Installation Guide
QAD QXtend

78-0952-1.8.5
QXtend 1.8.5
MFG/PRO eB, eB2, eB2.1, QAD 2007, QAD Enterprise Applications Standard and Enterprise Editions
March 2014

Qxtend_IG_v0185.pdf/biw/biw

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

Contents

QAD QXtend Install Guide Change Summary

The following table summarizes significant differences between this document and previous versions.

Date/Version	Description	Reference
March 2014/QXtend 1.8.5	Numerous editorial changes	---
	Corrected NS_LANG to NLS_LANG	page 72
September 2013/QXtend 1.8.4	Numerous editorial changes	---
	Updated Java version in Third-party Requirements	page 9
	Documented Windows 8 considerations	page 10 page 25
	Documented changes to the Global QXtend Panel	page 28
	Added Merge Database Triggers section	page 59
	Documented changes to the Initial Configuration (Instance) Panel	page 83
	Documented changes to the UI Connection Pool Panel	page 83
	Added QXtend Database Triggers appendix	page 93
March 2013/QXtend 1.8.3	Numerous editorial changes	---
	Added Upgrade Databases section	page 15
	Added QXtend Default Configuration GUI section	page 81
September 2012/QXtend 1.8.2	Numerous editorial changes	---
	Documented Oracle support	---
	Updated the Progress OpenEdge 10 version in QXO Server Requirements.	page 8
	Updated the Tomcat version requirements	page 9
	Added Configure Tomcat section	page 14
March 2012/QXtend 1.8	Added steps for the user to verify the .NET UI connmgr and telnet scripts have the QXtend adapter in `PROPATH`.	page 59
September 2011/QXtend 1.7.2	Rebranded for QXtend 1.7.2	---
	Documented Use OpenEdge DBMAN check box	page 29
	Documented Update Conmgr Properties check box	page 29
	Documented QXtend installer reminder feature	page 48
	Added new QXtend Post-migration section	page 60

Chapter 1

QXtend Overview

This section describes concepts to understand before attempting a QAD QXtend installation or upgrade.

Introduction

This chapter provides an overview of QXtend, QXtend deployment, the QAD Deployment Configuration Service, and QXtend installation.

QXtend

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 from 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, see *User Guide: QAD QXtend*.

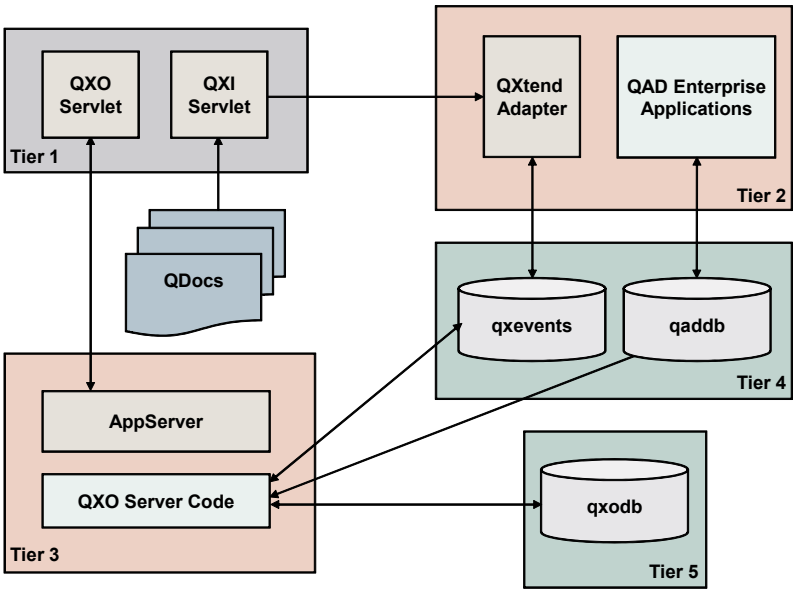
QXtend Deployment

The QAD QXtend application consists of several components. You can deploy these components in various ways on different hosts, platforms, and architectures.

QXtend Deployment Tiers

Figure 1.1 shows the QXI deployment tiers and their relationships.

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 support the QXO administration UI and contains the Tomcat application server.

  Note You can install the Inbound and Outbound servlets 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. Install these components 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 are installed on the same tier for data integrity reasons.
- Tier 5 consists of the qxodb database. This database contains data that the event service extracts from the qxevents database.

The sequence for a single-tier deployment is as follows:

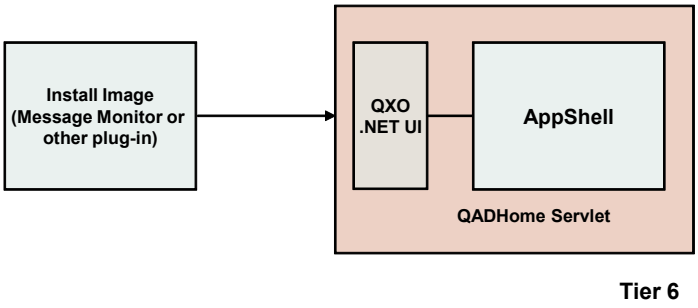
1. QXO servlet (tier 1)
2. qxodb database (tier 5)
3. qxevents and qaddb databases (tier 4)
4. QXO adapter code on the QAD Enterprise Applications Outbound server (tier 2)
5. QXO server code on the Progress AppServer (tier 3)

Note Tiers 2 and 4 (which contain the Adapter and QXEvents DB) can be multiplied according to the number of QAD Enterprise Applications instances installed.

QAD QXtend .NET UI AppShell Plug-in

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

Fig. 1.2
QAD QXtend .NET UI AppShell Plug-in



The QAD QXtend .NET UI AppShell plug-in deployment requires only one tier, consisting of the QXO .NET UI 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).

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

In a single-host environment, all logical tiers are on the same host, and can therefore run at the same time. In a multiple-host environment, hosts physically separate the logical tiers. Run the installer on each host involved in the installation.

QAD Deployment Configuration Service

When installing QXtend, QAD recommends that you use the QAD Deployment Configuration Service (QDCS).

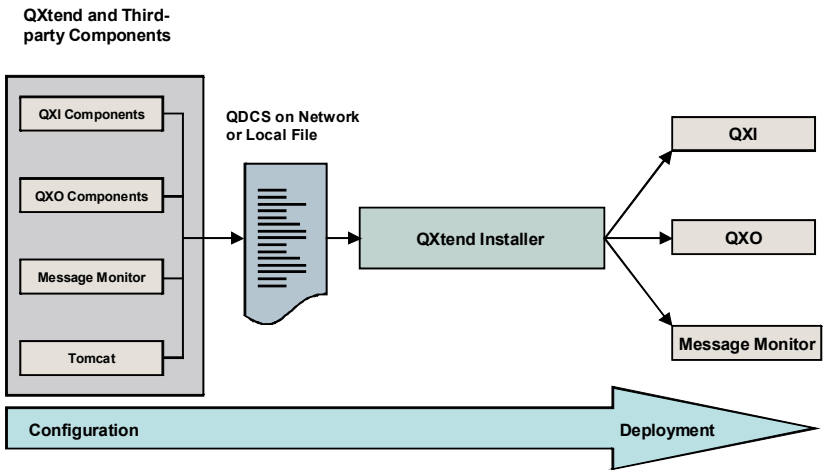
The QDCS is a repository that stores all of your deployment settings for QAD QXtend and its supporting third-party applications (Tomcat, for example) in a single place. Typically, the QDCS is stored on a network for easy access, but it can also be stored in a portable file. The QDCS is populated using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

Note If you are installing in a Linux or 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 one 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 be reused later, for example, to move a QAD QXtend deployment from a test environment into a production environment. See “QDCS Information Hierarchy” on page 5 for more information.

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



An installation using the QDCS consists of the following actions:

1. Specify the location of the QDCS: network or local host.
2. Specify the configuration file name and the environment name for the installation.
3. For each module you select to install (QXtend Adapter or QXI servlet, for example), provide the deployment configuration settings the installer requires. These settings are stored in the QDCS.
4. After you provide all of the settings for the selected components, review the installation summary before proceeding with the installation. The information stored in the QDCS guides the automated installation.

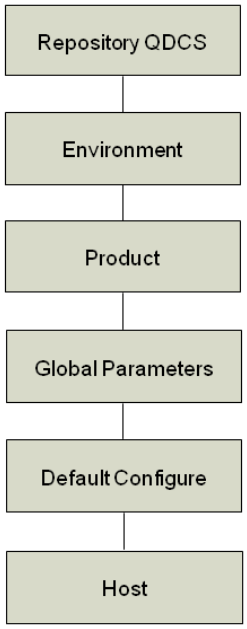
Installing QAD QXtend using the QDCS has the following advantages:

- For QAD QXtend components that share installation parameters (the location of the Progress AppServer, for example), the QDCS allows for the transfer of these settings between hosts. This functionality reduces 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 having to reenter the 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 can 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 allows for the easy movement of product and host deployments between environments.
- Since the QXtend 1.7 release, the hierarchy has only contained two product elements. These elements are for QAD's QXtend and Workflow products.
- Typically, there are several host elements. The elements enable the deployment of QAD QXtend in various configurations. Each host usually contains one or more QAD QXtend components based on component dependencies and organizational deployment requirements.

QXtend Installation

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 the prerequisite components.
2. Install QAD QXtend using the installer.

Chapter 2

System Requirements

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

Overview

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

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

http://support.qad.com

Software Requirements

This section describes the software required for QXtend installation, configuration, and use.

Unzip Utilities

QAD's Download Center provides product downloads as compressed ISO files in `.zip` or `.7z` format. The `.7z` format is used to overcome a limitation that prevents the creation of a `.zip` file larger than 2 GB.

To unzip `.7z` files, use the free 7-Zip utility. Windows, Linux, and UNIX versions of this utility are available from:

http://www.7-zip.org

Unzipping the compressed product file using the appropriate utility yields an uncompressed ISO file.

QXO Server Requirements

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

- A complete Progress OpenEdge 10 (10.2B or higher) or 11 installation including the following components:
  - Enterprise RDBMS
  - OpenEdge Application SVR Enterprise
  - Client Networking
- Java Runtime Environment (JRE), version 1.6 or higher

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, 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. Tomcat is available from:

  http://tomcat.apache.org
- Java 1.6 or higher

  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+:

  http://www.georgiasoftworks.com

Supporting Technologies

QAD QXtend transparently incorporates several web-based technologies. They are listed here to credit the open-source projects that created them.

- Struts is an open source framework for building web applications and is part of the Jakarta Project, sponsored by the Apache Software Foundation. For more information, see:

  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. Log4j is 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 only require Internet Explorer, version 9.0+.

Operating Systems

The QXI and QXO servers support the following platforms:

- Linux (Red Hat and SuSE)
- UNIX (HP-UX, HP-Tru64, Sun Solaris SPARC, Compaq UNIX Tru64, and IBM AIX)
- Windows (including Windows 2000, 2002 Server, 2003 Server, XP, and 64-bit variants)

Note To install QXtend on Windows 8 systems, you must run the QXtend installer in Compatibility Mode. This is described in “Windows 8 Environment” on page 25.

Installation User Account

To avoid potential access problems, carefully select the user account used to install QXtend and the start environment. An easy way to prevent these issues is to use an administrator (rather than root) account to perform the entire installation. This method 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 for QAD QXtend installation.

Overview

Before installing QAD QXtend, you perform the following tasks:

- Install the Windows Telnet Server (Windows only)
- Install the Tomcat Application Server.
- Configure Tomcat
- Prepare the Oracle Database (optional)
- Complete the Deployment Worksheet (optional)

Install the Windows Telnet Server

For telnet connections on Windows servers, install the Georgia SoftWorks Telnet Server (GSWTS). You can download the latest version from the Georgia SoftWorks web site at:

http://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 the 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.

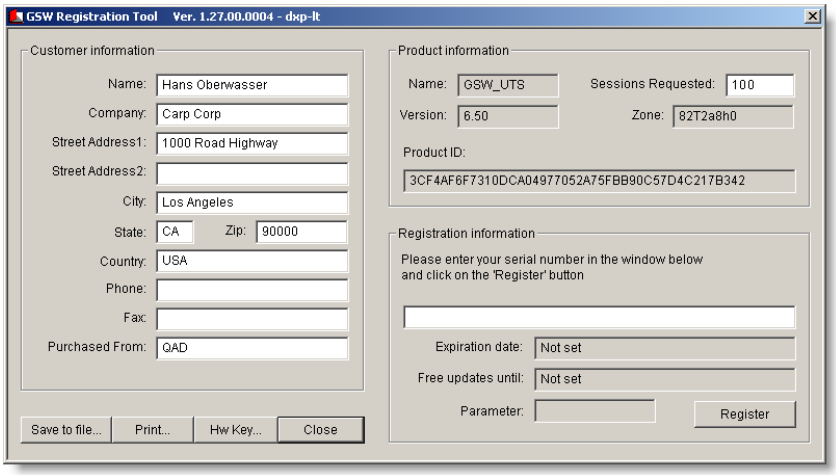
Register the Georgia SoftWorks Software

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 the 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. Enter your customer information in the Georgia SoftWorks Product Registration window. The information displayed in the Product Information section is system generated.
3. Set Sessions Requested to 100. This value 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 send the product ID to Georgia SoftWorks by e-mailing the saved registration form file to registration@georgiasoftworks.com or faxing the printed registration form to 706-265-1020. After Georgia SoftWorks receives your form, a serial number is generated for your product and sent to you.

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

Configure Georgia SoftWorks

You manually create the Georgia SoftWorks startup batch file, `k_start.bat`. 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. Separate user directories under the `gs_uts\scripts` identify each file.

For this second case, there are multiple ways to perform the setup. See the Georgia SoftWorks documentation that downloads with the product in the `\docs` directory for details.

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=6
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 previous file sample 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 (http://tomcat.apache.org).

Configure Tomcat

Add the new role `qadadmin` in `TOMCAT_HOME/conf/tomcat-users.xml` as follows:

```
<role rolename="qadadmin"/>
```

For Tomcat version 6.0.29 and before, make sure that the `manager` and `qadadmin` roles are associated with the administrator user. The final result looks like:

```
<user password="xxx" username="admin" roles="manager,qadadmin"/>
```

For Tomcat version 6.0.30 and later, make sure that the `manager-gui`, `manager-script`, and `qadadmin` roles are associated with the administrator user in the same file. The final result looks like:

```
<user password="xxx" username="admin" roles="manager-gui,manager-
script,qadadmin"/>
```

Prepare Oracle Database

If QAD Enterprise Applications is running on an Oracle database, perform the following procedure as the Oracle DBA to prepare the database for QXtend installation:

1. Use the following command to make the `$ORACLE_HOME/network/admin/tnsnames.ora` file readable to all users:

   ```
   chmod o+r $ORACLE_HOME/network/admin/tnsnames.ora
   ```

2. Create an Oracle Net Service for QXtend using the `netca` command.

3. Beginning with QAD Enterprise Edition 2012.1, tablespaces QXEVENTS and QXEVENTS_IDX for QXEvents database are created automatically when you install QAD Enterprise Applications. If they do not exist, create the QXEvents tablespace with database admin. Use the following SQL commands as an example (`/dbs/ueeora` is the Oracle Data Directory).

