Oracle Web Application Server™ Installation Guide

Release 3.0.1 for Windows NT

Part No. A54857-03

ORACLE®

Enabling the Information Age

Oracle Web Application Server™ Installation Guide
Release 3.0.1 for Windows NT

Oracle Government Purchasing, Oracle Government Receivables, Oracle Government Revenue Accounting, Oracle Graphical Schema Editor, Oracle Graphics, Oracle Human Resource Management Systems, Oracle Human Resources, Oracle Illustrated, Oracle Illustrated Series, Oracle Imaging, Oracle Incident, Oracle Industries, Oracle Installer, Oracle InstantSQL, Oracle Integrator, Oracle Internet Commerce, Oracle Internet Server, Oracle InterOffice, Oracle InterOffice Client, Oracle InterOffice Manager, Oracle InterOffice Server, Oracle Inventory, Oracle Magazine, Oracle Magazine Interactive, Oracle Manufacturing, Oracle Master Scheduling, Oracle Master Scheduling / MRP, Oracle Media Data Store, Oracle Media Library, Oracle Mission Control, Oracle Mobile Agents, Oracle Module Language, Oracle MRP, Oracle MultiProtocol Interchange, Oracle Names, Oracle NetSolutions, Oracle Network Manager, Oracle Newsroom Manager, Oracle Object Marketplace, Oracle Objects, Oracle Office, Oracle Office Directory, Oracle Office Mail, Oracle Office Manager, Oracle Office Scheduler, Oracle Online, Oracle Open Client Adapter, Oracle Open Gateways, Oracle Open World, Oracle Order Entry, Oracle Parallel Server [or Oracle7 Parallel Server], Oracle Payables, Oracle Payroll, Oracle Personal Time and Expense, Oracle Planner Workbench, Oracle PowerBrowser, Oracle Procedure Builder, Oracle Process Modeller, Oracle Product Configurator, Oracle Project Accounting, Oracle Project Billing, Oracle Project Costing, Oracle Projects, Oracle Public Sector, Oracle Purchasing, Oracle Quality, Oracle RALLY, Oracle Rdb7, Oracle Receivables, Oracle Release Management, Oracle Replication Manager, Oracle Replication Services, Oracle Reports, Oracle Reports Generator, Oracle Repository Administrator, Oracle Revenue Accounting, Oracle RMU, Oracle Sales Analysis, Oracle Sales Analyzer, Oracle Sales and Compensation, Oracle Sales and Marketing, Oracle Sales Brief, Oracle Sales Compensation, Oracle Server Generator, Oracle Server Manager, Oracle Smart Video, Oracle Store, Oracle System Sizer, Oracle SQL*Tutor, Oracle SQL / Services, Oracle Supplier Scheduling, Oracle Supply Chain Planning, Oracle SupportNotes, Oracle Systems Designer, Oracle Systems Modeller, Oracle Terminal, Oracle Text Server, Oracle TextServer3, Oracle Toolkit, Oracle TRACE, Oracle TRACE Collector, Oracle TRACE Option, Oracle Training Administration, Oracle Translation Manager, Oracle Universal Database, Oracle Upstream, Oracle Video Client, Oracle Video Server, Oracle Web Customers, Oracle Web Employees, Oracle Web Suppliers, Oracle Web Application Server, Oracle Work in Process, Oracle Workflow, Oracle Workgroup Server [or Oracle7 Workgroup Server], Oracle*Mail, Oracle7, Oracle7 Enterprise Backup Utility, Oracle7 Server, Oracle7 Spatial Data Option, Oracle8, Oracle 64 Bit Option, Oracle / 2000, PC Express, Personal Express, Personal Oracle [or Personal Oracle7], Personal Oracle Lite, PJM Advantage, PL / SQL, Profit, ProREXX, Pro*C, Pro*C / C++, Pro*REXX, Programmer / 2000, ProRexx, RDB7, Report Card, Security Without Compromise, Server / 2000, Services / 2000, Set-top / 2000, Smart Application Client, SmartBox, SmartCharts, SmartClient, SmartHints, SmartLayout, SmartSpring, SmartStandards, SmartTab, SmartTriggers, SQL*TextRetrieval, SQL*VDM, SupportAssistant, SupportNotes, SupportNews, The Oracle Network Builder, Trusted Oracle, Trusted Oracle7, Tutor, Video Client, Video Server, Web Request Broker, Workgroup / 2000, World / 2000.

SERVICE MARKS of Oracle Corporation:

BAP, Business Alliance Programme, CASE*Method, Cooperative Applications Initiative, International Oracle User's Group, International Oracle User's Week, IOUG, IOUW, Migration Technology Initiative, OOW, Operations Readiness Assessment, Oracle Alliance Program, Oracle Bronze, Oracle Business Alliance Programme, Oracle Consulting Services, Oracle Education, Oracle Gold, Oracle Master, Oracle Mercury, Oracle Metals, Oracle Platinum, Oracle Service, Oracle Silver, Oracle Sterling, Oracle SupportFax, Real Time Support Services, Systems Management Tools Initiative, Warehouse Technology Initiative, Web System Initiative.

All other products or company names are used for identification purposes only, and may be trademarks of their respective owners.

Contents

Preface

This *Installation Guide* is the primary source of introduction, installation, and configuration information for Oracle Web Application Server for Windows NT.

Audience

This *Installation Guide* is necessary for installing, configuring, or administering Oracle Web Application Server 3.0.1 for Windows NT.

Prerequisites

This *Installation Guide* assumes that you are familiar with:

- Windows NT and have installed and tested it for your PC and network hardware
- Oracle7 or Oracle8 Server and relational database concepts

Oracle Documentation Set

Online documentation is provided in HTML and Adobe Acrobat Portable Document Format (PDF). To read the HTML documentation, go to the Web

Listener Administrator Home page, and follow the online documentation link icon. The PDF documentation can be found by clicking the online documentation icon in the Oracle for Windows NT program group.

Note: To read the PDF online documentation, install the Adobe Acrobat Reader by double-clicking the file ACROREAD.EXE in the ACROBAT directory on the product CD.

The Web Application Server 3.0.1 for Windows NT contains the following documentation:

- *Oracle Web Application Server Installation Guide for Windows NT* - This contains installation information for the Oracle Web Application Server for Windows NT.
- *Overview* - This provides general information about the Oracle Web Application Server.
- *Using Oracle Web Application Server Cartridges* - This describes how to use the cartridges provided by Oracle for the Web Application Server.
- *Developing your own Web Application Server Cartridges* - This provides information for developing your own Web Application Server cartridges.
- *Security* - This describes how to use the security features of the Web Application Server. It also describes how to generate certificate requests.

Related Oracle Documentation

Oracle Server Getting Started for Windows NT

Oracle Server Administrator's Guide

Oracle Server Utilities User's Guide

Oracle Server Messages and Codes Manual

SQL Language User's Guide

SQL Language Reference Manual

*SQL*Plus User's Guide*

Conventions Used in This Manual

This table lists the typographical conventions used in this manual.

Convention	Meaning
Monospace text	Indicates text that must be typed exactly as shown. `set echo off`
Bold	Used for filenames, directories, and utilities such as: **owsctl**.
Italics	Indicate a value that you must provide. For example, if a command asks you to type *filename*, you must type the actual name of the file. Italics are also used for emphasis in the text and to indicate the titles of other manuals.
Horizontal ellipsis...	Indicate that parts of the statement or command not directly related to the example have been omitted. CHKVAL *fieldname value 1 value 2... valueN*
Vertical bar |	Represents an 'or' option between several options. You must enter only one of the options. Do not enter the vertical bar. The set of alternative choices is enclosed by curly braces if one of the items is required, or by square brackets if the item is an optional alternative.
Curly braces {}	Enclose required items. You must choose one of the alternatives. ..DEFINE { *macro1* | *macro2* }
Square brackets [ ]	Enclose optional items. You can choose one or none of the alternatives. cvtcrt *termname* [outfile] Square brackets also indicate a function key, for example [Enter].
C:\>	Represents the Windows NT command prompt of the current hard disk drive. Your prompt may differ and may, at times, reflect the subdirectory in which you are working.

Convention	Meaning
Symbols	Symbols other than brackets and vertical bars must be entered in commands exactly as shown.

Customer Support

Oracle Worldwide Customer Support (WWCS) can be reached at the following number

1-650-506-1500 in the United States of America

Be prepared to supply the following information:

- your CSI number (this helps Oracle Corporation track problems recorded for each customer)
- the release numbers of the Oracle Web Application Server and associated products
- the operating system name and version number
- details of error numbers and descriptions (write down the exact errors—it helps Oracle Technical Support track down the problem more quickly)
- a description of the problem
- a description of the changes made to the system

Documentation Sales

To order printed documentation, please call:

1-800-252-0303 in the United States

Your Comments Are Welcome

We value and appreciate your comments as an Oracle user. We encourage you to send your comments to us at the following address:

Web Application Server Documentation Manager

Oracle Corporation

600 Oracle Parkway
Redwood Shores, CA 94065

CHAPTER

1 Introduction

Overview of the Product

As the World Wide Web matures, a new generation of Internet and Intranet business applications is emerging. These new applications will incorporate real business transactions, data-driven multimedia content, and interactive information. Database-powered Web applications built on this new platform will enable corporations to expand their services and customer base as well as explore lucrative new business endeavors while still leveraging their investment in existing client-server systems. Such powerful Web applications demand a new breed of Web platform that combines all the power and reliability of traditional client-server environments with the flexibility and ease-of-use of the Internet.

Operating within the framework of Network Computing Architecture, Oracle Web Application Server 3.0.1 expands dramatically upon the power of standard Web servers, enabling the development and deployment of full-featured transaction-based Web sites that are scalable, reliable, and secure.

Note: Refer to the online glossary for definitions of terms with which you may not be familiar.

Open Architecture

A typical technology infrastructure involves a wide range of operating systems, languages, networks, applications, Web servers, and databases.

Oracle Web Application Server 3.0.1 operates across a wide range of operating systems, from PCs to workstations, and through its open cartridge API, supports a diverse class of languages and applications. Oracle Web Application Server 3.0.1 is the first Web server to support real transactions and it does so using X/Open DTP standards.

Note: A cartridge is a program, run on the server by the Web Request Broker (WRB), that interfaces to a Web server (Oracle or otherwise) through the WRB API. A given cartridge will have a varying number of execution instances called WRBXs.

Transaction Enable your Netscape or Microsoft Server

The WRB is portable not only across a range of operating systems, but across a range of Web servers. Oracle bundles its own HTTP listener with Web Application Server 3.0.1, but this portability means that it can also integrate with Netscape and Microsoft listeners. This allows application developers to protect their existing investments as technological infrastructures change.

Scalability

Building a presence on the Web can expand horizons considerably, but only if your Web site can perform and scale to support the vast Internet audience. Oracle Web Application Server 3.0.1 can scale to serve tens of thousands of users. Through the WRB, Oracle delivers vastly superior performance, dispatching, and access times, even in the extremely high network traffic environments that are synonymous with the Internet. Built to a true multi-threaded, multiprocess architecture, the Web Request Broker offers a superior application environment over low-level, first generation HTTP APIs.

Reliability

A mission-critical application demands high availability. Oracle Web Application Server 3.0.1 brings the robustness and reliability of the client-server world to the Web. Process separation, an object architecture, and independent cartridge management allow administrators to build, manage, and service their system on a component basis. Through its independent processing architecture, the WRB guarantees that third-party server extensions will not affect other parts of the system security.

