



Installation Guide

QAD Production Scheduler

This document contains proprietary information that is protected by copyright and other intellectual property laws. No part of this document may be reproduced, translated, or modified without the prior written consent of QAD Inc. The information contained in this document is subject to change without notice.

QAD Inc. provides this material as is and makes no warranty of any kind, expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. QAD Inc. shall not be liable for errors contained herein or for incidental or consequential damages (including lost profits) in connection with the furnishing, performance, or use of this material whether based on warranty, contract, or other legal theory.

QAD and MFG/PRO are registered trademarks of QAD Inc. The QAD logo is a trademark of QAD Inc.

Designations used by other companies to distinguish their products are often claimed as trademarks. In this document, the product names appear in initial capital or all capital letters. Contact the appropriate companies for more information regarding trademarks and registration.

Copyright ©2008 by QAD Inc.

QAD Inc.

100 Innovation Place

Santa Barbara, California 93108

Phone (805) 684-6614

Fax (805) 684-1890

<http://www.qad.com>

About This Guide	1
Installation Errata	2
Other Documentation	2
Online Help	3
QAD Web Site	3
Conventions	4
Menu References	4
Typographic	4
UNIX and Windows Conventions	5
Chapter 1 Installing and Converting QAD Production Scheduler ..	7
Overview	8
Architecture	8
Deployment	9
Implementing QAD Production Scheduler with QAD ERP on Oracle .	10
Conversion Overview	10
Installation and Conversion Summary	11
System Requirements	12
Operating Systems	12
Language Support	13
Prerequisites	14
Installation Worksheet	15
Modify the Services File	16
Conversion Prerequisites	18
Multiple QPS Version Prerequisites	18
Installing from Media	19
Mount the CD-ROM	19
Install the QAD Production Scheduler Application	19
QAD Production Scheduler Component Installs	20
Running the New Install Workflow	23
Set Up QAD Production Scheduler	25
Create qpsempty	26
Create qpsprod	28
Create Windows Icons or UNIX Scripts	32

Generate Sample QAD Production Scheduler Broker File	35
Compile QAD Production Scheduler	36
Running the Update Workflow	37
Set Up QAD Production Scheduler	39
Update qpsprod	41
Convert the Database	42
Update Schema	43
Set Up QAD Production Scheduler Database Set	44
Create Windows Icons or UNIX Scripts	45
Generate Sample QAD Production Scheduler Broker File	46
Compile QAD Production Scheduler	48
Configuring WebSpeed	49
Oracle Modifications	50
Append the Sample File	50
Copy WebSpeed Files to Apache	51
Configuring Additional Instances	51
Starting QAD Production Scheduler	54
Enter a Site to Test Background Processes	55
Validating the WebSpeed Broker	56
Test the QAD Production Scheduler Interface	59
Launching Background Processes	61
Modify Log Locations for Background Processes (Optional)	61
Launching the Background Processes	62
Configuring QXtend	63
Summarized QXtend Setup	63
Loading QAD Production Scheduler QDoc Schemas	64
Installation Reference	67
Running QAD Production Scheduler with QAD ERP Oracle	68
Starting QAD Production Scheduler	72
Stopping QAD Production Scheduler	73
Multi-Tier Installations	74
Configuring a Remote Messenger for QAD Production Scheduler	76



About This Guide

Installation Errata 2

Other Documentation 2

Online Help 3

QAD Web Site 3

Conventions 4

This guide describes how to install QAD Production Scheduler and upgrade it from earlier versions.

To learn about the QAD Production Scheduler on a functional basis, see the user guide.

Note This guide applies to multiple versions of QAD’s enterprise resource planning (ERP) application, from MFG/PRO eB through the current release. To account for product name changes, the guide refers to that product as the ERP application or QAD ERP.

Installation Errata

In addition to these instructions, you may receive a supplementary errata sheet with changes and additional instructions. Check your product package. In addition, when you begin the installation, always check the QAD Support Web site to verify that you have the latest version. See “QAD Web Site” on page 3 for details.

Other Documentation

- For an overview of new QAD Production Scheduler features and software updates, see the *Release Notes*.
- For software installation instructions for other QAD products, refer to the appropriate installation guide for your system.
- For information on installing and using QXtend, see *Technical Reference: QAD QXtend*.
- For information on using QAD ERP applications, refer to the *User Guides*.
- To view documents online in PDF format, see the *Documents on CD* and *Supplemental Documents on CD*. The CD-ROM media includes complete instructions for loading the documents on a Windows network server and making them accessible to client computers.

Note Installation guides are not included on a CD. Printed copies are packaged with your software. Electronic copies of the latest versions are available on the QAD Web site.

- For information on installing and setting up Progress WebSpeed, see the Progress documentation.

Online Help

To view user information from within the QAD Production Scheduler application, click Help.

The browser-based help system includes a table of contents and index of fields, as well as a keyword search tool.

To display the information about a particular function, click User Interface in the table of contents, then select a function.

QAD Web Site

QAD's Web site provides a wide variety of information about the company and its products. You can access the Web site at:

<http://www.qad.com>

For users with a QAD Web account, product documentation is available for viewing or downloading from the QAD Online Support Center at:

<http://support.qad.com/>

You can register for a QAD Web account at the QAD Online Support Center. Your customer ID number is required. Access to certain areas is dependent on the type of agreement you have with QAD.

Most user documentation is available in two formats:

- Portable document format (PDF). PDF files can be downloaded from the QAD Web site to your computer. You can view them with the free Adobe Acrobat Reader. A link for downloading this program is also available on the QAD Web site.
- HTML. You can view user documentation through your Web browser. The documents include search tools for easily locating topics of interest.

Features also include an online solution database to help QAD Production Scheduler users answer questions about setting up and using the product. Additionally, the QAD Web site has information about training classes and other services that can help you learn about QAD Production Scheduler.

Conventions

Menu References

This guide refers to multiple versions of QAD ERP applications. If menus were reorganized, differences in menu numbers are noted, when necessary, using this format:

User Maintenance (36.3.1; 36.3.18 before eB2.1)

The initial menu number identifies the program in the most recent release. The second menu number applies to the release specified and any earlier releases.

Typographic

This document uses the text or typographic conventions listed in the following table.

If you see:	It means:
monospaced text	A command or file name.
<i>italicized monospaced text</i>	A variable name for a value you enter as part of an operating system command; for example, <i>YourCDROMDir</i> .
indented command line	A long command that you enter as one line, although it appears in the text as two lines.
Note	Alerts the reader to exceptions or special conditions.
Important	Alerts the reader to critical information.
Warning	Used in situations where you can overwrite or corrupt data, unless you follow the instructions.

UNIX and Windows Conventions

This document supports the installation of QAD Production Scheduler for both UNIX and Windows platforms. Some steps are unique to a particular platform and are documented in separate sections and marked as UNIX or Windows only. In steps that are common to UNIX and Windows, UNIX file and path conventions are used when needed, unless otherwise noted. If you are installing on the Windows platform, substitute the drive letter and path conventions for your operating system.

6 Installation Guide — QAD Production Scheduler

Installing and Converting QAD Production Scheduler

This chapter includes an overview of the QAD Production Scheduler installation and conversion, system requirements, and specific instructions for installing or upgrading the product.

Information about how the application works is provided in the user guide.

<i>Overview</i>	8
<i>System Requirements</i>	12
<i>Prerequisites</i>	14
<i>Installing from Media</i>	19
<i>Running the New Install Workflow</i>	23
<i>Running the Update Workflow</i>	37
<i>Configuring WebSpeed</i>	49
<i>Configuring Additional Instances</i>	51
<i>Starting QAD Production Scheduler</i>	54
<i>Launching Background Processes</i>	61
<i>Configuring QXtend</i>	63
<i>Installation Reference</i>	67

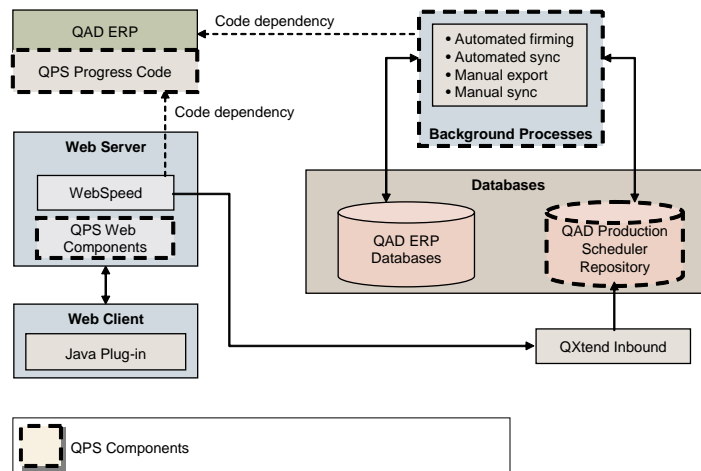
Overview

QAD Production Scheduler is a Web-based application that leverages a variety of demand, supply, inventory, and MRP data from the ERP database into a single workbench. It is used with Work Orders, Advanced Repetitive, and Repetitive.

Architecture

The QAD Production Scheduler consists of a database, a Web application component, a Progress application component, and background processes. QAD Production Scheduler also requires an instance of a QAD ERP application and an interoperability component.

Fig. 1.1
QAD Production Scheduler Deployment Architecture



In operation, QAD Production Scheduler accesses the ERP database through QXtend Inbound, updating the QAD Production Scheduler database with production schedule data. You can create, update, and delete production schedules in QAD Production Scheduler; the changes are made in the underlying ERP database automatically through QXtend.

The architecture supports four background processes, listed in Figure 1.1. ERP data is extracted using triggers in the Progress code and updates the QAD Production Scheduler database. Modifications to the ERP database are exported from QAD Production Scheduler and input through QXtend Inbound.

Deployment

In general, all QAD Production Scheduler components can reside on the same host. To install QAD Production Scheduler over multiple hosts, code and licensing dependencies must be taken into consideration. The most distributed multi-tier environment is shown in Figure 1.2 on four separate hosts.

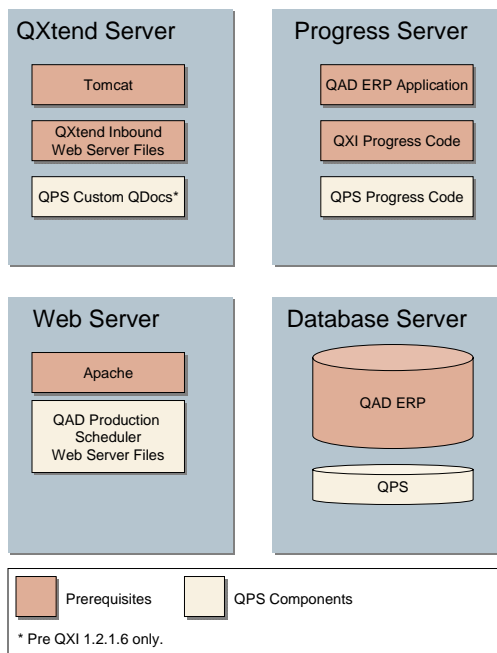


Fig. 1.2
Multi-tier QAD
Production
Scheduler
Deployment

The QAD Production Scheduler system is installed in three separate locations. Code and scripts are deployed to:

QPSInstallDir. This is the QAD Production Scheduler Progress code and is the location of the processing layer. The default location is the `/qps` directory under your existing *QADInstallDir*.

You create the database later on and can modify the standard database creation scripts to determine the host where the database resides.

Web Server Home Directory (Doc Root). This is the Web server layer. The default Apache location is `ApacheHome/html/qps`. On Windows, the Apache directory is `ApacheHome\htdocs`.

QXtend Inbound Webapp Directory. This is the QXtend Inbound Web installation directory, under the Web server home directory. Typically, this is `TOMCAT_HOME/webapps/qxtendserver`. Custom QDocs are installed here, but are only required for QXtend Inbound versions prior to 1.2.1.6.

The multi-tier installation process is covered in greater detail in “Multi-Tier Installations” on page 74.

Implementing QAD Production Scheduler with QAD ERP on Oracle

After either a new QAD Production Scheduler install or an update, you can modify the QAD Production Scheduler scripts to connect to QAD ERP Oracle databases. The QAD Production Scheduler database cannot be an Oracle database. However, QAD Production Scheduler can be run with QAD ERP Oracle implementations. To accomplish this, review the steps in “Running QAD Production Scheduler with QAD ERP Oracle” on page 68.

Conversion Overview

If you are converting from an earlier release of QAD Production Scheduler, you will go through many of the same steps that you would do a new install. The only difference is that instead of running the basic QAD Production Scheduler QPS/UTIL workflow, you will run the Update QAD Production Scheduler workflow.

There are detailed instructions in each section of the text.

Installation and Conversion Summary

Prerequisites

- Progress, Tomcat, and Java are installed.
- Environment variables are set.
- QAD ERP is installed and compiled.
- QXtend Inbound is installed and compiled.

Install QAD Production Scheduler

- Install QAD Production Scheduler from media.
- Build the production QAD Production Scheduler database.
- Create a QAD Production Scheduler database set.
- Generate QAD Production Scheduler server and client scripts.
- Compile the QAD Production Scheduler application.
- Set up a WebSpeed broker.
- Start and test the QAD Production Scheduler interface.

Convert QAD Production Scheduler

- Install new QAD Production Scheduler media.
- Update the production QAD Production Scheduler database.
- Edit the QAD Production Scheduler database set.
- Regenerate QAD Production Scheduler server and client scripts.
- Compile the updated QAD Production Scheduler application.
- Reconfigure the WebSpeed broker.
- Start and test the QAD Production Scheduler interface.

Configure QXI

- Add receivers for each QAD ERP instance or domain.

- Install the QAD Production Scheduler custom QDoc (QXI 1.2.1.5 or earlier only).
- Configure a QXI connection pool for each QAD ERP instance or domain.

Note The QXI steps are summarized in this guide. Refer to the QXtend documentation for details.

System Requirements

Operating Systems

- Linux
- HP-UX
- Sun Solaris (SPARC)
- Compaq UNIX (Tru64)
- IBM AIX
- Windows Server 2000

Software

Note For Windows implementations, both the QXtend Inbound Server and QAD ERP must be compiled on a character client.

- Progress RDBMS version 9.1D Service Pack 09 +
 - Note** Progress cannot be installed to a directory path that contains spaces. Scripts intended to start and stop QAD ERP and QAD Production Scheduler databases will fail.
- Progress WebSpeed version 3.1E + (If you are deploying the Web server on a separate host, you must download and install the WebSpeed Messenger. See “Multi-Tier Installations” on page 74.)
- QAD ERP application, versions eB, eB2, eB2.1, QAD 2007, QAD 2007.1, QAD 2008 - Standard and Enterprise Editions
- QAD Desktop (optional) version 2.6.1 +
- QAD QXtend Inbound, version 1.2 + (1.2.1.6 recommended)
- Apache Web server version 2 +

Important IBM AIX users require Apache version 2.2.4 or higher.

- Tomcat 5.5 +

Note Tomcat cannot be installed to a directory path that contains spaces. Scripts intended to start and stop the Tomcat service will fail.

Note Starting with Tomcat 5.5, the Admin tool must be downloaded and installed separately. To access the Admin tool, open a browser and enter `http://<HostIPAddress>:8080/admin`.

- Java SDK 1.4.2_07 +

Java must be installed on any machine where Apache Web server, Tomcat, or QXtend Inbound are deployed.

Note Apache 2.2 and later requires Java 1.5.0 +.

- Java 1.4.2_07 + plug-in on each client machine

The best results for the plug-in require that you install the file manually on each QAD Production Scheduler client. Download the Java 1.4.2_07 plug-in by selecting Download J2SE JRE from the following URL:

<https://java.sun.com/j2se/1.4.2/download.html>

- Windows only: Georgia Softworks Telnet Server, 6.50 +
- Internet Explorer, version 6.0 + on each client machine

Language Support

QAD Production Scheduler is double-byte enabled. This capability depends on QXtend Inbound, which defines supported code pages in the `QADInstall/qxtendserver/dtencode.dat` file.

Entries in `dtencode.dat` are in series of four. The first entry is the Progress code page, the second the equivalent JAVA code page, the third the HTML code page, and the fourth the XSLT code page. In most cases the second, third and fourth entries are identical because they use standard ISO code page names. Progress uses different names for some code pages.

