

Industry-specific

QAD SOLUTIONS

Manufacturing Applications

MFG/PRO Installation Guide

QAD Desktop

System Overview
Implementation
System Administration and Reference



78-0622B
QAD Desktop Version 2.6 and higher
MFG/PRO eB, eB2, and eB2.1
October 2004

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

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

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

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

Copyright © 2004 by QAD Inc.
78-0622B

QAD Inc.
6450 Via Real
Carpinteria, California 93013
Phone (805) 684-6614
Fax (805) 684-1890
<http://www.qad.com>

Contents

About This Guide	1
What Is in This Guide?	2
Audience	2
Installation Updates	3
Related Documentation	4
Progress Documentation	4
MFG/PRO Documentation	4
Multiple Releases	5
Conventions	5
QAD Support Services	6
 Section 1 System Overview	 7
 Chapter 1 Architectural Overview	 9
Implementation Overview	10
Desktop Components	10
Prerequisite Components	11
Third-Party Components	11
QAD-Supplied Components	12
Planning an Installation	15
Deployment Options	15
How to Use This Installation Guide	16
Installation Overview	17

Installation Tools	17
--------------------------	----

Chapter 2 System Requirements..... 19

Prerequisite Requirements	20
Database Server	20
Client Requirements	20
Operating Systems	20
Minimum Client Hardware	21
Client Software	21
Web Server Requirements	23
Operating Systems	23
Software	23
Hardware	24
Progress Server Requirements	25
Operating Systems	25
Java Requirements	25
Hardware	25
Servers	26
Windows Telnet Server	26
Georgia SoftWorks Licensing	26

Section 2 Implementation.....27

Chapter 3 Preparing the Desktop Environment..... 29

Verify Prerequisite Components	30
Progress	30
MFG/PRO Database and Character Client	31
Install the Web Server	32
Prerequisites	32
Installation and Setup Overview	32
Using UNIX Web Servers	33
Verify the Web Server	34

Install the Java Development Kit	34
Prerequisites	34
Overview	34
Installation	34
Set JAVA_HOME Variable	35
Verify the J2SDK	35
Install the Windows Telnet Server	35
Prerequisites	35
Install the Telnet Server	36
Register the Georgia SoftWorks Software	37
Chapter 4 Deploying Desktop on Single-Tier	39
Single-Tier Deployment Overview	40
Deploy the Desktop Components	41
Prerequisites	41
Mount the CD-ROM (UNIX Only)	41
Launch Installation Script	42
Install HTML and Script Files	44
Install the Tomcat Servlet Engine	45
Install Desktop MFG/PRO Code	47
Set Up Windows Telnet Environment	49
Configure Desktop Components	50
Verify and Update System Information	51
Build the Desktop System	55
Generate WebSpeed Configuration Information	60
Generate Telnet Connection Scripts	61
Generate Connection Manager Log-In Information	63
Print Desktop Information	65
QAD Desktop Quick Start	66
Chapter 5 Deploying Desktop Multi-Tier	69
Multi-Tier Deployment Overview	70
Desktop Deployment Configuration File	72
Plan the Deployment	72

Configure Desktop Components	72
Prerequisites	72
Launch the Installation Script	73
Deploy the Components	75
Setup Windows Telnet Environment	77
Configure and Prepare Desktop Files	78
Verify and Update the System Information	79
Build the Desktop System	83
Create Application Server JAR Files	88
Create Connection and Configuration Files	89
Launch the Deployment Script	90

Chapter 6 Configuring Desktop After Deployment 97

Post-Deployment Configuration	98
Configure and Start WebSpeed	98
Update the WebSpeed Configuration File	99
Verify WebSpeed Servers	99
Configure and Start Tomcat	100
Update Tomcat Port Numbers	101
Start Tomcat	101
Update User Information	102
Prepare Telnet UI Connection Scripts	103
Specify Telnet Server Settings	104
Define the Log-In Sequence Script Lines	105
Configure Telnet Connections Settings	109

Chapter 7 Configuring Connection Manager 111

Overview	112
Desktop Administration	112
Plan and Size Connection Manager	114
Minimum and Maximum Connections	114
Connection Sizing Guidelines	115
Administer Connection Manager	115
Use Connection Manager Menus	116

Configure Connection Manager	117
Monitor Connections and Users	120
Troubleshoot Errors	122
Error Handling	124
Start a Desktop Client	128
Prerequisites	128
Setup	129
First-Time Startup	129
Desktop Java Plug-in Version Control	130
Running the Desktop Client on a Windows Web Server	130
Chapter 8 Implementing Multiple Languages	133
Multiple-Language Overview	134
Supported Code Pages	134
Use MFG/UTIL to Install Multiple Languages	136
Building Languages with Shared Code Pages	137
Building Languages with Compatible Code Pages	137
Configure Non-English Languages	140
Update Fonts for Non-English Languages	140
Translated Process Maps	141
Language Reference	142
Overview of Locale Information	142
Updating Environment Data: MFG/PRO eB Only	144
Updating Environment Data: MFG/PRO eB2 and Higher	149
Section 3 System Administration and Reference.	153
Chapter 9 Administering Desktop	155
Common System Administration Tasks	156
Install a New Desktop Release	156
Clear the Client Cache	158
Install an MFG/PRO Service Pack	158
Desktop Output Administration	159

Manage Menu Updates in Desktop	160
Changing MFG/PRO Menus	160
Adding Browsers	161
Adding Lookup Browsers	161
Adding and Using Custom Programs	162
Regenerating Desktop Components	162

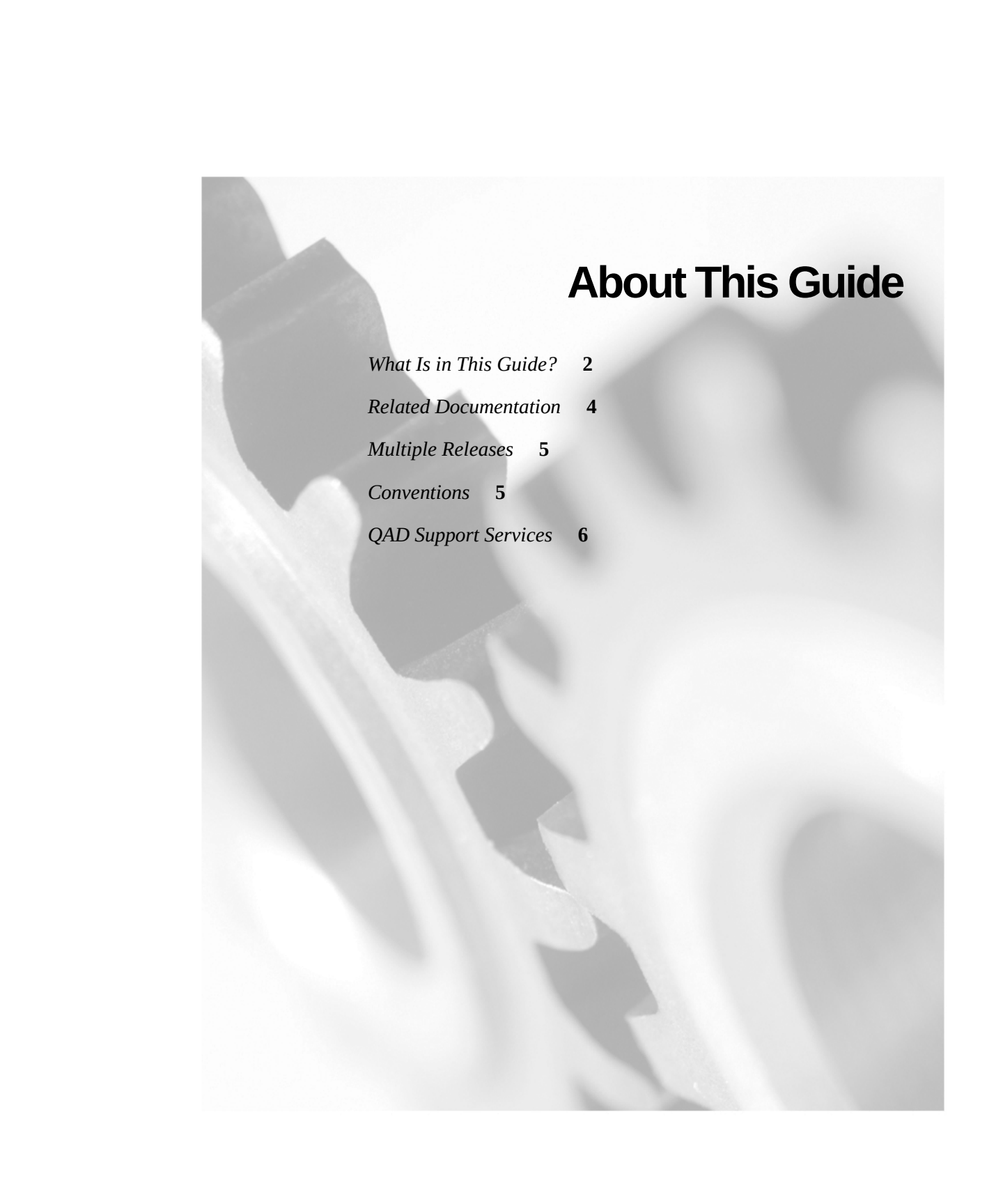
Chapter 10 Reference 165

Plan a Multi-Tier Deployment	166
Deployment Issues	167
Third-Party Component Considerations	169
Dedicated Database Server Deployment	172
Installed Components	173
MFG/PRO Installation Directory	173
Web Server Executables Directory	173
Web Server Documents Directory	174
Tomcat 4.1 Servlet Container	174
Language Directories for Menus	174
Connection Parameters	175
Local Host Connection Parameters	175
Client/Server Connection Parameters	176
Local Host Connection Parameter File Examples	177
Working with WebSpeed	178
Install and Configure WebSpeed Messenger	178
WebSpeed Agent Sizing and Behavior	181
Security	184
Securing Internet Communication	184
UNIX Telnet Environment Security Issues	184
Configure Multiple Desktop Systems	187
Configuring Multiple Systems on the Same Server	188
Configuring Desktop for Multiple MFG/PRO Releases	189
Administration with Tomcat	189
Install a New Tomcat Servlet Engine	189
Manually Install Tomcat and Merge QAD Components	190

Clear the Tomcat Server Cache	190
Tomcat Web Application Manager	191
Use Diagnostic Tools	192
Configure Logging	192
Enable the Java Console	193
Enable Performance Testing	194
Chapter 11 Deployment Tools	195
Deployment Script Reference	196
Single-Tier Deployment Options	196
Install HTML and CGI Files Menu	196
Install Tomcat Files Menu	196
Install MFG/PRO – Progress Desktop Files Menu	198
Multi-Tier Deployment Options	199
Install MFG/PRO – Progress Desktop Files Menu	199
Post MFG/UTIL Desktop Build Jar Install	201
MFG/UTIL Reference	202
Configure Desktop Environment Parameters	202
Verify and Build the Desktop System	210
Generate WebSpeed Configuration File	215
Generate Telnet Start Scripts	216
Add Parameter File Directories	217
Chapter 12 Configuring Optional User Settings	219
Configuration Overview	220
Define User Settings	221
Define Desktop URLs	224
Define Menu-Level URL Links	225
Define URL Links for Browsers	225
Monitor User Sessions	232
Other Desktop Configuration Options	233
Add New Main Menu Items	233
Assign Programs to the User Menu	234
Assign Links from One Program to Another	234

Glossary..... 235

Index..... 243

The background of the page is a grayscale image of several interlocking gears. The gears are of different sizes and are positioned in a way that they appear to be meshing together. The lighting creates highlights and shadows on the teeth of the gears, giving them a three-dimensional appearance. The overall tone is technical and industrial.

About This Guide

What Is in This Guide? 2

Related Documentation 4

Multiple Releases 5

Conventions 5

QAD Support Services 6

What Is in This Guide?

Use this guide to install and configure QAD Desktop 2. The guide is divided into three sections: Planning, Implementation, and System Administration and Reference.

▶ See “QAD Desktop Quick Start” on page 66.

Note If you have previous experience installing QAD Desktop 2 in a single-tier environment, you can go directly to the Quick Start section for a summary of installation steps.

The Desktop planning section contains instructions on how to prepare for the Desktop installation by planning the deployment and preparing these prerequisite components:

- Web server
- Java Development Kit
- Progress WebSpeed
- Windows telnet server

The implementation section includes instructions to:

- Install the Tomcat servlet engine.
- Install the QAD Desktop files.
- Configure single and multi-tier Desktop environments.
- Start up a Desktop client.
- Configure Connection Manager settings.

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

Audience

These instructions are intended for a system administrator with experience installing MFG/PRO, as well as configuring and managing hardware and operating system software. This person also should have a good understanding of networking concepts and administration, as well as an understanding of concepts and technologies discussed in “Desktop Components” on page 10.

If you do not have this expertise within your company, you should contact your QAD Support representative for information on the installation and customization offerings supplied by QAD's Global Services.

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/>

Installation Errata

In addition to these instructions, you may receive a supplementary errata sheet with changes and additional instructions. Download the most up-to-date errata sheets from the QAD Web site. New information may have been added to the errata since your product was shipped.

Installation Guides

To ensure a successful implementation, QAD installation guides are periodically updated. To determine whether your installation guide has been updated, refer to the QAD Web site. Compare the item number listed on your installation guide with the number listed on the QAD Web site. If your installation guide has been updated, download and use the most recent version.

Installation Media

The QAD Desktop 2 installation media is periodically updated with updates and enhancements. To ensure you have the most up-to-date Desktop installation media, compare the Desktop release on your installation media with the latest release available, which is listed on the latest errata sheet, posted on the QAD Web site.

Related Documentation

Progress Documentation

- For information on installing and configuring WebSpeed, see the *WebSpeed Installation and Configuration Guide*.
- For information on installing and configuring AppServer, see *Building Distributed Applications Using the Progress AppServer*.

Find the complete Progress documentation set online at:

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

MFG/PRO Documentation

- For information on installing MFG/PRO or converting to a more recent release, refer to the appropriate installation or conversion guides for your system.
- For information on using MFG/PRO functions, refer to the *User Guides*.
- For instructions on navigating the QAD Desktop, using the Process Editor, and setting up Desktop security, refer to *User Guide: QAD Desktop*.

For QAD customers with a Web account, the complete MFG/PRO documentation is available for review or downloading at:

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

Register for a QAD Web account by accessing the Web site and clicking the Accounts link at the top of the screen. 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 the product. Additionally, the QAD Web site has information about training classes and other services that can help you learn about MFG/PRO.

Multiple Releases

This guide can be used to install QAD Desktop Release 2.5 or higher on MFG/PRO eB, MFG/PRO eB2, and MFG/PRO eB2.1. Most installation and configuration steps are the same. Where needed, differences are pointed out. This list summarizes differences between the three releases that affect Desktop installation and configuration.

- In MFG/PRO eB, information required for locale-specific formatting is derived from records stored in the label master table in the database. In MFG/PRO eB2 and higher, this information is retrieved from two external data files: `locale.dat` and `encoding.dat`.
- MFG/PRO eB NetUI menus were renamed to Desktop in MFG/PRO eB2 and higher:
 - NetUI Security Menu (36.3.21) became Desktop Security Menu.
 - NetUI Security Control File (36.3.21.24) became Desktop Security Control.
 - NetUI Menu (36.20.10) became Desktop Menu.
- Other menus have changed, particularly the menu items on UI: Manager Functions (36.20) in MFG/PRO eB2 and functionality on the Security menu in MFG/PRO eB2.1. In this case, the latest number is displayed first, followed by the previous number as in the following examples:
 - User Tool Maintenance (36.20.4; 36.20.2 in MFG/PRO eB)
 - User Maintenance (36.3.1; 36.3.18 in MFG/PRO eB2)

▶ See “Language Reference” on page 142.

▶ See Chapter 12, “Configuring Optional User Settings,” on page 219.

▶ See Chapter 12, “Configuring Optional User Settings,” on page 219.

Conventions

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

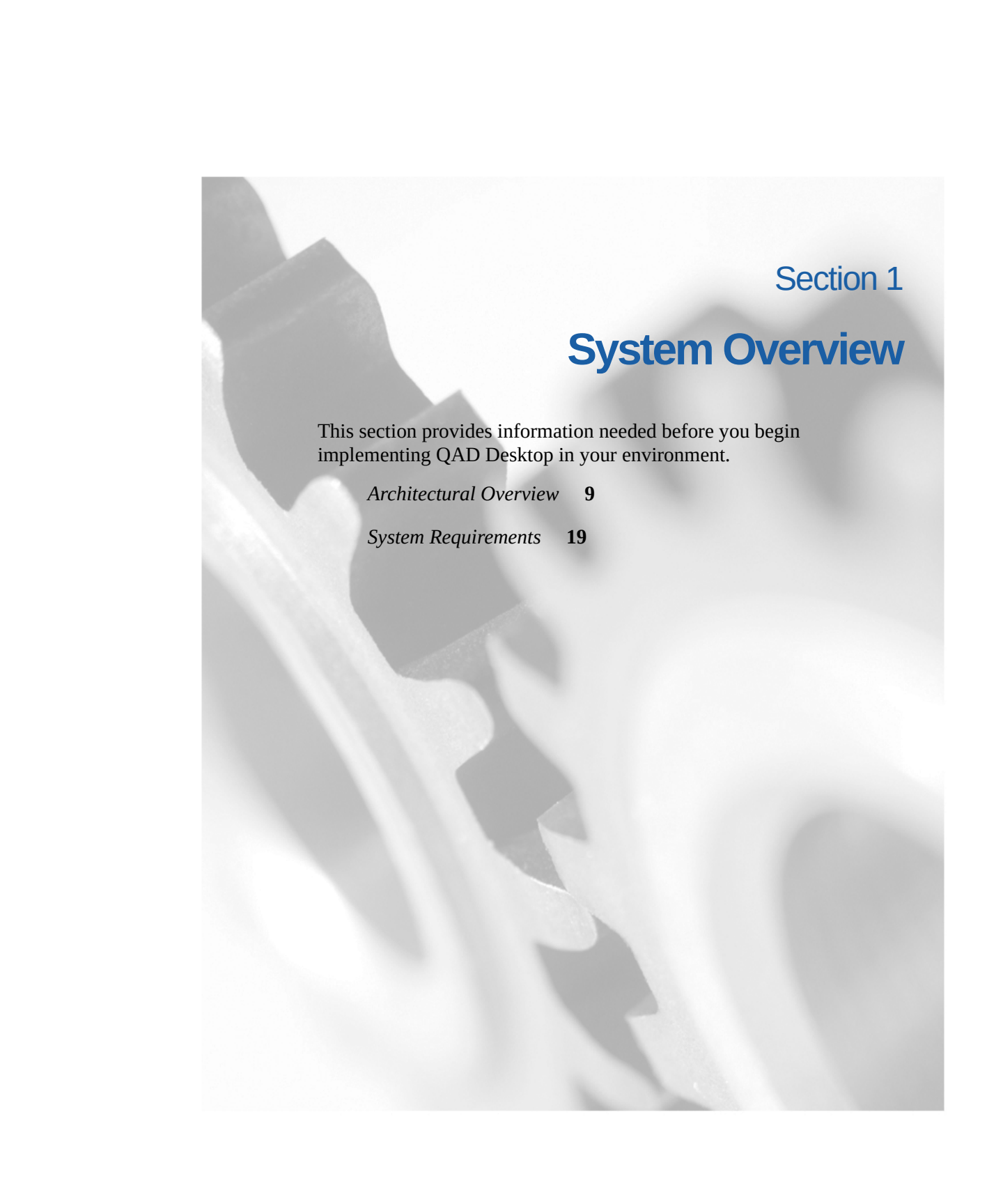
If you see:	It means:
monospaced text	A command or file name.
<i>italicized monospaced text</i>	A variable name for a value you enter as part of an operating system command; for example, <i>YourCDROMDir</i> .
indented command line	A long command that you enter as one line, although it appears in the text as two lines.

If you see:	It means:
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

QAD Desktop has a wide variety of configuration possibilities, is highly scalable, and can be customized easily. While this guide provides basic installation and configuration information, it cannot consider all of the possible computing environments and variations into which the Desktop can be implemented.

To take full advantage of the Desktop's flexibility and potential 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.

The background of the page is a grayscale image of several interlocking gears. The gears are of different sizes and are positioned in a way that they appear to be meshing together. The lighting creates highlights and shadows on the teeth of the gears, giving them a three-dimensional appearance. The overall tone is technical and industrial.

Section 1

System Overview

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

Architectural Overview **9**

System Requirements **19**

The background of the page is a grayscale photograph of several interlocking mechanical gears. The gears are of different sizes and are positioned in a way that creates a sense of depth and complexity. The lighting is soft, highlighting the metallic texture and the sharp edges of the gear teeth.

Chapter 1

Architectural Overview

This chapter provides the background information required to implement a new QAD Desktop environment. You should read this chapter *before* you begin any phase of a new implementation.

Implementation Overview **10**

Desktop Components **10**

Planning an Installation **15**

Implementation Overview

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

Important Beginning with Release 2.5 of QAD Desktop, the installation process has completely changed. Even if you have previous experience installing Desktop, you must read the instructions in this guide to ensure a successful implementation.

Desktop Components

QAD Desktop combines a variety of Web-based technologies:

- HTML/DHTML pages with multiple frames display the menu system and MFG/PRO programs.
- Cascading style sheets (CSS) define colors, fonts, styles, and images.
- A Java applet runs persistently and manages the interaction between the user and the interface.
- JavaScript dynamically generates HTML with applet tags.
- Perl scripts support a common gateway interface (CGI)-based search.
- Progress WebSpeed agents manage the interaction between the MFG/PRO database and Java browses and lookups using HTTP.
- Java servlets and server pages (JSP) manage the interactions between Desktop and the Progress character code using XML messaging between the r-code and client.
- Cookies are used to store information about the user.

The following sections provide brief descriptions of the Desktop components. These descriptions are intended to provide a framework for understanding the Desktop architecture before you begin your implementation. Specific details on different aspects of the Desktop components can be found in the appropriate chapters of this guide.

For example, for information on the system requirements for a specific component—such as version, disk space, and memory—see Chapter 2, “System Requirements,” on page 19.

Prerequisite Components

These components must exist in your environment before you begin implementing the Desktop. This guide assumes their existence and does not provide installation and configuration instructions for them. You must provide detailed information about the following components, including installation directories, versions, and network port number usage:

- Progress software components needed for MFG/PRO
- Proper version of MFG/PRO (latest service pack recommended)
 - MFG/PRO eB on Service Pack 4 or greater
 - MFG/PRO eB2
 - MFG/PRO eB2.1
- Internet access

▶ For more information, see the MFG/PRO installation guide for your system.

Third-Party Components

These components are supplied by software vendors or open source organizations other than QAD. This guide provides required information about installing and configuring the following components for use in a Desktop environment:

- Progress Servers

In addition to the Progress software required for MFG/PRO, the Desktop uses the Progress NameServer, AdminServer, and WebSpeed server products.

- Java

The Desktop uses the Java Development Kit (JDK) to enable communication between the client and the Tomcat servlet container. You may need to download the JDK from the Sun Microsystems Web site.

- Web Server

The Desktop uses a Web server to enable user access to the Desktop. The Desktop can operate with any CGI-compliant Web server. Examples include iPlanet Web Server, Microsoft Internet Information Server (IIS), and Apache HTTP Server.

- Web Browser

Each Desktop client machine must have a browser to access the Desktop. Desktop can be used only with Microsoft Internet Explorer; Netscape browsers are not supported.

- Telnet Server

The Desktop uses a telnet server for two purposes:

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

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 Desktop installation media.

▶ See “Windows Telnet Server Software” on page 14.

QAD-Supplied Components

The Desktop installation media contains various components needed to successfully install, set up, configure, and use the Desktop.

Table 1.1 lists the components included on the Desktop CD and gives a summary of their purpose.

Table 1.1
Desktop
Components

Directory Component	Description
desktop.war	The Web archive file containing the Desktop application files and directories
install.ksh	UNIX setup script
install.pl	Perl setup script called by install.sh or install.bat
install.exe	Windows setup executable
\zips\cgi.zip	Executable and archive files copied to the Web server cgi-bin directory
\zips\pro.zip	Modified Progress files needed to use Desktop
\zips\tomcat.zip	Tomcat LE Servlet/JSP Container
\zips\web.zip	Desktop HTML, Java, help, and image files

Directory Component	Description
\AIX	AIX operating system-specific configuration files
\HP - UX	HP-UX operating system-specific configuration files
\Linux	LINUX operating system-specific configuration files
\OSF1	OSF1 (TRU64) operating system-specific configuration files
\SunOS	Sun operating system-specific configuration files and WinZip utility
\Tools	Georgia SoftWorks telnet server software for Windows, Perl, and UnZip source
\win32	WinZip for Windows and multi-processor server executables

The following components are also supplied on the Desktop media. This guide provides installation and configuration information for these components.

Java Runtime Environment

The Java Virtual Machine (JVM) executes the platform-independent Java bytecode on a specific operating system. Desktop clients require a more recent version of the JVM than the browser default. Depending on the client machine configuration, the Desktop client startup process may automatically install a compatible Java Runtime Environment (JRE), which includes the Sun-developed JVM called the Java Plug-in.

Important If applications on the client machine require a specific version of the Java Plug-in, see “Desktop Java Plug-in Version Control” on page 130.

Servlet Engine

Desktop uses the Tomcat Servlet/JSP container to manage client connections between the Web server and the Desktop client sessions. Tomcat provides client state management for the Desktop HTML interface. Use the version on the Desktop media; otherwise, see “Install a New Tomcat Servlet Engine” on page 189.

Desktop Java Code

Desktop Java code is contained in the `multinet.jar` Java archive file. This file is copied to the Web server document directory during the Web server component deployment.

Archive files are managed by the JRE and are automatically downloaded when the version 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. Archive files are located in the user's Windows profile directory in a subdirectory called `java_plugin_AppletStore`.

Windows Telnet Server Software

The Desktop installation media contains Georgia SoftWorks Windows Telnet Server software. The telnet server enables Desktop clients to access a subset of MFG/PRO programs and any custom programs you have added. It is also used for the telnet sessions associated with HTML maintenance programs, reports, and inquiries.

The Georgia SoftWorks server is fully integrated with MFG/PRO telnet session functionality and works seamlessly in the Desktop environment. The server lets you monitor client sessions as well as detect and eliminate abnormally terminated client sessions.

For more information, access the Georgia SoftWorks Web site for a copy of their *Windows NT/2000 Telnet Server Software User Manual*:

<http://www.georgiasoftworks.com>

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 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/>
 - Log4j, a logging API for advanced logging options in Desktop 2, part of the Jakarta Project, sponsored by the Apache Software Foundation
<http://jakarta.apache.org/log4j/docs/index.html>
- ◆ See “Configure Logging” on page 192 for details.

Planning an Installation

If you have experience installing a version of Desktop earlier than 2.5, numerous significant changes have occurred to streamline the process. Review this installation guide prior to starting the install and carefully follow the instructions. Do not attempt a Desktop installation in a production environment until you have first installed and tested Desktop in a test environment.

Deployment Options

The various components of MFG/PRO and the Desktop can be deployed in several configurations. You can install all of the components on a single adequately sized server in a single-tier deployment or on multiple servers in a multi-tier deployment. Both deployments offer advantages for different MFG/PRO environments.

Note “Plan a Multi-Tier Deployment” on page 166 has complete details on how you can configure Desktop in various network scenarios.

To support these deployment options, this guide documents two types of Desktop installation:

- ▶ See page 39.
 - A new single-tier installation on one central database server
- ▶ See page 69.
 - A new multi-tier installation across two or more servers

The installation and configuration steps for these two scenarios differ and are presented in two separate chapters.

Important If you are upgrading from a pre-2.5 Desktop release, do not install the new release into directories containing components of an earlier release.

See “Install a New Desktop Release” on page 156 for an outline of issues to consider during an upgrade. Also review the Desktop Release Notes included on the installation media.

How to Use This Installation Guide

The entire installation process is documented in this guide. Each stage of the deployment has a corresponding chapter or section in the book. The installation guide walks you through the deployment of the Desktop components, the generation of required script files, the entry and validation of system data, the Desktop code compile, and the distribution of configured code and scripts managed by these tools.

This guide shows the deployment and configuration steps as they would most commonly be followed in a typical environment. In the chapters where step-by-step instructions are provided, field descriptions and detailed explanations are omitted. Instead, cross-references are provided to other sections of the installation guide where the detailed information can be found. If you require additional details about any menu items, functions, or field descriptions, refer to the pages indicated in the cross-references.

- ▶ “QAD Desktop Quick Start” on page 66.

Note If you have previous experience installing QAD Desktop 2 in a single-tier environment, you can go directly to the Quick Start section for a summary of installation steps.

Installation Overview

The Desktop installation consists of several sequential steps. Figure 1.1 shows a high-level work flow with each step.

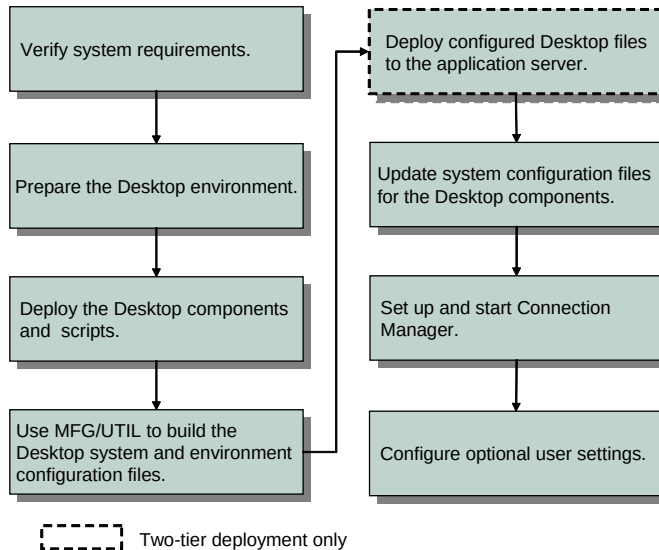


Fig. 1.1
Desktop
Installation and
Configuration
Work Flow

Installation Tools

Three tools are used to deploy and configure Desktop:

- Perl deployment scripts

Perl scripts manage the deployment stage of the Desktop installation. This includes deployment of applications, application code, HTML and archive files, and some script components. The Perl scripts are launched from the installation media, or from the directory where the installation media image was copied.

- MFG/UTIL

MFG/UTIL is the standard QAD tool for completing MFG/PRO and related application installations. You use it to compile Desktop code, generate Desktop scripts, and manage files on the database server. In multi-tier configurations, MFG/UTIL is used to customize and package HTML, script, and Java files for the Tomcat, WebSpeed, and Web server located on the application servers.

Refer to the MFG/PRO installation guide for your system for navigation and interface information for MFG/UTIL.

- Configuration .ini files

Configuration .ini files facilitate the deployment and configuration steps managed by MFG/UTIL. Each Desktop system has a corresponding file. This file is automatically created and placed in the log files directory you indicate. This file holds the configuration and deployment information used by the Desktop deployment tools to configure and build the Desktop system. It is subsequently updated with new information each time you run any deployment or configuration activity. It must be accessible by each deployment or configuration activity.

The file name is based on the name you specify for the system you are configuring, followed by _dt2.ini. For example, if you use the default deployment name, which is desktop, the configuration file for your deployment is desktop_dt2.ini.

System Requirements

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

Prerequisite Requirements **20**

Client Requirements **20**

Web Server Requirements **23**

Progress Server Requirements **25**

Windows Telnet Server **26**

Prerequisite Requirements

Database Server

The database server requirements for Desktop are the same as the requirements for the MFG/PRO version you are using. The following list shows the minimum requirements for QAD Desktop. For complete MFG/PRO requirements, refer to the MFG/PRO installation guide for your system.

Tip
Progress components should be updated to the latest patch level.

- The required Progress 9.1 software components for MFG/PRO, including:
 - Latest Progress version-specific patches
 - Progress AdminServer 9.1
 - Progress NameServer 9.1
 - Progress WebSpeed 3.1

Note If you are completing a multi-tier deployment and do not have WebSpeed installed on the application server, you can download and install the free WebSpeed Messenger.

- MFG/PRO eB Service Pack 4 or higher (latest service pack recommended)
- MFG/PRO eB2 and eB2.1 (latest service pack recommended)
- Internet access

▶ See “Install and Configure WebSpeed Messenger” on page 178.

Client Requirements

Operating Systems

The following operating systems are supported:

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

Minimum Client Hardware

These recommendations assume that the client machine will be used solely for the Desktop environment. These recommendations do not take into account any additional requirements for other client machine applications.

- Memory (RAM):
 - Windows 98 and ME: 64 MB minimum (128 recommended)
 - Windows NT, 2000 and XP: 128 MB minimum (256 recommended)
- P400 or faster processor (P800 recommended)
- At least 150 MB of disk space, depending on the version of Java, including 20 MB reserved for static HTML files being cached 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 Desktop display properties.

Client Software

Client Browser Requirements

- Desktop supports only Microsoft Internet Explorer version 5.5 with Service Pack 2 or later. The latest 6.0 version is recommended.
- Adobe Scalable Vector Graphics (SVG) plug-in 3.0 build 76 or newer for Desktop Process Editor. This component is copied to the Web server during the Desktop installation. It is downloaded by clients when needed.
- Java Runtime Environment (JRE): Any 1.3 version above 1.3.1_04.

Note Some 1.3 versions have a limitation that may affect the initial client startup process; see “Desktop Java Plug-in Version Control” on page 130 for details.

If your client machines are not currently using a 1.3 version of the Java Plug-in, an option on the MFG/UTIL build screen lets you continue using other versions of Java for client applications other

Tip
JRE includes the
Java Plug-in.

than QAD Desktop, but forces the Desktop client to use the QAD-provided version. For specific information, see “Use QAD Desktop Supplied Java Plugin” on page 209.

Client Browser Settings

With a few exceptions, Desktop assumes that default browser settings are in use.

Note To reset default browser settings, choose Tools|Internet Options|Advanced; then click Restore Defaults, followed by OK. This may affect other applications on the client machine.

The following is a list of non-default browser settings required:

- Temporary internet files should be set to check for newer versions of stored pages automatically.

To change this setting, select Tools|Internet Options|Temporary internet files|Settings. In the Setting window, select Automatic and click OK to make the change.

- Set disk space for temporary internet files to at least 100 MB.

To change this setting, select Tools|Internet Options|General, then click Settings in the Temporary Internet Files frame.

- When using a proxy server, disable access to the proxy for internal IP addresses. The Desktop connects to the proxy server using the settings defined for your browser.

The following is a partial list of required security defaults:

- You must have Run ActiveX controls and plug-ins enabled.
- Both cookie options must be enabled:
 - Allow cookies that are stored on your computer
 - Allow per-session cookies (not stored)
- In the miscellaneous security options, submit non-encrypted form data must be enabled. This is used for submitting search strings.
- All three scripting options must be enabled: active scripting, allow paste operations via script, scripting of Java applets.

Desktop uses colors and fonts to distinguish various screen elements such as menus and input frames. Setting any of the accessibility options may adversely affect screen appearance.

Web Server Requirements

The Web server is typically on the same machine as the Tomcat engine. The following requirements assume that both components are on the same server.

Operating Systems

Desktop clients communicate with the Web server using network protocols and do not require a specific operating system on the server. Use the operating system that best suits your needs based on criteria such as your experience or familiarity with the operating system, performance needs, and available support.

Software

- Any Web server that supports a common gateway interface (CGI)
Use the latest general release version of the product you choose.
Examples include:
 - iPlanet Web Server
 - Microsoft Internet Information Server (IIS)
 - Apache HTTP Web Server
- Any other software requirements for your Web server, including any Java requirements
- Progress WebSpeed messenger executable for multi-tier deployments

Important Java 2 SDK 1.4. for the Tomcat engine.

▶ See “Configure the WebSpeed Messenger” on page 179.

Hardware

Minimum hardware requirements differ depending on the Web server software and operating system you implement. The performance and scalability of the Desktop is directly related to proper server sizing. If you are unsure or inexperienced in performance and capacity planning, obtain professional assistance before purchasing any hardware.

Software and Hardware Sizing

The following Desktop components are most affected by Web server hardware resources:

- The Web server software. See the software vendor's sizing guidelines for your Web server's operating system.
- The Progress WebSpeed software. See the Progress WebSpeed documentation for information about your Web server's operating system.

CPU Requirements

The required number of central processing units (CPUs) and their speed depends on the type of hardware and operating system you are implementing. See the hardware vendor's product documentation and the operating system documentation for sizing guidelines.

Disk Space Requirements

The Web server must have enough disk space for the following Desktop components.

