**Oracle® Fusion Middleware**

Forms Services Deployment Guide

11*g* Release 2 (11.1.2)

**E24477-03**

November 2012

ORACLE®

Oracle Fusion Middleware Forms Services Deployment Guide 11*g* Release 2 (11.1.2)

E24477-03

Primary Author: Swati Thacker

Contributors: Ananth Satyanarayana, Vinay Agarwal, Suvarna Balachandra, Hemant Bansal, Ramesh Gurubhadraiah, Laiju Mathew, Gururaja Padakandla, Opendro Singh, Ashish Tyagi, Sudarshan Upadhya, Syed Nisar Ahmed, Dhiraj Madan, James Amalraj, Phil Kuhn, Arthur Housinger, Rubik Sadeghi, Naseer Syed, Emerson deLaubenfels, Grant Ronald

# Contents

## 4   Configuring and Managing Forms Services

# 5   Using Oracle Forms Services with the HTTP Listener and Oracle WebLogic Server

# 6 Oracle Forms and JavaScript Integration

# 7 Enhanced Java Support

# 8 Working with Server Events

# 9 Using Forms Services with Oracle Single Sign-On

## 10    Configuring and Managing Java Virtual Machines

## B    Configuring Java Plug-ins

## C    Locations and Samples of Configuration Files

## D    Forms Error Messages

## Index

# Preface

## Intended Audience

This manual is intended for software developers who are interested in deploying Oracle Forms applications to the Web with Oracle Fusion Middleware.

## Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at
http://www.oracle.com/us/corporate/accessibility/index.html.

### Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/support/contact.html or visit http://www.oracle.com/accessibility/support.html if you are hearing impaired.

## Related Documents

For more information, see the following manuals:

- *Oracle Fusion Middleware Release Notes for Linux x86*

- *Oracle Fusion Middleware Release Notes for Microsoft Windows*

- *Oracle Forms Upgrading Oracle Forms 6i* to Oracle Forms 11*g*

- Oracle Fusion Middleware Library on OTN

- Oracle Forms Builder Online Help, available from the Help menu in Oracle Forms Developer.

In addition, you will find white papers and other resources at
http://www.oracle.com/technetwork/developer-tools/forms/overview/index.html.

## 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. |
| *italic* | 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. |

# 1

# Introduction to Oracle Forms Services

This chapter introduces Oracle Forms. It provides an overview of the development and deployment environment for Oracle Forms, and provides references where you can find more information on associated components in Oracle Fusion Middleware.

This chapter contains the following sections:

## 1.1 Oracle Forms

Oracle Forms is a component of Oracle Fusion Middleware. Oracle Forms is used to develop and deploy Forms applications. The Forms applications provide a user interface to access Oracle Database in an efficient and tightly-coupled way. The applications can be integrated with Java and web services to take advantage of service oriented architectures (SOA).

Oracle Forms includes the following:

- Oracle Forms Developer, used to develop and compile Forms applications.

- Oracle Forms Services, a server component, used to deploy the applications.

### 1.1.1 Oracle Forms Developer

Oracle Forms Developer is used to develop a form that can access an Oracle database and present the data. Wizards and utilities are provided to speed up application development. The source form (*.fmb) is created and compiled into an "executable" (*.fmx). The Forms application is run (interpreted) by the Forms Runtime process.

For more information about the Oracle Forms Developer, refer to the following documentation:

- Oracle Forms Builder Online Help, which is accessible from Oracle Forms Builder, provides information on how to use Oracle Forms Developer to develop and compile Forms applications.

- *Upgrading Oracle Forms 6i to Oracle Forms 11g*: describes obsolete features of Oracle Forms Developer and instructions for upgrading your Forms applications.

### 1.1.2 Oracle Forms Services

Oracle Forms Services is a comprehensive application framework optimized to deploy Forms applications in a multitiered environment. It takes advantage of the ease and accessibility of the Web and elevates it from a static information-publishing mechanism to an environment capable of supporting complex applications.

The Form applications that you design and develop in Oracle Forms Developer are deployed on Oracle Fusion Middleware. These applications run on the middle tier (see Figure 1–2). The user interface is presented on the client tier as a Java applet in the client's browser.

This guide describes the configuration files, and environment variables that can be used to customize deployment of Forms applications. It also provides information on performance, logging and monitoring your deployment. You can use Oracle Fusion Middleware Enterprise Manager Control to manage the configuration files, and environment variables, and monitor the deployment.

### 1.1.3 How Oracle Forms Services Launches a Forms Application

When a user first starts an Oracle Forms application by clicking a link to the application's URL, the baseHTML file is read by the Forms servlet. Any variables (*%variablename%*) in the baseHTML file are replaced with the appropriate parameter values specified in the `formsweb.cfg` file, and from query parameters in the URL request (if any).

You can easily modify the configuration files with Oracle Enterprise Manager Fusion Middleware Control as per your requirements. You can also manually update the configuration files in those Oracle Forms installations that do not include the Enterprise Manager. Section 1.6, "Oracle Forms Services Architecture" describes the processes that are involved in deploying and running a typical Forms application.

## 1.2 Oracle Database

Oracle Database is the latest generation of RDBMS. Among the numerous capabilities are unlimited scalability and industry-leading reliability with Oracle Real Application Clusters; high availability technology including advancements in standby database technology (Oracle Data Guard); and built-in OLAP, data mining and Extract, Transform and Load (ETL) functions.

For more information on Oracle Database, refer to http://www.oracle.com/technetwork/indexes/documentation/index.html.

## 1.3 Oracle WebLogic Server

Oracle WebLogic Server 11*g* Release 1 is an application server for building and deploying enterprise Java EE applications with support for new features for lowering cost of operations, improving performance and supporting the Oracle applications portfolio.

Regardless of whether you want to create a staging, production, or testing environment, you begin by creating a WebLogic domain. A WebLogic domain includes instances of WebLogic Server, of which one is configured as an Administration Server. The Administration Server maintains configuration data for a domain. You can deploy your application on Administration Server but it is recommended to create a managed server and deploy your application in managed

server. For more information on Oracle WebLogic Server, refer to *Oracle Fusion Middleware Introduction to Oracle WebLogic Server*.

In the deployment mode, during configuration, a managed server for Forms is created (WLS_FORMS). For more information on WLS_FORMS, refer to Section 5.1, "About the Oracle WebLogic Managed Server."

## 1.4 Oracle Fusion Middleware

Oracle Fusion Middleware includes Web servers, application servers, content management systems, and developer tools that provide complete support for development, deployment, and management of software applications. Among the components are Oracle Forms Services, Oracle WebLogic Server, and Oracle Enterprise Manager Fusion Middleware Control, which together provide the technology to fully realize the benefits of Internet computing.

You can manage and monitor Oracle Forms using Oracle Enterprise Manager Fusion Middleware Control.

For a complete overview, list of components, and conceptual information about Oracle Fusion Middleware, refer to the following manuals:

- *Oracle Fusion Middleware Concepts*

- *Oracle Fusion Middleware Administrator's Guide*

## 1.5 About Installing or Upgrading Oracle Forms

In the installer, you can selectively configure any one of these products or all of them. For more information on installing Oracle Forms, refer to the following guides:

- *Oracle Fusion Middleware Installation Planning Guide*

- *Oracle Fusion Middleware Installation Guide for Oracle Portal, Forms, Reports and Discoverer*

- *Oracle Fusion Middleware Quick Installation Guide for Oracle Portal, Forms, Reports, and Discoverer*

For upgrade information, refer to the following documents:

- *Oracle Fusion Middleware Upgrade Planning Guide*

- *Oracle Fusion Middleware Upgrade Guide for Oracle Portal, Forms, Reports, and Discoverer*

For information on upgrading Forms 6*i* to Oracle Forms 11*g*, see Chapter 13, "Upgrading to Oracle Forms Services 11g."

For information about changed or obsolete features, see the *Oracle Forms Upgrading Oracle Forms 6i to Oracle Forms 11g Guide*.

> **Note:** If you have installed Oracle Forms using the **Deployment** installation option, you can use the following options to save RAM in a development-only environment:
>
> - You can choose to stop `WLS_REPORTS` (or other managed servers that may be running). To test Forms applications, only `WLS_FORMS` is required.
>
> - By default, `formsapp` and `formsconfigmbeans` run on `WLS_FORMS`. You can retarget these applications to run on the Administration Server and stop the Forms managed server (`WLS_FORMS`).

## 1.6 Oracle Forms Services Architecture

Figure 1–1 shows the three-tier architecture that makes up Forms Services:

- The **client tier**, at the top of the image, contains the Web browser, where the application is displayed. In addition to the browser, Java Runtime Environment (JRE) and Java Plug-In (JPI) are required. For more information, see Appendix B, "Configuring Java Plug-ins" and `http://www.oracle.com/technetwork/java/index-jsp-142903.html#documentation`.

- The **middle tier,** in the center of the image, is the application server, where application logic and server software are stored.

- The **database tier**, in the lower portion of the image, is the database server, where database server software is stored.

