Installation Guide
QAD QXtend

Automated Installation, Upgrade, and Migration Appendix

78-0692E
QXtend 1.6.3
MFG/PRO eB, eB2, eB2.1, QAD 2007, QAD Enterprise Applications Standard and Enterprise Editions
March 2010

Qxtend_IG_v0613.pdf/biw/biw

QAD Inc.
100 Innovation Place
Santa Barbara, California 93108
Phone (805) 566-6000
http://www.qad.com

Contents

Chapter 1

System Overview

This section contains basic topics that you should understand before attempting a QAD QXtend installation or upgrade.

System Overview

The QAD QXtend interoperability framework provides a standardized data interface between QAD products and between QAD products and external systems.

The framework consists of two components:

- QAD QXtend Outbound (QXO) exports business objects as QDocs out of QAD products, such as QAD Enterprise Applications and QAD JIT Sequencing (JIT/S), to external subscribers.
- QAD QXtend Inbound (QXI) imports QDocs from external applications into QAD Enterprise Applications and other QAD applications such as JIT/S.

For details about configuring and implementing QAD QXtend, refer to *Technical Reference: QAD QXtend*.

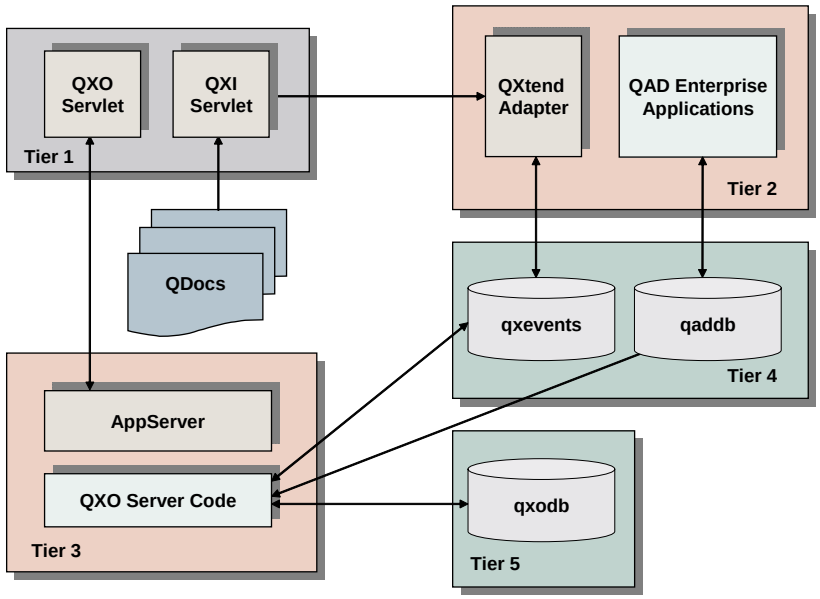
Deployment Overview

The QAD QXtend application comprises several components. These components may be deployed in various ways on different hosts, platforms, and architectures.

QXtend Deployment Tiers

Figure 1.1 shows the QXI deployment tiers and their relationship.

Fig. 1.1
QXI deployment Tiers



Note This deployment description uses tiers for explanatory purposes. While the grouping of components is significant, the numbering of the tiers is not.

The QXtend deployment consists of five tiers:

- Tier 1 consists of the QXI servlet and QXO servlet, which supports the QXO administration UI and contains the Tomcat application server.

  Note Both the Inbound and Outbound servlets can be installed on the same Tomcat server or on a different Tomcat server.
- Tier 2 consists of the QXtend adapter code and the QAD Enterprise Applications server. These two components must be installed on the same host. The QXO adapter code generates outbound documents through the QAD Enterprise Applications user interface or through direct calls to API programs. The QXO service interface adapter code specifically supports interoperability involving requests originating from the QAD EE Financials module. You may need to deploy Tier 2 multiple times (once for each QAD Enterprise Applications installation).
- Tier 3 consists of the Progress AppServer and the QXO server code (the server code handles data operations for the QXEvents, qaddb, and qxodb databases).
- Tier 4 consists of the QXEvents database and qaddb production database. These databases communicate with the QXO server code and should be installed on the same tier for data integrity reasons.
- Tier 5 consists of the qxodb database. This database contains data extracted by the event service from the QXEvents database.

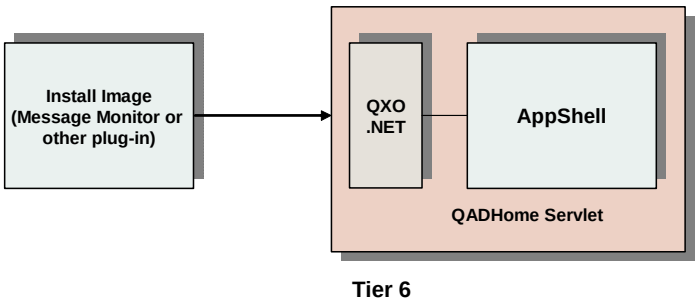
For a single-tier deployment, the sequence is as follows:

1. QXO servlet (tier 1)
2. qxodb database (tier 5)
3. QXEvents and qaddb databases (tier 4)
4. QXO adapter code on the QAD Enterprise Applications Outbound server (tier 2)
5. QXO server code on the Progress AppServer (tier 3)

QAD QXtend .NET AppShell Plug-in

Figure 1.2 shows the deployment of the QAD QXtend .NET AppShell plug-in.

Fig. 1.2
QAD QXtend .NET AppShell Plug-in



The QAD QXtend .NET AppShell plug-in deployment requires only one tier, consisting of the QXO .NET and the AppShell on the QADHome servlet. The AppShell supports various plug-ins designed to work with QAD applications—the QAD QXtend Message Monitor, for example. You must manually deploy this component.

For details about the .NET UI AppShell, refer to *User Guide: QAD User Interfaces*.

Deployment Options

The various components of QAD Enterprise Applications and QAD QXtend can be deployed in several configurations:

- On a single host in a unified deployment (single-tier deployment)
- On multiple hosts in a distributed deployment (multi-tier deployment)

In a single-host environment, all logical tiers are on the same host, and hence can be run at once. In a multiple-host environment, the logical tiers are separated physically by host. The installer must be run on each of the hosts involved in the installation.

QAD Deployment Configuration Service

When installing QAD QXtend, it is recommended that you use the QAD Deployment Configuration Service (QDCS) to facilitate the installation process.

Note Before starting an installation, collate your settings information and record it in the worksheet provided on page 14 for easy reference.

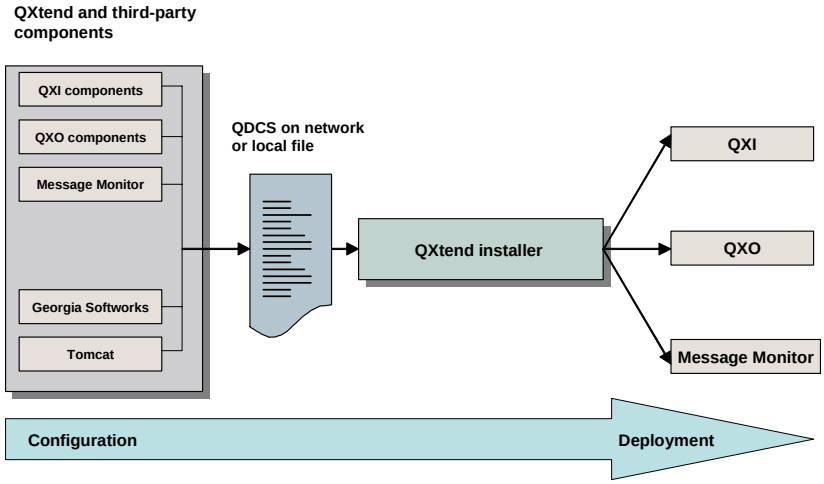
The QDCS is a repository that stores all your deployment settings for QAD QXtend and its supporting third-party applications (Tomcat, for example) in a single place. Typically, the QDCS is stored on a network for easy access, but it also can be stored in a portable file. The QDCS is populated using the GUI installer; therefore, access to a GUI environment is a prerequisite.

Note If you are installing in a UNIX environment and X-Windows is not available—or you are installing QAD QXtend in a character environment—you must first run the installer on Windows to collect the installation information.

There is only a single repository QDCS regardless of the number of environments and/or hosts you plan to use in your deployment. *You only have to enter your deployment settings once into the QDCS for a particular configuration*. The service can then be reused later to, for example, move a QAD QXtend deployment from a test environment into a production environment. See “QDCS Information Hierarchy” on page 5 for details about the structure of the QDCS.

Figure 1.3 illustrates how the QDCS works. The diagram assumes that all the QAD QXtend components and supporting third-party applications are being installed on the same host.

Fig. 1.3
QDCS Information Flow



During an installation using the QDCS:

1. Specify the location of the QDCS: network or local host.
2. Specify the name of the configuration filename and the name of the environment for the installation.
3. For each module you select to install—QXI adapter code, QXI servlet, for example— provide the deployment configuration settings required by the installer; these settings are stored in the QDCS.
4. After you provide all settings for the selected components, review the installation summary before you run the installation process. The information stored in the QDCS guides the automated installation.

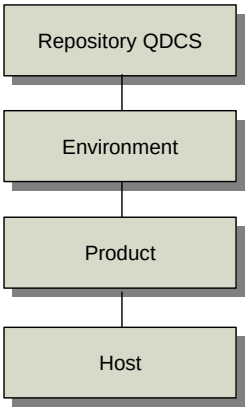
Installing QAD QXtend using the QDCS has the following advantages:

- For QAD QXtend components that share installation parameters—the location of the Progress AppServer, for example—the QDCS allows these settings to be passed between hosts, reducing the potential for error when entering configuration information.
- The QDCS preserves deployment data across sessions. If the installation fails, you can resume the installation from the point where it stopped without re-entering settings.
- Reinstallations need no further input.

QDCS Information Hierarchy

The QDCS stores information in XML format in a hierarchy, as shown in Figure 1.4.

Fig. 1.4
QDCS Information Hierarchy



The repository QDCS hierarchy consists of the following elements:

- The repository QDCS element is at the top of the hierarchy. There is only one repository, regardless of the number of subordinate environments, products, and hosts.
- Typically, organizations have more than one environment. For example, your organization may have a test environment for verifying deployments, and a production environment that accommodates the “live” system. The position of the environment element at the secondary level in the QDCS hierarchy enables product and host deployments to be moved easily between environments.
- In the current release, there is only one product element—for the QAD QXtend product. In the future, it is envisioned that the QDCS will store deployment settings for all QAD applications in your environment.
- Typically, there will be many host elements, enabling deployment of QAD QXtend in various configurations. Each host typically contains one or more QAD QXtend components determined by component dependencies and organizational deployment requirements.

Installation Overview

Installing and configuring QAD QXtend requires several prerequisite programs. This guide describes the installation and configuration of the QAD QXtend application on these programs.

Note See Chapter 2, “System Requirements,” on page 7 for additional information on requirements.

The steps are:

1. Install prerequisite components.
2. Complete the QDCS worksheet (optional).
3. Install QAD QXtend using the installer.

Chapter 2

System Requirements

This section contains component and version information for a QAD QXtend environment.

Overview

This section provides the software, client, operating system, and other requirements for QAD QXtend.

Note For the most current requirements information, refer to the Product Availability Guide on the QAD Online Support Center at:

http://support.qad.com

Software Requirements

This section describes the prerequisite requirements to install, configure, and use QAD QXtend.

QXO Server Requirements

The following software must be present on the QXO server at the time of installation:

- A complete Progress OpenEdge 10 (minimum version 10.1C03 or higher) installation including the following components:
  - Enterprise RDBMS
  - OpenEdge Application SVR Enterprise
  - Client Networking
  - OpenEdge Studio
- Java Software Development Kit, J2SDK, version 1.5x

QXI Server Requirements

QXI requires prerequisite components and includes optional installations on the product media for other required products. A complete QXI installation requires:

- Progress OpenEdge 9.1E and above components including:
  - If your QAD Enterprise Applications installation is for a language other than US English, ensure you also install US English from the Progress media. The US English version of PROMSGS is needed for the telnet connection scripts.
  - Latest Progress version-specific patches. For details, see the QAD Enterprise Applications installation guide for your system.
- QAD Enterprise Applications (MFG/PRO)
  - eB, SP4+
  - eB2, all versions
  - eB2.1 (QAD Enterprise Applications 2007), all versions
  - QAD Enterprise Applications Standard Edition
  - QAD Enterprise Applications Enterprise Edition

Third-Party Components

- Additional Progress Components

  In addition to the Progress software required for QAD Enterprise Applications, QXI requires the Progress AppServer, NameServer, and AdminServer if you use the code APIs.

- Apache Tomcat Application Server 5.5.20 or higher

  This is available from:

  http://tomcat.apache.org/

- Java 1.5.0 JDK (for Tomcat 5)

  For Linux, Sun, and Windows versions:

  http://java.sun.com

  For Hewlett-Packard systems:

  http://www.hp.com/products1/unix/java/index.html

  For AIX systems:

  http://www.ibm.com/developerworks/java/jdk/aix/service.html

- Telnet server

  QXI uses a telnet server to run a pool of telnet sessions for communicating between QXI and QAD Enterprise Applications. On UNIX machines, you can use the default telnet service provided with the operating system.

  For Windows systems, use:

  Georgia Softworks Telnet Server 6.50 +

  This is available from:

  http://www.georgiasoftworks.com/

Supporting Technologies

QAD QXtend incorporates various Web-based technologies to support its features. These technologies are included with the product and are transparent to you. They are listed here to give credit to the open-source projects that created them.

- Struts is an open source framework for building Web applications, part of the Jakarta Project, sponsored by the Apache Software Foundation.

  http://struts.apache.org/index.html

- Apache AXIS is an implementation of the SOAP (Simple Object Access Protocol) submission to W3C.

  http://ws.apache.org/axis/

- All QDoc requests and responses are logged using Log4j from Apache, a reliable, fast, and flexible logging framework for Java.

  http://logging.apache.org/log4j/docs/index.html

Client Requirements

QAD QXtend client systems are browser based and require only Internet Explorer, version 6.0+

Operating Systems

The QXI and QXO servers support the following platforms:

- Linux (Redhat and SuSE)
- HP-UX
- HP-Tru64
- Sun Solaris (SPARC)
- Compaq UNIX (Tru64)
- IBM AIX
- Windows (including Windows 2000, 2002 Server, 2003 Server, XP, and 64-bit variations of these platforms)

Installation User Account

The user account(s) used to install QXtend and the start environment must be carefully selected to avoid potential access permission problems. An easy and effective way to prevent permission issues is to use an administrator (rather than root) account to perform the entire installation. This includes starting Tomcat, installing QXtend, performing post-install activities, and starting the whole environment (QAD Enterprise Applications and QXtend).

Chapter 3

Prerequisites

This section describes the prerequisites that must be performed before QAD QXtend installation.

Overview

Before installing QAD QXtend:

1. Install the Windows Telnet Server (Windows only).
2. Install the Tomcat Application Server.
3. Complete the recommended QDCS Worksheet.

Install the Windows Telnet Server

For telnet connections on Windows servers, install the Georgia Softworks Telnet Server (GSWTS). The latest version can be downloaded from the Georgia Softworks Web site at:

```
www.georgiasoftworks.com/downloads.htm
```

Install the Telnet Server

1. Unzip the downloaded `.zip` file to a work directory.
2. In the work directory, double-click `setup.exe` to begin installation.
3. At the Welcome window, click Continue.
4. Select Full Install and click Continue.
5. Enter the GSWTS installation directory and click Continue.
6. Use the default `\gs_uts` installation directory. You can place this directory on any network drive.
7. When GSWTS is installed, the Setup Succeeded window displays and GSWTS starts automatically.

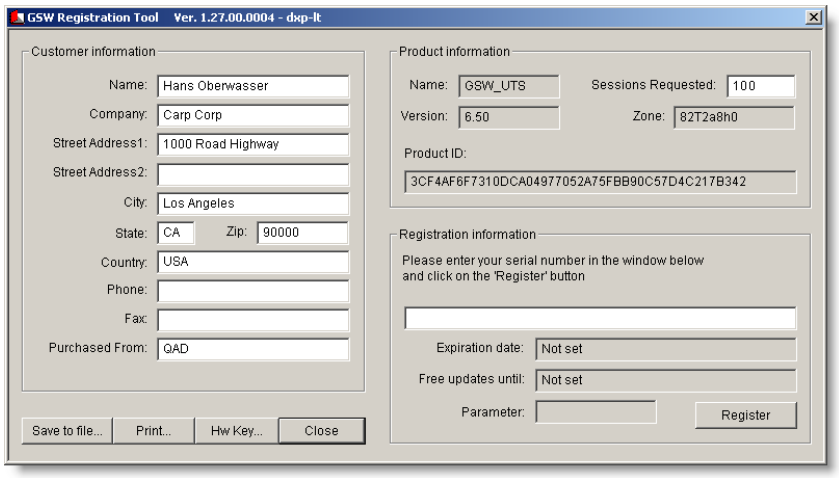
Register the Georgia SoftWorks Software

To register GSWTS, provide a product ID to Georgia SoftWorks so that a serial number can be generated for your product. The serial number identifies server hardware and software components. If these components change or are upgraded, contact Georgia SoftWorks about generating a new product ID and serial number.