   ```
   CREATE TABLESPACE "QXEVENTS" DATAFILE
   '/dbs/ueeora/qxevents_ORAU.dbf' SIZE 5242880
   AUTOEXTEND ON NEXT 8192 MAXSIZE 32767M
   LOGGING ONLINE PERMANENT BLOCKSIZE 8192
   EXTENT MANAGEMENT LOCAL AUTOALLOCATE SEGMENT SPACE MANAGEMENT AUTO;

   CREATE TABLESPACE "QXEVENTS_IDX" DATAFILE
   '/dbs/ueeora/qxevents_ORAU_idx01.dbf' SIZE 5242880
   AUTOEXTEND ON NEXT 8192 MAXSIZE 32767M
   LOGGING ONLINE PERMANENT BLOCKSIZE 8192
   EXTENT MANAGEMENT LOCAL AUTOALLOCATE SEGMENT SPACE MANAGEMENT AUTO;
   ```

4. Create the QXO database user. In the following example, the user name is qxodb and the password is qxodb.

   ```
   CREATE USER qxodb IDENTIFIED BY qxodb;
   GRANT CONNECT, RESOURCE TO qxodb;

   GRANT CREATE SESSION TO qxodb;
   ```

```
GRANT SELECT ON "SYS"."ARGUMENT$" TO qxodb;
GRANT SELECT ON "SYS"."COL$" TO qxodb;
GRANT SELECT ON "SYS"."COM$" TO qxodb;
GRANT SELECT ON "SYS"."CON$" TO qxodb;
GRANT SELECT ON "SYS"."DUAL" TO qxodb;
GRANT SELECT ON "SYS"."ICOL$" TO qxodb;
GRANT SELECT ON "SYS"."IND$" TO qxodb;
GRANT SELECT ON "SYS"."LINK$" TO qxodb;
GRANT SELECT ON "SYS"."OBJ$" TO qxodb;
GRANT SELECT ON "SYS"."PROCEDURE$" TO qxodb;
GRANT SELECT ON "SYS"."SEQ$" TO qxodb;
GRANT SELECT ON "SYS"."SYN$" TO qxodb;
GRANT SELECT ON "SYS"."TAB$" TO qxodb;
GRANT SELECT ON "SYS"."TS$" TO qxodb;
GRANT SELECT ON "SYS"."USER$" TO qxodb;
GRANT SELECT ON "SYS"."VIEW$" TO qxodb;
```

5 Beginning with QAD Enterprise Edition 2012.1, tablespaces QXODB, QXODB_IDX, and QXODB_LOB for QXO database are created automatically when you install QAD Enterprise Applications. If they do not exist, create the QXO tablespace. Use the following SQL commands as an example (`/dbs/ueeora` is the Oracle Data Directory).

```
CREATE TABLESPACE "QXODB" DATAFILE
'/dbs/ueeora/qxodb_ORAU.dbf' SIZE 5242880
AUTOEXTEND ON NEXT 8192 MAXSIZE 32767M
LOGGING ONLINE PERMANENT BLOCKSIZE 8192
EXTENT MANAGEMENT LOCAL AUTOALLOCATE SEGMENT SPACE MANAGEMENT AUTO;

CREATE TABLESPACE "QXODB_IDX" DATAFILE
'/dbs/ueeora/qxodb_ORAU_idx01.dbf' SIZE 5242880
AUTOEXTEND ON NEXT 8192 MAXSIZE 32767M
LOGGING ONLINE PERMANENT BLOCKSIZE 8192
EXTENT MANAGEMENT LOCAL AUTOALLOCATE SEGMENT SPACE MANAGEMENT AUTO;

CREATE TABLESPACE "QXODB_LOB" DATAFILE
'/dbs/ueeora/qxodb_lob_ORAU.dbf' SIZE 5242880
AUTOEXTEND ON NEXT 8192 MAXSIZE 32767M
LOGGING ONLINE PERMANENT BLOCKSIZE 8192
EXTENT MANAGEMENT LOCAL AUTOALLOCATE SEGMENT SPACE MANAGEMENT AUTO;
```

Upgrade Databases

If you are going to use a newer version of Progress OpenEdge when upgrading QXtend, follow the standard instructions for upgrading the QXO database before running the installer.

The steps are:

1 Shut down everything related to QXtend.

2 Back up the QXEvents and QXO databases.

3 Truncate the `bi` file for live and empty databases:

```
proutil <db-name> –C truncate bi
```

For example, to convert the OE10 databases to OE11, use `proutil db-name –C conv1011` (add `–cpinternal UTF-8` for unicode databases). See the Progress documentation and release notes for further information (http://www.progress.com/support).

4 Upgrade databases. The QXEvents database must use the same version of Progress OpenEdge as QAD Enterprise Applications. If you upgraded the QAD EA databases, you must upgrade the QXEvents database before upgrading QXtend.

Deployment Worksheet

QAD recommends that you use the following worksheet to record information such as server names and locations, port numbers, and other settings before you start an installation. You enter these settings during installation configuration. For details, see “Installing, Upgrading, and Migrating QAD QXtend in a GUI Environment” on page 23.

Table 3.1
Deployment Worksheet

Group	Description	Setting
General Install Configuration	Use QDCS network service? (Y/N) (Go to Local parameter file name if not using network service)	
	QDCS Tomcat host	
	QDCS Tomcat port	
	QDCS WebApp name	
	QDCS Tomcat admin user	
	QDCS Tomcat admin password	
	Local parameter file name (if not using QDCS)	
	Install log file	
	Environment name	
Tomcat	Tomcat home	
	Tomcat host	
	Tomcat admin user	
	Tomcat admin password	
	Tomcat port	
	Inbound WebApp name	
	Outbound WebApp name	
Migrate Inbound Configuration	Previous Inbound WebApp name	
	Tomcat directory	
	Previous Inbound version	

Table 3.1 — *Deployment Worksheet* (Page 1 of 4)

Group	Description	Setting
MFG/PRO	MFG/PRO version	
	MFG/PRO SP	
	MFG/PRO home	
	MFG/PRO PROPATH (xrc directory)	
	MFG/PRO database type (Progress, Oracle, or Other)	
	MFG/PRO database UTF	
	MFG/PRO database internal code page	
	MFG/PRO client stream codepage	
	MFG/PRO empty database directory	
	MFG/PRO qaddb empty database physical name	
	MFG/PRO admin empty database physical name	
	Additional connection parameters	
	Dataserver connection parameters (if data type is not Progress)	
	Progress for MFG/PRO	
	QDT directory (if applicable)	
	QDT environment name	
	Financial AppServer	
	Financial NameServer port	
	.NET UI home directory	
QXtend Adapter	QXtend adapter destination directory	
	Client startup parameters	
Adapter Native API AppServer	Native API AppServer Name	
	Native API AppServer port	
	Native API AppServer max connections	
	Name server	
	Name server port	
	AdminServer port	
MFG/PRO Databases	MFG/PRO database host	
	MFG/PRO database directory	
	MFG/PRO database physical name	
	MFG/PRO database service name	
	MFG/PRO admin database physical name	
	MFG/PRO admin database directory	
	MFG/PRO admin database host	
	MFG/PRO admin database service name	
	MFG database start script	
	MFG database connection pf file	

Table 3.1 — *Deployment Worksheet* (Page 2 of 4)

Group	Description	Setting
QXEvents Database	QXEvents database physical name	
	QXEvents database directory	
	QXEvents database host name	
	QXEvents database service name	
	QXEvents database service port	
	QXEvents database block size	
	Progress directory for MFG/PRO databases	
Migrate QXEvents	Source QXEvents database physical name	
	Source QXEvents connection type	
	Source QXEvents directory	
	Source QXEvents host	
	Source QXEvents service name	
	Source QXEvents version	
Outbound Database	QXODB host name	
	QXODB destination directory	
	QXODB physical name	
	QXODB service name	
	QXODB Block Size	
	Progress for Outbound database	
Outbound Database (advanced parameters)	Before image	
	Schema area	
	Configuration data	
	Configuration indexes	
	Transactional data	
	Transactional indexes	
	Raw messages (CLOB)	
	Subscriber messages (CLOB)	
Migrate QXODB	Source QXODB database physical name	
	Source QXODB connection type	
	Source QXODB directory	
	Source QXODB host	
	Source QXODB service name	
	Source QXODB version	

Table 3.1 — *Deployment Worksheet* (Page 3 of 4)

Group	Description	Setting
Outbound Server	Outbound Server destination directory	
	Progress for Outbound server	
	Outbound UI AppServer name	
	Outbound UI AppServer port	
	Outbound UI AppServer maximum connections	
	Outbound SI AppServer name	
	Outbound SI AppServer port	
	Outbound SI AppServer maximum connections	
	AdminServer port	
	Name server	
.NET UI Plug-in	.NET UI version	
	.NET UI WebApp name	
	.NET UI Tomcat Port	
	.NET UI Tomcat Home	
	.NET UI configuration name	
	Include Inbound menu entry (Y/N)	
	Include Outbound menu entry (Y/N)	
Oracle	Home Directory	
	System ID (SID)	
	`NLS_LANG`	
	Logical Name	
	Connection Parameters	
	QXtend Net Service Name	
	QXO Oracle User Name	
	QXO Oracle User Password	
	QXEvents Oracle User Name	
	QXEvents Oracle User Password	

Table 3.1 — *Deployment Worksheet* (Page 4 of 4)

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, which supports GUI and character-based installs, upgrades, and migrations.

The QAD QXtend installation, upgrade, and migration options are as follows:

- 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, and Migrating QAD QXtend in a Character Environment” on page 49.

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

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

Installer Limitations

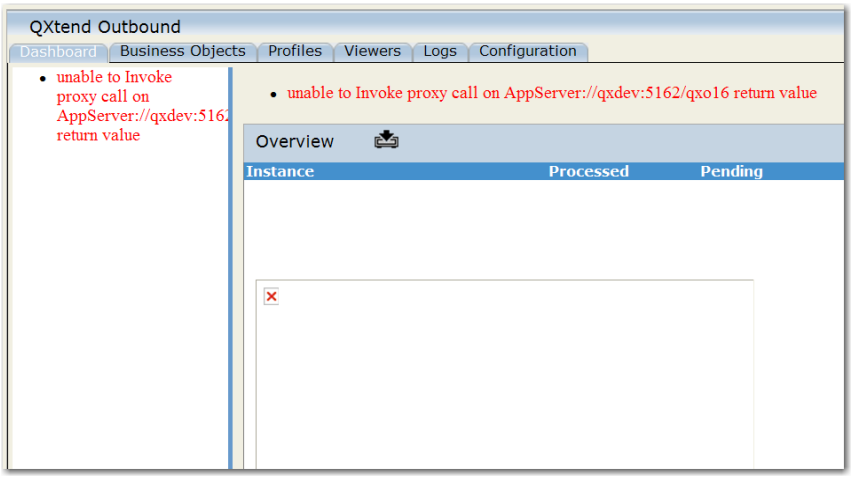
Installation or upgrade does not update the startup scripts for QAD Enterprise Applications. You manually update the scripts to include the locations of the newly installed components. For example, you include the new qxevents database in the QAD Enterprise Applications database set.

When migrating the QXO database configuration, the qxevents connection details are not automatically updated in the source application to point to the new version’s qxevents database installation. Before you attempt 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 cannot connect to it at upgrade time.

If a qxevents connection problem occurs, the Dashboard in the QXO UI does not display. The screen looks similar to Figure 4.1.

Fig. 4.1
QXEvents Connection Error Indication



Also these messages 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 can 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, Upgrades, and Migrations,” on page 61. If you still experience problems, contact QAD Support for assistance.

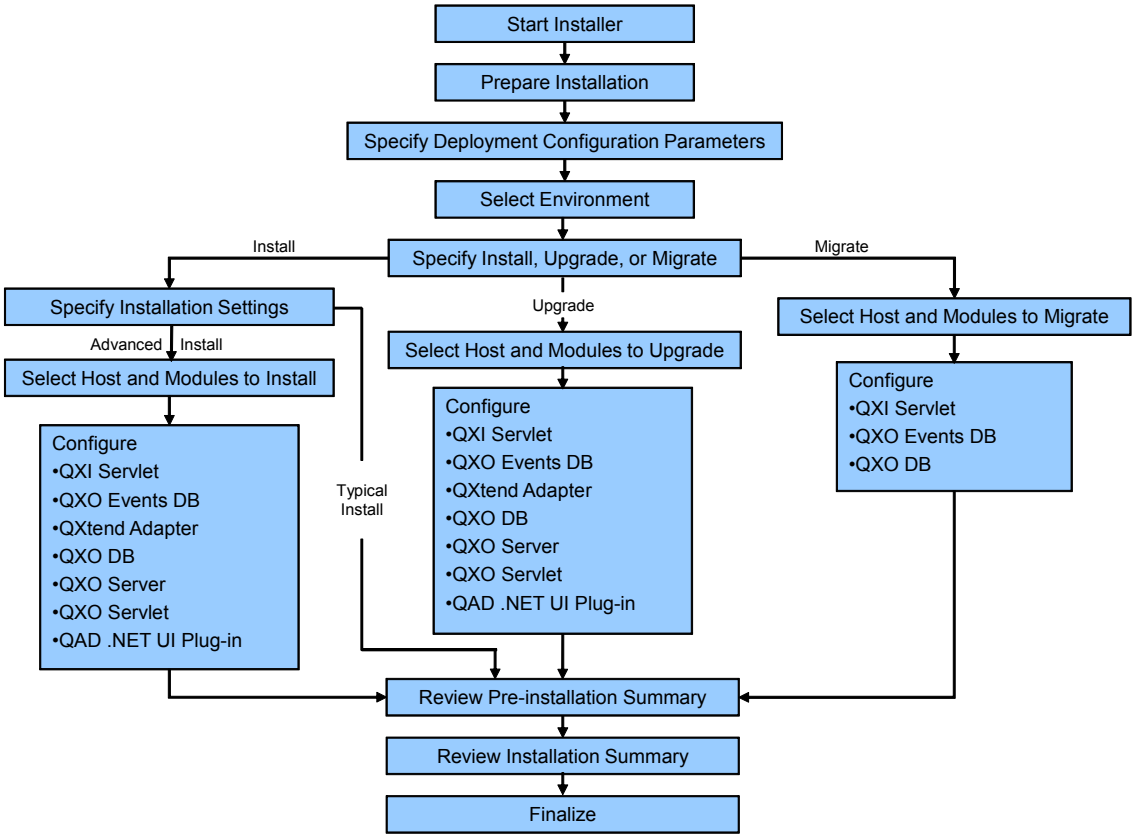
Installing, Upgrading, and Migrating QAD QXtend in a GUI Environment

This section describes the installation, upgrade, and migration of QAD QXtend 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 following sequence applies when all components are installed or upgraded on a single host. The deployment sequence varies depending on your configuration.

Fig. 4.2
QXtend Install, Upgrade, and Migration Workflows



Prerequisites

Before installing, upgrading, or migrating QAD QXtend, do the following:

1. QAD recommends that you shut down any virus protection programs.
2. Verify that Tomcat is running.
3. Verify that you have the appropriate folder permissions to perform the install, upgrade, or migration.
4. To use the GUI installer from an X-Windows session in a Linux or UNIX environment, set the `DISPLAY` variable using the syntax:

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

   For example,

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

5. If you are installing QXtend from QAD physical product media, go to “Start the Installer” on page 25.

   Otherwise, download the ISO file from the QAD Download Center or QAD Store, unzip it using the appropriate utility (see “Unzip Utilities” on page 8), and go to “Start the Installer” on page 25.

Start the Installer

1 Insert the release media into the CD-ROM drive or mount the release media image on your file system.

2 Navigate to the executable files. The executable files for each environment type are located under `Disk1/InstData/<env_type>/[No]VM/QXtend.[bin|exe]`.

Note If you choose to copy the files from the release media, make sure that the directory structure remains the same, including the folder called `Disk 1`. If the structure changes, the installer cannot run. The directory structure is as follows:

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

Windows 8 Environment

To install QXtend on Windows 8 systems, you must configure the QXtend installer to use Compatibility Mode. Use the following steps to configure the installer to use Compatibility Mode:

1 Right-click `QXtend.exe`.

2 Select Properties. The QXtend Properties dialog box appears.

3 Click the Compatibility tab.

4 Select the "Run this program in compatibility mode for” check box.

5 Select Windows 7 from the drop-down menu.

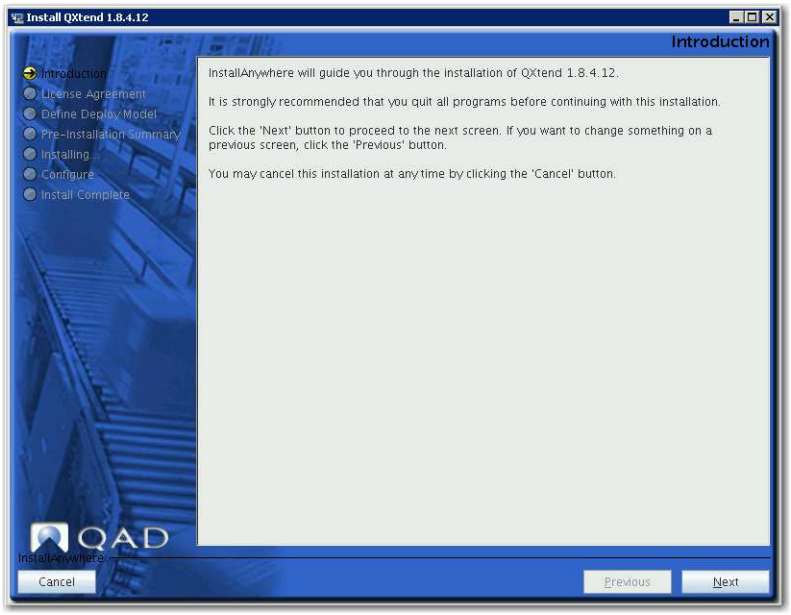
6 Click OK. The QXtend Properties dialog box closes.