Table 2.1.
Disk Space
Requirements

Desktop Component	Disk Space
Web server software	Required disk space depends on the server software you implement
WebSpeed software	100 MB
Desktop software	150 MB for the Desktop Java code and HTML files

Progress Server Requirements

Operating Systems

Desktop 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.

Desktop supports the following platforms:

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

Tip

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

Java Requirements

Depending on your Progress server's 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.

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

Hardware

The Progress servers are responsible for executing the MFG/PRO code for the Desktop clients. The machine on which these servers run should be sized in a manner similar to a server running MFG/PRO for an equivalent number of MFG/PRO character clients.

♦ See the System Requirements chapter in the MFG/PRO installation guide for your system.

Servers

See “WebSpeed Agent Sizing and Behavior” on page 181.

- Progress NameServer Enterprise Edition, Version 9.1
- Progress AdminServer 9.1
- Progress WebSpeed Transaction Server, Version 3.1, with sufficient WebSpeed agent licenses to support expected transaction volume

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

Windows Telnet Server

The Georgia SoftWorks Windows NT/2000 Telnet Server is required for Windows telnet server implementations. This component is provided on the Desktop installation media. Even so, download the latest version of the software to obtain the most recent patches and functionality. Download the software from the Georgia SoftWorks Web site:

<http://www.georgiasoftworks.com/>

Refer to the Georgia SoftWorks documentation for software and system sizing requirements.

Georgia SoftWorks Licensing

As a QAD customer, you receive a 100-user server license when you register your telnet server at no cost. Additionally, you are entitled to 60 days of free updates from Georgia SoftWorks beginning on the day you register your software.

Note To be entitled to future updates or upgrades after the initial 60-day period, you must purchase a subscription plan directly from Georgia SoftWorks. If you do not choose to purchase a subscription plan, you must pay for any upgrades after the initial 60 days.

Contact Georgia SoftWorks directly for additional information at:

<http://www.georgiasoftworks.com/>

For complete installation and registration procedures, see “Install the Telnet Server” on page 36.

The background of the page features a grayscale image of several interlocking gears. The gears are of different sizes and are positioned in a way that creates a sense of depth and mechanical complexity. The lighting is soft, highlighting the teeth of the gears and casting gentle shadows.

Section 2

Implementation

This section provides instructions for implementing QAD Desktop in your environment.

Preparing the Desktop Environment **29**

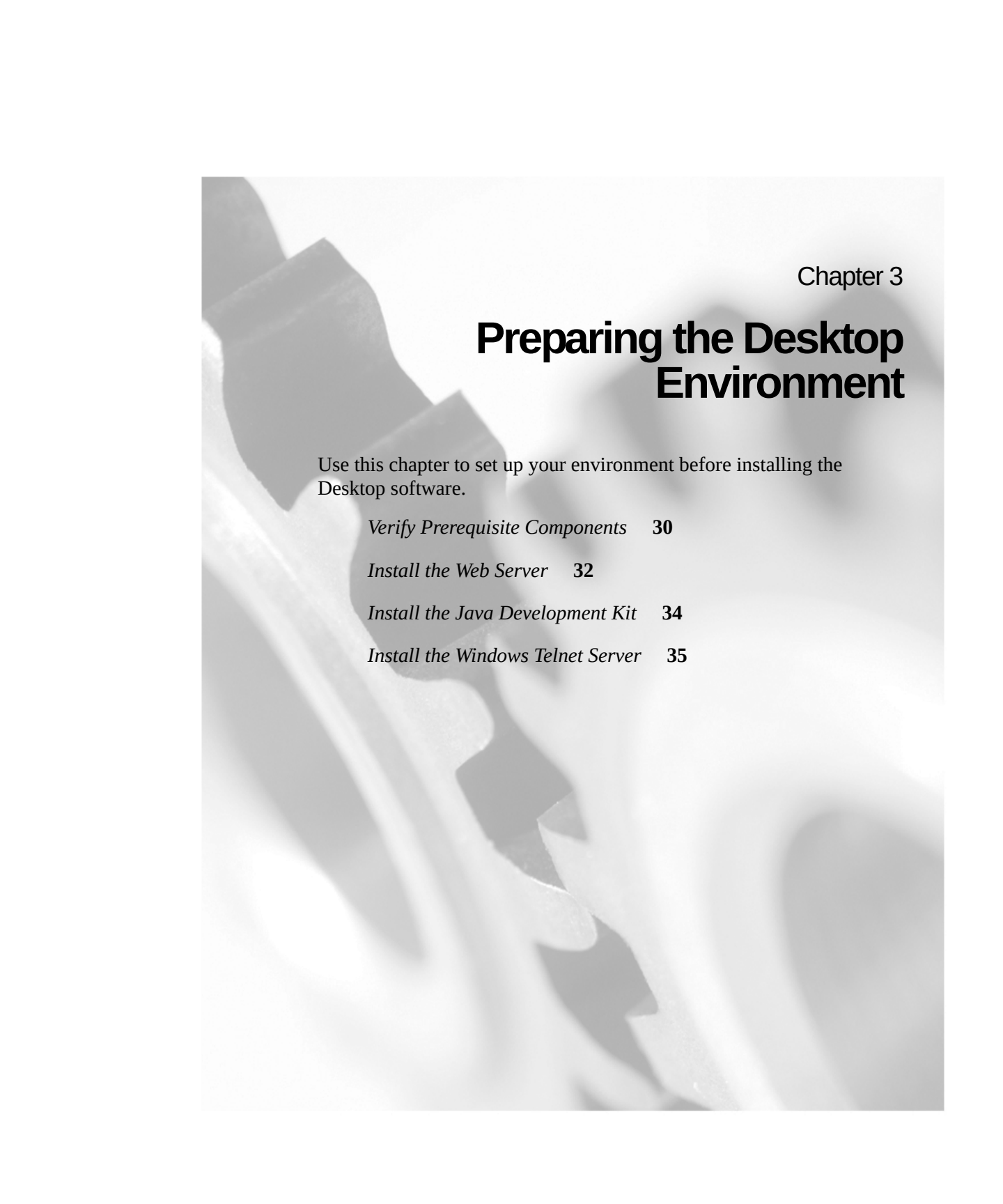
Deploying Desktop on Single-Tier **39**

Deploying Desktop Multi-Tier **69**

Configuring Desktop After Deployment **97**

Configuring Connection Manager **111**

Implementing Multiple Languages **133**

The background of the page is a grayscale image of several interlocking gears. The gears are of different sizes and are positioned in a way that they appear to be meshing together. The lighting creates highlights and shadows on the teeth of the gears, giving them a three-dimensional appearance. The overall tone is technical and industrial.

Chapter 3

Preparing the Desktop Environment

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

Verify Prerequisite Components **30**

Install the Web Server **32**

Install the Java Development Kit **34**

Install the Windows Telnet Server **35**

Verify Prerequisite Components

This installation guide assumes you have already installed Progress and have a running MFG/PRO database and a fully functional character client. You must provide or verify information regarding these components, including installation directories, PROPATH information, and server names.

Progress

Prerequisite components:

▶ See the Progress installation and configuration documentation.

- Complete Progress 9.1 installation

Make sure that you performed a complete installation of the Progress components by choosing the complete installation option.

- NameServer Enterprise Edition
- WebSpeed Enterprise Server

If you are using a local host connection for Desktop, WebSpeed must be installed on the MFG/PRO database server. For client/server connections, you can install the WebSpeed software anywhere.

- AdminServer

Verify the 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:

Tip
Optionally use the WSMAdmin to validate WebSpeed components.

- 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
```

Tip

Databases must be running to start WebSpeed.

MFG/PRO Database and Character Client

Prerequisite components:

- MFG/PRO eB Service Pack 4 or higher (latest service pack recommended)
- MFG/PRO eB2 or eB2.1 (latest service pack recommended)

▶ See the MFG/PRO installation guide for your system.

Verify MFG/PRO Components

If not already started, start your MFG/PRO database server and launch a character client session. Log in and make sure the character client is functioning.

Verify Language Compatibility

Desktop is configured to support a defined set of languages and associated code pages, based on the underlying requirements of the MFG/PRO database. The supported set has changed over time.

▶ See Chapter 8 on page 133.

If you are installing Desktop into a non-US English environment, verify that your database is using a currently supported code page as listed in Table 8.1, “Supported Progress Code Pages,” on page 135.

Important For non-US English installations, it is especially important that you have applied the latest MFG/PRO service packs to ensure the most up-to-date fixes for language-related issues.

If you do not have the latest service pack installed, be sure you carefully review the Desktop Release Notes for any language-related updates. These may contain instructions for additional steps required to ensure a successful installation. Note the following changes:

- In MFG/PRO eB, some of the label data used to map code pages has changed. You can verify the values in your database with the list in Table 8.4 on page 148.

- In MFG/PRO eB2, changes have been made to the file that manage code page mapping (`encoding.dat`).

Install the Web Server

Prerequisites

- Any CGI-compatible Web server
- Web server installation and user documentation
- Internet access to download any Web server updates

Installation and Setup Overview

Use the installation instructions provided by the Web server software vendor to install the Web server.

During the Web server installation process, you define a Web server host name and port number, and may create a Web server document directory and executables directory. You must enter this information during a later step.

▶ See “Sample Web Server Setup” on page 170.

If your Web server does not have a document directory, you must create one. This is the directory that will contain the Desktop Java, HTML help, and client files. For example, on an Apache Web server, the document directory is `\apache2\htdocs`.

The `cgi-bin` directory is the location on the server where most Web servers look for the CGI scripts and other executables.

myhost.mydomain.com/cgi-bin

Note Netscape Enterprise Server may use the `lcgi-bin` directory.

If an executables directory and alias do not already exist, you must create them. It is highly recommended that you use the default executables alias, `cgi-bin`. For example, on an Apache Web server, the executables directory is typically `\apache2\cgi-bin`.

While many Web server packages use the name `cgi-bin` for this directory, some use other names. If your Web server uses a different name, you need to do one of the following:

- Rename the directory to `cgi-bin`.
- Create a `cgi-bin` alias for the directory.

Using UNIX Web Servers

Install the Web Server

The UNIX Web server must be installed as root to bind to TCP ports 1-1023. The default port for UNIX and Windows Web servers is port 80.

Set Up the MIME Type

Web servers insert Multi-Purpose Internet Mail Extension (MIME) types in a header at the beginning of any Web transmission. Clients use this header to select an appropriate application for the type of data transmitted.

If you have a UNIX Web server, use the following instructions to enable clients to associate the proper application when downloading executable files. Depending on which server you are using, you edit different files. Refer to your Web server's documentation for details.

- 1 Remove `.exe` from `magnus-internal` and any other MIME-type.

```
type=magnus-internal/cgi      exts=cgi,bat
```

- 2 Add `.exe` to the `application/octet-stream` MIME-type.

```
type=application/octet-stream exts=bin,exe
```

Note When you are using an Apache Web server and the MIME types are not defined correctly, users may have problems displaying the Desktop HTML help when they click the question mark icon. Instead of displaying the help, the system prompts to download the file `mfwb01a.p`.

To correct this problem, edit the `ApacheInstallDir/conf/mime.types` file and comment out the following line:

```
application/x-sh          sh
```

Verify the Web Server

Verify that the Web server is functioning by starting a Web browser on a client machine. Then, enter a URL referencing only the Web server name. For example, enter the following and verify that there is a response.

`http://www.MyWebServerURL.com`

If this test fails, follow the troubleshooting steps outlined in your Web server's documentation.

Install the Java Development Kit

Prerequisites

▶ See “Java Requirements” on page 25.

- Internet access (to download J2SDK)
- J2SDK 1.4.x
- UNIX: Access to log in and install the J2SDK as root user

Overview

If a JDK version prior to JDK 1.4 was installed when you installed Progress and your Progress server and Web server are on the same machine, do not uninstall this JDK version. Instead, install the latest version in the default JDK installation directory. In most circumstances, the latest JDK version should be installed with Progress.

Tip
Check your server manufacturer's Web site for any server-specific custom or updated J2SDK versions.

Download J2SDK for your Web server from the Sun Microsystems Java Web site:

www.java.sun.com

The download time varies depending on your connection speed. The J2SDK installation file is approximately 44 megabytes.

Installation

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.

Set JAVA_HOME Variable

Desktop components such as Tomcat and Connection Manager require that the `java_home` variable be set correctly on your system. You can set this with operating system commands, although it typically is set in startup scripts.

In UNIX environments, use this command:

```
export JAVA_HOME=/pathtoj2sdk
```

In Windows environments, enter this command at the DOS prompt:

```
set java_home=c:\pathtoj2sdk
```

Note You can add this set command to the `catalina.bat` or `catalina.sh` file to automatically set the `JAVA_HOME` variable.

Record the `JAVA_HOME` information, You will need to verify it in a later step.

Verify the J2SDK

At the UNIX or DOS prompt, enter:

```
java -version
```

The system should display information about the current Java version. Make sure this is the correct version. If you receive an error message, make sure your `PATH` environment variable is set correctly.

Install the Windows Telnet Server

To support telnet connections on Windows servers, you must install the telnet server provided with Desktop.

Prerequisites

- Latest release of the Georgia SoftWorks Windows NT/2000 Telnet Server (GSWTS)

Highly recommended:

- Web access for downloading the latest telnet server and other patches.
- The `Kill.exe` and `Pulist.exe` Windows utilities. These utilities let you monitor and remove `_progres.exe` processes from the Windows telnet server without shutting down the server.
 - For Windows NT, you can find these utilities on the Resource Kit for Windows NT.
 - For Windows 2000, you can find `Pulist.exe` on the Windows 2000 Server Resource Kit Supplement 1. The `Kill.exe` utility is part of the Windows 2000 support tools.
 - You may also be able to download these utilities from the Microsoft Web site.
- Georgia Softworks Power Features Pack (included in general release).

This pack provides the Session Monitor, which is highly recommended for troubleshooting telnet connection issues. Review the GSWTS documentation.

Install the Telnet Server

- 1 If you are using the QAD-provided version, copy the release-specific zip file from the Desktop CD to a local drive. The zip file is in the `\tools` subdirectory on the Desktop media.
- 2 Unzip the source file you downloaded from the Web or copied from the Desktop media to a work directory. Double-click `setup.exe` to begin installation.
- 3 At the Welcome window, click Continue.
- 4 Select Full Install and click Continue.
- 5 Enter the GSWTS installation directory and choose Continue. Make sure to use the default `\gs_uts` installation directory. You can place this directory on any local drive that can be accessed through the network.

Note If you do not install GSWTS on the system drive—in most cases the C: drive—make sure that you have full access to the drive and to the GSWTS installation directory and subdirectories.

- 6 When GSWTS has been successfully installed, the Setup Succeeded window displays and GSWTS is started automatically.
- 7 Double-click the Registration icon in the GSWTS program group.

Register the Georgia SoftWorks Software

To register GSWTS, you must provide a product ID to Georgia SoftWorks so that a serial number can be generated for your product. This serial number is necessary to complete the GSWTS registration. Use the following instructions to register GSWTS.

Your serial number is based upon the product ID, which combines elements of the Windows server's hardware and software components. If these components are changed or upgraded, your GSWTS registration may no longer operate. If this occurs, 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 and you need to obtain new license codes from Georgia SoftWorks, be sure to indicate that you have registered previously. If you are a sales agent or a distributor, be sure to note that information on the registration as well.

- 1 In the Georgia SoftWorks Product Registration window, enter your customer information.

The information that displays in the Product Information section of the product registration window is automatically generated by the system.
- 2 Choose the Save to File button to save this information to a file, or choose print. Then, follow the appropriate step to supply the product ID to Georgia SoftWorks:
 - a If you printed your registration information, FAX it to Georgia SoftWorks at 706-265-1020. When your information is received, a serial number is generated for your product and is sent to you by FAX.
 - b If you saved your registration information to a file, e-mail the file to Georgia SoftWorks at the following address:

registration@georgiasoftworks.com

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

- 3** When you receive your serial number, enter it in the appropriate field in the registration information section and click Register.

Deploying Desktop on Single-Tier

This chapter discusses how the Desktop components are deployed by the deployment script, then configured using MFG/UTIL in a single-tier deployment.

Single-Tier Deployment Overview **40**

Deploy the Desktop Components **41**

Configure Desktop Components **50**

QAD Desktop Quick Start **66**

Single-Tier Deployment Overview

Installing and configuring the Desktop files on the Web server involves the following steps:

- Install HTML and Script Files
- Install the Tomcat Servlet Engine
- Install Desktop MFG/PRO Code
- Set Up Windows Telnet Environment
- Verify and Update System Information
- Build the Desktop System
- Generate WebSpeed Configuration Information
- Generate Telnet Connection Scripts
- Generate Connection Manager Log-In Information

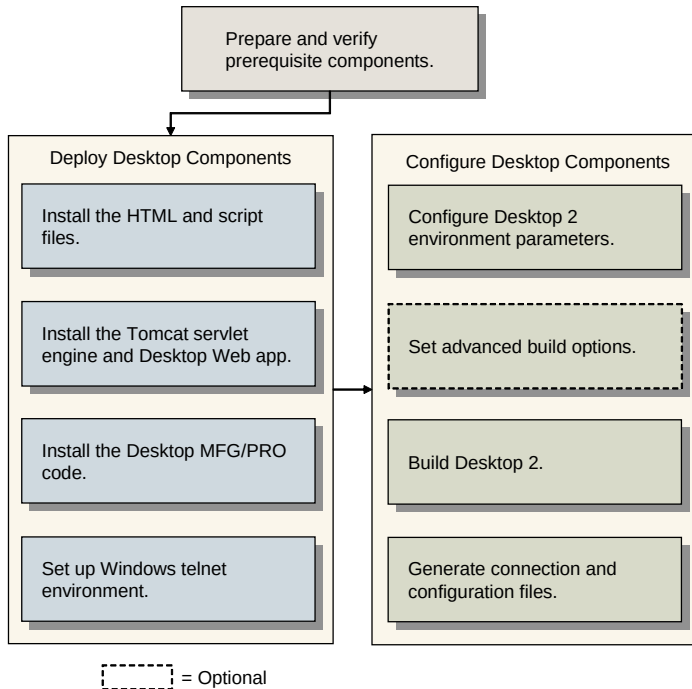
Figure 4.1 summarizes the deployment and configuration activities for a single-tier Desktop environment. Each topic has a corresponding installation section. Refer to that information for complete installation instructions. You may also be advised to see other installation reference material; make sure you have those materials before you begin that component of the installation.

For details about installing and configuring Desktop in a multi-tier environment, see Chapter 5, “Deploying Desktop Multi-Tier,” on page 69.

♦ See “QAD Desktop Quick Start” on page 66.

Note If you have experience with Desktop installations and are installing a single language, a Quick Start guide is provided at the end of the chapter. This summarizes the key steps in the installation process with cross-references to detailed explanations if they are needed.

Fig. 4.1
Single-Tier
Configuration
Work Flow



Deploy the Desktop Components

Prerequisites

- Latest QAD Desktop installation media, installation guide, Desktop release notes, and errata
- Your Web server user guide and installation documentation
- Windows: The latest Georgia SoftWorks Windows NT/2000 Telnet Server (GSWTS) release installed and tested

Mount the CD-ROM (UNIX Only)

- 1 Log on as a user that has write permission to the Web server, Tomcat, and MFG/PRO installation directories.
- 2 Mount the CD-ROM. Example commands are listed in Table 4.1.

Table 4.1
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 <code>man mount</code> to determine the correct command.

Launch Installation Script

Use these steps to start the installation process.

- 1 Launch the installation script. For UNIX:
`./install.ksh`
For Windows, run `install.exe` from Run on the Start menu.
- 2 You are prompted to enter a log file directory location. Typically, you should name the log file directory based on its associated MFG/PRO or Desktop release.

▶ See page 18 for additional log file details.

Each Desktop deployment should have its own log file directory, used for storing several log and support files. Later deployment and configuration processes look at this location for installation information stored in an `.ini` file.

Important Do not delete this directory or manually modify any files found here.

If the directory you specify does not exist, you are prompted to create it. You must also specify the name you want to use for your Desktop configuration, which is used in naming the log and `.ini` files.

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 'C:/mfgsvr'
->c:\mfgsvr

*** Note that each log directory applies to a single
MFG/PRO installation. Please see the Desktop install
guide for additional information.

Use c:\mfgsvr as log directory?
Default is 'y'
->y

```

- 3** When any application .ini files exist in this directory, they are displayed as selection options. In the following example, an .ini file already exists for a configuration named Test. 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>?

```

When no application .ini files exist, the following prompt displays:

```

No .ini was found in directory 'C:/logfiles'.
Do you want to start a new setup file (.ini)?
Default is 'y'
-> y

What is the application name for this Desktop 2 configuration?
Default is 'desktop'
->Production

```

Enter Yes to start a new application .ini file; then enter a name for this Desktop configuration or accept the default. Application names must be at least three characters, with no spaces. The name you specify is used as the .ini file name.

- 4** 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'
-> Yes

```

- 5** Specify this is a single-tier Desktop deployment.

```

*** Configuration: Production ***
*** Please select type of install. ***
1: Single-Tier
2: Multi-Tier (n-tier)

<1-2>? 1

```

▶ See Chapter 5 for multi-tier instructions.

6 The main menu displays.

```

*** Configuration:  Production ***
*** Main Menu ***
Please choose one of the following:
    1: Install HTML and CGI Files Menu
    2: Install Tomcat Files Menu
    3: Install MFG/PRO - Progress Desktop Files Menu
    4: Select a Different Configuration
    5: Exit
<1-5>?

```

Install HTML and Script Files

Use these steps to install the Desktop HTML and script files in the Web server installation directories.

- 1** On the main menu, choose 1 to access the HTML/CGI menu.
- 2** On the HTML/CGI menu, choose 1 to install the HTML and CGI files.

```

*** Configuration:  Production ***
*** HTML / CGI Menu ***
Please choose one of the following:
    1: Install HTML and CGI Files
    2: Uninstall HTML and CGI Files
    3: Return to Main Menu
<1-3>? 1

```

- 3** Verify the Web server port or enter a new port number.

```

Please enter the Web Server's port number
(usually this is 80).
Default is '80'
-> 80

```

- 4** Enter and confirm the Web server document root directory.

```

Please enter the Web Server's document root directory.
Default is '/apache/htdocs'
->c:\apache2\htdocs

```

- 5** Specify the subdirectory where you want to place the Desktop document files.

```

Please enter the name of the directory under
'c:\apache2\htdocs' where the html files will be placed.
Usually this is the same as the configuration name.
Default is 'Production'
-> Production

```

- 6 Enter and confirm the directory under the Web server where the script files should be placed, typically the `cgi-bin` directory.

Note Even though the prompt says Ok to overwrite?, no files in this directory are actually overwritten; files are only added.

```
Please enter the full path to the directory where the cgi-bin
files will be placed for this configuration.
Default is 'C:\apache2\cgi-bin'
-> C:\apache2\cgi-bin

The directory 'C:\apache2\cgi-bin' exists.
Ok to overwrite?
Default is 'y'
-> y
```

- 7 Enter the ScriptAlias for the `cgi-bin` directory.

```
Please enter the ScriptAlias which refers to C:\apache2\cgi-
bin. Generated HTML links will use this ScriptAlias name to
point to the cgi-bin directory.
Default is 'cgi-bin'
-> cgi-bin
```

The files are extracted to the Web server directories and the correct permissions are set. Press Enter when prompted; then review the log for errors before proceeding.

- 8 Choose option 3 to return to the main menu.

```
*** Configuration:  Production ***
*** HTML / CGI Menu ***
Please choose one of the following:
    1: Install HTML and CGI Files
    2: Uninstall HTML and CGI Files
    3: Return to Main Menu
<1-3>? 3
```

Install the Tomcat Servlet Engine

Complete this section to install the Tomcat servlet engine included on the Desktop media and deploy the Desktop Web application file.

If you want to use a previously installed Tomcat version, enter the installation directory for that version when prompted for the Tomcat installation directory. The installation scripts validate that the existing installed version is compatible with the Desktop release being installed.

- When the existing Tomcat installation is compatible with the Desktop version being installed, you are prompted whether to only install the Desktop Web applications required by the Tomcat engine. The

existing Tomcat version is not overwritten; instead the Desktop Web application files are installed into the existing Tomcat directory structure.

- When the existing Tomcat installation is not compatible with the Desktop release being installed, the installation scripts will prompt whether to overwrite the existing version with the version included on the Desktop installation media.

▶ See page 196.

The following steps assume this is a new Tomcat installation.

Deployment steps differ slightly if you are reinstalling Tomcat over an existing installation, or are not reinstalling Tomcat and instead installing only the Web application files into the existing Tomcat directory structure. In either case, carefully review the installation script prompts to install only the components you need.

- 1 Shut down any existing Tomcat instances running on the server
- 2 On the main menu, choose 2 to access the Tomcat installation menu.

Note If the JAVA_HOME variable is not set, you will be prompted to set it before continuing.

- 3 On the Tomcat menu, choose 1 to install Tomcat.

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

- 4 Enter and verify the directory where you want to install Tomcat. To avoid future configuration issues, install Tomcat in its own directory.

You can use an existing Tomcat installation as long as it meets the Desktop criteria. To do this, specify the directory where Tomcat is already installed.

```
Please enter the Tomcat installation directory.
No default value
->c:\tomcat
```

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

5 Optionally enter a new Tomcat port number.

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

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 6.

See “Update Tomcat Port Numbers” on page 101.

The Tomcat installation files are extracted. The Tomcat `setenv.sh` and `setenv.bat` scripts are also updated with Desktop-specific environment information. Review the displayed installation messages for errors before proceeding.

6 Enter 3 to exit to the main menu.

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

You now continue by installing the Desktop MFG/PRO code.

Install Desktop MFG/PRO Code

Complete this section to install Desktop-specific code for MFG/PRO and updates to MFG/UTIL that are included on the Desktop media.

Note This set of steps adds program files to an existing directory; no files are overwritten.

1 From the main menu, choose 3 to access the MFG/PRO and Progress Desktop files menu.

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

See page 35.

2 Choose option 1 to begin installing the Desktop Progress files.

```
*** Configuration: Production ***
*** MFG/PRO - Progress Desktop Files Menu ***
  1: Install MFG/PRO - Progress Desktop Files
  2: Update Additional MFG/UTIL Installations
  3: Telnet Setup (Windows Only)
  4: Uninstall MFG/PRO - Progress Desktop Files
  5: Return to Main Menu
<1-5>? 1
```

3 Enter the MFG/PRO character client installation directory.

In a typical MFG/PRO system, you install the character-client files in the MFG/PRO directory. By default, this is:

- c:\mfgsvr\charcli for MFG/PRO eB
- c:\mfgsvr for MFG/PRO eB2 and eB2.1

```
Please specify the MFG/PRO Character Client directory.
Default is 'c:\mfgsvr'
->
```

4 Enter and confirm the MFG/PRO code directory. This directory name defaults to the Desktop configuration name being installed. For consistency when installing multiple Desktop systems, accept the default values.

```
Please specify the MFG/PRO Desktop code directory.
The Desktop Progress code files will be extracted here.
Default is 'C:\mfgsvr\Production'
-> C:\mfgsvr\Production
```

```
The directory 'C:\mfgsvr\Production' does not exist. Would
you like to create this directory?
Default is 'y'
-> y
```

The process is logged to the screen. Files are copied, permissions are set correctly, and MFG/UTIL is updated with Desktop workflows. You are prompted to press Enter as each step completes.

When the process completes, review the log for errors when the installation is complete.

5 The next step depends on your operating system.

- In UNIX environments, choose option 5 to return to the main menu. Then exit the deployment script and begin the configuration activities with MFG/UTIL. You can also choose option 3 to launch MFG/UTIL directly from this menu to begin those activities. Continue your UNIX deployment with “Configure Desktop Components” on page 50.

```
*** MFG/PRO - Progress Desktop Files Menu ***
Please choose one of the following:
```

- ```
1: Install MFG/PRO - Progress Desktop Files
2: Update Additional MFG/UTIL Installations
3: Launch MFG/UTIL
4: Uninstall MFG/PRO - Progress Desktop Files
5: Return to Main Menu
```

```
<1-5>?
```

- For Windows, choose option 3 to generate the telnet scripts as detailed next in the “Set Up Windows Telnet Environment” section.

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

## Set Up Windows Telnet Environment

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

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

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

- 2 Verify the Georgia SoftWorks Windows NT/2000 Telnet Server (GSWTS) installation.

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.

**Note** The default installation directory that Georgia SoftWorks now uses is `gs_uts`, not `gs_tnet`. If you used the Georgia SoftWorks default, be sure to change it here.

```

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

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

```

- 4 You may be prompted to confirm the MFG/PRO character client directory.

```

Please specify the MFG/PRO Character Client directory.
Default is 'c:\mfgsvr'
->