Important If you must reinstall or are planning to move your installation to a different platform—or if you are a sales agent or a distributor—include that information on the registration.

1. Select the Registration icon from the GSWTS program group in the Start menu.
2. In the Georgia SoftWorks Product Registration window, enter your customer information. The information that displays in the Product Information section is system-generated.
3. Set Sessions Requested to 100. This is the number QAD automatically supplies with your registration.

Fig. 3.1
Georgia Softworks Registration



4 Choose Save to File to save this information or choose Print. Then, follow the appropriate step to supply the product ID to Georgia SoftWorks:

   a E-mail the saved registration form file to Georgia SoftWorks at:

   `registration@georgiasoftworks.com`

   When your form is received, a serial number is generated for your product and is returned to you by e-mail.

   b Fax the printed registration form to Georgia SoftWorks at 706-265-1020. When your form is received, a serial number is generated for your product and returned to you by fax.

5 When you receive your serial number, return to the Georgia SoftWorks Product Registration window and enter it in the appropriate field in the registration form. Click Register.

Configure Georgia Softworks

The Georgia Softworks start-up batch file, `k_start.bat`, must be created manually. In addition, if the QAD Desktop and QXI reside on the same host and use the same instance of Georgia Softworks, two `k_start.bat` files must exist, each identified by a separate user directory under the `gs_uts\scripts` directory.

For this second case, there are multiple methods for setting this up. Refer to the Georgia Softworks documentation that downloads with the product in the `\docs` directory for details.

You must manually create the batch file to launch the Georgia telnet server. Open a text editor and enter the following:

```
@echo off
set gwtn_term=1
set gwtn_graphics=3
set gwtn_color=0
set LRA_TERMINATION=SIG-CTRL-C,SLEEP5,SIG-CTRL-C,SLEEP5,F4,
SLEEP5,e,x,i,t,ENTER,SLEEP5,e,x,i,t,ENTER,SLEEP5,e,x,i,t,ENTER
C:
set prompt=$G
```

Make sure the drive designation, shown in the file sample above as `C:`, is the drive where you have installed Georgia Softworks.

Save the file as `k_start.bat` in the Georgia Softworks scripts directory, *`GSWTSHome`*`\scripts`. For example, `c:\GS_UTS\scripts`.

Install the Tomcat Application Server

Install the Tomcat application server using the installation instructions provided in the *Tomcat User Guide* on the Apache Tomcat Web site.

QDCS Worksheet

Use the worksheet in Table 3.1 to record installation-related information such as server names and locations, port numbers, and other settings. You enter these parameter settings into the QDCS; for details, see “Installing, Upgrading, and Migrating QAD Qxtend in a GUI Environment” on page 21.

Note Use of this worksheet is recommended. QAD suggests that you research this information before starting your installation.

Table 3.1
QDCS Worksheet

Group	Description	Setting
Tomcat	Tomcat home	
	Tomcat host	
	Tomcat admin user	
	Tomcat admin password	
	Tomcat port	
	Local parameter file name (if not using QDCS)	
Georgia Softworks	Georgia Softworks directory	
MFG/PRO	MFG/PRO home	
	MFG/PRO version	
	MFG/PRO SP	
	MFG/PRO PROPATH	
	MFG/PRO empty database directory	
	MFG/PRO empty database	
	MFG admin empty database	
	QXtend compile database set name	
	Progress for MFG/PRO server	
	Additional connection parameters	
	QDT directory (if applicable)	
MFG/PRO Database	MFG/PRO database physical name	
	MFG/PRO database directory	
	MFG/PRO database host	
	MFG/PRO database service name	

Table 3.1 — *QDCS Worksheet* ((Page 1 of 3)

Group	Description	Setting
MFG/PRO Admin Database	MFG/PRO admin database physical name	
	MFG/PRO admin database directory	
	MFG/PRO admin database host	
	MFG/PRO admin database service name	
QXEvents Database	QXEvents database physical name	
	QXEvents database directory	
	QXEvents database host name	
	QXEvents database service name	
	QXEvents database service port	
	Progress directory for MFG/PRO databases	
	QXEvents database block size	
Migrate QXEvents	Source QXEvents database name	
	Source QXEvents connection type	
	Source QXEvents directory	
	Source QXEvents host	
	Source QXEvents service name	
	Source QXEvents version	
Outbound Database	QXODB destination directory	
	Progress AppServer name	
	Progress AppServer port	
	Progress AppServer max. number of connections	
	QXODB physical name	
	QXODB service name	
	QXODB host name	
	.NET UI home directory	
	Progress for Outbound database	
Outbound Database (advanced parameters)	QXODB Block Size	
	Before image	
	Schema area	
	Configuration data	
	Configuration indexes	
	Transactional data	
	Transactional indexes	
	Raw msgs (CLOB)	
	Subscriber Msgs (CLOB)	

Table 3.1 — *QDCS Worksheet* ((Page 2 of 3)

Group	Description	Setting
Migrate QXODB	Source QXODB database name	
	Source QXODB connection type	
	Source QXODB directory	
	Source QXODB host	
	Source QXODB service name	
	Source QXODB version	
Inbound Adapter	Inbound adapter code directory	
Inbound Web Application	Inbound WebApp name	
Outbound Adapter	Outbound Adapter code	
Outbound Web Application	Outbound WebApp name	
Outbound Server	Outbound Appserver code	
	Progress for Outbound server	
	Outbound Appserver max connections	
	Outbound Appserver port	
	Outbound Nameserver	
	Outbound Appserver name	
	MFG AppShell source	
	SI AppServer name	
	SI AppServer port	
	SI AppServer max connections	
	SI AppServer NameServer	
Log	Install log	
	WebApp deployment directory (generated)	
	Use Tomcat live deployment	
	AppShell component deployment	
	QDCS host	
	QDCS port	
	QDCS Webapp name	
	Deploy QDCS	
	Use QDCS? (Y/N)	
	QDCS user	
	QDCS password	
	QDCS local file	

Table 3.1 — *QDCS Worksheet* ((Page 3 of 3)

Section 1

Automated Installation, Upgrade, and Migration

This section describes how to perform an automated QAD QXtend install, upgrade, or migration. It also describes post-installation and post-upgrade tasks and conversions, as well as troubleshooting information.

Chapter 4

Installing, Upgrading, and Migrating QAD QXtend

This section describes how to install, upgrade, and migrate QAD QXtend in a GUI or character environment.

Overview

The QAD QXtend installation disk contains the QAD QXtend installer. The installer supports both GUI and character-based installations or upgrades.

The following are the options for installing, upgrading, and migrating QAD QXtend:

- Install, upgrade, or migrate in a GUI environment using the QDCS.
- Install or upgrade in a character environment using the QDCS. You first run the installer in a GUI environment to populate the QDCS with the required settings. See “Installing, Upgrading, or Migrating QAD Qxtend in a Character Environment” on page 45.

Note You must have completed a successful installation before attempting to migrate the QXI servlet, QXO database, or QXO events database components.

The QXI adapter, QXO adapter, QXO servlet, and QXO server are code-based. There are no upgrade routines for these and they must be installed in place of the old version.

Installer Limitations

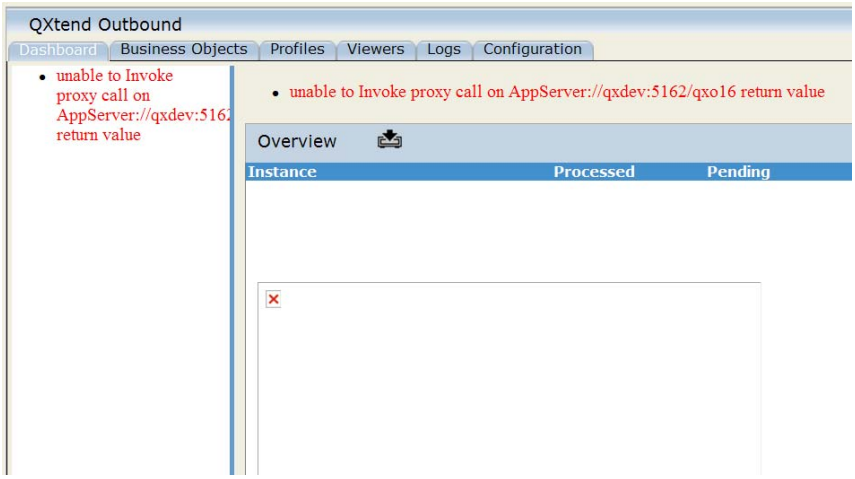
Installation or upgrade does not update startup scripts for QAD Enterprise Applications. You must update the scripts manually to include the locations of the newly installed components. For example, the new `qxevents` database must be included in the database set for QAD Enterprise Applications.

When migrating the QXO database configuration, the `qxevents` connection details are not automatically updated in the source application to point to the 1.6 `qxevents` database installation. Before attempting to create any events, verify and update the connection details for each source application in QXtend Outbound.

Depending on the connection type defined for the `qxevents` database, the installer may be unable to connect to it. For example, if the connection specified is client-server and the server is down, the installer will not be able to connect to it at upgrade time.

If a `qxevents` connection problem occurs, the Dashboard in the QXO UI will fail to display and look something like Figure 4.1.

Fig. 4.1
QXEvents Connection Error Indication



Also these messages will appear in the AppServer log for the QXO UI:

```
[09/04/02@14:52:42.296+1100] P-020654 T-000000 1 AS -- (Procedure: 'launchProcForSrcApps
connector.p' Line:638) ** CRC for ert_app_event does not match CRC in pl-msgconsol.p.
Try recompiling. (1896)

[09/04/02@14:52:42.296+1100] P-020654 T-000000 1 AS -- (Procedure: 'launchProcForSrcApps
connector.p' Line:638) Unable to locate index 'XPKevt_cnt' on table 'ert_evt_cnt' for
procedure 'pl-msgconsol.p'. (7967)
```

However, you will be able to navigate to the Configuration tab and update the database connection details for each of the source applications.

Workarounds exist for this issue. If you experience difficulties with your installation, see Chapter 6, “Troubleshooting QAD QXtend Installs and Upgrades,” on page 57. Also, review the install log file; see Appendix A, “Installation Log,” on page 63. If you still experience problems, contact QAD Support for assistance.

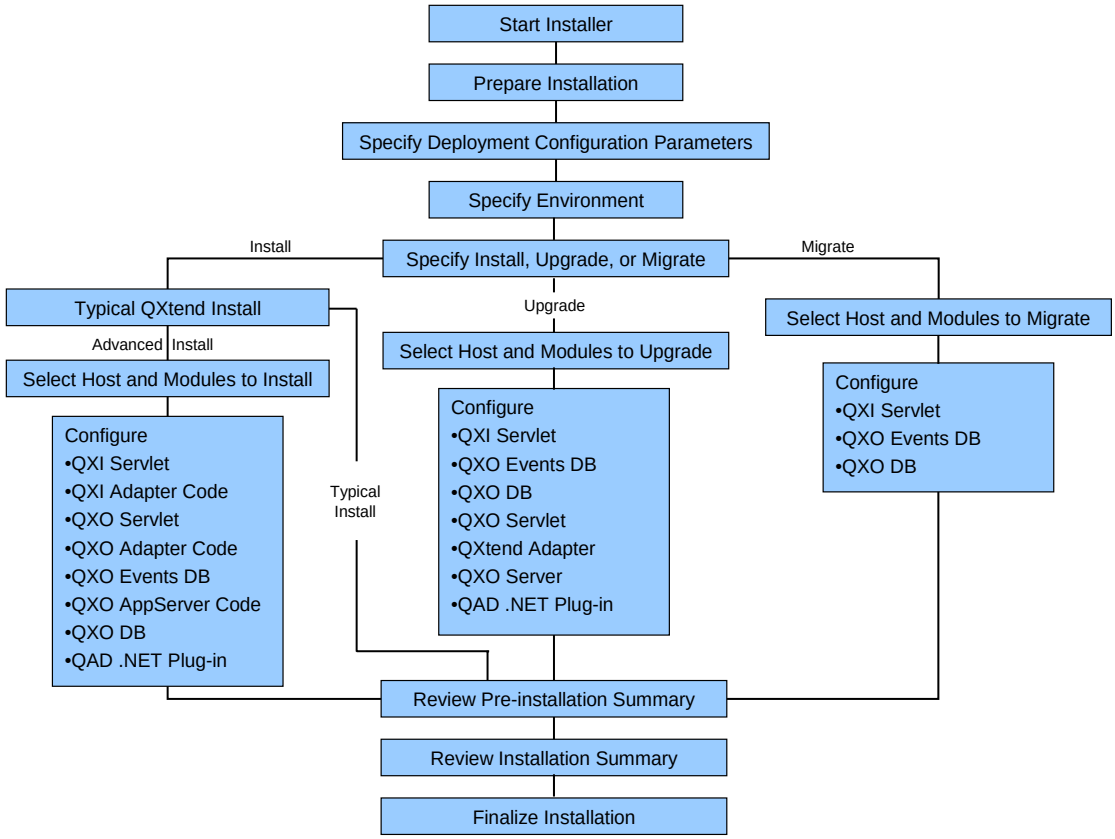
Installing, Upgrading, and Migrating QAD Qxtend in a GUI Environment

This section describes installing, upgrading, and migrating QAD QXtend by using the QDCS on the same host in a Windows GUI environment.

Figure 4.2 summarizes the QAD QXtend installation, upgrade, and migration workflows.

Note The sequence below applies when all components are installed or upgraded on a single host. The sequence of deployment may vary, depending upon your configuration.

Fig. 4.2
QXtend Install, Upgrade, and Migration Workflows



Before installing or upgrading QAD QXtend, ensure that Tomcat is running. You should also ensure that you have the appropriate folder permissions to perform the install, upgrade, or migration.

Note In a UNIX environment, before launching the GUI installer from an X-Windows session, you must first set the DISPLAY variable using the following command:

```
export DISPLAY=HOST_NAME:0.0
```

For example:

```
export DISPLAY=plli13:0.0
```

Start the Installer

1. Insert the CD-ROM into the CD-ROM drive or mount the CD image on your file system.
2. The executable files for each environment type can be found under `Disk1/InstData/<env type>/[No]VM/QXtend.[bin|exe]`.

   Note If you choose to copy the files from the CD, make sure the directory structure remains the same, including the folder called `Disk 1`. If this is changed, the installer will not run. The directory structure should be as follows:

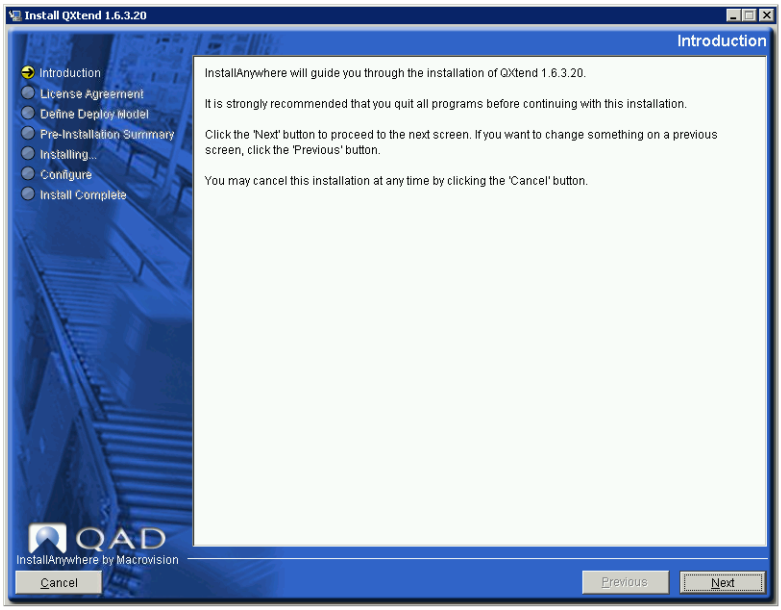
```
<CD MEDIA>
    +Disk1
        +InstData
            -Resource1.zip
            -MediaId.properties
            +{environment type}
                +NoVM
                    QXtend.[bin|exe]
                +VM
                    QXtend.[bin|exe]
```

Prepare the Installation

3 If you have downloaded the installer, double-click `QXtend.exe`.

The installer is extracted and the Introduction screen displays.

Fig. 4.3
Introduction Screen



4 Click Next. The License Agreement screen displays.

5 Scroll to the end of the license agreement.

6 Select the “I accept the terms of the License Agreement” option, then click Next.

Note The option to accept the license agreement is enabled only when you scroll to the bottom of the agreement.

The Log File Directory screen displays.

