

Oracle® Fusion Middleware

Administrator's Guide for Oracle Application Development
Framework

11g Release 1 (11.1.1)

E15470-02

April 2010

Oracle Fusion Middleware Administrator's Guide for Oracle Application Development Framework 11g
Release 1 (11.1.1)

E15470-02

Copyright © 2010, Oracle and/or its affiliates. All rights reserved.

Primary Authors: Peter Jew (Lead), Cammy Moore

Contributing Author: Odile Sullivan-Tarazi

Contributors: Lynn Munsinger, Duncan Mills, Dipankar Bajpai, Harry Hsu

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

This software and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Contents

Preface	v
Documentation Accessibility	v
Audience	v
Related Documents	vi
Conventions	vi
Part I Understanding Oracle ADF	
1 Introduction to Oracle ADF Administration	
1.1 Introducing Oracle ADF	1-1
1.2 Oracle ADF Architecture	1-1
1.2.1 ADF Business Components	1-2
1.2.2 ADF Model	1-2
1.2.3 ADF Controller.....	1-2
1.2.4 ADF Faces Rich Client.....	1-2
1.3 Administering Oracle ADF Applications.....	1-3
Part II Administering ADF Applications	
2 Deploying ADF Applications	
2.1 Introduction to Deploying ADF Applications.....	2-1
2.2 Deploying Using Oracle Enterprise Manager Fusion Middleware Control	2-2
2.3 Deploying Using WebLogic Scripting Tool (WLST) Commands	2-2
2.4 Deploying Using Scripts and Ant.....	2-2
2.5 Deploying Using Oracle WebLogic Server Administration Console.....	2-3
3 Monitoring and Configuring ADF Applications	
3.1 Introduction to ADF Application Monitoring and Configuration	3-1
3.2 Monitoring Performance Using Fusion Middleware Control.....	3-2
3.2.1 How to View Application Module Performance	3-2
3.2.2 How to view Application Module Pool Performance.....	3-3
3.2.3 How to View ADF Task Flow Performance	3-4
3.3 Configuring Application Properties Using Fusion Middleware Control.....	3-4
3.3.1 How to Modify ADF Business Components Parameters	3-5

3.3.2	How to Modify Connection Configurations.....	3-13
3.4	Configuring Application Properties Using the MBean Browser	3-20
3.4.1	How to Modify ADF Application Configuration Using MBean	3-20
3.4.2	How to Modify ADF Connections Using MBean	3-21
3.4.3	How to Modify ADF Business Components Configuration Using MBeans.....	3-23
3.4.4	How to Modify MDS Configuration Using MBean.....	3-23
3.5	Diagnosing Problems using the Diagnostic Framework	3-24

4 WLST Command Reference for ADF Applications

4.1	Overview of Custom WSLT Commands for Oracle ADF.....	4-1
4.2	Commands for ADF-based URL Connections.....	4-1
4.2.1	adf_createURLConnection	4-1
4.2.1.1	Description	4-2
4.2.1.2	Syntax	4-2
4.2.1.3	Example.....	4-2
4.2.2	adf_createURLConnection	4-2
4.2.2.1	Description	4-2
4.2.2.2	Syntax	4-2
4.2.2.3	Example.....	4-2
4.2.3	adf_setURLConnectionAttributes	4-2
4.2.3.1	Description	4-3
4.2.3.2	Syntax	4-3
4.2.3.3	Example.....	4-3
4.2.4	adf_listURLConnection	4-3
4.2.4.1	Description	4-3
4.2.4.2	Syntax	4-3
4.2.4.3	Example.....	4-3

Part III Appendices

A JDeveloper Runtime Libraries

A.1	adf.oracle.domain.webapp.war Library	A-1
A.2	adf.oracle.domain.ear Library	A-3
A.3	System Classpath	A-5

Preface

Welcome to *Administrator's Guide for Oracle Application Development Framework*.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible to all users, including users that are disabled. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at <http://www.oracle.com/accessibility/>.

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Deaf/Hard of Hearing Access to Oracle Support Services

To reach Oracle Support Services, use a telecommunications relay service (TRS) to call Oracle Support at 1.800.223.1711. An Oracle Support Services engineer will handle technical issues and provide customer support according to the Oracle service request process. Information about TRS is available at <http://www.fcc.gov/cgb/consumerfacts/trs.html>, and a list of phone numbers is available at <http://www.fcc.gov/cgb/dro/trsphonebk.html>.

Audience

This document is intended for system administrators who need to deploy, manage, monitor, and configure Oracle ADF applications using the Oracle Application Development Framework (Oracle ADF).

Related Documents

For more information, see the following documents:

Oracle Fusion Middleware Fusion Developer's Guide for Oracle Application Development Framework

Oracle Fusion Middleware Administrator's Guide

Oracle Fusion Middleware Web User Interface Developer's Guide for Oracle Application Development Framework

Oracle Fusion Middleware Developer's Guide for Oracle Application Development Framework Mobile

Oracle Fusion Middleware Desktop Integration Developer's Guide for Oracle Application Development Framework

Oracle Fusion Middleware Security Guide

Oracle Fusion Middleware WebLogic Scripting Tool Command Reference

Oracle Fusion Middleware High Availability Guide

Oracle JDeveloper 11g Online Help

Oracle JDeveloper 11g Release Notes, included with your JDeveloper 11g installation, and on Oracle Technology Network

Oracle Fusion Middleware Java API Reference for Oracle ADF Model

Oracle Fusion Middleware Java API Reference for Oracle ADF Controller

Oracle Fusion Middleware Java API Reference for Oracle ADF Lifecycle

Oracle Fusion Middleware Java API Reference for Oracle ADF Faces

Oracle Fusion Middleware JavaScript API Reference for Oracle ADF Faces

Oracle Fusion Middleware Java API Reference for Oracle ADF Data Visualization Components

Oracle Fusion Middleware Java API Reference for Oracle ADF Share

Oracle Fusion Middleware Java API Reference for Oracle ADF Business Components Browser

Oracle Fusion Middleware Java API Reference for Oracle Generic Domains

Oracle Fusion Middleware interMedia Domains Java API Reference for Oracle ADF Business Components

Oracle Fusion Middleware Java API Reference for Oracle Metadata Service (MDS)

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Part I

Understanding Oracle ADF

Part I contains the following chapters:

- [Chapter 1, "Introduction to Oracle ADF Administration"](#)

Introduction to Oracle ADF Administration

This chapter describes the administrative tasks you can perform and the tools you can use to deploy, manage, monitor, and configure applications developed for the Oracle Application Development Framework (Oracle ADF).

This chapter includes the following sections:

- [Section 1.1, "Introducing Oracle ADF"](#)
- [Section 1.2, "Oracle ADF Architecture"](#)
- [Section 1.3, "Administering Oracle ADF Applications"](#)

1.1 Introducing Oracle ADF

The Oracle Application Development Framework (Oracle ADF) builds on Java Platform, Enterprise Edition (Java EE) standards and open-source technologies to provide a complete framework for implementing service-oriented applications. You can use this framework to provide enterprise solutions across different platforms. You can build applications that search, display, create, modify, and validate data for web, web services, desktop, or mobile interfaces.

You use Oracle JDeveloper 11g with Oracle ADF to develop applications with an environment that supports the full development lifecycle of design, test, and deployment. For more information about ADF development, see *Oracle Fusion Middleware Fusion Developer's Guide for Oracle Application Development Framework*.

After you have developed and tested your ADF application in test environments, you can deploy your application to production environments using the tools described in this book. You can monitor the performance of applications as they are running. You can also manage and configure properties and attributes.

1.2 Oracle ADF Architecture

Oracle ADF supports the industry-standard model-view-controller architecture to achieve separation of business logic, navigation, and user interface. The MVC architecture provides:

- A model layer that represents the data values
- A view layer that contains the UI components
- A controller layer that handles input and navigation
- A business service layer that encapsulates business logic

The Fusion web application technology stack components are:

- ADF Model, for accessing declarative data binding metadata
- ADF Business Components, for building business services
- ADF Faces rich client, for AJAX-enabled UI components for web applications built with JavaServer Faces (JSF)
- ADF Controller, for input processing, navigation, and reusable task flows

1.2.1 ADF Business Components

ADF Business Components are application objects you can use to implement service-oriented Java EE applications. You implement ADF Business Components for clients to query, insert, update, and delete business data. You can apply business rules to the Business Components to enforce proper usage. The key components of ADF Business Components are the entity object, the view object, and the application module.

An *entity object* represents a row in a database table. It uses data manipulation language (DML) operations to modify data. Entity objects are used with others to reflect relationships in the database schema.

A *view object* represents a SQL query. You use the SQL Language to query the database to obtain the results. You can also link a view object with other entity objects to create master-detail hierarchies.

An *application module* is the transactional component that allows UI components to access data. It presents a data model and methods to perform certain tasks.

1.2.2 ADF Model

ADF Model implements a service abstraction called *data control*. Data control uses metadata interfaces to abstract business services. This metadata is used to describe data collections, properties, methods, and types. In JDeveloper, data controls appear in the Data Controls panel. When you drag and drop attributes, collections, and methods onto a page, JDeveloper automatically creates the bindings from the page to the associated services.

1.2.3 ADF Controller

ADF Controller provides a navigation and state management model that works with JSF. You can create navigational flows called task flows that encapsulate a specific task sequence.

1.2.4 ADF Faces Rich Client

ADF Faces provides over 100 rich components that can be used out of the box to create web applications. ADF Faces components provide built-in AJAX functionality to allow requests to be sent to the server without fully rendering the page. JSF provides server-side control to reduce the dependency on JavaScript. The components support skinning, internationalization, and accessibility options.

ADF Faces has a large set of components, including tables, trees, dialogs, accordions, and a variety of layout components. It also includes ADF Data Visualization components, which are Flash- and SVG-enabled, for displaying graphs, charts, and gauges.

1.3 Administering Oracle ADF Applications

You can perform a variety of administration tasks on ADF applications. You can deploy ADF applications using Enterprise Manager Fusion Middleware Control, WLST commands, the `ojdeploy` command, scripts, or the WebLogic Administration Console.

After the ADF application has been deployed, you can configure application properties using Enterprise Manager Fusion Middleware Control. You can also configure some properties using the MBean Browser to change values in the ADF MBeans. For example, you can use Enterprise Manager Fusion Middleware Control to change the URL connection or WebService connection endpoints or seed the production credentials.

When you run the application, you can monitor performance data on the application modules, application module pooling, and task flows.

Part II

Administering ADF Applications

Part I contains the following chapters:

- [Chapter 2, "Deploying ADF Applications"](#)
- [Chapter 3, "Monitoring and Configuring ADF Applications"](#)
- [Chapter 4, "WLST Command Reference for ADF Applications"](#)

Deploying ADF Applications

This chapter describes how to deploy Oracle ADF applications packaged as an EAR file to a target application server. It also describes how to use scripts and Ant to automate the deployment process. This chapter focuses on deploying ADF applications for production and later stage testing. For information about deploying ADF applications for development, see the *Oracle Fusion Middleware Developer's Guide for Oracle Application Development Framework*.

This chapter includes the following sections:

- [Section 2.1, "Introduction to Deploying ADF Applications"](#)
- [Section 2.2, "Deploying Using Oracle Enterprise Manager Fusion Middleware Control"](#)
- [Section 2.3, "Deploying Using WebLogic Scripting Tool \(WLST\) Commands"](#)
- [Section 2.4, "Deploying Using Scripts and Ant"](#)
- [Section 2.5, "Deploying Using Oracle WebLogic Server Administration Console"](#)

2.1 Introduction to Deploying ADF Applications

Deployment is the process of packaging application files and artifacts and transferring them to a target application server to be run. During application development using JDeveloper, developers can test the application using the Integrated WebLogic Server that is built into the JDeveloper installation, or they can use JDeveloper to directly deploy to a standalone WebLogic Server.

After the application has been developed, administrators can deploy the application to production WebLogic Servers. The tools that the administrators use for production-level deployment are:

- Oracle Enterprise Manager Fusion Middleware Control
- WebLogic Scripting Tool (WLST) commands
- Command scripts and Ant scripts
- Oracle WebLogic Administration Console

This chapter describes the tools and methods that administrators use to deploy ADF applications. For information about deploying ADF applications for development and testing purposes using JDeveloper, see the *Oracle Fusion Middleware Developer's Guide for Oracle Application Development Framework*.