**Figure 1–1   Oracle Forms Services Architecture**
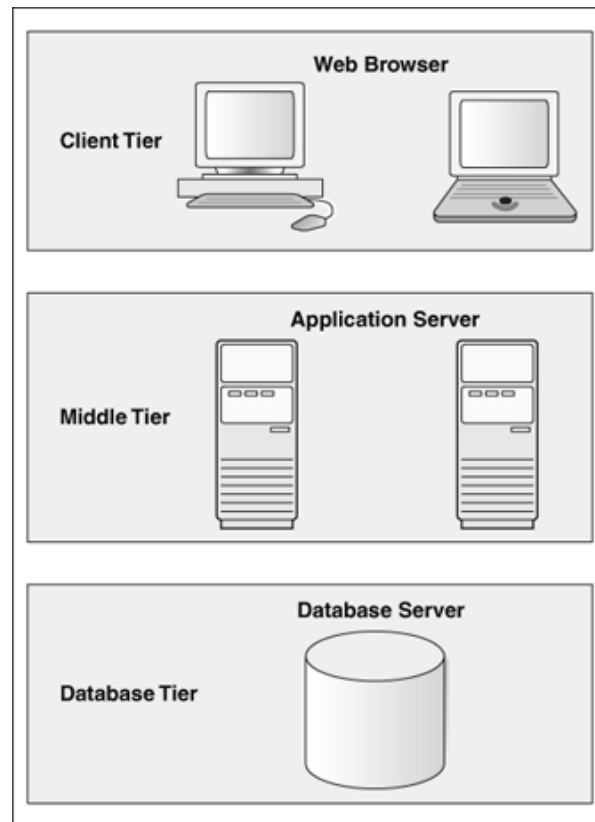


## 1.6.1  Oracle Forms Services Components

Oracle Forms Services is a middle-tier application framework for deploying complex, transactional forms applications to a network such as an intranet or the Internet. Developers build Forms applications with Forms Developer and deploy them with Forms Services. Developers can also take current applications that were previously deployed in client/server and move them to a three-tier architecture. Some minor changes in application code may be required when moving to a three-tier architecture.

As shown in Figure 1–2, the three-tier configuration for running a form consists of:

- The **Client**, at the top of the image, resides on the client tier

- The Forms Listener servlet, in the center of the image, resides on the middle tier

- The **Forms Runtime process**, also resides on the middle tier

*Figure 1–2   Three-tier configuration for running a form*
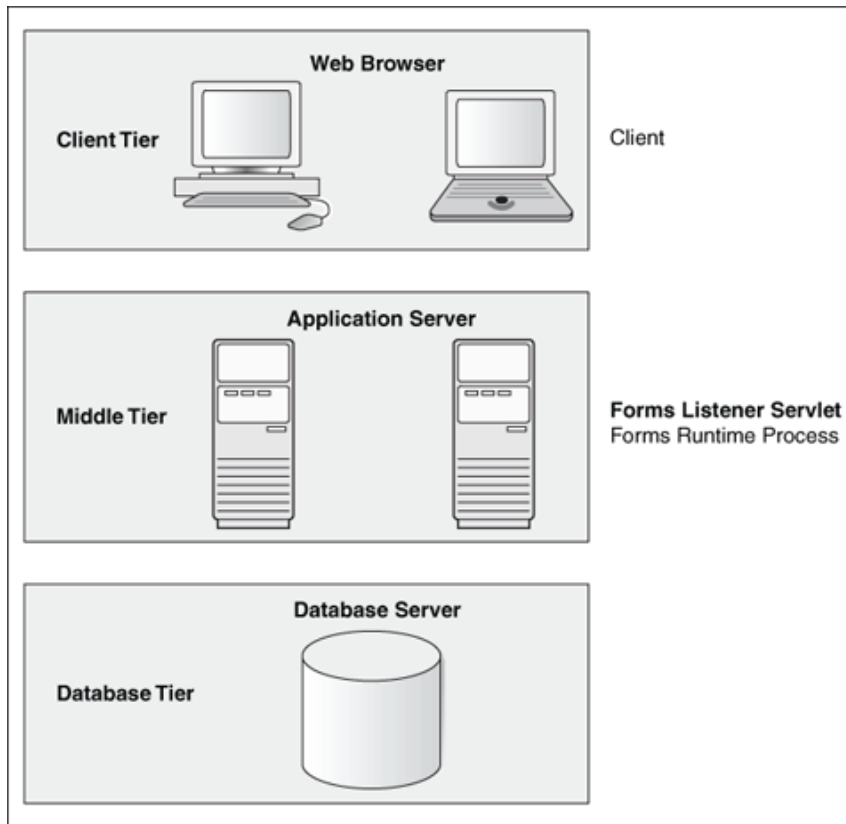


### 1.6.1.1 Forms Listener Servlet

The Forms Listener servlet is a broker between the Java client and the Forms Runtime process. It takes connection requests from Java client processes and initiates a Forms Runtime process on their behalf.

Figure 1–3 illustrates how the client sends HTTP requests and receives HTTP responses from Forms Services. Oracle Forms Services uses the Forms Listener servlet to start, stop, and communicate with the Forms Runtime process. In this image, the client is to the left. In the center of the image, the HTTP Listener acts as the network endpoint for the client, keeping the other server computers and ports from being exposed at the firewall.

The Forms Runtime process, in the right side of the image, executes the code contained in a particular Forms application. The Forms Listener servlet manages the creation of a Forms Runtime process for each client and manages the network communications between the client and its associated Forms Runtime process.

> **Note:**   The Forms Listener servlet is configured for you during the Oracle Fusion Middleware installation process.

*Figure 1–3   Architecture using the Forms Listener Servlet*
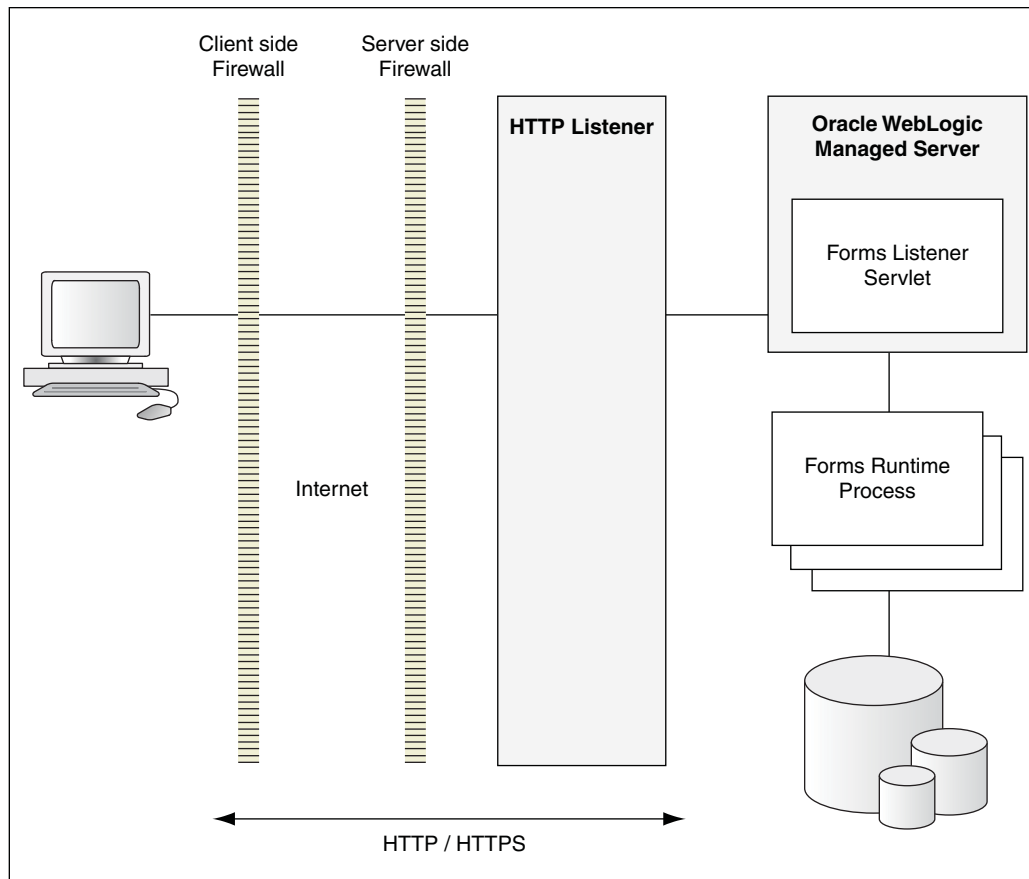


### 1.6.1.2 Forms Runtime Process

The Forms Runtime process plays two roles: when it communicates with the **client browser**, it acts as a server by managing requests from client browsers and it sends metadata to the client to describe the user interface; when it is communicating with the **database server**, it acts as a client by querying the database server for requested data.

For each Oracle Forms session, there is one Oracle Forms Runtime process on the application server. This process is where Oracle Forms actually runs, and manages application logic and processing. It also manages the database connection; queries and updates data; runs any PL/SQL in the Form; executes triggers; and so on. It uses the same forms, menus, and library files that were used for running in client/server mode.

The Forms Runtime process also contains the Java Virtual Machine (JVM) to run Java in your application. As an optimization feature, the JVM is started if the Forms application uses the Java Importer. In 10*g*, the JVM pooling feature is used only by the Java Importer. In 11*g*, Forms Runtime Process no longer creates a separate JVM when it calls Reports. Instead, if a JVM controller is configured for a form, the form can use the shared JVM when calling Reports. This results in a reduction of memory consumption, freeing more resources on the server. For more information about managing JVM usage and pooling, see Chapter 10, "Configuring and Managing Java Virtual Machines."