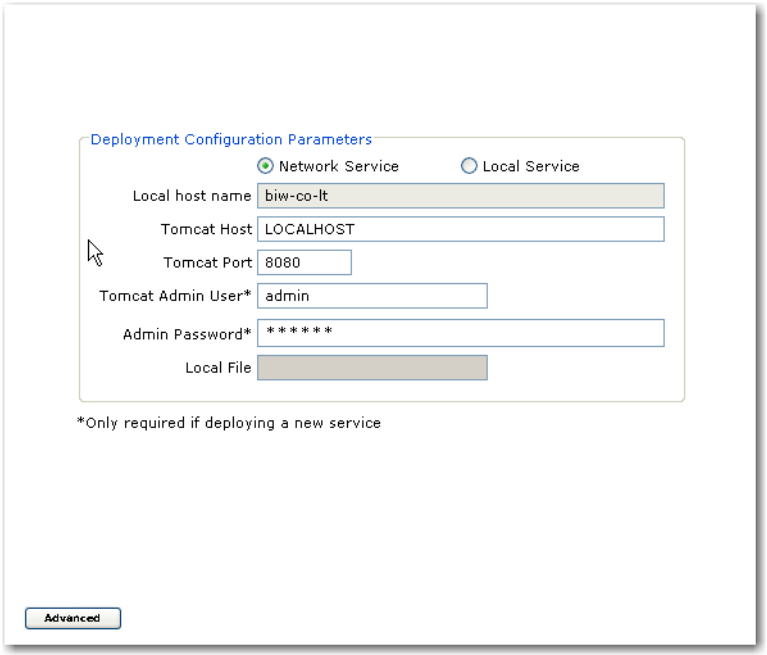
7 Accept the default location for the install log files (`C:\instlog`), or enter a different path.

8 Click Next. The QAD Deployment Configuration Service screen displays.

Specify Deployment Configuration Parameters

For details about the QDCS, see “QAD Deployment Configuration Service” on page 4.

Fig. 4.4
Specify Deployment Configuration Parameters



Network Service. Select this option to use a QDCS located on your network.

Note Tomcat must be installed and running to use Network Service. If you have an existing deployment service, you must create a new service because the version changed in the QXtend 1.6.2 release.

Local Service. Select this option to use a QDCS file located on your local machine. If you select this option, all fields are disabled, except Local File.

Local Host Name. Displays the name of the current machine (read-only).

Tomcat Host. Enter the name of the Tomcat server (for Network Service only).

Tomcat Port. Enter the number of the Tomcat port (for Network Service only).

Tomcat Admin User. Enter the user name for the Tomcat manager role (for Network Service only).

Admin Password. Enter the password of the user with the manager role (for Network Service only).

Local File. Enter the filename on the local machine that contains the parameter settings (for Local Service only).

Advanced. Click this button to display a dialog that allows you to specify the name of a new QDCS.

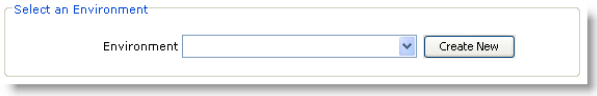
9 A pop-up window may display that says a QAD Deployment Configuration Service (QDCS) was not detected on a host. This means the QDCS is not deployed on the Tomcat server just specified or the wrong server was specified. Click Yes if you want the installer to deploy the QDCS.

10 Click Next. The Environment Selection screen displays.

Specify Environment

11 In the Select an Environment panel, specify the environment to use. To create a new environment, click Create New and enter the name of the environment you want to create.

Fig. 4.5
Select an Environment



12 Click Next. The Select an Installation Option screen displays.

Specify Install, Upgrade, or Migrate

Fig. 4.6
Specify Install, Upgrade, or Migrate

To perform an installation, verify Install is selected, click Next, and continue to “Installing QXtend” on page 25.

To perform an upgrade, select Upgrade, click Next, and go to “Upgrading QXtend” on page 35.

To perform a migration, refer to “Migrating QXtend” on page 39.

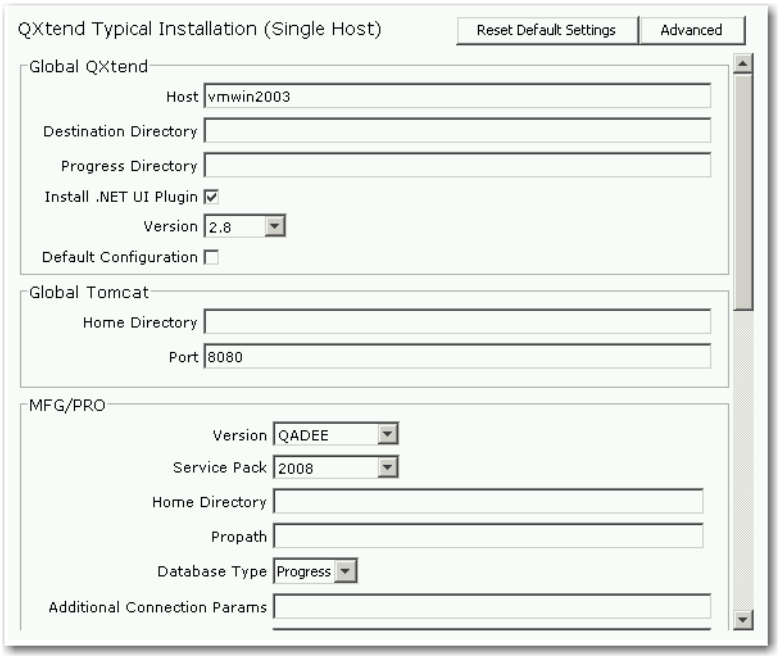
Note You must have completed a successful installation before attempting to upgrade the QXI servlet, QXO database, or QXO events database components.

Installing QXtend

QXtend Installation Settings

QXtend installation involves the automatic or manual population of fields on one or more screens. During installation, hovering over a field label displays the corresponding parameter name. Likewise, hovering over the field displays the fully resolved parameter as a tool tip. For more information on dynamic parameter referencing, refer to “Tokens” on page 72.

Fig. 4.7
QXtend Installation Settings



1 Enter the necessary information in the QXtend Typical Installation (Single Host) screen.

2 QXtend provides two installation methods: Typical and Advanced.

Typical Installation is a simplified install process that automatically fills the QXtend installer fields with parameters from a default QXtend installation. For detailed information regarding default installation parameters, refer to Appendix B, “Typical Installation Parameters,” on page 71.

Note If you check the Default Configuration box, QXtend will load with an initial set of default configuration parameters. Refer to Appendix C, “Default Configuration,” on page 77 for more information.

Advanced Installation allows you to enter the parameters individually to create a custom QXtend configuration. If you want to modify any of the default configuration parameters, use QXtend’s advanced installation mode.

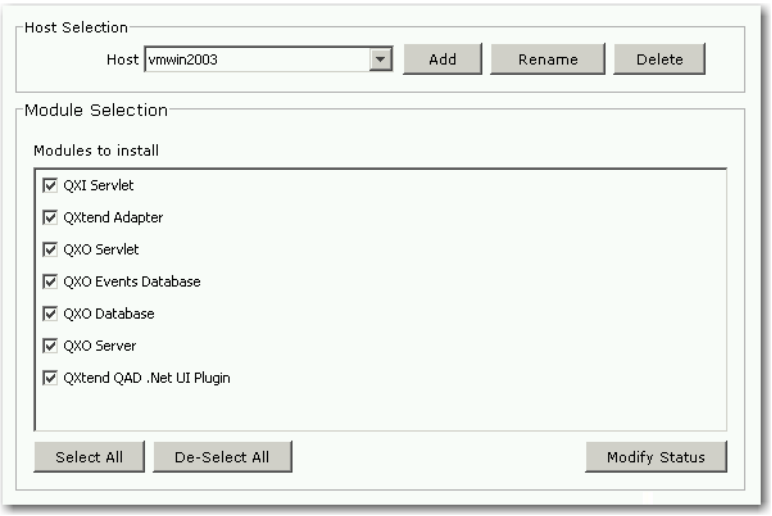
To continue with a typical installation, click Next and go to “Review the Pre-installation Summary” on page 41.

To perform a custom installation, click Advanced and go to “Advanced Installation” on page 27.

Advanced Installation

Select the Host and Modules to Install

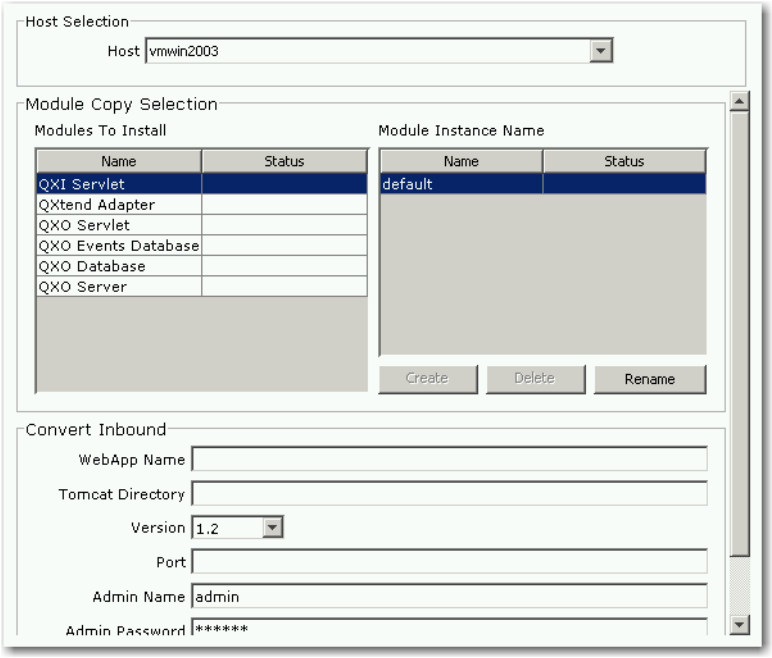
Fig. 4.8
Choose the Host and Install Set



1. In the Host Selection panel, enter the host on which to install the components.
2. Use the Add and Delete buttons to create and delete hosts as required.
3. In the Module Selection panel, select the components to install on the specified host.

   Note To install QXI only, select both the QXI servlet and adapter code components. To install QXO only, select all the QXO components and the adapter code component.
4. Click Next to display the Module Copy Selection screen.

Fig. 4.9
Module Copy Selection



The component list on the left shows the components available for installation. The Status column to the right indicates the installation status for each component. The first time the installer is run, the Status column is blank. After an installation, the Status column shows Incomplete—indicating the component has not been installed successfully—or Complete, indicating the component was installed successfully.

5 For each selected component, do the following as required:

a Create, rename, or delete instances.

Note New instances are enabled only for the QXtend Adapter and QXO Events database.

b Assign the required components to an instance by selecting the check box for a component when the instance in which it should be included is highlighted.

c For each selected component for that instance, enter all the required parameter settings, as described in the following sections.

Note You cannot select a different component until you have finished entering all the required configuration information for the currently selected component. Fields that must be completed are highlighted in orange.

Configure the QXI Servlet

6 In the Inbound Web Application panel, enter the name of the application as known by the Tomcat application server.

7 In the Tomcat for Inbound panel, complete the required fields.

Port. Enter the Tomcat port.

Home Directory. Enter the location of the home directory where Tomcat is installed.

Admin Name. Enter the user name for the Tomcat manager role.

Admin Password. Enter the password of the user with the manager role.

8 In the Migrate Inbound Configuration panel, make the necessary selections and complete the field.

Enable. Select the check box to enable migration of QXI servlet configuration data from a previously installed QXI servlet.

WebApp Name. Enter the name of the application as known by the Tomcat application server.

Tomcat Directory. Enter the location of the home directory where Tomcat is installed.

Version. Select the version being migrated from the drop-down list.

9 If necessary, select the next component to configure from the Modules to Install panel. Otherwise, go to step 34 on page 35.

Configure the QXtend Adapter Code

10 In the MFG/PRO panel, complete the required fields.

Version. Select the version of QAD Enterprise Applications from the drop-down list which the adapter is being installed against.

Service Pack. Enter the number of the service pack in use.

Home Directory. Enter the home directory where the QXtend adapter is installed.

Propath. If not populated automatically, enter the PROPATH used by the QXtend Adapter clients.

Database Type. Select Progress or Oracle.

Empty DB Directory. If not populated automatically, enter the directory where the QXtend adapter empty databases are located.

Empty DB. Enter the name of the empty QXtend adapter database.

Admin Empty DB. Enter the name of the empty QXtend adapter admin database.

Additional Connections Params. Enter any extra parameters required for compiling the QXtend adapter.

Progress Directory. Enter the location of Progress that this instance of the QXtend adapter uses. This should be the same version of Progress as the QAD Enterprise Application.

QDT Directory. Enter the location of the QDT. This is currently only applicable if installing against QAD Enterprise Edition.

Financial AppServer. Enter the name of the AppServer for Financials. This is currently only applicable if installing against QAD Enterprise Edition.

Financial NameServer Port. Enter the NameServer port that the Financials AppServer is on. This is currently only applicable if installing against QAD Enterprise Edition.

11 In the QXtend Adapter panel, complete the required fields.

Destination Directory. Enter the directory where the QXtend adapter is located

Client Startup Params. Enter the client startup parameters required for starting a character session in your QAD Enterprise Application.

12 In the Adapter Native API AppServer panel, complete the required fields.

Note This is only applicable to eB2.1 SP4 and later.

AppServer Name. Enter the name of the AppServer for using Native APIs.

AppServer Port. Enter the port number that the AppServer will use.

AppServer Max Connections. Enter the number of maximum connections to the AppServer.

Name Server. Enter the Name Server name to which the AppServer will belong.

Name Server Port. Enter the Name Server port that the AppServer will use.

Update Ubroker Properties. Check this box for the installer to update the Ubroker Properties. If you do not check the box, an example set of properties will be deployed to the component.

13 If necessary, select the next component to configure from the Modules to Install panel. Otherwise, go to step 34 on page 35.

Configure the QXO Servlet

14 In the Outbound Web Application panel, complete the required fields.

WebApp Name. Enter the name of the QXO servlet application as known by the Tomcat application server.

AppServer Name. Enter the name of the Progress AppServer.

AppServer Host. Enter the name of the host on which the AppServer is located.

Name Server Port. Enter the port number used by the Progress name server.

15 In the Tomcat for Outbound panel, complete the required fields. These values should be populated automatically if QXO will use the same AppServer configuration.

Port. Enter the number of the Tomcat port.

Home Directory. Enter the location of the home directory where Tomcat is installed.

Admin Name. Enter the user name for the Tomcat manager role.

Admin Password. Enter the password of the user with the manager role.

16 If necessary, select the next component to configure from the Modules to Install panel. Otherwise, go to step 34 on page 35.

Configure the QXO Events Database

17 In the MFG/PRO Version panel, complete the required fields.

Version. Enter the version of the QAD Enterprise Application the adapter is being installed against.

Service Pack. Enter the number of the service pack in use.

18 In the MFG/PRO Database panel, complete the required fields.

Physical Name. Enter the physical name of the qaddb for QAD Enterprise Applications.

Connection Type. Select either Local or Network.

Multi-user. Select this option to indicate that your database is operating in multi-user mode. Using this option means that you may not have to shut down your database before installing QAD QXtend. However, it is recommended that you do shut down your `qxevents` database.

Database Directory. This is the database where the QAD Enterprise Edition admin database is installed. Enter the directory if the connection type is Local.

Host. Enter the name of the host where the QAD Enterprise Edition database is installed.

Service Name. Enter the network service used by the QAD Enterprise Edition admin database

19 In the MFG/PRO Admin Database panel, complete the required fields.

Physical Name. Enter the physical name of the QAD Enterprise Edition admin database.

Connection Type. Select either Local or Network.

Multi-user. Select this option to indicate that your database is operating in multi-user mode.

If you use this option, the database must be up and running when installing QXtend. If you do not use this option, the database must be shut down before installing QXtend.

Database Directory. Enter the name of the database directory where the QAD Enterprise Edition admin database is installed.

Host. Enter the name of the host where the QAD Enterprise Edition admin database is installed.

Service Name. Enter the network service used by the QAD Enterprise Edition admin database.

20 In the QXEvents Database panel, complete the required fields.

Physical Name. Enter the physical name of the QXEvents database.

Database Directory. Enter the directory where the QXEvents database is located.

Use Client/Server. Select this check box to use client/server mode when connecting to the database.

Host. Enter the name of the host where the QXEvents database is installed.

Service Name. Entered into the services file by the installer. Otherwise, enter the name of the network service used by this database.

Create Services File Entry. Select this check box to enable the Service Port field and to have the port information added to the service.

Service Port. Entered into the services file by the installer. Otherwise, enter the number of the port used by this network service.

Block Size. Enter the block size to use for the qxevents database. This is dependent on the operating system block size.

Progress Directory. Enter the location of Progress used by the Progress AppServer.

21 If you are migrating an existing QXEvents database to the new install, complete the following required fields in the QXEvents Migrate Configuration panel (refer to the “Upgrading QXtend” on page 35 for more details). These fields may be populated already.

Enable. Select the check box to enable migration of QXEvents configuration data from a previously installed qxevents database.

Physical Name. Enter the physical name of the source qxevents database.

Connection Type. Enter the connection type to use to connect to the source QXEvents database.

Multi-user. If the source QXEvents Connection Type is Local, specify whether or not to connect in multi-user mode.

If you use this option, the database must be up and running when installing QXtend. If you do not use this option, the database must be shut down before installing QXtend.

Database Directory. Enter the directory for the source qxevents database. This is only applicable when the Connection Type is Local.

Host. Enter the name of the host where the source qxevents database is installed.

Service Name. If the Connection Type is Network, specify the service name or service port number to use.