If your application uses customization, you may need to set up the MDS repository in Oracle WebLogic Server. For more information about MDS, see the *Oracle Fusion Middleware Administrator's Guide*.

Note: Developers, Test, and QA personnel may also use these tools and the methods in this chapter to deploy ADF applications to staging WebLogic Servers.

2.2 Deploying Using Oracle Enterprise Manager Fusion Middleware Control

You can use Oracle Enterprise Manager Fusion Middleware Control to deploy the EAR file created in JDeveloper. Fusion Middleware Control is a Web browser-based, graphical user interface that you can use to monitor and administer a farm. For more information about deploying using Fusion Middleware Control, see the *Oracle Fusion Middleware Administrator's Guide*.

2.3 Deploying Using WebLogic Scripting Tool (WLST) Commands

Applications or modules can be deployed from JDeveloper without starting the JDeveloper IDE. You can run WLST commands from the command line or sequence them in scripts to run as a batch.

Before deploying from the command line, there must be deployment profiles for the application (EAR) or project (JAR or WAR). JDeveloper creates these deployment profiles automatically for certain types of applications, but before using WLST commands for deployment, it is important to verify that the deployment profile(s) exist. To verify that the profiles exist, choose the **Deployment** node from either the Application Properties or Project Properties dialogs in JDeveloper. For more information about deployment profiles, see the *Oracle Fusion Middleware Developer's Guide for Oracle Application Development Framework*.

JDeveloper can also be used to deploy an application's EAR, WAR, or JAR files. The same WLST scripts that are used for deployment via a command line are also used to deploy via JDeveloper, but JDeveloper creates the WLST syntax and provides a user interface for the deployment.

There are specific WLST commands for working with ADF applications. For a list of these commands, see [Chapter 4, "WLST Command Reference for ADF Applications."](#)

For more information about using WLST scripts, see the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

2.4 Deploying Using Scripts and Ant

You can deploy the application using commands and scripts. You create a script to deploy the application using the `ojdeploy` command and use the `ojaudit` command to audit projects, workspaces, or source files of the application. You can also set up the script to run automatically, for instance, whenever a developer checks in new changes.

`ojdeploy` scripts and Ant scripts can be used together or separately:

1. Create an `ojdeploy` script to compile, package, and deploy the application.
2. Create an `ojdeploy` script to compile and package the application. Then use an Ant script (such as `WLDeploy`) to deploy the application.
3. Create an Ant script to compile, package, and deploy the application. The Ant does not need to use `ojdeploy`.

For more information about the `ojdeploy` and `ojaudit` commands, see the JDeveloper online help.

You can deploy to most application servers from JDeveloper, or use tools provided by the application server vendor. You may also use Ant to package and deploy applications. The `build.xml` file, which contains the deployment commands for Ant, may vary depending on the target application server.

For deployment to other application servers, see the application server's documentation. If your application server does not provide specific Ant tasks, you may be able to use generic Ant tasks. For example, the generic `ear` task creates an EAR file for you.

For information about Ant, see <http://ant.apache.org>.

2.5 Deploying Using Oracle WebLogic Server Administration Console

You can use the Oracle WebLogic Server Administration Console to deploy the EAR file created in JDeveloper. For more information, see *Oracle Fusion Middleware Deploying Applications to Oracle WebLogic Server*.

Monitoring and Configuring ADF Applications

This chapter describes how to monitor ADF application performance. It also describes how to configure an ADF application's properties after it has been deployed to Oracle WebLogic Server.

This chapter includes the following sections:

- [Section 3.1, "Introduction to ADF Application Monitoring and Configuration"](#)
- [Section 3.2, "Monitoring Performance Using Fusion Middleware Control"](#)
- [Section 3.3, "Configuring Application Properties Using Fusion Middleware Control"](#)
- [Section 3.4, "Configuring Application Properties Using the MBean Browser"](#)

3.1 Introduction to ADF Application Monitoring and Configuration

After you have deployed an ADF application to Oracle WebLogic Server, you can monitor the application performance and configure application properties on the server. You can use Enterprise Manager Fusion Middleware Control to perform these tasks.

Enterprise Manager Fusion Middleware Control offers a user interface for the performance tasks. Some configuration tasks can be performed either from a user interface or by configuring an MBean, as listed in [Table 3-1](#).

Table 3-1 Configuration Tasks Using Fusion Middleware Control

Configuration tasks	Fusion Middleware Control UI	Fusion Middleware Control MBean Browser
ADF Business Components	Section 3.3.1, "How to Modify ADF Business Components Parameters"	Section 3.4.3, "How to Modify ADF Business Components Configuration Using MBeans"
ADF connections	Section 3.3.2, "How to Modify Connection Configurations"	Section 3.4.2, "How to Modify ADF Connections Using MBean"
ADF application configuration		Section 3.4.1, "How to Modify ADF Application Configuration Using MBean"
Metadata Services (MDS)		Section 3.4.4, "How to Modify MDS Configuration Using MBean"

By default, the post-deployment changes made using MBeans are stored in MDS with a layer name of `adfshare` and a layer value of `adfshare`. You can provide a specific layer name by specifying the `adfAppUID` property in the application's `adf-config.xml`.

[Example 3-1](#) shows the `adf-properties-child` code in `adf-config.xml`.

Example 3-1 MDS Layers in the `adf-config.xml` File

```
<adf:adf-properties-child xmlns="http://xmlns.oracle.com/adf/config/properties">
  <adf-property name="adfAppUID" value="DeptApp.myApp"/>
</adf:adf-properties-child>
```

If you are moving data between MDS repositories (for example, from a test to a production system), use the MDS `exportMetadata` and `importMetadata` commands as described in the chapter on managing the Oracle metadata repository in the *Oracle Fusion Middleware Administrator's Guide* and in the chapter on Metadata Services custom WLST commands in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

3.2 Monitoring Performance Using Fusion Middleware Control

You can monitor the performance of Oracle ADF applications using the Enterprise Manager Fusion Middleware Control.

You can:

- View application module performance
- View application module pool performance
- View task flow performance

3.2.1 How to View Application Module Performance

You can view performance information about application modules. Application module components can be used to support a unit of work which spans multiple browser pages.

Before you begin:

You must already have deployed an Oracle ADF application and have Enterprise Manager Fusion Middleware Control available to access the application.

To view ADF application module performance:

1. Log in to an Oracle Fusion Middleware farm using Fusion Middleware Control.
2. Click the **Farm** tab.
3. Expand the *Farm_domain* node.
4. Expand the **Application Deployments** node and click a J2EE application deployment, for example, *StoreFrontModule (AdminServer)*.

After you select an application, the Application Deployment page displays.

5. Click **Application Deployment** and select **ADF > ADF Performance Summary** from the dropdown menu.

The ADF Performance Summary page displays. It contains subtabs for viewing performance information about active application module pools and task flows.

3.2.2 How to view Application Module Pool Performance

An *application module pool* is a collection of instances of a single application module type which are shared by multiple application clients. One application module pool is created for each root application module used by an ADF web application (ADF Business Components, ADF Controller, or ADF Faces) in each Java virtual machine where a root application module of that type is used by the ADF Controller layer.

Before you begin:

You must already have deployed an Oracle ADF application and have Enterprise Manager Fusion Middleware Control available to access the application.

To view application module pooling performance:

1. Log in to an Oracle Fusion Middleware farm using Fusion Middleware Control.
2. Click the **Farm** tab.
3. Expand the *Farm_domain* node.
4. Expand the **Application Deployments** node and click a J2EE application deployment, for example, *StoreFrontModule (AdminServer)*.

After you select an application, the Application Deployment page displays.

5. Click **Application Deployment** and select **ADF > ADF Performance Summary** from the dropdown menu.

The ADF Performance Summary page displays. It contains subtabs for viewing performance information about active Application Module Pools and Task Flows.

6. Click the **Application Module Pools** tab.
7. In the **Module** column, select an application module to display its details in the Application Module Pools table.

No Data Available displays in the Module column if an application has never run.

8. Click a module to display additional informations about the module, for example, Lifetime, State Management, Pool Use, and Application Module Pools Page.

Use the Application Module Pools page to display active application module pools, a collection of application module instances of the same type. The Application Module Pools page:

- Displays size and performance information about pool connections
- Specifies settings that affect how application module pools behave
- Specifies credential information for the application module pools

Element	Description
Module	Displays the active application module pool name, for example, <code>model.BugTest5PM</code> . Click a module to display additional information about it, for example, Lifetime, State Management, Pool Use, Application Module Pools page.
Requests	Displays the number of requests that were made for the application during the selected time interval.

Element	Description
Average Creation Time (ms)	Displays the average time (in milliseconds) required to complete a request for the application module pool.
Maximum Creation Time (ms)	Displays the longest time (in milliseconds) required to complete any of the requests for the application module pool.
Free Instances	Displays the number of available instances of the application module pool.

3.2.3 How to View ADF Task Flow Performance

You can view performance information about task flows. Task flows provide a modular and transactional approach to navigation and application control. Task flows mostly contain pages that will be viewed, but they also can contain activities that call methods on managed beans, evaluate an EL expression, or call another task flow, all without invoking a particular page.

Before you begin:

You must already have deployed an Oracle ADF application and have Enterprise Manager Fusion Middleware Control available to access the application.

To view task flow performance:

1. Log in to an Oracle Fusion Middleware farm using Fusion Middleware Control.
2. Click the **Farm** tab.
3. Expand the *Farm_domain* node.
4. Expand the **Application Deployments** node and click a J2EE application deployment, for example, *StoreFrontModule (AdminServer)*.

After you select an application, the Application Deployment page displays.

5. Click **Application Deployment** and select **ADF > ADF Performance Summary** from the dropdown menu.

The ADF Performance Summary page displays. It contains subtabs for viewing performance information about active application module pools and task flows.

6. Click the **Task Flows** tab.

By default, Task Flow Performance charts on the tab display data for the preceding 15 minutes. To set a different interval, click the time at the top of the page or move the slider to another interval, for example, from 08:00 AM to 08:30 AM.

7. Click **TF Charts**.
 - **Request Processing Time** displays the average request processing time for all ADF task flows that execute during the selected interval.
 - **Active Task Flows** displays the number of active instances of each ADF task flow during the selected interval.

3.3 Configuring Application Properties Using Fusion Middleware Control

You can use Enterprise Manager Fusion Middleware Control to configure ADF application configuration parameters. These configuration parameters are stored in ADF MBeans. Fusion Middleware Control provides a user interface to configure the

ADF Business Components and ADF Connections MBeans. You can also use the System MBean Browser to directly access the underlying MBeans and configure their values. For more information about accessing the underlying MBeans, see [Section 3.4, "Configuring Application Properties Using the MBean Browser."](#)

Fusion Middleware Control provides a user interface for you to:

- Configure ADF Business Component parameters
- Configure connection parameters

3.3.1 How to Modify ADF Business Components Parameters

You control the runtime behavior of an application module pool by setting appropriate configuration parameters. Fusion Middleware Control provides a UI to configure ADF Business Components, as described in this section. You can also configure the ADF Business Components MBeans directly using the generic MBean Browser, as described in [Section 3.4.3, "How to Modify ADF Business Components Configuration Using MBeans."](#)

Before you begin:

You must already have deployed an Oracle ADF application and have Enterprise Manager Fusion Middleware Control available to access the application.

To modify business components parameters:

1. Log in to an Oracle Fusion Middleware farm using Fusion Middleware Control.
2. Click the **Farm** tab.
3. Expand the *Farm_domain* node.
4. Expand the **Application Deployments** node and click a J2EE application deployment, for example, *StoreFrontModule (AdminServer)*.

After you select an application, the Application Deployment page displays.

5. Click **Application Deployment** and select **ADF > Configure ADF Business Components** from the dropdown menu.
6. Click an **Application Module**.
7. Click the **Pooling and Scalability**, **Core Properties**, **Database Properties**, or **Security Properties** tabs to update configuration parameters.

If the application module uses data sources, you can configure the data sources by clicking **Edit DataSource** from the **Core** tab.

The ADF Business Components configurations page is arranged with the following sections or tabs:

- Application Modules section
- Pooling and Scalability tab - Application Pool Properties
- Pooling and Scalability tab - Connection Pool Properties
- Core tab
- Database Properties tab
- Security Properties tab

Application Modules Section

In the Application Modules section, select the application module you want to configure.