# 2

# What is New in Oracle Forms Services

This chapter describes the features and improvements in 11*g* Release 2 of Oracle Fusion Middleware Forms Services.

- Section 2.1, "Integration with Oracle Access Manager (OAM)"
- Section 2.2, "Schedule Forms Runtime Prestart"
- Section 2.3, "Forms Metric Agent"
- Section 2.4, "Enhanced Network Statistics Support"
- Section 2.5, "Support for Unicode Columns"
- Section 2.6, "Oracle Real User Experience Insight (RUEI)"
- Section 2.7, "Support for URLs in Image Items and Iconic Buttons"
- Section 2.8, "guiMode Configuration Parameter"

For the list of new features in Oracle Forms Services 11*g* Release 1, see "What's New in Oracle Forms Services" in *Oracle Fusion Middleware Forms Services Deployment Guide*.

## 2.1 Integration with Oracle Access Manager (OAM)

Oracle Access Manager 11*g* is a Java Platform, Enterprise Edition (Java EE)-based enterprise-level security application that provides restricted access to confidential information and centralized authentication and authorization services. Oracle Forms 11*g* Release 2 supports Oracle Access Manager (OAM) as the authentication server in Single Sign-On mode with mod_osso and webgate as the access clients.

For more information, see Chapter 9, "Using Forms Services with Oracle Single Sign-On".

## 2.2 Schedule Forms Runtime Prestart

The previous release of Oracle Forms allowed you to manage the startup of a configurable number of Forms Runtime engines prior to their usage. With Oracle Forms 11*g* Release 2, you can now schedule the prestart of Forms Runtime processes on a more flexible basis, at any appropriate time. You can schedule a Forms Runtime prestart, view existing schedules, delete any existing schedule, and export and import a schedule using the Enterprise Manager Fusion Middleware Control.

For more information, see Section 14.1.2.3, "Scheduling Runtime Pooling".

## 2.3 Forms Metric Agent

Forms Diagnostics Agent allows the user to collect Forms metrics in a repository. This feature enables the user to provide forms metrics information (that is available with DMS) into the database tables so that the users can access these metrics as historical data. The agent application provides an interface that enables the user to record the frequency of data collection as input and manage the starting or stopping of data collection.

For more information, see Chapter 15, "Forms Diagnostics Agent".

## 2.4 Enhanced Network Statistics Support

Use the Network Statistics feature that is now available with Oracle Forms 11*g* Release 2 to help diagnose performance related issues. The networkStats parameter enables the display of the aggregate statistics in the status bar. This feature provides the number of round trips between the Forms client and server, as well as the total number of bytes exchanged between client and server. The Network Statistics feature is enabled by specifying the networkStats parameter in the base HTML files.

For more information, see Section 4.2.5.7, "Advanced Configuration Parameters".

## 2.5 Support for Unicode Columns

A **Unicode Column** is a database column whose datatype is NCHAR or NVARCHAR2. Oracle Forms 11*g* Release 2 introduces a new datatype of NCHAR. Forms datatype NCHAR corresponds to the SQL datatypes NCHAR and NVARCHAR2.

For more information, see *Data Type Property* and *WHERE Clause Property* in Forms Builder Online Help.

## 2.6 Oracle Real User Experience Insight (RUEI)

Oracle Real User Experience Insight (RUEI) is a feature of Oracle Fusion Middleware that provides non-intrusive monitoring. It gives an insight into how a user interacts with an application. Oracle Forms 11g Release 2 has added new features to allow fine-grained recording of information in RUEI. This includes adding new arguments to the MESSAGE built-in, allowing application developers to effectively define their own logical units of work that they want timing information about.

## 2.7 Support for URLs in Image Items and Iconic Buttons

The previous versions of Oracle Forms allowed the users to specify a file name as image source for Image Items and Iconic Buttons. With Oracle Forms 11*g* Release 2, the user can also specify a URL as image source for Image items and Iconic Buttons.

For more information, see *READ_IMAGE_FILE Built-in* and *Icon Name Property* in Forms Builder Online Help.

## 2.8 guiMode Configuration Parameter

The guiMode parameter is designed to control the runtime GUI of a Form application. This applet configuration parameter gives the user the option to disable/enable the visibility of the default menubar and Windows title bar in the Forms application to allow for more seamless integration of the Forms applet into other HTML pages.

For more information, see Section 3.4.3, "guiMode Configuration Parameter".

# 3

# Basics of Deploying Oracle Forms Applications

This chapter describes how Forms Services run in Oracle Fusion Middleware, and describes the steps to deploy Forms applications. This chapter also describes the basic configuration files. After installation is completed, you can use the information in this chapter to change your initial configuration or make modifications as your needs change.

This chapter contains the following sections:

- Section 3.1, "Oracle Forms Services in Action"
- Section 3.2, "Configuration Files"
- Section 3.3, "Application Deployment"
- Section 3.4, "Client Browser Support"

## 3.1 Oracle Forms Services in Action

This section describes how Forms Services run in Oracle Fusion Middleware, and how the configuration files are used, with the assumption that the Forms servlet is used to generate the initial HTML page. For example, assume the Web server is running on port 8888 on a computer called "example.com". Also assume no modifications have been made to the standard configuration created during the Oracle Fusion Middleware installation process.

When a user runs an Oracle Forms Services application, the following sequence of events occur:

1. The user starts the Web browser and goes to a URL such as:

   ```
   http://example.com:8888/forms/frmservlet?config=myapp&form=hr
   app
   ```

   In this example, the top level form module to be run is called "hrapp" using the configuration section called "myapp".

2. Oracle HTTP Server listener receives the request. It finds `/forms` path in the URL and forwards the request to the correct Oracle WebLogic Managed Server based on the WebLogic handler mappings. The mapping is defined in `forms.conf`.

3. Oracle WebLogic Managed Server maps the request to the Oracle Forms Services application that has a context root named `/forms`. It maps the request to the Forms servlet using the frmservlet mapping specified in the `web.xml` file.

4. The Forms servlet running on the Oracle WebLogic Managed Server processes the request. The Forms servlet:

   ■ Opens the servlet configuration file (`formsweb.cfg` by default), which is located in `$DOMAIN_HOME/config/fmwconfig/servers/WLS_FORMS/applications/formsapp_11.1.2/config`.

   ■ Determines which configuration section to use in the `formsweb.cfg` file. In this example, the URL contains the query parameter `config=myapp`, therefore, the [`myapp`] section is used.

   ■ Determines which baseHTML file to use, based on (a) what browser (user-agent) made the request, (b) what platform the browser is running on, and (c) the settings of various parameters in the `formsweb.cfg` file (specifically, `basejpi.htm`, and `base.htm`).

   ■ Reads the baseHTML file, and returns the contents as an HTML page to the user's Web browser, after performing variable substitutions as follows:

   Whenever a variable (like `%myParam%`) is encountered, the Forms servlet looks for a matching URL query parameter (for example, `&myParam=xxx`), or, failing that, looks for a matching parameter in the `formsweb.cfg` file. If a matching parameter is found, the variable (`%myParam%`) is replaced with the parameter value.

   In this example, the baseHTML file contains the text `%form%`. This is replaced with the value "hrapp".

5. Depending on which baseHTML file the Forms servlet selected, the HTML page returned to the Web browser contains an applet, object or embed tag to start the Forms applet (thin client). The Forms client runs in the JVM environment provided by Oracle Java plug-in.

6. In order to start the Forms applet, its Java code must first be loaded. The location of the applet is specified by the applet codebase and archive parameters.

   The virtual path definition in the `weblogic.xml` file for `/forms/java` allows the applet code to be loaded from the Web server.

   **Note:** The Forms applet code is only loaded over the network the first time the user runs an Oracle Forms Services application or if a newer version of Oracle Forms Services is installed on the Web server. Otherwise, it is loaded from the cache of the Java plug-in on the local disk.

7. Once the Oracle Forms Services applet is running, it starts a Forms session by contacting the Forms Listener servlet at URL `http://example.com:8888/forms/lservlet`.

8. The Oracle HTTP Server listener receives the request. It forwards the request to Oracle WebLogic Managed Server, since the path `/forms/lservlet` matches a servlet mapping in the `web.xml` file (the one for the Forms Listener servlet).

9. The Forms Listener servlet (lservlet) starts a Forms run-time process (`frmweb.exe` or `frmweb`) for the Forms session.

10. Communication continues between the Forms applet and the Forms run-time process, through the Listener Servlet, until the Forms session ends.

11. The attribute value in a URL (such as the name of the form to run) is passed to the Forms run-time process. Part of the serverArgs value in the baseHTML file is `%form%`, which is replaced by "hrapp". Therefore, the run-time process runs the form in the file "hrapp.fmx".

This file must be present in any of the directories named in the FORMS_PATH environment setting, which is defined in the environment file (default.env by default). You can also specify the directory in formsweb.cfg (for example, form=c:\<path>\myform).

12. The Forms sessions end when either of the following occurs:

- The top-level form is exited (for example, by the PL/SQL trigger code which calls the "exit_form" built-in function). The user is prompted to save changes if there are unsaved changes. exit_form(no_validate) exits the form without prompting.

- If the user quits the Web browser, any pending updates are lost.