Version. Select the version being migrated in the drop-down list.

22 If necessary, select the next component to configure from the Modules to Install panel. Otherwise, go to step 34 on page 35.

Configure the QXO Database

23 In the Outbound Database panel, complete the required fields.

Physical Name. Enter the physical name of the QXO database.

Database Directory. Enter the location where the QXO database is located.

Use Client/Server. Select this check box to use client/server mode when connecting to the database.

Host. Enter the name of the host on which the QXO database is installed.

Service Name. Entered into the services file by the installer. Otherwise, enter the name of the network service for this database.

Create Services File Entry. Select this check box to enable the Service Port field and to have the port information added to the service.

Service Port. Entered into the services file by the installer. Otherwise, enter the port number for the network service.

Block Size. Enter the block size to use for the QXO database.

Progress Directory. Enter the location of the Progress directory used by this database.

Advanced Database Config. Selecting this box displays additional fields that allow you to specify the locations of the schema areas for the QXO database. Placing these areas on different disks can improve performance.

The fields displayed when this box is selected are as follows:

Before Image. This is the location of the BI file.

Schema Area. There are no tables in this area.

Configuration Data. This is a low volume and low capacity area.

Configuration Indexes. This is a low volume and low capacity area.

Transactional Data. This is a high volume, medium capacity area.

Transactional Indexes. This is a high volume, medium capacity area.

Raw Msgs (CLOB). This is a high volume and high capacity area.

Subscriber Msgs (CLOB). This is a high volume and high capacity area.

24 In the Migrate QXODB Configuration panel, complete the required fields.

Enable. Select the check box to enable migration of QXO database configuration data from a previously installed QXO database.

Physical Name. Enter the physical name of the source QXODB database.

Connection Type. Enter the connection type to use to connect to the source QXODB database.

Multi-user. If the source QXODB Connection Type is Local, specify whether to connect in multi-user mode.

If you use this option, the database must be up and running when installing QXtend. If you do not use this option, the database must be shut down before installing QXtend.

Database Directory. Enter the directory for the source QXODB database. This is only applicable when the Connection Type is Local.

Host. Enter the name of the host where the source QXODB database is installed.

Service Name. If Connection Type is Network, specify the service name or service port number to use.

Version. Select the version being migrated from the drop-down list.

25 If necessary, select the next component to configure from the Modules to Install panel. Otherwise, go to step 34 on page 35.

Configure the QXO Server

26 In the Outbound Server panel, complete the required fields.

Destination Directory. Enter the directory where the qxo server is located.

Outbound Database Name. Enter the name of the Outbound database name.

Use Client/Server. Select this check box to use client/server mode when connecting to the database.

Outbound Database Directory. Enter the location of the Outbound database directory.

Outbound Database Host. Enter the name of the host of the Outbound database.

Outbound Database Service. Enter the name of the service for the Outbound database.

.NET UI Home Directory. Enter the location of the .NET Progress code.

Progress Directory. Enter the location of Progress used by the Progress AppServer.

27 In the Outbound UI AppServer panel, complete the required fields.

AppServer Name. Enter the name of the Progress AppServer.

AppServer Port. Enter the port number of the Progress AppServer.

AppServer Max Connections. Enter the maximum number of connections allowed by this AppServer.

Name Server. Enter the name of the name server.

Update Ubroker Properties. Check this box if you would like the installer to update the Ubroker Properties. If you do not, an example set of properties will be deployed to the component.

28 In the Outbound SI AppServer panel, complete the required fields.

AppServer Name. Enter the name of the Progress AppServer.

AppServer Port. Enter the port number of the Progress AppServer.

AppServer Max Connections. Enter the maximum number of connections allowed by this AppServer.

Name Server. Enter the name of the name server.

Update Ubroker Properties. Check this box if you would like the installer to update the Ubroker Properties. If you do not, an example set of properties will be deployed to the component.

29 If necessary, select the next component to configure from the Modules to Install panel. Otherwise, go to step 34 on page 35.

Configure the QXtend QAD .NET Plug-in

30 Complete the required fields in the .NET UI Server panel.

Version. Enter the version of QAD .NET UI.

WebApp Name. Enter the name of the qad home server.

Port. Enter the Tomcat Port.

Tomcat Home. Enter the Tomcat installation home directory.

Admin Name. Enter the username for the Tomcat admin account.

Admin Password. Enter the Tomcat Admin password.

31 In the Outbound UI AppServer Setting panel, complete the required fields.

Direct Connect. Choose whether the plug-in is to connect in Direct Connect mode or via a Name Server.

AppServer Name. Enter the name of the QXO AppServer.

AppServer Host. Enter the host of the QXO AppServer.

Name Server Port. Enter the port of the NamServer (if Direct Connect is unchecked).

AppServer Port. Enter the port of the AppServer (if Direct Connect is checked).

32 Complete the required fields in the Include QXtend Menu Entries panel.

Include Inbound. Choose whether to add Inbound to the menu.

Include Outbound. Choose whether to add Outbound to the menu

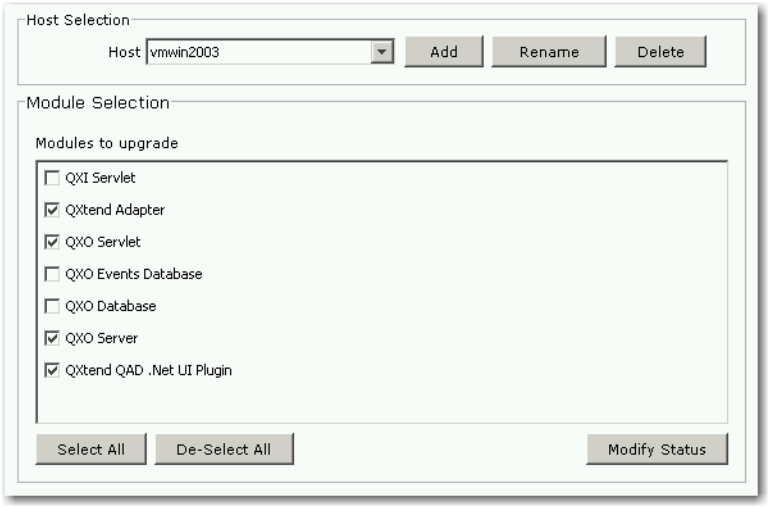
33 If you have entered all parameter settings for the required components, click Next and go to Default Configuration. Refer to Appendix C, “Default Configuration,” on page 77 for more information on default configuration.

34 Click Next and go to “Review the Pre-installation Summary” on page 41.

Upgrading QXtend

Select the Host and Modules to Upgrade

Fig. 4.10
Choose the Host and Install Set



1 In the Host Selection panel, enter the host on which to upgrade the components.

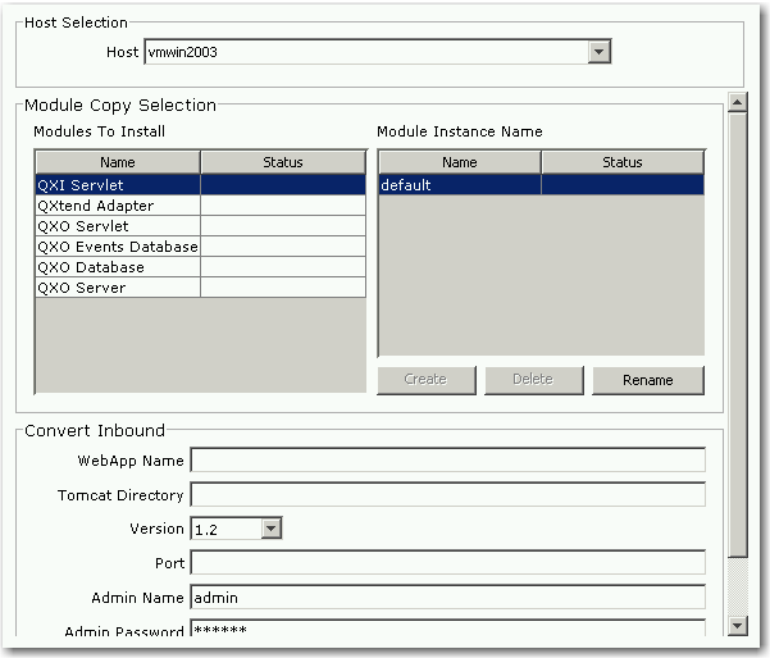
2 Use the Add and Delete buttons to create and delete hosts as required.

3 The components available for upgrade consist of code components and configuration components. In the Module Selection panel, select the components to upgrade on the specified host. Upgrading a code component installs a new instance of the component in the same location after backup of the contents to the `<target>/backup` directory.

If you select QXI Servlet, the system displays a warning message that says “PLEASE NOTE: The destination QXI will lose any existing configuration.”

4 Click Next to display the Module Copy Selection panel.

Fig. 4.11
Module Copy Selection



The component list on the left shows the components available for upgrade. The Status column on the right indicates the installation status for each component. The first time the installer is run, the Status column is blank. After an installation, the Status column shows Incomplete—indicating the component was not installed successfully—or Complete, indicating the component was installed successfully.

5 For each selected component for that instance, enter all of the required parameter settings, as described in the following sections.

Note

- Multiple instances are enabled only for the QXO Events Database and QXtend Adapter. The target QXODB and target QXEvents databases must be shut down before upgrading.
- You cannot select a different component until you have finished entering all of the required configuration information for the currently selected component. The fields that must be completed are highlighted in orange.

6 Click Next when you have entered all parameter settings for the required components.

Configure the QXI Servlet

7 In the Convert Configuration panel, complete the required fields.

WebApp Name. Enter the name of the application as known by the Tomcat application server.

Tomcat Directory. Enter the location of the home directory where Tomcat is installed.

Port. Enter the Tomcat port.

Admin Name. Enter the user name for the Tomcat manager role.

Admin Password. Enter the password of the user with the manager role.

Version. Enter the version being migrated from the drop-down list.

Note During QXI conversion, a prompt requests that Tomcat be shut down. This is done to replace the old servlet with the new one. Shut down Tomcat. Once it has shut down, click Ok. The installer replaces the servlet. When this process finishes, a second prompt displays requesting that Tomcat be restarted.

Important Shutdown/startup must be done before the button is pressed.

8 If necessary, select the next component to configure.

Configure the QXtend Adapter Code

9 Refer to steps 10 to 12 of “Configure the QXtend Adapter Code” on page 29 in the Installation section of this guide.

10 If necessary, select the next component to configure.

Note If the destination directory contains files, they are moved to `<destination directory>/<backup>`. The timestamp of each backed-up file is used in the filename.

Configure the QXO Servlet

11 Refer to steps 14 and 15 of “Configure the QXO Servlet” on page 30 in the Installation section of this guide.

12 If necessary, select the next component to configure.

Note If the WebApp directory exists, it is backed up to `<tomcat directory>/<backups>`. The timestamp of each backed-up file is used in the filename.

Configure the QXO Events Database

13 In the Convert Database panel, complete the required fields.

Physical Name. Enter the physical name of the `qxevents` database.

Database Directory. Enter the location of the database directory where the source `qxevents` database is installed.

Version. Select the version being converted in the drop-down list.

Progress Directory. Enter the location of Progress used by this QXEvents instance. This should be the same Progress version as the QAD Enterprise Application databases.

14 If necessary, select the next component to configure.

Configure the QXO Database

15 In the Convert Database panel, complete the required fields.

Physical Name. Enter the physical name of the QXO database.

Database Directory. Enter the location of the database directory where the source QXO database is installed.

Version. Select the version being converted in the drop-down list.

Progress Directory. Enter the location of Progress used by this instance of QXODB.

Configure the QXO Server

16 Refer to steps 26 to 28 of “Configure the QXO Server” on page 33 in the Installation section of this manual.

17 If necessary, select the next component to configure.

Note If the destination directory contains files, they are moved to `<destination directory>/<backup>`. The timestamp of each backed-up file is used in the filename.

Configure the QXtend QAD .NET Plug-in

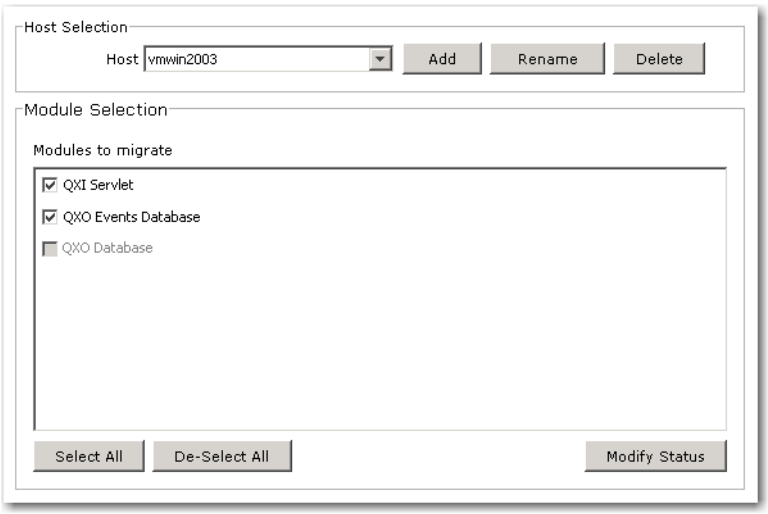
18 Refer to steps 30 to 33 under “Configure the QXtend QAD .NET Plug-in” on page 34 in the Installation section of this manual.

19 If you have entered all parameter settings for the required components, click Next and go to “Review the Pre-installation Summary” on page 41.

Migrating QXtend

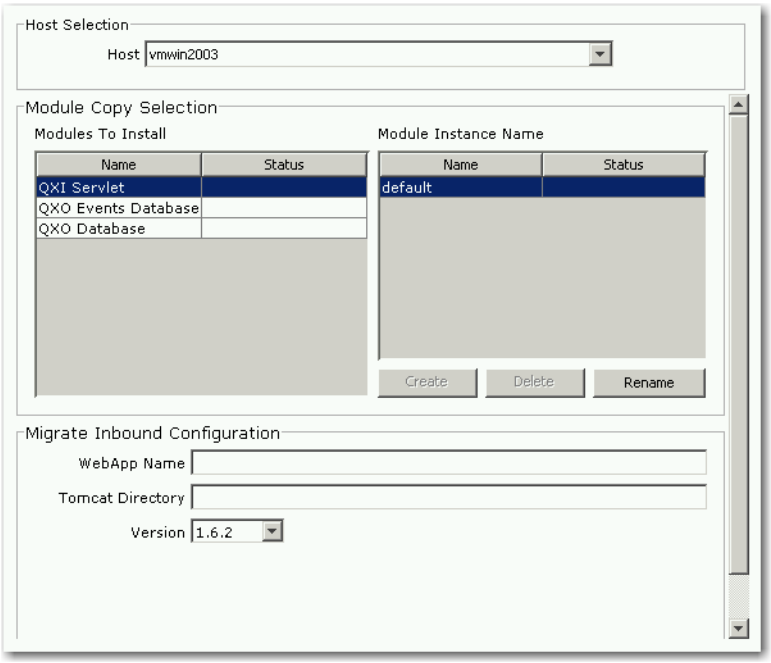
Select the Host and Modules to Upgrade

Fig. 4.12
Choose Host and Install Set



1. In the Host Selection panel, enter the host on which to install the components.
2. Use the Add and Delete buttons to create and delete hosts as required.
3. In the Module Selection panel, select the components to migrate on the specified host.
4. Click Next to display the Module Copy Selection screen.

Fig. 4.13
Module Copy Selection



The component list on the left shows the components available for installation. The Status column to the right indicates the installation status for each component. The first time the installer is run, the Status column is blank. After a migration, the Status column shows Incomplete—indicating the component has not been migrated successfully—or Complete, indicating the component was migrated successfully.

5 For each selected component, do the following as required:

a Create, rename, or delete instances.

Note

- New instances are enabled only for the QXO Events database.
- If you select QXO Database, a warning appears that says “PLEASE NOTE: The destination QXODB will lose any existing configuration.”

b Assign the required components to an instance by selecting the check box for a component when the instance in which it should be included is highlighted.

c For each selected component for that instance, enter all the required parameter settings as described in the following sections.

Note You cannot select a different component until you have finished entering all the required configuration information for the currently selected component. The fields that must be completed are highlighted in orange.

Configure the QXI Servlet

6 In the Migrate Configuration panel, complete the required fields.

WebApp Name. Enter the name of the application as known by the Tomcat application server.

Tomcat Directory. Enter the location of the home directory where Tomcat is installed.

Version. Enter the version being migrated in the drop-down list.

7 If necessary, select the next component to configure.

Configure the QXO Events Database

8 In the Migrate Configuration panel, complete the required fields.

Physical Name. Enter the physical name of the qxevents database.

Connection type. Select Local or Network.

Multi-User. Select this option to indicate that your database is operating in multi-user mode. Using this option means that you may not have to shut down your database before installing QAD QXtend. However, it is recommended that you do shut down your QXEvents database.

Database Directory. Enter the location of the database directory where the source qxevents database is installed.

Host. Enter the name of the host where the QAD Enterprise Applications database is installed.

Service Name. Enter the network service used by the QAD Enterprise Applications database.

Version. Select the version being migrated from the drop-down list.