```

- 5 The system generates a `k_start.bat` file and places executable files in several directories. Note the directories before you continue.

```

Creating a k_start.bat file in C:\gs_tnet\scripts
Thu Feb 5 16:48:57 2004: Created
C:\gs_tnet\scripts\k_start.bat
Copying 'win32\imagecfg.exe' to
'C:\mfgsvr\telnet\imagecfg.exe'
Copying 'win32\setMP.cmd' to 'C:\mfgsvr\telnet\setMP.cmd'

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 Desktop files.

## Configure Desktop Components

**Important** Before you begin configuring the Desktop components using MFG/UTIL, start the MFG/PRO databases.

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

- Verify and Update System Information
- Build the Desktop System
- Generate WebSpeed Configuration Information
- Generate Telnet Connection Scripts

- Generate Connection Manager Log-In Information
- Print Desktop Information

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



**Fig. 4.2**  
MFG/UTIL  
Desktop Menu  
Options

**Important** If you are building Desktop for an Oracle system, you must create a copy of your .pf file before beginning this sequence of steps.

▶ See “Building Desktop for an Oracle System” on page 58.

## Verify and Update System Information

Complete this section to set the Desktop 2 build parameters.

- 1 Launch MFG/UTIL from the database server. Then choose Desktop|Configure Desktop Environment Parameters.

**Note** In UNIX environments, you can also launch MFG/UTIL from the deployment script menu.

▶ See step 5 on page 48.

- 2 If MFG/UTIL detects a new Desktop 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.

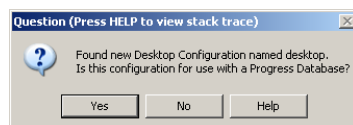
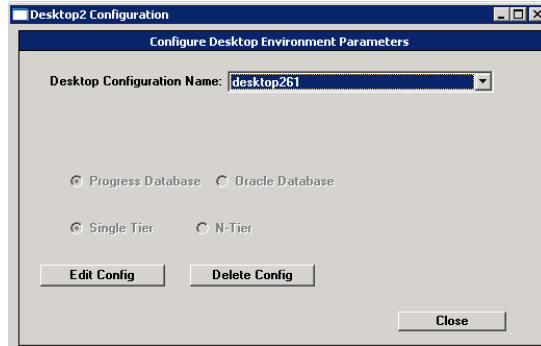


**Fig. 4.3**  
MFG/UTIL  
Database Type  
Prompt

- 3 The Desktop 2 Configuration screen displays. MFG/UTIL examines the system .ini files and provides the system names as selection options. Select the name of the system you are configuring; then choose Edit Config to review and update the build information.

**Fig. 4.4**  
Desktop 2  
Configuration  
Selection



- 4 The configuration screen displays.

**Fig. 4.5**  
Single-Tier  
Configuration  
Screen for Progress

|                                                                             |  |
|-----------------------------------------------------------------------------|--|
| Database Connection Parameter File with Path:                               |  |
| <input type="text" value="c:\eb2sp3\batch\batch.pl"/>                       |  |
| MFG/PRO Language Code:                                                      |  |
| <input type="text" value="us"/>                                             |  |
| Desktop Install Directory:                                                  |  |
| <input type="text" value="C:\eb2sp3\desktop261"/>                           |  |
| MFG/PRO Version:                                                            |  |
| <input type="text" value="eB2"/>                                            |  |
| Database Set Name:                                                          |  |
| <input type="text" value="Production"/>                                     |  |
| Service Pack Directory:                                                     |  |
| <input type="text" value=""/>                                               |  |
| Propath:                                                                    |  |
| <input type="text" value="C:\eb2sp3\desktop261...c:\eb2sp3.c:\eb2sp3\src"/> |  |
| Tomcat Operating System:                                                    |  |
| <input type="text" value="WIN32"/>                                          |  |
| Tomcat Hostname:                                                            |  |
| <input type="text" value="plli08.qad.com"/>                                 |  |
| Tomcat Port Number:                                                         |  |
| <input type="text" value="8080"/>                                           |  |
| Tomcat Directory:                                                           |  |
| <input type="text" value="c:\tomcat"/>                                      |  |
| Tomcat Webapp Name:                                                         |  |
| <input type="text" value="desktop261"/>                                     |  |
| Web Server Operating System:                                                |  |
| <input type="text" value="WIN32"/>                                          |  |
| Web Server Executables Directory:                                           |  |
| <input type="text" value="C:\Apache\Apache2\cgi-bin"/>                      |  |
| Web Server Scripts Alias:                                                   |  |
| <input type="text" value="cgi-bin"/>                                        |  |
| Web Server HTML Document Directory:                                         |  |
| <input type="text" value="C:\Apache\Apache2\htdocs"/>                       |  |
| Relative Path to Desktop HTML Document Directory:                           |  |
| <input type="text" value="desktop261"/>                                     |  |
| Web Server Hostname:                                                        |  |
| <input type="text" value="plli08.qad.com"/>                                 |  |
| Web Server Port Number:                                                     |  |
| <input type="text" value="80"/>                                             |  |
| WebSpeed Broker Name:                                                       |  |
| <input type="text" value="desktop261"/>                                     |  |
| WebSpeed Executable Name:                                                   |  |
| <input type="text" value="cgui.exe"/>                                       |  |
| WebSpeed Server Hostname:                                                   |  |
| <input type="text" value="plli08.qad.com"/>                                 |  |
| WebSpeed Server Port Number:                                                |  |
| <input type="text" value="80"/>                                             |  |
| Enable Java Console tracing:                                                |  |
| <input type="text" value="No"/>                                             |  |
| Use QAD Desktop Supplied Java Plugin:                                       |  |
| <input type="text" value="Yes"/>                                            |  |
| Maximum Search Results:                                                     |  |
| <input type="text" value="20"/>                                             |  |
| <input type="button" value="OK"/>                                           |  |
| <input type="button" value="Cancel"/>                                       |  |

See Figure 4.6.

See “Add Parameter File Directories” on page 217 for details.

See “Desktop Java Plug-in Version Control” on page 130 for details.

**Oracle:** The screen differs slightly depending on the server operating system and database type. Three additional fields display for Oracle database environments; ORASOPATHNAME is not used in Windows environments.

- a Choose the parameter file for your database in Database Connection Parameter File with Path.

This field displays all available .pf files found in your MFG/PRO and client installation directories. You can specify additional directories for MFG/UTIL to search for other .pf files by using the Desktop|Edit Search Directories for Desktop Configuration Lookups option on the MFG/UTIL menu.

Once you select a parameter file, default values are supplied for most of the configuration fields on the screen.

- b Review all field values and update any missing information. Use the field descriptions in “MFG/UTIL Reference” on page 202 to help you review and update the field values.
- c If you are installing into an environment that includes service pack files in separate files, enter the correct directory in the Service Pack Directory field.
- d Specify Yes to Use QAD Desktop Supplied Java Plugin to use the version of the plug-in supplied on the Desktop CD.
- e Use fully qualified server names for the Tomcat Host Name, Web Server Host Name, and the WebSpeed Host Server Name.

**Fig. 4.6**  
Configuration  
Fields for Oracle

Database Set Name: Training

Absolute Path to Oracle Install:

Oracle System Identifier Name:

ORASOPATHNAME: Not Applicable for Windows

Service Pack Directory:

Oracle-specific

- 5 Choose OK after entering and verifying existing information for all fields.
- 6 When the Desktop 2 Configuration screen redisplay, exit to the main menu.

## Build the Desktop System

You can optionally generate build scripts without actually completing the build at this time. This gives you more control over the build and can ease system administration activities in environments with multiple Desktop systems. You can create build scripts for all the Desktop systems without running the build and then create a batch or cron job to run the build for one or multiple Desktop systems at a later time when more system resources are available or during scheduled downtimes.

To generate the scripts only, choose the Advanced Build Options and select the Create Build Script Only option.

**Note** If you are building Desktop for an Oracle system, you must execute the build in two separate steps to prevent compile errors.

Use the following steps to verify the system information and build the Desktop system:

- 1 From the MFG/UTIL main menu, choose Desktop|Build Desktop.
- 2 In Desktop Configuration Name, choose the system to build.

See step 7 on page 57.

See “Building Desktop for an Oracle System” on page 58.

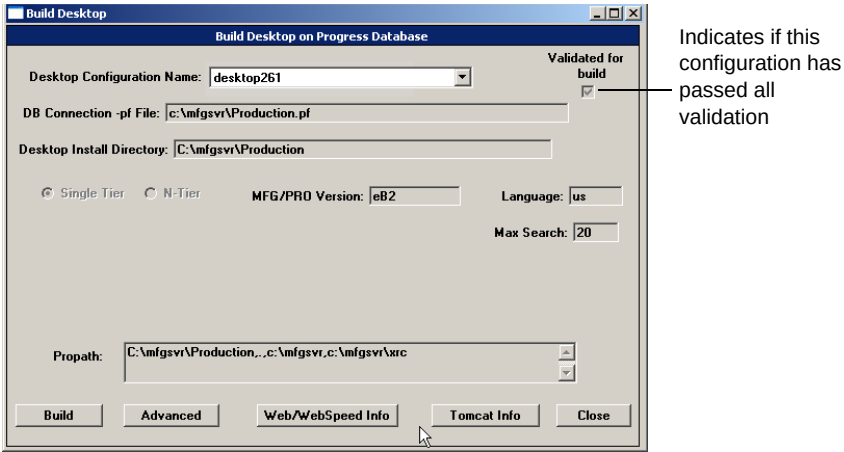
**Fig. 4.7**  
Select Desktop Configuration to Build

- 3 The fields are populated with the values for the system configuration you chose. Verify the build information by reviewing each set of fields before building your system.

See page 202 for field descriptions.

**Note** If Validated for Build is not checked, you can still build the system, but be aware that an MFG/UTIL validation failed, and the system being built may not function properly. You should return to the data-entry frame to update the build information as needed before building a non-validated system.

**Fig. 4.8**  
Build Desktop on Progress Database

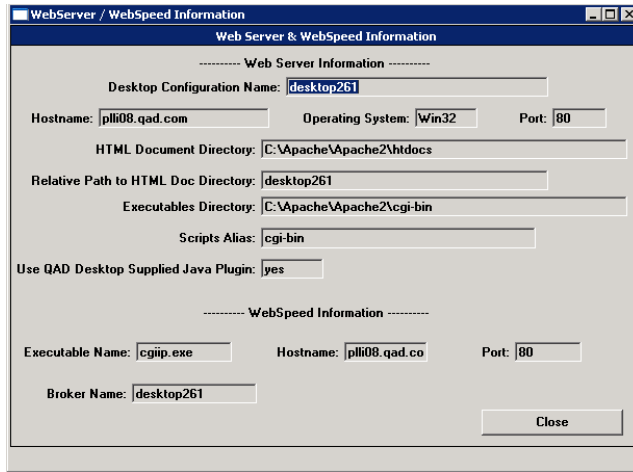


**Oracle:** If you are building Desktop for an Oracle database environment, the confirmation screen also displays the Oracle-specific settings. Verify the Oracle Home, Oracle SID, and for UNIX environments, the Oracle Shared Library Path values.

**Fig. 4.9**  
Oracle Build Parameter Confirmation Fields



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



**WebServer / WebSpeed Information**

----- Web Server Information -----

Desktop Configuration Name:

Hostname:  Operating System:  Port:

HTML Document Directory:

Relative Path to HTML Doc Directory:

Executables Directory:

Scripts Alias:

Use QAD Desktop Supplied Java Plugin:

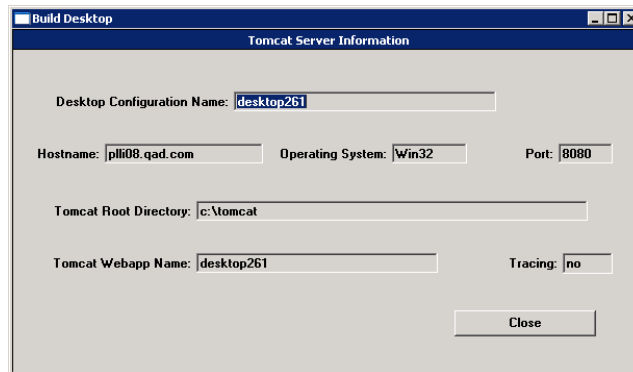
----- WebSpeed Information -----

Executable Name:  Hostname:  Port:

Broker Name:

**Fig. 4.10**  
Web Server and  
WebSpeed  
Information Fields

- 5 After verifying the Web server and WebSpeed information, choose Close. In the Desktop Build screen, choose Tomcat Info to review the build information for Tomcat.



**Build Desktop**

**Tomcat Server Information**

Desktop Configuration Name:

Hostname:  Operating System:  Port:

Tomcat Root Directory:

Tomcat Webapp Name:  Tracing:

**Fig. 4.11**  
Tomcat Server  
Information Fields

- 6 After verifying the Tomcat information, choose Close to return to the Build Desktop screen.
- 7 If you need to set advanced configuration options—typically, used when building Oracle systems, rebuilding Desktop, upgrading to a new release, creating build scripts without executing the build, or building multi-language environments—choose Advanced. Advanced build options are discussed in “Advanced Build Options” on page 210.

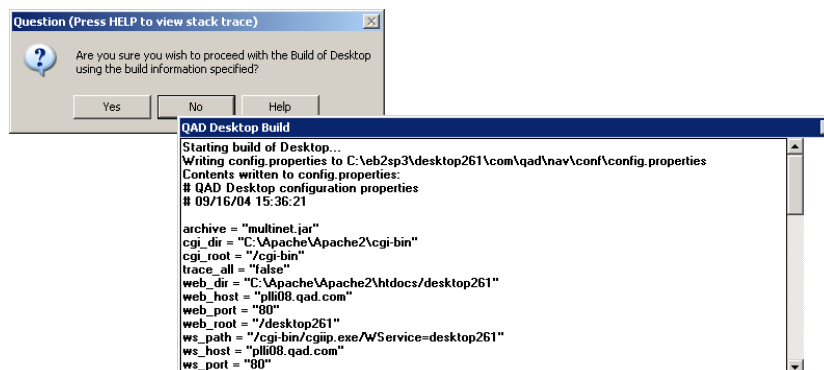
**Note** If you are building a multiple-language Desktop system, make sure you review the information in Chapter 8, “Implementing Multiple Languages,” on page 133.

See “Building Desktop for an Oracle System” on page 58.

**Important** If you are building Desktop for an Oracle system, you must access the Advanced Build options and complete the build in two steps.

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

**Fig. 4.12**  
Confirm the Build



- 9 Once the Desktop build completes, review the build log for errors. You must correct any errors before you proceed. Choose Close.

Once the Desktop system has been built, use the instructions in the next section to generate the telnet and Connection Manager telnet server scripts, the Desktop system-specific `ubroker.properties` file, and a detailed system configuration list. These files are required during the configuration steps detailed in the next chapter.

### Building Desktop for an Oracle System

When running the Desktop build, a compile and data load are performed using the same parameter file (`.pf`). For Oracle environments, the database should not be connected during the compile process, only during the data load process. Using the same `.pf` file causes the compile to fail.

To avoid this problem when setting up Desktop in an Oracle environment, first run the data load using a .pf file that connects to the schema holder and the Oracle database; then perform the compile process as a separate step using a .pf that does not connect to the Oracle database.

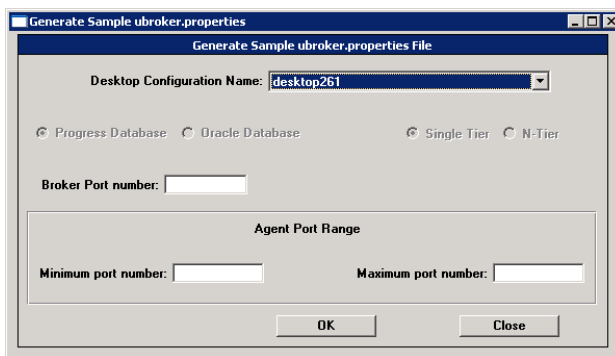
- 1 Before executing the steps under “Configure Desktop Components” on page 50, perform these steps:
  - a Make a copy of your .pf to modify for the compile process.
  - b In the copy, remove the Oracle connection parameters from the .pf file, leaving only the schema holder connection parameters.
  - c Note the directory where both .pf files are located. You enter this information in a later step.
- 2 Launch MFG/UTIL from the database server. Then choose Desktop|Configure Desktop Environment Parameters.
- 3 In Database Connection Parameter File with Path, choose the original .pf. (This corresponds to step 4 on page 52.) This .pf should include connection parameters for Oracle and the schema holder. Save your changes by pressing Go.
- 4 Follow the steps in “Build the Desktop System” on page 55 starting with choosing Desktop|Build Desktop from MFG/UTIL and verify the various system values.
- 5 In step 7 on page 57, choose the Advanced option in the Build Desktop screen.
- 6 On the Advanced Build Options screen, leave all build options checked except for Compile Desktop Progress Programs, which should be cleared. Choose Build.
- 7 Exit the Build Desktop screen and then repeat step 2. However, this time select the .pf file that only connects to the schema holder.
- 8 Choose Desktop|Build Desktop from MFG/UTIL and choose the Advanced option. On the Advanced Build Options screen, clear all the build options except for Compile Desktop Progress Programs, which should remain checked. Choose Build. This correctly compiles the Desktop code.

## Generate WebSpeed Configuration Information

Part of the Desktop post-installation configuration process is to update the WebSpeed configuration file, `ubroker.properties`, with information for the Desktop-required brokers. That step requires you to manually merge a MFG/UTIL-generated parameter file with the WebSpeed configuration file. For details, see “Configure and Start WebSpeed” on page 98.

To generate a file with the `ubroker.properties` information for your Desktop systems, launch MFG/UTIL. Choose Desktop|Generate Sample `ubroker.properties` File. The Generate Sample `ubroker.properties` File frame displays.

**Fig. 4.13**  
MFG/UTIL  
Generate Sample  
`ubroker.properties`  
File



Choose the system in Desktop Configuration Name; then enter the port to use and the minimum and maximum port numbers. Choose OK to build the sample definition for the system you selected. The file is saved to the MFG/PRO installation directory. It contains the complete broker definition required by Progress to launch and configure the brokers, such as:

- Universal unique identifier (UUID) for the broker, used by Progress to uniquely identify the WebSpeed broker
- The port numbers to use
- Directories for the PROPATH
- Location of log files

Generate this script for each system you are setting up.

## Generate Telnet Connection Scripts

Later steps in the Desktop configuration process require you to specify telnet connection scripts:

- You specify a telnet log-in sequence in MFG/PRO for programs that display in Desktop's integrated telnet display. ▶ See page 103.
- You specify a telnet startup script for Connection Manager log-in using MFG/UTIL. ▶ See page 63.

Use MFG/UTIL to generate the required startup scripts now.

To generate the Connection Manager and telnet connection scripts, choose Desktop|Create Connection Manager and telnet scripts from the main menu. When the Generate Desktop 2 Scripts selection screen displays, select the configuration you want to generate scripts for.

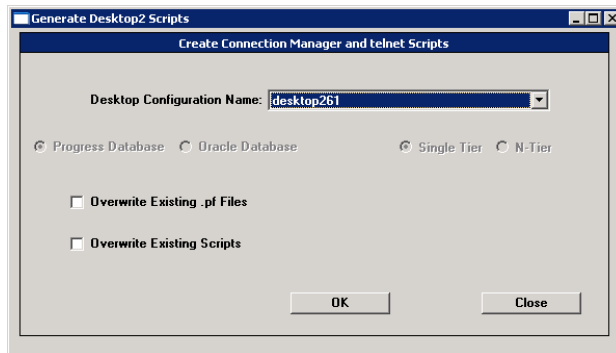


**Fig. 4.14**  
MFG/UTIL  
Create Connection  
Manager and telnet  
Scripts

Using the configuration information for the system you selected, MFG/UTIL generates two files in the Desktop installation directory. The file names depend on the operating system specified for the Web server in the configuration settings screen:

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

*DBSetName* refers to your system configuration name.

Use the check boxes to avoid being prompted to overwrite existing .pf files and scripts.

Depending on the release you are upgrading from, some files may be backed up.

- If any qma files are found in the Desktop directories, MFG/UTIL automatically creates a qmaobsolete subdirectory and places the old qma files there.
- If any .pf or telnet connection script files are found in the *MFGPROInstallDir* or *DesktopInstallDir* and you choose not to overwrite them, they are moved to an old\_dtscripts directory.

**Note** In releases of Desktop prior to 2.6, scripts were created in *MFGPROInstallDir*. They are now created in *DesktopInstallDir*. This ensures that scripts are not overwritten when multiple Desktop environments are being used, such as a test and production environment.

**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 server operating system documentation.

Desktop uses values defined in the Progress protermcap file referenced by the PROTERMCAP environment variable in the Connection Manager connection script generated in this step. The keys listed in Table 4.2 must have their default setting or actions in Desktop may fail. If you have modified any of these keys in the protermcap file, you must change the PROTERMCAP environment variable to point to an unchanged file.

**Table 4.2**  
Function Keys

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

## Generate Connection Manager Log-In Information

Before you can successfully start Connection Manager, a telnet connection script and several log-in and configuration details are required. Connection Manager uses these settings to log in to the telnet server and open a pool of telnet sessions. This pool is used to display the Desktop programs that display with HTML screens.

You can also specify this information directly in the Connection Manager Administration page. However, it is easier and less error-prone to use MFG/UTIL to create the script. MFG/UTIL updates the same configuration file that is updated when you use the Configuration Manager Administration page.

♦ See “Administer Connection Manager” on page 115.

**Note** You set up connection scripts for the programs accessed using Desktop’s integrated telnet display using User Option Telnet Maintenance (36.20.3). This activity is described in Chapter 6.

♦ See page 103.

**Important** The Connection Manager script requires a log-in ID. You should create a separate, exclusive log-in ID for Connection Manager that is used for no other purpose than creating a pool of sessions for Desktop users. This approach ensures that Desktop sessions are correctly accounted for by licensing functions.

Separate, exclusive log-in IDs should also be used for starting WebSpeed brokers and for no other purpose. You can use the same exclusive log-in ID starting the Desktop telnet sessions and WebSpeed brokers.

To generate Connection Manager log-in information, choose Desktop|Connection Manager Login Maintenance. When the screen shown in Figure 4.15 displays, select the configuration you want to generate scripts for.

**Fig. 4.15**  
MFG/UTIL  
Connection  
Manager Login  
Maintenance

The screenshot shows a window titled "Desktop2 Configuration" with a sub-header "Connection Manager Login Maintenance". The fields are as follows:

- Desktop Configuration Name: dropdown menu showing "desktop261"
- Login Script: text field with a "Browse..." button
- Working Directory: text field
- Hostname: text field showing "localhost"
- Host Operating System: dropdown menu showing "WIN32"
- Operating System Prompt: text field showing ">"
- Port: text field showing "23"
- Login: text field showing "guest"
- Password: text field
- Confirm Password: text field
- WIN32 Domain: text field
- Minimum Connections: text field showing "10"
- Maximum Connections: text field showing "20"
- Buttons: "Save" and "Close"

Use the following field descriptions to help you enter the correct values.

**Important** Prompt values are case sensitive. The values you enter must be identical to the prompts displayed by the telnet server when the Connection Manager logs in.

*Desktop Configuration Name.* Select the configuration that you are setting the Connection Manager log-in options for.

*Login Script.* Use the browse to select the MFG/UTIL-generated `connmgr.DBSetName` (UNIX) or `connmgrDBSetName.bat` (Win32) script to use for logging in to the telnet server.

*Working Directory.* Optionally enter a working directory. All files generated during the Desktop session are saved to this directory. If you leave this field blank, files are saved to the home directory of the user ID specified in Login.

*Hostname.* Enter the machine name or IP address of the telnet server. The Desktop uses this information to establish the telnet connection.

*Host Operating System.* Indicate whether the telnet sessions are executing on a computer with a Windows or UNIX operating system.

*Operating System Prompt.* Enter the prompt displayed at the server command line; for example, some servers use the pound sign (#) or the dollar sign (\$). On Windows systems, this defaults to >.

*Port.* Enter the port number for the telnet server.

*Login.* Enter the ID to use when logging in to the server to start the telnet sessions. This should be a separate, exclusive log-in ID only used by the Connection Manager.

*Password.* Enter the password for the indicated log-in ID to log in to the server.

*Confirm Password.* Confirm the password by reentering it.

*WIN32 Domain.* On Windows systems only, specify the domain the log-in user ID is a member of.

*Minimum Connections.* Enter the minimum number of open connections that the system should attempt to maintain. During startup, the Connection Manager opens this number of sessions. As sessions are used, it continues to open more so that this number of open sessions is maintained, until it reaches the value specified for Maximum Connections.

*Maximum Connections.* Enter a number that represents the maximum number of open connections that the system should allow. Connection Manager will not open any more sessions than this.

## Print Desktop Information

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

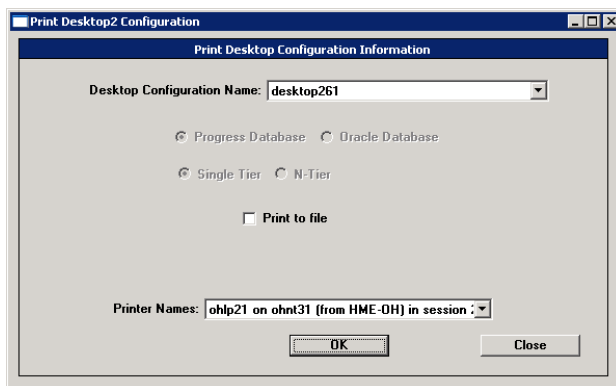
- Desktop Build PROPATH
- Telnet Client PROPATH
- Connection Manager PROPATH
- WebSpeed PROPATH

Optionally, you can send that information to your system printer. To use this function, launch MFG/UTIL; then choose Desktop|Print Desktop Configuration Information from the main menu.

When the Print Desktop Configuration screen displays, select your configuration name. Select Print to file to save a text file; otherwise, specify the printer to use to print the configuration information. The

sample file is saved to the MFG/PRO installation directory. Alternatively, enter a fully qualified directory name location and file name to save the file there.

**Fig. 4.16**  
MFG/UTIL Print  
Desktop  
Configuration  
Information



You are now ready to perform the Desktop setup and configuration steps detailed in Chapter 6, “Configuring Desktop After Deployment,” on page 97.

## QAD Desktop Quick Start

If you are an advanced user installing a single-tier, single-language system, you can use this section to quickly move through the installation steps. Cross-references are provided for each step where additional details can be found if necessary.

- 1 Log in as a user with write permissions to the Web server, Tomcat, and MFG/PRO installation directories.
- 2 Insert the CD-ROM into the drive and mount it on UNIX systems.
- 3 After the CD-ROM is mounted on UNIX systems, go to the CD-ROM directory and run:
 

```
install.ksh
```

 For Windows, run `install.exe` from Run on the Start Menu.
- 4 When prompted, specify a log file directory.
- 5 When prompted, specify the application name for your Desktop configuration.

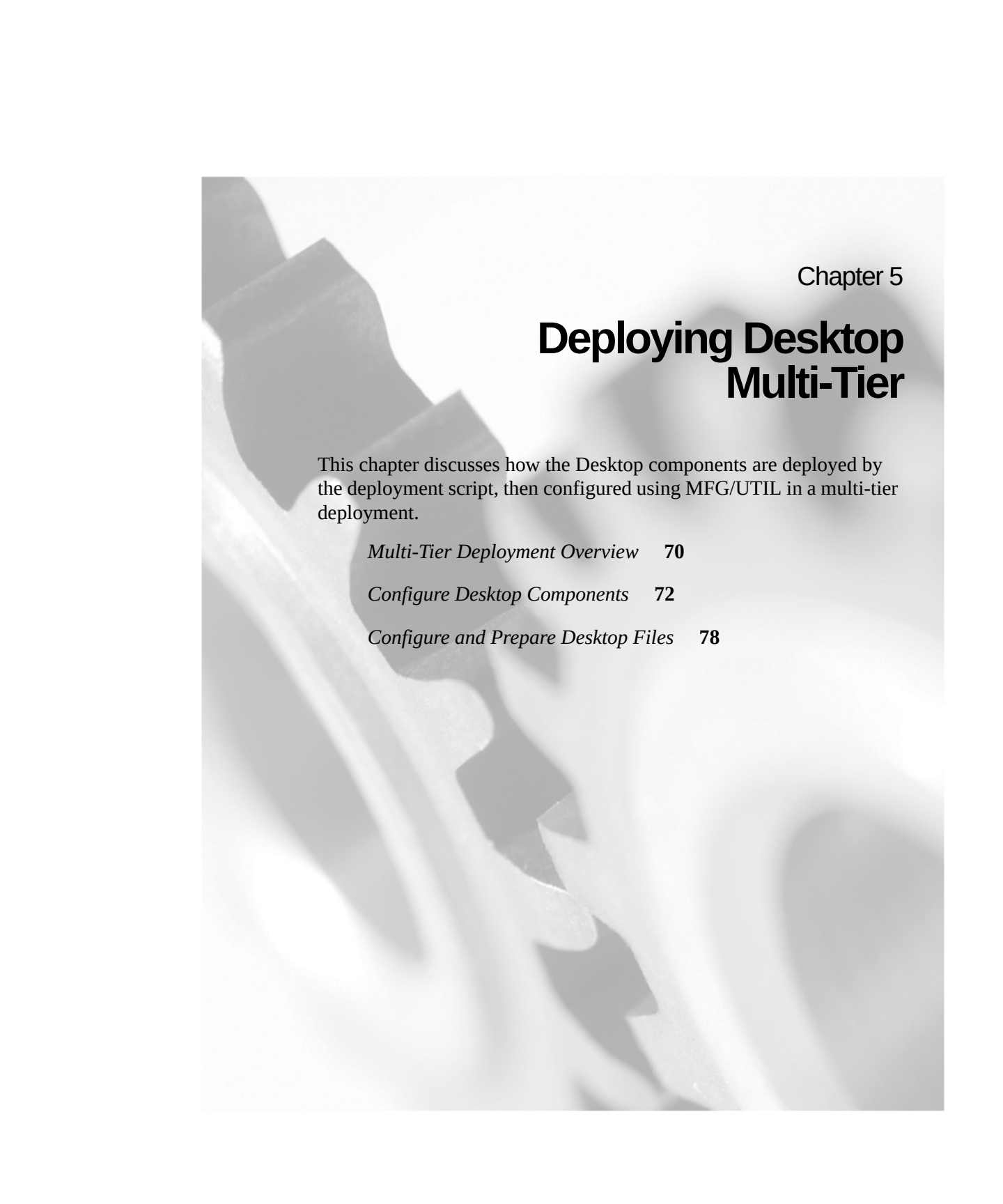
See page 41.

See page 42.

See page 43.

- 6 Review and accept the license agreement. ▶ See page 43.
  - 7 When prompted for the type of install, select 1 to specify a single-tier deployment. ▶ See page 43.
  - 8 When the main menu displays, choose 1 to access the HTML and CGI files menu. ▶ See page 44.
  - 9 Choose 1 again on the HTML/CGI menu to install the HTML and CGI files. Accept defaults when prompted for directories or enter appropriate values. When complete, exit to the main menu. ▶ See page 44.
  - 10 Choose 2 from the main menu to install Tomcat files. Choose 1 on the Tomcat menu to complete the installation. When complete, exit to the main menu. ▶ See page 46.
  - 11 Choose 3 from the main menu to install the MFG/PRO Desktop code. When complete, return to the main menu. ▶ See page 47.
  - 12 On UNIX systems, choose 3 on the MFG/PRO – Progress Desktop Files Menu to launch MFG/UTIL.  
  
On Windows systems, choose 3 on this menu to generate telnet connection scripts. Then exit the deployment menus and start MFG/UTIL from the Windows Start menu. ▶ See page 49.
  - 13 From MFG/UTIL, go to the Desktop menu and choose Configure Desktop Environment Parameters. Edit your Desktop configuration. ▶ See page 51.
  - 14 In the configuration parameters screen, review all field values and update any information if needed. Verify that you are using the correct .pf file and database set name. Then choose OK. ▶ See page 52.
  - 15 Select Build Desktop from the MFG/UTIL Desktop menu. Verify that all information is correct and choose Build. ▶ See page 55.
- Important** Ensure Validated for build field is checked before proceeding.
- 16 When the build completes, select Generate Sample ubroker.properties File from the Desktop menu. Use this file to create your WebSpeed broker. ▶ See page 60.  
  
You will need to copy the file to ubroker.properties. ▶ See page 99.

- ♦ See page 61.
- 17** From the Desktop menu, choose Create Connection Manager and telnet scripts to generate two scripts:
- `connmgr.DBSetName` and `telnet.DBSetName` in UNIX systems
  - `connmgrDBSetName.bat` and `telnetDBSetName.bat` in Windows systems
- ♦ See page 63.
- 18** From the Desktop menu, choose Connection Manager Login Maintenance to specify parameters the Connection Manager uses to create a pool of telnet sessions for accessing HTML programs in Desktop.
- ♦ See page 103.
- 19** Exit MFG/UTIL, log in to MFG/PRO, and set up User Option Telnet Maintenance (36.20.10.3).
- 20** Start your Web server, Tomcat, and your WebSpeed broker.
- 21** Go to:
- `http://ServerName/DesktopSysName/admin.html`
- Click Connection Manager and choose Update Configuration Settings.
- ♦ See page 117.
- 22** Specify and save your Connection Manager settings. Then restart Connection Manager and verify you have idle sessions.
- 23** From the Administration page, click Launch to log in to Desktop.

The background of the page is a grayscale image of several interlocking gears. The gears are of different sizes and are positioned in a way that they appear to be meshing together. The lighting creates highlights and shadows on the teeth of the gears, giving them a three-dimensional appearance. The overall tone is technical and industrial.

## Chapter 5

# Deploying Desktop Multi-Tier

This chapter discusses how the Desktop components are deployed by the deployment script, then configured using MFG/UTIL in a multi-tier deployment.

*Multi-Tier Deployment Overview*    **70**

*Configure Desktop Components*    **72**

*Configure and Prepare Desktop Files*    **78**

## Multi-Tier Deployment Overview

A multi-tier Desktop deployment requires multiple setup and configuration steps because Desktop components can be deployed to multiple application servers. The number of additional deployment and configuration tasks depends completely on network accessibility, bandwidth, server location, and your Desktop and MFG/PRO architecture.

♦ See “Configure Multiple Desktop Systems” on page 187.

**Note** Additional factors must be considered when you plan to install multiple Desktop systems on the same server or to set up Desktop systems for multiple MFG/PRO versions running on the same server. These issues are discussed in Chapter 10.

♦ See page 72.

This deployment is facilitated by a shared directory location accessible by the application server and database servers. This directory share is used to store the Desktop `.ini` file, the file that holds all the configuration and deployment information used by the Desktop deployment tools. Alternatively, a method to transfer this file and other shared files is required.

♦ See page 72.

A multi-tier deployment begins on the database server. During this step, Progress files are deployed to update MFG/UTIL with the required components to configure and build the Desktop environment. Several servlet engine and Web server CGI and HTML files are also placed in a staging area you specify for configuration with MFG/UTIL. In Windows environments, you also deploy the `k_start.bat` telnet script on the database server.

♦ See page 78

Once the database server files are installed, start MFG/UTIL on the database server to run the Desktop build. MFG/UTIL is also used to:

♦ See page 88.

- Generate customized application server files and create JAR files needed for deploying the files to the application servers.

♦ See page 90.

- Generate Connection Manager and MFG/PRO telnet connection scripts.
- Configure the Connection Manager log-in information.
- Generate the entry for the WebSpeed server `ubroker.properties` configuration file.
- Generate a detailed system configuration listing.

After transferring the JAR files to the application servers, launch the deployment script on the application server and deploy the customized Tomcat, HTML, and CGI files. When the file deployment is complete on the application and database servers, you configure Connection Manager and MFG/PRO components before launching Desktop.

See page 90.

Figure 5.1 depicts the typical multi-tier configuration flow and activities.

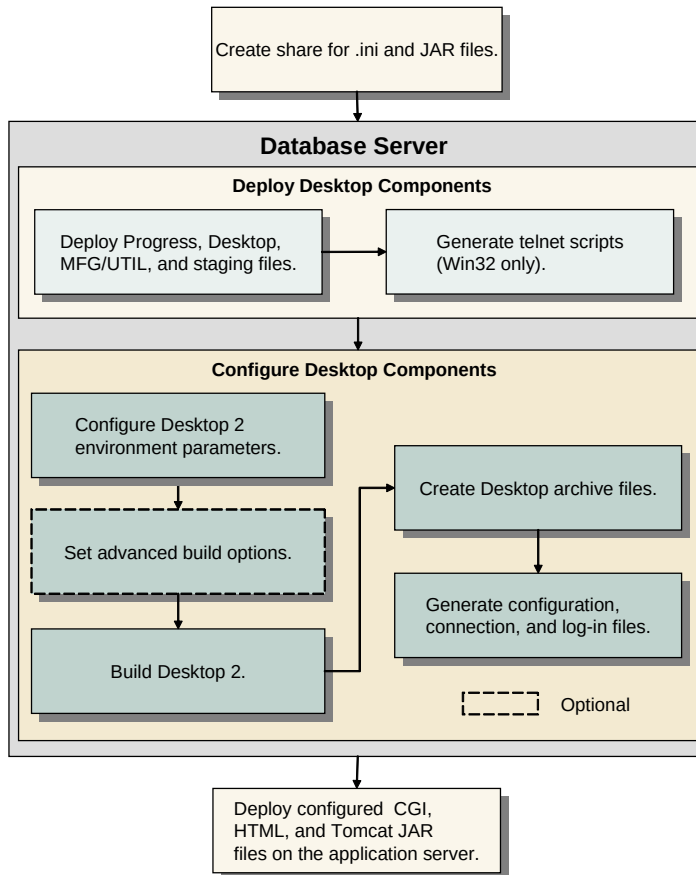


**Fig. 5.1**  
Multi-Tier  
Configuration  
Work Flow

## Desktop Deployment Configuration File

The deployment script and MFG/UTIL must have read/write access to the configuration `.ini` file throughout the deployment process. Create the Desktop log file directory on a network share visible to all the servers. Alternatively, create a log file directory on a local drive, then transfer the file to each server as you continue through each deployment and configuration step. You are prompted for the location of the `.ini` file when starting the deployment script.

## Plan the Deployment

Before you begin, plan the deployment for each distributed component. Make sure that:

- The user account you plan to use on each server has administrator access.
- The shared staging directory is accessible 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 sever, install the free WebSpeed Messenger.

▶ See “Plan a Multi-Tier Deployment” on page 166 for issues to consider.

▶ See page 178.

## Configure Desktop Components

### Prerequisites

A network share accessible by the database server and the application server or some other method to share files is required. For example, you must be able to transfer files from one server to the next using FTP, network shares, e-mail, or physical media, such as tape or CD-RW. The largest file to transfer is approximately 16 MB.

## Launch the Installation Script

**Important** In UNIX, make sure you are logged in as a user with write permissions to the Web server, Tomcat, and MFG/PRO installation directories. In Windows, make sure you have administrative privileges.

- 1 Launch the installation script.

```
./install.ksh
```

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

- 2 You are prompted to enter a log file directory location. Typically, you should name the log file directory based on its associated MFG/PRO or Desktop release.

Each Desktop deployment should have its own log file directory, used for storing several log and support files. Later deployment and configuration processes look at this location for installation information stored in an `.ini` file.

▶ See page 18 for additional log file details.

**Important** Do not delete this directory or manually modify any files found here.

If the directory you specify does not exist, you are prompted to create it. You must also specify the name you want to use for your Desktop configuration, which is used in naming the log and `.ini` files.

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 'C:/mfgsvr'
->c:\mfgsvr
```

```
*** Note that each log directory applies to a single
MFG/PRO installation. Please see the Desktop install
guide for additional information.
```

```
Use c:\mfgsvr as log directory?
Default is 'y'
->y
```

- 3 When any application `.ini` files exist in this directory, they are displayed as selection options. In the following example, an `.ini` file already exists for a configuration named `Test`. 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>?
```

When no application .ini files exist, the following prompt displays:

```
No .ini was found in directory 'C:/logfiles'.
Do you want to start a new setup file (.ini)?
Default is 'y'
-> y
```

```
What is the application name for this Desktop 2 configuration?
Default is 'desktop'
->Prod-Multi
```

Enter Yes to start a new application .ini file; then enter a name for this Desktop configuration or accept the default. Application names must be at least three characters, with no spaces. The name you specify is used as the .ini file name.

#### 4 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'
-> Yes
```

◆ See Chapter 4 for single-tier instructions.

#### 5 Specify this is a multi-tier Desktop deployment.

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

#### 6 The main menu displays.

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

## Deploy the Components

Complete this section to deploy the MFG/PRO code included on the Desktop media. This step installs Desktop-specific code for MFG/PRO and updates to MFG/UTIL.

- 1 From the main menu, access the MFG/PRO and Progress Desktop files menu.

**Note** You may be prompted to enter or confirm the JAVA\_HOME directory.

See page 35.

- 2 On the MFG/PRO – Progress Desktop Files Menu, choose option 1 to begin installing the Desktop Progress files.

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

- 3 Enter the MFG/PRO character client installation directory. In a typical MFG/PRO system, you install the character-client files in the MFG/PRO directory. The default MFG/PRO installation directories are:

- c:\mfgsvr\charcli for MFG/PRO eB
- c:\mfgsvr for MFG/PRO eB2 and eB2.1

```
Please specify the MFG/PRO Character Client directory.
Default is 'c:\mfgsvr'
->
```

- 4 Enter and confirm the MFG/PRO code directory. This directory name corresponds to the Desktop system name being deployed. For consistency when deploying multiple Desktop systems, accept the default values.

```
Please specify the MFG/PRO Desktop code directory.
The Desktop Progress code files will be extracted here.
Default is 'C:\mfgsvr\Prod-Multi'
-> C:\mfgsvr\Prod-Multi
```

```
The directory 'C:\mfgsvr\Prod-Multi' does not exist. Would
you like to create this directory?
Default is 'y'
-> y
```

The process is logged to the screen. Files are copied, permissions are set correctly, and MFG/UTIL is updated with Desktop workflows. You are prompted to press Enter as each step completes.

When the process completes, review the log for errors when the installation is complete.

- 5 When the installation of the Desktop files is complete, you are prompted to enter a directory location to use as a staging area for files that need to be configured by MFG/UTIL.

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 'C:\mfgsvr\staging'

->

Several Progress files are copied to a staging directory. Review the details that display.

- 6 You are prompted for the Web server HTML documents directory located on the application server.

It appears that the HTML files have not been installed yet.

Please enter the name of the directory under the html document root directory where the html files will be placed. Usually this is the same as the configuration name.  
Default is 'Prod-Multi'

->

HTML, Java, and script files for the Web server and the Tomcat servlet engine are placed in staging directories based on the directory you specify.

- 7 When the file staging process completes, you are returned to the Progress files menu. You are ready to configure and compile the staged files to prepare them to be moved to the application servers.

If this is a UNIX environment, launch MFG/UTIL directly from the deployment script or return to the main menu and exit the deployment script. Start MFG/UTIL manually.

If this is a Windows environment, you should now generate the Windows telnet script.

## Setup Windows Telnet Environment

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

Once the telnet script is generated, exit the deployment script and start MFG/UTIL to build the Desktop system and application server files.

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

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

- 2 Verify the Georgia SoftWorks Windows NT/2000 Telnet Server (GSWTS) installation.

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 previous 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.

**Note** The default installation directory that Georgia SoftWorks now uses is `gs_uts`, not `gs_tnet`. If you used the Georgia SoftWorks default, be sure to change it here.