Supported code page series in `dtencode.dat` are:

- iso8859-1,ISO8859_1,iso-8859-1,iso-8859-1
- shift-jis,SJIS,shift_jis,shift_jis

- gb2312,EUC-CN,gb2312,gb2312
- 1251,windows-1251,windows-1251,windows-1251
- 1250,windows-1250,windows-1250,windows-1250
- 1254,windows-1254,windows-1254,windows-1254
- 1257,windows-1257,windows-1257,windows-1257
- big-5,Big5,Big5,Big5
- ksc5601,EUC_KR,EUC-KR,EUC-KR
- IBM850,CP850,iso-8859-1,iso-8859-1

Other code page sets may function when added to the `dtencode.dat` file, but they have not been tested for the current version of QXtend Inbound and may return unexpected errors.

For additional information on supported code pages, see the *Progress Internationalization Guide*.

Prerequisites

The following installation instructions assume the requirements in Table 1.1 are met.

Table 1.1
Installation
Prerequisites

Requirement	Additional References
Progress is installed.	Installation guide for your version of the QAD ERP application
QAD ERP application is installed and compiled.	Installation guide for your version of the QAD ERP application
Tomcat, Apache, and Java are installed.	
Environment variables for JAVA_HOME, TOMCAT_HOME, and CATALINA_HOME are set.	<i>Technical Reference: QXtend</i>
Progress is added to the system PATH variable	
The <code>tomcat-users.xml</code> file is modified.	<i>Technical Reference: QXtend</i>

Requirement	Additional References
Windows only: Georgia Softworks is installed and configured.	<i>Technical Reference: QXtend</i>
QXtend Inbound is installed and compiled.	<i>Technical Reference: QXtend</i>
All QAD ERP databases are shut down.	

AIX Requirements

If you are installing QPS on an AIX system, additional Progress libraries are required. Do one of the following:

- Copy the following libraries from `$DLC/lib/*psc.so` to `/usr/lib`:
 - `libxerces-psc.so`
 - `libicuuc-psc.so`
 - `libicudata-psc.so`
- Alternately, create symbolic links in `/usr/lib` pointing to each of the three files listed above in `$DLC/lib`.

After completing the changes, run `slibclean` at the command line to clear the memory cache on the operating system.

Installation Worksheet

Use the following table to enter specific values for your environment. This list will be useful during the installation, and will serve as a reference for support issues and for future upgrades.

Description	Your Value	Default or Example
Multi-tier QPS install (yes/no)		
QAD ERP host machine		corpli01.corp.com
QAD ERP version and service pack		eB2.1 or earlier
QAD ERP install directory (QADInstallDir)		/home/mfgsvr
QAD ERP client script path		/home/mfgsvr/client.Production

Table 1.2
Installation
Worksheet

Description	Your Value	Default or Example
QAD ERP main database service		mfgprod-srv
QAD ERP admin database service		admprod-srv
QAD ERP help database service		hlpprod-srv
QPS database service		qpsprod-srv
QPS install directory		/home/mfgsvr/qps
QPS install log		/instlog/qps.log
Progress version and service pack		9.1E, SP 4
Progress install directory (ProgressInstallDir)		/dlc91e
Tomcat host machine		corpli05.corp.com
Tomcat version		5.5.17
Tomcat install directory (TOMCAT_HOME)		/tomcat
Java version		1.5.0_06
Java install directory (JAVA_HOME)		/jdk1.5.0_06
QXtend Inbound (QXI) version		1.2.1.6
QXI install directory		/mfgsvr/qxtendserver
Apache host machine		corpli09.corp.com
Apache version		2.0
Apache install directory (ApacheHome)		/apache
Georgia Softworks directory		c:\GS_UTS
QPS Web install directory		ApacheHome/htdocs/ qps

Modify the Services File

QAD Production Scheduler must be implemented as a client/server application to take advantage of Web services. Therefore, create services entries for the relevant databases before starting the install. If you have already created services for your Production database set during your QAD ERP installation, the services entries for `mfgprod` and `admprod` must be the same as those for the same databases in the QAD Production Scheduler database set. In addition, all service names, host names, and port numbers must match in the `services` files of each client and each database server on the network.

The location of the UNIX `services` file on the server is typically the `/etc` directory. On Windows it is located in `c:\windows\system32\drivers\etc`. In some Windows systems it is located in `c:\winnt\system32\drivers\etc`.

Add the name of your QAD Production Scheduler database service to your `services` file. Limit the service name to 16 characters. You can use any unused port number. Table 1.3 shows example service names and port numbers for a full QAD ERP production database set with QAD Production Scheduler.

Service Name	Port #/Protocol	Comment
mfgprod-srv	5500/tcp	# Production Database
admprod-srv	5510/tcp	# Admin Database
hlpprod-srv	5520/tcp	# Help Database
qpsprod-srv	5530/tcp	# QPS Database

Table 1.3
Services File
Example

Important These service names are the defaults used by MFG/UTIL and must match those used later in “Set Up QAD Production Scheduler” on page 25. If you change them here, you will need to edit your database sets later on.

Multiple Database Set Scenarios

If you are planning to create multiple QAD Production Scheduler database sets—for example, Production and Training—you will need to add a service for the additional databases; for example, `qpstrain-srv`, `mfgtrain-srv`, and `admtrain-srv`. See “Configuring Additional Instances” on page 51.

Conversion Prerequisites

If you are upgrading an earlier release of QAD Production Scheduler, complete the following tasks before proceeding:

Note The default broker name is `qps`. If your broker has a different name—for example `qpsprod`—the file and broker names will reflect this.

- 1 Back up the installation log and `.ini` file:

```
/instlog/qps.log
/instlog/instqps.ini
```

- 2 Stop the background processes from `QPSInstallDir`:

```
./stopall.QPS
```

- 3 Stop the WebSpeed broker from `ProgressInstallDir/bin`:

```
./wtbman -i qps -stop
```

- 4 Stop the database server from `QPSInstallDir`:

```
./stop.QPS
```

Multiple QPS Version Prerequisites

If you are installing more than one QAD Production Scheduler instance on the Apache server, you must first create the target directories manually. For example, if you plan to run a training version and a production version using the same instance of Apache, on UNIX systems you would need to create the following directories:

```
ApacheHome/html/qpsprod
ApacheHome/html/qpstrain
```

For Windows, the equivalent directories would be:

```
ApacheHome\htdocs\qpsprod
ApacheHome\htdocs\qpstrain
```

▶ See “Configuring Additional Instances” on page 51.

Installing from Media

These steps must be completed for both new installs and conversions.

Mount the CD-ROM

This guide assumes you are receiving QAD Production Scheduler on a CD-ROM. For Windows, insert the CD-ROM into the appropriate drive. For UNIX installations, use the following instructions:

- 1 Log on as `mfg`.
- 2 Mount the CD-ROM. Example commands are listed in Table 1.4.

Hardware	Mount Command
Sun	<code>volcheck cdrom</code>
HP	<code>/etc/mount -F cdfs /dev/dsk/YourCDDevice /cdrom</code>
Digital	<code>mount -r -o noversion -t cdfs /dev/YourCDDevice /cdrom</code>
AIX	<code>smitty mountfs</code> Then select file system, directory, and file system type (<code>cdarfs</code>).
Linux	<code>mount /dev/hdb /mnt/cdrom</code> Where <code>/hdb</code> could be <code>hdc</code> or <code>hdd</code> among other possibilities.
All others	Refer to your operating system documentation or vendor for requirements to mount a CD-ROM. You may be able to type <code>man mount</code> to determine the correct command.

Table 1.4
UNIX CD Drive
Mount Commands

Install the QAD Production Scheduler Application

Complete this section to install the QAD Production Scheduler files on the server.

- 1 In UNIX, log on as user `mfg` under the group `qad`. For Windows, log in as administrator.
- 2 On the CD, change to the `install` directory:
`cd install`
- 3 Launch the database server installation script in that directory:
`./install.ksh`

For Windows, run `install.exe` from Run on the Start menu.

A welcome screen displays. Press Enter.

```
Welcome to QAD's QPS 2.3 installation.  
We are installing QAD's QPS 2.3 for Linux.  
Press <Enter> to view license agreement.
```

- 4 Accept the QAD Production Scheduler software license agreement. (You can press `q` to jump to the end of the agreement.)

```
Do you accept all the terms of the preceding License  
Agreement? If you choose no, the install will stop.
```

```
To install QAD's QPS 2.3,  
you must accept this agreement. (y/n)?  
Default is n  
->y
```

- 5 Accept the default or enter a different installation log file location. Other installs on this host (QAD ERP, QXtend, and others) look in the default location for installation information. If you enter a different location, make note of it for later installations.

```
Please enter the location where the log file should be  
written. Default is /home/mfg/instlog  
->
```

On Windows systems, the default is `c:\instlog`.

- 6 On the initial installation, you are informed that `instqps.ini`, the data file to support the install, does not exist. Enter `y` to create it.
- 7 You are then asked to enter your Progress directory.

QAD Production Scheduler Component Installs

There are three QAD Production Scheduler components to be installed:

- The Progress code, typically installed under your QAD ERP directory
- The Web server components
- The QXtend components

The installation script asks if you want to install each separately to the current host, and then where to install that component.

- 8** You are asked if you want to install the Progress source code on this host:

```
Do you wish to install Schema for the QPS Repository and
Progress Source Files now?
```

If you want to install the Progress code to the current host, reply *y*.
The script verifies the location and version.

Answer *n* to skip to the Web server files install. If so, go to step 12.

- 9** You are then asked to enter the QAD ERP database server directory, also referred to as the *QADInstallDir*. By default this is */home/mfgsvr* on UNIX systems, *c:\mfgsvr* on Windows.

- 10** On Windows only, enter or accept the folder name for the MFG/UTIL icons. The default is QAD Production Scheduler.

If the folder does not exist, you are asked to confirm its creation.

- 11** You are asked whether your version of QAD ERP supports multiple domains. The default is *y*. If your installation is earlier than eB2.1, enter *n*.

- 12** You are then asked if you want to install the Web server files to the current host.

```
Do you wish to install Web server files (html, css,
javascript, property, and jar) now?
```

Answer *y* to install the Web server files to the current host.

Answer *n* to skip to the QXtend files install. If so, go to step 15.

- 13** For the Web server files install, you are asked to enter the QAD Production Scheduler location in your Web server deployment. For UNIX, this is typically *ApacheHome/html/qps*; on Windows, *ApacheHome\htdocs\qps*.

Note For multiple QAD Production Scheduler instances against a single copy of Apache, see “Multiple QPS Version Prerequisites” on page 18.

```
Enter a subdirectory of your Web Server Document directory for
the web application files.
```

- 14 You are prompted to enter the URL for your WebSpeed server. This entry references a file not yet created, but that is required later in the installation. By default, the correct WebSpeed URL for Linux or UNIX will look like:

```
http://host.domain.com:port/cgi-bin/qps.sh
```

For Windows, it will look like:

```
http://host.domain.com:port/cgi-bin/cgiip.exe/WService=qps
```

Where:

host.domain.com is your fully qualified host name (for example, *linx05.qad.com*).

port is the port number used by Apache.

Note The port must only be specified for Apache if you are using a port other than the default, 80.

cgi-bin is the standard Apache location used by WebSpeed.

qps.sh is the UNIX script file used to connect the QAD Production Scheduler to WebSpeed.

Note If you are implementing multiple instances of QAD Production Scheduler on the current host—for example, *qpsprod* and *qpstrain*—the file name here must reflect that; for example, *qpsprod.sh*.

cgiip.exe/WService=qps is the Windows WebSpeed executable and broker used to connect the QAD Production Scheduler to WebSpeed. The string is case sensitive.

Note For multiple instances of QAD Production Scheduler on the current host—for example, *qpsprod* and *qpstrain*—the *WService* value here must reflect that; for example, *qpsprod*.

- 15 You are then asked if you want to install the custom QAD Production Scheduler QDocs to the current host. This is only required for QXtend Inbound versions prior to 1.2.1.6.

```
Do you wish to install Custom QDocs for QXtend Inbound now?
```

Answer *y* to install the QDocs to the current host.

Answer *n* to skip the QXtend files install on this host and go ahead with the file extraction of the components you selected.

- 16** For the QDoc install, you are asked to enter a temporary location for the files. The default for UNIX is `/home/qps/qdocs`; Windows is `c:\qps\qdocs`. In this guide, this directory is referred to as *TMPDOC*. If the directory does not exist, you are asked to confirm its creation.

▶ See “Loading QAD Production Scheduler QDoc Schemas” on page 64.

- 17** A confirmation screen displays. Review the directories displayed and press Enter to start the file extractions for this host.
- 18** The file extraction process runs. The log is written to:

```
/instlog/qps.log
```

Running the New Install Workflow

These steps must be completed only for new installs.

For conversions, skip ahead to “Running the Update Workflow” on page 37.

QAD Production Scheduler is configured in your QAD ERP environment using a QPS/UTIL workflow. The QAD Production Scheduler Guided Setup program provides a workflow of the operations involved in setting up your QAD Production Scheduler environment. You can use this program to access the different configuration utilities in QPS/UTIL with proper default information and in proper sequence.

Note QPS/UTIL is a copy of MFG/UTIL that has been tailored for QAD Production Scheduler. QPS/UTIL is the program name and the name that appears on icons to launch the application under Windows. However, the name in the product itself is MFG/UTIL. This guide refers to the QAD Production Scheduler administrative application universally as QPS/UTIL. Do not use the MFG/UTIL located in *QADInstallDir*.

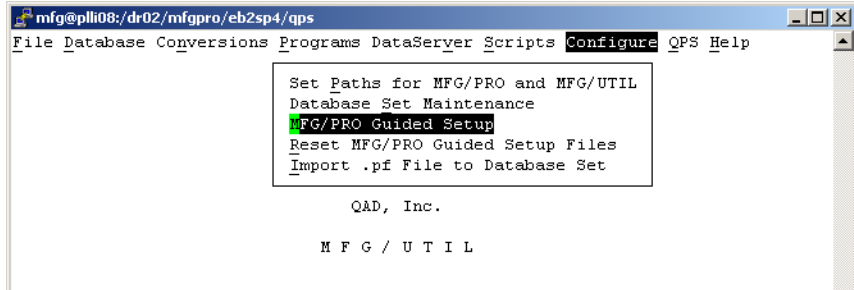
- 1** On UNIX systems, launch QPS/UTIL from *QPSInstallDir* using the following command:

```
./qpsutil
```

For Windows, launch QPS/UTIL from the QAD Production Scheduler icon on the Start menu.

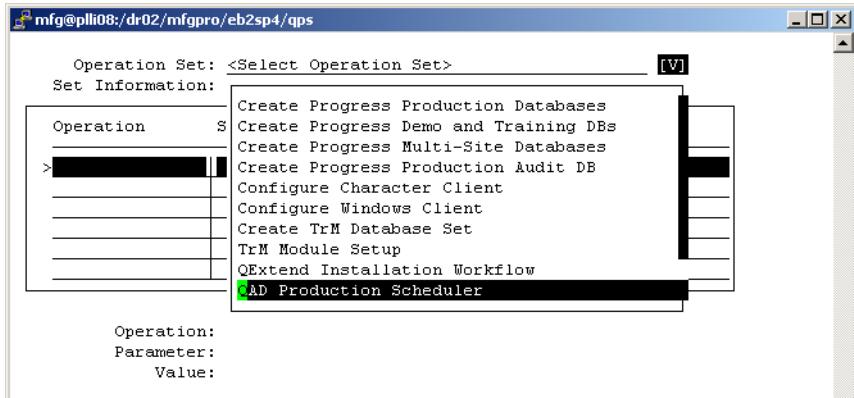
- 2** Choose QAD ERP Guided Setup from the Configure menu.

Fig. 1.3
QPS/UTIL
Configure Menu



- 3 Select QAD Production Scheduler in the Operation Set drop-down list box and press Enter.

Fig. 1.4
Selecting an
Operation Set



- 4 The QAD Production Scheduler Guided Setup displays. Review the steps by scrolling up or down in the window.

Operation sets are groups of installation activities. Below the Operation frame, the operation, the key variable required, and default value for that variable display.