9 If necessary, select the next component to configure.

Configure the Migration

10 In the Migrate Configuration panel, complete the required fields.

Physical Name. Enter the physical name of the QXO database

Connection Type. Select Local or Network.

MultiUser. Select this option to indicate that your database is operating in multi-user mode. Using this option means that you may not have to shut down your database before installing QAD QXtend. However, it is recommended that you do shut down your QXEvents database.

Database Directory. Enter the location of the database directory where the source QXO database is installed.

Host. Enter the name of the host where the QAD Enterprise Applications database is installed.

Service Name. Enter the network service used by the QAD Enterprise Applications database.

Version. Select the version being migrated in the drop-down list.

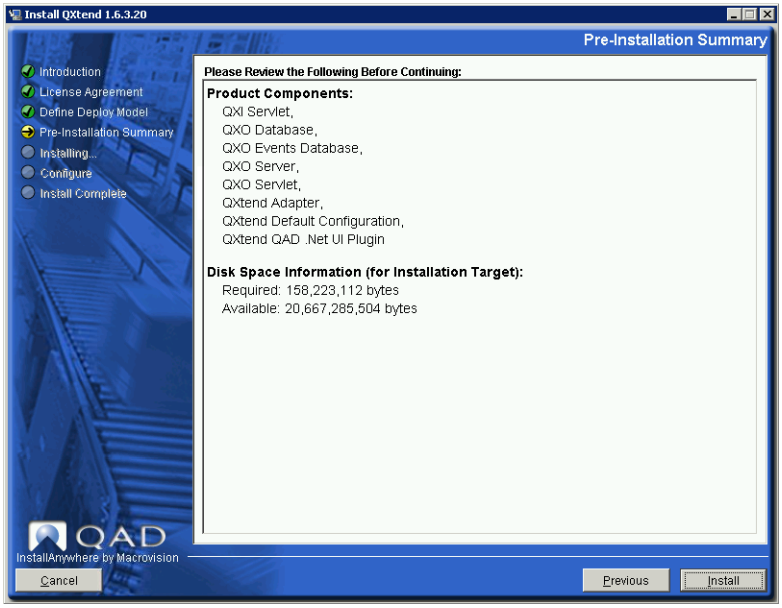
11 If you have entered all parameter settings for the required components, click Next and go to “Review the Pre-installation Summary” on page 41.

Review the Pre-installation Summary

The Pre-installation Summary screen displays the components selected for install and disk space information.

Note If no components were selected for installation on the host you are currently running, a message appears. Move to each host that is part of the installation and point to the configuration you have just created.

Fig. 4.14
Pre-installation Summary



Review the information and click Install to continue with the installation or upgrade.

Note If an error is encountered during installation, the installer will pause to allow you to correct the problem. Refer to Appendix D, “Process Control,” on page 81 for more information.

Review the Installation Summary

The Installation Summary screen displays three types of information about the installation or upgrade:

- Status
- Detailed Messages
- Detailed Errors

All system messages generated during installation are recorded in the install log; for details see Appendix A, “Installation Log,” on page 63.

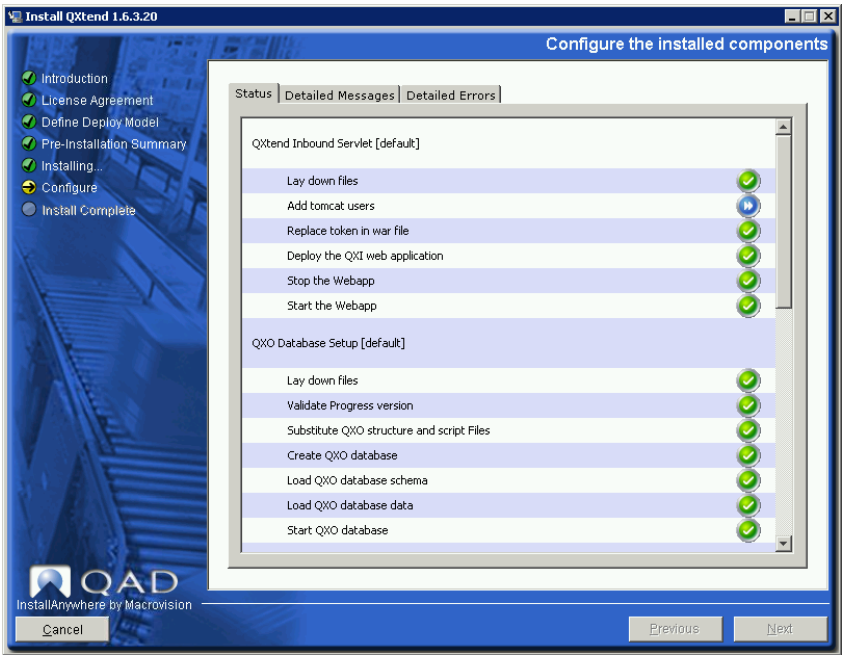
Status

The Status screen uses colored symbols to indicate the status of each component:

- A green check mark indicates the operation was completed successfully (success status).
- A red cross that indicates both non-fatal and fatal errors occurred (failure status).
- A blue double arrow indicates a task you can skip (previously completed or irrelevant).

A status also displays for any substeps performed during the installation or upgrade to allow you to easily identify issues. For example, Figure 4.15 shows that installing the `QXEvents` database consists of several substeps—creating the `QXEvents` database, loading the database schema, and so on; a status displays for each of these substeps.

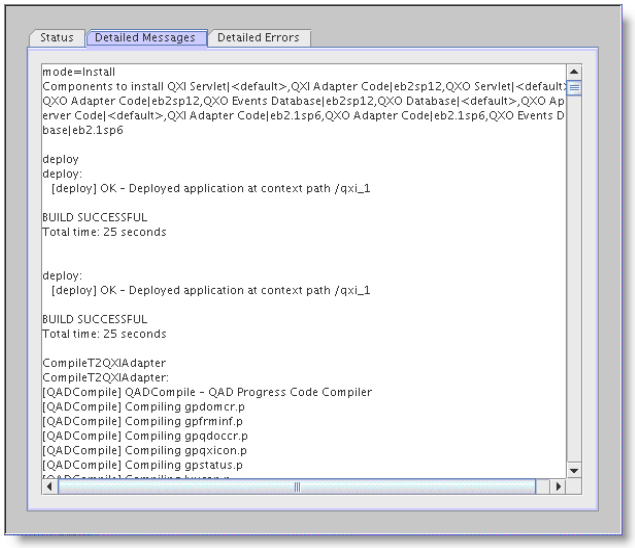
Fig. 4.15
Installation Status Screen



Detailed Messages

The Detailed Messages screen displays a record of all status messages generated during the installation or upgrade process. All the messages displayed are logged in the install log file.

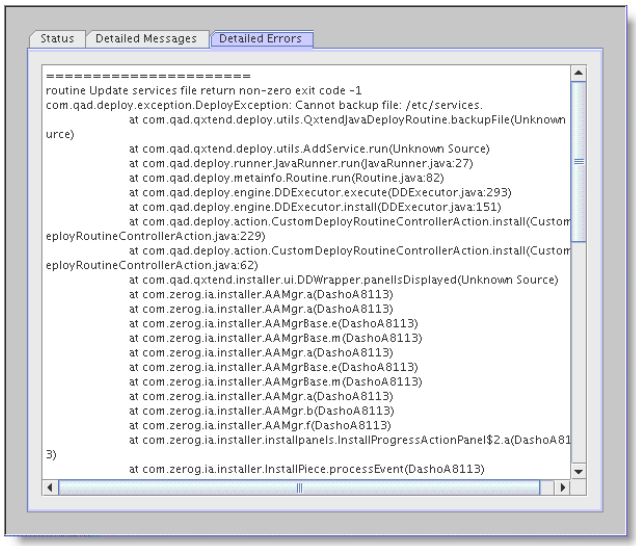
Fig. 4.16
Installation Detailed Messages Screen



Detailed Errors

The Detailed Errors screen shows a log of any fatal system errors encountered during the installation or upgrade. These system errors are typically out of your control and will display, for example, if the Java or Progress executable is corrupted. These messages indicate a serious problem with the system. Some of these errors are recorded in the install log file.

Fig. 4.17
Installation Detailed Errors Screen



Finalize the Installation

1. After reviewing the information, click Next. The Install Complete screen appears.
2. Press Done.

Installing, Upgrading, or Migrating QAD Qxtend in a Character Environment

Before installing, upgrading, or migrating QAD QXtend, ensure that Tomcat is running. You also should ensure that you have access permissions to the relevant folders. This procedure assumes you are performing a character installation in a Windows environment.

1. Run the GUI installer to populate the QDCS with the required parameter settings.
2. Specify an environment and host, select the components to install or upgrade, and specify the required parameters.

   Note For information about the QDCS see “QAD Deployment Configuration Service” on page 4.
3. Move to the host you created in Step 2.
4. Mount the CD-ROM.
5. Choose Start|All Programs|Accessories|Command Prompt to open a command window.
6. Navigate to the `InstData` directory on the CD-ROM.
7. Navigate to the appropriate directory for your environment. For example, if in a Linux environment, navigate to the Linux directory.
8. Navigate to the `VM` directory.
9. Start the executable by typing one of the following, depending on your environment:
   - On Windows: `QXtend.exe -i console`
   - On AIX, Linux, or HP-UX: `sh ./QXtend.bin -i console`
10. The Introduction text displays. Press Enter to continue. The first page of the License Agreement text displays.
11. Press Enter to move through and read the pages.
12. On the final page of the License Agreement, press Y to accept the terms, and then press Enter. The Log File Directory text displays.
13. Press Enter to accept the default location of the log file directory, or enter a different directory.
14. Select the Parameter Service type. Enter Y for network (the default) or N for local file.
15. The Get User Input text displays. These settings permit the use of the settings you defined using the QDCS in the GUI installer.
16. Enter the following:
    - Tomcat host name
    - Tomcat port number
    - WebApp name
    - Local parameter settings file name
    - Environment name

Note All of the above entries are case sensitive.

17 Choose whether to perform an install, upgrade, or migration.

18 The list of components being installed or upgraded displays. Review the list and press Enter to continue.

The installation or upgrade begins, using the parameter settings stored in the QDCS.

Note If an error is encountered during installation, the installer will pause to allow you to correct the problem. Refer to Appendix D, “Process Control,” on page 81 for more information.

Note After installing or upgrading QAD QXtend, you must restart the Tomcat AppServer and start QAD Enterprise Applications.

Chapter 5

Post-Installation and Post-Upgrade Tasks

This section describes post-installation and post-upgrade tasks. This section also contains information about conversion.

Post-Installation

These procedures are specific to QXO or QXI as indicated:

- Recompile the AppShell programs (required for eB2.1 SP4 to SP6 only)
- Ensure that the Java components are installed correctly (QXI)
- Use QDoc requestors instead of user IDs (QXI)
- Enable the Secure Socket Layer (QXI)
- Verify the QXO server installation (QXO)
- Verify the QXO server and source application communication (QXO)
- Verify the PROPATH and event recording (QXO)
- Verify QXO messages (QXO)

Logging In

To test that the Java components of the QXI component are correctly installed, do the following:

1. If you are not going from eB2.1 SP4 to SP6, go to step 2.

   If you are going from eB2.1 SP4 to SP6, you must recompile the AppShell programs with the QXtend Adapter in the PROPATH before the standard QAD Applications code. This is necessary because the ApShell uses an include file patched as part of the QXtend adapter.

2. Start Tomcat if it is not already running.

3. Launch a browser on your client machine.

4. Navigate to the following URL:

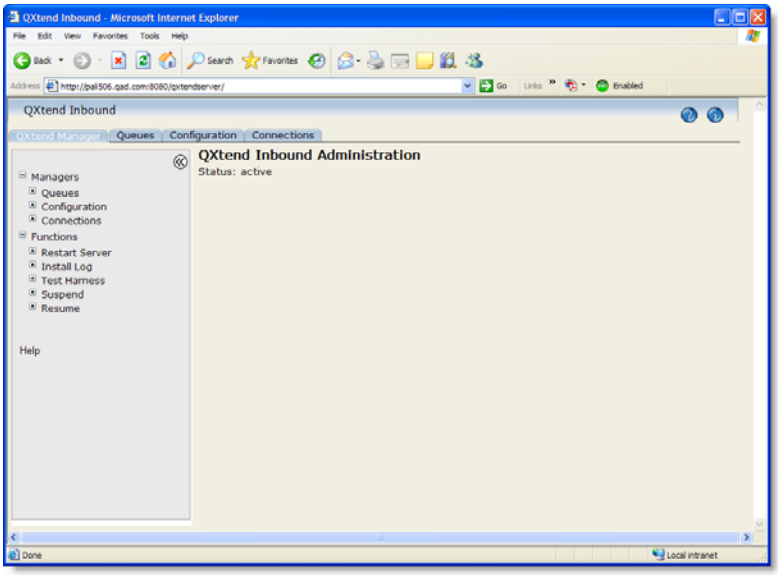
   `http://<qxtendhost>:8080/<QXI webapp>/index.jsp`

   The login screen displays.

5. Enter the admin user and password defined previously. Choose OK.

   The screen illustrated in Figure 5.1 displays.

Fig. 5.1
QXI Main Screen



Enable SSL

QAD QXtend supports SSL to encrypt all HTTP messages between the client and server. This requirement can be satisfied by correctly configuring Tomcat to provide an SSL connection to the QAD QXtend server Webapp.

To complete the configuration, you must:

- Modify server.xml.
- Modify catalina.bat or catalina.sh.
- Create Trusted Security Certificates in Java.

Modify server.xml

Define an SSL HTTP/1.1 connector on port 8443 in *TOMCAT_HOME*`\conf\server.xml`. The first section of the file content below appears in the default `server.xml`. Add the Connector definition as shown following the end of the comment marker (`-->`).

```
<!-- Define a SSL HTTP/1.1 Connector on port 8443 -->
<!--
<Connector port="8443" maxHttpHeaderSize="8192"
    maxThreads="150" minSpareThreads="25" maxSpareThreads="75"
    enableLookups="false" disableUploadTimeout="true"
    acceptCount="100" scheme="https" secure="true"
    clientAuth="false" sslProtocol="TLS" />
-->

<Connector
    className="org.apache.coyote.tomcat4.CoyoteConnector"
    port="8443"
    scheme="https"
    secure="true"
    useURIValidationHack="false"
    disableUploadTimeout="true">
</Connector>
```

Refer to the Tomcat documentation for further information.

To define SSL on AIX Java 5, set the `sslProtocol` attribute to `sslProtocol="SSL"` and add the attribute `algorithm="IbmX509"` as follows:

```
<!-- Define a SSL HTTP/1.1 Connector on port 8443 -->
<!--
<Connector port="8443" maxHttpHeaderSize="8192"
    maxThreads="150" minSpareThreads="25" maxSpareThreads="75"
    enableLookups="false" disableUploadTimeout="true"
    acceptCount="100" scheme="https" secure="true"
    clientAuth="false" sslProtocol="SSL"
    algorithm="IbmX509" />
-->
```

For more information, refer to the Tomcat SSL documentation.

Modify catalina.bat or catalina.sh

This sections describes the tasks to perform for Tomcat versions before 5.5.

In order for the QAD QXtend servlets to work with the HTTPS connector, properties must be set in *TOMCAT_HOME*`/bin/catalina.bat` for Windows and *TOMCAT_HOME*`\bin\catalina.sh` for UNIX. The QAD QXtend servlets properties are:

```
-Djavax.net.ssl.trustStore=<Location of the keystore>
-Djavax.net.ssl.trustStorePassword=<keystore password>
-Djava.protocol.handler.pkgs=com.sun.net.ssl.internal.www.protocol
```

Enter these lines into the file after each instance of RUNJDB or RUNJAVA. One entry is shown below as an example; however, there are typically seven or eight locations where the additions must be made.

```
exec "$_RUNJDB" $JAVA_OPTS $CATALINA_OPTS \
    -Djavax.net.ssl.trustStore=<Location of the keystore> \
    -Djavax.net.ssl.trustStorePassword=<keystore password> \
    -Djava.protocol.handler.pkgs=\com.sun.net.ssl.internal.www.protocol \
    -Djava.endorsed.dirs="$JAVA_ENDORSED_DIRS" -classpath "$CLASSPATH" \
    -sourcepath "$CATALINA_HOME"/../../jakarta-tomcat-4.0/catalina/src/share \
    -Djava.security.manager \
    -Djava.security.policy=="$CATALINA_BASE"/conf/catalina.policy \
    -Dcatalina.base="$CATALINA_BASE" \
    -Dcatalina.home="$CATALINA_HOME" \
    -Djava.io.tmpdir="$CATALINA_TMPDIR" \
    org.apache.catalina.startup.Bootstrap "$@" start
else
exec "$_RUNJDB" $JAVA_OPTS $CATALINA_OPTS \
    -Djavax.net.ssl.trustStore=<The location of the keystore> \
    ...
```

Create Trusted Security Certificates

The machine on which Tomcat is running requires a trusted security certificate. If the HTTPS client cannot find a trusted security certificate on the QAD QXtend server, it may issue a warning to that effect or may refuse to work with a HTTPS connection to the server. This behavior can cause a fault on the client side.