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

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

- 4 You may be prompted to confirm the MFG/PRO home directory.

```
Please Enter/Confirm the MFG/PRO Home directory (MFG_HOME).
Default is 'c:\mfgsvr'
-> c:\mfgsvr
```

- 5 The system generates a k\_start.bat file and places executable files in several directories. Note the directories before you continue.

```
Creating a k_start.bat file in C:\gs_tnet\scripts
Thu Feb 5 16:48:57 2004: Created
C:\gs_tnet\scripts\k_start.bat
Copying 'win32\imagecfg.exe' to
'C:\mfgsvr\telnet\imagecfg.exe'
Copying 'win32\setMP.cmd' to 'C:\mfgsvr\telnet\setMP.cmd'

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 Desktop files.

## Configure and Prepare Desktop Files

**Note** Before you begin configuring the Desktop components using MFG/UTIL, make sure your MFG/PRO databases are running.

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

- Verify and Update the System Information
- Build the Desktop System
- Create Application Server JAR Files
- Create Connection and Configuration Files
- Launch the Deployment Script

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



**Fig. 5.2**  
MFG/UTIL  
Desktop Menu  
Options

**Important** If you are building Desktop for an Oracle system, you must create a copy of your .pf file before beginning this sequence of steps.

See “Building Desktop for an Oracle System” on page 86.

## Verify and Update the System Information

Complete this section to set the Desktop 2 build parameters.

- 1 Launch MFG/UTIL from the database server. Then choose Desktop|Configure Desktop Environment Parameters.

**Note** In UNIX environments, you can also launch MFG/UTIL from the deployment script menu.

- 2 If MFG/UTIL detects a new Desktop 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.

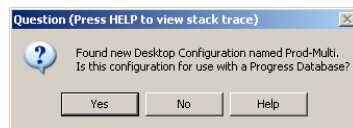
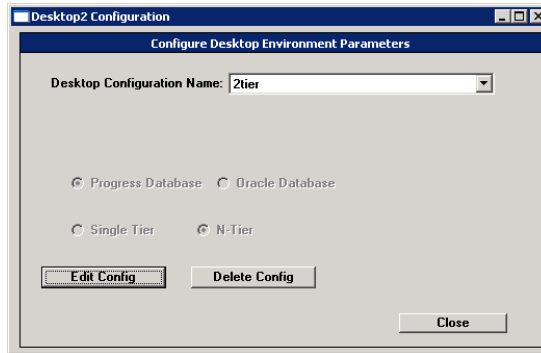


**Fig. 5.3**  
MFG/UTIL  
Database Type  
Prompt

- 3 The Desktop 2 Configuration screen displays. MFG/UTIL examines the system .ini files and provides the system names as selection options. Select the name of the system you are configuring; then choose Edit Config to review and update the build information.

**Fig. 5.4**  
Desktop 2  
Configuration  
Selection



#### 4 The configuration screen displays.

Database Connection Parameter File with Path:

MFG/PRO Language Code:

Desktop Install Directory:

Java Home for JAR Build:

MFG/PRO Version:

Database Set Name:

Service Pack Directory:

Propath:

Tomcat Operating System:

Tomcat Hostname:

Tomcat Port Number:

Tomcat Stage Directory:

Tomcat Webapp Name:

Web Server Operating System:

Web Server Scripts Alias:

Web Server Executables Stage Directory:

Relative Path to Desktop HTML Document Directory:

Web Server HTML Document Stage Directory:

Web Server Hostname:

Web Server Port Number:

WebSpeed Broker Name:

WebSpeed Executable Name:

WebSpeed Server Hostname:

WebSpeed Server Port Number:

Enable Java Console tracing:

Use QAD Desktop Supplied Java Plugin:

Maximum Search Results:

OK Cancel

**Fig. 5.5**  
Multi-Tier  
Configuration  
Screen for Progress

See Figure 5.6.

**Oracle:** The screen differs slightly depending on the server operating system and database type. Three additional fields display for Oracle database environments; ORASOPATHNAME is not used in Windows environments.

See “Add Parameter File Directories” on page 217 for details.

- a Choose the parameter file for your database in Database Connection Parameter File with Path.

This field displays all available .pf files found in your MFG/PRO and client installation directories. You can specify additional directories for MFG/UTIL to search for other .pf files by using the Desktop|Edit Search Directories for Desktop Configuration Lookups option on the MFG/UTIL menu.

Once you select a parameter file, default values are supplied for most of the configuration fields on the screen.

- b If you are installing into an environment that includes service pack files in separate files, enter the correct directory in the Service Pack Directory field.
- c Specify Yes to Use QAD Desktop Supplied Java Plugin to use the version of the plug-in supplied on the Desktop CD.
- d Review all field values and update any missing information. Use the field descriptions in “MFG/UTIL Reference” on page 202 to help you review and update the field values.
- e Use fully qualified server names for the Tomcat Host Name, Web Server Host Name, and the WebSpeed Host Server Name.

See “Desktop Java Plug-in Version Control” on page 130 for details.

**Fig. 5.6**  
Configuration  
Fields for Oracle

Absolute Path to Oracle Install:

Oracle System Identifier Name:

ORASOPATHNAME:  
Not Applicable for Windows

Service Pack Directory:

Oracle-specific fields

- 5 Choose OK after entering and verifying existing information for all fields.
- 6 When the Desktop 2 Configuration screen displays again, exit to the main menu.

## Build the Desktop System

You can optionally generate build scripts without actually completing the build at this time. This gives you more control over the build and can ease system administration activities in environments with multiple Desktop systems. You can create build scripts for all the Desktop systems without running the build and then create a batch or cron job to run the build for one or multiple Desktop systems at a later time when more system resources are available or during scheduled downtimes.

To generate the scripts only, choose the Advanced Build Options and select the Create Build Script Only option.

See step 7 on page 85.

**Note** If you are building Desktop for an Oracle system, you must execute the build in two separate steps to prevent compile errors.

See “Building Desktop for an Oracle System” on page 86.

Use the following steps to verify the system information and build the Desktop system:

- 1 From the MFG/UTIL main menu, choose Desktop|Build Desktop.
- 2 In Desktop Configuration Name, select the system to build.

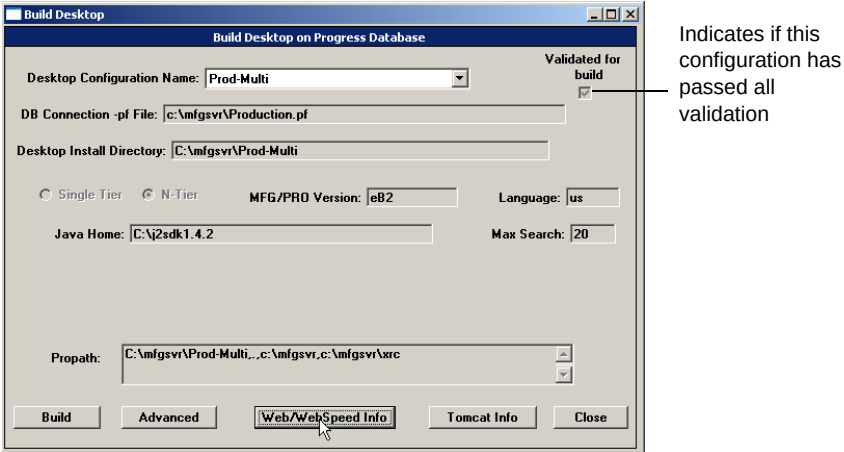
**Fig. 5.7**  
Select Desktop Configuration to Build

- 3 The fields are populated with the values for the system configuration you chose. Verify the build information by reviewing each set of fields before building your system.

See page 202 for field descriptions.

**Note** If Validated for Build is not checked, you can still build the system, but be aware that an MFG/UTIL validation failed, and the system being built may not function properly. You should return to the data-entry frame to update the build information as needed before building a non-validated system.

**Fig. 5.8**  
Build Desktop on  
Progress Database

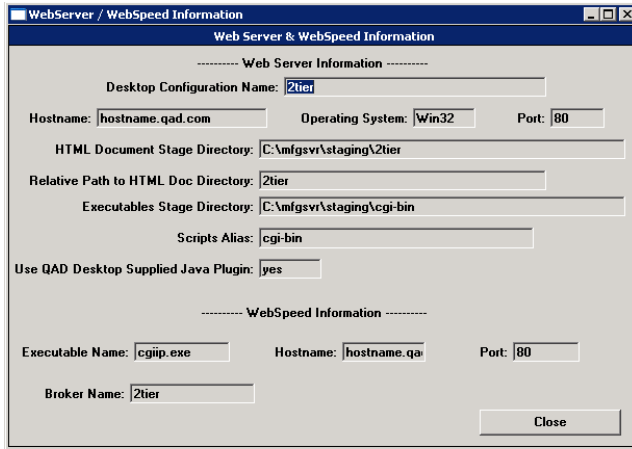


**Oracle:** If you are building Desktop for an Oracle database environment, the confirmation screen also displays the Oracle-specific settings. Verify the Oracle Home, Oracle SID, and for UNIX environments, the Oracle Shared Library Path values.

**Fig. 5.9**  
Oracle Build  
Parameter  
Confirmation  
Fields



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



**WebServer / WebSpeed Information**

Web Server & WebSpeed Information

----- Web Server Information -----

Desktop Configuration Name: 2tier

Hostname: hostname.qad.com    Operating System: Win32    Port: 80

HTML Document Stage Directory: C:\mfgsrv\staging\2tier

Relative Path to HTML Doc Directory: 2tier

Executables Stage Directory: C:\mfgsrv\staging\cgi-bin

Scripts Alias: cgi-bin

Use QAD Desktop Supplied Java Plugin: yes

----- WebSpeed Information -----

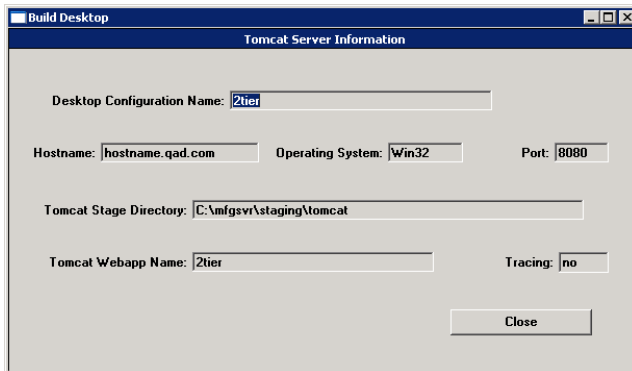
Executable Name: cgiip.exe    Hostname: hostname.qa    Port: 80

Broker Name: 2tier

Close

**Fig. 5.10**  
Web Server and  
WebSpeed  
Information Fields

- 5 After verifying the Web server and WebSpeed information, choose Close. At the Desktop Build screen, select Tomcat Info to review the build information for Tomcat.



**Build Desktop**

Tomcat Server Information

Desktop Configuration Name: 2tier

Hostname: hostname.qad.com    Operating System: Win32    Port: 8080

Tomcat Stage Directory: C:\mfgsrv\staging\Tomcat

Tomcat Webapp Name: 2tier    Tracing: no

Close

**Fig. 5.11**  
Tomcat Server  
Information Fields

- 6 After verifying the Tomcat information, choose Close to return to the Build Desktop screen.
- 7 If you need to set advanced configuration options—typically, used when building Oracle systems, rebuilding Desktop, upgrading to a new release, creating build scripts without executing the build, or building multi-language environments—choose Advanced. Advanced build options are discussed in “Advanced Build Options” on page 210.

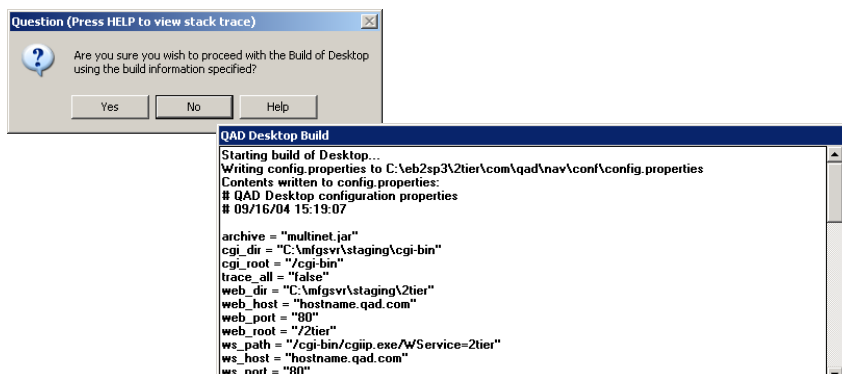
**Note** If you are building a multiple-language Desktop system, make sure you review the information in Chapter 8, “Implementing Multiple Languages,” on page 133.

See “Building Desktop for an Oracle System” on page 86.

**Important** If you are building Desktop for an Oracle system, you must access the Advanced Build options and complete the build in two steps.

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

**Fig. 5.12**  
Confirm the Build



- 9 Once the Desktop build completes, review the build log for errors. You must correct any errors before you proceed. Choose Close.

You are ready to create the archive files that will be used to deploy configured WebSpeed, Tomcat, and Web server files on the application server.

## Building Desktop for an Oracle System

When running the Desktop build, a compile and data load are performed using the same parameter file (.pf). For Oracle environments, the database should not be connected during the compile process, only during the data load process. Using the same .pf file causes the compile to fail.

To avoid this problem when setting up Desktop in an Oracle environment, first run the data load using a .pf file that connects to the schema holder and the Oracle database; then perform the compile process as a separate step using a .pf that does not connect to the Oracle database.

- 1** Before executing the steps under “Configure Desktop Components” on page 72, perform these steps:
  - a** Make a copy of your .pf to modify for the compile process.
  - b** In the copy, remove the Oracle connection parameters from the .pf file, leaving only the schema holder connection parameters.
  - c** Note the directory where both .pf files are located. You enter this information in a later step.
- 2** Launch MFG/UTIL from the database server. Then choose Desktop|Configure Desktop Environment Parameters.
- 3** In Database Connection Parameter File with Path, choose the original .pf. (This corresponds to step 4 on page 81.) This .pf should include connection parameters for Oracle and the schema holder. Save your changes by pressing Go.
- 4** Follow the steps in “Build the Desktop System” on page 83 starting with choosing Desktop|Build Desktop from MFG/UTIL and verify the various system values.
- 5** In step 7 on page 85, choose the Advanced option in the Build Desktop screen.
- 6** On the Advanced Build Options screen, leave all build options checked except for Compile Desktop Progress Programs, which should be cleared. Choose Build.
- 7** Exit the Build Desktop screen and then repeat step 2. However, this time select the .pf file that only connects to the schema holder.
- 8** Choose Desktop|Build Desktop from MFG/UTIL and choose the Advanced option. On the Advanced Build Options screen, clear all the build options except for Compile Desktop Progress Programs, which should remain checked. Choose Build. This correctly compiles the Desktop code.

## Create Application Server JAR Files

Use the following steps to build the archive files for the application servers:

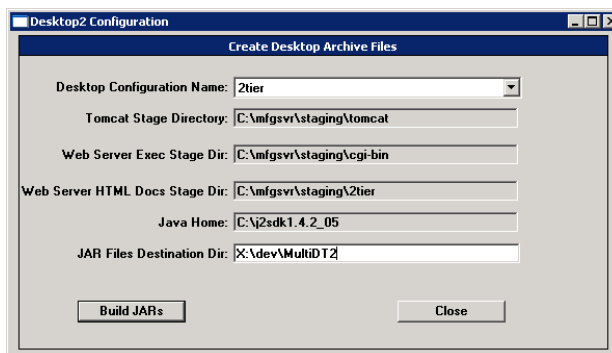
- 1 In MFG/UTIL, choose Desktop|Create Desktop Archive Files.

**Fig. 5.13**  
Build JAR Files for  
Desktop 2 N-Tier



- 2 The Create Desktop Archive Files screen displays. Select your configuration name; the stage directories specified in the configuration screen display in each field.

**Fig. 5.14**  
Create Desktop  
Archive Files



- 3 Specify the destination directory for the JAR files; then choose Build JARs.

The destination directory must be an existing directory location. In the example, h:\multidt2 is a network share. This process takes several minutes. The log files display on the screen.

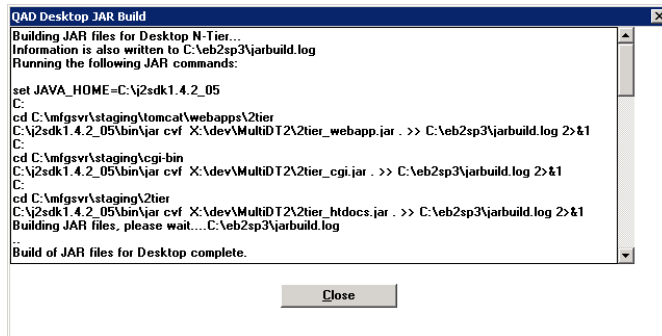


**Fig. 5.15**  
Build JAR Log

- 4 Once the process completes, review the log for errors; then if the JAR files were not generated directly to a network share, transfer them to a network share or to a local drive on the application server.

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

## Create Connection and Configuration Files

Before you deploy the JAR files on the application server, generate the following configuration and startup scripts:

- The entry for the WebSpeed ubroker.properties configuration file  
Use the instructions in “Generate WebSpeed Configuration Information” on page 60 to generate this entry.
- The Connection Manager and telnet connection scripts  
Use the instructions in “Generate Telnet Connection Scripts” on page 61 to generate these files. These files are used by Connection Manager to connect to the MFG/PRO database server and specified in MFG/PRO in User Telnet Options Maintenance.
- Connection Manager log-in information  
Use the instructions in “Generate Connection Manager Log-In Information” on page 63 to specify this information, which is used to configure the Connection Manager.

- The system configuration listing

Use the instructions in “Print Desktop Information” on page 65 to generate this file. It includes complete configuration information for the Desktop system environment.

## Launch the Deployment Script

Use the following steps to launch the deployment script and access the JAR file menu to deploy the JAR files with the Tomcat, HTML, and CGI components to corresponding directories on the application server. Before you proceed, make sure the .ini and JAR files are accessible by the application server or you have copied them to a local directory on the application server.

- 1 On the application server, launch the deployment script from the Desktop installation media.
- 2 At the prompt, point to the .ini file for the multi-tier deployment. This is the configuration .ini file that was created when you deployed and staged the Desktop components on the database server. You should have moved this file to a network location, or to a local drive on the application server.

In the following example, the .ini file for this configuration, Prod-Multi\_dt2.ini, was copied to the H:\Multidt2 Windows network share.

```
Please enter a directory to write log files
Default is 'C:\mfgsvr'
->H:\Multidt2
```

- 3 At the configuration file selection prompt, select the configuration file for your multi-tier system.

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

1: Prod-Multi_dt2.ini -> Prod-Multi
2: Create new configuration
3: Cancel
<1-3>? 1
```

- 4 On the main menu, choose the Post MFG/UTIL Desktop Build Jar Install menu option.

**Note** You may be prompted to enter or confirm the JAVA\_HOME value.

```
*** Configuration: Prod-Multi ***
*** Main Menu ***
Please choose one of the following:

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

The JAVA_HOME environment variable for this machine is set to
'C:\j2sdk1.4.2'. Is this correct?
Default is 'y'
->
```

- 5 When prompted, enter the directory where the JAR files are located. This should be your network share or the local application server directory where you copied the JAR files. In this example, the files were moved to the same network share where the .ini file was placed. The default is the database directory where the files were originally generated:

```
Please enter the location of the MFG/UTIL created JAR files.
Default is 'C:\mfgsvr\staging'
-> H:\Multidt2
```

- 6 The JAR extract menu displays.

```
*** Configuration: Prod-Multi ***
*** Jar Extract Menu ***
Please choose one of the following:

 1: Prod-Multi_webapp.jar
 2: Prod-Multi_htdocs.jar
 3: Prod-Multi_cgi.jar
 4: Return to Main Menu
<1-4>?
```

This menu displays the archive JAR files in the directory you specified. File names are based on the system name, followed by an extension identifying the contents. For example, Prod-Multi\_webapp.jar contains the Web application files for the Prod-Multi Desktop system. The following extensions are used:

- \_webapp.jar contains the application files for the Tomcat server.
- \_htdocs.jar contains the HTML files for the Web server.
- \_cgi.jar contains the script files for the Web server.

## Deploy the Customized Web Application Files

Use the following steps to expand the Web application JAR file. This archive file 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 `_webapp.jar` file to the local application server directories.
- 2 The system reviews your current configuration to see if the file can be distributed to the local server. Review and confirm the system configuration prompts. This include verifying the server name and whether it is a Tomcat server.

```
This machine's name is 'dzv-lt.qad.com'.
Is dzv-lt.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 Tomcat is already installed, only the customized Web application and HTML files are installed.

```
Please enter the Tomcat installation directory.
Default is 'c:\tomcat'
->
```

```
The directory 'c:\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 101.

**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 6.

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

```
It does not appear that Tomcat is installed at 'c:\tomcat'.
Do you want to install Tomcat to this directory now?
Answering 'n' will cancel the extraction of
'H:\multidt2\Prod-Multi_webapp.jar'.
```

```
Default is 'y'
->
```

The deployment scripts expand the archive JAR file and deploy the configured files to the proper directories. Review the logs. At the conclusion of the file expansion process, the JAR extract menu redisplays. Continue by deploying the `_htdocs.jar` file.

### Deploy the Customized Web Server HTML Files

Use the following steps to expand the `htdocs` JAR file. This archive file contains the customized HTML files that reside in the Web server documents directory.

- 1 From the JAR extract menu, choose the option to deploy the `_htdocs.jar` file to the Web server documents directory on the application server.

```
*** Configuration: Prod-Multi ***
*** Jar Extract Menu ***
Please choose one of the following:
```

```
1: Prod-Multi_webapp.jar
2: Prod-Multi_htdocs.jar
3: Prod-Multi_cgi.jar
4: Return to Main Menu
```

```
<1-4>? 2
```

```
Java version on machine 'dzv-lt.qad.com' is 1.4.1_01
```

- 2 Confirm the server name and function; then enter the Web server document directory. This example uses an Apache Web server.

```
This machine's name is 'dzv-lt.qad.com'.
Is dzv-lt.qad.com a web server?
Default is 'y'
-> y
```

```
Please enter the Web Server's document root directory.
Default is 'c:\apache2\htdocs'
-> C:\apache2\htdocs
```

- 3 Enter the directory name where the HTML files should be placed under the documents directory. You should always accept the default value, which is the configuration name. This helps keep the directory structure organized when you are deploying multiple systems.

```
Please enter the name of the directory under
c:\apache2\htdocs' where the HTML files will be placed.
Usually this is the same as the configuration name.
Default is 'Prod-Multi'
->Prod-Multi
```

The JAR file extraction begins and the directory structure is updated with the Desktop HTML documents. When the process concludes, review the extraction logs for error. The JAR extract menu redisplay. You are ready to deploy the last archive JAR file.

### Deploy the Customized Web Server Script Files

Use the following steps to expand the cgi-bin JAR file. This archive file contains the customized script files that reside in the Web server cgi-bin directory.

- 1 From the JAR extract menu, choose the option to deploy the `_cgi.jar` file to the Web server scripts directory on this application server. Most Web servers use the `cgi-bin` directory as the scripts directory. Refer to your Web server documentation for details.

```
*** Configuration: Prod-Multi ***
*** Jar Extract Menu ***
Please choose one of the following:

1: Prod-Multi_webapp.jar
2: Prod-Multi_htdocs.jar
3: Prod-Multi_cgi.jar
4: Return to Main Menu
<1-4>? 3

Java version on machine 'dzv-lt.qad.com' is 1.4.1_01
```

- 2 Confirm the server name and function.

```
This machine's name is 'dzv-lt.qad.com'.
Is dzv-lt.qad.com a web server?
Default is 'y'
-> y
```

- 3 At the prompt, enter the full path to the Web server scripts directory.

```
Please enter the full path to the directory where the cgi-bin
files will be placed for this configuration.

Default is 'C:\apache2\cgi-bin'
->
```

The JAR file extraction begins. When the process concludes, review the extraction logs for errors. The JAR extract menu redisplay.

Deployment activities on the application server are complete; you can now exit the deployment script. You are ready to perform the prerequisite Connection Manager setup activities detailed in “Post-Deployment Configuration” on page 98.



# Configuring Desktop After Deployment

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

*Post-Deployment Configuration*    **98**

*Configure and Start WebSpeed*    **98**

*Configure and Start Tomcat*    **100**

*Prepare Telnet UI Connection Scripts*    **103**

## Post-Deployment Configuration

Before you can launch Connection Manager, you must complete the following tasks:

- Configure and Start WebSpeed
- Configure and Start Tomcat
- Prepare Telnet UI Connection Scripts

The following sections detail how to complete each of these steps.

### Configure and Start WebSpeed

The WebSpeed server is used for the MFG/PRO log-in sequence, for user validation, and for running browses and lookups. The WebSpeed `ubroker.properties` file holds all the configuration information used by WebSpeed.

You should have generated the `ubroker.properties` information for your system using MFG/UTIL as part of configuring Desktop components. If you have not already generated the configuration data for your Desktop system, use the steps in “Generate WebSpeed Configuration Information” on page 60 to do so now.

The only WebSpeed configuration task that remains is to copy the updated sample contents into the WebSpeed `ubroker.properties` file. Since MFG/UTIL has already recorded most of your system configuration details, the generated sample file includes all the required information in the required format. Typically you can simply copy the entire sample settings directly into your system `ubroker.properties` file.

**Important** Always back up configuration files before editing them.

## Update the WebSpeed Configuration File

Use the following steps to update the WebSpeed `ubroker.properties` configuration file by adding the MFG/UTIL-generated Desktop configuration data to it.

- 1 If you have not already generated the configuration data for your Desktop system, use the steps in “Generate WebSpeed Configuration Information” on page 60 to do so now.
- 2 Make a backup copy of the `ubroker.properties` file found in the `properties` subdirectory of your Progress installation directory. Name it `ubroker.orig`.
- 3 Using a text editor, open the MFG/UTIL-generated `SysName-ubroker.properties` file in your MFG/PRO installation directory. `SysName` refers to your Desktop system name.
- 4 Open the `ubroker.properties` file in the `properties` subdirectory of your Progress installation directory.
- 5 Copy the text from the MFG/UTIL-generated file to the end of the `ubroker.properties` file.
- 6 Close the MFG/UTIL-generated `ubroker.properties`.
- 7 Save the file and exit the text editor.

## Verify WebSpeed Servers

Use these steps to verify the WebSpeed servers are active.

- 1 Check to see if the AdminServer is running using the following command:  
`Progress/bin/proadsv -query`  
 If the AdminServer is not already running, start it using the following command:  
`Progress/bin/proadsv -start`
- 2 Start the NameServer process for your broker. These instructions use the default NameServer, NS1:  
`nsman -name NS1 -start`

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

You can check the status of the NameServer using the following command:

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

- 3 Start the MFG/PRO database servers for the Production database set. To start the database servers, use the database startup script for the Production database set; for example, `start.Production`.

When you start a WebSpeed broker, the broker's agents attempt to connect to the databases specified in the Desktop .pf file. This connection fails if the servers for the databases are not running.

- 4 Start the WebSpeed server instance.

```
Progress/bin/wtbman -name BrokerName -start
```

- 5 Check the status of the WebSpeed server instance.

```
Progress/bin/wtbman -name BrokerName -query
```

- 6 After you check the status, leave the instance running. If you need to stop the server, use this command:

```
Progress/bin/wtbman -name BrokerName -stop
```

♦ For additional commands, see the Progress WebSpeed Installation and Configuration Guide.

In addition to the wtbman management utility, you may also need to refer to the log files for the WebSpeed server instance. These files are located in the directory you specified in the agent and broker log file entries (`srvrLogFile=` and `brokerLogFile=` respectively).

## Configure and Start Tomcat

Two separate Tomcat configuration tasks may need to be completed: updating user IDs and ports. Updating the Tomcat user IDs and passwords is always required. Updating port numbers is required only in these situations:

♦ See page 45.

- 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.

## Update Tomcat Port Numbers

If you are using more than one Tomcat instance on a single server or have other applications that use the default Tomcat ports such as 8080 or 8005, you must update the ports in the Tomcat `server.xml` configuration file so that each servlet engine is using unique ports. Each Tomcat instance has its own configuration file.

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

- 1 Using a text editor, open the `server.xml` file for editing. This file is located in `TomcatInstallDir/conf`.
- 2 Look for the following parameter block and update the port reference from port 8080 to the port you plan to use:

```
<!-- Define a non-SSL HTTP/1.1 Connector on port 8080 -->
<Connector className=
 "org.apache.catalina.connector.http.HttpConnector"
 port="8080" minProcessors="15" maxProcessors="175"
 enableLookups="false" redirectPort="8443" compression="on"
 bufferSize="8192" tcpNoDelay="true"
 acceptCount="100" debug="0" connectionTimeout="15000"/>
```

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

```
<Server port="8005" shutdown="SHUTDOWN" debug="0">
```

- 4 Save the file.

## Start Tomcat

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

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

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

## Update User Information

Connection Manager security is implemented through the Tomcat servlet engine. Use the following instructions to modify the default QAD user information. For ease of administration, you can use the same user ID and password you specified for the Desktop Administration page.

The pronav user ID is used to access the Process Editor functions only. This user cannot access the Connection Manager or Desktop Administration page.

- 1 Open the Tomcat Administration Manager located at:  
`http://localhost:8080/admin/login.jsp`  
**Note** You may need to use a different port if you updated it in the `server.xml` file.
- 2 Log in and choose OK. The Administration Manager displays.
- 3 Select Users from User Definition in the left column. The list of currently defined users displays.
- 4 Click on a user name to edit the log-in and password information as well as the defined groups and roles.
- 5 Select Create New User from the Available Actions drop-down to enter new user information.
- 6 Choose Save for each new or modified user record.
- 7 Choose Log Out to exit the Tomcat Administrator.

When you open Connection Manager, you are prompted for the new log-in ID and password.

## Prepare Telnet UI Connection Scripts

You can access maintenance and transaction programs in Desktop in two ways.

- A few maintenance programs display in Desktop’s integrated telnet interface.
- Most maintenance programs display with HTML screens. These screens are generated through XML transformation from a pool of telnet sessions.

A telnet connection script is used to identify the MFG/PRO database that you want each type of program to run against. In general, the connection script performs the following functions:

- Changes to the user’s home directory
- Provides connection parameters for the databases in the set

You use MFG/UTIL to generate the two required telnet connection scripts for each database set you configure. You also use MFG/UTIL to generate the required log-in information for Connection Manager to create the telnet pool for HTML programs. This is discussed in “Generate Connection Manager Log-In Information” on page 63.

◆ See page 61.

No additional configuration steps are required for Desktop to correctly display HTML programs. Additional setup is required to configure the system for the programs that display in the Desktop’s integrated telnet display. You do this setup in User Option Telnet Maintenance (36.20.10.3).

This program lets you define unique settings for individual users or enter an asterisk (\*) in the User ID field to create a generic record of settings. This generic record is automatically associated with every user in the system that does not have a user-specific record.

In addition to settings, you use this program to specify the Windows `telnetDBSetName.bat` or UNIX `telnet.DBSetName` connection script that identifies the MFG/PRO database the Desktop telnet session will run against. `DBSetName` refers to the database set name. This is the connection script that was generated using MFG/UTIL during the Desktop component configuration steps.

◆ See page 61.

Most of the information you define in this program is the same regardless of your telnet server’s operating system. The instructions indicate when there are differences.

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

- “Specify Telnet Server Settings” on page 104
- “Define the Log-In Sequence Script Lines” on page 105
- “Verify the Log-In Sequence” on page 109
- “Configure Telnet Connections Settings” on page 109

Specify Telnet Server Settings

Use the following instructions to configure telnet server settings:

- 1 Start an MFG/PRO session and access User Option Telnet Maintenance (36.20.10.3).
- 2 In the User ID field, enter \* to specify the generic record and press Go.
- 3 Use the following table to complete the Telnet Options fields in User Option Telnet Maintenance.

mgusrtmt.p b+36.20.10.3 User Option Telnet Maintenance08/11/00

User ID: \*

Telnet Options

Host: Host O/S: UNIXPort: 23Image:

Script Timeout: 240Idle Timeout: 240

Field	Description
Host	Enter the fully qualified machine name or IP address of the telnet server. The Desktop 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.
Image	Leave this field blank; images do not apply to Desktop.

Field	Description
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.

- 4 When you are ready, press Go to access the Script Lines frame.

## Define the Log-In Sequence Script Lines

To log in to and begin a telnet session on the telnet server, you must provide the Desktop Java applet with the telnet server log-in prompts and responses to these prompts. Taken together, the prompts and responses make up the telnet log-in sequence.

**Note** The last step in the sequence specifies the script you created in MFG/UTIL.

You define the telnet log-in sequence in the Script Lines frame in User Option Telnet Maintenance.

Sequence:	Script Lines
Script Pattern: login:	
Script Value: netui	
Script Status: logging in	

Each line in the log-in sequence is stored in the MFG/PRO administration schema as a single chui\_det record containing three values: pattern, value, and status. Each of these values is represented by a field in the Script Lines frame.

Table 6.1 provides a general description of the fields in the Script Lines frame.

**Table 6.1**  
Script Lines Fields

Field	Description
Sequence	A number for the script line that indicates the order that the line appears in the overall log-in sequence.
Script Pattern	Enter the prompt generated by the telnet server when a telnet log-in occurs.  <b>Important</b> The values you enter in the Script Pattern field must be identical to the prompts the telnet server displays when users attempt to 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 activity that the prompt and response represent. For example, if the prompt is <b>Login:</b> and the response is a user ID, enter <b>Logging In</b> to describe the activity.  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.

### 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 6.2 illustrates sample script lines for a Windows system.

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	path\telnetdir>	c:\telnet\user1	Start GTNTS as user1	Enter the user- or group-specific directory below the telnet directory.
5	path\telnetdir>	telnetProd.bat	Launching script to connect to Prod	This is the telnet script generated by MFG/UTIL.

**Table 6.2**  
Sample Windows  
Script Values

**Important** In MFG/PRO eB2.1, the value entered in sequence 2 is not displayed on the screen. You are prompted to re-enter it for verification purposes.

Table 6.3 illustrates sample script lines for a UNIX system. Record your log-in script values in Table 6.4.

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.

**Table 6.3**  
Sample UNIX  
Script Values

Seq	Script Pattern	Script Value	Script Status	Notes
3	\$	cd /qad/eB/	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 184.
4	\$	./telnet.Prod	Launching script to connect to Prod	Launch the telnet connection script. See “Generate Telnet Connection Scripts” on page 61.

Use Table 6.4 to record your telnet server log-in information.

**Table 6.4**  
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 6.5.

**Table 6.5**  
Sample Telnet  
Log-In Script

Seq	Script Pattern	Script Value	Script Status
1	Login	<i>UserID</i>	Logging in
2	Password	<i>UserPassword</i>	Supply the password
3	\$	cd /qad/eB/	Accessing DBServer Dir
4	\$	./telnet.Prod	Launching Prod telnet connect script

## Verify the Log-In Sequence

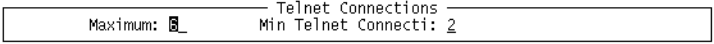
To verify the log-in sequence, from a remote machine attempt to log in to the telnet server using the log-in sequence you configured in MFG/PRO. You should receive a blank telnet screen after the telnet connection script is launched.

## Configure Telnet Connections Settings

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.

**Important** These settings apply to both telnet and HTML maintenance programs in Desktop.

### 1 Access the Telnet Connections frame.



The screenshot shows a window titled "Telnet Connections". Inside the window, there are two fields: "Maximum: 5" and "Min Telnet Connecti: 2". The "Maximum" field has a dropdown arrow next to the number 5, and the "Min Telnet Connecti" field has a dropdown arrow next to the number 2.

### 2 Use the following field descriptions to complete the Telnet Connections fields.

**Maximum.** This value specifies two different settings:

- 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. For example, if Maximum Telnet Settings is 5, a user can have 5 HTML 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 to MFG/PRO through the 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. However, setting this value too high depletes system resources.

QAD recommends that you set Minimum to 2 for users using Desktop telnet maintenance programs extensively. Set Minimum to zero (0) for all other users, including the generic user. The generic user is defined with an asterisk (\*).

Chapter 7

# Configuring Connection Manager

This chapter provides configuration and administration information for the Desktop Connection Manager. You should read this chapter before you begin configuring Connection Manager.

*Overview*    **112**

*Desktop Administration*    **112**

*Plan and Size Connection Manager*    **114**

*Administer Connection Manager*    **115**

*Troubleshoot Errors*    **122**

*Start a Desktop Client*    **128**

## Overview

The Desktop Connection Manager controls the pool of telnet sessions that are used when users execute HTML-based maintenance programs, reports, and inquiries.

You can access Connection Manager administration functions from the Desktop Administration page:

- Some functions are directly associated with Connection Manager. You can close, start, and reset Connection Manager; update configuration settings; and view a log file of Connection Manager actions.
- You can view and monitor the status of connections in the connection pool.
- You can view and monitor the activity of users and close active user sessions, if needed.

This chapter provides instructions on performing general startup, setup and system administration tasks for the Desktop Connection Manager. The following topics are discussed:

- Starting Connection Manager
- Planning and sizing
- Setup and configuration
- Managing connections and users
- Maintaining and troubleshooting

## Desktop Administration

Once you complete the initial setup, configuration, and build steps, and you have started Tomcat, you can begin using the Desktop Administration page. To start the Admin page, point your browser at:

`http://YourWebServer/DesktopInstallDir/admin.html`

Figure 7.1 shows the Desktop Administration page.