## 3.2 Configuration Files

This section introduces the basic files used to configure Forms applications. For more advanced configuration topics, see Chapter 4, "Configuring and Managing Forms Services."

This section contains the following:

- Section 3.2.1, "Oracle Forms Configuration Files"
- Section 3.2.2, "Forms Java EE Application Deployment Descriptors"
- Section 3.2.3, "Oracle HTTP Listener Configuration File"
- Section 3.2.4, "Standard Fonts and Icons File"
- Section 3.2.5, "baseHTML Files"
- Section 3.2.6, "WebUtil Configuration Files"

> **Note:** Location of files are given relative to the DOMAIN_HOME and ORACLE_INSTANCE directory. Forward slashes should be replaced by back slashes on Windows. For more information on terminology used such as Middleware home, Oracle home, Oracle instance, and so on, see the *Oracle Fusion Middleware Administrator's Guide*.

### 3.2.1 Oracle Forms Configuration Files

Oracle Forms configuration files allow you to specify parameters for your Forms. You can manage these files through the Oracle Enterprise Manager Fusion Middleware Control. These configuration files include:

- default.env
- formsweb.cfg
- ftrace.cfg

> **Note:** For a list of Forms configuration files and their respective locations, refer to Table C–1.

#### 3.2.1.1 default.env

Location: $DOMAIN_HOME/config/fmwconfig/servers/<MANAGED_SERVER>/applications/<appname>_<appversion>/config

Typically, this location is `$DOMAIN_HOME/config/fmwconfig/servers/WLS_FORMS/applications/formsapp_11.1.2/config`

This file contains environment settings for Forms run time. On UNIX and Linux, `default.env` includes the `PATH` and `LD_LIBRARY_PATH`.

For a sample `default.env` file, see Appendix C.3, "Platform Specific default.env Files."

For more information about `default.env`, see Chapter 4.3, "Managing Environment Variables."

#### 3.2.1.2  formsweb.cfg

Location: `$DOMAIN_HOME/config/fmwconfig/servers/<MANAGED_SERVER>/applications/<appname>_<appversion>/config`

Typically, this location is `$DOMAIN_HOME/config/fmwconfig/servers/WLS_FORMS/applications/formsapp_11.1.2/config`

This Forms configuration file contains the following:

- Values for Forms run-time command line parameters, and the name of the environment file to use (`envFile` setting).

- Most of the servlet configuration parameter settings that you set during installation. You can modify these parameters, if needed.

Variables (`%variablename%`) in the `base.htm` file are replaced with the appropriate parameter values specified in the `formsweb.cfg` file and from query parameters in the URL request (if any).

For a sample `formsweb.cfg` file, see Appendix C.2, "Default formsweb.cfg."

For more information about `formsweb.cfg`, see Chapter 4.2.2, "Configuring Parameters with Fusion Middleware Control."

#### 3.2.1.3  ftrace.cfg

Location:  `$ORACLE_INSTANCE/config/FormsComponent/forms/server`

This file is used to configure Forms Trace. Forms Trace replaces the functionality that was provided with Forms Runtime Diagnostics (FRD) and Performance Event Collection Services (PECS), which were available in earlier releases of Oracle Forms. Forms Trace traces the execution path through a form (for example, steps the user took while using the form).

For more information about `ftrace.cfg`, see Chapter 12, "Tracing and Diagnostics."

### 3.2.2  Forms Java EE  Application Deployment Descriptors

The Forms Services Java EE application EAR (Enterprise Archive) file `formsapp.ear` is deployed to the WLS_FORMS (Oracle WebLogic Managed Server) when you configure Oracle Forms.

This results in the creation of a directory structure under  `$DOMAIN_HOME/servers/WLS_FORMS/tmp/_WL_user/formsapp_11.1.2/<random_string1>/APP-INF` directory that is similar to the following:

```
./APP-INF
./APP-INF/lib
./APP-INF/lib/frmconfig.jar
./APP-INF/lib/frmconfigmbeans.jar
./META-INF
```

```
./META-INF/application.xml
./META-INF/jazn-data.xml
./META-INF/jps-config.xml
./META-INF/mbeans.xml
./META-INF/weblogic-application.xml
```

This following directory structure is created under `$DOMAIN_HOME/servers/WLS_FORMS/tmp/_WL_user/formsapp_11.1.2/<random_string2>/war/WEB-INF` directory.

```
./WEB-INF
./WEB-INF/lib
./WEB-INF/lib/frmsrv.jar
./WEB-INF/web.xml
./WEB-INF/weblogic.xml
```

> **Note:** The sub-directories in `$DOMAIN_HOME/servers/WLS_FORMS/tmp/_WL_user/formsapp_11.1.2` are created by the nostage deployment process of Oracle WebLogic Server. They are named with a random string. For example, e18uoi, wb1h9e and so on.

Deployment descriptors:

- `application.xml` and `weblogic-application.xml` define the structure of the EAR file.
- `web.xml` defines the aliases `frmservlet` and `lservlet` for the Forms servlet and the Forms Listener servlet.
- `weblogic.xml` defines the context parameters and any user defined virtual directory mappings.

For a sample `web.xml` file, see Appendix C.5, "web.xml."

### 3.2.3 Oracle HTTP Listener Configuration File

This section describes the file used to configure Oracle HTTP Listener for Oracle Forms Services.

Location: `$ORACLE_INSTANCE/config/OHS/<OHS INSTANCE NAME>/moduleconf`

`forms.conf` is the Oracle HTTP listener configuration file for Oracle Forms Services. It includes Forms Services related directives, like Forms WebLogic Managed Server handler mappings and AliasMatch.

#### 3.2.3.1 About Editing forms.conf

`forms.conf` is an Oracle HTTP Server directives file. In Oracle Fusion Middleware, the `forms.conf` file is included in the Oracle HTTP Server configuration directory at `$ORACLE_INSTANCE/config/OHS/<OHS INSTANCE NAME>/moduleconf`.

If you add any custom Oracle HTTP Server directives to `forms.conf,` you must restart the Oracle HTTP Server node where it resides.

For more information about `forms.conf,` see Appendix C.7, "forms.conf."

### 3.2.3.2  Configuring OHS on a Separate Host

If you choose to configure Oracle HTTP Server on a separate host, then perform the following tasks:

1. Copy the Forms OHS directives file, `forms.conf.backup` from the tier hosting Forms to the tier hosting OHS and rename it to `forms.conf`.

   Source location (on Forms tier):

   ```
   $ORACLE_
   INSTANCE/config/FormsComponent/forms/server/forms.conf.backup
   ```

   Destination location (on OHS tier):

   ```
   $ORACLE_INSTANCE/config/OHS/<OHS Component
   Instance>/moduleconf/forms.conf
   ```

2. Specify the appropriate managed server cluster or the managed server for the default forms Java EE application context root (`/forms`).

   Example of cluster entry:

   ```
   <Location /forms>
           SetHandler weblogic-handler
           WebLogicCluster    <HOSTNAME>:<WLS_PORT>
           DynamicServerList OFF
   </Location>
   ```

   Example of non-cluster entry:

   ```
   <Location /forms>
           SetHandler weblogic-handler
           WebLogicHost = <HOSTNAME>
           WebLogicPort = <PORT>
   </Location>
   ```

3. Make sure that any directories referenced in user-added directives are accessible on the OHS tier.

4. Restart OHS instance on the OHS tier.

## 3.2.4  Standard Fonts and Icons File

`Registry.dat` is the file that contains the default font, font mappings, and icon information that Forms Services uses.

Location: `$DOMAIN_HOME/config/fmwconfig/servers/WLS_ FORMS/applications/formsapp_ 11.1.2/config/forms/registry/oracle/forms/registry`

For a sample of the default `Registry.dat`, see Appendix C.8, "Registry.dat."

For more information about Registry.dat, see Chapter 4.7, "Deploying Fonts, Icons, and Images Used by Forms Services."

## 3.2.5  baseHTML Files

Location: `$ORACLE_INSTANCE/config/FormsComponent/forms/server/`

The `base.htm` and `basejpi.htm` are used as templates by the Forms servlet when generating the HTML page used to start an Oracle Forms application.

Oracle recommends that you make configuration changes in the `formsweb.cfg` file using Enterprise Manager and avoid editing these files. To change the baseHTML files, create your own versions and reference them from the `formsweb.cfg` file by changing the appropriate settings.

For a sample baseHTML file, see Appendix C.4, "base.htm and basejpi.htm Files."

## 3.2.6  WebUtil Configuration Files

This section describes the files used to configure WebUtil at run time. For information about using WebUtil at design time, see the Oracle Forms Developer Online Help. WebUtil configuration files include:

- Default webutil.cfg
- Default webutilbase.htm
- Default webutiljpi.htm

### 3.2.6.1  Default webutil.cfg

Location: `$ORACLE_INSTANCE/config/FormsComponent/forms/server.`

This file provides all of the configuration settings for WebUtil, including:

- Logging Options
- Installation Options
- File Upload and Download Options
- Server Side Logging Options for logging errors and log messages

For a sample of the `webutil.cfg` file, see Appendix C.10, "Default webutil.cfg."

> **Note:**

### 3.2.6.2  Default webutilbase.htm

Location: `$ORACLE_INSTANCE/config/FormsComponent/forms/server/`

This is the default baseHTML file for running a form on the Web using a generic APPLET tag to include a Forms applet with a certificate registration for WebUtil.