Element	Description
Application Modules	Displays the active application module name. Click the module name to display the applications in the module.

Pooling and Scalability Tab - Application Pool Properties

In the Pooling and Scalability tab, select the application pool properties you want to configure.

Element	Description
AmpoolDoampooling	Select to enable application module pooling by default. Whenever you deploy your application in a production environment the default setting of <code>jbo.ampool.doampooling</code> is <code>true</code> and is the way you will run your application. But, as long as you run your application in a test environment, setting the property to <code>false</code> can play an important role in your testing. When this property is <code>false</code> , there is effectively no application pool.
AmpoolWritecookieclient	Select to write the <code>SessionCookie</code> value to the client browser.
AmpoolMaxavailablesize	Enter the maximum number of available application modules that should be referenced by an application pool. This is the ideal maximum number of available application module instances in the pool when not under abnormal load. When the pool monitor wakes up to do resource cleanup, it will try to remove available application module instances to bring the total number of available instances down to this ideal maximum. Instances that have been not been used for a period longer than the idle instance timeout will always get cleaned up at this time, and then additional available instances will be removed, if necessary to bring the number of available instances down to this size. The default maximum available size is 25 instances. Configure this value to leave the maximum number of available instances desired after a resource cleanup. A lower value generally results in more application module instances being removed from the pool on a cleanup.
AmpoolSessioncookiefactoryclass	Enter a custom session cookie factory implementation. This class creates the session cookies that allow clients to retrieve application modules in stateful mode
AmpoolMaxinactiveage	Enter the maximum amount of time (in milliseconds) that an application module may remain inactive before it is removed from the pool. The default is 600000 milliseconds of idle time (which is 600 seconds, or ten minutes). A lower value results in more application module instances being marked as candidates for removal at the next resource cleanup. A higher value results in fewer application module instances being marked as candidates for removal at the next resource cleanup.

Element	Description
AmpoolMinavailablesize	<p>Enter the minimum number of available application modules that should be referenced by an application pool. This is the minimum number of available application module instances that the pool monitor should leave in the pool during a resource cleanup operation.</p> <p>Set to 0 (zero) if you want the pool to shrink to contain no instances when all instances have been idle for longer than the idle timeout after a resource cleanup.</p> <p>The default is 5 instances.</p>
Doconnectionpooling	<p>Select if the application pool should release the application module connection upon checkin. This forces the application module pool to release the JDBC connection used each time the application module is released to the pool.</p>
Recyclethreshold	<p>Enter the maximum number of application module instances in the pool that attempt to preserve session affinity for the next request made by the session. This session used them last before releasing them to the pool in managed-state mode.</p>
AmpoolConnectionstrategyclass	<p>Enter a custom connection strategy implementation, for example <code>oracle.jbo.common.ampool.DefaultConnectionStrategy</code>. This is the class that implements the connection strategy.</p>
Maxpoolcookieage	<p>Enter the maximum browser cookie age for pooled application module sessions. This is the maximum age of the browser cookies used to help clients retrieve stateful application modules. If these cookies do not time out, the value is -1. It is recommended that the maximum cookie age be always set less than or equal to the session cookie age. It is set that way by default (both are -1). If you change the maximum cookie age, then you must also change the session cookie age to the same value.</p>
AmpoolInitpoolsize	<p>Enter an initial number of application module instances to be created in a pool. This is the number of application module instances to created when the pool is initialized.</p> <p>The default is 0 (zero) instances. A general guideline is to configure this value to 10% more than the anticipated number of concurrent application module instances required to service all users.</p> <p>Creating application module instances during initialization takes the CPU processing costs of creating application module instances during the initialization instead of on-demand when additional application module instances are required.</p>
AmpoolDynamicjdbccredentials	<p>Select if an application pool may support multiple JDBC users. This property enables additional pooling lifecycle events to allow developer-written code to change the database credentials (username/password) each time a new user session begins to use the application module.</p> <p>This feature is enabled by default (<code>true</code>); however this setting is a necessary but not sufficient condition to implement the feature. The complete implementation requires additional developer-written code.</p>
AmpoolIsuseexclusive	<p>Select if application module use is exclusive.</p>

Element	Description
AmpoolResetnontransactionalstate	<p>Select if the nontransactional application module state should be reset upon an unmanaged checkin. This forces the application module to reset any nontransactional state like view object runtime settings, JDBC prepared statements, bind variable values, and so on. when the application module is released to the pool in unmanaged, or "stateless," mode.</p> <p>This feature is enabled by default (<code>true</code>). Disabling this feature can improve performance; however, since it does not clear bind variable values, your application needs to ensure that it systemically sets bind variable values correctly. If your application does not do so, and this feature is disabled, then it is possible for one user to see data with another user's bind variable values.</p>
AmpoolMaxpoolsize	<p>Enter the maximum number of application module instances that the pool can allocate. The pool will never create more application module instances than this limit imposes.</p> <p>The default is 5000 instances. A general guideline is to configure this value to 20% more than the initial pool size to allow for some additional growth. If the value is set too low, then some users may see an error when they tries to access the application if no application module instances are available.</p>
AmpoolTimetolive	<p>Enter the connection pool time to live for connection instances. This is the number of milliseconds after which an application module instance in the pool is considered as a candidate for removal during the next resource cleanup, regardless of whether it would bring the number of instances in the pool below <code>minavailablesize</code>.</p> <p>The default is 3600000 milliseconds of total time to live (which is 3600 seconds, or one hour). The default value is sufficient for most applications.</p>
AmpoolMonitorsleepinterval	Enter the length of time (in milliseconds) between pool resource cleanups.
Dofailover	<p>Select if failover should occur upon checkin to the application module pool. This feature enables eager passivation of pending transaction state each time an application module is released to the pool in managed state mode. Web applications should set enable failover (<code>true</code>) to allow any other application module to activate the state at any time. This feature is disabled by default (<code>false</code>).</p>
poolClassName	Enter the custom application pool implementation class.
Show Connection Pool Properties	Expand to display fields containing current advanced connection pool properties, or enter new values in the fields.
Hide Connection Pool Properties	Click to hide all Connection Pool Properties fields.

Pooling and Scalability Tab - Connection Pool Properties

In the Pooling and Scalability tab, select the connection pool properties you want to configure.

Element	Description
Initpoolsize	<p>Enter the initial size of a JDBC connection pool. This is the number of JDBC connection instances created when the pool is initialized.</p> <p>The default is an initial size of 0 instances.</p>
Maxpoolsize	<p>Enter the maximum size of a JDBC connection pool. This is the maximum number of JDBC connection instances that the pool can allocate. The pool will never create more JDBC connections than this allows.</p> <p>The default is 5000 instances.</p>
Poolmaxinactiveage	<p>Enter the maximum amount of time (in milliseconds) that a connection may remain inactive before it is removed from the pool. This is the number of milliseconds after which to consider an inactive application module instance in the pool as a candidate for removal during the next resource cleanup.</p> <p>The default is 600000 milliseconds of idle time (which is 600 seconds, or ten minutes). A lower value results in more application module instances being marked as candidates for removal at the next resource cleanup. A higher value results in fewer application module instances being marked as candidates for removal at the next resource cleanup.</p>
Poolmaxavailablesize	<p>Enter the maximum number of available connections that should be referenced by a connection pool. This is the ideal maximum number of JDBC connection instances in the pool when not under abnormal load.</p> <p>When the pool monitor wakes up to do resource cleanup, it will try to remove available JDBC connection instances to bring the total number of available instances down to this ideal maximum. Instances that have been not been used for a period longer than the idle instance timeout will always get cleaned up at this time, and then additional available instances will be removed, if necessary, to bring the number of available instances down to this size.</p> <p>The default is an ideal maximum of 25 instances (when not under load).</p>
Poolrequesttimeout	<p>Enter the time (in milliseconds) that a request should wait for a JDBC connection to be released to the connection pool.</p>
Poolminavailablesize	<p>Enter the minimum number of available connections that should be referenced by a connection pool. This is the minimum number of available JDBC connection instances that the pool monitor should leave in the pool during a resource cleanup operation.</p> <p>Set to zero (0) if you want the pool to shrink to contain no instances when instances have been idle for longer than the idle time-out.</p> <p>The default is to not let the minimum available size drop below 5 instances.</p>

Element	Description
Poolmonitorsleepinterval	<p>Enter the time (in milliseconds) that the connection pool monitor should sleep between pool checks. This is the length of time in milliseconds between pool resource cleanup.</p> <p>While the number of application module instances in the pool will never exceed the maximum pool size, available instances which are candidates for getting removed from the pool do not get "cleaned up" until the next time the application module pool monitor wakes up to do its job.</p>
ConnectionPoolManager	Enter the implementation of the connection pool manager which will be used.
Pooltimetolive	Enter the application pool time to live (in milliseconds) for application module instances.

Core Tab

Use the core tab to view or edit core properties for the application module.

Element	Description
DefaultLanguage	Enter the default business components session language, which is part of the locale.
Passivationstore	<p>Enter the type of store, file, or database file that should be used for application module passivation.</p> <p>database is the default choice. While it may be a little slower than passivating to file, it is by far the most reliable choice.</p> <p>file may offer faster performance because access to the file is faster than access to the database.</p>
Default Country	Enter the default business components session country, which is part of the Locale.
AssocConsistent	Select if entity row set associations have been kept consistent.
XmlValidation	Select to determine the validation mode for the XML parser. If selected, the XML parser uses strict XML validation.
DatabaseConfig	Database Configuration.
Name	Enter the name of the application module.
OracleSchema	Enter the name of the schema in which the business components runtime libraries are deployed.
Show Advanced Properties	Expand to display fields containing current advanced core properties, or enter new values in the fields.
PersMaxRowsPerNode	Enter the maximum size of a node for view row spillover.
PassivationTrackInsert	If selected when an application module is activated, it will be updated to include rows inserted into the database while it was passive.
ApplicationPath	For EJB deployment, enter the JNDI path to the business components.

Element	Description
ViewlinkConsistent	If selected, the view object row sets retrieved through view link accessors will include rows that have been added, even if these changes have not been posted to the database.
ConnectionMode	Deprecated property, formerly used for deployment to VisiBroker. VisiBroker deployment is no longer supported.
Maxpassivationstacksize	Enter the maximum size of the passivation stack (default is 10)
TxnHandleafterpostexc	Select to cause ADF Business Components to take a transaction snapshot before beginning a commit operation. If an exception is thrown after changes have been posted to the database, ADF Business Components will use this snapshot to roll back the in-memory state of your application module to the point before commit operation began.
SnapshotstoreUndo	Enter the target for undo snapshots {transient persistent}
Project	Enter the name of the project containing extended business components to be substituted for base ones, if Factory-Substitution-List is not empty.
Tmpdir	Enter the directory for temporary Oracle ADF Business Components files.
DeployPlatform	The deployment platform: select LOCAL, EJB_IAS (for an EJB deployed to Oracle Application Server), or WLS (for an EJB deployed to Oracle WebLogic Server).
PersMaxActiveNodes	Enter the maximum number of nodes that will be cached in memory for view row spillover.
Saveforlater	Select Save snapshots for the lifetime of the transaction.
ViewCriteriaAdapter	Enter a custom class that will be used by view objects to convert between view criteria and view object SQL.
Connectfailover	Select business components transparent JDBC connection failover
Hide Advanced Properties	Click to hide all Connection Pool Properties fields.

Database Properties Tab

If you are using a JDBC URL for your connection information so that the ADF database connection pool is used, then the configuration parameters listed here can be used to tune the behavior of the database connection pool.

Element	Description
MaxCursors	Enter the maximum number of cursors to be used by the session. This is the maximum number of cursors the business components may have open. The framework will clean up free JDBC statements as the number of cursors approaches this number.
Sql92DbTimeQuery	Enter the database system time SQL query string.
SQLBuilder	Enter the SQLBuilder implementation (Oracle, OLite, DB2, or SQL92 for other SQL92-compliant databases).

