QAD Enterprise Applications

Installation Guide
QAD User Interfaces

System Overview
Implementation
System Administration and Reference

78-XXXX
QAD .NET UI 2.9.x
June 2010

UserInterfaces_IG_v020901.pdf/dmk/dmk

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

Contents

Chapter 1

Introduction

What is in this Guide?

Use this guide to install, configure, and update QAD user interfaces. The guide is divided into three sections: System Overview, Implementation, and System Administration and Reference.

The system overview section contains instructions on how to prepare for the user interface installations by planning the deployment and preparing these prerequisite components:

- Java Development Kit
- Progress WebSpeed
- Windows Telnet Server

The implementation section includes instructions to:

- Install the Tomcat servlet engine.
- Install the files for the QAD user interfaces.
- Configure single and multi-tier environments.
- Configure the UI environment.
- Install and start a QAD .NET UI client.

The reference section contains instructions for performing optional user and system administration activities. It also contains reference information to help you better understand the user interface environment and a glossary of terms.

Audience

These instructions are intended for a system administrator with experience of installing QAD Enterprise Applications, as well as configuring and managing hardware and operating system software. This person should also have a good understanding of networking concepts and administration.

Product Naming

MFG/PRO eB2.1 Service Pack 5 was identified as QAD Enterprise Applications 2007 and MFG/PRO eB2.1 Service Pack 6 was identified as QAD Enterprise Applications 2007.1. MFG/PRO eB2.1 Service Pack 7 is identified as QAD Enterprise Applications 2008 - Standard Edition (QAD 2008 Standard or QAD 2008 SE).

This document typically uses QAD 2008 Standard or QAD 2008 SE instead of MFG/PRO. For example, MFG/PRO database is now QAD 2008 SE database. Please be aware that although the product name has changed, the term MFG/PRO continues to be used in certain aspects of the installation process. Consequently, this guide includes references to QAD Standard Edition, as well as MFG/PRO.

Important The installation procedures described in this document apply to QAD 2008 Standard Edition, and to all Standard Editions subsequent to this version.

Installation Updates

Check the QAD Web site to make sure you have the latest installation errata, installation guides, and installation media.

http://support.qad.com/

Related Documentation

Progress Documentation

- For information on installing and configuring WebSpeed, see the *WebSpeed Installation and Configuration Guide*.

Find the complete Progress documentation set online at:

http://www.progress.com/products/documentation/index.ssp

QAD Documentation

- For information on installation or converting to a more recent release, refer to the appropriate installation or conversion guides for your system.
- For information on using product features, refer to the *User Guides*.
- For information on using the QAD .NET UI, refer to *User Guide: QAD .NET User Interface* and *User Guide: Introduction to Enterprise Applications*.

For QAD customers with a Web account, documentation is available for review or downloading at:

http://support.qad.com/

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

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

Conventions

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

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

QAD Support Services

The QAD user interfaces have a wide variety of configuration possibilities, are highly scalable, and can be customized easily. While this guide provides basic installation and configuration information, it does not cover all possible computing environments and variations into which the QAD user interfaces can be implemented.

To take full advantage of the flexibility and potential of QAD's user interfaces in your specific environment, contact your QAD Support representative for information on the installation and customization offerings supplied by QAD Support Services. These offerings include performance enhancements as well as technical and administration training.

Section 1

System Overview

This section provides information needed before you begin implementing QAD user interfaces in your environment.

Chapter 2

Architectural Overview

This chapter provides an overview of the architecture of the QAD user interfaces. It describes the various components you will install and configure and discusses issues you should understand as you begin to plan your installation.

Introduction

There are three distinct user interfaces:

- The Character interface, which you can use after a simple installation of the product.
- The GUI client, which lets you run the system in a full graphical interface against a Windows file server.
- The QAD .NET UI, which provides a common framework for multiple QAD applications based on Microsoft .NET technology.

The installation of the first two interfaces is covered in the base installation guides for QAD 2008 Standard. The installation and configuration of the QAD .NET UI is described in this guide.

The QAD .NET UI provides a full-fledged user interface with a high level of user flexibility using Microsoft .NET technology. Advantages include:

- Multiple open workspaces
- Rearranging of windows in a workspace
- Enhanced menus with full search, multiple application access, security controls, and user-configuration
- Enhanced browses with drag-and-drop columns; column hide, resize, and sort; simple and complex searches; user-defined data views; drill-down and linking; and output to Microsoft Excel

When you have completed a standard installation, you can open a browser on a client machine and point to the QAD .NET UI home URL; a brief automated client install runs. When this process completes, the QAD .NET UI can be launched from the Windows Start menu or a desktop icon.

The following sections briefly describe the shared user interface components. These descriptions are intended to provide a framework for understanding the user interface architecture as you begin your implementation.

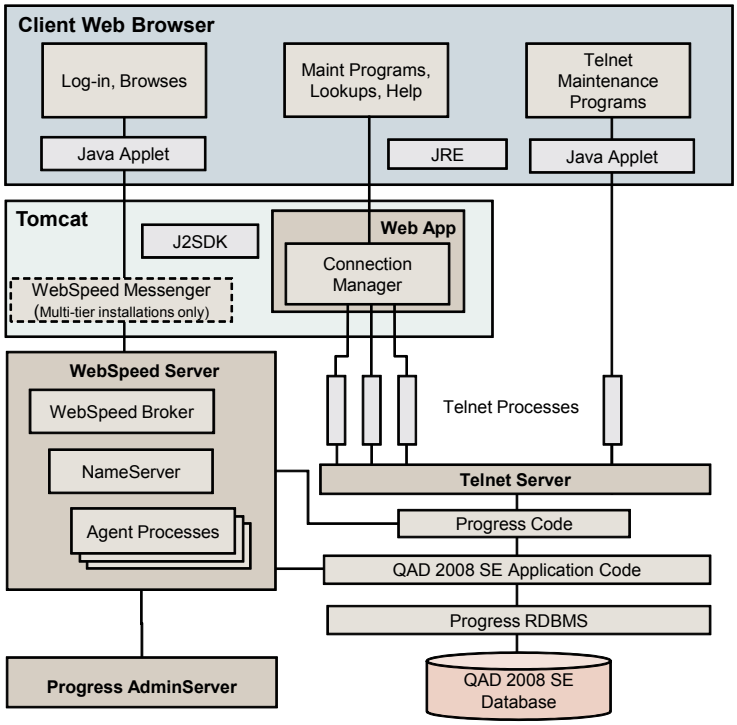
UI Architecture

Although all system functions display in the UI browser, the underlying technology used to support the different displays varies. There are three types of screens in the UI supported with different application components:

- Maintenance programs, lookups and help, which are generated and managed by the Tomcat application server.
- Java browses are generated by Progress WebSpeed agents. WebSpeed also manages the UI log-in.
- Embedded telnet screens look and function similarly to character client screens. For telnet programs, the Web browser runs a telnet client. This telnet client is used for screens that do not function properly as HTML maintenance screens.

Figure 2.1 illustrates the basic components in the UI architecture and their relationships. Each component is discussed briefly following the diagram. For details on specific versions and other component requirements, see Chapter 3, “System Requirements,” on page 15.

Fig. 2.1
UI Architecture



In Figure 2.1, the two components that comprise the .NET UI are the UI application installed under Tomcat and the UI Progress code, installed in a subdirectory under your QAD 2008 Standard installation directory. These directories are placed in the proper locations by the installation script and MFG/UTIL during the UI installation and build process.

UI systems are typically built for multiple environments—pilot, production, training, development, and so forth. Each QAD UI system has a unique name, which is reflected in the QAD 2008 Standard and Tomcat directory structures:

- *MFGPROInstallDir*\qaduiConfig
- *TomcatInstallDir*\webapps*qaduiConfig*

A list of top-level directories created beneath these directories and a description of their content are found in “Installed Components” on page 134.

Multiple-Tier Installation

There are several ways to deploy the .NET UI in a multi-tier environment. Each environment has different characteristics; this section describes two common scenarios.

Standard Two-Tier Installation

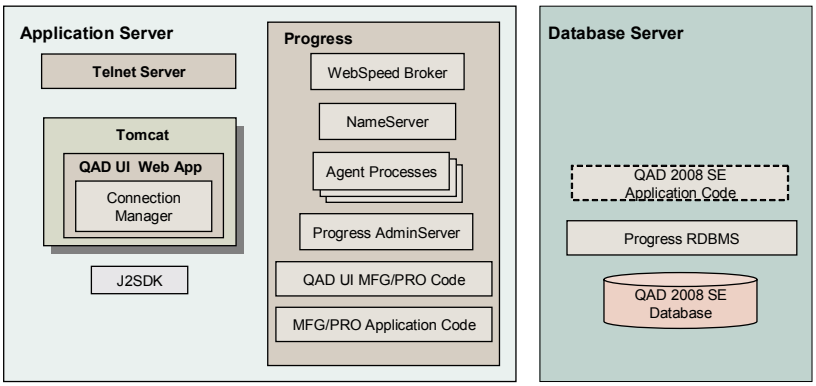
In a standard two-tier installation, the QAD 2008 Standard (MFG/PRO) database server hosts most—if not all—of the system and Progress components. A second server hosts Tomcat and the UI Web application. WebSpeed Messenger must be installed on the Tomcat server.

Isolated Database Installation

Another type of two-tier deployment supports a dedicated database server. The UI installation, compile, and configuration occur on the Tomcat server. This two-tier deployment works well in regulated environments where any change to the database server configuration requires validation.

All UI and related components are installed and configured on the Tomcat server. Communication between the servers occurs using client networking. A full Progress installation, including WebSpeed, is required on the Tomcat server.

Fig. 2.2
Isolated Database Server Deployment



In this arrangement, having the QAD 2008 Standard code on the database server is not required, but increases performance for disk-intensive tasks such as MRP and batch processing.

Client Web Browser

Each UI client requires Microsoft Internet Explorer; other browsers are not supported. Other components used by the Web browser on the client include:

Java Runtime Environment (JRE). The Java Virtual Machine (JVM) executes platform-independent Java code on a specific operating system. .NET UI clients may require a more recent version of the JVM than the browser default. Depending on the client configuration, the UI client startup process may automatically install a compatible JRE, which includes the appropriate JVM.

UI Java Applet. The UI Java code is contained in the `multinet.jar` Java archive (JAR) file. This applet manages the communication between the log-in, browsers, and telnet programs and the QAD 2008 Standard system. It is installed as part of UI under Tomcat.

JAR files are managed by the JRE and are automatically downloaded when the version of the JAR file on the client is older than the version on the server. They are also automatically downloaded the first time a client needs them; special setup is not required. JAR files are located in the user's Windows profile directory in a subdirectory called `java_plugin_AppletStore`.

Tomcat Web Application Server

The .NET UI uses the Tomcat Servlet/JSP container to manage the communication between .NET UI client sessions and the QAD 2008 Standard system through the HTML interface. The UI media includes a customized version of Tomcat, which is the recommended version to use. It is installed as part of the installation script.

The .NET UI is installed as a client application under Tomcat. After files are installed, you must configure the UI Connection Manager with installation-specific settings to control and manage the pool of telnet sessions on the telnet server.

Components of the Java Development Kit (JDK) are used by Tomcat to enable communication between the client and the Tomcat servlet container. You may need to download the JDK from the Sun Microsystems Web site.

In a multi-tier environment when Tomcat is on a different host than the WebSpeed server, you must install the WebSpeed Remote Messenger under Tomcat to support communication with WebSpeed. If you use multiple Tomcat servers, each should have the same version of Tomcat and JDK installed.

QAD and Progress Components

The .NET UI provides a user interface for updating and viewing data in a QAD 2008 Standard (MFG/PRO) database. QAD 2008 Standard must exist in your environment before you begin implementing a UI. This guide assumes you have already installed and configured it.

In addition to the Progress software required for QAD 2008 Standard, the .NET UI uses the Progress NameServer, AdminServer, and WebSpeed server products to support the Java browses and authenticate users during log-in.

Telnet Server

The .NET UI uses a telnet server for two purposes:

- On the database server, it is used to run a pool of telnet sessions that support maintenance programs, reports, and inquiries.
- It enables the client telnet interface for a limited subset of QAD 2008 Standard programs and any custom programs that do not conform to QAD programming standards.

If you plan to use a UNIX machine for the telnet server, you can use the default telnet service provided with the operating system. If you plan to use a Windows machine for the telnet server, you must use the Georgia SoftWorks telnet server software. This software is included on the QAD UI installation media.

QAD .NET UI

The QAD .NET UI provides a common framework for multiple QAD applications. This framework—based on Microsoft .NET technology—has excellent performance and provides best-practice usability and deployment features, including powerful, configurable browses.

Important Be sure to refer to he *QAD .NET UI Release Notes* for the version of the QAD .NET UI you are installing.

QAD .NET UI Deployment

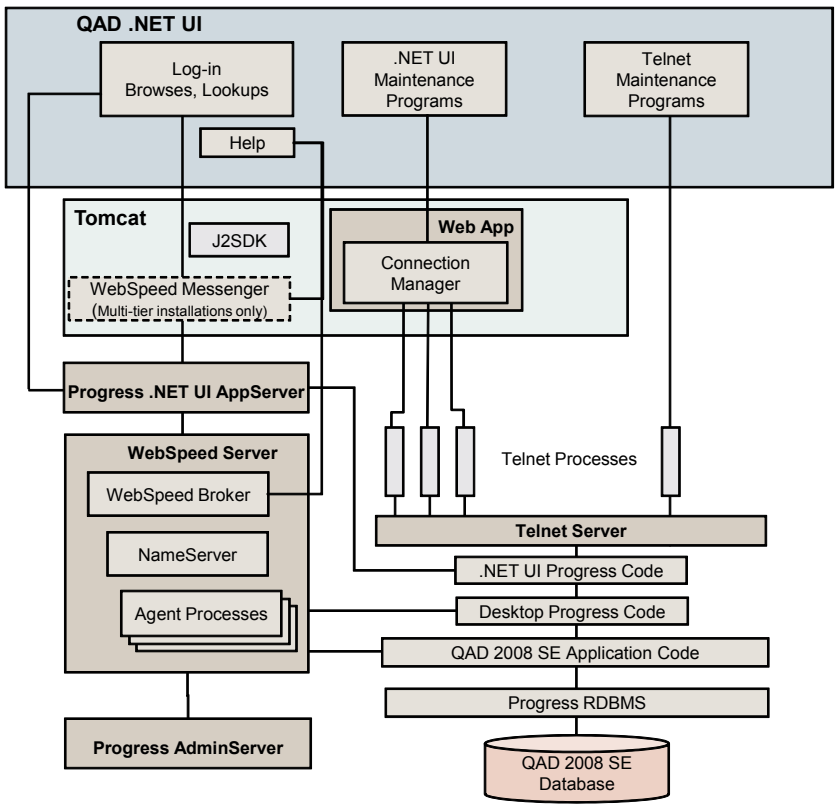
One QAD .NET UI configuration is required for each of your named QAD configurations. QAD .NET UI server components are created on disk as part of the general install, but they are not integrated or used until you have explicitly gone through the QAD .NET UI install. During the general install, some additional fields are required during configuration, and a Progress AppServer must be started on the server.

The QAD .NET UI Appserver manages the data and HTML page flow from the database server. The QAD .NET UI server is installed as a Tomcat Web application, called by Progress through the `ubroker.properties` file.

The QAD .NET UI client runs in the Microsoft .NET framework with a brief automated installation on the client machine.

Note The direct client telnet server connection in the QAD .NET UI can only be used on UNIX systems with US English.

Fig. 2.3
QAD .NET UI Architecture



See “Configure Multiple UI Systems” on page 140.

If you have more than one installed version of the UI, you can choose the UI configuration you want to access from the QAD .NET UI using the System Preferences setting accessed from Tools| Options. This can be useful if you have test and production systems that you are using at the same time.

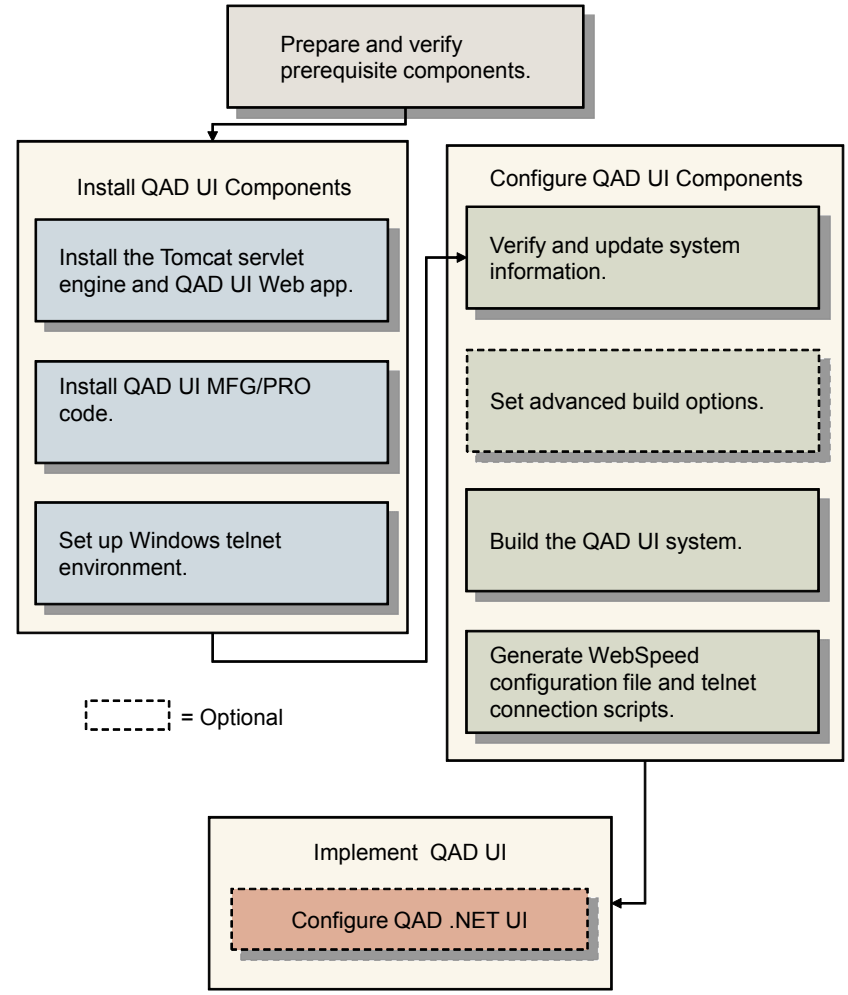
Planning an Installation

As a general rule, do not install the QAD user interfaces in a production environment until you have installed it in a test environment.

Single-Tier Installation Overview

Figure 2.4 summarizes the installation and configuration activities for a single-tier QAD UI environment. Once a UI environment is in place, the QAD .NET UI requires a few additional steps.

Fig. 2.4
Single-Tier Configuration Work Flow



Multi-Tier Installation Overview

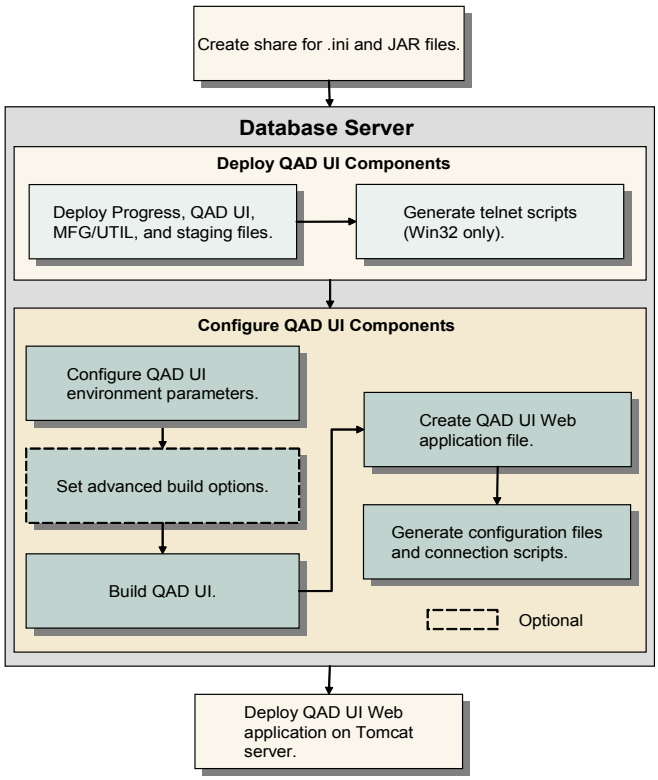
A multi-tier QAD UI installation requires additional steps. Several deployment options are possible. If you are not using a basic two-tier installation, ensure you understand the QAD architecture and requirements first.

Note Multiple QAD UI systems can be installed on the same server or for multiple MFG/PRO versions on the same server. For more information, see “Configure Multiple UI Systems” on page 140.

A multi-tier installation requires a shared network directory accessed by both the Tomcat server and the database server. This directory share stores an `.ini` file, which contains information used by the installation tools. Alternatively, transferring this file and other shared files is required.

As with a single-tier install, once a .NET UI environment is in place, the QAD .NET UI requires additional steps. These are not shown in Figure 2.5.

Fig. 2.5
Multi-Tier Configuration Work Flow



Plan the Installation

Before you begin, plan where each distributed component will be installed. Make sure that:

- The user account you plan to use on each server has administrator access.
- The shared staging directory for the configuration `.ini` file is accessible from all host machines and you have read/write access to it from all servers, or that the file transfer mechanism you plan to use is working properly for all servers.
- If a full WebSpeed installation is not possible on the application server, the free WebSpeed Messenger has been installed. See “Install WebSpeed Messenger” on page 135.

Chapter 3

System Requirements

A QAD UI implementation requires planning for adequate system resources. This chapter provides guidelines you can use to generate estimates of the hardware and software requirements.

Database Server Requirements

The following list shows the minimum requirements for QAD .NET UI. For QAD 2008 Standard system requirements, refer to the installation guide for your system.

- Operating systems include most Linux, UNIX, and Windows releases.
- The required Progress 9.1 or OpenEdge 10 software components for MFG/PRO, including:
  - Latest Progress version-specific patches (minimum Progress 9.1D SP7 or OpenEdge 10 SP3)
  - Progress AdminServer
  - Progress WebSpeed 3.1

  Note If you are completing a multi-tier deployment and do not have WebSpeed installed on the Tomcat server, you can download and install the free WebSpeed Messenger. For more information, see "Install WebSpeed Messenger" on page 135.
- Internet access
- MFG/PRO eB2 Service Pack 7 or higher or MFG/PRO eB2.1 Service Pack 2 or higher (latest service pack recommended).

  Note MFG/PRO eB2.1 Service Pack 5 was identified as QAD Enterprise Applications 2007 and MFG/PRO eB2.1 Service Pack 6 was identified as QAD Enterprise Applications 2007.1. MFG/PRO eB2.1 Service Pack 7 is identified as QAD Enterprise Applications 2008 - Standard Edition (QAD 2008 Standard).
- For MFG/PRO eB2 Service Pack 7, download ECO P2TW from the QAD support Web site.
- For MFG/PRO eB2 Service Pack 8, only ECO P2TW is needed.

Verify Language Compatibility

QAD .NET UI is configured to support a defined set of languages and associated code pages based on the underlying requirements of the QAD 2008 Standard (MFG/PRO) database. The supported set has changed over time. See Chapter 10, "Implementing Multiple Languages," on page 105.

If you are installing QAD .NET UI in a non-US English environment, verify that your database is using a currently supported code page as listed in Table 10.2, "Supported Progress Code Pages," on page 109.

QAD .NET UI Client Requirements

Operating Systems

The following operating systems are supported:

- Microsoft Windows 2000 and XP recommended
- Microsoft Windows 98, ME, and NT 4.0 supported

Minimum Client Hardware

- Memory (RAM):

  - Windows NT, 2000, and XP: 256 MB minimum (512+ recommended)
  - Windows 98 and ME: 256 MB minimum (512+ recommended)
- Intel P4 or faster processor (1.6+ GHz recommended)
- At least 150 MB of disk space, depending on the version of Java, including 20 MB reserved for static HTML file caching and 100 MB for other temporary internet files
- Ability to display 65,536 or more colors
- Ability to display at least 1024 x 768 resolution

Note Display settings should be set to the small font size. Using large fonts affects some QAD .NET UI display properties.

Client Software

Client Browser Requirements

- The latest version of Microsoft Internet Explorer.
- Adobe Scalable Vector Graphics (SVG) plug-in 3.03 for the Process Editor or viewing process maps created with the Process Editor. This component is copied to the Tomcat server during the QAD UI installation. It is downloaded by clients when needed.
- Java Runtime Environment (JRE): 1.5.0_06 is recommended. 1.4 versions are not supported, but can be used using the workaround instructions described in “Compatability with Java Versions” on page 18. The JRE includes the Java Virtual Machine (JVM).

Important Depending on the languages supported by the Windows operating system on the client PC, the typical install option for JRE may not install components required to support international character sets. To avoid this, the JRE must be installed by choosing the custom setup type and selecting the additional language support option. To verify support for non-European languages, check for the existence of the following file:

```
JavaInstallDir/jre/lib/charsets.jar
```

QAD .NET UI Server Requirements

All requirements for the QAD .NET UI server are identical to those for the database server with the following limitations:

- Progress OpenEdge 10, SP3 or higher
- MFG/PRO eB2.1 Service Pack 4 and higher.

  Note MFG/PRO eB2.1 Service Pack 5 was identified as QAD Enterprise Applications 2007 and MFG/PRO eB2.1 Service Pack 6 was identified as QAD Enterprise Applications 2007.1. MFG/PRO eB2.1 Service Pack 7 is identified as QAD Enterprise Applications 2008 - Standard Edition (QAD 2008 Standard).

The requirements for the client machine are:

- Windows XP Service Pack 2 or higher
- Memory (RAM) 512 MB or higher

Note If Microsoft Excel is not installed on the client machine, exporting browse data to Excel results in the error: No application is associated with the specified file for this operation.

Microsoft .NET 1.1 Framework Installation Issue

Version 2.0 of the Microsoft .NET Framework is currently supplied on the QAD User Interfaces installation media. The following only applies if you are installing QAD .NET UI version 1.1.

In previous releases, QAD .NET UI version 1.1 included version 1.1 of the Microsoft .NET Framework. The QAD .NET UI works correctly with version 1.1 of the .NET Framework in almost all cases. In a very few cases, the following error has been encountered during initial log-in to the QAD .NET UI using Version 1.1:

```
ERROR [24] QAD.Shell.ShellLoader?. HandleUncaughtException(0) => { Outer: The
type initializer for "System.Net.Dns" threw an exception. Base: An operation
on a socket could not be performed because the system lacked sufficient buffer
space or because a queue was full.
```

If this error occurs, follow these steps to correct the problem:

1. Completely uninstall Version 1.1 of .NET.
2. Go to the Microsoft Web site and download .NET Service Pack 1.

   `http://www.microsoft.com/downloads/details.aspx?familyid=A8F5654F-088E-40B2-BBDB-A83353618B38&displaylang=en`
3. Reinstall Version 1.1 of .NET and then install the service pack.

Tomcat Server Requirements

Software

- Progress WebSpeed Messenger (multi-tier deployments only)
- Java 2 SDK 1.5.0_06+
- Tomcat 5.5.9 ships on QAD UI media and is recommended; you may also use Tomcat version 5.0.27+

Preserving Earlier Tomcat Versions

By default, if you have a Tomcat version that is earlier than 5.5.9, the QAD UI install script overwrites the earlier version. To avoid this, add a text file named `version.txt` into *`TomcatInstallDir`*. The content of the file should read `tomcat.version=<`*`your current Tomcat version`*`>`. For example:

```
tomcat.version=5.5.15
```

Compatability with Java Versions

Java 1.4 versions may be used but are not supported. 1.4 versions are incompatible with the Tomcat 5.5.9 that is delivered on the QAD UI media without additional configuration.

In general, the best solution is to install J2SE 5.0 or later on the Tomcat server. If you require J2SDK 1.4, refer to the `RUNNING.txt` file located in the *`TomcatInstallDir`* for configuration steps. The compatibility files referenced in `RUNNING.txt` are available on the QAD UI media in the `/tools` directory in `jakarta-tomcat-5.5.9-compat.zip`.

Make sure to unzip `jakarta-tomcat-5.5.9-compat.zip` to a temp directory, which creates a `/jakarta-tomcat-5.5.9` directory. You then copy all files under the temp directory over your existing *`TomcatInstallDir`*:

```
cd unzip/jakarta-tomcat-5.5.9
cp -rp * TomcatInstallDir
```

You then start Tomcat with `setenv.sh` pointing to the Java 1.4 SDK.

Note If you encounter on-going problems with the Java 1.4 SDK, replace this version with version 1.5.0_06x.

Disk Space Requirements

The Tomcat server must have enough disk space for the components listed in Table 3.1.

Table 3.1
Disk Space Requirements

QAD UI Component	Disk Space
Tomcat installation files	10 MB
WebSpeed software	100 MB, less than 1 MB for WebSpeed Messenger
QAD UI client application	100 MB

Note On AIX systems, you must remove the `-server` argument from the `CATALINA_OPTS` value in `setenv.sh`. The `setenv.sh` file is located in *`TomcatInstallDir`*`/bin`.

Progress Server Requirements

Operating Systems

QAD UI clients communicate with the Progress servers using network protocols and do not require that the machines on which these servers run use a specific operating system.

You can use QAD .NET UI with the following platforms:

Note Contact your QAD sales representative for the latest supported platforms and versions.

- Linux (recommended)
- Sun Solaris (SPARC)
- Compaq UNIX (Tru64)
- IBM AIX
- HP-UX
- Windows (Intel)

Java Requirements

Depending on your Progress server hardware and operating system configuration, a version of the JDK may be installed as part of your Progress installation, or you may be required to install Java before installing Progress.

The minimum Java version is J2SE 5.0.

If Progress installed its own Java version and your Progress server is on the same machine as Tomcat, do not remove the Progress-installed JDK. If you need to install a later JDK version for Tomcat, do so without uninstalling the Progress version.

If you must install Java before installing Progress, install the version required by the .NET UI.

Hardware

The Progress servers execute the QAD 2008 Standard code for the QAD UI clients. The machine on which these servers run should be sized similar to a server running QAD 2008 Standard for an equivalent number of character clients.

Progress Components

- Complete Progress installation

  Perform a complete installation of the Progress components by choosing the complete installation option.
- Progress OpenEdge 10, Service Pack 3+.
- Progress WebSpeed Transaction Server, Version 3.1, with sufficient WebSpeed agent licenses to support expected transaction volume. See “WebSpeed Agent Sizing and Behavior” on page 136.

  If you are using a local host connection, WebSpeed must be installed on the QAD 2008 Standard database server. For client/server connections, install the WebSpeed software on either the database or Web server.

Important If your telnet server is on the same machine as your Progress components, the 4GL code is also required.

Verify Progress Components

Depending on your operating system, validation steps and commands will vary. The first step should be to validate that you have the needed components using the `showcfg` command.

Use the following sequence of command-line entries to validate Progress components. The database servers must be running before starting the WebSpeed brokers. Run these from *`ProgressInstallDir`*`/bin`:

- AdminServer

```
./proadsv -start
./proadsv -query
```

- NameServer

```
./nsman -name NS1 -start
./nsman -name NS1 -query
```

- WebSpeed

```
./wtbman -name wsbroker1 -start
./wtbman -name wsbroker1 -query
./wtbman -name wsbroker1 -stop
```

Windows Telnet Server (Windows Only)

The Georgia SoftWorks Windows Telnet Server is required for Windows telnet server implementations. This component is provided on the QAD UI installation media. However, you should download the latest version of the software to obtain the most recent patches and functionality:

http://www.georgiasoftworks.com/

The minimum version is 6.5.0.38.

Refer to the Georgia SoftWorks documentation for software and system sizing requirements. For installation information, see “Install the Windows Telnet Server” on page 36.

Section 2

Implementation

This section provides instructions for implementing the QAD user interfaces in your environment.

Chapter 4

Preparing the Environment

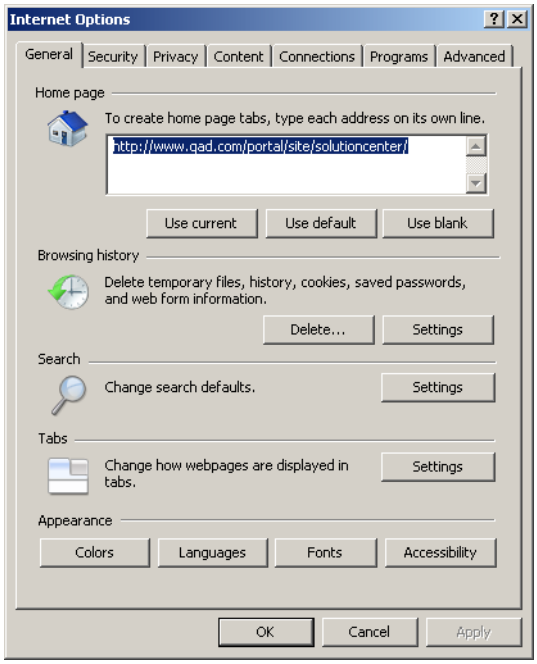
Use this chapter to set up your environment before installing the QAD UI software.