For a sample of the `webutilbase.htm` file, , see Appendix C.11, "Default webutilbase.htm."

### 3.2.6.3  Default webutiljpi.htm

Location: `$ORACLE_INSTANCE/config/FormsComponent/forms/server/`

This is the default baseHTML file for running a form on the Web using the JDK Java Plugin. For example, this file can be used when running a form on the Web with Firefox on UNIX and a certificate registration for WebUtil.

For a sample of the `webutiljpi.htm` file, , see Appendix C.12, "Default webutiljpi.htm."

## 3.3  Application Deployment

Once you have created your application in Forms Developer, you are ready for application Web deployment. Oracle Forms Services accesses an application in Oracle

Fusion Middleware through a specified URL. The URL then accesses the HTTP Listener, which communicates with the Listener Servlet. The Listener Servlet starts a Forms run-time process (`frmweb.exe` on Windows or `frmweb` on UNIX and Linux) for each Forms Services session.

For more information about how Forms Services run, see Section 3.1, "Oracle Forms Services in Action."

### 3.3.1 Deploying Your Application

To deploy a basic form with the default parameters set up by the installer:

1. Create your application in Forms Developer and save it.

   The .fmb file is a design time file that can only be opened in Forms Developer. The .fmx file is the run-time file created when you compile the .fmb and is used for Web deployment.

   For more information about Forms Developer, see the Help menu in Forms Developer.

2. Modify the `formsweb.cfg` file so that Oracle Forms Services can access your application module. You edit this file in the **Web Configuration** page of Fusion Middleware Control. For more information, see Section 4.2, "Configuring Forms Services".

   Table 3–1 shows the configuration of an application called "my_application" with a form module called "`form=hrapp.fmx`":

*Table 3–1    Example of Configuration Section Parameter Values*

| Configuration Section Name | Forms Module Name Value |
|---|---|
| my_application | hrapp.fmx |

   When configured, the Oracle Forms Services module hrapp.fmx is accessible on the Web by entering "`...?config=my_application`" in the browser URL (the name of the **Web Configuration** section in `formsweb.cfg`).

   > **Note:**   The name of the configuration section must not include spaces and must contain only alphanumeric characters.

3. Make sure the .fmx file location is specified in the `FORMS_PATH` environment variable.

   For example, in Windows, if your .fmx file is located in `d:\my_files\applications`, in the `FORMS_PATH`, include `d:\my_files\applications`. On Windows, use semi-colons to separate directory locations if specifying multiple locations. On UNIX/Linux, use colons for separators. Specify this information in the **Environment Configuration** page for the environment file.

4. To modify an environment file, select the file in the **Environment Configuration** page of Fusion Middleware Control and add or edit environment variables as needed by your application. For example, you can add the environment variable shown in Table 3–2.

*Table 3–2   Example of Environment Variable Values*

| Environment Variable Name | Environment Variable Value |
|---|---|
| NLS_LANG | `NLS_LANG=GERMAN_GERMANY.WE8ISO8859P1` |

If you specified these environment variables in an environment file, specify this environment file in the respective configuration section of the `formsweb.cfg` in the **Web Configuration** page.

**5.** Enter the name of your application in the URL as shown:

`http://example.com:8888/forms/frmservlet?`

where "example" is the hostname of your computer and "8888" is the port used by your HTTP Listener.

Once you have created a configuration section, add "config=" and the name of the configuration section. In this example, the URL to access `hrapp.fmx` is:

`http://example.com:8888/forms/frmservlet?config=my_`
`application`

## 3.3.2  Specifying Parameters

There are two ways to predefine parameter values for your Oracle Forms Services applications. You can define parameters by:

- Editing your application settings in the default section of the **Web Configuration** page of Fusion Middleware Control. The default configuration section displays the default values that are used by Oracle Forms Services.

- Managing (adding, editing, copying, deleting) other system and user parameter values in the named application configuration section (see Section 3.3.3, "Creating Configuration Sections in Fusion Middleware Control"). For example, in the configuration section you create for `myApp`, you can add or change these parameters and their values, as shown in Table 3–3.

*Table 3–3   Example Configuration Section: Parameter Values for myApp*

| Parameter Name | Parameter Value |
|---|---|
| baseHTML | mybase.htm |
| baseHTMLjpi | mybasejpi.htm |
| form | hrapp.fmx |
| userid | scott/tiger@orcl |

> **Note:** Parameters specified in the named configuration section of a **Web Configuration** override the settings in the default section.

> **Note:** System Parameters cannot be overridden in the URL, while user parameters can.

### 3.3.3 Creating Configuration Sections in Fusion Middleware Control

Under the configuration sections you created in step 2 of Section 3.3.1, "Deploying Your Application", you can specify parameters for your Oracle Forms Services applications. You can specify any application and system parameters that are available in the default section for **Web Configuration** page.

For example, you can set the look and feel of the application to the Oracle look and feel by setting the `lookAndFeel` parameter to the value of `oracle` and clicking **Apply**.

You can also override the default parameter values in the named configuration section. For example, to predefine the connect information of an application to `scott/tiger@orcl`, the parameter value for `userid` must be set in the named configuration section by changing the parameter value of `userid` to `scott/tiger@orcl`.

For other parameters that you can edit, see Chapter 4.2.5, "Forms Configuration Parameters."

#### 3.3.3.1 Editing the URL to Access Oracle Forms Services Applications

You can directly type parameters in the URL that accesses your Oracle Forms Services application. Using the previous example, instead of specifying the `form` parameter in your configuration file, you could also type it into the URL as follows:

```
http://example.com:8888/forms/frmservlet?config=my_application&form=hrapp
```

You can use the ampersand (&) to call a combination of a form and named configuration parameters. In the above example, you are calling the form "hrapp" with the parameter settings you specified in "my_application".

> **Note:** Parameters specified in the URL override the parameters set in the configuration section. See Chapter 4.5, "Managing URL Security for Applications" for more information.

### 3.3.4 Specifying Special Characters in Values of Runform Parameters

Certain considerations apply if values passed to runform parameters contain special characters. This section describes these considerations, and compares the default behavior in this release with the behavior in prior releases.

Runform parameters are those that are specified in the serverArgs applet parameter of the template HTML file. The value specified for the serverArgs parameter in the template HTML file, after variable substitution, is sometimes referred to as the command-line parameters string. It consists of a series of blank-separated `name=value` pairs. The name must consist solely of alphanumeric or underscore characters. The value portion of a name=value pair can be an arbitrary string.

#### 3.3.4.1 Default Behavior in the Current Release

The value of a runform parameter can be specified in one of three places:

1. In the value of the `serverArgs` parameter in the template HTML file (for example, base.htm).

2. In the value of a variable specified in the configuration file (for example, `formsweb.cfg`), which is substituted (directly or recursively) for a variable reference in (1). Such values are typically maintained using Fusion Middleware Control; see Chapter 4.2, "Configuring Forms Services."

**3.** As an attribute value in a URL, which is substituted directly for a variable reference in (1) or (2).

For case (3), URL syntax rules (as enforced by the browser and the application server) require that certain characters be entered as URL escape sequences ('%' followed by 2 hexadecimal digits representing the ASCII value of the character, for a total of three characters).

This requirement includes the `%` character itself (which must be entered as `%25`). In addition, Oracle Forms Services currently requires that the quote character ('"') be entered as `%22`, even if the browser and the application server allow a quote to be entered without escaping.

URL syntax rules also allow a space to be entered as a + (as an alternative to the URL escape sequence `%20`). However in the value of the `otherparams` configuration parameter, a + is treated specially; it separates name=value pairs as opposed to indicating a space embedded in the value of a runform parameter.

For example, if a runform application has user parameters `param1` and `param2`, and you want to assign them the values 'a b' and 'c d', you do so by incorporating the following into a URL:

```
&otherparams=param1=a%20b+param2=c%20d
```

When specifying runform parameters in the template HTML files or in the configuration files (cases (1) and (2)), Forms requires URL escape sequences in some circumstances, allows them in others, and forbids them in still others.

Outside of the values of runform parameters, URL escape sequences must not be used. For example, the = in a name=value pair must always be specified simply as =, and the space that separates two adjacent name=value pairs must always be specified simply as " " (a single space character).

Within the value of a runform parameter, space (' ') must be specified as a URL escape sequence (`%20`). The HTML delimiter character (specified in the configuration file) must also be specified as a URL escape sequence. And when the runform parameter is specified in the template HTML file (case (1)), quote ('"') must also be specified as a URL escape sequence (%22).

Any other 7-bit ASCII character may also be specified as a URL escape sequence, although this is not required (except possibly for `%`, as noted below). Certain additional restrictions apply to the `%` character. These include:

■ If the HTML delimiter is `%` (the default), then an occurrence of `%` within the value of a runform parameter must be escaped (specified as `%25`). (This actually follows from the requirement stated above, that the HTML delimiter character be escaped). Furthermore, variable names must never begin with two hexadecimal digits that represent a 7-bit ASCII value (that is, two hexadecimal digits, the first of which is in the range 0-7).

■ If the HTML delimiter is not `%`, then an occurrence of `%` must be escaped if it is immediately followed by an octal digit and then a hexadecimal digit. It is recommended that other occurrences of '%' also be escaped; but this is not a requirement.