On completion of an operation, the status changes to Done. If errors occur or if you cancel processing prior to completing a step, the status is Error. If you stop the workflow and an Error or Cancelled status is written to a step, you must open the workflow, *QPSInstallDir/wk0650.ini*, and edit the status to read Not Run. Save the workflow and rerun it in QPS/UTIL. The Not Run step is the first step run when you restart the operation set.

- 5 Select Run Set and press Enter.

Set Up QAD Production Scheduler

- 6 The QPS Setup screen displays. Enter the correct values based on the screen example and the field descriptions that follow.

The screenshot shows a window titled "QPS Setup" with the following fields and values:

```

Log Directory:
  /var/log/qps
Webapp Name:
  qps
QPS PROGRESS Source Directory:
  /qad/mfgpro/eb21sp5_pro/qps
Web Server Home (Doc Root):
  /var/www/html/qps
WebServer Host Name:
  vml102
QXtend Inbound Webapp Directory:
  /ul/qad/qxi/tomcat-5.5.20/webapps/qxtendserver
  
```

At the bottom of the window, there are navigation buttons: "< OK >" and "<Cancel>".

Fig. 1.5
QPS Setup Screen

Log Directory. Enter the path to the directory where processing logs will be stored.

Important You must specify a directory to which user ID `mfg` has write access.

Webapp Name. By default, this is `qps`. For a single instance of QAD Production Scheduler, accept this default. If you are implementing multiple instances of QAD Production Scheduler on this host as described in “Multiple Database Set Scenarios” on page 17, enter the first instance name here.

QPS Progress Source Directory. This is the `QPSInstallDir` from step 8 of the media installation. This is `QADInstallDir/qps`.

Web Server Home (Doc Root). The location where QAD Production Scheduler resides on the Web server as specified during the media installation in step 12. Typically this would be `ApacheHome/html/qps`. On Windows, this may be `ApacheHome\htdocs\qps`.

Note The `mfg` user must have write permissions on `ApacheHome/html`.

Web Server Host Name. The host name of the machine where WebSpeed will run. This should include a domain designation; for example, *host.domain.com*. For multi-tier installs, this is the host where Tomcat and the QXI webapps are located.

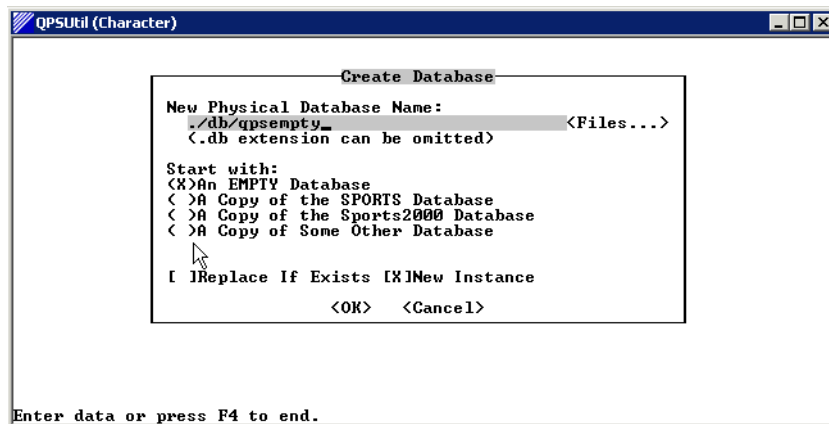
QXtend Inbound Webapp Directory. The QXtend webapp location; for example, *TOMCAT_HOME/webapps/qxtendserver*. For multi-tier installs, this is the path where the Tomcat QXI webapp is located.

Create qpsempy

In the next few steps, you create the QAD Production Scheduler empty database, `qpsempy`. You then connect to the new database, load the schema, and truncate the database before-image (BI) file.

- 1 The Create Database screen displays. Accept the defaults and choose OK.

Fig. 1.6
Create Database
qpsempy



- 2 The database is created and you are prompted to connect to `qpsempy`.

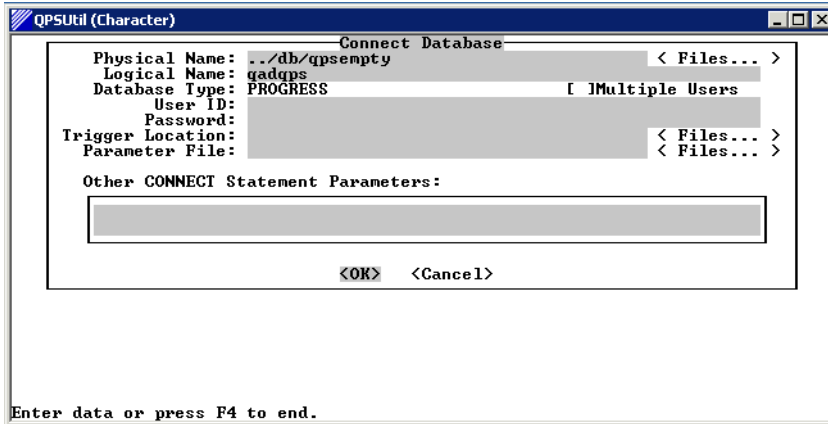


Fig. 1.7
Connecting to qpsempty

- 3 Accept all defaults and choose OK to connect.
- 4 A sample schema file (.df) is displayed in the Enter a Data Definition to Load screen. The file, qps.df, is located in QADInstallDir/db. Choose OK to load the schema.



Fig. 1.8
Loading Data Definitions for qpsempty

- 5 A log window displays the schema load progress. When the load is complete, choose Close to exit the log and continue.

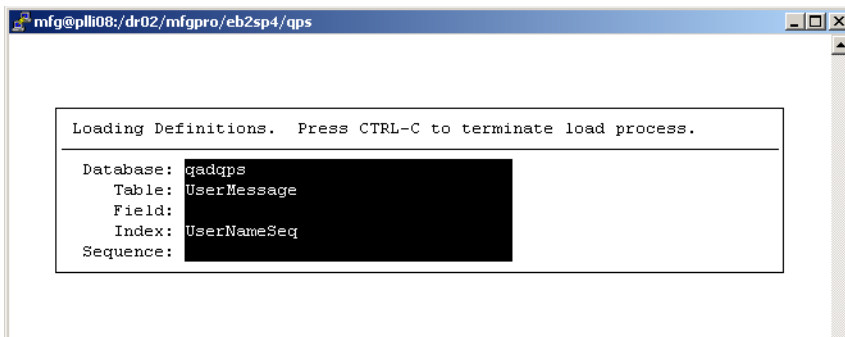
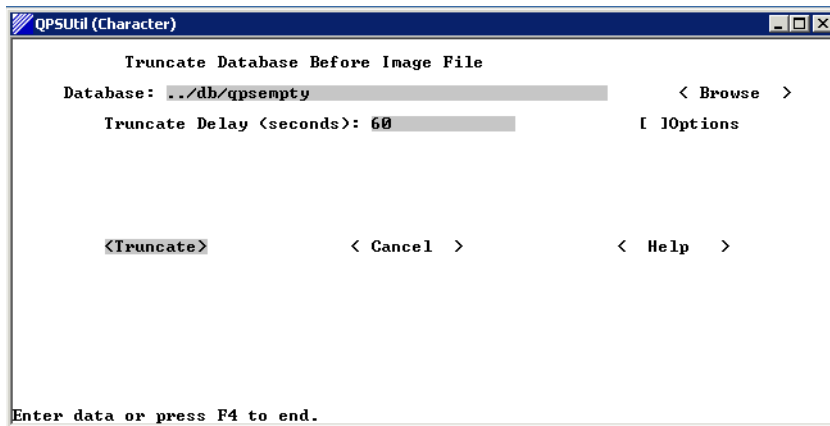


Fig. 1.9
Data Definition Load Progress Screen

- 6 You are then prompted to truncate the database. Choose Truncate.

Fig. 1.10
Truncating qpsprod



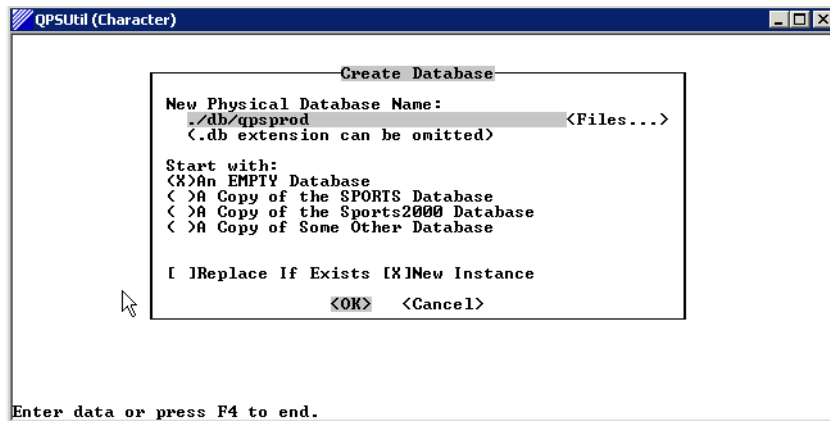
- 7 A log window displays. When the truncate is complete, tab out of the log to the Close button. Choose Close to exit the log and continue with the workflow.

Create qpsprod

You now create the qpsprod database.

- 1 The Create Database screen displays. Accept the defaults and choose OK.

Fig. 1.11
Creating qpsprod



- The database is created and you are prompted to connect to `qpsprod`.

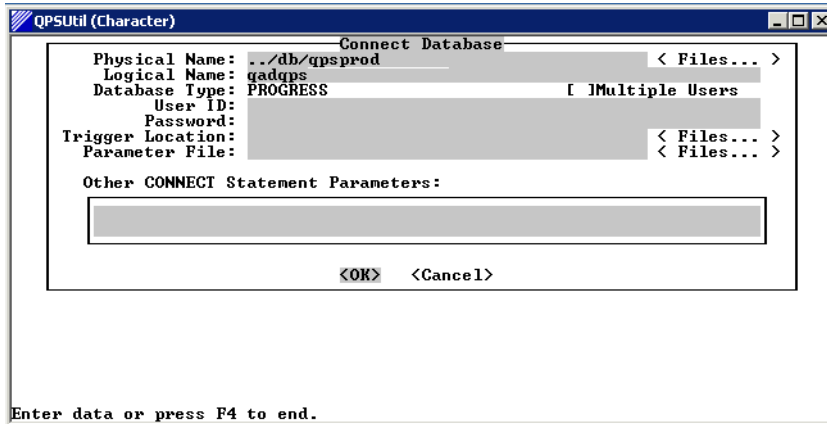


Fig. 1.12
Connecting to
qpsprod

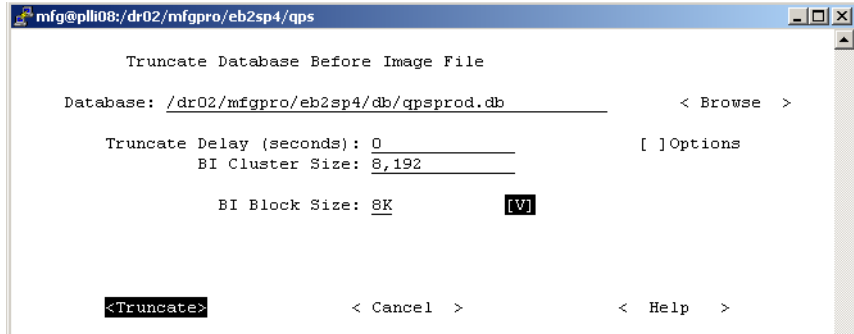
- Accept all defaults and choose OK to connect.
- A sample schema file (`.df`) is displayed in the Enter a Data Definition to Load screen. The file, `qps.df`, is located in `QADInstallDir/db`. Choose OK to load the schema.



Fig. 1.13
Loading Data
Definitions for
qpsprod

- A log window displays the schema load progress. When the load is complete, choose Close to exit the log and continue.
- You are then prompted to truncate the database. Choose Truncate.

Fig. 1.14
Truncating qpsprod



- 7 A log window displays. When the truncate is complete, tab out of the log to the Close button. Choose Close to exit the log and continue with the workflow.

Set Up QAD Production Scheduler Database Set

While you are working in Database Set Maintenance, you will need to validate both the QPS and QPS Compile database sets. Make sure:

- The databases in both sets have the full path to the databases defined.
- Each database in the QPS set must be defined as client/server and have valid host and services entries.
- Each database in the QPS Compile set may be defined as local if the databases are local; otherwise define them as client/server; the QPS Compile set should contain empty databases only.

Important The settings for the databases defined for the QPS database set must match the settings defined for the Production database set. Even more importantly, since both QAD ERP and QAD Production Scheduler must connect to the databases at the same time, the Production databases must all be defined as client/server. If your Production databases are defined as local, you will need to change this for each database, add settings for Host and Service for each database, and regenerate the Production database set scripts.

The QPS database set is different than standard QAD ERP database sets, primarily because it generates different scripts. One important difference is that the client script it creates is for a “headless” user; it does not launch an interface you can interact with. Therefore, you cannot simply add your

other side databases such as help, AIM, or custom databases to the QPS database and then attempt to launch a client session that connects to all of these databases.

Note The QPS database set scripts start and stop the main and admin QAD ERP databases by default. This will impact all users. To avoid this, you can edit the QAD ERP databases out of the database set start and stop scripts. If you do this, you must make sure the QAD ERP database servers are running prior to starting the QPS database server.

- 1 Database Set Maintenance displays. The QPS database set is selected.

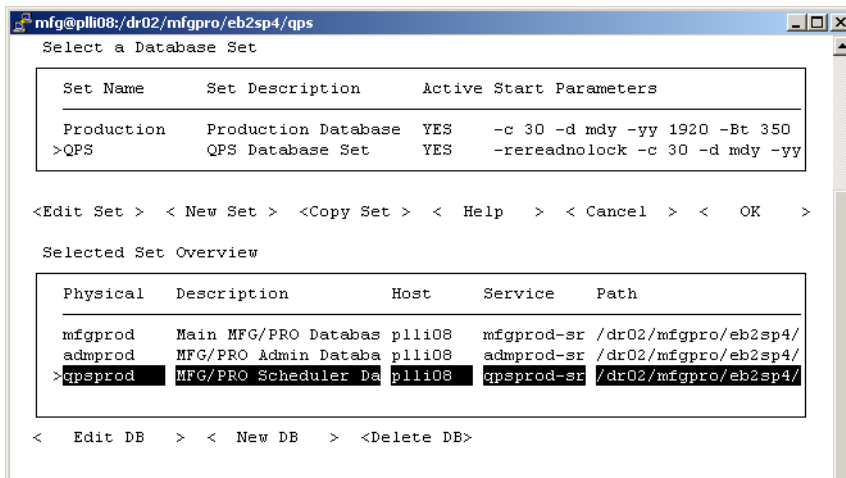
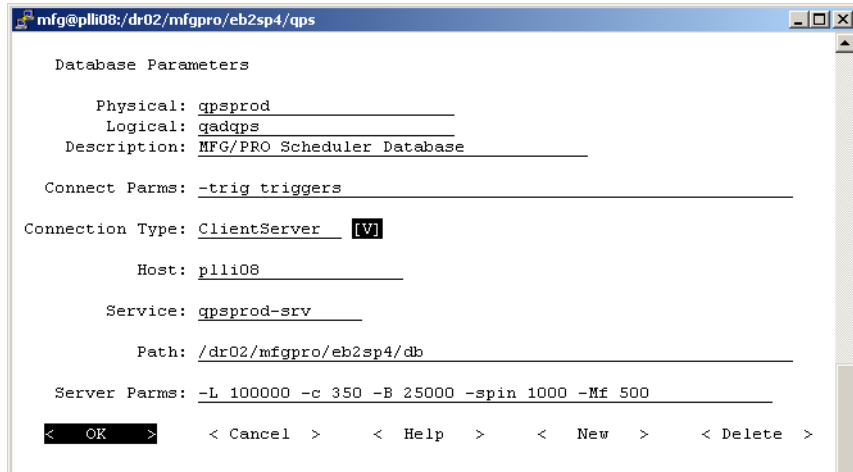


Fig. 1.15
QPS Database Set

- 2 Tab to Selected Set Overview, and select mfgprod. Choose Edit DB.
- 3 In the Edit Database screen, enter the correct Host. Verify that the Service defined for the database is the same as that defined in “Modify the Services File” on page 16 and matches the service assigned in the Production database set. Choose OK.
- 4 Repeat steps 2 and 3 for admprod and qpsprod.