Prepare the Installation

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

The installer is extracted and the Introduction screen displays.

Fig. 4.3
Introduction Screen



8 Click Next. The License Agreement screen displays.

9 Scroll to the end of the license agreement.

10 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.

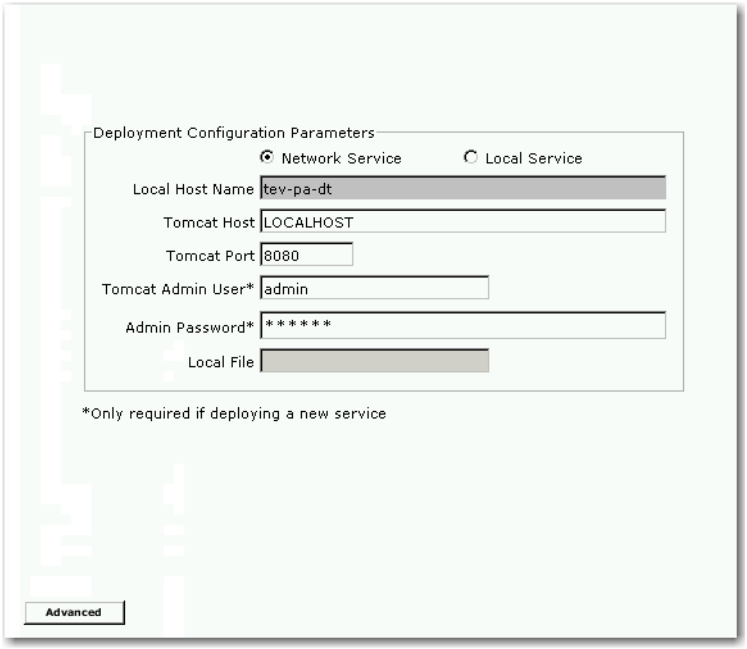
11 Accept the default location for the install log files (`C:\instlog`), or enter a different path.

12 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 on your network.

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

Local Service. Select this option to use a QDCS file on your local machine. Using this option disables all fields except Local File.

Local Host Name. This read-only field displays the name of the current machine.

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

Tomcat Port. Enter the Tomcat port number (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 file name on the local machine containing the parameter settings (for Local Service only).

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

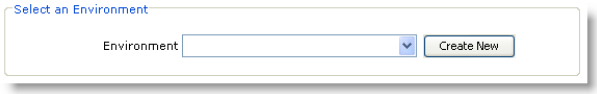
13 A pop-up window may display that says a QAD Deployment Configuration Service (QDCS) was not detected on a host. This means that the QDCS is not deployed on the Tomcat server specified or the wrong server was specified. Click Yes if you want the installer to deploy the QDCS.

14 Click Next. The Environment Selection screen displays.

Select Environment

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

Fig. 4.5
Select an Environment



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

Specify Install, Upgrade, or Migrate

To perform an installation, verify Install is selected, click Next, and continue to “Installing QXtend” on page 28.

Fig. 4.6
Specify Install, Upgrade, or Migrate

To upgrade components of an existing QXtend installation to the current QXtend release, select Upgrade, click Next, and go to “Upgrading QXtend” on page 41.

To migrate the existing application data from a previous QXtend installation to the current QXtend release, select Migrate, click Next, and see “Migrating QXtend” on page 44.

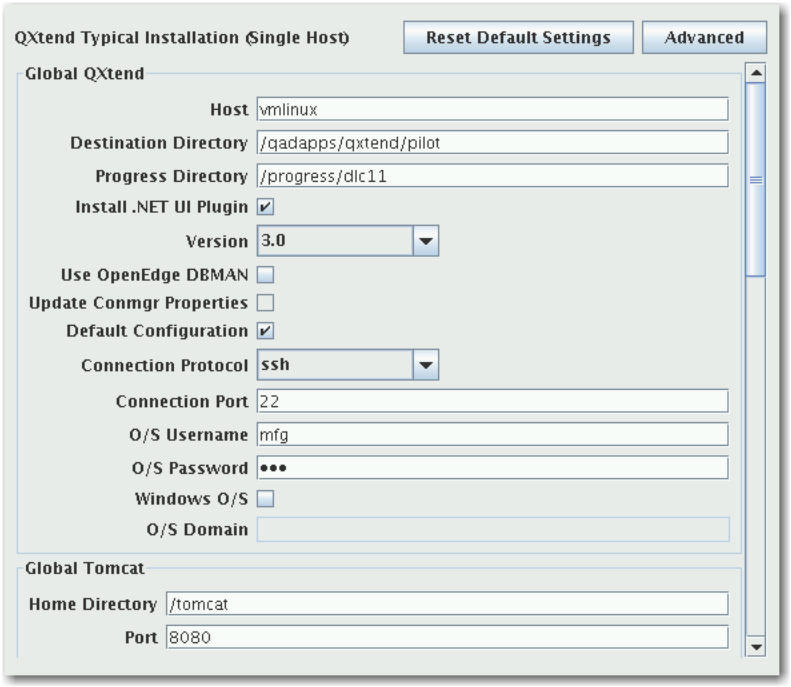
Note You must have completed a successful installation before attempting to upgrade the QXI servlet, QXO database, or QXO events database components.

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, see “Tokens” on page 72.

Fig. 4.7
QXtend Installation Settings



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

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

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

Install .NET UI Plugin. Select this check box if you want to install the .NET UI plug-in.

Version. Select the .NET UI version from the drop-down list.

Use OpenEdge DBMAN. Select this check box to use DBMAN to manage the QXO and qxevent databases. This box is enabled only if you specify QAD EE 2011.1 or later.

Update Conmgr Properties. Select this check box to update the `$DLC/properties/conmgr.properties` file using the installer. Not selecting this box allows you to update the file manually. This box is only enabled if Use OpenEdge DBMAN is selected.

Default Configuration. Select this check box to load QXtend with an initial set of default configuration parameters. See Appendix B, “Default Configuration,” on page 79 for more information.

Connection Protocol. The connection protocol for the UI connection pool. The options are telnet and ssh.

Connection Port. The connection port for the UI connection pool. The default values are 23 for telnet and 22 for ssh.

O/S Username. The user name for the destination host operating system.

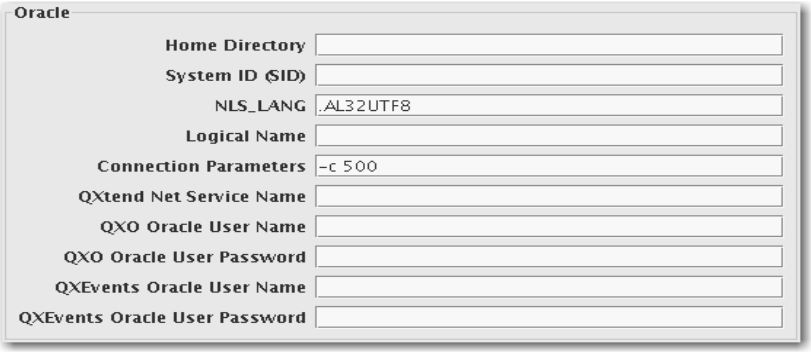
O/S Password. The password for the destination host operating system.

Windows O/S. Select this box if the destination host uses the Windows operating system.

O/S Domain. The name of the domain to which the destination host belongs. The field is only enabled if the destination host operating system is Windows.

If the QAD Enterprise Application is running on an Oracle database, and you select Oracle as the Database Type in the MFG/PRO panel, a new panel (shown in the following figure) appears at the bottom of Typical Installation.

Fig. 4.8
Oracle Settings



Enter the necessary information in the Oracle screen:

Home Directory. The home directory for the Oracle database.

System ID (SID). The system ID for the Oracle database.

NLS_LANG. The `NLS_LANG` environment variable setting for the Oracle database.

Logical Name. The logical name of the Oracle database.

Connection Parameters. Additional Oracle database connection parameters.

QXtend Net Service Name. The Oracle net service name for QXtend.

QXO Oracle User Name. The Oracle user name for the QXO database.

QXO Oracle User Password. The Oracle user password for the QXO database.

QXEvents Oracle User Name. The Oracle user name for the QXEvents database.

QXEvents Oracle User Password. The Oracle user password for the QXEvents database.

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 for a default QXtend installation. For detailed information regarding default installation parameters, see Appendix A, “Typical Installation Parameters,” on page 71.

Advanced Installation allows you to individually enter the parameters to create a custom QXtend configuration. You are also able to add additional QXtend adapters and QXO Events Databases according to the number of QAD Enterprise Application installations. Finally, if you want to modify any of the default configuration parameters, use QXtend’s advanced installation mode.

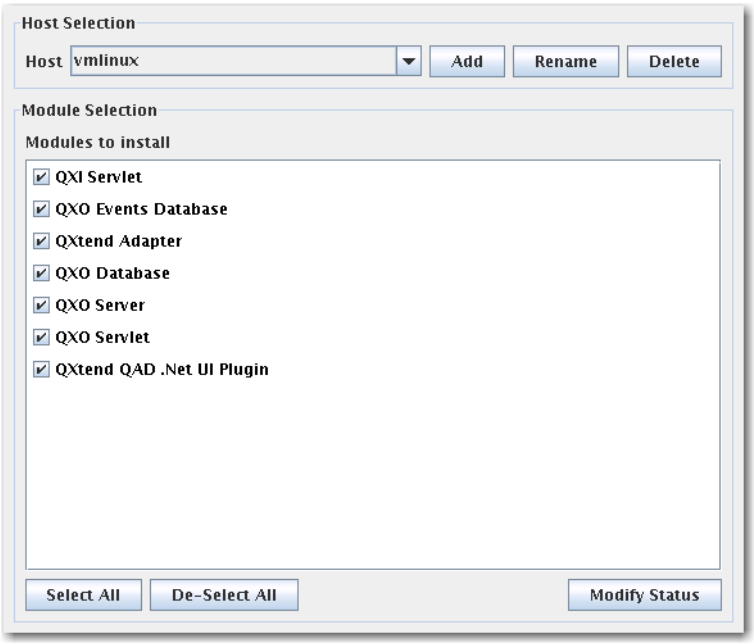
To continue with a typical installation, click Next and go to “Review the Pre-installation Summary” on page 47.

To perform a custom installation, click Advanced and go to “Advanced Installation” on page 31.

Advanced Installation

Select the Host and Modules to Install

Fig. 4.9
Choose the Host and Install Set



1 Enter the host on which to install the components in the Host Selection panel.

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

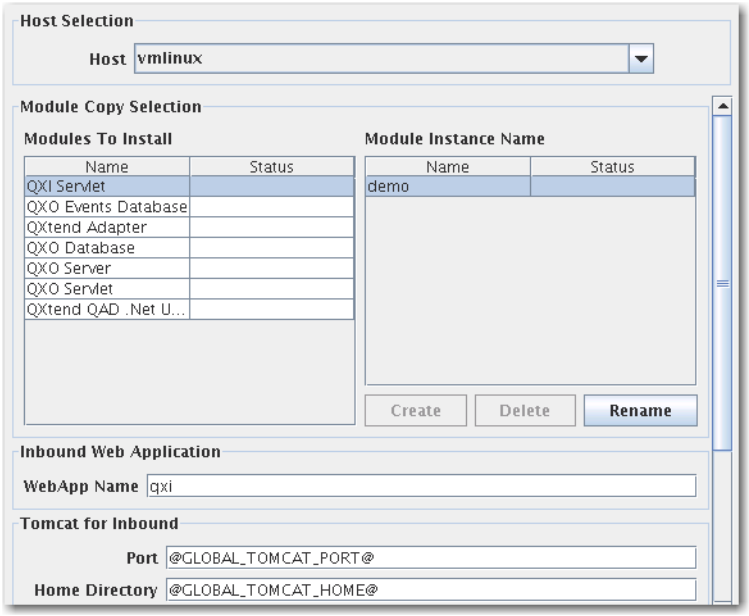
3 In the Module Selection panel, select the components to install on the specified host.

Note To install QXI only, select the QXI servlet and adapter code components. To install QXO only, select all of the QXO components and the adapter code component.

Note Under each environment (“QDCS Information Hierarchy” on page 5 and “Select Environment” on page 28), you can select each component once across all hosts that make up the configuration except the QXtend Adapter and QXO Events Database. This functionality allows you to configure against multiple QAD Enterprise Applications installations.

4 Click Next to display the Module Copy Selection screen.

Fig. 4.10
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 of 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, 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. This functionality allows you to configure against multiple QAD Enterprise Applications installations.

b Assign the required components to an instance. Select the check box for a component if its corresponding instance is highlighted.

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

You cannot select a different component until you have entered all of the required information for the currently selected component.

Note Enter information in the fields highlighted in orange.

Configure the QXI Servlet

6 In the Inbound Web Application panel, enter the application name known to the Tomcat application server.

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

Port. Enter the Tomcat port.

Home Directory. Enter 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 application name known to 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 35 on page 40.

Configure the QXO Events Database

10 Complete the required fields in the MFG/PRO version panel.

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

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

Database Type. Select Progress, Oracle, or Other, depending on the type of database that QAD Enterprise Applications uses. Other refers to non-Progress and non-Oracle databases.

Database UTF. Whether the database codepage is UTF.

11 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 Local or Network.

Multi-user. Select this option if your database is operating in multi-user mode. If you use this option, the database must be running during QXtend installation. If you do not use this option, shut down the database before installing QXtend.

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

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

Service Name. Enter the network service that the QAD Enterprise Applications admin database uses.

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

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

Connection Type. Select Local or Network.

Multi-user. Select this option if your database is operating in multi-user mode. If you use this option, the database must be running when you install QXtend. If you do not use this option, shut down the database before installing QXtend.

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

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

Service Name. Enter the network service that the QAD Enterprise Applications admin database uses.

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

Physical Name. Enter the qxevents database physical name. If the Database Type is Oracle, this field has a fixed value of `qxevents_sh` and cannot be edited.

Database Directory. Enter the qxevents database directory.

Connect Using Client/Server. Select this check box to use client/server mode to connect to the database.

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

Service Name. The name the installer enters into the services file. Otherwise, enter the name of the network service that this database uses.

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 that this network service uses.

MFG DB Start Script. Enter the location of the script that starts the QAD Enterprise Applications databases.

MFG DB Conn.pf. Enter the full path (including the file name) of the connection parameter file that the client session uses.

Block Size. Enter the block size to use for the qxevents database. The value depends on the operating system block size.

Progress Directory. Enter the Progress location the Progress AppServer uses.

14 If you are migrating an existing qxevents database to the new install, complete the following required fields in the QXEvents Migrate Configuration panel (see “Upgrading QXtend” on page 41 for more details). These fields may already be populated.

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 to connect in multi-user mode. If you use this option, the database must be running during QXtend installation. If you do not use this option, shut down the database before installing QXtend.

Database Directory. Enter the source qxevents database directory. The entry only applies 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 ServiceName or service port number to use.

Version. Select the version being migrated.

15 If necessary, select the next component to configure from the Modules to Install panel. Otherwise, go to step 35 on page 40.

Configure the QXtend Adapter

16 Complete the required fields in the MFG/PRO panel.

Version. Use the drop-down list to select the QAD Enterprise Applications version that 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 MFG/PRO source code directory (`xrc` directory) to use during QXtend Adapter compilation.

Database Type. Select Progress, Oracle, or Other, depending on the type of database that QAD Enterprise Applications uses. Other refers to non-Progress and non-Oracle databases.

