

# Oracle8 ConText® Cartridge Workbench

User's Guide

Release 2.3

December 1997

Part No. A57700-01

---

Oracle8 ConText Cartridge Workbench User's Guide

Part No. A57700-01

Release 2.3

Copyright © 1997, Oracle Corporation. All rights reserved.

Primary Authors: Graham R. Hoare, D. Yitzik Brenman

Contributors: Dave Allewell, John Barry, Roy Clarke, Franco Cravero, Colin McGregor

**The programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be licensee's responsibility to take all appropriate fail-safe, back up, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and Oracle disclaims liability for any damages caused by such use of the Programs.**

This Program contains proprietary information of Oracle Corporation; it is provided under a license agreement containing restrictions on use and disclosure and is also protected by copyright patent and other intellectual property law. Reverse engineering of the software is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. Oracle Corporation does not warrant that this document is error free.

If this Program is delivered to a U.S. Government Agency of the Department of Defense, then it is delivered with Restricted Rights and the following legend is applicable:

**Restricted Rights Legend** Programs delivered subject to the DOD FAR Supplement are 'commercial computer software' and use, duplication and disclosure of the Programs shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement. Otherwise, Programs delivered subject to the Federal Acquisition Regulations are 'restricted computer software' and use, duplication and disclosure of the Programs shall be subject to the restrictions in FAR 52.227-14, Rights in Data -- General, including Alternate III (June 1987). Oracle Corporation, 500 Oracle Parkway, Redwood City, CA 94065.

Oracle, SQL\*Net, SQL\*Plus, and ConText are registered trademarks of Oracle Corporation. Oracle8, Oracle Forms, Oracle Server, PL/SQL, and Gist are trademarks of Oracle Corporation.

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

---

---

# Contents

<b>Send Us Your Comments .....</b>	<b>vii</b>
<b>Preface.....</b>	<b>ix</b>
<b>1 Introduction</b>	
<b>ConText Workbench Components.....</b>	<b>1-2</b>
<b>Administration Tools .....</b>	<b>1-3</b>
Configuration Manager .....	1-3
System Administration Tool .....	1-3
<b>Document Viewers .....</b>	<b>1-4</b>
Viewer Plugin.....	1-4
Viewer Cartridge .....	1-4
Viewer Control.....	1-5
<b>I/O Utility .....</b>	<b>1-6</b>
<b>CTXQUERY Forms Sample Application .....</b>	<b>1-7</b>
<b>TextServer3 Dictionary Migration Tool .....</b>	<b>1-8</b>
<b>2 Document Viewers</b>	
<b>Using the Viewer Control.....</b>	<b>2-2</b>
Online Help for the Viewer Control .....	2-2
Registering the Viewer Control .....	2-2
<b>Using the Viewer Plugin .....</b>	<b>2-3</b>
Installing the Viewer Plugin File .....	2-3
Activating the Viewer Plugin .....	2-3

<b>Using the Viewer Cartridge .....</b>	<b>2-4</b>
Pre-Installation Requirements for the Viewer Cartridge.....	2-4
Installing the Viewer Cartridge .....	2-5
Specifying URLs.....	2-5
Using URLs.....	2-6
Security Mechanisms.....	2-8
Viewer Cartridge Configuration Files .....	2-9
Using an Alternative Oracle JDBC Driver.....	2-11

### **3 Input/Output (I/O) Utility**

<b>Overview of the I/O Utility .....</b>	<b>3-2</b>
<b>Command-Line Syntax .....</b>	<b>3-3</b>
<b>Parameter File Structure .....</b>	<b>3-4</b>
<b>Examples .....</b>	<b>3-10</b>
LoadDir Example.....	3-10
GeneratedFile Example.....	3-10
Simple Parameter File Example.....	3-10
Advanced Parameter File Example.....	3-15
<b>The I/O Utility dll Interface .....</b>	<b>3-19</b>

### **4 CTXQUERY Forms Sample Application**

<b>Overview of CTXQUERY .....</b>	<b>4-2</b>
Concepts.....	4-2
Layout.....	4-2
<b>Setting Up the Sample .....</b>	<b>4-3</b>
Demonstration Installation Scripts.....	4-3
Oracle Forms Installation.....	4-4
<b>CTXQUERY Architecture .....</b>	<b>4-5</b>
Startup.....	4-5
Enter the query screen.....	4-6
Start the pre-query trigger.....	4-6
Build the query expression.....	4-7
Display an article in the view block .....	4-9
Invoke the HIGHLIGHT procedure.....	4-10
Perform DML Operations.....	4-12

Cleanup .....	4-13
---------------	------

## 5 TextServer3 Dictionary Migration Tool

<b>Overview of the TextServer3 Dictionary Migration Tool .....</b>	<b>5-2</b>
What is the Migration Tool?.....	5-2
Who Performs a TextServer3 Migration.....	5-2
<b>Overview of the TextServer3 Migration Process .....</b>	<b>5-3</b>
<b>Requirements and Limitations.....</b>	<b>5-4</b>
SQEs.....	5-4
Thesaurus.....	5-5
Section Groups.....	5-5
<b>Installing the TextServer3 Dictionary Migration Tool .....</b>	<b>5-6</b>
Pre-Installation Tasks.....	5-6
Installing the Migration Tool .....	5-7
<b>Text Table Migration .....</b>	<b>5-8</b>
Preferences and Policies .....	5-8
How Attributes are Determined for Preferences .....	5-8
Generating Scripts For Column Policies .....	5-10
Editing the Generated Scripts .....	5-10
Column Policy Script Example.....	5-11
<b>Thesaurus Migration.....</b>	<b>5-12</b>
<b>Section Group Migration .....</b>	<b>5-13</b>
<b>CTXWMG_MIGRATE: TextServer3/ConText Migration Tool Package .....</b>	<b>5-14</b>
MIGRATE .....	5-15
MIGRATE_ALL_SYNS .....	5-16
MIGRATE_ALL_THES .....	5-17
MIGRATE_SG .....	5-18
MIGRATE_SYN .....	5-19
MIGRATE_THES .....	5-20
<b>Example Migration Using Supplied Script .....</b>	<b>5-22</b>
Example Description.....	5-22
Using the Example.....	5-22
Example Results.....	5-23

## **A Viewer Cartridge: Manual Configuration**

<b>Notational Conventions for Configuration/Deconfiguration</b> .....	A-2
<b>Configuring Oracle WebServer 2.1 for the Viewer Cartridge</b> .....	A-3
Configuration Requirements .....	A-3
Configuration Tasks .....	A-4
<b>Deconfiguring Oracle WebServer 2.1</b> .....	A-10
Deconfiguration Requirements .....	A-10
Deconfiguration Tasks .....	A-10
<b>Configuring Oracle Web Application Server 3.0 for the Viewer Cartridge</b> .....	A-14
Configuration Requirements .....	A-14
Configuration Tasks .....	A-15
<b>Deconfiguring Oracle Web Application Server 3.0</b> .....	A-22
Deconfiguration Requirements .....	A-22
Deconfiguration Tasks .....	A-22

## **B Configuration Manager: Manual Installation and Configuration**

<b>Manual Installation of Database Packages</b> .....	B-2
Location of Installation Files (Local Installation) .....	B-2
Location of Installation Files (Remote Installation) .....	B-2
Installing the Database Packages .....	B-2
<b>Notational Conventions for Configuration/Deconfiguration</b> .....	B-3
<b>Configuring Oracle WebServer 2.1 for the Configuration Manager</b> .....	B-4
Configuration Requirements .....	B-4
Configuration Tasks .....	B-5
<b>Deconfiguring Oracle WebServer 2.1</b> .....	B-10
Deconfiguration Requirements .....	B-10
Deconfiguration Tasks .....	B-11
<b>Configuring Oracle Web Application Server 3.0 for the Configuration Manager</b> .....	B-14
Configuration Requirements .....	B-14
Configuration Tasks .....	B-15
<b>Deconfiguring Oracle Web Application Server 3.0</b> .....	B-22
Deconfiguration Requirements .....	B-22
Deconfiguration Tasks .....	B-22

## **Index**

---

---

# Send Us Your Comments

**Oracle8 ConText Cartridge Workbench User's Guide, Release 2.3**

**Part No. A57700-01**

Oracle Corporation welcomes your comments and suggestions on the quality and usefulness of this publication. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most about this manual?

If you find any errors or have any other suggestions for improvement, please indicate the chapter, section, and page number (if available). You can send comments to us in the following ways:

- FAX - (650) 506-7200. Attn: ConText Documentation Manager
- postal service:  
Oracle Corporation  
Attn: ConText Documentation Manager  
500 Oracle Parkway  
Redwood Shores, CA 94065  
USA

If you would like a reply, please give your name, address, and telephone number below.

---

---

---





---

# Preface

This manual introduces the Oracle8 ConText Cartridge Workbench components and provides a description of each, as well as any setup and usage information that is needed for each component.

## Audience

This manual is intended for system administrators responsible for installing and maintaining the ConText Workbench Release 2.3.6 components. It is also intended for ConText administrators and application developers who use the ConText Workbench components to administer ConText or develop applications.

## Prerequisites

This document assumes that you have experience with the Oracle relational database management system, SQL, SQL\*Plus, PL/SQL, and with Web-based software such as Oracle Web Request Broker and Oracle Web Application Server. See the other documentation provided with your hardware and software for additional information.

## Related Publications

For more information about ConText, see:

- *Oracle8 ConText Cartridge QuickStart*
- *Oracle8 ConText Cartridge Application Developer's Guide*
- *Oracle8 ConText Cartridge Administrator's Guide*
- *Oracle8 Error Messages*

For more information about the Oracle8 server, see:

- *Oracle8 Server Concepts*
- *Oracle8 Server Administrator's Guide*
- *Oracle8 Server Utilities*
- *Oracle8 Server Tuning*
- *Oracle8 Server SQL Reference*
- *Oracle8 Server Reference Manual*
- *Oracle8 Server Application Developer's Guide*

For more information about PL/SQL, see:

- *PL/SQL User's Guide and Reference*

# How This Manual Is Organized

The manual is divided into five chapters and two appendices:

## Chapter 1: Introduction

This chapter introduces the ConText Workbench components and provides an overview of their features.

## Chapter 2: Document Viewers

This chapter describes the viewers that are contained in the ConText Workbench, and shows how they are used.

## Chapter 3: Input/Output (I/O) Utility

This chapter describes the I/O utility and provides guidance for its use.

## Chapter 4: CTXQUERY Forms Sample Application

This chapter presents the CTXQUERY Oracle Forms sample application.

## Chapter 5: TextServer3 Dictionary Migration Tool

This chapter describes the TextServer3 Dictionary Migration Tool and provides guidance for its use.

## Appendix A: Viewer Cartridge: Manual Configuration

This appendix details the manual configuration and deconfiguration procedures for the Viewer Cartridge.

## Appendix B: Configuration Manager: Manual Installation and Configuration

This appendix details the manual configuration and deconfiguration procedures for the Configuration Manager.

## Type Conventions

This manual adheres to the following type conventions:

Type	Meaning
UPPERCASE	Uppercase letters indicate Oracle commands, standard database objects and constants, and standard Oracle PL/SQL procedures.
<i>lowercase italics</i>	Italics indicate variable names, names of user objects (tables, views, preferences, policies, etc.), PL/SQL parameter/argument names, and names of example PL/SQL procedures.  Italics also indicate emphasis.
<code>monospace</code>	Monospace type indicates example SQL*Plus commands and example PL/SQL code. Type in the command or code exactly as it appears.

## Customer Support

You can reach Oracle Worldwide Customer Support 24 hours a day.

In the USA: **1.415.506.1500**

In Europe: + **44.344.860.160**

Please be prepared to supply the following information:

- your CSI number (This helps Oracle Corporation track problems for each customer)
- the release numbers of the Oracle Server and associated products
- the operating system name and version number
- details of error numbers and descriptions (Write down the exact errors you encounter)
- a description of the problem
- a description of the changes made to the system

## **Your Comments Are Welcome**

Please use the Reader's Comment Form at the back of this document to convey your comments to us. You can also contact us at:

Documentation Manager  
ConText Group  
Server Technologies  
Oracle Corporation  
500 Oracle Parkway  
Redwood Shores, California 94065  
Phone: 1.415.506.7000 FAX: 1.415.506.7360



---

# Introduction

This chapter introduces the Oracle8 ConText Cartridge Workbench components.

The topics covered in this chapter are:

- ConText Workbench Components
- Administration Tools
- Document Viewers
- I/O Utility
- CTXQUERY Forms Sample Application
- TextServer3 Dictionary Migration Tool

## ConText Workbench Components

The ConText Workbench consists of the following components:

- Administration Tools:
  - Configuration Manager (platform-independent)
  - System Administration Tool (Windows NT and 95 only)
- Document Viewers:
  - Viewer Plugin, in combination with Viewer Cartridge (Web-based)
  - Viewer Control (client/server-based)
- I/O Utility (Windows NT and 95 only)
- Samples
  - CTXQUERY Sample Forms application
  - Sample I/O Utility parameter file
- TextServer3 Dictionary Migration Tool



## Administration Tools

The ConText Workbench provides two separate tools for administering ConText

**See Also:** For more information about ConText administration, see *Oracle8 ConText Cartridge Administrator's Guide*

### Configuration Manager

The Configuration Manager is a Web-based application that allows a ConText administrator to manage various administration tasks quickly and easily. It also incorporates a simple ad-hoc query tool.

The Configuration Manager is platform-independent, whereas the System Administration Tool is for Windows NT and Windows 95 platforms only. Also, in contrast to the System Administration Tool, which may be installed on multiple client machines, there is one installation of the Configuration Manager per database; each installation runs under the CTXSYS user.

The Configuration Manager includes a generic PL/SQL package for processing HTML pages extended with special tags that allow transparent interaction with a database.

An HTML-based help system is delivered with the Configuration Manager. This incorporates full text indexing facilities that enable you to search the help system; therefore, usage information for the Configuration Manager is not included in this user's guide.

### System Administration Tool

The System Administration Tool is a client-based Visual Basic application that provides a graphical user interface (GUI) for administering ConText servers, text, and the Linguistic Services.

The System Administration Tool can be installed on any PC running Microsoft Windows NT or Windows 95.

A Windows-based help system is delivered with the System Administration Tool; therefore, usage information for the tool is not included in this user's guide.

## Document Viewers

The ConText Workbench provides two methods for viewing text retrieved by ConText:

- Viewer Plugin (in combination with WebServer-side Viewer Cartridge)
- Viewer Control

### Viewer Plugin

The client-side Viewer Plugin can display most documents on the World Wide Web just as they would appear in their native format.

The Viewer Plugin actually consists of a set of libraries. These provide the means for viewing the file. The plugin currently supports WYSIWYG viewing and query term highlighting in the following formats:

- WordPerfect 5.x, 6.x
- MS Word 2, 6

All other formats supported by ConText are displayed, by default, as plain (ASCII) text.

In the Configuration Manager, the View Data form uses the Viewer Plugin to view a hit from the available query hitlist.

**See Also:** For more information about using the Viewer Plugin, see Chapter 2, “Document Viewers”.

### Viewer Cartridge

The Viewer Plugin requires the Viewer Cartridge. The Viewer Cartridge is installed separately on the WebServer-side. It is used to package data for the Viewer Plugin; that is, it generates the highlight information for a document, fetches the document from the ConText database, and then sends the document as a file (Mime-type “application/x-ctxv”) to the Web browser, which uses the Viewer Plugin (if available) to display the document.

Alternatively, for other Mime-types, the Web browser can use its own mapping to invoke a helper application or plugin which can display the ConText-indexed documents. In such cases, Viewer Cartridge does not need the Viewer Plugin as long as suitable URLs have been specified.

**See Also:** For more information about using the Viewer Cartridge, see Chapter 2, “Document Viewers”.

For more information about manually configuring the Viewer Cartridge, see Appendix A, “Viewer Cartridge: Manual Configuration”.

## Viewer Control

The Viewer Control is a 32-bit Windows custom control for use in client/server configurations. It can be embedded in Windows applications or in HTML pages (for display in Internet Explorer using the Object tag).

The Viewer Control is used to display retrieved documents using the intended layout and with search terms highlighted. It is delivered with a sample application implemented as a stand-alone container.

A viewed document can be in any of the supported server-side formats. For example, a Word document can be viewed in a way similar to how it appears within Microsoft Word. The user can scroll through the document using the Next and Previous buttons to jump to other occurrences of the search term(s).

A Windows-based help system is delivered with the Viewer Control; therefore, usage information for the viewer is not included in this user’s guide.

**See Also:** For more information about using the Viewer Control, see Chapter 2, “Document Viewers”.

For more information about document viewing, see *Oracle8 Con-Text Cartridge Application Developer’s Guide*

## I/O Utility

The I/O Utility enables you to load and unload data between database tables and client-side files.

The I/O Utility is a 32-bit, stand-alone executable that performs a series of operations carried out in sequence as defined by a parameter file.

Example parameter files are documented in this user's guide; in addition, one of the example parameter files is included in the ConText Workbench as a sample.

**See Also:** For more information about using the I/O Utility, see Chapter 3, "Input/Output (I/O) Utility".

## CTXQUERY Forms Sample Application

The CTXQUERY sample application is delivered with the ConText Workbench. This Oracle Forms 4.5 application demonstrates one way of developing Forms applications using ConText.

---

---

**Note:** An icon is not created for CTXQUERY during installation of the ConText Workbench because the sample application requires some setup tasks to be performed on the server before the application can be used.

---

---

**See Also:** For a complete description of CTXQUERY, including setup information, see Chapter 4, “CTXQUERY Forms Sample Application”.

For more information about developing ConText applications, see *Oracle8 ConText Cartridge Application Developer's Guide*

## TextServer3 Dictionary Migration Tool

The TextServer3 Dictionary Migration tool has been developed to assist users who are migrating from TextServer3 to ConText. It is not a complete migration tool, but it does allow you to:

- create a policy in ConText for each indexable column of a TextServer3 table
- migrate TextServer3 thesauri and synonym rings to ConText
- create a section group in ConText for each section group in TextServer3

The tool is delivered as a PL/SQL package called CTXWMG\_MIGRATE. The package generates SQL scripts that you can customize to your requirements before running them.

This package must be created in the ConText database on the server after the ConText Workbench is installed on a client machine.

---

**Note:** The TextServer3 Dictionary Migration Tool is not listed, in the Oracle Installer, under the list of available components for the ConText Workbench.

The scripts for creating the CTXWMG\_MIGRATE package are automatically copied to the client machine during installation of the ConText Workbench. The scripts must then be run after installation to create the package.

---

**See Also:** For a complete description of the Migration Tool, see Chapter 5, “TextServer3 Dictionary Migration Tool”.

For more information about policies, thesauri, and section groups in ConText, see *Oracle8 ConText Cartridge Administrator's Guide*

---

## Document Viewers

This chapter introduces and describes the document viewers provided in the Con-Text Workbench. In particular, this chapter provides installation and usage information for the Viewer Plugin and Viewer Cartridge. Ensure you read the section entitled “Document Viewers” in Chapter 1, “Introduction” which briefly explains the purpose of the various viewers.

The topics covered in this chapter are:

- Using the Viewer Control
- Using the Viewer Plugin
- Using the Viewer Cartridge

## Using the Viewer Control

This section provides usage information for the Viewer Control provided by the ConText Workbench.

The Viewer Control can be used on Windows NT or 95 in any application environment that supports custom controls.

It can also be used in a Web-based environment:

- as a client/server component on a page as an Active-X control to view a document directly from the viewer's own database connection
- in a helper application to view documents that are provided by means other than via the viewer's own database connection (see "URL as a Helper Application" in this chapter)

## Online Help for the Viewer Control

Full usage information for the Viewer Control is provided online as a WinHelp file.

## Registering the Viewer Control

The Viewer Control should normally be automatically registered for use during installation. If, for some reason, the Viewer Control is not registered, it can be registered manually using the regsvr32 utility.

---

---

**Note:** The Viewer Control is not unregistered during deinstallation. You must manually unregister it using regsvr32 before deinstallation.