**Fig. 7.1**  
Desktop  
Administration  
Page

The Admin page provides a direct access point to the setup, configuration, and startup pages. This page is available only after building the Desktop system. The following URL links are available on the Administration page:

**Tomcat Manager.** This link launches the Tomcat Web Application Manager, which 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 the entire container.

◆ See page 191.

**Connection Manager.** This link launches Connection Manager. This link is used to access all Connection Manager administration and configuration pages.

◆ See page 117.

**Process Admin.** This link launches the Context Parameters page, used to define parameters for the Desktop Process Editor.

◆ See *User Guide: QAD Desktop* for a description of Process Admin and Process Editor.

**Process Editor.** This link launches the Desktop Process Editor in a new window. Use this link to launch and test the Process Editor after making parameter updates in the Process Admin pages.

**QAD Downloads.** Use this link to access the QAD Download Center. This secure Web site provides access to new MFG/PRO service packs and QAD Desktop 2 releases as soon as they are available. This is an optional service available only to QAD customers that have ordered specific MFG/PRO products and QAD Desktop 2. Contact your QAD sales representative for details.

*Launch.* This link launches the Desktop. Use it after making any setup or configuration updates from the Desktop Administration page.

*WebSpeed Developer Console.* Use this link to access the WebSpeed Developer Console. The console is used to develop, diagnose, and debug WebSpeed applications. While the console applies to any application that uses WebSpeed, it is provided particularly to support custom development for Lean Manufacturing workbench programs on the Kanban menu.

## Plan and Size Connection Manager

Most maintenance programs run in Desktop's HTML interface. Custom programs—as well as a limited number of MFG/PRO maintenance programs—run in the Desktop's integrated telnet character mode.

▶ See page 103.

- Desktop telnet character screens are managed by settings in User Option Telnet Maintenance (36.20.10.3).

▶ See page 63.

- Desktop HTML screens are serviced by a pool of telnet sessions managed by Connection Manager.

Connection Manager controls a pool of telnet sessions available for users when they execute HTML maintenance programs, reports, and inquiries. These sessions are all started under one administrative user ID and password, and are then available to any Desktop user.

## Minimum and Maximum Connections

When you configure Connection Manager, you must specify the minimum and maximum number of sessions to be available.

During startup, Connection Manager opens the minimum number of sessions. As sessions are used, it continues to open more sessions so that the minimum number of open sessions is maintained. It opens sessions until it reaches the value specified for maximum connections.

**Example** Minimum Connections is 5 and Maximum Connections is 50. During startup, 5 sessions are opened. When three users log in to Desktop and begin executing HTML maintenance programs, Connection Manager

starts 3 more sessions. This continues until 50 sessions are started. When a request for session 51 occurs, an error displays.

You should determine the maximum number of sessions based on the number of user licenses you have. If you want a large number of sessions to be available on startup, you can set the minimum and maximum to the same value.

### Connection Sizing Guidelines

To save system resources and aid performance, you should always keep the number of connections to a minimum. Add connections as needed by increasing the value of maximum connections. In general, the faster your server, the fewer idle sessions need to be opened initially. Slower servers require more sessions.

**Example** Configure 3 sessions for a faster server with 10-15 users and 6 sessions for the same number of users for a slower server.

## Administer Connection Manager

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



Fig. 7.2  
Connection  
Manager Menus

## Use Connection Manager Menus

### Functions Menu

*Launch.* Launches Connection Manager. Connection Manager is started automatically when you launch Tomcat.

*Close Connection Manager.* Immediately terminates all active Desktop connections. Any data being processed by active Desktop sessions is lost. Any processes begun by active Desktop 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.

▶ See “Configure Connection Manager” on page 117.

*Update Configuration Settings.* Displays the Configuration Settings Update Page, which is used to set up and configure the Connection Manager options.

*View Log.* Displays a complete log of all connection activity since Connection Manager was started, or from the last time the activity log was cleared.

### Connections Menu

▶ See “Monitoring Connections” on page 120.

The Connections menu contains links to four connection views. Use each view to see what connection activity is occurring on the system. The connection information that displays is a snapshot of current activity; use the refresh link to refresh the view with the latest information.

*All.* Displays all connections.

*Busy.* Displays currently busy connections.

*Idle.* Displays currently idle connections.

*Initializing.* Displays currently initializing connections.

## Users Menu

Choose Users to view a list of currently logged in users. Click a user ID to see information related to that user.

◆ See “Monitoring Users” on page 121.

## Configure Connection Manager

Settings used by Connection Manager are defined using the Connection Manager Login Maintenance function in MFG/UTIL. MFG/UTIL will create a log-in script without syntax errors that can occur when you specify it manually in the Connection Manager Administration page. You should only use the Administration page to make minor adjustments after setting up the Connection Manager with MFG/UTIL.

◆ See page 63.

Use the following steps to launch the Connection Manager Administration page and update configuration settings as needed:

- 1 Click Connection Manager in the Desktop Administration page.
- 2 Once Connection Manager is active, click Update Configuration Settings in the Functions menu. The Configuration Settings Update page displays.

**Fig. 7.3**  
Configuration  
Settings Update  
Page

**Configuration Settings Update**

Configuration Parameters

Host: localhost

Port: 23

Startup Script: login:[uibeta]Password:[atebiu]{}|

Minimum Connections: 2

Maximum Connections: 20

Maximum Failures: 15

Connection Timeout: 1800000

Processing Timeout: 2000

Wait time for Idle Connection: 20000

mode: static

Initializing Timeout: 180000

Operating System Win32/NT: false

Connections Monitor Frequency: 180000

Connections Log: connectionslog.txt

Connections Log Dir: logs

Save Cancel

- 3 On the displayed page, edit the values as needed for your environment. Use the following field descriptions to enter the appropriate settings for your environment.

**Host.** Enter the machine name or IP address of the telnet server. The Desktop uses this information to establish the telnet connection.

**Port.** Enter the port number for the telnet server.

See page 61.

**Startup Script.** Specify the telnet server log-in prompts and the responses to these prompts, separated with the pipe symbol (|). Connection Manager uses these to log in to the telnet server and start the pool of sessions.

See page 114.

**Minimum Connections.** Enter the minimum number of open connections that the system should attempt to maintain. During startup, Connection Manager opens this number of sessions. As sessions are used, it continues to open more so that this number of open sessions is maintained, until it reaches the value specified for Maximum Connections.

**Maximum Connections.** Enter the maximum number of open connections that the system should allow. Connection Manager will not open any more sessions than this.

**Maximum Failures.** Enter how many times Connection Manager should attempt to start a session without success before it stops. This number is reset when a successful connection is made. You can also reset it by using the Reset Failed Init Count command on the Function menu.

**Connection Timeout.** Enter, in milliseconds, how long an HTML session can remain inactive before Connection Manager closes it. The default value is 1800000 (30 minutes).

**Note** A session timeout value can also be defined in User Option Telnet Maintenance (36.20.10.3) for browses and telnet maintenance programs.

**Processing Timeout.** Enter, in milliseconds, how long a session can be in processing mode. Processing mode indicates a locked or busy screen. The default value is 3600000 (60 minutes).

**Wait Time for Idle Connection.** Enter, in milliseconds, how long Connection Manager should wait for an initializing session to become idle before the connection fails. The default value is 20000 (20 seconds).

**mode.** Leave the default static value. The dynamic mode is used for troubleshooting page caching issues.

**Initializing Timeout.** Determines how long to wait for a telnet session to initialize. The default value is 180000 (3 minutes).

**Operating System Win32/NT.** Set this to true if the Progress telnet sessions are executing on a computer with a Windows operating system. Otherwise, set this to false.

**Connections Monitor Frequency.** Enter, in milliseconds, how frequently Connection Manager checks to ensure that initializing sessions have not exceeded the maximum time allowed for this status. This time is set in the Wait Time for Idle Connection parameter. The default value is 180000 (3 minutes).

**Connections Log.** Enter the name of the file Connection Manager should use for recording actions and errors. If this file exists, information is appended.

*Connections Log Directory.* Enter the directory where the log file is located. The directory path is relative to the Desktop installation directory on the application server (Tomcat).

- 4 Review your settings and click Save to save any values you entered or modified.
- 5 Click Restart Connection Manager in the Function menu. If prompted to continue, click OK.
- 6 Once Connection Manager restarts, click Connections to monitor the Desktop sessions.

## Monitor Connections and Users

You can use the Connection Manager administration functions to monitor the connection pool and to monitor users.

### Monitoring Connections

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

- Initializing. A session with this status is just starting and is not yet available for use.
- Idle. These sessions are active and available for the next user request.
- Busy. These sessions are currently executing a user request.
- Pause. This 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, so that 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 7.4 illustrates the screen for monitoring connections.

Status	ID	Process ID	User ID	User IP	Max Connections	Program	User Connected Time	Close
Idle	5	14843	null	null	0	null		<a href="#">Close</a>
Idle	8	14848	null	null	0	null		<a href="#">Close</a>
Idle	17	0	null	null	0	null		<a href="#">Close</a>
Idle	16	0	null	null	0	null		<a href="#">Close</a>
Idle	4	14844	null	null	0	null		<a href="#">Close</a>
Idle	13	22926	null	null	0	null		<a href="#">Close</a>
Idle	14	22981	null	null	0	null		<a href="#">Close</a>
Idle	23	0	null	null	0	null		<a href="#">Close</a>
Idle	15	22985	null	null	0	null		<a href="#">Close</a>
Idle	2	14845	null	null	0	null		<a href="#">Close</a>
Busy	3	14840	jws	167.3.11.103	6	sosomt.p	Tue Apr 16 12:54:27 PDT 2002	<a href="#">Close</a>
Total	11							<a href="#">Close</a> <a href="#">All</a>

Fig. 7.4  
Monitoring  
Connections

Use Refresh to update the screen display. Click the Close link to close unneeded sessions.

Monitoring Users

Choose Users to view a list of currently logged in users. Click a user ID to see information related to that user. A screen like the one in Figure 7.5 displays.

Connection Manager

Tue Apr 16 12:56:04 PDT 2002

All: 0 Busy: 1 Idle: 10 Initializing: 0

Name: Busy - jws

Status	ID	Process ID	User ID	User IP	Max Connections	Program	User Connected Time	Close
Busy	3	14840	jws	167.3.11.103	6	sosomt.p	Tue Apr 16 12:54:27 PDT 2002	<a href="#">Close</a>
Total	1							<a href="#">Close</a> <a href="#">All</a>

Refresh

Fig. 7.5  
Monitoring Users

Use Refresh to update the screen display. Click the Close link to close a user's session. This might be needed if a user has locked a database record and left their session running.

## Troubleshoot Errors

A variety of types of messages can be generated when running MFG/PRO programs from Desktop. 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 Desktop 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.

When a client generates an error, the general information page initially displays. This includes an error number, description, and other general information.

**Fig. 7.6**  
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. See Figure 7.7 for an example.

- Click Support Information for support-related details. The same details are included in the message sent if you choose to send a message to your help desk by clicking Forward to Support. See Figure 7.8 for an example of the message text.

### Information Detail

Clicking More Details in the information page displays a page like the one in Figure 7.7.



**Fig. 7.7**  
Error Information  
Detail Page

Click Forward to Support to bring up a new e-mail message containing error details and information from the Connection Manager log file. Users can send this e-mail to your internal help desk so that the issue can be corrected. Clicking Support Information in the Information page displays the same log details.

**Note** Desktop uses the e-mail system configured to work with the browser.

### Support Information Detail

Figure 7.8 illustrates the type of information that is included in the body of an e-mail message to be sent to support.

**Fig. 7.8**

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 Desktop. Use this information to help resolve any errors you encounter. Only the most common errors and their causes are listed.

Failure Exception Errors

This class of exception occurs when a utility type function fails, in particular while processing XSL or XSLT documents.

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.

Table 7.1  
Failure Exception  
Errors

## Connection Exception Errors

This class of exception is related to the activities of creating, retrieving, and resetting server sessions.

**Table 7.2**  
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 <code>config.xml</code> 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 <code>-param mfgwrapper=true</code> 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.
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 <code>config-files.xml</code> in the <code>\webapps\YourApp\WEB-INF\config</code> 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.

Connection Exception Error	Description
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.

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 into the Desktop.
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.
ex005 Failure occurred sending spacebar message	A pause event occurs within the MFG/PRO application and the Desktop 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.

**Table 7.3**  
Process Exception  
Errors

## Adapter Exception Errors

Each client request is filtered through a servlet. Adapter exceptions are raised if an error occurs in this servlet.

**Table 7.4**  
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 the Desktop. 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 <code>config-files.xml</code> .

## Start a Desktop Client

### Prerequisites

◆ See “Desktop Administration” on page 112.

- Ensure that you have completed the Desktop administration tasks outlined in this chapter and have configured and started Connection Manager.
- Verify that Internet Explorer 5.5, Service Pack 2 or Internet Explorer 6.0 or above is installed. QAD recommends you use the latest 6.0 version available.
- Set the minimum 1024 x 768 display setting.

## Setup

Desktop client setup is an automated process. To update client machines, simply point the client browser to the Desktop Web server and follow the instructions, if any. The correct Java components are automatically installed on the client machine.

## First-Time Startup

When starting Desktop on a client machine for the first time, some automated setup tasks may occur. Typically, this includes upgrading the client version of JRE, which includes the Java Plug-in. A security warning also displays. After the client chooses to accept files from QAD, this warning does not display again.

The following steps are typically required the first time Desktop is run on a client machine. These steps may vary depending on factors such as the client machine Java Plug-in version and the browser version.

- 1 Point the browser to the following URL to launch Desktop:  
`http://WebServerName/WebAppName`
- 2 When the Desktop window displays, click Login in the upper right corner.
- 3 The JRE begins to download. The installation program runs automatically.
  - a Accept the license agreement.
  - b Enter an installation directory or accept the default.
  - c Set Microsoft Internet Explorer as the default browser.
  - d The Desktop .jar file is downloaded from the server.
  - e When the Java Plug-in security warning displays asking if you want to install and run the QAD Inc. signed applet, select Grant Always.
- 4 The MFG/PRO log-in screen displays. Log in to begin using QAD Desktop.

## Desktop Java Plug-in Version Control

Desktop does not work correctly with all versions of the Java Plug-in. Java Plug-in version 1.3.1\_08 is supplied with Desktop. When the Desktop system is built, you can choose whether client sessions should use the QAD-supplied plug-in or one already installed on the client.

Typically, this setting should be set to Yes so that all client sessions use Plug-in 1.3.1\_08. This ensures that no plug-in related issues occur with Desktop processing. If other versions of Java exist on the client, they are not replaced and can continue to be used by applications other than Desktop.

When Use QAD Desktop Supplied Java Plug-in is set to Yes for the Desktop system, the Desktop client startup routine automatically downloads and installs version 1.3.1\_08 if it does not exist on the client.

When this field is No, Desktop clients use the latest plug-in already installed on the client machine. If no plug-in exists on the client or the minimum plug-in version required by Desktop (1.3.1\_03) is not found, the QAD-supplied version is downloaded.

You should set this value to No only when you are sure that all clients using Desktop already have a supported version of the Java Plug-in installed, such as 1.3.1\_04. In this case, you can avoid requiring each client to download the 1.3.1\_08 version.

**Important** As of Release 2.6.1, the following plug-in versions have been tested and are considered compatible with Desktop: 1.3.1\_03, 1.3.1\_04, 1.3.1\_08, 1.3.1\_10 through 1.3.1\_12, 1.4.2\_04, 1.4.2\_05. Earlier versions of Desktop do not support using any of the 1.4 plug-in versions.

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

## Running the Desktop Client on a Windows Web Server

**Important** The Web server should not have multiple Java versions installed, unless an earlier version was installed by Progress.

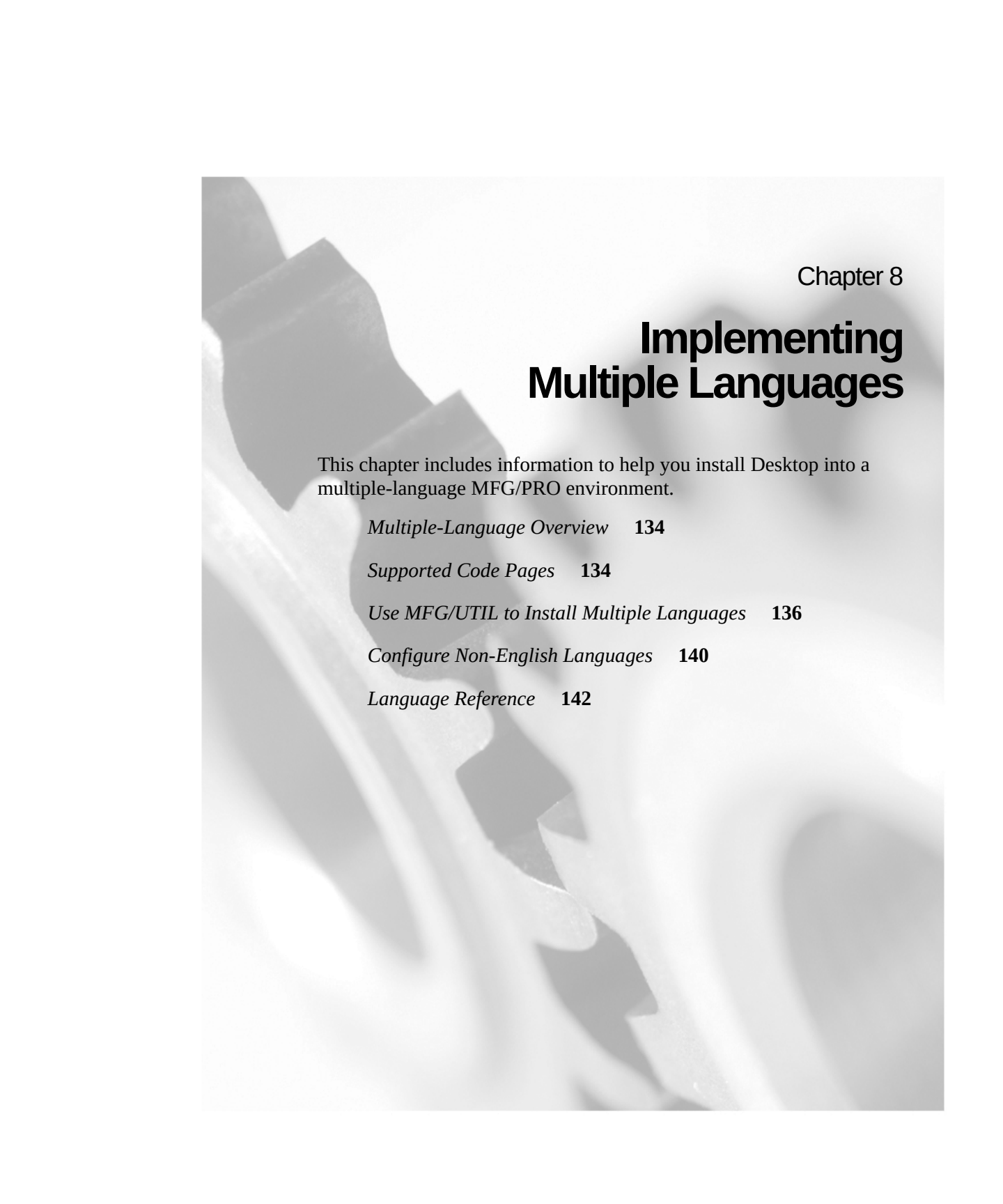
If you try to run the Desktop client on the Web server machine, you may encounter various Java errors because the client startup installs JRE 1.3.1\_08, which conflicts with the 1.4 JDK.

To avoid this problem, the Java 1.4 registry entries must be removed from the Web server. Tomcat, the only Desktop component using J2SDK 1.4, does not require these registry entries. The Desktop client startup process installs JRE 1.3.1\_08 and updates the registry entries as needed.

Once your Desktop environment is operational, use these instructions to resolve the registry conflicts before starting a client on the Web server.

- 1 Stop the Tomcat engine.
- 2 Back up the J2SDK 1.4 installation directory.
- 3 Uninstall J2SDK 1.4 (Control Panel|Add/Remove Programs|J2SDK 1.4).
- 4 Move the J2SDK backup directory to the original installation directory.
- 5 Restart Tomcat.
- 6 Start the Desktop client on the Web server host to automatically install JRE 1.3.1\_08.





## Chapter 8

# Implementing Multiple Languages

This chapter includes information to help you install Desktop into a multiple-language MFG/PRO environment.

*Multiple-Language Overview*    **134**

*Supported Code Pages*    **134**

*Use MFG/UTIL to Install Multiple Languages*    **136**

*Configure Non-English Languages*    **140**

*Language Reference*    **142**

## 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 language translations. Each MFG/PRO 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.

Each user can then view interface elements in their own language. Language affects the fields and menus for HTML screens, telnet screens, browses, report output, and Desktop administrative functions such as Connection Manager.

This chapter discusses topics you need to know for a successful multiple-language implementation, including the following:

- Code pages supported by the Desktop
- Building multiple systems with MFG/UTIL
- Additional configuration tasks you may need to complete
- Background information about how Desktop stores settings that affect language-related displays

## 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, Ukrainian, and Romanian that 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.

Installing languages with incompatible code pages in the same database is not currently a standard MFG/PRO supported option.

Multiple-language display in the Desktop is complicated by the various technology layers involved in the Desktop architecture. Data stored in the Progress database can be passed through Java, HTML, and XML representations 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 Desktop installations. Your database must use the Progress code page that QAD supports. Table 8.1 lists the code page supported for each supported language.

If your database is using a code page other than the one specified in Table 8.1, 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 Desktop using MFG/UTIL without any changes.

Language	MFG/PRO Language Code	Database Code Page
Chinese (Traditional)	TW	Big-5
Chinese (Simplified)	CH	Gb2312
Czech	CZ	1250
Danish	DA	ISO8859-1
Dutch	DU	ISO8859-1
Finnish	FI	ISO8859-1
French	FR	ISO8859-1
German	GE	ISO8859-1
Hungarian	HU	1250
Italian	IT	ISO8859-1
Japanese	JP	Shift-jis
Korean	KO	ksc5601
Lithuanian	LT	1257
Norwegian	NO	ISO8859-1

**Table 8.1**  
Supported Progress  
Code Pages

Language	MFG/PRO Language Code	Database Code Page
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 differently in different MFG/PRO releases:

- ▶ See page 148.
  - In MFG/PRO eB, it is maintained as terms in Label Master Maintenance (36.4.17.1).
- ▶ See page 151.
  - In MFG/PRO eB2 and higher, it is maintained in a text file.

## Use MFG/UTIL to Install Multiple Languages

- ▶ See “Advanced Build Options” on page 210 for details.

You can use the advanced build options in MFG/UTIL to build multiple-language versions of a Desktop system. To do this, you first generate the Desktop system 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 Desktop|Configure Desktop 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 Desktop|Build Desktop. Review the settings and complete the build for the first language.
- 4 Now choose Desktop|Build Desktop for the second language. Choose the same Desktop system 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.

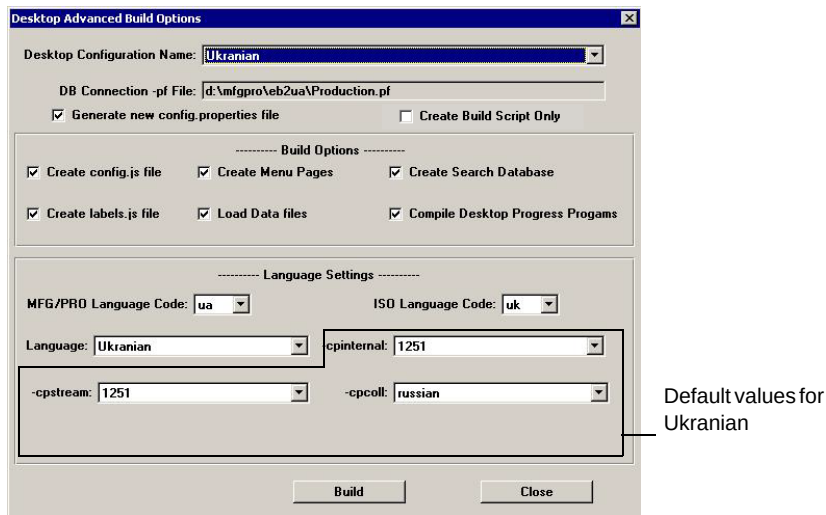
## Building Languages with Compatible Code Pages

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 for the languages that share the database code page first; then build the Desktop for English. Follow these steps:

- 1 Launch MFG/UTIL and choose Desktop|Configure Desktop Environment Parameters. Build the languages with the same code page as the database first.

- 2 After specifying parameters, choose Desktop|Build Desktop. Review the settings and complete the build for the first language. Default language settings should not need to be changed. In Figure 8.1, the language being built is Ukrainian.

**Fig. 8.1**  
Advanced Build  
Options for  
Ukrainian



- 3 Now choose Desktop|Build Desktop for the English language. Choose the same Desktop system 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.

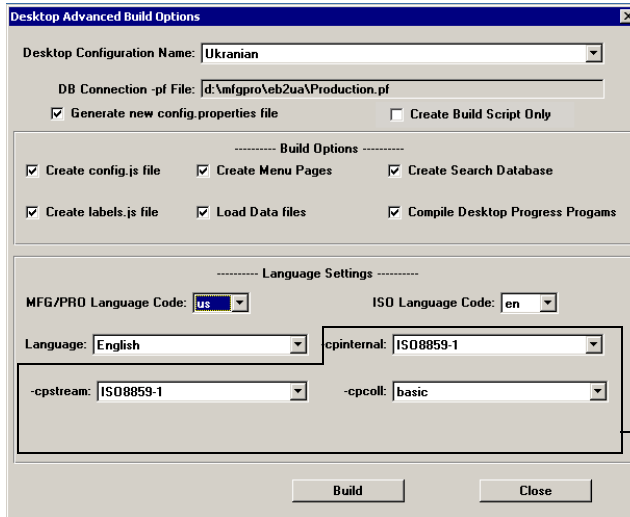


**Fig. 8.2**  
Default Values for  
English

Default values for  
English must be  
changed.

See “Advanced Build Options” on page 210 for details about the significance of each of these fields.

- 5 Change the value of the marked fields to match values of the database code page. In this example, these were default values for Ukrainian.

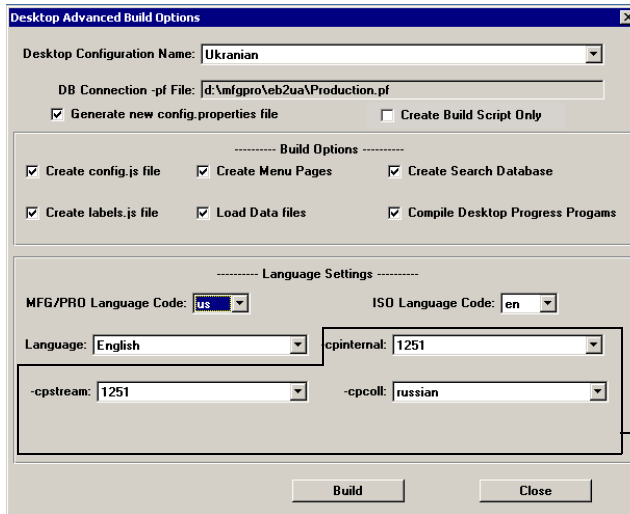


**Fig. 8.3**  
Building English  
with a Ukrainian  
Database

Change to values  
for Ukrainian  
code page.

- 6 Choose Build to complete the process.

## Configure Non-English Languages

After installing the Desktop files on the Web server, 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 using the Apache HTTP Server on the Windows platform, set the `AddDefaultCharset` directive to `utf-8` to account for any possible locale variances. See the Apache documentation for details. In UNIX platforms, this directive can remain unset.
- If you are not using the latest MFG/PRO service pack for your system, you may need to update language-related data. See the 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 MFG/PRO database, make sure you include the proper language-related parameters in the MFG/PRO startup scripts. See the Progress documentation for details.