Modify Client Browser Settings

Make the following changes to your Internet Explorer settings.

1 Open an Internet Explorer browser and choose Internet Options from the Tools menu.

On Internet Explorer 7, the Internet Options menu is as follows:



If you are using Internet Explorer 7, under the Advanced tab, choose Restore Advanced Settings and then choose OK.

2 Open Internet Options again.

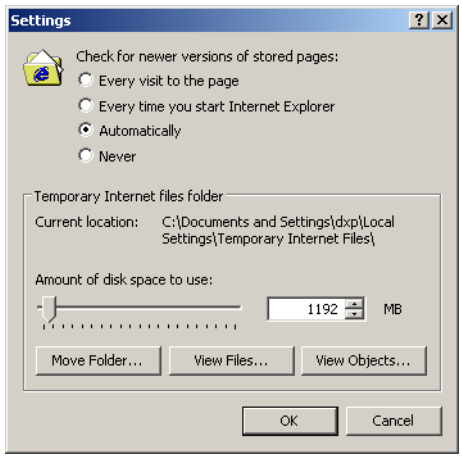
If using Internet Explorer 7, select the General tab and in the Browsing History section, choose Settings.

Make the following changes:

- Click on Automatically under Check for newer version of stored pages.
- Set the Amount of disk space to use to greater than 100 MB.

Choose OK to continue.

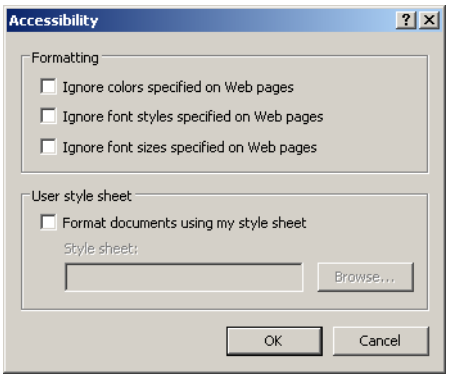
Fig. 4.1
Temporary Internet Files Settings



3 Under the General tab, choose Accessibility.

QAD user interfaces use colors and fonts to distinguish various screen elements such as menus and input frames. Uncheck all accessibility options.

Fig. 4.2
Internet Accessibility Options



Choose OK to continue.

4 Under the Security tab, choose Custom Level. Scroll down to make the following changes.

- Under ActiveX controls and plug-ins, scroll to Run ActiveX controls and plug-ins and choose Enable.
- Under Miscellaneous, scroll to Non-encrypted form data and select Enable.
- Under Scripting set all three types—Active scripting, Allow paste operations via script, and Scripting of Java applets—to Enable.

Choose OK to continue.

5 Under the Privacy tab, choose Advanced. Make the following changes:

- Select Override automatic cookie handling.
- Under First-party Cookies, select Accept.
- Under Third-part Cookies, select Prompt.
- Select Always allow session cookies.

Fig. 4.3
Cookies Settings

Choose OK to continue.

6 Choose OK to exit Internet Options.

Windows XP Service Pack 2 Settings

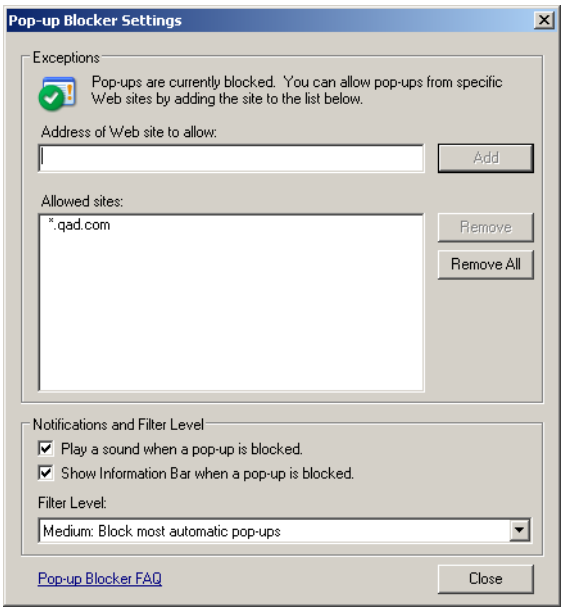
If your client PC is running Windows XP Service Pack 2, you must add the QAD UI launch point as a trusted site in Internet Explorer in order to use the browse output-to-spreadsheet option.

1 In Internet Explorer, choose Tools|Internet Options.

2 Under the Privacy tab in the Pop-Up Blocker section, choose Settings.

3 Enter the QAD UI server domain name in Address of Web site to allow; for example, `corpname.com`. Choose Add to save the name.

Fig. 4.4
Pop-Up Blocker Settings

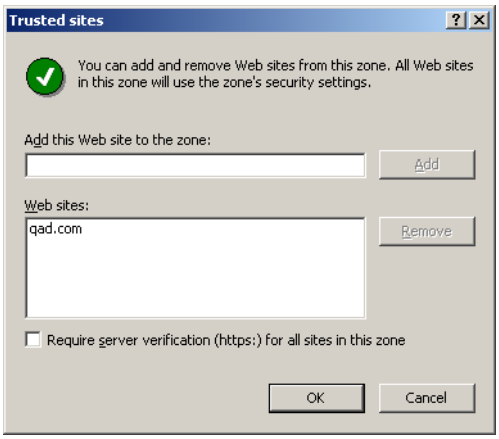


4 Choose Close to continue.

5 Under the Security tab, click Trusted Sites and then choose Sites.

6 In the Trusted Sites screen, uncheck Require server authentication (https:) for all sites. Then enter the same domain name again and choose Add. Choose OK to continue.

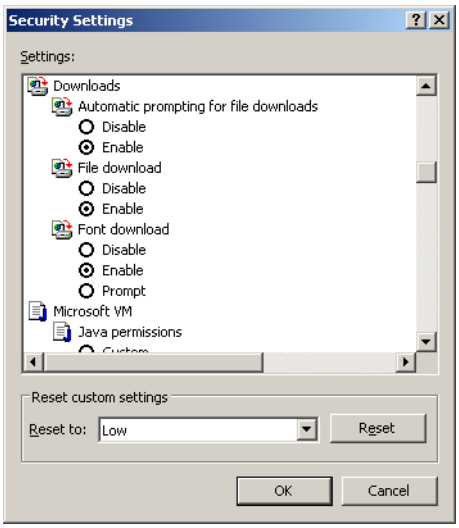
Fig. 4.5
Adding a Trusted Site



7 Under the Security tab, choose Custom Level. Scroll to Downloads. Enable the following settings and choose OK.

- Automatic prompting for file downloads
- File download

Fig. 4.6
Setting Download Options



8 Choose OK to exit Internet Options.

Install the Java Development Kit

Install the latest version of the Java J2SDK on the Tomcat server. Download J2SDK for Tomcat from the Sun Microsystems Java Web site:

www.java.sun.com

After you download the setup file, start the installation by launching it. The installation process is automatic; accept all the default settings and grant any authorization requests.

To verify the install, at the UNIX or DOS prompt, enter:

```
java -version
```

The system displays information about the current Java version. If you receive an error message, make sure your PATH environment variable is set correctly.

Set JAVA_HOME

Tomcat and the .NET UI Connection Manager require the JAVA_HOME variable. Set this with operating system commands or add it to system or user startup scripts.

In UNIX environments, enter:

```
export JAVA_HOME=/javapath
```

In Windows environments, enter in a DOS window:

```
set java_home=c:\javapath
```

The Tomcat installation configures the `catalina.bat` or `catalina.sh` file to automatically set the JAVA_HOME variable based on your input. Record the JAVA_HOME information; you will need to verify it in later steps.

Install the Windows Telnet Server

For telnet connections on Windows servers, install the Georgia Softworks Telnet Server (GSWTS). A version is provided on the QAD UI installation media. Check the Georgia Softworks Web site for an updated version.

The Georgia Softworks Power Features Pack included in the general release provides a session monitor, which is helpful for troubleshooting telnet connection issues.

Install the Telnet Server

1. If you are using the QAD-provided version, copy the release-specific `.zip` file from the `/tools` directory on the QAD UI CD to a local drive.
2. Unzip the `.zip` file to a work directory.
3. In the work directory, double-click `setup.exe` to begin installation.
4. At the Welcome window, click Continue.
5. Select Full Install and click Continue.
6. Enter the GSWTS installation directory and choose Continue. Use the default `\gs_uts` installation directory. You can place this directory on any drive that can be accessed through the network.

7 When GSWTS is installed, the Setup Succeeded window displays and GSWTS starts automatically.

Register the Georgia SoftWorks Software

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

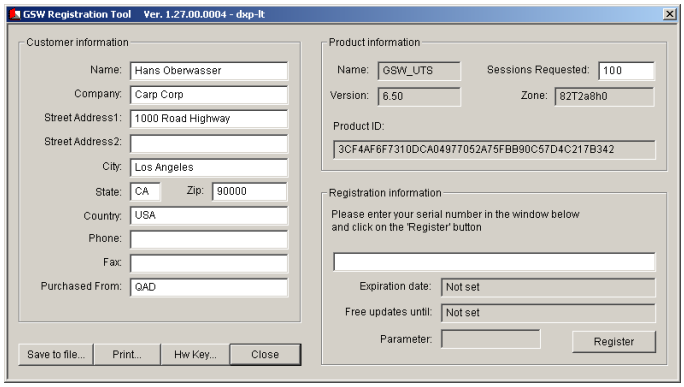
Important If you need to reinstall or are planning to move your installation to a different platform, or if you are a sales agent or a distributor, include that information on the registration.

1 Select the Registration icon from the GSWTS program group in the Start menu.

2 In the Georgia SoftWorks Product Registration window, enter your customer information. The information that displays in the Product Information section is system-generated.

3 Set Sessions Requested to 100. This is the number QAD automatically supplies with your registration.

Fig. 4.7
Georgia Softworks Registration



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

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

registration@georgiasoftworks.com

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

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

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

Define QAD UI Security Groups

Three user groups have access to QAD UI functions:

- A system administration group
- A design group to create simplified screens
- A Process Editor user group

For more information, see “QAD UI Administration” on page 96.

Note There is also a Tomcat administrative group. A link to the Tomcat admin page is provided from the UI administration page. Users clicking this link are prompted for a user ID and password authenticated through Tomcat; this is not the same as the QAD 2008 Standard user ID and password unless you set them up to be the same.

UI Administrative Group

Users accessing the UI administration page must supply a user ID and password. These users must be members of a security group. By default this is `qadadmin`, but can be defined by the system administrator. This group has the authority to manage the QAD UI system. The default user is `mfg`, but other users can be defined for this group.

UI Design Group

A separate UI design group that can create simplified screens is also defined in the UI administration page. This can be the same user group specified during installation, but it does not have to be. Typically, the administrative group for QAD UI is systems personnel, while the UI design group includes users with in-depth knowledge of the system.

Important Make sure this members of this group have read, write, and execute permissions to the following directory:

TomcatInstallDir`/webapps/`*qaduiConfig*`/WEB-INF/pro/com/ mfgpro`

Process Editor Group

A separate user group can access the Process Editor. This group must be named `pronav`; you cannot change the name.

Create QAD UI Administrative Users and Groups

The steps are different depending on the version of the system. In addition, if you are completing a non-US English QAD UI installation, modify the language and country code for each user so that QAD UI administrative screens display in the correct language.

QAD UI Admin User Setup for MFG/PRO eB2

1. Launch an MFG/PRO character session.
2. Go to User Maintenance (36.3.18) and add the MFG/PRO security group, by default `qadadmin`, to all QAD UI admin users, including `mfg`.

3 For all UI design group users, create the user records and assign each to a common administrative group; for example, `qaddsgn`.

4 Repeat step 3 for all Process Editor users, assigning each to the `pronav` group.

5 Exit the MFG/PRO session.

QAD UI Admin User Setup for eB2.1 and QAD 2008 Standard

1 Launch a character session.

2 Go to User Group Maintenance (36.3.4) to define a security group. By default this is `qadadmin`. You then assign users, including `mfg`, to the group. You can also associate existing groups with users in User Maintenance (36.3.1).

3 For all UI design group users, create a new group—for example, `qaddsgn`—and assign users to the group.

4 Repeat step 3 for all Process Editor users, creating the `pronav` group and assigning users.

5 Exit the session.

Chapter 5

Installing QAD UI Single-Tier

This chapter provides the steps to install the QAD .NET UI on a single-tier server.

Overview

This chapter details the steps required to install the QAD UIs on a single-tier server. If you ordered the QAD .NET UI, the server installation occurs as a part of the basic .NET UI installation.

MFG/UTIL and QAD UI Installations

Any installation process updates MFG/UTIL. This is especially true when you install the QAD UI over MFG/PRO, or an MFG/PRO service pack over the QAD UI, since MFG/UTIL for both of these is installed into the QAD Enterprise Applications database server directory. Some MFG/UTIL functions, including conversion routines, are specific to certain versions of MFG/UTIL. As a result, some functions may not work correctly if you overwrite the existing MFG/UTIL version.

To avoid any mismatches, back up your existing MFG/UTIL version. Once you have installed the QAD UI over QAD 2008 Standard (or MFG/PRO) or an MFG/PRO service pack over the QAD UI, back up that version of MFG/UTIL as well. If you need the prior version of MFG/UTIL—for example if you are running a conversion after installing the QAD UI—restore the prior version.

To create a backup, include the following directories:

- *MFGPROInstallDir*
- *MFGPROInstallDir*/xmfgusrc

Install QAD UI Components

Mount the CD-ROM (UNIX Only)

1. Log on as a user that has write permission to the Tomcat and QAD 2008 Standard installation directories. If you do not have a Tomcat directory, it will be created during the install.
2. Mount the QAD UI CD-ROM. Example commands are listed in Table 5.1.

Table 5.1
UNIX CD Drive Mount Commands

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

Launch Installation Script

Use these steps to start the installation process. The following instructions work for all QAD UI installation CDs.

Important Run through the install script for one UI configuration at a time. Rerun the script to add UI configurations to the same host. Performing multiple passes during a single script execution can cause problems in the MFG/UTIL configuration.

1 Launch the installation script. For UNIX:

```
./install.ksh
```

For Windows, run `install.exe` from Run on the Start menu. (Ensure you have administrative privileges.)

2 You are prompted to enter a log file directory location. Each QAD UI configuration (*`qaduiConfig`*) should have its own log file directory. Later installation and configuration processes look at this location. If the directory you specify does not exist, you are prompted to create it. The default directory in UNIX is `/home/mfg/mfgsvr`; in Windows, it is `C:\mfgsvr`.

```
Please enter a directory to write log files
Default is '/home/mfg/mfgsvr'
        ->/dr02/mfgpro/eb2.1sp3/dt01/log
```

3 If no QAD UI install `.ini` file exists in the log file directory, you are prompted to create one:

```
No .ini was found in directory '/dr02/mfgpro/eb2.1sp3/ dt01/log'. Do you want to
start a new setup file (.ini)?
Default is 'y'
        -> y
```

If a QAD UI install `.ini` file exists, it displays as a selection option. In the following example, an `.ini` file exists for `Test_dt2`. Choose option 2 to create a new configuration.

```
*** Configuration file selection ***
Please choose one of the following:

        1: Test_dt2.ini -> Test
        2: Create new configuration
        3: Cancel
<1-3>?
```

4 Specify the name for this UI configuration; this name is applied to the log and `.ini` files.

```
What is the application name for this UI configuration?
Default is 'qadui'
        ->dt01
```

Configuration names must be at least three characters, with no spaces.

5 Press Enter to review and accept the license agreement. Press q to quit the agreement and reach the acceptance prompt.

```
Do you accept the terms of the preceding License Agreement?
If you choose no, the install will stop.
Default is 'n'
        -> y
```

6 Specify whether this is a single- or multi-tier QAD UI configuration. This chapter addresses the single-tier option. To implement a multi-tier scenario, go to Chapter 6, “Installing QAD UI Multi-Tier,” on page 59.

```
*** Configuration: qadui ***
*** Please select type of install. ***
```

```
        1: Single-Tier
        2: Multi-Tier (n-tier)
<1-2>? 1
```

7 The main menu displays.

```
*** Configuration:  qadui ***
*** Main Menu ***
Please choose one of the following:
        1: Install Tomcat Files Menu
        2: Install MFG/PRO - Progress QAD UI Files Menu
        3: Select a Different Configuration
        4: Exit
<1-4>?
```

Install Tomcat and UI Web Application (required)

Complete this section to install the Tomcat servlet engine included on the QAD UI media and deploy the QAD UI Web application file. If you use a previously installed QAD-supplied Tomcat version, the install script validates that the existing installed version is compatible with the QAD UI release being installed.

1 Shut down any existing Tomcat instances running on the server. In a separate terminal window, navigate to *TomcatInstallDir*/bin and run the shutdown script.

```
./shutdown.sh
```

2 On the main menu, choose 1 to access the Tomcat installation menu.

3 If the JAVA_HOME variable is not set, you are prompted to set it before continuing. If it is set, you are prompted to confirm. For more information, see “Set JAVA_HOME” on page 30.

4 On the Tomcat menu, choose 1 to install Tomcat.

```
*** Configuration: dt01 ***
*** Tomcat Menu ***
Please choose one of the following:
        1: Install Tomcat Files
        2: Uninstall Tomcat
        3: Return to Main Menu
<1-3>? 1
```

5 Enter the directory where you want to install Tomcat. Always install Tomcat in its own directory.

To use an existing Tomcat installation, specify the directory where Tomcat is installed. If the existing Tomcat is valid for the current version of QAD UI, you install the QAD UI Web application only. If the existing Tomcat installation is not compatible, you are prompted to overwrite the existing Tomcat.

```
Please enter the Tomcat installation directory.
No default value
        ->/dr01/tomcat

The directory '/dr01/tomcat' does not exist.  Would you like to create this
directory?
Default is 'y'
        -> y
```

6 Optionally enter a new Tomcat port number.

```
Please enter the Tomcat port number.
Default is '8080'
        ->
```

See “Update Tomcat Port Numbers” on page 85.

Important Manual updates are required when Tomcat does not use the default 8080 port or if another application is already using port 8080. These steps are detailed in Chapter 7.

7 You are informed about the compatibility of the Tomcat version and asked to confirm the install. Enter Y and press Enter to start the install.

The Tomcat installation files are extracted. The Tomcat `setenv.sh` and `setenv.bat` scripts are also updated with QAD UI-specific environment information. You are prompted to press Enter as each step completes.

8 When the Tomcat Menu displays again, choose 3 to return to the main menu.

Install QAD UI Code

Complete this section to install QAD UI-specific code for QAD 2008 Standard and the required updates to MFG/UTIL.

1 From the main menu, choose 2 to access the MFG/PRO – Progress QAD UI Files Menu.

2 You may be prompted to enter or confirm the JAVA_HOME directory.

3 The MFG/PRO — Progress QAD UI Files Menu displays. Choose 1 to begin installing the QAD UI Progress files.

```
*** Configuration:  dt01 ***
*** MFG/PRO - Progress QAD UI Files Menu ***
        1: Install MFG/PRO - Progress QAD UI Files
        2: Update Additional MFG/UTIL Installations
        3: Launch MFG/UTIL
        4: Uninstall MFG/PRO - Progress QAD UI Files
        5: Return to Main Menu
<1-5>? 1
```

4 Enter the character client installation directory. In a typical system, the character-client files are located in the *`MFGPROInstallDir`* directory.

```
Please specify the MFG/PRO Client directory.
No default value
        ->/dr02/mfgpro/eb2.1sp3
```

See “Define QAD UI Security Groups” on page 32.

5 Enter the user group that will be given UI administration privileges. By default, this is `qadadmin`.

```
Please enter the existing MFG/PRO security group that will have the admin privileges
in qadui.
Default is 'qadadmin'
        ->
```

6 Enter and confirm the QAD UI code directory. The default is the QAD UI configuration name being installed. For consistency when installing multiple QAD UI configurations, accept the default value.

```
Please specify the MFG/PRO QAD UI code directory.
The QAD UI Progress code files will be extracted here.
Default is '/dr02/mfgpro/eb2.1sp3/dt01'
          ->
```

If the directory does not exist, you are prompted to confirm the creation. If the directory exists, you are prompted to confirm the overwrite.

The file copy process is logged to the screen. Files are copied, permissions are set correctly, and MFG/UTIL is updated with QAD UI workflows. You are prompted to press Enter as each step completes. When the process completes, review the log for errors.

7 When the process completes, you are returned to the Progress QAD UI Files menu. The next step depends on your operating system.

- In UNIX environments, exit from the install menus, then continue with “Configure QAD UI Components” on page 41.
- For Windows, choose option 3 to generate the telnet scripts as detailed in the next section.

Important Do not update or create additional UI configurations without restarting the script. Performing activities on more than one configuration during a single script execution can cause problems in MFG/UTIL.

Set Up Windows Telnet Environment

Use the following steps to copy files required for the Windows telnet server and to generate the required `k_start.bat` script. Make sure you are in the MFG/PRO – Progress QAD UI Files Menu.

1 Choose option 3 to set up your Windows telnet environment.

```
*** MFG/PRO - Progress QAD UI Files Menu ***
Please choose one of the following:
        1: Install MFG/PRO - Progress QAD UI Files
        2: Update Additional MFG/UTIL Installations
        3: Telnet Setup (Windows Only)
        4: Uninstall MFG/PRO - Progress QAD UI Files
        5: Return to Main Menu
<1-5>? 3
```

See “Install the Windows Telnet Server” on page 30.

2 Validate that the Georgia SoftWorks Windows Telnet Server (GSWTS) has been installed.

```
The latest release of Georgia SoftWorks Windows NT/2000 Telnet Server(GSWTS) should
be installed before continuing.

Is Georgia SoftWorks Telnet Server installed? (answering 'n' will return to the
previoius menu).
Default is 'y'
        -> y
```

3 Enter the GSWTS installation directory and confirm whether to overwrite existing files, if any. Back up any files before allowing them to be overwritten.

```
Please enter the GSWTS installation directory, including the drive letter.
Default is 'c:\gs_uts'
        ->


File 'C:\gs_uts\scripts\k_start.bat' exists.  Do you want to overwrite this file?
Default is 'n'
        -> y
```

4 You may be prompted to confirm the character client directory. The default value is the one you specified in step 4 on page 39.

```
Please specify the MFG/PRO Character Client directory.
Default is 'D:\mfgpro\eb2'
        ->
```

5 The system generates the `k_start.bat` file and places executable files in several directories.

```
Setup of Telnet files finished.

Press Enter to Continue
```

6 Press Enter to continue, return to the main menu, and exit the installation script.

You are now ready to use MFG/UTIL to configure the deployed QAD UI files.

Configure QAD UI Components

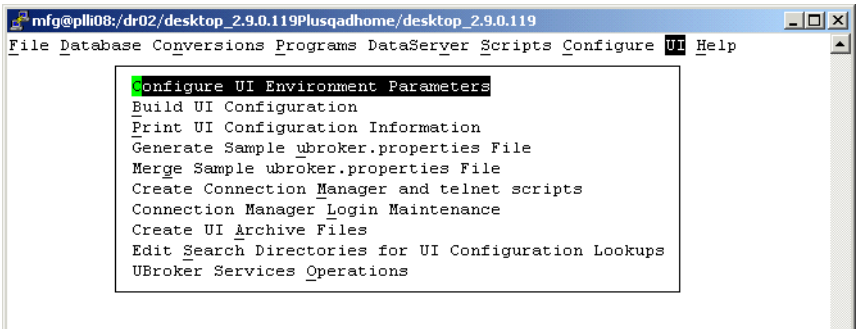
This section discusses the configuration activities you perform after deploying the QAD UI components. These steps are:

- Add Parameter Files (Optional)
- Verify and Update System Information
- Build the QAD UI System
- Generate WebSpeed Configuration Information
- Generate Telnet Log-in and Connection Scripts
- Print Configuration Information

During the installation activities, MFG/UTIL on the database server was updated with the functions needed to configure a QAD UI system. When you launch MFG/UTIL, choose UI to view the UI-specific menu options. Only the options used for a single-tier installation are discussed in this section.

During QAD UI configuration, you must specify the database set to use. If you have created `.pf` files manually without using MFG/UTIL functions, you can import these files into MFG/UTIL using the Import .pf Files to Database Set function on the MFG/UTIL Configure menu.

Fig. 5.1
MFG/UTIL UI Menu Options



Add Parameter Files (Optional)

By default, MFG/UTIL displays all available `.pf` files found in the QAD 2008 Standard system and client installation directories in the QAD UI configuration screen. You can specify additional directories for MFG/UTIL to search for other `.pf` files by using the UI|Edit Search Directories for UI Configuration Lookups option on the MFG/UTIL menu.

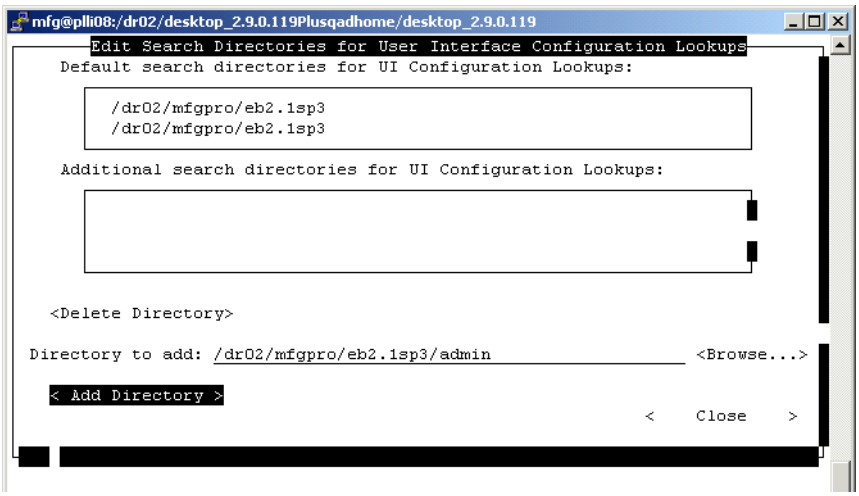
If your WebSpeed broker is on a different machine than the QAD 2008 Standard databases, the parameter files must include client/server parameters to access the databases. These are the host (`-H`), server (`-S`), and network protocol (`-N`) parameters.

1 Choose 3 on the main menu to launch MFG/UTIL.

2 Select UI|Edit Search Directories for UI Configuration Lookups from the MFG/UTIL main menu.

3 To add a directory, enter its full path name in Directory to add and choose Add Directory. A `.ksh` file from a QAD UI installation must exist in the directory.

Fig. 5.2
Edit Search Directories for UI Configuration Lookups



Note If MFG/UTIL cannot find a directory specified here or if no `.pf` files are located in a specified directory, an error displays for each affected directory when you access the Configure QAD UI Environment Parameters screen.

Delete Directory. Deletes the selected directory from the list of directories to search. You cannot delete the default directories.

Directory to add. Adds a directory to the search list. Optionally, use the browse to locate the directory in the file system and select it. After entering the directory, choose Add Directory to add it.

Verify and Update System Information

Complete this section to set QAD UI build parameters.

1 Start the database servers:

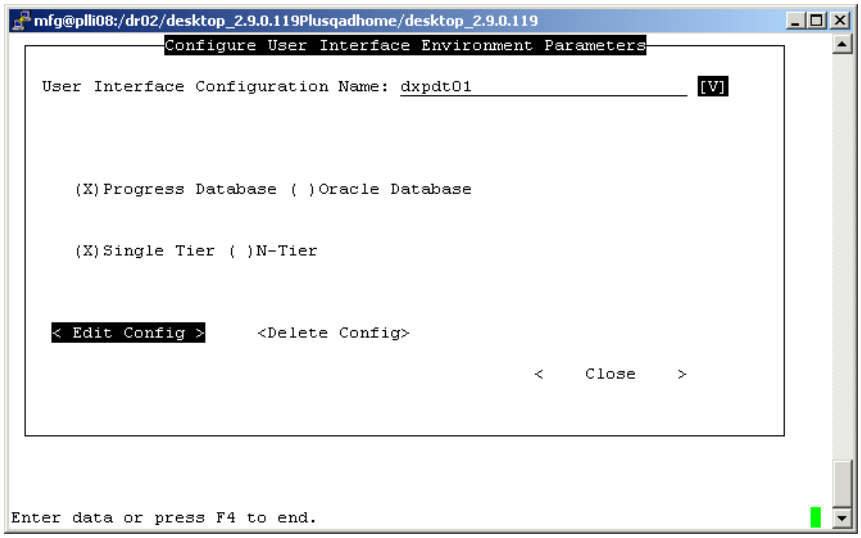
```
./start.Production
```

2 In MFG/UTIL, choose UI|Configure UI Environment Parameters.

3 If MFG/UTIL detects a new UI configuration, you are asked whether this is a Progress database configuration. Choose No if the new configuration is for an Oracle database; otherwise, choose Yes.

4 The Configure User Interface Environment Parameters screen displays. Select the name of the system you are configuring and choose Edit Config.

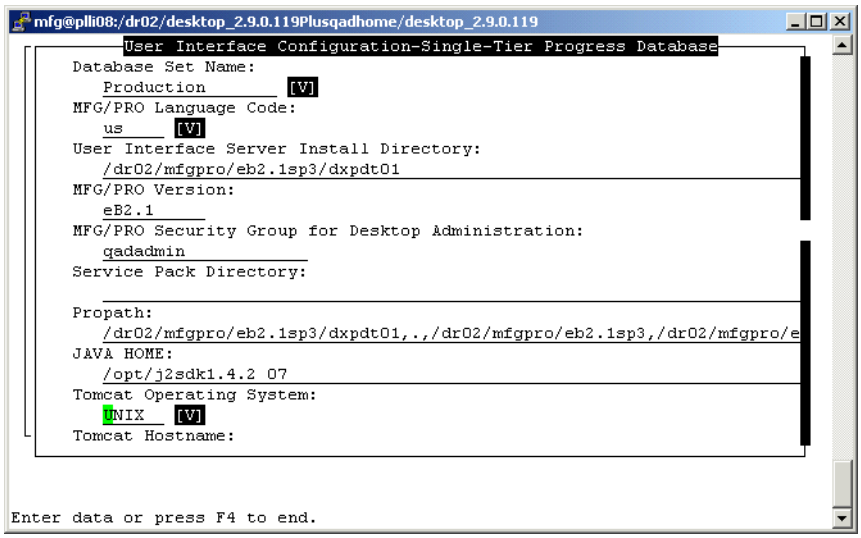
Fig. 5.3
Configure User Interface Environment Parameters



5 The system gathers configuration information and the configuration screen displays.

The screen differs slightly depending on the server operating system and database type. Some fields display only for Oracle database environments; the use of these fields is detailed in the field descriptions.

Fig. 5.4
First Single-Tier Configuration Screen



QAD UI System Information

The first set of fields prompts for information related to your QAD UI configuration.

Database Set Name. Select the database set where QAD UI will be used.

MFG/PRO Language Code. Select the language you want to use for screen displays.

User Interface Server Install Directory. Verify the location of the QAD UI installation directory.

MFG/PRO Version. Verify the version.

MFG/PRO Security Group for Desktop Administration. Specify the security group from "Define QAD UI Security Groups" on page 32. By default, this is `qadadmin`.

Absolute Path to Oracle Install. (Not shown) This is the path to the Oracle installation directory, also known as ORACLE_HOME.

Oracle System Identifier Name. (Not shown) Enter the Oracle System Identifier (ORACLE_SID). This is the environment variable for the Oracle database and is typically the same as the Oracle database name.

ORASOPATHNAME. (UNIX only, not shown) Enter the path to *ORACLE_HOME*`/lib/libclntsh.`*xx* client library, where *xx* refers to the Oracle version. Your specific Oracle version and system configuration define whether this parameter must be set and the specific setting. Refer to your Oracle documentation for details.

Service Pack Directory. If your MFG/PRO install includes a separate service pack installation, enter the service pack directory here. The appropriate service pack directories—*BaseServicePackDir* and the *BaseServicePackDir*`/xrc`—are automatically added to the QAD UI compile PROPATH.

PROPATH. Use this field to enter any additional values for the PROPATH. The PROPATH for the build process requires the following directories, in this order:

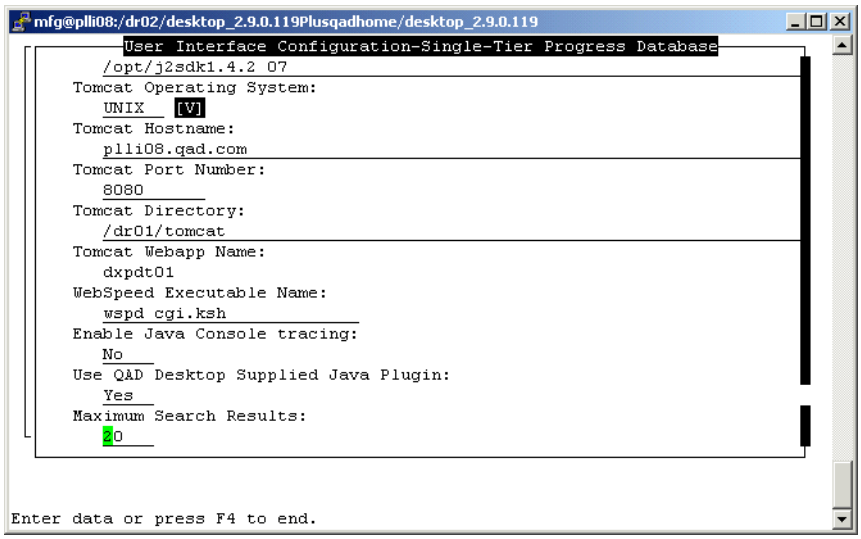
- QAD UI install directory
- Current directory (`.`)
- *MFGPROInstallDir*
- *MFGPROInstallDir*`/us/xrc`

All directories specified in the PROPATH must exist.

