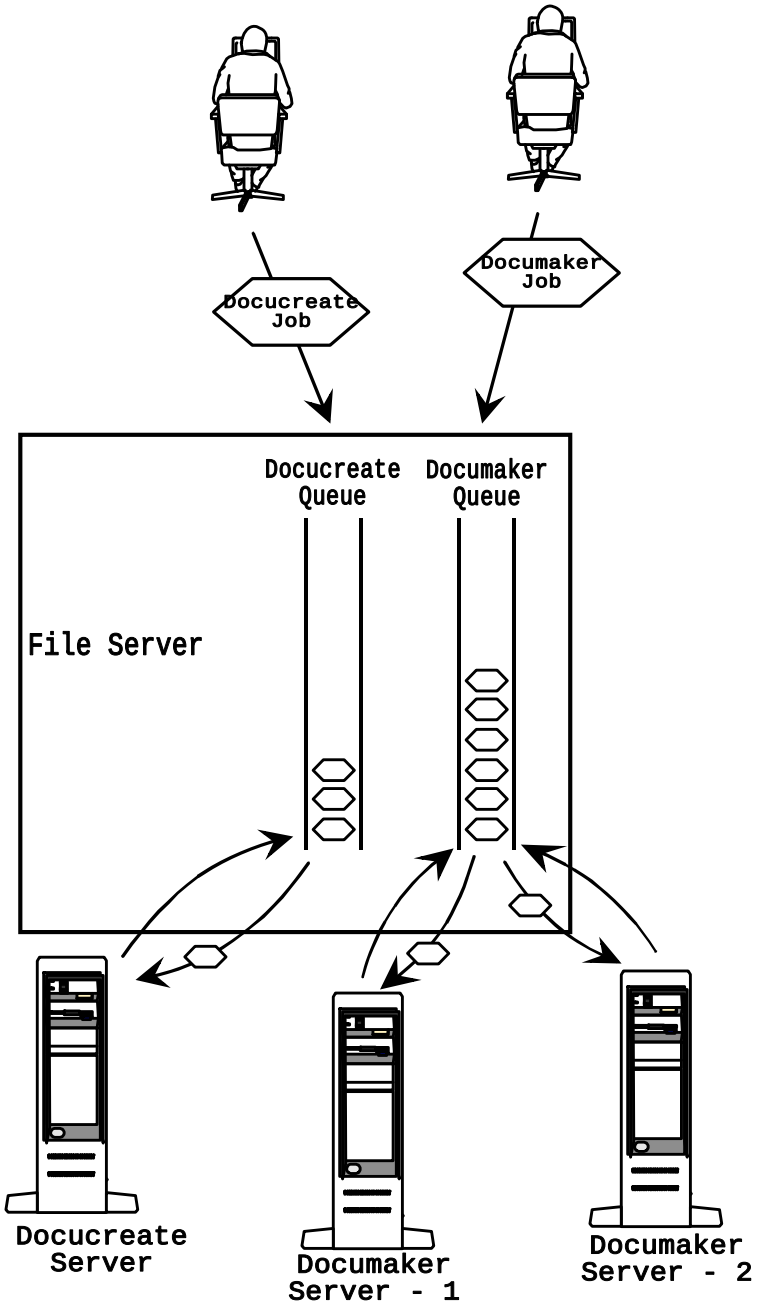


Jobs sent by users throughout the local area network are stored in the appropriate product queue.

Product Servers

Another avenue in the distributed processing solution is to dedicate separate machines to process jobs for each product. Product servers reside on individual machines and are dedicated to a single product.

A product server periodically checks its product queue on the file server. If there's a job in the queue, the product server removes it (jobs are removed on a First-In, First-Out (FIFO) basis) and begins processing. When the processing is complete, the server again checks the product queue.



In the illustration above, there is one product server for Docucreate® jobs and two for Documaker® jobs. If many jobs are sent to a single queue, attaching additional servers is a straight-forward way to increase throughput.

About This Manual

This manual explains the setup and use of the following Docucorp products:

- **Using Queue Systems™** — discusses the process of creating and managing data in the queues using ISIQMSD, QCUTIL, and QCMGR.
- **Queue Systems/NT** — this product creates and names queues in the Windows NT environment using user-defined names.
- **Queue Systems/UNIX** — this product creates and names queues in the UNIX environment using vacant sockets and user-defined names.

TIP

Throughout this guide, the generic term "UNIX" represents multiple UNIX-based platforms — currently, Queue Systems is implemented on IBM's AIX, Red Hat's Linux, and Sun SOLARIS.

- **Queuecommander™ for Windows** — this product manages queues in the Windows NT environment using dialog boxes.

The network administrator will use Queuecommander plus the chapter dealing with the desired network operating system, whereas the general user will only use Queuecommander.

In this book, you'll find the information you need to implement and customize not only the Queue Systems product for your operating system, but also Queuecommander for your particular processing environment.

Installing Queue Systems

Overview

In general, queues are a method by which individual workstations can organize jobs to be processed on the server. Before the workstation can send a job to the server in the first place, it must be able to communicate with the server.

Queue Systems is a communication and processing solution for document creation in a business environment where you have a mixture of computers and operating systems.

System Requirements

In a client/server networking environment, you'll require different configurations for the workstations and the servers. Queue Systems is a low-overhead application, so you'll need very few additional resources over and above the requirements for the operating system itself.

Windows

- Microsoft Windows 2000

  -or-
  Microsoft Windows NT version 3.5 or higher

UNIX

- Red Hat Linux 2.1
  -or-
  Sun SOLARIS 5.8
  -or-
  AIX version 5.1 or later for the RISC System/6000

Client Requirements

Since the client requirements vary greatly with the type of workstation you're using, you should discuss your particular needs with your network administrator.

Queuecommander Requirements

Hardware Requirements

These are the minimum hardware requirements to install and run Queuecommander:

- 100%-compatible PC with a Pentium microprocessor
- 256 megabytes (MB) of memory
- A hard disk and a CD-ROM drive.

Software Requirements

These are the minimum software requirements to install and run Queuecommander:

- Microsoft Windows 2000
  -or-
  NT Workstation 4.0 or higher (Service Pack 3 or higher)

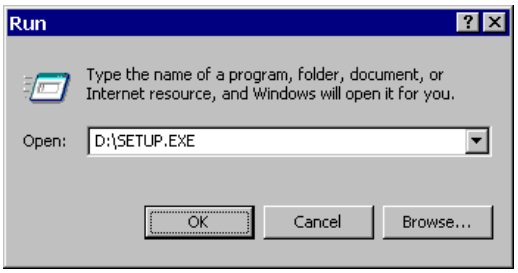
Installing Queue Systems

Queue Systems is controlled by **ISIQMSD.EXE**, which starts the queue management system from a command prompt and activates a previously-established queue. When you install Queue Systems, the procedure follows the standard Windows conventions.

If you want to	See:
Install Queue Systems for Windows NT	"To Install Queue Systems on Windows NT" on page 14
Install Queue Systems for UNIX	"To Install Queue Systems on UNIX" on page 19

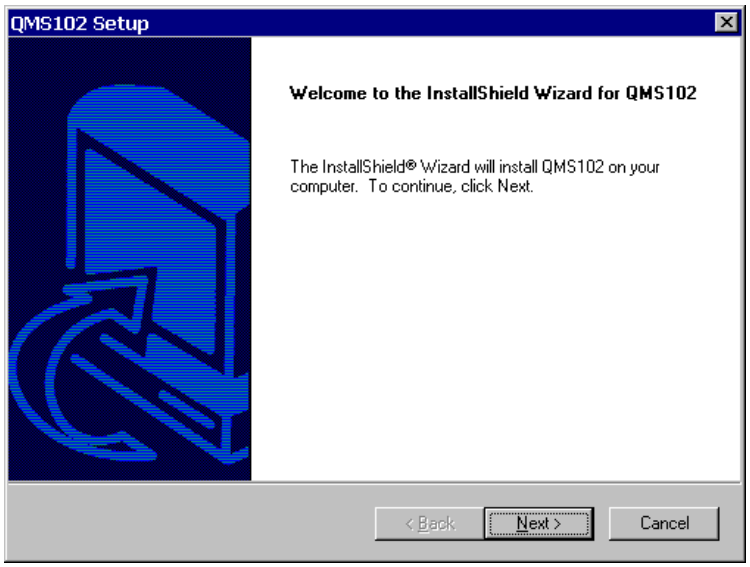
To Install Queue Systems on Windows NT

1. Insert the Queue Systems Installation disc in the CD-ROM drive of the workstation on which you intend to install the program.
2. Choose **Start>Run** from the Windows Desktop and Windows displays the Run dialog box.



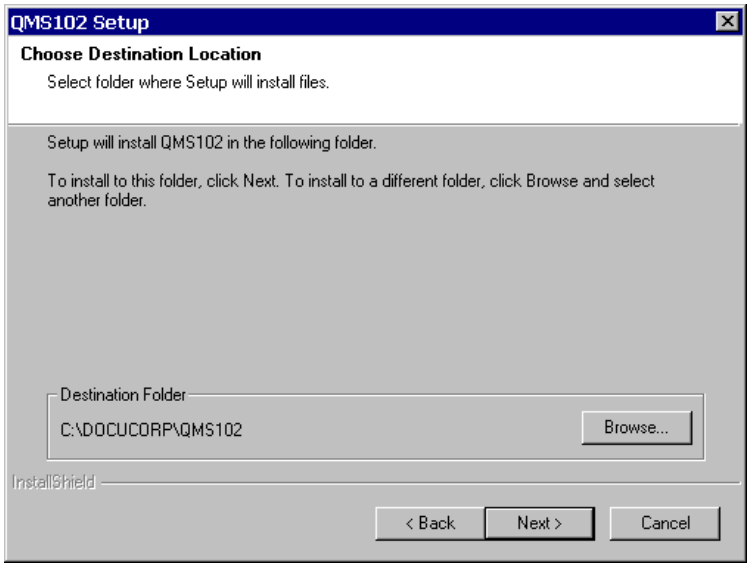
3. Enter **D:\SETUP.EXE** in the **Open:** text box and click **OK**. If the installation disc is in a drive with a different drive letter, substitute the appropriate letter.

The installation routine displays a dialog box indicating the InstallShield Wizard's progress, followed by the Queue Systems Setup dialog box.



4 Click **Next** to continue with the installation or **Cancel** to quit the program.

The routine then displays the Choose Destination Location dialog box, prompting you for the folder name/directory path into which you want to install the program.

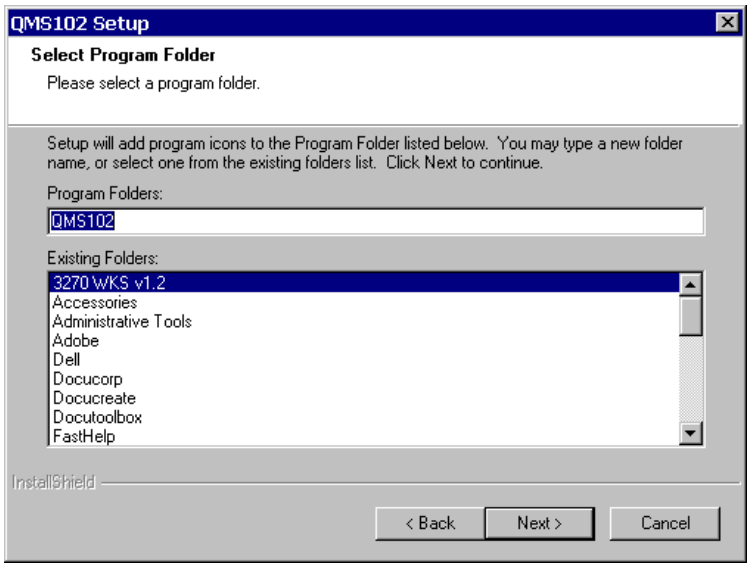


5 Perform one of the following procedures:

To	Perform this action
Accept the default path the installation routine proposes	Click **Next**.
Enter another path for the installation	Click **Browse...**, choose a new **path**, and then Click **Next**.

6 Click **Next** to continue with the installation if you haven't already done so. You can also click **Back** to return to the previous screen or **Cancel** to abort the routine.

The Select Program Folder dialog box displays the folder where you'll store the program shortcut icons.