Database UTF. Whether the database codepage is UTF.

Internal Codepage. Enter the name of the codepage used in memory.

Stream Codepage. Enter the name of the codepage used for stream I/O.

Use Alternate DB Set. Enable this option to specify non-standard empty QAD databases. If this option is unchecked, skip steps 17 and 18 because the MFG/PRO Database and MFG/PRO Admin Database panels are disabled.

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 to compile the QXtend adapter.

Dataserver Connection Params. Used if the database type is Other. Specify the parameters to use to connect to the database through Progress.

Progress Directory. Enter the location of Progress that this instance of the QXtend adapter uses. This is the same Progress version required by QAD Enterprise Applications.

QDT Directory. Enter the location of the QDT. The location is only applicable if you are installing against QAD Enterprise Edition.

Environment Name. Enter the configuration name that was specified in QDT.

Financial AppServer. Enter the name of the AppServer for Financials. The name is only applicable if you are installing against QAD Enterprise Edition.

Financial NameServer Port. Enter the NameServer port that the Financials AppServer uses. The port is only applicable if you are installing against QAD Enterprise Edition.

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

17 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 Local or Network.

Multi-user. Select this option if your database is operating in multi-user mode. If you use this option, the database must be running during QXtend installation. If you do not use this option, shut down the database before installing QXtend.

Database Directory. The database where the QAD Enterprise Applications qaddb database is installed. Enter the directory if the connection type is Local.

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

Service Name. Enter the network service that the QAD Enterprise Applications qaddb uses.

18 Complete the required fields in the MFG/PRO Admin Database panel.

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

Connection Type. Select 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 running during QXtend installation. If you do not use this option, shut down the database before installing QXtend.

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

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

Service Name. Enter the network service that the QAD Enterprise Applications admin database uses.

19 Complete the required fields in the QXtend Adapter panel.

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 QAD Enterprise Applications.

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

Note This information only applies to eB2.1 SP4 and later.

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

AppServer Port. Enter the AppServer port number.

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

Name Server. Enter the NameServer name to which the AppServer belongs.

Name Server Port. Enter the NameServer port that the AppServer uses.

AdminServer Port. Enter the AdminServer port that the AppServer uses.

Update Ubroker Properties. Select this check box for the installer to update the Ubroker Properties. Ensure that the installer has the permissions required to update the file. If you do not select the check box, you can create the entry manually. You can find the template under the properties subdirectory of the QXO Adapter directory.

21 If necessary, select the next component to configure from the Modules to Install panel. Otherwise, go to step 35 on page 40.

Configure the QXO Database

22 Complete the required fields in the Outbound Database panel.

Physical Name. Enter the physical name of the QXO database. If the Database Type is Oracle, this field has a fixed value of `qxodb_sh` and cannot be edited.

Database Directory. Enter the QXO database location.

Service Name. The installer enters the name into the services file. 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. The installer enters the port into the services file. Otherwise, enter the port number for the network service.

Block Size. Enter the QXO database block size.

Progress Directory. Enter the location of the Progress directory this database uses.

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

Selecting this box displays the following fields:

Before Image. The location of the BI file.

Schema Area. There are no tables in this area.

Configuration Data. A low-volume, low-capacity area.

Configuration Indexes. A low-volume, low-capacity area.

Transactional Data. A high-volume, medium-capacity area.

Transactional Indexes. A high-volume, medium-capacity area.

Raw Msgs (CLOB). A high-volume, high-capacity area.

Subscriber Msgs (CLOB). A high-volume, high-capacity area.

23 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 running when you install QXtend. If you do not use this option, shut down the database before installing QXtend.

Database Directory. Enter the directory for the source QXODB database. The entry only applies 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.

24 If necessary, select the next component to configure from the Modules to Install panel. Otherwise, go to step 35 on page 40.

Configure the QXO Server

25 Complete the required fields in the Outbound Server panel.

Destination Directory. Enter the directory containing the qxo server.

Outbound Database Name. Enter the Outbound database name.

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

Outbound Database Directory. Enter the Outbound database directory location.

Outbound Database Host. Enter the Outbound database host name.

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

Progress Directory. Enter the Progress location the Progress AppServer uses.

26 Complete the required fields in the Outbound UI AppServer panel.

AppServer Name. Enter the Progress AppServer name.

AppServer Port. Enter the Progress AppServer port number.

AppServer Max Connections. Enter the maximum number of connections this AppServer allows.

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

Name Server. Enter the name of the NameServer.

Update Ubroker Properties. Select this check box for the installer to update the Ubroker Properties. Make sure that the installer has the permissions required to update the file. If you do not select the check box, create the entry manually. The template is located under the scripts subdirectory of the QXO Server directory. The Outbound UI and Outbound SI AppServers are required for correct QXO operation.

27 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 this AppServer allows.

Admin Server Port. Enter the port number of the Progress AdminServer.

Name Server. Enter the name of the NameServer.

Update Ubroker Properties. Select this check box for the installer to update the Ubroker Properties. Make sure that the installer has the permissions required to update the file. If you do not select the check box, create the entry manually. The template is located under the scripts subdirectory of the QXO Server directory. The Outbound UI and Outbound SI AppServers are both required for correct QXO operation.

28 If necessary, select the next component to configure from the Modules to Install panel. Otherwise, go to step 35 on page 40.

Configure the QXO Servlet

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

WebApp Name. Enter the QXO servlet application name known to the Tomcat application server.

AppServer Name. Enter the Progress AppServer name.

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

Name Server Port. Enter the Progress NameServer port number.

30 In the Tomcat for Outbound panel, complete the required fields. These values are populated automatically if QXO uses the same AppServer configuration.

Port. Enter the Tomcat port number.

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.

If necessary, select the next component to configure from the Modules to Install panel. Otherwise, go to step 35 on page 40.

Configure the QXtend QAD .NET UI Plug-in

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

Version. Enter the QAD .NET UI version.

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

Port. Enter the Tomcat Port.

Tomcat Home. Enter the Tomcat installation home directory.

Configuration Name. Enter the configuration name set for the .NET UI (the default is qadui).

32 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 NameServer.

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 NameServer (if Direct Connect is not selected).

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

33 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

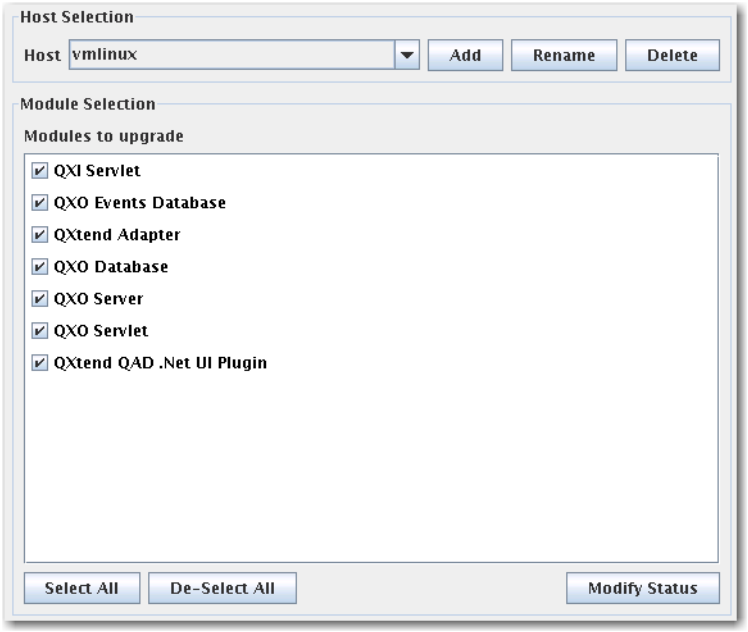
34 If you have entered all of the parameter settings for the required components, click Next and go to Default Configuration. See Appendix B, “Default Configuration,” on page 79 for more information.

35 Click Next and go to “Review the Pre-installation Summary” on page 47.

Upgrading QXtend

Select the Host and Modules to Upgrade

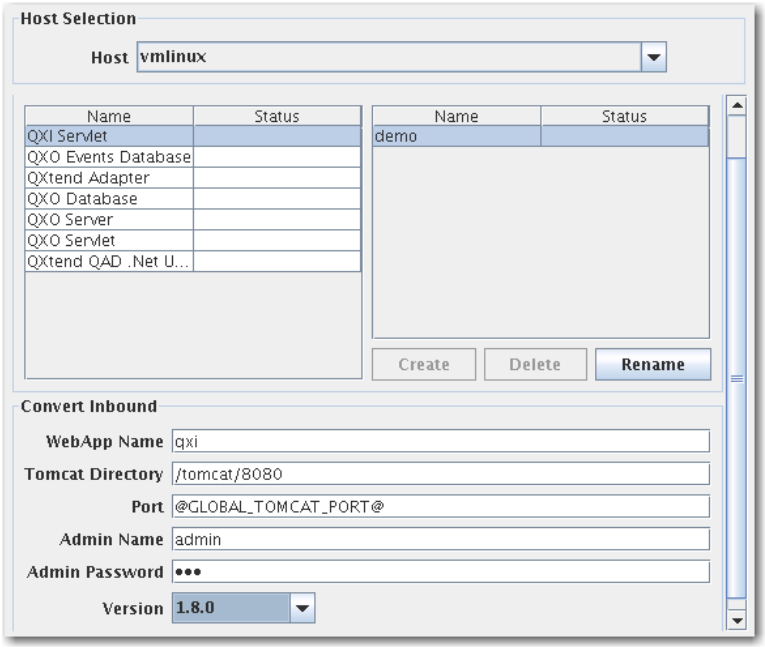
Fig. 4.11
Choose the Host and Install Set



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 (QXtend Adapter, QXO Servlet, QXO Server, and .NET UI Plug-in) and configuration components (QXI Servlet, QXO Event Database, and QXO Database). The configuration components have system information and application data that require transfer to the new QXtend instance. If you do not want to transfer the existing data of a configuration component, use install instead of upgrade.

   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.
4. Click Next to display the Module Copy Selection panel.

Fig. 4.12
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 of 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. Shut down the target QXODB and target QXEvents databases before upgrading.
- You cannot select a different component until you have entered all of the required configuration information for the currently selected component. Enter information in the fields highlighted in orange.

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

Configure the QXI Servlet

7 Complete the required fields in the Convert Inbound panel.

WebApp Name. Enter the application name known to the Tomcat application server.

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

Version. Use the drop-down list to specify the version being migrated.

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.

Note During QXI conversion, a prompt asks you to shut down Tomcat. A shutdown replaces 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 asks you to restart Tomcat.

Important Perform a shutdown/startup before pressing the button.

8 If necessary, select the next component to configure.

Configure the QXO Events Database

9 In the Convert QXEvents 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 Progress location that this qxevents instance uses. This directory should be the same Progress version as the QAD Enterprise Application databases.

10 If necessary, select the next component to configure.

Configure the QXtend Adapter

11 See steps 16 to 20 in “Configure the QXtend Adapter” on page 35.

12 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 file name.

Configure the QXO Database

13 In the Convert QXODB 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 Progress location that this QXODB instance uses.

Configure the QXO Server

14 See steps 25 to 27 in “Configure the QXO Server” on page 38.

15 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 file name.

Configure the QXO Servlet

16 See steps 29 to 30 in “Configure the QXO Servlet” on page 39.

17 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 file name.

Configure the QXtend QAD .NET UI Plug-in

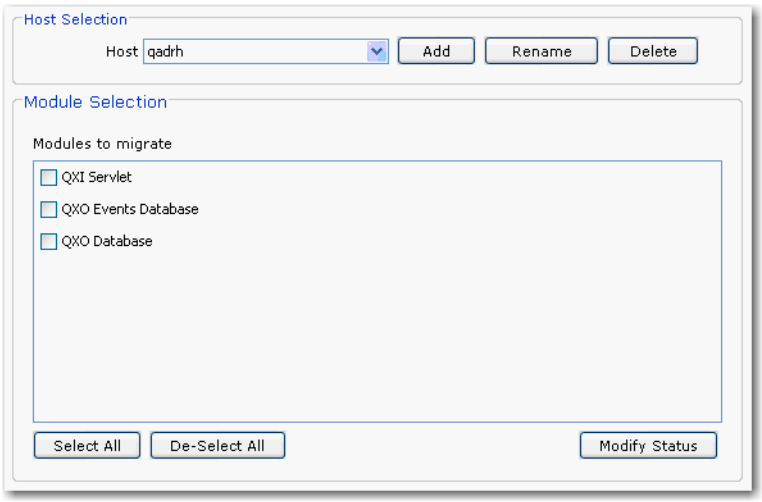
18 See steps 31 to 34 in “Configure the QXtend QAD .NET UI Plug-in” on page 40.

19 If you have entered all parameter settings for the required components, click Next and go to “Review the Pre-installation Summary” on page 47.

Migrating QXtend

Select the Host and Modules to Migrate

Fig. 4.13
Choose Host and Install Set



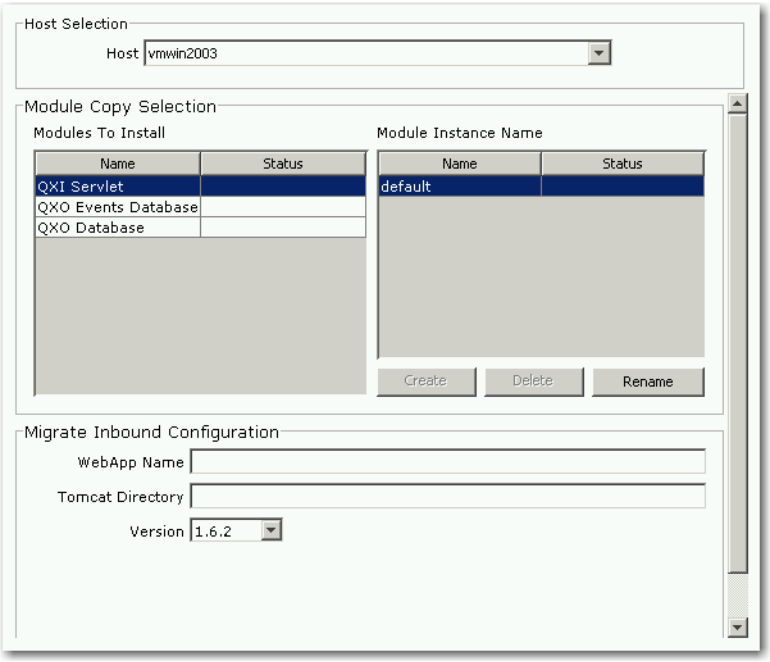
1 In the Host Selection panel, enter the host on which to install the components.

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. 4.14
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 of 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 was not 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. Select the check box for a component when its corresponding instance is highlighted.

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

Note You cannot select a different component until you have entered all of the required configuration information for the currently selected component. The required fields are highlighted in orange.

Configure the QXI Servlet

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

WebApp Name. Enter the application name known to 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.

7 If necessary, select the next component to configure.

Configure the QXO Events Database

8 In the Migrate QXEvents 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 if your database is operating in multi-user mode. If you use this option, you may not have to shut down your database before you install QAD QXtend. However, QAD recommends that you 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 the QAD Enterprise Applications database uses.

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

9 If necessary, select the next component to configure.

Configure the QXO Database

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

Physical Name. Enter the physical name of the QXO database

Connection Type. Select Local or Network.

Multi-user. Select this option if your database is operating in multi-user mode. If you use this option, you may not have to shut down your database before you install QAD QXtend. However, QAD recommends that you 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 on which the QAD Enterprise Applications database is installed.

Service Name. Enter the network service the QAD Enterprise Applications database uses.

Version. Use the drop-down list to select the version being migrated.