Note Add custom and patched code directories to the end of the value in PROPATH.

Annotation 1 The following screen includes a reference to Desktop

Fig. 5.5
Second Single-Tier Configuration Screen



Tomcat and WebSpeed Configuration Fields

This set of fields contains Tomcat information.

JAVA HOME. Verify the location of your Java installation.

Tomcat Operating System. Verify the operating system for the server where Tomcat resides.

Tomcat Hostname. Enter the fully qualified host name of the Tomcat server.

Tomcat Port Number. This value is automatically populated with 8080. Change this value as needed.

See “Update Tomcat Port Numbers” on page 85.

Note If you do not accept 8080, you must manually update the Tomcat server configuration file with the new port.

Tomcat Directory. For single-tier deployments, enter the fully qualified absolute path to the Tomcat installation directory.

Tomcat Webapp Name. The name of your QAD UI configuration.

WebSpeed Executable Name. Enter `cgiip.exe` for Windows, `wspd_cgi.ksh` for UNIX.

See “Install WebSpeed Messenger” on page 135.

Note The WebSpeed Messenger executable must be located in the Tomcat `cgi` directory:

TomcatInstallDir`/webapps/`*qaduiConfig*`/WEB-INF/cgi/`

Java and Search Defaults

This set of fields contains Java JVM and QAD UI search information.

See “Enable the Java Console” on page 143.

Enable Java Console Tracing. Enter Yes to send trace messages to the Java Virtual Machine. To view the messages, you must also enable the console. Enabling trace may adversely affect performance. If you enable trace for debugging purposes, you must rebuild QAD UI to disable this option later.

Use QAD Desktop Supplied Java Plugin. The plug-in is now called the Java Virtual Machine (JVM). Indicate how you want to manage the JVM version for client sessions.

- Set to No to have Desktop UI clients use a JVM already installed on the client machine. If no JVM exists, or the minimum version is not found, the QAD-supplied version is downloaded.
- Set to Yes to have all clients use the JVM supplied on the QAD UI installation media. If other versions of Java exist on the client, they are not replaced and can continue to be used by applications other than the Desktop UI.

To change this value at a later time, you must rebuild your QAD UI system.

Annotation 2 This field label is still Use QADDesktop Supplied Java Plugin. Is the functional description of the field still valid?

Maximum Search Results. By default, the search function in the UI returns 20 records at a time. You can change this value by specifying a different number here.

6 Choose OK to save the configuration. This may take a moment.

7 Choose Close to exit to the main menu.

Build the QAD UI System

This portion of the UI creation lets you build a new configuration based on the values entered in the UI Configuration screen. It also lets you:

- Limit optional build features such as the search database.
- Build for a specific language.
- Rebuild a UI selectively based on what portions of the UI you have modified.
- Create the build script without running the build, which lets you run the build for one or more UIs at a later time when system resources are available.

Use the following steps to verify the system information and build the QAD UI system:

1 From the MFG/UTIL main menu, choose UI|Build UI Configuration.

2 In UI Configuration Name, choose the configuration to build.

Fig. 5.6
Build User Interface

The fields are populated with values for the configuration you chose.

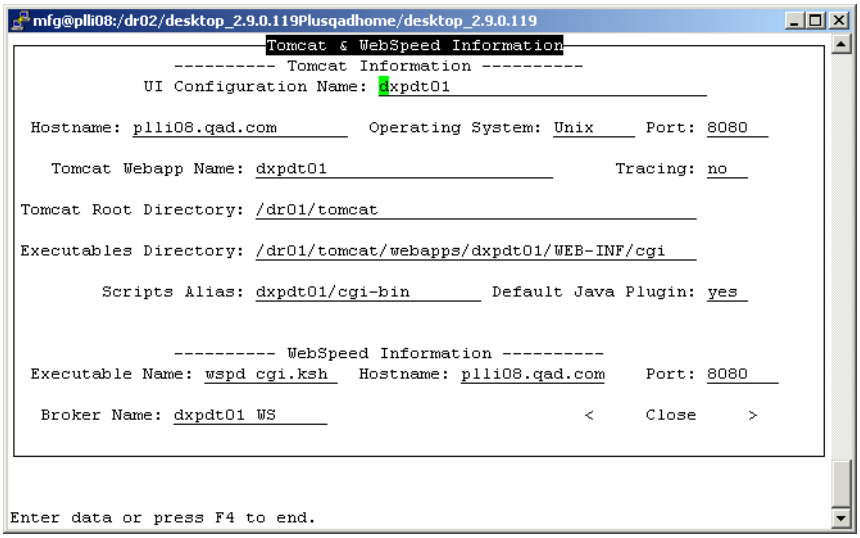
A UI is Validated for Build for single database systems. Multi-database systems that have more than one `qaddb` included in the `.pf` files are not validated; they will build, but the compile will return errors.

If Validated for Build is not checked, you may proceed; however, after the build you must edit your `.pf` files to ensure that only the first instance of the `qaddb` database appears. This allows the compile to run without errors.

When you are building a UI for an Oracle environment, the confirmation screen displays additional Oracle-specific settings. Verify the Oracle Home, Oracle SID, and for UNIX environments, the Oracle Shared Library Path values.

3 Once you verify the values on this screen, choose Tomcat/WebSpeed Info to review the build information for the Tomcat server and WebSpeed components.

Fig. 5.7
Tomcat Server and WebSpeed Information



4 After verifying the Tomcat and WebSpeed information, choose Close to return to the Build UI Configuration screen.

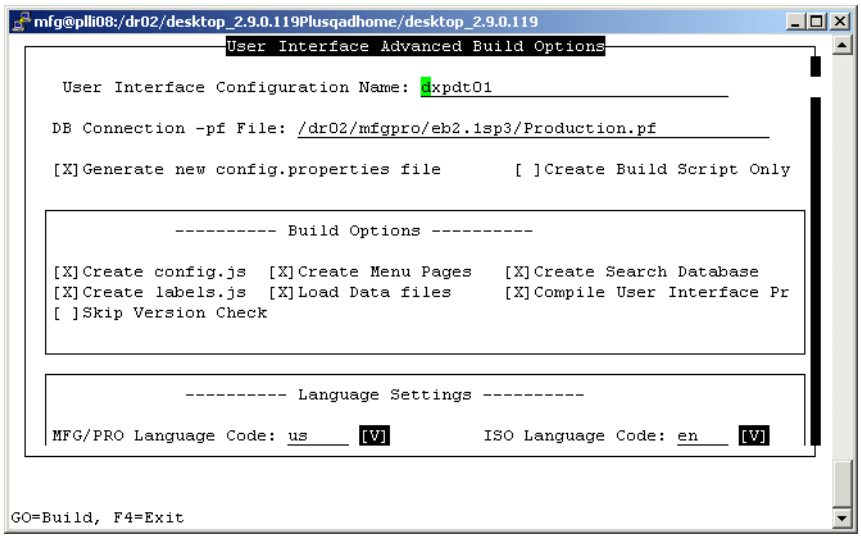
5 To set advanced configuration options—typically, used when building Oracle systems, rebuilding a UI, upgrading to a new release, creating build scripts without executing the build, or building multi-language environments—choose Advanced.

Note If you are building a multiple-language UI configuration, make sure you review the information in Chapter 10, "Implementing Multiple Languages," on page 105.

Advanced Build Options

As with other MFG/UTIL screens, this one may differ slightly depending on the QAD UI configuration and database environment you are working with.

Fig. 5.8
First Advanced Build Options Screen



User Interface Configuration Name. This field displays the QAD UI configuration you are currently updating; it cannot be modified.

DB Connection -pf File. This is the database connection parameter file that the build process uses to connect to the database to load system data and compile the QAD UI code. This file is set based on the database set you identified in the UI Configuration screen. If the wrong `.pf` is displayed here or you need to change it, you must exit this screen and specify the correct database set in the UI Configuration screen.

Generate new config.properties file. This option should be selected when a new QAD UI system is to be built. However, once a QAD UI system has been built and you are rerunning a specific build function—such as generating new menus because you changed menus in the system— deselect this option. This option creates a new `config.properties` file and overwrites the existing one. If the existing file contains any customizations, they will be lost.

Create Build Script Only. Select this to generate build scripts without running the build. These scripts can later be used to build the QAD UI system. This option helps ease system administration activities in environments with multiple QAD UI systems. By creating build

scripts for all the QAD UI systems without running the build, the system administrator can create a batch or cron job to run the build for one or multiple QAD UI systems at a later time when system resources are available, or during scheduled down times.

The generated build script is saved to the QAD 2008 Standard installation directory. The file name is the QAD UI system name with the standard system extension, such as `qaduiConfig.bat` for Windows, or `qaduiConfig.ksh` for UNIX.

Annotation 3 The following options refer to JS, HTML, and Desktop. Are they still valid for .NET UI?

Build Options

Select the options you want to use to control this build. These options are affected by values defined in the `config.properties` file.

See “Changing Menus” on page 123.

Typically, you enable all build options the first time you build the system. Later, you can change the `config.properties` file and rebuild with or without all options selected. In some cases, you may not need all options. For example, if you change your menu system, you can build only the search database and menu pages to incorporate your changes.

Create config.js file. Enable this option to create the JavaScript code that incorporates your environment values. This script is used to build the menu system for QAD Desktop.

See “Language Directories for Menus” on page 135.

Create Menu Pages. Enable this option to create the HTML menu pages for the UI. This option updates the menus in the two-letter language code subdirectory in the Tomcat Web applications directory:

TomcatInstallDir`/webapps/`*qaduiConfig*`/menus`

Create Search Database. Enable this option to build the database accessed by the Search option in the Desktop UI. You can search by menu label, menu number, short name, and execution file based on data specified in Menu System Maintenance (36.4.4) and Menu Substitution Maintenance (36.20.6).

Create labels.js file. Enable this option to create the JavaScript code for displaying labels that are used on the Desktop UI main pages. Labels and their associated terms are stored in the label master table in the database and maintained with Label Master Maintenance (36.4.17.1). Labels are extracted and placed in language-specific files based on the language specified for the language code prompt.

Load Data files. Enable this option to load terms and labels used in the UI from an external file into your QAD 2008 Standard database. This data needs to be loaded only once to update your database. If you are creating multiple UI configurations for the same databases, you can run each UI build with this option enabled with no negative effect.

Compile User Interface Programs. Enable this option to compile the Progress programs supplied with the Desktop UI.

Skip Version Check. The build code checks the QAD Enterprise Applications version against the UI version. If an unsupported version is detected, the build stops. This is true even for currently supported versions that were previously not supported.

If you are installing this UI configuration against a version that was not previously supported, check this box. You will see a warning about the version mismatch, but will not be stopped from completing the build.

Language Settings

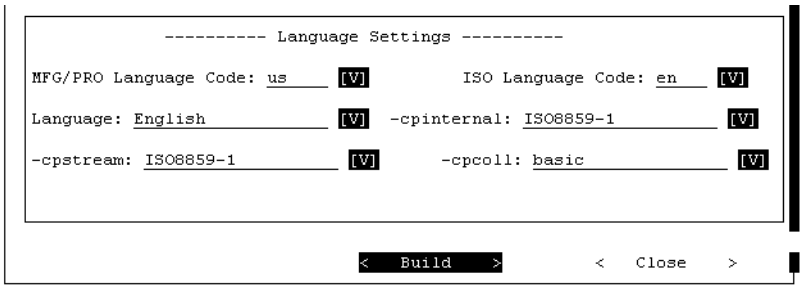
Language settings are important if you are installing a UI configuration into a system that supports multiple languages. For a single- language implementation, you typically do not need to change any of these settings. QAD UI is preconfigured with appropriate settings for each supported language and these display by default, based on the QAD (MFG/PRO) language code you selected in the previous screen.

Each language has predefined values for ISO language code, Progress startup parameters, and the Oracle NLS setting.

For additional details, see “Use MFG/UTIL to Install Multiple Languages” on page 110.

However, if you are installing multiple languages, you must execute the advanced build options for each language to create language-specific menus, search database, and other required support data. In this case, you may need to modify the language settings for additional languages from the default values to conform to your database settings.

Fig. 5.9
Advanced Build Options, Language Settings



MFG/PRO Language Code. Select the language code from the drop-down list that represents the language you want to use for screen display, such as US for US English. MFG/UTIL builds the list of available languages based on the language directories you have currently installed.

Note This language code is not necessarily the same as the ISO standard language code used by Java.

ISO Language Code. This field displays the ISO language code associated with the QAD (MFG/PRO) language code in this screen. You should not change this value. The .NET UI is preconfigured with the correct mapping between supported QAD (MFG/PRO) codes and ISO codes.

See “Language Directories for Menus” on page 135.

When menus are generated, the HTML files are placed in a directory that is named based on this ISO language code. However, even when a non-English language is specified, some files are also created in the `en` directory.

Language. This field displays the full language name associated with the language code you selected. Do not change this value.

For details on these parameters, see the Progress *Startup Command and Parameter Reference*.

–cpinternal. The database startup parameter that identifies the Progress code page for graphical clients. This should be the same as the operating system code page. If a value is not specified, iso8859-1 is used. When you build a multiple-language UI configuration, change this value to the code page associated with the database.

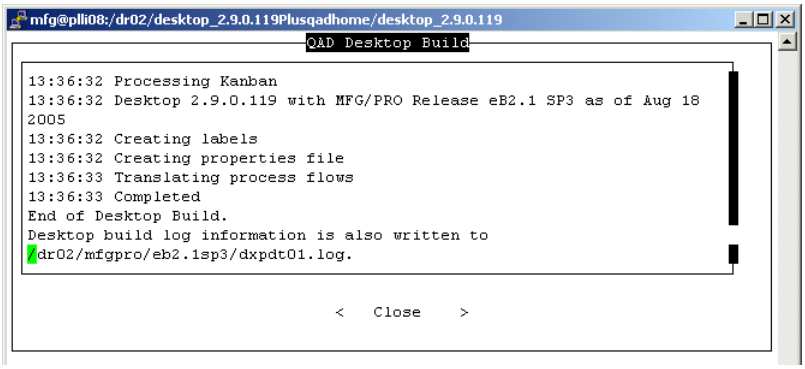
–cpstream. The database startup parameter that identifies the code page for stream I/O. Stream I/O includes character clients, data files, and code. When you build a multiple-language UI configuration, change this value to the code page associated with the database.

–cpcoll. The database startup parameter that identifies the collation table to use with the code page identified by the `-cpinternal` parameter. Progress uses the collation rules that you specify to compare characters and sort records. When you build a multiple-language UI configuration, change this value to the collation table associated with the database.

Oracle NLS Setting. Sets the NLS_LANG parameter required in the WebSpeed `ubroker.properties` file and telnet startup scripts for the Oracle dataserver where Native Language Support (NLS) is required for non-English and double-byte languages.

6 Choose Build and confirm you have reviewed the build information to build the system. The build log displays as the build occurs.

Fig. 5.10
Sample Build Log Display



7 When the UI build completes, review the log for errors. Correct any errors before you proceed. Choose Close.

Continue with the instructions in the next section to generate the QAD UI system-specific `ubroker.properties` file.

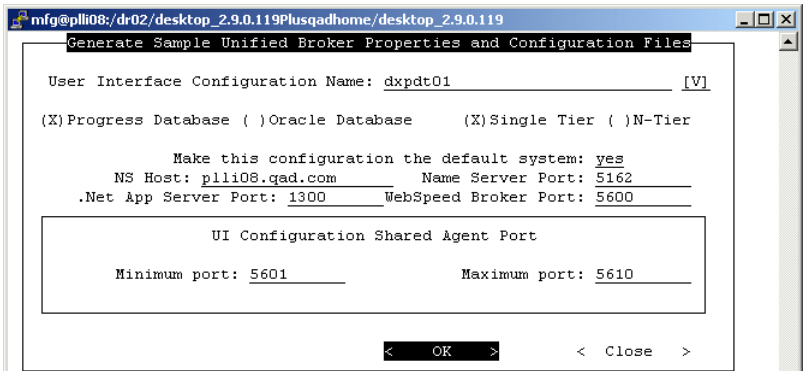
Generate WebSpeed Configuration Information

To update the WebSpeed `ubroker.properties` file with the QAD UI broker definitions, generate a parameter file in MFG/UTIL. The parameter file is merged into `ubroker.properties` in the next set of steps.

If you are implementing the QAD .NET UI, this screen displays additional fields in order to generate the Progress AppServer definitions.

1 In MFG/UTIL, choose UI|Generate Sample ubroker.properties File. The Generate Sample Unified Broker Properties and Configuration Files frame displays.

Fig. 5.11
Generate Sample Unified Broker Properties and Configuration Files



2 Choose the configuration you are generating the file for in User Interface Configuration Name. Update the other fields as follows:

Progress Database/Oracle Database. Display-only. If this value is incorrect, the install script must be run again.

Single-Tier/N-Tier. Display-only. If this value is incorrect, the install script must be run again.

See “Sample XML Configuration File” on page 121.

Make this configuration the default system. This field applies to the QAD .NET UI. Set this to Yes to generate both a `qaduiConfig.xml` and a `default.xml` file. The default configuration is the first configuration users log into. After initial login, they can access other configurations in the same environment by choosing Tools|Options and setting the System Preferences. When multiple systems are available, users can also choose to select from a list during login.

NS Host. The machine where the Progress Name Server is installed.

Name Server Port. By default this is 5162.

.NET App Server Port. The port for the .NET UI Progress AppServer. The default is 1300. Make sure the port you enter here is unused and available.

WebSpeed Broker Port. This port is used only by the .NET UI WebSpeed broker. The default is 5600. Make sure the port you enter here is unused and available.

UI Configuration Shared Agent Minimum Port. The minimum port for a range of shared agent ports available for UI user sessions. The typical range is 10. Make sure the entire range of ports you enter here is unused and available.

UI Configuration Shared Agent Maximum Port. The maximum port for a range of shared agent ports available for UI user sessions. Make sure the entire range of ports you enter here is unused and available.

3 Choose OK to generate the file for the UI configuration you selected.

The system displays a message that the file, *qaduiConfig*-ubroker.properties, was saved in your *MFGPROInstallDir*. The generated file contains the complete definitions required by Progress to launch the AppServer and WebSpeed broker.

Generate this file for each UI configuration you are setting up.

Merge ubroker.properties Files

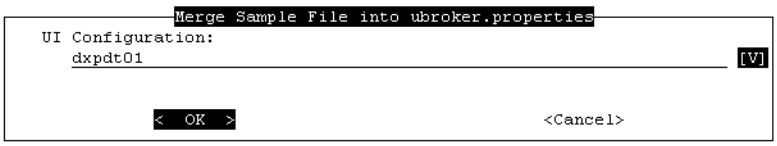
The automated merge process creates a backup of the standard Progress ubroker.properties file and then inserts the file you generated at the bottom of the original file. For example, *ProgressInstallDir*/ properties/ubroker.properties.002 is created, then the content of *MFGPROInstallDir/qaduiConfig*-ubroker.properties is inserted at the bottom of *ProgressInstallDir*/properties/ ubroker.properties.

If you are rerunning the merge process for a given UI configuration, the existing broker and AppServer definitions are deleted from the base ubroker.properties file before the new definitions are added.

It is worthwhile to review the content of *MFGPROInstallDir/ qaduiConfig*-ubroker.properties prior to merging. Make sure all the values look correct given the setup completed to this point.

1 In MFG/UTIL, choose UI|Merge Sample ubroker.properties File. The Merge Sample File into ubroker.properties frame displays.

Fig. 5.12
Merge Sample File into ubroker.properties



2 Choose the configuration you are merging in.

3 Choose OK. The log displays.

4 Choose Close to complete the process.

Generate Telnet Log-in and Connection Scripts

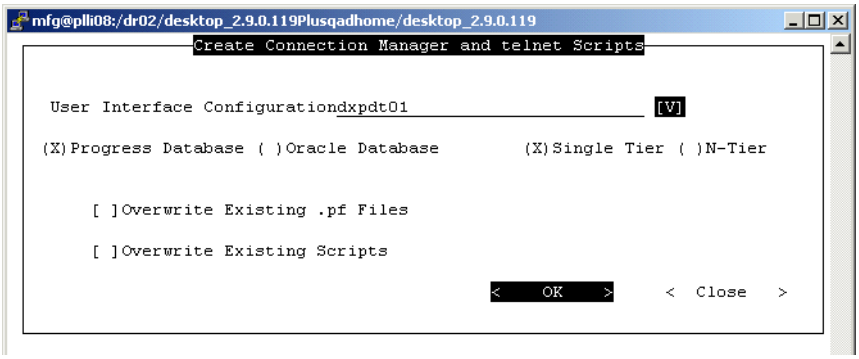
See “Update Configuration Settings” on page 98.

Use MFG/UTIL to generate the Connection Manager and telnet connection scripts. Later on, these values default into the Connection Manager.

1 In MFG/UTIL, choose UI|Create Connection Manager and telnet scripts from the main menu. The create scripts screen displays.

2 Select the UI configuration. The database type and tier type display. Use the check boxes to overwrite existing .pf files and scripts built during your base install. Choose OK to generate the scripts.

Fig. 5.13
MFG/UTIL
Create Connection Manager and telnet Scripts



3 Using the specified configuration information, MFG/UTIL generates two files in the UI installation directory. The file names depend on the operating system specified in the configuration settings screen:

- For UNIX servers, the `connmgr.`*`DBSetName`* and `telnet.`*`DBSetName`* files are generated.
- For Windows servers, the `connmgr`*`DBSetName`*`.bat` and `telnet`*`DBSetName`*`.bat` files are generated.

New *`DBSetName`*`.pf` and `.ini` files are created in *`MFGPROInstallDir`*.

`DBSetName` refers to the database set name specified in the build screen.

Depending on the release you are upgrading from, some files may be backed up. If any `.pf` or telnet connection script files are found in the *`MFGPROInstallDir`* or *`QADUIInstallDir`* and you choose not to overwrite them, they are moved to an `old_dtscripts` directory.

Important In some UNIX environments (HP-UX and AIX), you may need to include additional parameters such as SHLIB_PATH and LIBPATH in the generated files. For details, refer to your operating system documentation.

The .NET UI uses values defined in the Progress `protermcap`. The keys listed in Table 5.2 must have their default setting or actions in the Desktop UI may fail. If you have modified keys in the `protermcap` file, change the PROTERMCAP environment variable to point to an unchanged file.

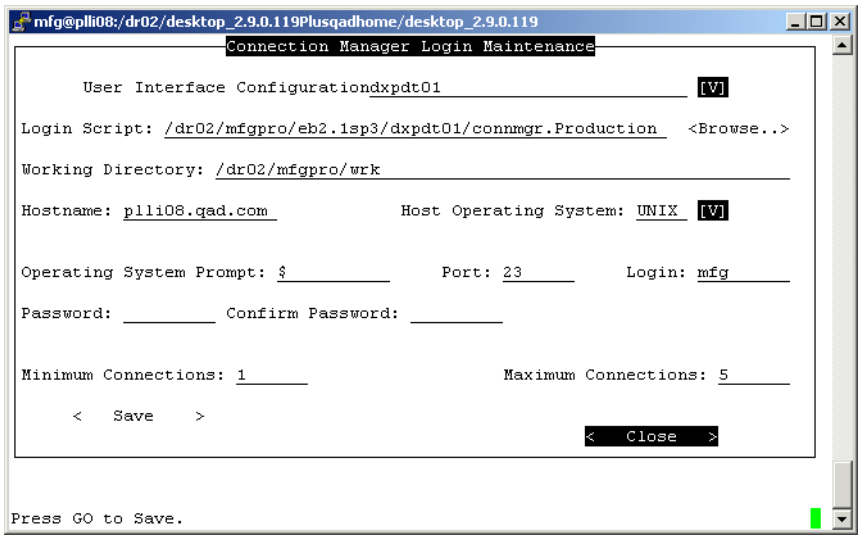
Table 5.2
Function Keys

UNIX	Windows
F16	Shift-F5
F17	Shift-F6
F18	Shift-F7
F19	Shift-F8
F20	Shift-F9

When the Connection Manager scripts have been created, use MFG/UTIL to create Connection Manager log-in scripts:

1 In MFG/UTIL, choose UI|Connection Manager Login Maintenance. The Login Maintenance screen displays.

Fig. 5.14
Connection Manager Login Maintenance



2 Choose the configuration you are setting up in User Interface Configuration. Update the other fields as follows:

Login Script. Locate or type in the `connmgr` script generated in the previous set of steps; for example, *`MFGPROInstallDir/ qaduiConfig`*`/connmgr.Production`.

Working Directory. Enter a directory where temporary files can be written during telnet sessions. This directory must exist and must have write privileges for user `mfg`.

Hostname. The database server host machine, including domain value; for example, hpux01.corp.com.

Host Operating System. This defaults in and should be correct.

Operating System Prompt. This, and the next few fields through the password, are used to construct the log-in sequence for the session telnet connections to the system. The prompt is typically a dollar sign ($), pound sign (#), or greater than sign (>) depending on your operating system and UNIX shell.

Port. The operating system telnet port. Typically this is 23. Check your operating system to verify.

Login. The user that will be used to login to the operating system and run the Character UI Progress sessions that underlie the .NET UI screens. This user is not required to be a user in the application. QAD recommends that this user not be used to run anything other than these sessions so that you can easily identify these Progress processes on the operating system.

Password/Confirm Password. Enter the user password and then confirm it. No characters display during data entry in these fields for your security.

Annotation 4 The following connections values refer to Desktop. Still valid for .NET UI?

Minimum Connections. Enter the minimum number of telnet connections that the system will keep open during a Desktop session.

Maximum Connections. Enter the maximum number of telnet connections that the system will open during a Desktop session.

3 Choose Save to update the login script.

4 When the save is complete, you are informed that file `connectionManagerConfig.xml` was updated and a backup of the original file was saved to `connectionManagerConfig.xml.back`. Choose OK to continue.

5 Choose Close to exit.

Print Configuration Information

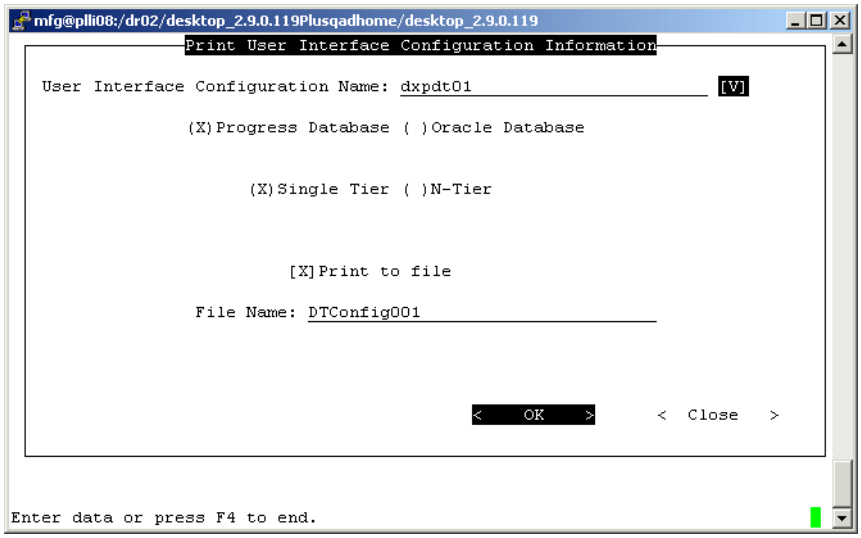
You can use MFG/UTIL to generate a text file containing the complete configuration information, including PROPATH information, directory locations, and port information for your UI configurations. A separate PROPATH is displayed for each of the following:

- QAD UI Build PROPATH
- Telnet Client PROPATH
- Connection Manager PROPATH
- WebSpeed PROPATH

1 In MFG/UTIL, choose UI|Print UI Configuration Information.

2 When the Print UI Configuration screen displays, select your UI configuration and choose whether to send the output to a file or printer.

Fig. 5.15
MFG/UTIL Print UI Configuration Information



3 Choose OK. File output is saved by default to *MFGPROInstallDir*.

You are now ready to perform the UI setup and configuration steps detailed in Chapter 7, “Configuring UIs After Deployment,” on page 83.

Chapter 6

Installing QAD UI Multi-Tier

This chapter provides the steps to install the Desktop UI and .NET UI in a multi-tier environment.

Overview

This chapter details the steps required to install the QAD UIs in a multi-tier environment.

MFG/UTIL and QAD UI Installations

Any installation process updates MFG/UTIL. This is especially true when you install the QAD UIs over QAD 2008 Standard, or a service pack over the QAD UIs, since MFG/UTIL for both of these is installed into the database server directory. Some MFG/UTIL functions, including conversion routines, are specific to certain versions of MFG/UTIL. As a result, some functions may not work correctly if you overwrite the existing MFG/UTIL version.

To avoid any mismatches, back up your existing MFG/UTIL version. Once you have installed the QAD UI over QAD 2008 Standard or a service pack over the QAD UI, back up that version of MFG/UTIL as well. If you need the prior version of MFG/UTIL—for example if you are running a conversion after installing the QAD UI—restore the prior version.

To create a backup, include the following directories:

- *MFGPROInstallDir*
- *MFGPROInstallDir*/xmfgusrc

Install QAD UI Components

Mount the CD-ROM (UNIX Only)

1. Log on as a user that has write permission to the Tomcat and QAD 2008 Standard installation directories. If you do not have a Tomcat directory, it will be created during the install.
2. Mount the QAD UI CD-ROM. Example commands are listed in Table 6.1.

Table 6.1
UNIX CD Drive Mount Commands

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

Launch Installation Script

Use these steps to start the installation process.

Important Run through the install script for one UI configuration at a time. Rerun the script to add UI configurations to the same host. Performing multiple passes during a single script execution can cause problems in the MFG/UTIL configuration.

1 Launch the installation script. For UNIX:

```
./install.ksh
```

For Windows, run `install.exe` from Run on the Start menu. (Ensure you have administrative privileges first.)

2 You are prompted to enter a log file directory location. Each QAD UI configuration (*`qaduiConfig`*) should have its own log file directory. Later installation and configuration processes look at this location. If the directory you specify does not exist, you are prompted to create it. The default directory in UNIX is `/home/mfg/mfgsvr`; in Windows, it is `C:\mfgsvr`.

```
Please enter a directory to write log files
Default is '/home/mfg/mfgsvr'
        ->/dr02/mfgpro/eb2.1sp3/dt02/log
```

3 If no QAD UI install `.ini` file exists in the log file directory, you are prompted to create one:

```
No .ini was found in directory '/dr02/mfgpro/eb2.1sp3/ dt02/log'. Do you want to
start a new setup file (.ini)?
Default is 'y'
        -> y
```

If a QAD UI install `.ini` file exists, it displays as a selection option. In the following example, an `.ini` file exists for `Test_dt2`. Choose option 2 to create a new configuration.

```
*** Configuration file selection ***
Please choose one of the following:

        1: Test_dt2.ini -> Test
        2: Create new configuration
        3: Cancel
<1-3>?
```

4 Specify the name for this UI configuration; this name is applied to the log and `.ini` files.

```
What is the application name for this User Interface (UI) configuration?
Default is 'qadui'
        ->dt02
```

Configuration names must be at least three characters, with no spaces.

5 Press Enter to review and accept the license agreement.

```
Do you accept the terms of the preceding License Agreement?
If you choose no, the install will stop.
Default is 'n'
        -> Y
```

6 Specify whether this is a single- or multi-tier QAD UI configuration. This chapter addresses the multi-tier option. To implement a single-tier scenario, go to Chapter 5, “Installing QAD UI Single-Tier,” on page 35.

```
*** Configuration: QAD UI_Prod ***
*** Please select type of install. ***
        1: Single-Tier
        2: Multi-Tier (n-tier)
<1-2>? 1
```

7 The main menu displays.

```
*** Configuration:  dt02 ***
*** Main Menu ***
```

```
Please choose one of the following:
        1: Install MFG/PRO - Progress QAD UI Files Menu
        2: Post MFG/UTIL QAD UI Build Jar Install
        3: Select a Different Configuration
        4: Exit
<1-4>?
```

Install QAD UI Code

Complete this section to install QAD UI-specific code for QAD 2008 Standard Edition (the MFG/PRO - Progress QAD UI Files) and updates to MFG/UTIL.