### 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\JavaSoft\JRE1.3.1_08\lib
```

◆ See Table 8.2 on page 143 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 Desktop installation, all existing process maps remain in the default `/WEB-INF/pronav/xml` directory. Whenever you configure a Desktop 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 Desktop 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: *YourWebServer/YourDesktopSystem/admin.html*.
- 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.

## Language Reference

This section includes background information on the way the system manages the mapping of languages and code pages. The approach to this changed from MFG/PRO eB to eB2. Following the overview, a section specific to each release provides detailed information.

### Overview of Locale Information

When a user logs in to MFG/PRO from either the Windows or character interface, MFG/PRO determines the language to use for labels, messages, and other interface elements based on the MFG/PRO language code associated with that user in User Maintenance (36.3.1, 36.3.18 in eB2).

A similar process occurs when a user logs in to MFG/PRO from the Desktop. However, the Desktop uses Java language codes. These codes are defined by an ISO (International Organization for Standardization) standard. Since not all MFG/PRO language codes follow this standard, a mapping must occur.

In addition to language codes, Java also uses country codes to accommodate regional language differences. For example, while both the US and Great Britain use the English language, there are many differences between the two—for example, spelling differences and date format differences. In this case, Java uses the ISO language code for English (en) and then adds the ISO country code to define the correct display for each country—en,US for the US and en,GB for Great Britain.

When a user logs into MFG/PRO from the Desktop, the system performs the following process to determine the MFG/PRO language code and map it to the appropriate ISO language code:

- 1 The system checks the user master table (usr\_mstr) to find the MFG/PRO language code associated with the user's ID.
- 2 Once the MFG/PRO language is determined, the system determines the associated ISO language code, country code, date separator, number format, and date format.
  - In MFG/PRO eB, the system determines format information from labels created in Label Master Maintenance (36.4.17.1).

- In MFG/PRO eB2 and higher, the system determines format information from an external data file, `locale.dat`.

**3** The system uses the ISO language code to determine the language-related aspects of the interface.

If the combination of language and country that you need does not exist in label master (for MFG/PRO eB) or `locale.dat` (for MFG/PRO eB2), you may need to create the necessary records using the guidelines provided here.

Table 8.2 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.

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
Danish	DA	da	DK	Denmark
Dutch	DU	nl	NL	Netherlands
Finnish	FI	fi	FI	Finland
French	FR	fr	FR	France
German	GE	de	DE	Germany
Hungarian	HU	hu	HU	Hungary
Italian	IT	it	IT	Italy
Japanese	JP	ja	JP	Japan
Korean	KO	ko	KR	South Korea
Lithuanian	LT	lt	LT	Lithuania
Norwegian	NO	no	NO	Norway
Polish	PL	pl	PL	Poland
Portuguese	PO	pt	BR	Brazil
Romanian	RO	ro	RO	Romania
Russian	RU	ru	RU	Russian

**Table 8.2**  
MFG/PRO  
Language Codes

Language	MFG/PRO Language Code	ISO Language Code	ISO Country Code	Country
Spanish (Castilian)	CS	es	ES	Spain
Spanish (Latin)	LS	es	MX	Mexico
Swedish	SW	sv	SE	Sweden
Turkish	TU	tr	TR	Turkey
Ukrainian	UA	uk	UA	Ukraine
US English	US	en	US	United States

### Updating Environment Data: MFG/PRO eB Only

At run time, Desktop client sessions obtain two types of environment information from label master records, updated using Label Master Maintenance (36.4.17.1):

- Mapping of MFG/PRO language codes to ISO language and country codes, and to appropriate number and date formats. In most cases, you can use the default mapping for your environment. However, in some cases you may need to modify the ISO country code to ensure correct display and field validation in the Desktop.
- Mapping of Progress code page values to Java code page values. Use the default mapping supplied by QAD.

Use the following instructions to verify and define Desktop constants for your environment.

▶ See your Service Pack Installation Guide.

**Important** All constants in the label master are stored under the US language ID. If you are installing the Desktop in a non-English environment, you still need to load this required subset of US label data. The file containing this data is provided on MFG/PRO eB Service Pack 2 and above.

## Access Language Code Records

Use the following instructions to view the term for your language:

- 1 Launch an MFG/PRO session and access Label Master Maintenance (36.4.17.1).
- 2 In the Language ID field, specify US and press Go.
- 3 Each MFG/PRO language code has a predefined term used to map to the appropriate ISO language, country code, and associated date and number format information. In the Term field, identify the record for your language by selecting the MFG/PRO language code followed by the \_LANGUAGE suffix.

**Tip**  
Select US regardless of the language used in your environment.

The lbl\_mstr record used to map the MFG/PRO language code to the ISO language code and display formats is shown in the Long Label field. The following example record is for US-English, MFG/PRO language code US.

Field	Value
Language ID	US
Term	US_LANGUAGE
Long Label	en,US,/,A,mdy
Description	MFG/PRO language code to Java language, country codes. Do not translate.

**Important** The language ID associated with all language terms is US. To modify the language term for German, you must specify US in the Language ID field as in the following example.

Field	Value
Language ID	US
Term	GE_LANGUAGE
Long Label	de,GE,,E,dmy
Description	MFG/PRO language code to Java language, country codes. Do not translate.

## Language Code Elements

This section contains information on each of the elements in the Long Label field of the language code record.

The long label record for each language in the lbl\_mstr has the following format:

*ISO language code,ISO country code,date separator,numeric format code,  
date format*

There are five components to the long label:

▶ See Table 8.2 on page 143.

- ISO language code.
- ISO country code.
- Date separator. This element is not currently referenced in the Desktop. The only date separator that is supported is the slash (/).
- Numeric format. This element uses one of the following values:
  - A (American). The comma separates thousands and the period is a decimal point; for example, 100,000.00.
  - E (European). The period separates thousands and the comma is a decimal point; for example, 100.000,00.
- Date format. This element uses one of the following values:
  - mdy (Month/Day/Year)
  - dmy (Day/Month/Year)
  - ymd (Year/Month/Day)

## Modify Language Code Records

If you are installing MFG/PRO in an English-speaking country that does not use American date and decimal formats, then you need to modify the value of the Long Label field associated with the US\_LANGUAGE term. You can change the country code, numeric format, and date format values.

For example, in England, define the label as en,GB,/,E,dmy. If you are unsure of the correct ISO country code to use, you can refer to the information at the following URL:

[http://www.chemie.fu-berlin.de/diverse/doc/ISO\\_3166.html](http://www.chemie.fu-berlin.de/diverse/doc/ISO_3166.html)

**Important** Be careful when you make modifications to preserve consistency between the language code and date format. These two values are used differently within the Desktop and setting them inconsistently can result in inconsistent date formats between Desktop programs and printed reports.

The date format determines the date format used by the Progress session. The client uses the language code to set the locale, including the date. Setting the language code to fr (French) and the date format to mdy has the following results:

- Browse and report input screens display dates in dmy based on the date format associated with the fr locale.
- The report output, generated from Progress, displays with mdy based on the date format in the label master.

Table 8.3 lists the term and long label defined in the label master for each supported MFG/PRO language.

Language	MFG/PRO Language Code	Label Master Term	Long Label
Chinese (Traditional)	TW	TW_LANGUAGE	zh,TW,-,A,ymd
Chinese (Simplified)	CH	CH_LANGUAGE	zh,CN,-,A,ymd
Czech	CZ	CZ_LANGUAGE	cs,CZ-,E,ymd
Danish	DA	DA_LANGUAGE	da,DK,-,E,dmy
Dutch	DU	DU_LANGUAGE	nl,NL,-,E,dmy
Finnish	FI	FI_LANGUAGE	fi,FI,,E,dmy
French	FR	FR_LANGUAGE	fr,FR,/,E,dmy
German	GE	GE_LANGUAGE	de,DE,,E,dmy
Hungarian	HU	HU_LANGUAGE	hu,HU,,E,ymd
Italian	IT	IT_LANGUAGE	it,IT,/,E,dmy
Japanese	JP	JP_LANGUAGE	ja,JP,/,A,ymd
Korean	KO	KO_LANGUAGE	ko,KR,-,A,ymd
Lithuanian	LT	LI_LANGUAGE	lt,LT,,E,ymd
Norwegian	NO	NO_LANGUAGE	no,NO,,E,dmy
Polish	PL	PL_LANGUAGE	pl,PL,-,E,ymd
Portuguese	PO	PO_LANGUAGE	pt,BR,/,E,dmy

**Table 8.3**  
Pre-defined  
Language Terms

Language	MFG/PRO Language Code	Label Master Term	Long Label
Romanian	RO	RO_LANGUAGE	ro,RO,,E,dmy
Russian	RU	RU_LANGUAGE	ru,RU,,E,dmy
Spanish (Castilian)	CS	CS_LANGUAGE	es,ES,/E,dmy
Spanish (Latin)	LS	LS_LANGUAGE	es,MX,/A,dmy
Swedish	SW	SW_LANGUAGE	sv,SE,-E,ymd
Turkish	TU	TU_LANGUAGE	tr,TR,,E,dmy
Ukranian	UA	UA_LANGUAGE	uk,UA,/E,dmy
US English	US	US_LANGUAGE	en,US,/A,mdy

### Map Code Pages

Java uses a different set of values to represent code pages than Progress does. When a user logs into the Desktop, the Progress code page value must be mapped to the Java one. This is accomplished through a set of records stored in `lbl_mstr`.

The Progress code page value is stored as a term consisting of the code page value with `_CODEPAGE` appended. The long label associated with that term represents the Java equivalent of the Progress value.

By default, the Progress code page value for your environment is based on the code page value used by your MFG/PRO databases. The Progress value is then mapped to the Java value using the `_CODEPAGE` terms in `lbl_mstr`. These terms are predefined and you do not need to perform any actions for the mapping to occur.

Table 8.4 lists the Progress code page to Java code page mapping that the Desktop uses on UNIX systems and Windows systems. Values are tailored during installation for Windows.

**Important** Do not modify these records.

**Table 8.4**  
Code Page  
Cross-Reference

Term (Progress Value)	Long Label (Java Value) UNIX Platforms	Long Label (Java Value) Windows Platforms
ISO8859-1_CODEPAGE	ISO8859_1	utf_8
1250_CODEPAGE	windows-1250	utf_8
1251_CODEPAGE	windows-1251	utf-8

Term (Progress Value)	Long Label (Java Value) UNIX Platforms	Long Label (Java Value) Windows Platforms
1252_CODEPAGE	windows-1252	utf-8
1253_CODEPAGE	windows-1253	utf-8
1254_CODEPAGE	windows-1254	utf-8
1257_CODEPAGE	windows-1257	utf-8
BIG-5_CODEPAGE	Big-5	utf-8
GB2312_CODEPAGE	EUC-CN	utf-8
SHIFT-JIS_CODEPAGE	SJIS	utf-8

## Updating Environment Data: MFG/PRO eB2 and Higher

In MFG/PRO eB2 and eB2.1, information that was previously stored in the label master table is now retrieved from two external data files: `locale.dat` and `encoding.dat`. Both files are located in the `./mfgpro` directory on the MFG/PRO release CD. After installation, they are located in the root directory of the user-named MFG/PRO installation directory.

The data in these files is used for the same purpose as the label master data. However, in MFG/PRO eB2 and eB2.1, more information is associated with users in User Maintenance (36.3.1, 36.3.18 in eB2) and not required in the data file.

### 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 MFG/PRO user.

When a user logs into MFG/PRO eB2, the system validates the log-in identifier in User Maintenance and determines the associated 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 all MFG/PRO interfaces: character, Windows, and Desktop.

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. Most of the 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](http://www.chemie.fu-berlin.de/diverse/doc/ISO_3166.html)
- 2** Create or update the user profile in User Maintenance (36.3.1, 36.3.18 in eB2).
  - 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.
 

Open the `locale.dat` file and verify that an entry exists for the language and country code assigned to the user.
- 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.

▶ See "Locale Record Format" on page 150.

### 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 upon 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.

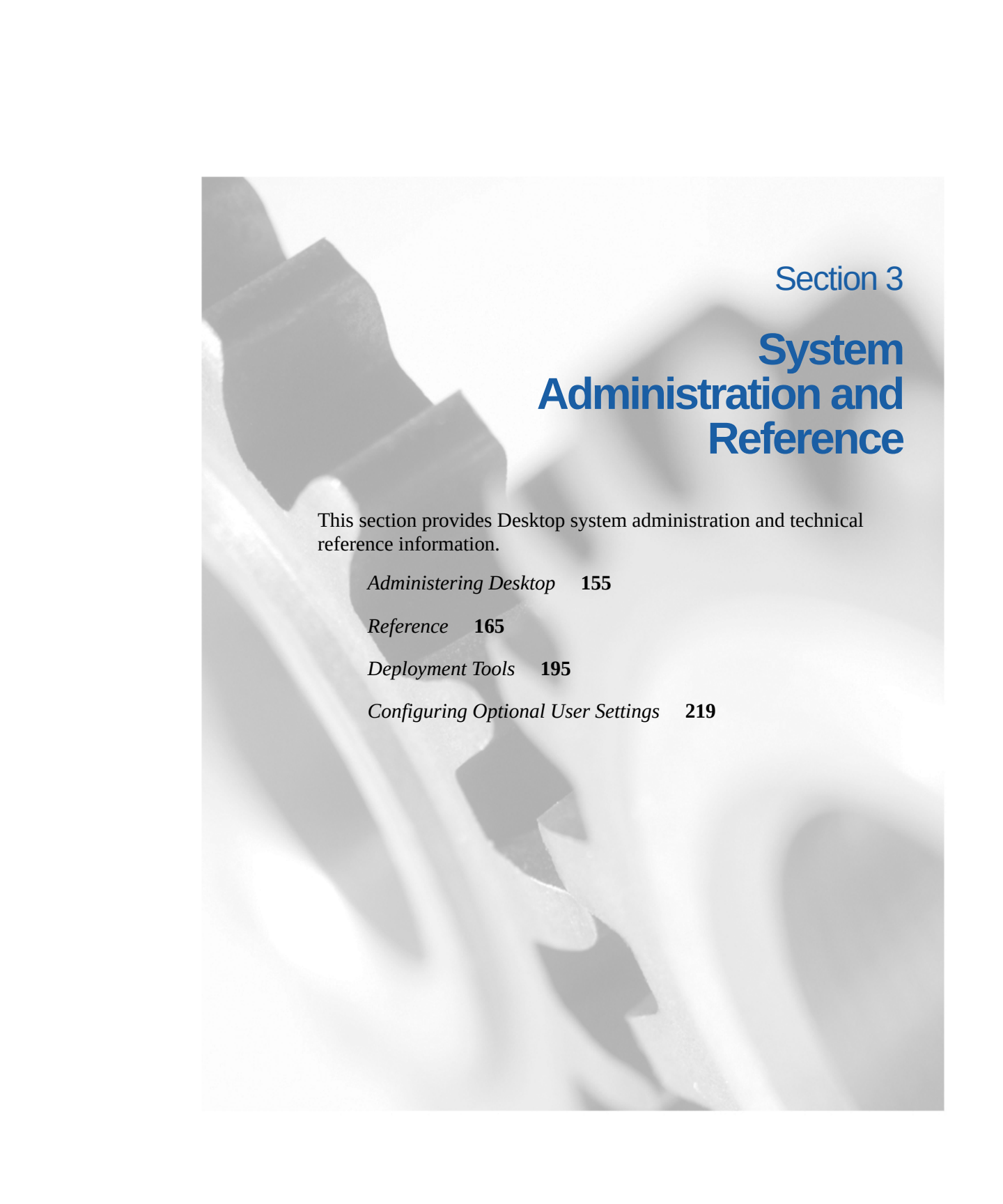
## Encoding Data

The system gets information on encoding from the `encoding.dat` file supplied with MFG/PRO. 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.

Desktop also uses the `dtencode.dat` file supplied with the Desktop 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.



The background of the page features a grayscale image of several interlocking mechanical gears. The gears are of different sizes and are positioned in a way that creates a sense of depth and movement. The lighting is soft, highlighting the metallic texture of the gears.

## Section 3

# System Administration and Reference

This section provides Desktop system administration and technical reference information.

*Administering Desktop*    **155**

*Reference*    **165**

*Deployment Tools*    **195**

*Configuring Optional User Settings*    **219**





## Chapter 9

# Administering Desktop

This chapter contains Desktop system administration and other reference information.

*Common System Administration Tasks*    **156**

*Install a New Desktop Release*    **156**

*Install an MFG/PRO Service Pack*    **158**

*Desktop Output Administration*    **159**

*Manage Menu Updates in Desktop*    **160**

## Common System Administration Tasks

This section provides information about common system administration tasks you may need to perform for the Desktop environment.

### Install a New Desktop Release

QAD highly recommends you deploy only the Desktop components included on the latest QAD Desktop 2 release. The latest release always contains the most recent component versions, patches, and performance improvements. Review `Desktop2_ReleaseNotes.html` on the Desktop media to see if any updated Desktop system components have been included.

Consider the following items before you begin installing a new Desktop release:

- Desktop version component dependencies

Some Desktop releases cannot be installed directly into directories that contain any components from a previous Desktop version. For example, Desktop 2.5 cannot be installed over an existing pre-Desktop 2.5 environment. Mixing previous version components with Desktop 2.5 may cause significant configuration problems. Instead, you must install Desktop 2.5 into new directories that do not contain components from a previous Desktop release.

Always check the Desktop release notes to see if this condition applies to the Desktop release you plan to implement in an existing Desktop environment.

- MFG/PRO menus and configuration settings

Running the Desktop build, required when installing a new Desktop release, resets the HTML and telnet screen configuration to its original settings.

- PROPATH issues

The Desktop installation process installs updated MFG/PRO code. Depending on the Desktop release and the service pack you have installed, your client, Desktop, and WebSpeed PROPATHs are affected differently. The Desktop installation directory must be placed at the front of all your system PROPATHs.

- Tomcat

The latest QAD-optimized Tomcat version is always included on the Desktop media. Always install the new version. This can be done manually or automatically using the Perl deployment tools.

- Java

If the Java Plug-in is updated, initial client startup after setting up the new Desktop might require additional time to allow for the new Java download.

- SVG plug-in

The latest QAD-tested version is always included on the Desktop media. Desktop Process Editor client machines will need to download the new plug-in.

- Client and server cache

The server cache must be cleared after installing a new Desktop release. You do not have to manually clear the client cache. When you install a new Desktop release, new files are automatically cached by the client. The old files are deleted automatically based on client browser history settings.

▶ See “Clear the Tomcat Server Cache” on page 190.

- HTML page caching

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 performance impact, if any, is temporary, and only affects the initial client sessions while new pages are being cached.

- Process maps

If you are upgrading an existing Desktop environment, existing process maps are located in the *SysName* /WEB-INF/pronav/xml directory in the Tomcat Webapps directory. *SysName* refers to the previous system name. If you are using a Desktop release that requires a complete new installation, such as Release 2.5, 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 is 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 Desktop build, new Desktop 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 Desktop release is installed.

There are, however, times when you do need to manually clear the client cache, especially when initially setting up a Desktop environment. These are when:

- The Desktop menu system is modified and rebuilt.
- A browse is attached to or detached from a field.
- During initial Desktop 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 an MFG/PRO Service Pack

Installing an MFG/PRO service pack can affect your Desktop system in two ways:

- The service pack may include software fixes that correct issues in Desktop since some Desktop issues are caused by problems in the underlying MFG/PRO programs, not the Desktop programs.
- The service pack installation media may include a different version of MFG/UTIL than the version of MFG/UTIL delivered with your Desktop release.

The second possibility may result in an older version of MFG/UTIL being installed over the one installed with QAD Desktop, which either removes the Desktop menu or removes Desktop updates.

For example, the Desktop menu entry in MFG/UTIL is removed under MFG/PRO eB2 SP6 and earlier MFG/PRO eB SP8 and earlier. Later service packs do not overwrite the Desktop menu, but may remove some Desktop code fixes.

After installing a service pack that does not contain Desktop-specific MFG/UTIL updates, use the Desktop installation media to reinstall MFG/UTIL. Use the following steps to reinstall this code:

- 1 Start the Desktop install and select the configuration that you have already installed.
- 2 Select the Install MFG/PRO – Progress Desktop 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.

## Desktop Output Administration

When Desktop clients output MFG/PRO browses to the screen, the number of pages sent to the client is determined by the Max Pages setting for the JavaUI printer in Printer Setup Maintenance (36.13.2).

By default, the Max Pages field for the JavaUI printer is set to zero, unlimited. This may cause the Desktop client to run out of memory if too many pages are output at once. You should set the Max Pages field to a reasonable number of pages for your Desktop client machines.

HTML reports and inquiries can be directed to the screen using any printer defined with Scroll Output set to Yes in Printer Setup Maintenance. Users can direct output to Page; this device is supplied by default with MFG/PRO and is defined with Max pages set to 200. You may want to set up a special output device named Desktop to help users

select an appropriate device. Choosing Terminal for output of HTML reports and inquiries is not recommended, since the output will not be formatted properly.

## Manage Menu Updates in Desktop

♦ See *User Guide: QAD Desktop* for details.

This section includes instructions for adding custom programs, browses, and lookup browses to MFG/PRO. You must rebuild the menus and search databases in order for the new options to display in Desktop.

### Changing MFG/PRO Menus

The Desktop system must be configured dynamically so that it can reflect the menu system in your MFG/PRO 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; 36.20.3 in MFG/PRO eB), these changes are reflected in the Desktop menu system when it is built.

Menu substitution is managed differently in Desktop than in the other MFG/PRO 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 HTML menu system for the Desktop, browses (instead of inquiries) and standard programs (instead of custom ones) are always placed on the menus.

♦ See “Clear the Client Cache” on page 158.

If you make menu or lookup changes after Desktop is initially installed, you must rebuild the HTML menus and the search database to reflect those changes. You must also clear the cache for all client and server machines.

♦ See “Advanced Build Options” on page 210.

Use the Advanced Build Options available from the MFG/UTIL Desktop|Build Desktop option to rebuild your menus whenever you make changes in:

- Menu System Maintenance (36.4.4)
- Menu Substitution Maintenance (36.20.6; 36.20.3 in MFG/PRO eB)

- Drilldown/Lookup Maintenance (36.20.1) in MFG/PRO eB2 and eB2.1
- Drilldown Maintenance (36.20.1) and Window Help Maintenance (36.4.21) in MFG/PRO eB
- Program Information Maintenance (36.3.21.1)

## Adding Browsers

Use these steps to add a browse to the Desktop system:

- 1 Add the browse to MFG/PRO 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 Desktop menus and search databases. See the detailed steps in “Regenerating Desktop Components” on page 162.

## Adding Lookup Browsers

Use these steps to associate a lookup field for use in the Desktop:

- 1 Associate the lookup with a program or field in:
  - Drilldown/Lookup Maintenance (36.20.1) in MFG/PRO eB2 and eB2.1
  - Drilldown Maintenance (36.20.1) or Window Help Maintenance (36.4.21) in MFG/PRO eB
- 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 the Desktop 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 Desktop menu system. If created following QAD programming constructs, your custom programs can be run in the Desktop HTML interface. If the QAD programming constructs were not followed, your custom program may not function properly in the Desktop HTML interface. Instead, run them 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 MFG/PRO menu structure.

Option 2: Add the custom program to the MFG/PRO 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 HTML interface:

- Set Web Logic Implemented to No.
- Set Type to Desktop.

Use these settings to run the custom program in the embedded telnet interface:

- Set Type to blank.
- Set Web Logic Implemented to No.

**3** Rebuild the Desktop menus and search databases as detailed in the following section.

## Regenerating Desktop Components

Use these steps to regenerate the Desktop menu pages and the search database after adding any custom programs and browses or making other menu changes in MFG/PRO, 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 Desktop|Build Desktop.
- 2 Select your Desktop system in Desktop Configuration Name.
- 3 Review the field values in the Verify and Build Desktop 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.
- 7 Clear the cache for all client machines.
- 8 Clear the Tomcat server cache.

♦ See page 158.

♦ See page 190.

Once the Desktop build completes, launch a Desktop client and test your program and menu changes.



# Reference

This chapter provides additional reference. This information is provided to give you a better understanding of the Desktop system and its interaction with MFG/PRO.

*Plan a Multi-Tier Deployment*    **166**

*Installed Components*    **173**

*Connection Parameters*    **175**

*Working with WebSpeed*    **178**

*Security*    **184**

*Configure Multiple Desktop Systems*    **187**

*Administration with Tomcat*    **189**

*Use Diagnostic Tools*    **192**

## Plan a Multi-Tier Deployment

Desktop can be deployed in multiple configurations using two or more servers, typically a database server and one or more application servers. In a typical two-tier deployment, the first server is the MFG/PRO database server, which should already be configured and running your MFG/PRO system. It hosts most—if not all—of the following MFG/PRO and Progress components:

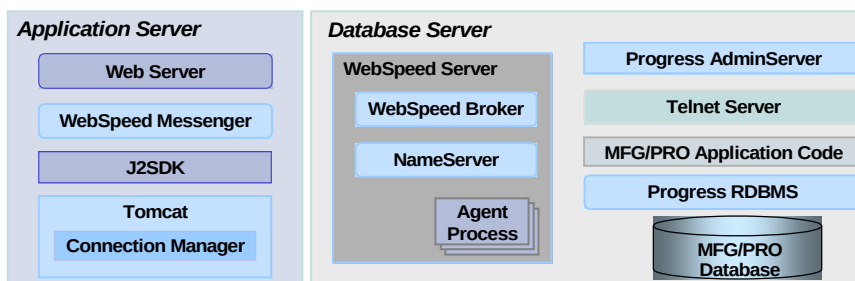
- Progress 4GL development environment
- Enterprise database
- Progress Query/Results application
- WebSpeed Transaction Server
- Telnet server
- Progress client and MFG/PRO code base
- Progress databases

The second server, referred to as the application server, is used for the Desktop components. These include:

- CGI-compatible Web server
- Tomcat server
- WebSpeed Messenger

Figure 10.1 shows a typical two-tier Desktop deployment with one database server and one application server.

**Fig. 10.1**  
Basic Two-Tier  
Deployment



For sample two-tier deployment component layouts see:

- Figure 10.1, “Basic Two-Tier Deployment,” on page 166.
- Figure 10.2, “Two-Tier Windows Deployment,” on page 171.
- Figure 10.3, “Isolated Database Server Deployment,” on page 172.

## Deployment Issues

When planning a multi-tier Desktop deployment, be sure to consider each of the following issues:

- System Characteristics
- Location of Prerequisite Components
- Existing Infrastructure
- Desktop Architecture
- Desktop Directory Components

### System Characteristics

Consider your expectations for system characteristics such as accessibility, reliability, scalability, performance, and maintenance. Different Desktop deployments may improve one characteristic while degrading another. Always implement Desktop in the deployment that best achieves your configuration and performance objectives.

For example, a single-tier deployment may ease administration because all of the components are located on the same server. However, to provide better performance for remote users, you may want to deploy a multi-tier configuration with components located on the remote user’s local-area network (LAN).

### Location of Prerequisite Components

The location of prerequisite components—such as the MFG/PRO databases, the MFG/PRO character client code, and the related Progress software—affects the deployment you implement. Keep in mind that spreading components across multiple machines can increase network delays and reduce overall system performance if not implemented properly by a trained specialist.

## Existing Infrastructure

If you decide to deploy the Desktop in a multi-tier configuration, the overall system performance is directly affected by:

- Network topology, bandwidth, and throughput
- The processing power of the machines involved in the deployment

For example, you may find that while a component runs faster on a dedicated server—removing CPU contention with other components—the network delays that result from moving the item actually decrease system performance.

## Desktop Architecture

Desktop components can be spread across multiple machines to suit your needs. When considering deployment options, be aware that the WebSpeed Messenger executable must be deployed with your Web server if you do not have a full WebSpeed installation on the Web server host.

There are several ways to deploy Desktop in a multi-tiered environment. Each environment has different characteristics, which makes documenting all possible situations impossible. The following sections discuss the basic concepts and tasks required for performing multi-tiered Desktop deployments; several examples are also discussed.

## Desktop Directory Components

Desktop is comprised of several main directory structures that contain the code and files used by Desktop. These directories are placed in the proper server locations by the deployment script and MFG/UTIL during the Desktop deployment and build process.

The following are the main top-level Desktop directories deployed during the installation and configuration process. If you are deploying Desktop across multiple servers, the deployment tools facilitate the installation of remote components using JAR archive files. In the following directory structure example, SysName refers to the Desktop system name; your system name will be different, and should reflect some aspect of the system, such as Test, Train, or Prod.

- `\WebServerDir\cgi-bin\SysName`

- `\WebServerDir\htdocs\SysName`
- `\TomcatInstallDir\SysName`
- `\TomcatInstallDir\Webapps\SysName`

## Third-Party Component Considerations

Third-party components used by Desktop include items such as the Web server, the Tomcat server, the telnet server, Progress, and WebSpeed.

For multi-tier deployments, Desktop components can be placed across a network on multiple servers. If you are configuring a multi-tier deployment, each Desktop component has specific sizing, configuration, and communication requirements. While system sizing for each Desktop component remains fairly constant in most deployments, configuration requirements will vary widely. This section discusses the main configuration issues you must resolve for each component when deploying across multiple servers.

### Tomcat

The Tomcat engine configuration in a multi-tier deployment remains unchanged from the single-tier deployment configuration. One important issue is that the telnet server startup scripts generated by MFG/UTIL and referenced in Connection Manager and in MFG/PRO both point to the correct log-in sequence for the telnet server, which may be placed locally on the same server or on a separate server. Additionally, if you have multiple Tomcat installations, you need to update the Tomcat server.xml configuration file to prevent port conflicts.

▶ See “Update Tomcat Port Numbers” on page 101.

The Tomcat configuration settings become more complex when you set up multiple application servers that connect to the same database server. Each application server requires its own Tomcat installation. Depending on where the telnet server is located—either on the database server or locally on the application server—the Tomcat connection script must point to the correct telnet server.

Tomcat requires several J2SDK 1.4.x components. If you are using multiple Tomcat instances for your deployment, make sure you use the same Tomcat and J2SDK 1.4.x versions on each application server.

## Tomcat LE

QAD Desktop media always contains a customized version of the Tomcat Lite edition (LE) engine. This version includes several QAD-modified files, as well as a subset of libraries included with a full Tomcat version. The LE version requires J2SDK 1.4 on the same server where Tomcat is implemented.

## Sample Web Server Setup

Desktop runs with any CGI-compatible Web server. Each Web server has specific system and configuration requirements. Desktop requires the existence of a `cgi-bin` executable directory, and an `htdocs` document directory under your Web server. Refer to your Web server's documentation for instructions on creating a ScriptAlias if required, or on setting up your server for this purpose.

## Progress

An important consideration when deploying Desktop is that the Desktop installation process requires compiling the installed Desktop code. Progress is required for this compilation, so install the Desktop code on the server with your Progress implementation. Alternatively, install Progress on the server where you intend to run the Desktop installation and configuration process.

## WebSpeed

♦ See “Install and Configure WebSpeed Messenger” on page 178 for details.

One possible configuration option has a full Progress and WebSpeed installation located on the database server, with a WebSpeed Messenger installed on each application server. The WebSpeed Messenger enables communication between the application server and the WebSpeed broker agents on the database server.

If licenses allow, the application servers can also have a full WebSpeed installation, which removes the WebSpeed Messenger requirement. This configuration is not required; additional Progress licenses are not needed to use the WebSpeed Messenger on the application server.

In some regulated environments, adding components to the database server is not an option. Instead, the MFG/PRO databases are isolated completely from any other components. All communication with the database is accomplished using client networking. In these environments a full Progress installation—including WebSpeed—is placed on the application server. Alternatively, two or more application servers can be used to distribute the application server load.

A second configuration has a full WebSpeed installation on each application server. Communication with the database server is via host and services (-H -S) protocol to the database server.

With either configuration, the connection parameters must be properly set in the `Progress\properties\ubroker.properties` file on each application server. This may require additional manual configuration updates to the MFG/UTIL-generated `ubroker.properties` parameter file.

Figure 10.2 shows the final component distribution and interaction for a basic two-tier Desktop deployment.

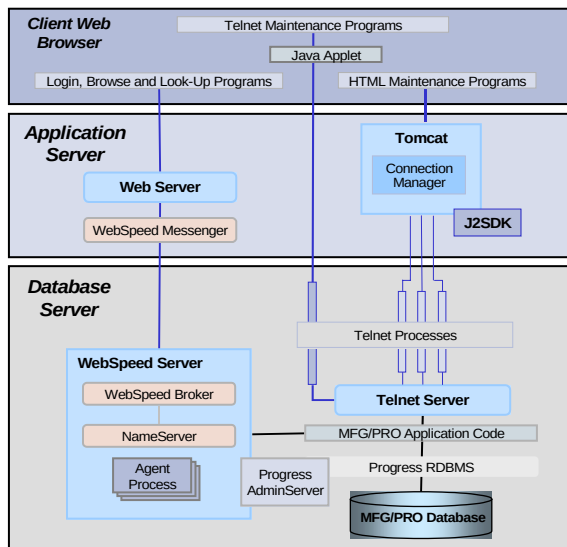


**Fig. 10.2**  
Two-Tier Windows  
Deployment

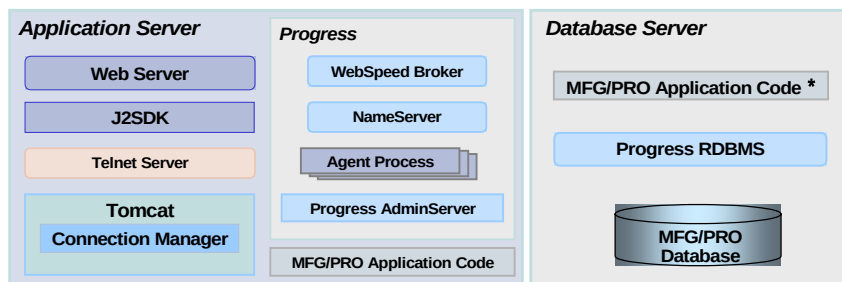
## Dedicated Database Server Deployment

This section discusses a two-tier deployment with a dedicated database server. The Desktop installation, compile, and configuration occur on the application server. No additional components are added to the database, nor are any configuration changes made to this server.

This two-tier deployment works well for environments where MFG/PRO databases are all located on a dedicated database server. This is especially true in regulated environments where any changes to the database server configurations would require additional validation overhead. All Desktop and related components are installed and configured on an application server. Communication between the servers occurs using client networking.

Figure 10.3 shows the component layout for this Desktop deployment. Since all components reside on this server, you may need to acquire additional licenses for the Progress components. A full Progress installation, including WebSpeed, is required on the application server.

**Fig. 10.3**  
Isolated Database  
Server Deployment



\* Not required, but increases performance for tasks such as MRP and Batch Processing

## Prerequisites

- The MFG/PRO code must be installed and fully functional on the application server. See your MFG/PRO installation guide for complete setup and configuration instruction.
- The MFG/PRO database must be running on a separate server.

## Configuration Notes

In this configuration the databases are on a separate network-connected server, and all communication occurs via client networking, instead of shared memory.

## Installed Components

This section provides a brief description of the directory structure created or updated by the Desktop system-generation process.

**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 `com/qad/nav` directory path below the MFG/PRO installation directory for Desktop files.

```
MFGPROInstallDir/DesktopSysName/com/qad/nav
- /conf <contains template for config.properties>
- /data <directory for data files>
- /html <HTML templates>
*.r <Progress r-code>
```

It also creates a `com/mfgpro/xx` directory path below the MFG/PRO installation directory for Desktop files, where `xx` refers to the two-letter language code installed. Several directories are placed here that contain compiled Desktop-specific MFG/PRO code.

### Web Server Executables Directory

Several files are added to the executables directory. Many Web servers use an executables directory named `cgi-bin`.

```
YourWebServer/ExecutablesDir
<executable files for configuring, building, searching>
- /DesktopSysName
 - /bin <Perl executable for UNIX only>
 - /data <templates for building Desktop>
 config.js <JavaScript code incorporating environment
 values>
 desktop.war <Web archive file containing the Desktop
 application files and directories>
```

## Web Server Documents Directory

Several files are placed in the Web server documents directory. Many Web servers use the `htdocs` directory as the documents directory.:

*YourWebServer/DocDir/DesktopSysName*

```
*.html <top-level HTML files>
ReleaseNotes.html <information about updates to Desktop>
- /client_setup <JRE and client install script>
- /css <Desktop and process flow style sheets>
- /custom <sample process flow>
- /doc <directory for user information files>
- /images <directory for graphic files>
- /java <contains the multinet.jar file>
- /js <directory for JavaScript files>
- /menus/xx <language-specific menu HTML>
```

## Tomcat 4.1 Servlet Container

The Desktop media includes the latest Lite edition of the Tomcat engine. If you are using J2SDK 1.4, you do not need the full Tomcat release. That version provides additional libraries not utilized in a Desktop implementation. If you cannot use J2SDK 1.4, you should download the latest full Tomcat 4.1 version from the Apache Web site:

<http://jakarta.apache.org/tomcat>

The downloaded version requires some QAD-supplied components. See “Administration with Tomcat” on page 189.

The following directory and files are added to the Tomcat installation directory:

```
TomcatInstallDir/webapps/DesktopSysName/yourwebapp/WEB-INF/conf/
config.xml <Connection Manager configuration file>
process-config.xml <Configuration file for Process Editor>
```

## Language Directories for Menus

► See “Language Reference” on page 142.

HTML menu files are generated and placed on the Web server in a two-letter language subdirectory of the menus directory during the build process; *YourWebServer/htdocs/SysName/menus/xx*. In this directory name, the two-letter ISO language code is used.

This code is derived in two different ways based on MFG/PRO release:

- In MFG/PRO eB, the build maps the MFG/PRO language code to the ISO language code required by Java using a set of terms and labels defined in Label Master Maintenance (36.4.17.1).
- In MFG/PRO eB2 and higher, the mapping data is derived from the `locale.dat` file.

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`.

## Connection Parameters

Desktop clients connect to the MFG/PRO databases using the `.pf` file you specify in MFG/UTIL when the Desktop system is being built. You can choose to build the Desktop using a `.pf` not located in the MFG/PRO directories by specifying a location using the MFG/UTIL Desktop|Edit Search Directories for Desktop Configuration Lookup menu option.

♦ See “Add Parameter File Directories” on page 217.

Review this section if you plan to build customized Desktop parameter files.

### Local Host Connection Parameters

If the WebSpeed broker is on the same server as the MFG/PRO databases, you can use local host or shared memory connections to access the databases. This can improve Desktop performance by eliminating network overhead between the processes and the databases.

To enable local host connections, the Desktop parameter files must contain the connection parameters described in Table 10.1.

**Table 10.1**  
Local Host  
Connection  
Parameters

Parameter	Description
-db	The physical database name parameter. Follow this parameter with the path and physical name of an MFG/PRO database.
-trig	The triggers parameter. Follow this parameter with the name of the directory containing the database triggers for the main MFG/PRO database. For MFG/PRO, this is the <code>triggers</code> subdirectory. The parameter value is: <code>-trig triggers</code> . This parameter is used only with main databases.
-ld	The logical database name parameter. Follow this parameter with the logical name of an MFG/PRO database.  <b>Oracle:</b> This parameter is not typically used in Oracle environments. Instead the database logical name is embedded into the schema holder and the MFG/PRO code embeds the logical name at runtime.

**Oracle:** Do not use the `-ld` parameter. In most MFG/PRO environments you do not typically connect to the Oracle database using logical database names. Instead, the database logical name is embedded into the schema holder and the MFG/PRO code embeds the logical name at runtime.

## Client/Server Connection Parameters

If the WebSpeed broker is on a different machine than the MFG/PRO databases, you must use client/server connections to access the databases. This can improve Desktop system performance by spreading the MFG/PRO and Desktop resource requirements between two servers.

To enable the server processes to make client/server connections, the Desktop parameter files must contain the connection parameters described in Table 10.2.

**Table 10.2**  
Client/Server  
Connection  
Parameters

Parameter	Description
-db	The physical database parameter name. Follow this parameter with the physical name of the MFG/PRO database. You do not need to include the path to the database file.
-trig	The triggers parameter. Follow this parameter with the name of the directory containing the database triggers for the main MFG/PRO database. For MFG/PRO, this is the <code>triggers</code> subdirectory. The parameter value is: <code>-trig triggers</code> . This parameter is used only with main databases.

Parameter	Description
-ld	The logical database name parameter. Follow this parameter with the logical name of an MFG/PRO database. This parameter is used only with support databases.  <b>Oracle:</b> This parameter is not typically used in Oracle environments. Instead the database logical name is embedded into the schema holder and the MFG/PRO code embeds the logical name at runtime.
-S	The database service name parameter. Follow this parameter with an MFG/PRO database service name. You can use the Database Set Maintenance utility in MFG/UTIL to find the service name for a database.
-N	The network parameter. Follow this parameter with the network protocol used to connect to the MFG/PRO databases. For MFG/PRO, this is TCP/IP. The parameter value is: -N TCP.
-H	The host name parameter. Follow this parameter with the machine name or IP address of the MFG/PRO database server. This guide uses <i>DBServer</i> as an example in place of an actual machine name or IP address.

## Local Host Connection Parameter File Examples

The following examples are edited versions of the `Production.pf` file for both Progress and Oracle database environments.

### Progress Database Environments

**Example** Local Host Connections: This example is for local host connections and contains only the `-db`, `-ld`, and `-trig` parameters. Remember when configuring local host connections that you must include the full path and physical name of the database after the `-db` parameter:

```
-db /qad/mfgpro/eB/db/mfgprod.db -trig triggers
-db /qad/mfgpro/eB/db/hlpprod.db -ld qadhelp
-db /qad/mfgpro/eB/db/admprod.db -ld qadadm
-d mdy -yy 1920 -Bt 350 -c 30 -D 100 -mmax 8192 -nb 200-s 63
```

**Example** Client/Server Connections: This example is for client/server connections and contains the `-H`, `-S`, and `-N` parameters:

```
-db mfgprod -H uxsvr1 -S mfgprod-srv -N tcp -trig triggers
-db mfghelp -ld qadhelp -H uxsvr1 -S mfghelp-srv -N tcp
-db admin -ld qadadm -H uxsvr1 -S admin -srv -N tcp
-d mdy -yy 1920 -Bt 350 -c 30 -D 100 -mmax 8192 -nb 200 -s 63
```

▶ See “Connecting the DataServer” in the *Progress DataServer for Oracle Guide*.

## Oracle Database Environment

Generally, to execute most database functions, the `-c` should not exceed the Oracle database setting. If `OPEN_CURSORS` is 512 in Oracle, then the Progress `-c` should not exceed this value. Typically, the `-c` value should be approximately 10% less than the `OPEN_CURSORS` value.

**Example** Oracle Database Connections:

```
-db SchemaHolder -ld qad -RO -znotrim -trig triggers
-db ORACLE_SID -ld qaddb -dt ORACLE -U qad@NetService
-P qad -c 255
-cpinternal iso8859-1 -cpstream iso8859-1 -cpcoll basic
-d mdy -yy 1920 -Bt 350 -D 100 -mmax 8192 -nb 200 -s 63
```

*NetService* refers to the system Net Service name, such as *OracleSID*.

## Working with WebSpeed

This section includes additional information about WebSpeed, including:

- How to install and configure WebSpeed Messenger
- How to determine the required number of WebSpeed agents

### Install and Configure WebSpeed Messenger

If you do not have a full WebSpeed installation on the application server, you must download and install the WebSpeed Messenger on the application server. For this Desktop implementation, WebSpeed resides on the database server and the WebSpeed Messenger will reside on the application server.

#### Prerequisites

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

#### Overview

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

## Installation

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

- 1 Download the WebSpeed Messenger executable from the Progress Web site.  
<http://www.progress.com/products/downloads/index.ssp>  
 Make sure you download the executable for your specific server. Additionally, download any related documentation. The installation instructions are also provided on the download page. Make sure you print them out. They are required to install this Progress component.
- 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 the WebSpeed Messenger

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

**UNIX Servers:** If your application server is a UNIX server, go to the `\WebSpeedInstallDir\properties` directory. Find `msngrs.properties`, make a copy of this file, and rename the copy to `ubroker.properties`.

- 1 Using a text editor, open the `ubroker.properties` file in `\WebSpeedInstallDir\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 application server OS, add the following parameters; use `wspd_cgi.ksh` for UNIX or `cgiip.exe` for Windows.

```
msngrExecFile=@{Startup\DLC}\bin\wspd_cgi.ksh
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 Web server `\cgi-bin` 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.
- 6 Go to the *ProgressInstallDir*\properties subdirectory on the database server. Using a text editor, open the `ubroker.properties` file for editing. Find the [WebSpeed.Messengers.CGIIP] section and add these two parameters:

UNIX server

```
msngrExecFile=@{Startup/DLC}/bin/wspd_cgi.ksh
AllowMsngrCmds=1
```

Windows server

```
msngrExecFile=@{Startup\DLC}\bin\cgiip.exe
AllowMsngrCmds=1
```

## Verify the WebSpeed Messenger

### Ping

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

```
http://WebSvrHost/cgi-bin/wspd_cgi.ksh/WSservice=BrokerName/ping
```

- *WebSvrHost* is the Web server host where the remote messenger was installed.
- *BrokerName* is your WebSpeed broker name.