For example:

```
regsvr32 /u ctxv32.ocx
```

---

---



## Using the Viewer Plugin

This section provides details on installing the client-side Viewer Plugin, which is used to display documents on the World Wide Web as they would appear in their native format.

However, the Viewer Plugin cannot work without the Viewer Cartridge, which must be installed first. The Viewer Cartridge generates the highlight information for a document, fetches the document from the ConText database, and then sends the document as a file to the Viewer Plugin for display.

Once installed, the Viewer Plugin is automatically invoked by the Web browser when:

- a document with the Mime-type “application/x-ctxv” is delivered to the Web browser via the Viewer Cartridge and Oracle Web Application Server, or
- a local file with the extension “.ctxv” is loaded into the Web browser

---

---

**Note:** Before using the Viewer Plugin, the Viewer Cartridge must be installed, and at least one suitable URL must be specified.

For more information, see “Using the Viewer Cartridge” in this chapter.

---

---

## Installing the Viewer Plugin File

Under automatic installation, the Viewer Plugin file - called *npctxvpi.dll* - is placed in the Web browser 'plugins' directory.

If there is no Web browser installed, you must install it before copying the file manually, as follows:

Copy the Viewer Plugin file (*npctxvpi.dll*) manually from the `<ORACLE_HOME>\ctxw\Viewer\plugin` directory to the Web browser 'plugins' directory.

## Activating the Viewer Plugin

To activate the Viewer Plugin file, you must first close down all open Web sessions (if applicable).

Then, start a new Web browser session to ensure that the file has been registered. You can do this by selecting the **Help -> About Plug-ins** menu item from Netscape.

## Using the Viewer Cartridge

The Viewer Cartridge is installed on Oracle Web Application Server. Its purpose is to deliver ConText-indexed documents, and any other data held in an Oracle database to Web browsers. These documents can then be viewed.

URLs are used to instruct the Viewer Cartridge to view ConText-indexed documents from an Oracle database. When these URLs are specified, there is a Mime-type that determines how the documents will be delivered. With the default Mime-type, “application/x-ctxv”, the Web browser tries to use the Viewer Plugin, if it is available.

URLs can be used to display documents in various ways. Documents can be viewed full-frame, in a separate window, or to predetermined dimensions.

Access to ConText-indexed documents can be restricted through the Oracle Web Application Server by the use of realms.

A Viewer Cartridge configuration file is used to determine how the Viewer Cartridge connects to the database. Multiple configuration files can be used to allow different database connection details to be protected by different realms. This allows varying levels of security for documents.

**See Also:** For more information about URLs in the Viewer Cartridge, see “Specifying URLs” and “Using URLs” in this chapter.

For more information about using realms, see “Security Mechanisms” in this chapter.

For more information about configuration files, see “Viewer Cartridge Configuration Files” in this chapter

## Pre-Installation Requirements for the Viewer Cartridge

First, ensure the WebServer version to be installed against is 2.1.0.3.2 or higher.

Then, if necessary, install an Oracle JDBC driver (version 7.3.3.1.3 or later) and note the connection requirements. These are used when writing additional configuration file entries.

To obtain the JDBC driver, navigate to [www.oracle.com](http://www.oracle.com) and look for the free software page. Choose the Oracle JDBC option. Or you can navigate directly to the page; currently it is:

[www.oracle.com/products/free\\_software/](http://www.oracle.com/products/free_software/)

Otherwise, you can use an alternative driver.

**See Also:** For more information about configuration files, see “Viewer Cartridge Configuration Files”.

For more information about alternative JDBC drivers, see “Using an Alternative Oracle JDBC Driver”.

## Installing the Viewer Cartridge

This section provides details for installing the Viewer Cartridge to a remote machine.

On installation, the middle-tier cartridge files are placed in the following directory:

```
<ORACLE_HOME>\ctxw\middle\ctxvcart
```

When installing the Viewer Cartridge to a remote machine, you need to copy or move all files from the locally installed directory `<ORACLE_HOME>\ctxw\middle\ctxvcart` to a directory `<ORACLE_HOME>/ctxw/middle/ctxvcart` on the target machine.

The “install” subdirectory is not required on the remote machine.

---

**Note:** If the target machine is not running Windows NT, all file and directory names *must* be in lowercase.

---

## Specifying URLs

Once the Viewer Plugin and Viewer Cartridge have been installed, the next step is to specify URLs and embed them into HTML pages.

These URLs are required to be able to view ConText-indexed documents from an Oracle database through the Viewer Cartridge. They contain sufficient information for Web Request Broker to direct the request to the Viewer Cartridge. This enables the Viewer Cartridge to generate a database connection, plus parameters that define what the document is and any highlight information needed to service the request.

Parameters to the URL can be encoded within the URL, or supplied through HTML form fields.

To specify URLs:

1. To get a representation of a highlighted document (in WYSIWYG, if supported, or ASCII, if not supported), use the URL parameters to define:

```
indexing policy:      COLSPEC=<policy_name>
```

document selection:     TEXTKEY=<tk\_value>

and, optionally:

query expression:     [QUERYEX=<term\_specification>]

To retrieve the document in its original, unhighlighted form:

type info:             MIMETYPE=<mime\_type\_string>

Then the browser can use its own mapping to invoke a helper application or plugin to handle the data.

With the default Mime-type “application/x-ctxv”, the browser tries to use the Viewer Plugin, if it is available.

**See Also:** For information about installing the Viewer Plugin, see “Using the Viewer Plugin” in this chapter.

2. If the document in its original, unhighlighted form is required (without using ConText), and is stored internally in the database - that is, *not* URL or external data store - it is more efficient to go directly to the base table:

table spec:	TABLE=<table_name>
row selection	WHERE=<where_clause>
col to fetch:	COLUMN=<column_name>
Mime type:	MIMETYPE=<mime_type_string>

The following URL example displays a document from the table field defined by <table\_name>, <column\_name> and <where\_clause> as these elements would appear in a SQL select statement:

```
http://<host>/<virtual_path>/CtxwViewCart?table=<table_name>&column=<column_name>&where=<where_clause>
```

The optional parameter, MIMETYPE, may be used to define the Mime-type of the data. The MIME-type defaults to “application/octet-stream” for binary data and “text/plain” for textual data.

## Using URLs

Once suitable URLs have been specified, they can be used in HTML in various ways:

- URL as a Link Target

- URL as an Embedded Plugin Tag
- URL as a Helper Application

### URL as a Link Target

When you use URLs as a link target, highlighted documents are viewed full-frame if the Viewer Plugin has been installed on the client. The frame is resizable, but plugin parameters cannot be specified.

Documents of other Mime-types (such as jpeg files) are supported. These are downloaded and viewed on the client by the best means available to the browser for that Mime-type.

### URL as an Embedded Plugin Tag

Here, the dimensions of the plugin are pre-determined. So the height and width of the document view are set by the Height and Width EMBED tag parameters.

Plugin parameters can also be specified. A link can be provided to the location of the plugin archive for an 'install-on-demand' mechanism. Other Mime-types are not supported.

### EMBED Tag Parameters

To embed the Viewer Plugin, include the following EMBED tag.

This example displays a document from the table field implied by <policy> and <textkey>, using <query> to highlight hit words:

```
<EMBED width="nnn" height="nnn" src="http://<host>/<virtual_path>/  
CtxwViewCart?colSpec=<policy>&queryEx=<query>&textKey=<textkey>">
```

where *nnn* is a positive number, <host> is the WebServer host and <virtual\_path> is the Viewer Cartridge alias. The *src* attribute value must be on a single line. An optional parameter, *conf*, is used to indicate which section name of the configuration file is used for database connection data. This parameter defaults to use the 'DEFAULT' section.

**See Also:** For more information, see “Viewer Cartridge Configuration Files” in this chapter.

### Plugin Parameters

You can specify the following parameters to the plugin:

```
toolbar=true|false  
silenterrors=true|false  
idleinterval=<n>
```

```
shortcutmenu=true|false  
pageview=true|false
```

If *toolbar* is set to true, the toolbar is shown. The default is true.

If *silenterrors* is set to true, it suppresses error information dialog boxes. The default is false.

*idleinterval* represents a value (in milliseconds) between 1 and 500 that sets the interval between document section reads. The default is 50. A value of 0 means that there is no idle time reading of document sections.

If *shortcutmenu* is set to true, you can click the right mouse button on the viewer window to obtain a short-cut menu. The default is true.

If *pageview* is set to true, the plugin will attempt to represent the document page as the page would appear when printed. The default is false.

### URL as a Helper Application

URLs can be used to invoke a helper application (which must have been defined in the applicable Web browser) to display the document in a separate window as a separate process.

The URL is used as a link URL where a Mime-type is mapped to the ConText Cartridge Viewer Example that uses the Viewer Control. The ConText Cartridge Viewer Example is delivered as part of the ConText Workbench, and is accessible via the icon of that name.

## Security Mechanisms

Access to ConText-indexed documents can be restricted by using realms.

A realm is a group of users and (other) groups assigned by an authentication scheme to regulate access to specific documents and directories through the Oracle Web Application Server.

Authentication schemes allow you to define named groups of user name/password combinations, and named realms that are groups of these groups. You can then assign user, group, and realm names to virtual files and directories, requiring any client requesting access to input one of the specified username/password combinations.

This way, some groups can be granted access to ConText-indexed documents that are held in a variety of databases, while access is denied to other groups.

To restrict access through realms:

1. Map a virtual path to a realm through the Oracle WebServer Administration pages.
2. Map that virtual path to Viewer Cartridge through the WebServer Administration pages.
3. Create the configuration files to which the virtual paths are mapped.

If the virtual path contains more than one element, the last element is used to map onto the name of the configuration file. In this manner, each realm can have a separate configuration file. This ensures that users in groups with access to a realm can be denied access to information available in other realms.

4. Enter the connection details into the configuration file (using the *confURL* parameter) for each database connection that is within that realm.
5. Specify URLs using the appropriate virtual paths.

**See Also:** For more information about using the Oracle WebServer Administration pages, see Appendix A, “Viewer Cartridge: Manual Configuration”, and also the Oracle WebServer documentation.

For more information about the configuration file, see “Viewer Cartridge Configuration Files” in this chapter.

For more information about specifying URLs, see “Using the Viewer Cartridge” in this chapter.

## Viewer Cartridge Configuration Files

A Viewer Cartridge configuration file contains one or more entries that describe how the Viewer Cartridge connects to a database. This description includes the username and password, the database details, and the JDBC driver to be used.

The configuration files must reside on the same tier as the Web Server, in the following directory:

```
$ORACLE_HOME/ctxw/middle/ctxvcart
```

### Multiple Configuration Files

Multiple configuration files can be created to allow different database connection details to be protected by different realms. These files can be used instead of, or in conjunction with, the default file *ctxvcart.cfg*.

Multiple files are created by giving the file the same name as the last element of the virtual path of the Viewer Cartridge, with a .cfg extension.

For example:

Virtual Path	Configuration File
/ctxview	ctxvcart.cfg
/ctxview/secure	secure.cfg
/ctxview/public	public.cfg

**Section Name**

Each section of a configuration file starts with a section name enclosed within square brackets and has a number of parameters associated with the section, as follows:

```
[SECTION_NAME]
username
password
database
driver
subprotocol
```

All lines with any other parameter names are ignored. If one or more parameters of the same name are found within one section, the value associated with last one is used. A value cannot be split over more than one line.

**username**

This parameter specifies the name of the Oracle user that the database connection uses. Defaults to CTXSYS.

**password**

This parameter specifies the password of the Oracle user. Can be null.

**database**

This parameter depends on the value of the *subprotocol* parameter.

For any subprotocol, a full connect descriptor, as used in *tnsnames.ora* files, can be used. For example:

```
database=(DESCRIPTION=(ADDRESS=(COMMUNITY=DECCOM.FIN.HQ.ACME)(PROTOCOL=DECNET)(NODE=NY_VAX.FIN.HQ.ACME)(OBJECT=LSNR))(CONNECT_DATA=(SID=DB1)(GLOBAL_NAME=NY_FIN.FIN.HQ.ACME)))
```



For the oci7 or oci8 subprotocol, a service name present in a *tnsnames.ora* file can be used.

For the thin protocol, a value in the form *<host>:<port>:<sid>* can be used. For example:

```
hq_server:1521:ORCL
```

If *database* is null, it defaults to the installation's default database.

#### **driver**

This parameter specifies the Java class name of the JDBC driver (of version 7.3.3.1.3 or later). It should always be:

```
oracle.jdbc.driver.OracleDriver
```

#### **subprotocol**

This parameter should be assigned one of the following values:

```
oci7 | oci8 | thin
```

If the *subprotocol* parameter name does not appear within the section, a default of oci7 is used.

**See Also:** For more information about the values you can specify for section names, see the documentation for the JDBC driver that you are using.

## **Using an Alternative Oracle JDBC Driver**

To use an alternative Oracle JDBC driver:

1. If, for example, the OCI driver is not available for the Oracle WebServer host platform, you must install into the Oracle WebServer's *<ORACLE\_HOME>*.
2. Modify the Viewer Cartridge CLASSPATH to include a path to (the zip archive containing) the class file root directory.
3. Modify the Viewer Cartridge LD\_LIBRARY\_PATH to include a path to the directory containing any native library object files required by the driver.
4. Modify the configuration file for that CONF section, specifying:

```
database connection string: database=<instance_specification>
java class name:           driver=<driver_class_path>
subprotocol identifier:     subprotocol=<oci7|oci8|thin>
```

**See Also:** For more information about modifying the CLASSPATH and LD\_LIBRARY\_PATH, see Appendix A, “Viewer Cartridge: Manual Configuration”.

For the precise requirements you must specify for your JDBC driver, see the appropriate JDBC driver documentation.

---

## Input/Output (I/O) Utility

This chapter describes how to use the Oracle8 ConText Cartridge Input/Output (I/O) utility. You can use this utility in Windows 32-bit environments such as Windows NT and Windows 95.

The topics covered in this chapter are:

- Overview of the I/O Utility
- Command-Line Syntax
- Parameter File Structure
- Examples
- The I/O Utility dll Interface

## Overview of the I/O Utility

The I/O utility enables you to move text from database to client-side files in a Windows 32-bit environment and vice-versa.

The I/O utility can be used to perform the following operations:

- write data from a (possibly RAW) database field to an operating system file
- read data from an operating system file and update a (possibly RAW) database field
- execute any piece of PL/SQL or SQL that can be executed in an anonymous PL/SQL block
- spawn an operating system command
- load comma-delimited data into the database
- load directory hierarchies into the database
- filter any ASCII text file before loading into the database

These operations are performed in a sequence defined by a parameter file that you specify on the command line.

## Command-Line Syntax

The I/O utility has the following command-line syntax:

```
ctxio32 [-s] [-w] [-p file] [-l file] [-v editor] [-d connect_string]
```

where:

Parameter	Description
-s	Shows login box.  This allows the user to confirm the connect string and the name of the parameter file.
-w	Displays status window.  Messages in the parameter file are displayed in this window during processing.
-p file	Specifies the name of parameter file.  For more information about the format of the file, see “Parameter File Structure” in this chapter.
-l file	Specifies the name of log file.  Error and debug messages, as well as any messages in the parameter file, are output to this file. No messages are reported directly to the user.
-v editor	Displays log file, using given editor (a Windows executable, such as Notepad).
-d connect_string	Specifies an explicit database username/password connection string.  If "-d none" is given, there is no log on and no prompt so no database session is started. This option would normally be used to spawn an operating system command.  If -d is omitted, the user is prompted to enter the connection string.

# Parameter File Structure

The general structure of a parameter file is similar to the tag structure used in HTML/SGML:

```
<START>
many lines of text
<END>
<START>
many lines of text
<END>
```

Start Tag	End Tag	Text	Description
<SQL>	</SQL>	Limited to a maximum of 32k of SQL split over many lines. Keep each line below 2k.  If the first character is @, the SQL is read from the file that follows; for example, @oco.sql loads SQL from the file oco.sql, and executes it.  SQL read from a file must not exceed 5Mb.	Encloses SQL commands.
<GET>	</GET>	Line 1: Destination filename Line 2: Table containing source document Line 3: Column containing source document Line 4: WHERE clause (excluding WHERE keyword)	Writes a document from a database table to an operating system file.
<PUT>	</PUT>	Line 1: Source filename Line 2: Destination table name Line 3: Destination column name Line 4: WHERE clause (excluding WHERE keyword)	Puts an operating system file into a database table.

Start Tag	End Tag	Text	Description
<REG>	</REG>	Reg_key, Reg_value [, Trunc_string]	<p>Registration entries can be used within &lt;EXE&gt; &lt;/EXE&gt; pairs.</p> <p><i>Reg_key</i> is the registration key, as it appears in the lefthand pane of the Registry Editor window.</p> <p><i>Reg_value</i> is the registry entry name as it appears in the righthand pane of the Registry Editor window. This defaults to “(Default)”.</p> <p><i>Trunc_string</i> is optional and causes truncation of the fetched registry value from the given string. Many useful registry entries have '%1' or similar placeholders. See the parameter file examples further on.</p>

Start Tag	End Tag	Text	Description
<LOAD>	</LOAD>	<p>Line 1: Table_name [TAB_OPTIONS]</p> <p>Line2: Column_name[COL_OPTIONS],Co lumn_name[COL_OPTIONS]...</p> <p>Line 3+: Column_data, column_data, ...</p> <p><b>Note:</b> The backslash represents the escape character. Therefore, to obtain \, use \\. Or to obtain ", use \".</p>	<p>Line 1: TAB_OPTIONS := (Replace)</p> <p><i>Replace:</i> Data is updated rather than inserted into the database. This option is used if the row that satisfies the 'where' clause already exists.</p> <p>Line 2: COL_OPTIONS := (COL_OPTION, COL_OPTION...) COL_OPTION := ReadFromFile   Replace "X" "Y" ["C"]   Ignore "X" "Y" ["Z"] ["C"]   ExtractOne "X" "Y" ["Z"] ["C"]   ExtractAll "X" "Y" ["Z"] ["C"]   SaveDir "X"   TempFile "X"   Where   Directory "X"   UseFilter "X"   Sequence n   LoadDir "A" "B" "C" "D" "E" "F"   GeneratedFile</p> <p><i>ReadFromFile</i> Reads the contents of the given file into the database column.</p> <p><i>Replace "X" "Y" ["C"]</i> After reading the file replaces all occurrences of X with Y. There can be more than one Replace call per column. X and Y are case-sensitive if C is omitted or if C equals "N". X and Y are case-insensitive if C equals "Y".</p> <p><i>Ignore "X" "Y" ["Z"] ["C"]</i> After reading the file removes all text between X and Y inclusive of X and Y. If Z is given, remove the text only if Z occurs between X and Y. X, Y and Z are case-sensitive if C is omitted or if C equals "N". X and Y are case-insensitive if C equals "Y".</p>



Start Tag	End Tag	Text	Description
			<i>ExtractOne "X" "Y" ["Z"] ["C"]</i>
			<i>ExtractAll "X" "Y" ["Z"] ["C"]</i>
			After reading the file, extracts all text between X and Y, excluding X and Y. If Z is given, extracts the text only if Z occurs between X and Y. <i>ExtractOne</i> extracts only the first instance of the text; <i>ExtractAll</i> extracts all instances with a carriage return/line feed between instances. X, Y and Z are case-sensitive if C is omitted or if C equals "N". X and Y are case-insensitive if C equals "Y".
			<i>SaveDir "X"</i>
			This parameter copies the post-filtered versions of the input files into directory X. This is useful for checking that search/replace processing has worked as expected.
			<i>TempFile "X"</i>
			Any temporary files are created by default in the home directory; X can be used to specify another file name that is used instead. Typically, this is not required.
			<i>Where</i>
			This defines the columns to be used to define the where clause. This is used to insert the file data into the database and to perform the updates during <i>Replace</i> mode.
			<i>Directory "X"</i>
			The file names given refer to files in this directory. X must end in a double backslash ("\\").
			<i>UseFilter "X"</i>
			For each file, this spawns a user-defined filter program to pre-filter the files before any Extract, Replace or Ignore actions. The program must take an input file as the first argument, and an output file as the second argument. X is the program name (including the full path).