Security

With all the potential of real business applications, but no face-to-face contact, the Web presents new security challenges. Oracle Web Application Server 3.0.1 supports full end-to-end security at the client, at every level within the Web server architecture, and through the firewall to an Oracle database. This unprecedented degree of granularity supports not only username-password protection, but also custom security schemes. For even greater protection of vital data, Oracle Web Application Server 3.0.1 supports industry standard SSL 3.0.1 encryption, as well as Oracle Advanced Networking Option for secure communications and transactions.

Supported Features

Oracle Web Application Server 3.0.1 — Standard Edition

- cartridge-based development platform
- CORBA compliant ORB-based Web Request Broker
- support for fully distributed applications
- listener independence
- language-independent development
- third-party cartridges available through the cartridge solutions network

CORBA-compliant ORB

- Web Request Broker (WRB) services implemented as CORBA-compliant object services
- supports distributed listener and cartridge architecture

Web Request Broker

- automatic server redirection on maximum connections reached
- safe, scalable architecture
- dynamic load-balancing
- automatic context management
- open API for custom extensions

Fully portable WRB API

- CORBA-compliant ORB-based Web Request Broker
- unifying API for Netscape, Microsoft, and Oracle HTTP servers

Java Cartridge

- native Java environment
- auto-generated wrapper classes for PL/SQL
- native access to Oracle
- HTML presentation classes
- National Language Support (NLS)
- support for persistent database connections
- HTML 3.2 support

PL/SQL Cartridge

- 100-percent data encapsulation through stored procedures
- 100-percent portable code
- transparent dispatch to Oracle server
- automatically translates HTML parameters to PL/SQL calls
- HTML 3.2 support
- ICX and transaction support
- enhanced error mapping to relay RDBMS messages to user
- object-oriented design

LiveHTML Cartridge

- enhanced Server Side Include (SSI)
- access to other cartridges through ICX
- HTML files can be target of an HTML form submission

Perl Cartridge

- Perl Version 5 interpreter
- Oracle DBI/DBD extensions
- OraPerl Emulation

VRML Cartridge

- VRML 2.0 standard support
- platform for building and deploying business applications in VRML
- VRML Data Repository that manages persistent, scalable, and secure VRML worlds
- Logic Repository ties together database triggers with the VRML event model
- easy-to-use mechanism for embedding dynamic content, generated by custom scripts or SQL statements within a VRML scene

Inter-cartridge Exchange

- transport-independent, stateless protocol
- mirrors the HTTP request model
- set of APIs to allow a cartridge to address, send, request, and receive a response from another cartridge

Security

- IP address restriction
- domain name restriction
- basic authentication
- digest authentication
- SSL 3.0 (International version uses 40-bit key)
- client-side digital ID authentication

Built-in Logging and Analysis Tools

- support for clf / xlf system message formats and client-defined statistics
- support for log file cycling / archiving based on size / date
- support for logging into the database
- graphical log analyzer tool
- report generation on accesses, errors, clients, URLs, etc.

Common Gateway Interface

- CGI 1.1 compliant

- dedicated process per request
- automatic cleanup

Miscellaneous

- native imagemap support
- multiple imagemap extensions
- configurable DNS resolution
- Common Log Format

Optional Extensibility

- OCI cartridge (Oracle Call Level Interface)
- Rdb cartridge
- Oracle Security Server
- Oracle Internet Commerce Server (cartridge-based solution for electronic commerce)
- third-party cartridges available through the Cartridge Solutions Network

Oracle Web Application Server 3.0.1 — Advanced Edition

In addition to the standard features, the advanced edition offers the following:

ODBC Cartridge

- accepts SQL statements
- returns HTML table with formatted results
- optional use of format strings
- callable through ICX from any other cartridge

Transaction support

- supports X/Open DTP model
- supports open standards — SQL, X/Open's XA and X/Open's TX
- defines a set of APIs that is modeled on the XA interface

- APIs to start or join a transaction, retrieve transaction information, commit and rollback a transaction
- transactional support across multiple cartridges

Persistent Storage Services

- APIs for storing and retrieving content or managing content from a SQL database
- schema attributes include content-type, author, creation date, etc.
- service supports Oracle DBMS, or file system storage

Products Available for Installation

Components

- **Oracle Web Listener**
- **Web Application Server Cartridges** - The PL/SQL, Java, and Live HTML cartridges are bundled under the label Web Application Server cartridges and are automatically installed on the single node and primary multi-node install.
- **Web Request Broker**

Additional Cartridges

You may choose to install the following additional cartridges during installation:

- ODBC
- JDBC
- VRML

CHAPTER

2 Installation Requirements

System Requirements

Hardware Requirements

Table 2-1: Hardware

Hardware Item	Required
CPU	An Intel compatible 486 or higher processor
Memory	64 MB (32 MB for a listener only machine)
Disk Space	150 MB
Swap Space	64 MB
CD-ROM Device	RockRidge format

Software Requirements

Table 2-2: Software

Software Item	Version
Operating System	Windows NT Server v. 4.0
Web Browser	Netscape Navigator 3.0 or later Any browser that supports tables and forms
Listener	Oracle 40-bit Oracle 128-bit Netscape FastTrack V2.0, V2.0.1 Netscape Enterprise Server V2.0, V2.0.1 Microsoft Internet Information Server V2.0, V3.0
Oracle RDBMS	7.1.6 7.2.2x 7.2.3x 7.3.2.x 7.3.3.x 8.0.3 8.0.4 8.0.5
JAVA Developer Kit (JDK)	1.0.2

Note: For cartridges that link with the Oracle client libraries (OCI/PRO*C and transactional), the Oracle 7.3 RDBMS client libraries must be used. With these libraries, the cartridge can connect to supported versions of either Oracle 7.x or Oracle 8.0.x databases.

Product Dependencies

If you want to use Oracle Web Application Server with an Oracle database, you are required to install other products. The following table lists the required

Oracle products and minimum release levels for using the Web Application Server with an Oracle database.

Table 2-3: Required Products

Products	Minimum Release
Oracle RDBMS	7.1.6
PL/SQL	2.1.6
SQL*Net	2.1.6
TCP/IP Protocol Adaptor	2.1.6

Note: The Oracle Server and PL/SQL are not installed automatically with Web Application Server.

Remote Database Installation

If you wish to access an Oracle database on a remote machine, install the products listed above on the remote machine.

On your local machine, you must also install SQL*Net and the TCP/IP Protocol Adapter provided on the Web Application Server CD.

Local Database Installation

If you wish to install Oracle Web Application Server as a stand-alone Internet server, and you want to access a database on your local machine, you need only install an Oracle database and PL*SQL.

Note: If you wish to use either Oracle8 or the Multi-Threaded Server option to connect to a local database, you must install SQL*Net 2.3 and the TCP/IP Protocol Adaptor. These products are automatically installed with Oracle Web Application Server 3.0.1.

Supported Configurations

Oracle Web Application Server can be installed as a single-node, where everything is installed on a single computer; or as a multi-node, which consists of a primary-node installed on one computer and multiple remote-nodes installed on different computers.

Oracle Web Application Server is made up of several processes, and you can run these processes on different machines on the network. You can do this because the architecture of the Web Application Server is based on CORBA (common object request broker architecture), which is a standard for distributed objects.

One advantage of distributing the processes on different machines is performance and scalability. You can handle more requests without using up too much resources from one machine.

See Appendix C, "Multi-node Configuration" on page B-1 for additional information.

The following are examples of typical Web Application Server installation configuration choices:

Single-Node

This installation shows components installed on a single-node:

- WRB with Oracle Administrator Listener
- Listener (Oracle, Microsoft, or Netscape)
- Web Application Server cartridges (required)
- Additional cartridges (optional)

Multi-Node Primary

This installation shows components installed on the primary-node in a multi-node configuration.

- WRB with Oracle Administrator Listener
- Listener (Oracle, Microsoft, or Netscape)
- Web Application Server cartridges (required)
- Additional cartridges (optional)

Multi-Node Remote - Cartridge Only Installation

This installation shows an example of a remote-node in a multi-node configuration.

- Web Application Server cartridges (required)
- Additional cartridges (optional)

Multi-Node Remote - Listener Only Installation

This installation shows an example of a remote-node in a multi-node configuration.

- Listener (Oracle, Microsoft, or Netscape)

Multi-Node Remote - Cartridge and Listener Installation

This installation shows an example of a remote-node in a multi-node configuration.

- Web Application Server cartridges (required)
- Additional cartridges (optional)
- Listener (Oracle, Microsoft, or Netscape)

CHAPTER

3 Installation Procedure

Supported Installation Activities

The following installation activities are described in this chapter:

- Pre-installation Information
- First-time Web Application Server Installation
- Installing over an Existing Web Application Server
- Auto-Starting Oracle Web Application Server Listeners
- Uninstalling Oracle Web Application Server

You may also use the Installer to configure third-party Web servers for use with Oracle Web Application Server. See the chapters on configuring third-party servers and migrating to the Oracle Web Listener for more information on using third-party servers.

Note to Oracle8 Users: You must follow the pre-installation steps described in the section "Special Oracle8 User Information" on page 3-2 to be able to successfully install Oracle Web Application Server Release 3.0.1 with your Oracle8.0.x database.

Installing a Single Node

You may do a fresh, single-node installation of Oracle Web Application Server Release 3.0.1. Refer to "Upgrading from Previous Releases" on page 7-1 for information on upgrading from Version 2.x.

Installing a Primary Node in a Multi-node Configuration

The following activities are supported when installing a primary node in a multi-node setup.

- first time 3.0.1 installation
- 3.0.1 reinstallation (configuration of third-party HTTP products saved automatically on partial install)
- 3.0.1 reinstallation and configure third-party HTTP products using complete installation option
- configure if third-party server detected on first time installation

Installing a Remote Node in a Multi-node configuration — Cartridges Only

The following installation activities are supported when installing cartridges only:

- first time 3.0.1 installation
- partial 3.0.1 reinstallation
- complete 3.0.1 reinstallation

Add Components

You may add components, such as listeners, Oracle Web Application Server cartridges, or other cartridges to an existing 3.0.1 installation using the Oracle Installer.

Pre-installation Information

In addition to deciding whether to install a single-node or a multi-node configuration, you will need the following information. Oracle suggests that you gather this information before beginning the installation.

Special Oracle8 User Information

An Oracle8 database and Oracle Web Application Server can be installed on the same machine or different machines. However, doing so requires special considerations, which are outlined in the following steps:

1. Make sure that the **net80** networking component is installed with the Oracle8 database.

2. Install Oracle Web Application Server. During installation, make sure that you check the “Remote Oracle RDBMS Connection (SQL*Net V2.3)” option in the optional cartridge installation section.

   If during installation “Remote Oracle RDBMS Connection” was not selected, then make sure that SQL*Net V2.3 on the Oracle Web Application Server CD is installed.

3. Use the SQL*Net Easy Configuration tool to create a database alias for the database instance.
4. Ensure that TRACE_CLIENT_LEVEL = ON in the **sqlnet.ora** file.
5. When configuring DADs, you must connect to the Oracle8 database as a remote database even if the Oracle8 database is on the same machine as Oracle Web Application Server.

Standard Settings

The following settings are common to all installations:

- **Language** - The default language is English.
- **Company Name** - A text string identifying the name of your company.
- **Oracle Home Directory** - The directory for the %ORACLE_HOME% environment variable. The default is **c:\orant**
- **Site Name** - The site name for the server. The default is “WEBSITE30”.
- **Host Name** - This setting is used for the primary node in a multi-node installation. The default is the local machine name as defined in the Registry.
- **Remote List** - This is a list of the remote node names that will form this site and is needed when installing the primary node in a multi-node installation.
- **UDP Service Port** - The UDP service port is used by the ORB and WRB process. The default is 2649. You can use any number between 1024 and 65535.
- **Shared Key** (in Hex) - This key is used for encryption in a multi-node configuration. You will need to enter the same key in all nodes. If you are setting up a single node, you should still specify a shared key to prevent unauthorized processes from connecting to your server.