1 From the main menu, choose option 1 to access the MFG/PRO – Progress QAD UI files menu.

2 You may be prompted to enter or confirm the JAVA_HOME directory. For more information, see “Set JAVA_HOME” on page 30.

3 The MFG/PRO — Progress QAD UI Files Menu displays. Choose 1 to begin installing the QAD UI Progress files.

```
*** Configuration:  dt02 ***
*** MFG/PRO - Progress QAD UI Files Menu ***
        1: Install MFG/PRO - Progress QAD UI Files
        2: Update Additional MFG/UTIL Installations
        3: Launch MFG/UTIL
        4: Uninstall MFG/PRO - Progress QAD UI Files
        5: Return to Main Menu
<1-5>? 1
```

4 Enter the character client installation directory. In a typical system, the character client files are located in the *MFGPROInstallDir* directory.

```
Please specify the MFG/PRO Client directory.
No default value
        ->/dr02/mfgpro/eb2.1sp3
```

See “Define QAD UI Security Groups” on page 32.

5 Enter the user group that will be given UI administration privileges. By default this is `qadadmin`.

```
Please enter the existing MFG/PRO security group that will have the admin privileges
in QAD UI.
Default is 'qadadmin'
        ->
```

6 Enter and confirm the QAD UI code directory. The default is the QAD UI configuration name being installed. For consistency when installing multiple QAD UI configurations, accept the default value.

```
Please specify the MFG/PRO QAD UI code directory.
The QAD UI Progress code files will be extracted here.
Default is '/dr02/mfgpro/eb2.1sp3/dt02'
-> /dr02/mfgpro/eb2.1sp3/dt02
```

If the directory does not exist, you are asked to confirm the creation. If the directory exists, you are asked to confirm the overwrite.

The file copy process is logged to the screen. Files are copied, permissions are set correctly, and MFG/UTIL is updated with QAD UI workflows. You are prompted to press Enter as each step completes. When the process completes, review the log for errors.

7 You are prompted to enter a directory location for staging files that need to be configured by MFG/UTIL. By default, this is located in a `/staging` directory beneath the `/log` directory specified earlier.

```
Since this installation is a MULTI_TIER installation, this installation program will
need to copy files off the CD to a temporary staging area so that MFG/UTIL can modify
the files.

Please enter a temporary directory with adequate free space.
Default is '/dr02/mfgpro/eb2.1sp3/dt02/log/staging'
        ->
```

Files are copied to the staging directory.

8 When the process completes, you are returned to the Progress QAD UI Files menu. The next step depends on your operating system.

- In UNIX environments, exit from the install menus, then continue with “Configure QAD UI Components” on page 64.
- For Windows, choose option 3 to generate the telnet scripts as detailed in the next section.

Important Do not update or create additional UI configurations without restarting the script. Performing activities on more than one configuration during a single script execution can cause problems in MFG/UTIL.

Set Up Windows Telnet Environment

Use the following steps to copy files required for the Windows telnet server and to generate the required `k_start.bat` script. Make sure you are in the MFG/PRO – Progress QAD UI Files Menu.

1 Choose option 3 to set up your Windows telnet environment.

```
*** MFG/PRO - Progress QAD UI Files Menu ***
Please choose one of the following:
        1: Install MFG/PRO - Progress QAD UI Files
        2: Update Additional MFG/UTIL Installations
        3: Telnet Setup (Windows Only)
        4: Uninstall MFG/PRO - Progress QAD UI Files
        5: Return to Main Menu
<1-5>? 3
```

See “Install the Windows Telnet Server” on page 30.

2 Validate that the Georgia SoftWorks Windows Telnet Server (GSWTS) has been installed.

```
The latest release of Georgia SoftWorks Windows NT/2000 Telnet Server(GSWTS) should
be installed before continuing.

Is Georgia SoftWorks Telnet Server installed? (answering 'n' will return to the
previoius menu).
Default is 'y'
        -> y
```

3 Enter the GSWTS installation directory and confirm whether to overwrite existing files, if any. Back up any files before allowing them to be overwritten.

```
Please enter the GSWTS installation directory, including the drive letter.
Default is 'c:\gs_uts'
        ->


File 'C:\gs_uts\scripts\k_start.bat' exists.  Do you want to overwrite this file?
Default is 'n'
        -> y
```

4 You may be prompted to confirm the character client directory. The default value is the one you specified in step 4 on page 62.

```
Please specify the MFG/PRO Character Client directory.
Default is 'D:\mfgpro\eb2'
        ->
```

5 The system generates the `k_start.bat` file and places executable files in several directories.

```
Setup of Telnet files finished.

Press Enter to Continue
```

6 Press Enter to continue, return to the main menu, and exit the installation script.

You are now ready to use MFG/UTIL to configure the deployed QAD UI files.

Configure QAD UI Components

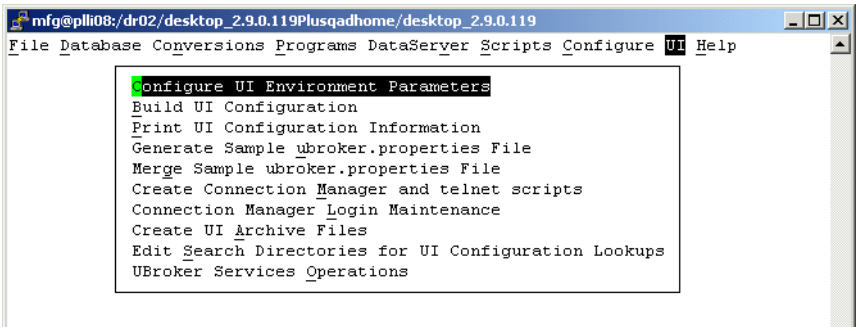
This section discusses the configuration activities you perform after deploying the database components and staging the Tomcat server files. These steps are:

- Add Parameter Files (Optional)
- Verify and Update System Information
- Build the QAD UI System
- Create Application Server JAR File
- Generate WebSpeed Configuration Information
- Generate Telnet Connection Scripts
- Print Configuration Information
- Extract the Web Application JAR File

During the installation activities, MFG/UTIL on the database server was updated with the functions needed to configure a QAD UI system. When you launch MFG/UTIL, choose UI to view the UI-specific menu options. Only the options used for a single-tier installation are discussed in this section.

During QAD UI configuration, you must specify the database set to use. If you have created `.pf` files manually without using MFG/UTIL functions, you can import these files into MFG/UTIL using the Import .pf Files to Database Set function on the MFG/UTIL Configure menu.

Fig. 6.1
MFG/UTIL UI Menu Options



Add Parameter Files (Optional)

By default, MFG/UTIL displays all available `.pf` files found in QAD 2008 Standard and client installation directories in the QAD UI configuration screen. You can specify additional directories for MFG/UTIL to search for other `.pf` files by using the UI|Edit Search Directories for UI Configuration Lookups option on the MFG/UTIL menu.

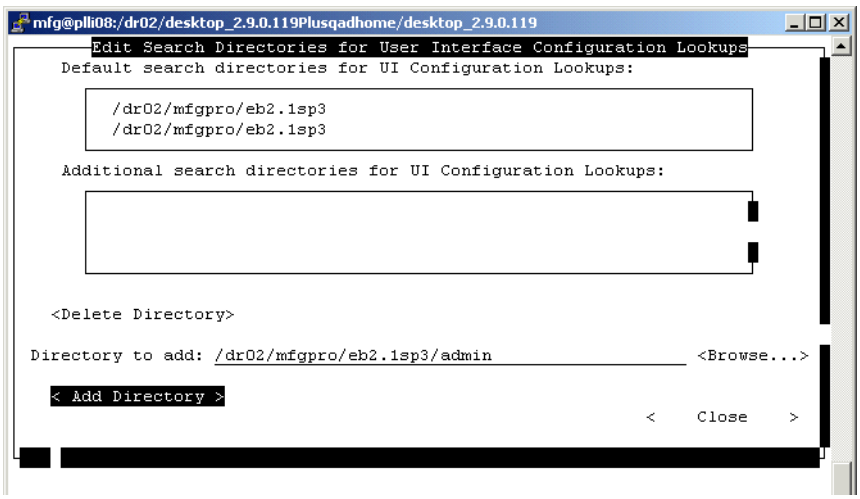
If your WebSpeed broker is on a different machine than the MFG/PRO databases, the parameter files must include client/server parameters to access the databases. These are the host (`-H`), server (`-S`), and network protocol (`-N`) parameters.

1 Choose 3 on the main menu to launch MFG/UTIL.

2 Select UI|Edit Search Directories for UI Configuration Lookups from the MFG/UTIL main menu.

3 To add a directory, enter its full path name in Directory to add and choose Add Directory. A `.ksh` file from a QAD UI installation must exist in the directory.

Fig. 6.2
Search Directories for UI Configuration Lookups



Note If MFG/UTIL cannot find a directory specified here or if no `.pf` files are located in a specified directory, an error displays for each affected directory when you access the Configure QAD UI Environment Parameters screen.

Delete Directory. Deletes the selected directory from the list of directories to search. You cannot delete the default directories.

Directory to add. Adds a directory to the search list. Optionally, use the browse to locate the directory in the file system and select it. After entering the directory, choose Add Directory to add it.

Verify and Update System Information

Complete this section to set QAD UI build parameters.

1 Start the database servers:

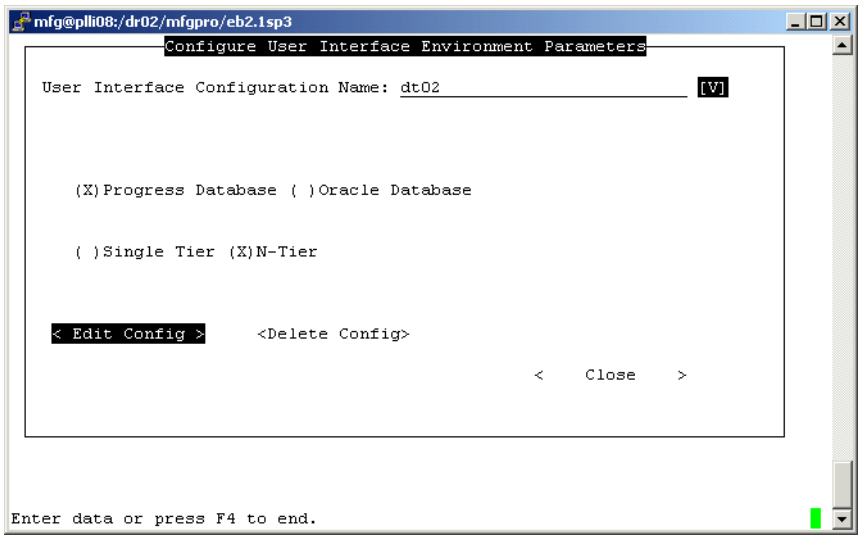
```
./start.Production
```

2 In MFG/UTIL, choose UI|Configure UI Environment Parameters.

3 If MFG/UTIL detects a new UI configuration, you are asked whether this is a Progress database configuration. Choose No if the new configuration is for an Oracle database; otherwise, choose Yes.

4 The Configure User Interface Environment Parameters screen displays. Select the name of the system you are configuring and choose Edit Config.

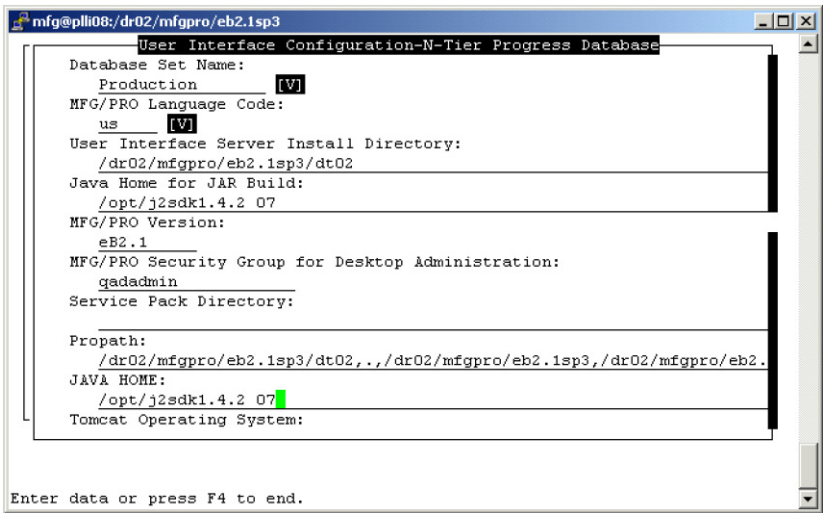
Fig. 6.3
Configure UI Environment Parameters



5 The system gathers configuration information and the configuration screen displays.

The screen differs slightly depending on the server operating system and database type. Some fields display only for Oracle database environments; the use of these fields is detailed in the field descriptions.

Fig. 6.4
First Multi-Tier Configuration Screen



QAD UI System Information

The first set of fields prompts for information related to your QAD UI configuration.

Database Set Name. The database set this UI connects to.

MFG/PRO Language Code. Select the language you want to use for screen displays.

User Interface Install Directory. Verify the location of the Desktop configuration installation directory.

Java Home for JAR build. This field displays only for a multi-tier installation. It displays the database server JAVA_HOME, which is used by MFG/UTIL to build archive files and run the QAD UI build. The Tomcat server and database server JAVA_HOME values will likely be different.

MFG/PRO Version. Verify the version.

Absolute Path to Oracle Install. (Not shown) This is the path to the Oracle installation directory, also known as ORACLE_HOME.

Java Home for JAR Build. This field displays only for a multi-tier installation. It displays the database server JAVA_HOME, which is used by MFG/UTIL to build archive files and run the QAD UI build. The Tomcat server and database server JAVA_HOME values will likely be different.

Oracle System Identifier Name. (Not shown) Enter the Oracle System Identifier (ORACLE_SID). This is the environment variable for the Oracle database and is typically the same as the Oracle database name.

ORASOPATHNAME. (UNIX only, not shown) Enter the path to *ORACLE_HOME*`/lib/libclntsh.`*xx* client library, where *xx* refers to the Oracle version. Your specific Oracle version and system configuration define whether this parameter must be set and the specific setting. Refer to your Oracle documentation for details.

Service Pack Directory. If your install includes a separate service pack installation, enter the service pack directory here. The appropriate service pack directories—*BaseServicePackDir* and the *BaseServicePackDir*`/xrc`—are automatically added to the QAD UI compile PROPATH.

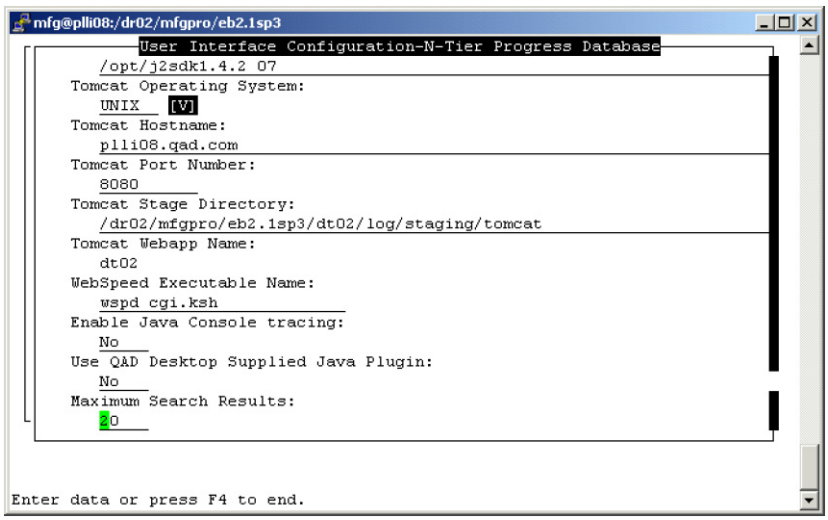
PROPATH. Use this field to enter any additional values for the PROPATH. The PROPATH for the build process requires the following directories, in this order:

- QAD UI install directory
- Current directory (`.`)
- *MFGPROInstallDir*
- *MFGPROInstallDir*`/us/xrc`

All directories specified in the PROPATH must exist.

Important Add custom and patched code directories to the end of the value in PROPATH.

Fig. 6.5
Second Multi-Tier Configuration Screen



Tomcat and WebSpeed Configuration Fields

This set of fields contains Tomcat information.

JAVA HOME. Verify the location of your Java installation.

Tomcat Operating System. Verify the operating system for the server where Tomcat resides.

Tomcat Hostname. Enter the fully qualified host name of the Tomcat server.

Tomcat Port Number. This value is automatically populated with 8080. Change this value as needed.

See “Update Tomcat Port Numbers” on page 85.

Note If you do not accept 8080, you must manually update the Tomcat server configuration file with the new port.

Tomcat Stage Directory. For multi-tier deployments, the staging directory where the Tomcat files were placed during the installation. MFG/UTIL customizes these files for your Tomcat server.

Tomcat Webapp Name. The name of your QAD UI configuration.

WebSpeed Executable Name. Enter `cgiip.exe` for Windows, `wspd_cgi.ksh` for UNIX.

See “Install WebSpeed Messenger” on page 135.

Note The WebSpeed Messenger executable must be located in the Tomcat `cgi` directory:

TomcatInstallDir`/webapps/`*qaduiConfig*`/WEB-INF/cgi/`

Java and Search Default

This set of fields contains Java Virtual Machine (JVM) and QAD UI search information.

See “Enable the Java Console” on page 143.

Annotation 5 As with Single-Tier, these fields refer to a Desktop-supplied plugin

Enable Java Console Tracing. Enter Yes to send trace messages to the Java JVM Console. To view the messages, you must also enable the console. Enabling trace may adversely affect performance. If you enable trace for debugging purposes, you must rebuild QAD UI to disable this option later.

Use QAD Desktop Supplied Java Plugin. The plug-in is now called the Java Virtual Machine (JVM). Indicate how you want to manage the JVM version for client sessions.

- Set to No to have UI clients use a JVM already installed on the client machine. If no JVM exists, or the minimum version is not found, the QAD-supplied version is downloaded.
- Set to Yes to have all clients use the JVM supplied on the QAD UI installation media. If other versions of Java exist on the client, they are not replaced and can continue to be used by applications other than the .NET UI.

To change this value at a later time, you must rebuild your QAD UI system.

Maximum Search Results. By default, the search function in the .NET UI returns 20 records at a time. You can change this value by specifying a different number here.

6 Choose OK to save the configuration.

7 Choose Close to exit to the main menu.

Build the QAD UI System

This portion of the UI creation lets you build a new .NET UI configuration based on the values entered in the UI Configuration screen. It also lets you:

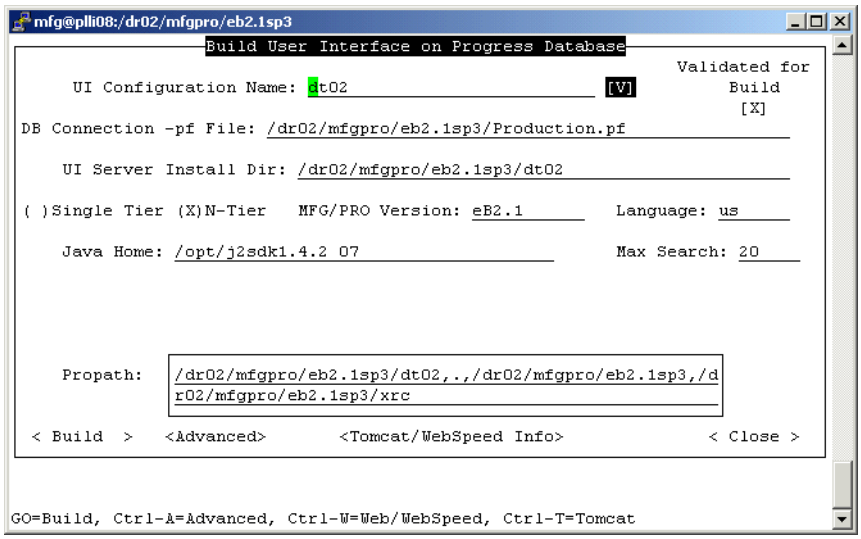
- Limit optional build features such as the search database.
- Build for a specific language.
- Rebuild a UI selectively based on what portions of the UI you have modified.
- Create the build script without running the build, which lets you run the build for one or more UIs at a later time when system resources are available.

Use the following steps to verify the system information and build the QAD UI system:

1 From the MFG/UTIL main menu, choose UI|Build UI Configuration.

2 In UI Configuration Name, choose the configuration to build.

Fig. 6.6
Build User Interface



The fields are populated with values for the configuration you chose.

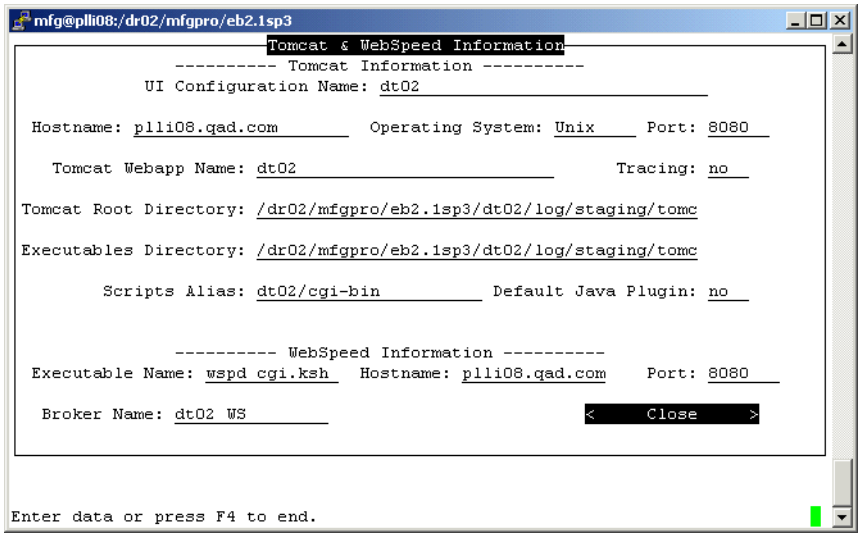
A UI configuration is Validated for Build for single database systems. Multi-database systems that have more than one `qaddb` included in the `.pf` files are not validated; they will build, but the compile will return errors.

If Validated for Build is not checked, you may proceed; however, after the build you must edit your `.pf` files to ensure that only the first instance of the `qaddb` database appears. This allows the compile to run without errors.

When you are building a UI for an Oracle environment, the confirmation screen displays additional Oracle-specific settings. Verify the Oracle Home, Oracle SID, and for UNIX environments, the Oracle Shared Library Path values.

3 Once you verify the values on this screen, choose Tomcat/WebSpeed Info to review the build information for the Tomcat server and WebSpeed components.

Fig. 6.7
Tomcat Server and WebSpeed Information Fields



4 After verifying the Tomcat and WebSpeed information, choose Close to return to the Build UI screen.

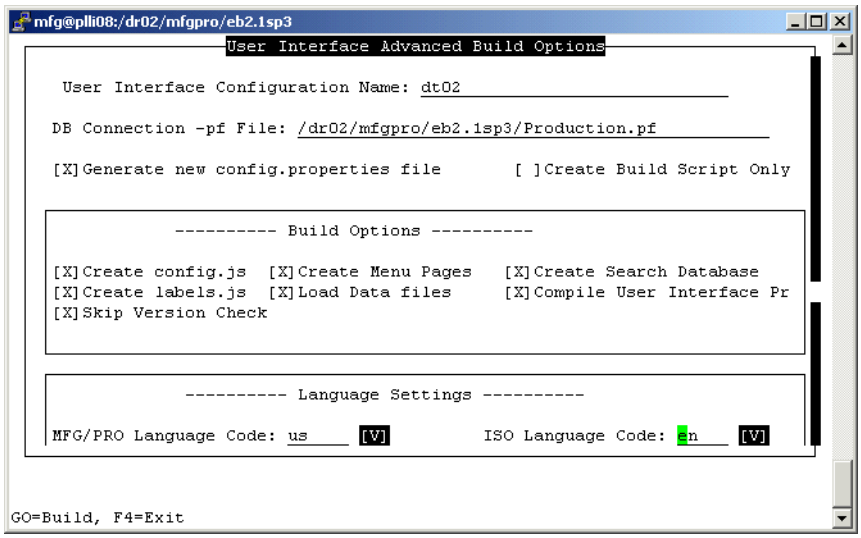
5 To set advanced configuration options—typically, used when building Oracle systems, rebuilding a UI, upgrading to a new release, creating build scripts without executing the build, or building multi-language environments—choose Advanced.

Note If you are building a multiple-language UI configuration, make sure you review the information in Chapter 10, "Implementing Multiple Languages," on page 105.

Advanced Build Options

As with other MFG/UTIL screens, this one may differ slightly depending on the QAD UI configuration and database environment you are working with.

Fig. 6.8
First Advanced Build Options Screen



User Interface Configuration Name. This field displays the QAD UI configuration you are currently updating; it cannot be modified.

DB Connection -pf File. This is the database connection parameter file that the build process uses to connect to the database to load system data and compile the QAD UI code. This file is set based on the database set you identified in the UI Configuration screen. If the wrong `.pf` is displayed here or you need to change it, you must exit this screen and specify the correct database set in the UI Configuration screen.

Generate new config.properties file. This option should be selected when a new QAD UI system is to be built. However, once a QAD UI system has been built and you are rerunning a specific build function—such as generating new menus because you changed menus—deselect this option. This option creates a new `config.properties` file and overwrites the existing one. If the existing file contains any customizations, they will be lost.

Create Build Script Only. Select this to generate build scripts without running the build. These scripts can later be used to build the QAD UI system. This option helps ease system administration activities in environments with multiple QAD UI systems. By creating build

scripts for all the QAD UI systems without running the build, the system administrator can create a batch or cron job to run the build for one or multiple QAD UI systems at a later time when system resources are available, or during scheduled down times.

The generated build script is saved to the QAD 2008 Standard installation directory. The file name is the QAD UI system name with the standard system extension, such as `qaduiConfig.bat` for Windows, or `qaduiConfig.ksh` for UNIX.

Build Options

Annotation 6 As with Single-Tier, are these options relevant for .NET UI?

Select the options you want to use to control this build. These options are affected by values defined in the `config.properties` file.

See “Changing Menus” on page 123.

Typically, you enable all build options the first time you build the system. Later, you can change the `config.properties` file and rebuild with or without all options selected. In some cases, you may not need all options. For example, if you change your menu system, you can build only the search database and menu pages to incorporate your changes.

Create config.js file. Enable this option to create the JavaScript code that incorporates your environment values. This script is used to build the QAD Desktop menu system.

See “Language Directories for Menus” on page 135.

Create Menu Pages. Enable this option to create the HTML menu pages for the Desktop UI. This option updates the menus in the two-letter language code subdirectory in the Tomcat Web applications directory:

TomcatInstallDir`/webapps/`*qaduiConfig*`/menus`

Create Search Database. Enable this option to build the database accessed by the Search option in the Desktop UI. You can search by menu label, menu number, short name, and execution file based on data specified in Menu System Maintenance (36.4.4) and Menu Substitution Maintenance (36.20.6).

Create labels.js file. Enable this option to create the JavaScript code for displaying MFG/PRO labels that are used on the Desktop UI main pages. Labels and their associated terms are stored in the label master table in the database and maintained with Label Master Maintenance (36.4.17.1). Labels are extracted and placed in language-specific files based on the language specified for the MFG/PRO language code prompt.

Load Data files. Enable this option to load terms and labels used in the Desktop from an external file into your MFG/PRO database. This data needs to be loaded only once to update your MFG/PRO database. If you are creating multiple UI configurations for the same databases, you can run each UI configuration build with this option enabled without any negative effect.

Compile User Interface Progress Programs. Enable this option to compile the Progress programs supplied with the user interfaces.

Skip Version Check. The build code checks the version against the UI version. If an unsupported version is detected, the build stops. This is true even for currently supported versions that were previously not supported.

If you are installing this UI configuration against a version that was not previously supported, check this box. You will see a warning about the version mismatch rather be stopped from completing the build.

Language Settings

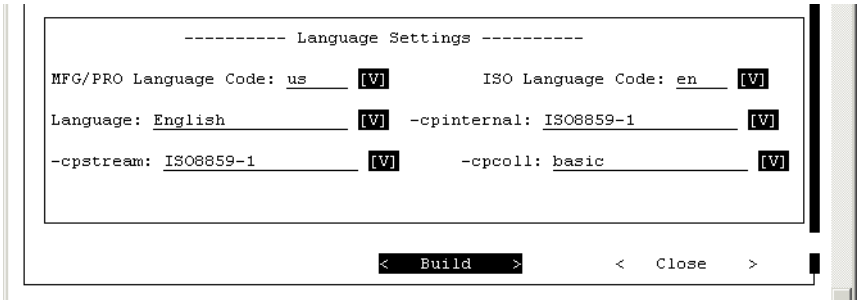
Language settings are important if you are installing a UI configuration into a system that supports multiple languages. For a single- language implementation, you typically do not need to change any of these settings. QAD UI is preconfigured with appropriate settings for each supported language and these display by default, based on the QAD (MFG/PRO) language code you selected in the previous screen.

Each language has predefined values for ISO language code, Progress startup parameters, and the Oracle NLS setting.

For additional details, see the section “Use MFG/UTIL to Install Multiple Languages” on page 110.

However, if you are installing multiple languages, you must execute the advanced build options for each language to create language-specific menus, search database, and other required support data. In this case, you may need to modify the language settings for additional languages from the default values to conform to your database settings.

Fig. 6.9
Advanced Build Options, Language Settings



MFG/PRO Language Code. Select the MFG/PRO language code from the drop-down list that represents the language you want to use for screen display, such as US for US English. MFG/UTIL builds the list of available languages based on the language directories you have currently installed.

Note This language code is not necessarily the same as the ISO standard language code used by Java.

ISO Language Code. This field displays the ISO language code associated with the MFG/PRO language code in this screen. You should not change this value. The Desktop UI is preconfigured with the correct mapping between supported MFG/PRO codes and ISO codes.

See “Language Directories for Menus” on page 135.

When menus are generated, the HTML files are placed in a directory that is named based on this ISO language code. However, even when a non-English language is specified, some files are also created in the `en` directory.

Language. This field displays the full language name associated with the language code you selected. Do not change this value.

For details on these parameters, see the Progress *Startup Command and Parameter Reference*.

–cpinternal. The database startup parameter that identifies the Progress code page for graphical clients. This should be the same as the operating system code page. If a value is not specified, iso8859-1 is used. When you build a multiple-language UI configuration, change this value to the code page associated with the database.

–cpstream. The database startup parameter that identifies the code page for stream I/O. Stream I/O includes character clients, data files, and code. When you build a multiple-language UI configuration, change this value to the code page associated with the database.

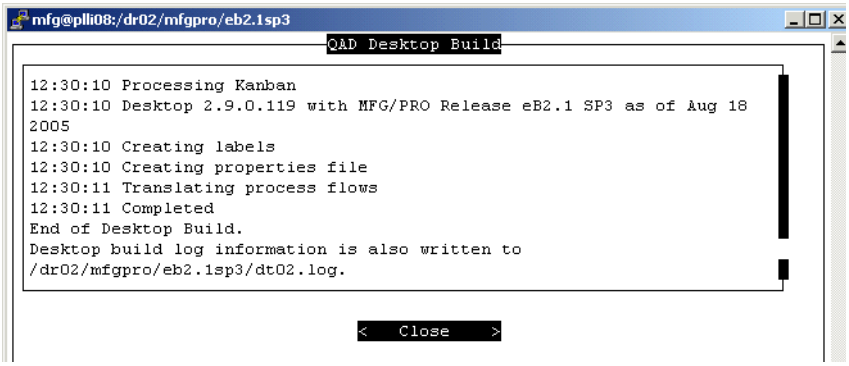
–cpcoll. The database startup parameter that identifies the collation table to use with the code page identified by the `-cpinternal` parameter. Progress uses the collation rules that you specify to compare characters and sort records. When you build a multiple-language UI configuration, change this value to the collation table associated with the database.

Oracle NLS Setting. This field sets the NLS_LANG parameter required in the WebSpeed `ubroker.properties` file and telnet startup scripts for the Oracle dataserver when Native Language Support (NLS) is required for non-English and double-byte languages.

6 Choose Build and confirm you have reviewed the build information to build the system. The build log displays as the build occurs.

Annotation 7 This screen shows a Desktop build. Does the install still show Desktop build details at this stage.

Fig. 6.10
Sample Build Log Display



7 When the UI build completes, review the log for errors. Correct any errors before you proceed. Choose Close.

One of the files generated by the build is *MFGPROInstallDir/*`dt2`*qaduiConfig*`.ini`. For a configuration named dt02, this would be `dt2dt02.ini`. This file is required during the JAR file extraction.

You are ready to create the archive file that will be used to deploy the configured Web application on the Tomcat server.