Element	Description
Sql92LockTrailer	Enter the SQL statement trailer clause for locking.
JdbcTrace	Select to trace all JDBC activity with lines flagged by + <code>PropertyConstants.JDBC_MARKER</code> +
oracleDefineColumnLength	Enter the column length for all JDBC CHAR or VARCHAR2 columns. Use <code>as_bytes</code> to make column precision specifications in bytes. Use <code>as_chars</code> to make column precision specifications in characters. This is important for larger character sets, such as Unicode.
Sql92JdbcDriverClass	Enter the name of the class implementing JDBC Driver, for example, <code>sun.jdbc.odbc.JdbcOdbcDriver</code> .
TypeMapEntries	Enter the type map implementation. This specifies a custom type map between Java types and SQL types.
ControlTableName	Enter the persistent collection control table name.
FetchMode	Enter the control fetch behavior of View Objects (+ <code>PropertyConstants.ENV_FETCH_AS_NEEDED</code> + " " + <code>PropertyConstants.ENV_FETCH_ALL</code> +). AS.NEEDED causes view objects to fetch rows only when they are requested. ALL causes them to fetch the entire results of their queries.
LockingMode	Enter the default locking mode for an application module. This prevents the application module pool from creating a pending transaction state on the database with row-level locks each time the application module is released to the pool. Fusion web applications should set the locking mode to optimistic to avoid creating the row-level locks.
JdbcBytesConversion	Indicate whether to use JDBC default bytes conversion or to perform such conversion in the framework.
Show Advanced Properties	Expand to display fields containing current advanced database properties, or enter new values in the fields.
TxnSeqInc	Select persistent transaction sequence increment.
UsePersColl	Select enable view row spillover to help manage large rowsets.
TxnSeqName	Enter persistent transaction sequence name.
Hide Advanced Properties	Click to hide all advanced property fields.

Security Properties Tab

Use the Security Properties tab to configure application module security information.

Element	Description
SecurityContext	Enter the JAAS context. This element specifies a particular JAAS implementation. The default is JAZN.
Show Advanced Properties	Expand to display fields containing current advanced security properties, or enter new values in the fields.
UserPrincipal	Enter the authenticated user principal name.

Element	Description
SecurityConfig	<p>Enter the complete path and file name of JAZN configuration, for example, k:\j2ee\home\config\jazn.xml.</p> <p>If this property value is null or length 0, runtime will assume that jazn.xml is in the same path as jazn.jar and append /config/jazn.xml before it accesses login module or gets the JAZN context for getting permission manager.</p>
javaNamingSecurityCredentials	For EJB deployment, enter the password for the application server connection.
AppModuleJndiName	For EJB deployment, enter the JNDI name used to look up the application module factory.
SecurityLoginmodule	<p>Enter a custom login module for authentication, for example, oracle.security.jazn.realm.RealmLoginModule.</p> <p>The default is the JAZN login module.</p>
ServerUseNullDbTransaction	Use 9.0.2 compatible oracle.jbo.server.NullDbtransactionImpl when not connected to the database.
SecurityEnforce	<p>Enter one of the following values:</p> <p>None - No authentication.</p> <p>Test - Requires authentication. If using the tester or ADF Swing, a dialog will prompt for login. If authentication fails, the application module is still instantiated.</p> <p>Must - Like Test, but if authentication fails, the application module will not be instantiated. Instead, you will get an exception.</p> <p>Auth - Like Must, but in addition, if you have used the Entity Wizard Authorization editor to define entity or attribute permissions, the permissions will be checked. For example if the permission on Dept.Deptno was granted update_while_new to role users, then the users role can set the Deptno value only when the row is new. Otherwise, it is not editable.</p> <p>Note that even if there are permissions granted via the wizard, they will not be enforced unless jbo.security.enforce is set to Auth.</p>
javaNamingSecurityPrincipal	For EJB deployment, enter the password for the application server connection.
Hide Advanced Properties	Click to hide all advanced property fields.

3.3.2 How to Modify Connection Configurations

A connection configuration contains information that a client application uses to identify the ADF application module's deployment scenario. You use Oracle Enterprise Manager Fusion Middleware Control to:

- Register and manage back-end services such as mail, discussion forums servers, and so on
- Register and manage external applications that users need access to while working with applications

- Register and manage any portlet producers that the application uses or that users may need access to

Fusion Middleware Control provides a UI to configure ADF connections, as described in this section. You can also configure the ADF connections MBean directly using the generic MBean Browser, as described in [Section 3.4.2, "How to Modify ADF Connections Using MBean."](#)

Before you begin:

You must already have deployed an Oracle ADF application and have Enterprise Manager Fusion Middleware Control available to access the application.

You must have MDS configured in your application before you can modify the ADF application and connection configurations. ADF connection attributes are persisted to MDS.

If you deployed an application to several nodes within a cluster, any ADF connection changes to a single node will be propagated to all the other nodes. MDS will store a single set of connection information for all versions of an application.

To modify connection configurations:

1. Log in to an Oracle Fusion Middleware farm using Fusion Middleware Control.
2. Click the **Farm** tab.
3. Expand the *Farm_domain* node.
4. Expand the **Application Deployments** node and click a J2EE application deployment, for example, *StoreFrontModule (AdminServer)*.

After you select an application, the Application Deployment page displays.

5. Click **Application Deployment** and select **ADF > Configure ADF Connections** from the dropdown menu.
6. In the **Connection Type** drop-down list, choose the type of connection you want to configure:
 - ADF BC Service
 - Discussions and Announcements
 - File System
 - Mail Server
 - Secure Enterprise Search
 - URL
 - Web Service

You cannot create an Essbase connection, however, you can edit an existing Essbase connection that was deployed with the application.

7. In the **Connection Name** field, enter a unique name for the connection configuration.
8. Click **Create Connection**.

The Connection Configuration page updates with a section where you can specify options for the connection type you chose.

The following connection types are supported:

- ADF Business Components Service connection

- Essbase connection
- Discussions and Announcements connection
- File system connection
- Mail server connection
- Secure enterprise search connection
- URL connection
- Web Service connection

ADF Business Components Service Connection

Use the ADF Business Components Service connection page to create a new ADF Business Components Service connection or to modify existing connection details.

Element	Description
serviceInterfaceName	Enter the class name of the service endpoint interface.
serviceSchemaName	Enter the name of the service schema file.
serviceSchemaLocation	Enter the relative path of the service schema file.
serviceEndpointProvider	Enter the provider of the service endpoint. Valid types are <code>ADFBC</code> , <code>Fabric</code> .
jndiName	Enter the JNDI name of the EJB that implements the service interface. Applicable when the endpoint is <code>ADF BC</code> .
jndiFactoryInitial	Enter the class name of initial context factory for JNDI lookup. Applicable when the endpoint is <code>ADF BC</code> .
jndiProviderURL	Enter configuration information for the JNDI lookup. Applicable when endpoint is <code>ADF BC</code> .
jndiSecurityPrincipal	Enter the identity of the principal (e.g. user) for the JNDI lookup. Applicable when the endpoint is <code>ADF BC</code> .
jndiSecurityCredentials	Enter the principal's credentials for JNDI lookup. Applicable when the endpoint is <code>ADF BC</code> .
fabricAddress	Enter the service name of the SOA composite. Applicable when the endpoint is <code>Fabric</code> .

Essbase Connection

You cannot create an Essbase connection; however, you can edit an existing Essbase connection that was deployed with the application.

Element	Description
Host	Enter the host that this connection represents.
Port	Displays the default port that this connection uses to connect to Essbase. Clear the Default option to enter a port other than the default.
Username	Enter the user name authorized to connect to Essbase during design time. This user name is replaced at runtime with the user name specified by the application.

Element	Description
Password	Enter the password of the user. An asterisk (*) is displayed for each character you enter in this field.

Discussions and Announcements Connection

Use the Discussion Forum Connection pages to connect to a new discussions server connection or to modify existing connection details. Forum Connections configuration includes configurations for name, connection details, and advanced.

Discussions and Announcements Connection - Name

Element	Description
Name	Enter a unique name for the connection.

Discussions and Announcements Connection - Connection Details

Element	Description
Server URL	Enter the URL of the discussion server hosting the discussion forums. For example: <code>http://discuss-server.com:8888/owc_discussions</code>
Administrator User Name	Enter the user name of the discussion server administrator. Administrative privileges are required for this connection so that operations can be performed on behalf of WebCenter users.
Connection Secured	Indicate whether or not the discussion server connection is secure.

Discussions and Announcements Connection - Advanced Configuration

Element	Description
Cache Size (in MB)	Specify the amount of space reserved for the cache (in MB). The default is 0.
Cache Expiration Time (in Minutes)	Specify a suitable expiration period for the cache. This is the maximum length of time (in minutes) that cached content is valid. The default is 0.
Connection Timeout (in Seconds)	Specify a suitable timeout for the connection. This is the length of time (in seconds) that the WebCenter application waits for a response from the discussion server before issuing a connection timeout message. The default is 60 seconds.

File System Connection

Use the Add/New Content Repository Connection pages to connect to a new content repository or to modify existing connection details.

Note: All configuration changes are stored in the MDS repository.

File System Connection Details - File System

Element	Description
Base Path	Enter the full path to a folder on a local file system in which your content is placed. For example: <code>C:/MyContent</code> Caution: File system content <i>must not</i> be used in production or enterprise application deployments. This feature is provided for development purposes only.

Mail Server Connection

Use the Mail Server connection pages to configure LDAP and advanced mail server configurations

Element	Description
IMAP Host	Enter the host name of the machine where the IMAP service (Internet Message Access Protocol) is running.
IMAP Port	Enter the port on which the IMAP service listens.
IMAP Secured	Indicate whether a secured connection (SSL) is required for incoming mail over IMAP.
SMTP Host	Enter the host name of the machine on which the SMTP service (Simple Mail Transfer Protocol) is running.
SMTP Port	Enter the port on which the SMTP service listens.
SMTP Secured	Indicate whether a secured connection (SSL) is required for outgoing mail over SMTP.
Associated External Application	Associate the mail server with an external application. External application credential information is used to authenticate users against the IMAP server.

Mail Server Connection - LDAP Configuration

Element	Description
LDAP Domain	Enter the LDAP domain.
LDAP Host	Enter the host name of the LDAP (Lightweight Directory Access Protocol) server.
LDAP Port	Enter the port on which the LDAP service listens.
LDAP Secured	Indicate whether a secured connection (SSL) is required for the LDAP connection.
LDAP Administrator User Name	Enter the user name of the LDAP server administrator.
LDAP Administrator Password	Enter the password for the LDAP server administrator.
LDAP Base DN	Enter the base-distinguished name for the LDAP schema.
LDAP Default User	Enter the LDAP default user.

Mail Server Connection - Advanced Configuration

Element	Description
Connection Timeout (in Seconds)	Specify a suitable timeout for the connection. This is the length of time (in seconds) that the WebCenter application waits for a response from the mail server before issuing a connection timeout message. The default is 60 seconds.
Cache Expiration Time (in Minutes)	Specify a suitable expiration period for the cache. This is the maximum length of time (in minutes) that cached content is valid. The default value (-1) means that the cache never expires.

Secure Enterprise Search Connection

Use the Secure Enterprise Search Connection pages to connect the WebCenter application to a new Oracle Secure Enterprise Search server or to modify existing connection details.

Secure Enterprise Search Connection Provider configuration includes configurations for name, connection details, and advanced configurations.

Secure Enterprise Search Connection - Name

Element	Description
Connection Name	Enter a unique name for the connection.
Active Connection	Select to use this connection for search-related services in the WebCenter application. You can register multiple search connections through Oracle Enterprise Manager Fusion Middleware Control, but only one connection is active at a time.

Secure Enterprise Search Connection - Connection Details

Element	Description
SOAP URL	Enter the Web Service URL that Oracle Secure Enterprise Search exposes to enable search requests. Use the format: <code>http://<host>:<port>/search/query/OracleSearch</code> For example: <code>http://myHost:7777/search/query/OracleSearch</code>
Application User Name	Enter the name of a valid user. You can specify the name of any user in the identity store. The user must be present in both the Oracle Identity Management server configured for your WebCenter application and the Oracle Identity Management server configured for Oracle SES. The WebCenter application must authenticate itself as a trusted application to Oracle Secure Enterprise Search so that it may perform searches on behalf of WebCenter users.
Application Password	Enter the appropriate user password.

Secure Enterprise Search Connection - Advanced Configuration

Element	Description
Oracle Secure Enterprise Search Data Group	Enter the Secure Enterprise Search data group in which to search.
Execution Timeout	Enter the search execution timeout in milliseconds.
Executor Preparation Timeout	Enter the search executor preparation timeout in milliseconds.
Number of Saved Searches	Enter the number of saved searches displayed.
Simple Search Result Rows	Enter the number of results displayed from a simple search, for each service.
Search Result Rows	Enter the number of search results displayed for each service.
Global Search Result Rows	Enter the number of search results displayed (on the toolbar) for each service.