Start Tag	End Tag	Text	Description
			<i>Sequence n</i> This generates a number of the value of this column. This value is initially n (default = 1).
			<i>LoadDir "A" "B" "C" "D" "E" "F"</i> This recursively scans files below directory A (which must have a trailing slash) and looks for files of type B (containing wildcards and separated by ';').  Use C as the directory separator. Prefix all files with D. Recursion ceases when more than E directory levels is reached (a value of 0 means no limit is set). Do not recurse into directories F (where multiple directories are separated by ';').  See the LoadDir Example further on.
			<i>GeneratedFile</i>  This is related to LoadDir. The generated file from LoadDir is used as input for this column.  Line 3: Column_data = Strings, dates and so on should be quoted as if they were entered in an INSERT SQL statement.  <b>Note:</b>  Only ASCII files can be filtered with Replace, Ignore, ExtractOne, ExtractAll.
<EXE>	</EXE>	OS command.	Operating system command. This can be split over many lines. If so, carriage returns are removed and replaced by spaces.
<QU>	</QU>	Line 1: Question Line 2: Yes_label Line 3: No_label	This prompts the user with the given question and resumes execution after the line starting 'Yes_label:' if the user responds 'yes'. If the user responds 'no', execution is continued after the line starting 'No_label:'.  Alternatively, labels may take the form '#<string>'
<GOTO>	label	None.	Resumes execution after the line starting 'label:'

Start Tag	End Tag	Text	Description
<MSG>	end of line	Message	<p>Displays <i>message</i> in log file and, if -w option is chosen, displays in status window.</p> <p><b>Note:</b></p> <p>&lt;MESS&gt; is still supported as an alternative to &lt;MSG&gt;.</p>
<ERROR-CODE>	end of line	Error number	<p>Sets the error code to the given number.</p> <p>If a command fails, and the current error code is non-zero, then that error code is returned by ctxio32.exe. If the -r command line option is used in this case, the error code is also placed in the registry at the following location:</p> <p>"HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\ConText Workbench\ctxio_ret"</p>
<AUTO-COMMIT>	end of line	On   Off	<p>Sets the standard autocommit function. If On, there is a commit after every SQL statement until either it is turned Off, or the end of the session.</p>
#	end of line	None	Parameter file comment - not used.

## Examples

This section provides the following four examples for using the I/O utility:

- LoadDir Example
- GeneratedFile Example
- Simple Parameter File Example
- Advanced Parameter File Example

### LoadDir Example

```
"y:\\webpages\\" "*" .htm *.html" "/" "http://oracle.uk.oracle.com" 20 ""
```

This example loads all .htm and .html files into and below y:\webpages (up to a maximum of 20 levels). The path from y:\webpages\ is then built by using '/' as the separator, and this relative path is then prefixed by 'http://oracle.uk.oracle.com/'.

So, if y:\webpages contains a.htm, subdir\b.html and subdir\c.htm, then the following three strings are loaded into the database:

```
'http://oracle.uk.oracle.com/a.htm'  
'http://oracle.uk.oracle.com/subdir/b.html'  
'http://oracle.uk.oracle.com/subdir/c.htm'
```

### GeneratedFile Example

```
Head(GeneratedFile,ExtractOne "<TITLE>" "</TITLE>" "" "Y")
```

This example loads the file found by LoadDir and extract the text between 'TITLE' and '/TITLE'. Case is ignored because of the "Y" parameter.

### Simple Parameter File Example

The file below is a sample CTXIO32 parameter file. It is installed as part of the ConText Workbench.

The command to execute this parameter file is:

```
ctxio32 -v notepad -s -w -p pfile.txt -l log.txt -d ctxdemo/ctxdemo
```

```
#  
# CTXIO sample script
```

```
# File : pfile.txt

#
# Script split into 2
#
# 1) First section (#ReplaceAll) replaces all employees.
# 2) Second section (#ReplaceSome) replaces some employees.
#
# The <QU> .. </QU> section asks a question and jumps to the relevant tag
#

<QU>
Do you wish to replace ALL employees?
#ReplaceAll
#ReplaceSome
</QU>

#
# ----- This section replaces all current employees -----

#
#ReplaceAll

<MESS>Drop employee table
<SQL>
DROP TABLE CTXIO_EMP
</SQL>

<MESS>Create employee table
<SQL>
CREATE TABLE CTXIO_EMP (EMPNO NUMBER, EMPNAME VARCHAR2(100), RESUME LONG)
</SQL>
<SQL>
alter table ctxio_emp add constraint unique_ctxioemp unique(empno)
</SQL>

<MESS>Truncate employee table
<SQL>
truncate table CTXIO_EMP
</SQL>

<MESS>Loading ALL employees into database...
# Use SaveDir so we can check search and replace has required affect
<LOAD>
```

## Examples

---

```
CTXIO_EMP
EMPNO(WHERE), EMPNAME, RESUME(ReadFromFile, Ignore "<Confidential>" "</Confidential>", Replace
"brewing" "chemistry", Replace "Brewing" "Chemistry", SaveDir "temp")
1,"Joe Bloggs","jbloggs.htm"
2,"Ray Smith","rsmith.htm"
3,"Lisa Turner","lturner.htm"
</LOAD>

# Skip next bit which is for single update only

<GOTO> #Common

#
# ----- This section updates the resumes of some employees -----
#
#ReplaceSome

<MESS>Replacing resumes in database...
# Use SaveDir so we can check search and replace has required affect
<LOAD>
CTXIO_EMP (REPLACE)
EMPNO(WHERE), EMPNAME, RESUME(ReadFromFile, Ignore "<Confidential>" "</Confidential>", Replace
"brewing" "chemistry", Replace "Brewing" "Chemistry", SaveDir "temp")
2,"Ray Smith","rsmith.htm"
</LOAD>

# Load this resume without search and replace
<PUT>
jbloggs.htm
ctxio_emp
resume
EMPNO=1
</PUT>

#
# ----- This section common to both -----
#
#Common

# Check Smith's CV was filtered and stored OK
<GET>
temp\rsmith.htm
ctxio_emp
resume
```

```
EMPNO=2
</GET>

#
# ----- Create OCO index -----
#
<MESS>Dropping policy and index (may produce errors but this is OK)...
<SQL>
begin
ctxsys.ctx_ddl.drop_index('CTXIO_EMPRES');
end;
</SQL>
<SQL>
begin
ctxsys.ctx_ddl.drop_policy('CTXIO_EMPRES');
end;
</SQL>

<MESS>Creating policy on test table and creating index...
<SQL>
begin
ctxsys.ctx_ddl.create_policy(policy_name=>'CTXIO_EMPRES',
colspec=>'CTXIO_EMP.RESUME',
textkey=>'EMPNO');
end;
</SQL>
<SQL>
begin
ctxsys.ctx_ddl.create_index('CTXIO_EMPRES');
end;
</SQL>

#
# ----- Do some OCO querying -----
#

<MESS>Create marked up table
<SQL>
drop table mutab
</SQL>
<SQL>
create table mutab (id number, document long)
</SQL>
```

## Examples

---

```
<MESS>Get highlights...
<SQL>
begin
  ctxsys.ctx_query.highlight(CSPEC=>'CTXIO_EMPRES', TEXTKEY=>2,
    query=>'chemistry,Chemistry', mutab=>'MUTAB', ID=>999,starttag=>'<EM>',
    endtag=>'</EM>');
end;
</SQL>

<MESS>Write marked up text to file...
<GET>
mudoc.htm
mutab
document
id=999
</GET>

# This views the marked-up resume in notepad
#<EXE>
#notepad mudoc.htm
#</EXE>

# This views the HTML resume in Netscape
<EXE>
<REG>HKEY_CLASSES_ROOT\NetscapeMarkup\shell\open\command,, "</REG> file:///mudoc.htm
</EXE>

#Finished
```



## Advanced Parameter File Example

This example demonstrates how to load a website. In particular, it shows how to use the LoadDir option of the I/O utility.

---

---

**Note:** This example is not included as part of the ConText Workbench distribution

---

---

```
<MESS>Drop table
<SQL>
DROP TABLE WEBPAGE
</SQL>
<SQL>
DROP TABLE WEBPAGE_OPTIONS
</SQL>
<MESS>Create tables..
<SQL>
CREATE TABLE WEBPAGE(DOCNO NUMBER, DOC LONG, HEAD VARCHAR2(1000))
</SQL>
<SQL>
CREATE TABLE WEBPAGE_OPTIONS(NAME VARCHAR2(100), VALUE LONG, DES VARCHAR2(1000))
</SQL>
<SQL>
CREATE TABLE WEBPAGE_HL(ID NUMBER, DOCUMENT LONG)
</SQL>
<MESS>Create sequence
<SQL>
CREATE SEQUENCE HL_SEQ
</SQL>
<SQL>
alter table WEBPAGE add constraint unique_WEBPAGE unique(docno)
</SQL>
<MESS>Truncate table
<SQL>
truncate table WEBPAGE
</SQL>
<AUTOCOMMIT> On
<MESS>Scanning and loading roll ...
<LOAD>
WEBPAGE
Docno(Where, Sequence 1), Doc(LoadDir "j:\www\ocowsite\\" "**.htm;*.html" "/" "http://
roll.uk.oracle.com/" 20 "www"), Head(GeneratedFile, ExtractOne "<TITLE>" "</TITLE>" "" "Y")
</LOAD>
```

## Examples

---

```
<MESS>Scanning and loading shake ...
<LOAD>
WEBPAGE
Docno(Where, Sequence 1000), Doc(LoadDir "y:\\webpages\\" "*"*.htm;*.html" "/" "http://
shake.uk.oracle.com/" 20), Head(GeneratedFile, ExtractOne "<TITLE>" "</TITLE>" "" "Y")
</LOAD>
<MESS>Scanning and loading uksn16 ...
<LOAD>
WEBPAGE
Docno(Where, Sequence 2000), Doc(LoadDir "h:\\etc\\httpd\\htdocs\\" "*"*.htm;*.html" "/" "http://
uksn16.uk.oracle.com/" 20), Head(GeneratedFile, ExtractOne "<TITLE>" "</TITLE>" "" "Y")
</LOAD>
<MESS>Dropping policy and index (may produce errors but this is OK)...
<SQL>
begin
ctxsys.ctx_ddl.drop_index('WEBPAGE');
end;
</SQL>
<SQL>
begin
ctxsys.ctx_ddl.drop_policy('WEBPAGE');
end;
</SQL>
<MESS>Creating policy on test table and creating index...
<SQL>
begin
ctxsys.ctx_ddl.create_policy(policy_name=>'WEBPAGE',
colspec=>'WEBPAGE.DOC',
filter_pref => 'CTXSYS.HTML_FILTER',
dstore_pref => 'CTXSYS.DEFAULT_URL',
textkey=>'DOCNO');
end;
</SQL>
<SQL>
begin
ctxsys.ctx_ddl.create_index('WEBPAGE');
end;
</SQL>
<SQL>
TRUNCATE TABLE WEBPAGE_OPTIONS
</SQL>
<MESS>Initializing search page options ...
<LOAD>
WEBPAGE_OPTIONS
NAME, VALUE, DES
```

```

'GenShortTitle','OCO Websearch','Short generic title'
'FrontTitle','Oracle8 ConText Cartridge Web Site Search','Title of front page'
'FrontHeading','<center><h1>Oracle8 ConText Cartridge Web Site Search</h1></center>','Heading of
front page'
'FrontIncludeCount','Y','Include web pages count (Y/N)'
'FrontContentsPreQuery','','Contents of page after count and before query section (max 32k)'
'FrontContentsPostQuery','','Contents of page after query section (max 32k)'
'ResultTitle','Oracle8 ConText Cartridge Search Results','Title of results page'
'ResultHeading','<center><h1>Oracle8 ConText Cartridge Search Results</h1></center>','Heading
of results page'
'ResultShowExpression','Y','Show query expression (Y/N)'
'ResultLabelHeading','Page heading','Label to describe heading'
'ResultLabelScore','Score','Label to describe score'
'ResultLabelPage','Page','Label to describe page'
'ResultHighStart','<font color=red size=+4>','Highlight on'
'ResultHighEnd','</font>','Highlight off'
'ResultTagStart','<OCOH>','Internal use'
'ResultTagEnd','</OCOH>','Internal use'
'ResultContentsPreTable','','Contents of page before the results table (max 32k)'
'ResultContentsPostTable','','Contents of page after the results table (max 32k)'
'ResultHighNextYN','Y','Show the [Next] link after each highlight (Y/N)'
'ResultHighNext','<font size=-1>[Next]</font>','If ResultHighNextYN=Y, this is displayed after
each highlight'
'ResultHighFirst','[First Highlight]','If ResultHighNextYN=Y, this is displayed at start and
end of document'
</LOAD>
<PUT>
frontpre.txt
webpage_options
value
name='FrontContentsPreQuery'
</PUT>
<PUT>
frontpos.txt
webpage_options
value
name='FrontContentsPostQuery'
</PUT>
<PUT>
respre.txt
webpage_options
value
name='ResultContentsPreTable'
</PUT>
<PUT>

```

## Examples

---

```
respos.txt  
webpage_options  
value  
name='ResultContentsPostTable'  
</PUT>
```

## The I/O Utility dll Interface

All the functionality of ctxio is incorporated in a single dll (ctxio32l.dll). Developers can call the C routines within this dll directly.

**See Also:** For more information, see the required header file, ctx-exp.h, which is installed in \$ORACLE\_HOME/ctxw/io/public as part of the ConText Workbench distribution.



---

# CTXQUERY Forms Sample Application

This chapter presents the CTXQUERY sample application, developed with Oracle Forms 4.5, and illustrates one method for developing applications using ConText.

The topics covered in this chapter are:

- Overview of CTXQUERY
- Setting Up the Sample
- CTXQUERY Architecture

## Overview of CTXQUERY

The CTXQUERY sample application is a basic Oracle Forms 4.5 application which provides a graphical interface for performing text queries and DML operations with ConText. It supports all the query expression operators and can provide query term highlighting using asterisks in the specified documents.

---

---

**Note:** The CTXQUERY sample Forms application does not utilize any of the theme querying functionality provided by ConText. However, the sample application could be theme-query enabled for English-language documents simply by creating a theme indexing policy, indexing the column with the policy, and referencing the policy in the query block.

---

---

## Concepts

The ConText query concepts demonstrated in this example are:

- query expression syntax
- boolean and statistical (logical) operators
- expansion operators
- score thresholds
- result limits
- two-step queries
- sharing result tables
- term highlighting

## Layout

The form is laid out in four blocks on three canvases. There is a block and a canvas for each step of the query, hitlist, and view with one extra block for all the buttons. All canvases are displayed in a single window, CTXQUERY.



## Setting Up the Sample

This section provides information about the setup tasks that must be performed before using the sample application:

- Demonstration Installation Scripts
- Oracle Forms Installation

### Demonstration Installation Scripts

Before the sample Forms application can be used, the demonstration installation scripts must be run. The demonstration installation scripts are provided with ConText and are installed automatically on the server machine during ConText installation; however, they must be run manually after ConText is installed.

The demonstration scripts perform the following tasks:

- create the *articles* sample table
- populate the *articles* table with text
- create a text index for the text column in the table

---

**Note:** Because the CTXQUERY sample application requires setup before it can be used, an icon is not created for CTXQUERY during ConText Workbench installation.

Once setup has been performed on the server machine, to start CTXQUERY, navigate to the directory where the `ctxquery.fmb` file is located (usually `CTXW\DEMO` in either `C:\ORANT` or `C:\ORA95`) and double-click on the file.

---

**See Also:** For more information about the location of the demonstration scripts for setting up the sample application, see the Oracle8 Server installation documentation specific to your server operating system.

## Oracle Forms Installation

The sample application requires Oracle Forms (32-bit), version 4.5.7.4.0 or higher, to be installed.

Oracle Forms is not installed automatically during ConText Workbench installation and must be installed manually before using the sample application.

---

---

**Note:** If a version higher than 4.5.6.3.3 is installed on your machine, to use the sample application, the ctxquery.fmb file may have to be regenerated after installation.

---

---

## CTXQUERY Architecture

The CTXQUERY form is divided into eight distinct sections:

- Startup
- Enter the query screen
- Start the pre-query trigger
- Build the query expression
- Display an article in the view block
- Invoke the HIGHLIGHT procedure
- Perform DML Operations
- Cleanup

### Startup

On startup, the When-New-Form-Instance trigger initializes the form.

```
When-New-Form-Instance
set_window_property
  (FORMS_MDI_WINDOW, WINDOW_STATE, MAXIMIZE);
set_window_property('CTXQUERY', WINDOW_STATE, MAXIMIZE);
select query_id.nextval
  into :global.query_id
  from dual;
```

This maximizes the windows for display, then selects a query ID from a sequence.

This form is designed so that it can be used by multiple concurrent users, which means that the results tables are shared. When result tables are shared, a unique query ID separates each user's results from other users. In this form the unique ID is created as an increasing sequence of numbers with the last value stored in a global variable.

## Enter the query screen

The query screen is the first screen of the form displayed.

Each widget is part of the *query* block, which is a non-base-table block. There is one hidden field in this screen, *query\_string*.

When the Query button is pressed, the When-Button-Pressed trigger goes to the *hitlist* block and executes a query:

```
When-Button-Pressed on BUTTONS.QUERY
```

```
go_block('hitlist');
clear_block(no_validate);
execute_query(all_records);
```

The *hitlist* block is based on the *article\_hitlist* view. This view joins the query result table *query\_temp* and the base table *articles* into a hitlist which has both score and article information. Although this block contains items for all the fields in this view, most are hidden. The visible page displays score, author, section, and title.

## Start the pre-query trigger

Before the query on the hitlist block is executed, the *pre-query* trigger fires. This trigger executes the first step of a two-step query, then limits the relational fields to the query criteria:

```
Pre-Query on HITLIST
```

```
build_query_string;

ctx_query.contains('DEMO_POLICY',
                  :query.query_string,
                  'QUERY_TEMP',
                  1,
                  :global.query_id,
                  0,1,null);

--
--limit the relational fields to the query criteria
--

:hitlist.conid := :global.query_id;
:hitlist.author := :query.author;
:hitlist.section := :query.section;
copy(:query.pub_date,'hitlist.pub_date');
```

## Build the query expression

The *build\_query\_string* procedure constructs the query expression by concatenating the term, weight, and operator fields from the form and putting them into the *query\_query\_string* hidden field.

*build\_query\_string* is divided into sections A through E.

### Section A

Section A of the code clears the query string of any previous contents.

### Section B

Section B takes each term from the user input form and builds the query expression as follows:

- attaches expansion operators (*fuzzy*, *stem*, *soundex*) to the front of each specified query term
- encases user-specified query arguments in curly brackets

Query arguments are enclosed in curly brackets so that any special characters within the argument (e.g., the "&" in Q&A) will not be misinterpreted as Con-Text operators

- adds weighting factor

If the user has filled in a weighting factor in the appropriate field of the input screen, this value is appended to the search argument.

- appends any specified logical operators to the end of the query expression

When the concatenation is finished, a sample term looks like this:

Expansion	Search Criteria	Weight	Boolean
!	{cat}	*2	&

**Note:** In CTXQUERY, the user can enter up to three separate query expression. These four steps are repeated to for each query expression.

## Section C

Section C of the code strips off extraneous operators at the end of the final query string since the last term has no boolean operator.

## Section D

Section D adds the score threshold value to the query expression.

## Section E

Section E appends the result limit.