A self-signed server certificate must be generated with the Java keytool. The keytool program prompts you for several values.

1. In a command window, navigate to a temporary directory such as `c:\temp` and run the following command:

   ```
   %JAVA_HOME%\bin\keytool -genkey -alias tomcat -keyalg RSA -keystore
   c:\temp\.keystore
   ```

2. You are prompted to enter the keystore password. The default SSL password for Tomcat is `changeit`. If you change this, you must place the new password in `server.xml` and `catalina.bat`.

   You are prompted to enter your first and last name.

3. Enter the name of the host where the QXI webapp is installed; for example, `corp_pc01`. It must be the same host specified in `qxtendconfig.xml` for MessageReceiverServlet. This and all subsequent prompts use the following format:

   ```
   [Unknown]: <Enter data value here.>
   ```

4. You are prompted for the following information for which you can enter the appropriate values for your implementation:
   - The name of your organizational unit
   - The name of your organization
   - The name of your city or locality
   - The name of your state or province
   - The two-letter country code for this unit

5. You are prompted to confirm:

   ```
   Is CN=corp_pc01, OU=QAD, O=QAD, L=Summerland, ST=California, C=US correct?
   [no]:  y
   ```

   The common name (CN) must be the correct QXI webapp host or the Web service call reports the validation error “HTTPS hostname wrong.”

6. You are prompted to enter the key password for `<tomcat>`.

   Press Return if the key password for Tomcat is the same as the keystore password. Do not do this unless you are experienced with HTTPS.

Refer to the Java tools documentation for details.

Verify QXO Server Installation

This test verifies that the QXO Server was correctly installed, the required AppServer was correctly configured, and the QXO user interface was correctly installed and configured.

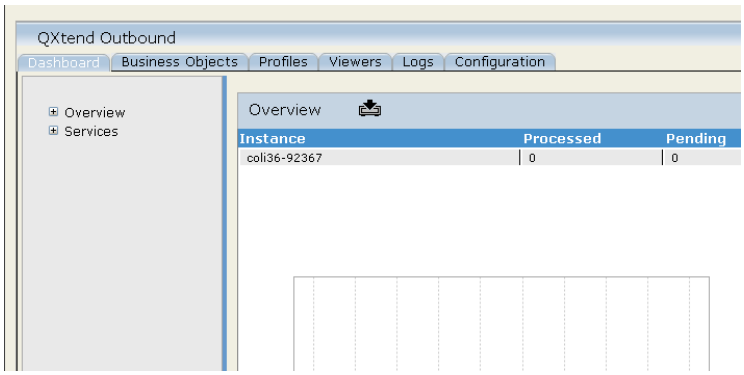
1. Ensure the Tomcat application server is running.
2. Ensure the Progress AppServer is running.
3. Ensure the QAD Enterprise Applications production and `QXEvents` database servers are running.

4 Start a browser session on the client and set the URL to:

```
http://<qxtendhost>:<port>/<QXO webapp>
```

5 The QXO Dashboard should display as in Figure 5.2.

Fig. 5.2
QXO Dashboard



Note Refer to *Technical Reference: QAD QXtend* for details about the administrative functions.

If your system is not configured properly, the following message might display when you try to display the QXO Console:

```
Unable to start Appserver AppServer://servername:9999/QXOSession appserver
com.qad.qxo.proxy.QXOProxy
```

This message indicates that one of the following has occurred:

- The AppServer has not been started on the server *`servername`*, possibly because of configuration problems.
- The configuration file for the QXO UI (`environmentmanager.xml`) has not been correctly modified to point to the AppServer. Check that:
  - The name of the server where the AppServer is running is correct.
  - The correct port is specified for the NameServer with which the AppServer is registered.
  - The name of the AppServer is correct. The default name is QXOSession, but this can be modified if multiple AppServers are running on the same machine.

Verify QXO Server and Source Application Connection

This test ensures that the QXO server connects correctly to the `QXEvents` database. It assumes that a suitable QXO-adapted QAD Enterprise Applications is available as a source application.

1 Start a browser session on the client and set the URL to:

```
http://<qxtendhost>:<port>/<QXO webapp>
```

2 In the QXO Console, click the Configuration tab and then the Source Applications node in the left-hand tree view. Create a new source application and define its details.

3 Click Event Types under your Source Application and ensure that the event types appear.

If your system is not configured properly, the following message might display when you try to display the event types:

```
Database not available
```

This message indicates that one of the following has occurred:

1. The database name is configured incorrectly; correct the database name using the Configuration tab.
2. The `QXEvents` database server is not running. Verify that the servers are started for the QAD Enterprise Applications databases including `QXEvents` using your `start.`*`DBName`* script.

If a blank screen displays, this indicates that the database connection parameters for the `QXEvents` database are incorrect. Correct the database connection parameters using the Configuration tab.

Verify PROPATH and Event Recording

This test ensures that the QAD Enterprise Applications database set was configured correctly with the correct PROPATH and database connections. The test verifies that an event is recorded when a change in the source application occurs.

It assumes that a suitable QXO-adapted QAD Enterprise Applications is available as a source application.

1. Start a browser session on the client and set the URL to:

   ```
   http://<qxtendhost>:<port>/<QXO webapp>
   ```

2. In the QXO Console, click the Configuration tab and then click Event Types under your source application.
3. Edit the source application event types and select the Active check box for the cm_mstr event type.
4. Launch a QAD Enterprise Applications session. Use Customer Maintenance (2.1.1) to create a new customer.

   If the following error message displays in Customer Maintenance, the PROPATH is incorrect:

   ```
   ** "filename" was not found. (293)
   ```

   Correct the PROPATH for the QXO database set and create new startup scripts.
5. Go to QXtend Outbound Control (36.16.19) and verify that QXO is enabled in the QAD application. The Enable QXtend Outbound option must be set to Yes before any event is logged to the QXEvents database.
6. From your QAD Enterprise Applications session, access the Progress Editor. From Tools|Data Dictionary|Database, choose Select Working Database and select the `QXEvents` database.
7. Choose Admin|Dump Data and Definitions|Table Contents (`.d`) file. Choose the ert_app_event table, press Go, and specify an output file name.
8. Open the output file in a text editor and verify that there is a record similar to the following:

   ```
   1 "0x0057d684" "TMP1497" "mfg" "" "cm_mstr" 2453319.175590277 "mfgprod" 1 3
   ```

If no records display in the file, one of the following may have occurred:

- The cm_mstr event type is not active. Restart this test and ensure you select the Active check box for cm_mstr in the QXO Console.
- The source application database schema was not updated correctly during installation. Verify that replication write and replication delete triggers exist for the cm_mstr table. If these triggers do not exist, you must reinstall the adapter to QAD Enterprise Applications.

To verify the existence of the triggers:

1. From your QAD Enterprise Applications session, access the Progress Editor.
2. From Tools|Data Dictionary|Database, choose Select Working Database and select the `qaddb` database.
3. Choose Database|Reports|Trigger.
4. In the Trigger Report, ensure that the following displays:

```
Table/Field Name  Event     Check CRC  Flags   Procedure
cm_mstr           RP-DEL    no                 cmrd.t
                  RP-WRI    no                 cmrw.t
```

Verify QXO Messages

This test ensures that a subscriber to a particular profile receives an outbound message after a change to the profile. It assumes that the QAD Enterprise Applications source application was added and the databases were correctly set up.

1. Load the Customer business object (`Customer.xml`) using the XML Import screen under the Configuration tab in the QXO Console.
2. In the Configuration tab, click Subscribers in the navigation tree and create a new subscriber.
3. Register the default profile for the Customer BO with the subscriber.
4. In the Configuration tab, click Event Services in the navigation tree and create a new event service.
5. Register the QAD Enterprise Applications source application with the event service.
6. In the Configuration tab, click Message Publishers in the navigation tree and create a new message publisher session profile.
7. Register the Customer BO with the message publisher.
8. In the Configuration tab, click Message Senders in the navigation tree and create a new message sender session profile.
9. Register the new subscriber with the message sender.

10 In the QXO Dashboard, start the event service, message publisher, and message sender sessions. The event created in the previous test when you created a new customer will be processed by QXO.

11 Verify that a message is sent to the subscriber you created by clicking the Log tab in the QXO Console.

Post-Upgrade

QXtend Outbound Configuration

Following a QXtend 1.6 Outbound upgrade, the following steps are required:

1 Update the source application database details to ensure that the QXEvents database points to the 1.6 version of QXEvents database. This applies when migrating the QXtend Outbound configuration. During an in-place conversion, the existing QXEvents database specified in the source application connection details is converted to the 1.6 version.

   To update the source application database details, go to the QXtend Outbound UI, select the Configuration tab, and make the necessary changes in the Databases pane.

2 Update QXtend Web Service subscribers in QXtend Outbound. For each QXtend Web Service subscriber, verify that the entries in the Tomcat Host, Tomcat Port, and Webapp Name fields point to the 1.6 Web Application.

3 Set the license manager details for QXtend Outbound. This can be set in QXO by selecting Configuration|Outbound Settings|License Manager.

4 When migrating from QXO 1.1 or QXO 1.3, all business objects must be validated manually before an event message can be extracted. To validate business objects, select the Business Objects tab followed by the business object, and click Validate.

   This regenerates the schema definition for the business objects.

Chapter 6

Troubleshooting QAD QXtend Installs and Upgrades

This section describes how to resolve QAD QXtend installation and upgrade problems.

Overview

This section describes how to resolve any issues encountered during or after installation or upgrade. Automated installation or upgrade creates a log file; for details see Appendix A, “Installation Log,” on page 63.

Manage Lock Table Sizes

Setting up the `qxodb` database requires setting the size of the lock table for the database server. The value recommended is at least 200,000. This is based on use of the QAD-defined business objects.

The following factors affect how this setting should be adjusted:

- The number of tables in a business object
- The number of fields in each business object table
- The number of records extracted for a table in a single business object message
- The number of event service processes running in parallel

An increase in any of these results in an increase in the number of locks required. Should the number of locks required exceed the number available, the Progress session trying to get those locks is automatically restarted (STOP condition) and an error is logged in the session log file and the database log file.

If this happens, do the following:

1. Shut down the QAD QXtend service processes.
2. Shut down the `qxodb` database server.
3. Double the `-L` parameter value in the database startup script.
4. Restart the database server.
5. Restart the QAD QXtend service processes.

The messages being processed when the process went down are backed out. They are picked up and processed correctly when the processes come back up.

You may need to carry out these steps several times to find a value that works with your business objects.

Invoke Proxy Calls

If an “Unable to invoke proxy call” error message is generated during reconnection, increase the maximum servers in the AppServer pool. The default is 1; try a setting of 5.

On Windows

1. Open the Progress Explorer.
2. Connect to the AppServer QXOSession host machine.

3 View the AppServer properties.

4 Navigate to Agent|Pool Range.

5 Set Maximum servers to 5.

On UNIX

1 Open *ProgressInstallDir*`\properties\ubroker.properties` in a text editor.

2 Modify the QXOSession entry by adding:

```
minSrvrInstance=1
maxSrvrInstance=5
```

Section 2

Appendix

This appendix contains additional information related to QAD QXtend installation.

Appendix A

Installation Log

This section describes the install log file generated by the QAD QXtend automated installation process.

Overview

The `QXtendInstallLog.log` file records all system messages generated during an automated install of the various QAD QXtend components selected for installation: QXI servlet, QXI adapter code, QXO servlet, QXO adapter code, QXO events database, QXO database, and QXO AppServer code. This file is located in the directory specified during installation; by default this is `C:\instlog`.

Note This section describes messages generated by the QAD QXtend application. It does not describe messages generated by third-party supporting applications, such as Progress. For descriptions of errors generated by these applications, refer to their product documentation.

QXtendInstallLog File

This section details the various sections of the install log file and the information it contains.

Inbound Servlet Deployment

```
deploy:
   [deploy] OK - Deployed application at context path /QXI
```

Inbound Adapter Compilation

```
CompileT2QXIAdapter:
[QADCompile] QADCompile - QAD Progress Code Compiler
[QADCompile] Compiling gpfrminf.p
[QADCompile] Compiling gpqxicon.p
[QADCompile] Compiling gpstatus.p
[QADCompile] Compiling lvucap.p
[QADCompile] Compiling mfqxtend.p
[QADCompile] Compiling mfsyad02.p
[QADCompile] Compiling mfwb01aa.p
[QADCompile] Compiling mfwsock2.p
[QADCompile] Compiling mfww01.p
[QADCompile] Compiling mfww01b.p
[QADCompile] Compiling mfwwc01.p
[QADCompile] 11 file(s) compiled

BUILD SUCCESSFUL
Total time: 17 seconds
```

Outbound Servlet Deployment

```
deploy:
   [deploy] OK - Deployed application at context path /QXO
```

Outbound Adapter Compilation

1. Create an empty `QXEvents` database.

   This database is created where the QAD Enterprise Applications empty databases are located.

   ```
   CreateT2EmptyQXEvents:
   BUILD SUCCESSFUL
   Total time: 0 seconds
   ```

2. Load the `QXEvents` schema.

   ```
   LoadT2QXEventsSchema:
   ```

```
[PCTLoadSchema] Loading C:\QXtend\qxoAdapter\df\qxevents.df into database

BUILD SUCCESSFUL
Total time: 3 seconds
```

3 Compile the QXO adapter programs.

```
CompileT2QXOAdapter:
[QADCompile] QADCompile - QAD Progress Code Compiler
[QADCompile] Compiling apipl-db.p
[QADCompile] Compiling apipl.p
[QADCompile] Compiling gpqxocnf.p
[QADCompile] Compiling gpqxohdl.p
[QADCompile] Compiling qxoevent.p
[QADCompile] Compiling util\viewfiles.p
[QADCompile] 6 file(s) compiled

BUILD SUCCESSFUL
Total time: 5 seconds
```

4 Compile the QAD Enterprise Applications database triggers and in-line triggers.

```
CompileT2MFGTriggers:
[QADCompile] QADCompile - QAD Progress Code Compiler
[QADCompile] Compiling abdrd.t
[QADCompile] Compiling abdrw.t
[QADCompile] Compiling absccrd.t
[QADCompile] Compiling absccrw.t
[QADCompile] Compiling abscrd.t
[QADCompile] Compiling abscrw.t
[QADCompile] Compiling absdrd.t
<etc ...>
QADCompile] Compiling worw.t
[QADCompile] Compiling wrrd.t
[QADCompile] Compiling wrrw.t
[QADCompile] Compiling wtxrd.t
[QADCompile] Compiling wtxrw.t
[QADCompile] 1509 file(s) compiled

BUILD SUCCESSFUL
Total time: 3 minutes 26 seconds
```

QXEvents Database

1 Update the QAD Enterprise Applications database schema with the replication-write and replication-delete trigger definitions.

```
UpdateT4MFGSchema:
[PCTLoadSchema] Loading C:\QXtend\qxevents\df\mfgdelta.df into database

BUILD SUCCESSFUL
Total time: 13 seconds
```

2 Load menus and messages into QAD Enterprise Applications database. This data may already exist in the QAD Enterprise Applications database and would give errors if it does. These can be safely ignored.

```
LoadT4MFGData:
[PCTLoadData]   Loading Data.   Press CTRL-BREAK to terminate the load process.
[PCTLoadData] Records   Total   Expected
[PCTLoadData]   Table Load file Loaded    Errors   Records
[PCTLoadData]   ----------------- --------------------------
[PCTLoadData] mnd_det C:\QXtend\qxevents\data\qad 0 1 1
[PCTLoadData] Load of database contents completed.
[PCTLoadData] Errors/Warnings listed in .e files placed into same directory as .d files
[PCTLoadData]   Loading Data.   Press CTRL-BREAK to terminate the load process.
[PCTLoadData] Records   Total   Expected
```

```
[PCTLoadData]   Table Load file Loaded    Errors   Records
[PCTLoadData]   ----------------- -------------------------- ----
[PCTLoadData]   mnt_det C:\QXtend\qxevents\data\qad 0 1 1
[PCTLoadData] Load of database contents completed.
[PCTLoadData] Errors/Warnings listed in .e files placed into same directory as .d files
[PCTLoadData]   Loading Data.   Press CTRL-BREAK to terminate the load process.
[PCTLoadData] Records   Total   Expected
[PCTLoadData]   Table Load file Loaded    Errors   Records
[PCTLoadData]   ----------------- -------------------------- -----
[PCTLoadData]   msg_mstr C:\QXtend\qxevents\data\qad 0 1 2
[PCTLoadData] Load of database contents completed.
[PCTLoadData] Errors/Warnings listed in .e files placed into same directory as .d files

BUILD SUCCESSFUL
Total time: 12 seconds
```