SpeedStart

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 any 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 in by resetting the Web Logic Implemented field in Program Information Maintenance.

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.

Table 10.3 shows the approximate amount of time during which WebSpeed agents are held for various combinations of Desktop activities and agent states.

Activity	Agent State	Time Agent Held
Building log-in screen	Stateless	Less than one second
Building menu or desktop	Stateless	Less than one second
Navigating down menu	Stateless	Less than one second
Starting browse	Stateless	Less than one second
Starting report	Stateless	Less than one second
Running report	Stateless	While report is generating data

Table 10.3  
Summary of Agent Usage

Activity	Agent State	Time Agent Held
Running a browse for which the number of records is less than the Download Size	Stateless	Less than one second
Running a browse for which the number of records exceeds the Download Size	Stateless	Less than one second
Running a browse for which the number of records exceeds the Download Size	State-aware	While browse is incomplete or until timeout is reached

### 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 the Desktop has not yet displayed all the data to the user.

### Agent Usage for Browsers and Lookups

♦ See *User Guide: QAD Desktop*.

For browses and lookups, 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).

- If the browse request generates fewer records than Max Browse Records, the records display in the browse and the agent is released immediately.

**Example** If Max Browse Records is 100 and a browse query returns 80 records, MFG/PRO displays the 80 records and releases the agent.

When download size is small, this might require only a few seconds per request. This means that a WebSpeed agent might be busy only a fraction of the time a user has a browse data window open in the Desktop.

- If the browse request generates more records than Max Browse Records, the agent is sometimes held the entire time the browse data window is open—that is, from the time the user requests the data to the time the browse times out or is manually closed. A browse in this condition is known as *state-aware*. This occurs when:
  - The user sorts browse data by a field that is not indexed in the database. Indexed fields are indicated by an asterisk (\*) in the Sort By field.
  - The WebSpeed agent returns a group of data that ends with a non-unique key.
  - The browse takes an unusually long time to begin returning data to the Web server.

When a browse is state-aware, it displays a time-out message in the title bar—for example, `Browse times out in 2 minutes`.

**Example** Max Browse Records is 100 and the query returns 200 records; the browse data window displays approximately the first 100 records. It may actually display slightly more than 100 records, depending on where an appropriate break in the active index is found. When the user scrolls down to the bottom of the data window, the browse requests the next 100 records for display.

When any of the conditions above are met, the agent becomes state-aware and is held until the browse is closed or times out. In this case, the browse data window displays the following message to indicate that the agent is being held:

`Times Out in X Minutes (Port #).`

## Mechanisms to Optimize Agent Usage

Several settings in the Desktop can be used to assist in optimizing WebSpeed agent usage. Define these settings in User Option Maintenance (36.20.10.1) for individual MFG/PRO user IDs.

▶ See “Define User Settings” on page 221.

**Max Web Connections.** Use this setting to indicate the maximum number of agents that can be simultaneously locked by the associated user. When this amount is exceeded, an error message displays and the user cannot run a report or a browse until they have released agents.

*Browse Timeout.* Use this setting to indicate the maximum amount of time an agent remains locked for the associated user before it is automatically released and is available to other users.

*Session Timeout.* Use this setting to indicate the maximum amount of time in minutes that a Desktop session can be inactive for the associated user before the system automatically logs them out.

## Security

This sections includes additional information about Desktop security issues.

### Securing Internet Communication

QAD recommends using Virtual Private Network (VPN) connections, which are encrypted, when accessing Desktop over the Internet. VPN provides a secure communication channel for Desktop sessions. This requires no additional setup for Desktop and requires that the user is authenticated and has authorization to log on to the network.

Using SSL (Secure Socket Layer) is not recommended. SSL over VPN is redundant and reduces Desktop performance. Additionally, the use of SSL security has the following limitations with Desktop:

- SSL security does not secure Desktop telnet communications.
- SSL does not secure communications between the Desktop Web server and the MFG/PRO database server. If these servers are different machines that communicate over a network, their communications may also be compromised.

### UNIX Telnet Environment Security Issues

To access a subset of MFG/PRO maintenance programs, the Desktop uses telnet to log on to the server where MFG/PRO is installed. Because the Desktop 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 special server-side security measures in mind.

A range of security options exists to solve the unencrypted log-in and password problem. This section outlines two sample security setups: 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.

**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.

**Tip**  
The default shell for a user is located in the /etc/passwd file.

## Examples of Security Measures

### Case 1: Maximum Security

One UNIX account with the following characteristics is used for all Desktop 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 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 maintenance 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: Less Security

Users have their own unique log in and password, but run the restricted shell by default:

- Write permission to 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 command line or script.

Use the following instructions to set up Case 2:

- 1 Create or modify accounts for Desktop 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 the Desktop maintenance program users.
- 4 Set the minimal environment variables, remembering to include `/usr/rbin`.
- 5 Copy the Desktop 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 maintenance 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 Desktop Systems

This section discusses issues you should be aware of when you:

- Configure multiple Desktop systems on one server.
- Configure multiple Desktop systems for more than one MFG/PRO release.

## Configuring Multiple Systems on the Same Server

**Note** If you want to provide Desktop clients with access to additional database sets—for example, the Training database set—you cannot share the same administration database between the sets.

If needed, you can create multiple Desktop systems on the same server. However, some aspects of each Desktop 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.

◆ See page 189.

**Note** Special Desktop log file directory requirements exist if you are configuring Desktop systems for multiple MFG/PRO releases.

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 system has a unique Web server document directory under the Web server document directory:

- c:\apache2\htdocs\test
- c:\apache2\htdocs\production
- c:\apache2\htdocs\train

The setup process creates additional subdirectories for each system. It creates \mfgpro for configuration files and MFG/PRO Progress programs (.p files) and \qad for the compiled Desktop-specific r-code. For example, for the test Desktop 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 example assume you used the MFG/PRO installation directory as the logs directory:

- `c:\mfgsvr\test_dt2.ini`
- `c:\mfgsvr\production_dt2.ini`
- `c:\mfgsvr\train_dt2.ini`

## Configuring Desktop for Multiple MFG/PRO Releases

If your environment requires setting up Desktop systems for multiple MFG/PRO versions running on the same server, the `.ini` and log files for Desktop systems running on each MFG/PRO release must be in their own MFG/PRO release-specific log files directory.

For example, if you have MFG/PRO eB2 Service Pack 3 and eB2.1 installed on a single server, one log file directory is required for each MFG/PRO release. Do not store `.ini` files for different MFG/PRO releases in the same Desktop log directory. You can store multiple `.ini` files for the same MFG/PRO 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
- Using the Tomcat Web Application Manager

### Install a New Tomcat Servlet Engine

**Note** To find your current Tomcat version, review the release note documents in the Tomcat root directory.

To install the new Tomcat version included on the Desktop 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 exiting 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.

## Manually Install Tomcat and Merge QAD Components

**Important** The Desktop 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.

## Clear the Tomcat Server Cache

In order to see updates in Desktop, you should clear the server cache:

- Before you install a new Desktop release
- After you rebuild the Desktop
- After you attach or detach a lookup from any field

**Note** In the following steps *SysName* refers to your Desktop system name.

To clear the cache, follow these steps:

- 1 Shut down Tomcat.
- 2 Go to the `$TOMCAT_HOME/webapps/SysName` directory.
- 3 Open the `WEB-INF/conf/config.xml` file with a text editor.
- 4 Search for the `staticpagedirectory` entry. This entry defines the directory where the static HTML files are cached; for example:  

```
<staticpagedirectory value="/usr/local/tomcat/webapps/
desktop/cache"/>
```
- 5 Close the text file.
- 6 Delete the contents of the directory specified in the `staticpagedirectory` entry.
- 7 If you are installing a new Desktop release, complete the installation steps.
- 8 Restart Tomcat.

## Tomcat Web Application Manager

The Tomcat Web Application Manager is a Web application used to manage Web applications on the Tomcat server. It lets you list, install, reload, deploy, and remove existing Web applications, without having to shut down and restart the entire container. This utility is a Web application itself; Tomcat must be running in order to use it.

This installation guide does not include configuration or user documentation for the Tomcat Web Application Manager. Refer to the Tomcat how-to document for Tomcat documentation:

<http://jakarta.apache.org/tomcat/tomcat-4.1-doc/manager-howto.html>

**Fig. 10.4**  
Tomcat Web  
Application  
Manager

Tomcat Web Application Manager					
Message:					
Applications					
Path	Display Name	Running	Sessions		
/	Welcome to Tomcat	true	0	Start	<a href="#">Stop</a> <a href="#">Reload</a> <a href="#">Remove</a>
/admin	Tomcat Administration Application	true	0	Start	<a href="#">Stop</a> <a href="#">Reload</a> <a href="#">Remove</a>
/du		true	0	Start	<a href="#">Stop</a> <a href="#">Reload</a> <a href="#">Remove</a>
/examples	Tomcat Examples	true	0	Start	<a href="#">Stop</a> <a href="#">Reload</a> <a href="#">Remove</a>
/jp		true	0	Start	<a href="#">Stop</a> <a href="#">Reload</a> <a href="#">Remove</a>
/manager	Tomcat Manager Application	true	0	Start	<a href="#">Stop</a> <a href="#">Reload</a> <a href="#">Remove</a>
/multi		true	2	Start	<a href="#">Stop</a> <a href="#">Reload</a> <a href="#">Remove</a>
/testdt91		true	0	Start	<a href="#">Stop</a> <a href="#">Reload</a> <a href="#">Remove</a>
/tomcat-docs	Tomcat Documentation	true	0	Start	<a href="#">Stop</a> <a href="#">Reload</a> <a href="#">Remove</a>
/webdav	Webdav Content Management	true	0	Start	<a href="#">Stop</a> <a href="#">Reload</a> <a href="#">Remove</a>
Install					
Path:		Config URL:		WAR URL:	Install
Server					
Server Version	JVM Version	JVM Vendor	OS Name	OS Version	OS Arch
Apache Tomcat/4.1.12	1.4.0-b92	Sun Microsystems Inc.	Linux	2.4.7-10enterprise	i386

## Use Diagnostic Tools

This section includes information about a number of diagnostic techniques and tools you can use with your Desktop environment. These tools can be especially helpful when first installing and tuning the application or upgrading to a new release of Desktop.

Details are included about:

- Setting logging parameters
- Enabling the Java Console
- Enabling performance timing

## Configure Logging

Logging in Desktop 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:

`$TOMCAT_HOME/webapps/SysName/WEB-INF/conf`

All log information is written by default to the desktop .log file located in:

`$TOMCAT_HOME/webapps/SysName/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 is beyond the scope of this guide; further information 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

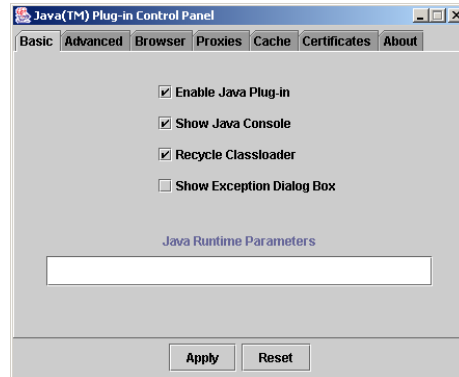
## Enable the Java Console

If you selected Enable Java Console Tracing when you configured your Desktop 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 Desktop client machine, click Start|Settings|Control Panel.
- 2 Double-click the icon for Java Plug-in.
- 3 When the Java Plug-in Control Panel displays, click Show Java Console in the Basic tab.

◆ See “Enable Java Console Tracing” on page 209.

**Fig. 10.5**  
Java Plug-in  
Control Panel



- 4 Click Apply to save your changes.

## Enable Performance Testing

To help you troubleshoot Desktop performance related to your network, you can activate a timer for Desktop 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 timing.

To activate the performance timer function:

- 1 Using a text editor, open `config.js` for editing. This file is in the `js` subdirectory of the Desktop installation directory.
- 2 Add the following new line at the bottom of the file and save the file:  

```
var htmlTiming = "true";
```

# Deployment Tools

This chapter discusses the deployment script and MFG/UTIL features not discussed in the step-by-step installation chapters. It includes additional details for advanced functions and features.

*Deployment Script Reference*    **196**

*Single-Tier Deployment Options*    **196**

*Multi-Tier Deployment Options*    **199**

*MFG/UTIL Reference*    **202**

## Deployment Script Reference

The Desktop component deployment scripts are written in Perl, which is a cross-platform scripting language. It uses an Unzip utility to expand compressed files and the Java jar functions to create and expand JAR files for configured Desktop component distribution. The source code used to build the Perl and Unzip executables as well as the binaries for each of the supported platforms is included in a set of platform-specific directories on the CD.

This section gives a general overview of each menu option on the deployment script. The deployment script main structure differs between single and multi-tier configurations and for Windows and UNIX servers.

## Single-Tier Deployment Options

### Install HTML and CGI Files Menu

This option displays the htdocs and cgi-bin installation submenu to install the htdocs and cgi-bin files on the Web server. In order to successfully complete this menu option, you must enter or validate the following Web server values:

- JAVA\_HOME
- The installation directory
- The scripts directory (cgi-bin)
- The Web server port number
- The ScriptAlias
- The HTML documents directory (htdocs)

### Install Tomcat Files Menu

This option displays the Tomcat installation submenu to install the Tomcat servlet engine and the Desktop application files. You must execute this menu option, even if you have already installed Tomcat on

your server. In this case, only the Desktop required application files are installed into the proper Tomcat directories. To run this option, you must enter or verify these values:

- Tomcat installation directory
- JAVA\_HOME
- Tomcat port number

### New Tomcat Installations

When you select this option, the system installs Tomcat in the specified directory if no other Tomcat installation is present. Once the Tomcat application files are installed, the script installs the Desktop-specific Web application files used by Tomcat.

### Replacing an Existing Tomcat Installation

If you are deploying components on a server that already has Tomcat installed, the script automatically detects if the existing Tomcat installation is compatible with the Desktop release you are deploying. If the installed Tomcat version is not compatible with Desktop, you are prompted whether to back up the incompatible Tomcat installation and install the version included on the Desktop installation media.

### Using an Existing, Compatible Tomcat Installation

If you are deploying components on a server where a compatible Tomcat version is already installed, the deployment script verifies that the installed Tomcat version is compatible with the Desktop release being configured. You are then prompted whether to install only the Desktop Web application and configuration files. These files are required. The Web application files are placed in the Tomcat `\webapps\SysName` subdirectory, where *SysName* refers to your Desktop system name. Several Tomcat-specific files are updated in other Tomcat directories.

◆ See “Configure and Start Tomcat” on page 100.

**Note** Back up your Tomcat installation directory if you need to maintain any customizations required for other Web applications.

## Install MFG/PRO – Progress Desktop Files Menu

Use the options on this menu to install Progress files on the database server or update any MFG/UTIL installations not located in the character client directory. In Windows environments, an additional menu option is used to set up your telnet environment and generate the telnet server `k_start.bat` script. To complete the options on this menu, you must enter or validate these values:

- Database server `JAVA_HOME`
- MFG/PRO character client installation directory
- Telnet server installation directory (Windows)

## Install MFG/PRO – Progress Desktop Files

Use this option to install the complete set of MFG/PRO Progress files required on the database server for Desktop. These files include updates to MFG/UTIL and updated MFG/PRO code required for Desktop fixes not included in the current MFG/PRO service packs. To use this option, you must enter the character client installation directory. Based on this value, the deployment script updates the MFG/UTIL and MFG/PRO-related code in the `\xrc` directory and places the Desktop code in a Desktop system-named subdirectory under the character client installation directory.

## Update Additional MFG/UTIL Installations

This option is used in environments where either the character client is installed in separate directories or you have multiple MFG/PRO installations on the same server. This option lets you update multiple MFG/UTIL installations with the Desktop configuration and build options.

When you select this option, you enter the directory where the other MFG/UTIL installation is located. The script validates whether the indicated directory contains an MFG/PRO installation and its version. That MFG/UTIL installation is then updated with the required changes. If the MFG/PRO version cannot be detected from the `version.mfg` file, you must specify what MFG/PRO version is being updated.

### Telnet Setup (Windows Only)

This menu option does not appear when running the scripts on a UNIX server.

You must use this menu option to set up your telnet environment and generate the Windows-specific telnet script, `k_start.bat`. This file is required for Windows systems. The script is used to automatically start a telnet server session and to determine which items a user can access on the host when launching an MFG/PRO program that executes in Desktop's integrated telnet interface. The generated file is placed in a `\telnet` subdirectory in the MFG/PRO installation directory.

### Uninstall MFG/PRO – Progress Desktop Files

Use this menu option to uninstall the Progress files installed by the deployment script for the Desktop system you select. Use this option when you want to back out the Desktop directories for a system. This option does not remove the MFG/UTIL updates.

Using this option removes the `\mfgsvr\SysName` directory. *SysName* is the Desktop configuration name.

## Multi-Tier Deployment Options

The main menu for multi-tier configurations has two significant menu options. The first option—the Deploy the Progress files menu—is used on the database server. The Post MFG/UTIL Desktop Build Jar Install Menu option is used on the application servers. In order to use these menu options, you need to know and confirm the database server `JAVA_HOME`.

### Install MFG/PRO – Progress Desktop Files Menu

Use the options on this menu on the database server to:

- Deploy Desktop-specific code and MFG/PRO and MFG/UTIL updates.
- Stage the HTML and cgi-bin files that MFG/UTIL will customize and prepare for the application servers.

- Update any MFG/UTIL installations not located in the character client directory.
- Generate the telnet server `k_start.bat` script in Windows environments.

To complete the items on this menu, you must enter or validate these values:

- MFG/PRO character client installation directory
- Telnet server installation directory (Windows)

### Install MFG/PRO – Progress Desktop Files

This menu option installs the Desktop code and makes Desktop-specific updates to MFG/PRO and MFG/UTIL. Additionally, it stages the HTML, cgi-bin, and Tomcat code that MFG/UTIL will customize and package for deployment on the application servers.

You must enter or confirm the following information to successfully complete this option:

- The MFG/PRO character client installation directory
- The Desktop code directory, typically in the client installation directory and named the same as your Desktop configuration
- A staging directory location with approximately 20 MB free space

### Update Additional MFG/UTIL Installations

◆ See page 198.

This option is the same as the corresponding option for single-tier deployments.

### Telnet Setup (Windows Only)

◆ See “Telnet Setup (Windows Only)” on page 199.

This option is the same as the corresponding option for single-tier Windows deployments.

## Uninstall MFG/PRO – Progress Desktop Files

This option is the same as the corresponding single-tier menu item.

▶ See “Uninstall MFG/PRO – Progress Desktop Files” on page 199.

## Post MFG/UTIL Desktop Build Jar Install

Run this menu option on the application server to deploy the MFG/UTIL-created JAR archive files. These are the files that contain the configured application server files for the Tomcat servlet engine and the cgi-bin and htdocs files for the Web server. Depending on the JAR file you are deploying, you will be prompted for different information.

To deploy the JAR archive file with the configured Tomcat files, you must enter or confirm the following information:

- Tomcat installation directory
- The JAVA\_HOME value
- The port number used for Tomcat

To deploy the JAR files with the configured HTML and CGI scripts, you must enter or confirm the following information:

- The application server name
- The Desktop system name
- The JAVA\_HOME value
- The Web server htdocs directory
- The Web server cgi-bin directory
- The Web server port number

Before you can use this option:

- The MFG/UTIL-created JAR files must be accessible from the application server.
- The Desktop configuration .ini file for the multi-tier system being configured must be accessible from the application server.
- A Java 1.4.x version must be installed on the application server.

## MFG/UTIL Reference

This section includes additional field descriptions for all the Desktop-specific screens in MFG/UTIL. Use this information as reference to verify the MFG/UTIL build information and to complete any additional configuration and build activities.

**Note** Some fields and screens may not apply or display in your environment depending on the database type and whether you are completing a single or multi-tier deployment.

## Configure Desktop Environment Parameters

This section discusses all of the fields that display when you choose the Configure Desktop Environment Parameters option on the Desktop menu in MFG/UTIL.

### Desktop System Information

The first set of fields prompt for information related to your Desktop system.

**Fig. 11.1**  
Desktop System  
Fields

Database Connection Parameter File with Path:  

MFG/PRO Language Code:  

Desktop Install Directory:  

MFG/PRO Version:  

Database Set Name:  

Absolute Path to Oracle Install:  

Oracle System Identifier Name:  

ORASDPATHNAME:  

Service Pack Directory:  

Propath:

*Database Connection Parameter File with Path.* This field displays any parameter files found in the MFG/PRO and character client installation directory.

If you defined any additional search directories, files found in those directories are also listed for selection.

▶ See “Add Parameter File Directories” on page 217.

**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 generates a list of language codes based on MFG/PRO language installations detected in your environment.

Other language settings are available in the Advanced Options. For a single-language implementation, you should not need to change any default values.

▶ See “Language Settings” on page 213.

**Desktop Install Directory.** MFG/UTIL uses the value you specified during the Perl script deployment activities that was saved to the configuration `.ini` file. The system validates that the directory exists and whether the Desktop code directories are located in it.

When an installation directory is not recorded in the Desktop `.ini` file, MFG/UTIL searches for the Desktop code subdirectories in the the known MFG/PRO installation directories and any additional search directories you defined. When MFG/UTIL finds a Desktop code directory, this field is populated with that value.

▶ See “Add Parameter File Directories” on page 217.

**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 Desktop build. In multi-tier configurations, you will be prompted for `JAVA_HOME` for the application server when installing Tomcat or deploying configured archive files. The application server and database server `JAVA_HOME` values will likely be different.

**MFG/PRO Version.** The MFG/PRO version listed in the `version.mfg` file in the MFG/PRO installation directory defaults here. eB displays for MFG/PRO eB; eB2 displays for MFG/PRO eB2 and MFG/PRO eB2.1.

You can change this value, but MFG/UTIL validates that the version specified is the same version found in the `version.mfg` file. If it is not, a message displays and you are prompted whether to continue.

**Note** When a service pack level is found, it is ignored and has no effect on this setting.

**Database Set Name.** This is the database set where the Desktop will be used. This value should correspond to the .pf file you choose for building Desktop. The default value is the first database set MFG/UTIL finds in the mfgutil.ini file.

**Absolute Path to Oracle Install.** This is the path to the Oracle installation directory, also known as the ORACLE\_HOME parameter.

**Oracle System Identifier Name.** Enter the Oracle System Identifier (ORACLE\_SID). This is the environment variable that the system uses for the Oracle database and is typically the same as the Oracle database name.

**ORASOPATHNAME.** This Oracle environment variable is used in UNIX environments only. This value can be used to point to the ORACLE\_HOME/lib/libcIntsh.xx client library, where XX refers to your Oracle version. Your specific Oracle version and system configuration define whether this parameter must be set and the specific setting. Refer to the Oracle documentation for details.

**Service Pack Directory.** Enter the service pack directory to include in the Desktop PROPATH. The directory you enter is validated by MFG/UTIL to ensure it contains the version.mfg file. Service pack directories should always contain this file. If version.mfg does not exist, a warning message is issued, but you are given the option to continue with the directory specified.

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

MFG/UTIL uses the service pack base install directory to compile the Desktop code. This directory is also added to the Desktop PROPATH.

- For the code compile in eB2 executed during the Desktop build, MFG/UTIL appends *BaseSPDir* and the *BaseSPDir/xrc* to the PROPATH after *DesktopInstallDir* and before *BaseMFGDir*, *BaseMFGDir/xrc*.
- For the code compile in eB executed during the Desktop build, MFG/UTIL appends *BaseSPDir* and *BaseSPDir/langcode/xrc* to the PROPATH after *DesktopInstallDir* and before *BaseMFGDir*, *BaseMFGDir/xrc*.

- For telnet, the PROPATH is:

```
DesktopInstallDir, ., BaseSPDir, BaseSPDir/
bbi, BaseMFGDir, BaseMFGDir/bbi
```

- For the Connection Manager script, the PROPATH is:

```
DesktopInstallDir/com/mfgpro, ., BaseSPDir, BaseSPDir/
bbi, BaseMFGDir, BaseMFGDir/bbi
```

- For WebSpeed the PROPATH is:

```
DesktopInstallDir/com/
mfgpro, DesktopInstallDir, ., BaseSPDir, BaseSPDir/
bbi, BaseMFGDir, BaseMFGDir/bbi
```

**PROPATH.** Use this field to enter any additional values for the PROPATH, such as an additional client code directory. Make sure that all the proper MFG/PRO code and any directories containing service pack code are included. The PROPATH for the build process requires the following directories, in this order:

- a** The Desktop installation directory.
- b** The MFG/PRO installation directory. This is the directory where the `version.mfg` file is found.
- c** The `/usr/src` directory in the MFG/PRO installation directory.

The default PROPATH value is generated dynamically using:

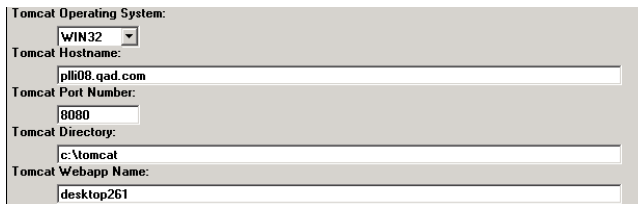
- The Desktop directory from the system configuration `.ini` file.
- The PROPATH for MFG/PRO from the `mfgutil.ini` file.
- For eB, the `src` directory is appended to the first language directory MFG/UTIL encounters.
- For eB2 and 2.1, the `src` directory is appended to the PROPATH.

All directories specified in the PROPATH must exist.

## Tomcat Configuration Fields

This set of fields prompts for details related to your Tomcat installation.

**Fig. 11.2**  
Tomcat  
Configuration  
Fields



Tomcat Operating System:	WIN32
Tomcat Hostname:	plli08.qad.com
Tomcat Port Number:	8080
Tomcat Directory:	c:\Tomcat
Tomcat Webapp Name:	desktop261

**Tomcat Operating System.** Specify the operating system for the application server where Tomcat will reside. This value defaults from information in the configuration .ini file, if available.

**Tomcat Host Name.** 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 101.

**Note** If you do not accept the default value of 8080, you will need to manually update the Tomcat server configuration file with the port you plan to use.

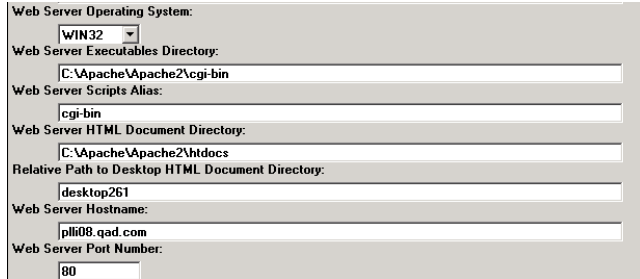
**Tomcat Stage Directory.** For multi-tier deployments, this is the staging directory where the Tomcat files were placed by the deployment script. These are the files MFG/UTIL will customize for your application server.

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

**Tomcat Webapp Name.** This field is automatically populated with the name of your Desktop system configuration.

## Web Server Information

This set of fields prompts for details related to the Web server you plan to use with Desktop.



Web Server Operating System:	WIN32
Web Server Executables Directory:	C:\Apache\Apache2\cgi-bin
Web Server Scripts Alias:	cgi-bin
Web Server HTML Document Directory:	C:\Apache\Apache2\htdocs
Relative Path to Desktop HTML Document Directory:	desktop261
Web Server Hostname:	plli08.qad.com
Web Server Port Number:	80

**Fig. 11.3**  
Web Server Fields

*Web Server Operating System.* Enter the Web server operating system.

*Web Server Executables Stage Directory.* For multi-tier deployments, this is the staging directory where the Web server executables files were placed by the deployment script. These files will be customized by MFG/UTIL for deployment to the Web server.

*Web Server Executables Directory.* For single-tier deployments, enter your Web server's executables directory, also known as the script alias.

The WebSpeed messenger executable is installed in this directory during the WebSpeed installation process.

*Web Server Scripts Alias.* Enter the Web server ScriptAlias. For single-tier deployment, this value comes from the configuration .ini file. You entered it during the component deployment phase of the Desktop implementation.

*Web Server HTML Document Directory.* For single-tier deployments, this is the full path to the directory where the Web server document files were placed by the deployment script. You entered this directory during the deployment phase of this Desktop configuration. For many Web servers, this directory is referred to as the htdocs directory.

*Relative Path to Desktop HTML Document Directory.* This is the relative path to the directory where the Web server document files for this Desktop configuration were placed. You indicated this directory

during the deployment activities. If you left the default value when you were prompted for this directory, the directory name is the same as your Desktop configuration name.

**Web Server HTML Document Stage Directory.** For multi-tier deployments, this is the staging directory where the Web server HTML document files were placed by the deployment script. These files will be updated by MFG/UTIL for deployment to the Web server.

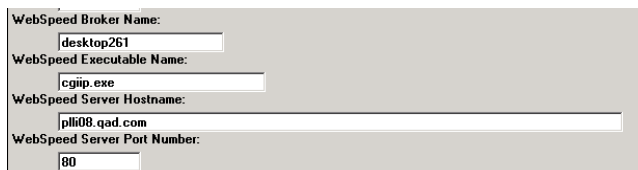
**Web Server Host Name.** Enter the fully qualified server name for the Desktop Web server. Fully qualified server names are required for all servers being used for any Desktop component. Even if the server name is populated by MFG/UTIL, you must update it with a fully qualified name.

**Web Server Port Number.** Enter your Web server's HTTP port number. This port was set up during the Web server installation process. MFG/UTIL always populates this field with port 80; you can change it as needed.

## WebSpeed Information

This set of fields prompts for details used with your WebSpeed configuration.

**Fig. 11.4**  
WebSpeed  
Configuration  
Fields



WebSpeed Broker Name:	desktop261
WebSpeed Executable Name:	cgiip.exe
WebSpeed Server Hostname:	plli08.qad.com
WebSpeed Server Port Number:	80

**WebSpeed Broker Name.** Specify the name of the WebSpeed broker you set up for the Desktop. The broker name is defined in the `ubroker.properties` file, located in the `properties` subdirectory of the Progress installation directory. Typically, the broker name is the same as the Desktop configuration name.

**WebSpeed Executable Name.** Enter the name of the WebSpeed messenger executable, located in your Web server's executables directory. On Windows, it is named `cgiip.exe`. On UNIX systems, use `wspd_cgi.ksh`.

**WebSpeed Server Host Name.** Enter the fully qualified server name for the Desktop Web server. Fully qualified server names are required for all servers being used for any Desktop component. Even if the server name is populated by MFG/UTIL, you must update it with a fully qualified name.

In a multi-tier environment where WebSpeed Messenger is being used, you must enter the server where the WebSpeed Messenger is installed.

**WebSpeed Server Port Number.** Enter a port number for the WebSpeed server. For most environments this value is already set to the same port used for the Web server.

## Java and Search Default

This set of fields prompts for details used with the Java Plug-in and Desktop search.



**Enable Java Console Tracing.** Enter Yes or No to indicate if the session trace messages should be sent to the Java Plug-in Console. To view the messages, you must also enable the console.

Enabling trace may adversely affect performance because display messages are continually sent to the Java console. If you enable trace for debugging purposes, you must rebuild the Desktop once you are ready to disable this option.

**Use QAD Desktop Supplied Java Plugin.** Indicate how you want to manage the Java Plug-in version for client sessions.

- Set this field to Yes when you want all clients to use the 1.3.1\_08 plug-in version supplied on the Desktop installation media. If this version is not available on the client, it is automatically downloaded and installed. Yes is the default setting, and most users should leave this set to Yes. If other versions of Java exist on the client, they are not replaced and can continue to be used by applications other than Desktop.

**Fig. 11.5**  
Java and Search  
Fields

▶ See “Enable the Java Console” on page 193.

- Set this field to No to have Desktop clients use the latest plug-in already installed on the client machine. If no plug-in exists on the client or the minimum plug-in version required by Desktop (1.3.1\_03) is not found, the QAD-supplied version is downloaded.

You should set this value to No only when you are sure that all clients using Desktop already have a supported version of the Java Plug-in installed, such as 1.3.1\_04. In this case, you can avoid requiring each client to download the 1.3.1\_08 version.

**Important** Currently the only supported plug-in versions are 1.3.1\_03, 1.3.1\_04, and 1.3.1\_08 through 1.3.1\_12. If the client has a 1.4 version of the plug-in installed and you specify No to this setting, some functions in Desktop may fail.

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

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

## Verify and Build the Desktop System

Before performing the Desktop build, you verify the system build parameters. To begin, choose Desktop|Build Desktop on the MFG/UTIL main menu. When the Build Desktop screen displays, choose the system to build. The same screens used to enter system parameters display, but with read-only values. Verify the values on all four screens and then choose Build to proceed with the Desktop build. At the completion of the Desktop build, a message displays the name and directory location of the build log file.

### Advanced Build Options

This section discusses the fields that display when you choose the Build Desktop option on the Desktop menu in MFG/UTIL and access the Desktop Advanced Build Options screen.

These options are used when configuring multi-language Desktop environments, to rebuild specific Desktop components without rerunning the entire build process, or to create build scripts without running the actual build process. As with other MFG/UTIL screens, this one may differ slightly depending on the Desktop configuration and database environment you are working with.

**Fig. 11.6**  
Advanced Build  
Options

Displays for  
Oracle database  
environments  
only

**Desktop Configuration Name.** This is the configuration that you are setting the build options for. You cannot select any other configuration; this is a display-only field.

**DB Connection -pf File.** This is the database connection parameter file that the build process will use to connect to the database to load system data and compile the Desktop code. This file was selected previously in the Desktop 2 Configuration screen. If the wrong .pf is displayed here or you need to change it, you must exit this screen and restart the process by choosing the MFG/UTIL Desktop|Configure Desktop Environment Parameters menu option. Then select the correct .pf in the Desktop 2 Configuration screen.

See “Database Connection Parameter File with Path” on page 202.

**Note** The database must be running for the compile and load to complete, or your database parameters must include -1 to connect to the database in single-user mode. In eB2 and eB2.1 environments, it must contain the appropriate -d mdy parameter.

*Generate new config.properties file.* This option should be selected when a new Desktop system is to be built. However, once a Desktop system has been built and you are rerunning a specific build function—such as generating new menus because you changed menus in MFG/PRO—you may want to 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 option to generate build scripts without running the build. These scripts can later be used to build the Desktop system. This option helps ease system administration activities in environments with multiple Desktop systems. By creating build scripts for all the Desktop systems without running the build, the system administrator can create a batch or cron job to run the build for one or multiple Desktop systems at a later time when more system resources are available, or during scheduled down times.

The generated build script is saved to the MFG/PRO installation directory. The file name is the Desktop system name with the standard system extension, such as `Production.bat` for Windows, or `Production.ksh` for UNIX.

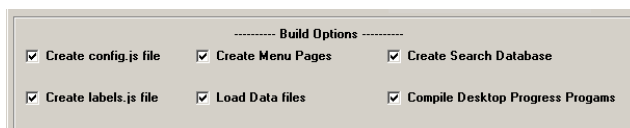
## Build Options

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 MFG/PRO Menus” on page 160.

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.

**Fig. 11.7**  
Build Options



*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.

**Create Menu Pages.** Enable this option to create the HTML menu pages for Desktop. This option updates the menus in the 2-letter language code subdirectory in the Web server document directory at `htdocs/SysName/menus`, where *SysName* refers to the Desktop system name.

**Create Search Database.** Enable this option to build the database accessed by the Search option in Desktop. 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; 36.20.3 in MFG/PRO eB).

**Create labels.js file.** Enable this option to create the JavaScript code for displaying MFG/PRO labels that are used on the Desktop 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 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 Desktop systems for the same databases, you can run each Desktop system build with this option enabled without any negative effect.

**Compile Desktop Progress Programs.** Enable this option to compile the Progress programs supplied with Desktop.

## Language Settings

Language settings can be important if you are installing Desktop into an MFG/PRO system that supports multiple languages. For a single-language implementation, you typically do not need to change any of these settings. Desktop is preconfigured with appropriate settings for each supported language and these display by default, based on the 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 136.

**Fig. 11.8**  
Advanced Build  
Options, Language  
Settings

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.

**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. Desktop is preconfigured with the correct mapping between supported MFG/PRO codes and ISO codes.

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 created in the en directory.

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

**–cpinternal.** This Progress database startup parameter identifies the code page that Progress uses in memory for graphical clients. This should be the same as the code page that the operating system uses. If a value is not specified, iso8859-1 is used by default. When you are building a multiple-language system, you must change this value to the code page associated with the database.

See “Language Directories for Menus” on page 174.

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

–*cpstream*. This Progress database startup parameter identifies the code page used for stream I/O. Stream I/O includes character-based terminals, data files, and code. When you are building a multiple-language system, you may need to change this value to the code page associated with the database.