(You might choose to ignore this recommendation if you have existing template HTML files or configuration files created in prior releases, which use an HTML delimiter other than '%', and which contain '%' in runform parameter values).

### 3.3.4.2  Behavior in Previous Releases

Release 9.0.4 and later behave the same as the current release except that a quote must be escaped (%22) within the value of a runform parameter in a configuration file, and in the template HTML file.

Releases before 9.0.4 did not allow URL escape sequences in runform parameter values specified in the template HTML file or the configuration file (cases (1) and (2) above). In all three cases, it was difficult or impossible to specify certain special characters, notably space, quote, and apostrophe. Also, certain transformations were applied to the parameter value before passing it to runform. Most notably, if a value began and ended with an apostrophe, these were typically stripped off. However, these transformations were not well-defined, and they differed between the Web and client/server environments.

### 3.3.4.3  Obtaining the Behavior of Prior Releases in the Current Release

If your applications are dependent on the behavior of prior releases, you can obtain that behavior in the current release, by simply setting the value of the `escapeparams` variable to `False` in the configuration file (this can be accomplished using Fusion Middleware Control).

If you want to obtain the old behavior only for selected applications, you can specify different values for the `escapeparams` variable in different configuration sections. Applications that require the old behavior can specify a configuration section in which the `escapeparams` variable is set to `False`; applications that require (or tolerate) the behavior in the current release can specify a configuration section in which the `escapeparams` variable is set to `True`.

### 3.3.4.4  Considerations for Template HTML Files

If you are creating your own template HTML files, then bear in mind the following:

It is recommended that a reference to the `escapeparams` variable (the string `%escapeparams%`, if '%' is the HTML delimiter character) appear at the beginning of the value of the `serverArgs` applet parameter, followed by a space. See the shipped base.htm file for an example.

References to the `escapeparams` variable must appear nowhere else in the template HTML file. If you choose to enclose the value of the `serverArgs` applet parameter in apostrophes instead of quotes, then within the value of a runform parameter in your template HTML file, apostrophes must be escaped (%27). Quotes do not require escape sequences.

It is permissible to omit the reference to the `escapeparams` variable from the beginning of the value of the `serverArgs` applet parameter. This results in the behavior of prior releases, regardless of the value specified in the configuration file for the `escapeparams` variable.

### 3.3.4.5  Considerations for Static HTML Pages

If you are invoking the runform engine using static HTML, and you want to obtain the behavior in the current release, then you must take certain steps.

The basic rule is that your static HTML must look like the HTML generated by the Forms servlet. Specifically, the value of the `serverArgs` applet parameter must begin with the string `escapeparams=true` (case-insensitive).

Also, in the value portion of each name=value pair, in the value of the `serverArgs` applet parameter, certain characters must be specified by a URL escape sequence, as listed in Table 3–4:

*Table 3–4    URL Escape Sequences for Static HTML pages*

| Characters that must be escaped | URL Escape Sequence |
| --- | --- |
| newline ' \n ' | %0a |
| space ' ' | %20 |
| quote ' " ' | %22 |
| percent ' % ' | %25 |
| apostrophe ' ' ' | %27 |
| left parenthesis ' ( ' | %28 |
| right parenthesis ' ) ' | %29 |

It is also permissible to escape other 7-bit ASCII characters in the value portion of a `name=value` pair.

Here's an example of what the `serverArgs` applet parameter might look like in static HTML. This is for a form named "my form" (quotes not included), which is being passed the value "foo'bar" (quotes again not included) to the user-defined parameter named `myparam`.

```
<PARAM NAME="serverArgs" VALUE="escapeparams=true module=my%20form
userid=scott/tiger@mydb myparam=foo%27bar">
```

### 3.3.5  Accessing the Listener Servlet Administration Page

You can display a test page for the Listener Servlet by accessing the following URL:

`http://<hostname>:<port>/forms/frmservlet/admin`

The information displayed depends on the value of the initialization parameter `TestMode`. This parameter is set in the `$DOMAIN_HOME/servers/WLS_FORMS/tmp/_WL_user/formsapp_11.1.2/<random_string>/war/WEB-INF/web.xml` file. An example is shown below:

```
<init-param>
<!-- Display sensitive options on the /admin page ? -->
    <param-name>TestMode</param-name>
    <param-value>true</param-value>
</init-param>
```

## 3.4  Client Browser Support

Users can view Oracle Forms applications on the Web using Oracle Java Plug-in. In future patch releases other virtual machines may be supported.

For more information about client browser support, including the latest supported platforms, go to the Forms Developer menu and choose **Help | Forms on OTN...** to locate the Client Platform Statement of Direction.

You can also find information on certification on OTN at http://www.oracle.com/technetwork/developer-tools/forms/clientsod-forms10gr2-088253.html

### 3.4.1 How Configuration Parameters and BaseHTML Files are Tied to Client Browsers

When a user starts a Web-enabled application (by clicking a link to the application's URL), the Forms servlet:

1. Detects which browser is being used.

2. Selects the appropriate baseHTML file using Table 3–5:

*Table 3–5    baseHTML file descriptions*

| Detected Browser | Base HTML file used |
|---|---|
| Internet Explorer | `basejpi.htm` |
| Mozilla FireFox 3.0 | `basejpi.htm` |
| All other browsers and Macintosh clients | `base.htm` |

3. Replaces variables (%*variablename*%) in the baseHTML file with the appropriate parameter values specified in the Forms `servlet.initArgs` file, `formsweb.cfg` file, and from query parameters in the URL request (if any).

4. Sends the HTML file to the user's browser.

### 3.4.2 Forms Single Sign-On on Mozilla 3.x

Ensure that you have enabled cookies from Oracle site when using Forms and Single Sign-On on Mozilla 3.x. To enable the cookies, perform the following steps:

1. Open Mozilla Firefox, select Tools.

2. Select Options and then Privacy.

3. Select the Accept Cookies from site box.

   These steps are not required for other browsers.

   For more information on default browser settings in Mozilla, refer to http://www.mozilla.com.

### 3.4.3 guiMode Configuration Parameter

The `guiMode` parameter controls the runtime GUI of a Form application. This parameter can be specified in the URL or a value can be provided in the `formsweb.cfg` file (Forms configuration file). The `guiMode` parameter affects the visibility of the following GUI components:

■ The visibility of default Windows menubar provided by the client. When no menubar is specified for a Form in the Forms builder, the client provides a default menubar at runtime. The `guiMode` value affects only this default menubar provided by the client.

> **Note:** The `guiMode` value takes effect for  menubars only when the Forms menu module parameter is set to null. If the Form has any other server specified menubar (including the Forms default menu) or toolbar associated with it, then this parameter is not applicable. In case of window-bars, this parameter is applicable even if there is a menu specified for that form in the Forms Builder.

- The visibility of the title bars of all the windows in a Form. This parameter does not affect title bars in windows like - alert windows, pop-up windows.

Table 3–6 shows the effect of guiMode values on the default Windows menubar and Windows title bar. The default guiMode value is 0. Any value other than the four valid values mentioned in the table, will be ignored. In such a case, guiMode will return to its default value.

*Table 3–6    Effect of guiMode Values*

| guiMode | Default Menubar Visible | Windows Title bar Visible |
|---------|-------------------------|---------------------------|
| 0 | Yes | Yes |
| 1 | No | Yes |
| 2 | Yes | No |
| 3 | No | No |

> **Note:**   At guiMode = 2 or 3, when windows title bar is not visible, windows cannot be maximized or minimized. Though it is possible to maximize or minimize the window using built-ins, users should not minimize the window. This is  because once the window is minimized, it cannot be restored.

# 4

# Configuring and Managing Forms Services

This chapter contains the following sections:

- Section 4.1, "Fusion Middleware Control and Oracle Forms"
- Section 4.2, "Configuring Forms Services"
- Section 4.3, "Managing Environment Variables"
- Section 4.4, "Managing User Sessions"
- Section 4.5, "Managing URL Security for Applications"
- Section 4.6, "Creating Your Own Template HTML Files"
- Section 4.7, "Deploying Fonts, Icons, and Images Used by Forms Services"
- Section 4.8, "Enabling Language Detection"
- Section 4.9, "Enabling Key Mappings"

## 4.1 Fusion Middleware Control and Oracle Forms

The Fusion Middleware Control is a Web-based tool that you launch from your default browser. The default URL is:

```
http://<example.com>:7001/em
```

Use the Web-based Oracle Enterprise Manager Fusion Middleware Control to:

- Monitor metrics for a Forms Services instance. See Section 14.1.1.1, "Monitoring Forms Services Instances" for more information.
- Manage user sessions. See Section 4.4, "Managing User Sessions" for more information.
- Configure parameters for a Forms Services instance. See Section 4.2.2, "Configuring Parameters with Fusion Middleware Control" for more information.
- Configure Forms Trace and monitor trace metrics. See Section 12.2, "Enabling and Configuring Forms Trace" and Section 12.6, "Taking Advantage of Oracle Diagnostics and Logging Tools" for more information.
- Configure multiple environment files. See Section 4.3, "Managing Environment Variables" for more information.
- Configure and use JVM pooling. See Section 10.8, "Managing JVM Pooling from Fusion Middleware Control" for more information.