3 Load labels into the QAD Enterprise Applications admin database. This data may already exist in the QAD Enterprise Applications admin database and would give errors if it does. These can be safely ignored.

```
LoadT4AdminData:

[PCTLoadData]   Loading Data.   Press CTRL-BREAK to terminate the load process.
[PCTLoadData] Records   Total   Expected
[PCTLoadData]   Table Load file Loaded    Errors   Records
[PCTLoadData]   ----------------- --------------------------
[PCTLoadData]   lbl_mstr C:\QXtend\qxevents\data\qad 0 1 2
[PCTLoadData] Load of database contents completed.
[PCTLoadData] Errors/Warnings listed in .e files placed into same directory as .d files
[PCTLoadData]   Loading Data.   Press CTRL-BREAK to terminate the load process.
[PCTLoadData] Records   Total   Expected
[PCTLoadData]   Table Load file Loaded    Errors   Records
[PCTLoadData]   ----------------- --------------------------
[PCTLoadData]   lbld_det C:\QXtend\qxevents\data\qad 0 1 1
[PCTLoadData] Load of database contents completed.
[PCTLoadData] Errors/Warnings listed in .e files placed into same directory as .d files
[PCTLoadData]   Loading Data.   Press CTRL-BREAK to terminate the load process.
[PCTLoadData] Records   Total   Expected
[PCTLoadData]   Table Load file Loaded    Errors   Records
[PCTLoadData]   ----------------- --------------------------
[PCTLoadData]   pgmi_mstr C:\QXtend\qxevents\data\qad 0 1 1
[PCTLoadData] Load of database contents completed.
[PCTLoadData] Errors/Warnings listed in .e files placed into same directory as .d files

BUILD SUCCESSFUL
Total time: 13 seconds
```

4 Create the `QXEvents` database.

```
CreateT4QXEvents:
[PCTCreateBase] Procopy session begin for Administrator on batch. (451)
[PCTCreateBase] Formatting extents:
[PCTCreateBase] size area name   path name
[PCTCreateBase] 4 Primary Recovery Area C:\QXtend\qxevents\qxevents.b1 00:00:00
[PCTCreateBase] 16 Schema Area C:\QXtend\qxevents\qxevents.d1 00:00:01
[PCTCreateBase] Copying C:\dlc101b\empty8 to qxevents... (6715)
[PCTCreateBase] Start writing data blocks. (6718)
[PCTCreateBase] 11:07:27 10 Percent complete.
[PCTCreateBase] 11:07:27 20 Percent complete.
[PCTCreateBase] 11:07:27 30 Percent complete.
[PCTCreateBase] 11:07:27 40 Percent complete.
[PCTCreateBase] 11:07:27 50 Percent complete.
[PCTCreateBase] 11:07:27 60 Percent complete.
[PCTCreateBase] 11:07:27 70 Percent complete.
[PCTCreateBase] 11:07:27 80 Percent complete.
[PCTCreateBase] 11:07:27 90 Percent complete.
[PCTCreateBase] 11:07:27 100 Percent complete.
[PCTCreateBase] 297 blocks copied. (6720)
[PCTCreateBase] ...Copy complete. (6722)
```

```
[PCTCreateBase] Database copied from C:\dlc101b\empty8. (1365)
[PCTCreateBase] Procopy session end. (334)

BUILD SUCCESSFUL
Total time: 14 seconds
```

5 Load the `QXEvents` schema.

```
LoadT4QXEventsSchema:
[PCTLoadSchema] Loading C:\QXtend\qxevents\df\qxevents.df into database

BUILD SUCCESSFUL
Total time: 2 seconds
```

6 Load the `QXEvents` default data.

```
LoadT4QXEventsData:
[PCTLoadData]   Loading Data.   Press CTRL-BREAK to terminate the load process.
[PCTLoadData] Records   Total   Expected
[PCTLoadData]   Table Load file Loaded    Errors   Records
[PCTLoadData]   ----------------- -------------------------- -
[PCTLoadData] erm_event_type C:\QXtend\qxevents\data\qxe 45 0 45
[PCTLoadData] Load of database contents completed.

BUILD SUCCESSFUL
Total time: 5 seconds
```

7 Start the `QXEvents` database in multi-user mode.

```
StartT4QXEvents:

BUILD SUCCESSFUL
Total time: 2 seconds
```

Outbound Database

1 Validate the Progress version is 10.1C minimum.

```
ValidateT5Progress:

BUILD SUCCESSFUL
Total time: 16 seconds
```

2 Create the Outbound database.

```
CreateT5QXODB:
[PCTCreateBase] Procopy session begin for Administrator on batch. (451)
[PCTCreateBase] Formatting extents:
[PCTCreateBase]    size area name   path name
[PCTCreateBase]       4 Primary Recovery Area C:\QXtend\qxodb\db\qxodb.b1 00:00:00
[PCTCreateBase] 16 Schema Area C:\QXtend\qxodb\db\qxodb.d1 00:00:00
[PCTCreateBase] Copying C:\dlc101b\empty8 to qxodb... (6715)
[PCTCreateBase] Start writing data blocks. (6718)
[PCTCreateBase] 11:08:13 10 Percent complete.
[PCTCreateBase] 11:08:13 20 Percent complete.
[PCTCreateBase] 11:08:13 30 Percent complete.
[PCTCreateBase] 11:08:13 40 Percent complete.
[PCTCreateBase] 11:08:13 50 Percent complete.
[PCTCreateBase] 11:08:13 60 Percent complete.
[PCTCreateBase] 11:08:13 70 Percent complete.
[PCTCreateBase] 11:08:13 80 Percent complete.
[PCTCreateBase] 11:08:13 90 Percent complete.
[PCTCreateBase] 11:08:13 100 Percent complete.
[PCTCreateBase] 297 blocks copied. (6720)
[PCTCreateBase] ...Copy complete. (6722)
[PCTCreateBase] Database copied from C:\dlc101b\empty8. (1365)
[PCTCreateBase] Procopy session end. (334)
```

```
BUILD SUCCESSFUL
Total time: 21 seconds
```

3 Load the Outbound database schema.

```
LoadT5QXODBSchema:
[PCTLoadSchema] Loading C:\QXtend\qxodb\db\qxodb.df into database

BUILD SUCCESSFUL
Total time: 5 seconds
```

4 Load the Outbound database default data.

```
LoadT5QXODBData:
[PCTLoadData] Loading Data. Press CTRL-BREAK to terminate the load process.
[PCTLoadData] Records   Total   Expected
[PCTLoadData]   Table Load file Loaded    Errors   Records
[PCTLoadData]   ----------------- -------------------------
[PCTLoadData]   qxodb_ctrl C:\QXtend\qxodb\db\qxodb_ct 1 0 1
[PCTLoadData] Load of database contents completed.
[PCTLoadData]   Loading Data.   Press CTRL-BREAK to terminate the load process.
[PCTLoadData] Records   Total   Expected
[PCTLoadData]   Table Load file Loaded    Errors   Records
[PCTLoadData]   ----------------- -------------------------
[PCTLoadData] esm_profile_type C:\QXtend\qxodb\db\esmproty 2 0 2
[PCTLoadData] Load of database contents completed.
[PCTLoadData]   Loading Data.   Press CTRL-BREAK to terminate the load process.
[PCTLoadData] Records   Total   Expected
[PCTLoadData]   Table Load file Loaded    Errors   Records
[PCTLoadData]   ----------------- -------------------------
[PCTLoadData]   adm_src_apptype C:\QXtend\qxodb\db\src_appt 5 0 5
[PCTLoadData] Load of database contents completed.

BUILD SUCCESSFUL
Total time: 13 seconds
```

5 Start the Outbound database in multi-user mode.

```
StartT5QXODB:

BUILD SUCCESSFUL
Total time: 2 seconds
```

Outbound Server

1 Validate the Progress version is 10.1C minimum.

```
ValidateT3Progress:

BUILD SUCCESSFUL
Total time: 0 seconds
```

2 Create an empty Outbound database (`qxodbempty`).

```
CreateT3EmptyQXODB:
[PCTCreateBase] Procopy session begin for Administrator on batch. (451)
[PCTCreateBase] Formatting extents:
[PCTCreateBase]    size               area name   path name
[PCTCreateBase]       4    Primary Recovery Area C:\QXtend\appServer\db\qxodbempty.b1
00:00:00
[PCTCreateBase]      16             Schema Area C:\QXtend\appServer\db\qxodbempty.d1
00:00:00
[PCTCreateBase] Copying C:\dlc101b\empty8 to qxodbempty... (6715)
[PCTCreateBase] Start writing data blocks. (6718)
[PCTCreateBase] 11:09:00 10 Percent complete.
[PCTCreateBase] 11:09:00 20 Percent complete.
[PCTCreateBase] 11:09:00 30 Percent complete.
```

```
[PCTCreateBase] 11:09:00 40 Percent complete.
[PCTCreateBase] 11:09:00 50 Percent complete.
[PCTCreateBase] 11:09:00 60 Percent complete.
[PCTCreateBase] 11:09:00 70 Percent complete.
[PCTCreateBase] 11:09:00 80 Percent complete.
[PCTCreateBase] 11:09:00 90 Percent complete.
[PCTCreateBase] 11:09:00 100 Percent complete.
[PCTCreateBase] 297 blocks copied. (6720)
[PCTCreateBase] ...Copy complete. (6722)
[PCTCreateBase] Database copied from C:\dlc101b\empty8. (1365)
[PCTCreateBase] Procopy session end. (334)

BUILD SUCCESSFUL
Total time: 17 seconds
```

3 Load the Outbound database schema.

```
LoadT3QXODBSchema:
[PCTLoadSchema] Loading C:\QXtend\appServer\db\qxodb.df into database

BUILD SUCCESSFUL
Total time: 5 seconds
```

4 Create the empty `QXEvents` database (`qxevtempty`).

```
CreateT3EmptyQXEvents:
[PCTCreateBase] Procopy session begin for Administrator on batch. (451)
[PCTCreateBase] Formatting extents:
[PCTCreateBase] size area name   path name
[PCTCreateBase] 4   Primary Recovery Area C:\QXtend\appServer\db\qxevtempty.b1 00:00:00
[PCTCreateBase] 16 Schema Area C:\QXtend\appServer\db\qxevtempty.d1 00:00:00
[PCTCreateBase] Copying C:\dlc101b\empty8 to qxevtempty... (6715)
[PCTCreateBase] Start writing data blocks. (6718)
[PCTCreateBase] 11:09:25 10 Percent complete.
[PCTCreateBase] 11:09:25 20 Percent complete.
[PCTCreateBase] 11:09:25 30 Percent complete.
[PCTCreateBase] 11:09:25 40 Percent complete.
[PCTCreateBase] 11:09:25 50 Percent complete.
[PCTCreateBase] 11:09:25 60 Percent complete.
[PCTCreateBase] 11:09:25 70 Percent complete.
[PCTCreateBase] 11:09:25 80 Percent complete.
[PCTCreateBase] 11:09:25 90 Percent complete.
[PCTCreateBase] 11:09:25 100 Percent complete.
[PCTCreateBase] 297 blocks copied. (6720)
[PCTCreateBase] ...Copy complete. (6722)
[PCTCreateBase] Database copied from C:\dlc101b\empty8. (1365)
[PCTCreateBase] Procopy session end. (334)

BUILD SUCCESSFUL
Total time: 17 seconds
```

5 Load the `QXEvents` schema.

```
LoadT3QXEventsSchema:
[PCTLoadSchema] Loading C:\QXtend\appServer\db\qxevents.df into database

BUILD SUCCESSFUL
Total time: 2 seconds
```

6 Compile the Outbound server programs.

```
CompileT3QXOServer:
[QADCompile] QADCompile - QAD Progress Code Compiler
[QADCompile] Compiling alertAPI.p
[QADCompile] Compiling apipl-db.p
[QADCompile] Compiling apipl.p
[QADCompile] Compiling archiveservice.p
[QADCompile] Compiling bo-bo.p
[QADCompile] Compiling bo-dashboard.p
```

```
[QADCompile] Compiling bo-dumpboxml.p
[QADCompile] Compiling bo-eventmsg.p
[QADCompile] Compiling bo-forcepublish.p

<etc ... >

[QADCompile] Compiling trg\mpt_msg_subs_resp_c.t
[QADCompile] Compiling trg\mpt_publ_msg_c.t
[QADCompile] Compiling valconfig.p
[QADCompile] Compiling xml-bo-load.p
[QADCompile] Compiling xml-profile-load.p
[QADCompile] 222 file(s) compiled

BUILD SUCCESSFUL
Total time: 2 minutes 28 seconds
```

7 Create an AppServer entry in the `ubroker.properties` file.

```
T3UpdateAppserver:
[PCTASBroker] OpenEdge Release 10.1B02 as of Fri Jul 27 00:45:24 EDT 2007

BUILD SUCCESSFUL
Total time: 10 seconds
```

Appendix B

Typical Installation Parameters

This section describes the typical parameters provided for basic QXtend installations.

Overview

Beginning with version 1.6.2, QXtend provides a simplified installation process using a default QXtend configuration bundled with the product to automatically populate the various installer fields.

If you wish to modify any of default configuration parameters, use the QXtend's advanced installation mode.

Tokens

Tokens enable you to reference parameters from other fields dynamically. Any parameter name can be used as a token by encasing the parameter name in "@" symbols. The tokens also work recursively.

For example, to resolve `QXODB_DIR`:

```
QXODB_DIR = @QXOSERVER_DIR@/db
QXOSERVER_DIR = @GLOBAL_QXTEND_DIR@/qxoserver
GLOBAL_QXTEND_DIR = /qad/qxtend
```

The result is:

```
QXODB_DIR = /qad/qxtend/qxoserver/db
```

Values are stored as tokens in the repository and resolved at runtime during configuration in the UI, or at runtime as ant properties.

Note During configuration, hovering over a field label shows you the corresponding parameter name. If you hover over the field itself, the fully resolved parameter is shown as a tool tip.

Parameters

The following table describes the parameters provided with the default QXtend configuration.

Table B.1 Default QXtend Installation Parameters (Page 1 of 5)

Module	Parameter Name	Description	Default Value
Qxi Servlet	INBOUND_TOMCAT_ADMIN	Admin Name	admin
Qxi Servlet	INBOUND_TOMCAT_HOME	Home Directory	@GLOBAL_TOMCAT_HOME@
Qxi Servlet	INBOUND_TOMCAT_HOST	Host	@GLOBAL_INSTALL_HOST@
Qxi Servlet	INBOUND_TOMCAT_PASS	Admin Password	mfgpro (encrypted)
Qxi Servlet	INBOUND_TOMCAT_PORT	Tomcat Port	@GLOBAL_TOMCAT_PORT@
Qxi Servlet	INBOUND_WEBAPP_NAME	WebApp Name	qxi
Qxo Database	ADV_DB_CONFIG	Advanced Database Config	false
Qxo Database	CREATE_QXODB_SERVICE	Create Services File Entry	false
Qxo Database	QXODB_BLOCK_SIZE	Block Size	4
Qxo Database	QXODB_DIR	Database Directory	@GLOBAL_QXTEND_DIR@\qxodb

Table B.1 Default QXtend Installation Parameters (Page 2 of 5)

Module	Parameter Name	Description	Default Value
Qxo Database	QXODB_DLC	Progress Directory	@GLOBAL_QXTEND_DLC@
Qxo Database	QXODB_HOST	Host	
Qxo Database	QXODB_PHY	Physical Name	qxodb
Qxo Database	QXODB_SERVICE	Service Name	
Qxo Database	QXODB_SERVICE_PORT	Service Port	
Qxo Database	SA_BEFORE_IMAGE	Before Image	
Qxo Database	SA_CONFIG_DATA	Configuration Data	
Qxo Database	SA_CONFIG_INDEX	Configuration Indexes	
Qxo Database	SA_RAW_MSG	Raw Msgs (CLOB)	
Qxo Database	SA_SCHEMA_AREA	Schema Area	
Qxo Database	SA_SUBSCRIBER_MSG	Subscriber Msgs (CLOB)	
Qxo Database	SA_TRANS_DATA	Transactional Data	
Qxo Database	SA_TRANS_INDEX	Transactional Indexes	
Qxo Events	CREATE_QXEVENTS_SERVICE	Create Services File Entry	false
Qxo Events	MFG_ADMIN_CONNECTION_TYPE	Connection Type	mandatory field
Qxo Events	MFG_ADMIN_DIR	Database Directory	depends on connection type
Qxo Events	MFG_ADMIN_HOST	Host	depends on connection type
Qxo Events	MFG_ADMIN_MULTI_USER	Multi-user	depends on connection type
Qxo Events	MFG_ADMIN_PHY	Physical Name	mandatory field
Qxo Events	MFG_ADMIN_SERVICE	Service Name	depends on connection type
Qxo Events	MFG_DB_CONNECTION_TYPE	Connection Type	mandatory field
Qxo Events	MFG_DB_DIR	Database Directory	depends on connection type
Qxo Events	MFG_DB_MULTI_USER	Multi-user	depends on connection type
Qxo Events	MFG_DB_PHY	Physical Name	mandatory field
Qxo Events	MFG_DB_SERVICE	Service Name	depends on connection type
Qxo Events	MFG_HOST	Host	depends on connection type
Qxo Events	MFGPRO_DB_DLC	Progress Directory	@MFGPRO_DLC@
Qxo Events	QXEVENTS_BLOCK_SIZE	Block Size	4
Qxo Events	QXEVENTS_CLIENT_SERVER	Client Server Mode	false
Qxo Events	QXEVENTS_DB_PHY	Physical Name	qxevents
Qxo Events	QXEVENTS_DIR	Database Directory	@MFG_DB_DIR@