URL Connection

Use the URL Connection pages to configure URL connections.

Element	Description
URL	Enter the URL of the desired data stream, but omit any URL parameters.
Username	Enter the username require to enter the web site.
Password	Enter the password required to enter the web site.
AuthenticationRealm	Defines the Realm as in HTTP authentication. Defined by the server hosting the protected resources.
Proxy	Defines the proxy to be used for connecting to HTTP/HTTPS resources. Specifies the host/port and any authentication details needed to authenticate against the proxy itself.
ProxyUseDefault	Uses the default proxy at the system level instead of the connection level at both DT or RT, or wherever the connection instance is active. At design time, the default proxy will be the JDeveloper IDE proxy settings, at runtime, it will be the one configured for WLS.
ConnectionClassName	Indicates the type of challenge authentication. The two supported modes are Basic and Digest authentication (HTTP basic & digest).
ChallengeAuthenticationType	The class name of the connection that gets loaded into the reference to be used by the factory to construct the connection instance.

Web Service Connection

Use the Web Service Connection page to configure a connection using the WebService MDDS model based on the service WSDL to call and invoke the WebService.

Use the **Configure Web Service** dropdown list to configure the Web Service Client, including attaching and detaching policy. After you have finished the configuration in the web services page, you can use the breadcrumbs to navigate back to the ADF Connections page.

Element	Description
Model	Enter the WebService MDDS model elements generated based on the service WSDL.
WsdUrl	Enter the WebService service WSDL URL.
DefaultServiceName	Enter the default service Name of the service WSDL.

3.4 Configuring Application Properties Using the MBean Browser

You can use the Enterprise Manager Fusion Middleware Control MBean Browser to access and modify the values in ADF MBeans deployed with the ADF application into Oracle WebLogic Server.

You can view and modify:

- ADFcConfiguration MBean
- ADF Connections MBean
- ADF Business Components BC4J MBeans
- MDS Configuration MBeans

3.4.1 How to Modify ADF Application Configuration Using MBean

You can modify ADF application configurations MBeans using the MBean Browser.

Before you begin:

You must have MDS configured in your application before you can modify the ADF application and connection configurations. ADF application attributes are persisted to MDS.

If you deployed an application to several nodes within a cluster, any ADF application configuration changes to a single node via an MBean will be propagated to all the other nodes. MDS will store a single set of ADF application configuration information for all versions of an application.

To modify ADF application configuration using MBean Browser:

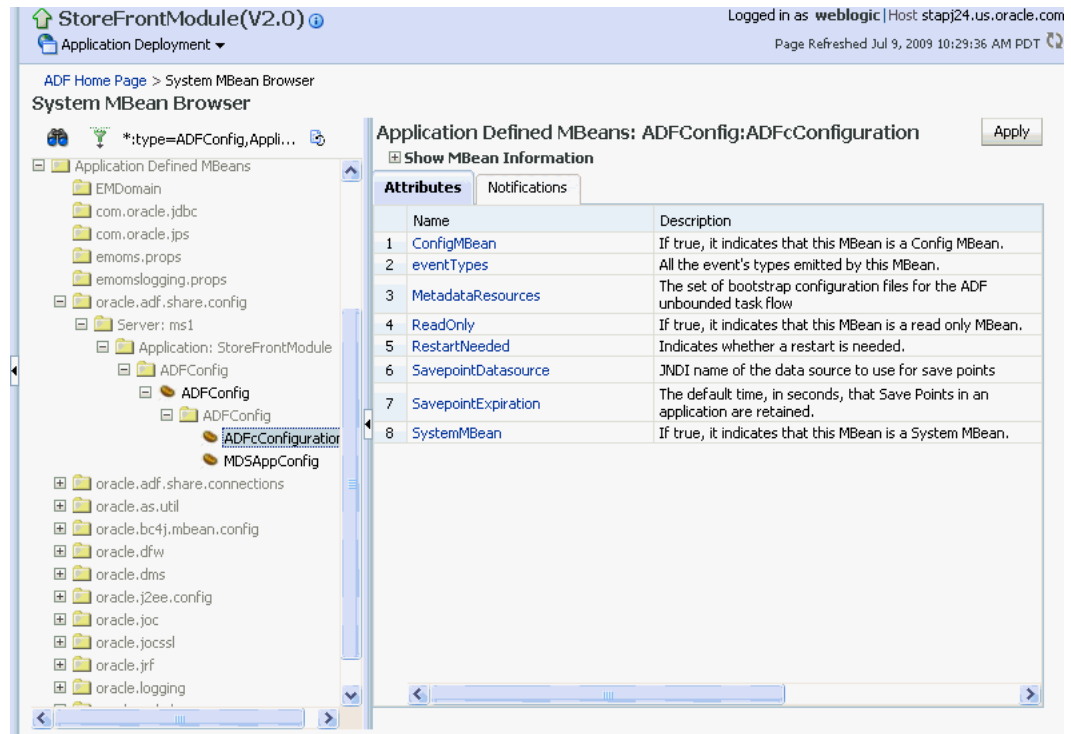
1. Log in to an Oracle Fusion Middleware farm using Fusion Middleware Control.
2. Click the **Farm** tab.
3. Expand the *Farm_domain* node.
4. Expand the **Application Deployments** node and click a J2EE application deployment, for example, *StoreFrontModule (AdminServer)*.
After you select an application, the Application Deployment page displays.
5. Click **Application Deployment** and select **ADF > Configure ADF (adf-config)** from the dropdown menu.
6. In the left pane of the System MBean Browser, expand the parent ADF MBean **ADFConfig** and then the **ADFConfig** folder to expose the child ADF MBeans.

You may see the child ADF MBeans **ADFCConfiguration** and **MDSAppConfig**.

7. In the left pane, select the **ADFCConfiguration** MBean, and in the right pane, select the attribute you want to view or modify.

Figure 3–1 shows an ADF Configuration MBean in the Fusion Middleware Control MBean Browser.

Figure 3–1 ADF Configuration MBean



8. Change the attribute value and click **Apply**.
9. In the left pane, select the parent ADF MBean **ADFCConfig**.
10. In the right pane, click the **Operations** tab and click **save**.

The new values you have edited are written to MDS after you click **save** from the parent MBean.

3.4.2 How to Modify ADF Connections Using MBean

You can modify ADF connection configurations MBean using the MBean Browser.

You can also modify ADF connections using the Fusion Middleware UI described in [Section 3.3.2, "How to Modify Connection Configurations."](#)

Before you begin:

You must have MDS configured in your application before you can modify the ADF application and connection configurations. ADF application attributes are persisted to MDS.

If you deployed an application to several nodes within a cluster, any ADF connection changes to a single node via an MBean will be propagated to all the other nodes. MDS

will store a single set of ADF application configuration information for all versions of an application.

To modify ADF application configuration using MBean Browser:

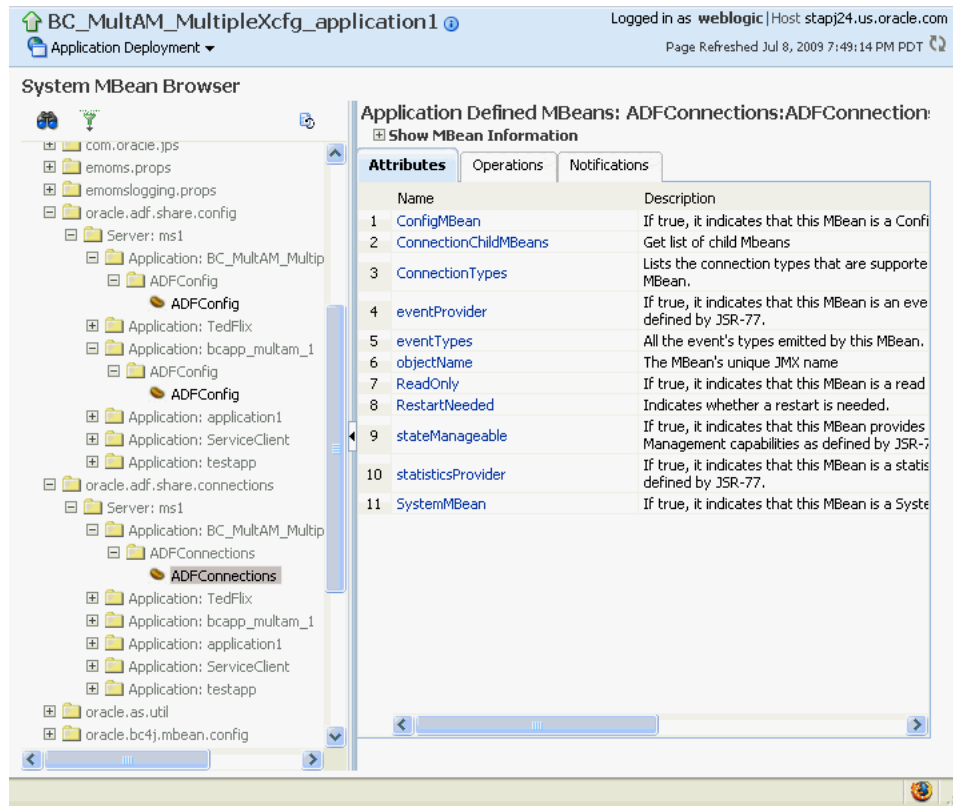
1. Log in to an Oracle Fusion Middleware farm using Fusion Middleware Control.
2. Click the **Farm** tab.
3. Expand the *Farm_domain* node.
4. Expand the **Application Deployments** node and click a J2EE application deployment, for example, *StoreFrontModule (AdminServer)*.

After you select an application, the Application Deployment page displays.

5. Click **Application Deployment** and select **System MBean Browser** from the dropdown menu.
6. In the left pane of the System MBean Browser, navigate to the **ADFConnections** MBean. The MBean should be in **oracle.adf.share.connections > server name > application name**.
7. In the left pane, select the ADF Connections MBean, and in the right pane, select the attribute you want to view or modify.

Figure 3–2 shows an ADF Connections MBean displayed in the Fusion Middleware Control MBean Browser.

Figure 3–2 ADF Connections MBean



8. Change the attribute value and click **Apply**.
9. In the right pane, click the **Operations** tab and click **save**.

The new values you have edited are written to MDS after you click **save**.

3.4.3 How to Modify ADF Business Components Configuration Using MBeans

You can modify ADF Business Components configurations MBeans using the MBean Browser. ADF Business Component configuration information are stored in MBeans that are specific for each application. Unlike ADF connections and ADF application configuration information which you can configure once for all versions of the same application, you will need to configure ADF Business Components for each version of the application.

You can also modify ADF Business Components configuration information using the Fusion Middleware UI described in [Section 3.3.1, "How to Modify ADF Business Components Parameters."](#)

Before you begin:

You must have MDS configured in your application before you can modify the ADF application and connection configurations. ADF application attributes are persisted to MDS.

If you deployed an application to several nodes within a cluster, any ADF Business Components changes to a single node via MBeans will be propagated to all the other nodes. MDS will store a single set of ADF application configuration information for all versions of an application.

To modify ADF application configuration using MBean Browser:

1. Log in to an Oracle Fusion Middleware farm using Fusion Middleware Control.
2. Click the **Farm** tab.
3. Expand the *Farm_domain* node.
4. Expand the **Application Deployments** node and click a J2EE application deployment, for example, *StoreFrontModule (AdminServer)*.

After you select an application, the Application Deployment page displays.

5. Click **Application Deployment** and select **System MBean Browser** from the dropdown menu.
6. In the left pane of the System MBean Browser, navigate to the BC4J MBeans. These MBeans should be in **oracle.bc4j.mbean.share > server name > application name**.
7. In the left pane, select the ADF Connections MBean, and in the right pane, select the attribute you want to view or modify.
8. Change the attribute value and click **Apply**.

3.4.4 How to Modify MDS Configuration Using MBean

You can use the MBean Browser to perform advanced configuration of MDS parameters. For more information about configuring MDS using MBeans, see the *Oracle Fusion Middleware Administrator's Guide*.