Note: If you plan to add multiple nodes in the future, you should enter a value for the shared key during installation.

Administrator Listener Settings

This information is required for a single node or primary node (in a multi-node) installation.

- **Port Number** - Defaults to 8888.
- **User Name** - Defaults to “admin”.
- **User Password** - The password you will use for the node.

Web Listener Settings

The following information is needed for the general usage Web listener. You may use the Oracle listener or a third-party HTTP listener such as Netscape FastTrack or Microsoft IIS.

- **Web Listener Name** - Defaults to “www” for the Oracle listener.
- **Port Number** - Defaults to 80.

Note: The Microsoft IIS listener does not take a listener name. You only need to set the port number.

NLS_LANG Variable

If you will need to access an NLS database, you must set the NLS_LANG variable in the environment using the Control Panel. Double-click on the System icon and select the Environment tab.

First-time Web Application Server Installation

These instructions assume that your Oracle home directory is **c:\orant** and that your CD-ROM drive is mapped to the G: drive.

At any time during the installation, you can select the Help button to receive additional information about the information requested on the installation form.

If you have already installed Oracle products, shut down all instances and stop all Oracle services currently running before you attempt to install the new software.

1. Start the Oracle Installer.

   When you insert the Oracle Web Application Server CD-ROM into the

CD-ROM drive, the Windows “auto run” feature automatically starts the Oracle Installer program.

If your CD-ROM drive does not support the “auto run” feature, you will need to navigate to the following directory using any of the standard Windows navigation methods.

G:\NT_X86\INSTALL

Then launch the **setup.exe** program, which starts the Oracle Installer.

2. Choose the language you want to use during installation.
3. Enter the following Oracle Installation Settings:

   Company Name

   Enter the name of your company.

   Oracle Home

   Enter the directory for the ORACLE_HOME environment variable. This will be the directory under which the Oracle Web Application Server directory tree will be placed. ORACLE_HOME can be any alphanumeric value up to 256 characters.

   The default is **c:\orant**.

   Warning: Windows NT does not support multiple ORACLE_HOMEs. If you already have an existing ORACLE_HOME, changing it here will disable your currently installed Oracle products.

   If you decide to change the default, you can use the browser folder button to browse your computer’s directory tree, or type in a new directory directly into the ORACLE_HOME field.
4. The Installer displays two README files with useful information about the installation. Use the scrollbar to page through the files. Click OK to proceed.
5. Select the Oracle Web Application Server Installation activity you wish to perform. The options are:
   - Install a Single Node
   - Install a Multi-Node

Install a Single Node

Select a single-node installation if you want to complete a full Oracle Web Application Server installation on a single machine. If you choose a single-

node installation, the Web Request Broker (WRB), Web Listener, and cartridges are installed on the same machine.

Install a Multi-Node

Select a multi-node installation if you want to install different components on separate machines. For example, if you want WRB, Listener, and cartridges installed on three separate machines, you should select a multi-node installation.

If you install Oracle Application Server in a multi-node configuration, you must install one primary node and at least one remote node.

- **Primary multi-node installation -** The primary node is where your WRB and configuration files are installed.
- **Remote multi-node installation -** Related remote nodes are nodes that share the same WRB. Remote node installation allows you to specify which components of the Oracle Web Application Server you want for a specific node.

If you select remote multi-node installation, you are prompted to choose from the following components:

- Oracle Web Listener
- Oracle Web Request Broker
- Oracle Web Application Server Cartridge

Remote node installations may be run as listener only, cartridge only, or listener and cartridge nodes.

Note: If you are performing a multi-node installation, you will need to repeat the installation procedure for each node. That is, you can only install one node with each pass through the Oracle Installer. Each pass must be run on the individual node.

For example, if you want to install a primary node and three remote nodes, you will need to make four passes through the installer making the appropriate selection for each node.

Note: For a single- or primary multi-node installation, the bundled cartridges are automatically installed. The Product Components screen does not appear. The bundled cartridges are PL/SQL, JAVA, LiveHTML, Perl, and VRML.

6. Choose the optional components/cartridges that you want to install.

   You may choose to install additional components or cartridges that are packaged outside of the Web Application Server bundle.

Note: You must install a copy of the cartridge on the node where you will perform administration (single or primary node) in order for the Web Application Server Manager to be able to register the cartridge. You will be able to administer the actual remote node cartridge from the primary node.

The optional components / cartridges are:

- ODBC Cartridge
- Oracle JDBC Drivers (Beta)
- Remote Oracle RDBMS Connection (SQL*Net V 2.3)
- VRML Cartridge

Note: If you use Oracle8, you must install SQL*Net regardless of whether your Oracle8 installation is local or remote.

7. Site Installation

   Enter the following information about the installation site.

   Site Name - Defaults to "WEBSITE30". Oracle Web Application Server allows you to have multiple server sites running in a single installation environment. Servers are differentiated by site names, so you must enter the current Web server site name.

   Host Name - Defaults to the local machine name as defined in the Registry. If this is a multi-node remote installation, you need to supply the name of the primary node for this remote installation.

   Host List - If this is a multi-node primary installation, you need to supply the list of the remote nodes that will be used in this configuration. Enter the fully qualified hostnames of all remote nodes in this site. For example, **lindros.us.oracle.com**. Hostnames should be separated by a comma and a space.

   UDP Service Port - Defaults to 2649. The UDP service port is used by the Object Request Broker (ORB) and Web Request Broker (WRB) processes. If the default is already being used by another process, provide a different number. You may choose a number between 1024 and 65535. For a remote multi-node, you need to supply the UDP Service Port of the primary host.

   Shared Key (in Hex) - If you use a multi-node configuration, the shared key feature allows you to encrypt messages between the primary node and remote nodes for security purposes. This ensures

that only users with proper authority can see the contents of the messages. To encrypt these messages, you use a shared key that is known among the different nodes of the Web Application Server distributed installation. The shared key is used to drive encryption of all messages sent or received by an ORB program.

Note: If you are installing a multi-node, you should enter the same key value for each node here.

The shared key can vary from 0 to 255 bytes. The string is made up of only hexadecimal characters; for example, 0-9 and a-f. Larger shared keys have higher security. Shared key size has minimal impact on performance.

This field should be left blank if you do not wish to use encryption.

8. Administration Listener Information

   This screen only displays on a single- or primary-node installation when the Web Request Broker is installed. System administrators will use these settings when they manage the system using the Oracle Web Application Server Administration Tool.

   Enter the following Administration listener settings:

   Port Number - Defaults to 8888. You should not change this value.

   User Name - Defaults to “admin”. It is recommended that you not change this value.

   Password - Enter your administration password.

   Confirm Password - Confirm the password.

9. Listener Choice

   This dialog appears when one of the following products is detected and the Web Listener component has been selected:

   - Oracle Web Application Server 3.0.1 Listener
   - Microsoft Internet Information Server
   - Netscape FastTrack
   - Netscape Enterprise Server

   If you use a third-party server from Microsoft or Netscape, you may chose to configure it during installation. If you wish to use the Oracle Listener included with Web Application Server, select the choice “Oracle Web Application Server 3.0.1 Listener”.

10. Oracle Web Listener Configuration

    Note: Only one general usage listener can be configured during installation. To configure additional listeners, use the Web Application Service Manager.

    Enter the following Oracle Web Listener Configuration information.

    Web Listener Name - Defaults to "www" for the Oracle listener. This is the application specific name. You can change it to match your listener name.

    Note: The Microsoft Internet Information Server does not use a listener name.

    Port Number - Defaults to 80. Do not change this number.

    Note: For Microsoft, you need to follow the Post-Install steps in "Post-Installation and Administration" on page 4-1**.**

11. Copy Files

    This dialog allows you to confirm your configuration parameters. When you click OK, the installation is performed.

12. If you chose to install SQL*Net V 2.3 products, you are now asked to specify which SQL*Net V2.3 products you want to install.

    a. Select the SQL*Net products you wish to install. Your choices are: SQL*Net Client Version 2.3.2.1.6A and SQL*Net Protocol Adapters.

    b. If you select the SQL*Net Protocol Adapters, you will be asked which Protocol Adapters you want to install. Choose the Oracle TCP/IP Adapter for use with Oracle Web Application Server.

    Note: Oracle Named Pipes Adapter, Oracle SPX Adapter, and Oracle DECNet Adapter are available, but are not supported for use with Oracle Web Application Server Release 3.0.1.

    Note: For first-time installations, you must restart your machine for the PATH settings to take effect.

13. After you restart your system, start the Web Request Broker (WRB) and listeners. Refer to "Starting Web Application Server" on page 4-1 for instructions on how to start these components.

Installing over an Existing Web Application Server

You may upgrade from Release 3.0 to Release 3.0.1 of Web Application Server. If the Installer detects an existing Web Application Server installation, it may ask questions that are slightly different from those in a new installation. Some of the questions you encounter are:

Reinstall Web Application Server

If the Installer detects an earlier version of Web Application Server, you have two choices. You may:

- **Upgrade** - This option installs Web Application Server Release 3.0.1 and *preserves* your existing Web Application Server configuration information. An upgrade copies library and message files only based on component choices and skips configuration questions.
- **New Install** - This option installs Web Application Server Release 3.0.1 and *overwrites* your existing Web Application Server configuration information. A new install backs up old files before overwriting them.

  In a new install, the configuration files for the administration listener and the default listener (for example **svadmin.cfg**, **wrb.app**, and **svwww.cfg**) will be replaced with the default versions. Your configuration files will be backed up with **.bak** appended to the filenames. The web page for your default web listener, **index.html**, will also be backed up.

  Existing NT registry values used by the Web Application Server will be replaced; however, these old registry values will not be backed up.

Note: You may want to use standard Windows NT techniques to back up your Windows NT registry prior to re-installing Oracle Web Application Server.

Add Components

This option appears in the Installation Activity dialog if the Installer detects an existing installation of Web Application Server. It also appears during a Remote multi-node installation. Choose the components that you want to install on this node. You may choose any number of the following components:

- **Oracle Installer**

- **Oracle Web Listener** - You will be given the opportunity to re-configure this Listener to run Microsoft or Netscape HTTP server in a later dialog in this install.
- **Oracle Web Request Broker** - Installs the Oracle Web Request Broker.
- **Web Application Server Cartridges** - Installs the bundled cartridges. These include PL/SQL, JAVA, LiveHTML, Perl, and VRML cartridges.

Most other installation prompts are similar to the new installation prompts outlined in the preceding section.

Auto-Starting Oracle Web Application Server Listeners

The Oracle Installer sets the OracleWRBPrimaryService to Automatic startup, which means that it will start when the machine boots. Because correct listener startup depends on a running WRB, you cannot use the Services Control Panel to configure the listeners to start automatically at boot time as well. However, you can configure the listeners and WRB to start in the correct order at login time.

If you would like Oracle Web Application Server to automatically start after you have logged on to the machine, use the following instructions:

1. In your Services Control Panel, set the OracleWRBPrimaryService to Manual startup rather than Automatic.
2. Create a text file called **startweb3.cmd**.
3. Put the following lines in the file:

   ```
   owsctl start wrb

   owsctl start admin
   ```

4. Place the file in the **\WINNT\Profiles\All Users\Start Menu\Programs\Startup** directory.
5. Restart your machine; the admin listener should come up.

Uninstalling Oracle Web Application Server

To uninstall the Oracle Web Application Server for Windows NT, follow these steps:

1. Shutdown all the listeners (including the Administration listener, Netscape, and Microsoft IIS listeners).

   Warning: Do not shut down services using the Control Panel Services window. Use the following command at the DOS prompt:

   ```
   owsctl stop <ListenerName>
   ```

   Where `<ListenerName>` is the name of the listener you want to shut down.