Note QAD Production Scheduler does not use the QAD ERP help database.

Fig. 1.16
qpsprod Default
Connection
Information



- 5 Make any necessary modifications and choose OK to return to Database Set Maintenance.

Important If you are running QAD Production Scheduler with QAD ERP on Oracle, review “Running QAD Production Scheduler with QAD ERP Oracle” on page 68 regarding changes required in Database Set Maintenance.

- 6 Tab back to the database set names and select QPS Compile. Repeat steps 2 through 4 for the QPS Compile set, except for the following differences:
 - Unless the databases are defined on a separate host, the Connection Type is Local and hosts and services do not need to be defined.
 - The admin database is admempty instead of admprod.
- 7 Choose OK to continue the workflow.

Create Windows Icons or UNIX Scripts

Once QAD Production Scheduler is fully configured, you can start any of the QAD Production Scheduler processes individually—auto firm, auto synchronization, manual synchronization, and auto export—or you can

start them all at the same time. The next set of steps creates a total of ten scripts: four to start the individual processes, four to stop them, one to start all the processes at once, and one to stop them all.

- 1 The Create Database Set Icons (Windows) or Create UNIX Server Scripts (UNIX) screen displays. Arrow to the QPS set and press spacebar to select it. Choose OK to create the icons or scripts.

Note The QPS Compile set does not require scripts.

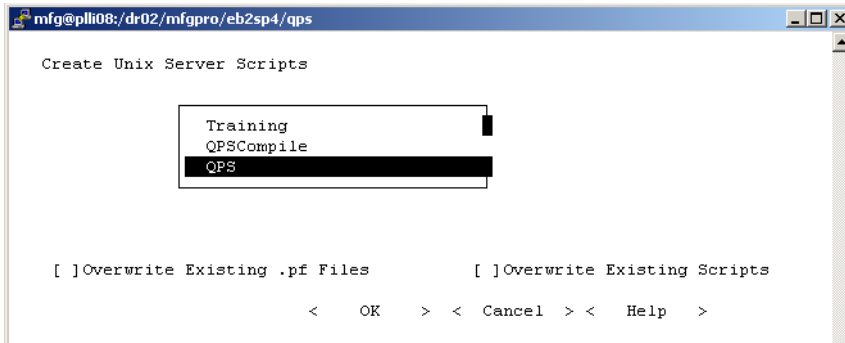


Fig. 1.17
Creating Set Icons

- 2 On Windows systems, you are prompted to create or select an icon folder. The default is QPS 2.0. Choose OK to continue.
You are prompted to confirm before proceeding.
- 3 A log window displays as the scripts and icons (for Windows) are generated. Table 1.5 describes the files generated.

Important QAD Production Scheduler requires that the QAD ERP main and admin databases be running. Therefore, the `start.<DBSetName>` and `stop.<DBSetName>` scripts include these databases. If QAD ERP databases are started separately in your production environment, edit these scripts after they have been created to remove the QAD ERP databases.

Table 1.5
QPS Generated
Database Set
Scripts

File name (UNIX version first, then Windows)	Description
envvars.ini	Updates to QPS paths, host name, and so on.
start.<DBSetName> start<DBSetName>.bat	For example, start.QPS. The script used to start the database servers. Since the QAD ERP databases should already be started when you run this script, you will see messages that those databases are already running. This script ultimately launches the qpsprod database server.
stop.<DBSetName> stop<DBSetName>.bat	For example, stop.QPS. The script used to stop the QPS database servers. Since this script shuts down mfgprod and admprod as well as qpsprod, it should be run only after the QAD ERP database servers have been shut down using stop.Production or the equivalent. Alternately, you could edit this script to remove mfgprod and admprod.
QPSclient.<DBSetName> QPSclient<DBSetName> .bat	For example, QPSclient.QPS. The QPS telnet client launch script. This script is used by the telnet connections in QXtend Inbound; it does not launch an interactive session.
<DBSetName>.pf <DBSetName>.pf	For example, QPS.pf. This contains the three databases in the QPS database set and their connection parameters. It is called by QPSclient.<DBSetName>.
startall.<DBSetName> startall<DBSetName> .bat	For example, startall.QPS. This script launches all four QPS background processes. Each of the background processes also has a separate script. A matching stopall.QPS script shuts down all four processes.
startprocautofirm. <DBSetName> startprocautofirm <DBSetName>.bat	This script launches the automatic firming process in the background; stopprocautofirm.<DBSetName> stops the process.
startprocautosynch. <DBSetName> startprocautosynch <DBSetName>.bat	This script launches the automatic synchronization process in the background; stopprocautosynch.<DBSetName> stops the process.
startprocmanexport. <DBSetName> startprocmanexport <DBSetName>.bat	This script launches the manual export process in the background; stopprocmanexport.<DBSetName> stops the process.
startprocmansynch. <DBSetName> startprocmansynch <DBSetName>.bat	This script launches the manual synchronization process in the background; stopprocmansynch.<DBSetName> stops the process.

4 Choose Close to continue.

Important On UNIX and Linux systems, if the QPS broker is named something other than `qps`, you must copy `qps.sh` to `ApacheHome/cgi-bin/<qpsbrokername>.sh`. You must then edit the file so that the `-i` argument is `<qpsbrokername>`.

Important If you are running QAD Production Scheduler with QAD ERP on Oracle, review the steps under “Running QAD Production Scheduler with QAD ERP Oracle” on page 68 regarding changes required in the background process scripts.

Generate Sample QAD Production Scheduler Broker File

The interface between WebSpeed and your client Internet Explorer browser is managed through the `Progress ubroker.properties` file. This file identifies the Progress application and name servers that QAD Production Scheduler needs up and running in order to connect as a WebSpeed broker. The next steps use the values created during the install thus far to create a WebSpeed broker definition that is pasted into the `ubroker.properties` file in a separate step.

- 1 You now generate a sample `ubroker.properties` file entry. All defaults are correct, as shown.
- 2 Choose Generate to continue. This may take a moment.

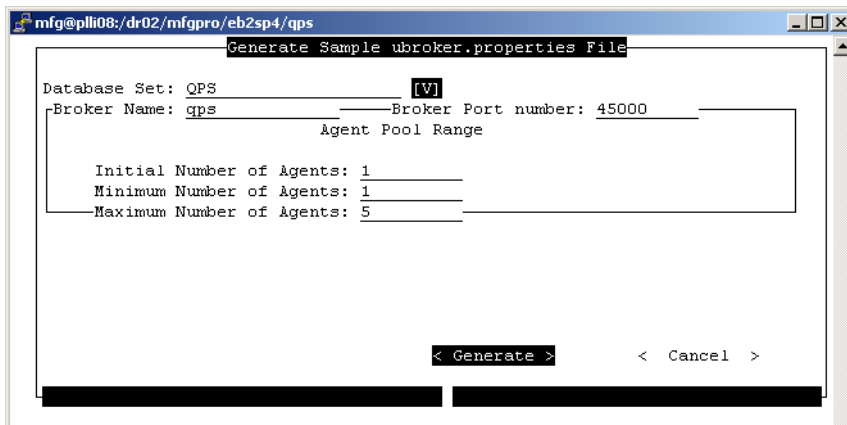


Fig. 1.18
Generating a
Sample ubroker
Broker Definition

- 3 A message appears informing you that a sample file named `gadps-ubroker.properties` has been written to your `QPSInstallDir`. Choose OK.

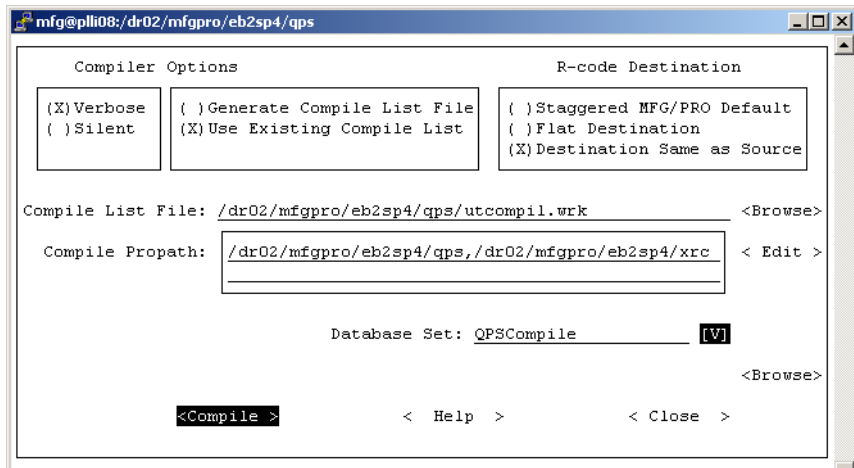
If you are implementing more than one QAD Production Scheduler instance on this host, repeat the workflow for the next QPS Webapp name, starting with “Running the New Install Workflow” on page 23.

Compile QAD Production Scheduler

The next steps result in compiled application code for QAD Production Scheduler under `QPSInstallDir`.

- 1 The Compiler Options screen displays. The Compile List File location and the other information default from your system. Accept all defaults. Choose Compile.

Fig. 1.19
QAD Production
Scheduler Compiler
Options



- 2 A summary screen displays. Choose Continue to start the compile.
- 3 A log window displays the compile progress.
- 4 On completion of the compile, check for errors. If there are no errors, choose Close in the log window. If you encounter errors, these must be resolved before you continue.
- 5 The workflow completion screen displays. Choose Close to end the workflow.

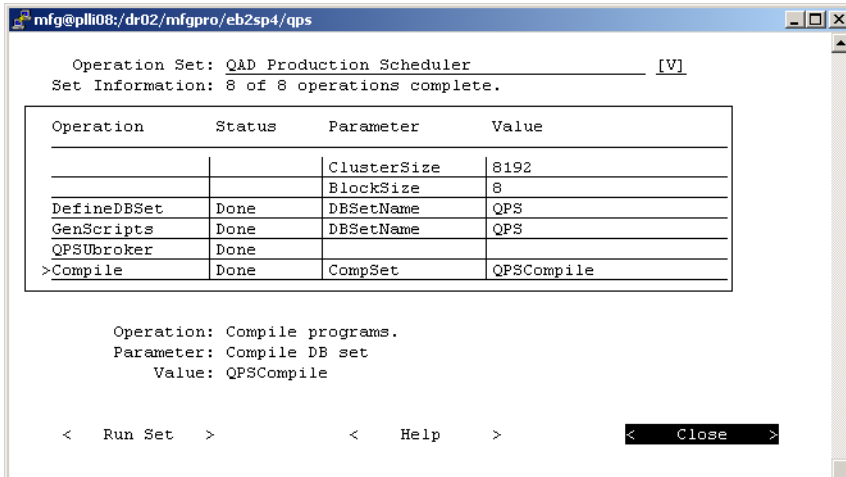


Fig. 1.20
Workflow
Completion

For new installs, skip ahead to “Configuring WebSpeed” on page 49.

Running the Update Workflow

These steps must be completed only for conversions.

The QAD Production Scheduler conversion is performed using a QPS/UTIL workflow. You can use this program to access the different configuration utilities in QPS/UTIL with proper default information and in proper sequence.

Note QPS/UTIL is the program name and the name that appears on icons to launch the application under Windows. However, the name in the product itself is MFG/UTIL. This guide refers to the QAD Production Scheduler administrative application universally as QPS/UTIL.

- 1 On UNIX systems, launch QPS/UTIL from *QPSInstallDir* using the following command:

```
./qpsutil
```

For Windows, launch QPS/UTIL from the QAD Production Scheduler icon on the Start menu.

Important Do not launch the version of QPS/UTIL that resides in *QADInstallDir*. This version does not include the functions required for the QAD Production Scheduler configuration.

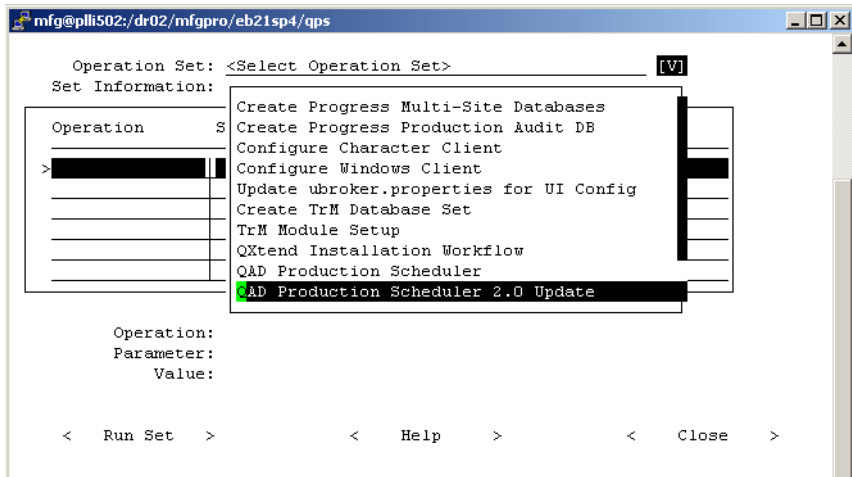
- 2 Choose QAD ERP Guided Setup from the Configure menu.

Fig. 1.21
QPS/UTIL
Configure Menu



- 3 Select QAD Production Scheduler 2.0 Update from the Operation Set drop-down list box.

Fig. 1.22
Selecting the
Update Operation
Set



- 4 The QAD Production Scheduler Guided Setup displays. Review the steps by scrolling up or down in the window.

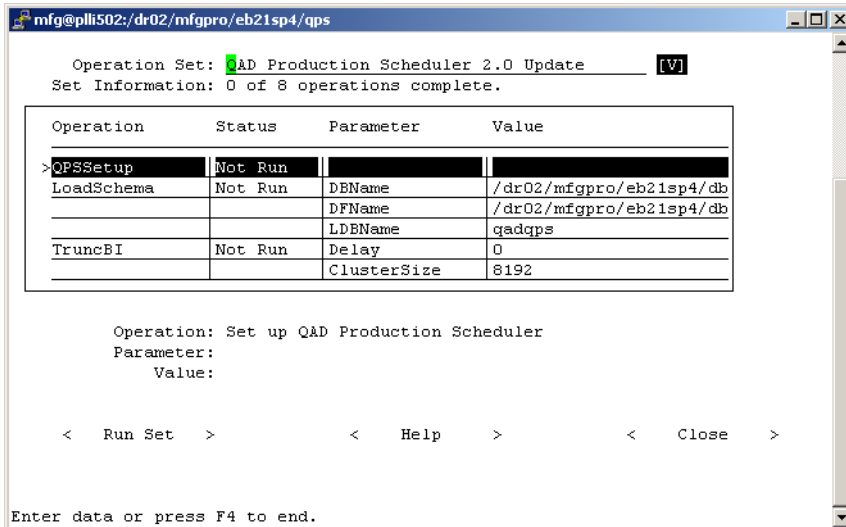


Fig. 1.23
QAD Production
Scheduler
Workflow

Operation sets are groups of installation activities. Below the Operation frame, the operation, the key variable required, and default value for that variable display.

On completion of an operation, the status changes to Done. If errors occur or if you cancel processing prior to completing a step, the status is Error. If you stop the workflow and an Error or Cancelled status is written to a step, you must open the workflow, *QPSInstallDir/wk0660.ini*, and edit the status to read Not Run. Save the workflow and rerun it in QPS/UTIL. The Not Run step is the first step run when you restart the operation set.

- 5 Select Run Set and press Enter.

Set Up QAD Production Scheduler

- 6 The QPS Setup screen displays. Enter the correct values based on the screen example and the field descriptions that follow.

Fig. 1.24
QPS Setup Screen

The screenshot shows a terminal window titled "mfg@plli502:/dr02/mfgpro/eb21sp4/qps" with a sub-window titled "QPS Setup". The setup screen contains the following fields and values:

```

Webapp Name:
  qps
QPS PROGRESS Source Directory:
  /dr02/mfgpro/eb21sp4/qps
Web Server Home (Doc Root):
  /var/www/html/qps
WebServer Host Name:
  plli502.qad.com
QXtend Inbound Webapp Directory:
  /dr01/apache-tomcat-5.5.15/webapps/qxtendserver
  
```

At the bottom of the form are buttons for "< OK >" and "<Cancel>". Below the form, the text "Enter data or press F4 to end." is displayed.