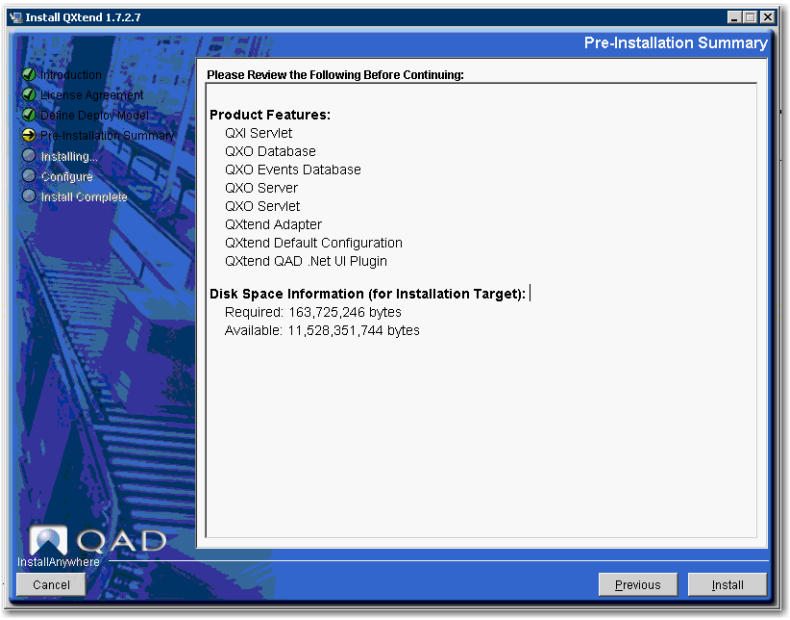
11 If you have entered all parameter settings for the required components, click Next and go to “Review the Pre-installation Summary” on page 47.

Review the Pre-installation Summary

The Pre-installation Summary screen lists the components selected for installation, upgrade, or migration and displays disk space information.

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

Fig. 4.15
Pre-installation Summary



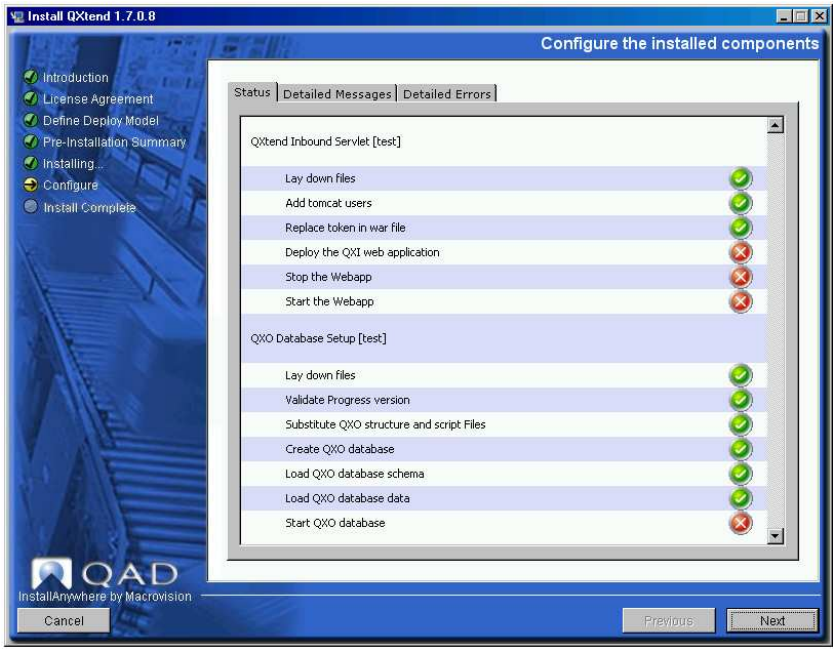
Review the information and click Install to continue with the installation, upgrade, or migration.

Note If the installer encounters an error, it pauses, allowing you to correct the problem. See Appendix C, “Process Control,” on page 85 for more information. If you have difficulty resolving a problem, see Chapter 6, “Troubleshooting QAD QXtend Installs, Upgrades, and Migrations,” on page 61.

When the install, upgrade, or migration finishes, proceed to “Review the Installation Summary” on page 48.

Review the Installation Summary

Fig. 4.16
Install Status Screen



1 Use the Status screen to review the results of the install, upgrade, or migration.

2 A red cross symbol indicates that an error occurred. If one or more red crosses are displayed, see Chapter 6, “Troubleshooting QAD QXtend Installs, Upgrades, and Migrations,” on page 61.

   If the installer encounters an error, a dialog box displays. You can select the “Detailed Messages” and “Detailed Errors” tabs for details to help you diagnose or fix the problem. You then choose the appropriate action to continue or quit the installation. See Appendix C, “Process Control,” on page 85 for more information. If you have difficulty resolving a problem, see Chapter 6, “Troubleshooting QAD QXtend Installs, Upgrades, and Migrations,” on page 61.

   If only green check marks or green check marks and one or more blue double arrows are displayed, the install, upgrade, or migration was successful. Click Next.

3 The Install Complete screen appears.

   Depending on your previous selections, the installer may display a reminder that describes post-installation work you must perform. Note the information for further reference.

   When you are finished, press Done.

To finalize an installation, see “Overview” on page 52. To complete an upgrade, see “Post-upgrade” on page 59. To finalize a migration, see “Post-migration” on page 60.

Installing, Upgrading, and Migrating QAD QXtend in a Character Environment

The following procedure describes how to install, upgrade, and migrate QAD QXtend in a character environment. It assumes that you are using Windows.

1 Verify that Tomcat is running.

2 Verify that you have the required permissions to access the relevant folders.

3 Run the GUI installer to populate the QDCS with the required parameter settings.

4 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.

5 Move to the host you created in step 4.

6 Mount the release media.

7 Choose Start|All Programs|Accessories|Command Prompt to open a command window.

8 Navigate to the `InstData` directory on the release media.

9 Navigate to the appropriate directory for your environment. For example, if you are in a Linux environment, navigate to the Linux directory.

10 Navigate to the `VM` directory.

11 Start the executable by entering one of the following commands:

- Windows: `QXtend.exe -i console`
- Linux, AIX, or HP-UX: `sh ./QXtend.bin -i console`

12 The Introduction displays. Press Enter to continue. The first page of the License Agreement text displays.

13 Press Enter to move through and read the pages.

14 On the final page of the License Agreement, press Y to accept the terms, and then press Enter. The Log File Directory text displays.

15 Press Enter to accept the default location of the log file directory, or enter a different directory.

16 Select the Parameter Service type. Enter Y for network (the default) or N for local file.

17 Get User Input displays. These settings permit the use of the settings you defined using the QDCS in the GUI installer.

18 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.

19 Select install, upgrade, or migration.

20 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.

21 The Install Complete screen appears.

Note If the installer encounters an error, it pauses, allowing you to correct the problem. See Appendix C, “Process Control,” on page 85 for more information. If you have difficulty resolving a problem, see Chapter 6, “Troubleshooting QAD QXtend Installs, Upgrades, and Migrations,” on page 61.

22 After reviewing the results of the install, upgrade, or migration, press Enter.

23 Restart the Tomcat AppServer and start QAD Enterprise Applications.

Adding Additional QAD Enterprise Applications Instances to QXtend

To install additional source applications (except a DDP-only installation) for Outbound, install the QXO Events Database and QXtend Adapter as another instance on the same or a new host from the configured environment. To add additional receivers for Inbound, install the QXtend Adapter as another instance on the same or a new host from the configured environment.

To use a QAD ERP core product as the source application and receiver, install the QXO Events Database and QXtend Adapter.

See the advanced installation information in “QXtend Installation Settings” on page 28.

Chapter 5

Post-installation, Post-upgrade, and Post-migration Tasks

This section describes post-installation and post-upgrade tasks. It also contains post-migration information.

Overview

Following QXtend installation, upgrade, or migration, the product requires configuration to make it ready for use. The following sections describe the required product configuration tasks.

Post-installation

The following 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 determine if 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, recompile the AppShell programs with the QXtend Adapter in the `PROPATH` before the standard QAD Applications code. This action is necessary because the AppShell 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:

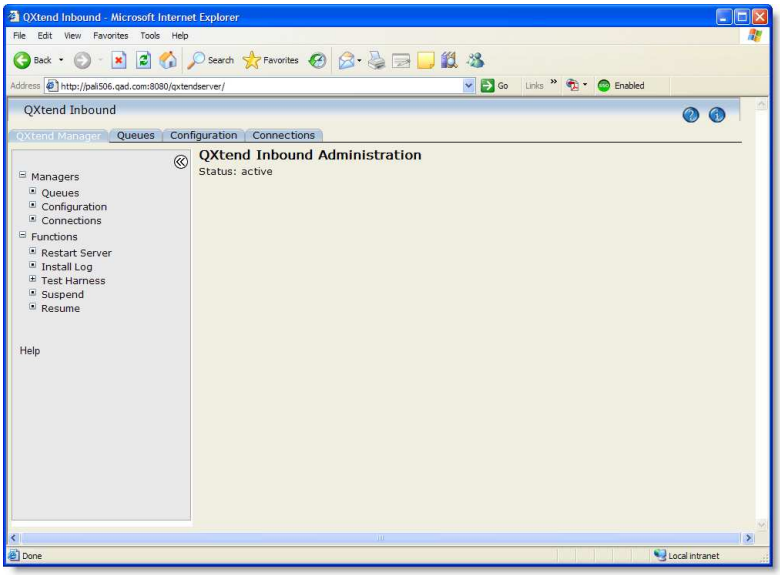
   `http://<qxtendhost>:8080/<QXI webapp>/index.jsp`

   The login screen displays.

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

   The screen shown in Figure 5.1 displays.

Fig. 5.1
QXI Main Screen



Enable SSL

You can use SSL with QAD QXtend to encrypt all HTTP messages between the client and server. To use SSL, configure 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 following file content appears in the default `server.xml`. Add the Connector definition following the end of the comment marker (`-->`) as shown in the following example.

```
<!-- 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>
```

See 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, see the Tomcat SSL documentation.

Modify catalina.bat or catalina.sh

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

For the QAD QXtend servlets to work with the HTTPS connector, set the properties 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 in the following code as an example; however, you typically make this change in seven or eight locations.

```
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 can issue a warning or refuse to work with an HTTPS connection to the server. This behavior can cause a fault on the client side.

Generate a self-signed server certificate using the Java keytool. This task is done as follows:

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

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

2 The system prompts you to enter the keystore password. The default SSL password for Tomcat is `changeit`. If you change the password, place it 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 prompt and all subsequent prompts use the format:

   `[Unknown]: <Enter_data_value_here.>`

4 The system prompts you for the following information:

   - 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

   Enter the appropriate values for your implementation.

5 The system prompts you to confirm the following information:

```
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 The system prompts you to enter the key password for `<tomcat>`.

   Press Enter if the key password for Tomcat is the same as the keystore password. Do not attempt this task unless you are an experienced HTTPS user.

See 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.

To perform the test, do the following:

1 Ensure that the Tomcat AppServer is running.

2 Ensure that the Progress AppServer is running.

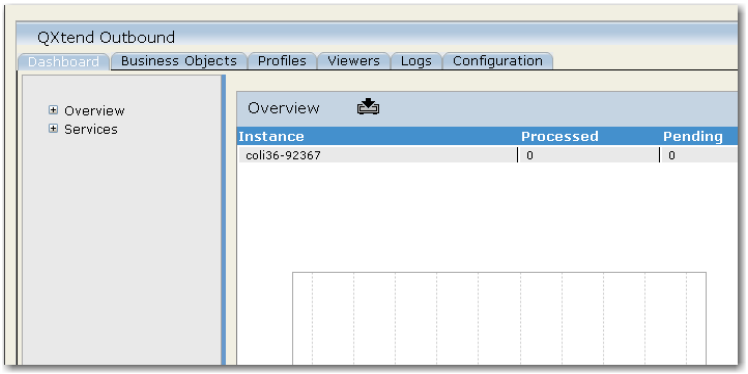
3 Ensure that the QAD Enterprise Applications production and qxevents database servers are running.

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

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

5 The QXO Dashboard is displayed as shown in Figure 5.2.

Fig. 5.2
QXO Dashboard



Note See *User Guide: QAD QXtend* for details about the administrative functions.

If your system is not configured properly, the following message can appear 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 did not start on the server *servername*, possibly because of configuration problems.
- The configuration file for the QXO UI (`environmentmanager.xml`) was not correctly modified to point to the AppServer. Verify that:
  - The name of the server running the AppServer 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 qxoui_AS. If multiple AppServers are running on the same machine, you can modify the name.

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 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 can display when you try to display the event types:

```
Database not available
```

This message indicates that one of the following has occurred:

- The database name is configured incorrectly; correct the database name using the Configuration tab.
- The qxevents database server is not running. Verify that the servers are started for the QAD Enterprise Applications databases (including qxevents) using the `start.DBName` script.

A blank screen 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 properly configured with the correct PROPATH and database connections. It verifies that an event is recorded when a change occurs in the source application.

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

To perform the test, do the following:

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 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 the Enable QXtend Outbound option is set to Yes. This setting is required to log events 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.

8. Choose the ert_app_event table, press Go, and specify an output file name.

9. Open the output file in a text editor and verify that it contains 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 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, verify that the following information 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 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 an 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 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 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. QXO processes the event that resulted from creation of the new customer in the previous test.

11 Click the Log tab in the QXO Console to verify that QXtend sends a message to the subscriber you created.

Check PROPATH for QXtend Adapter in .NET UI Scripts

After you have successfully installed QXtend with the .NET UI plug-in, use the following steps to verify that the `PROPATH` entries in the .NET UI connmgr and telnet scripts contain the QXtend adapter:

1 Open the `connmgr.Demonstration` and `telnet.Demonstration` scripts. For QAD Enterprise Edition, they are located in the `$QDT_HOME/envs/$ENV_NAME/scripts` directory. For QAD Standard Edition or previous versions, they are located in the `$MFG_HOME/qadui` directory or another place, depending on how the .NET UI was installed.

2 Verify that the QXtend adapter directory is in the `PROPATH`. If it is not, add it.

3 Restart the qadui AppServer and WebSpeed server.

4 Verify that the .NET UI connmgr and telnet scripts support QXtend. Modify the QAD Enterprise Applications data in the .NET UI and examine the data that QXtend Outbound publishes.

   QXtend Outbound must be properly configured to do this. Since you can open a typical program in Desktop and Terminal modes in .NET UI, be sure to test both modes.

Merge Database Triggers

If custom database triggers were installed before QXtend installation and they conflict with the QXtend triggers, you must merge the corresponding trigger code with the QXtend trigger code. The installed QXtend triggers are listed in Appendix E.

Post-upgrade

QXtend Outbound Configuration

The following steps are required following a QXtend Outbound upgrade:

1 Update the source application database details to ensure that the qxevents database points to the new version's qxevents database. This action 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 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 Verify 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 new version's Web Application.

3 Verify the license manager details for QXtend Outbound. You can set them in QXO by selecting Configuration|Outbound Settings|License Manager.

4 You must manually validate all business objects before an event message can be extracted. To validate business objects, select the Business Objects tab, and click Validate All. This action regenerates the schema definition for the business objects.

Note Since QXtend 1.8.4, QXtend only allows alphanumeric characters, dashes, and underscores in BO or profile names. During upgrade and migration, if a BO or Profile name contains other special characters from a previous QXtend installation, the installer raises warning messages and writes them to the install log file. After upgrade or migration, a BO with special characters is not validated. You must change the BO or Profile name to make it available.

Post-migration

The following steps are required following a QXtend migration:

1 Update the source application database details to ensure that the qxevents database points to the new version's qxevents database. This action 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 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 Verify the 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 new version's web application.

3 Verify the license manager details for QXtend Outbound. You can set them in QXO by selecting Configuration|Outbound Settings|License Manager.

4 Additional work can be required depending on the QXtend version. The work is described in a reminder the installer displays after a successful migration.

5 You must manually validate all business objects before an event message can be extracted. To validate business objects, select the Business Objects tab, and click Validate All. This action regenerates the schema definition for the business objects.

Note Since QXtend 1.8.4, QXtend only allows alphanumeric characters, dashes, and underscores in BO or profile names. During upgrade and migration, if a BO or Profile name contains other special characters from a previous QXtend installation, the installer raises warning messages and writes them to the install log file. After upgrade or migration, a BO with special characters is not validated. You must change the BO or Profile name to make it available.

Chapter 6

Troubleshooting QAD QXtend Installs, Upgrades, and Migrations

This section describes how to resolve QAD QXtend install, upgrade, and migration problems.

Overview

This section describes how to resolve issues encountered during or after a QXtend installation, upgrade, or migration.

Diagnosing the Problem

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.