2. Once all services have been terminated, shutdown the Web Request Broker (WRB) services.

   ```
   owsctl stop wrb
   ```

3. If you are running Microsoft Internet Information Server (IIS) configured with the Oracle WRB, use the **regedt32** to delete the following entries from the NT Registry:

   - Under **HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\W3SVC\Parameters**, remove **ndwfis30.dll** reference in the Filter DLLs entry.

4. If you are running the Netscape servers, restore the old configuration to the **obj.conf** file:

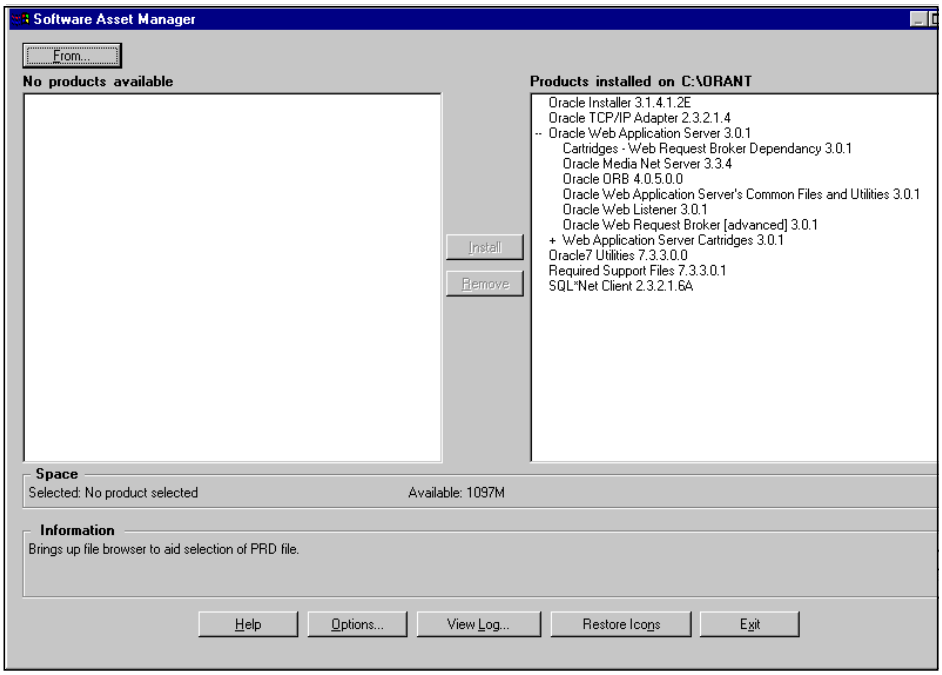
   - For Netscape Enterprise Server, in the **%Netscape Server Home%\https-***wrb-configured server name***\config** directory, move the **obj.conf** file and restore the old configuration from **obj.conf.sav**.
   - For Netscape FastTrack Server, in the **%Netscape Server Home%\https-***wrb-configured server name***\config** directory, move the **obj.conf** file and restore the old configuration from **obj.conf.sav**.

5. Run the Oracle Installer:

   - From the Start menu, select Programs > Oracle for Windows NT > Oracle Installer.

- Respond to the initial Installer prompts as described in "First-time Web Application Server Installation" on page 3-4. After you specify your Oracle Home, the Software Asset Manager is displayed.

Figure 3-1: Software Asset Manager



The right side of the window displays products installed on your system. Select the products you wish to remove. Use Shift + click, or click and drag to select multiple items.

- Click Remove. The Installer will remove the selected products from your system.

Note: The deinstallation process removes all of the files and directories under ORAWEB_HOME except for the configuration files (.app, .cfg). This is to facilitate future reinstallation or partial reinstallations.

CHAPTER

4 Post-Installation and Administration

Starting Web Application Server

Start the Web Request Broker (WRB) and the Administration Listener by using the **owsctl** command. At the DOS prompt, type:

```
owsctl start wrb
```

This starts the WRB. To start the Administration Listener, type the following at the DOS prompt:

```
owsctl start admin
```

To start your listener, type the following at the DOS prompt:

```
owsctl start <listener_name>
```

where `<listener_name>` is the name of your listener. The default listener name that you may have set up during installation is “www”.

Note The WRB **must** be started first, and allowed to come up and stabilize before starting another component.

Warning: Do not use the Control Panel Services to start or stop Web Application Server services.

Note: The **owsstat** utility is now a Windows-based utility and can be found in **NT_X86\CoolStuff\OWSSTAT.**

To verify that a listener is running, at the DOS prompt, type:

```
owsctl status <listener_name>
```

Listeners

The installation process creates two listeners using the values input at install time, the Administration Listener and the Default Web Listener. If the defaults are chosen, the values are as listed below. If other values were input, please note them.

Administration Listener

The Administration Listener is used for Web Application Server configuration and administration tasks.

- Host Name: *hostname.domainname*
- Web Listener Name: *ADMIN*
- Port Number: *8888*
- Username: *admin*

Default Web Listener

One general usage web listener is created by the install. This can be the Oracle listener or a third-party HTTP listener.

- Web Listener Name: *www*
- Port Number: *80*

Database Access Descriptors

Database Access Descriptors (DADs) are not created automatically by the installation. DADs are needed only for database access. See the section "Web-based Administration" on page 4-3 for information on how to set up DADs.

Web-based Administration

Use the Oracle Web-based administration pages to configure and administer Web Application Server.

1. Use your Web browser to navigate to the Web Application Server Administrator Welcome Page at:

   `http://<localhost:port>/`

   where *localhost* is the hostname of the machine on which Oracle Web Application Server is installed, and *port* is the port number you have assigned to the administration web listener (8888 by default).

   A dialog box appears, requesting you to authenticate.

2. Provide your username and password (these settings were determined during the installation process), and click OK.

Note: The username and password are case sensitive.

   The Oracle Web Application Server Administrator Home Page appears.

3. Follow the link to install DADs for database access.

4. Click on Web Application Server Manager icon to begin configuring your Web Application Server.

Web Application Server Control Utility

The **owsctl** utility is used to stop, start, and monitor status of the Web Request Broker, Object Request Broker, and Web listeners. This command is described in detail in Appendix A.

Configuration for Multi-node Install

After installing the files on the primary and remote nodes, you need to configure the Web Application Server to tell it about the remote nodes.

Tasks to Perform at the Primary Node

On the primary node, you need to:

1. Specify the remote node names using the Oracle Web-based administration pages.
   a. Use your Web browser to navigate to the Web Application Server Administrator Welcome Page at:

      `http://<localhost:port>/`

      where *localhost* is the hostname of the machine on which Oracle Web Application Server is installed, and *port* is the port number you have assigned to the administration web listener (8888 by default).

      A dialog box appears, requesting you to authenticate.

   b. Provide your username and password (these settings were determined during the installation process), and click OK.

      The Oracle Web Application Server Administrator Home Page appears.

   c. Click Web Application Server Manager.

   d. Click Oracle Web Application Server.

   e. Click Cartridge Administration.

   f. Click Cartridge Summary (Web Request Broker).

   g. In the left frame, click System.

   h. In the WRB System Parameters form, enter the list of remote host names, separated by spaces. For example, `www.blob.com www.blob2.com`.

   i. Click Modify.

2. Restart the WRB on the primary node by entering:

   ```
   owsctl stop wrb
   owsctl start wrb
   ```

3. Start up the listener processes by typing:

   ```
   owsctl start <listener_name>
   ```

   where *listener_name* specifies the name of the listener.

Tasks to Perform at the Remote Node

1. Install the correct components on the remote nodes.

2. Configure the cartridges that you want to run remotely.

On each of the remote nodes, you need to start up the cartridge factory using this command:

```
owsctl start cartridge
```

The cartridge factory manages the cartridges that run on that machine. It communicates with the Dispatcher (which runs on the primary node), telling it about new cartridge instances that are started up and about unoccupied cartridge instances. The Dispatcher then knows where to direct requests.

Setting Up a Secure Oracle Web Application Server

Refer to the Web Application Server Manager page at:

http://*localhost:port*/ows-adoc/Intro.html

for instructions on generating a certificate request and obtaining a certificate from the certifying authority (CA).

CHAPTER

5 Configuration of Third-party HTTP Servers

Overview

A feature of Oracle Web Application Server Version 3.0.1 is the HTTP daemon adapter which enables the Web Request Broker (WRB) to run seamlessly using third-party HTTP servers (listeners) instead of the Oracle Web Listener shipped with Oracle Web Application Server. Oracle Web Application Server supports:

- Netscape FastTrack 2.0 and 2.01
- Netscape Enterprise Server 2.0 and 2.01
- Microsoft Internet Information Server (IIS) 2.0 and 3.0

Note: This version of Oracle Web Application Server does not support Netscape Enterprise 3.x and Netscape FastTrack 3.x listeners; therefore, the Administration Listener is unable to show their status.

Note: When you register an external listener, such as Netscape or IIS, with Oracle Web Application Server, you cannot configure it using the Admin Web pages. You must hand edit the appropriate configuration files.

Configuring Netscape Using the Oracle Installer

Oracle Installer automatically detects Netscape FastTrack and Enterprise HTTP servers. During installation you can migrate one Netscape HTTP server automatically. To migrate additional servers, or to perform the migration after the Oracle Web Application Server installation, use the Oracle migration utility:

http://*localhost:port*/ows-adoc/migrat.html

Oracle Installer automatically creates registry entries under **HKEY_LOCAL_MACHINE\Software\Oracle** when you upgrade a Netscape HTTP server to use the Oracle WRB:

```
OWS30:<ORANT>\OWS30

OWS_ADdPCFG30:<OWS30>\ADMIN\ADPSPY.CFG
```

The installation automatically backs up original Netscape FastTrack and Enterprise HTTP server settings during the migration. The new settings filename is **obj.conf.sav**, and it is saved to the same directory as the original **obj.conf** file.

Configuring Using the Oracle Migration Utility

For the Netscape server you want to run with Oracle Web Request Broker:

1. Register the Netscape server with the Oracle Web Application Server Manager by using the External Listener Registration utility. The Web Application Server Manager is located at:

   http://*localhost:port*/ows-adoc/Intro.html

2. Using your web browser, navigate to

   http://*localhost:port*/ows-adoc/migrat.html

   Authenticate by providing your username and password which were set during installation.

3. Follow the link to Configure FastTrack/Enterprise Server 2.0 to use Oracle Web Request Broker 3.0.1.

4. Enter the information for each listener you want to migrate.

   - Netscape Server Type
   - NS_HOME (Netscape home directory)
   - Netscape Server Name

5. Click Configure.
6. Make sure the **.app** file mentioned in the **\config\obj.conf** file exists, assuming you are running the Web Request Broker. If it does not exist, use the sample file **serverapp.dfl** to create the **.app** file: **orant\ows30\admin\sv***listener***.app**.
7. Stop and start your Netscape HTTP server.

After configuration, use the **%ORAWEB_HOME%\bin\owsctl** to start up the Netscape server. For example:

```
owsctl start fnsl
```

To stop the server:

```
owsctl stop servername
```

To check the status of the server:

```
owsctl status servername
```

Note: If you use **owsctl** to start the Netscape server, you must register the Netscape server with the Oracle Web Application Server.

Changes Made During Configuration

The **wlmigrat** program does the following:

- Configures the Netscape server configuration file **obj.conf** under:

  %NS_HOME%\httpd-*server***\config** (FastTrack Server), or

  %NS_HOME%\https-*server***\config** (Enterprise)

- The migration tool links these files with Oracle WRB and saves the original files into *filename***.sav**.