Webapp Name. By default, this is `qps`. If you have multiple instances of QAD Production Scheduler on this host as described in “Multiple Database Set Scenarios” on page 17, enter the first instance name here.

QPS Progress Source Directory. This is the `QPSInstallDir` from step 8 of the media installation. This is `QADInstallDir/qps`.

Web Server Home (Doc Root). The location where QAD Production Scheduler resides on the Web server as specified during the media installation in step 12 on page 21. Typically this would be `ApacheHome/html`. On Windows, this may be `ApacheHome\htdocs`.

Web Server Host Name. The host name of the machine where WebSpeed will run. This should include a domain designation; for example, `host.domain.com`.

QXtend Inbound Webapp Directory. The QXtend webapp location; for example, `TOMCAT_HOME/webapps/qxtendserver`. For multi-tier installs, this is the path where the Tomcat QXI webapp is located.

Update qpsprod

In the next few steps, you connect to the existing QAD Production Scheduler database, `qpsprod`, load new schema, and truncate the database.

- 1 You are prompted to connect to the existing `qpsprod` database.

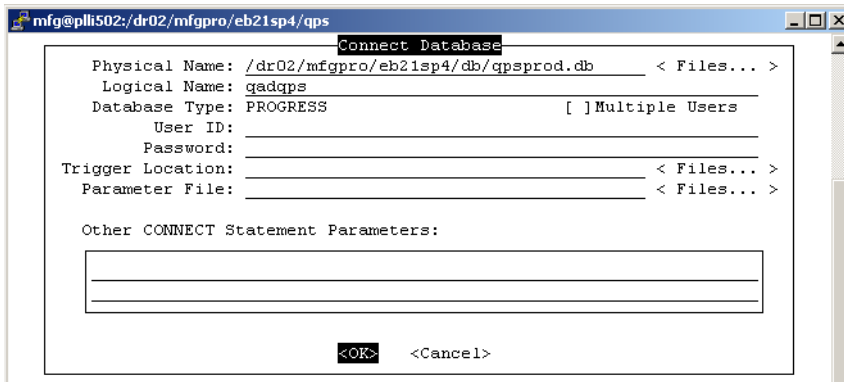


Fig. 1.25
Connecting to
qpsprod

- 2 Accept all defaults and choose OK to connect.
- 3 A sample schema file (`.df`) is displayed in the Enter a Data Definition to Load screen. Choose the delta data definitions file that matches your starting QAD Production Scheduler version. For example, if you are upgrading from version 1.0.4, you would select `delta-1.0.4.df`. The delta `.df` files are located in `QADInstallDir/db`. Choose OK to load the schema.

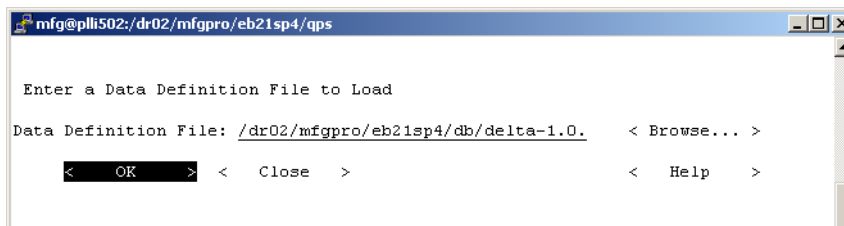
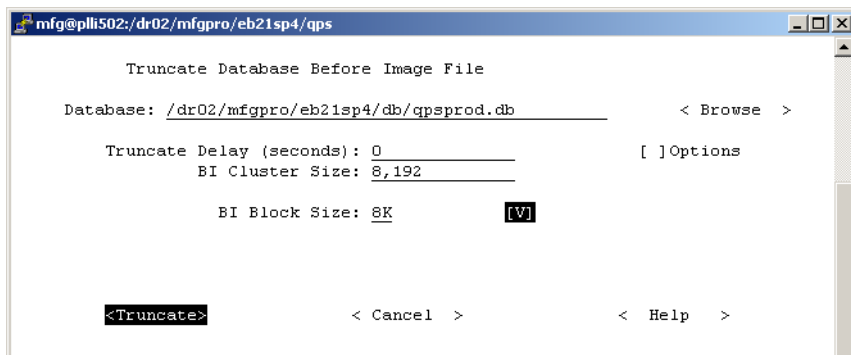


Fig. 1.26
Loading Data
Definitions for
qpsprod

- 4 A log window displays the schema load progress. When the load is complete, choose Close to exit the log and continue.
- 5 You are then prompted to truncate the database. Choose Truncate.

Fig. 1.27
Truncating qpsprod

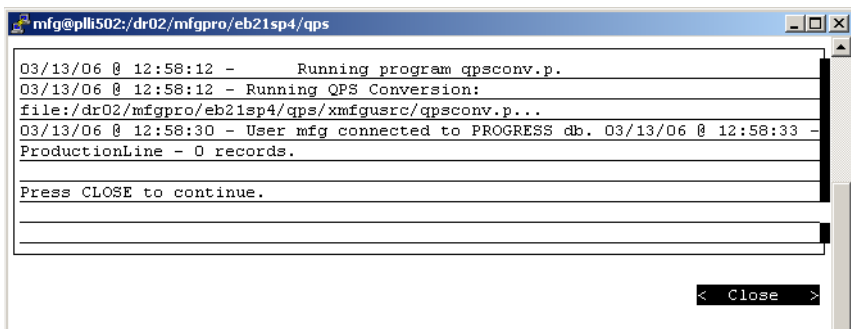


- 6 A log window displays. When the truncate is complete, tab out of the log to the Close button. Choose Close to exit the log.

Convert the Database

- 1 You are prompted to connect to `qpsprod`.
- 2 Accept all defaults and choose OK to connect.
- 3 A confirmation log displays. Choose Close to continue.
- 4 The database conversion routines are run.
- 5 Before choosing Close, complete the following set of steps outside the workflow.

Fig. 1.28
Connection File
Create Confirmed



Update Schema

Important Instructions in this section apply only to users who are upgrading from QAD Production Scheduler version 2.0.4 or earlier.

For all QPS databases including and prior to version 2.0.4, you must create new `qpsprod` and `qpsempty` databases containing the QPS database structure file. (You still needed to complete the load of the delta `.df` under “Update `qpsprod`” on page 41.)

The QPS structure file is `qpsprod.st`. This structure file defines the recommended schema areas for the QPS database.

Modify the structure file prior to these steps if required. For example, specify the directories where QPS schema areas are stored. If you make no changes, then the schema areas will be created in the `/db` directory below the directory where the Progress database has been created. See the Progress documentation for additional information.

Update `qpsprod`

Use the following instructions to create the new `qpsprod` database.

- 1 Dump the data from the `qpsprod` database using Progress tools.
- 2 Back up the QPS database `qpsprod`.

```
$DLC/bin/procopy qpsprod qpsprod-bu
```
- 3 Delete `qpsprod`.

```
prodel qpsprod
```
- 4 Create a new Progress database using the `prostrct` utility. The name of the database should be the same as the original database.

```
$DLC/bin/prostrct create qpsprod qpsprod.st
```
- 5 To create the Progress metaschema information in the new database, create an empty Progress database called `emptydb`.

```
$DLC/bin/prodb emptydb
```

You are asked whether to create a copy of the sports, system, or empty Progress database. Choose empty.

- 6 Copy the new empty database into your `qpsprod` database.

```
$DLC/bin/procopy emptydb qpsprod
```

- 7 Load the new `qps.dft` into the `qpsprod` database.

- 8 Load the `.d` data from step 1 into `qpsprod`.

Update `qpsemt`

Use the following instructions to create the new `qpsemt` database.

- 1 Back up the QPS database `qpsemt`.

```
$DLC/bin/procopy qpsemt qpsemt-bu
```

- 2 Delete `qpsemt`.

```
prodel qpsemt
```

- 3 Create a new Progress database using the `prostrct` utility. The name of the database should be the same as the original database.

```
$DLC/bin/prostrct create qpsemt qpsprod.st
```

- 4 Copy the new empty database `emptydb`—created in step 5 on page 43—into your `qpsemt` database.

```
$DLC/bin/procopy emptydb qpsemt
```

- 5 Load the new `qps.dft` into the `qpsemt` database.

Set Up QAD Production Scheduler Database Set

Your database sets should be correct without any additional modifications. The Update workflow opens the screen to allow you to review the sets, or modify anything that might need changing due to changes in your environment.

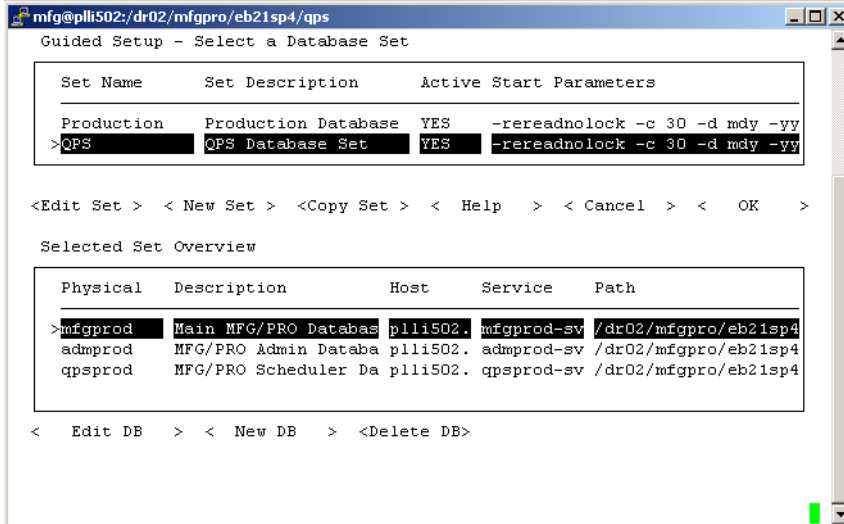


Fig. 1.29
QPS Database Set

Note The QPS database set scripts start and stop the main and admin QAD ERP databases by default. This will impact all users. To avoid this, you can edit the QAD ERP databases out of the database set start and stop scripts. If you do this, you must make sure the QAD ERP database servers are running prior to starting the QPS database server.

Important If you are running QAD Production Scheduler with QAD ERP on Oracle, review “Running QAD Production Scheduler with QAD ERP Oracle” on page 68 regarding changes required in Database Set Maintenance.

Create Windows Icons or UNIX Scripts

The next set of steps regenerates the ten QAD Production Scheduler scripts: four start the individual processes, four stop them, one starts all the processes at once, and one stops them all.

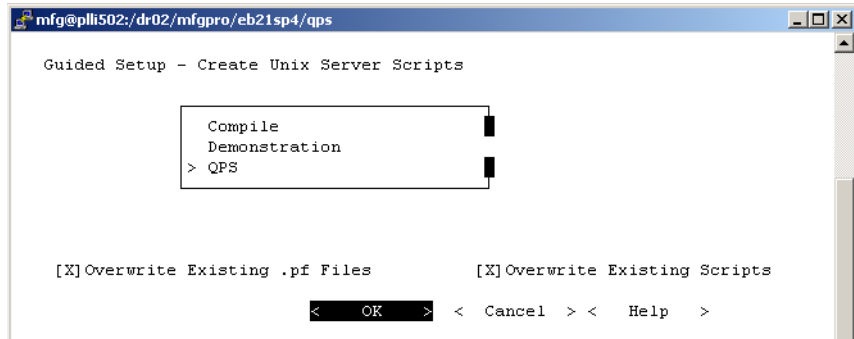
- 1 The Create Database Set Icons (Windows) or Create UNIX Server Scripts (UNIX) screen displays. Arrow to the QPS set and press spacebar to select it.

For conversions, overwrite the existing scripts and .pf files.

Choose OK to create the icons or scripts.

Note The QPS Compile set does not require scripts.

Fig. 1.30
Creating Set Icons



- 2 A log window displays as the scripts and icons (for Windows) are generated. Table 1.5 on page 34 describes the files generated. Choose Close to continue.

Important On UNIX and Linux systems, if the QPS broker is named something other than `qps`, you must copy `qps.sh` to `ApacheHome/cgi-bin/<qpsbrokername>.sh`. You must then edit the file so that the `-i` argument is `<qpsbrokername>`.

Important If you are running QAD Production Scheduler with QAD ERP on Oracle, review “Running QAD Production Scheduler with QAD ERP Oracle” on page 68 regarding changes required in the background process scripts.

Generate Sample QAD Production Scheduler Broker File

The `ubroker.properties` file must be updated. This step uses the values created during the install thus far to create an updated WebSpeed broker definition that is pasted into the `ubroker.properties` file in a separate step.

- 1 You now generate a sample `ubroker.properties` file entry. All defaults are correct, as shown.
- 2 Choose Generate to continue. This may take a moment.

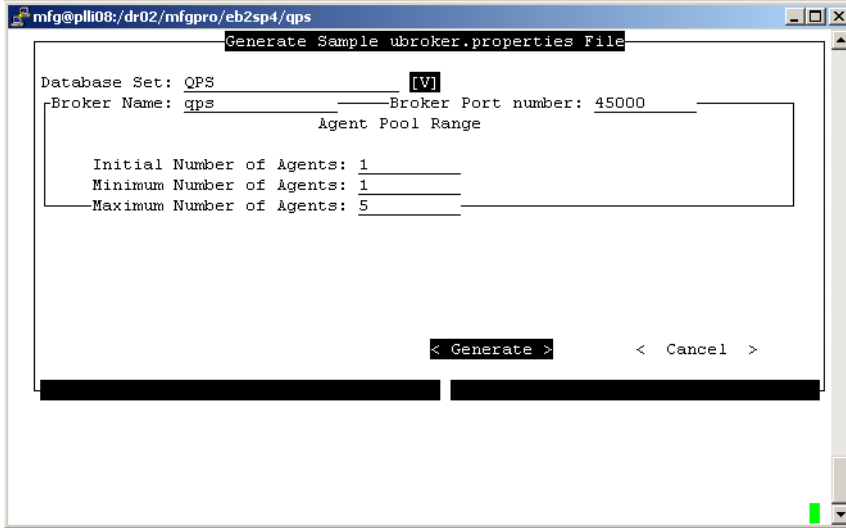


Fig. 1.31
Generating a
Sample ubroker
Broker Definition

- 3 A message appears informing you that a sample file named `gadps-ubroker.properties` has been written to your `QPSInstallDir`. Choose OK.

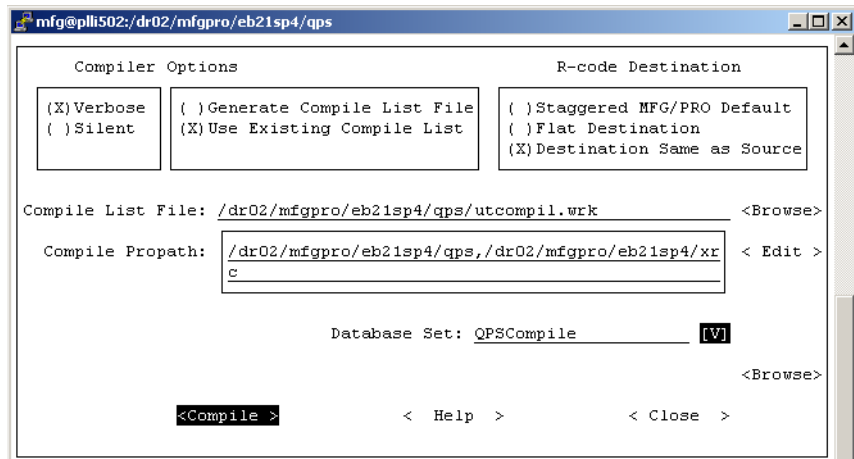
If you are updating more than one QAD Production Scheduler instance on this host, repeat the workflow for the next QPS Webapp name, starting with “Running the Update Workflow” on page 37.