### BUILD\_QUERY\_STRING Procedure - Sample Code

```
PROCEDURE build_query_string IS
BEGIN
  --
  --Section A
  --
    :query.query_string := null;
  --
  --Section B
  --
    if (:query.qterm1 is not null) then
      :query.query_string := :query.query_string ||
        :query.qexp1 ||
        '{ ' || :query.qterm1 || ' } ' ||
        '* ' || to_char(nvl(:query.qwt1, 1)) ||
        substr(:query.qop1,2,1);
    end if;
    if (:query.qterm2 is not null) then
      :query.query_string := :query.query_string ||
        :query.qexp2 ||
        '{ ' || :query.qterm2 || ' } ' ||
        '* ' || to_char(nvl(:query.qwt2, 1)) ||
        substr(:query.qop2,2,1);
    end if;
    if (:query.qterm3 is not null) then
      :query.query_string := :query.query_string ||
        :query.qexp3 ||
        '{ ' || :query.qterm3 || ' } ' ||
        '* ' || to_char(nvl(:query.qwt3, 1));
    end if;
  --
  --Section C
```

```

--
:query.query_string :=
rtrim(:query.query_string, '&|,;-');
--
--Section D
--
    if (:query.qthresh is not null) then
        :query.query_string := '(' || :query.query_string ||
        ')>' || to_char(:query.qthresh);
    end if;
--
--Section E
--
    if (:query.qlimit is not null) then
        :query.query_string := '(' || :query.query_string ||
        '):' || to_char(:query.qlimit);
    end if;
END;

```

## Display an article in the view block

When an article in the hitlist is double-clicked, the When-Mouse-Double-Click trigger on *hitlist.title* displays the article in the VIEW block:

```

When-Mouse-Double-Click on HITLIST.TITLE
GO_BLOCK('VIEW');
EXECUTE_QUERY;

```

In order to display the correct article, the Pre-Query trigger limits the *view* block to display the article highlighted in the *hitlist* block:

```

Pre-Query on VIEW:
:view.article_id := :hitlist.article_id;

```

The *view* block is based on *articles*, and displays the full text of the article.

## Invoke the HIGHLIGHT procedure

When the highlight button on the *view* block is pressed, the When-Button-Pressed trigger on *buttons.highlight* invokes the CTX\_QUERY.HIGHLIGHT procedure.

The *highlight* procedure is divided into sections A and B.

### Section A

Section A call the HIGHLIGHT procedure and performs the following tasks:

- fetches the document
- parses the query
- identifies the matching terms
- inserts results into the *highlight\_temp* table.

The highlight result table (*highlight\_temp*) has the following structure:

Columns	TYPE
ID	NUMBER
OFFSET	NUMBER
LENGTH	NUMBER
STRENGTH	NUMBER

This procedure highlights matching query terms in the text window by surrounding them with <<< and >>>.

It is not possible to do an INSTR for the search terms because of the expansion operators; for example, *go=going=gone* in a stem expansion. Instead, use the highlight procedure in Section A to generate the highlights table.

This table holds the offset and the length of each word to be highlighted.

### Section B

With the highlights generated, the cursor loop in Section B works through each offset, length pair backwards from last offset to first.

The asterisks insert won't change the other offsets. For each offset, length pair, the asterisks are inserted by reassigning the full text to: everything before the term (line 1), then the asterisks, then the term and some more asterisks (line 2) then everything after the term (line 3). Doing this repeatedly highlights all the terms in the document.



**HIGHLIGHT Procedure -- Sample Code**

```

PROCEDURE highlight IS
  cursor highcur is select offset, length
    from highlight_temp
    where id = :global.query_id
    order by offset desc;

  --
  -- Section A
  --
BEGIN
  ctx_query.highlight(
    'DEMO_POLICY',           -- policy name
    to_char(:view.article_id), -- textkey
    :query.query_string,     -- query string
    :global.query_id,        -- query_id
    null,
    null,
    'HIGHLIGHT_TEMP',       -- highlight table
    null,
    null,
    null,
    null);
  --
  -- Section B
  --
  for hc in highcur loop
1.    :view.text := substr(:view.text,1,hc.offset - 1) ||
2.    '***' || substr(:view.text,hc.offset,hc.length) || '***' ||
3.    substr(:view.text,hc.offset+hc.length);
    end loop;
END;

```

## Perform DML Operations

When a document is changed it needs to be re-indexed and new linguistic information needs to be extracted. This can be performed in a table trigger or in the application.

The *view* block contains pre-update and pre-index triggers that perform the necessary re-indexing when there is a change to article data. Both triggers invoke the *reindex\_article* procedure.

The *reindex\_article* procedure performs the following tasks:

- submits the document for text re-index (Section A)
- clears existing linguistic information (Section B)
- extracts new linguistic information (Section C)
- submits the linguistic requests (Section D)

---

**Note:** There is no COMMIT; it is performed by the COMMIT of the change to the VIEW block. In SUBMIT, you must explicitly ask for it not to COMMIT (the FALSE second parameter).

---

**See Also:** For more information about DML and re-indexing, see *Oracle8 ConText Cartridge Administrator's Guide*.

### REINDEX\_ARTICLE Procedure -- Sample Code

```
PROCEDURE reindex_article IS
    handle number;
begin
    #
    # Section A
    #
    ctx_dml.reindex('demo_policy', TO_CHAR(:view.article_id));
    #
    # Section B
    #
    delete from article_themes
    where pk = :view.article_id;

    delete from article_gists
    where pk = :view.article_id;
    #
    # Section C
```

```

#
ctx_ling.REQUEST_themes('demo_policy',
                        TO_CHAR(:view.article_id),
                        'article_themes');
ctx_ling.REQUEST_gist('demo_policy',
                     TO_CHAR(:view.article_id),
                     'article_gists');

#
Section D
#
    handle := ctx_ling.submit(0,FALSE,0);
end;

```

## Cleanup

Before the session is ended, clean up any leftover rows in the hitlist result table (*query\_temp*) and the highlight result table (*highlight\_temp*) tables for the query ID:

```

Post-Form
delete from query_temp
where conid = to_number(:global.query_id);
delete from highlight_temp
where id = to_number(:global.query_id);
standard.commit;

```

The *standard.commit* statement executes an Oracle COMMIT without doing a Forms *commit\_form*.



---

# TextServer3 Dictionary Migration Tool

This chapter describes how to use the TextServer3 Dictionary Migration Tool.

The topics covered in this chapter are:

- Overview of the TextServer3 Dictionary Migration Tool
- Overview of the TextServer3 Migration Process
- Requirements and Limitations
- Installing the TextServer3 Dictionary Migration Tool
- Text Table Migration
- Thesaurus Migration
- Section Group Migration
- CTXWMG\_MIGRATE: TextServer3/ConText Migration Tool Package
- Example Migration Using Supplied Script

## Overview of the TextServer3 Dictionary Migration Tool

The TextServer3 Dictionary Migration Tool is for users who need to migrate from Oracle TextServer3 to Oracle8 ConText Cartridge. The tool will:

- create ConText policies on columns that are currently indexed by TextServer3
- recreate TextServer3 thesauri and synonym rings in ConText
- recreate TextServer3 section groups in ConText

### What is the Migration Tool?

The Migration Tool is a PL/SQL package, CTXWMG\_MIGRATE, that is installed on the same database as ConText. It queries the TextServer3 dictionary to find as much useful information as possible that can be used in the creation of column policies. This is done using a database link to the database where the TextServer3 text dictionary resides. It also uses the database link to access any TextServer3 thesauri or section groups that are to be migrated.

The PL/SQL package that implements the tool makes use of the UTL\_FILE package. This package adds File I/O capabilities to PL/SQL in Oracle8 Server, Release 8.0.

### Who Performs a TextServer3 Migration

The Migration Tool is intended for use by application developers or TextServer3/ConText administrators. It is assumed that the user has an understanding of the following:

- how different settings for a TextServer3 text table affect indexing, filtering and querying
- the content and structure of any TextServer3 thesauri, synonym rings, and section groups to be migrated, and how they are used in the TextServer3 application
- ConText concepts such as policies, preferences, and attributes

Familiarity with the Oracle8 Server IMP and EXP utilities is also assumed.

All references to migration refer to the TextServer3/ConText data dictionary migration process. All references to the Migration Tool refer to the TextServer3/ConText Dictionary Migration Tool.

**See Also:** For more information about policies, preferences, and attributes, see *Oracle8 ConText Cartridge Administrator's Guide*

## Overview of the TextServer3 Migration Process

Although TextServer3 and ConText share similar functionality, there are differences that prevent the automatic migration of text tables, thesauri and section groups. The user or application developer needs to make decisions about preferences and policies using their knowledge about the text application.

For this reason, the Migration Tool does not directly create column policies and thesauri in the ConText data dictionary. Instead, it generates:

- scripts which are run from SQL\*Plus to create column policies and section groups
- load files to be used with ctxload to create thesauri and synonym rings

The generated scripts are edited by the user in order to customize the migration to suit the needs of the application. The user needs to alter the creation of some preferences because of differences in ConText functionality (eg. filters supported internally) or to take advantage of new functionality.

## Requirements and Limitations

The Migration Tool is aimed at migrating from any version of TextServer3 to Oracle8 ConText Cartridge 2.3.6. The database used by each product must be accessible to the user carrying out the migration.

The table that is a TextServer3 text table must exist on the target database (where ConText is installed). It need not be populated, but must have a column that has a unique or primary key constraint on it - this column must be the TextServer3 text-key column. The exp/imp utilities can be used for this purpose, or the TextServer3 database can be upgraded to 8.0.4.

---

**Note:** A separate 7.2 ORACLE\_HOME must be maintained if the user wants to carry on using the TextServer3 application.

---

ConText 2.3.6 must be installed on a 8.0.4 database. The user who uses the Migration Tool to generate scripts and load files need not be the user who ultimately owns the column policies and thesauri. The Migration Tool user must have privileges that allow the creation and dropping of database links for each TextServer3 user whose text tables and thesauri are to be migrated.

## SQEs

TextServer3 has areas of functionality that are available in ConText, but the ConText functionality does not exactly parallel the TextServer3 functionality. One of these areas is Stored Query Expressions (SQEs). In TextServer3, an SQE can be a query against multiple text columns. These text columns do not have to be in the same text table. In addition, the query also supports the full 'SELECT ... FROM ... WHERE ...' clauses in a SQL statement.

The ConText implementation binds an SQE to a single column policy and specifies only the query expression. For these reasons, no attempt is made to migrate TextServer3 SQEs to ConText.



## Thesaurus

Another area of TextServer3 functionality that differs from the ConText implementation is the thesaurus. A term in a TextServer3 thesaurus or synonym ring can be a proper term, a reference to another thesaurus or synonym ring, or a reference to an SQE. In ConText, a term can only be a proper term or phrase. When doing the thesaurus migration, the user has control over what happens when a term is a thesaurus or synonym ring, but no attempt to process a term is made when that term is an SQE.

## Section Groups

If section groups are to be migrated, the section groups must be created in the ConText data dictionary *before* policy creation and indexing is carried out.

## Installing the TextServer3 Dictionary Migration Tool

This section provides details for installing the Migration Tool.

During installation of the ConText Workbench, the installation scripts for the CTXWMG\_MIGRATE package are copied to the client machine and an icon is created for executing the scripts; however the scripts are not executed to create the package.

Installing the Migration Tool involves creating the CTXWMG\_MIGRATE package in the database on the server machine.

### Pre-Installation Tasks

Before installing the Migration Tool, you must perform the following tasks:

1. Choose target directory for generated files.

The Migration Tool, once installed on the ConText database, can generate files for the tables and thesauri that are to be migrated. These files are owned by the owner of the Oracle8 shadow process. Therefore, choose a directory that can be accessed by the Oracle Server user who runs SQL\*Plus and ctxload. This is because the generated files may need to be customized.

2. Set the UTL\_FILE\_DIR initialization parameter.

Accessible directories must be specified in the INIT.ORA file for the 8.0.4 instance. Each accessible directory is specified by the following parameter:

```
UTL_FILE_DIR = <directory name>
UTL_FILE_DIR = <directory name> ...
```

The named director(ies) are then available to the PL/SQL File I/O package.

---

**Note:** On UNIX-based systems, and possibly other operating systems, the shadow process, using the PL/SQL File I/O package, can write to *all* file locations specified by the UTL\_FILE\_DIR parameter. There are *no* user-level file permissions. As a result, the UTL\_FILE\_DIR parameter can override operating system file permissions.

For this reason, create a new directory for the purposes of migration so there is no possibility of existing files being overwritten.

---

3. Set the GLOBAL\_NAMES initialization parameter.

You may require that the names of database links are different from the name of the database where they are linking. To enforce this, add the following parameter to INIT.ORA:

```
GLOBAL_NAMES = FALSE
```

4. Restart the database.

The database instance where ConText is installed must be restarted so that the UTL\_FILE\_DIR and GLOBAL\_NAMES parameters can take effect.

5. Drop and recreate database links.

The Migration Tool uses a database link to access the TextServer3 dictionary. The link is called *ts3db*.

For every TextServer3 user that owns a text table or thesaurus to be migrated, the link must be dropped, and then recreated.

*Creating a Link*

Links should be created by the Oracle user that runs the Migration Tool. From SQL\*Plus, use the following syntax:

```
SQL> create database link ts3db connect to <user>  
2 identified by <pw> using '<connect_string>';
```

*Dropping a Link*

When the database link must be created for another TextServer3 user, drop the existing ts3db link:

```
SQL> drop database link ts3db;
```

## Installing the Migration Tool

To install the CTXWMG\_MIGRATE package, double-click on the TextServer3 Migration Tool Installation icon.

## Text Table Migration

During text table migration, a script is generated that creates, as closely as possible, a policy that reflects an indexable column in a TextServer3 text table. The user must specify a prefix for the naming of the preferences in the policy. If a TextServer3 text table contains multiple indexable columns, then a script is created for each column.

### Preferences and Policies

ConText has a concept of preferences and policies. A preference specifies one of the size indexing options that are necessary for creating a text index for a column:

- data store (where is the text stored?)
- filter (what format is the text in?)
- lexer (how are tokens in the text identified for indexing?)
- engine (how is the text index stored?)
- stoplist (are there words that should not be indexed?)
- wordlist (are there special querying features that you want to use?)

The preferences are grouped into a policy and the policy is assigned to a column. Then, a text index is created for the column policy.

A preference consists of a Tile, which identifies the indexing option for the preference, and a number of attributes for the Tile. The attributes are set for a Tile using the CTX\_DDL.SET\_ATTRIBUTE procedure.

**See Also:** For more information about policies, preferences, and attributes, see *Oracle8 ConText Cartridge Administrator's Guide*

### How Attributes are Determined for Preferences

A brief outline of how the Migration Tool handles each type of preference is given here.

#### Data Store Preference

If the TextServer3 column is EXTERNAL, then the OSFILE Tile is used to create a Data Store preference. If an External File Path for EXTERNAL columns is specified, then the *path* attribute of OSFILE is set appropriately. Otherwise, the predefined CTXSYS.DEFAULT\_DIRECT\_DATASTORE preference is used.

### Filter Preferences

If the TextServer3 text table contains documents of a single format supported internally by ConText, then the FILTER attribute is set appropriately. If multiple formats are used, that are all supported internally by ConText, the predefined CTX-SYS.AUTOB preference is used. However, if unsupported formats are used, the tool generates a commented-out list of CTX\_DDL.SET\_ATTRIBUTE calls, for the *executable* attribute, to match the formats that are recognized by Autorec. The user should then uncomment the appropriate ones and edit the executable name.

### Lexer Preference

The BASIC LEXER Tile is used to create the Lexer preference. If the TextServer3 text column has column-level options defined, then these are used when setting lexer attributes, otherwise table-level options are used. The options map as follows:

TextServer3 Option	BASIC LEXER Tile Attribute(s)
Alpha Join	printjoins
Numeric Join	numjoin, numgroup
Continuation	continuation
case Conversion = BASE	BASE_LETTER set to 1

If a particular TextServer3 option is not set, then the equivalent BASIC LEXER attribute value is not set.

Additionally, if the TextServer3 text column references a section group, then the following two BASIC LEXER Tile attributes are set:

Attribute Name	Value(s)
startjoin	'<' and '/'
endjoin	'>'

The user may wish to edit these.

### Engine Preference

The GENERIC ENGINE Tile is used to create the Engine preference. The resulting script contains a call to CTX\_DDL.SET\_ATTRIBUTE for each attribute of GENERIC ENGINE, but they are commented out. The values need to be edited by the user as appropriate, and the line uncommented.

### Wordlist

The GENERIC WORDLIST tile is used to create the Wordlist preference. For the *stclause* and *instclause* attributes, a commented-out call to CTX\_DDL.SET\_ATTRIBUTE() exists.

*stemmer* is set to 1 (English) and *fuzzy\_match* is set to 1 (English).

If the TextServer3 text column references a section group, then the *section\_group* attribute is set appropriately.

### Stoplist

If there are no stop words in the appropriate TextServer3 wordlist, then the pre-defined CTXSYS.NO\_STOPLIST preference is used.

If there are less than 4096 lowercase stop words in the TextServer3 table wordlist with a length of less than 65 characters, then these words are used to make up a stoplist.

If there are more than 4096 stop words, the CTXSYS.DEFAULT\_STOPLIST preference is used.

## Generating Scripts For Column Policies

The generation of a script is straightforward after the tool has been installed and the UTL\_FILE\_DIR initialization parameter has been set.

The user needs to decide which table to migrate and what the prefix for the preference names are going to be. Then, the user calls the CTXWMG\_MIGRATE.MIGRATE procedure to create the necessary scripts for the migration.

The preference names that the tool generates in the scripts are a concatenation of the prefix supplied by the user, the name of the preference category, and the name of the column for which each script is created.

## Editing the Generated Scripts

The generated scripts contain calls to CTX\_DDL.SET\_ATTRIBUTE that have been commented out with the SQL\*Plus command rem. The user must examine the script to determine whether such calls need uncommenting. The comments are used for attributes where the Migration Tool is unable to determine a reasonable value.

An example of this is for the storage parameters used by various tables and indexes in the Engine preference.

## Column Policy Script Example

This example uses the MIGRATE procedure to create column policy scripts.

In this example, a TextServer3 table, owned by *jbloggs*, is called *resume* and has two indexable columns called *cv* and *address*. The directory that has been chosen as the target directory for the scripts is called `/usr/home/joebloggs/migrate`. The prefix convention chosen is `RESUME_APP`. A database link called *ts3db* has been created for *jbloggs*.

To generate scripts, the procedure MIGRATE is called:

```
begin
    ctwmg_migrate.migrate (tablename=>'RESUME',
                          pref_conv=>'RESUME_APP',
                          scriptdir=>'/usr/home/joebloggs/migrate');
end;
```

From this example, two scripts would be generated, one for each of the indexable columns in *resume*:

- `JBLOGGS_CV.sql`, which creates a column policy called *jbloggs\_cv*
- `JBLOGGS_ADDRESS.sql`, which creates a column policy called *jbloggs\_address*

## Thesaurus Migration

The migration of TextServer3 thesauri and synonym rings to ConText is not as straightforward as the migration of TextServer3 tables. This is because in ConText each term in a thesaurus must be a phrase. In TextServer3, narrower terms can be another thesaurus or a synonym ring. A related term can also be a thesaurus or synonym ring. This means that the functionality of a TextServer3 thesaurus is determined just as much by its structure and its relationship with other thesauri/synonym rings, as by the query operators made available.

Another difference is that a thesaurus in TextServer3 can contain only one hierarchy. In ConText, a thesaurus can contain multiple hierarchies. A ConText thesaurus can also contain synonym rings. The load files create Standard Narrower Term hierarchies.

A term in a ConText synonym ring must be unique to that synonym ring within a particular thesaurus. All synonym ring terms are given a qualifier so that conflicts do not arise. The qualifier is the name of the TextServer3 synonym ring.

The Migration Tool provides four procedures for use in the migration of thesauri and synonym rings. Each of these procedures generates a load file that can be used with ctxload. The ctxload utility can be used for loading thesauri into the ConText dictionary.

**See Also:** For more information about thesauri and synonym rings, see *Oracle8 ConText Cartridge Administrator's Guide*



## Section Group Migration

Section groups can be migrated from TextServer3 to ConText. The generated script should be run from SQL\*Plus to create the section group and add sections to it.