### 4.1.1 Accessing Forms Services with Fusion Middleware Control

To perform most management tasks for a Forms instance using Fusion Middleware Control, you start by navigating to the Forms home page in Fusion Middleware Control.

**To navigate to the Forms Home page in Fusion Middleware Control:**

1. Navigate to the home page for the Fusion Middleware Control that contains the Forms instance you want to manage.

   For introductory information about using the Enterprise Manager Fusion Middleware Control, see "Overview of Oracle Fusion Middleware Administration Tools" in the *Oracle Fusion Middleware Administrator's Guide*.

2. In the Farm pane, click the Fusion Middleware folder, then click the link for the Forms instance. This displays the Forms Home page (Figure 4–1) in the Fusion Middleware Control.

*Figure 4–1   Forms Home page*
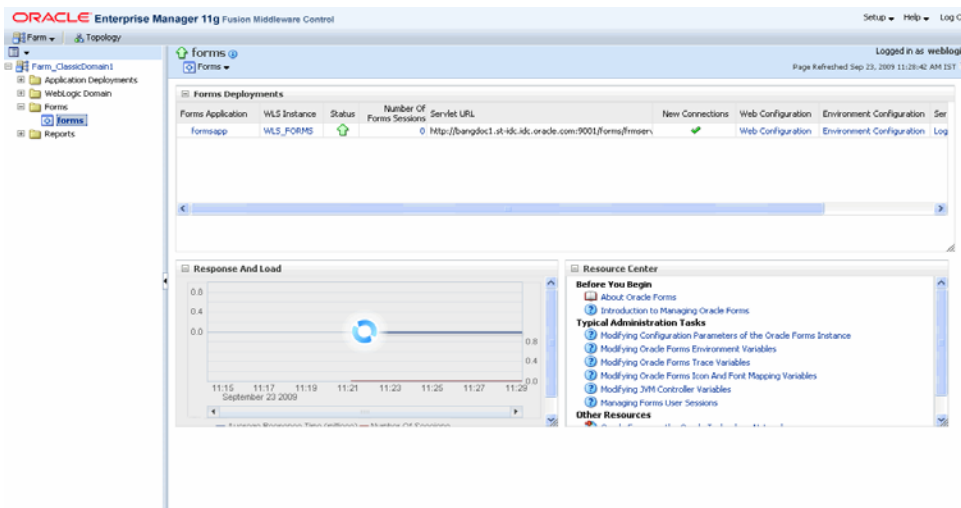


3. The Forms Home page provides information on the Forms applications that are deployed on the Oracle instance. Table 4–1 describes the information displayed on the Forms Home page.

*Table 4–1     Forms Deployment Fields*

| Field | Description |
| --- | --- |
| Forms Application | Lists the names of the Forms applications that are deployed on the Oracle WebLogic Server instance. Click the name to view the Forms application home page. |
| WLS Instance | Name of Oracle WebLogic Server instance where the application is deployed. |
| Status | Indicates the status of the forms application. A green up arrow indicates the application is running. A red down arrow indicates the application is not started. |
| Number of Forms Sessions | Displays the number of active forms sessions. |
| Servlet URL | Displays the URL for the Forms servlet. |
| New Connections | Indicates whether new connections are enabled or not. |

*Table 4–1 (Cont.) Forms Deployment Fields*

| Field | Description |
| --- | --- |
| Web Configuration | Link to the Web Configuration page. |
| Environment Configuration | Link to the Environment Configuration page. |
| Servlet Logs | Link to the Servlet Logs. |
| Prestart Scheduling | Link to the Prestart scheduling page. |

**To access the Forms Menu in Fusion Middleware Control:**

1. Navigate to the Forms home page in Fusion Middleware Control.

2. Click **Forms** on the top left. This displays the Forms Menu. Table 4–2 lists the Menu Selections that are available in the Forms Menu.

*Table 4–2 Forms Menu Options*

| Select | To Display |
| --- | --- |
| Home | Forms Home page. This page displays a list of the Forms deployments and their details. This page also displays the Response and Load statistics and a set of useful links in the Resource Center. |
| Monitoring - Performance Summary | Performance Summary page. This page displays a set of default performance charts that show the values of specific performance metrics.<br><br>For more information, see the *Oracle Fusion Middleware Performance Guide*. |
| Monitoring - Servlet Log | Log Messages page. Oracle Fusion Middleware components generate log files containing messages that record all types of events. |
| JVM Controllers | JVM Controllers page. This page is used to manage the JVM controller for the Forms instance. |
| Schedule Prestart | Prestart scheduling page. This page is used to manage Forms prestart scheduling. |
| User Sessions | User Sessions page. This page is used to monitor and trace User Sessions within a Forms instance. |
| Web Configuration | Web Configuration page. This page is used to configure deployment of Forms applications and manage configuration sections and parameters in `formsweb.cfg`. |
| Trace Configuration | Trace Configuration page. This page is used to manage the settings used for tracing of user sessions. |
| Fonts and Icons Mapping | Fonts and Icons Mapping page. This page is used to change, add, or delete parameters in the Registry.dat file. |
| JVM Configuration | JVM Configuration page. This page is used to modify the JVM controllers that can be subsequently spawned for the Forms instance. |
| Environment Configuration | Environment Configuration page. This page is used to manage environment variables that define environment settings for Forms run time. |
| Associate/Disassociate OID | Associate/Disassociate OID page. This page is used to associate and disassociate a forms deployment with an Oracle Internet Directory host to enable Single Sign-On functionality. |

**Table 4–2   (Cont.)  Forms Menu Options**

| Select | To Display |
| --- | --- |
| General Information | Displays information about the Target Name, Version, Oracle Home, Oracle Instance, and Host. |

---

**Note:**   For the pages that include a **Help** icon, click the **Help** icon to access the page-level help. The page-level help describes each element in the page.

---

## 4.2  Configuring Forms Services

Use the **Web Configuration** page in Fusion Middleware Control to configure deployment of Forms applications by modifying `formsweb.cfg`.

**To access Web Configuration page:**

1.  Start Fusion Middleware Control.

2.  From the Fusion Middleware Control main page, click the link to the Oracle Forms Services instance that you want to configure.

3.  From the Forms menu list, select **Web Configuration**.

    The **Web Configuration** page (Figure 4–2) is displayed.

**Figure 4–2   Web Configuration Page**
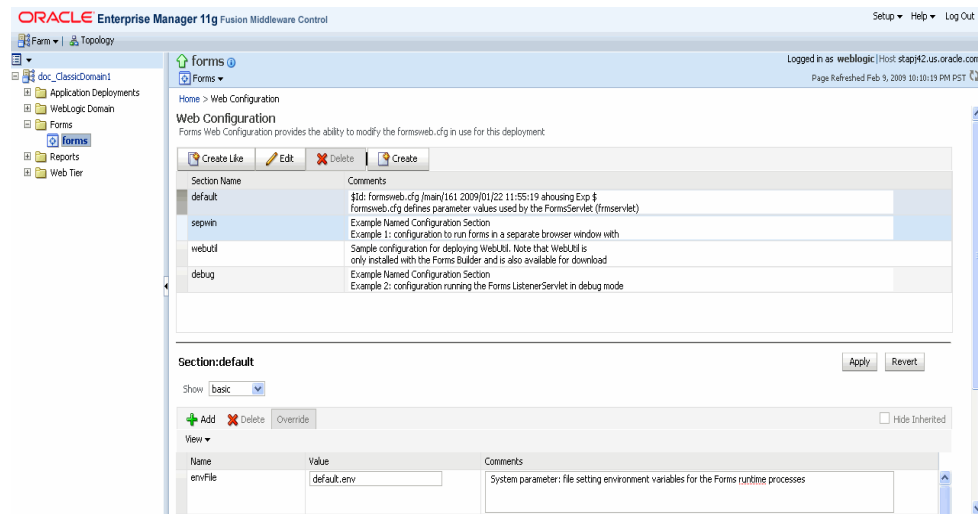


4.  See Table 4–3 and Table 4–4 for the tasks that you can do.

> **Note:** As with most Web applications, it is easy to lose unsaved changes by switching pages. Be sure to save any changes you make through Fusion Middleware Control to Forms configuration or environment files before proceeding to other pages.
>
> The length of time it takes for changes to be saved is affected by the number of lines you have changed. For example, an additional fifty lines of comments takes longer to save than just the deletion of a single entry.

## 4.2.1 Common Tasks in the Web Configuration Page

Table 4–3 describes the common tasks that you can do to edit configuration with the sections of a configuration file and their parameters.