The **init** function causes the Netscape server to load the WRB modules and adds to the **obj.conf** and **magnus.conf** files as follows.

```
Init fn=load-modules shlib=%ORAWEB_HOME%\lib \ndwfns30.dll
funcs="oracle-adp-init,oracle-adp-auth-trans, \
oracle-adp-service,oracle-adp-addlog,oracle-adp-error, \
oracle-adp-name-trans,oracle-adp-path-check,oracle-adp- \
object-type"
Init fn=oracle-adp-init adapter=%ORAWEB_HOME%\lib\ndwfn30.dll \
cfgfile="%ORAWEB_HOME%\admin\adpnsapi.cfg" \
serverconf="%NS-HOME%\httpd-[server]\config\obj.conf" \
servererrfile="%NS-HOME%\httpd-[server]\logs\errors" \
ORACLE_HOME="%ORACLE_HOME%" \
ORAWEB_HOME="%ORAWEB_HOME%" \
```

```
userdbdir="%NS-HOME%\userdb" \
mimetypesfiles="%NS-HOME%\http-[server]\config\mime.types"
```

Additional directory mappings for the Netscape server are needed for viewing Web Application Server release 3.0.1 pages. These are added to the default object section of the **obj.conf** file as follows:

```
NameTrans fn="oracle-adp-name-trans" \
NameTrans fn=pfx2dir from=\ows-bin \
dir="%ORAWEB_HOME%\bin" name="cgi"
NameTrans fn=pfx2dir from=\ows-doc \ dir="%ORAWEB_HOME%\doc"
NameTrans fn=pfx2dir from=\ows-img \ dir="%ORAWEB_HOME%\img"

NameTrans fn=pfx2dir from=\tr-img \
dir="%ORAWEB_HOME%\demo\img" \
NameTrans fn=pfx2dir from=\sample\bin \
dir="%ORAWEB_HOME%\sample\bin"name="cgi"
NameTrans fn=pfx2dir from=\sample \
dir="%ORAWEB_HOME%\sample" \
NameTrans fn=pfx2dir from=\oracle \ dir="%ORAWEB_HOME%\doc"
```

The following functions are added to the **obj.conf** default object section:

```
AuthTrans fn="oracle-adp-auth-trans"
PathCheck fn="oracle-adp-path-check"
ObjectType fn="oracle-adp-object-type"
Service fn="oracle-adp-service"
AddLog fn="oracle-adp-addlog"
```

Note: If Netscape FastTrack / Enterprise 2.0 is chosen, **ndwfns20.dll** and **ndwfn20.dll** are used. If FastTrack / Enterprise 2.1 is chosen, **ndwfns201.dll** and **ndwfn201.dll** are used.

The default text / plain type is commented out (with ##) in the default object section because it does not interpret the output of Oracle Web Application Server Java and WRB samples as HTML:

```
##ObjectType fn=force-type type=text/plain
```

Microsoft Internet Information Server

For this release you must use the Oracle Installer to migrate Microsoft IIS. Therefore, Microsoft IIS must be installed before you run the Oracle Web Application Server installation. Choose to configure Microsoft as the general usage listener in the Install and proceed with the Post-installation steps listed below.

Post Installation

To configure the Microsoft Internet Information Server (IIS) to use the Oracle Web Request Broker, follow these steps:

1. Install IIS Internet Service Manager (HTML) before performing any web-based administration for IIS. Install the Internet Service Manager by running IIS setup.
2. Set authentication to BASIC to allow non-Microsoft Web browsers to administer IIS. Administration of IIS is protected by the Windows NT operating system security check. If BASIC authentication is not set, only browsers that support NTLM (Microsoft Internet Explorer) can configure the server.
3. Enable anonymous access to IIS.
4. Configure the Web Request Broker for IIS by going to the Web Request Broker Administration page at

   http://*localhost*:*port*/ows-doc/Intro.html

   where *localhost* is the name of the host machine and *port* is the port on which your administration server is running (default is 8888).
5. Confirm that the following files exist:
   - **%ORAWEB_HOME%\bin\ndwfis30.dll**
   - **%ORAWEB_HOME%\admin\adpiis.cfg**

   If not, there was a problem with the installation. Try re-installing Oracle Web Application Server.
6. Restart Microsoft IIS.

Authentication

When using an IIS listener, it is possible to perform user authentication based on the NT user domain. This is achieved by securing access to the adaptor **.dll** file through the validation of the client username and password against the NT user accounts.

This effectively validates access to Oracle Application Server cartridges and services at the NT listener level, and individual cartridges now cannot be protected with their own specific authentication schemes.

The benefits of this scheme are that access to an entire site encompassing static HTML pages, dynamic content such as CGI scripts and ASP application, and

Oracle Application Server cartridges can all be protected with the same user domain.

Note: If the IIS authentication mechanism is used, all cartridge level authenticating must be turned off.

All requests for Oracle Application Server cartridges and services are routed through the adaptor **.dll** file, and therefore, you can deny / grant authentication based on the user NT domain. You can use either Oracle Application Server or IIS authentication to protect your Oracle Application Server cartridges. Disabling anonymous access causes IIS to attempt to authenticate (based on NT accounts) any request.

To use IIS authentication:

You must disable anonymous access to the "ows" virtual path. To do this, do the following:

- Do not use Oracle Application Server authentication.
- Set the variable OAS_IIS_AUTH_ENABLED to TRUE. This can be set in two places:
  - in the System Environment

    or

  - in the Registry under HKEY_LOCAL_MACHINE > SOFTWARE > ORACLE > OAS40.

To use Oracle Application Server authentication:

- Allow anonymous access to the IIS.
- Either leave the variable OAS_IIS_AUTH_ENABLED unset, or set it to FALSE.
- Configure the Oracle Application Server authentication schemes.

CHAPTER

6 Migrating to Oracle Web Listener

This chapter describes how to migrate from a Netscape server to Oracle Web Application Server.

Requirements

Before migrating a Netscape FastTrack or Enterprise server to Oracle Web Application Server, make sure the system meets the following requirements:

- The Oracle Web Application Server 3.0.1 is installed on the same machine as the Netscape server. The Oracle Web Application Server Administration Server must be running.
- Current Netscape server names do not conflict with names in the Oracle Web Application Server.
- If you want the port number to be re-used by the migrated Oracle Web Application Server make sure the Netscape server is stopped.

Note: Only the first six characters of the Netscape server name are used as the migrated Oracle Web Application Server listener name. The migration does not alter any configuration of the existing Netscape server.

Introduction

During Migration of the Netscape server to Oracle Web Application Server, the Netscape configuration files **obj.conf**, **magnus.conf**, and **mime.types** are read and parsed, then translated into the Oracle Web Application Server listener and WRB configuration files such as **sv***server***.cfg** and **sv***server***.app**. However, some of the Netscape server features are not converted to the Oracle Web Application Server listener.

Netscape Server Information

The Netscape FastTrack server has a directory structure similar to the following:

```
%NS_HOME%\
     bin\
     docs\
     nsapi\
     userdb\
     httpd-<server1>\
          config\
               magnus.conf
               obj.conf
               mime.types
          logs\
               access
               errors
     httpd-<server2>\
          config\
               magnus.conf
               obj.conf
               mime.types
          logs\
               access
          errors
     start-admin*
     stop-admin*
```

For the Netscape Enterprise server, the **httpd-***server* directory is named **https-***server*. Each Netscape server has three important configuration files associated with it, which are stored in the **\config** directory:

- **magnus.conf** - This file contains information that the listener uses for initialization. It contains data such as the port the listener should use, the name of the listener, and the owner of the listener process.

- **obj.conf** - This file provides the object configuration for the Web Application Server. When a request is received, the server uses this file to determine if and how it should service the request. This file contains information like directory mappings and security restrictions.
- **mime.types -** This file provides the server with a mapping from file extensions to the MIME types. Both migration choices (configuring the Netscape server to use the Oracle WRB, and migrating the Netscape server to Oracle Web Application Server) use these configuration files extensively.

Migrating a Netscape Server to Oracle Web Application Server

1. Access the top level of the Migration tool at the following location:

   http://*hostname.domain*:8888/ows-adoc/Migrat.html

2. Select the "MIGRATE the Netscape server to Oracle Web Application Server 3.0.1" option.
3. Provide the following information:
   - Netscape server type (FastTrack, Enterprise)
   - Netscape server home directory (full pathname)
   - the name of the Netscape server to be migrated
   - the port number on which the Netscape server is running
4. Select the "Migrate Netscape server" button. The cgi-bin program **wlmigrat** migrates the Netscape server.

The **wlmigrat** program generates a report showing which Netscape configurations will not be converted and gives an "ok to migrate" prompt. Select "ok to migrate" to migrate the Netscape server to the Oracle server.

You must use the Oracle Manager to start up the newly migrated server, after which you can access your pages as usual. You can access the Oracle Web Application Server page from:

http://hostname.domain:*port*/oracle

Notice that *port* is either the port number of the Netscape server, or the port number you assigned on the Migration page.

Changes Made During Migration

The migration program does the following:

- reads and parses the Netscape **obj.conf**, **magnus.conf**, and **mime.types** files and translates the items that can be migrated into the Oracle Web Application Server configuration
- writes the translated configuration and the default Oracle Web Application Server configuration to the Oracle Web Listener **sv***server***.cfg** file
- registers the newly migrated Oracle Web Application Server with the **owl.cfg** file

Migrating the magnus.conf File

The ServerName directive defines the server host name. For example:

```
ServerName wchan-sun.us.oracle.com
```

It is mapped into the Web Application Server **sv***server***.cfg** MULTIPORT section.

The Port directive defines the TCP port the server listens to. For example:

```
Port 7000
```

It is mapped into the Web Application Server **sv***server***.cfg** MULTIPORT section.

The Security directive tells the server whether encryption is enabled. For example:

```
Security off
```

It is mapped into the Web Application Server **sv***server***.cfg** MULTIPORT section. The Security column of the MULTIPORT section will be SSL if encryption is enabled.

If security is activated, the user must generate a new SSL Key certificate for the newly migrated Oracle Web Application Server. Refer to the Oracle Web Administration Server for setting up SSL for Oracle Web Application Server.

The DNS directive specifies whether DNS lookup is performed on the clients that access the server. For example:

```
DNS off
```

It is mapped into the Web Application Server **sv***server***.cfg** [NetInfo] section. DNS Resolution is **always**, if activated, otherwise DNS Resolution is **never**.