**See Also:** For more information about sections and section groups, see *Oracle8 ConText Cartridge Administrator's Guide*

## CTXWMG\_MIGRATE: TextServer3/ConText Migration Tool Package

The CTXWMG\_MIGRATE PL/SQL package contains the following stored procedures:

Name	Description
MIGRATE	Generates a script for each indexable column in a TextServer3 text table. Each script is used to create a column policy.
MIGRATE_ALL_SYNS	Generates a load file containing all TextServer3 synonym rings owned by a TextServer3 user.
MIGRATE_ALL_THES	Generates a load file containing all TextServer3 thesauri owned by a TextServer3 user.
MIGRATE_SG	Generates a script for a section group.
MIGRATE_SYN	Generates a load file containing a named TextServer3 synonym ring.
MIGRATE_THES	Generates a load file containing a named TextServer3 thesaurus and, optionally, any synonym rings owned by the TextServer3 user.

## MIGRATE

The MIGRATE procedure creates a SQL script for a TextServer3 text table. When the script is run, it generates a ConText policy for an existing table in a ConText database.

### Syntax

```
CTXWMG_MIGRATE.MIGRATE_(tablename    IN VARCHAR2,  
                           pref_conv   IN VARCHAR2,  
                           scriptdir   IN VARCHAR2);
```

#### **tablename**

Specify the name of the TextServer3 text table that contains one or more indexable columns for which you want to create a ConText policy.

#### **pref\_conv**

Specify the string of characters used to generate a prefix for the preference names in the script.

#### **scriptdir**

Specify the directory for the script generated by MIGRATE.

### Notes

The preference names that the tool generates in the script are a concatenation of the prefix supplied by the user, the name of the preference category, and the name of the column for which the script is being created.

For example, if the tool is creating a script for the *doc* column, and the user specifies a prefix of *myapp*, then the Engine preference is called *myapp\_engine\_doc*.

If the preference is being created for an internal filter, then the name of the preference category is substituted with four letters representing the format (e.g. WP51 for WordPerfect 5.1).

## MIGRATE\_ALL\_SYNS

The MIGRATE\_ALL\_SYNS procedure creates a single load file for all the synonym rings owned by the user.

### Syntax

```
CTXWMG_MIGRATE.MIGRATE_ALL_SYNS(loaddir IN VARCHAR2);
```

#### **loaddir**

Specify the directory for load file generated by MIGRATE\_ALL\_SYNS.

### Notes

No 'in-place' expansion occurs.

When the load file generated by MIGRATE\_ALL\_SYNS is used to generate a ConText thesaurus, all the synonym rings from TextServer3 are created within one ConText thesaurus.

The name of the generated load file is based on the user ID for the TextServer3 user. For example, for a user with a TextServer3 user ID of 3, the generated file is called ALL\_SYNONYMS\_3.syn.

## MIGRATE\_ALL\_THES

The MIGRATE\_ALL\_THES procedure creates a single load file for all the thesauri owned by the user.

### Syntax

```
CTXWMG_MIGRATE.MIGRATE_ALL_THES(load_dir IN VARCHAR2);
```

#### **load\_dir**

Specify the directory for load file generated by MIGRATE\_ALL\_THES.

### Notes

No 'in-place' expansion occurs.

When the load file generated by MIGRATE\_ALL\_THES is used to generate a ConText thesaurus, all the hierarchies from the TextServer3 thesauri are created within one ConText thesaurus.

The name of the generated load file is based on the user ID for the TextServer3 user. For example, for a user with a TextServer3 user ID of 3, the generated file is called ALL\_THESAURI\_3.ths.

## MIGRATE\_SG

The MIGRATE\_SG procedure generates a script for a named section group.

### Syntax

```
CTXWMG_MIGRATE.MIGRATE_SG(group_name IN VARCHAR2,  
                           loadaddr   IN VARCHAR2);
```

#### **group\_name**

Specify the name of the section group to be migrated.

#### **loadaddr**

Specify the directory for script generated by MIGRATE\_SG.

### Notes

The name of the generated script is based on the name of the TextServer3 section group. For example, for a TextServer3 section group called MYSECTIONGROUP, the generated script is called SG\_MYSECTIONGROUP.sql.

## MIGRATE\_SYN

The MIGRATE\_SYN procedure generates a load file for a named TextServer3 synonym ring.

### Syntax

```
CTXWMG_MIGRATE.MIGRATE_SYN(syn_sname IN VARCHAR2,  
                             loaddir   IN VARCHAR2);
```

#### **syn\_name**

Specify the name of the synonym ring to be migrated.

#### **loaddir**

Specify the directory for load file generated by MIGRATE\_SYN.

### Notes

No 'in-place' expansion occurs.

The name of the TextServer3 synonym ring is used as a qualifier for each term.

The name of the generated load file is based on the thesaurus name and the user ID for the TextServer3 user. For example, for a TextServer3 synonym ring called MYSYNRING belonging to a user with a TextServer3 user ID of 3, the generated file is called MYSYNRING\_3.ths.

## MIGRATE\_THES

The MIGRATE\_THES procedure generates a load file for a named TextServer3 thesaurus.

### Syntax

```
CTXWMG_MIGRATE.MIGRATE_THES(thesname          IN VARCHAR2,  
                             t_expand_mode     IN INTEGER DEFAULT 0,  
                             s_expand_mode     IN INTEGER DEFAULT 0,  
                             dump_syms        IN INTEGER DEFAULT 0,  
                             loadaddr         IN VARCHAR2);
```

#### **thesname**

Specify the name of the thesaurus to be migrated.

#### **t\_expand\_mode**

Specify the expansion mode for referenced thesauri:

- 0 No thesaurus expansion
- 1 Expand user referenced thesauri in-place
- 2 Expand user and public referenced thesauri in-place

#### **s\_expand\_mode**

Specify the expansion for referenced synonym rings:

- 0 No synonym ring expansion
- 1 Expand single user synonym ring in-place
- 2 Expand multiple user synonym rings in-place
- 3 Expand single user or public synonym ring in-place
- 4 Expand multiple user or public synonym rings in-place

#### **dump\_syms**

Specify to dump all synonym rings owned by current user.

#### **loadaddr**

Specify the directory for load file generated by MIGRATE\_THES.



## Notes

The *t\_expand\_mode* and *s\_expand\_mode* parameters allow users to control what happens when the thesaurus hierarchy contains a narrower term that is a synonym ring.

If thesaurus expansion is used, a referenced thesaurus is expanded 'in-place'. This means that instead of the term being a reference to a thesaurus, it becomes the top-term of the referenced thesaurus. This effectively makes the referenced hierarchy part of the main hierarchy. If a thesaurus that has been made part of the main hierarchy references other thesauri itself, then those thesauri also become part of the main hierarchy. Referenced thesauri may be owned by the user or may be public. Expansion of referenced thesauri can be limited to those owned by the user if required.

In-place synonym ring expansion is similar to thesaurus expansion, but the terms in the referenced ring become narrower terms of the referencing term. The user can limit expansion to those synonym rings owned by the user. It is possible for a synonym ring term to be a reference to another synonym ring. Expansion of such a synonym ring would result in those terms also being narrower terms of the term in the thesaurus. The user can control this, as well as being able to limit the expansion to synonym rings owned by the user.

If a value of 1 is specified for the *dump\_syms* parameter, MIGRATE\_THES dumps all synonym rings into the same load file. When the load file is used to create a thesaurus in ConText, all the dumped TextServer3 synonym rings are part of that ConText thesaurus.

The name of the generated load file is based on the thesaurus name and the user ID for the TextServer3 user. For example, for a TextServer3 thesaurus called MYTHES belonging to a user with a TextServer3 user ID of 3, the generated file is called MYTHES\_3.ths.

## Example Migration Using Supplied Script

A sample script is supplied with the Migration Tool. It can be altered to suit the user's requirements. It prompts the user for values.

The following example makes use of the sample script to demonstrate how a user's text table and thesaurus can be migrated from TextServer3 to ConText.

### Example Description

In this example, the TextServer3 user is *jbloggs* on a TextServer3 database using a connect string 'prod\_db'. The *jbloggs* user has a TextServer3 user ID of 3 and one text table called *odc\_papers*:

- the textkey column is *paper\_id*
- the text columns are *summary* and *paper*
- the table has a format ID column called *format\_id*

The *summary* column and the documents in the *paper* column are in the same format, which is either Word for Windows 6 or WordPerfect 6 format. Both of these formats are supported internally by ConText. Neither of these two columns are external.

The application uses a single thesaurus, called *MY\_THESAURUS*, but it does contain terms that are references to other thesauri and synonym rings. Some of these are owned by the user, others are public. The user requires the same terms to be returned from operations on the thesaurus using ConText.

The directory chosen for creating the column policy script and load files is */usr/home/jobloggs/migrate*. The script creates the database link, and drops it after finishing the script generation.

### Using the Example

The steps for migrating the text table and thesaurus are:

1. Install the Migration Tool on the ConText database. For the purpose of this exercise, it is installed for the *jbloggs* user.
2. In the ConText database, create the TextServer3 table to be migrated. It need not be populated at this stage.
3. Add the UTL\_FILE\_DIR parameter to the init.ora file used by the ConText database and restart this database.

4. As the Oracle user for whom the migration package was installed, run the script `samp_mig.sql`:

```
% sqlplus jbloggs/jbloggs @samp_mig
```

Enter the following information at the prompts:

```
jbloggs
jbloggs
prod_db
/usr/home/joebloggs/migrate
odc_papers
joes_app
2
4
1
```

## Example Results

The Migration Tool creates three script files in the directory `/usr/home/joebloggs/migrate`:

- *JBLOGGS\_SUMMARY.sql* - creates a policy called `JBLOGGS_SUMMARY`
- *JBLOGGS\_PAPER.sql* - creates a policy called `JBLOGGS_PAPER`
- *MY\_THESAURUS\_3.ths*

If these scripts are then run from SQL\*Plus, they create column policies for each of the text columns. Both column policies use the Autorec functionality to invoke the internal filters since the Migration Tool determines that all the formats used are supported by ConText. Before running the scripts, the user may choose to edit the preference name or any of the attribute values.

The thesaurus is migrated to ConText by using the `ctxload` utility. Assuming that the thesaurus should keep the name 'MY\_THESAURUS', the following command would import the thesaurus:

```
% ctxload -user jbloggs/jbloggs -name MY_THESAURUS -file MY_THESAURUS_3.ths -thes
```

This completes the migration of the TextServer3 Dictionary information for the text table, `ODC_PAPERS`. The user should now import the data from the text table in the TextServer3 database and create a ConText index for the data.



---

## Viewer Cartridge: Manual Configuration

This appendix documents the various procedures for configuring and deconfiguring Oracle WebServer 2.1 and Oracle Web Application Server 3.0 for use with the ConText Viewer Cartridge.

This appendix contains the following topics:

- Notational Conventions for Configuration/Deconfiguration
- Configuring Oracle WebServer 2.1 for the Viewer Cartridge
- Deconfiguring Oracle WebServer 2.1
- Configuring Oracle Web Application Server 3.0 for the Viewer Cartridge
- Deconfiguring Oracle Web Application Server 3.0

## Notational Conventions for Configuration/Deconfiguration

The section provides details for the notational conventions that apply to all of the configuration and deconfiguration procedures for Oracle WebServer 2.1 and Oracle Web Application Server 3.0.

### Buttons

Buttons referenced appear within square brackets, with capitalization the same as it appears in the browser.

For example: [Modify Listener]

### Links

Links appear within angle brackets.

For example: <Web Request Broker>

### Pathnames

All paths mentioned in the configuration steps follow the UNIX convention of directory names separated by forward slashes (i.e. '/'). If a path is a physical path (as opposed to a virtual path), and the platform is Windows NT, instead of a UNIX-based platform), then use the backslash character (i.e. '\') to separate directory names.

## Configuring Oracle WebServer 2.1 for the Viewer Cartridge

Configuration of Oracle WebServer 2.1 for use with the Viewer Cartridge is normally performed through prompts provided by the Oracle Installer during installation of the ConText Workbench.

In the event that the WebServer configuration portion of the Viewer Cartridge installation fails, manual configuration is necessary to set up Oracle WebServer 2.1 for use with ConText Viewer Cartridge.

---

**Note:** These procedures are different from those that apply for Oracle Web Application Server 3.0.

---

The manual configuration is carried out through a browser, such as Netscape Navigator or Microsoft Internet Explorer. The browser should have forms and frame capabilities.

Deconfiguration procedures can also be applied to remove the Viewer Cartridge from Oracle WebServer 2.1.

**See Also:** For more information about deconfiguring Oracle WebServer 2.1 to remove the Viewer Cartridge, see “Deconfiguring Oracle WebServer 2.1” in this chapter.

### Configuration Requirements

The following information is required during configuration:

- the name of the machine on which Oracle WebServer is running
- the URL of the WebServer Administration Page; usually:  
`http://<hostname>:8888/ows-adoc/Intro.html`
- the name of the listener used to access the Viewer Cartridge
- the type of platform that Oracle WebServer is running on (UNIX-based or Windows NT)
- the Oracle home directory for the Oracle WebServer 2.1 installation

---

---

**Note:** In certain steps of the configuration process for Oracle WebServer 2.1, it is necessary to enter filesystem paths. Part of the path may be the Oracle home directory, which is represented in the configuration steps as %ORACLE\_HOME%. The actual value of ORACLE\_HOME should be used instead.

The following example assumes an Oracle home directory of D:\ORANT on Windows NT. If the step asks you to enter the following value in a field:

```
%ORACLE_HOME%/ctxw/middle/ctxvcart/
```

then enter the value:

```
D:\ORANT\ctxw\middle\ctxvcart\
```

---

---

## Configuration Tasks

There are three major tasks in the configuration procedure:

- Configuring Oracle Web Listener
- Creating the Web Request Broker Cartridge Configuration
- Configuring Oracle Web Request Broker

### Configuring Oracle Web Listener

This section describes how to configure a Web Listener for use with the Viewer Cartridge. A MIME type is mapped to a File Extension for use with the Viewer Cartridge. Security for the virtual directory is also configured.

1. Access the Oracle WebServer Administration page.
2. On the Oracle WebServer Administration page:
  - Select the <Oracle Web Listener> link.
  - The Oracle Web Listener Administration page appears.
3. On the Oracle Web Listener (OWL) Administration page:
  - Select <CONFIGURE> for the listener used to access the Viewer Cartridge.
  - The OWL Administration - Server Advanced Configuration page appears.
4. On the OWL Administration - Server Advanced Configuration page:



Scroll down to the MIME Types block of fields to insert values into the first empty row.

In the Mime Type column, enter:

`application/x-ctxv`

In the File Extension(s) column, enter:

`ctxv`

Click [Modify Listener].

The Oracle Web Listener Administration page appears.

5. On the Oracle Web Listener (OWL) Administration page, select <CONFIG-URE> for the listener used to access the Viewer Cartridge.

The OWL Administration - Server Advanced Configuration page appears.

6. On the OWL Administration - Server Advanced Configuration page, select <Security: Access Control and Encryption>.

The Oracle Web Listener Security page appears.

7. On the Oracle Web Listener Security page, scroll to the Basic Users block of fields. In the User Name column of a blank row, enter:

`ctxquery`

In the Password field, enter:

`ctxquery`

Scroll to the Basic Groups block of fields. In the Group Name column, enter:

`ctxview_group`

In the User(s) column, enter:

`ctxquery`

Scroll to the Basic Realms block of fields. In the Realms column, enter:

`ctxview_realm`

In the Group(s) column, enter:

`ctxview_group`

Scroll to the Protection block of fields. In the Virtual Path column, enter:

```
/ctxwview/
```

In the Basic/Digest column, select Basic.

In the Realm column, enter:

```
ctxview_realm
```

Click [Modify Listener].

The Oracle Web Listener Security page reappears.

Scroll to the end of the page and click [Administration].

### Creating the Web Request Broker Cartridge Configuration

This section describes how to create the Web Request Broker cartridge configuration for the ConText Viewer Cartridge. The configuration provides the Viewer Cartridge with information it requires when processing requests.

1. Access the Oracle WebServer Administration page.
2. On the Oracle WebServer Administration page, select the <Web Request Broker> link.

The Web Request Broker Administration page appears.

3. On the Web Request Broker Administration page, select <Modify> for the listener used to access the Viewer Cartridge.

The WRB Cartridge Administration Page appears.

4. On the WRB Cartridge Administration Page:
5. In the Cartridges field, enter:

```
CTXWVIEW
```

Click [Create Cartridge].

The WRB Cartridge Configuration Page appears.

6. On the WRB Cartridge Configuration Page, in row 1, column 1 of the Cartridge Parameters block of fields, enter:

```
CLASSPATH
```

If the Oracle WebServer platform is UNIX, in row 1, column 2, enter:

%ORACLE\_HOME%/ows21/java/classes.zip;%ORACLE\_HOME%/ows21/javaoracle.zip;%ORACLE\_HOME%/jdbc/lib/classes102.zip;%ORACLE\_HOME%/ctxw/middlectxvcart/ctxvcart.zip

If the Oracle WebServer platform is NT, in row 1, column 2, enter:

%ORACLE\_HOME%\ows21\java\classes.zip;%ORACLE\_HOME%\ows21\java\oracle.zip;%ORACLE\_HOME%\jdbc\lib\classes102.zip;%ORACLE\_HOME%\ctxw\middle\ctxvcart\ctxvcart.zip

In row 2, column 1, enter:

LD\_LIBRARY\_PATH

If the Oracle WebServer platform is UNIX, in row 2, column 2, enter:

%ORACLE\_HOME%/ows21/bin:%ORACLE\_HOME%/ows21/lib:%ORACLE\_HOME%/ows21/java/lib:%ORACLE\_HOME%/jdbc/lib

If the Oracle WebServer platform is NT, in row 2, column 2, enter:

%ORACLE\_HOME%\ows21\bin

In row 3, column 1, enter:

JAVA\_HOME

In row 3, column 2, enter the physical path:

%ORACLE\_HOME%/ows21/java

Click the [Modify Cartridge Configuration] button.

The WRB Cartridge Configuration page reappears.

## Configuring Oracle Web Request Broker

This section describes how to set up the ConText Viewer Cartridge so that the Web Request Broker knows how to invoke it. The physical cartridge is mapped to the virtual path of the Viewer Cartridge, and security for the Viewer Cartridge is configured.

1. Access the Oracle WebServer Administration page.
2. On the Oracle WebServer Administration page, select the <Web Request Broker> link.

The Web Request Broker Administration page appears.

3. On the Web Request Broker Administration page, select <Modify> for the listener used to access the Viewer Cartridge.

The WRB Cartridge Administration Page appears.

4. On the WRB Cartridge Administration Page, scroll to the Application and Objects block of fields.

In the App. column, enter:

`CTXWVIEW`

If the Oracle WebServer platform is UNIX, in the Object Path column, enter:

`%ORACLE_HOME%/ows21/lib/libjava.so`

If the Oracle WebServer platform is NT, in the Object Path column, enter:

`%ORACLE_HOME%\ows21\bin\javai.dll`

In the Entry Point column, enter:

`ojsdinit`

In the Min column, enter:

`0`

In the Max column, enter:

`30`

Scroll to the Applications and Directories block of fields. In the Virtual Path column, enter:

`/ctxwview`

In the App. column, enter:

`CTXWVIEW`

In the Physical Path column, enter:

`%ORACLE_HOME%/ctxw/middle/ctxvcart/ctxvcart.zip`

Scroll to the Protecting Applications block of fields. In the Virtual Path column, enter:

`/ctxwview/`

In the Basic/Digest column, select Basic.

In the Realm column, enter:

`ctxview_realm`

Click [Modify WRB Configuration].

The WRB Cartridge Administration page reappears.

The configuration of Oracle WebServer for use with the ConText Viewer Cartridge is now complete.

To use WebServer and the Viewer Cartridge, stop and restart the associated Web-Server.

## Deconfiguring Oracle WebServer 2.1

This section provides detailed instructions for deconfiguring Oracle WebServer 2.1 for the Viewer Cartridge.