Compile QAD Production Scheduler

The next steps compile the revised application code for QAD Production Scheduler under *QPSInstallDir*.

- 1 The Compiler Options screen displays. The Compile List File location and the other information default from your system. Accept all defaults. Choose Compile.

Fig. 1.32
QAD Production
Scheduler Compiler
Options



- 2 A summary screen displays. Choose Continue to start the compile. A log window displays the compile progress.
- 3 On completion of the compile, check for errors. If there are no errors, choose Close in the log window. If you encounter errors, these must be resolved before you continue.
- 4 The workflow completion screen displays. Choose Close to end the workflow.

Configuring WebSpeed

These steps must be completed for both new installs and conversions.

The first task is to verify the `qadps-ubroker.properties` file. Because different versions of Progress provide different files and directory structures, and produce different results; and because multi-tier QAD Production Scheduler implementations have additional requirements, you must open the `ubroker` sample file created at the end of the new install or conversion workflows and modify it.

The following output is a Windows example. The only difference for UNIX systems is the file names and paths.

```
[UBroker.WS.qps]
  appserviceNameList=qps
  brokerLogFile=c:\eb21sp4\qps\logs\qps.broker.log
  srvrLogFile=c:\eb21sp4\qps\logs\qps.server.log
  description=qps
  environment=qps
  controllingNameServer=NS1
  portNumber=45000
  PROPATH=c:\eb21sp4,c:\eb21sp4\qps,c:\eb21sp4\qxtendserver
  uuid=434f443f55229669:639a3e:10ba612054a:-8000
  workDir=c:\eb21sp4\qps
  fileUploadDirectory=c:\eb21sp4\qps\upload
  defaultService=0
  groupName=
  initialSrvrInstance=1
  maxSrvrInstance=5
  minSrvrInstance=1
  srvrAppMode=Development
  srvrDebug=Disabled
  srvrLoggingLevel=1
  srvrLogAppend=0
  srvrMinPort=45001
  srvrMaxPort=45010
  brkrLogAppend=0
  brkrLoggingLevel=1
  userName=
  srvrStartupParam=-rereadnolock -p scheduler\qps-disp.p -
  weblogerror -pf c:\eb21sp4\qps\QPS.pf -cpstream UTF-8 -cpinternal
  UTF-8
  #
```

By default everything should be right. There are some settings you will need to adjust depending on your implementation.

Note The example file refers to a `fileUploadDirectory`. This is not used by QAD Production Scheduler and can typically be ignored. However, adding a valid directory to the definition as shown is a good policy.

Oracle Modifications

For QPS brokers that work against an QAD ERP on Oracle database, additional environment settings should be added to the file. The environment section must listed before the WebSpeed broker information. In the following example, the broker is `qps`, which requires the `Environment.qps` definition.

```
[Environment.qps]
ORACLE_SID=eb21ora
ORACLE_HOME=/dr02/app/Ora10GR2/oracle/product/10.2.0/db_1
PATH=$ORACLE_HOME:$ORACLE_HOME/bin:$DLC:$DLC/bin:$PATH
LIBPATH=$ORACLE_HOME/lib:$DLC/lib:/usr/lib:/lib
NLS_LANG=AMERICAN_AMERICA.WE8ISO8859P1
NLS_NUMERIC_CHARACTERS=,.

[UBroker.WS.qps]
appserviceNameList=qps
...
```

Append the Sample File

To define the QAD Production Scheduler WebSpeed broker, append the sample broker file created in step 1 on page 35 (`qadps-ubroker.properties`) to the bottom of your existing `ubroker.properties` file.

- 1 Stop the Progress admin server.


```
./proadsv -stop
```
- 2 Create a backup of `ProgressInstallDir/Properties/ubroker.properties`.
- 3 Open your working copy of `ubroker.properties` in a text editor.
- 4 Open `QPSInstallDir/qadps-ubroker.properties` in a separate editor.
- 5 Copy `qadps-ubroker.properties` and paste it at the bottom of `ubroker.properties`.

6 Save the file.

If you have created multiple QAD Production Scheduler instances on this host—for example, `qpsprod` and `qpsstrain`—you must repeat these steps for each instance.

To test WebSpeed in a multi-tier environment, set `svrDebug=Enabled`. See “Verifying the WebSpeed Messenger” on page 79.

Copy WebSpeed Files to Apache

These steps must be completed for both new installs and conversions.

Configuration of Apache for communication to WebSpeed takes place through the Common Gateway Interface (CGI).

- For Linux and UNIX environments, a shell script is used to implement this CGI integration, `qps.sh`.

Copy `QPSInstallDir/qps.sh` to your Web server `/cgi-bin` directory under Apache.

For additional QPS instances such as `qpsstrain`, copy those shell scripts to the `/cgi-bin` directory as well.

- For Windows, copy the WebSpeed `cgip.exe` file from `ProgressInstallDir\bin` into your Web server `\cgi-bin` directory.

Configuring Additional Instances

If you need to create more than one QAD Production Scheduler instance on a single host machine—one for production, one for training, and so on—there are several required steps. The following information lists the steps, and where relevant, the page in this document where the step is described.

This guide refers to these additional database sets as instances of QAD Production Scheduler. They can also be referred to as Webapps on the Apache Web server, as QPS brokers, and QPS databases or database sets. One name such as `qpsprod` should be used for each of these different aspects of each instance.

The following steps use the instance name of `qpstrain` as an example.

- 1 Create the additional services for the training databases: `qpstrain-srv`, `mfgtrain-srv`, and `admtrain-srv`.
- 2 Create a directory on the Apache Web server for each QPS instance; for example, `ApacheHome/html/qpstrain`. See “Multiple QPS Version Prerequisites” on page 18.
- 3 Archive or rename the `qadps-ubroker.properties` file located in `QPSInstallDir/qps` to avoid having this overwritten.
- 4 Using the installation media, skip the installation of the QPS Repository and Progress Source Files under “QAD Production Scheduler Component Installs” on page 20 and just install the Web server files.
 - a When you are asked to enter a subdirectory of your Web Server Document directory for the web application files, enter:
`ApacheHome/html/qpstrain` on Linux/UNIX and
`ApacheHome\htdocs\qpstrain` on Windows. See step 12 on page 21.
 - b When you are prompted to enter the WebSpeed URL, the broker name must be the name of the QAD Production Scheduler instance you are creating. See step 14 on page 22.
- 5 In `QPSInstallDir`, copy `wk0650.ini` to `wk0655.ini`. In the new version make the following changes:
 - a Make sure all Status entries are `Status=Not Run`.
 - b Change `WorkFlowDesc` to something recognizable; for example, `QPS Train Database`.
 - c Change all instances of `qpsprod.db` to `qpstrain.db`.
 - d Change all instances of `DBSetName=QPS` to `DBSetName=QPSTrain`.
 - e Delete the section entitled `Compile`.
 - f Save and exit the file.

- 6 Prior to running the revised workflow, open Configure|Database Set Maintenance in QPS/UTIL. In Database Set Maintenance, do the following:
 - a Select the QPS database set and choose Copy Set. Name the new set appropriately—for example, QPSTrain—and choose OK. (Do not use spaces in the name.)
 - b With the new set selected, tab to the Set Overview frame and select the `qpsprod` database. Choose Edit DB.
 - c In Edit DB, change the database name to match your planned new database; for example `qpstrain`.
 - d Change the Service Name to match the service you added in step 1; for example, `qpstrain-srv`. Choose OK to save your changes.
- 7 Create a new database structure file for `qpstrain`. Copy `QADInstallDir/db/qpsprod.st` to `QADInstallDir/db/qpstrain.st`.
- 8 Create a new Web script file. Copy `ApacheHome/cgi-bin/qps.sh` to `ApacheHome/cgi-bin/qpstrain.sh`.
 - a Open and modify the new `qpstrain.sh`. Change the line:
`$DLC/bin/cgiip -i qps`
to read:
`$DLC/bin/cgiip -i qpstrain`
- 9 From the Configure menu, select QAD ERP Guided Setup. When the Operation Sets screen displays, locate the workflow and choose Run Set. During the workflow, do the following:
 - a In the QPS Setup screen, enter a new Webapp name; for example, `qpstrain`.
 - b Also in the QPS Setup screen, enter a new Web server home—for example, `/var/www/html/qpstrain` for Linux or UNIX; `c:\Apache2.2\htdocs\qpstrain` on Windows.
 - c The database creation step should use your new database name from the edited workflow; for example, `qpstrain`.

- d Use `qps.df` to build the new database.
 - e Verify the services and paths for the new database set in Database Set Maintenance; for example, QPSTrain.
 - f For Windows, in the Select Icon Folder screen, enter a new folder name for the set; for example, QPS Train 2.0.
 - g Generate scripts for the new database set; for example, QPSTrain.
 - h When you generate the `ubroker.properties` definition for the new database set, make sure the correct database set is selected, enter a new Broker Name; for example, `qpstrain`, and set a new Broker Port; for example 45500.
 - i Complete and exit the workflow.
- 10 Stop the Progress admin server.


```
./proadsv -stop
```
 - 11 Back up `ubroker.properties` and copy the newly generated `ubroker` section to the bottom of the file. See “Generate Sample QAD Production Scheduler Broker File” on page 35.
 - 12 Create a QXtend Inbound receiver and connection pool for the new instance as documented in *Technical Reference: QXtend Inbound*. Remember to use the `QPSclient.QPSTrain` file when you define the connection pool rather than `QPSclient.QPS`.
 - 13 Follow the steps for starting and testing the additional instance using the steps in the next section.

Starting QAD Production Scheduler

These steps must be completed for both new installs and conversions.

There are three major components of QAD Production Scheduler: the QAD Production Scheduler interface, the QAD Production Scheduler background processes, and the QXtend component. In the following section, you validate that the QAD Production Scheduler interface runs. To do this, you load validation data, start the WebSpeed broker, and start QAD Production Scheduler.

Following this, you configure and validate the QAD Production Scheduler background processes, then the QXtend component.

Enter a Site to Test Background Processes

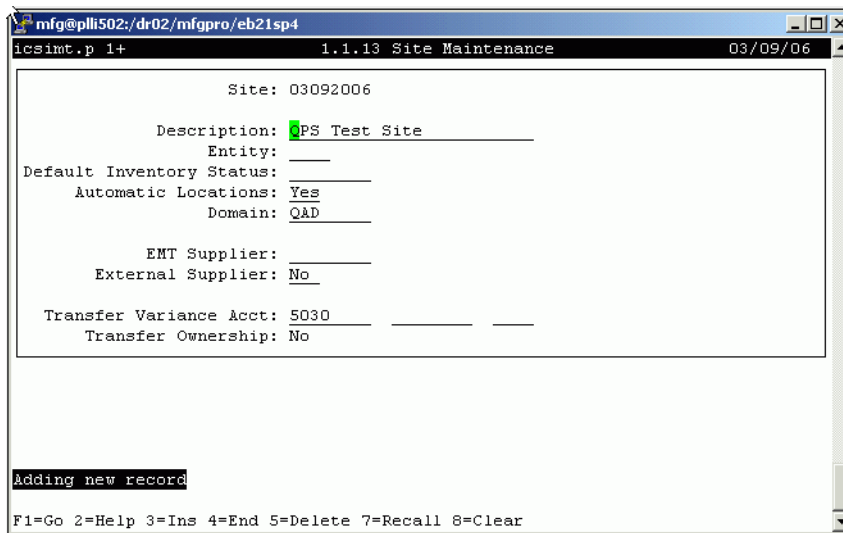
These steps can be completed for both installations and conversions.

Entering a new site into the production database allows you to verify that the background processes are working when you reach “Launching the Background Processes” on page 62.

If you are performing a conversion and have existing sites in the production database, enter a test site that can be easily located; for example, one starting with zero so that it sorts to the top of the list.

- 1 For UNIX systems, start your production client from
QADInstallDir:
`./client.Production`
On Windows, start the Production client from the Start menu.
- 2 Log in and navigate to Site Maintenance (1.1.13).
- 3 Create a recognizable new site value and press Go to save the site.
Note For newly-installed production databases without converted data, you will need to create an entity (as well as a domain where applicable) and set up an inventory status prior to creating a new site.

Fig. 1.33
Entering a QPS
Test Site



4 Exit from QAD ERP and continue with the next section.

After completing the test, you can delete the site from within QAD ERP.

Validating the WebSpeed Broker

These steps must be completed for both new installs and conversions. The steps must be followed exactly in the sequence provided.

To start the QAD Production Scheduler interface, you must launch the required servers and background processes.

- 1 If Apache is not already running, start it on UNIX systems by navigating to the *ApacheHome/bin* directory and executing the Apache start command:

```
apachectl start
```

On Windows, the method is dependent on whether you have set Apache up as a service or to run in a console. See the Apache documentation for additional information.

To test whether Apache is running, open a browser to `http://\host.domain.com:port`. If Apache is running, the Apache home page displays.

Note The port must only be specified if you are using a port other than the Apache default, 80.

- 2 Start Tomcat if it is not running. In *TomcatHome/bin*, launch:

```
./startup.sh
```

- 3 Start the QAD ERP database servers. This command is run from *QADInstallDir*. On Windows, the servers are started from an icon on the Start menu.

```
./start.Production
```

For additional instances such as QPSTrain, launch the appropriate database servers; for example, `./start.Train`.

- 4 Start the QAD Production Scheduler database servers. This command is run from *QPSInstallDir*. On Windows, the servers are started from an icon on the Start menu.

```
./start.QPS
```

You should see messages alerting you that the production databases in the set, `mfgprod` and `admprod`, are already running.

For additional instances, this command would be

```
./start.<InstanceName>; for example, ./start.QPSTrain.
```

- 5 Start the Progress admin server. Run this command from the *ProgressInstallDir/bin* directory.

```
./proadsv -start
```

Verify it is started:

```
./proadsv -query
```

- 6 From the same location, start the Progress name server.

```
./nsman -i NS1 -start
```

Verify it is started:

```
./nsman -i NS1 -query
```

- 7 From the same location, start the QAD Production Scheduler broker.

```
./wtbman -i qps -start
```

Verify it is started:

```
./wtbman -i qps -query
```

You should see a broker status summary. It should be ACTIVE.

▶ If you are on AIX, make sure you have completed “AIX Requirements” on page 15.

For additional QPS instances such as QPSTrain, the commands would be:

```
./wtbman -i qpstrain -start
./wtbman -i qpstrain -query
```

- 8 You should now be able to connect to the QAD Production Scheduler database through WebSpeed using a test program on your browser client, `sxwbmt.p`. For Linux and UNIX environments this URL looks like:

```
http://host.domain.com:port/cgi-bin/qps.sh/scheduler/sxwbmt.p
```

For additional instances, replace `qps.sh` in the URL with the instance script; for example, `qpstrain.sh`.

For Windows environments, this URL looks like:

```
http://localhost:port/cgi-bin/cgiip.exe/WService=qps/scheduler/sxwbmt.p
```

Note The port must only be specified if you are using a port other than the Apache default, 80.

The WebSpeed test page displays.

Fig. 1.34
WebSpeed Test
Program in Internet
Explorer

The screenshot shows a web browser window with the following content:

- Address bar: `http://localhost/cgi-bin/cgiip.exe/WService=qps/scheduler/sxwbmt.p`
- Function:
- Context:
- XML:
- Messages:
- Submit button

- 9 To test the WebSpeed connection, type `getEnabledSiteList` into the Function field and choose Submit.

Once the system has been fully configured, the form will return a list of QAD Production Scheduler enabled sites. However, since no sites have been brought into the QAD Production Scheduler database yet, the resulting list displayed in the XML pane is empty. It should show:

```
<SiteList>
</SiteList>
```

The Messages pane displays `SESSION_HAS_BEEN_DELETED`.

If there are problems with connecting to WebSpeed, validate that the database connections in your `.pf` file are correct by executing:

```
$DLC/bin/_progres -pf QPSInstallDir/QPS.pf
```

If there are problems connecting to the databases through this Progress client, resolve the problems in the `QPS.pf` file.

Test the QAD Production Scheduler Interface

These steps must be completed for both new installs and conversions.

You can now validate that QAD Production Scheduler runs in your browser and validate the auto-synchronization process.