Before you begin:

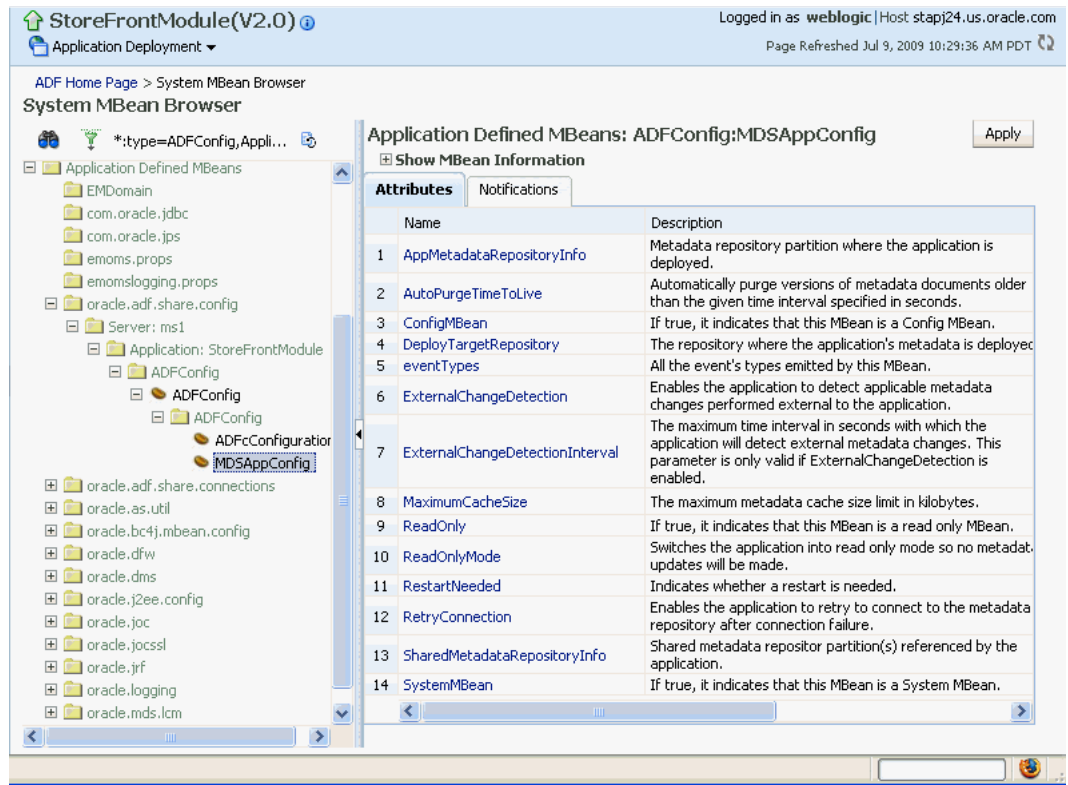
You must already have deployed an Oracle ADF application and have Enterprise Manager Fusion Middleware Control available to access the application.

To modify MDS configuration using MBean Browser:

1. Log in to an Oracle Fusion Middleware farm using Fusion Middleware Control.
2. Click the **Farm** tab.
3. Expand the *Farm_domain* node.
4. Expand the **Application Deployments** node and click a J2EE application deployment, for example, *StoreFrontModule (AdminServer)*.
After you select an application, the Application Deployment page displays.
5. Click **Application Deployment** and select **MDS Configuration** from the dropdown menu.
6. Click **Configuration MBean Browser** or **Runtime MBean Browser**.
7. Select the MBean and the attribute you want to view or modify.

Figure 3–3 show an MDS MBean in the Fusion Middleware Control MBean Browser.

Figure 3–3 MDS MBean



8. Change the value and click **Apply**.

3.5 Diagnosing Problems using the Diagnostic Framework

Oracle Fusion Middleware provides a Diagnostic Framework to help you detect, diagnose, and resolve problems with your application

When a critical error occurs, the Diagnostic Framework immediately captures diagnostic data and associates the data and error with an incident number. Using this

number, you can retrieve the data for analysis from the Automatic Diagnostic Repository (ADR).

For more information about the Diagnostic Framework, see the chapter on diagnosing problems in the *Oracle Fusion Middleware Administrator's Guide*.

WLST Command Reference for ADF Applications

This chapter describes the WLST commands you can use to deploy, manage, and configure Oracle ADF applications.

This chapter includes the following sections:

- [Section 4.1, "Overview of Custom WLST Commands for Oracle ADF"](#)
- [Section 4.2, "Commands for ADF-based URL Connections"](#)

4.1 Overview of Custom WLST Commands for Oracle ADF

Use the ADF-based URL Connections WLST commands to navigate the hierarchy of configuration or runtime beans and control the prompt display.

To use the custom WLST commands for Oracle ADF, you must invoke the WLST script from the Oracle Common home. For more information about other WLST commands, such as custom Metadata Services (MDS) commands, see the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

4.2 Commands for ADF-based URL Connections

Use the commands in [Table 4-1](#) to manage URL-based connections.

Table 4-1 Browse Commands for WLST Configuration

Use this command...	To...	Use with WLST...
<code>adf_createFileUrlConnection</code>	Create a new ADF file connection.	Online or Offline
<code>adf_createHttpUrlConnection</code>	Create a new ADF URL connection.	Online or Offline
<code>adf_setUrlConnectionAttributes</code>	Set or edit the attributes of a newly created or existing ADF connection.	Online or Offline
<code>adf_listUrlConnection</code>	List a new URL connection.	Online or Offline

4.2.1 `adf_createFileUrlConnection`

Use with WLST: Online or Offline.

4.2.1.1 Description

Use this command to create a new connection based on the `oracle.adf.model.connection.url.FileURLConnection` connection class.

4.2.1.2 Syntax

```
adf_createFileURLConnection(appName, name, URL)
```

Argument	Definition
<code>appName</code>	Application name for which the connection will be created.
<code>name</code>	The name of the new connection.
<code>URL</code>	The URL associated with this connection.

4.2.1.3 Example

```
adf_createFileURLConnection('myapp', 'tempDir', '/scratch/tmp')
```

4.2.2 adf_createHttpURLConnection

Use with WLST: Online or Offline.

4.2.2.1 Description

Use this command to create a new connection based on the `oracle.adf.model.connection.url.HttpURLConnection` connection type class.

4.2.2.2 Syntax

```
adf_createHttpURLConnection (appName, name, [URL], [authenticationType], [realm], [user], [password])
```

Argument	Definition
<code>appName</code>	Application name for which the connection will be created.
<code>name</code>	The name of the new connection.
<code>url</code>	(Optional) The URL associated with this connection.
<code>authenticationType</code>	(Optional) The default is basic.
<code>realm</code>	(Optional) If this connection deals with authentication, then this should be set. The default is basic.
<code>user</code>	(Optional)
<code>password</code>	(Optional)

4.2.2.3 Example

```
adf_createHttpURLConnection('myapp', 'cnn', 'http://www.cnn.com')
```

4.2.3 adf_setURLConnectionAttributes

Use with WLST: Online or Offline.

4.2.3.1 Description

Use this command to set or edit the attributes of a newly created or existing ADF connection.

4.2.3.2 Syntax

```
adf_setURLConnectionAttributes(appname, connectionname, attributes)
```

Argument	Definition
appname	Application name.
connectionname	The name of the connection.
attributes	The array containing attributes to set in key/value pairs.

4.2.3.3 Example

```
adf_setURLConnectionAttributes
('myapp', 'cnn', 'ChallengeAuthenticationType:digest',
'AuthenticationRealm:XMLRealm')
```

4.2.4 adf_listUrlConnection

Use with WLST: Online or Offline.

4.2.4.1 Description

Use this command to list the connections of the application.

4.2.4.2 Syntax

```
adf_listURLConnection(appname)
```

Argument	Definition
appname	Application name.

4.2.4.3 Example

```
adf_listURLConnection ('myapp')
```


Part III

Appendices

Part I contains the following chapters:

- [Appendix A, "JDeveloper Runtime Libraries"](#)

JDeveloper Runtime Libraries

This appendix provides a reference of the contents of JDeveloper runtime libraries that are deployed into Oracle WebLogic Server to support ADF applications.

The following JDeveloper runtime libraries are described:

- [Section A.1, "adf.oracle.domain.webapp.war Library"](#)
- [Section A.2, "adf.oracle.domain.ear Library"](#)
- [Section A.3, "System Classpath"](#)

A.1 adf.oracle.domain.webapp.war Library

[Table A-1](#) lists the JAR files that are packaged into the `adf.oracle.domain.webapp.war` file and their corresponding JDeveloper runtime library.

Table A-1 *adf.oracle.domain.webapp.war Library*

JAR	JDeveloper Library
oracle.adf.controller_ 11.1.1\adf-controller-api.jar	ADF Controller Runtime
oracle.adf.controller_ 11.1.1\adf-controller-rt-common.jar	ADF Controller Runtime
oracle.adf.controller_ 11.1.1\adf-controller.jar	ADF Controller Runtime
oracle.adf.pageflow_ 11.1.1\adf-pageflow-dtrt.jar	ADF Page Flow Runtime ADF Designtime API
oracle.adf.pageflow_ 11.1.1\adf-pageflow-fwk.jar	ADF Page Flow Runtime
oracle.adf.pageflow_ 11.1.1\adf-pageflow-impl.jar	ADF Page Flow Runtime
oracle.adf.pageflow_ 11.1.1\adf-pageflow-rc.jar	ADF Page Flow Runtime
oracle.adf.view_11.1.1\adf-dt-at-rt.jar	ADF Model Runtime ADF Designtime API
oracle.adf.view_ 11.1.1\adf-dynamic-faces.jar	ADF Faces Dynamic Components
oracle.adf.view_ 11.1.1\adf-faces-changemanager-rt.jar	ADF Faces Change Manager Runtime 11

Table A-1 (Cont.) adf.oracle.domain.webapp.war Library

JAR	JDeveloper Library
oracle.adf.view_ 11.1.1\adf-faces-databinding-dt-core.jar	ADF Designtime API
oracle.adf.view_ 11.1.1\adf-faces-databinding-rt.jar	Trinidad Databinding Runtime ADF Faces Databinding Runtime
oracle.adf.view_ 11.1.1\adf-faces-registration.jar	NA
oracle.adf.view_ 11.1.1\adf-faces-templating-dt-core.jar	ADF Designtime API
oracle.adf.view_ 11.1.1\adf-faces-templating-dtrt.jar	ADF Designtime API
oracle.adf.view_ 11.1.1\adf-richclient-api-11.jar	Trinidad Databinding Runtime ADF Faces Runtime 11
oracle.adf.view_ 11.1.1\adf-richclient-automation-11.jar	Oracle Customized Selenium
oracle.adf.view_ 11.1.1\adf-richclient-impl-11.jar	ADF Faces Runtime 11
oracle.adf.view_11.1.1\adf-share-web.jar	NA
oracle.adf.view_ 11.1.1\adf-view-databinding-dt-core.jar	ADF Designtime API
oracle.adf.view_ 11.1.1\dvt-databindings.jar	BI Data Control Runtime Essbase Data Control Runtime ADF DVT Faces Databinding Runtime
oracle.adf.view_11.1.1\dvt-faces.jar	ADF DVT Faces Runtime
oracle.adf.view_ 11.1.1\dvt-facesbindings.jar	BI Data Control Runtime Essbase Data Control Runtime ADF DVT Faces Databinding Runtime
oracle.adf.view_11.1.1\dvt-jclient.jar	Oracle BI Graph ADF DVT Core Runtime ADF Swing Runtime ADF DVT Faces Runtime
oracle.adf.view_11.1.1\dvt-trinidad.jar	ADF DVT Core Runtime ADF DVT Faces Runtime
oracle.adf.view_11.1.1\dvt-utils.jar	BI Data Control Runtime Oracle BI Graph ADF DVT Core Runtime ADF Swing Runtime Essbase Data Control Runtime ADF DVT Faces Runtime
oracle.adf.view_ 11.1.1\oracle-page-templates.jar	Oracle Page Templates
oracle.adf.view_11.1.1\trinidad-api.jar	ADF Faces Runtime 11 Trinidad Runtime 11

Table A-1 (Cont.) adf.oracle.domain.webapp.war Library

JAR	JDeveloper Library
oracle.adf.view_11.1.1\trinidad-impl.jar	ADF Faces Runtime 11 Trinidad Runtime 11
oracle.facesconfigdt_11.1.1\facesconfigmodel.jar	ADF Faces Change Manager Runtime 11
oracle.facesconfigdt_11.1.1>taglib.jar	ADF Faces Change Manager Runtime 11
velocity-dep-1.4.jar	ADF Designtime API

A.2 adf.oracle.domain.ear Library

Table A-2 lists the JAR files that are packaged into the adf.oracle.domain.ear file and their corresponding JDeveloper runtime library.