### Deconfiguration Requirements

The following information is required before deconfiguration starts:

- the name of the listener configured to access the Viewer Cartridge
- the URL of the WebServer Manager Home Page

---

---

**Note:** In certain steps of the deconfiguration process for Oracle WebServer 2.1, it is necessary to remove filesystem paths. A portion of the path may be the Oracle home directory, which is represented in the configuration steps as %ORACLE\_HOME%. The actual value of ORACLE\_HOME should be used instead.

The following example assumes an Oracle home directory of D:\ORANT on Windows NT. If the step asks you to remove the following value from a field:

```
%ORACLE_HOME%/ctxw/middle/ctxvcart/
```

then remove the value:

```
D:\ORANT\ctxw\middle\ctxvcart\
```

---

---

### Deconfiguration Tasks

There are three major tasks in the deconfiguration procedure:

- Deconfiguring Oracle Web Listener
- Deleting the Web Request Broker Cartridge Configuration
- Deconfiguring Oracle Web Request Broker

---

---

**Note:** Oracle WebServer must be restarted once deconfiguration has completed.

---

---

## Deconfiguring Oracle Web Listener

This section describes how to remove the Viewer Cartridge configuration from the Web Listener used to access the Viewer Cartridge.

1. Access the Oracle WebServer Administration page.
2. On the Oracle WebServer Administration page, select the <Oracle Web Listener> link.

The Oracle Web Listener Administration page appears.

3. On the Oracle Web Listener (OWL) Administration page, select <CONFIGURE> for the listener which is used to access the Viewer Cartridge.

The OWL Administration - Server Advanced Configuration page appears.

4. On the OWL Administration - Server Advanced Configuration page, scroll down to the MIME Types block of fields.

Remove all values from the row that has the following value in the Mime Type column:

```
application/x-ctxv
```

Click [Modify Listener].

The Oracle Web Listener Administration page appears.

5. On the Oracle Web Listener (OWL) Administration page, select <CONFIGURE> for the listener which is used to access the Viewer Cartridge.

The OWL Administration - Server Advanced Configuration page appears.

6. On the OWL Administration - Server Advanced Configuration page, select <Security: Access Control and Encryption>.

The Oracle Web Listener Security page appears.

7. On the Oracle Web Listener Security page, scroll to the Basic Users block of fields.

Remove all values from the row that has the following value in the User Name column:

```
ctxquery
```

Scroll to the Basic Groups block of fields. Remove all values from the row that has the following value in the Group Name column:

```
ctxview_group
```

Scroll to the Basic Realms block of fields. Remove all values from the row that has the following value in the Realms column:

`ctxview_realm`

Scroll to the Protection block of fields. Remove all values from the row that has the following value in the Virtual Path column:

`/ctxwview/`

Click [Modify Listener].

The Oracle Web Listener Security page reappears.

Scroll to the end of the page and click [Administration].

### **Deleting the Web Request Broker Cartridge Configuration**

This section describes how to remove the Web Request Broker cartridge configuration for the ConText Viewer Cartridge.

1. Access the Oracle WebServer Administration page.
2. On the Oracle WebServer Administration page, select the <Web Request Broker> link.

The Web Request Broker Administration page appears.

3. On the Web Request Broker Administration page, select <Modify> for the listener which is used to access the Viewer Cartridge.

The WRB Cartridge Administration Page appears.

4. On the WRB Cartridge Administration Page, in the Cartridges section, select <Delete> for the CTXWVIEW cartridge.

The Confirm Cartridge Deletion page appears.

Click [Confirm Delete].

The resulting page shows that the CTXWVIEW cartridge configuration has been deleted.



## Deconfiguring Oracle Web Request Broker

This section describes how to remove the ConText Viewer Cartridge configuration from the Web Request Broker of the Web Listener used to access the Viewer Cartridge.

1. Access the Oracle WebServer Administration page.
2. On the Oracle WebServer Administration page, select the <Web Request Broker> link.

The Web Request Broker Administration page appears.

3. On the Web Request Broker Administration page, select <Modify> for the listener which is used to access the Viewer Cartridge.

The WRB Cartridge Administration Page appears.

4. On the WRB Cartridge Administration Page, scroll to the Application and Objects block of fields.

Remove all values from the row that has the following value in the App. column:

`CTXWVIEW`

Scroll to the Applications and Directories block of fields. Remove all values from the row that has the following value in the Virtual Path column:

`/ctxwview`

Scroll to the Protecting Applications block of fields. Remove all values from the row that has the following value in the Virtual Path column:

`/ctxwview/`

Click [Modify WRB Configuration].

The WRB Cartridge Administration page reappears.

## Configuring Oracle Web Application Server 3.0 for the Viewer Cartridge

Configuration of Oracle Web Application Server 3.0 for use with the Viewer Cartridge is normally performed through prompts provided by the Oracle Installer during installation of the ConText Workbench.

In the event that the Web Application Server configuration portion of the Viewer Cartridge installation fails, manual configuration is necessary to set up Web Application Server for use with the ConText Viewer Cartridge.

---

**Note:** These procedures are different from those that apply for Oracle WebServer 2.1.

---

The manual configuration is performed through a browser, such as Netscape Navigator or Microsoft Internet Explorer. The browser should have forms and frame capabilities.

Deconfiguration procedures can also be applied to remove the Viewer Cartridge from Oracle Web Application Server 3.0.

**See Also:** For more information about deconfiguring Oracle WebServer 3.0 to remove the Viewer Cartridge, see “Deconfiguring Oracle Web Application Server 3.0” in this chapter.

## Configuration Requirements

The following information is required during configuration:

- the name of the machine on which Oracle Web Application Server is running
- the URL of the Web Application Server Manager Home Page; usually:  
`http://<hostname>:8888/ows-adoc/Intro.html`
- the name of the listener used to access the Viewer Cartridge
- the type of platform that Oracle Web Application Server is running on (UNIX-based or Windows NT)

---

**Note:** In certain steps of the configuration process for Oracle Web Application Server 3.0, it is necessary to enter filesystem paths. Part of the path may be the Oracle or Oracle Web Application Server home directory, which is represented in the configuration steps as either %ORACLE\_HOME% or %ORAWEB\_HOME%.

*Do not* substitute the real values unless explicitly instructed to do so.

---

## Configuration Tasks

There are four major tasks in the configuration procedure:

- Configuring the Oracle Web Listener
- Creating the Web Request Broker Cartridge
- Configuring Viewer Cartridge Specific Parameters
- Configuring Authorization Server

---

**Note:** Oracle Web Application Server must be restarted once configuration has completed.

---

### Configuring the Oracle Web Listener

This section describes how to configure a Web Listener for use with the Viewer Cartridge. A MIME type is mapped to a File Extension for use with the Viewer Cartridge. Security for the virtual directory is also configured.

1. Access the Oracle Web Application Server Administration page.
2. On the Oracle Web Application Server Administration page, select the <Oracle Web Listener> link.

The Oracle Web Listener Administration page appears.

3. On the Oracle Web Listener (OWL) Administration page, select <CONFIGURE> for the listener used to access the Viewer Cartridge.

The OWL Administration - Server Advanced Configuration page appears.

4. On the OWL Administration - Server Advanced Configuration page, scroll down to the MIME Types block of fields to insert values into the first empty row.

In the Mime Type column, enter:

`application/x-ctxv`

In the File Extension(s) column, enter:

`ctxv`

Click [Modify Listener].

The Oracle Web Listener Administration page appears.

5. On the Oracle Web Listener (OWL) Administration page, select <CONFIG-URE> for the listener used to access the Viewer Cartridge.

The OWL Administration - Server Advanced Configuration page appears.

6. On the OWL Administration - Server Advanced Configuration page, select the <Security> link.

The Oracle Web Listener Security page appears.

7. On the Oracle Web Listener Security page, scroll to the Basic Users block of fields.

In the User Name column of a blank row, enter:

`ctxquery`

In the Password field, enter:

`ctxquery`

Scroll to the Basic Groups block of fields. In the Group Name column, enter:

`ctxview_group`

In the User(s) column, enter:

`ctxquery`

Scroll to the Basic Realms block of fields. In the Realms column, enter:

`ctxview_realm`

In the Group(s) column, enter:

`ctxview_group`

Scroll to the Protection block of fields. In the Virtual Path column, enter:

`/ctxwview/`

In the Basic/Digest column, select Basic.

In the Realm column, enter:

`ctxview_realm`

Click [Modify Listener].

The Oracle Web Listener Security page reappears.

Scroll to the end of the page and click [Admin].

### Creating the Web Request Broker Cartridge

This section describes how to create a configuration for the ConText Viewer Cartridge. The configuration provides the Web Request Broker with information required to access the Viewer Cartridge; it also defines and protects the virtual path used to access the cartridge.

1. Access the Oracle Web Application Server Administration page.
2. On the Oracle Web Application Server Administration page, select the <Oracle Web Application Server> link.

The Oracle Web Application Server Administration page appears.

3. On the Oracle Web Application Server Administration page, select the <Cartridge Administration> link.

The Cartridge Administration Page appears.

4. On the Cartridge Administration Page, select the <Add New Cartridge> link.

The New Cartridge Configuration Page appears.

5. On the New Cartridge Configuration Page, select the <Add New Cartridge with Manual Configuration> link.

The New Cartridge Configuration Page appears.

6. On the New Cartridge Configuration Page, in the Cartridge Name field, enter:

`CTXWVIEW`

If the Oracle Web Application Server platform is UNIX-based, in the Object Path column, enter:

`%ORAWEB_HOME%/lib/libjava.so`

If the Oracle Web Application Server platform is Windows NT, in the Object Path column, enter:

`%ORAWEB_HOME%\bin\javai.dll`

In the Entry Point field, enter:

`ojsdinit`

In the Minimum # of Instances field, enter:

`0`

In the Maximum # of Instances field, enter:

`100`

In the Cartridge Description field, enter:

`ConText Viewer Cartridge`

In the Cartridge Icon field, enter:

`/ows-img/32java.gif`

In the Client Certificate field, select:

`Disabled`

In the Client Sessions field, select:

`Disabled`

In the Max session idle time field, select:

`0`

In the Error Page field, enter

`%ORAWEB_HOME%/admdoc/wrberr.html`

Scroll to the Virtual Paths block of fields. In the Virtual Path column, enter:

`/ctxwview`

In the Physical Path column, enter:

`%ORACLE_HOME%/ctxw/middle/ctxvcart/ctxvcart.zip`

Scroll to the Virtual Path Protection block of fields. In the Virtual Path column, enter:

```
/ctxwview/
```

In the Scheme field, select:

```
Basic
```

In the Realm field, enter:

```
ctxwview_realm
```

Click [Register New Cartridge].

The Cartridge Administration page reappears.

### **Configuring Viewer Cartridge Specific Parameters**

This section describes how to create specific parameters used by the ConText Viewer Cartridge when processing requests.

1. Access the Oracle Web Application Server Administration page.
2. On the Oracle Web Application Server Administration page, select the <Oracle Web Application Server> link.

The Oracle Web Application Server Administration page appears.

3. On the Oracle Web Application Server Administration page, select the <Cartridge Administration> link.

The Cartridge Administration Page appears.

4. On the Cartridge Administration Page, select the <ConText Viewer Cartridge> link.

The CTXWVIEW Cartridge Configuration page appears.

Select the <CTXWVIEW Cartridge specific parameters> link.

The Cartridge Configuration Page appears.

5. On the Cartridge Configuration Page, in row 1, column 1 of the Cartridge Parameters block of fields, enter:

```
CLASSPATH
```

If the Oracle Web Application Server platform is UNIX-based, in row 1, column 2, enter:

```
%ORAWEB_HOME%/java/classes.zip:%ORAWEB_HOME%/java/oracle.zip:%ORACLE_HOME%/jdbc/lib/  
classes102.zip:%ORACLE_HOME%/ctxw/middle/ctxvcart/ctxvcart.zip
```

If the Oracle Web Application Server platform is NT, in row 1, column 2, enter:

```
%ORAWEB_HOME%\java\classes.zip:%ORAWEB_HOME%\java\oracle.zip:%ORACLE_HOME%\jdbc\lib\classes102.zip:%ORACLE_HO  
ME%\ctxw\middle\ctxvcart\ctxvcart.zip
```

In row 2, column 1, enter:

```
LD_LIBRARY_PATH
```

If the Oracle Web Application Server platform is UNIX, in row 2, column 2, enter:

```
%ORAWEB_HOME%/bin:%ORAWEB_HOME%/lib:%ORAWEB_HOME%/java/lib:%ORACLE_HOME%/jdbc/lib
```

If the Oracle Web Application Server platform is NT, in row 2, column 2, enter:

```
%ORAWEB_HOME%\bin
```

In row 3, column 1, enter:

```
JAVA_HOME
```

In row 3, column 2, enter the physical path:

```
%ORAWEB_HOME%/java
```

Click [Modify Cartridge Configuration].

The Cartridge Configuration page reappears.

## Configuring Authorization Server

This section describes how to configure the Authorization Server for use with Con-Text Viewer Cartridge. It provides authentication for Web Application Server cartridges to authenticate users before allowing them to use the cartridge.

1. Access the Oracle Web Application Server Administration page.
2. On the Oracle Web Application Server Administration page, select the <Oracle Web Application Server> link.

The Oracle Web Application Server Administration page appears.



3. On the Oracle Web Application Server Administration page, select the <Authorization Server> link.

The Authentication Configuration page appears.

4. On the Authentication Configuration page, select the <Basic> link.

The Basic Authentication page appears.

5. On the Basic Authentication page, scroll to the Basic Users block of fields.

In the User Name column, enter:

`ctxquery`

In the Password column, enter:

`ctxquery`

Scroll to the Basic Groups block of fields. In the Group Name column, enter:

`ctxview_group`

In the User(s) column, enter:

`ctxquery`

Scroll to the Basic Realms block of fields. In the Realms column, enter:

`ctxview_realm`

In the Group(s) column, enter:

`ctxview_group`

Click [Modify].

The resulting page shows that the Basic Authentication was updated.

The configuration of Oracle Web Application Server for use with the ConText Viewer Cartridge is now complete. Scroll to the end of the page and click [Administration].

## Deconfiguring Oracle Web Application Server 3.0

This section describes how to remove the ConText Virtual Cartridge configuration from the Web Listener used to access the Viewer Cartridge.

Oracle Web Application Server needs to be restarted after the deconfiguration has been completed.

### Deconfiguration Requirements

The following information is required before deconfiguration starts:

- the name of the listener configured to access the Viewer Cartridge
- the URL of the Web Application Server Manager Home Page

### Deconfiguration Tasks

There are four major tasks in the deconfiguration procedure:

- Deconfiguring the Oracle Web Listener
- Deleting Web Request Broker Cartridge
- Deconfiguring Viewer Cartridge Specific Parameters
- Deconfiguring Authorization Server

#### Deconfiguring the Oracle Web Listener

This section describes how to remove the ConText Viewer Cartridge configuration from the Web Listener used to access the cartridge.

1. Access the Oracle Web Application Server Administration page.
2. On the Oracle Web Application Server Administration page, select the <Oracle Web Listener> link.

The Oracle Web Listener Administration page appears.

3. On the Oracle Web Listener (OWL) Administration page, select <CONFIG-URE> for the listener which is used to access the Viewer Cartridge.

The OWL Administration - Server Advanced Configuration page appears.

4. On the OWL Administration - Server Advanced Configuration page, scroll down to the MIME Types block of fields.

Remove all values from the row that has the following value in the Mime Type column:

`application/x-ctxv`

Click [Modify Listener].

The Oracle Web Listener Administration page appears.

5. On the Oracle Web Listener (OWL) Administration page, select <CONFIGURE> for the listener which is used to access the Viewer Cartridge.

The OWL Administration - Server Advanced Configuration page appears.

6. On the OWL Administration - Server Advanced Configuration page, select <Security>.

The Oracle Web Listener Security page appears.

7. On the Oracle Web Listener Security page, scroll to the Basic Users block of fields.

Remove all values from the row that has the following value in the User Name column:

`ctxquery`

Scroll to the Basic Groups block of fields. Remove all values from the row that has the following value in the Group Name column:

`ctxview_group`

Scroll to the Basic Realms block of fields. Remove all values from the row that has the following value in the Realms column:

`ctxview_realm`

Scroll to the Protection block of fields. Remove all values from the row that has the following value in the Virtual Path column:

`/ctxwview/`

Click [Modify Listener].

The Oracle Web Listener Security page reappears.

Scroll to the end of the page and click [Administration].

### **Deleting Web Request Broker Cartridge**

This section describes how to delete the Viewer Cartridge from the Web Request Broker.

1. Access the Oracle Web Application Server Administration page.
2. On the Oracle Web Application Server Administration page, select the <Oracle Web Application Server> link.

The Oracle Web Application Server Administration page appears.

3. On the Oracle Web Application Server Administration page, select the <Cartridge Administration> link.

The Cartridge Administration Page appears.

4. On the Cartridge Administration Page, select <Delete Cartridge>.

The Delete Cartridge Configuration Page appears.

5. On the Delete Cartridge Configuration Page, in the Select a Cartridge field, select:

CTXWVIEW

Click [Delete].

The Confirm Cartridge Deletion page appears.

Click [Delete].

The resulting page shows that the CTXWVIEW cartridge configuration has been deleted.

### **Deconfiguring Viewer Cartridge Specific Parameters**

This section describes how to remove the ConText Viewer Cartridge configuration from the Web Request Broker.

1. Access the Oracle Web Application Server Administration page.
2. On the Oracle Web Application Server Administration page, select the <Oracle Web Application Server> link.

The Oracle Web Application Server Administration page appears.

3. On the Oracle Web Application Server Administration page, select the <Cartridge Administration> link.

The Cartridge Administration Page appears.

4. On the Cartridge Administration Page, select the <Cartridge Summary (Web Request Broker)> link.

The Web Request Broker Administration page appears.

5. On the Web Request Broker Administration Page, scroll to the Protecting Applications block of fields.

Remove all values from the rows that have the following value in the Virtual Path column:

```
/ctxwview/
```

Click [Modify WRB Configuration].

The WRB Cartridge Administration page reappears.

### Deconfiguring Authorization Server

This section describes how to remove the ConText Viewer Cartridge configuration from the Authorization Server.

1. Access the Oracle Web Application Server Administration page.
2. On the Oracle Web Application Server Administration page, select the <Oracle Web Application Server> link.

The Oracle Web Application Server Administration page appears.

3. On the Oracle Web Application Server Administration page, select the <Authorization Server> link.

The Authentication Configuration page appears.

4. On the Authentication Configuration page, select the <Basic> link.

The Basic Authentication page appears.

5. On the Basic Authentication page, scroll to the Basic Users block of fields.

Remove all values from the row that has the following value in the User Name column:

```
ctxquery
```

Scroll to the Basic Groups block of fields. Remove all values from the row that has the following value in the Group Name column:

```
ctxview_group
```

Scroll to the Basic Realms block of fields. Remove all values from the row that has the following value in the Realms column:

`ctxview_realm`

Click [Modify].

The resulting page shows that the Basic Authentication was updated.

The Oracle Web Application Server Authentication Configuration reappears.

The deconfiguration of Oracle Web Application Server 3.0 for use with the ConText Viewer Cartridge is now complete.

---

# **Configuration Manager: Manual Installation and Configuration**

This appendix documents the various manual procedures for installing the Configuration Manager packages, as well as configuring and deconfiguring Oracle WebServer 2.1 and Oracle Web Application Server 3.0 for use with the Configuration Manager.

The following topics are discussed in this chapter:

- Manual Installation of Database Packages
- Notational Conventions for Configuration/Deconfiguration
- Configuring Oracle WebServer 2.1 for the Configuration Manager
- Deconfiguring Oracle WebServer 2.1
- Configuring Oracle Web Application Server 3.0 for the Configuration Manager
- Deconfiguring Oracle Web Application Server 3.0

## Manual Installation of Database Packages

Installation of Configuration Manager is handled through prompts provided by the Oracle Installer during installation of the ConText Workbench; however, if the I/O utility (ctxio32), which is called by the Oracle Installer to create the database packages used by the Configuration Manager, fails to create the database packages, it may be necessary to manually install the database packages.

A SQL\*Plus script is provided for this purpose. It is installed with the Configuration Manager. The script, and the packages it installs, only need to be installed if the normal installation procedure fails.