–*cpcoll*. This Progress database startup parameter 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 are building a multiple-language system, you must change this value to the collation value associated with the database.

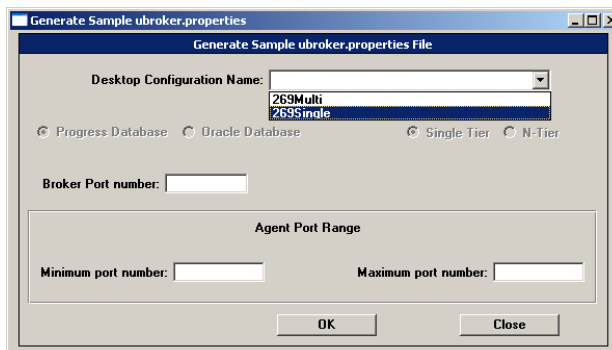
*Oracle NLS Setting*. This field sets the `NLS_LANG` parameter required in the `WebSpeed ubroker.properties` file and telnet startup scripts for the Oracle dataserver when Native Language Support (NLS) is required for non-English and double-byte languages.

## Generate WebSpeed Configuration File

Use the `Generate Sample ubroker.properties` option to generate a text file containing the parameters you need to add to the `WebSpeed ubroker.properties` configuration file for your Desktop system. Before you use this option, make sure the configuration information for your Desktop system is updated and complete in the single or multi-tier Desktop 2 Configuration window.

**Important** The generated parameter file includes the settings used in the most typical environment. Depending on your specific server and environment configuration, you may need to add other parameters to the `WebSpeed` configuration file.

**Fig. 11.9**  
Generate Sample  
ubroker.properties  
File



To create the file, follow these steps:

- 1 Choose Desktop|Generate Sample ubroker.properties File on the MFG/UTIL main menu.
- 2 Select the proper Desktop configuration name.
- 3 Enter the port number to use; the default is 20000.
- 4 Enter the minimum and maximum port numbers. These values set the `svrMinPort` and `svrMaxPort` values in `ubroker.properties` and default to 20001 and 20100.

The sample file is generated and saved in the MFG/PRO installation directory. A pop-up window indicates the exact directory where the file is saved. The file name includes the Desktop system name, followed by `-ubroker.properties`. For example, for the Production system, the file name is `Production-ubroker.properties`.

## Generate Telnet Start Scripts

Use this option to generate the telnet server startup scripts and an updated parameter file for the Desktop system. The telnet scrips are used to start the pool of Telnet connections for Connection Manager and to start a telnet session when running MFG/PRO maintenance programs in the embedded telnet interface. The generated parameter file includes any additional Desktop required directories, and replaces the existing system parameter file.

**Important** The generated scripts are set up for use in the most typical environment. Depending on your specific server and environment configuration, you may need to add other parameters to each script.

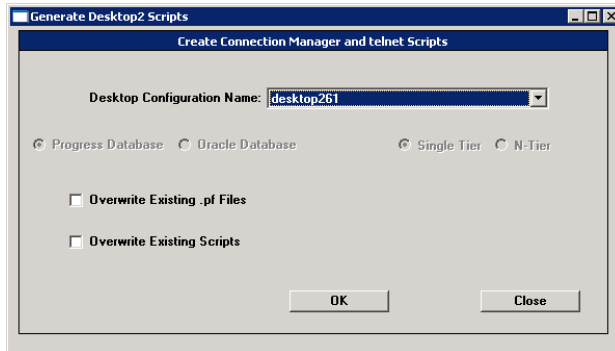


**Fig. 11.10**  
Generate  
Connection  
Manager and Telnet  
Server Scripts

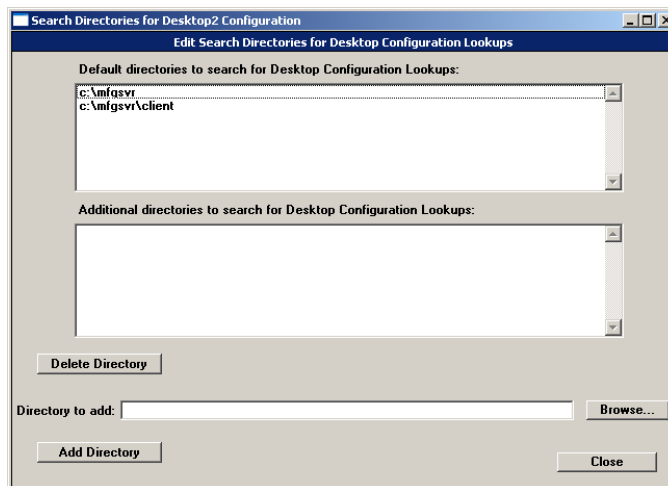
- 1 Access this option by choosing Desktop|Create Connection Manager and telnet scripts from the MFG/UTIL main menu.
- 2 Select the Desktop configuration in Desktop Configuration Name.
- 3 Use the check boxes to indicate whether to overwrite existing files. Otherwise, you are prompted whether to back up and replace the existing parameter file or telnet scripts.
- 4 To generate the scripts, choose OK.

## Add Parameter File Directories

This option lets you specify additional directories that MFG/UTIL should review to find .pf files that can be used to connect to an MFG/PRO database when compiling the Desktop-specific code. Use this option when you create custom .pf files for compiling Desktop code that are located in directories other than the standard MFG/PRO .pf files.

Access this function by selecting Desktop|Edit Search Directories for Desktop Configuration Lookups from the MFG/UTIL main menu. To add a directory, enter its full path name in Directory to add and choose Add Directory.

**Fig. 11.11**  
Search Directories  
for Desktop  
Configuration  
Lookups



You can add as many directories as you need. Any additional .pf files located in these directories display in the Database Connection Parameter File with Path field on the single or multi-tier Desktop 2 Configuration screen.

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

*Delete Directory.* Use this option to delete the selected directory from the list of directories to search. You cannot delete the default directories.

*Directory to add.* Enter the fully qualified directory name to add 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.

*Add Directory.* Choose this option to add the directory entered in Directory to add.

# Configuring Optional User Settings

This chapter discusses optional Desktop settings you can configure after the initial implementation. These settings let you tailor the Desktop for specific users and groups.

*Configuration Overview*    **220**

*Define User Settings*    **221**

*Define Desktop URLs*    **224**

*Monitor User Sessions*    **232**

*Other Desktop Configuration Options*    **233**

## Configuration Overview

This chapter includes information about the programs used to configure Desktop that are located on the Desktop Menu (36.20.10). For example, using functions on the Desktop Menu, you can define uniform resource locator (URL) links to intranet or Internet resources.

**Note** The Desktop uses some features originally introduced with QAD's Network User Interface (NetUI). The Desktop interface replaces the NetUI interface, but still uses some of the underlying NetUI functions. In MFG/PRO eB, this menu is still named NetUI Menu. In MFG/PRO eB2, all references to NetUI have been changed to Desktop.

▶ See *User Guide Volume 9: Manager Functions* for details.

Some of the manager functions that are used to configure character and Windows interfaces can also be used to affect the appearance and behavior of the Desktop. These programs are also discussed briefly in this chapter.

Table 12.1 lists the functions available on the Desktop Menu.

**Table 12.1**  
Desktop Menu  
(36.20.10)

Number	Program	Menu Label	Notes
36.20.10.1	mgusrmt.p	User Option Maintenance	Lets you tailor the Desktop for specific users and groups
36.20.10.2	mgbr062.p	User Option Browse	Lets you browse user option settings
36.20.10.3	mgusrtmt.p	User Option Telnet Maintenance	Discussed in Chapter 3.
36.20.10.4	mgusrrp.p	User Option Report	Lets you generate a report of user option settings
36.20.10.8	mgurlmt.p	Menu URL Maintenance	Lets you define the URLs that display in the My Bookmarks menu under My Desktop
36.20.10.9	mgbr220.p	Menu URL Browse	Lets you browse the URLs in My Bookmarks
36.20.10.11	mgurlmt.p	Browse URL Maintenance	Lets you define the URLs that display in Desktop browses
36.20.10.12	mgbr221.p	Browse URL Browse	Lets you browse the URLs in the Desktop browses
36.20.10.15	mgsessmt.p	Session Master Maintenance	Lets you monitor current Desktop sessions

## Define User Settings

Use User Option Maintenance (36.20.10.1) to define settings that control Desktop features and determine which interface options users can configure in browses, inquiries, and reports.

You can use this program to define unique settings for individual users or enter an asterisk (\*) in the User ID field to create a generic record of settings. This generic record is automatically associated with every user in the system that does not have a user-specific record.

**Note** This program was originally designed for the NetUI, which has been replaced by Desktop. Some of the settings that can be defined by a system administrator in User Option Maintenance do not apply to Desktop or do not function the way they did in NetUI. These include the following:

- Menu restriction has no effect. Standard MFG/PRO menu security does apply, but all menus are visible to users.
- Menu substitution has no effect. During the build of Desktop menus, browses (instead of inquiries) and standard programs (instead of custom ones) are always placed on the menus. The search function in Desktop finds both the standard and substituted programs and users can choose which one to execute.
- Session timeout is used only for maintenance programs that display in Desktop's integrated telnet character display. For HTML screens, timeout is determined by settings in Connection Manager; the setting in this program has no effect.

♦ See Chapter 7, "Configuring Connection Manager," on page 111.

The system is installed with a default record that applies to all users. You can change it to accommodate your own requirements.

You should create separate records for individual users only when appropriate settings vary from user to user or User Restriction is Yes in Desktop Security Control (36.3.21.24). When User Restriction is Yes, you must use this program to create a record of settings for each user in the system; you cannot create a generic record and apply it to multiple users.

♦ See *User Guide: QAD Desktop*.

Individual user IDs must be defined in User Maintenance (36.3.1, 36.3.18 in eB2) before you can reference them in User Option Maintenance.

**Fig. 12.1**  
User Option  
Maintenance  
(36.20.10.1)

Use the following field descriptions to configure User Option Maintenance settings for your environment.

► For more details on these fields, see *User Guide: QAD Desktop*.

***Restrict Menu.*** This field has no effect on the display of menus in Desktop. You can restrict user access to executing programs as needed using Menu Security Maintenance (36.3.1) or Desktop security features.

***Menu Substitution.*** This field has no effect on Desktop menus.

***Program Execution.*** Enter No to indicate that the associated user cannot execute programs that are not included on the Desktop menu. Enter Yes to indicate that they can. To execute a program not on the menu, an exclamation point (!) must be added at the end of the program name, as in the following example:

custom.p!

***Server Print.*** Enter No to indicate that the associated user cannot print browses to a server-based (remote) printer. Enter Yes to indicate that they can. If this is No, the remote print option is disabled for the user in browses.

***Save Brw Settings.*** Enter Yes to allow the associated user to save custom browse settings or configurations—for example, column order or column hiding settings.

When you save custom browse features, the associated browse displays the same way the next time you execute it.

***Expiration Date.*** Enter a date in this field to indicate that the associated user cannot access the system after this date. Leave this field blank to indicate that an expiration date does not apply to this user.

**Max Web Connections.** Specify the maximum number of WebSpeed agent connections that the associated user can use simultaneously. 0 (zero) indicates that no limit applies to this user.

Max Web Connections applies to browses. It also applies to look-ups accessed from another browse, inquiry, report, or maintenance program. It does not apply to HTML or telnet maintenance and transaction programs.

Web connections are calculated in the following way:

- One connection is used when the Desktop is initially launched. This connection is also used for the first browse or look-up program accessed.
- Each additional browse or look-up accessed while the first program is active counts as an additional connection.

When a program is closed, the associated connection is released.

In an environment with a limited number of available WebSpeed agents, setting Max Web Connections to a high value may result in decreased performance for other users. However, if enough WebSpeed processes have been configured system-wide, you should set this value to the maximum number of applicable Desktop programs the associated user is expected to execute simultaneously.

You can use the following guidelines to specify the initial value in this field. As the access patterns of different Desktop users become clear, you can modify this value to fit your environment.

- For users who typically run short reports or inquiries, assign a lower Max Web Connections value than for users who must simultaneously generate multiple reports or inquiries containing large amounts of data.
- Since it is difficult to predict in advance whether browses will require WebSpeed agents for short or long time periods, you should plan to have one WebSpeed agent available per open browse for high-priority users.

♦ See “Common System Administration Tasks” on page 156.

♦ See “Agent Usage for Reports and Inquiries” on page 182.

♦ See “Agent Usage for Browses and Lookups” on page 182.

**Max Browse Records.** Enter the maximum number of records to display in a browse for this user before notifying the server to retrieve more records. The default value is 0 (zero), which corresponds to a 10,000-record maximum.

*Session Timeout.* Specify an amount of time in minutes after which the system automatically logs the associated user out of an inactive telnet session. This setting has no effect on the timeout of programs that display with HTML screens. Session timeout is determined by Connection Manager.

*Browse Timeout.* Enter the maximum amount of time in minutes during which the associated user can view a state-aware browse without requesting the server to retrieve additional records. When this time period is exceeded, the browse displays a time-out message when the user scrolls to the bottom of the window and does not retrieve any more records. The user must restart the browse if they need to view additional records.

When a user accesses a browse containing more records than the specified download size, the system retrieves the first set of records and then waits for the user to scroll down to the bottom of the browse window before retrieving the next set of records. If the browse is state-aware during this time, other users cannot use the WebSpeed agent that is waiting to execute this user's request. Setting a Browse Timeout value prevents WebSpeed agents from being inaccessible to other users for extended time periods.

This field cannot be left blank.

**Note** Do not confuse this parameter with the Timeout Minutes parameter in the standard Security Control (36.3.24) program. Timeout Minutes specifies when to log users with inactive sessions out of MFG/PRO.

## Define Desktop URLs

Uniform resource locators (URLs) are text strings that indicate the location of an intranet or Internet resource. In Desktop, you can use URLs to access the resources associated with them.

- Links defined in Menu URL Maintenance (36.20.10.8) are accessible from the My Bookmarks menu under My Desktop.
- Links defined in Browse URL Maintenance (36.20.10.11) are accessible in the Desktop browse windows.

## Define Menu-Level URL Links

Use Menu URL Maintenance to establish URL links that users can access from the My Bookmarks menu under My Desktop. You can set up URLs that are available to all users or selected individuals.

**Fig. 12.2**  
Menu URL  
Maintenance  
(36.20.10.8)

Use the following field descriptions to configure Menu URL Maintenance settings for your environment:

**User ID.** Enter a user ID to associate with the specified URL link. To associate all users in the system with that link, enter an asterisk (\*) in this field. The link displays only under the associated user's My Bookmarks menu.

**URL to Execute.** Enter a URL to include under the My Bookmarks menu for this user (maximum 132 characters).

**Description.** Enter a descriptive label to associate with the specified URL. This label displays as a menu item on the My Bookmarks menu for this user.

**GIF File Name.** Leave this field blank. Image files do not apply to Desktop URLs.

## Define URL Links for Browsers

Use Browse URL Maintenance (36.20.10.11) to create URL links that users can activate from Desktop browse data windows.

♦ See *User Guide: QAD Desktop*.

When a browse cell contains a URL link, double-clicking it launches a new browser window and displays the intranet or Internet resource associated with the URL. You can use these URLs in two ways:

- Create links to external Web sites that users can activate from Desktop browse data windows, such as a supplier Web site, associated with a supplier ID.
- Create links—also known as *hotlinks*—to other MFG/PRO programs running under the Desktop and pass specific data values to the programs. This lets you use browses as a means of navigating directly to maintenance programs.

**Note** You can access hotlinks only from drill-down browses, not lookups. Drill-down browses are typically available directly from the MFG/PRO menus, but can also be associated with program fields in Drill Down/Look-Up Maintenance (36.20.1). When associated with a field, you use the Alt+B key combination to display the drill-down in the Desktop.

Define URLs to External Web Sites

Use Browse URL Maintenance to create links to external URLs with information that is related to items in the browse, as in the following example.

**Example** You want to establish a URL link in the Purchase Order Browse from supplier ID GS10100 to the corresponding supplier’s company Web site, located at <http://www.generalsupplies.com>. To do this, enter the following values in Browse URL Maintenance.

Table 12.2  
Sample Field  
Entries

Field Name	Value
Browse	pobr006.p
User ID	*
Field Name	so_vend
Value	gs10100
URL	<a href="http://www.generalsupplies.com">http://www.generalsupplies.com</a>
Description	General Supplies Web Site
Primary	Yes

URLs can contain special strings that are automatically replaced by field values in the browse. Selecting a link containing this type of string automatically replaces that string with the corresponding field value in the row.

Follow these steps to define this type of special string in a URL:

- 1 Enter **#b#** to indicate the beginning of the string.
- 2 After the **#b#**, enter an MFG/PRO field name associated with the specified browse.
- 3 Enter **#e#** to indicate the end of the string.

**Example** The Web site for one of your primary suppliers contains a catalog of items. Entering an item's identifier at this Web site accesses the catalog entry for that item, containing information such as item cost, quantity available, and ship weight. To create links from the MFG/PRO supplier item numbers to their corresponding catalog entries at the supplier's Web site, create the following URL:

*[http://www.generalsupplies.com/catalog/#b#vp\\_vend\\_part#e#](http://www.generalsupplies.com/catalog/#b#vp_vend_part#e#)*

Then, associate it with the Supplier Item column in the Supplier Item Browse. After you establish this link, selecting a supplier item number in the Supplier Item Browse automatically inserts the selected field value. For example, selecting supplier item 10-1005 creates this URL:

*<http://www.generalsupplies.com/10-1005>*

The system then launches a Web browser to display the relevant catalog information for that item located at that URL address.

### Define Hotlinks to Other MFG/PRO Programs

Use Browse URL Maintenance to create hotlinks to other MFG/PRO programs, as in the following example.

You can set up links in an item browse to directly access Item Master Maintenance (1.4.1), passing the current item number to the maintenance program, and executing the Go command any number of times. When a user clicks the link, Item Master Maintenance displays in a detached window.

Multiple columns of data in a browse can contain links so that you can access maintenance programs for any of the data related to a record. However, data for only one field can be passed to each program.

To support this kind of URL link, you use the `run_html` setting to indicate that you want to build a Desktop URL. The string must include the beginning and ending indicators required for other strings in URLs:

- 1 Enter `#b#` to indicate the beginning of the string.
- 2 Enter `#e#` to indicate the end of the string.

You then specify values that determine:

- The name of the MFG/PRO program to be executed when a user clicks the link
- The field in the designated MFG/PRO program to which you want to supply a value
- The value to be passed to the specified field
- The number of times the Go command should be executed in order to reach the field

To make it easy to build the URL, you can leave the URL and URL Script fields blank and press Go to display a pop-up that prompts you for the values required to build the URL. In this case, the system builds the URL including the `run_html` setting using the values you supply.

**Table 12.3**  
Sample Field  
Entries for Hotlink

Field Name	Value
Browse	sobr009.p
User ID	*
Field Name	sod_part
Value	*
URL	Leave Blank
Description	Hotlink to Item Master Maint.
Program Name	ppptmt
Field	pt_part
Value	sod_part
Index	2

The URL that the system builds based on these input values looks like the following example:

```
#b#run_html#e#?id=ppptmt.p&f=pt_part&v=#b#sod_part#e#&x=2
```

When the user clicks item 01053 in the sobr009.p browse, Item Master Maintenance displays with 01053 entered in the Item Number field and the cursor displayed in the Name field below it.

## Using Browse URL Maintenance

Figure 12.3 illustrates Browse URL Maintenance.

**Fig. 12.3**  
Browse URL  
Maintenance  
(36.20.10.11)

Use the following field descriptions to configure Browse URL Maintenance settings for your environment:

**Browse.** Enter the name of the Web-enabled browse program to contain the specified URL link.

Entering an asterisk (\*) in this field allows the specified URL to be associated with any Web-enabled browse in the system.

**Example** To associate a specific URL with the sales order number column in all Desktop browses, enter an asterisk in this field and specify the sales order column's corresponding field name (so\_nbr) in Field.

Entering a value in this field automatically displays a list of valid values for Field in the Defined URL Parameters frame.

**User ID.** Enter a user ID to associate with the specified URL link. To associate all users in the system with that link, enter an asterisk (\*) in this field.

**Field Name.** Designate the browse column in which you want to establish a URL link by entering the database field name associated with that column. Entering a value in the Browse field automatically displays valid values for this field in the Defined URL Parameters frame.

This field cannot be left blank.

**Value.** Specifying a value in this field associates the designated URL with every browse cell that contains that value and belongs to the browse column indicated in Field. Enter an asterisk (\*) to associate the URL with every cell in the column.

**Tip**

You cannot specify both a URL and a URL script.

**URL.** Specify a URL referring to an Internet or intranet location (maximum 132 characters). Leave blank to specify a URL script or to create a hotlink.

**Description.** Optionally enter a description to display when this URL is selected.

**Primary.** Enter Yes to indicate that the specified URL is the primary URL for a cell.

In browse data windows, right-clicking a cell or column containing multiple URLs displays a list of all the URLs associated with that cell or column. The primary link displays at the top of the list and is the default link for the cell.

This value applies to defined (non-scripted) URLs only.

**Tip**

You cannot specify both a URL and a URL script.

**URL Script.** To associate the specified user, browse, column, field value, or combination of these with a custom URL script, enter the full path to the directory containing the custom script. Leave this and the URL field blank to create a hotlink.

**Note** Scripts should be based on the supplied template located in the source code directory: `/src/urltempl.p`.

**Determine.** Enter Yes to have the system run the specified custom URL script upon selection of the associated cell or column to determine whether that cell or column has an associated URL.

When this field is No, the script is not run and the designated column or cell is defaulted to having a URL.

### HTTP Parameters Pop-Up

This pop-up displays when both URL and URL Script are blank and lets you specify a set of values used in a hotlink.

**Fig. 12.4**  
Browse URL  
Maintenance,  
HTTP Parameters

**Program Name.** Enter the name of the MFG/PRO program to launch when a user clicks this link. You can include or omit the .p extension. For example, specify ppptmt or ppptmt .p.

**Field.** Enter the field in the program that you want to pass a value to from the browse; for example, enter pt\_part to pass a value to the Item Number field in Item Master Maintenance. Only one field can be specified.

**Value.** Enter the field in the browse that contains the value that you want to pass to the named field. Use the beginning and ending delimiters for this value. For example, to pass the value of the sod\_part field in the Sales Order Browse (7.1.2) to the pt\_part field in Item Master Maintenance, enter #b#sod\_part#e#.

**Index.** Enter the number of times you want the system to execute a Go command when it invokes the program specified for Program Name. To access the first frame in a maintenance program, set this value to 1. If you set this to more than 1, all validations are executed before processing the Go command, so if appropriate values do not exist for

all required fields, an error will be generated when the user clicks the link.

# Monitor User Sessions

Use Session Master Maintenance (36.20.10.15) to view information about users who are currently logged in to MFG/PRO through the Desktop. This information displays in the form of session records, each identified by a unique session ID that is generated by the system. Session records are automatically created when a user successfully logs in to MFG/PRO from the Desktop and are deleted when they log out.

**Fig. 12.5**  
Session Master  
Maintenance  
(36.20.10.15)

The screenshot shows the 'Session Master Maintenance' window. At the top, the title bar says 'Session Master Maintenance'. Below the title bar, there are three input fields: 'Session ID:' with the value '1027446287795', 'User ID:' with the value 'hme', and 'User Name:'. To the right of these fields are two circular buttons with left and right arrows. Below this section is a 'Session Master' section containing two columns of information. The left column lists 'Client IP Address: 167.3.11.103', 'Session Timeout: 60 Min', 'Active Web: 0', and 'Number of Records: 100'. The right column lists 'Security Profile:', 'Idle Time: 0.12 Min', 'Active Telnet: 1', and 'Menu Substitution: [checked]'. Below the 'Session Master' section is a 'Session Context Detail' section containing two columns of information. The left column lists 'Context ID: 1' and 'GL Entity: 1000'. The right column lists 'Logical Database:', 'Current Entity: 1000', and 'Base Currency: USD'.

Session Master	
Client IP Address: 167.3.11.103	Security Profile:
Session Timeout: 60 Min	Idle Time: 0.12 Min
Active Web: 0	Active Telnet: 1
Number of Records: 100	Menu Substitution: <input checked="" type="checkbox"/>

Session Context Detail	
Context ID: 1	Logical Database:
GL Entity: 1000	Current Entity: 1000
	Base Currency: USD

The information displayed in each session record includes user settings established in User Option Maintenance, User Telnet Option Maintenance, and Desktop security programs. Additional information about the session context also displays, including the logical database name, the current entity, GL entity, and base currency.

See Chapter 7, “Configuring Connection Manager,” on page 111.

**Note** You can also monitor the status of sessions for HTML maintenance programs, reports, and inquiries using Connection Manager.

## Other Desktop Configuration Options

Most of the other functions that can be used to tailor the Windows and character user interfaces can also be used with the Desktop. For example, when you use Drill Down/Lookup Maintenance (36.20.1) to add a look-up or drill-down browse, the new browse displays in all three interfaces.

### Tip

In MFG/PRO eB, use Window Help Maintenance (36.4.21) and Drill Down Maintenance (36.20.1).

However, in some cases, you must set up features in a slightly different way in the Desktop. This section discusses some of the functions that vary when used in the Desktop:

- Menu System Maintenance (36.4.4)
- User Function Maintenance (36.4.11)
- User Tool Maintenance (36.20.4; 36.20.2 in MFG/PRO eB)

## Add New Main Menu Items

Use Menu System Maintenance (36.4.4) to add new top-level menu items that display in the Desktop menu system. The menu groups represented by top-level menu items are referenced through the letter A. For example A.1 is Distribution, A.2 is Manufacturing and so on.

Menus added in this way are available to all users.

If you require only one menu, use My Programs to define a set of frequently used programs instead.

To add a new top-level menu, specify A for the menu. Then specify a selection with a number greater than the last top-level menu item.

**Important** Whenever you change the menu system, you have to rebuild the menus and search database in order to see your changes.

**Example** You have five accounting clerks who always use the same set of programs. Set up menu Accounting Tasks by entering A for menu, 8 for selection, Accounting Tasks for selection label, and A.8 for the file to execute. Then press Go to save the record. Add functions under this menu by specifying A.8 for menu and the program description and name you want to execute.

♦ See “Configuring Multiple Systems on the Same Server” on page 188.

## Assign Programs to the User Menu

Use User Function Maintenance (36.4.11) to define a list of functions:

- To be executed with function keys in the character interface
- To appear on the User Menu in the character and Windows interfaces so they can be accessed directly from other programs
- To display under the My Programs link in Desktop

**Note** In the Windows or Desktop interface, you cannot use function keys to execute programs automatically. Instead, the Function Key field determines the order in which programs appear on the User Menu pull-down menu or in the My Desktop menu.

You can define a general User Menu as well as one for each user. To set up a general menu for all users, leave User ID blank when adding programs. To set up a user-specific menu, indicate the ID for which you are setting up each user menu item.

## Assign Links from One Program to Another

Use User Tool Maintenance (36.20.4; 36.20.2 in MFG/PRO eB) to create links for running programs from another program. Assign links by user and program. These links display at the bottom of the calling program screen in Desktop. They display in the order in which they are defined.

You can assign a text label to the link; image files do not apply.

# Glossary

**Accelerator Keys.** Keystroke equivalents to mouse actions. For example, you can delete a record in Desktop 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 Desktop 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 Desktop, 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)*.

**Constant.** In Desktop security, a generic variable identifying a category of data affected by a rule. Constant values are used with rules to determine which database records users can access in Web-enabled browses, reports, and inquiries. See also *Rule*.

**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 Desktop, 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 Desktop 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 NT/2000 Telnet Server (GSWTS).** The third-party telnet server software supplied with the Desktop 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 *Security Group*.

**GUI.** See *Graphical User Interface (GUI)*.

**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 Microsystems.

**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)*.

**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 Desktop, 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. Some features of NetUI are used by Desktop, such as security.

**Node.** In the Process Editor, nodes represent steps in a process and can contain URL links to other resources.

**Operator.** A symbol or mnemonic used to perform calculations or data comparisons—for example, =, <, AND, OR, and XOR.

**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.

**Profile.** In Desktop security, a set of MFG/PRO programs grouped together to simplify security setup. Profiles are assigned to security groups. For example, a group consisting of salespeople might be assigned a profile consisting of only sales programs. See also *Security Group*.

**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.

**Rule.** In Desktop security, rules define conditions under which specific database records can be accessed by browses, reports, and inquiries programs. Types of rules include field rules, logical rules, and dual rules. Rules are associated with individual programs as part of Desktop security setup.

**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 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.

**Security Group.** A category of users who typically access the same programs in Desktop. Using Desktop security features, you can apply standard security settings to groups of users, then define exceptions for particular individuals within these groups as needed. See also *Profile*.

**Security Profile.** See *Profile*.

**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 Desktop.

**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 Desktop. 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

## Symbols

.ini file. *See* configuration .ini file

## Numerics

36.3.21.24 221  
36.4.4 233  
36.4.11 234  
36.4.17.1 144  
36.20.2 234  
36.20.10 220  
36.20.10.1 221  
36.20.10.3 103  
36.20.10.8 225  
36.20.10.11 225  
36.20.10.15 232

## A

accessibility options 23  
ActiveX controls 22  
admin.html 112  
Administration page 112  
agent sizing for WebSpeed 181  
alias 32

## B

broker, WebSpeed 208  
Browse URL Maintenance 225  
browses  
    saving settings 222  
    timing out 224

## C

cache 157  
    client 158  
    Tomcat server 190  
CD  
    installing 41  
cgi-bin

    alias 33  
cgi-bin directory 32  
cgiip.exe 208  
client installation 129  
code pages 134  
    advanced build options 214  
    compatible 137  
    mapping Progress and Java 148  
    shared 137  
com\qad\nav directory 173  
compiling 213  
config.js file 212  
Connection Manager 111–122  
    configuration settings 117  
    errors 122  
    maximum sessions 114  
    monitoring sessions 120  
    monitoring users 121  
constants 144  
    code pages 148  
    language codes 142  
cookies  
    browser setting 22  
-cpoll 215  
-cpinternal 214  
-cpstream 215

## D

database server  
    dedicated deployment 172  
date formats 146  
date separator 146  
Desktop  
    Administration 112  
    client installation 129  
desktop installation directory  
    building system 207  
Desktop Menu 220

Desktop Security Control 221  
Download Center 113

## E

errata sheet 3  
errors  
    Connection Manager 122  
expiration date, user option 222

## F

fonts, non-English languages 140

## G

Georgia SoftWorks (GSNTS) 14  
    installing 35  
    registering 37

## H

hotlinks 226

## I

installation  
    CD media 41  
    Desktop media 12  
    items installed 173  
installation tools 17  
    .ini files 18  
    MFG/UTIL 17  
    reference 202  
    Perl scripts 17, 196  
internationalization  
    code pages 148  
    font.properties 140  
ISO language and country codes 143

## J

JAR files  
    create with MFG/UTIL 88  
    deploying 90  
Java code  
    Desktop 14  
Java console  
    enabling 193  
Java Development Kit (JDK)  
    overview 11  
    Tomcat release 174  
    verifying 35  
Java Plug-in  
    versions used 130  
Java Plug-in Control Panel  
    displaying console 193

Java Runtime Environment (JRE) 21  
Java Virtual Machine (JVM) 13  
java\_home variable 35  
JavaUI printer 159

## K

Kill.exe 36

## L

Label Master Maintenance 144  
labels.js file 213  
language codes  
    accessing 145  
    date formats 146  
    date separator 146  
    directory 175  
    elements 146  
    for menus 214  
    non-English 175  
    numeric formats 146  
    overview 142  
language, non-English  
    constants 146  
    font.properties 140  
    US data required for 144  
locale, determining 142  
log files  
    Connection Manager 119  
    WebSpeed 100  
Log4j  
    logging levels 192  
log-in scripts  
    telnet 103

## M

management utilities  
    Desktop Administration page 112  
    Tomcat Web Application Manager 191  
Max Browse Records field 223  
Max Web Connections field 223  
Menu Security Maintenance 222  
menu substitution 160, 222  
Menu Substitution Maintenance 160  
Menu System Maintenance 160, 233  
Menu URL Maintenance 225  
messenger executable 208  
MFG/PRO character code directory  
    items installed in 173  
MFG/PRO code  
    deploying multi-tier 75  
    deploying single-tier 47  
MFG/PRO databases

- client/server connection parameters 176
- local host connection parameters 175
- MFG/UTIL 202–218
  - build Desktop
    - multi-tier 83
    - single-tier 55
  - building languages 136
  - configure Desktop parameters
    - multi-tier 79
    - single-tier 51
  - create Desktop JAR files 88
  - generating scripts 61
  - multi-tier deployment 78
  - overview 17
  - printing Desktop information 65
  - single-tier deployment 51
- MIME types 33
- mount commands 42
- multiple languages 133–151
  - advanced build options 213
  - code pages 134
  - compatibility 31
  - installing 136
  - process maps 141
- Multi-Purpose Internet Mail Extension (MIME) 33
- multi-tier deployment 69–95
  - build Desktop 83
  - create JAR files 88
  - deploy JAR files 90
  - launch scripts 73
  - MFG/PRO code 75
  - MFG/UTIL 78
  - planning 166

## N

- NameServer
  - starting 99
- numeric formats 146

## O

- Oracle
  - database connection parameters 178
  - Desktop build settings 56, 84
  - NLS setting 215

## P

- parameter file
  - client/server connection parameters 176
  - local host connection parameters 175
- performance 194
- Perl scripts
  - overview 17

- reference 196
- print Desktop information 65
- Printer Setup Maintenance 159
- printers, remote 222
- process maps
  - migrating between releases 157
  - translated 141
- PROPATH 205
  - Desktop system setting 205
- Pulist.exe 36

## Q

- QAD Support Services 6
- quick start 66

## R

- remote printers 222
- requirements
  - client 20
  - client browser 21
  - prerequisites 20
  - Web server 23
- Restrict Menu field 222
- restricted shells 185

## S

- saving browse settings 222
- search
  - creating database 213
  - maximum results 210
- security
  - restricted shells 185
  - telnet, UNIX 184
- server
  - database 172
  - Web
    - installing 32
- Service Pack
  - directory field 54
- service pack
  - directory field 82
  - effect on Desktop 158
- Session Master Maintenance 232
- session timeout 224
  - Connection Manager 119
- Show Java Console check box 193
- single-tier deployment 39–68
  - build Desktop 55
  - generate connection scripts 61
  - HTML and script files 44
  - launch scripts 42
  - MFG/PRO code 47

- MFG/UTIL 51
- telnet scripts 49, 77
- Tomcat 45
  - WebSpeed configuration 60
- single-tier installation
  - quick start 66
- SpeedStart 181
- Support Services 6

## T

- telnet scripts
  - generating 61
- telnet server
  - Connection Manager settings 117
  - connection settings, configuring 109
  - log-in sequence, configuring 105
  - overview 12
  - registering on Windows 37
  - server settings, configuring 104
  - UNIX security, configuring 184
  - verifying on UNIX 109
  - Windows server settings, configuring 104
- timeout
  - browse 224
  - Connection Manager 119
  - minutes 224
  - session 224
- Tomcat
  - deploy single-tier 45
  - host name 206
  - installing new version 189
  - J2SDK version 174
  - multi-tier deployment 169
  - port 206
  - security 102
  - starting 101
  - Webapp name 206
- Tomcat installation directory
  - items installed in 174
- Tomcat Web Application Manager 113, 191
- troubleshooting
  - Connection Manager 122
  - performance 194
  - tools 192
- two-tier Desktop deployment 166

## U

- ubroker.properties 60
  - two-tier deployment 179
- uniform resource locators (URLs)
  - browse links 225
  - menu-level links 225
- UNIX telnet security 184
- User Function Maintenance 234
- User Option Maintenance 221
  - effect in Desktop 221
- User Option Telnet Maintenance 103
  - log-in sequence 105
  - session timeout 119
  - telnet connections 109
  - telnet server settings 104
- user options
  - expiration date 222
  - menu substitution 222
  - program execution 222
  - restrict menu 222
- User Tool Maintenance 234

## W

- Web archive file (WAR) 12
- Web browser
  - overview 12
- Web server
  - host name 208
  - installing 32
  - multi-tier deployment 170
  - overview 11
  - port number 208
  - setting up MIME-types on UNIX 33
  - verifying 34
- WebSpeed
  - broker name 208
  - messenger executable 208
  - multi-tier deployment 170
  - server host name 209
  - server port number 209
  - verify servers 99
- WebSpeed Messenger
  - two-tier deployment 178
- WebSpeed server
  - agent sizing 181
  - log files in ubroker.properties 100
  - wspd\_cgi.ksh 208