Create Application Server JAR File

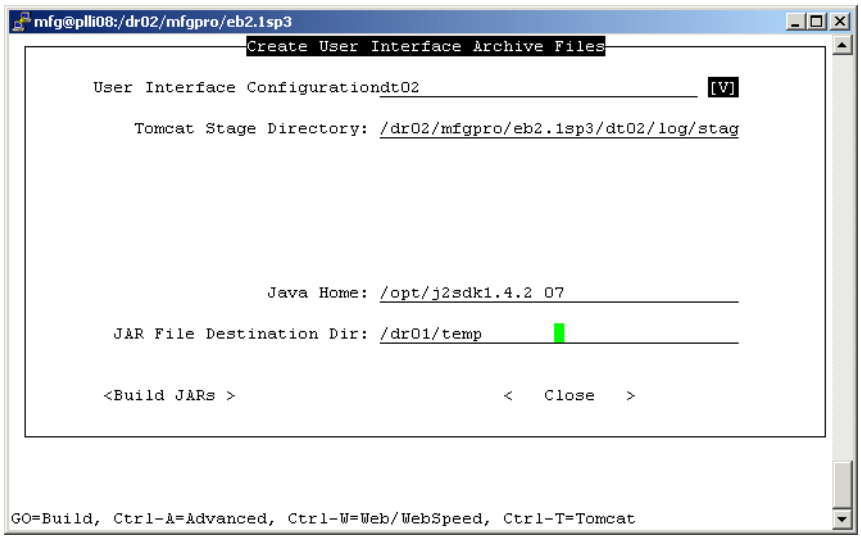
Use the following steps to build the archive file for the remote Tomcat server.

Important If you are deploying on two different operating systems, for example Linux and Windows, you must create the JAR files in the same format as the destination operating system.

1 In MFG/UTIL, choose UI|Create UI Archive Files.

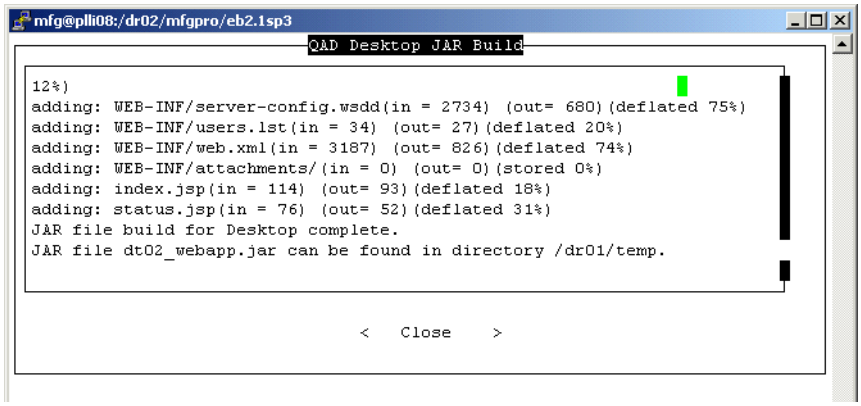
2 The Create QAD UI Archive Files screen displays. Select your configuration name. Your stage directory displays.

Fig. 6.11
Create UI Archive Files



3 Specify the destination directory for the JAR file and choose Build JARs. The destination directory must be an existing directory location. In the example, `/dr01/temp` is a network share where user `mfg` has write permissions.

Fig. 6.12
Build JAR Log



4 When the process completes, review the log for errors. If the JAR file was not generated directly to a network share, transfer it to a network share or to a local drive on the Tomcat server.

You are now ready to create the connection and configuration files. These files are required later during Connection Manager and WebSpeed configuration steps.

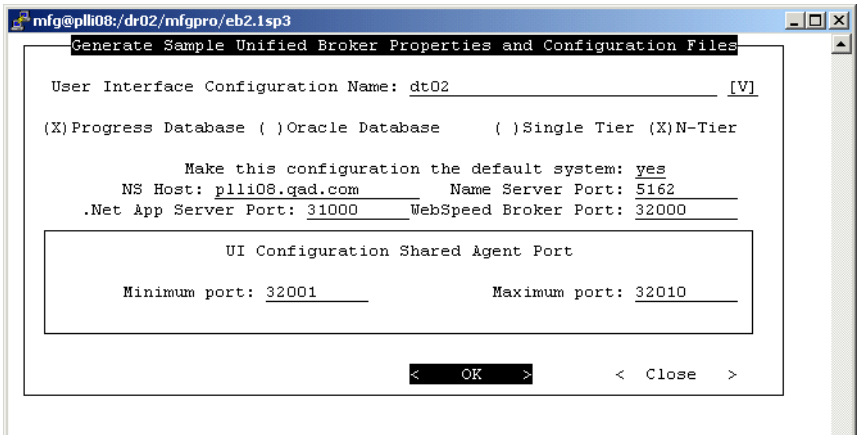
Generate WebSpeed Configuration Information

To update the WebSpeed `ubroker.properties` file with the QAD UI broker definitions, generate a parameter file in MFG/UTIL. The parameter file is merged into `ubroker.properties` in the next set of steps.

If you are implementing the QAD .NET UI, this screen displays additional fields in order to generate the Progress AppServer definitions.

1 In MFG/UTIL, choose UI|Generate Sample ubroker.properties File. The Generate Sample Unified Broker Properties and Configuration Files frame displays.

Fig. 6.13
Generate Sample Unified Broker Properties and Configuration Files



2 Choose the configuration you are generating the file for in User Interface Configuration Name. Update the other fields as follows:

Progress Database/Oracle Database. Display-only. If this value is incorrect, the install script must be run again.

Single-Tier/N-Tier. Display-only. If this value is incorrect, the install script must be run again.

See “Sample XML Configuration File” on page 121.

Make this configuration the default system. Set this to Yes to generate both a `qaduiConfig.xml` and a `default.xml` file. The default configuration is the first configuration users log into. After initial login, they can access other configurations in the same environment by choosing Tools|Options and setting the System Preferences. When multiple systems are available, users can also choose to select from a list during login.

NS Host. The machine where the Progress Name Server is installed.

Name Server Port. By default this is 5162.

.NET App Server Port. The port for the .NET UI Progress AppServer. The default is 1300. Make sure the port you enter here is unused and available.

WebSpeed Broker Port. This port is used only by the .NET UI WebSpeed broker. The default is 5600. Make sure the port you enter here is unused and available.

UI Configuration Shared Agent Minimum Port. The minimum port for a range of shared agent ports available for UI user sessions. The typical range is 10. Make sure the entire range of ports you enter here is unused and available.

UI Configuration Shared Agent Maximum Port. The maximum port for a range of shared agent ports available for UI user sessions. Make sure the entire range of ports you enter here is unused and available.

3 Choose OK to generate the file for the UI configuration you selected.

The system displays a message that the file, *qaduiConfig*-ubroker.properties, was saved in your *MFGPROInstallDir*. The generated file contains the complete definitions required by Progress to launch the AppServer and WebSpeed broker.

Generate this file for each UI configuration you are setting up.

Merge ubroker.properties Files

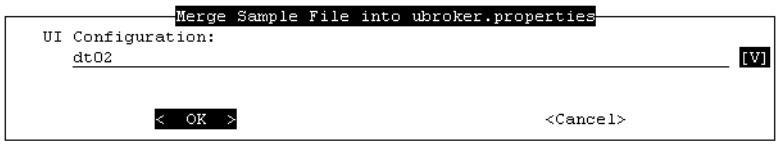
The automated merge process creates a backup of the standard Progress ubroker.properties file and then inserts the file you generated at the bottom of the original file. For example, *ProgressInstallDir*/ properties/ubroker.properties.002 is created, then the content of *MFGPROInstallDir/qaduiConfig*-ubroker.properties is inserted at the bottom of *ProgressInstallDir*/properties/ ubroker.properties.

If you are rerunning the merge process for a given UI configuration, the existing broker and AppServer definitions are deleted from the base ubroker.properties file before the new definitions are added.

It is worthwhile to review the content of *MFGPROInstallDir/ qaduiConfig*-ubroker.properties prior to merging. Make sure all the values look correct given the setup completed to this point.

1 In MFG/UTIL, choose UI|Merge Sample ubroker.properties File. The Merge Sample File into ubroker.properties frame displays.

Fig. 6.14
Merge Sample File into ubroker.properties



2 Choose the configuration you are merging in.

3 Choose OK. The log displays.

4 Choose Close to complete the process.

Generate Telnet Connection Scripts

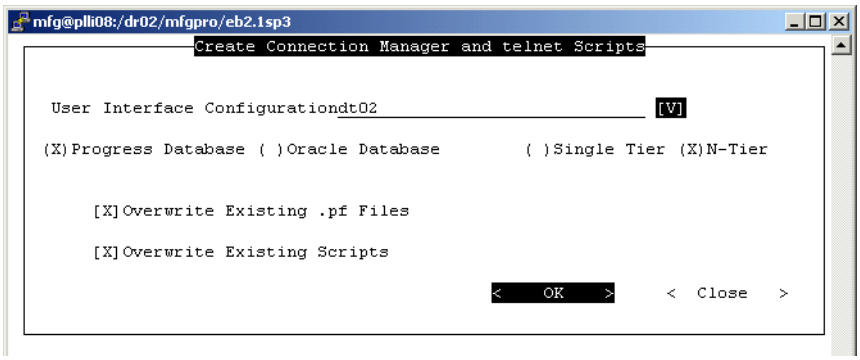
See “Update Configuration Settings” on page 98.

Use MFG/UTIL to generate the Connection Manager and telnet connection scripts. Later on, these values default into the Connection Manager.

1 In MFG/UTIL, choose UI|Create Connection Manager and telnet scripts from the main menu. The create scripts screen displays.

2 Select the UI configuration. The database type and tier type display. Use the check boxes to overwrite existing `.pf` files and scripts built during your base QAD 2008 Standard install. Choose OK to generate the scripts.

Fig. 6.15
MFG/UTIL
Create Connection Manager and telnet Scripts



3 Using the specified configuration information, MFG/UTIL generates two files in the UI installation directory. The file names depend on the operating system specified in the configuration settings screen:

- For UNIX servers, the `connmgr.`*`DBSetName`* and `telnet.`*`DBSetName`* files are generated.
- For Windows servers, the `connmgr`*`DBSetName`*`.bat` and `telnet`*`DBSetName`*`.bat` files are generated.

New *`DBSetName`*`.pf` and `.ini` files are created in *`MFGPROInstallDir`*.

`DBSetName` refers to the database set name specified in the build screen.

Depending on the release you are upgrading from, some files may be backed up. If any `.pf` or telnet connection script files are found in the *`MFGPROInstallDir`* or *`QADUIInstallDir`* and you choose not to overwrite them, they are moved to an `old_dtscripts` directory.

Important In some UNIX environments (HP-UX and AIX), you may need to include additional parameters such as SHLIB_PATH and LIBPATH in the generated files. For details, refer to your operating system documentation.

Annotation 8 Are these values relevant for .NET UI ?

The Desktop UI uses values defined in the Progress `protermcap`. The keys listed in Table 6.2 must have their default setting or actions in the Desktop UI may fail. If you have modified keys in the `protermcap` file, change the PROTERMCAP environment variable to point to an unchanged file.

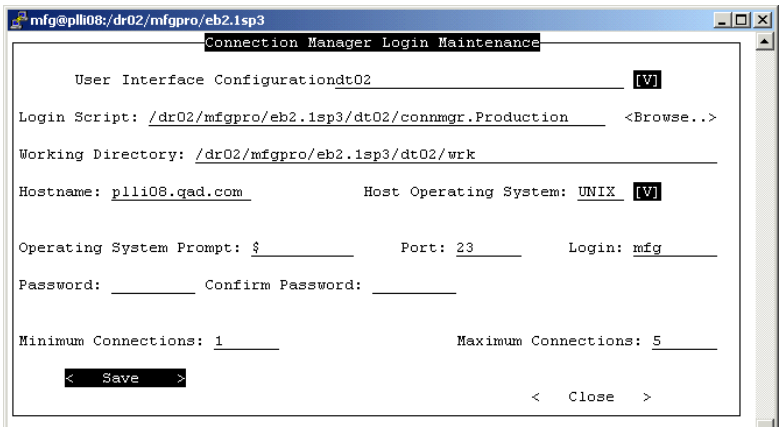
Table 6.2
Function Keys

UNIX	Windows
F16	Shift-F5
F17	Shift-F6
F18	Shift-F7
F19	Shift-F8
F20	Shift-F9

When the Connection Manager scripts have been created, use MFG/UTIL to create Connection Manager log-in scripts:

1 In MFG/UTIL, choose UI|Connection Manager Login Maintenance. The Login Maintenance screen displays.

Fig. 6.16
Connection Manager Login Maintenance



2 Choose the configuration you are setting up in User Interface Configuration. Update the other fields as follows:

Login Script. Locate or type in the `connmgr` script generated in the previous set of steps; for example, *`MFGPROInstallDir/ qaduiConfig/`*`connmgr.Production`.

Working Directory. Enter a directory where temporary files can be written during telnet sessions. This directory must exist and must have write privileges for user `mfg`.

Hostname. The database server host machine, including domain value; for example, hpux01.corp.com.

Host Operating System. This defaults in and should be correct.

Operating System Prompt. This, and the next few fields through the password, are used to construct the log-in sequence for the session telnet connections to QAD 2008 Standard. The prompt is typically a dollar sign ($), pound sign (#), or greater than sign (>) depending on your operating system and UNIX shell.

Port. The operating system telnet port. Typically this is 23. Check your operating system to verify.

Login. The telnet user, typically `mfg`.

Password/Confirm Password. Enter the user password and then confirm it. No characters display during data entry in these fields for your security.

Annotation 9 Are these connections values valid for .NET UI?

Minimum Connections. Enter the minimum number of telnet connections that the system will keep open during a Desktop session.

Maximum Connections. Enter the maximum number of telnet connections that the system will open during a Desktop session.

3 Choose Save to update the login script.

4 When the save is complete, you are informed that file `connectionManagerConfig.xml` was updated and a backup of the original file was saved to `connectionManagerConfig.xml.back`. Choose OK to continue.

5 Choose Close to exit.

Print Configuration Information

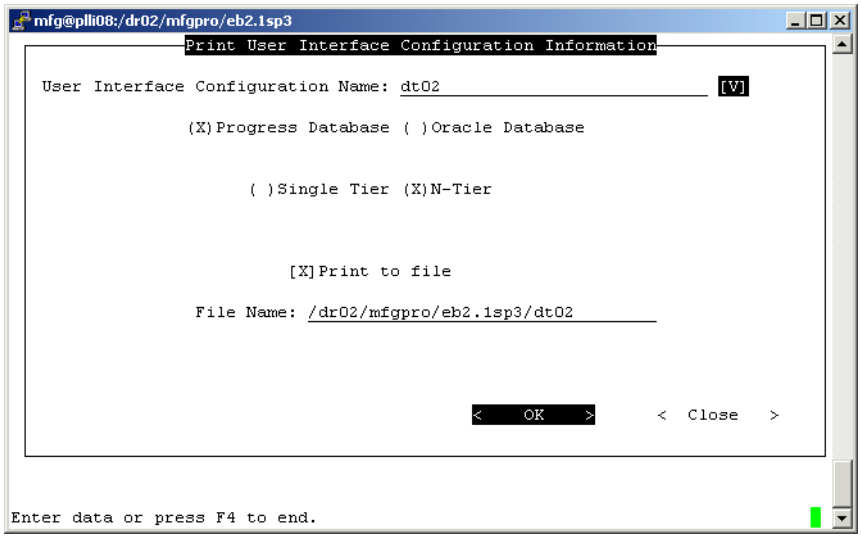
You can use MFG/UTIL to generate a text file containing the complete configuration information, including PROPATH information, directory locations, and port information for your UI configurations. A separate PROPATH is displayed for each of the following:

- QAD UI Build PROPATH
- Telnet Client PROPATH
- Connection Manager PROPATH
- WebSpeed PROPATH

1 In MFG/UTIL, choose UI|Print UI Configuration Information.

2 When the Print UI Configuration screen displays, select your configuration name and choose whether to send the output to a file or printer.

Fig. 6.17
MFG/UTIL Print UI Configuration Information



3 Choose OK. File output is saved by default to *MFGPROInstallDir*.

Extract the Web Application JAR File

Use the following steps to launch the installation script and deploy the Web application JAR file to the Tomcat server. Before you proceed, make sure the `.ini` and JAR files are accessible from the Tomcat server or you have copied them to a local directory on the Tomcat server.

1 On the Tomcat server, launch the installation script from the QAD UI installation media.

2 At the log file prompt, point to the `.ini` file created for the original UI configuration install (if it resides on an accessible drive). The file was created in step 2 on page 61.

```
Please enter a directory to write log files
Default is '/home/mfg/mfgsvr'
        ->/dr02/mfgpro/eb2.1sp3/dt02/log
```

3 At the configuration file selection prompt, choose the number for the configuration you are extracting.

```
*** Configuration file selection ***
Please choose one of the following:

        1: dt02-dt.ini -> dt02
        2: Create new configuration
        3: Cancel
<1-3>? 1
```

4 The main installation menu displays. Choose 2 to launch the Post MFG/UTIL QAD UI Build Jar Install.

See “Set JAVA_HOME” on page 30.

Note You may be prompted to enter or confirm the JAVA_HOME value.

```
*** Configuration: dt02 ***
*** Main Menu ***
Please choose one of the following:

        1: Install MFG/PRO - Progress QAD UI Files Menu
        2: Post MFG/UTIL QAD UI Build Jar Install
        3: Select a Different Configuration
        4: Exit
<1-4>? 2

The JAVA_HOME environment variable for this machine is set to '/opt/j2sdk1.5.0_06'.
Is this correct?
Default is 'y'
        ->
```

5 When prompted, enter the directory where the JAR file is located.

```
Please enter the location of the MFG/UTIL created JAR files.
Default is '/home/mfg/mfgsvr/staging'
        -> /dr01/temp
```

The JAR files are located in the shared file location identified in step 3 of “Create Application Server JAR File” on page 74.

6 The JAR extract menu displays.

```
*** Configuration:  dt02 ***
*** Jar Extract Menu ***
  Please choose one of the following:

        1: dt02_webapp.jar
        2: Return to Main Menu
```

```
<1-2>?
```

This menu displays the JAR files in the directory you specified. The file name consists of the configuration name, followed by `_webapp.jar`.

Deploy the Customized Web Application Files

Use the following steps to expand the Web application JAR file, which contains the customized Web application files that reside on the Tomcat server. Additionally, if Tomcat is not already installed on this server, this step also installs the QAD-provided version.

1. From the JAR extract menu, choose the option to deploy the *qaduiConfig*`_webapp.jar` to the local Tomcat server directories.

2. The system reviews your current configuration to see if the file can be extracted to the local server. Review and confirm the system configuration prompts. This includes verifying the server name and whether it is a Tomcat server.

   ```
   This machine's name is 'hpux09.qad.com'.
   Is hpux09.qad.com a Tomcat server?
   Default is 'y'
           ->
   ```

3. Enter and confirm the Tomcat directory. If Tomcat is not already installed, it will be installed here. If a QAD-compatible version of Tomcat is already installed, only the customized Web application file is installed.

   ```
   Please enter the Tomcat installation directory.
   Default is '/tomcat'
           ->

   The directory '/tomcat' does not exist.  Would you like to create this directory?
   Default is 'y'
           ->
   ```

4. Confirm the default Tomcat port or enter a new one.

   ```
   Please enter the Tomcat port number.
   Default is '8080'
           ->
   ```

See “Update Tomcat Port Numbers” on page 85.

Important Manual updates are required when Tomcat does not use the default 8080 port or if another application is already using port 8080.

5. You may be prompted to install Tomcat; confirm that Tomcat should be installed.

   ```
   It does not appear that Tomcat is installed at '/tomcat'.
   Do you want to install Tomcat to this directory now?
   Answering 'n' will cancel the extraction of '/dr01/temp/dt02_webapp.jar'.
   Default is 'y'
           ->
   ```

   The installation scripts extract the JAR file and deploy the configured files to the proper directories. Review the logs. At the conclusion of the file extraction, the JAR extract menu redisplays.

   Installation activities on the Tomcat server are complete; you can now exit the installation script. You are ready to perform the setup activities in Chapter 7, “Configuring UIs After Deployment,” on page 83.

Chapter 7

Configuring UIs After Deployment

Use this chapter to configure your environment after you install and configure the QAD UI files.

Post-Deployment Configuration

In this chapter complete the following tasks:

- Start WebSpeed.
- Configure and Start Tomcat.
- Prepare Telnet UI Connection Scripts
- Set Up SSH for QAD .NET UI

Start WebSpeed

The WebSpeed server is used for admin log-in sequences and for user validation. The WebSpeed `ubroker.properties` file holds all the configuration information used by WebSpeed. Use these steps to start the WebSpeed servers and broker.

Note Review the log files in the Progress work directory for diagnostic help.

1 Start the AdminServer using the following command:

```
ProgressHome/bin/proadsv -start
```

Validate that it is running using:

```
ProgressHome/bin/proadsv -query
```

2 Start the NameServer for your broker:

```
nsman -name NS1 -start
```

Check the status of the NameServer using:

```
nsman -name NS1 -query
```

3 The MFG/PRO database servers for the Production database set should already be running. You can verify this by attempting to restart them. Go to *MFGPROInstallDir* and run (for UNIX):

```
./start.Production
```

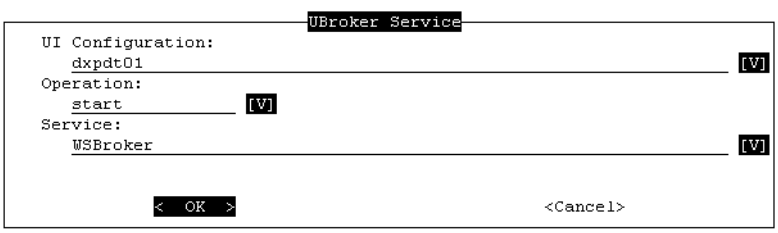
For Windows:

```
startProduction.bat
```

If they are not running, they will start. If they are running, a message for each database displays informing you the databases are running.

4 The remaining steps can be done in MFG/UTIL. In MFG/UTIL, open UI|UBroker Services Operations.

Fig. 7.1
UBroker Services Operations Set to Start the WebSpeed Broker



5 Select the UI configuration.

6 Set the Operation to start, stop, or query.

7 Select either the WebSpeed broker or the AppServer. Start the WebSpeed broker first, then the AppServer. Query each one after you start it to make sure it is up and running.

8 Choose OK to launch the operation. The log window displays.

9 Choose Close in the log window to proceed to the next operation.

To start, stop, or query the UI configuration WebSpeed broker from the Progress `/bin` directory, use the following syntax:

```
wtbman -name uiconfig_WS -start
wtbman -name uiconfig_WS -query
wtbman -name uiconfig_WS -stop
```

For the UI configuration AppServer, the syntax is:

```
asbman -name uiconfig_AS -start
asbman -name uiconfig_AS -query
asbman -name uiconfig_AS -stop
```

Configure and Start Tomcat

The configuration task consists only of updating the Tomcat port number. Updating the port number is required only in these situations:

- You did not accept the default 8080 port when you installed the Tomcat servlet engine.
- You have multiple Tomcat instances on the same server, which may result in port conflicts.
- Other applications already use the default Tomcat port. For example, Oracle 9i uses port 8080 as the default port for its Web server.

See page 38.

Update Tomcat Port Numbers

Use these steps to update the `server.xml` configuration file for Tomcat with the new port.

1 In a text editor, open *TomcatInstallDir*`/conf/server.xml`.

2 Look for the following parameter block and update the port reference:

```
<!-- Define a non-SSL Coyote HTTP/1.1 Connector on port 8080 -->
<Connector port="8900"
    maxThreads="150" minSpareThreads="25" maxSpareThreads="75"
    enableLookups="false" redirectPort="8443" acceptCount="100"
    debug="0" connectionTimeout="20000"
        compression="on"
    disableUploadTimeout="true" />
<!-- Note : To disable connection timeouts, set connectionTimeout value to 0 -->
```

3 If you need to change the server port from the default 8005 value, locate the following parameter and update the port reference:

```
<Server port="8905" shutdown="SHUTDOWN" debug="0">
```

4 Save the file.

Start Tomcat

To start Tomcat, launch the startup file in the *TomcatInstallDir*/bin directory:

- In Windows environments, use the `startup.bat` file.
- In UNIX environments, use `startup.sh`.

Note On UNIX, you may have to log in as `root` to start Tomcat.

Prepare Telnet UI Connection Scripts

The telnet setup completed in MFG/UTIL creates scripts used by the Connection Manager for managing programs. Additional telnet scripts must be created using User Option Telnet Maintenance (36.20.10.3). These scripts are used for character-based sessions in stand-alone Desktop and for running terminal sessions within the QAD .NET UI. In addition, the QAD .NET UI reads the port value defined in this program to determine which port to use to connect to the server for terminal programs.

Configuring the settings in User Option Telnet Maintenance consists of the following tasks:

- Specifying telnet server settings
- Defining the log-in sequence
- Configuring the telnet connection settings
- Verifying the script log-in sequence

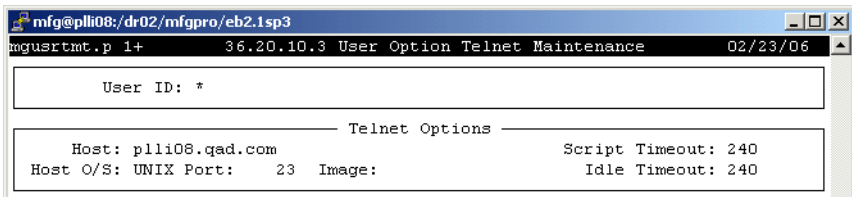
Specify Telnet Server Settings

Use the following instructions to configure telnet server settings:

1 In MFG/PRO, access User Option Telnet Maintenance (36.20.10.3).

2 In the User ID field, enter * to specify a generic record and press Go.

Fig. 7.2
User Option Telnet Maintenance, Telnet Options



3 Use the following descriptions to complete the Telnet Options fields in User Option Telnet Maintenance.

Host. Enter the fully qualified machine name or IP address of the telnet server. The script uses this information to establish the telnet connection.

Host O/S. Enter `UNIX` for UNIX systems. Enter `NT` for Windows systems.

Port. Enter the port number for the telnet server. The default value is 23. This is the value you would normally use, unless you are using SSH under the .NET UI. In this case, the port value is 22. See “Set Up SSH for QAD .NET UI” on page 91.

Image. Leave this field blank; images do not apply to the QAD user interfaces.

Script Timeout. Enter the number of seconds (1-999) the system will wait for the telnet log-in script to execute. If this value is exceeded, a time-out message displays and the session closes.

Idle Timeout. Enter the number of seconds (1-999) the system will wait after a telnet session begins for an MFG/PRO program to execute.

Note Idle timeout is not used in the QAD .NET UI.

4 Press Go to continue.

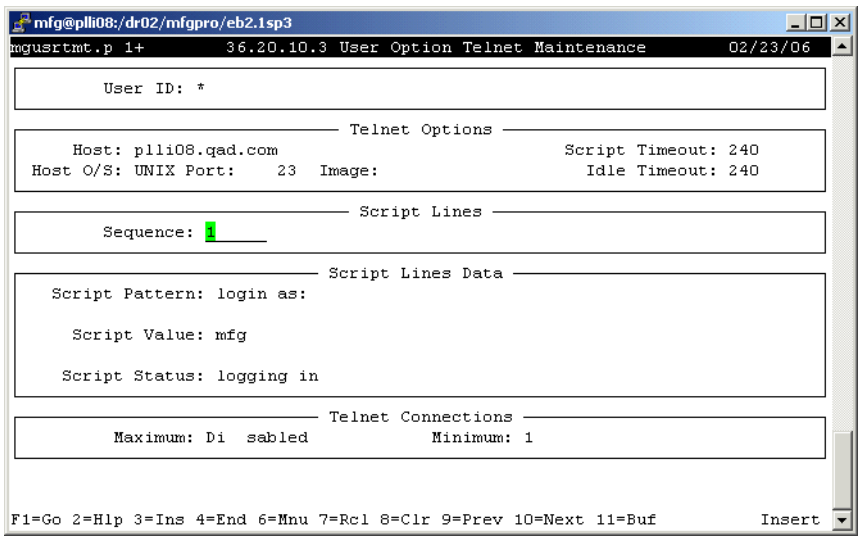
Define the Log-In Sequence Script Lines

To log in to and begin a telnet session on the telnet server, you must provide the sequence of telnet server log-in prompts and responses. The last value in the sequence specifies the script you created in MFG/UTIL. Following these step-by-step instructions is a set of sample script values for Windows and UNIX systems.

Important In the .NET UI, the log-in sequence must be specified, but the values are ignored. In character display mode, the user’s .NET UI user ID and password must match their UNIX user ID and password exactly.

When defining paths for scripts used in the .NET UI, avoid using relative paths since each user’s access may be different.

Fig. 7.3
Log-In Script Line



5 Specify the telnet log-in sequence number in the Script Lines frame. For each telnet command, enter a sequence number beginning with 1, and press Go. In the next frame, enter the following:

Script Pattern. Enter the prompt generated by the telnet server when a telnet log-in occurs. The values in this field must be identical to the prompts the telnet server displays when users log in.

Script Value. Enter the response to the telnet log-in prompt defined in Script Pattern.

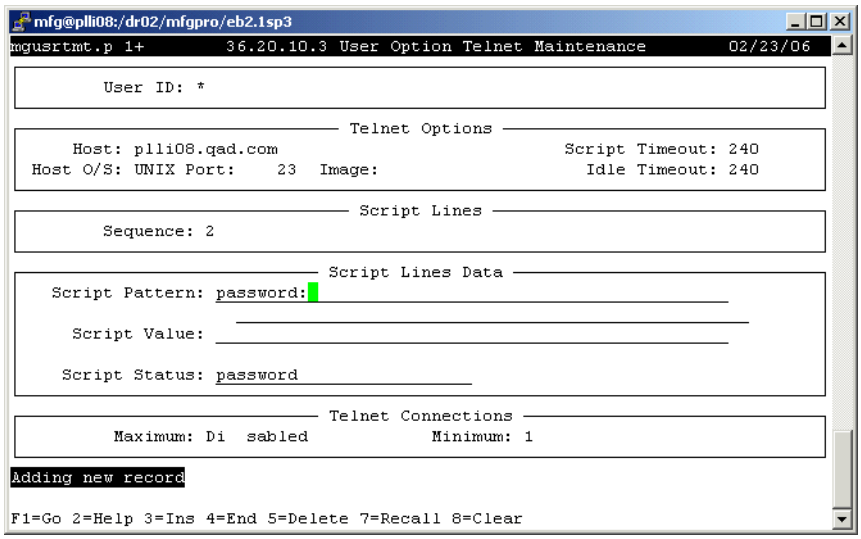
Script Status. Optionally enter a description of the prompt and response; for example, Logging In.

If you have tracing enabled and the Java console is displayed, the description in the Script Status field displays in the Java console on the Desktop client when an error occurs during the execution of the prompt and response. You can use these descriptions as an aid in troubleshooting telnet session issues.

Note In QAD 2008 Standard, when you enter a password as a script value, only blanks display. When you press Go at the end of the sequence, you are asked to confirm the password.

6 Press Go after entering the sequence values. You return to the Sequence field to enter the next sequence number and values.

Fig. 7.4
Second Log-In Script Line



7 After entering the final sequence, press Go to return to the Sequence field. Then press End to move to the Telnet Connections fields.

Configure Telnet Connection Settings

Annotation 10 are these setting relevant for .NET UI? See the note below about HTML in Desktop

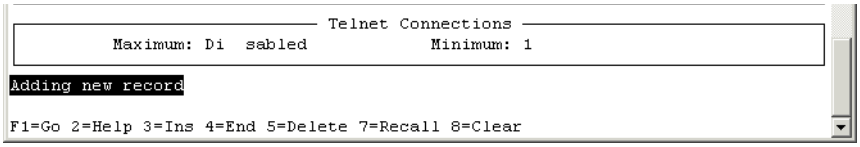
Once you configure and verify your telnet log-in sequence, access the Telnet Connections frame and specify telnet connection settings. These settings define the maximum and minimum number of telnet connections available to the associated user.

Note These settings apply to both telnet and HTML maintenance programs in Desktop; they are ignored in the .NET UI. The maximum number of open connections in the .NET UI is determined by the `MaximumDesktopsPerWorkspace` setting in *`qaduiConfig`*`.xml`, located in:

`TomcatInstallDir``/webapps/qadhome/client/plugins/mfgpro`

Recommended settings are 10 or more for Maximum; 1 for Minimum.

Fig. 7.5
Telnet Connections



Maximum. This value specifies:

- The maximum number of concurrent embedded telnet screen connections this user can have open per session.
- The maximum number of detached windows running HTML programs allowed for the user. If a specific record does not exist for a user with this setting defined, that user can continue opening detached windows until the maximum number of sessions allowed for the entire pool is reached.

This setting applies separately to HTML telnet sessions and standard telnet sessions in the Desktop UI. For example, if Maximum Telnet Settings is 5, a user can have 5 maintenance programs running and 5 telnet programs running in one Desktop session before an error displays.

Note The maximum number of concurrent browses is determined by Max Web Connections in User Option Telnet Maintenance. Valid values are:

- Unlimited: The associated user can have an unlimited number of concurrent telnet connections open.
- Disabled: The associated user cannot log in through Desktop. Until you create a log-in script to initiate telnet sessions for this user, you cannot set this field to any value other than Disabled.
- Any numerical value between 1 and 99.

Minimum. Enter a value between 0 and 9 to indicate the minimum number of telnet connections to be available to the associated user at all times.