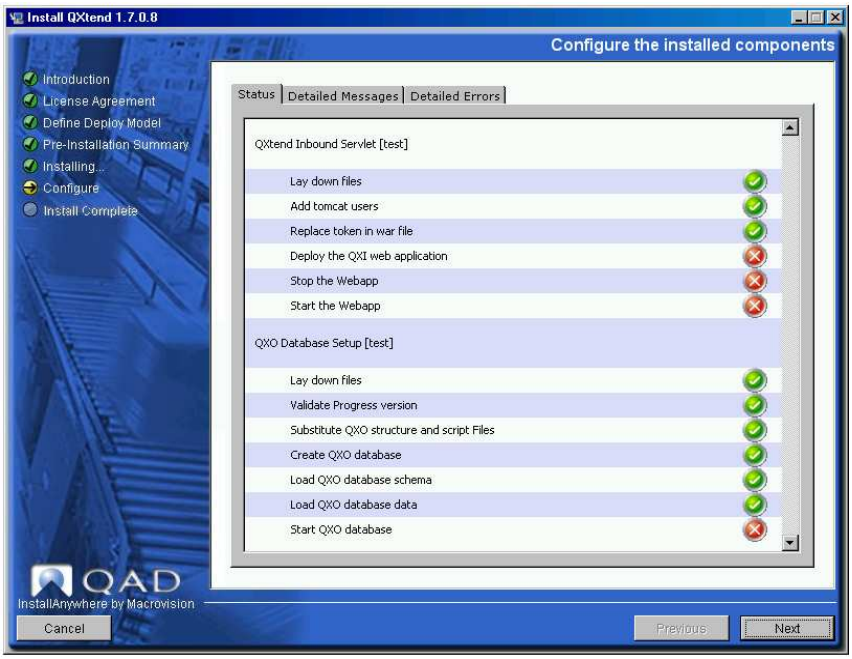
Status

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

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

To allow you to easily identify issues, a status also displays for any substeps performed during the installation or upgrade. For example, Figure 6.1 shows that installing the Inbound Servlet and Outbound Database consists of several substeps (replace token in archive, loading the database schema, and so on). A status displays for each of these substeps.

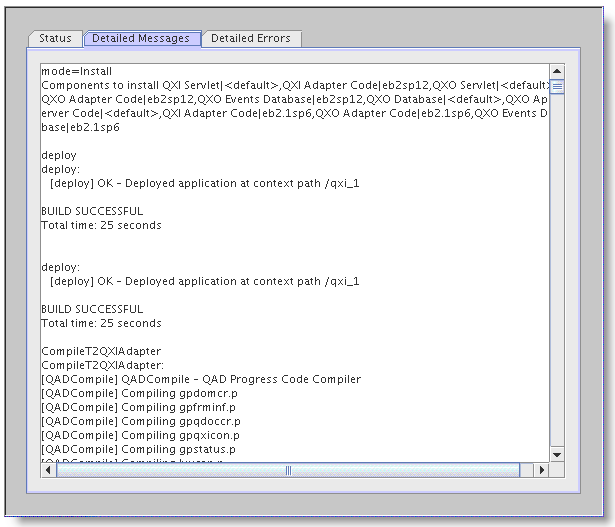
Fig. 6.1
Installation Status Screen



Detailed Messages

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

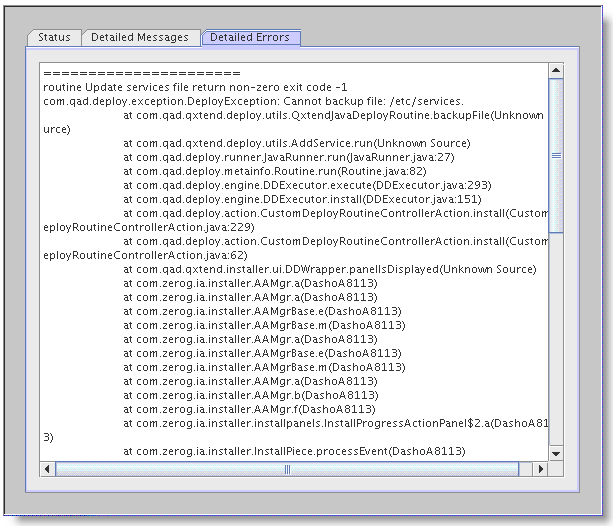
Fig. 6.2
Installation Detailed Messages Screen



Detailed Errors

The Detailed Errors screen shows a log of any fatal system errors noted during the installation or upgrade. These messages indicate a serious problem with the system (for example, a corrupted Java or Progress executable). Some of these errors are recorded in the log file.

Fig. 6.3
Installation Detailed Errors Screen



Checking repository.xml

The installer keeps the installation configuration and status of all install-related tasks in the install `repository.xml` file.

Note Do not manually modify this file. Instead, modify it through the GUI installer (see “Installing, Upgrading, and Migrating QAD QXtend in a GUI Environment” on page 23). Beginning with QXtend 1.6.3, you can also modify the status of routines and therefore modules. For more information see Appendix C, “Process Control,” on page 85.

The `repository.xml` file can give you a high-level view of the results of an install, particularly when doing a console install, which presents less information.

Figure 6.4 shows the contents of a typical `repository.xml` file.

Fig. 6.4
repository.xml File

```
<?xml version="1.0" encoding="UTF-8"?>

<repository>
  <environment name="test" createDate="2010-05-25 11:55:55 +0000">1
    <product name="QXtend" version="1.6.3" servicePack="IR">2
      <global>
        …
      </global>
      <defaultconfig>
        …
      </defaultconfig>
      <host hostname="xxx">3
        <component name="QXI Servlet" copy="default">5
          <install status="Complete" moduleStatus="Complete">
            …
          </install>
          <parameters>
            …
          </parameters>
        </component>
        <component name="QXO Servlet" copy="default">
          <install status="Incomplete" moduleStatus="Incomplete">
            <routines>
              <routine name="install" status="Done"/>4
              <routine name="add_tomcat_users" status="Error"/>
              <routine name="replace_token_in_war" status="Pending"/>
              <routine name="deploy_qxo" status="Pending"/>
            </routines>
          </install>
          <parameters>
            …
          </parameters>
        </component>
      </host>
    </product>
  </environment>
</repository>
```

A `repository.xml`file has the following features:

1 Environment: The name attribute is the environment set during the installation.

2 Product: There can be more than one product in the environment. Verify that the name and version match the targeted install.

3 Host: There can be multiple hosts in an installation. Verify that this host is the correct one.

4 Routine: Routines are granular tasks that the installer performs. A group of routines makes up a component. Routines can have a status of Done, Skipped, Error, or Pending.

5 Component: The component node is the parent to the status and the parameters. Check the component name and proceed to observe the install or upgrade nodes.

a Status: The status of that particular component instance (indicated with the copy attribute). It can be:

- Complete if all routines are Done
- Forced Complete if any routine is marked as Skipped
- Incomplete if any routine is marked as Pending or Error

b ModuleStatus: The status of all the instances of a component. If there are three instances (indicated by copy), all three must have a status of Complete or Forced Complete before the ModuleStatus is marked as Complete.

Checking the status of a component (more specifically a routine) leads you to the installation step that failed. This functionality improves installation log navigation because you know what to look for.

The `repository.xml` file is located under the `data` directory in the QADDeployService servlet in the `tomcat/webapps` directory. If you are using a local file, `repository.xml` is in the location specified during the install.

Reading the Installation Log

The QXtend installation log, named `QXTEND-InstallLog-<TimeStamp>.log`, resides in the configured directory that was chosen when running the installer. The log file captures all of the standard output from the JVM during the install. If you are running the GUI installer, this file is the same as the Detailed Messages tab.

If an error occurred, you can read the log file during or after the install. If the pause occurred during installation (see Appendix C, “Process Control,” on page 85), you can look at the file without exiting the installer. In some cases, you can fix the problem and rerun the routine that failed.

If the installation process completed with a non-fatal error, you can identify the error by looking in the `repository.xml` file (see “Checking repository.xml” on page 64). To find the problem, look for the component that owns the failed routine. Then search the log file for the beginning of that component by finding the name followed by the instance in square brackets:

```
QXtend Inbound Servlet [default]
===============================================
```

The failed routine has a corresponding ant task. Look for an ant task with a name that is similar to that of the failed routine.

```
add-tomcat-users:
   [copy] Copying 1 file to /qad/tomcat-164-8110/conf
   [xslt] Processing /qad/tomcat-164-8110/conf/tomcat-users.xml.bak
   to /qad/tomcat-164-8110/conf/tomcat-users.xml
   [xslt] Loading stylesheet
   /tmp/656641.tmp/resources/Transformations/tomcat-users.xsl
```

If the routine is successful, you see a message similar to the following:

```
BUILD SUCCESSFUL
Total time: 1 second
Trying to override old definition of datatype resources
```

If the routine is unsuccessful, you see a message similar to the following:

```
BUILD FAILED
java.net.ConnectException: Connection refused
```

The failure message includes a Java stack trace of the error. Since stack traces are sometimes hard to understand, QAD provides Helpful Hints when the installer pauses for an error.

Reading the steps around the error can provide clues regarding its cause.

Environment Issues and Common Mistakes

This section describes problems that can occur during QXtend installation. For more detailed information regarding potential installation issues, visit the QAD KnowledgeBase or contact QAD support.

No X11 DISPLAY Variable was Set

This error appears when you try to run the install in a console without GUI capability. To run the installer in console mode, add the `-i console` option to the command.

Unable to Deploy

If the Web Applications do not deploy properly, Tomcat may have failed to fully start during the installation. Verify that the Tomcat server is fully operational (a good test is to access the Manager application in your browser). If it is running, but produces out-of-memory exceptions, you can increase the maximum and default Tomcat heap size (`-Xmx` and `-Xms`). See your operating system documentation.

Cannot Connect to Database

The install log reports this problem as an error code 14. If the installer is trying to connect to a database using client-server or shared-memory mode, this error probably means that the database is not running. It could also mean that the database is trying to run in single-user mode, but the database is already in use.

If the install log reports error messages like “SYSTEM ERROR: Shared memory access permission denied (1136),” log in as the user who created the database, restart the installer, and continue with the installation.

Default Configuration

A default configuration failure is often the result of a few issues.

The Inbound Web application and Outbound AppServer must be running so that the Web Service can be accessed, and in turn, contact the AppServer.

The QAD Enterprise Application databases must be running for the default configuration to create subscribers for the source application.

If the problem seems to be configuration-related, consult the AppServer logs in the server installation logs directory.

No Features to Install on this Host

The following message can appear during QXtend installation:

> “The information stored in the QAD Deployment Configuration Service indicates that there are no features to install on this host (xxx). If this is unexpected, you may wish to re-run the installer on a Windows or X-Platform and update the configuration. If this is correct, then continue the installer on the next host. The installer will now exit.”

This message indicates that there are no installs to do on the machine on which you are running the install. You can continue the install on the servers you are using.

This message can display for a number of reasons:

- When you first run the installer on a PC in a multi-tier install, it builds the QAD Deployment Service (QADDeployService.V1), interrogates the user for configuration values (data gathering), and checks if it should also deploy on the same PC. Generally, the answer is no because you are just building the configuration data through a GUI interface, but you plan to deploy the components on a different server or servers. In this case, you can safely ignore the message and proceed.
- During `QXtend.bin -i console`, you were prompted for an environment to install and entered an invalid environment name. Most likely, when you ran the installer, you provided an environment name like Test, Prod, and so on. If your answer was not valid, the installer looks in the `respository.xml` file for an environment tag called *<what you entered>*. The tag is missing, resulting in "nothing to install on this Host." If you forgot your environment name, you can check the `respository.xml` file for the name you provided. You can also run a GUI installer and observe the option at the environment selection screen. Then rerun the installer.
- To determine if an incorrect host name was entered, compare `xxx` from the error message with the result of `hostname` or equivalent command in a console window. When using the QXtend installer, always use the shortened host name (do not include the domain).

IATEMPDIR Space

The installer must self-extract several files before it can run. This requirement can cause an error before the installer finishes loading, particularly when using the bundled Java VM. This issue occurs because there is insufficient space in the default extract location. To resolve this problem, you can set the environment variable `IATEMPDIR` to a storage area with sufficient space.

Java Memory

If you experience out-of-memory errors during installation before the installer finishes initializing, do the following:

1. Create a directory called `bin` under `$HOME` as user `mfg`.
2. Create a file named `java` under the `bin` directory and add following parameters in the file:
   - `- /opt/java1.5/bin/java -verbose -Djava.awt.headless=true -XX:HeapDumpOnOutOfMemoryError -client`
   - `-Xms1024M -Xmx2048M $@`
   - `-XX:PermSize=512M`
   - `-XX:MaxPermSize=512M`
   - Set `JAVA_HOME` to `$HOME/bin`.
   - Add `$HOME/bin` in the beginning of the `PATH`.

The installer should now be able to complete the installation process.

UI API Adapter Performance

When using QXtend with AIX, change the AIX kernel setting `tcp_nodelayack` to 0. Otherwise, QXtend UI API performance is severely limited.

Known Issues

Installer Menu Items Difficult to Select

Starting the QXtend Installer UI in X-Windows while using X-Windows clients (such as xterm) can make installer UI menu items difficult to select.

This behavior is a known environment-related issue with the Java X-Windows component. To work around it, use the Up and Down keys to make a selection.

Appendix A

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 want to modify any of default configuration parameters, use QXtend's advanced installation mode.

Tokens

Tokens enable you to dynamically reference parameters from other fields. You can use any parameter name as a token by surrounding 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 displays the corresponding parameter name. If you hover over the field itself, the fully resolved parameter appears as a tool tip.

Parameters

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

Table A.1 Default QXtend Installation Parameters.

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 Configuration	false
QXO Database	CREATE_QXODB_SERVICE	Create Services File Entry	false
QXO Database	QXODB_BLOCK_SIZE	Block Size	4

Table A.1 — *Default QXtend Installation Parameters.* — (Page 1 of 7)

Table A.1 Default QXtend Installation Parameters.

Module	Parameter Name	Description	Default Value
QXO Database	QXODB_DIR	Database Directory	@GLOBAL_QXTEND_DIR@ \qxodb
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 Messages (CLOB)	
QXO Database	SA_SCHEMA_AREA	Schema Area	
QXO Database	SA_SUBSCRIBER_MSG	Subscriber Messages (CLOB)	
QXO Database	SA_TRANS_DATA	Transactional Data	
QXO Database	SA_TRANS_INDEX	Transactional Indexes	
QXO Database	QXODB_ORACLE_USER_NAME	Oracle User Name for QXO Database	
QXO Database	QXODB_ORACLE_USER_ PASSWORD	Oracle User Password for QXO Database	
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

Table A.1 — *Default QXtend Installation Parameters.* — (Page 2 of 7)

Table A.1 Default QXtend Installation Parameters.

Module	Parameter Name	Description	Default Value
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@
QXO Events	QXEVENTS_DB_LANG_UTF	Database UTF	Depends on MFG version
QXO Events	QXEVENTS_HOST	Host	
QXO Events	QXEVENTS_SERVICE	Service Name	
QXO Events	QXEVENTS_SERVICE_PORT	QXO Events	
QXO Events	QXEVENTS_ORACLE_USER_NAME	Oracle User Name for QXEvents	
QXO Events	QXEVENTS_ORACLE_USER_ PASSWORD	Oracle User Password for QXEvents	
QXO Server	APPSHELL_SOURCE	.NET UI Home Directory	
QXO Server	APPSRV_ADMIN_SI	AdminServer Port	20931
QXO Server	APPSRV_ADMIN_UI	AdminServer Port	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	NameServer	NS1
QXO Server	APPSRV_NAMESERVER_UI	NameServer	NS1
QXO Server	APPSRV_PORT_SI	AppServer Port	3092
QXO Server	APPSRV_PORT_UI	AppServer Port	3091
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@

Table A.1 — *Default QXtend Installation Parameters.* — (Page 3 of 7)

Table A.1 Default QXtend Installation Parameters.