Table A-2 adf.oracle.domain.ear Library

JAR	JDeveloper Library
groovy-all-1.6.3.jar	ADF Model Runtime ADF Model Generic Runtime BC4J Runtime
oracle.adf.model_11.1.1\adf-controller-schema.jar	ADF Controller Schema
oracle.adf.model_11.1.1\adf-controller-system.jar	NA
oracle.adf.model_11.1.1\adf-runtime-mbean.jar	NA
oracle.adf.model_11.1.1\adf-sec-idm-dc.jar	User and Role Data Control
oracle.adf.model_11.1.1\adfbcsvc-client.jar	BC4J Service Client
oracle.adf.model_11.1.1\adfbcsvc-share.jar	BC4J Service Runtime BC4J Service Client
oracle.adf.model_11.1.1\adfbcsvc.jar	BC4J Service Runtime
oracle.adf.model_11.1.1\adfdeployrt.jar	NA
oracle.adf.model_11.1.1\adfdt_common.jar	ADF Model Runtime ADFm Designtime API
oracle.adf.model_11.1.1\adflibfilter.jar	ADF Common Web Runtime
oracle.adf.model_11.1.1\adflibrary.jar	NA
oracle.adf.model_11.1.1\adfm-debugger.jar	BC4J Tester

Table A-2 (Cont.) adf.oracle.domain.ear Library

JAR	JDeveloper Library
oracle.adf.model_11.1.1.1\adfm.jar	BC4J EJB Client ADF Model Runtime BC4J Oracle Domains ADF Model Generic Runtime BC4J Runtime ADF Swing Runtime JSR-227 API BC4J EJB Runtime BC4J Client BC4J IAS Client
oracle.adf.model_11.1.1.1\adfmportlet.jar	NA
oracle.adf.model_11.1.1.1\adfmweb.jar	ADF Web Runtime
oracle.adf.model_11.1.1.1\adftags.jar	Oracle ADF DataTag
oracle.adf.model_11.1.1.1\adftransactionsdt.jar	ADF Designtime API
oracle.adf.model_11.1.1.1\bc4j-mbeans.jar	BC4J Runtime
oracle.adf.model_11.1.1.1\bc4jhtml.jar	BC4J Struts Runtime BC4J HTML
oracle.adf.model_11.1.1.1\bc4jimdomains.jar	Oracle Intermedia ADF Swing Oracle Intermedia
oracle.adf.model_11.1.1.1\bc4jsyscat.jar	BC4J Tester
oracle.adf.model_11.1.1.1\bc4jwizard.jar	BC4J Tester BC4J Runtime
oracle.adf.model_11.1.1.1\datatags.jar	BC4J HTML
oracle.adf.model_11.1.1.1\db-ca.jar	BC4J EJB Client ADF Model Runtime BC4J Tester BC4J Runtime BC4J Client BC4J IAS Client
oracle.adf.model_11.1.1.1\jdev-cm.jar	BC4J EJB Client ADF Model Runtime BC4J Tester BC4J Runtime Obsolete JDeveloper Extension SDK BC4J Client BC4J IAS Client Connection Manager
oracle.adf.model_11.1.1.1\jmxdc.jar	NA
oracle.adf.model_11.1.1.1\jr_dav.jar	Resource Catalog Service

Table A–2 (Cont.) *adf.oracle.domain.ear* Library

JAR	JDeveloper Library
oracle.adf.model_11.1.1\oicons.jar	ADFm Designtime API
oracle.adf.model_11.1.1\oraclexsql.jar	NA
oracle.adf.model_11.1.1\ordhttp.jar	Oracle Intermedia ADF Swing Oracle Intermedia
oracle.adf.model_11.1.1\ordim.jar	Oracle Intermedia ADF Swing Oracle Intermedia
oracle.adf.model_11.1.1\rca-adflib-rt.jar	NA
oracle.adf.model_11.1.1\rca.jar	Resource Catalog Service
oracle.adf.model_11.1.1\regexp.jar	NA
oracle.adf.model_11.1.1\xsqlserializers.jar	XSQL Runtime
oracle.bali.share_11.1.1\share.jar	MDS Runtime Dependencies BC4J Tester ADF Model Generic Runtime Oracle Help for Java Oracle JJWT

A.3 System Classpath

Table A–3 lists the JAR files that are loaded into the system classpath and their corresponding JDeveloper runtime library.

Table A–3 System Classpath

JAR	JDeveloper Library
com.sun.msv.datatype.xsd_20030530.jar	NA
commonj.sdo_2.1.0.jar	BC4J Service Runtime BC4J Service Client Java EE 1.5 EJB SDO Client
features\adf.model.generic_11.1.1.jar	NA
features\adf.model_11.1.1.jar	NA
features\adf.share.ca_11.1.1.jar	NA
features\adf.share_11.1.1.jar	NA
groovy-all-1.6.0.jar	NA
jakarta.jstl_1.0\lib\jaxen-full.jar	JSTL 1.0
jakarta.jstl_1.0\lib\jdbc2_0-stdext.jar	JSTL 1.0
jakarta.jstl_1.0\lib\jstl.jar	JSTL 1.1 JSTL 1.0
jakarta.jstl_1.0\lib\saxpath.jar	JSTL 1.0

Table A-3 (Cont.) System Classpath

JAR	JDeveloper Library
jakarta.jstl_1.0\lib\standard.jar	JSTL 1.1 Tags JSTL 1.0 Tags
jakarta.jstl_1.0\lib\xalan.jar	JSTL 1.0
jakarta.jstl_1.0\lib\xercesImpl.jar	JSTL 1.0
jakarta.jstl_1.0\lib\xml-apis.jar	JSTL 1.0
jakarta.jstl_1.1\lib\jstl.jar	JSTL 1.1 JSTL 1.0
jakarta.jstl_1.1\lib\standard.jar	JSTL 1.1 Tags JSTL 1.0 Tags
jsf.facelets_1.1.14\jsf-facelets.jar	Facelets Runtime
oracle.adf.dconfigbeans_11.1.1.jar	NA
oracle.adf.management_11.1.1\adf-em-config.jar	ADF Management Pages
oracle.adf.model.generic_11.1.1\bc4jdomgnrc.jar	BC4J Generic Domains
oracle.adf.model_11.1.1\redist\adfbinding-samples.jar	System Classpath
oracle.adf.model_11.1.1\redist\graphtags.jar	BC4J HTML
oracle.adf.share.ca_11.1.1\adf-share-base.jar	ADF Common Web Runtime MDS Runtime Dependencies ADF Model Generic Runtime BC4J Runtime ADF Swing Runtime BC4J Security ADF Common Runtime
oracle.adf.share.ca_11.1.1\adf-share-ca.jar	MDS Runtime Dependencies ADF Model Generic Runtime BC4J Runtime BC4J Security ADF Common Runtime
oracle.adf.share_11.1.1\adf-share-support.jar	MDS Runtime Dependencies ADF Model Generic Runtime BC4J Runtime BC4J Security ADF Common Runtime
oracle.adf.share_11.1.1\adf-share-wls.jar	NA

Table A-3 (Cont.) System Classpath

JAR	JDeveloper Library
oracle.adf.share_11.1.1\adflogginghandler.jar	MDS Runtime Dependencies BC4J Tester ADF Model Generic Runtime BC4J Runtime ADF Common Runtime
oracle.adf.share_11.1.1\adfsharebean.jar	BC4J Runtime ADF Common Runtime
oracle.adf.share_11.1.1\commons-cli-1.0.jar	MDS Runtime Dependencies
oracle.adf.share_11.1.1\commons-el.jar	ADF Model Runtime MDS Runtime Dependencies ADF Model Generic Runtime BC4J Runtime
oracle.adf.share_11.1.1\jsp-el-api.jar	ADF Model Runtime MDS Runtime Dependencies ADF Model Generic Runtime BC4J Runtime
oracle.adf.share_11.1.1\oracle-el.jar	ADF Model Runtime MDS Runtime Dependencies ADF Model Generic Runtime BC4J Runtime
oracle.auditprovider_11.1.1\jps-wls-auditprovider.jar	System Classpath
oracle.bulkops_11.1.1\bulkoperationsmbean.jar	NA
oracle.classloader_11.1.1.jar	NA
oracle.dconfig-infra_11.1.1.jar	NA
oracle.dms_11.1.1\dms.jar	MDS Runtime Dependencies Java EE 1.5 Essbase Data Control Runtime J2EE 1.4 Oracle JDBC JAX-RPC Client
oracle.ejb_11.1.1\ejbsvc-share.jar	EJB SDO Client

Table A-3 (Cont.) System Classpath

JAR	JDeveloper Library
oracle.fabriccommon_11.1.1.1\fabric-common.jar	Java EE 1.5 J2EE 1.4 Web Service Data Control JAX-RPC Client
oracle.fabriccommon_11.1.1.1\policy-api.jar	NA
oracle.help_5.0\help-indexer.jar	NA
oracle.help_5.0\help-share.jar	BC4J Tester Oracle Help for Java
oracle.help_5.0\help-wizard.jar	NA
oracle.help_5.0\ohj.jar	BC4J Tester Oracle Help for Java
oracle.help_5.0\ohw-rcf.jar	NA
oracle.help_5.0\ohw-share.jar	NA
oracle.help_5.0\ohw-uix.jar	NA
oracle.help_5.0\oracle_ice.jar	BC4J Tester Ice Oracle Help for Java
oracle.http_client_11.1.1.1.jar	URL Data Control Java EE 1.5 Oracle SOAP J2EE 1.4 Web Service Data Control JAX-RPC Client
oracle.iau_11.1.1.1\fmw_audit.jar	Java EE 1.5 J2EE 1.4 BC4J Security JAX-RPC Client
oracle.iau_11.1.1\reports\AuditReportTemplates.jar	NA
oracle.idm_11.1.1.1\identitystore.jar	ADF Model Runtime ADF Model Generic Runtime BC4J Runtime BC4J Security ADF Common Runtime
oracle.idm_11.1.1.1\identityutils.jar	NA
oracle.javacache_11.1.1.1\cache.jar	MDS Runtime Dependencies Java Cache ADF Common Runtime
oracle.javacache_11.1.1.1\jocconfmbean.jar	NA

Table A-3 (Cont.) System Classpath

JAR	JDeveloper Library
oracle.javatools_11.1.1.1\javamodel-rt.jar	Java EE 1.5 J2EE 1.4 JAX-RPC Client
oracle.javatools_11.1.1.1\javatools-nodeps.jar	ADF Common Web Runtime MDS Runtime Dependencies Java EE 1.5 ADFm Designtime API J2EE 1.4 JAX-RPC Client
oracle.javatools_11.1.1.1\resourcebundle.jar	ADF Desktop Integration Runtime Resource Bundle Support BC4J Runtime
oracle.jdbc_11.1.1.1\ojdbc6dms.jar	NA
oracle.jmx_11.1.1.1\jmxframework.jar	MDS Runtime Dependencies Java EE 1.5 BC4J Tester BC4J Runtime J2EE 1.4 JAX-RPC Client
oracle.jmx_11.1.1.1\jmxspi.jar	MDS Runtime Dependencies Java EE 1.5 BC4J Tester BC4J Runtime J2EE 1.4 JAX-RPC Client
oracle.jps_11.1.1.1\jacc-spi.jar	BC4J Security
oracle.jps_11.1.1.1\jps-api.jar	BC4J Tester BC4J Security JAX-RPC Client
oracle.jps_11.1.1.1\jps-common.jar	BC4J Security JAX-RPC Client
oracle.jps_11.1.1.1\jps-ee.jar	JPS Designtime BC4J Security
oracle.jps_11.1.1.1\jps-internal.jar	BC4J Security
oracle.jps_11.1.1.1\jps-manifest.jar	NA
oracle.jps_11.1.1.1\jps-mbeans.jar	NA
oracle.jps_11.1.1.1\jps-unsupported-api.jar	BC4J Security