Set this value to the number of telnet programs the user is likely to run simultaneously. Specifying a value here can dramatically reduce the wait time for these programs to display in the Desktop UI. However, setting this value too high depletes system resources.

QAD recommends that you set Minimum to 0 (zero) for most users, including the generic user—defined with an asterisk (*). If users access Desktop telnet maintenance programs extensively, set Minimum to 2.

8 Press Go to save the record.

Sample Scripts

Review the following sample log-in scripts to become familiar with the configuration process. Then create your log-in sequence based on your environment. For reference purposes record your scripts in the table provided. View the MFG/PRO field help for additional field-specific details.

Note Windows log-in scripts typically include the log-in domain. Domain names should adhere to the Internet Engineering Task Force (IETF) published standards. For more information, refer to document RFC1035 found at the following URL:

http://ietf.org/rfc/rfc1035.txt?number=1035

Table 7.1 illustrates sample script lines for a Windows system.

Table 7.1
Sample Windows Script Values

Seq	Script Pattern	Script Value	Script Status	Notes
1	login:	user1	Logging In	This ID must correspond to a local user account on the Windows server.
2	password:	pswd	Supply Password	Enter the password assigned to the user ID specified in step 1.
3	domain:	domain1	Supply Domain	Enter your Windows telnet server's domain, if required.
4	>	c:\telnet\user1	Start GTNTS as user1	Enter the user- or group-specific directory below the telnet directory.
5	>	telnetProd.bat	Launching script to connect to Prod	This is the telnet script generated by MFG/UTIL.

Table 7.2 illustrates sample script lines for a UNIX system. Record your log-in script values in Table 7.3.

Table 7.2
Sample UNIX Script Values

Seq	Script Pattern	Script Value	Script Status	Notes
1	login:	user1	Logging In	This ID must correspond to a local user account on the UNIX server.
2	password:	pswd	Supply Password	Enter the password assigned to the user ID specified in step 1.

Seq	Script Pattern	Script Value	Script Status	Notes
3	$	cd /qad/eB2/	Accessing DBServer Dir	Change directories to the MFG/PRO database server administration directory. Depending on how you set up your UNIX telnet environment, the response you enter varies. For example, for maximum security, you might enter a command to change to a restricted shell before changing to the directory containing the telnet script. See “UNIX Telnet Environment Security Issues” on page 138.
4	$	./telnet.Prod	Launching script to connect to Prod	Launch the telnet connection script. See “Generate Telnet Log-in and Connection Scripts” on page 53.

Use Table 7.3 to record your telnet server log-in information.

Table 7.3
Telnet Log-In Script Information

Seq	Script Pattern	Script Value	Script Status
1			
2			
3			
4			
5			

Your completed table for the telnet script should be similar to Table 7.4.

Table 7.4
Sample Telnet Log-In Script

Seq	Script Pattern	Script Value	Script Status
1	Login	*UserID*	Logging in
2	Password	*UserPassword*	Supply the password
3	`$`	`cd /qad/eB2/`	Accessing DBServer Dir
4	`$`	`./telnet.Prod`	Launching Prod telnet connect script

Verify the Log-In Sequence

To verify the log-in sequence, from a remote machine attempt to log in to the telnet server using the log-in sequence you configured in MFG/PRO. You should receive a blank telnet screen after the telnet connection script is launched.

Set Up SSH for QAD .NET UI

Under the .NET UI, users can choose to display programs in a character rather than .NET format.

Note Currently, this is only supported for US English and UNIX servers.

To support the character screens, you can use SSH rather than standard telnet. SSH (or Secure SHell) is a protocol for creating a secure connection between a .NET UI client and the server. The safeguards provided by SSH include:

- User authentication and key exchange.
- Negotiate encryption, compression, and message integrity verification.
- All data is encrypted using a sysmetric key algorithm and verified against a keyed-hash message authentication code (HMAC).

Annotation 11 If SSH is enabled for QAD .NET UI, you will not be able to use telnet for Desktop sessions because the Desktop UI does not support SSH. You can separate telnet and SSH users by creating settings in User Option Telnet Maintenance (36.20.10.3). Remove this note complete, or keep section about separating telnet and SSH users?

To set up SSH, follow these steps:

1. Download `granados200.tar.gz` from the following link to a temporary directory:

   `http://www.routrek.co.jp/en/product/varaterm`

   Note This file is not included with QAD software because of exportation laws regarding encryption.

2. Extract `Routrek.granados.dll` from the archive using `gunzip` and `tar`. You must use version 2.0.0.0 of the DLL signed by Routrek Networks or SSH will not work.

   The `gunzip` and `tar` programs are UNIX utilities. The commands follow. Some WinZip versions support `*.tar.gz` files.

   a. Create `granados200.tar`:

      ```
      gunzip granados200.tar.gz
      ```

   b. Extract the tar archive:

      ```
      tar -xf granados200.tar
      ```

   The DLL file will be located in the temporary directory's `/bin` directory.

3. Copy the DLL to *`TomcatInstallDir`*`/webapps/qadhome`. This is the default location.

See "Sample XML Configuration File" on page 121.

4. Open *`qaduiConfig`*`.xml` in *`TomcatInstallDir`*`/webapps/qadhome/client/configs` where *`qaduiConfig`* is your QAD UI configuration name.

   Note Starting with the QAD .NET UI 2.5, the client session file defines client session characteristics of the QAD .NET UI. By default, the file is located in `TomcatInstallDir/webapps/qadhome/configurations/default/client-session.xml`. In the following step, the elements you must edit in `client-session.xml` include `<SshProviderUrl>` and `<TerminalProtocol>`.

5. Make the following changes:

   a. Update the location of the DLL file:

      ```
      <add key="SshProviderUrl" value="${HomeServer}/ Routrek.granados.dll" />
      ```

      `${HomeServer}` is automatically replaced with the HomeServer configuration value defined at installation time. All `${<variable>}` references are resolved against other configuration keys. If no key is found, then the string is not replaced.

b Change the terminal protocol entry from telnet to either SSH1 or SSH2. SSH2 is preferred because it is more secure.

```
<add key="TerminalProtocol" value="SSH2" />
```

6 Save the configuration file.

7 Repeat these steps for `default.xml` in the same directory.

8 In User Option Telnet Maintenance (36.20.10.3), change the port from 23 to 22. This is the default port for SSH.

Chapter 8

Using Connection Manager

This chapter provides instructions for using the Connection Manager to configure connection parameters and manage connection sessions.

Overview

The Connection Manager controls the pool of telnet sessions used for maintenance programs, lookups, reports, and inquiries. The Connection Manager lets you:

- Close, start, and reset Connection Manager.
- Update configuration settings.
- View a log file of Connection Manager actions.
- Monitor connections in the connection pool.
- Monitor users and close user sessions, if needed.

Once Connection Manager is configured, you can start a .NET UI client session.

QAD UI Administration

After you have built your UI configuration and completed the post-deployment steps, you can begin using the administration page.

To start the Administration page, point your browser to:

```
http://TomcatServer/qaduiConfig/admin/
```

The *TomcatServer* is typically *TomcatHost:TomcatPort* as in `prcx06.qad.com:8080`. The *qaduiConfig* is the name of the configuration.

See “Define QAD UI Security Groups” on page 32.

You are prompted to enter your user ID and password. You must have a valid user ID defined in User Maintenance (36.3.18; 36.3.1 in eB2.1) and be a member of the QAD UI Administration user group, by default `qadadmin`, to access this function.

Fig. 8.1
Log-In Page

Once you log in, run the Connection Manager, which is a program in the Administration program group. Connection Manager options are described in “Administering Connection Manager” on page 97.

Other Administration Programs

This section describes other QAD .NET UI administration programs.

Tomcat Manager

Tomcat Manager is a program in the Administration program group, and is used to manage Web applications. It lets you list, install, reload, deploy, and remove existing Web applications, without having to shut down and restart Tomcat.

> ***Note*** You must have the correct access set up using Tomcat administrative functions to use this link.

Process Admin

Use the Process Admin program to define parameters for the Process Editor. This feature is described in *User Guide: Introduction to QAD Enterprise Applications*.

Configurable Screens

Configurable screens is a design tool for configuring selected screens. The design tool lets users in a designated UI design group disable fields for input, hide fields, set a predefined default value for a field, mark a field as required, add fields and frames, and indicate that navigation through a frame should occur automatically. This feature is described in *Administration Guide: QAD .NET UI*.

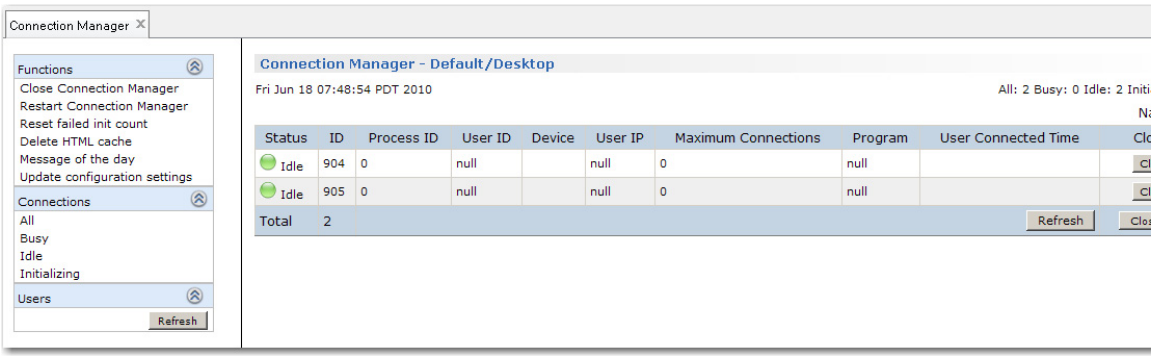
Process Editor

The Process Editor is used for creating and editing process maps. Use this function after making parameter updates in the Process Admin pages. Only users who are members of the `pronav` user group can access the Process Editor. The Process Editor is described in *User Guide: Introduction to QAD Enterprise Applications*.

Administering Connection Manager

Connection Manager is composed of several administrative functions and views that help you manage the pool of telnet connections used to run the QAD .NET UI. The initial Connection Manager page displays options with three menu headings: Functions, Connections, and Users.

Fig. 8.2
Connection Manager



Functions

Close Connection Manager. Terminates all active connections. Any data being processed by active UI sessions is lost. Any processes begun by active sessions are terminated.

Restart Connection Manager. Shuts down and restarts Connection Manager. This option has the same effect as Close Connection Manager, but also restarts it after complete shutdown.

Reset failed init count. Resets the initialization failed counter. The system maintains a count of the number of times Connection Manager unsuccessfully attempts to start a session. When this counter reaches the maximum number, as indicated in Maximum Failures on the Connection Manager configuration page, it stops further automatic attempts to start the session.

This number is automatically reset when a successful connection is made.

Delete HTML cache. This option clears all cached maintenance screens. Removing the cache is required whenever updates are made to screen elements—such as adding lookups to a field—to ensure that the new screen information is read by the system.

Message of the day. Select this option to set up a brief message that displays in the header of the main page each time a user logs in. A screen displays where you can enter up to 80 characters of message text. Use the message to alert users of upcoming system maintenance, other updates, or other business needs. You can specify how long the message displays.

Update configuration settings. Displays the Configuration Settings Update page (see Figure 8.3), which is used to set up and configure the Connection Manager options.

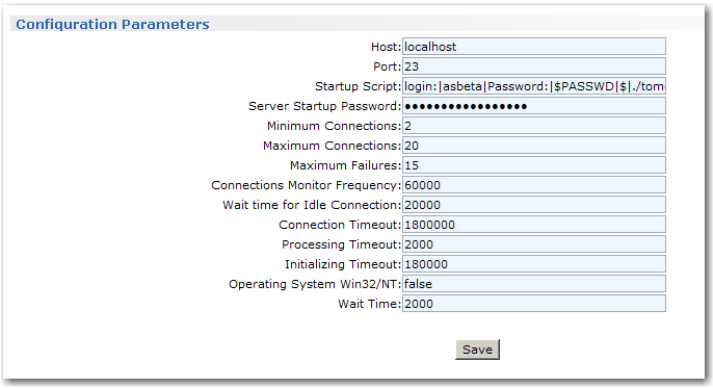
Update Configuration Settings

Use the following steps to update configuration settings as needed. The values that appear were generated during the Connection Manager and telnet setup you completed during “Generate Telnet Log-in and Connection Scripts” on page 53 for single-tier, and under the same heading on page 78 for multi-tier. Use the Connection Manager configuration screen to make minor modifications.

1 Run Connection Manager.

2 Select Functions|Update Configuration Settings. The Configuration Parameters page displays.

Fig. 8.3
Configuration Parameters



3 Review and edit the values as needed for your environment.

Host. The machine name or IP address of the telnet server.

Port. The port number for the telnet server. This is normally set to 23.

See “Generate Telnet Log-in and Connection Scripts” on page 53 and page 78.

Startup Script. The telnet server log-in prompts and the responses to these prompts, separated with the pipe symbol (|). The standard order is:

```
loginPrompt|userid|passwordPrompt|$PASSWD|osPrompt|cd UIConfigDir|osPrompt|
startScript
```

For example:

```
login:|mfg|Password:|$PASSWD|$|cd /dr02/mfgpro/eb2.1sp3/ uiconfig01|$|
./connmgr.Production
```

Server Startup Password. The password for the telnet session startup script. It is encrypted on entry.

Minimum Connections. The minimum number of open connections that the system should maintain.

Maximum Connections. The maximum number of open connections that the system should allow.

Maximum Failures. Number of times Connection Manager should attempt to restart an unsuccessful connection.

Connections Monitor Frequency. Number of milliseconds between checks for all connections.

Wait Time for Idle Connection. The maximum wait time in milliseconds for a requested connection from the Connection Manager. If the maximum number of connections has been reached, or if new connections are in the initializing state, the wait time may expire. The default value is 20000 (20 seconds).

Connection Timeout. Number of milliseconds an HTML session can remain inactive before Connection Manager closes it. The default value is 1800000 (30 minutes).

Note The session timeout value defined in User Option Telnet Maintenance (36.20.10.3) determines when an inactive user is logged out of a QAD .NET UI session.

Processing Timeout. How often, in milliseconds, a locked or busy screen is pinged (that is, checked for changes). A locked or busy screen is considered to be in processing mode, which is when the underlying program is processing and is not prompting for user input. This setting does not apply to reports. The default value is 2000 (2 seconds).

Initializing Timeout. Number of milliseconds Connection Manager will wait for a telnet session to successfully initialize. The default value is 180000 (3 minutes).

Operating System Win32/NT. True if the Progress telnet sessions are executing on a computer with a Windows operating system.

Wait Time. Number of milliseconds between checks of an HTML session while it runs reports. The default value is 2000 (2 seconds).

4 Choose Save to save your changes.

5 Choose Restart Connection Manager. If prompted to continue, choose OK.

6 Once Connection Manager restarts, click Connections to monitor .NET UI sessions.

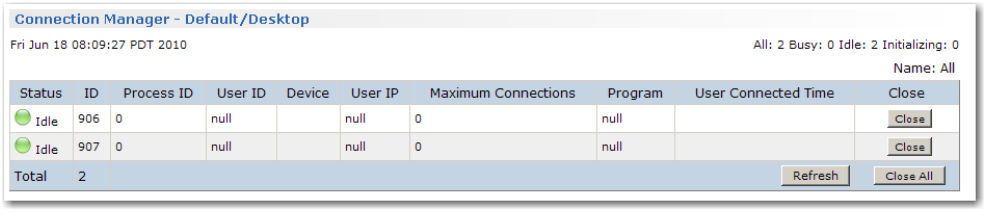
Monitor Connections

Choose Connections to view the pool of active sessions. Each session has one of the following statuses:

- Initializing. The session is starting and is not available for use.
- Idle. The session is active and available for the next user request.
- Busy. The session is executing a user request.
- Pause. The session is waiting for a response from the user; for example, the user may need to press the spacebar to continue.
- Processing. The session is actively updating the Progress database and database records are locked.
- Force Disconnect. This is a temporary state that occurs when the administrator closes an initializing session.
- Disconnected. This is a temporary state that occurs when idle sessions are closed.

You can choose to view connections by status or view all connections. If you choose the Busy status, any sessions that have the Pause or Processing status also display. You must choose All to see sessions that have Disconnect statuses. These statuses do not occur in normal operations. Figure 8.4 illustrates the screen for monitoring connections.

Fig. 8.4
Monitoring Connections



Use Refresh to update the display. Choose Close to close unneeded sessions.

Monitor Users

Connection Manager's Users menu displays a list of currently logged in users.

- Click a User ID to see information related to that user, including the following: Status, ID, Process ID, User ID, Device, User IP, Maximum Connections, Program, and User Connected Time.
- Click Refresh to update the display.
- Click Close to close a user session. This might be needed if a user has locked a database record and left their session running.

Chapter 9

Starting UI Clients

Use this chapter to install and configure the QAD .NET UI clients.

Overview

This chapter provides the prerequisites and steps to launch the QAD .NET UI.

Start a .NET UI Client

Important Be sure to read the *QAD .NET UI Release Notes* for your version of the QAD .NET UI. If using QAD Enterprise Applications 2007 or 2007.1, be sure to refer to *User Guide: QAD .NET User Interface*. If using QAD 2008 Standard, be sure to refer to *User Guide: QAD User Interfaces*.

Refer to *User Guide: Introduction to Enterprise Applications* for the latest Standard and Enterprise Edition features and functions.

Prerequisites

Prior to installing the QAD .NET UI client, make sure the following steps have been completed.

- Tomcat is installed and running on the QAD .NET UI server.
- MFG/PRO eB2.1, QAD Enterprise Applications 2007, 2007.1, or 2008 is installed and correctly configured.

Note When installing the QAD .NET UI client on Windows Vista, all users must install it as an Administrator user. In Windows Vista, a user has to right-click the Internet Explorer icon, and select Run as Administrator. This is a Windows Vista requirement.

First-Time Start-Up of the .NET UI

To install the QAD .NET UI client and begin using it, complete the following steps:

1. Open an Internet Explorer 6.0+ browser.
2. Enter the following URL and press Enter:
   `http://TomcatHost:TomcatPort/qadhome/client/setup.html`.

   Note If connecting to a URL with secure HTTP (`https://`), be aware that Microsoft introduced changes in how Internet Explorer 7 (IE7) connects with secure HTTP compared to how Internet Explorer 6 (IE6) connects with secure HTTP. For your default browser and the QAD .NET UI to display data, your default browser must be configured to accept the secure HTTP security certificate. The steps for accepting a security certificate vary depending on whether you are using IE6 or IE7. Please contact your system administrator for details.
3. The installation screen displays and the installation starts automatically. If it does not, click the Install link on the displayed page.
4. If you have already installed the QAD .NET UI, the installation program prompts you to modify, repair, or remove it.

   Note If you have previously installed the QAD .NET UI client and want find out the URL that you installed it from, choose Help|View Configuration and enter homeserver in the Search field.

5 If the install detects that you do not have Macrovision installed on your system, you are asked to confirm this portion of the installation.

6 Choose Install to continue.

7 InstallShield starts and displays a security warning. To continue the install, select I understand the security risk; then choose Next.

8 InstallShield sets up the installation environment.

9 On completion, you are asked whether you want launch icons on your desktop and under Programs on the Start menu. Select the options you want and choose Install.

10 The installation begins. A progress screen displays.

11 When the client installation is complete, a confirmation screen displays. Choose Finish.

12 If the client installation includes QAD .NET UI plug-in updates, you will be prompted whether to accept the updates. Choose OK.

13 To launch the client, select the QAD Applications icon or menu item under Programs on the Start menu.

The login screen is now displayed.

Setting System-Specific Desktop Shortcut

You can define multiple Windows Desktop icons for accessing different systems.

1 Copy and paste the QAD Applications icon.

2 Place your cursor over the new icon, right-click, and select Properties.

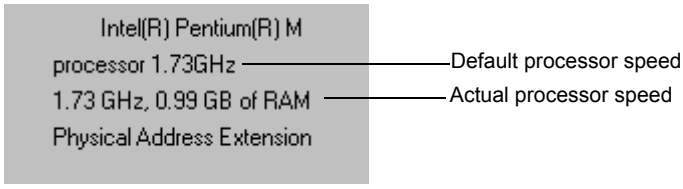
3 In the Target: field, add the command line option `-mfgsys` *`SystemName`*, where *`SystemName`* is the name of an installed system.

If the specified system does not exist or if the command line option is included but no system is specified, QAD .NET UI displays a list of available systems when a user logs in.

Setting Client Power

The power setting on the client machine may cause the CPU to work at a reduced speed, which affects the performance of the QAD .NET UI. To correct this issue:

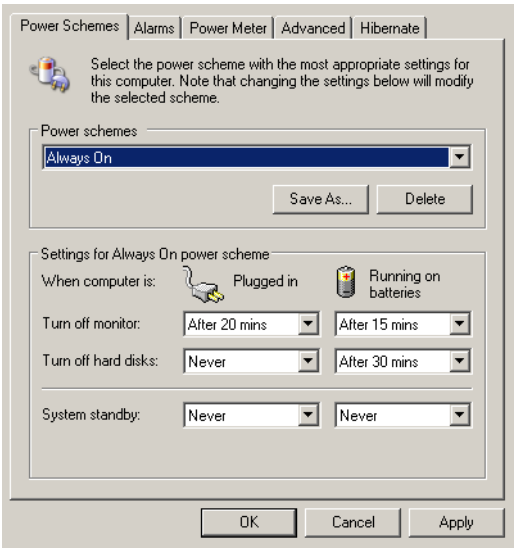
1 Determine your CPU speed. Go to Start|Control Panel|System and view the information on the General tab.



2 If your machine is running below the default speed—the first number is much greater than the second—go to Start|Control Panel|Power Options.

3 In the Power Schemes tab, select Always On in the drop-down list.

Fig. 9.1
Power Schemes Settings



4 Close your Control Panel|System window and then reopen it from the Start menu to refresh the CPU reading.

Displaying QAD .NET UI Help

QAD .NET UI help is currently English-only, although QAD applications help is available in multiple languages.

Typically, pressing F1 or selecting Help from the QAD .NET UI window displays the help for the field and program in which the cursor is placed.

QAD Assist

QAD Assist is a QAD .NET UI application that hosts user guide and related content for the QAD .NET UI and for QAD Enterprise Applications. The function was introduced in QAD .NET UI 2.8 and features a search engine you can use to search across user guides, field help, program help, and process maps. QAD Assist is described in detail in *User Guide: Introduction to Enterprise Applications*.

QAD Assist requires some configuration for Standard Edition. This is described in “Installing QAD Assist” on page 125.

QAD Guide Me

The QAD Guide Me feature was introduced in QAD .NET UI 2.9 and provides immediate mouse-over descriptions of fields in both component and non-component programs. The feature is disabled by default. You can enable it and control the length of time it takes to display the field description, as well as the length of time the description is displayed on screen, using Guide Me parameters in the `client-session.xml` file.

Chapter 10

Implementing Multiple Languages

This chapter includes information to help you install the Desktop UI into a multiple-language environment.

Multiple-Language Overview

For details on setting up multiple languages, see the MFG/PRO installation guide for your system.

MFG/PRO is available in a number of different languages. Each language is identified by a two-letter language code. You can use this code to associate different language displays with different MFG/PRO users in a multiple-language environment.

In the .NET UI, language affects the fields and menus for maintenance screens, telnet screens, browses, report output, and administrative functions such as Connection Manager.

Information required to establish the user locale is retrieved from two external data files: `locale.dat` and `encoding.dat`. Both files are located in the `mfgpro` directory on the release CD. After installation, they are located in *`MFGPROInstallDir`*.

The data in these files is used in conjunction with details about the user defined in User Maintenance (36.3.18; 36.3.1 in eB2.1).

Languages in the QAD .NET UI

There are no additional install steps to run the QAD .NET UI under a language other than English. You select the language in the log-in screen after clicking the Options link.

However, the menus and labels are in the language associated with the user ID. For example, to run the QAD .NET UI in German, set up users with the language set to German in QAD 2008 Standard—assuming you have installed QAD 2008 Standard with German as a language. Then, when a German user logs in to QAD .NET UI, the user selects German on the Options link and enters user credentials. The QAD .NET UI then display German labels.

Locale Data

Information on language, country code, and, optionally, the variant is maintained in the `locale.dat` file, along with date and numeric formats. The system uses this information to set the locale formatting (the date and number formats) for an QAD 2008 Standard user.

When a user logs in, the system validates the log-in identifier in User Maintenance and determines the associated QAD (MFG/PRO) language and country code. It uses this information to access the corresponding record in `locale.dat`.

The system uses the date and number format information and ISO language code in `locale.dat` to set the user's locale formats in character and in the QAD .NET UI.

If locale information cannot be determined at log-in, US settings are used by default.

Setting Different Locale Formats

Three values associated with a user determine locale formatting: language, country code, and variant.

1 Define country codes and corresponding ISO country code values in Country Code Maintenance (2.14.1):

   a Specify the MFG/PRO country code (up to three letters) in the Country Code field.

   b Specify the corresponding ISO code in the Alternate Code field.

   ISO country codes are already defined in Generalized Codes Maintenance (36.2.13). If a code is not available in generalized codes, you can add it. Valid ISO country codes can be found on the following Web site:

   http://www.chemie.fu-berlin.de/diverse/doc/ISO_3166.html

2 Create or update the user profile in User Maintenance (36.3.18; 36.3.1 in eB2.1).

   a Specify the user's language in the Language field.

   b Specify the user's country in the Country Code field. The system uses the alternate code to determine the ISO country code value.

   c Optionally, specify a value in the Variant field.

3 Verify that the required locale entry is available in `locale.dat` file.

4 If necessary, use any standard text editor to edit the file and add or update entries, using the proper format; then save the file.

Locale Record Format

The format of records in `locale.dat` is as follows:

```
US,en,US,,mdy,American
```

Where:

- US is the MFG/PRO language code.
- en is the ISO language code.
- US is ISO country code.
- Optional variant is blank (indicated by `,,`).
- mdy (month/day/year) is the date format.
- American is the numeric format (period as the decimal separator; comma as the thousand separator).

Two number formats are defined in Progress: American and European. Decimal and thousand separators are defined differently for each numeric format type. Periods (.) and commas (,) are used to separate decimal places and thousand places, depending on which format is specified. For example, in the US format, a period separates decimal, while a comma separates thousands as in 10,000.33. A user with the European numeric format will see the number as 10.000,33.

Language Codes and Country Codes

Table 10.1 lists MFG/PRO language codes and the ISO language and country codes to which they map.

ISO language codes are lowercase and country codes are uppercase. Java is case sensitive and expects the codes in the ISO format.

Table 10.1
MFG/PRO Language Codes

Language	MFG/PRO Language Code	ISO Language Code	ISO Country Code	Country
Chinese (Traditional)	TW	zh	TW	Taiwan
Chinese (Simplified)	CH	zh	CN	China
Czech	CZ	cs	CZ	Czech Republic
Dutch	DU	nl	NL	Netherlands
French	FR	fr	FR	France
German	GE	de	DE	Germany
Italian	IT	it	IT	Italy
Japanese	JP	ja	JP	Japan
Portuguese	PO	pt	BR	Brazil
Russian	RU	ru	RU	Russian
Spanish (Castilian)	CS	es	ES	Spain
Spanish (Latin)	LS	es	MX	Mexico
US English	US	en	US GB AU	United States Great Britain Australia

Encoding Data

The system gets information on encoding from the `encoding.dat` file supplied with QAD 2008 Standard. This file includes code page information for Progress, Java, HTML/XML, and XSL. The code page values vary for Windows and UNIX operating systems and are tailored by the installation scripts.

The system uses encoding information to correctly display language-specific characters in the user interface. Typically, system administrators do not need to add or edit information in `encoding.dat`. All required information is in the file.

The QAD .NET UI also uses the `dtencode.dat` file supplied with the QAD UI media and located in `com/mfgpro` and `com/qad/mfgpro`. This file is tailored by the MFG/UTIL scripts when necessary, and should not require manual updates.

Supported Code Pages

Progress manages the character set used for data storage through code page settings that apply to the entire database. For any one database, only a limited set of characters is available, as defined by the code page. This means that some combinations of language data cannot be stored together. Currently, the two supported multiple-language scenarios are:

1 All languages in the database share the same code page.

   This is true for English, French, German, Spanish, and other languages that share the ISO-8859-1 code page, or for Russian, Ukranian, and Romanian, which share the 1251 code page, as well as for other shared code page sets.

2 The languages in the database have compatible code pages.

   This scenario is only true of English with other code pages. Because English is a subset of all other code pages, English can be combined, for example, in a single database with Japanese, or with Polish and Romanian, or with Turkish.

Multiple languages with incompatible code pages in the same database are not supported.

Multiple-language display is complicated by the various technology layers involved in the UI architecture. Data stored in the Progress database can be passed through Java, HTML, and XML before being viewed by the user in a Web browser. In some cases the data transformation also requires code page mapping.

To ensure that data can be passed correctly without corruption, QAD has defined appropriate code page configurations based on the Progress code page in use. This adds an additional constraint for UI installations. Your database must use the Progress code page that QAD supports. Table 10.2 lists the code page supported for each supported language.

If your database is using a code page other than the one specified in Table 10.2, additional changes may be required to ensure that data displays correctly. You should contact QAD Support for help with these changes.

If your database code page is the standard code page for your language, you should be able to install the UI using MFG/UTIL without any changes.

Table 10.2
Supported Progress Code Pages

Language	MFG/PRO Language Code	Database Code Page
Bulgarian	BU	1251
Chinese (Traditional)	TW	cp950
Chinese (Simplified)	CH	cp936
Czech	CZ	1250
Danish	DA	iso8859-1
Dutch	DU	iso8859-1
French	FR	iso8859-1
German	GE	iso8859-1
Hungarian	HU	1250
Italian	IT	iso8859-1
Japanese	JP	shift-jis

Language	MFG/PRO Language Code	Database Code Page
Korean	KO	cp949
Lithuanian	LT	1257
Norwegian	NO	iso8859-1
Polish	PL	1250
Portuguese	PO	iso8859-1
Romanian	RO	1250
Russian	RU	1251
Spanish (Castilian)	CS	iso8859-1
Spanish (Latin)	LS	iso8859-1
Swedish	SW	iso8859-1
Turkish	TU	1254
Ukranian	UA	1251
US English	US	iso8859-1

Note To determine your database code page, enter the following command in the Progress Editor:

```
message session:cpinternal
```

Information for mapping code page values from Progress settings to those used by different interface layers is maintained in `encoding.dat`.

Use MFG/UTIL to Install Multiple Languages

You can use the advanced build options in MFG/UTIL to build multiple-language versions of a .NET UI configuration. To do this, you first generate the UI configuration for your default language. Then you rerun the process and choose the advanced build options to specify your second language.

Building Languages with Shared Code Pages

The process is straightforward when all languages share the same code page as the database. This would be the case, for example, if you are installing French, German, and English together. For this scenario, follow these steps:

1. Launch MFG/UTIL and choose UI|Configure UI Environment Parameters.
2. Specify the parameters for the first language you are going to build. When you are building the system for languages with shared code pages, you can build them in any order.
3. After specifying parameters, choose UI|Build UI. Review the settings and complete the build for the first language.
4. Now choose UI|Build UI Configuration for the second language. Choose the same QAD UI configuration to build.
5. Choose Advanced in the Verify and Build screen. In the Advanced Build Options screen, choose the second MFG/PRO language code from the drop-down list. The other language-related fields should default appropriately.

6 Choose Build to complete the process.

7 Repeat these steps for each additional language with the same code page.

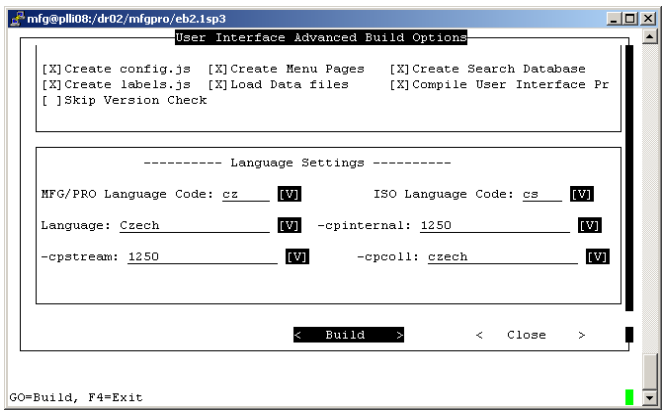
Building Languages with Compatible Code Pages

Because English is a subset of all other code pages, English can be combined in a single database with any other language. In this scenario, you should build the Desktop UI for the languages that share the database code page first; then build the Desktop UI for English. Follow these steps:

1 Launch MFG/UTIL and choose UI|Configure UI Environment Parameters. Build the languages with the same code page as the database first.

2 After specifying parameters, choose UI|Build UI Configuration. Review the settings and complete the build for the first language. Default language settings should not need to be changed. In Figure 10.1, the language being built is Czech.