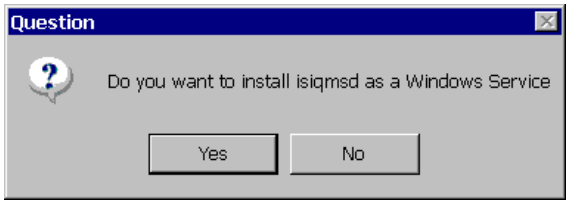


7 Perform one of the following procedures:

To	Perform this action
Accept the default program folder the installation routine proposes	Click **Next**.
Select an existing folder for the installation	1 Use the scroll bar to click on an existing folder. **-or-** Type the desired folder name in the space provided. 2 Click **Next**.

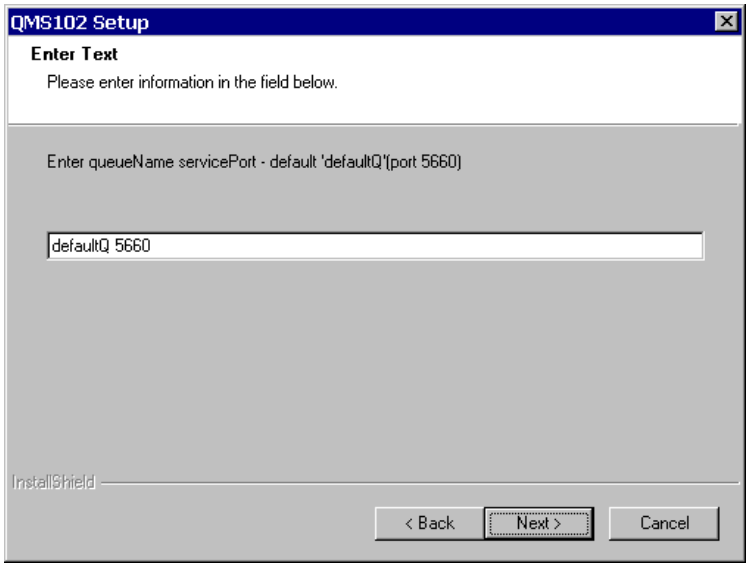
8 If you haven't already done so, click **Next** to continue with the installation. You can also click **Back** to return to the previous screen or **Cancel** to abort the routine.

A dialog box displays asking whether you want to install the QMS Server as a Windows Service.



9 Click **Yes** to install the Server as a Service or **No** to decline the option.

If you choose **Yes**, enter the desired queue Name and service Port separated by a blank. If you choose **No**, skip to “Step 12” on page 19.

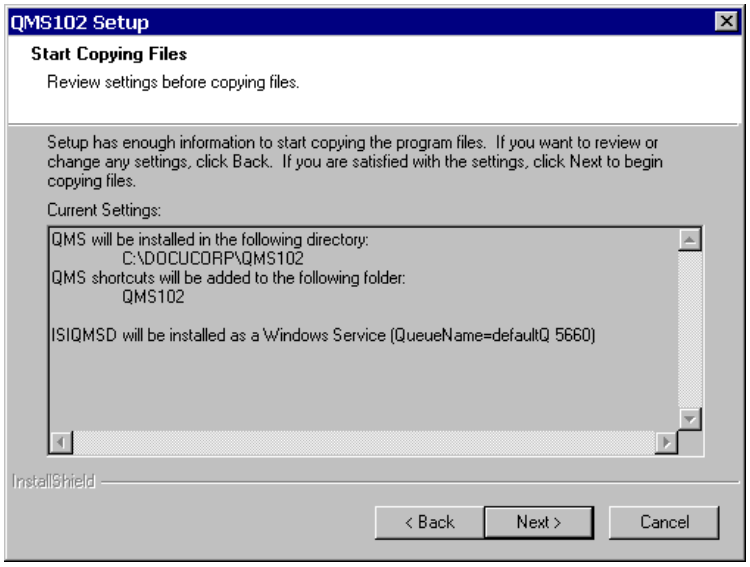


where:

Parameter	Explanation
queue Name	This is the logical user-supplied name of a given queue. If the queue doesn’t exist, one is established using the **queue name**. If the queue name does exist, it will be re-established. This value is optional; if you don’t supply the queue name, the queue and service port are created as in the above illustration, (e.g., **defaultQ 5560**).
service Port	This is the desired TCPIP service port number. This value is optional; if you omit the service port number when specifying the queue name, the queue Name is matched against the services file to resolve the port number (see "Configuring Queues for TCP/IP" on page 21).

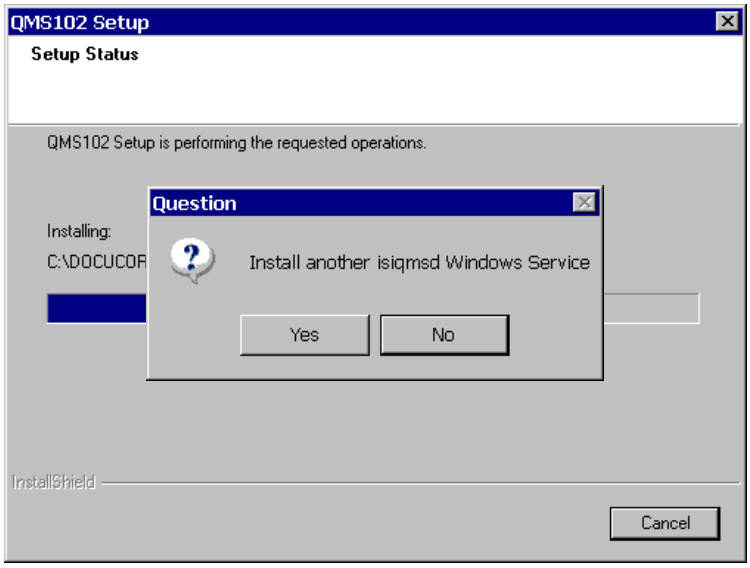
10 Click **Next** to proceed with the installation or **Back** to return to previous screens and make changes.

A verification dialog box displays listing the choices you've made thus far in the installation routine.



11 Click **Next** to proceed with the installation or **No** to return to previous screens and make changes.

A dialog box indicating the setup program's progress displays until all the files have been copied to your PC, followed by a dialog box asking whether you want to install another QMS Server as a Windows Service.

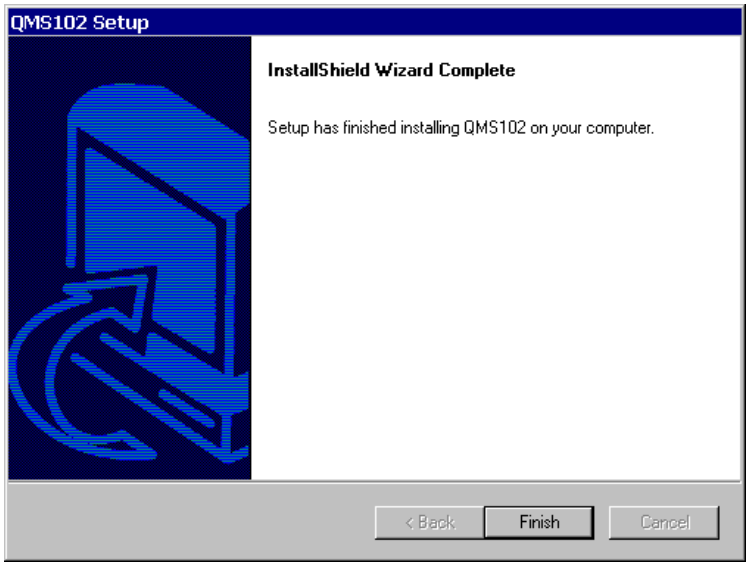


12 Click **Yes** to install another service or **No** to continue with installation.

NOTE

If you make a mistake while installing a service (e.g., an incorrect gate name), you must uninstall and then re-install the service.

When the installation is 100% complete, the program displays the InstallShield Wizard Complete dialog box.



13 Choose **Finish** to complete the installation program and return to Windows.

TIP

If you need to install more services after you've completed the installation routine, you can execute *isiqmsd.exe* as a stand-alone application (see "Manually Operating ISIQMSD" on page 53 for more information).

To Install Queue Systems on UNIX

1 Create an installation directory for the Queue Systems files (e.g., /u/qms102).

2 Copy the installation CD into the installation directory.

3 From a command prompt, change directories to the installation directory.

4 Execute the UNIX "tar" command to install Queue Systems for UNIX (e.g., "tar -xvf name.tar," where *name.tar* is the name of the Queue Systems distribution *tar* file).

The *tar* file name varies, depending on the version and platform to which you're installing. The following list represents the current *tar* file names for various platforms; newer release names will be distributed as they're developed:

- qms102AIX51.tar — IBM AIX 5.1
- qms102LINUX.tar — Red Hat Linux 2.1
- qms102SUN.tar — Sun SOLARIS 5.8

Configuring TCP/IP Networks

Configuring Queues for TCP/IP

Each computer has it own unique TCP/IP Internet Address number, associated with the Hostname in the Hosts file, which serves as the “server” name. The “queues” are associated with a Service name in the Services file. When sharing information between two networked machines, each computer must have the same socket number in its Services file. It’s a good practice to use the same Service name for the socket on each machine on your network. You may need to contact your network administrator to coordinate these entries.

WARNING!

You must be the root user to modify the **/etc/services** file.

To Configure Queues for TCP/IP

1. Change to the /etc directory and edit the Services file.

```
#
# /etc/services
# —             Socket #/
Service         Protocol        # Comments
-----------------------------------------------------------
echo            7/tcp
echo            7/udp
discard         9/tcp           sink null
discard         9/udp           sink null
systat          11/tcp          users
daytime         13/tcp
daytime         13/udp
netstat         15/tcp
qotd            17/tcp          quote
chargen         19/tcp          ttytst source
chargen         19/udp          ttytst source
ftp-data        20/tcp
 .                .
 .                .
 .                .
documaker       6650/tcp        # Documaker queue
docucreate      6651/tcp        # Docucreate queue
docusave        6652/tcp        # Docusave queue
qc-in           6969/tcp        # Queuecommander IN queue
```

2. Select a socket number (in the Socket # / Protocol column) that isn’t reserved by any other networking application you may have installed at your enterprise. This socket number must be unique on all the workstations in use for the specific QMS installation. You may need to contact your network administrator for assistance in selecting this number.

3. Decide on a Service name to be associated with the socket — it can be any name you want (e.g., **qc-in**). Type this name into the Service column and press **TAB** to advance the cursor to the Socket # / Protocol column.

4 Type in the unique socket number you chose in Step 2, followed by **/tcp** (e.g., **6969/tcp**).

NOTE

The Service name and Socket # are case-sensitive. If your Socket # is **6969/tcp**, for example, you must enter the same case mixture in all applicable files.

5 Save the file. This process establishes the service name and socket number.

6 Change to the ETC directory and edit the HOSTS file.

```
#
# /etc/hosts
#
# Internet Address  Hostname             # Comments
-------------------------------------------------------------------
# xxx.x.xxx.1       net0sample           # ethernet name/address
# xxx.xxx.x.1       token0sample         # token ring name/address
# xx.x.x.2          x25sample            # x.25 name/address
xxx.x.x.1           loopback localhost   # loopback (lo0) name/address
xxx.xx.x.1          prod-serv
xxx.xx.x.2          dev-serv
xxx.xx.x.3          hp-serv
```

7 From your network administrator, obtain the IP Address of the server on which the queues reside and type this number into the Internet Address column.

8 Type a Hostname that you want to associate with the Internet Address.

9 Save the file. This process establishes the Internet Address and Hostname.

Then, when you're identifying the *servername* and *queuename* from within a client application, the “server” is the name associated with the TCP/IP machine number in the Hosts file, while the “queue” is the service name from the Services file.

Queue Management Systems Utilities

Overview

Docucorp Queue Management Systems is delivered with the following components:

- **ISIQMSD** — is the QMS “server” that creates and destroys a queue and processes submitted jobs. ISIQMSD allows workstations on Windows NT or UNIX to communicate with servers on those same operating systems via TCP/IP.

  For more information about using the server, see "ISIQMSD" on page 24.

- **QCMDR32** — (Queuecommander) is the QMS Windows GUI “client” that manages submitted jobs. Queuecommander also monitors job queues, either at a specified time interval or on-demand.

  For more information about Queuecommander, see "Queuecommander" on page 35.

- **QCUTIL** — is a QMS command-line utility designed for the person performing daily maintenance on a queue.

  For more information about QCUTIL, see "QCUTIL" on page 27.

- **QCMGR** — is the QMS Manager command-line utility designed to manage ISIQMSD (e.g., in a manner similar to Queuecommander). QCMGR, however, contains fewer functions than Queuecommander. In addition, a network administrator has the ability to restrict its use by granting rights only to designated employees.