If the normal installation procedure fails, the location of the installed files depends on whether the installation was attempted against a Web Server running on the local machine or a remote machine.

### Location of Installation Files (Local Installation)

If the installation was against a Web Server on the local machine, the files are located in the following directory:

```
%ORACLE_HOME%\ctxw\middle\cfgmgr\install
```

### Location of Installation Files (Remote Installation)

If the installation was against a Web Server on a remote machine, the files are located in the following directory:

```
$ORACLE_HOME\ctxw\middle\cfgmgr\remote\install
```

## Installing the Database Packages

To install the packages, change directories to the directory in which the packages are installed and use the following command from the command-line prompt:

```
> plus33 ctxsys/ctxsys@connect_string @cfgpack
```



## Notational Conventions for Configuration/Deconfiguration

The section provides details for the notational conventions that apply to all of the configuration and deconfiguration procedures for Oracle WebServer 2.1/Oracle Web Application Server 3.0.

### Buttons

Buttons referenced appear within square brackets, with capitalization the same as it appears in the browser.

For example: [Modify Listener]

### Links

Links appear within angle brackets.

For example: <Web Request Broker>

### Pathnames

All paths mentioned in the configuration steps follow the UNIX convention of directory names separated by forward slashes (i.e. '/'). If a path is a physical path (as opposed to a virtual path), and the platform is Windows NT, instead of a UNIX-based platform), then use the backslash character (i.e. '\') to separate directory names.

## Configuring Oracle WebServer 2.1 for the Configuration Manager

Configuration of Oracle WebServer 2.1 for use with the Configuration Manager is normally performed through prompts provided by the Oracle Installer during installation of the ConText Workbench.

In the event that the WebServer configuration portion of the Configuration Manager installation fails, manual configuration is necessary to set up Oracle WebServer 2.1 for use with ConText Configuration Manager.

---

**Note:** These procedures are different from those that apply for Oracle Web Application Server 3.0.

---

The manual configuration is carried out through a browser, such as Netscape Navigator or Microsoft Internet Explorer. The browser should have forms and frame capabilities.

Deconfiguration procedures can also be applied to remove the Configuration Manager from Oracle WebServer 2.1.

**See Also:** For more information about deconfiguring Oracle WebServer 2.1 to remove the ConText Configuration Manager, see “Deconfiguring Oracle WebServer 2.1” in this chapter.

## Configuration Requirements

The following information is required during configuration steps:

- the name of the machine on which Oracle WebServer is running
- either the SID or Connect String of the database on which ConText is installed
- the password of the CTXSYS user
- the URL of the WebServer Administration Page, usually  
`http://<hostname>:8888/ows-adoc/Intro.html`
- the name of the listener used to access the Configuration Manager
- the port number of the listener used to access the Configuration Manager
- the type of platform that Oracle WebServer is running on (UNIX-based or Windows NT)
- Oracle home directory of the Oracle WebServer 2.1 installation

---

**Note:** In certain steps of the configuration process for Oracle WebServer 2.1, it is necessary to enter filesystem paths. Part of the path may be the Oracle home directory, which is represented in the configuration steps as %ORACLE\_HOME%. The actual value of ORACLE\_HOME should be used instead.

The following example assumes an Oracle home directory of D:\ORANT on Windows NT. If the step asks you to enter the following value in a field:

```
%ORACLE_HOME%/ctxw/middle/cfgmgr/
```

then enter the value:

```
D:\ORANT\ctxw\middle\cfgmgr\
```

---

## Configuration Tasks

There are three major tasks in the configuration procedure:

- Creating a DCD
- Configuring Oracle Web Listener
- Configuring Oracle Web Request Broker

---

**Note:** Oracle WebServer must be restarted once configuration has completed.

---

### Creating a DCD

This section describes the creation of a new Database Connection Descriptor (DCD) which provides PL/SQL Agent with the required information when connecting to a database. It also installs the WebServer Developer's Toolkit packages to the database. The DCD is used by the ConText Configuration Manager to process administration requests from a browser.

1. Access the Oracle WebServer Administration page.
2. On the Oracle WebServer Administration page, select the <PL/SQL Agent> link.

The PL/SQL Agent Administration page appears.

3. On the PL/SQL Agent Administration page, select the <Create New DCD> link.

The PL/SQL Agent Administration - Create New DCD page appears.

4. On the Create New DCD page, in the DCD Name field, enter:

`ctx_dba`

In the PL/SQL Agent Database User, enter:

`CTXSYS`

Click the Identified by Password radio button.

In the PL/SQL Agent User Password and Confirm Password fields, enter the password of the CTXSYS user.

If the database on which ConText is installed is a local database, enter the ORACLE\_SID in the ORACLE\_SID field; otherwise, enter a connect string in the SQL\*Net V2 Service field.

In the Authorized Ports field, enter the port number of the listener. This port number is used to access the Configuration Manager.

Select the Install WebServer Developer's Toolkit PL/SQL Packages checkbox.

In the DBA Username field, enter:

`CTXSYS`

In the Password field, enter the password of the CTXSYS user.

Click the [Submit New Service] button.

The resulting page shows that the DCD has been created. This page may take some time to appear since packages are installed in the database.

### Configuring Oracle Web Listener

This section describes how to configure a Web Listener for use with the Configuration Manager. A physical file-system directory is mapped to a virtual directory to specify a root directory for the Configuration Manager. Security for the virtual directory is also configured.

1. Access the Oracle WebServer Administration page.
2. On the Oracle WebServer Administration page, select the <Oracle Web Listener> link.

The Oracle Web Listener Administration page appears.

3. On the Oracle Web Listener (OWL) Administration page, select <CONFIG-URE> for the listener used to access the Configuration Manager.

The OWL Administration - Server Advanced Configuration page appears.

4. On the OWL Administration - Server Advanced Configuration page, scroll down to the Directory Mappings block of fields to insert values into the first empty row.

In the File-System Directory column, enter the physical path:

```
%ORACLE_HOME%/ctxw/middle/cfgmgr/
```

In the Virtual Directory column, enter:

```
/ctxw_cm/
```

Click [Modify Listener].

The Oracle Web Listener Administration page appears.

5. On the Oracle Web Listener (OWL) Administration page, select <CONFIG-URE> for the listener used to access the Configuration Manager.

The OWL Administration - Server Advanced Configuration page appears.

6. On the OWL Administration - Server Advanced Configuration page, select <Security: Access Control and Encryption>.

The Oracle Web Listener Security page appears.

7. On the Oracle Web Listener Security page, scroll to the Basic Users block of fields.

In the User Name column of a blank row, enter:

```
ctxsys
```

In the Password field, enter the CTXSYS password, scroll to the Basic Groups block of fields.

In the Group Name column, enter:

```
ctxdba_group
```

In the User(s) column, enter:

```
ctxsys
```

Scroll to the Basic Realms block of fields. In the Realms column, enter:

`ctxdba_realm`

In the Group(s) column, enter:

`ctxdba_group`

Scroll to the Protection block of fields. In the Virtual Path column, enter:

`/ctx_dba/owa/`

In the Basic/Digest column, select Basic.

In the Realm column, enter:

`ctxdba_realm`

Click [Modify Listener].

The Oracle Web Listener Security page reappears.

Scroll to the end of the page and click [Administration].

### Configuring Oracle Web Request Broker

This section describes how to configure the Web Request Broker for a particular Web Listener for use with the Configuration Manager. The PL/SQL Agent is mapped to the virtual path of the Configuration Manager, and security for the PL/SQL Agent is configured.

1. Access the Oracle WebServer Administration page.
2. On the Oracle WebServer Administration page, select the <Web Request Broker> link.

The Web Request Broker Administration page appears.

3. On the Web Request Broker Administration page, select <Modify> for the listener used to access the Configuration Manager.

The WRB Cartridge Administration Page appears.

4. On the WRB Cartridge Administration Page, scroll to the Applications and Directories block of fields.

In the Virtual Path column, enter:

`/ctx_dba/owa`

In the App. column, enter:

OWA

In the Physical Path column, enter:

%ORACLE\_HOME%/ows21/bin

Scroll to the Protecting Applications block of fields. In the Virtual Path column, enter:

/ctx\_dba/owa/

In the Basic/Digest column, select Basic.

In the Realm column, enter:

ctxdba\_realm

Click the [Modify WRB Configuration] button.

The configuration of Oracle WebServer for use with the ConText Configuration Manager is now complete.

Before using the Configuration Manager, restart Oracle WebServer.

## Deconfiguring Oracle WebServer 2.1

This section describes how to remove the Configuration Manager configurations from Oracle WebServer 2.1.

---

---

**Note:** Some steps instruct the user to remove a particular value from a field. This can be done by placing the cursor at the start of the data in the field and holding down the Delete key until all the data has been removed. Alternatively, it may be possible to double-click on a value then press Delete.

For any pull-down menu fields, select a blank entry from the pull-down. If there is no blank entry, select the top entry.

Other steps instruct the user to remove a certain row from a block of fields. This means removing the value from each field in the row.

---

---

### Deconfiguration Requirements

The following information is required during deconfiguration steps:

- the URL of the WebServer Manager Home Page, usually:  
`http://<hostname>:8888/ows-adoc/Intro.html`
- the name of the listener configured to access the Configuration Manager

---

---

**Note:** In certain steps of the deconfiguration process for Oracle WebServer 2.1, it is necessary to remove filesystem paths. Part of the path may be the Oracle home directory, which is represented in the configuration steps as %ORACLE\_HOME%. The actual value of ORACLE\_HOME should be used instead.

The following example assumes an Oracle home directory of D:\ORANT on Windows NT. If the step asks you to remove the following value from a field:

`%ORACLE_HOME%/ctxw/middle/cfgmgr/`

then remove the value:

`D:\ORANT\ctxw\middle\cfgmgr\`

---

---



## Deconfiguration Tasks

There are three major tasks in the deconfiguration procedure:

- Removing the DCD
- Deconfiguring Oracle Web Listener
- Deconfiguring Oracle Web Request Broker

---

---

**Note:** Oracle WebServer must be restarted once deconfiguration has completed.

---

---

### Removing the DCD

This section describes how to remove the Database Connection Descriptor (DCD) called CTX\_DBA. It does not remove packages from the database account associated with the DCD.

1. Access the Oracle WebServer Administration page.
2. On the Oracle WebServer Administration page, select the <PL/SQL Agent> link.

The PL/SQL Agent Administration page appears.

3. On the PL/SQL Agent Administration page, select the <CTX\_DBA> link.

The Modify DCD (Database Connection Descriptor) page appears.

4. On the Modify DCD (Database Connection Descriptor) page, click the [Delete Service] button.

The Delete DCD (Database Connection Descriptor) page appears.

5. On the Delete DCD (Database Connection Descriptor) page, click the [Confirm Delete] button.

The resulting page shows that the CTX\_DBA DCD has been deleted.

### Deconfiguring Oracle Web Listener

This section describes how to remove the ConText Configuration Manager configuration from the Web Listener used to access the Configuration Manager.

1. Access the Oracle WebServer Administration page.
2. On the Oracle WebServer Administration page, select the <Oracle Web Listener> link.

The Oracle Web Listener Administration page appears.

3. On the Oracle Web Listener (OWL) Administration page, select <CONFIG-URE> for the listener which is used to access the Configuration Manager.

The OWL Administration - Server Advanced Configuration page appears.

4. On the OWL Administration - Server Advanced Configuration page, scroll down to the Directory Mappings block of fields.

Remove all values from the row that has the following value in the File-System Directory column:

```
%ORACLE_HOME%/ctxw/middle/cfgmgr/
```

Click [Modify Listener].

The Oracle Web Listener Administration page appears.

5. On the Oracle Web Listener (OWL) Administration page, select <CONFIG-URE> for the listener which is used to access the Configuration Manager.

The OWL Administration - Server Advanced Configuration page appears.

6. On the OWL Administration - Server Advanced Configuration page, select <Security: Access Control and Encryption>.

The Oracle Web Listener Security page appears.

7. On the Oracle Web Listener Security page, scroll to the Basic Users block of fields.

Remove all values from the row that has the following value in the User Name column:

```
ctxsys
```

Scroll to the Basic Groups block of fields.

Remove all values from the row that has the following value in the Group Name column:

```
ctxdba_group
```

Scroll to the Basic Realms block of fields.

Remove all values from the row that has the following value in the Realms column:

```
ctxdba_realm
```

Scroll to the Protection block of fields.

Remove all values from the row that has the following value in the Virtual Path column:

`/ctx_dba/owa/`

Click [Modify Listener].

The Oracle Web Listener Security page reappears.

Scroll to the end of the page and click [Administration].

### **Deconfiguring Oracle Web Request Broker**

This section describes how to remove the ConText Configuration Manager configuration from the Web Request Broker of the particular Web Listener used to access the Configuration Manager.

1. Access the Oracle WebServer Administration page.
2. On the Oracle WebServer Administration page, select the <Web Request Broker> link.

The Web Request Broker Administration page appears.

3. On the Web Request Broker Administration page, select <Modify> for the listener which is used to access the Configuration Manager.

The WRB Cartridge Administration Page appears.

4. On the WRB Cartridge Administration Page, scroll to the Applications and Directories block of fields.

Remove all values from the row that has the following value in the Virtual Path column:

`/ctx_dba/owa`

Scroll to the Protecting Applications block of fields. Remove all values from the row that has the following value in the Virtual Path column:

`/ctx_dba/owa/`

Click the [Modify WRB Configuration] button.

The deconfiguration of Oracle WebServer 2.1 for use with the ConText Configuration Manager is now complete.

## Configuring Oracle Web Application Server 3.0 for the Configuration Manager

Configuration of Oracle Web Application Server 3.0 for use with the Configuration Manager is normally performed through prompts provided by the Oracle Installer during installation of the ConText Workbench.

In the event that the Web Application Server configuration portion of the Configuration Manager installation fails, manual configuration is necessary to set up Web Application Server for use with the ConText Configuration Manager.

---

---

**Note:** These procedures are different from those that apply for Oracle WebServer 2.1.

---

---

The manual configuration is performed through a browser, such as Netscape Navigator or Microsoft Internet Explorer. The browser should have forms and frame capabilities.

Deconfiguration procedures can also be applied to remove the Configuration Manager from Oracle Web Application Server 3.0.

**See Also:** For more information about deconfiguring Oracle WebServer 3.0 to remove the Configuration Manager, see “Deconfiguring Oracle Web Application Server 3.0” in this chapter.

## Configuration Requirements

The following information is required during configuration steps:

- the name of the machine on which Oracle Web Application Server is running
- either the SID or Connect String of the database on which ConText is installed
- the password of the CTXSYS user
- the URL of the Web Application Server Manager Home Page, usually:  
`http://<hostname>:8888/ows-adoc/Intro.html`
- the name of the listener that used to access the Configuration Manager
- the port number of the listener used to access the Configuration Manager
- The type of platform that Oracle Web Application Server is running on (UNIX-based or Windows NT)

- the owner and group of the admin listener
- the Oracle home directory of the Oracle Web Application Server 3.0 installation

---



---

**Note:** In certain steps of the configuration process for Oracle Web Application Server 3.0, it is necessary to enter filesystem paths. A portion of the path may be the Oracle or Oracle Web Application Server home directory, which is represented in the configuration steps as either %ORACLE\_HOME% or %ORAWEB\_HOME%.

*Do not* substitute the real values unless explicitly instructed to do so.

---



---

## Configuration Tasks

There are five major tasks in the configuration procedure:

- Creating a Database Access Descriptor
- Creating a PL/SQL Agent
- Configuring the Oracle Web Listener
- Configuring the Oracle Web Request Broker
- Configuring Authorization Server

---



---

**Note:** Oracle Web Application Server must be restarted once configuration has completed.

---



---

### Creating a Database Access Descriptor

This section describes the creation of a new Database Access Descriptor (DAD) which provides the PL/SQL Agent with information used when connecting to a database. The DAD is used by the Configuration Manager to process administration requests from a browser.

1. Access the Oracle Web Application Server Administration page.
2. On the Oracle WebServer Administration page, select the <Oracle Web Application Server> link.

The Oracle Web Application Server Administration page appears.

Select the <DAD Administration> link.

The Database Access Descriptor Administration page appears.

3. On the Database Access Descriptor Administration page, select the <Create New DAD> link.

The Database Access Descriptor Creation page appears.

4. On the Database Access Descriptor Creation page, in the DAD Name field, enter:

`ctx_dba`

In the Database User, enter:

`CTXSYS`

Click the Identified by Password radio button.

In the Database User Password and Confirm Password fields, enter the password of the CTXSYS user.

If the database on which ConText is installed is a local database, enter the ORACLE\_SID in the ORACLE\_SID field; otherwise, enter a connect string in the SQL\*Net V2 Service field.

Click the Store The User Name And Password in the DAD checkbox.

In the DBA Username field, enter:

`CTXSYS`

In the Password field, enter the password of the CTXSYS user.

Click the [Submit New Service] button.

The resulting page shows that the DAD was created.

### Creating a PL/SQL Agent

This section describes the creation of a new PL/SQL Agent configuration for use with the Configuration Manager. It also installs the Web Application Server Developer's Toolkit packages to the database.

1. Access the Oracle Web Application Server Administration page.
2. On the Oracle WebServer Administration page, select the <Oracle Web Application Server> link.

The Oracle Web Application Server Administration page appears.

3. On the Oracle Web Application Server Administration page, select the <Cartridge Administration> link.

The Cartridge Administration page appears.

4. On the Cartridge Administration page, select the <PLSQL Cartridge> link.

The PL/SQL Agent Administration page appears.

5. On the PL/SQL Agent Administration page, select the <Create New PL/SQL Agent> link.

The PL/SQL Agent Administration Service Creation page appears.

6. On the PL/SQL Agent Administration Service Creation page, in the Name Of PL/SQL Agent field, enter:

`ctx_dba`

In the Name Of DAD field, select:

`ctx_dba`

In the Authorized Ports field, enter the port number of the listener used to access the Configuration Manager.

In the DAD Username field, enter:

`CTXSYS`

In the DAD Password field, enter the password of the CTXSYS user.

Select the Install Web Application Server Developer's Toolkit PL/SQL Packages checkbox.

If the database is remote to the Oracle Web Application Server host OR if the database is local and admin listener owner is NOT a DBA then:

Enter the CTXSYS and password in the DBA Username and Password fields.

Click the [Submit New Agent] button.

The resulting page shows that the PL/SQL Agent was created. This may take time as packages are installed in the database.

## Configuring the Oracle Web Listener

This section describes how to configure a Web Listener for use with the Configuration Manager. A physical filesystem directory is mapped to a virtual directory to specify a root directory for the Configuration Manager. Security for the virtual directory is also configured.

1. Access the Oracle WebServer Administration page.
2. On the Oracle WebServer Administration page, select the <Oracle Web Listener> link.

The Oracle Web Listener Administration page appears.

3. On the Oracle Web Listener (OWL) Administration page, select <CONFIGURE> for the listener used to access the Configuration Manager.

The OWL Administration - Server Advanced Configuration page appears.

4. On the OWL Administration - Server Advanced Configuration page, scroll down to the Directory Mappings block of fields to insert values into the first empty row.

In the File-System Directory column, enter the physical path (substitute real value of the Oracle home directory):

```
%ORACLE_HOME%/ctxw/middle/cfgmgr/
```

In the Virtual Directory column, enter:

```
/ctxw_cm/
```

Click [Modify Listener].

The Oracle Web Listener Administration page appears.

5. On the OWL Administration page, select <CONFIGURE> for the listener used to access the Configuration Manager.

The OWL Administration - Server Advanced Configuration page appears.

6. On the OWL Administration - Server Advanced Configuration page, select the <Security> link.

The Oracle Web Listener Security page appears.

7. On the Oracle Web Listener Security page, scroll to the Basic Users block of fields.

In the User Name column of a blank row, enter:



`ctxsys`

In the Password field, enter the CTXSYS password.