Module	Parameter Name	Description	Default Value
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	NameServer Port	5162
QXtend Adapter	ADD_CONN_PARAMS	Additional Connection Parameters	
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 Parameters	-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_DB_LANG_UTF	Database UTF	Depends on MFG version
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_INTERNAL_CODEPAGE	Internal Codepage	utf-8
QXtend Adapter	MFG_PROPATH	MFG/PRO xrc directory	Mandatory field
QXtend Adapter	MFG_SP	Service Pack	Mandatory field
QXtend Adapter	MFG_STREAM_CODEPAGE	Stream Codepage	utf-8

Table A.1 — *Default QXtend Installation Parameters.* — (Page 4 of 7)

Table A.1 Default QXtend Installation Parameters.

Module	Parameter Name	Description	Default Value
QXtend Adapter	MFG_VERSION	Version	Mandatory field
QXtend Adapter	MFGPRO_DLC	Progress Directory	Mandatory field
QXtend Adapter	QDT_DIR	QDT Directory	Mandatory field (Enterprise Edition 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
Standard Install	ORACLE_HOME	Oracle Database Home Directory	
Standard Install	ORACLE_SID	Oracle Database System ID	
Standard Install	NLS_LANG	NLS_LANG Environment Variable for Oracle Database	.AL32UTF8
Standard Install	ORACLE_DB_LD	Oracle Database Logical Name	
Standard Install	ORACLE_CONN_PARAMS	Oracle Connection Parameters	-c 500
Standard Install	QXTEND_ORACLE_NET_SERVICE_NAME	Oracle Net Service Name for QXtend	
Default Config	ENABLE_DEFAULT_CONFIG		false
Default Config	DEFAULT_CONFIG_OUTBOUND_RECEIVER		QADQXO
Default Config	CP_CONNECTION_PROTOCOL	Connection Protocol	telnet
Default Config	CP_CONNECTION_PORT	Connection Port	23
Default Config	CP_ SSH_USE_PRIVATE_KEY	Use SSH Private Key	Depends on Connection Protocol
Default Config	CP_SSH_PRIVATE_KEY_FILE	SSH Private Key File	Depends on Use SSH Private Key

Table A.1 — *Default QXtend Installation Parameters.* — (Page 5 of 7)

Table A.1 Default QXtend Installation Parameters.

Module	Parameter Name	Description	Default Value
Default Config	CP_SSH_PRIVATE_KEY_PASSWORD	SSH Private Key Password	Depends on Use SSH Private Key
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		word:
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 UI Plug-in	NET_UI_SERVER_TOMCAT_PORT		@GLOBAL_TOMCAT_PORT@
QXtend QAD .NET UI Plug-in	NET_UI_SERVER_WEBAPP_NAME		qadhome
QXtend QAD .NET UI Plug-in	NET_UI_SERVER_TOMCAT_HOME		@GLOBAL_TOMCAT_HOME@
QXtend QAD .NET UI Plug-in	NET_UI_DIRECT_CONNECT_ APPSERVER		false
QXtend QAD .NET UI Plug-in	NET_UI_APPSERVER_NAME		@APPSRV_NAME_UI@
QXtend QAD .NET UI Plug-in	NET_UI_APPSERVER_HOST		@GLOBAL_INSTALL_HOST@
QXtend QAD .NET UI Plug-in	NET_UI_NAME_SERVER_PORT		5162
QXtend QAD .NET UI Plug-in	NET_UI_APPSERVER_PORT		
QXtend QAD .NET UI Plug-in	NET_UI_INCLUDE_INBOUND		true

Table A.1 — *Default QXtend Installation Parameters.* — (Page 6 of 7)

Table A.1 Default QXtend Installation Parameters.

Module	Parameter Name	Description	Default Value
QXtend QAD .NET UI Plug-in	NET_UI_INCLUDE_OUTBOUND		true
QAD .NET UI	NET_UI_CONFIG_NAME	QAD .NET UI Configuration Name	qadui

Table A.1 — *Default QXtend Installation Parameters.* — (Page 7 of 7)

Appendix B

Default Configuration

This section describes the QXtend default installation configuration.

Overview

The default configuration feature provides an initial set of configuration data for QXtend. This feature allows you to quickly set up QXtend or create a template for setting up interoperability.

Once the installation successfully completes the Inbound Servlet, QXO Database, and QXO Server, the installation process calls the ant target that creates the default configuration inside the newly installed QXtend.

You can enable the default configuration through the Typical and Advanced Install. Enabling the default configuration feature creates the following:

Inbound

- Receiver
- UI, SI, and Fin(QADEE) connection pool for the receiver
- E-mail settings

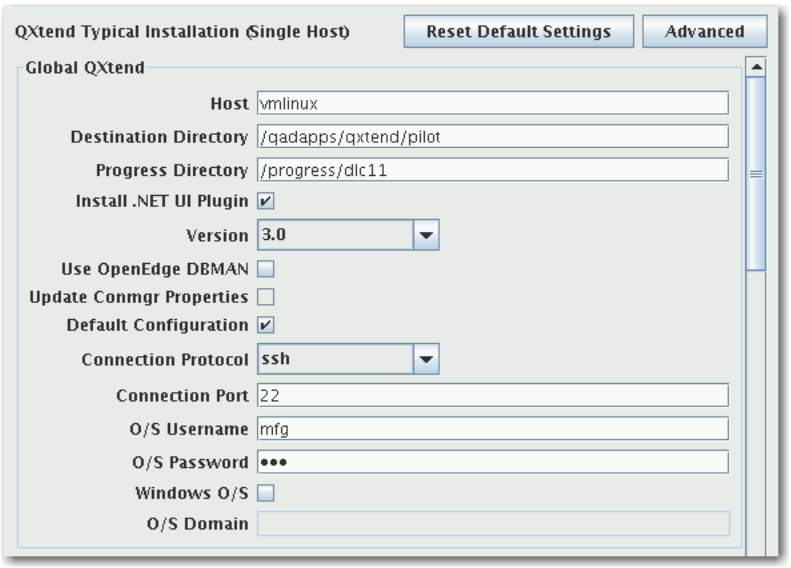
Outbound

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

Note The installer attempts to start the QXO Database and AppServers along with the QAD databases. These components must all be running to successfully complete a default configuration. Verify that they are running before the default configuration routine runs.

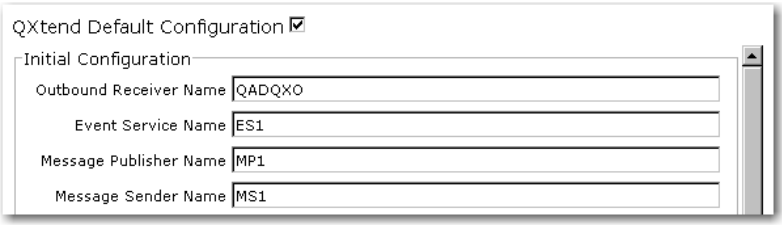
Default Configuration Process

Fig. B.1
Enable Default Configuration in Typical Install



Select the check box and the extra mandatory fields display. For more control over the initial configuration, click the Advanced button. See “Parameters” on page 72 for a list of the parameters used in Default Configuration.

Fig. B.2
Enable Default Configuration in Advanced Install



This screen displays 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 you enable the default configuration, it is initialized automatically by the installer, which uses ant to call the configuration service.

After the default configuration runs, use the following steps to perform a data synchronization:

1. Inbound already has the connection pools and receiver with enabled schemas ready.
2. Outbound would require the assigning of a source domain (which would be a master domain) for each subscriber.
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.

QXtend Default Configuration GUI

The QXtend Default Configuration screen (Figure B.3 and Figure B.4) displays the default QXtend installation parameters. You use this screen to define the default settings for QXtend Inbound and Outbound.

Fig. B.3
QXtend Default Configuration GUI

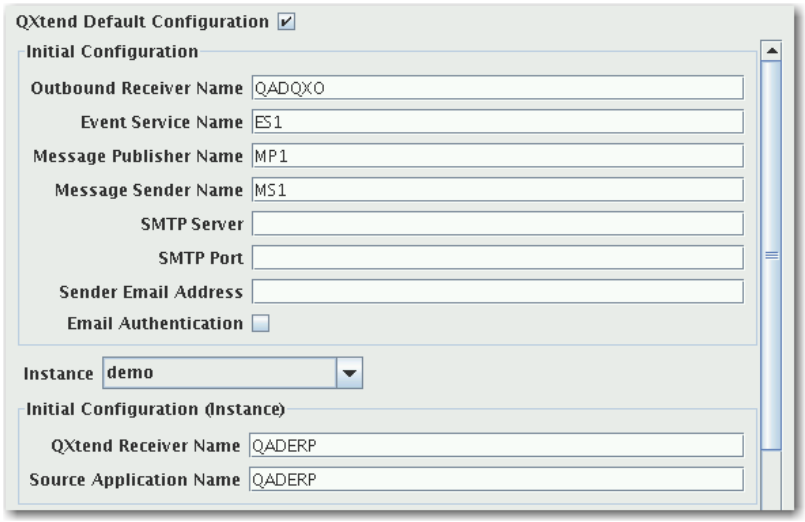
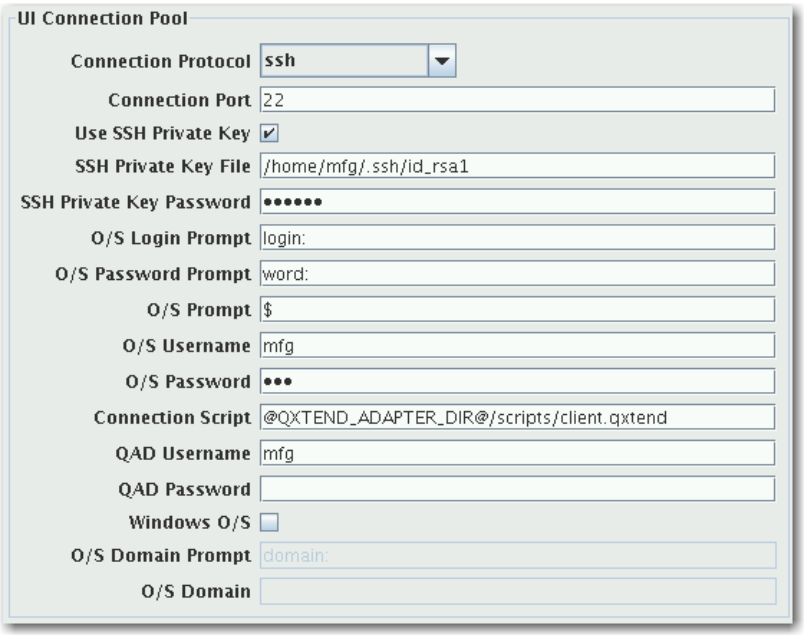


Fig. B.4
QXtend Default Configuration Panel



This GUI contains the following check boxes, pull-down menus, and fields:

QXtend Default Configuration. This check box specifies whether to load the default configurations for QXtend.

Instance Pulldown. The QXtend installation instance that the default configurations load to.

Initial Configuration Panel

Outbound Receiver Name. The default receiver used for Outbound queries.

Event Service Name. The default Event Service name for Outbound.

Message Publisher Name. The default Message Publisher name for Outbound.

Message Sender Name. The default Message Sender name for Outbound.

SMTP Server. The SMTP server name to send out e-mails.

SMTP Port. The SMTP port to send out e-mails.

Sender Email Address. The e-mail address for the e-mail sender.

Email Authentication. Whether e-mail authentication is required.

Initial Configuration (Instance) Panel

QXtend Receiver Name. The default receiver name in Inbound.

Source Application Name. The default source application name in Outbound.

UI Connection Pool Panel

Connection Protocol. Connection protocol for the UI connection pool. The options are telnet and ssh.

Connection Port. Connection port for the UI connection pool. The default values are 23 for telnet and 22 for ssh.

Use SSH Private Key. Whether to use SSH private key for the UI connection pool.

SSH Private Key File. The full path for the SSH private key file.

SSH Private Key Password. The password for the SSH private key.

O/S Login Prompt. The login prompt displayed when connecting to the server.

O/S Password Prompt. The password prompt.

O/S Prompt. The shell prompt.

O/S Username. The O/S user name to connect to the server.

O/S Password. The O/S user password.

Connection Script. The script to execute to connect to QAD Enterprise Applications.

QAD Username. The QAD client name.

QAD Password. The QAD client password.

Windows O/S. Whether the system is a Windows system.

O/S Domain Prompt. The domain prompt.

O/S Domain. The Windows domain to log in to.

Appendix C

Process Control

This section describes QXtend installation process control.

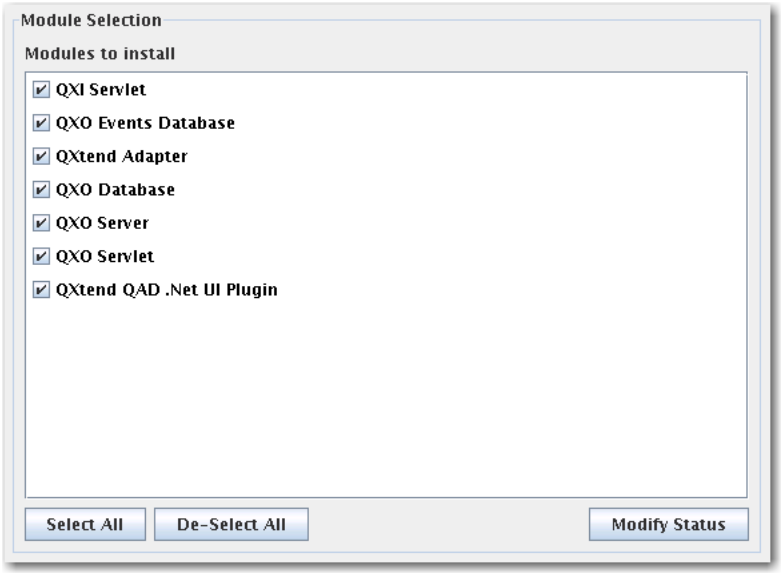
Overview

Process Control can enhance QXtend installation by providing a higher level of control over the installation process.

Using Process Control

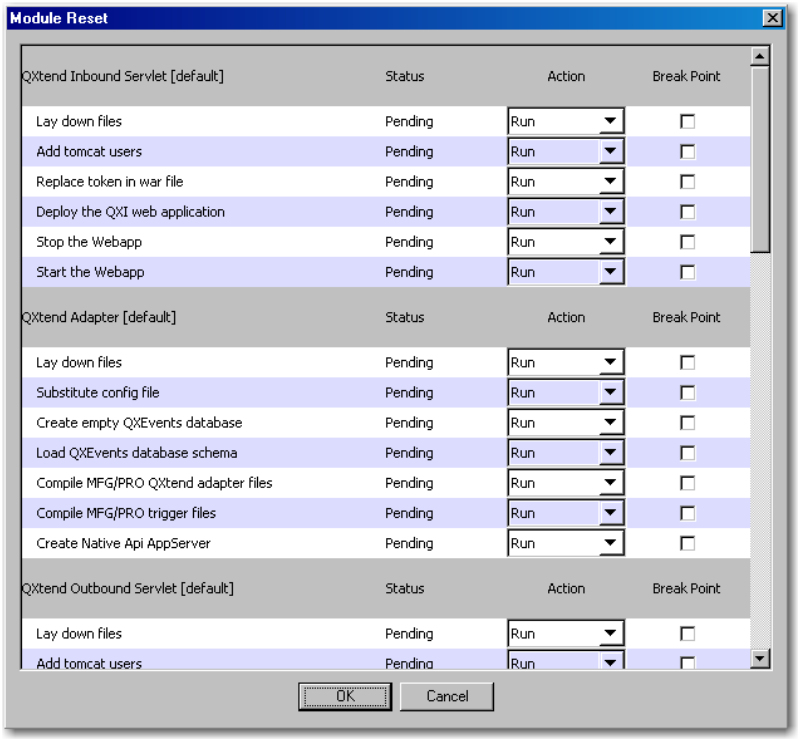
Click the Modify Status button on the Module Selection Screen.

Fig. C.1
Module Selection Screen



The Modify Status pop-up appears.

Fig. C.2
Modify Status Pop-up



Changing the action via the selection lists enables you to run, rerun, or skip each of the routines. This feature allows you to skip 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 could result in an incomplete or non-functional installation. Only skip routines that you are sure are not needed.

Selecting a routine's breakpoint check box pauses the installer before it runs the routine. This feature allows you to pay particular attention to that part of the install.