  For more information about QCMGR, see "QCMGR" on page 29.

Docucorp Queue Systems is designed for use on either Windows NT or UNIX. If your operating platform is Windows NT, for example, you have a choice: you can use either the Queuecommander GUI or a combination of QCUTIL and QCMGR, whereas the Queuecommander GUI isn’t available on UNIX.

Even though QCMGR doesn’t contain all of the functionality included in Queuecommander, you can combine it with QCUTIL to fully replicate all of the Queuecommander functions. As an added bonus, QCUTIL and QCMGR are command-line applications and are suitable for use in batch files.

ISIQMSD

ISIQMSD is extremely flexible in the methods by which you can launch the server, including:

- Installing as a Windows Service
- Starting from a Windows command prompt

If you install the server as a Windows Service, you can define the service with one of two startup types:

- **Automatic (default)** — the Windows Service starts automatically every time you boot your server.
- **Manual** — the server administrator must start the service from the Windows Control Panel>Administrative Tools>Services dialog.

TIP

If you need to install more services after you've completed the installation routine, you can execute *isiqmsd.exe* as a stand-alone application.

Choose from one of the following ISIQMSD tasks:

If you want to	See:
Start a queue	"To Start Queues in Windows (Non-NT Service)" on page 25
Stop a queue	"To Stop Queues in Windows (Non-NT Service)" on page 25
Start a queue (NT Service)	"To Start Queues in Windows (NT Service)" on page 25
Stop a queue (NT Service)	"To Stop Queues in Windows (NT Service)" on page 26
Start a UNIX queue	"To Start Queues in UNIX" on page 26
Stop a UNIX queue	"To Stop Queues in UNIX (non-Daemon)" on page 26
Stop a UNIX queue (Daemon)	"To Stop Queues in UNIX (Daemon)" on page 26

To Start Queues in Windows (Non-NT Service)

WARNING!

You can only create a queue if you possess Console rights on your network.

You can create a queue from the command prompt each time you need it.

- Make sure you're located in the desired sub directory for storing your queues, as the following command creates queues in the directory where you issue the command.

  ```
  ISIQMSD <queue Name> <service Port> <ENTER>
  ```

where:

Parameter	Explanation
ISIQMSD	is the name of the executable file
queue Name	This is the logical user-supplied name of a given queue. If the queube doesn't exist, one is established using the **queue name**. If the queue name does exist, it will be re-established. This value is optional; if you don't supply the queue name, the queue and service port are created with the default values (e.g., **defaultQ 5560**).
service Port	This is the desired TCPIP service port number. This value is optional; if you omit the service port number when specifying the queue name, the queue Name is matched against the services file to resolve the port number (see "Configuring Queues for TCP/IP" on page 21).

TIP

The following command is an example of setting up a queue under NT Server:

```
ISIQMSD qc-in <ENTER>
```

To Stop Queues in Windows (Non-NT Service)

- In the window where the server is running, press **CTRL-C** to shut down the Queue Daemon.

To Start Queues in Windows (NT Service)

- Use **Control Panel**>**Administrative Tools>Services** to start the server. For more information about manually starting the QMS server, see "To Start ISIQMSD in Windows" on page 55.

  The startup type for queues established during installation is Automatic (which means the desired queue may already be active).

  -or-

  From a command prompt, enter the following command:

  ```
  ISIQMSD -INSTSERV <queue Name> <service Port> <ENTER>
  ```

  This command adds and starts the specified service via Windows Services.

To Stop Queues in Windows (NT Service)

- Use **Control Panel**>**Administrative Tools>Services** to stop the server. For more information about manually starting the QMS server, see "To Start ISIQMSD in Windows" on page 55.

  -or-

  From a command prompt enter the following command:

  ```
  ISIQMSD -REMSERV <queue Name> <ENTER>
  ```

  This method stops the specified service and removes it from the Windows Services pool.

To Start Queues in UNIX

You can create a queue from the command prompt each time you need it.

- Make sure you're located in the desired sub directory for storing your queues, as the following command creates queues in the directory where you issue the command.

  ```
  isiqmsd [-daemon] <queue Name> <service Port> <ENTER>
  ```

where:

Parameter	Explanation
isiqmsd	is the name of the executable file
-daemon	(Optional) Include this parameter when you want to start a queue as a UNIX Daemon. The use of a daemon requires additional configuration (see "To Configure ISIQMSD as a UNIX Daemon" on page 60).
queue Name	This is the logical user-supplied name of a given queue. If the queue doesn't exist, one is established using the **queue name**. If the queue name does exist, it will be re-established. This value is optional; if you don't supply the queue name, the queue and service port are created with the default values (e.g., **defaultQ 5560**).
service Port	This is the desired TCPIP service port number. This value is optional; if you omit the service port number when specifying the queue name, the queue Name is matched against the services file to resolve the port number (see "Configuring Queues for TCP/IP" on page 21).

TIP

The following command is an example of setting up a queue under UNIX:

```
isiqmsd qc-in <ENTER>
```

To Stop Queues in UNIX (non-Daemon)

- In the window where the server is running, press **CTRL-C** to shut down the Queue Daemon.

To Stop Queues in UNIX (Daemon)

1. Use the ps command to locate the process ID (pid) of the daemon that you want to stop (e.g., **ps -ef**).
2. Use the kill command, along with the process ID (pid), of the daemon that you want to stop (e.g., **kill** *pid*).

QCUTIL

QCUTIL is a QMS utility designed for the person performing daily maintenance on queues and their contents. With this utility, you can use the following commands to maintain queues:

- **add** — adds a job to the queue for processing
- **get** — retrieves a job from the queue for saving
- **list** — lists the jobs in the queue
- **drain** — drains or empties the queue, meaning all jobs are deleted from the queue

You can see a listing of available commands and their parameter by typing QCUTIL from a command prompt:

```
Queue Management System QUtility Mod:  M010015  as of May  2 2003 09:41:56
 COPYRIGHT 1993 TO 2003 DOCUCORP INTERNATIONAL - ALL RIGHTS RESERVED

usage: qcutil add protocol queueName serverName fileName [description] [userID]
usage: qcutil get protocol queueName serverName [fileName]
usage: qcutil list protocol queueName serverName [jobsize]
usage: qcutil drain protocol queueName serverName
         protocol: [ls|tcpip|disk]
```

Syntax

▶ To use QCUTIL.EXE, type the following syntax at a command prompt:

`qcutil` *`command protocol queuename servername filename description userid`*

where:

Parameter	Explanation	
command	**add**	adds a job to the queue to await processing
	get	retrieves a job from the queue, writes it to a local file, and then permanently removes it from the queue.
	list	polls the queue and lists all waiting jobs
	drain	drains or empties the queue, leaving the queue intact
protocol	**ls**	LANServer queue
	tcpip	TCP/IP queue
	disk	local disk queue
queuename	the logical, user-supplied name of a given queue. This name must match the Socket # / Protocol listed in the Services file (see "Configuring Queues for TCP/IP" on page 21).	
servername	the machine name of the server where the queue files will reside. This name must match the Hostname listed in the Hosts file (see "Configuring Queues for TCP/IP" on page 21).	
filename	the file that is to be added to the queue. In the case of the Get command, the filename represents the local file to which you want to save the job.	
description	(*optional*) a description to be associated with a given queue entry	
userid	(*optional*) a user identification designation to be associated with a given queue entry	

To Use the Add Command

- To add a new job (file) to the Queue Server queue, type the following syntax at a command prompt:

  qcutil **add** *protocol queuename servername filename [description] [userid]*

To Use the Get Command

- To retrieve and remove a job from the Queue Server queue, type the following syntax at a command prompt:

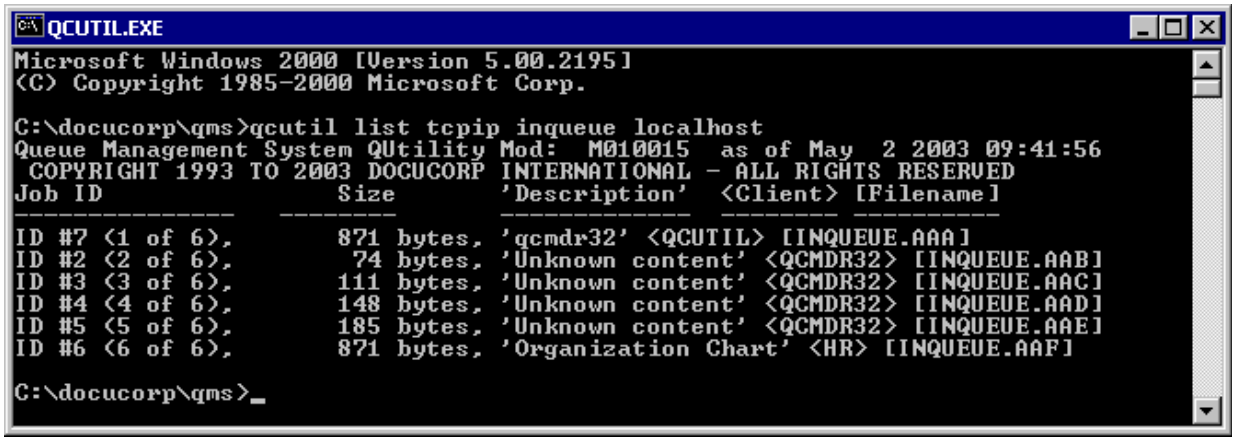
  qcutil **get** *protocol queuename servername [filename]*

To Use the List Command

- To show the output results, type the following syntax at a command prompt:

  qcutil **list** *protocol queuename servername*

QCUTIL displays a listing of the contents of the current server/queue:



Field	Contents
Job ID#/ Position	The Job ID Number is a sequential number assigned by Queuecommander. The jobs are assigned a number in the order in which they were submitted to the queue. The job's actual position in the queue is displayed next. Since you can reorder jobs, the position number isn't necessarily the same as the Job ID.
Size	The size of the job, expressed in bytes.
Description	Queuecommander displays a description of the job that was supplied by the application or keyed by the user when the job was added to the Queue Server.
Client	Queuecommander displays the user ID supplied by the application or keyed by the user when the job was added to the Queue Server.
Filename	This is the generated file name of the job in the Queue Server, consisting of the queue name plus a three-character identifier.

To Use the Drain Command

- To permanently remove (delete) all jobs from the Queue Server queue, type the following syntax at a command prompt:

  `qcutil drain protocol queuename servername`

QCMGR

QCMGR is the QMS Manager Utility designed to manage the QMS "server" (e.g., ISIQMSD). QCMGR provides many, but not all, of the queue management functions included in Queuecommander. It's also designed so that network administrators can control the "management" types of functions by limiting the availability and distribution of QCMGR within an organization.

You can operate QCMGR in one of two modes:

- **command line mode** — if you issue all command arguments and parameters with QCMGR on the command line, the function processes and the program ends.

  This mode is beneficial when invoking QCMGR from another process or batch file.

- **interactive mode** — if you omit any necessary parameters, QCMGR operates as an interactive application and prompts you for command arguments and parameters.

Most of the parameters are optional, depending on whether you've used them with a previous command. If you don't use a required parameter, you'll be prompted for it.

TIP

Since the **CONnect** command requires three of the most common parameters, and you have to issue it as the first command, you don't have to re-type *protocol*, *queueName*, or *serverName* as long as you're connected to the same queue.

You can see a listing of available commands and their parameter by typing QCMGR from a command prompt:

```
QCMGR - QMS Queue Manager Utility
    COPYRIGHT 2002 TO 2002 DOCUCORP INTERNATIONAL - ALL RIGHTS RESERVED
   QCMGR Mod:  M010000 , Built: Dec 11 2002 10:08:02

QCMGR: Enter a Command:
         CONnect protocol queueName serverName
         ChangePos [protocol queueName serverName] jobid position
         CheckStatus [protocol queueName serverName]
         List [protocol queueName serverName]
         DELetejob [protocol queueName serverName] jobid
         HOldjob  [protocol queueName serverName] jobid [type]
                jobid = ID of job to be held
                type  = type flag placed on held jobid
                        Operator = operator flag [default]
                        User = user flag
                        Both = both operator and user flag
         RELeasejob  [protocol queueName serverName] jobid [type]
                jobid = ID of job to be released
                        nnn or All or * = A or * releases all jobs
                type  = type flag of job to release
                        Operator = operator flag [default]
                        User = user flag
                        Both = both operator and user flag
         DRain [protocol queueName serverName]
         RESume  [protocol queueName serverName]
         HAlt  [protocol queueName serverName]
         SHUTdown  [protocol queueName serverName]
         Help - display help information for all commands.
         /? - display help information for all commands.
         ? - display help information for all commands.
         Quit - end program.
         eXit - end program.
                protocol: [nw|tcpip]
Enter Command:
```

When you type the command, you can abbreviate by typing the beginning characters or enough to make it unique from the others (i.e., “command cascading”). The acceptable abbreviations are capitalized in the help facility and in the following explanations.