Table B.1 Default QXtend Installation Parameters (Page 3 of 5)

Module	Parameter Name	Description	Default Value
Qxo Events	QXEVENTS_HOST	Host	
Qxo Events	QXEVENTS_SERVICE	Service Name	
Qxo Events	QXEVENTS_SERVICE_PORT	Qxo Events	
Qxo Server	APPSHELL_SOURCE	.NET UI Home Directory	
Qxo Server	APPSRV_ADMIN_SI	AdminServer Name	20931
Qxo Server	APPSRV_ADMIN_UI	AdminServer Name	20931
Qxo Server	APPSRV_MAX_CONN_SI	AppServer Max Connections	5
Qxo Server	APPSRV_MAX_CONN_UI	AppServer Max Connections	5
Qxo Server	APPSRV_NAME_SI	AppServer Name	qxosi_AS
Qxo Server	APPSRV_NAME_UI	AppServer Name	qxoui_AS
Qxo Server	APPSRV_NAMESERVER_SI	Name Server	NS1
Qxo Server	APPSRV_NAMESERVER_UI	Name Server	NS1
Qxo Server	APPSRV_PORT_SI	AppServer Port	-1
Qxo Server	APPSRV_PORT_UI	AppServer Port	-1
Qxo Server	QXOSERVER_DIR	Destination Directory	@GLOBAL_QXTEND_DIR@\qxo server
Qxo Server	QXOSERVER_DLC	Progress Directory	@GLOBAL_QXTEND_DLC@
Qxo Server	QXOSERVER_QXODB_CLIENT_ SERVER	Client Server Mode	false
Qxo Server	QXOSERVER_QXODB_DIR	Directory	@QXODB_DIR@
Qxo Server	QXOSERVER_QXODB_HOST	Host	
Qxo Server	QXOSERVER_QXODB_PHY	Physical Name	@QXODB_PHY@
Qxo Server	QXOSERVER_QXODB_SERVICE	Service Name	
Qxo Server	UPDATE_UBROKER_PROP_SI	Update Ubroker Properties	true
Qxo Server	UPDATE_UBROKER_PROP_UI	Update Ubroker Properties	true
Qxo Servlet	OUTBOUND_TOMCAT_ADMIN	Admin Name	admin
Qxo Servlet	OUTBOUND_TOMCAT_HOME	Home Directory	@GLOBAL_TOMCAT_HOME@
Qxo Servlet	OUTBOUND_TOMCAT_HOST	Host	@GLOBAL_INSTALL_HOST@
Qxo Servlet	OUTBOUND_TOMCAT_PASS	Admin Password	mfgpro (encrypted)
Qxo Servlet	OUTBOUND_TOMCAT_PORT	Port	@GLOBAL_TOMCAT_PORT@
Qxo Servlet	OUTBOUND_WEBAPP_NAME	WebApp Name	qxo
Qxo Servlet	QXO_AS_HOST	AppServer Host	@GLOBAL_INSTALL_HOST@
Qxo Servlet	QXO_AS_NAME	AppServer Name	@APPSRV_NAME_UI@
Qxo Servlet	QXO_AS_NS_PORT	Name Server Port	5162

Table B.1 Default QXtend Installation Parameters (Page 4 of 5)

Module	Parameter Name	Description	Default Value
Qxtend Adapter	ADD_CONN_PARAMS	Additional Connection Params	mandatory field
Qxtend Adapter	APPSRV_NAME_NATIVE	AppServer Name	qadsi_AS
Qxtend Adapter	APPSRV_PORT_NATIVE	AppServer Port	3093
Qxtend Adapter	APPSRV_MAX_CONN_NATIVE	AppServer Max Connections	5
Qxtend Adapter	APPSRV_NAMESERVER_NATIVE	NameServer Name	NS1
Qxtend Adapter	APPSRV_NAMESERVER_PORT_ NATIVE	NameServer Port	5162
Qxtend Adapter	APPSRV_ADMIN_NATIVE	AdminServer Port	20931
Qxtend Adapter	CLIENT_STARTUP_PARAMS	Client Startup Params	-rereadnolock -c 30 -d mdy -yy 1920 -Bt 350 -D 100 -mmax 3000 -nb 200 -s 128 -noshvarfix
Qxtend Adapter	EMPTY_DATABASE_TYPE	Database Type	mandatory field
Qxtend Adapter	FIN_NAME_SERVER_PORT	NameServer Port	5162
Qxtend Adapter	MFG_ADMIN_EMPTY_DB	Admin Empty DB	admempty
Qxtend Adapter	MFG_EMPTY_DB	Empty DB	mfgempty
Qxtend Adapter	MFG_EMPTYDB_DIR	Empty DB Directory	@MFG_HOME@\db
Qxtend Adapter	MFG_HOME	Home Directory	mandatory field
Qxtend Adapter	MFG_PROPATH	Propath	mandatory field
Qxtend Adapter	MFG_SP	Service Pack	mandatory field
Qxtend Adapter	MFG_VERSION	Version	mandatory field
Qxtend Adapter	MFGPRO_DLC	Progress Directory	mandatory field
Qxtend Adapter	QDT_DIR	QDT Directory	mandatory field (EE only)
Qxtend Adapter	QXTEND_ADAPTER_DIR	Destination Directory	@MFG_HOME@\qxtend
Qxtend Adapter	UPDATE_UBROKER_PROP_NATIVE	Update Ubroker Properties	true
Qxtend Adapter	USE_ALTERNATE_DB_SET	Alternate DB Set	false
Standard Install	GLOBAL_DEFAULT_CONFIG	Default Configuration	true
Standard Install	GLOBAL_QXTEND_DIR	Destination Directory	mandatory field
Standard Install	GLOBAL_QXTEND_DLC	Progress Directory	mandatory field
Standard Install	GLOBAL_TOMCAT_HOME	Home Directory	mandatory field
Standard Install	GLOBAL_TOMCAT_PORT	Port	mandatory field
Standard Install	GLOBAL_INSTALL_HOST	Host	mandatory field
Default Config	ENABLE_DEFAULT_CONFIG		false
Default Config	DEFAULT_CONFIG_OUTBOUND_ RECEIVER		QADQXO

Table B.1 Default QXtend Installation Parameters (Page 5 of 5)

Module	Parameter Name	Description	Default Value
Default Config	DEFAULT_CONFIG_EVENT_ SERVICE		ES1
Default Config	DEFAULT_CONFIG_MSG_ PUBLISHER		MP1
Default Config	DEFAULT_CONFIG_MSG_SENDER		MS1
Default Config	DEFAULT_CONFIG_QAD_RECEIVER		QADERP
Default Config	DEFAULT_CONFIG_SOURCE_APP		QADERP
Default Config	CP_LOGIN_PROMPT		login:
Default Config	CP_PASSWORD_PROMPT		Password:
Default Config	CP_SERVER_PROMPT		$
Default Config	CP_CLIENT_SCRIPT		@QXTEND_ADAPTER_DIR@ /scripts/client.qxtend
Default Config	CP_CLIENT_USERNAME		mfg
Default Config	CP_CLIENT_PASSWORD		
Default Config	CP_IS_WINDOWS		false
Default Config	CP_DOMAIN_PROMPT		domain:
Default Config	CP_SERVER_DOMAIN		
Default Config	CP_SERVER_USERNAME		mandatory field
Default Config	CP_SERVER_PASSWORD		mandatory field
QXtend QAD .NET Plug-in	NET_UI_SERVER_TOMCAT_PORT		@GLOBAL_TOMCAT_PORT@
QXtend QAD .NET Plug-in	NET_UI_SERVER_WEBAPP_NAME		qadhome
QXtend QAD .NET Plug-in	NET_UI_SERVER_TOMCAT_HOME		@GLOBAL_TOMCAT_HOME@
QXtend QAD .NET Plug-in	NET_UI_SERVER_ADMIN_NAME		admin
QXtend QAD .NET Plug-in	NET_UI_SERVER_ADMIN_ PASSWORD		mfgpro (encrypted)
QXtend QAD .NET Plug-in	NET_UI_DIRECT_CONNECT_ APPSERVER		false
QXtend QAD .NET Plug-in	NET_UI_APPSERVER_NAME		@APPSRV_NAME_UI@
QXtend QAD .NET Plug-in	NET_UI_APPSERVER_HOST		@GLOBAL_INSTALL_HOST@
QXtend QAD .NET Plug-in	NET_UI_NAME_SERVER_PORT		5162
QXtend QAD .NET Plug-in	NET_UI_APPSERVER_PORT		
QXtend QAD .NET Plug-in	NET_UI_INCLUDE_INBOUND		true
QXtend QAD .NET Plug-in	NET_UI_INCLUDE_OUTBOUND		true

Appendix C

Default Configuration

This section describes QXtend default installation configuration.

Overview

The installation process calls the ant target that creates default configuration inside the newly installed QXtend. This can be enabled through both the Typical and Advanced Install and creates:

Inbound

- Receiver
- UI and SI connection pool for the receiver
- E-mail settings

Outbound

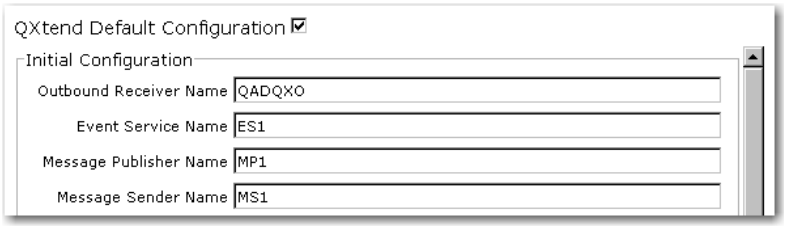
- Source Application
- A subscriber for each domain in the source application
- A default Event Service, Message Publisher, and Message Sender
- E-mail settings

Default Configuration Process

Fig. C.1
Enable Default Configuration in Typical Install

Select the check box and the extra mandatory fields are displayed. For more control over the initial configuration, click the Advanced button. Refer to the table in “Parameters” on page 72 for the parameters used in Default Configuration.

Fig. C.2
Enable Default Configuration in Advanced Install



This screen is displayed after configuring the modules for the installation.

The module itself is run once the QXI Servlet, QXO Database, and QXO Server are successfully installed. It can run from any host because it is a web service call. Once the above has been completed, you can call the configuration service. You will use ant to do this—and return the results to the installer.

After the default configuration has run, there are a few simple steps to take to achieve data synchronization.

1. Inbound will already have the connection pools and receiver with enabled schemas ready to go.
2. Outbound would require the assigning of a source domain (which would be a master domain) for each of the subscribers.
3. Based on the scenario, you would register the required business objects to the message publisher and profiles to subscribers, activate the matching events, and start the services.

Appendix D

Process Control

This section describes QXtend installation process control.

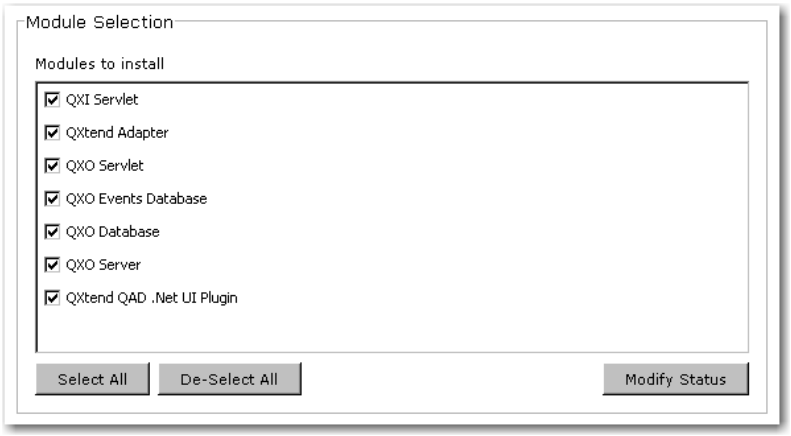
Overview

Process Control provides a higher level of control of the installation that can greatly enhance the efficiency of performing an install.

Using Process Control

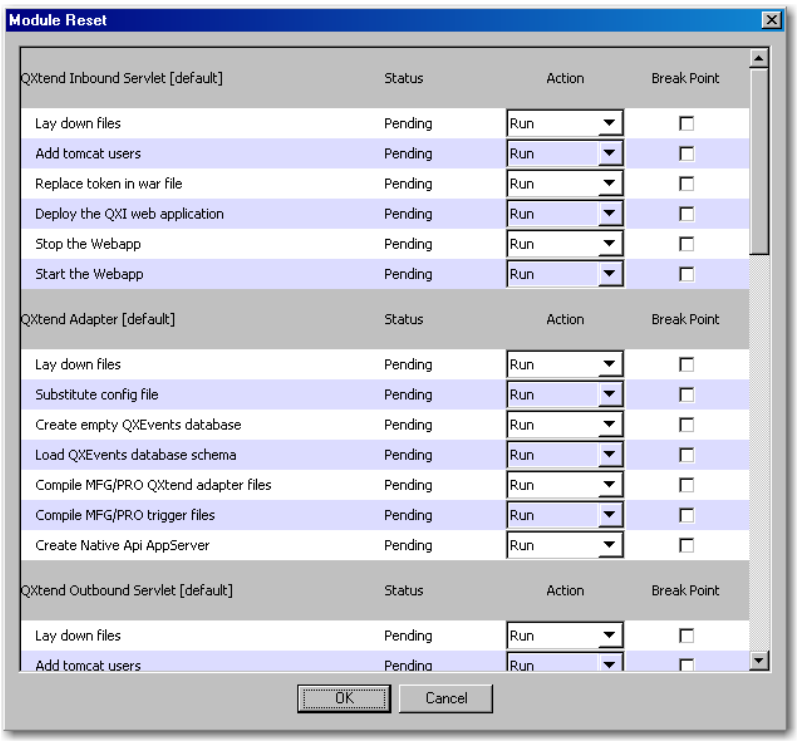
Click the Modify Status button on the Module Selection Screen.

Fig. D.1
Module Selection Screen



The Modify Status pop-up appears.

Fig. D.2
Modify Status Pop-up



Changing the action via the selection lists enables you to run, rerun or skip each of the routines. This may be useful to skip redundant tasks that were run in previous installs, such as adding the Tomcat users, or to rerun a specific routine such as compiling the adapter.

Note All routines are included for a reason; skipping a routine means an install is potentially incomplete and may not work. Only skip routines that are definitely not needed.

Checking the breakpoint boxes enables you to preemptively pause the installer before the routine runs if they want to pay particular attention to a part of the install.

If the installer encounters an error while running, it will pause, allowing you to fix environment problems on the fly. For example if Tomcat is not running when it tries to deploy the Servlets, the installer will pause. You have the chance to investigate through the log file (see Appendix A), see that Tomcat was not started, and start-up Tomcat. The installation continues as if nothing happened.

Fig. D.3
Console Prompt After an Error

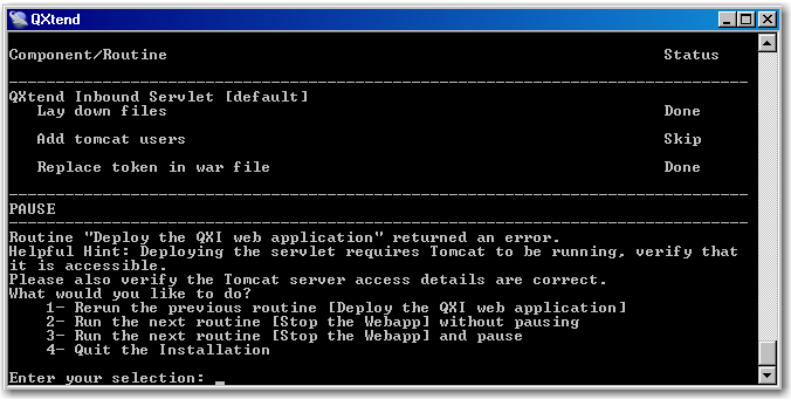
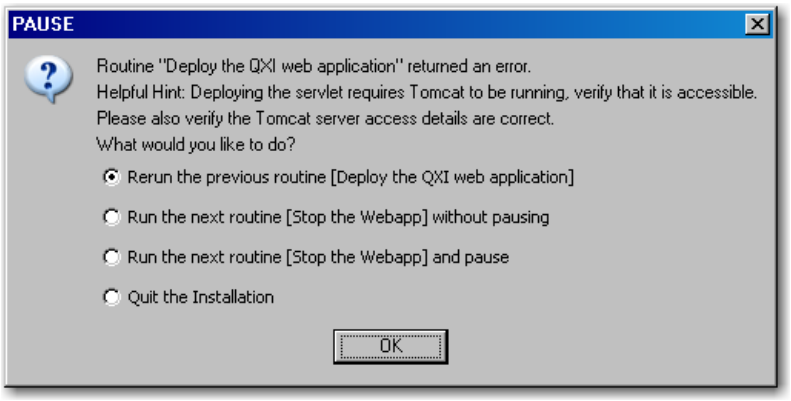


Fig. D.4
GUI Prompt After an Error



Note Errors that you cannot fix are configuration mistakes inside the repository file. For instance, if you forgot to add a custom element to the compile PROPATH, you cannot add it after the installer starts installing and rerun the routine after it fails. You must restart the installer and let it pick up the new value.

Index