- 1 Test the QAD Production Scheduler application in the browser client by entering:

```
http://host.domain.com:port/qps/login.html?url=qps.html
```

Note The port must only be specified if you are using a port other than the default, 80.

- 2 If you do not already have the appropriate Java plug-in installed, you will be prompted to install it now. Complete the installation and choose Finish.

Note Automatic upgrades to the Java plug-in do not always work properly. It may be necessary to go to the following URL and manually download and install the offline version of the 1.4 Java Runtime Environment (JRE):

<http://java.sun.com>

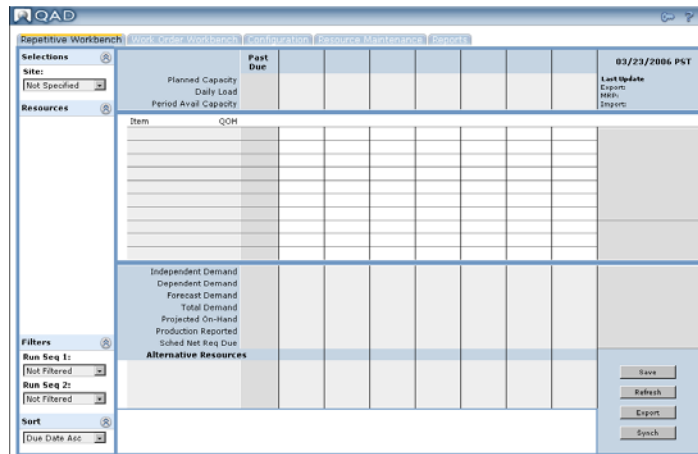
- 3 When QAD Production Scheduler comes up, the login screen displays. The initial login is `mfg` with a valid password. By default, the `mfg` password is blank.

Fig. 1.35
QAD Production
Scheduler Login
Screen



4 The Repetitive Workbench displays.

Fig. 1.36
QAD Production
Scheduler
Repetitive
Workbench



5 Select the Configuration tab. The Configuration screen displays with Global Settings displaying in the Site field. Enter the correct QXtend Version (1.2, 1.4, 1.5), QXtend URL, QXtend Receiver, and Server Timezone for your environment. The QXtend URL is:

`http://qxtendhost:8080/qxtendserver`

Where *qxtendhost* is the fully qualified host where QXtend Inbound resides, such as `linx09.corp.com`.

The QDoc versions are covered in greater detail in the user guide. The required values are:

QDoc Type	eB	eB2	eB2.1 or later
Work Order	eB_1	eB2_1	eB2_1
Repetitive	eB_1	eB_1	eB_1
MRP	eB_1	eB_1	eB_1

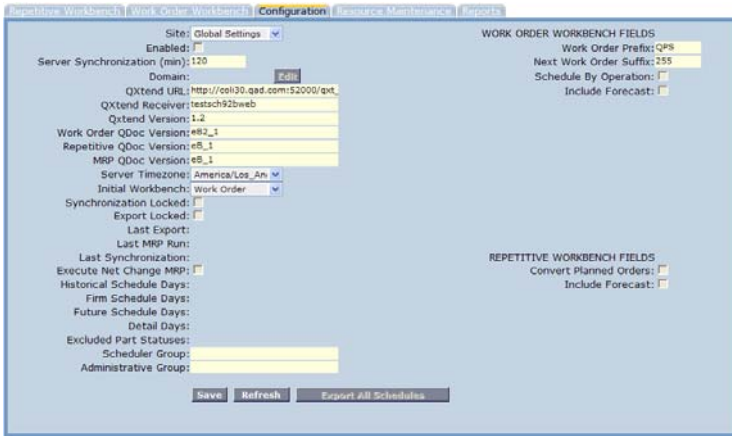


Fig. 1.37
QAD Production Scheduler Configuration With QXTend Values Entered

- 6 Choose Save. No additional data is visible until the steps in the next section are completed.

You can either leave the browser open or close it at this time.

Launching Background Processes

These steps must be completed for both new installs and conversions.

The QAD Production Scheduler background processes run the synchronization between QAD ERP and QAD Production Scheduler and enable the manual export of QAD Production Scheduler data to QAD ERP. These processes are launched by the scripts created in “Create Windows Icons or UNIX Scripts” on page 32.

Modify Log Locations for Background Processes (Optional)

The log files for each background process are located in the Log Directory specified in step 6 on page 25. A `cd` (change directory) command sets this directory in the start scripts before the process is launched. You can modify the `cd` command to a directory of your choice; for example:

```
cd /dr01/qps/logs
```

Make the changes in each of the start scripts. If you make these changes, add the identical `cd` command to each stop script as well. Do not use the `/tmp` directory on UNIX to avoid background file clean-ups. The scripts are:

- `startall.QPS`
- `startprocautofirm.QPS`
- `startprocautosynch.QPS`
- `startprocmanexport.QPS`
- `startprocmansynch.QPS`
- `stopall.QPS`
- `stopprocautofirm.QPS`
- `stopprocautosynch.QPS`
- `stopprocmanexport.QPS`
- `stopprocmansynch.QPS`

No log file changes are required in the `startall` or `stopall` scripts. They merely call the other four start or stop scripts.

Note You should monitor the log directory and regularly delete old files to avoid using excessive disk space.

Launching the Background Processes

You should now be able to start the processes.

- 1 In a command window, launch the background processes from `QPSInstallDir`:

```
./startall.QPS
```

For Windows, use the QAD QPS start icon from the Start menu.

Each process creates log files in the directory where the process was started. The auto-synchronization process should update QAD Production Scheduler with QAD ERP site data.

- 2 Open QAD Production Scheduler in a browser, `http://host.domain.com:port/qps/login.html?url=qps.html`, or reopen it in the open browser window.

Note The port must only be specified if you are using a port other than the Apache default, 80.

▶ See “Modify Log Locations for Background Processes (Optional)” on page 61.

- Return to the Configuration tab and open the Site drop-down. For conversions, you should see the existing QAD ERP sites displayed. For both new installs and conversions, you should also see the test site you created in “Enter a Site to Test Background Processes” on page 55.

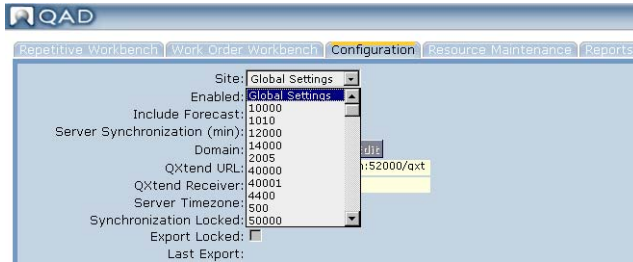


Fig. 1.38
QAD Production Scheduler Site Values After Auto Synchron

Configuring QXtend

These steps must be completed for new installs only.

Once the QAD Production Scheduler application is installed and compiled, you can configure QXtend Inbound to accept messages from QAD Production Scheduler to be directed to QAD ERP.

Summarized QXtend Setup

This section provides a summarized set of steps to follow. See the QXtend documentation for detailed instructions for each step.

Step	Task
1.	Start Tomcat.
2.	Ensure database servers for QAD ERP and QAD Production Scheduler are running. This was done during “Starting QAD Production Scheduler” on page 54 of this guide.
3.	Launch QXtend Inbound: <code>http://localhost:8080/qxtendserver</code> . Note: This example uses the default Tomcat port, 8080. Adjust as needed if this has been changed in your installation.
4.	If you have not done so already, load the QAD ERP license codes for QXI.
5.	Modify <code>myQdocWebService.wsdl</code> .

Table 1.6
QXtend Inbound Configuration Steps

Step	Task
6.	<p>Add a receiver for each QAD ERP instance or domain. The example used in this guide is eB2QPS.</p> <p>For each receiver, it is fine to assign all available QDocs to the receiver. (The maintainWorkOrder QDoc is required for QPS.)</p>
7.	Launch the Connection Pool Manager.
8.	<p>Add a new UIAPI connection pool for each receiver.</p> <p>Make sure that:</p> <ul style="list-style-type: none"> • The Pool Name is the same as the Receiver Name (eB2QPS); the Server Startup Script is <code>QPScilent.QPS</code>, located in <code>QPSInstallDir</code>. • For Windows implementations only, the prompt in the Server Startup Script should be <code>></code>. In addition, the script must change directories into the appropriate directory before launching <code>QPScilentQPS.bat</code>. A sample Windows script is as follows: <pre>login: mfg password: \$PASSWD domain: qad > cd c:\mfgpro\eb2sp10\qps > QPScilentQPS.bat</pre> • Maximum Connections is set to 9. • For Windows implementations only, set the Operating System Win32/NT to true. • The Connection Setup Userid is the QXtend Inbound telnet connection user. • Leave Connection Setup Password as is. • For QAD ERP versions that support domains, the Domain value is set, for example, to <code>Domain1</code>. <p>For multi-tier, the URL in step 3 of this table points to your QXtend server; the Host value points to your Progress server. See “Multi-Tier Installations” on page 74.</p>
9.	Save the new connection pool. The new pool is started automatically. Make sure the connection is idle and active before proceeding.

Loading QAD Production Scheduler QDoc Schemas

Important This set of steps is required only for QXtend Inbound versions prior to 1.2.1.6. Starting in 1.2.1.6, the QPS QDocs are included in the QXI source.

When you have completed the QXtend Inbound steps in Table 1.6, you must load the QAD Production Scheduler QDoc schemas.

During step 16 on page 23 of the media extraction process, a temporary directory was created for the QAD Production Scheduler QDoc schemas. The following steps load these into QXtend.

- 1 In the Configuration Manager, select your QAD ERP version under the Schemas menu.
- 2 When the schemas display, choose Add.

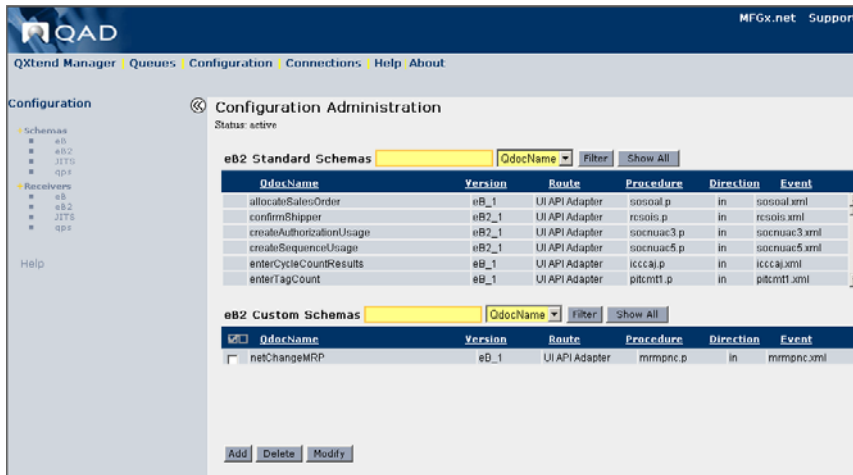


Fig. 1.39 QXtend Schemas for eB2

- 3 If QXtend Inbound is active, you are asked whether to suspend or continue. Assuming QXI is not in production mode during these updates, choose Continue. The Add Schemas screen displays.

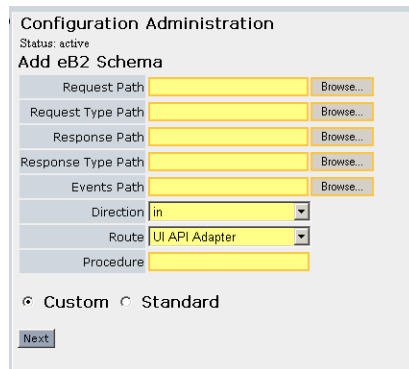


Fig. 1.40 Add QXtend Schemas

- 4 Use the Browse button for each of the path values, and enter or accept the field values listed in Table 1.7. These files are located in *TMPOC*, created in step 16 on page 23.

Table 1.7
QXtend Schema
Add Values

Field	Value
Request Path	netChangeMRP-eB_1.xsd
Request Type Path	netChangeMRPType-eB_1.xsd
Response Path	netChangeMRPResponse-eB_1.xsd
Response Type Path	netChangeMRPResponseType-eB_1.xsd
Events Path	mrmpnc.xml
Direction	In
Route	UI API Adapter
Procedure	mrmpnc.p

- 5 When all the values are entered correctly, choose Next. The Add Receiver Support screen displays. Use this screen to assign the new schemas to your QAD Production Scheduler receivers.

Fig. 1.41
Add Receiver
Support



- 6 Select the QAD Production Scheduler receiver and choose Done. The .xsd files are written to:

TOMCAT_HOME/webapps/qxtendserver/WEB-INF/**schemas**/eB2/custom

Where eB2 is your QAD ERP version. A message displays informing you that the schemas were added successfully to your receivers.

The .xml file is written to:

TOMCAT_HOME/webapps/qxtendserver/WEB-INF/**events**/eB2/custom

Installation Reference

The files generated and tailored either by the generic QAD installer or by QPS/UTIL for QPS 2.0 on a UNIX or Linux System are shown in Table 1.8.

Description	Location and File Name	Page	
QPS installation settings	<i>InstallLogDirectory/instqps.ini</i>	20	
QPS installation log	<i>InstallLogDirectory/qps.log</i>		
QPS/UTIL executable	<i>QPSInstallDir/qpsutil</i>		
QPS/UTIL workflow	<i>QPSInstallDir/wk0650.ini</i>		
QPS/UTIL support files	All in <i>QPSInstallDir</i> mfgutil.ini mfguprog.ini mfgutora.ini mfgutpro.ini		
QPS database set scripts	All in <i>QPSInstallDir</i> start.QPS QPSclient.QPS startall.QPS startprocautofirm.QPS startprocautosynch.QPS startprocmanexport.QPS startprocmansynch.QPS stopall.QPS stopprocautofirm.QPS stopprocautosynch.QPS stopprocmanexport.QPS stopprocmansynch.QPS stop.QPS	32	
QPS WebSpeed broker definition	<i>QPSInstallDir/qadps-ubroker.properties</i>		35
QPS WebSpeed shell script	<i>QPSInstallDir/qps.sh</i>		
Copied to <i>ApacheHome/cgi-bin</i>			

Table 1.8

Generated and Tailored Files for UNIX and Linux Systems During Install

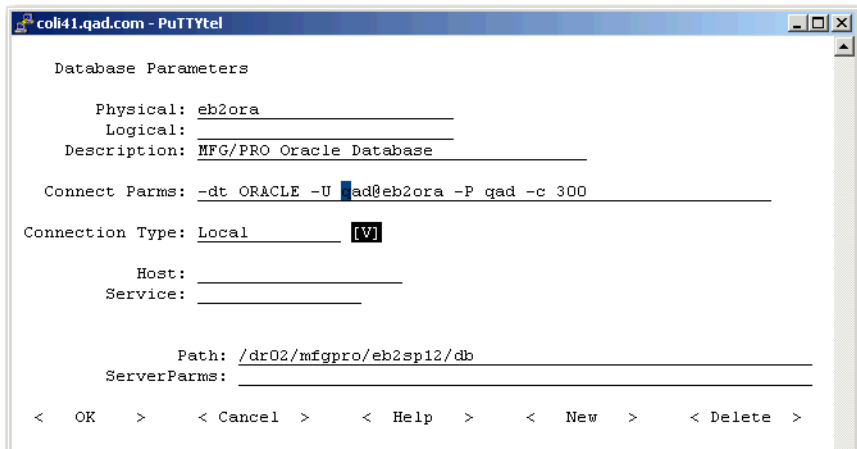
Description	Location and File Name	Page
QPS application properties files	All in <i>ApacheHome</i> /html/qps/	n/a
	config.properties	
	domain.properties	
	login.properties	
	repetitive.properties	
	report.properties	
	resource.properties	

Running QAD Production Scheduler with QAD ERP Oracle

After either a new QAD Production Scheduler install or an update, you can modify the QAD Production Scheduler scripts to connect to QAD ERP Oracle databases. The QAD Production Scheduler database itself is defined only as a Progress database. However, QAD Production Scheduler can be run with QAD ERP Oracle implementations. To accomplish this, complete the following steps:

- 1 Set up the QAD ERP Oracle database connection in Database Set Maintenance. This ensures that the database connection parameters are correct for each script generation. For more information on setting up an Oracle database connection, refer to *Installation Guide: Oracle Database* for your QAD ERP version.