To Start QCMGR

- Do one of the following to CONnect to the desired job queue:

To CONnect in	Do this:
command-line mode	See "To Use the Connect Command" on page 31
interactive mode	Type **QCMGR** and press **ENTER**

To Use the Connect Command

- Type the following syntax from a command prompt:

  qcmgr **CONnect** *protocol queueName serverName*

where:

Parameter	Explanation	
protocol	**nw**	NetWare queue
	tcpip	TCP/IP queue
queueName	the logical, user-supplied name of a given queue. This name must match the Socket # / Protocol listed in the Services file (see "Configuring Queues for TCP/IP" on page 21).	
serverName	the machine name of the server where the queue files will reside. This name must match the Hostname listed in the Hosts file (see "Configuring Queues for TCP/IP" on page 21).	

To Use the ChangePos Command

The ChangePos command changes the logical position of a job within the queue. Since the application retrieves jobs in a FIFO (first-in, first-out) order, changing the job position either causes the job to be processed earlier or later, depending on the direction of movement.

- Type the following syntax from a command prompt:

  qcmgr **ChangePos** *[protocol queuename servername] jobid position*

where:

Parameter	Explanation
jobid	the Job ID number assigned to the job
position	a number specifying the NEW position (POS) of the job

To Use the CheckStatus Command

The CheckStatus command ensures the queue you're polling is active.

- Type the following syntax from a command prompt:

  qcmgr **CheckStatus** *[protocol queuename servername]*

To Use the List Command

The List command displays the jobs in the queue and describes the output results.

- Type the following syntax from a command prompt:

  qcmgr **List** *[protocol queuename servername]*

For an explanation of the contents of the List command, see "To Select a Server Connection" on page 40.

To Use the DeleteJob Command

The DeleteJob command permanently removes a job from the queue. New jobs can be added to the queue in various ways. A Docucorp product, for example, can add jobs to the queue, or you can use QCUTIL or Queuecommander (qcmdr32.exe).

- Type the following syntax from a command prompt:

  qcmgr **DELetejob** *[protocol queuename servername] jobid*

where:

Parameter	Explanation
jobid	the Job ID number assigned to the job

To Use the HoldJob Command

The HoldJob command puts a job in hold status base on its type. Applications retrieving jobs normally bypass jobs with a hold status and leave them in the queue.

- Type the following syntax from a command prompt:

  qcmgr **HOldjob** *[protocol queuename servername] jobid [type]*

where:

Parameter	Explanation	
jobid	the Job ID number of the job to be held	
type (optional)	**Operator**	operator flag (default)
	User	user flag
	Both	both operator and user flags

To Use the ReleaseJob Command

The ReleaseJob command removes the hold status of a job placed by the HoldJob command.

- Type the following syntax from a command prompt:

  qcmgr **RELeasejob** *[protocol queuename servername] jobid [type]*

where::

Parameter	Explanation	
jobid	the Job ID number(s) of the job(s) to be released. **Note:** **nnn**, **All**, ***=A**, or ***** releases all jobs	
type (optional)	**Operator**	operator flag (default)
	User	user flag
	Both	both operator and user flags

To Use the Drain Command

The Drain command permanently removes (deletes) all jobs from the queue, leaving the queue intact.

- Type the following syntax from a command prompt:

  qcmgr **DRain** *[protocol queuename servername]*

To Use the Resume Command

The Resume command reactivates a queue for processing that had been temporarily stopped by the Halt command.

- Type the following syntax from a command prompt:

  qcmgr **RESume** *[protocol queuename servername]*

To Use the Halt Command

The Halt command temporarily stops a queue from processing. The queue can be restarted by the Resume command.

- Type the following syntax from a command prompt:

  qcmgr **HAlt** *[protocol queuename servername]*

To Use the Shutdown Command

The Shutdown command terminates a queue, which also terminates the Queue Server (isiqmsd) processing. To restart the queue, you must execute the Queue Server (isiqmsd) application.

- Type the following syntax from a command prompt:

  qcmgr **SHUTdown** *[protocol queuename servername]*

To Use the Help Command

The three help commands display a listing of the correct syntax for all commands.

- Type the following syntax from a command prompt:

  qcmgr **Help**
  -or-
  qcmgr **?**
  -or-
  qcmgr **/?**

To Use the Quit Command

The two quit commands exit the program.

- Type the following syntax from a command prompt:

  qcmgr **eXit**
  -or-
  qcmgr **Quit**

Queuecommander

Overview

Queuecommander is a Microsoft Windows-based program with which you can perform the following tasks:

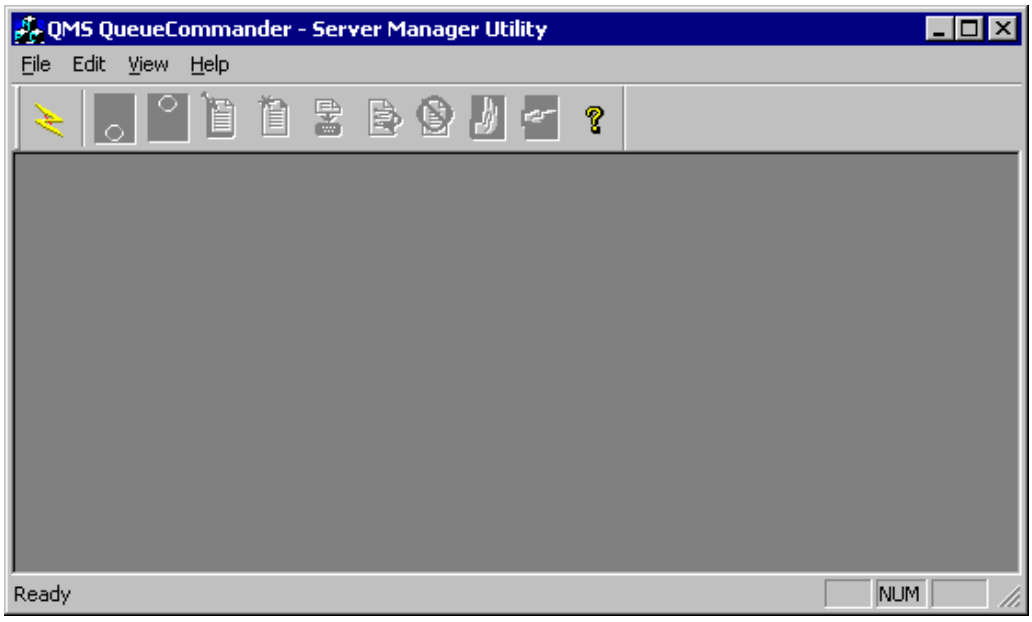
- Monitor the status of jobs in the queue
- Simultaneously monitor multiple queues
- Change the processing order of jobs in the queue
- Hold/Release a job in the queue
- Add a job to the queue
- Retrieve a job from the queue for saving
- Delete a job from the queue
- Halt/Resume processing a job queue
- Create, drain, or destroy a job queue

Queuecommander is useful for managing your document processing load because you're given the flexibility and control to oversee the upkeep of the job queues.

To Start Queuecommander

After you install Queuecommander in your Windows system, the Start menu's Programs sub-menu displays the Docucorp folder.

- Select the Queucommander folder, then click **Queuecommander** to display the main window:



For more information on operating Queuecommander, see "Setting Up Queuecommander" on page 37.

To Close Queuecommander

- Select **Exit** from the **File** menu.

  Queuecommander closes and returns to the Windows Desktop.

Activating Command Buttons with a Mouse

Queuecommander command buttons (also called icons) let you execute some of the most frequently used commands with a point-and-click of your mouse. The following table shows the Queuecommander buttons and their associated actions.

	**Connect to Server** lets you open a new window and monitor a job server/queue.		**Start Auto Refresh** initiates the process of polling the queue at specified intervals.
	**Stop Auto Refresh** terminates automatic polling.		**Job List** obtains and displays of list of all jobs in the queue..
	**Add Job** automates the procedure of adding a job to be processed.		**Get Job** retrieves a job from the queue that you can then save to a file.
	**Change Job Position** moves the current job to a specified position within the queue.		**Delete Job** removes the job from the queue and places it in the trash can.
	**Halt Server** (indicated with a red light) stops processing jobs in the active queue.		**Resume Server** (indicated with a green light) begins processing jobs in the halted queue.
	**Help** invokes the on-line help utility.		

TIP

The icon on the Tool Bar indicates the action that **will be** taken on the queue/job. If you press the **Halt Server** icon, for example, the current queue will be held and prevented from processing.

To Activate a Command Button

- Position the cursor on the command button and click.

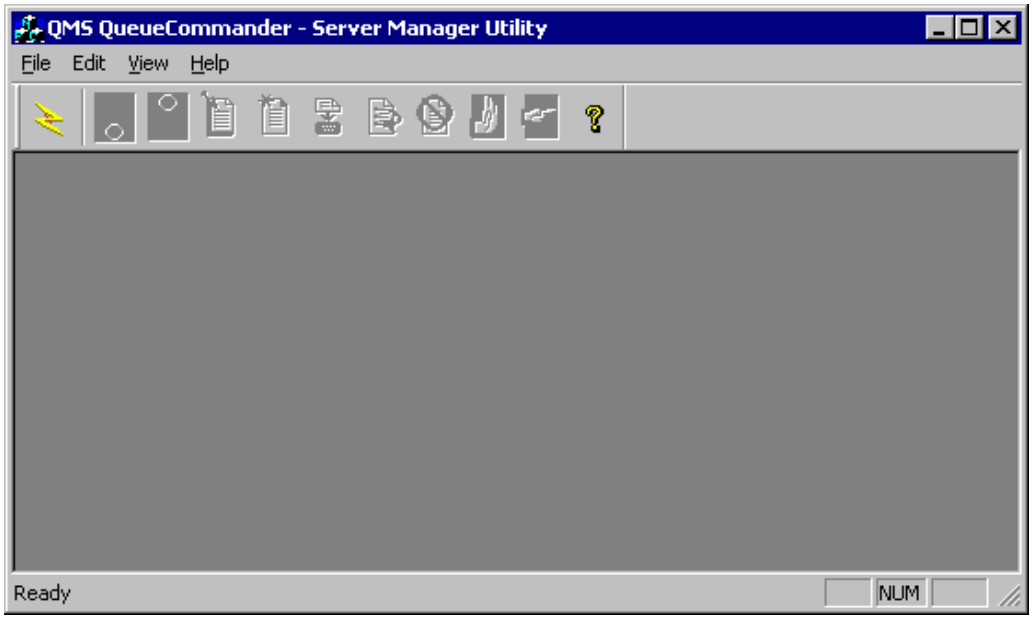
Setting Up Queuecommander

When you open Queuecommander for the first time, you might be presented with a blank main screen. If this is the case (or you're setting up a new queue), you'll need to follow the procedure for setting up queues. In doing so, you can choose from the server(s) present on your network and the queues on those server(s).

To Set Up Queuecommander

1 Double-click the **Queuecommander** icon.

Queuecommander displays its main window.