Scroll to the Basic Groups block of fields. In the Group Name column, enter:

`ctxdba_group`

In the User(s) column, enter:

`ctxsys`

Scroll to the Basic Realms block of fields. In the Realms column, enter:

`ctxdba_realm`

In the Group(s) column, enter:

`ctxdba_group`

Scroll to the Protection block of fields. In the Virtual Path column, enter:

`/ctx_dba/owa/`

In the Basic/Digest column, select Basic.

In the Realm column, enter:

`ctxdba_realm`

Click [Modify Listener].

The Oracle Web Listener Security page reappears.

Scroll to the end of the page and click [Administration].

### **Configuring the Oracle Web Request Broker**

This section describes how to configure the Web Request Broker for a particular Web Listener for use with the Configuration Manager. The PL/SQL Agent is mapped to the virtual path of the Configuration Manager, and security for the PL/SQL Agent is configured.

1. Access the Oracle WebServer Administration page.
2. On the Oracle WebServer Administration page, select the <Oracle Web Application Server> link.

The Oracle Web Application Server Administration page appears.

3. On the Oracle Web Application Server Administration page, select the <Cartridge Administration> link.

The Cartridge Administration Page appears.

Select the <Cartridge Summary (Web Request Broker)> link.

The Web Request Broker Administration page appears.

4. On the Web Request Broker Administration page, scroll to the Applications and Directories block of fields.

In the Virtual Path column, enter:

```
/ctx_dba/owa
```

In the App. column, enter:

```
PLSQL
```

In the Physical Path column, enter:

```
%ORAWEB_HOME%/bin
```

Scroll to the Protecting Applications block of fields. In the Virtual Path column, enter:

```
/ctx_dba/owa/
```

In the Scheme column, select Basic.

In the Realm column, enter:

```
ctxdba_realm
```

Click the [Modify WRB Configuration] button.

The resulting page shows that the update of Web Request Broker was successful.

### Configuring Authorization Server

This section describes how to configure the Authorization Server for use with the Configuration Manager. It provides authentication for Web Application Server cartridges to authenticate users before allowing them to use the Configuration Manager.

1. Access the Oracle WebServer Administration page.

2. On the Oracle Web Application Server Administration page, select the <Oracle Web Application Server> link.

The Oracle Web Application Server Administration page appears.

Select the <Authorization Server> link.

The Authentication Configuration page appears.

3. On the Authentication Configuration page, select the <Basic> link.

The Basic Authentication page appears.

4. On the Basic Authentication page, scroll to the Basic Users block of fields.

In the User Name column, enter:

`ctxsys`

In the Password column, enter the password of the CTXSYS user.

Scroll to the Basic Groups block of fields. In the Group Name column, enter:

`ctxdba_group`

In the User(s) column, enter:

`ctxsys`

Scroll to the Basic Realms block of fields. In the Realms column, enter:

`ctxdba_realm`

In the Group(s) column, enter:

`ctxdba_group`

Click the [Modify] button.

The resulting page shows that the Basic Authentication was updated.

The configuration of Oracle Web Application Server for use with the Configuration Manager is now complete.

## Deconfiguring Oracle Web Application Server 3.0

This section describes how to remove the Configuration Manager configurations from Oracle Web Application Server 3.0.

---

---

**Note:** Some steps instruct the user to remove a particular value from a field. This can be done by placing the cursor at the start of the data in the field and holding down the Delete key until all the data has been removed. Alternatively, it may be possible to double-click on a value then press Delete.

For any pull-down menu fields, select a blank entry from the pull-down. If there is no blank entry, select the top entry.

Other steps instruct the user to remove a certain row from a block of fields. This means removing the value from each field in the row.

---

---

### Deconfiguration Requirements

The following information is required during deconfiguration steps:

- the URL of the Web Application Server Manager Home Page, usually:  
`http://<hostname>:8888/ows-adoc/Intro.html`
- the name of the listener configured to access the Configuration Manager.

### Deconfiguration Tasks

There are five major tasks in the deconfiguration procedure:

- Deleting the Database Access Descriptor
- Deleting the PL/SQL Agent
- Deconfiguring the Oracle Web Listener
- Deconfiguring the Oracle Web Request Broker
- Deconfiguring Authorization Server

---

---

**Note:** Oracle Web Application Server must be restarted once configuration has completed.

---

---

### Deleting the Database Access Descriptor

This section describes how to remove the Database Access Descriptor (DAD) called `ctx_dba`.

1. Access the Oracle Web Application Server Administration page.
2. On the Oracle WebServer Administration page, select the <Oracle Web Application Server> link.

The Oracle Web Application Server Administration page appears.

Select the <DAD Administration> link.

The Database Access Descriptor Administration page appears.

3. On the Database Access Descriptor Administration page, select the <ctx\_dba> link.

The Modify DAD (Database Access Descriptor) page appears.

4. On the Modify DAD (Database Access Descriptor) page, click the [Delete DAD] button.

The Delete DAD (Database Access Descriptor) page appears.

Click the [Confirm Delete] button.

The resulting page shows that the `ctx_dba` DAD was deleted.

### Deleting the PL/SQL Agent

This section describes how to delete the PL/SQL Cartridge Agent used by the Configuration Manager.

1. Access the Oracle Web Application Server Administration page.
2. On the Oracle WebServer Administration page, select the <Oracle Web Application Server> link.

The Oracle Web Application Server Administration page appears.

3. On the Oracle Web Application Server Administration page, select the <Cartridge Administration> link.

The Cartridge Administration page appears.

4. On the Cartridge Administration page, select the <PLSQL Cartridge> link.

The PL/SQL Agent Administration page appears.

5. On the PL/SQL Agent Administration page, select the <Create New PL/SQL Agent> link.

The PL/SQL Agent Administration Service Creation page appears.

6. On the PL/SQL Agent Administration Service Creation page, select the <ctx\_dba> link.

The Modify PL/SQL Cartridge Agent (PLSQL) page appears.

7. On the Modify PL/SQL Cartridge Agent (PLSQL) page, click the [Delete Agent] button.

The Delete PL/SQL Agent (PLSQL) page appears.

8. On the Delete PL/SQL Agent (PLSQL) page, click [Confirm Delete].

The resulting page shows that the ctx\_dba PL/SQL Agent has been deleted.

### Deconfiguring the Oracle Web Listener

This section describes how to remove the ConText Configuration Manager configuration from the Web Listener used to access the Configuration Manager.

1. Access the Oracle WebServer Administration page.
2. On the Oracle WebServer Administration page, select the <Oracle Web Listener> link.

The Oracle Web Listener Administration page appears.

3. On the Oracle Web Listener (OWL) Administration page, select <CONFIGURE> for the listener which is used to access the Configuration Manager.

The OWL Administration - Server Advanced Configuration page appears.

4. On the OWL Administration - Server Advanced Configuration page, scroll down to the Directory Mappings block of fields.

Remove all values from the row that has the following value in the File-System Directory column:

```
%ORACLE_HOME%/ctxw/middle/cfgmgr/
```

Click [Modify Listener].

The Oracle Web Listener Administration page appears.

5. On the Oracle Web Listener (OWL) Administration page, select <CONFIGURE> for the listener which is used to access the Configuration Manager.

The OWL Administration - Server Advanced Configuration page appears.

6. On the OWL Administration - Server Advanced Configuration page, select <Security>.

The Oracle Web Listener Security page appears.

7. On the Oracle Web Listener Security page, scroll to the Basic Users block of fields.

Remove all values from the row that has the following value in the User Name column:

`ctxsys`

Scroll to the Basic Groups block of fields.

Remove all values from the row that has the following value in the Group Name column:

`ctxdba_group`

Scroll to the Basic Realms block of fields.

Remove all values from the row that has the following value in the Realms column:

`ctxdba_realm`

Scroll to the Protection block of fields.

Remove all values from the row that has the following value in the Virtual Path column:

`/ctx_dba/owa/`

Click [Modify Listener].

The Oracle Web Listener Security page reappears.

Scroll to the end of the page and click [Administration].

## **Deconfiguring the Oracle Web Request Broker**

This section describes how to remove the ConText Configuration Manager configuration from the Web Request Broker.

1. Access the Oracle Web Application Server Administration page.

2. On the Oracle Web Application Server Administration page, select the <Cartridge Administration> link.

The Cartridge Administration page appears.

3. On the Cartridge Administration page, select the <Cartridge Summary (Web Request Broker)> link.

The Web Request Broker Administration page appears.

4. On the Web Request Broker Administration page, scroll to the Applications and Directories block of fields.

Remove all values from the row that has the following value in the Virtual Path column:

`/ctx_dba/owa`

Scroll to the Protecting Applications block of fields.

Remove all values from the row that has the following value in the Virtual Path column:

`/ctx_dba/owa/`

Click the [Modify WRB Configuration] button.

The WRB Cartridge Administration page reappears.

### **Deconfiguring Authorization Server**

This section describes how to remove the Configuration Manager configuration from the Authorization Server.

1. Access the Oracle WebServer Administration page.
2. On the Oracle Web Application Server Administration page, select the <Oracle Web Application Server> link.

The Oracle Web Application Server Administration page appears.

Select the <Authorization Server> link.

The Authentication Configuration page appears.

3. On the Authentication Configuration page, select the <Basic> link.

The Basic Authentication page appears.

4. On the Basic Authentication page, scroll to the Basic Users block of fields.



Remove all values from the row that has the following value in the User Name column:

`ctxsys`

Scroll to the Basic Groups block of fields.

Remove all values from the row that has the following value in the Group Name column:

`ctxdba_group`

Scroll to the Basic Realms block of fields.

Remove all values from the row that has the following value in the Realms column:

`ctxdba_realm`

Click the [Modify] button.

The Oracle Web Application Server Authentication Configuration appears.



---

# Index

## A

---

- administration tools
  - 32-bit Windows, 1-3
  - Web-based, 1-3
- Authorization Server
  - configuring, A-20, B-20
  - deconfiguring, A-25, B-26

## C

---

- C routines
  - in I/O Utility dll, 3-19
- cartridges
  - Viewer Cartridge, 2-9
  - Web Request Broker Cartridge, A-6, A-17
- command-line syntax
  - I/O Utility, 3-3
- commands
  - executing in I/O Utility, 3-8
- components
  - ConText Workbench, 1-2
- configuration files
  - for Viewer Cartridge, 2-9
- Configuration Manager, 1-3
  - manually configuring Oracle Web Application Server 3.0 for, B-14
  - manually configuring Oracle WebServer 2.1 for, B-4
  - manually deconfiguring Oracle Web Application Server for, B-22
  - manually deconfiguring Oracle WebServer 2.1 for, B-10
  - manually installing, B-2

- configuring
  - Authorization Server, A-20, B-20
  - Oracle Web Listener, A-4, A-15, B-6, B-18
  - Oracle Web Request Broker, A-7, B-8, B-19
  - Viewer Cartridge parameters, A-19
- ConText Workbench
  - components, 1-2
  - related publications, x
- controls
  - custom, 1-5
- conventions
  - notational, xii, A-2, B-3
- creating
  - Database Access Descriptors (DADs), B-15
  - Database Connection Descriptors (DCDs), B-5
  - PL/SQL Agents, B-16, B-23
  - Web Request Broker cartridges, A-6, A-17
- CTXIO32 I/O utility, 3-2
- CTXQUERY sample application, 1-7, 4-2
  - architecture, 4-5
  - setting up, 4-3
- CTXWMG\_MIGRATE package
  - MIGRATE, 5-15
  - MIGRATE\_ALL\_SYNS, 5-16
  - MIGRATE\_ALL\_THES, 5-17
  - MIGRATE\_SG, 5-18
  - MIGRATE\_SYN, 5-19
  - MIGRATE\_THES, 5-20
- custom controls, 1-5
- customer support
  - contacting, xii

## D

---

- Data Store preferences, 5-8
- Database Access Descriptors (DADs)
  - creating, B-15
  - deleting, B-23
- Database Connection Descriptors (DCDs)
  - creating, B-5
  - deleting, B-11
- deconfiguring
  - Authorization Server, A-25, B-26
  - Oracle Web Listener, A-11, A-22, B-11, B-24
  - Oracle Web Request Broker, A-13, B-13, B-25
  - Viewer Cartridge parameters, A-24
- deleting
  - Database Access Descriptors (DADs), B-23
  - Database Connection Descriptors (DCDs), B-11
  - Web Request Broker cartridges, A-12, A-24
- dll interface
  - I/O Utility, 3-19
- DML
  - example, 4-12
- document viewers, 1-4

## E

---

- embedded tags
  - in URLs, 2-7
- end tags
  - in I/O Utility parameter files, 3-4
- Engine preferences, 5-9
- error handling
  - in I/O Utility, 3-9
- examples
  - HIGHLIGHT procedure in Forms
    - application, 4-10
  - I/O Utility, 3-10
  - loading HTML text, 3-10
  - MIGRATE procedure, 5-11
  - query expression, 4-7
  - TextServer3 migration, 5-22
- executing
  - OS commands, 3-8

## F

---

- Filter preferences, 5-9

## H

---

- helper applications
  - URLs as, 2-6, 2-8
- HIGHLIGHT procedure
  - example, 4-11, 4-12
  - example in sample Forms application, 4-10
  - result tables in, 4-10
- HTML text
  - loading example, 3-10

## I

---

- I/O Utility, 1-6, 3-2
  - command-line syntax, 3-3
  - dll interface, 3-19
  - examples, 3-10
  - overview, 3-2
- I/O Utility parameter file, 3-4
- input/output
  - text, 1-6, 3-2
- installing
  - Configuration Manager, B-2
  - Migration Tool, 5-6
  - Viewer Cartridge, 2-5
  - Viewer Plugin file, 2-3

## J

---

- JDBC drivers, 2-11

## L

---

- limitations in Migration Tool
  - migrating section groups, 5-5
  - migrating SQEs, 5-4
  - migrating synonym rings, 5-5
  - migrating thesauri, 5-5
- link targets, 2-7

## M

---

### manual configuration

- Oracle Web Application Server 3.0 for Configuration Manager, B-14
- Oracle Web Application Server 3.0 for Viewer Cartridge, A-14
- Oracle WebServer 2.1 for Configuration Manager, B-4
- Oracle WebServer 2.1 for Viewer Cartridge, A-3

### manual deconfiguration

- Oracle Web Application Server 3.0 for Configuration Manager, B-22
- Oracle Web Application Server 3.0 for Viewer Cartridge, A-22
- Oracle WebServer 2.1 for Configuration Manager, B-10
- Oracle WebServer 2.1 for Viewer Cartridge, A-10

### MIGRATE procedure, 5-15

### MIGRATE\_ALL\_SYNS procedure, 5-16

### MIGRATE\_ALL\_THES procedure, 5-17

### MIGRATE\_SG procedure, 5-18

### MIGRATE\_SYN procedure, 5-19

### MIGRATE\_THES procedure, 5-20

### migration

- process, 5-3

### Migration Tool, 1-8

- editing migration scripts, 5-10
- generating migration scripts, 5-10
- installing, 5-6
- limitations, 5-4
- overview, 5-2
- requirements, 5-4

## N

---

### notational conventions

- general, xii

### npctxvpi.dll

- Viewer Plugin file, 2-3

## O

---

### Oracle Corporation

- customer support, xii

### Oracle Forms, 1-7

- for CTXQUERY sample application, 4-4

### Oracle JDBC driver, 2-4, 2-11

### Oracle Web Application Server 3.0

- manually configuring, A-14, B-14
- manually deconfiguring, A-22, B-22

### Oracle Web Listener

- configuring, A-4, A-15, B-6, B-18
- deconfiguring, A-11, A-22, B-11, B-24

### Oracle Web Request Broker

- configuring, A-7, B-8, B-19
- deconfiguring, A-13, B-13, B-25

### Oracle WebServer 2.1, 2-3

- DCD for, B-5
- manually configuring, A-3, B-4
- manually deconfiguring, A-10

### OS commands

- executing in I/O Utility, 3-8

### OS files

- importing into database, 3-4

## P

---

### parameter file

- advanced, 3-15
- for I/O Utility, 3-4
- simple, 3-10

### parameters

- EMBED tag, 2-7
- for configuration file sections, 2-10
- Viewer Plugin, 2-7

### PL/SQL Agent

- creating, B-16, B-23
- DCD for, B-5
- virtual paths for, B-8

### policies

- creating for migration, 5-8
- migration example, 5-11

### preface

- Send Us Your Comments, vii

### preferences

- creating for migration, 5-8
- Data Store, 5-8
- Engine, 5-9
- Filter, 5-9

- Lexer, 5-9
- Stop List, 5-10
- Word List, 5-10
- procedures
  - MIGRATE, 5-15
  - MIGRATE\_ALL\_SYNS, 5-16
  - MIGRATE\_ALL\_THES, 5-17
  - MIGRATE\_SG, 5-18
  - MIGRATE\_SYN, 5-19
  - MIGRATE\_THES, 5-20

## Q

---

- query expression
  - build example, 4-7
  - build procedure, 4-8
- query ID
  - in results tables, 4-5

## R

---

- realms
  - using, 2-8
- reindexing a document, 4-12
- remote machines
  - installing Viewer Cartridge on, 2-5
- requirements
  - Oracle Web Application Server 3.0
    - configuration, A-14, B-14
  - Oracle Web Application Server 3.0
    - deconfiguration, A-22, B-22
  - Oracle WebServer 2.1 configuration, A-3, B-4
  - Oracle WebServer 2.1 deconfiguration, A-10, B-10
- result tables
  - shared in CTXQUERY sample application, 4-5

## S

---

- samples
  - CTXQUERY Forms application, 1-7, 4-2
  - TextServer3 migration script, 5-22
- scripts
  - editing, 5-10
  - for migrating, 5-10

- migration example, 5-11
- section groups
  - migrating, 5-5, 5-13
- sections
  - in configuration files, 2-10
- security
  - in Viewer Plugin, 2-8
- Send Us Your Comments
  - boilerplate, vii
- setting up
  - CTXQUERY sample application, 4-3
- sharing result tables
  - Forms example, 4-5
- SQEs
  - migrating, 5-4
- SQL
  - calling in I/O utility, 3-4
- start tags
  - in I/O Utility parameter files, 3-4
- Stop List preferences, 5-10
- synonym rings
  - migrating, 5-5, 5-12
- System Administration tool, 1-3

## T

---

- tags
  - embedded, 2-7
- text
  - exporting to OS file, 3-4
  - importing from OS file, 3-4
  - input/output, 1-6, 3-2
  - loading, 3-15
  - updating, 3-6
- text tables
  - migrating, 5-8
- TextServer3 Dictionary Migration Tool, 1-8
  - overview, 5-2
- thesauri
  - migrating, 5-5, 5-12
- tools
  - Configuration Manager, 1-3
  - Migration Tool, 1-8
  - System Administration, 1-3

## U

---

### URLs

- as helper applications, 2-8
- for viewing ConText-indexed documents, 2-4
- specifying, 2-5
- using, 2-6

### using

- alternative JDBC driver, 2-11
- I/O Utility, 3-10
- URLs in Viewer Plugin, 2-6
- Viewer Cartridge, 2-4
- Viewer Control, 2-2

### utilities

- I/O, 1-6, 3-2

## V

---

### Viewer Cartridge, 1-4

- configuration files, 2-9
- configuring parameters for, A-19
- deconfiguring parameters for, A-24
- installing, 2-5
- introduction to, 2-4
- manually configuring Oracle Web Application Server 3.0, A-14
- manually configuring Oracle WebServer 2.1, A-3
- pre-installation requirements for, 2-4

### Viewer Control, 1-5

- registering, 2-2
- using, 2-2

### Viewer Plugin, 1-4

- activating, 2-3
- copying file manually, 2-3
- embedded tags, 2-7
- installing the file for, 2-3
- introduction to, 2-3
- security, 2-8

### viewers

- Viewer Cartridge, 1-4
- Viewer Control, 1-5
- Viewer Plugin, 1-4

### virtual paths

- for PL/SQL Agent, B-8

- in realms, 2-8

## W

---

### Web Request Broker cartridges

- creating, A-6
- deleting, A-12, A-24

### Windows 32-bit

- administration tool, 1-3
- I/O Utility, 3-2

### Word List preferences, 5-10