Fig. 1.42
Main Oracle
Database Definition
in Database Set
Maintenance



The completed database sets should include QPS and QPS Compile. QPS should contain the Oracle schema holder and the main QAD ERP Oracle database, as well as the QPS database.

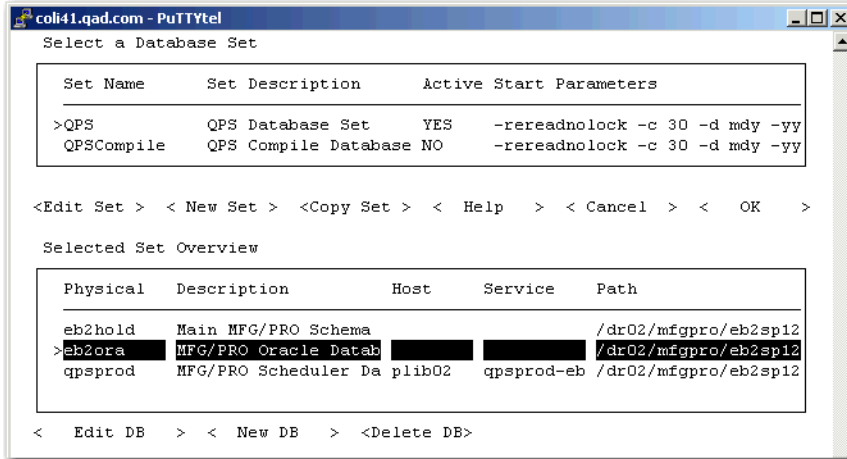


Fig. 1.43 Completed Database Sets

The QPS Compile database set should reference the QAD ERP oraempty database and the QPS qpsemtly database.

Important The database parameters for the oraempty database must include the logical name qad, as shown in the following example.

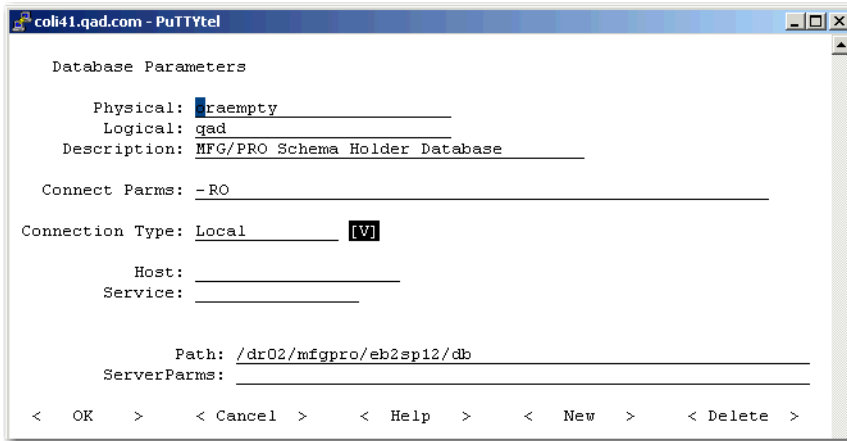


Fig. 1.44 Sample Connection Parameters for oraempty Database in Compile Database Set

2 Generate new connection scripts.

Once the scripts are generated, edit the `start.QPS` and `stop.QPS` scripts located in `QPSInstallDir` to remove references to the schema holder and QAD ERP production databases. This avoids conflicts between your standard QAD ERP launch scripts and the QPS scripts. Comment the connections out using the pound sign (#) as shown in the following example:

```
#$DLC/bin/_mprosrv /dr02/mfgpro/eb21sp4/db/eb21holder
#$DLC/bin/_mprosrv eb21ora
$DLC/bin/_mprosrv /dr02/mfgpro/eb21sp4/db/qpsprod -L
100000 -c 350 -B 25000 -spin 1000 -Mf 500
```

- 3 After the QPS scripts have been generated for either an install or an update, set the Oracle environment variables. For UNIX, do this in each of the start scripts for each of the background processes. Insert the Oracle environment variables after the Progress variables. Make sure to set the Oracle password for the environment using the Oracle method best suited for your environments.

Note The `ORASOPATHNAME` variable is required only for Sun and Hewlett-Packard operating systems on Progress 9.1.

```
ORACLE_HOME=/dr01/oracle/sun/product/9.2
ORASOPATHNAME=
/dr01/oracle/sun/product/9.2/lib32/libclntsh.so.9.0
ORASID=ora92b
NLS_LANG=AMERICAN_AMERICA.WE8ISO8859P1
export ORACLE_HOME
export ORASOPATHNAME
export ORASID
export NLS_LANG
```

For Windows environments, insert the variables in `QPS.ini` located in `QPSInstallDir`. These would look like:

```
SET ORACLE_HOME=c:\oracle\sun\product\9.2
SET ORASID=ora92b
SET NLS_LANG=AMERICAN_AMERICA.WE8ISO8859P1
```

- 4 In UNIX environments, the `PATH` variable must contain both `$$$DLC` and `$$$DLC/bin`.

5 Modify the QXtend Inbound compile list and recompile QXtend.

- a Copy `QADInstallDir/qps/scheduler/export/oracle_mfwb01aa.p` to `QADInstallDir/qxtendserver/qxxrc`.
- b Modify the QXtend Inbound `QADInstallDir/qxtendserver/qxtendcompile.wrk` to include `oracle_mfwb01aa.p` at the end of the file as shown:

```
gpfrminf.p
gpstatus.p
mfqxtend.p
mfsyad02.p
mfwb01aa.p
mfwsock2.p
mfww01.p
mfww01b.p
mfwwc01.p
oracle_mfwb01aa.p
```

- c Save your changes.
- d Open `QADInstallDir/toplevel.lst` and add `oracle_mfwb01aa.r` to the end.

```
pxdeclre.r
pxdiscnt.r
pxgblmgr.r
pxldgbl.r
pxlogdb.r
pxmsghdl.r
pxpgmmgr.r
pxrpc.r
pxtools.r
rpc.r
oracle_mfwb01aa.r
```

- e Compile QXtend Inbound as described in the QXtend documentation. Make sure the Compile database set selected for the compile contains the QAD ERP `oraempty` database.

Note You must also add the QPS installation directory to the QXI compilation path to ensure that compilation is successful.

- 6 Edit `QADInstallDir/qps/QPSClient.QPS`. Change the `-p` parameter as shown:

```
#
# Start MFG/PRO.

# exec $DLC/bin/_progres &DB etc
$DLC/bin/_progres -rereadnolock -c 30 -d mdy -yy 1920 -BT 350
-D 100 -mmax 3000
-nb 200 -s 128 -noshvarfix -pf
/dr02/mfgpro/eb21sp4/qps/QPS.pf -p oracle_mfwb01aa.p -param
mfgwrapper=true,apimode=true
```

Starting QAD Production Scheduler

The following procedures should be followed in the specified sequence for QAD Production Scheduler to work correctly.

Note For UNIX/Linux systems, always log in as the `mfg` user.

- 1 Start the database servers for `mfgprod`, `admprod`, and `qpsprod`.

```
QPSInstallDir> ./start.QPS
```

- 2 Start Tomcat if it is not running:

```
TomcatHome/bin> startup.sh
```

- 3 Validate that the Admin server is running:

```
ProgressInstallDir/bin> proadsv -q
```

If the Admin server is not running:

```
ProgressInstallDir/bin> proadsv -start
```

- 4 Validate that the Name server is running:

```
ProgressInstallDir/bin> nsman -q
```

If the Name server is not running:

```
ProgressInstallDir/bin> nsman -name NS1 -start
```

5 Start the WebSpeed broker:

```
ProgressInstallDir/bin> wtbman -i <broker name> -start
```

Check the broker status before proceeding to the next step.

```
ProgressInstallDir/bin> wtbman -i <broker name> -q
```

Repeat until the status shows as ACTIVE.

6 Start the background processes:

```
QPSInstallDir> ./startall.QPS
```

7 Open the browser and type the URL to start QAD Production Scheduler:

```
<http://hostname.domain.com>/qps/login.html?url=qps.html
```

Note If this is for a different QAD Production Scheduler instance, enter that name here as the directory and html name; for example:

```
<http://hostname.domain.com>/qpstrain/login.html?url=qpstrain.html
```

8 Check that idle connections are available in the connection pool setup in Qxtend Inbound for QAD Production Scheduler.

Stopping QAD Production Scheduler

The processes to be followed to shut QAD Production Scheduler down are as follows:

1 Shut down the background processes:

```
QPSInstallDir> ./stopall.QPS
```

2 Shut down the WebSpeed broker:

```
ProgressInstallDir/bin> wtbman -i <broker name> -stop
```

3 Shut down the database servers for mfgprod, admprod, and qpsprod.

```
QPSInstallDir> ./stop.QPS
```

Note If you experience a decrease in system performance on UNIX or Linux systems, it may be due to background processes that were not shut down prior to the database servers being shut down. Check the active processes on the system following a complete shutdown, and kill any open background processes.

▶ See Figure 1.2, “Multi-tier QAD Production Scheduler Deployment,” on page 9.

Multi-Tier Installations

QAD Production Scheduler can be deployed over one, two, three, or four hosts. A two-host deployment would combine the Web and QXtend servers and the Progress and database servers. A three-tier environment would split QXtend and the Web servers and keep the Progress and database servers on a single host.

This section describes the steps required to deploy a four-host system consisting of a Progress server, a database server, a Web server, and a QXtend server.

Configuring the Qxtend Server

The QXtend server requires the following components:

- Java 1.4.2_07 +
- Tomcat 5.5 +
- QXtend Inbound servlet code 1.2.1.4 +
- QAD Production Scheduler custom QDocs (for pre-QXI 1.2.1.6 only)

When you configure QXtend Inbound, as described in Table 1.6 on page 63, you will point the browser URL to your QXtend server; the Host value in Connection Pool Configuration Settings points to your Progress server.

Configuring the Apache Web Server

The Web server requires the following components:

- Java 1.4.2_07 +
- Apache 2 +
- **Important** IBM AIX users require Apache version 2.2.4 or higher.
- WebSpeed Messenger
- QAD Production Scheduler Web server files

To install and configure the WebSpeed Messenger, see “Configuring a Remote Messenger for QAD Production Scheduler” on page 76.

Configuring the Database Server

The database server requires the following components:

- Progress 9.1e +
- QAD ERP database server and language files
- QAD Production Scheduler Progress code

Create the empty and production QAD ERP databases on this server.

Since the QAD Production Scheduler media does not separate out the application code from the database code, you must install the complete QPS Progress component to both the database and Progress servers. Do the following to complete the install on the database server:

- 1 Modify the workflow file *QPSInstallDir/wk0650.ini*:
 - a Remove the QAD Production Scheduler setup step, and all steps after the database truncate step.
 - b Save and exit the file.
- 2 Run the workflow in QPS/UTIL as described in “Running the New Install Workflow” on page 23.

Configuring the Progress Server

The Progress server requires the following components:

- Progress 9.1e +
- Database Server QAD ERP
- Client Application QAD ERP
- QXtend Inbound Progress code 1.2.1.6 +
- For Windows only: Georgia Softworks Telnet Server 6.50 +
- QAD Production Scheduler Progress code

Create the QAD ERP database sets on this server, making sure to correctly define the client/server parameters for each database, generate scripts, and compile the application code.

Since the QAD Production Scheduler media does not separate out the application code from the database code, you must install the complete QAD Production Scheduler Progress component to both the database and Progress servers. Do the following to complete the install on the Progress server:

- 1 Modify the workflow file *QPSInstallDir/wk0650.ini*:
 - a Remove only the database create, load, and truncate steps.
 - b Save and exit the file.
- 2 Run the workflow in QPS/UTIL as described in “Create qpsempty” on page 26.

Configuring a Remote Messenger for QAD Production Scheduler

The WebSpeed Messenger provides communication links between the Apache Web server and the Progress WebSpeed server when the Web server resides on a different server than the WebSpeed broker and agents.

Prerequisites

- Web access
- Administrator access to Progress installation on the database server

Installation

Use the following steps to download and install the appropriate WebSpeed Messenger files:

- 1 Download the WebSpeed Messenger executable from the Progress Web site.

<http://www.progress.com/esd/index.ssp>

Make sure you download the executable for your specific server. Additionally, download any related documentation.

- 2 Review the documentation provided on the download page for additional installation or server requirements information.

- 3 Install the WebSpeed Messenger on the host where the Web server is located. The default installation directory will be the same as the default Progress directory on the Progress host. We recommend that you choose a distinctive directory name such as

`/dlc91e/WS3msngr` to avoid confusion between the two servers.

Use the installation instructions provided on the download page. Note the installation directory; during the configuration steps, you edit and copy several files in this directory.

- 4 When the install is complete, you must correct the results of a WebSpeed bug. During the install, files are written to the Apache directory, `/var/www/html/webspeed31D`. This should be `/var/www/html/webspeed31E`. Either rename the directory (Windows), or for Linux or UNIX, create the `webspeed31E` directory in the correct location and move the files across.

This problem can occur in later versions of Progress as well, with OpenEdge 10.0B installing to a directory named `/webspeed` on the Apache server. In all events, make sure the files are located in `/var/www/html/webspeed31E`.

Configuring the Remote WebSpeed Messenger

These steps assume you are configuring WebSpeed on a UNIX system. If you are installing on a Windows server, these steps also apply; the relative paths are the same. The only difference is that Windows requires the `cgiiip.exe` executable instead of the `qps.sh` executable.

On the Web server or QXtend server, go to the `properties` directory under the Messenger installation directory. Find `msngrs.properties`, make a copy of this file, and rename the copy to `ubroker.properties`.

Note On Progress 10.0B, there is no `msngrs.properties`. Just use the `ubroker.properties` file in the same location.

- 1 Using a text editor, open the `ubroker.properties` file in `MessengerInstallDir/properties`.
- 2 Find the `[NameServer.NS1]` section and make sure the following parameters appear:

```
[NameServer.NS1]
  srvrLogFile=$WORKDIR/NS1.ns.log
```

```
environment=NS1
autoStart=0
hostName=ProgressServerName
location=remote
```

The Nameserver on the Web server Remote Messenger `ubroker.properties` file must match exactly the Nameserver on the WebSpeed Progress server side `ubroker.properties`. QAD Production Scheduler uses `[Nameserver.NS1]` by default in the `gadps-ubroker.properties` file generated during the installation.

- 3 Find the `[WebSpeed.Messengers.CGIIP]` section. Make sure the following parameters appear:

```
# WebSpeed CGIIP Messenger Properties
#
[WebSpeed.Messengers.CGIIP]
msngrExecFile=/usr/dlc/WS3msngr/bin/cgiip
msngrScriptFile=
AllowMsngrCmds=1
Port=5162
registerNameServer=1
```

Comment out all other lines using the pound or hash symbol (#) at the beginning of each line.

- 4 Verify your edits and save the file.
- 5 On UNIX systems only, create `qps.sh` in `ApacheHome/cgi-bin`. The contents of this file are:

```
#!/bin/sh
DLC=WSMessengerInstall;export DLC
PROMSGS=$DLC/promsgs;export PROMSGS
WRKDIR=/tmp;export WRKDIR
$DLC/bin/cgiip -i qps
```

- 6 On Windows systems only, copy `cgiip.exe`, the WebSpeed executable from the `WebSpeedMessengerInstallDir/bin` directory, to `ApacheHome/cgi-bin`.

Verifying the WebSpeed Messenger

Ping

For Ping to work, your `ubroker.properties` broker definition must include `srvrDebug=Enabled`. See “Configuring WebSpeed” on page 49.

Ping verifies that the WebSpeed Messenger has been installed and configured correctly. Launch a Web browser and enter this URL:

```
http://WebServerHost.domain.com:port/cgi-bin/qps.sh/WService=
BrokerName/ping
```

- *WebServerHost* is the Web server host where the remote messenger was installed.

port is the port number used by Apache.

Note The port must only be specified for Apache if you are using a port other than the default, 80.

- *BrokerName* is your WebSpeed broker name.

The WebSpeed home page should display.

Open `SpeedStart` from the WebSpeed Program Group and follow the instructions to test the WebSpeed Messenger configuration.