2 Click  on the Toolbar.

-or-

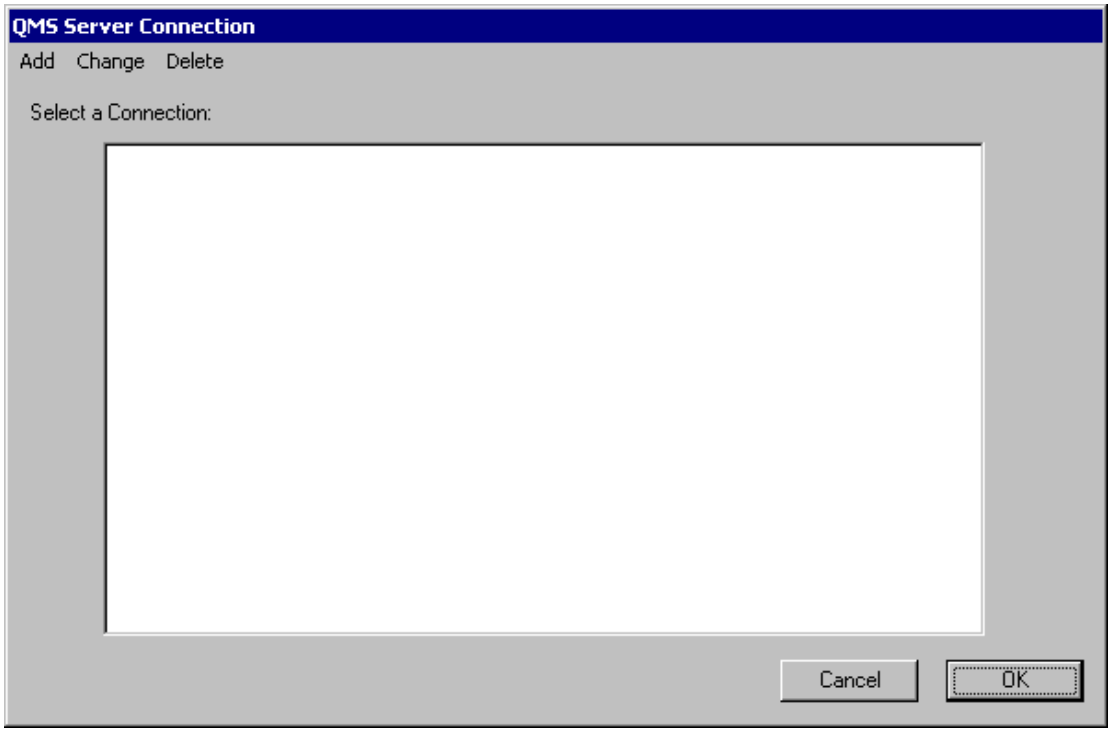
Select **New** from the **File** menu.

NOTE

The first time you use Queuecommander, you'll receive the following dialog box:

Click **OK** and proceed with the following steps to establish QMS Server Connection.

Queuecommander displays the QMS Server Connection dialog box.



Because you can use the QMS Server Connection dialog box to add, change, delete, and select queue connections, this guide provides a separate topic for each action.

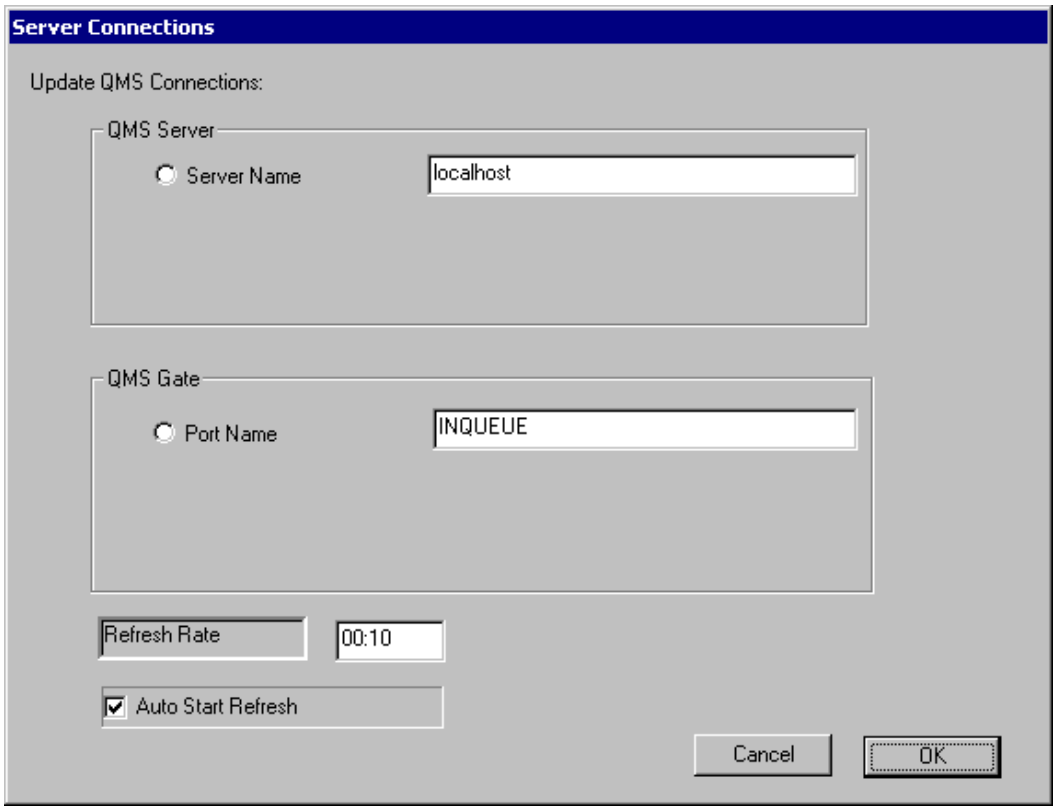
To	See
Add a Queue	"To Add a Server Connection" on page 38
Change a Queue	"To Change a Server Connection" on page 39
Delete a Queue	"To Delete a Server Connection" on page 39
Select a Queue	"To Select a Server Connection" on page 40

To Add a Server Connection

NOTE

For more information about setting up queues in the supported environments, see "Configuring Queues for TCP/IP" on page 21.

1 Click **Add** to display the Server Connections dialog box.



2 In the **Server Name** text box, type the name of the machine on which the queues reside. This host name must match a name listed in the HOSTS file (e.g., C:\Windows\System32\Drivers\Etc\Hosts).

3 In the **Port Name** text, box, type the name of the queue to which the jobs are placed. This service name must match a name listed in the SERVICES file (e.g., C:\Windows\System32\Drivers\Etc\Services).

4 In the **Refresh Rate** text box, type the interval time you want to use for polling the queue. The default value is five seconds. This time is honored by the Auto Refresh feature (see below).

5 Enable the **Auto Start Refresh** check box if you want Queuecommander to automatically poll the queue for the indicated time interval.

6 Click **OK** to save the Server Connection.

Queuecommander returns you to the QMS Server Connection dialog and lists the new/updated queue. The Server Connection information is written to the QCMDR32.INI file.

To Change a Server Connection

1 In the **Select a Connection:** list box, highlight the server connection you want to edit.

2 Click **Change** to display the Server Connections dialog box.

3 Complete the changes by following the procedure beginning with “Step 2” on page 39.

To Delete a Server Connection

1 In the **Select a Connection:** list box, highlight the server connection you want to delete.

2 Click **Delete** to remove the queue from the list box. The queue is also deleted from the QCMDR32.INI file. If you want to monitor the queue at a later date, you must recreate the queue.

To Select a Server Connection

NOTE

The server/queue connections you want to monitor with Queuecommander should be **active** before you add them to the list. If they aren't, you'll receive appropriate error warnings.

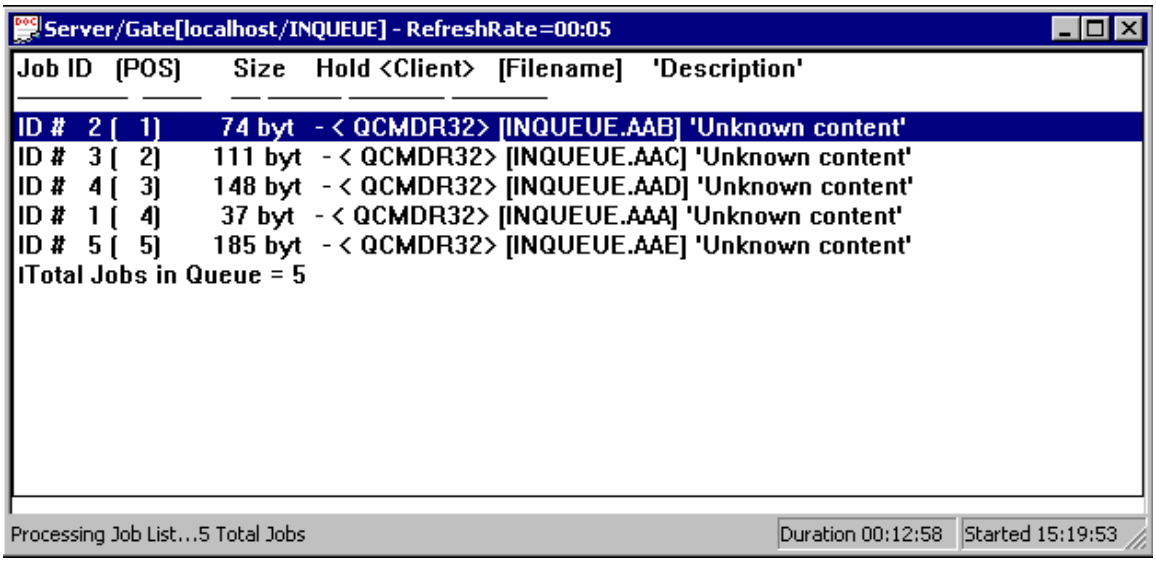
1 In the **Select a Connection:** list box, highlight the server connection you want to monitor. The list of available connections is obtained from the QCMDR32.INI file.

2 Click **OK** and Queuecommander displays the Job Queue Window.

TIP

The jobs are only listed if you chose "Auto Start Refresh" in the Server Connections dialog. To poll the queue and list its contents, select

- **List Jobs** from the Jobs menu,
- **Start Auto Refresh** from the QMS Server menu, or
- **Server Connections** from the Edit menu and Change the settings.



3 Repeat these steps for each additional queue you need to monitor.

When Queuecommander displays the Job Queue Window, the following fields are displayed:

Field	Contents
Job ID#	The Job ID Number is a sequential number assigned by Queuecommander. The jobs are assigned a number in the order in which they were submitted to the queue.
Position	The job's position in the queue. Since you can reorder jobs, the position number isn't necessarily the same as the Job ID.
Size	The size of the job, expressed in bytes.

Field	Contents
Hold	Queuecommander displays the status of the job in the queue. The three most common status are **Ready**, **Operator Held**, and **Processing**.
Client	Queuecommander displays the user ID supplied by the application or keyed by the user when the job was added to the Queue Server.
Filename	This is the generated file name of the job in the Queue Server, consisting of the queue name plus a three-character identifier.
Description	Queuecommander displays a description of the job that was supplied by the application or keyed by the user when the job was added to the Queue Server.

The Title Bar of the dialog box lists the Server Name/ Port Name, followed by the automatic refresh rate.

The following section discusses the menu options to manipulate the contents of the window.

File Menu

Creating a Connection

Use the New Connection command to establish a new monitoring window for a queue.

To Create a Connection

1. Select **New Connection** from the **File** menu.
   -or-
   Click  on the Toolbar.
   Queuecommander displays the Server Connections dialog box.
2. Follow the procedure to create a Server Connection beginning with “Step 2” on page 39.

Closing a Connection

Use the Close Connection command to terminate the monitoring of a queue.

To Close a Connection

1. If you’re monitoring more than one queue, select the queue monitoring window you want to close.
2. Select **Close Connection** from the **File** menu.

Queuecommander closes the queue monitoring window.

Closing Queuecommander

Use the Exit command to close Queuecommander.

To Close Queuecommander

- Select **Exit** from the **File** menu..

  Queuecommander closes and returns to the Windows Desktop.

Edit Menu

Managing Server Connections

You can use the Server Connections command to quickly access the area in which you service the active queues you monitor.

To Manage Server Connections

1. Select **Server Connections** from the **Edit** menu.
   Queuecommander displays the QMS Server Connection dialog box.
2. See the procedure on 38 to determine the appropriate action in maintaining queues.

View Menu

Toolbar

Use the Toolbar command to toggle the display of the handy shortcut icons.

To Toggle the Toolbar Display

- Select **Toolbar** from the **View** menu.

Queuecommander displays or hides the Toolbar icons.

Status Bar

Use the Status Bar command to toggle the display of the messages at the bottom of the Queuecommander window.

The Status Bar displays three separate items in its area.

Status/Message Description

This section of the Status Bar provides a message test that describes the current status of the Queue Server being monitored. This section also provides any informational or error messages describing any processing circumstances that may have been encountered during the monitoring of the Queue Server. Some of the messages are as follows:

- **Ready - Request /Start Job List** — received after a Queue Server connection has been made and Queuecommander is ready to begin monitoring.
- **Processing...** — a status message indicating a request has been sent to the Queue Server and you're waiting for the response.
- **Processing Job LIst... *nn* Total Jobs** — displays the total number of jobs in the Queue Server after the response has been received. If other applications are adding jobs to the Queue Server and/or retrieving and processing jobs, this number will necessarily change. The number of jobs displayed here is simply the number in the queue at the point in time that Queuecommander sent the request to the Queue Server.