Fig. 10.1
Advanced Build Options for Czech

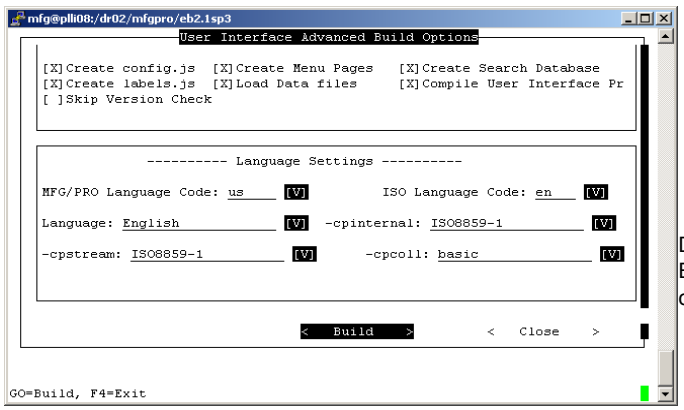


Default values for Czech

3 Now choose UI|Build QAD UI for the English language. Choose the same QAD UI configuration to build.

4 Choose Advanced in the Verify and Build screen. In the Advanced Build Options screen, choose US as the MFG/PRO language code from the drop-down list. English values default to the other language-related fields and must be changed.

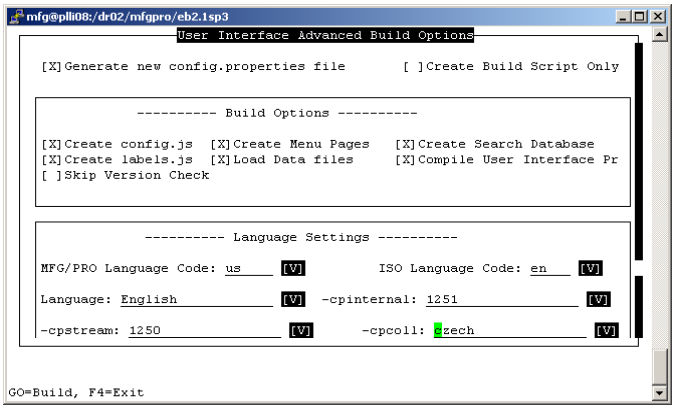
Fig. 10.2
Default Values for English



Default values for English must be changed.

5 Change the value of the marked fields to match values of the database code page. In this example, these were default values for Czech.

Fig. 10.3
Building English with a Czech Database



6 Choose Build to complete the process.

Configure Non-English Languages

After installing the UI files, you may need to perform a few additional configuration tasks, depending on your environment:

- Update fonts for non-English languages. Check your operating system documentation for font requirements for the language you are implementing.
- Update the Process Editor to access translated process maps.
- If you are not using the latest service pack for your system, you may need to update language-related data. See the QAD Desktop Release Notes for details about language-related changes.
- If your version of Progress was installed with a default language that uses a different code page than your database, make sure you include the proper language-related parameters in the startup scripts. See the Progress documentation for details.

Important Depending on the languages supported by the Windows operating system on the client PC, the typical install option for JRE may not install components required to support international character sets. To avoid this, the JRE must be installed by choosing the custom setup type and selecting the additional language support option. To verify support for non-European languages, check for the existence of the following file:

```
JavaInstallDir/jre/lib/charsets.jar
```

Update Fonts for Non-English Languages

The `font.properties` file may need to be updated to access locale-specific fonts, such as the Japanese font, on an English environment platform. A number of alternate `font.properties` files are provided with the JRE, located by default in:

```
C:\Program Files\Java\JRE1.5.0_06\lib
```

See Table 10.1 on page 108 for a list of language codes.

These files are identified by locale-specific extensions such as ko for Korean, ja for Japanese, and zh for Chinese.

The simplest way to make these fonts available on a client machine is to back up the original `font.properties` file and rename the locale-specific file to `font.properties`.

For example, to access Chinese fonts, rename:

```
font.properties.zh
```

To:

```
font.properties
```

For more information about the `font.properties` file and how to modify it in an international environment, review the documentation on the Sun Microsystems Java Web site:

http://java.sun.com

Translated Process Maps

During the UI installation, all existing process maps remain in the default `/WEB-INF/pronav/xml` directory. Whenever you configure the Desktop UI environment for a new language, existing process maps located in this directory are automatically translated to the new language. The translated process maps are placed in a two-letter subdirectory in the `/WEB-INF/pronav/xml` directory. Your environment's two-letter ISO language code is used as the directory name.

To use the translated process maps, update the XML Directory field in the Process Admin|Context Parameters page after completing the UI installation. Use these steps to make the required change:

1. Launch a Web browser and point to the Desktop Administration HTML page. This page is typically found at:

   TomcatServer`/`*qaduiConfig*`/admin/`

2. Click the Process Admin link.
3. Click Context Parameters.
4. Edit the XML Directory value to point to the two-letter subdirectory created during installation. You should only need to add `/`*xx* to the existing value, where *xx* is a two-letter ISO language code.
5. Click Save.
6. Click the Refresh link to update the server.

Section 3

System Administration and Reference

This section provides QAD UI system administration and technical reference information.

Chapter 11

Administering the Desktop UI

This chapter contains Desktop system administration and other reference information.

Install a New UI Release

QAD highly recommends you deploy only the components included with the latest QAD UI release. The latest release always contains the most recent component versions, patches, and performance improvements. Review Release Notes on the QAD UI media to see if any updated Desktop system components have been included.

Consider the following items before you install a new release:

- Some releases cannot be installed directly into directories that contain any components from a previous UI version. Install QAD .NET UI into new directories that do not contain components from a previous release.

  Always check the release notes to see if this condition applies to the release you plan to implement.
- The installation process installs updated code; therefore, the installation directory must be placed at the front of all your system PROPATHs.
- The latest QAD-optimized Tomcat version is always included on the QAD UI media. Always install the new version. This can be done manually or automatically using the Perl installation tools.
- If the client Java version has not been updated, initial client startup after setting up the new QAD .NET UI might require a Java download.
- The latest QAD-tested version of the SVG plug-in is always included on the QAD UI media. Process Editor client machines need to download the new plug-in.
- The Tomcat server cache must be cleared after installing a new release. You do not have to manually clear the client cache. When you install a new release, new files are automatically cached by the client. The old files are deleted automatically based on client browser history settings.
- After a release upgrade, client machine performance may be slightly affected while new HTML pages are generated on the server and cached locally on each client. This only affects the initial client sessions while new pages are being cached.
- Process maps are located in `/WEB-INF/pronav/ xml` in the Tomcat `webapps` directory. If you are using a release that requires a complete new installation, copy the existing process maps from the previous system to the new system directory structure.

Clear the Client Cache

Depending on your client machine configuration, a copy of all the Web pages displayed during a session are automatically stored on the local hard drive. This is called the client cache. These files are normally deleted after a time period determined in a history setting for Internet Explorer.

After installing a new build, new client files are generated and served to the client machines. Under normal operating conditions, you do not need to manually clear the client cache each time a new release is installed.

There are, however, times when you do need to manually clear the client cache, especially when initially setting up a environment. These are when:

- The menu system is modified and rebuilt.
- A browse is attached to or detached from a field.

- During initial system configuration, for testing and troubleshooting purposes.

Use the following instructions to manually clear a client machine cache:

1. Start Internet Explorer on the client machine.
2. Choose Tools|Internet Options and click the General tab.
3. Click the Delete Files button in the Temporary Internet Files section.

Install a Service Pack

Installing a service pack can affect your system in two ways:

- The service pack may include software fixes that correct issues in the .NET UI, since some issues are caused by problems in the underlying programs, not the .NET UI programs.
- The service pack installation media may include a different version of MFG/UTIL than the version delivered with your release.

Nothing beyond the standard service pack install is required to take advantage of the changes for the first case. The second case may result in an older version of MFG/UTIL being installed over the one installed with QAD .NET UI, which may remove the menu or updates.

After installing a service pack that contains older .NET UI-specific MFG/UTIL updates, use the .NET UI installation media to reinstall MFG/UTIL. Use the following steps to reinstall this code:

1. Start the UI install and select the configuration that you have already installed.
2. Select the Install MFG/PRO – Progress QAD UI Files Menu option.
3. In the next menu, select Option 2, Update Additional MFG/UTIL Installations.
4. Specify the directory where you want to update MFG/UTIL. This is typically the same directory where you originally installed MFG/UTIL.

Use the same process to update additional MFG/UTIL installs.

Administer QAD .NET UI

For the most current information on administering the QAD .NET UI, be sure to read the *QAD .NET UI Release Notes* for your version of the QAD .NET UI. If using QAD Enterprise Applications 2007 or 2007.1, be sure to refer to *User Guide: QAD .NET User Interface*. If using QAD Enterprise Applications 2008, be sure to refer to *User Guide: QAD User Interfaces*. Starting with the QAD .NET UI version 2.5, the default configuration files were consolidated as XML files in a *`TomcatInstallDir`*`/webapps/qadhome/configurations/default` directory.

For more information on the configurations directory, see the user guide for your version of the QAD .NET UI.

Other administration functions are described in *Administration Guide: QAD .NET UI*.

Set Browse Record Thresholds

See “Sample XML Configuration File” on page 121.

By default, the .NET UI sets a limit on browse records a given user can download, navigate in page mode, select for printing, or select for export to Excel. These defaults can be modified by the administrator.

These settings improve performance when browsing against large databases. For smaller databases or on highly performant systems, you may want to increase these values. All four parameters are configured on the server in *qaduiConfig*.xml, located on the Web server in *TomcatInstallDir*\webapps\qadhome\client\ plugins\mfgpro.

MaximumBrowseRecordsToCount. The default value is 50,000. This limits the total record count allowed in page navigation mode. If the count is above this threshold, the total count is not displayed in the upper right of the browse, and the To-From records display vanishes if the user clicks to go to the last page.

MaximumBrowseRecordsToDownload. The default value is 50,000. This limits the number of records downloaded to the client when a user requests all records. If the number of records exceeds this limit, the download is aborted and an error message displays.

BrowseRecordsForPrintWarning. The default value is 10,000. A browse print size warning message displays if a user selects more than this number of records for a print job.

BrowseRecordsForExcelWarning. The default value is 10,000. A browse print size warning message displays if a user exports more than this number of records.

Excluding Menus in .NET UI

To exclude specific menu-level programs from the QAD .NET UI interface, you can add them to an exclusion file.

Important Do not remove any menus from the file as shipped.

1. Open menuexclude.dat located in *MFGPROInstallDir*/*qaduiConfig*/com/qad/shell/interface.
2. Enter the menu level programs you want to exclude from the .NET UI interface. An example is shown in the file.
3. Save the file.

Set Session Configuration

The administrator can set the number of forms each user can open in a single session. Limit the values to a reasonable number for your users; setting these values too high will impact performance.

This parameter is configured on the server in *qaduiConfig*.xml, located on the Web server in:

TomcatInstallDir\webapps\qadhome\client\ plugins\mfgpro

```
<add key="MaximumDesktopsPerWorkspace" value="5" />
```

Note You can ignore the three other maximum settings; they currently have no effect.

```
<add key="MaximumBrowsesPerWorkspace" value="6" />
<add key="MaximumWebspeedsPerWorkspace" value="3" />
<add key="MaximumTelnetsPerWorkspace" value="5" />
```

Sample XML Configuration File

The `default.xml.sample` file is a template tailored by MFG/UTIL to create *`qaduiConfig`*`.xml` and `default.xml`, in *`TomcatInstallDir`*`/webapps/qadhome/client/configs`.

Some of these settings can be configured; others should not be changed, because changing them may cause the system to fail.

The following table describes the settings in `default.xml.sample`, located on the Web server in:

`TomcatInstallDir``\webapps\qadhome\client\ plugins\mfgpro`

The values with dollar signs ($) are updated during installation with system-specific values.

Table 11.1
.QAD .NET UI Application Settings

Setting	Default Value	Description
RemoteConfiguration		Not used in this file, as this is already the remote configuration.
ConnectionProtocol	AppServer	This is the protocol used to connect to the AppServer; it is called AppServer and should not be changed.
ConnectionHost	$NameServerHost	The name of the machine that is running the Progress Name Server.
ConnectionPort	$NameServerPort	This is the port that the Name Server listens on; by default this is 5162, which is the Progress default.
ConnectionService	$AppServerBroker Name	The name of the Progress AppServer created for this configuration.
http	http	This is the protocol to use when launching HTML maintenance or WebSpeed programs within the QAD .NET UI. This might be changed to https for that kind of communication.
DesktopHost	$TomcatHostName	The name of the machine that Tomcat is running on.
DesktopPort	$TomcatPort	This is the port number that Tomcat listens on; by default this is 8080, and could be 8443 (if using https).
DesktopService	$DesktopWebapp Name	The name of the UI configuration directory under Tomcat webapp folder.
DesktopAPI	shell.jsp	This is the program used to run maintenance programs; it does not change.

Setting	Default Value	Description
DesktopUsercount Servlet	UsercountServlet	This is the name of the servlet used to track user log-ins; it does not change.
WebSpeedService	$WebSpeedBroker Name	The name of the Progress WebSpeed broker created for this configuration.
WebSpeedPath	cgi-bin/$WebSpeed ExecutableName/ WService	This is the relative path for WebSpeed calls; only the middle section changes. It is `cgiip.exe` on Windows machines.
WebSpeedProgram	com/qad/desktop/ interface/wsep1.p	This is the launching program for WebSpeed pieces such as Browse Maintenance; it does not change.
HelpService	mfwb01.p	This is an admin piece; it does not change.
MaximumBrowsesPer Workspace	6	Not currently used.
MaximumDesktopsPer Workspace	5	Limits the number of opened tabs a .NET UI client can open.
MaximumTelnetsPer Workspace	5	Not currently used.
MaximumWebspeeds PerWorkspace	3	Not currently used.
MaximumBrowse RecordsToCount	50000	This is the maximum number to check for (count) before a > symbol is displayed on a client.
MaximumBrowse RecordsToDownload	50000	This is the maximum number of records that a client can download.
BrowseRecordsForPrint Warning	10000	When this number of records is reached or exceeded during printing, a warning is displayed on the client.
BrowseRecordsFor ExcelWarning	10000	When this number of records is reached or exceeded during export to Excel, a warning is displayed on the client.
SshProviderUrl	${HomeServer}/ Routrek.granados.dll	This is the location for the DLL to be downloaded to the clients when they use SSH for terminal connections.
TerminalProtocol	telnet	This is the protocol used for terminal connections.
TerminalEncoding	UTF-8	This is the encoding for terminal connections; it does not change.
TerminalAuthentication	ShellUser	This is the credentials to use for terminal connections; it does not change.

Manage Menu Updates

This section includes instructions for adding custom programs, browses, and lookup browses. You must rebuild the menus and search databases in order for the new options to display in QAD >NET UI.

Changing Menus

The .NET UI system must be configured dynamically so that it can reflect the menu system in your database. If you have moved programs or made any other changes with either Menu System Maintenance (36.4.4) or Menu Substitution Maintenance (36.20.6), these changes are reflected in the menu system when it is built.

Menu substitution is managed differently in .NET UI than in the other interfaces. In other interfaces, menu substitution affects the display of:

- Browses (Off) or inquiries (On)
- Standard programs (Off) or custom programs (On)

During the build of the menu system, browses (instead of inquiries) and standard programs (instead of custom ones) are always placed on the menus.

See “Clear the Client Cache” on page 118.

If you make menu or lookup changes after the initial installation, you must rebuild the menus and the search database to reflect those changes. You must also clear the cache for all client and server machines.

Use the Advanced Build Options available from the MFG/UTIL UI|Build UI Configuration option to rebuild your menus whenever you make changes in:

- Menu System Maintenance (36.4.4)
- Menu Substitution Maintenance (36.20.6)
- Drilldown/Lookup Maintenance (36.20.1)
- Program Information Maintenance (36.3.21.1)

Adding Browses

To add a browse to the system:

1. Add the browse in Menu System Maintenance (36.4.4).
2. Add the program details in Program Information Maintenance (36.3.21.1) with these settings:
   - Set Web Logic Implemented to Yes.
   - Set Type to Blank.
3. Rebuild the menus and search databases. See the detailed steps in “Regenerating Desktop Components” on page 124.

Adding Lookup Browses

To associate a lookup with a field for use in .NET UI:

1. Associate the lookup with a program or field in Drilldown/Lookup Maintenance (36.20.1).
2. Add the program details in Program Information Maintenance (36.3.21.1) with these settings:
   - Set Web Logic Implemented to Yes.
   - Set Type to Blank.
3. You do not need to rebuild menus for lookups to be visible, but you do need to clear the client and server cache to see your changes.

Adding and Using Custom Programs

Use the following instructions to add custom programs to the menu system. If created following QAD programming constructs, your custom programs can be run in the .NET UI interface. If the QAD programming constructs were not followed, your custom program may not function properly in the .NET UI interface. Instead, run it in the embedded telnet interface.

1. Make the custom program accessible to users.

   Option 1: Set Program Execution to Yes in User Option Maintenance (36.20.10.1). This allows users to use programs even if they are not on the menu structure.

   Option 2: Add the custom program to the menu using Menu System Maintenance (36.4.4).
2. Add the program to Program Information Maintenance (36.3.21.1).

   Use these settings to run custom programs in the .NET UI interface:
   - Set Web Logic Implemented to No.
   - Set Type to .NET UI.

   Use these settings to run the custom program in the embedded telnet interface:
   - Set Web Logic Implemented to No.
   - Set Type to blank.
3. Rebuild the Desktop menus and search databases as detailed in the following section.

Regenerating Desktop Components

Use these steps to regenerate the menu pages and the search database after adding any custom programs and browses or making other menu changes, such as adding program shortcuts in Menu System Maintenance. You must regenerate the menus and search database to see your menu and program changes and use new program name shortcuts.

1. Launch MFG/UTIL. Choose UI|Build UI Configuration.
2. Select your QAD UI configuration in QAD UI Configuration Name.
3. Review the field values in the Verify and Build QAD UI screen for accuracy.
4. Choose Advanced.
5. Select the Create Search Database and Create Menu Pages options.

Note All other options should be deselected.

6 Click the Build button.

See page 118.

7 Clear the cache for all client machines.

8 Clear the Tomcat server cache.

Once the QAD UI build completes, launch a .NET UI client and test your program and menu changes.

Installing QAD Assist

QAD Assist, introduced in QAD .NET UI 2.8, provides online help for the QAD .NET UI and for QAD Enterprise Applications.

The QAD .NET UI 2.8.1 release media includes the QAD Assist content for both the QAD .NET UI and QAD Enterprise Applications 2009–Standard Edition (eB2.1 SP9).

To install QAD Assist, you need to install and configure two Web applications that are included on the release media in the /zips directory:

```
/zips/epub.war
/zips/search.war
```

Use the following steps to install QAD Assist. Note that the following steps are based on using a Linux environment with JDK 1.6.

1 Shut down Tomcat.

2 Copy `epub.war` and `search.war` from the `/zips` directory on the release media into the `/webapps` directory.

3 Start Tomcat.

4 Copy the content from the release media's `zips/content/*` directory to the `/webapps/epub` directory.

Note that the QAD Assist content for the QAD .NET UI 2.8.1 and QAD Enterprise Applications 2009—Standard Edition (eB2.1 SP9) is included on the release media in the /zips/content directory.

The QAD User Interfaces User Guide content is available in `netui-en-2.8.1.zip`.

The QAD Enterprise Applications 2009–Standard Edition content, which includes program help, field help, and user guides, is available in `.zip` files for the following languages:

- German: `erp-de-2009se-sp9.zip`
- English: `erp-en-2009se-sp9.zip`
- Spanish (Castillian): `erp-es-es-2009se-sp9.zip`
- Spanish (Latin American): `erp-es-mx-2009se-sp9.zip`
- French: `erp-fr-2009se-sp9.zip`
- Italian: `erp-it-2009se-sp9.zip`

- Japanese: `erp-ja-2009se-sp9.zip`
- Portuguese (Brazilian): `erp-pt-br-2009se-sp9.zip`
- Chinese (Simplified): `erp-zh-cn-2009se-sp9.zip`
- Chinese (Traditional): `erp-zh-tw-2009se-sp9.zip`

5 Unzip the content using the `unzip` command.

For example, for `netui-en-2.8.1.zip`, enter:

```
unzip netui-en-2.8.1.zip
```

Unzipping creates a `/netui` directory under the `/epub` directory containing the content.

You next set up the search engine index.

6 Using a web browser, access the search administration page URL.

Open `http://host:port/search`, where `host` and `port` are the host and port for the Tomcat instance under which you installed the epub and search Web applications.

7 Click the New Index button and in the field enter:

```
default
```

This creates a new index and directory in the following location:

```
search/WEB-INF/indexes/default
```

Next, you need to add index (links) content to the search engine by copying the `links_*.xml` files (which contain the default link information for the search engine index) under `/epub` to `/search/WEB-INF/indexes/default`.

You can use the following command to copy all the files that match '`links-*.xml`' into the directory.

8 Using a command line, navigate to the `/epub` directory and enter:

```
cp `find . -name "links-*.xml"` ../search/WEB-INF/indexes/default
```

The search engine automatically imports these files and removes them upon completion.

The search administration page (`http://host:port/search`) updates the information about the default index, including the number of documents in the Documents column and the total size under the Size column.

Note that you can import the same content multiple times without affecting the number of documents loaded into the search engine, although that action will increase the allocated file size. In the Actions column, you can click Optimize to reclaim the excess disk space.

To load the default index, you must stop and start the search engine for that index:

9 In the Actions column for default, click Stop.

10 In the Actions column for default, click Start.

Finally, you need to update the `client-session.xml` file.

11 Edit the `client-session.xml` file (`<HomeServer>/configurations/<SysEnvName>/client-session.xml`) so that it includes the following settings:

```
<ContentIndexServer>http://host:port/search</ContentIndexServer>
<ContentIndex>default</ContentIndex>
<ePublisherServer>http://host:port/epub</ePublisherServer>
```

The Assist feature's search is now ready for use. You can access it from the QAD .NET UI by choosing Help|Application Help or pressing the F1 key.

Note Refer to the Release Notes for the QAD .NET UI version you are installing for updates to this procedure.

Troubleshoot Connection Manager Errors

A variety of types of messages can be generated when running MFG/PRO programs through the QAD .NET UI. Application messages such as invalid user input or an error creating a record are managed by MFG/PRO.

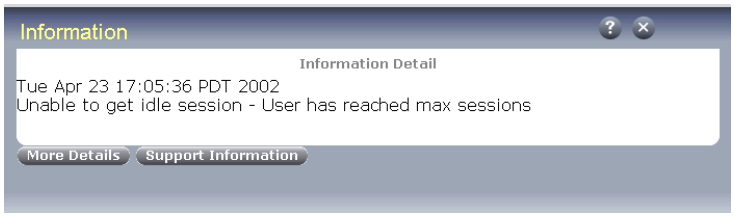
Another type of error is related to communication among the various .NET UI components. These errors may occur when a session is not available or the telnet server does not respond.

When these types of errors occur, a Java Server Page (JSP) displays system-generated error information. The error JSP displays three levels of error details. Currently only the basic information level is implemented.

Annotation 12 Is this page still displayed for .NET UI?

When a client generates an error, the general information page initially displays. This includes an error number, description, and other general information.

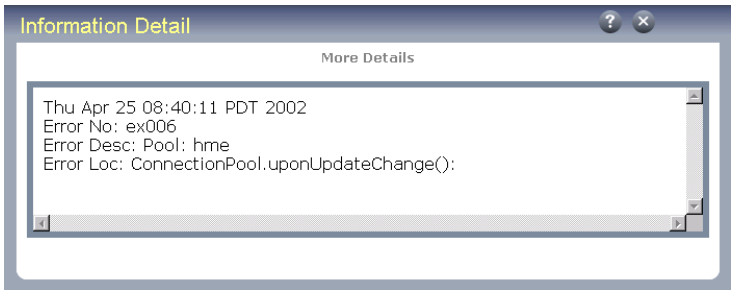
Fig. 11.1
Error Information Page



When the client error page displays, a user has several options:

- Click the exit (x) icon to exit and start a new client session.
- Click the question mark (?) icon for additional information about the error. Requesting help displays summary help information in a new browser window.
- Click More Details to display more error details. The Information Detail page displays additional information related to the error generated.

Fig. 11.2
Error Information Detail Page



- Click Support Information for support-related details.

Fig. 11.3
Desktop Client Error Support Information Detail

```
mailBody=
Page Title:  Information
Mailed From: http://crsu04.qad.com:8080/Desktop/error.jsp

Thu Apr 25 08:40:11 PDT 2002

Error No: ex006
Error Desc: Pool: hme
Error Loc: ConnectionPool.uponUpdateChange():

Thu Apr 25 08:40:11 PDT 2002

ConnectionException: Pool: hme
        at
com.qad.MFGPROWrapper.pools.ConnectionPool.uponUpdateChange(ConnectionPool.java:
        at com.qad.MFGPROWrapper.ProcessAgent.announceUpdateChange(ProcessAgent
        at com.qad.MFGPROWrapper.ProcessAgent.runRequest(ProcessAgent.java:757)
        at com.qad.MFGPROWrapper.adapter.HTMLAdapter.service(HTMLAdapter.java:2
        at javax.servlet.http.HttpServlet.service(HttpServlet.java:853)
        at org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(Unk
        at org.apache.catalina.core.ApplicationFilterChain.doFilter(Unknown Sou
        at org.apache.catalina.core.StandardWrapperValve.invoke(Unknown Source)
        at org.apache.catalina.core.StandardPipeline.invokeNext(Unknown Source)
        at org.apache.catalina.core.StandardPipeline.invoke(Unknown Source)
        at org.apache.catalina.core.ContainerBase.invoke(Unknown Source)
        at org.apache.catalina.core.StandardContextValve.invoke(Unknown Source)
```

Error Handling

The following tables list some common error messages you may encounter when using .NET UI. Only the most common errors and their causes are listed.

Failure Exception Errors

These errors occur when a utility function fails, in particular while processing XSL or XSLT documents.

Table 11.2
Failure Exception Errors

Failure Exception Error	Description
ex001 Error with processing XSLT Document	XSLT translation occurs after the XML document retrieved from the server has been successfully parsed. This exception is generated when an error occurs at the point of translation. The likely cause is an invalid change made to the XSL document (currently `/xsl/screen.xsl`).
ex002 Error with processing XML Document	The XML document is retrieved from the Progress server and then parsed. This error occurs when parsing of the retrieved document fails. A possible cause of this is an error in the creation of the XML document. Another possible cause is if the server connection fails while retrieving the XML document, so that the complete document is not retrieved.
ex003 Error writing document to file	When the user is updating configuration settings using the Connection Manager update page, the new settings must be written to file. This exception is generated when an error occurs writing to the file.

Connection Exception Errors

This class of exception is related to the activities of creating, retrieving, and resetting server sessions.

Table 11.3
Connection Exception Errors

Connection Exception Error	Description
ex001 Unable to initialize session	This exception occurs when a problem occurs initializing the back-end sessions. Possible causes would be that the maximum number of sessions has been reached, or the back-end databases are not up. There may also be a problem with the log-in script or a step omitted. This can be tested by following the log-in script section of `config.xml` and ensuring all stages are correct. Another possible issue is that the telnet session is running in normal mode instead of MFGWrapper mode. Ensure that the MFGPRO startup script contains the `-param mfgwrapper=true` statement.
ex002 Unable to get working session	Each application running on the client corresponds to a telnet session on the server. Each time a request is made on the client, the server has to link up the client's request to the server session. This error occurs when the system is unable to find the telnet session. Possible cause is that the telnet session has been shut down, perhaps as a result of an error, or it may have been closed down by Connection Manager.

Connection Exception Error	Description
ex003 Failure initializing Connection Manager	This error is generated when a problem occurs during initialization of Connection Manager. One responsibility of Connection Manager is to open XML configuration files. This error can occur if there is an invalid configuration file. To see a list of files, look at `config-files.xml` in the `\webapps\qaduiConfig\WEB-INF\config` directory.
ex004 Unable to get an idle session	When a client requests a new program to be run, an idle session is retrieved from Connection Manager. Possible cause of this exception is that Connection Manager has exceeded its maximum number of connections. It may be possible that Connection Manager is in the process of initializing new sessions but they are unavailable at this time.
ex005 Unable to get session because pool has been shutdown	This exception occurs when the user is attempting to either create a new application or make a request on an existing one and Connection Manager has been shut down. In this case, all current connections are invalid and Connection Manager has to be restarted.
ex006 User has exceeded max session	A user has requested to run another application; however, this user has exceeded the maximum number of allowed connections.

Process Exception Errors

This class of exception occurs during interaction between the client and server while sending and retrieving data and during the management of that process.

Table 11.4
Process Exception Errors

Process Exception Error	Description
ex001 Unable to initialize session with program	This error occurs after a dedicated session has been assigned to a client request. The client is attempting to run a new program and the launch of that program fails. This can occur if the user is not properly logged in.
ex002/ex003 Failure occurred sending message to begin to submit data	This exception occurs when data submission to the server fails. An acknowledgement may fail due to the trigger failing to fire. To find the root cause of this exception, follow the process through a character session and look for any unusual user interface functions such as alert boxes, selection lists, or browses.
ex004 Failure occurred sending get frame message	This exception occurs when a request for the screen XML times out. This could be due to the trigger not firing (see ex003) or the XML failing to be received by the XMLReceiverServlet. To attempt to debug this further, enable tracing on Progress and ProcessAgent.

Process Exception Error	Description
ex005 Failure occurred sending spacebar message	A pause event occurs within the application and the .NET UI system is unable to respond to this event. To attempt to debug this further, enable tracing on Progress and ProcessAgent.
ex006 Failure occurred parsing xml document	This exception incorporates the occurrence of FailureException (ex002) but will aid to highlight the cause of the exception.

Adapter Exception Errors

Each client request is filtered through a servlet. Adapter exceptions are raised if an error occurs in this servlet.

Table 11.5
Adapter Exception Errors

Adapter Exception Error	Description
ex001 Parameters not passed correctly	This exception occurs if the servlet is unable to interpret the parameters sent to it from the browser.
ex002 Unable to create xml document from parameters	The servlet creates an XML document from the parameters it is passed. This exception occurs if a problem occurs parsing the XML document.
ex003 Connection Manager is unavailable	The servlet requests Connection Manager to provide the back-end session. If Connection Manager has not been launched, then this exception is generated.
ex004 Error occurred initializing Parameter Manager	The Parameter Manager is responsible for reading and parsing the configuration files used in .NET UI . Under normal circumstances, Connection Manager will be launched prior to launching any new applications and as part of its initialization it will create the Parameter Manager. However, if the connection pool is not initialized, then the servlet will initialize the Parameter Manager. This error occurs if there is an invalid configuration file. To see a list of files opened by the Parameter Manager, look at `config-files.xml`.

Chapter 12

Reference

This chapter provides additional reference. This information is provided to give you a better understanding of the .NET UI system and its interaction with MFG/PRO.

Installed Components

This section provides a brief description of the directory structure created or updated by the .NET UI system-generation process.

If you want to look at information about the current release, release notes are provided at the top level of the QAD .NET UI CD: they are not copied during installation.

Note In the following listing, UNIX directory conventions are used. If you installed on a Windows server, the relative paths are the same.

MFG/PRO Installation Directory

The installation creates a set of directories in the MFG/PRO installation directory for UI files.

```
MFGPROInstallDir/qaduiConfig/com/
    - /mfgpro
        - /configxml <files for simplified screens>
        - /queries <files for HTML lookups>
    - /qad/
        - /kb <files used by lean workbenches>
        - /mfgpro <release-specific MFG/PRO code>
        - /nav
            - /conf <contains template for config.properties>
            - /data <directory for data files loaded during build>
            - /html <HTML templates>
            *.r <Desktop-specific Progress r-code>
```

Tomcat Servlet Container

The QAD .NET UI media includes the latest edition of the Tomcat engine; it is recommended that you install and use this version.

Annotation 13 Are these installed for .NET UI?

The following directory and files are added as part of the Desktop Web application under the Tomcat installation directory:

```
TomcatInstallDir/webapps/qaduiConfig
    - /admin <Desktop Administration pages>
    - /applet <location of multinet.jar>
    - /cache <stored versions of HTML pages, may need to be cleared>
    - /client_setup <JRE and client install script>
    - /css <Desktop and process flow style sheets>
    - /doc <directory for online user guide files>
    - /images <directory for graphic files>
    - /java <contains the multinet.jar file>
    - /js <directory for JavaScript files>
    - /jsp <Java server pages>
    - /menus/xx <language-specific menu HTML>
    - /pronav <files for Process Editor>
    - /uiconfig <location of simplified screens>
    - /WEB-INF
        - /conf <configuration files for Desktop components>
        - /cgi <WebSpeed Messenger and other executables>
        - /log <log files for Desktop>
    - /XSL <XSL files>
```

Configuration Files

Two important configuration files in the `/WEB-INF/conf` directory are:

- `connectionManagerConfig.xml`, which stores configuration values for Connection Manager
- `process-config.xml`, which stores configuration values for the Process Editor

Language Directories for Menus

.NET UI menu files are generated and placed in the Tomcat `webapps` directory in a two-letter language subdirectory of the `/menus` directory during the build process: *`TomcatInstallDir`*`/webapps/` *`qaduiConfig`*`/menus/`*`xx`*. The two-letter ISO language code derived from the `locale.dat` file is used as the directory name.