If the installer encounters an error while running, it pauses, allowing you to fix environment problems on the fly. For example, if Tomcat is not running when the installer tries to deploy the servlets, the installer pauses. You can investigate using the log file, see that Tomcat was not started, and start it. The installation continues as if nothing had happened.

Fig. C.3
Console Prompt After an Error

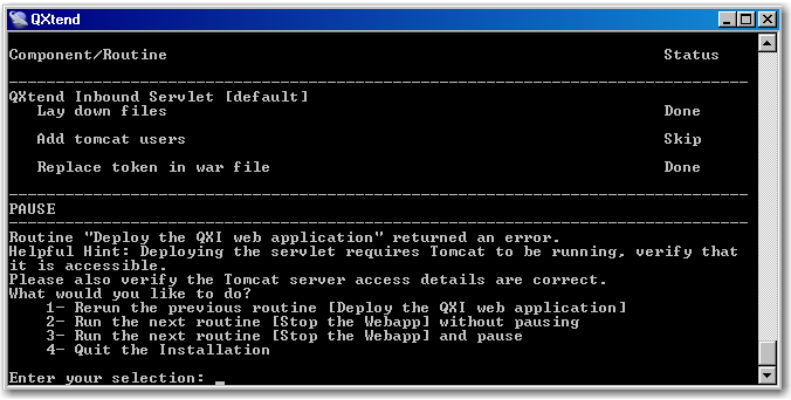
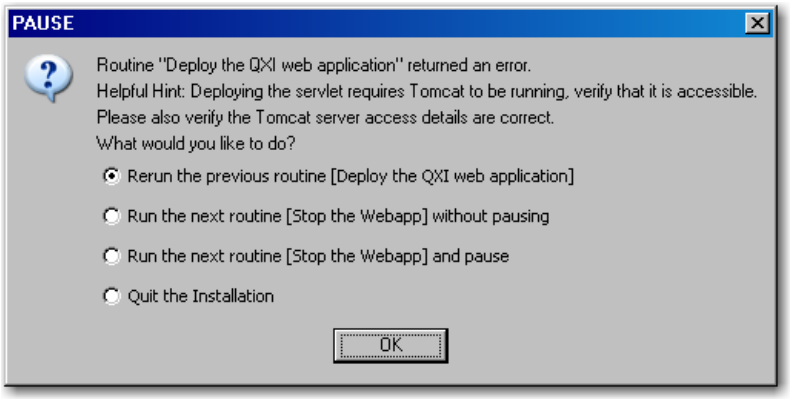


Fig. C.4
GUI Prompt After an Error



You cannot correct configuration errors in the repository file. For instance, if you forgot to add a custom element to the compile PROPATH, you cannot add it once the installation starts and rerun the routine after it fails. Rerun the installer and let it pick up the new value.

In summary, do the following when errors occur:

- To correct the parameters, quit the installer, fix the configuration, and rerun the installer.
- For environment issues such as permissions problems, the database failing to start, and so on, you can fix the issue and resume the installer by selecting “Rerun the previous routine.”
- In rare situations, the installer hangs and does nothing. You can kill the process (typically by pressing Ctrl+c for the console installer), identify the problem, fix the issue, and rerun the installer.

Appendix D

Adapter Program Usage

This section describes QXtend adapter program usage by QXO and QXI.

Overview

QXtend implementation planning, configuring QXtend to work with customized QAD programs, and QXtend installation regression testing following maintenance program modification often require information about QXtend adapter program usage. This section lists the QXtend adapter programs and if QXO or QXI uses them.

The QAD Enterprise Applications version and service pack do not list the following QXtend Adapter programs. The programs in your environment are only a subset of these programs.

Adapter Program Utilization

QXO Programs

The following are QXO programs, or MFG/PRO programs with in-line triggers (some of which have context fixes that QXI also requires):

`gpqxocnf.p`, `gpqxohdl.p`, `gpqxost.i`, `qxodef.i`, `qxoevent.i`, `qxoevent.p`, `qxoglob.i`, `qxopm.p`, `qxotrig.i`, `qxotrign.i`, `adcamt.p`, `adcrtmt.p`, `adcsmt.p`, `adctrymt.p`, `adstmt.p`, `advnmt.p`, `glacmt.p`, `glcalmt.p`, `glfmmt.p`, `gpcontxt.i`, `gpisapi.i`, `icstmt.p`, `mccuacmt.p`, `mccumt.p`, `mcexrmt.p`, `mgcodem1.p`, `mgcodemt.p`, `ppcommt.p`, `ppplmt1.p`, `pppsmt01.p`, `ppptmta.p`, `ppptmtb.p`, `rcctwba1.p`, `rcsomt.p`, `soivpst1.p`, `sosomt1.p`.

QXI Programs

The following are QXI-only programs that contain context fixes to make them work with QXI:

`adcsmtp.p`, `adstiq.p`, `advnai.p`, `advnai01.p`, `advnairs.p`, `advngtad.i`, `advngtsp.i`, `applhelp.p`, `apvomta.p`, `apvomta5.p`, `apvomtb.p`, `arcsrp03.p`, `ardunrpa.p`, `arsbac01.p`, `assedef.i`, `assettrp.i`, `assettrq.i`, `blankpage.p`, `bmpsmt.p`, `bmttps.i`, `bussvc.i`, `cmbkmta.p`, `compile.p`, `csptmt3a.p`, `datetime.i`, `domtitle.p`, `drppm.p`, `dscontxt.i`, `dsdois.p`, `dsexcptn.i`, `dsreqexc.i`, `ecapsbmt.p`, `eddmnset.i`, `eddomchg.p`, `edimsh02.p`, `edmfd.p`, `edmfdd.p`, `edmid.p`, `edmidd.p`, `edsd.p`, `edsdd.p`, `edxfd.p`, `edxfdd.p`, `excpmgpr.i`, `excpmsgs.i`, `fsbmismt.p`, `fscaimta.p`, `fscamte.p`, `fscamtm.p`, `fscareo1.p`, `fscarmta.p`, `fscarmtb.p`, `fscarmtc.p`, `fscarmtm.p`, `fscarmtr.p`, `fsci2frm.i`, `fsegrp03.p`, `fseomt02.p`, `fsquem1a.p`, `fsquem2a.p`, `fsquem3a.p`, `fsquem4a.p`, `fsrmamtu.p`, `fssam01.p`, `fssama01.p`, `fssamt1.p`, `fssamtm.p`, `fssrmg.p`, `fstzsel.p`, `GetTemplates.p`, `glcctrra.p`, `glcrfmt.p`, `gldabrp.p`, `gldabrpa.p`, `gldyiq2.p`, `glenchg.p`, `glrtrmta.p`, `glrvmta.p`, `gltriq2.p`, `gltriq2a.p`, `gltriq2b.p`, `gltriq2c.p`, `gltriq2e.p`, `gltrmta.p`, `gpaigach.p`, `gpaighst.i`, `gpapphlp.p`, `gpaudd.p`, `gpcmmt.p`, `gpcmmt01.p`, `gpdomcr.i`, `gpdomcr.p`, `gpfrcalc.p`, `gpfrlwt.p`, `gpgenfld.p`, `gpgetver.p`, `gpisutf8.i`, `gpiswrap.i`, `gplabel.p`, `gplocale.p`, `gpmenupl.p`, `gpmfgusr.p`, `gppage.p`, `gpqdoccr.p`, `gpqxicon.p`, `gpqdocrs.p`, `gpqxtddp.p`, `gprunp.i`, `gpstatus.p`, `gpttcmmt.i`,

gpttcmt.i, gpwait.i, gpxmldt.i, grrxmta.p, grrxmtx.i, gttxncmt.i, icintr.p, icisrc01.p, iclad01.p, iclomt.p, icshhdr.p, icshtrl.p, icsimt.p, icsrup.p, ictriq3.i, kbkbmtb.p, lgarinv0.p, lgload0.p, lvdef.i, lvlvcrp.p, lvlvcrpa.p, lvtt.i, lvucap.i, lvucap.p, mf1.p, mf1a.p, mfaiacc.i, mfaiact.p, mfaicon.p, mfaidea.p, mfaidflt.i, mfaidis.p, mfaigsta.p, mfaigtkn.p, mfaiiter.i, mfaimfg.i, mfaimsgh.p, mfaiocon.i, mfaipcon.i, mfairunh.p, mfaistrt.p, mfaisy01.i, mfaisy02.i, mfaisy03.i, mfaisy04.i, mfaisy05.i, mfaitrn.i, mfaiunrf.p, mfaiutl.p, mfapi.p, mfcqa.p, mfasai.p, mfcustom.i, mfdtitle.i, mfgp01.p, mfgproxy.i, mfinitpl.p, mflang.p, mflvai.p, mfsb.p, mfsemgfn.i, mfsessmg.p, mfsyad02.i, mfsyad02.p, mfwb01a.p, mfwb01aa.p, mfwb02.p, mfwsock2.p, mfww01.i, mfww01.p, mfww01a.p, mfww01b.p, mfwwc01.i, mfwwc01.p, mfwwsock.i, mgdccg.p, mgdcinit.p, mgflbook.p, mgprpgmt.p, mgtoolmt.p, mgusrtmt.p, mrmpiq01.p, mrmrppm.p, nrint.i, pjpjsub.i, poairsp.p, pogetscx.i, pogtie.i, pogtied.i, pogtpdct.i, pogtpo.i, pogtpoct.i, pogtpod.i, pogttx2d.i, popoai.p, popoai01.p, poporp.i, ppacln.p, ppacmt.p, ppacrl.p, ppdsisp.i, ppdsps01.i, ppdsps03.i, ppicxr.p, pppdspt.i, pppimt.p, ppplmt.p, pppsmt02.p, pppsmt03.p, ppptcp.p, ppptmta2.p, ppptmtc.p, ppptmtd.i, ppptmtd.p, pppttpt.i, ppttisp.i, ppttps01.i, ppttps03.i, pxdeclre.p, pxgblmgr.p, pxmsg.i, pxmsglib.i, pxmsglib.p, pxsevcon.i, pxtools.p, pxttmsg.i, pxunref.p, qqdp.p, qraexmsg.i, qxlu001.p, rcctwbb.p, rcrsupa.p, rcshmtb.p, rcshwb.p, rcshwbb.p, rcshwcar.p, rcsodf.p, rcsois1a.p, rcsomtb1.p, rcsomte1.p, rebkfl.p, rebkfli1.p, reisrc01.p, reisslst.p, reisssw.p, reoptr1a.p, rercvlst.p, reshmta.p, rqrqiq1a.p, rqrqiq1b.p, rqrqiq1c.p, rqrqmt0.p, rqrtmt0.p, rqtotdsp.p, rsgtschd.i, rsgtscho.i, rsporc.p, rssai.p, rssai01.p, rssairsp.p, rwdsro.i, rwromt.p, rwromt01.p, rwttro.i, scrolwkf.i, sectxtpr.i, sessctxt.i, sessmgpr.i, sfoptra.p, soise01.p, sopiwitt.i, sorp0501.p, sorp5a01.p, sosoisd.p, sosoism.p, sosomtf2.p, sosomtf3.p, sottatap.i, sovtesfm.i, svclibpr.i, svcproxy.i, svcpxcom.i, svcrqdsp.i, swindowb.i, swview.i, swviewn.i, ttcontxt.i, uiconfig.p, uiconfigentrytrig.p, uiconfiggotrig.p, uiconfigtriggers.p, utcahist.p, utdomain.p, utfsddel.p, utitmdtd.p, uxmsdtcr.p, vqapi.p, wbbr01.p, wbgp02.p, wbgpms01.p, wbhp01.p, wbhp01a.p, wbhp01b.p, wbhp02.p, wbhp01c.p, wbqu01.p, window6b.i, windowxt.i, woisrc01.p, wowois.p, wowoisc.p, wowoisrc.p, woworc.p, xmlengineshell.p, xxgpcmmt.p.

Appendix E

Installed Database Triggers

This section describes the database triggers that are installed with QXtend.

Overview

This section describes the database triggers that are installed with QXtend.

Triggers

The following table lists the QXtend database triggers.

Table E.1 QXtend Database Triggers

Table	Enterprise Edition		Standard Edition	
	Replication Write	Replication Delete	Replication Write	Replication Delete
abs_mstr	Y	Y	Y	Y
absc_det	Y	Y	Y	Y
absd_det	Y	Y	Y	Y
absl_det	Y	Y	Y	Y
absr_det	Y	Y	Y	Y
ac_mstr			Y	Y
ad_mstr			Y	Y
an_mstr	Y	Y	Y	Y
anl_det	Y	Y		
ans_det	Y	Y		
bom_mstr	Y	Y		
ca_mstr	Y	Y	Y	Y
ccd_mstr			Y	Y
ccd_mstr	Y	Y	Y	Y
cd_det	Y	Y	Y	Y
cm_mstr			Y	Y
cmt_det	Y	Y	Y	Y
code_mstr	Y	Y		
com_mstr			Y	Y
cp_mstr	Y	Y	Y	Y
cr_mstr			Y	Y
cs_mstr	Y	Y		
dpt_mstr	Y	Y		
ecd_det	Y	Y	Y	Y
egt_mstr	Y	Y	Y	Y
eng_mstr	Y	Y	Y	Y
eu_mstr			Y	Y
fcs_sum	Y	Y		
fsc_mstr	Y	Y	Y	Y
fspd_det	Y	Y		
fwk_mstr	Y	Y	Y	Y

Table E.1 — *QXtend Database Triggers* — (Page 1 of 3)

Table E.1 QXtend Database Triggers

Table	Enterprise Edition		Standard Edition	
	Replication Write	**Replication Delete**	**Replication Write**	**Replication Delete**
idh_hist	Y	Y	Y	Y
ie_mstr	Y	Y	Y	Y
ied_det	Y	Y	Y	Y
ih_hist	Y	Y	Y	Y
in_mstr	Y	Y	Y	Y
isb_mstr	Y	Y	Y	Y
is_mstr	Y	Y		
isd_det	Y	Y		
itm_det	Y	Y	Y	Y
lacd_det	Y	Y	Y	Y
ls_mstr			Y	Y
pc_mstr	Y	Y		
pi_mstr	Y	Y	Y	Y
pid_det	Y	Y	Y	Y
pj_mstr			Y	Y
pl_mstr	Y	Y		
po_mstr			Y	Y
pod_det			Y	Y
ps_mstr	Y	Y	Y	Y
pt_mstr	Y	Y	Y	Y
ptp_det	Y	Y	Y	Y
ro_det	Y	Y		
sb_mstr			Y	Y
sch_mstr	Y	Y	Y	Y
schd_det	Y	Y	Y	Y
sct_det			Y	Y
scx_ref	Y	Y	Y	Y
si_mstr	Y	Y		
sob_det	Y	Y	Y	Y
sod_det	Y	Y	Y	Y
so_mstr	Y	Y	Y	Y
sodlc_det	Y	Y	Y	Y
spt_det			Y	Y
tr_hist	Y	Y	Y	Y
tx2d_det	Y	Y	Y	Y
um_mstr	Y	Y		
vp_mstr	Y	Y		
wc_mstr	Y	Y		

Table E.1 — *QXtend Database Triggers* — (Page 2 of 3)

Table E.1 QXtend Database Triggers

Table	Enterprise Edition		Standard Edition	
	Replication Write	**Replication Delete**	**Replication Write**	**Replication Delete**
wo_mstr	Y	Y	Y	Y
wod_det	Y	Y	Y	Y
wr_route	Y	Y	Y	Y

Table E.1 — *QXtend Database Triggers* — (Page 3 of 3)

Product Information Resources

QAD offers a number of online resources to help you get more information about using QAD products.

QAD Forums (community.qad.com)
Ask questions and share information with other members of the user community, including QAD experts.

QAD Knowledgebase (knowledgebase.qad.com)*
Search for answers, tips, or solutions related to any QAD product or topic.

QAD Document Library (www.qad.com/documentlibrary)
Get browser-based access to user guides, release notes, training guides, and so on; use powerful search features to find the document you want, then read online, or download and print PDF.

QAD Learning Center (learning.qad.com)*
Visit QAD's one-stop destination for all courses and training materials.

*Log-in required

Index