- **Active** — shows that this window is in the “Auto Refresh” mode and is waiting on the elapsed time to expire before sending a request to the Queue Server and beginning it’s automatic monitoring of that Queue Server.
- **Inactive** — indicates that the auto refresh has been stopped.

Duration: hh:mm:ss

The amount of time, displayed in *hours:minutes:seconds* format, since the Queue Server monitoring process began. This duration is the elapsed time since the connection to the Queue Server was initially made. This figure is only updated when some request is completed by the Queue Server.

Started: hh:mm:ss

The time, displayed in *hours:minutes:seconds* format, when the Queue Server processing started. This time represents the starting point of the connection to the Queue Server. This time doesn’t change with requests and responses.

To Toggle the Status Bar Display

▶ Select **Status Bar** from the **View** menu.

Queuecommander displays or hides the Status Bar message.

QMS Server Menu

Start Auto Refresh

Use the Start Auto Refresh command to begin polling the queue at specified intervals.

> **NOTE**
> The Start/Stop Auto Refresh icons indicate the state to which you'll be changing. For example, the “stop light” is green if you're currently viewing a manually-refreshed queue and you want to make it automatic; whereas the “stop light” is red if you're viewing an automatically-refreshed queue and you want to make it manual.

To Start Auto Refresh

1. If you're monitoring more than one queue, select the queue monitoring window you want to automatically poll.
2. Select **Start Auto Refresh** from the **QMS Server** menu.

   -or-

   Click  on the Toolbar.

Queuecommander begins polling the queue using the specified time interval. If you want to change the time interval, you must close, edit, and then restart the connection. For more information about changing the time interval, see "To Change a Server Connection" on page 39.

Stop Refresh

Use the Stop Refresh command to stop automatic polling of the queue.

> **NOTE**
> The Start/Stop Auto Refresh icons indicate the state to which you'll be changing. For example, the “stop light” is green if you're currently viewing a manually-refreshed queue and you want to make it automatic; whereas the “stop light” is red if you're viewing an automatically-refreshed queue and you want to make it manual.

To Stop Refresh

1. If you’re monitoring more than one queue, select the queue monitoring window for which you want to terminate automatic polling.
2. Select **Stop Refresh** from the **QMS Server** menu.

   -or-

   Click  on the Toolbar.

Queuecommander halts the automatic polling of jobs in the queue.

Halt QMS Server

Use the Halt QMS Server command to temporarily suspend the monitoring of jobs in the queue. This action doesn’t affect the jobs in the queue, only the processing thereof.

To Halt QMS Server

1. If you’re monitoring more than one queue, select the queue monitoring window for which you want to suspend processing.
2. Select **Halt QMS Server** from the **QMS Server** menu.

   -or-

   Click  on the Toolbar.

Queuecommander halts the processing of jobs in the queue.

Resume QMS Server

Use the Resume QMS Server command to reactivate processing of a job queue.

To Resume QMS Server

1. If you’re monitoring more than one queue, select the queue monitoring window for which you want to reactivate processing.
2. Select **Resume QMS Server** from the **QMS Server** menu.

   -or-

   Click  on the Toolbar.

Queuecommander resumes the processing of jobs in the queue.

Shut Down QMS Server

Use the Shut Down QMS Server command to terminate the ISIQMSD queue daemon application (i.e., ISIQMSD.EXE is the “server” and Queuecommander is the “client”). If you want to monitor the affected queue again, you must restart ISIQMSD.

To Shut Down QMS Server

1. If you're monitoring more than one queue, select the queue monitoring window for the server connection you want to terminate.
2. Select **Shut Down QMS Server** from the **QMS Server** menu.

Queuecommander automatically terminates the server application; however, it doesn't affect the jobs in the queue.

NOTE

For more information about starting queues in the supported environments, see "Configuring Queues for TCP/IP" on page 21.

Jobs Menu

List Jobs

Use the List Jobs command to manually poll the queue and list all jobs awaiting processing.

TIP

You can use the Start Auto Refresh command to automatically poll the queue at specified time intervals. For more information, see "Start Auto Refresh" on page 43.

To List the Jobs

1. If you're monitoring more than one queue, select the queue monitoring window for which you want to obtain a list of waiting jobs.
2. Select **List Jobs** from the **Jobs** menu.

   -or-

   Click  on the Toolbar.

Queuecommander polls the queue and displays the current jobs.

Add Job

Use the Add Job command to insert a job into the queue. If you've stored a job from an earlier process to a file, you can add that job to the job queue for processing now.

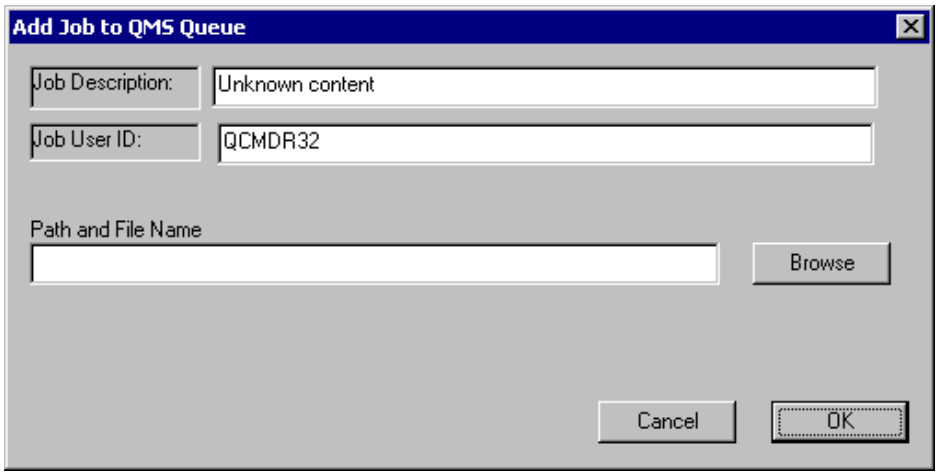
To Add a Job

1. If you're monitoring more than one queue, select the queue monitoring window in which you want to add a job.
2. Select **Add Job** from the **Jobs** menu.

   -or-

   Click  on the Toolbar.

Queuecommander displays the Add Job to QMS Queue dialog box and prompts you for the path and file name of the job you want to add to the job queue.



3 In the **Job Description:** text box, type a description of the contents, if known.

4 In the **Job User ID:** text box, type the ID of the machine submitting the job.

5 In the **Path and File Name** text box, type the path and file name of the job you want to add.
-or-
Click **Browse** to locate the file.

6 Click on **OK** to proceed with the task or **Cancel** to exit the procedure.

Queuecommander adds the job to the queue.

Get Job

Use the Get Job command to retrieve a job from the queue and save it to a local file. After a successful save, Queuecommander permanently removes the job from the queue. Queuecommander displays the Save As dialog box and prompts you for the path and file name to which you want to save the job.

To Get a Job

1 If you're monitoring more than one queue, select the queue monitoring window in which you want to get (save) a job.

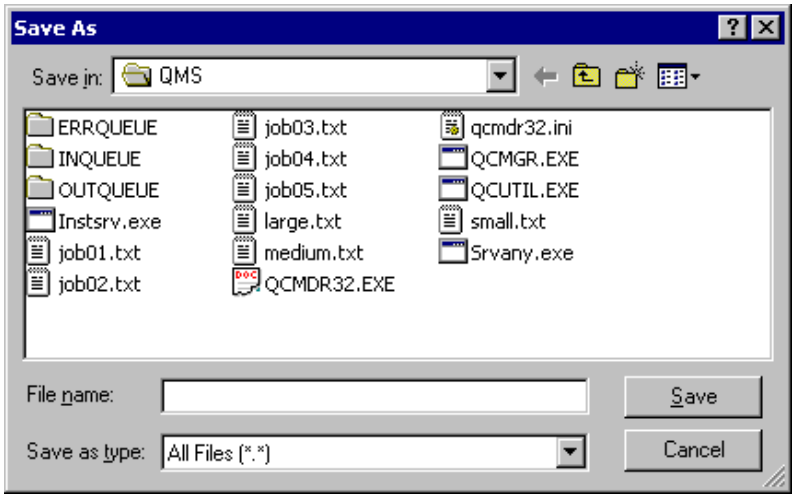
2 Highlight the job you want to save.

3 Select **Get Job** from the **Jobs** menu.

-or-

Click  on the Toolbar.

Queuecommander displays the Save As dialog box.



4 In the **Save in** drop-down list box, select the directory where you want to save the job.

5 In the **File name** list box, type the name you want to assign to the file.

6 Click **Save**.

Queuecommander saves your job to the specified file.

Change Job Position

Use the Change Job Position command to alter the processing order of the jobs in the queue. While it's true the jobs are initially ranked in the order in which they were submitted, you might find a need to reorganize the order of the jobs. All jobs are processed via the FIFO method (i.e., first-in, first-out). You can easily manipulate the processing order of the jobs in the queue by changing the order (e.g., move a job to a “later” position for “later” processing, and vice versa).

To Change the Job Position

1 If you’re monitoring more than one queue, select the queue monitoring window in which you want to reorder a job.

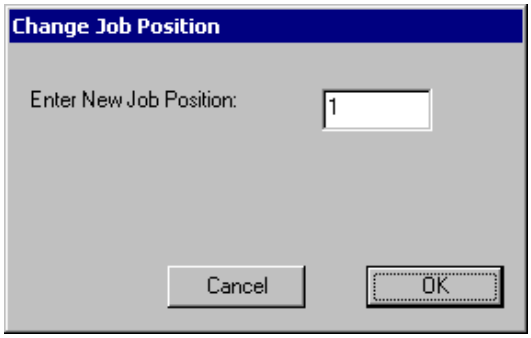
2 Highlight the job you want to move.

3 Select **Change Job Position** from the **Jobs** menu.

-or-

Click  on the Toolbar.

Queuecommander displays the Change Job Position dialog box.



4 In the **Enter New Job Position:** text box, type the number representing the new place in the queue.

Queuecommander places the job in the new position and shuffles the other jobs accordingly.

Hold a Job in QMS Queue

Use the Hold a Job in QMS Queue commands to temporarily restrict a job from processing. You can choose between two hold states, as follows:

- **Hold Operator** — places the job in an operator hold status.
- **Hold User** — places the job in a user hold status.

After you've held a job, a status code displays in the Hold field of the queue monitoring window: **O** for Hold Operator, **U** for Hold User, or **B** for both. You can use the Release command to remove the hold status of a job; however, hold status codes are independent of one another, meaning you can place multiple codes on a single job.

TIP

If you place an Operator Hold on a job, the status becomes **O**. If you then place a User Hold on the same job, the status changes to **B**.

To Hold a Job

1 If you're monitoring more than one queue, select the queue monitoring window in which you want to hold a job.

2 Highlight the job you want to hold.

3 Select **Hold Operator** from the **Jobs>Hold a Job in QMS Queue** sub-menu.
-or-
Select **Hold User** from the **Jobs>Hold a Job in QMS Queue** sub-menu.

Queuecommander places a hold on the indicated job.

Release Job

Use the Release Job commands to resume processing a job. This command removes the hold status placed on the job by the Hold command. After you've released a job, a hyphen (ready state) displays in the Hold field of the queue monitoring window. Since the hold status codes are independent of one another, releasing the status code of a job with multiple hold codes only releases that particular hold code, unless you use the Release Both Hold types command (see below).

To Release a Job

1 If you're monitoring more than one queue, select the queue monitoring window in which you want to release a job.

2 Highlight the job you want to release.

3 Do one of the following:

If you want to release	See
a job with a Hold status of O	▪ Select **Release Operator Hold** from the **Jobs>Release Job** sub-menu.
a job with a Hold status of U	▪ Select **Release User Hold** from the **Jobs>Release Job** sub-menu.
a job with a Hold status of B	▪ Select **Release Both Hold types** from the **Jobs>Release Job** sub-menu.

If you want to release	See
all jobs	▪ Select **Check to Release All Jobs** from the **Jobs>Release Job** sub-menu.

Queuecommander removes the hold code and releases the job.

Delete a Job

Use the Delete a Job command to permanently remove a job from the queue. You should, however, use caution with this command, as the only recourse is to resubmit the job.

To Delete a Job

1. If you're monitoring more than one queue, select the queue monitoring window in which you want to delete a job.
2. Highlight the job you want to delete.
3. Select **Delete a Job** from the **Jobs** menu.

   -or-

   Click  on the Toolbar.

Queuecommander deletes the indicated job from the queue.