Table A-3 (Cont.) System Classpath

JAR	JDeveloper Library
oracle.jps_11.1.1.1\jps-upgrade.jar	NA
oracle.jps_11.1.1.1\jps-wls.jar	NA
oracle.jrf_11.1.1.1\jrf-api.jar	JRF API
oracle.jrf_11.1.1.1\jrf-client.jar	JRF Client
oracle.jrf_11.1.1.1\jrf.jar	JAX-WS Client JRF Runtime
oracle.jsf_1.2.7.1\jsf-api.jar	JSF
oracle.jsf_1.2.7.1\jsf-ri.jar	NA
oracle.jsf_1.2.7.1\sun-commons-beanutils.jar	NA
oracle.jsf_1.2.7.1\sun-commons-collections.jar	NA
oracle.jsf_1.2.7.1\sun-commons-digester.jar	NA
oracle.jsf_1.2.7.1\sun-commons-logging.jar	NA
oracle.jsf_1.2.9\glassfish.jsf_1.2.9.0.jar	JSF 1.2
oracle.jsf_1.2.9\glassfish.jstl_1.2.0.1.jar	JSF 1.2 JSTL 1.2 JSTL 1.2 Tags
oracle.jsf_1.2.9\javax.jsf_1.2.0.1.jar	JSF 1.2 Java EE 1.5 API
oracle.jsf_1.2.9\wls.jsf.di.jar	JSF 1.2
oracle.jsp_11.1.1.1\ojsp.jar	NA
oracle.ldap_11.1.1.1\jremtool.jar	NA
oracle.ldap_11.1.1.1\ldapjclnt11.jar	BC4J Security
oracle.ldap_11.1.1.1\ojmisc.jar	BC4J EJB Client ADF Model Runtime BC4J Tester ADF Model Generic Runtime BC4J Runtime J2EE 1.4 JAX-RPC Client BC4J IAS Client
oracle.logging-utils_11.1.1.1.jar	NA
oracle.mds_11.1.1.1\mdslcm.jar	NA
oracle.mds_11.1.1.1\mdsrt.jar	MDS Runtime
oracle.mds_11.1.1.1\oramds.jar	MDS Runtime Dependencies
oracle.nlsrtl_11.1.1.0\orai18n-collation.jar	NA
oracle.nlsrtl_11.1.1.0\orai18n-mapping.jar	NA
oracle.nlsrtl_11.1.1.0\orai18n-servlet.jar	NA

Table A-3 (Cont.) System Classpath

JAR	JDeveloper Library
oracle.nlsrtl_11.1.1.0\orai18n-translation.jar	NA
oracle.nlsrtl_11.1.1.0\orai18n-utility.jar	NA
oracle.nlsrtl_11.1.1.0\orai18n.jar	Java EE 1.5 J2EE 1.4 JAX-RPC Client
oracle.oamprovider_11.1.1.1\oamAuthnProvider.jar	NA
oracle.oamprovider_11.1.1.1\oamcfgtool.jar	NA
oracle.oc4j_10.1.3.4\oc4jclient.jar	NA
oracle.odl_11.1.1.1\ojdl-log4j.jar	NA
oracle.odl_11.1.1.1\ojdl.jar	MDS Runtime Dependencies Java EE 1.5 Essbase Data Control Runtime J2EE 1.4 Oracle JDBC JAX-RPC Client
oracle.odl_11.1.1.1\ojdl2.jar	Java EE 1.5 J2EE 1.4 Web Service Data Control JAX-RPC Client
oracle.oer_11.1.1.1\activation-1.0.2.jar	NA
oracle.oer_11.1.1.1\aler-axis-1.2.1.jar	NA
oracle.oer_11.1.1.1\aler-axis-jaxrpc-1.2.1.jar	NA
oracle.oer_11.1.1.1\client.rex-10.3.1.0.jar	NA
oracle.oer_11.1.1.1\commons-discovery-0.2.jar	NA
oracle.oer_11.1.1.1\commons-logging-1.0.4.jar	NA
oracle.oer_11.1.1.1\mail-1.2.jar	NA
oracle.oer_11.1.1.1\oracle.jdeveloper.oer.jar	NA
oracle.oer_11.1.1.1\soap-2.2.jar	NA
oracle.oer_11.1.1.1\wsdl4j-1.6.2.jar	NA
oracle.osdt_11.1.1.1\ojdigsig.jar	NA
oracle.osdt_11.1.1.1\osdt_cert.jar	Java EE 1.5 J2EE 1.4 BC4J Security Web Service Data Control JAX-RPC Client
oracle.osdt_11.1.1.1\osdt_cmp.jar	NA
oracle.osdt_11.1.1.1\osdt_cms.jar	NA

Table A-3 (Cont.) System Classpath

JAR	JDeveloper Library
oracle.osdt_11.1.1.1\osdt_core.jar	Java EE 1.5 J2EE 1.4 BC4J Security Web Service Data Control JAX-RPC Client
oracle.osdt_11.1.1.1\osdt_core_fips.jar	Web Service Data Control
oracle.osdt_11.1.1.1\osdt_jce.jar	Web Service Data Control
oracle.osdt_11.1.1.1\osdt_ldap.jar	NA
oracle.osdt_11.1.1.1\osdt_lib_v11.jar	Web Service Data Control
oracle.osdt_11.1.1.1\osdt_lib_v12.jar	Web Service Data Control
oracle.osdt_11.1.1.1\osdt_ocsp.jar	NA
oracle.osdt_11.1.1.1\osdt_saml.jar	Java EE 1.5 J2EE 1.4 Web Service Data Control JAX-RPC Client
oracle.osdt_11.1.1.1\osdt_saml2.jar	Java EE 1.5 J2EE 1.4 Web Service Data Control JAX-RPC Client
oracle.osdt_11.1.1.1\osdt_smime.jar	NA
oracle.osdt_11.1.1.1\osdt_tsp.jar	NA
oracle.osdt_11.1.1.1\osdt_wss.jar	Java EE 1.5 J2EE 1.4 Web Service Data Control JAX-RPC Client
oracle.osdt_11.1.1.1\osdt_xkms.jar	NA
oracle.osdt_11.1.1.1\osdt_xmlsec.jar	Java EE 1.5 J2EE 1.4 BC4J Security Web Service Data Control JAX-RPC Client
oracle.osdt_11.1.1.1\ospnego.jar	NA
oracle.ossoiap_11.1.1.1\ossoiap.jar	NA
oracle.pki_11.1.1.1\oraclepki.jar	J2EE 1.4 BC4J Security JAX-RPC Client
oracle.pki_11.1.1.1\owm-3_0.jar	NA
oracle.pki_11.1.1.1\owm-images.jar	NA
oracle.pki_11.1.1.1\owm_help.jar	NA

Table A-3 (Cont.) System Classpath

JAR	JDeveloper Library
oracle.security-api_11.1.1.jar	NA
oracle.ssofilter_11.1.1\ssofilter.jar	NA
oracle.toplink_11.1.1\eclipselink-dbwsutils.jar	NA
oracle.toplink_11.1.1\toplink-grid.jar	NA
oracle.toplink_11.1.1\toplink-javadoc.jar	NA
oracle.toplink_11.1.1\toplink-oc4j.jar	NA
oracle.ucp_11.1.0.jar	MDS Runtime Dependencies
oracle.web-common_11.1.1.jar	NA
oracle.webservices_11.1.1\dbws.jar	NA
oracle.webservices_11.1.1\lwdom.jar	JAX-RPC Client
oracle.webservices_11.1.1\mdds.jar	J2EE 1.4 Web Service Data Control JAX-RPC Client
oracle.webservices_11.1.1\oc4j-ws-support.jar	NA
oracle.webservices_11.1.1\orasaa.jar	Java EE 1.5 J2EE 1.4 Web Service Data Control JAX-RPC Client
oracle.webservices_11.1.1\orawsdl.jar	Java EE 1.5 J2EE 1.4 Web Service Data Control JAX-RPC Client Oracle JWSDL
oracle.webservices_11.1.1\orawarm.jar	Java EE 1.5 J2EE 1.4 Web Service Data Control JAX-RPC Client
oracle.webservices_11.1.1\testpage.jar	NA
oracle.webservices_11.1.1\wsa.jar	NA
oracle.webservices_11.1.1\wsclient.jar	BC4J Service Runtime ADF Desktop Integration Runtime Java EE 1.5 Oracle JAX-WS Client JAX-RPC 11 Web Services J2EE 1.4 Web Service Data Control Oracle JAX-RPC Client OWSM Policy Lib

Table A-3 (Cont.) System Classpath

JAR	JDeveloper Library
oracle.webservices_11.1.1\wsif.jar	WSIF Client Java EE 1.5 J2EE 1.4 JAX-RPC Client
oracle.webservices_11.1.1\wssecurity.jar	Java EE 1.5 J2EE 1.4 Web Service Data Control JAX-RPC Client
oracle.webservices_11.1.1\wsserver.jar	Java EE 1.5 J2EE 1.4 JAX-RPC Client
oracle.webservices_11.1.1\ws_confmbeans.jar	NA
oracle.wsm.agent.common_11.1.1\wsm-agent-core.jar	Web Service Data Control
oracle.wsm.agent.common_11.1.1\wsm-agent-fmw.jar	NA
oracle.wsm.agent.common_11.1.1\wsm-agent-wls.jar	NA
oracle.wsm.agent.common_11.1.1\wsm-agent.jar	Web Service Data Control JAX-RPC Client
oracle.wsm.agent.common_11.1.1\wsm-pap.jar	Web Service Data Control JAX-RPC Client
oracle.wsm.common_11.1.1\wsm-audit-core.jar	NA
oracle.wsm.common_11.1.1\wsm-dependencies.jar	NA
oracle.wsm.common_11.1.1\wsm-pmlib.jar	Web Service Data Control JAX-RPC Client
oracle.wsm.common_11.1.1\wsm-policy-core.jar	Web Service Data Control JAX-RPC Client
oracle.wsm.common_11.1.1\wsm-secpol.jar	Web Service Data Control JAX-RPC Client
oracle.wsm.pm_11.1.1\wsm-pm-ejb-client-api.jar	NA
oracle.wsm.pm_11.1.1\wsm-pmserver.jar	NA
oracle.wsm.policies_11.1.1\wsm-policytool.jar	NA
oracle.wsm.policies_11.1.1\wsm-seed-policies.jar	Web Service Data Control
oracle.xdb_11.1.0.jar	J2EE 1.4 JAX-RPC Client
oracle.xdk_11.1.0\xml.jar	MDS Runtime Dependencies Oracle XML Parser v2 XSQL Runtime

Table A-3 (Cont.) System Classpath

JAR	JDeveloper Library
oracle.xdk_11.1.1.0\xmlparserv2.jar	BC4J EJB Client ADF Model Runtime MDS Runtime Dependencies Oracle XML Parser v2 BC4J Tester ADF Model Generic Runtime BC4J Runtime XSQL Runtime Oracle SOAP BC4J Security JAX-RPC Client BC4J IAS Client
oracle.xdk_11.1.1.0\xquery.jar	Oracle XQuery
oracle.xdk_11.1.1.0\xsu12.jar	XSQL Runtime Oracle XML SQL Utility
oracle.xds_11.1.1.1.jar	NA
oracle.xmldef_11.1.1.1\xmldef.jar	MDS Runtime Dependencies ADF Faces Change Manager Runtime 11 ADF Model Generic Runtime Obsolete JDeveloper Extension SDK
oracle.xqs-api_11.1.1.1.jar	NA
org.apache.bcel_5.1.jar	NA
org.apache.commons.beanutils_1.6.jar	Commons Beanutils 1.6.1 Commons Beanutils 1.6
org.apache.commons.digester_1.7.jar	Java EE 1.5 Commons Digester 1.5 Commons Digester 1.7 J2EE 1.4 JAX-RPC Client
org.apache.commons.logging_1.0.4.jar	Commons Logging 1.0.3 Commons Logging 1.0.4
org.dom4j_1.6.1.jar	NA
org.jaxen_1.1.1.jar	Java EE 1.5 J2EE 1.4 Web Service Data Control JAX-RPC Client

Table A-3 (Cont.) System Classpath

JAR	JDeveloper Library
org.objectweb.asm.attrs_1.5.3.jar	NA
org.objectweb.asm_1.5.3.jar	NA
org.osoa.sca_2.0.jar	NA
org.springframework_2.0.jar	Java EE 1.5 J2EE 1.4 JAX-RPC Client
sun.tools_1.6.0.jar	NA
ws.databinding.plugins_1.0.0.0.jar	NA
ws.databinding_1.0.0.0.jar	NA