Migrating the obj.conf and mime.types Files

The **obj.conf** file defines how the server should handle the incoming requests for documents and programs. An object contains a name or a pattern match which defines the resources it applies to. For example:

```
<Object ppath=...>
Directives.
</Object>
```

Object also contains a series of directives. For example:

```
Directive fn=function [parameters]
```

The following directives can be converted:

- Init fn=load-types mime-types=xxx

  The value for mime-types= is the file name that contains the MIME type and ENCODING extension mapping, formatted as follows:

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

  euc=x-gzip exts=gz,gzip,gzzipp
  ```

  These MIME type and ENCODING extensions are translated into the **sv***server***.cfg** [MIMETypes] and [Encoding] sections as:

  ```
  [MIMETypes]
  application/octet-stream bin exe

  [Encoding]
  x-gzip gz gzip
  ```

Note: Oracle Web Application Server only allows extensions with less than five characters. Any extensions with more than four characters are not migrated and are reported after the migration is complete.

- NameTrans fn="pfx2dir" from=/img dir=/Netscape/img

  The value for **from=** is the virtual path; the value for **dir=** is the physical path. This directive is mapped into the **sv***server***.cfg** [DirMaps] section as:

  ```
  [DirMaps]
  \Netscape\img\ NR \img\
  ```

- NameTrans fn="pfx2dir" from="\cgibin" dir=\Netscape \
  \cgibin name=cgi

  The value for **from=** is the virtual path and the value for **dir=** is the physical path. The **name=cgi** means this is a cgi-bin directory. This directive is mapped into the **sv***server***.cfg** [DirMaps] section as

```
[DirMaps]
\Netscape\cgibin\ CR \cgibin\
```

- NameTrans fn=document-root root=\Netscape\docs

  The value of **root=** is the document root. This directive is mapped into the **sv***server***.cfg** [DirMaps] section as:

```
[DirMaps]
\Netscape\docs\ NR \
```

- NameTrans fn=unix-home from=\~subdir="public_html,home.html"

  The value of **subdir=** is the user directory if the URL specifies **~***user*. It is mapped into the **sv***server***.cfg** [Server] section as:

```
[Server]
UserDir = public_html
```

Note: Oracle Web Application Server allows only one directory to be specified in the **Userdir** parameter, while Netscape allows multiple directories to be specified in a comma-separated list. Only the first directory specified in the list is used by Oracle Web Application Server.

- ObjectType fn=force-type type=text/html

  The value of **type=** is the default MIME type for the server. It is mapped into the **sv***server***.cfg** [Server] section as:

```
[Server]
DefaultMIMEType=text/html
```

- Service fn=imagemap method=(GET|HEAD) type=magnus-internal/imagemap

  The value of **type=** represents the MIME types used for **imagemap**. It is required to look for its corresponding MIME extension from the file listed in (1) and mapped into **sv***server***.cfg** [Server] section as:

```
[Server]
ImageMap = map
```

Note: Oracle Web Application Server only allows one extension to be specified in the **imageMap** parameter, while Netscape allows multiple extensions for the **imagemap**. Only the *first* extension specified in the list is used by Oracle Web Application Server.

- Service fn=index-simple method=(GET|HEAD) type=magnus-internal/directory

  The value of **fn=** can be **index-simple** or **index-common**. If either appears,

it is mapped into the **sv***server***.cfg** [Server] section as:

```
[Server]
UseDirIndexing = true
```

- PathCheck index-names=index.html,home.html fn=find-index

  The value of **index-names=** is the initial file to be searched if a URL does not specify the file name. It is mapped into the **sv***server***.cfg** [Server] section as:

```
[Server]
InitialFile = index.html
```

Note: Oracle Web Application Server only allows one file to be specified in the InitialFile parameter, while Netscape allows multiple files to be specified in a comma-separated list. Only the first file specified in the list is used by Oracle Web Application Server.

Access Control and Server Side Includes (parse-html)

The Netscape server schema for the access control and Server Side Includes (SSI) is quite different from that of Oracle Web Application Server. Therefore **wlmigrat** does not migrate any Netscape access control information or SSI (**parse-html**) information to Oracle Web Application Server.

Access Control

Netscape keeps all the user and group information in binary format under **%NS_HOME%\authdb\<dbname>***

Oracle Web Application Server keeps all user, group and realm information in the **sv***server***.cfg** file in ASCII format.

The Netscape administration tool generates the ACL command, saves it under **%NS_HOME%\httpacl**, and uses the ACL in **obj.conf** to apply the access control on any files on the server machine. The ACL can be a combination of user, group, IP address, and hostname.

Oracle Web Application Server applies access control on files that exist in its virtual directory map. Its access control can be the combination of realms and / or IP address, or realms and / or hostname.

The user can use Oracle Web Application Server Administration Manager to group all Netscape user and group information into Oracle Application Server's

user, group and realms, and apply the access control on the file, based on the virtual path (but not the physical path).

Server Side Includes (parse-html)

On Netscape, the **parse-html** option means it will parse the HTML files with any SSI before sending them to the client. The **parse-html** option is associated with the optional **opts=no-exec** (which prevents the client from executing anything on the server machine), and can be applied to any files and directories on the server machines, whether the files and directories appear on its directory mapping or not. For example:

```
#
# dirmap
#
...
NameTrans from="\ows\3.0" fn="pfx2dir" dir="\d1\ows\3.0"
...
#
# file with parse-html enabled
#
 ...
<Object ppath="\d1\ows\3.0\docs\sstest.html">
Service fn="parse-html" method="(GET|HEAD)" type="magnus-
internal\parsed-html"
<\Object>
<Object ppath="\d1\noway\sstest.html">
Service fn="parse-html" opts=no-exec method="(GET|HEAD)"
type="magnus-internal\parsed-html"
<\Object>
```

Both **\d1\noway\sstest.html** and **\d1\ows\3.0\docs\sstest.html** are parsed before being sent to the client, although **\d1\noway\sstest.html** does not appear on the directory mapping. In addition, accessing **\d1\noway\sstest.html** does not allow any **exec** on the server machine, but accessing **\d1\ows21\docs\sstest.html** does allow **exec** on the server machine.

Oracle Web Application Server only allows the SSI processing of the files under the directory specified in **wrb.app** [AppDirs] section with use of SSI. For example, in **wrb.app**:

```
[AppDirs]
/ssi SSI \oracle\ows\3.0\sample\ssi
```

All files under **\oracle\ows\3.0\sample\ssi** are processed with the SSI cartridge before being sent to the client. In addition, the global SSI flag **ParseHTMLExtn** must stay TRUE to parse the file with any extension, or you

can set the **ExtensionList = html shtml** to include the extension of files that must be parsed. Similarly, the global SSI flag **EnableExecTag** is global to the whole server and cannot be associated with any particular files. Therefore, the user can move all parsed-html or SSI files into the Oracle Application Server's virtual directory to be processed with SSI. The SSI parameters **ParseHTMLExtn**, **EnableExecTag**, and **ExtensionList** can be configured as the user requires.

The **wlmigrat** tool lists all files and directories with the **parse-html** option set to **on** at the end of migration. The user is encouraged to move them into a directory where they can be processed by the Oracle Web Application Server SSI.

CHAPTER

7

Upgrading from Previous Releases

This chapter provides information for upgrading from a previous version of Oracle Web Application Server. The available upgrades are:

- Upgrading from Oracle Web Application Server 3.0.x
- Upgrade from WebServer 2.x

Upgrading from Oracle Web Application Server 3.0.x

Use the Oracle Installer to upgrade your existing 3.0.x Web Application Server to Release 3.0.1. Choose the following installation option:

- Install / Upgrade Software Only - upgrades product files only from Release 3.0.x to Release 3.0.1, while retaining any existing configuration files.

Warning: Do not select "Add / Upgrade Software". This selection will destroy any existing configuration information and install the complete product using a non-OFA file structure.

Note: If you are upgrading an OFA compliant installation, you must have ORACLE_BASE set. If the ORACLE_BASE variable is not set during an upgrade,

the installer assumes that the existing installation version is non-OFA compliant and proceeds accordingly.

The prompts you see will be similar to those seen in a new installation.

Upgrade from WebServer 2.x

To upgrade from WebServer 2.x to Web Application Server 3.0.1:

1. Install Oracle Web Application Server 3.0.1 on the same machine as Oracle WebServer 2.x. Make sure you do not use the same listener name and DCD name as used in WebServer 2.x.

   Warning: Windows NT does not support multiple Oracle Homes. If you already have an existing Oracle Home, changing it during installation of Release 3.0.1 will disable the Oracle products currently installed on your machine.

2. Reboot your machine.
3. Start the Web Request Broker and the Administration Listener.
4. Launch your Web browser, and navigate to the upgrade page. Detailed instructions on how to access and use the Upgrade Tool follow.

Upgrade Tool

When you upgrade, you will use the following files:

- **%ORAWEB_HOME%\admdoc\Migrat.html**
- **%ORAWEB_HOME%\admdoc\OWS2xto30.html**
- **%ORAWEB_HOME%\admbin\wlupgrade**

Access the Upgrade Page

Make sure that the Oracle Administration Server 3.0.1 is running.

1. Access the upgrade page from the Web Application Server 3.0.1 Administration Server by entering the following location in your Web browser:

   http://*hostname.domain:port*/ows-adoc/Migrat.html

   You may also access this page by clicking on the following links:

a. Click on the Web Application Server Manager icon, and enter your username and password as specified during installation.This brings you to the Web Application Server Administration Home Page.

b. Click on the Oracle Web Application Server icon.

2. Click on the Migration icon.

3. Select the "Upgrade the Oracle WebServer 2.x to Oracle Web Application Server 3.0.1" option. The page prompts you to:

   - upgrade from 2.0 or 2.1
   - enter your ORACLE_HOME

The cgi-bin program, **wlupgrade**, upgrades the server and does the following:

- Lists the available Oracle WebServer 2.x from the ORACLE_HOME. You may then choose to upgrade.

Note: The upgrade utility does not list the Oracle WebServer 1.0 available for upgrade. If you want to upgrade from WebServer 1.0, create a new Oracle Web Listener 3.0.1.

- Upgrades the Oracle Web PL/SQL Agent DCD in WebServer 1.0 and 2.x, and registers them in Web Application Server 3.0.1.

Warning: If you are still using Oracle Web PL/SQL Agent 1.0, you will not be able to access it after you upgrade. Refer to "Upgrade Oracle PL/SQL Agent 1.0" on page 7-9 to make the application available after upgrade.

- Detects the cartridge, application directory mapping, cartridge section, and the application protection that might conflict with that defined in the current Web Application Server 3.0.1 **wrb.app** file.
- If conflicts are found, it lists the conflicts and prompts you to 'keep 2.x parameters' or 'use 3.0.1 parameters'. If there is no conflict, you are prompted with 'ok to upgrade'.
- Merges the Oracle WebServer 2.x **sv***server_name***.app** information into Web Application Server 3.0.1 **wrb.app** file.
- Merges the Oracle WebServer 2.x **sv***server_name***.cfg** security information such as user, group, and realms into the **wrb.app** file.
- Updates the **sv***server_name***.cfg** for Web Application Server 3.0.1.
- Upgrades the Oracle Web Application Server **sv***server***.cfg** file from Oracle 1.2 format to Oracle 2.1 format.
- Continues to upgrade the next chosen Oracle Web Application Server 2.x to be upgraded.

Determine a List of Registered WebServer 2.x Listeners

The following sections describe the upgrade tool in detail. The upgrade tool examines the **%ORACLE_HOME%\ows2\admin\owl.cfg** file or **%ORACLE_HOME%\ows21\admin\owl.cfg** to determine which registered Oracle WebServer is available for upgrade.

Upgrade Oracle Web PL/SQL Agent DCDs

The upgrade tool upgrades DCDs and merges them into the Web Application Server 3.0.1 **wrb.app** file. It shows a list of DCDs that have been successfully upgraded and a list of DCDs that cannot be upgraded because the names have already been used by the Web Application Server 3.0.1.

Oracle WebServer 1.0 and 2.x stores the DCD information in the **%ORACLE_HOME%\ows2\admin\owl.cfg**, and each entry has the following information:

Table 7-1: DCD Information

Entry	Description
owa_service	DCD name
owa_user	database login
owa_password	database user password
oracle_home	database ORACLE_HOME
oracle_sid	sid if using local database
owa_connect_string	SQL*Net v2 connect string if using remote database
owa_valid_ports	ports allow access to database
owa_log_dir	this was obsoleted for the 3.x versions, it was replaced by two parameters: logger_logsys_destdir and logger_logsys_destfname directory that stores database errors
owa_nls_lang	database language type
owa_error_page	html page that shows errors

There are two built-in DCDs, **owa_dba** and **owa_default_service,** in the **owa.cfg**. These DCDs are not upgraded because they are created during installation.

Upgrade and merge the entries into **wrb.app**. The entries are split into two entries that are DAD and PL/SQL descriptors.

In Oracle WebServer 2.x **owa.cfg,** you will find an entry. For example:

```
#
(
owa_service=DOHC_WEB
(
owa_user=www_dba
)
(
owa_password=manager
)
(
oracle_home=\private\home\wchan\work\ows-home
(
)
owa_connect_string=DOHC
)
(
owa_valid_ports=8818 3368
)
(
owa_log_dir=\private\home\wchan\work\ows-home\ows21\log
(
)
owa_nls_lang=AMERICAN_AMERICA.US7ASCII
```

The above entry is updated into **wrb.app** as:

```
[DAD_DOHC_WEB]
username=www_dba
password=manager
oracle_home=private\home\wchan\work\ows-home
connect_string=DOHC
nls_lang=AMERICAN_AMERICA.US7ASCII
[PLSQL_DOHC_WEB]
owa_valid_ports=8818 3368
logger_logsys_destdir=\private\home\wchan\work\ows-
home\log\
owa_dad_name=DOHC_WEB
```

You can refer to the DCD by using DOHC_WEB.

The following table shows the parameter mappings between WebServer 2.x DCD, Web Application Server 3.0.1 DAD, and Web Application Server 3.0.1 PL/SQL.

Table 7-2: Parameter Mapping

OWS 2.x DCD	OWS 3.0.1 DAD_xx	OWS 3.0.1 PL/SQL_xx
owa_service	suffix of [DAD_xx]	suffix of [PL/SQL_xx] section name
owa_user	username	
owa_password	password	
oracle_home	oracle_home	
owa_connect_string	connect_string	
owa_oracle_sid	oracle_sid	
owa_nls_lang	nls_lang	
owa_valid_ports		owa_valid_ports
owa_log_dir		logger_logsys_destdir
owa_error_page		owa_error_page

In order to direct the PL/SQL to use the appropriate DAD, the owa_dad_name=xx parameter is added to the PL/SQL_xx section of the Web Application Server 3.0.1 **wrb.app** file.

Upgrade the Oracle WebServer 2.x sv*server_name*.cfg

The upgrade tool updates the **sv***server_name*.**cfg** file in two steps.

1. The upgrade tool updates the [DynApps] section to Web Application Server 3.0.1.

   In WebServer 2.0, the section is:

   ```
   [DynApps]
   %ORACLE_HOME%\ows2\lib\libndwll.so ndwlld_DynamicInit
   %ORACLE_HOME%\ows2\lib\ndwrd.so sndwrdini_init
   %ORACLE_HOME%\ows2\lib\ndwp.so ndwp_dinit
   ```

   In WebServer 2.1, the section is:

```
[DynApps]
%ORACLE_HOME%\ows21\lib\ndwfss.so oracle_adp_init
```

The upgrade tool will update the entry to:

```
[DynApps]
%ORAWEB_HOME%\lib\libwrl.so oracle_adp_init
```

2. The upgrade tool updates the **sv***server_name*.**cfg** from Oracle 1.2 format to 2.1 format.

Merge WebServer 2.x sv*server_name*.app into Web Application Server 3.0.1 wrb.app

In Web Application Server 3.0.1 there is only one **wrb.app** file that stores the cartridge application related information, as opposed to one **sv***server*.**app** per listener in Oracle WebServer 2.x.

The upgrade tool does not alter default cartridges such as JAVA, Live HTML, PL/SQL and HELLO in the Web Application Server 3.0.1 **wrb.app** file, which allows you to use the Web Application Server 3.0.1 default cartridges after the upgrade.

The upgrade tool updates JAVA settings such as CLASSPATH and LD_LIBRARY_PATH. The JAVA settings CLASSPATH and LD_LIBRARY_PATH are found in WebServer 2.x **sv***server*.**app** and are appended into Web Application Server 3.0.1 **wrb.app**'s JAVA setting. After this, the upgrade tool detects conflicts in:

- user-defined cartridges in the [Apps] section of **sv***server*.**app**
- application directory mapping in the [AppDirs] section of the **sv***server*.**app**
- cartridge section parameters and application protection in the [AppsProtection] section of the **sv***server*.**app**

It also indicates conflicts between the WebServer 2.x **sv***server*.***app*** and Web Application Server 3.0.1 **wrb.app** by showing the conflicts in table format before it merges **sv***server*.**app** into **wrb.app**.

Conflicts occur when cartridges, application directory mapping, section parameters, and application protection defined in Web Application Server 3.0.1 **wrb.app** are already defined in WebServer 2.x **sv***server*.**app** with different values. For example:

In WebServer 2.x **sv***server*.**app**:

```
[Apps]
. .
```

```
MYAPP1 \private\oracle\ows21\myapp1\libmyapp1.so
myapp1entry 0 100
[AppDirs]
\usr\myapp1 MYAPP1 \private\oracle\ows21\myapp1
[MYAPP1]
MyApp1Param1 = True
MyApp1Param2 = False
```

In Web Application Server 3.0.1, **wrb.app**:

```
[Apps]
...
MYAPP1 \usr\hr\myapp1\Myapp1.so Myentry 0 100
...
[AppDirs]
...
\usr\myapp1 MYAPP1 \private\oracle\ows\3.0\myapp1
[MYAPP1]
MyApp1Param1 = False
Param2 = False
```

In this case, the MYAPP1 cartridge is defined in WebServer 2.x **sv***server*.**app** and Web Application Server 3.0.1 **wrb.app**; however, they are different cartridges. The application directory mapping **\usr\myapp1** has a conflict in WebServer 2.x **sv***server*.**app** and Web Application Server 3.0.1 **wrb.app** because they have different physical paths. The cartridge section parameter MyApp1Param1 has a conflict because it has a different value.

When conflicts occur, the upgrade tool prompts you with two choices:

- keep 2.x parameters
- use 3.0.1 parameters

If you choose to keep the WebServer 2.x parameters, the **sv***server*.**app** is used to replace the parameters in Web Application Server 3.0.1 **wrb.app**.

If you choose to use Web Application Server 3.0.1 parameters, conflicting parameters are not merged into Web Application Server 3.0.1 **wrb.app**. In the example shown above, the cartridge MYAPP1 is in conflict, thus the definition for MYAPP1 in WebServer 2.x **sv***server*.**app** is not merged into Web Application Server 3.0.1 **wrb.app**. Its related parameters such as application directory mapping **\usr\myapp1**, and the cartridge section [MYAPP1] are not merged into Web Application Server 3.0.1 **wrb.app** either. Parameters without conflict are upgraded with the tool and merged into the Web Application Server 3.0.1 **wrb.app** file.

In addition, the upgrade tool migrates all the security information from the chosen Web Listener into the **wrb/app** file.

Register WebServer 2.x with Web Application Server 3.0.1

After updating the **sv***server_name*.**cfg** and merging it into Web Application Server 3.0.1 **wrb.app**, the upgrade tool registers the WebServer 2.x, which has been upgraded with Web Application Server 3.0.1. Now you can connect to the Web Application Server 3.0.1 Administration server and start the listener.

Upgrade Oracle PL/SQL Agent 1.0

If you use the Oracle PL/SQL Agent 1.0, the agent application is accessed through the cgi-bin program **owa** under **%ORACLE_HOME%\ows\bin**. The DCD information is derived from the virtual path defined in **sv***lsnr***cfg** file.

For example, the **sv***lsnr***cfg** file directory mapping is in **\private\oracle\ows\bin CR \ows1.** When the URI **\ows1\owa\hr.tree** is requested, the Web Listener looks for DCD **ows1** information and executes the cgi-bin based PL/SQL agent **owa**. Then, **owa** executes the PL/SQL package **hr.tree**.

To abandon the use of Oracle PL/SQL Agent 1.0, so you can use the new PL/SQL cartridge, do the following:

- Comment out the virtual path entry in **sv.***lsnr***.cfg.** For example, in the **sv.***lsnr***.cfg** file, comment out the directory mapping entry that uses the cgi-based PL/SQL Agent; such as:

  \private\oracle\ows\bin CR \ows1

- Add the application virtual path entry to the **wrb.app** file. Use the WRB administration page to add the application path into the application directories section. For example:

  \ows1\owa PLSQL \private\oracle\ows\bin

After completing the above steps, the next time the URI **\ows1\owa \...** is requested, the cartridge version of PL/SQL agent is used.

APPENDIX

A Starting and Stopping the Listener

owsctl Utility Commands

The **owsctl** utility starts, stops, reloads, and displays the status of the Oracle Web Listener, Oracle Web Status Monitor, WRB, ORB and Cartridge processes. You can also use it to display the version number of the cartridge.

Syntax

The syntax has the following options:

```
owsctl [ start | stop | reload | status ] [ listener_name ]

owsctl [ start | stop | status ] [-e]wrb | cartridge |
       -p process_name | ncx }

owsctl [ start | stop| status ] -stat [ listener_name ]

       [-poll<polling period> | -uri <uri> | -timeout <timeout>|

       -action<trigger script> ]
```

Options

start

Starts the process specified by the next argument as follows:

- If *listener_name* is supplied, it starts the named Oracle Web Listener.
- If the Netscape server is registered with Oracle Web Application Server, **owsctl** starts up the Netscape server based on the registration information.
- The ORB processes are **mnaddrsrv, mnrpcnmsrv,** and **mnorbsrv**. Use NCX or ncx to run these processes. Make sure you run the NCX processes before exclusively starting the WRB process.
- If WRB or wrb is supplied, it starts the **mnaddrsrv, mnrpcnmsrv, mnorbsrv, wrbcfg, wrblog, wrbasrv, wrbahsrv, wrbroker, wrbvpm,** and **wrbfac** processes. Always start the WRB process before starting an Oracle Web Listener or Netscape server on any machine.
- If CARTRIDGE or cartridge is supplied, the Cartridge process such as **wrbfac** is started.

stop

Stops the process specified by the next argument as follows:

- If *listener_name* is supplied, it stops the named Oracle Web Listener.
- If the Netscape server is registered with Oracle Web Application Server, **owsctl** stops the Netscape server based on the registration information.
- Uses NCX or ncx to stop the NCX (ORB) processes, such as **mnaddrsrv, mnrpcnmsrv,** and **mnorbsrv**.
- If WRB or wrb is supplied, it stops the WRB processes, such as **mnaddrsrv, mnrpcnmsrv, mnorbsrv, wrbcfg, wrblog, wrbasrv, wrbahsrv, wrbroker, wrbvpm,** and **wrbfac** and child processes forked by **wrbfac**, such as **wrbc**.
- If CARTRIDGE or cartridge is supplied, it stops the Cartridge process, such as **wrbfac,** and child processes such as **wrbc**.

reload

Reloads the Oracle Web Listener configuration. This option is ONLY valid for Oracle Web Listener, not for WRB or cartridge processes or Netscape server.

status

Displays the status of the process specified by the next argument as follows:

- If *listener_name* is supplied, it shows the current status of the named Oracle Web Listener.
- If NCX or ncx is given, it shows the current status of the ORB process.
- If WRB or wrb is supplied, it shows the current status of WRB processes.

- If CARTRIDGE or cartridge is supplied, it shows the current status of cartridge processes, such as **wrbfac**.

-e

This option exclusively starts / stops the WRB processes. When you use this option to start / stop the WRB, the NCX (ORB) processes are not started / stopped.

Environment

To use **owsctl**, you must have the following environment variables set:

- **ORAWEB_HOME** - Absolute path where Oracle Web Application Server is installed.
- **ORAWEB_SITE** - This is the site name for the Oracle Web Application Server.
- **ORACLE_HOME** - ORACLE_HOME is the absolute path where you install Oracle products.

WRB, ORB and Cartridge Processes

The Web Request Broker (WRB), Object Request Broker (ORB), and cartridge processes coordinate the distributed interprocess communications for Oracle Web Application Server 3.0.1.

The ORB processes are: **mnaddrsrv**, **mnrpcnmsrv**, and **mnorbsrv**. Run the ORB processes with NCX or ncx. You should run these processes before you start the WRB processes. To start the ORB processes, use:

```
owsctl start ncx
```

If you want to exclusively start the WRB processes without the ORB processes, use:

```
owsctl start -e wrb
```

The WRB Oracle Web Logger process, **wrblog**, can be started individually and executed with multiple instances.

When a cartridge execution request is entered, **wrbfac** starts one or more child processes, **wrbc**, to execute the cartridge. Start the cartridge process after all WRB processes are started and running on a primary machine.

The cartridges can run on the same primary machine as the WRB processes, or they can run on a remote machine. Multiple cartridge processes can be run on the same machine.

The following examples show how to run the processes:

Example 1: Starting the WRB Process

Start the WRB processes on the primary machine by entering:

```
owsctl start wrb
```

Example 2: Displaying the WRB Status

Show the status of WRB processes on the primary node by entering:

```
owsctl status wrb
```

Example 3: Starting the Listener (Optional)

Start the Oracle Web Listener Admin by entering:

```
owsctl start admin
```

Example 4: Displaying the Listener Status (Optional)

Show the status of the Oracle Web Listener Admin by entering:

```
owsctl status admin
```

Example 5: Starting the Cartridge on a Remote Machine (Optional)

Start the cartridge process on the remote machine by entering:

```
owsctl start cartridge
```

Example 6: Starting the Oracle Web Logger Process (Optional)

Start the Oracle Web Logger process on the primary machine by entering:

```
owsctl start -p wrblog
```

Example 7: Checking the Oracle Web Logger Status (Optional)

Check the status of the Oracle Web Logger by entering:

```
owsctl status -p wrblog
```

Example 8: Starting the Oracle Web Status Monitor for the Listener 'admin'

Start the Oracle Web Status Monitor for the Listener 'admin' by entering:

```
owsctl start -stat admin
```

Example 9: Starting the ORB Processes

Start the ORB processes on the primary machine by entering:

```
owsctl start ncx
```

Example 10: Starting the WRB Processes without the ORB processes:

To exclusively start the WRB processes use:

```
owsctl start -e wrb
```

Files

WRB/Cartridge pid:

%ORAWEB_ADMIN%\ows\%ORAWEB_SITE%\wrb\config\.ows_*machine_name***.pid**

WRB/Cartridge address:

%ORAWEB_ADMIN%\ows\%ORAWEB_SITE%\wrb\config\.omnaddr

Web Listener registration file:

%ORAWEB_ADMIN%\ows\%ORAWEB_SITE%\httpd_*machine_name***\owl.cfg**

WRB process files:

%ORAWEB_HOME%\bin\mnaddrsrv.exe
%ORAWEB_HOME%\bin\mnrpcnmsrv.exe
%ORAWEB_HOME%\bin\mnorbsrv.exe
%ORAWEB_HOME%\bin\wrbcfg.exe
%ORAWEB_HOME%\bin\wrblog.exe
%ORAWEB_HOME%\bin\wrbasrv.exe
%ORAWEB_HOME%\bin\wrbahsrv.exe
%ORAWEB_HOME%\bin\wrbroker.exe
%ORAWEB_HOME%\bin\wrbfac.exe
%ORAWEB_HOME%\bin\wrbvpm.exe
%ORAWEB_HOME%\bin\oraweb.exe
%ORAWEB_HOME%\bin\owstat.exe

Notes:

- Always use **owsctl** to start the WRB processes on the primary machine and then start the Oracle Web Listener.
- If you want to distribute the cartridge over remote machines, use **owsctl** with the **start CARTRIDGE** option.
- If you want to start the individual WRB or cartridge process, you must first invoke **owsctl start wrb** on a primary machine.

APPENDIX

B

Multi-node Configuration

The Oracle Web Application Server is made up of several processes — most of which are defined as services on Windows NT. You can run these processes on different machines on the network. You can do this because the architecture of the Web Application Server is based on CORBA (common object request broker architecture), which is a standard for distributed objects. Two advantages of distributing the processes on different machines are performance and scalability.

Processes

The Web Application Server consists of the following processes and services:

Table B-1: Processes

Type	Process	Services	Runs on	Description
ORB Processes	`mnorbsrv`	`OraMediaNet_mnorbsrv`	Primary	The ORB service
	`mnaddsrv`	`OraMediaNet_mnaddsrv`	Primary	The ORB address server
	`mnrpcmnsrv`	`OraMediaNet_mnrpcmnsrv`	Primary	The RPC name server

Type	Process	Services	Runs on	Description
WRB Processes	wrbcfg	OraWeb_wrbcfg	Primary	The WRB configuration provider
	wrblog	OraWeb_wrblog	Primary	The WRB logger
	wrbasrv wrbahsrv	OraWeb_wrbsrv OraWeb_wrbahsrv	Primary and Remote	The WRB authentication servers
	wrbroker	OraWeb_wrbroker	Primary	The Web Request Broker
	wrbvpm	OraWeb_wrbvpm	Primary	The WRB virtual path manager
	wrbfac	OraWeb_wrbfac	Primary and Remote	The cartridge factory
	wrbc		Primary and Remote	Cartridge processes
Listener Process	oraweb	OracleWWWListen-er30%NAME%	Primary	The web listener process.

Cartridge Instances on Each Node

This section describes when cartridge instances are started up and how the Dispatcher directs requests to them. The step numbers correspond to the numbers in the figure below.

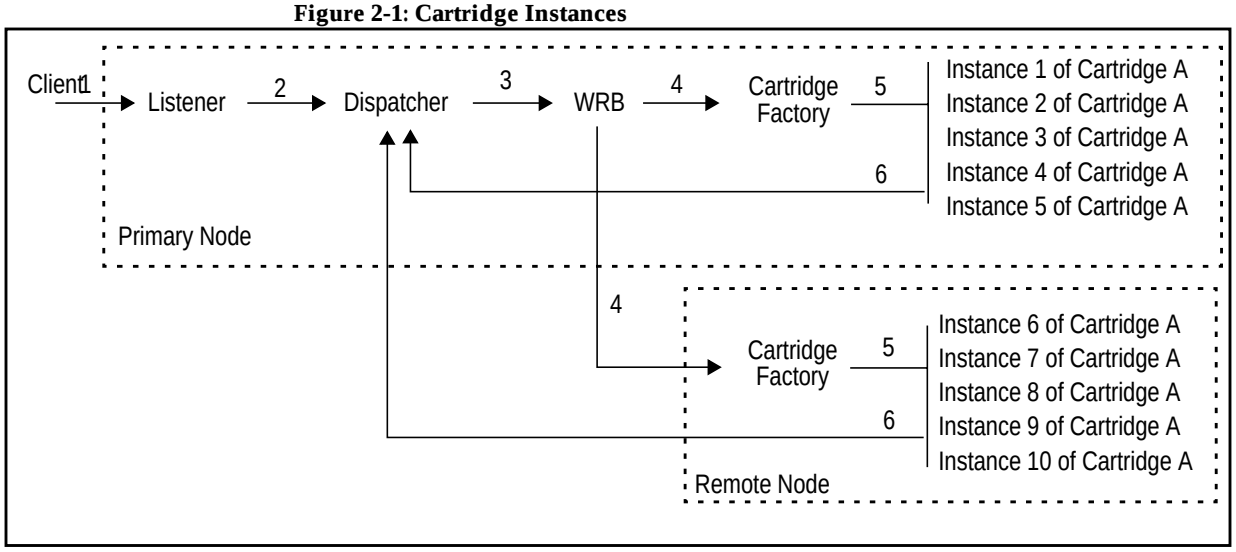
1. A client sends a request for a cartridge to the Listener.
2. The Listener sees that the request is for a cartridge, and sends it to the Dispatcher.
3. If the Dispatcher knows of no free cartridge instances for that cartridge, it sends the request to the Web Request Broker.
4. The WRB then directs one or more cartridge factories to allocate cartridge instances. The WRB tries to ensure that each cartridge factory allocates approximately the same number of cartridge instances for a particular cartridge type for each node.
5. Each cartridge factory spawns the appropriate number of cartridge

instances for its node.

6. The cartridge instances are then registered with the Dispatcher, so that the Dispatcher can direct requests to them.

When subsequent requests for the cartridge come in, the Dispatcher sends the requests to unoccupied cartridge instances.

Figure 2-1: Cartridge Instances



The MIN and MAX configuration parameters for the cartridge type determine the minimum and maximum number of cartridge instances. Note the following points about the meaning of these parameters if you are running in a distributed environment:

- The MIN and MAX values include cartridge instances running on all nodes. For example, if you set a MIN for the PL/SQL Cartridge at 10, then 10 instances can be running on one node or spread over several nodes.
- If you set the MIN value to 0, the cartridge factory spawns one cartridge instance to handle a request, but it does not spawn any extra cartridge instances. In this case, the Dispatcher does not have a list of cartridge instances, and it sends all requests for the cartridge to the WRB. The WRB then directs a cartridge factory to start up a cartridge instance to handle the request.

Distributing the Authentication Server Processes

You can run the Authentication Server on machines other than the primary node, and you can run multiple copies of the Authentication Server. The main reasons for doing this are performance and reliability.

- If the primary node is running many processes and resources on the node are scarce, you might get better performance if you move the Authentication Server to a less busy machine.
- If you are using the Oracle database server to authenticate clients, you can improve performance if you move the Authentication Server to the same machine as the database.
- If you have only one Authentication Server and several clients are requesting authentication, then the requests are queued up. You can improve this bottleneck situation by running multiple copies of the Authentication Server.
- If you run running multiple copies of the Authentication Server on different machines and one machine fails, clients can still access the Authentication Servers that are running on other machines.

The services for the Authentication Server are **ORAWEB_wrbasrv** and **ORAWEB_wrbahsrv**. **ORAWEB_wrbasrv** is the Authentication Broker service, and **ORAWEB_wrbahsrv** is the Authentication Provider service.

Installing the Authentication Server on Remote Nodes

To run the Authentication Server on a remote node, you run **orainst** on the remote node. In **orainst**, select the multi-node option, then select “remote”, and then select “WRB” when prompted on what to install on the remote node.

During installation of the WRB on the remote node, you need to provide the name of the primary node. The name of this node appears in the registry entry **HKEY_LOCAL_MACHINE\Software\Oracle\OMN_ADDR**.

Unlike running cartridges on remote nodes, you do not need to list the names of the remote nodes in the Web Application Server Manager.

Running the Authentication Server

To start up the processes for the Authentication Server:

```
owsctl start -p wrbasrv
owsctl start -p wrbahsrv
```

To stop the processes for the Authentication Server:

```
owsctl stop -p wrbasrv
owsctl stop -p wrbahsrv
```

Troubleshooting and Tips

If requests are not being sent to the remote nodes, consider the following:

- You have to use the same UDP port number on all nodes. This number is saved in the registry entry **HKEY_LOCAL_MACHINE\Software\Oracle\Medianet\OMNADDR**.
- If you are using remote nodes to run the Java or the LiveHTML cartridges, you have to use the same definition of the ORACLE_HOME environment variable for all nodes. This is not required for the PL/SQL cartridge.
- If you are running a custom cartridge that uses the physical path parameter, you need to set the same value for all nodes. If your physical path parameter uses environment variables, you need to check that the environment variables refer to the same physical path.
- Check that you have entered the name of the primary node correctly during installation. The name is not verified until the Web Application Server needs to use it.
- You need to start up the WRB before starting up the Listener or the cartridge processes. To stop these processes, you stop them in the reverse order.

Index

V

W