Drain>Delete All Jobs

You can use the Drain>Delete All Jobs command when you need to permanently delete all the jobs in the queue, but not delete the queue itself. In other words, you'll see an empty job queue after you've drained the queue.

To Drain>Delete All Jobs

1. If you're monitoring more than one queue, select the queue monitoring window for which you want to drain/delete all the jobs.
2. Choose **Drain>Delete All Jobs** from the **Jobs** menu.

Queuecommander removes all the jobs from the queue, leaving the queue itself intact.

Window Menu

The Window menu contains commands that let you manipulate the different windows on the Queuecommander main window. You can reorder and rearrange the windows at any time without affecting the information inside.

Cascade

Use the Cascade command to produce a staggered, layered arrangement of all windows which are currently open in the Queuecommander main window. Because the Tile command won't re-size the windows as it tries to fit them in the available main window space, we recommend you choose **Cascade** rather than **Tile** to manage access to windows.

Once the windows are arranged with their title bars showing, you can click on the title bar to activate a particular window and bring it to the foreground in the main window.

Tile

Use the Tile command to arrange and maximize the display areas of all windows which are currently open in the Queuecommander main window. We recommend you choose **Cascade** rather than **Tile** to manage access to multiple windows because the Tile command won't re-size the windows as it tries to fit them in the available main window space.

Once the windows are tiled in the main window, you can click inside a window to activate and bring it to the foreground in the main window.

Help Menu

The Help menu contains helpful hints about the use and maintenance of Queuecommander.

About

To get the version number, product type, and copyright date for your current version of Queuecommander, choose **Help>About...**.

Customizing Queue Systems

Overview

Docucorp Queuecommander and Queue Systems are primarily designed as mechanisms to transfer work-in-progress jobs between Docucorp products. If you need to transfer jobs in a unique fashion, you can access the queue functions contained in the Docucorp API.

Using the Docucorp API Functions

Docucorp provides a set of functions which you can use to customize the way in which Queuecommander and Queue Systems transfer your jobs. The Docucorp API functions reside in a Dynamic Link Library (DLL). You can access these DLLs and their functions in an application written for your enterprise's processing environment. For more information about customizing Queuecommander and Queue Systems, please see the *Docucorp API (DAPI) Functions Reference Guide*.

Manually Operating ISIQMSD

Installing ISIQMSD

NOTE

The installation routine installs the Queue Daemon as an "Automatic" NT Service. This section is only for those wanting a better understanding of the procedure.

You can operate the Docucorp Queue Management System (QMS) Queue Daemon (ISIQMSD) as a Windows NT/2000 Service. An application operating as a Windows Service allows the machine to be logged off without stopping the application process. This feature allows an application like ISIQMSD to continue operating and, therefore, to continue to service client application requests. Also, the ISIQMSD application, while operating as a Service, doesn't have to be assigned to any specific Windows "user" ID. This enhancement to ISIQMSD is implemented with a new execution parameter (e.g., **-instserv**, including the hyphen). This parameter instructs ISIQMSD.EXE to install and start a version of itself as a Windows NT Service, as opposed to its normal command-line invocation.

To Install ISIQMSD

- Type the following syntax to invoke ISIQMSD as a Windows Service or UNIX Daemon:

```
ISIQMSD <-parm> <queue Name> <service Port> <ENTER>
```

where:

Parameter	Explanation
ISIQMSD	The name of the executable file.
-INSTSERV	(**Windows Service only**) Install and start Docucorp Queue Management System as a Windows Service.
-REMSERV	(**Windows Service only**) Stop and remove Docucorp Queue Management System from the Windows Services pool.
-DAEMON	(**UNIX only**) Run as a UNIX system Daemon process.
queue Name	This is the logical user-supplied name of a given queue. If the queue doesn't exist, one is established using the **queue name**. If the queue name does exist, it will be re-established. This value is optional; if you don't supply the queue name, the queue and service port are created with the default values (e.g., **defaultQ 5560**).
service Port	This is the desired TCPIP service port number. This value is optional; if you omit the service port number when specifying the queue name, the queue Name is matched against the services file to resolve the port number (see "Configuring Queues for TCP/IP" on page 21).

Configuring ISIQMSD in Windows

The following procedure lists the steps necessary to manually configure the newly installed Windows NT Service to operate the QMS Queue Daemon, ISIQMSD.EXE.

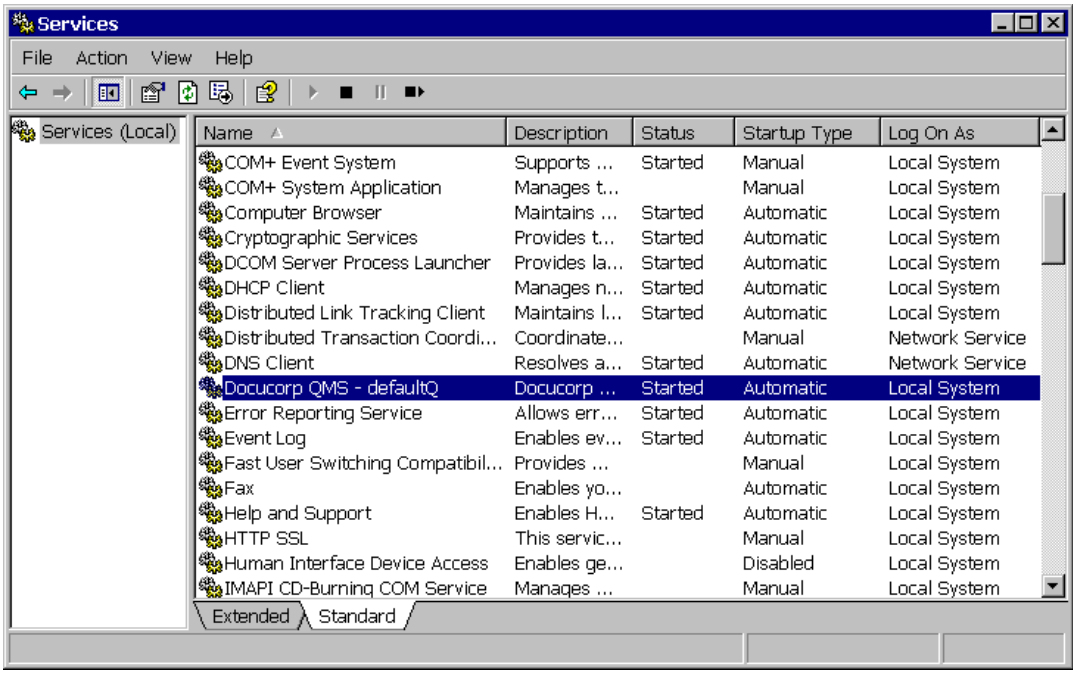
This procedure also assumes you have some knowledge of installing, configuring, and managing Windows NT Services. The README file included with the Windows NT 3.5 Resource Kit contains a more detailed explanation.

To Configure ISIQMSD in Windows

1 Set the Service attributes for ISIQMSD.EXE.

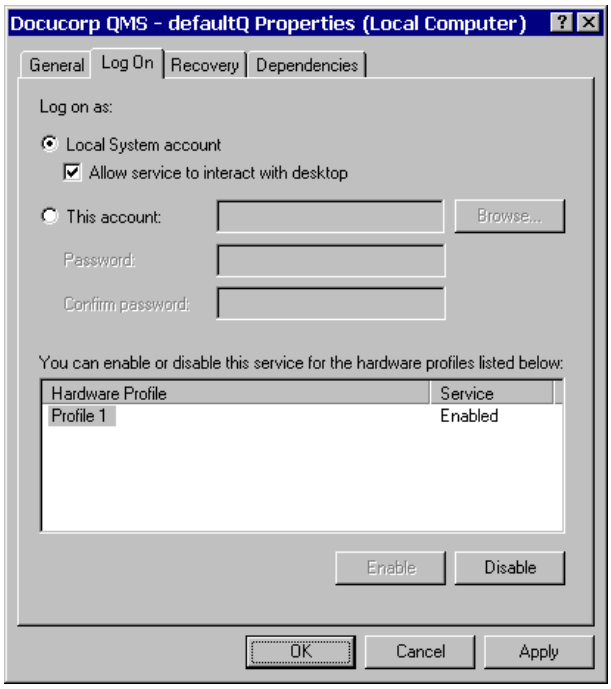
a From the **Start** menu, select **Settings**, then **Control Panel**.

b Once the Control Panel displays, double-click **Administrative Tools**, then double-click **Services**.



c Select the **Service** name (**Docucorp QMS - defaultQ**), then click the **Properties** tab.

d Click the **Log On** tab.



e Under the **Log on as:** group box, select the **Local System account** option button, then check **Allow service to interact with desktop**.

2 Click **OK** to save the settings.

NOTE

If you want to configure/update manually the Windows Registry to supply the required execution parameters for the Service, the README file included with the Windows NT 3.5 Resource Kit contains a more detailed explanation.

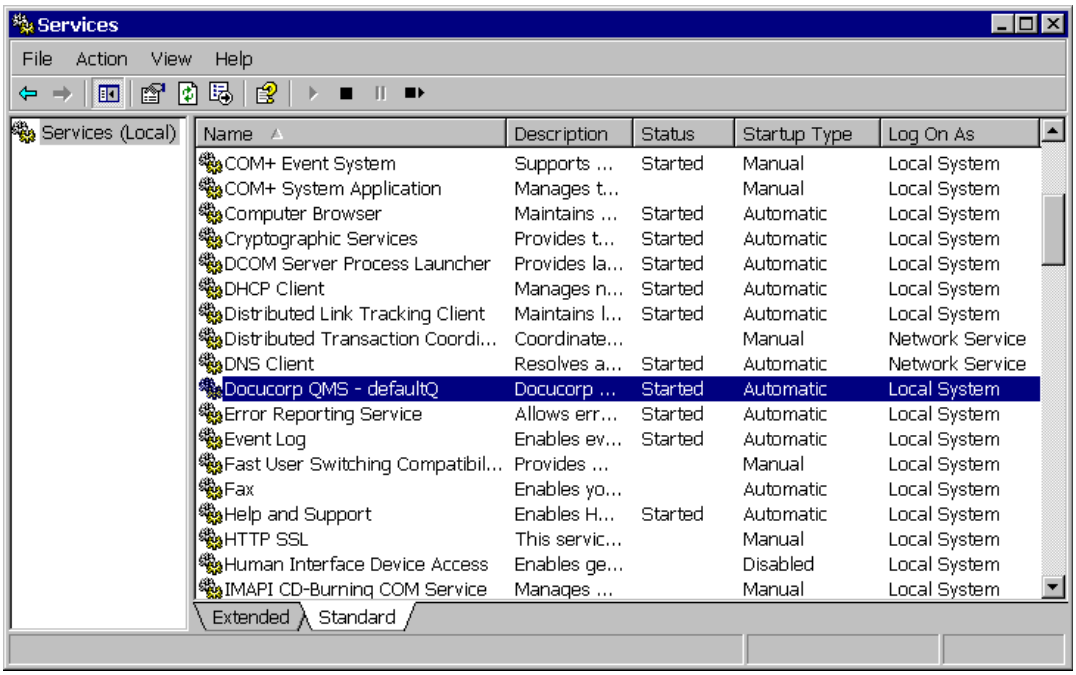
Starting ISIQMSD in Windows

The following procedure lists the steps necessary to manually start the Windows NT Service. This procedure also assumes you have some knowledge of installing, configuring, and managing Windows NT Services.

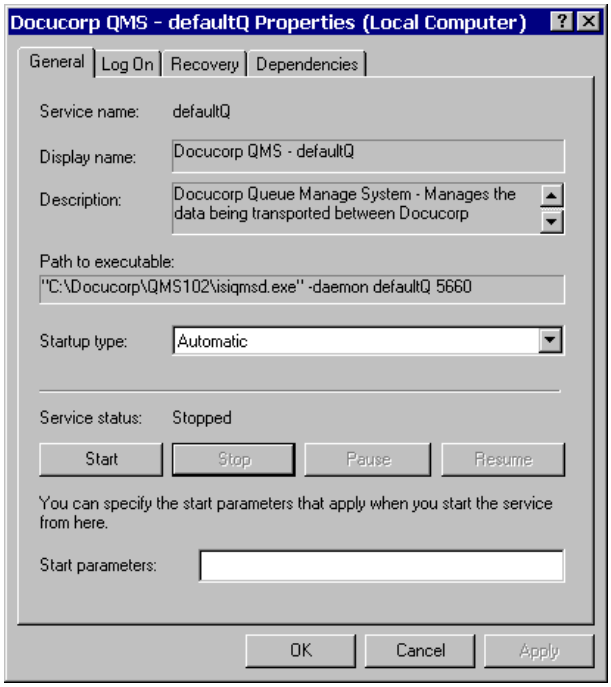
To Start ISIQMSD in Windows

1 From the **Start** menu, select **Settings**, then **Control Panel**.

2 Once the Control Panel displays, double-click **Administrative Tools**, and then double-click **Services**.



3 Select the **Service** name (**Docucorp QMS - defaultQ**), and then click the **Properties** tab.



4 In the **Service status:** group box, click **Start** to start the queue. QMS displays a window verifying that the queue daemon is operating. You should minimize this window.