If you build menus in a language other than English, two directories are created: `en` (for US English) and one with the ISO version of the language code you specify during the build. For example, if you install French files, the directory would be `fr`.

Working with WebSpeed

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

Install WebSpeed Messenger

If you do not have a full WebSpeed installation on the Tomcat server in a multi-tier installation, you must download and install the WebSpeed Messenger in the Tomcat `cgi` directory. You must have Web access and administrator access to the *`ProgressInstallDir`*.

Use the following steps to download and install WebSpeed Messenger:

1. Download WebSpeed Messenger from the Progress Web site.

   http://www.progress.com/end/index.ssp

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

3. Install the WebSpeed Messenger using the installation instructions provided on the download page. Note the installation directory; during the configuration steps, you edit and copy several files in this directory.

Configure WebSpeed Messenger

UNIX Servers: If Tomcat is installed on UNIX, go to the *`WebSpeedInstallDir`*`/properties` directory and copy `msngrs.properties` to `ubroker.properties`.

1. Using a text editor, open `ubroker.properties`.

2. Find the `[NameServer.NS1]` section and add the following parameters:

```
location=remote
hostName=DataBaseServerName
```

3 Find the `[WebSpeed.Messengers.CGIIP]` section. Depending on the OS where Tomcat is installed, add the following parameters; use `wspd_cgi.ksh` for UNIX or `cgiip.exe` for Windows.

UNIX server

```
msngrExecFile=@{Startup/DLC}/bin/wspd_cgi.ksh
AllowMsngrCmds=1
Host=DataBaseServerName
Port=NameServerPortOnDBServer
registerNameServer=1
```

Windows server

```
msngrExecFile=@{Startup\DLC}\bin\cgiip.exe
AllowMsngrCmds=1
Host=DataBaseServerName
Port=NameServerPortOnDBServer
registerNameServer=1
```

4 Verify your edits and save the file.

5 Copy the WebSpeed executable from the `\dlc\bin` directory on the application server to the Tomcat `cgi` directory on the application server.

- For UNIX servers, copy `wspd_cgi.sh`. In the target directory, rename the file to `wspd_cgi.ksh`.
- For Windows servers, copy the `cgiip.exe` file.

The file should be placed in the following directory:

TomcatInstallDir`/webapps/`*qaduiConfig*`/WEB-INF/cgi/`

Verify the WebSpeed Messenger

To verify that WebSpeed Messenger is installed and configured correctly, launch a Web browser and enter this URL:

`http://`*TomcatHost*`/cgi-bin/wspd_cgi.ksh/WService=BrokerName/ping`

- *TomcatHost* is the Tomcat server where the remote messenger was installed.
- *BrokerName* is your WebSpeed broker name.

The WebSpeed home page should display.

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

WebSpeed Agent Sizing and Behavior

WebSpeed agents are used to extract information from the database for browses, inquiries, and reports that have the Web Logic Implemented field set to Yes in Program Information Maintenance (36.3.21.1). Each WebSpeed agent services one user request at a time.

Note By default, browses and lookups are Web-enabled; reports and inquiries display with HTML screens. However, this can be changed by resetting the Web Logic Implemented field in Program Information Maintenance.

Annotation 14 Is this still true for .NET UI?

When calculating the number of WebSpeed agent licenses required for your site, you should consider the following:

- The number of reports and large browses being run simultaneously
- The number of seconds of inactivity before an agent is automatically released
- The number of simultaneous activities each end user is allowed

For most activities, the agent is used only briefly to retrieve the information required to present or update the user interface.

Agent Usage for Reports and Inquiries

For reports and inquiries that have been changed to use WebSpeed (by setting Web Logic Implemented to Yes), a WebSpeed agent is in use—that is, it is servicing a user request—when it is extracting report or inquiry data from the database. After delivering this data to the Web server, the WebSpeed agent is free to service another request, even if .NET UI has not yet displayed all the data to the user.

Agent Usage for Browses

For browses, a WebSpeed agent is in use when it is delivering the number of records specified as the Max Browse Records setting in User Option Maintenance (36.20.10.1).

Security

This section includes additional information about .NET UI security.

Securing Internet Communication

QAD recommends using encrypted Virtual Private Network (VPN) connections when accessing Desktop over the Internet. VPN provides a secure communication channel for .NET UI sessions. This requires no additional setup for .NET UI and requires that the user is authenticated and has authorization to log on to the network.

Secure Sockets Layer is recommended for local network traffic.

Setting up SSL with .NET UI

During installation of the .NET UI, you are prompted for the Tomcat port number, which defaults to 8080. This is the port for the HTTP listener, referenced by http://servername:8080. The default HTTPS listener is port 8443, referenced by https://servername:8443.

If you need to use a different port for HTTPS, follow the instructions in “Update Tomcat Port Numbers” on page 85 to modify `server.xml` in the *`TomcatInstallDir`*`/conf` directory. For HTTPS, edit the section for SSL HTTP/1.1 and modify the Connector port shown here:

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

```
clientAuth="false" sslProtocol="TLS" />
```

Desktop installs a placeholder certificate located in the directory specified by:

```
keystoreFile="bin/keystore"
```

This directory is relative to the Tomcat install directory. If you use SSL, replace the default certificate with your own valid, signed certificate. While you can run under HTTPS with the certificate supplied by QAD, users will see security warnings about the certificate.

To find more information on how to configure SSL, review the information on the Tomcat Web site:

http://jakarta.apache.org/tomcat/tomcat-5.5-doc/ssl-howto.html

Note If you implement HTTPS, do not change the Menu Lookup URL in the administration settings for the Process Editor from HTTP to HTTPS. This URL is used on the server and does not affect client communication.

UNIX Telnet Environment Security Issues

To access a subset of MFG/PRO maintenance programs, the .NET UI uses telnet to log on to the server where MFG/PRO is installed. Because .NET UI communicates via HTTP, the account log-in ID and password are sent using unencrypted text. Since this may compromise system security, you should configure the telnet environment with server-side security measures.

A range of security options exists to solve the unencrypted log-in and password problem. This section outlines two security scenarios: one providing a maximum level of security and one providing less security but more flexibility for Progress client session and home directory access. In both setups, it is recommended that you use a restricted shell (rsh).

Restricted Shells

Restricted shells are restricted versions of the common UNIX Bourne shell or Korn shell. In the Bourne shell, the restricted shell is run as `rsh` (`/usr/lib/rsh`), while in the Korn Shell it is known as `rksh` (`/usr/bin/rksh`). The restricted versions of these shells allow users to log in with restricted access. They cannot:

- Use the `cd` command to change directories.
- Specify a path or command using `/`.
- Use redirection (>, >>).
- Set the value of `$PATH`.

Tip
The default shell for a user is located in the `/etc/passwd` file.

Note A user's path should not include `/usr/bin`. This lets the user run another shell, thereby inheriting access to any commands that the child shell allows.

Examples of Security Measures

Case 1: Maximum Security

One UNIX account with the following characteristics is used for all telnet sessions:

- No write permissions to home directory. Temporary files are written elsewhere.
- $PATH, $DLC, and $PROPATH environment variables are set in `.profile` and inaccessible to the user.
- Startup command and/or scripts are run from `.profile`.
- Telnet disconnects immediately after the user exits the MFG/PRO session.

Use the following instructions to set up Case 1:

1. Create the unique Desktop account for log-in to UNIX through telnet.
2. Make the default shell for this account the restricted shell.
3. Remove all write permissions for this user in their home directory. Use the `-T` option in the remote script to specify an alternate temporary directory.
4. Set up the `.profile` to set minimal environment variables.
5. Set up the `.profile` to run the script automatically.

Example `.profile` for Case 1:

```
/*Sample .profile for Desktop mainten session, single Desktop login*/

#set default for error (STOP) condition handling
stty intr ^C

#set environment variables
PATH=/dlc91:/dlc91/bin
DLC=${DLC - /dlc91}
PROEXE=${PROEXE - $DLC/bin/_progres}
export PATH DLC PROEXE

#Autorun remote script for Desktop Access and automatically exit
exec remote.script
exit
```

Case 2: Moderate Security

Users have their own unique log-in and password, but run the restricted shell by default:

- Write permission to the home directory is possible, but not necessary.
- $PATH, $DLC, and $PROPATH environment variables are set in `.profile` and inaccessible to the user.
- Users run a subset of UNIX commands, which you add to `/usr/rbin`.
- Users can run MFG/PRO manually from a command line or script.

Use the following instructions to set up Case 2:

1. Create or modify accounts for users of MFG/PRO maintenance programs by changing their default shell in the `/etc/passwd` file to the restricted shell.
2. Create the directory `/usr/rbin` and copy the UNIX commands necessary for these users. Make the `/usr/bin` directory read-only so users cannot change path variables.

3 Set up a special `.profile` for maintenance program users.

4 Set the minimal environment variables, remembering to include `/usr/rbin`.

5 Copy the telnet connection script to each user's home directory with read-only access.

6 Put any other necessary read-only script files in the home directory.

Example `.profile` for Case 2:

```
/*Sample .profile for Desktop session for individual logins
remote.script should be in home directory; executable by Desktop*/

#set default for error (STOP) condition handling
stty intr ^C

#set environment variables
PATH=/dlc91:/dlc91/bin:/usr/rbin#don't forget /rbin directory
DLC=${DLC - /dlc91}
PROEXE=${PROEXE - $DLC/bin/_progres}
export PATH DLC PROEXE
```

Configure Multiple UI Systems

This section discusses issues you should be aware of when you:

- Configure multiple .NET UI systems on one server.
- Configure multiple .NET UI systems for more than one QAD Enterprise Applications (MFG/PRO) release.

Configuring Multiple Systems on the Same Server

If needed, you can create multiple .NET UI systems on the same server. However, some aspects of each .NET UI system are set up individually. Review the following example to understand these requirements. They assume a Test, Production, and Train system were configured on a Windows server. This setup on a UNIX server would use the same relative paths.

If you want to provide .NET UI clients with access to additional database sets—for example, the Training database set—you cannot share the same administration database between the sets.

When you use the Perl deployment script to set up these systems, several directories are created and populated based on the unique Web application names you defined:

- Test
- Production
- Train

Each .NET UI system has a unique directory in Tomcat `webapps`:

- c:*TomcatInstallDir*\webapps\test
- c:*TomcatInstallDir*\webapps\production
- c:*TomcatInstallDir*\webapps\train

The setup process creates additional subdirectories for each system. It creates `\mfgpro` for configuration files and QAD (MFG/PRO) Progress programs (`.p` files) and `\qad` for the compiled .NET UI-specific r-code. For example, for the test .NET UI system these directories are:

- `c:\mfgsvr\test\com\qad`

- `c:\mfgsvr\test\com\mfgpro`

After running the setup process for the three environments, each system also has a unique configuration `.ini` file stored in the log files directory you specify. These examples assume you used the *`MFGPROInstallDir`* as the logs directory:

- `c:\mfgsvr\test_dt2.ini`
- `c:\mfgsvr\production_dt2.ini`
- `c:\mfgsvr\train_dt2.ini`

Configuring .NET UI for Multiple Releases

If your environment requires setting up .NET UI systems for multiple QAD Enterprise Applications versions running on the same server, the `.ini` and log files for .NET UI systems running on each release must be in their own release-specific log files directory.

For example, if you have MFG/PRO eB2 Service Pack 8 and eB2.1 installed on a single server, a separate log file directory is required for each MFG/PRO release. Do not store `.ini` files for different releases in the same .NET UI log directory. You can store multiple `.ini` files for the same release in the same directory.

Administration with Tomcat

This section includes additional information on Tomcat, including:

- How to install a new version of Tomcat
- Clearing the Tomcat server cache

Install a New Tomcat Servlet Engine

To install the new Tomcat version included on the .NET UI media, launch the deployment script and use the Tomcat installation menu option. You can choose to install Tomcat in the same directory as an existing Tomcat installation, or in a new directory. If you choose to install the new version in an existing Tomcat installation directory and the deployment tools detect the Tomcat versions are compatible, you can optionally merge the existing Tomcat configuration files with the new Tomcat version.

Find your current Tomcat version by reviewing the release notes in the Tomcat root directory.

Manually Install Tomcat and Merge QAD Components

Important To ensure system integrity, .NET UI deployment tools should be used to manage the Tomcat installation process. These details are provided for information purposes only.

Use the following instructions to manually merge QAD-provided Tomcat components with your Tomcat version:

1. Unzip the QAD-provided Tomcat version:
   a. UNIX: Mount the Desktop installation CD.
   b. Navigate to the `\zips` subdirectory on the Desktop installation media.

c. Unzip the `Tomcat.zip` file to a temporary directory.

2. Merge the Tomcat components.

a. Go to the temporary directory where you installed the QAD-provided Tomcat engine. Copy `server.xml` and `tomcat-users.xml` from the `\conf` subdirectory to the `\conf` subdirectory of your Tomcat version. Overwrite existing files.

b. Merge the QAD-provided *`TempDir`*`\bin\setenv.sh` file with the `setenv.sh` file in the *`TomcatInstallDir`*`\bin`. If your Tomcat `\bin` directory does not contain a `setenv.sh` file, copy the QAD-provided file to your *`TomcatInstallDir`*`\bin` subdirectory.

c. Create a file named `version.txt` that references the QAD-compatible Tomcat version number. The entry should look like the following:

```
tomcat.version=5.0.27
```

Clear the Tomcat Server Cache

When the following events occur, you should clear the server cache to ensure new information is properly displayed in .NET UI:

- Before you install a new .NET UI release over an existing one

  Note It is recommended to complete installations in a new directory.
- After you rebuild the .NET UI
- After you attach or detach a lookup from any field

To clear the cache, follow these steps:

1. Shut down Tomcat.
2. Go to the *`TomcatInstallDir`*`/webapps/`*`qaduiConfig`*`/cache` directory.
3. Delete all subdirectories and files located in this directory.
4. If you are installing a new .NET UI release, complete the installation.
5. Restart Tomcat.

Use Diagnostic Tools

This section includes information about a number of diagnostic techniques and tools you can use with your .NET UI environment. These tools can be especially helpful when first installing and tuning the application or upgrading to a new release of .NET UI.

Details are included about:

- Setting logging parameters
- Enabling the Java Console
- Enabling performance timing

Configure Logging

Annotation 15 Is this true for .NET UI?

Logging in .NET UI is facilitated by the Apache Jakarta Project Log4j infrastructure. This means that log statements are in the code; their output can be configured in an external XML file. This file is called `logging.xml` and is located in:

TomcatInstallDir`/webapps/`*qaduiConfig*`/WEB-INF/conf`

All log information is written to the `Desktop.log` file located in:

TomcatInstallDir`/webapps/`*qaduiConfig*`/WEB-INF/logs/`

It may be useful to move these files to backup storage, or to delete them, once the records are no longer required.

Log4j is highly configurable, based on settings that can be modified in the `logging.xml` file. Information on how to edit this file can be found on the Log4j Web site:

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

One configurable setting of interest is the logging level. This can be set to any of the following and can be useful in troubleshooting errors:

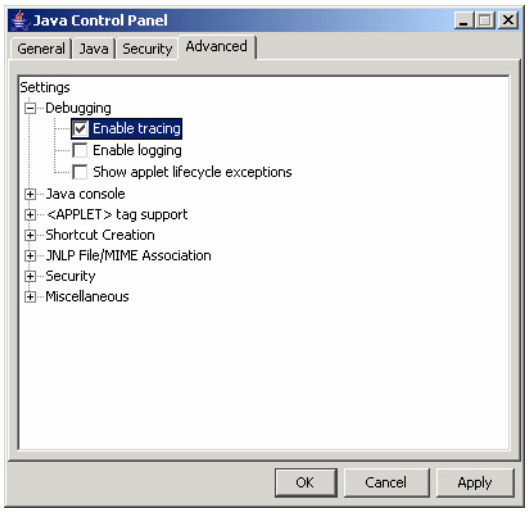
- Error – error messages only
- Warning – error and warning messages only
- Info – error, warning, and info messages only
- Debug – all messages

Enable the Java Console

If you selected Enable Java Console Tracing when you configured your .NET UI system with MFG/UTIL, you must enable the Java console to view the trace output. Follow these steps:

1. From the Windows Start button on the .NET UI client machine, click Start|Control Panel.
2. Double-click the icon for Java.
3. When the Java Control Panel displays, click the Advanced tab.
4. Expand the Debugging node and select Enable tracing.

Fig. 12.1
Java Control Panel



5 Click Apply to save your changes.

Enable Performance Testing

Annotation 16 Is this utility still used?

To help you troubleshoot .NET UI performance related to your network, you can activate a timer for .NET UI maintenance programs. Use this function to help you pinpoint network issues affecting the Desktop environment by timing Desktop performance on multiple clients and recording the performance.

1 Using a text editor, open `general.js` for editing. This file is in the `js` subdirectory of the Desktop installation directory.

2 Add the following new line at the top of the file and save the file:

```
var htmlTiming = "true";
```

Support Programs

Desktop incorporates various Web-based technologies to support features such as the Process Editor and the display of MFG/PRO programs in HTML screens. These technologies are included with the product and are transparent to the user. They are listed here to give credit to the open-source projects that created them.

- SAXON XSLT Processor from Michael Kay

  http://saxon.sourceforge.net
- Struts, an open-source framework for building Web applications, part of the Jakarta Project, sponsored by the Apache Software Foundation

  http://jakarta.apache.org/struts/index.html
- Xerces for parsing and generating XML, from the Apache Software Foundation

  http://xml.apache.org/xerces2-j/index.html

- Batik, a Java-based toolkit for use with images in the Scalable Vector Graphics (SVG) format, from the Apache Software Foundation

  http://xml.apache.org/batik/

- JDOM, a Java-based solution for accessing, manipulating, and outputting XML data from Java code from the JDOM Project

  http://www.jdom.org/

- JFreeChart, a Java class library for generating charts

  http://www.jfree.org/jfreechart/

See “Configure Logging” on page 143 for details.

- Log4j, a logging API for advanced logging options in Desktop, part of the Jakarta Project, sponsored by the Apache Software Foundation

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

- SecurityFilter, a Java Servlet Filter that offers the features of container managed security, from SourceForge.net.

  http://sourceforge.net/projects/securityfilter/

Glossary

Accelerator Keys. Keystroke equivalents to mouse actions. For example, you can delete a record in QAD UI by choosing Delete from the context menu or by pressing Ctrl+Shift+D.

Apache Software Foundation. A membership-based, not-for-profit corporation that exists to provide organizational, legal, and financial support for the Apache open-source software projects.

Applet. See *Java Applet*.

Array. A field or variable with multiple elements, each element having the same data type.

Bookmark. A saved link to a Web page that has been added to a list. Bookmarks are found on the Favorites menu in Internet Explorer.

Browse. An inquiry program that displays records in a tabular format. Browse types in QAD UI include look-up and drill-down browses. Drill-down browses can be specified on the menu or associated with a field. Look-up browses are always associated with individual fields for use in selecting entry values. See also *Drill-Down Browse* and *Look-Up Browse*.

Browser. An application used to view and navigate the World Wide Web and other Internet resources. Common browsers include Microsoft Internet Explorer and Netscape Navigator.

Bytecode. Compiled Java code containing instructions to the Java virtual machine. The virtual machine, in turn, interprets these instructions so they can be performed by the system. See also *Java Virtual Machine (JVM)*.

Cascading Style Sheet (CSS). A Web page derived from multiple sources with a defined order of precedence to manage conflicts among style element definitions. The cascading style sheet, level 1 (CSS1) recommendation from the W3C is implemented in the latest versions of the Netscape and Microsoft Web browsers.

CGI. See *Common Gateway Interface (CGI)*.

Character Interface. Sometimes abbreviated as CUI or CHUI, an interface to MFG/PRO that displays only keyboard characters on the screen rather than icons and buttons. In QAD UI, telnet maintenance programs display in the character interface, while browses, reports, inquiries and HTML maintenance programs display in a full graphical user interface. Compare with *Graphical User Interface (GUI)*.

Comma-Separated Values (CSV). Sometimes referred to as *flat files*, CSV files organize values as a series of ASCII text lines where each column value is separated by a comma from the next column's value and each row starts a new line. Data in CSV files can be conveyed as input to other table-oriented applications such as Microsoft Excel.

Common Gateway Interface (CGI). A standard way for a Web server to pass a Web user's request to an application program and to receive data in response, which it then forwards to the user. CGI is part of the Web's hypertext transfer protocol (HTTP). See also *HTTP (Hypertext Transfer Protocol)*.

Context Menu. A pop-up menu that displays when you use a right mouse click. The choices on the menu vary depending on where you click; only actions valid in the current context display.

Cookie. Information a Web site puts on your hard disk so that it can retrieve information about you at a later time. Typically, a cookie records your preferences when using a particular site.

CSS. See *Cascading Style Sheet (CSS)*.

CSV. See *Comma-Separated Values (CSV)*.

Data Encryption. See *Encryption*.

DHTML (Dynamic Hypertext Markup Language). A combination of new HTML tags and options that support more animated and interactive Web pages. Much of dynamic HTML is specified in HTML 4.0 with Javascript and cascading style sheets. See also *HTML (Hypertext Markup Language)*.

Document Object Model (DOM). A programming interface specification being developed by the W3C that lets a programmer create and modify HTML pages and XML documents as program objects, rather than data structures.

Download Size. In browses, this is the number of records the browse window displays before the client notifies the server to retrieve more records.

Drill-Down Browse. A type of browse that is available from a menu or associated with individual fields. You can filter, graph, and print from drill-down browses. In QAD UI, drill-down browses apply to fields in other browses, inquiries, and reports. Compare with *Look-Up Browse*.

Emulation. See *Terminal Emulation*.

Encryption. Conversion of data into a form that cannot be easily intercepted by unauthorized people.

Extensible Markup Language (XML). A markup language that describes data. The QAD UI uses XML to create HTML screens.

Extensible Stylesheet Language (XSL). A language for formatting an XML document; for example, showing how the data described in the XML document should be presented in a Web page. XSLT shows how the XML document should be reorganized into another data structure (which could then be presented by following an XSL style sheet).

Extensible Stylesheet Language Transformation (XSLT). A standard way to describe how to transform the structure of an XML document into an XML document with a different structure. The coding for the XSLT is also referred to as a style sheet and can be combined with an XSL style sheet or be used independently.

Extranet. A private network designed to securely share portions of business information or operations with suppliers, partners, customers, or other businesses. An extranet can be viewed as part of a company's intranet that is extended to users outside the company. See also *Internet Server Application Programming Interface (ISAPI)*.

Field Tips. Context-specific references to fields consisting of the Progress field name.

Georgia SoftWorks Windows Telnet Server (GSWTS). The third-party telnet server software supplied with the QAD UI for Windows NT servers.

Graphical User Interface (GUI). A user interface that presents computer actions and options as pictures, buttons, and icons. The most common example of a graphical user interface is Microsoft Windows. Compare with *Character Interface*.

Graphics Interchange Format (GIF). A graphics file format that is the industry standard for Web and Internet use.

Grid. A screen design element that organizes related data fields in a format similar to a spreadsheet or table.

Group. See *Servlet*.

GUI. See *Graphical User Interface (GUI)*.

HMAC (Hash Message Authentication Code). A type of message authentication code (MAC) calculated using a cryptographic hash function in combination with a secret key. It may be used to simultaneously verify both the data integrity and the authenticity of a message.

HTML (Hypertext Markup Language). A tag-based ASCII language used to create pages on the World Wide Web. HTML uses codes surrounding a block of text to indicate how it should display. In HTML, you can also specify that a block of text, or a word, is linked to another file on the Internet.

HTTP (Hypertext Transfer Protocol). The set of rules for exchanging text, graphic images, sound, video, and other multimedia files on the World Wide Web. See also *Common Gateway Interface (CGI)*.

Interface. See *User Interface (UI)*.

International Organization for Standardization. See *ISO*.

Internet. A system of linked computer networks—international in scope—that facilitates data communication services such as remote log-in, file transfer, electronic mail, and newsgroups. The Internet is a way of connecting existing computer networks.

Internet Information Server (IIS). Web server software from Microsoft Corporation that supports a common gateway interface.

Internet Server Application Programming Interface (ISAPI). A program-level means of communicating with the Microsoft Internet Explorer Web server.

Intranet. A private network inside a company or organization that uses the same kinds of software found on the public Internet.

ISO. ISO (International Organization for Standardization), founded in 1946, is a worldwide federation of national standards bodies from some 100 countries. ISO is not an abbreviation. It is a word, derived from the Greek *isos*, meaning equal. The name ISO is used around the world to denote the organization.

ISS. See *Internet Information Server (IIS)*.

JAR File. See *Java Archive File (JAR)*.

Java 2 Platform, Enterprise Edition (J2EE). A recent release of Java designed to support the requirements of large-scale computing systems. Features include Java servlets and Java Server Pages (JSPs), which facilitate dynamic Web-enabled data access and manipulation.

Java. An object-oriented programming language created by Sun Microsystems. Java is a device-independent language. Programs compiled in Java can be run on any computer. Java programs can be run as free-standing applications or as applets placed on a Web page.

Java Applet. A small application program that can be sent along with a Web page to a user. Applets written in Java are served from a Web site but executed on the client computer.

Java Archive File (JAR). A file that contains the class, image, and sound components of a Java applet gathered into a single file and compressed for faster downloading to a Web browser.

Java Bean. A reusable program building block developed with a Beans Development Kit (BDK) from Sun Microsystem.

Java Class. In object-oriented programming, a class is a template definition of the methods and variables in a particular kind of object.

Java Development Kit (JDK). A software development environment from Sun Microsystems for writing applets and applications in the Java programming language.

Java Plug-in. Software provided by Sun Microsystems that replaces the default virtual machine associated with a Web browser. Using the Java plug-in allows developers to deploy Java applets that depend on the latest features of the Java platform and be assured that their applets will run reliably and consistently in both Microsoft Internet Explorer and Netscape Navigator.

Java Runtime Environment (JRE). A subset of the Java Development Kit for end users and developers who want to redistribute the Java runtime environment. The Java runtime environment consists of the Java virtual machine (JVM), the Java core classes, and supporting files.

Java Server Page (JSP). A technology for controlling the content or appearance of Web pages through the use of servlets, small programs that are specified in the Web page and run on the Web server to modify the page before it is sent to the user who requested it.

Java Virtual Machine (JVM). The part of the Java runtime environment responsible for interpreting bytecode. See also *Bytecode*.

JDK. See *Java Development Kit (JDK)*.

JRE. See *Java Runtime Environment (JRE)*.

JSP. See *Java Server Page (JSP)*.

JVM. See *Java Virtual Machine (JVM)*.

Keyed-Hash Message Authentication Code (HMAC). A type of message authentication code (MAC) calculated using a cryptographic hash function in combination with a secret key.

Look-Up Browse. A type of browse that is only associated with individual program fields for use in selecting entry values. Look-up browses contain less detail than drill-down browses and cannot be used to filter, graph, or print data. Compare with *Drill-Down Browse*.

Maintenance Programs. Programs used to add, modify, and delete records and codes in the MFG/PRO database. In QAD UI, most maintenance programs display as HTML pages.

Menu Substitution. Replacing one program with another on the MFG/PRO menu; for example, replacing a standard program with a customized version.

MFGX.net. The QAD portal, which is a collaborative environment where members of the manufacturing community can share information and work together to achieve overall competitiveness.

MIME (Multipurpose Internet Mail Extension). A protocol for exchanging different kinds of information on the Internet. The MIME header is inserted at the beginning of a Web transmission so that client programs can select the appropriate associated application.

Netscape Server Application Programming Interface (NSAPI). A program-level means of communicating with the Netscape Web server.

Network. A series of points or nodes interconnected by communication paths. The Internet is a common example of a network. See also *Internet*, *Extranet*, *Internet Server Application Programming Interface (ISAPI)*, and *Wide Area Network (WAN)*.

Network User Interface (NetUI). An earlier version of the Web-enabled interface now replaced by QAD Desktop.

Node. In the Process Editor, nodes represent steps in a process and can contain URL links to other resources.

Perl. A script programming language (Practical Extraction and Reporting Language) that is similar in syntax to the C language and that includes a number of popular UNIX facilities. Perl is regarded as a good choice for developing Common Gateway Interface (CGI) programs because it has good text manipulation facilities.

Platform. An underlying computer system on which application programs run. Historically, most application programs had to be written to run on a particular platform. Products written in Java, however, are *cross-platform*, meaning they can be run anywhere the Java runtime environment is installed.

Point. A unit of about 1/72 inch used to measure type size.

PROPATH. An environment variable containing the list of directories Progress searches when looking for a program to execute.

Relational Database Management System (RDBMS). The Progress software that manages and provides access to the MFG/PRO databases.

RSA (Rivest-Shamir-Adleman). An Internet encryption and authentication system that uses an algorithm developed in 1977 by Ron Rivest, Adi Shamir, and Leonard Adleman. The RSA algorithm is included with Netscape Navigator and Microsoft Explorer.

Scalable Vector Graphics (SVG) . An XML technology for defining vector-based two-dimensional graphics for the Web.

Script. A program or sequence of instructions that is interpreted or carried out by another program.

Secure Shell (SSH). A secured connection protocol between a client and server. Provides user authentication and key exchange; negotiates encryption, compression, and message integrity verification against a keyed-hash message authentication code (HMAC).

Secure Sockets Layer (SSL). A program layer for managing the security of message transmissions in a network. The program layer exists between an application (such as a Web browser or HTTP) and the Internet's TCP/IP layers. Sockets refers to the sockets method of passing data back and forth between a client and a server program in a network or between program layers in the same computer.

Servlet. Programs similar to Java applets, that run on the server rather than the client and are used to run interactive Web applications.

Socket. A convention for connecting with and exchanging data between two program processes within the same computer or across a network. A socket represents the end point in a network connection. Sockets are created and used with a set of programming requests or function calls sometimes referred to as the sockets application program interface (API). The most common sockets API is the Berkeley UNIX C language interface.

SSL. See *Secure Sockets Layer (SSL)*.

State-Aware. Describes the condition of a browse when a WebSpeed agent is in use during the entire time it is open. This can happen when the user sorts browse data by a non-indexed field, the WebSpeed agent returns a group of data that ends with a non-unique key, or the browse takes an unusually long time to begin returning data to the Web server.

SVG. See *Scalable Vector Graphics (SVG)*.

TCP/IP. See *Transmission Control Protocol/Internet Protocol (TCP/IP)*.

Telnet. A user command and underlying TCP/IP protocol that lets you access applications and data on remote, or *host*, computers. See also *Transmission Control Protocol/Internet Protocol (TCP/IP)*.

Terminal Emulation. Use of a personal computer to interact with a computer with a different operating system. The terminal emulation program runs as a separate task with its own window. The application interface presented in this window is character-based or text-only.

Thin-Client. In a thin-client model, the client machine takes on only the user interface role. No business logic processing is accomplished on the client.

Tomcat. The servlet container used in the official reference implementation for the Java Servlet and Java Server Pages (JSP) technologies. Tomcat is developed in an open and participatory environment and released under the Apache Software Foundation license.

Tool Tips. Context-specific descriptions that display whenever your cursor is positioned over a tool button in QAD UI.

Transmission Control Protocol/Internet Protocol (TCP/IP). The basic communication language or protocol of the Internet. It can also be used as a communications protocol for intranets and extranets.

UI. See *User Interface (UI)*.

Uniform Resource Locator (URL). A text string that indicates the location of an intranet or Internet resource.

Universal Unique Identifier (UUID). A hexadecimal number including a time stamp and a host identifier. Applications use uuids to identify many kinds of entities.

User Interface (UI). The portion of an application that is visible to the user and the mechanism by which the end user interacts with the application, enters information into the application, and sees the results of the interaction.

UUID. See *Universal Unique Identifier (UUID)*.

W3C. See *World Wide Web Consortium (W3C)*.

WAN. See *Wide Area Network (WAN)*.

WAR. See *Web Archive File (WAR)*.

Web Archive File (WAR). A compressed file containing a Web application and its related files. Assists in easily deploying an entire application.

Web Browser. See *Browser*.

Web-Enabled. Refers to a program that has code added to it so that it displays with a Java-based GUI in QAD UI. User interface actions in the Progress code are bypassed and rerouted to the Java client.

Web Site. A related collection of Web files that includes an introductory file called a home page. From the home page, you can get to all the other pages at that site.

WebSpeed. A product from Progress Software consisting of two parts: a set of Web-centric development tools and a transaction Web server. The server manages high-volume database transactions across multiple servers.

Wide Area Network (WAN). Generally a corporate private network that connects computers between remote company sites.

Widget. In Java screen-design terms, identifies a unique screen element. Widgets display information or provide specific ways for users to interact with application programs.

Wildcards. Characters or symbols used in search or command functions in place of one or more letters or numbers.

World Wide Web Consortium (W3C). An international industry consortium that seeks to promote standards for the evolution of the Web and interoperability among Internet products by producing specifications and reference software.

XML. See *Extensible Markup Language (XML)*.

XSL. See *Extensible Stylesheet Language (XSL)*.

XSLT. See *Extensible Stylesheet Language Transformation (XSLT)*.

Index