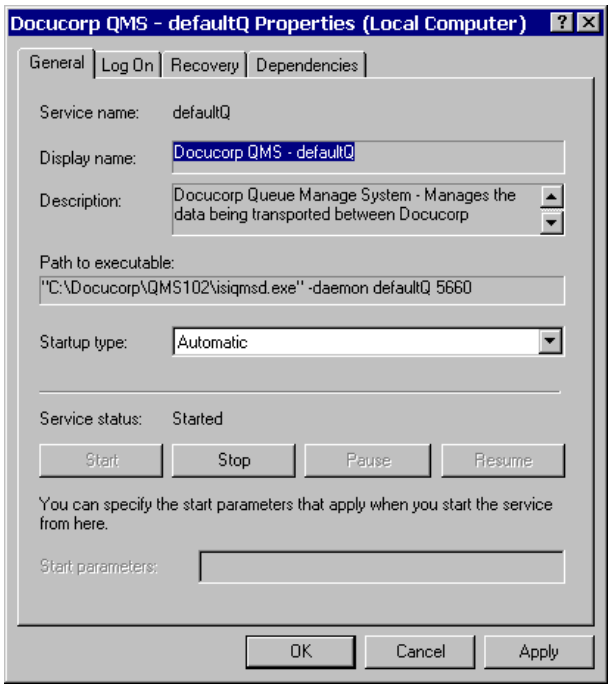
5 Click **OK** to save the settings.

Stopping ISIQMSD in Windows

The following procedure lists the steps necessary to manually stop the Windows NT Service. This procedure also assumes you have some knowledge of installing, configuring, and managing Windows NT Services.

To Stop ISIQMSD in Windows

1. From the **Start** menu, select **Settings**, and then Control Panel.
2. Once the Control Panel displays, double-click **Administrative Tools**, and then double-click **Services**.
3. Select the **Service** name (**Docucorp QMS - defaultQ**), and then click the **Properties** tab.



4. In the **Service status:** group box, click **Stop** to stop the queue.
5. Click **OK** to save the settings.

Uninstalling QMS from Windows

When uninstalling QMS, you can just remove ISIQMSD from Windows Service Management control or remove all QMS components from your workstation.

To remove:	See:
ISIQMSD as a Service	"To Remove ISIQMSD from Windows Service Management Control" on page 57
all QMS components	"To Uninstall QMS Components" on page 57

To Remove ISIQMSD from Windows Service Management Control

- Type the following syntax to stop the Server and remove it from Windows Service Management control:

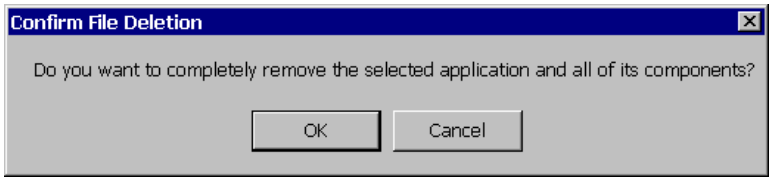
```
isiqmsd.exe -remserv queueName
```

To Uninstall QMS Components

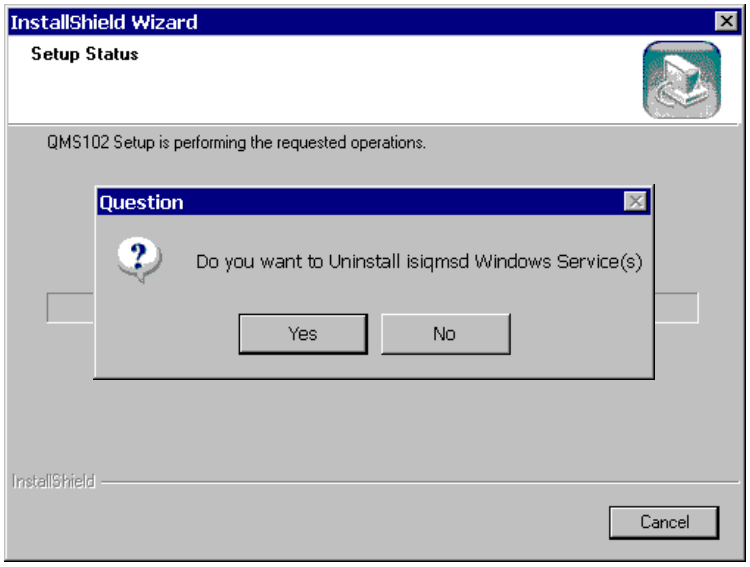
You can uninstall QMS components in one of the following ways:

1. (Preferred method) **Start>Programs>Control Panel>Add or Remove Programs**.
   -or-
   Rerun the Windows installation media.

   The following confirmation dialog displays.

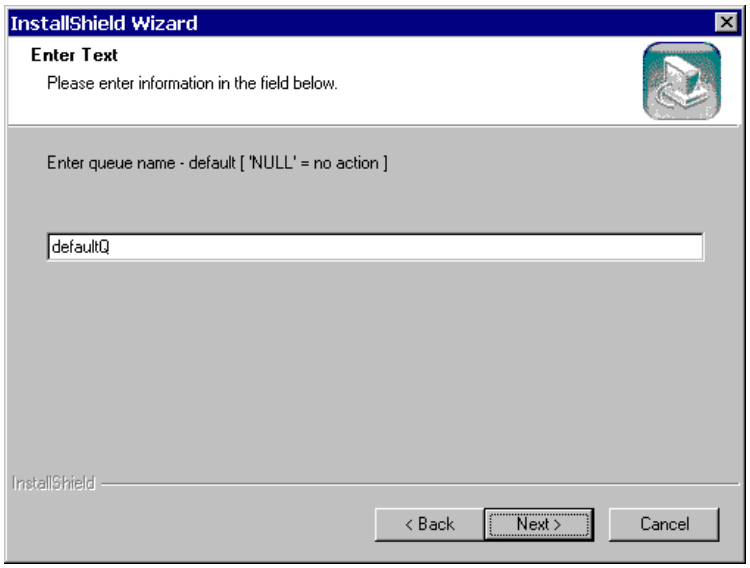


2 Click **OK** to uninstall the QMS components or **Cancel** to abort the process.



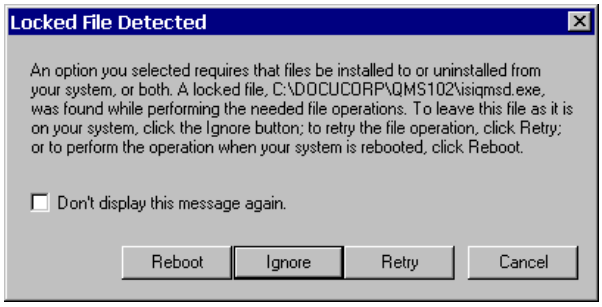
3 Click **Yes** to stop and remove active queues from the Windows Service pool or **No** to skip this option.

If you chose **Yes**, type the desired queue Name—the logical user-supplied name of a given queue.



4 Click **Next** to proceed with the routine or **Back** to return to previous screens and make changes.

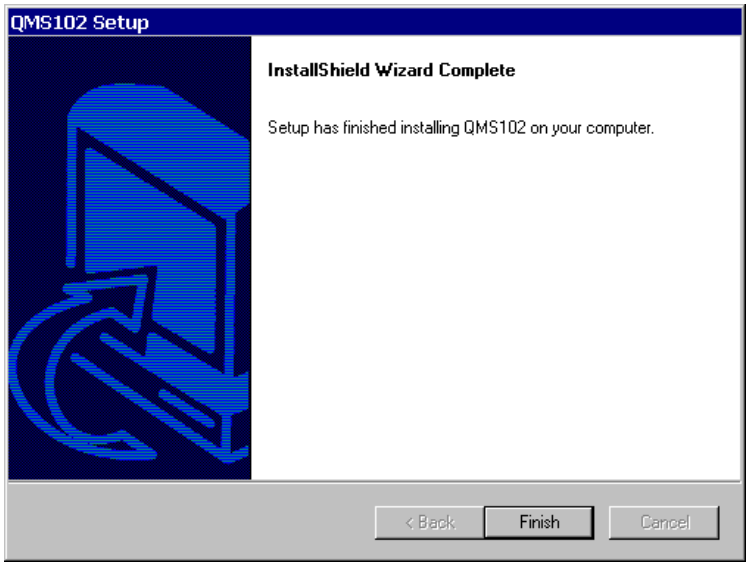
If you're presented with a "Locked File Detected" dialog, click **Ignore** to proceed with the un-installation process. If this occurs, you must manually remove the ISIQMSD module (see "To Remove the ISIQMSD Module Manually" on page 59).



5 If you haven't already done so, click **Ignore** and the following dialog displays.

6 Click **Yes** to uninstall another Windows Service or **No** to decline this option.

When the routine is complete, the program displays the InstallShield Wizard Maintenance Complete dialog box.



7 Choose **Finish** to complete the routine and return to Windows.

To Remove the ISIQMSD Module Manually

1 From a command prompt, change to the directory where the ISIQMSD executable resides.

2 Type `isiqmsd –remserv defaultQ`

3 Type `erase isiqmsd.exe`

Installing ISIQMSD as a UNIX Daemon

You can operate the Docucorp Queue Management System (QMS) Queue Daemon (ISIQMSD) as a UNIX Daemon Service. An application operating as a UNIX Daemon allows the machine to be logged off without stopping the application process. This feature allows an application like ISIQMSD to continue operating and, therefore, to continue to service client application requests. This enhancement to ISIQMSD is implemented with the execution parameter (e.g., **-daemon**, including the hyphen). This parameter instructs ISIQMSD to invoke a version of itself as a UNIX Daemon Service, as opposed to its normal command-line invocation.

To Install ISIQMSD as a UNIX Daemon

- See "To Install ISIQMSD" on page 53.

Configuring ISIQMSD as a UNIX Daemon

The following procedure lists the steps necessary to manually configure the newly-installed application to operate as a UNIX Daemon.

To Configure ISIQMSD as a UNIX Daemon

- Before invoking ISIQMSD as a UNIX Daemon, have your **systems administrator** create the following directory entry on the machine where ISIQMSD is to run:

  ```
  /usr/spool/isiqms
  ```

  This directory should have read\write\create permissions.

To Start ISIQMSD as a UNIX Daemon

- You should type the following syntax to invoke ISIQMSD as a Daemon:

  ```
  isiqmsd -daemon queueName servicePort
  ```

To Stop and Remove ISIQMSD as a UNIX Daemon

1. Using the **ps** command (e.g., ps –ef), locate the process ID (pid) of the daemon that you want to stop.
2. Using the **kill** command (e.g., kill **pid**) with the process ID (pid) that you want to stop.
3. Issue the **rm** command (e.g., rm **isiqmsd**) to remove the ISIQMSD application module.

Glossary

distributed processing
A form of information processing in which work is performed by separate computers that are linked through a communications network.

FIFO
A method of processing jobs where the jobs are processed in the order in which they were submitted (e.g., the first job submitted is processed before the second job, the second job is submitted before the third job, and so on).

file server
A file storage device on a local area network that's accessible to all users on the network. A file server is a sophisticated device that not only stores files but manages them and maintains order as network users request files and make changes to them.

On local area networks, a file server is often a computer with a large hard disk that's dedicated only to the task of managing shared files.

LAN
Local Area Network. A communications network that interconnects a variety of office equipment and computers within a fairly small area, such as an office building.

product server
A computer on a network dedicated to processing jobs for a single product. The processing may include data transfers and interaction with other products, informing the user when a given task is complete, and routing data to a printer or specified file.

queue
A temporary holding area on the network where jobs are stored for later processing. The jobs are then processed in the order in which they were submitted, commonly referred to as FIFO (First-In, First-Out). See also *FIFO*.

Index