*Table 4–3    Common Tasks for Working with Configuration Sections*

| Task | Description | Comment |
| --- | --- | --- |
| Create Like | Creates a copy of a configuration section. | Use to create a configuration section based on the parameters of an existing configuration section. |
| Edit | Opens the Edit Description dialog. | Allows editing of the text description of a configuration section. |
| Delete | Opens the Confirmation dialog when deleting a configuration section. | Irrevocably deletes a configuration section and its contents when you click Delete in the Confirmation dialog. |
| Create | Opens the Create Section dialog. | Creates a configuration section. You must supply a required name and an optional description for it. |

Table 4–4 describes the tasks that you can do to modify the parameters within a named configuration section:

*Table 4–4    Common Tasks for Working with Parameters*

| Task | Description | Comment |
|------|-------------|---------|
| Show | Drop down list for selecting named groups of parameters in a configuration section. | Use for viewing and editing groups of parameters. The groups of parameters include:<br><br>■ basic<br><br>■ sso<br><br>■ trace<br><br>■ plugin<br><br>■ HTML<br><br>■ applet<br><br>■ advanced<br><br>■ all<br><br>For more information, see Section 4.2.5, "Forms Configuration Parameters". |
| Revert | Enables you to revert all changes made to parameters in a configuration section since the last apply. | Does not allow you to revert individual changes in a configuration section. |
| Apply | Applies and activates all changes made to parameters in a configuration section. | Once applied, you cannot revert changes to individual parameters. |
| Hide Inherited | Enables you to hide or display parameters that are inherited from a parent configuration section. | Use this to view parameters that have been explicitly added to a configuration section or to view all parameters (including those that are inherited from the default section). |
| Add | Displays the Add Parameter dialog. | Add a parameter to a configuration section based on a mandatory name and an optional value and description. |
| Delete | Deletes a parameter. | There is no Confirmation dialog. Once applied, you cannot revert changes to individual parameters. |
| Override | Allows overriding and editing of a parameter which is inherited from the default section. | Click Apply to save and activate your changes. |

## 4.2.2  Configuring Parameters with Fusion Middleware Control

For a description and the location of the Forms servlet configuration file (`formsweb.cfg`), see Section 3.2.1.2, "formsweb.cfg".

### 4.2.2.1  Parameters that Specify Files

Three configuration parameters specify files. Of these, two baseHTML parameters must point to appropriate `.htm` files. Typically, the following values and their parameters should appear in the default configuration section, as shown in Table 4–5.

*Table 4–5    Default Configuration Parameters that Specify Files*

| Parameter | Value |
|---|---|
| baseHTML | `base.htm` |
| baseHTMLjpi | `basejpi.htm` |
| envFile | `default.env` |

All of these parameters specify file names. If no paths are given (as in this example), the files are assumed to be in the same directory as the Forms servlet configuration file (`formsweb.cfg`), that is `$DOMAIN_HOME/config/fmwconfig/servers/WLS_FORMS/applications/formsapp_11.1.2/config`

## 4.2.3  Managing Configuration Sections

This section describes creating, editing, duplicating, and deleting named configuration sections.

### 4.2.3.1  Creating a Configuration Section

You can create a configuration section in `formsweb.cfg` from the **Web Configuration** page of Fusion Middleware Control. These configurations can be requested in the end-user's query string of the URL that is used to run a form.

**To create a configuration section:**

1.  Start the Enterprise Manager Fusion Middleware Control.

2.  From the Fusion Middleware Control main page, click the link to the Forms Services instance that you want to configure.

3.  From the Forms menu list, select the **Web Configuration**.

4.  Click **Create** at the top of the **Web Configuration** region.

    The **Create Section** dialog appears.

5.  Enter a name and description for the configuration section and click **Create**.

    > **Note:**  The name must not contain any special characters such as #, *.

    The configuration section is added.

For example, to create a configuration to run Forms in a separate browser window with the `Oracle` look and feel, create a section called sepwin and add the following parameters from Table 4–6:

*Table 4–6    Sample Parameters to Add to a Configuration Section*

| Parameter | Value |
|---|---|
| form | `<module>` |
| separateFrame | `True` |
| lookandfeel | `Oracle` |

Your users would type the following URL to launch a form that uses the "sepwin" (or the name you applied) configuration:

```
http://server:port/forms/frmservlet?config=sepwin
```

### 4.2.3.2 Editing a Named Configuration Description

You can edit the description (comments) for a named configuration from the **Web Configuration** page.

> **Note:** You can make a backup of the configuration section you are about to edit by duplicating it first. For more information, see Section 4.2.3.3, "Duplicating a Named Configuration"

**To edit a named configuration description:**

1. In the **Web Configuration** region, select the row containing the configuration section you want to edit.

2. Click **Edit**.

3. The **Edit Description** dialog appears.

4. Enter the text for the comment.

5. Click **Save**.

   The **Edit Description** dialog box is dismissed, and your changes are saved.

### 4.2.3.3 Duplicating a Named Configuration

You can make a copy of a named configuration for backup purposes, or create configuration sections from existing configurations or other duplicates.

**To duplicate a named configuration:**

1. In the **Web Configuration** region, select **Create Like**.

2. In the Create Like dialog, from the **Section to Duplicate** menu list, select the name of an existing configuration section you want to duplicate.

3. In the **New Section Name** field, enter a name for the configuration section. The name for the configuration section must be unique.

4. Click **Create**.

   A section with the same parameters, parameter values and comments of the section you are duplicating is created.

### 4.2.3.4 Deleting a Named Configuration

When you delete a named configuration, you delete *all* the information within it. If you only want to delete specific parameters, see Section 4.2.4, "Managing Parameters".

**To delete a named configuration:**

1. From the **Web Configuration** region, select the row of the configuration section you want to delete.

2. Click **Delete**.

   The **Confirmation** dialog appears.

3. Click **Delete**.

   The configuration section is deleted.

Oracle Enterprise Manager returns to the **Web Configuration** page and displays the remaining configurations.

> **Note:** You cannot delete the Default configuration section.

## 4.2.4 Managing Parameters

Use Fusion Middleware Control to manage parameters within a named configuration. You can add, edit, or delete parameters from the Section pane of Fusion Middleware Control.

**To edit a new or overridden parameter in a configuration section:**

1. From the **Web Configuration** region, select the row of the configuration section that contains the parameter(s) you want to edit.

2. In the Section region, select the parameter group from the **Show** menu list. The parameters of the group are  displayed.

3. Select the row of the parameter you want to edit. Enter the Value and Comments.

> **Note:** You can edit new or overridden parameters. Inherited parameters must first be overridden so they can be edited. In Figure 4–3, `test1` is an example of a new parameter and `lookandfeel` is an example of an overridden parameter.

4. Click **Apply** to save the changes or **Revert** to discard them.

**To add a parameter to a configuration:**

1. In Fusion Middleware Control, from the **Web Configuration** region, select the configuration section row to which you want to add a parameter.

2. Click Add to add a parameter.

   The Add dialog box is displayed.

3. Enter the Name, Value and Comments for the parameter.

4. Click **Create** to add the parameter.

5. Click **Apply** to save the changes or **Revert** to discard them.

**To delete a parameter in a configuration:**

1. In Fusion Middleware Control, from the **Web Configuration** region, select the configuration section row that contains the parameter you want to delete.

2. In the Sections region, from the Show menu list, select the parameter group that contains the parameter you want to delete.

3. Select the row that contains the parameter you want to delete.

4. Click **Delete**.

5. Click **Apply** to save the changes or **Revert** to discard them.

> **Note:** You can delete/edit multiple parameters at a time.

> **Note:** You can only delete user-defined parameters. Inherited parameters (such as `enableJavascriptEvent` in Figure 4–3) cannot be deleted.

> **Note:** When you delete an overridden parameter, the parameter is not deleted but instead regains its inherited status.

**Figure 4–3   Parameter States**



## 4.2.5  Forms Configuration Parameters

The section provide information about Forms configuration parameters. These parameters can be specified in the Forms configuration file (`formsweb.cfg`), as described in preceding sections. Many of these parameters can also be specified in the URL. Parameters that cannot be specified in the URL are listed in Section 4.2.5.8. A value in the URL overrides a value from `formsweb.cfg`. The following notes apply to all the parameter tables from Section 4.2.5.1 to Section 4.2.5.7:

- Required/Optional:  A parameter is required if the Forms Services requires a non-null value (from `formsweb.cfg` or, where allowed, from the URL) to function correctly.

- Default values:  For required parameters, the parameter description lists the default value from the default section of the `formsweb.cfg` that is shipped with the Forms product (or at least indicates that it specifies an appropriate value).

  For optional parameters, the parameter description may show a non-null default value from the default section of the `formsweb.cfg` that is shipped with the Forms product.  In addition, the parameter description may show the default value that is assumed if no value is specified. (This is the non-null value that produces the same behavior as a null value). When the description for an optional parameter simply shows an unqualified default value, the implication is that this value is both the default value from the default section of the `formsweb.cfg` that is shipped with the Forms product, and also the default value that is assumed if no value is specified.

  When the description for an optional parameter does not explicitly specify a default value, the implication is that the default value is null.

- Runform parameters:  The descriptions for some parameters indicate that they are runform parameters. They are passed to the frmweb process using the serverArgs applet parameter. For such a parameter, the syntax rules documented in Section 3.3.4 must be adhered to when specifying a value that contains special characters.

- Sub-arguments for otherparams: The descriptions for some parameters indicate that they are sub-arguments for otherparams. That means that in order for the parameter to take effect (when specified in `formsweb.cfg` or the URL), it must appear in the form "`name=%name%`" within the value of the otherparams parameter. So, for example, if you are adding the parameter "array" (with a value of "no") to a configuration section, you must also add "`array=%array%`" to the value of the otherparams parameter.

  Note that these parameters are all runform parameters (since the otherparams parameter is itself a runform parameter), and so the syntax rules documented in Section 3.3.4 must be adhered to when specifying a value that contains special characters.

This section includes:

- Section 4.2.5.1, "Basic Configuration Parameters"

- Section 4.2.5.2, "Single Sign-On Configuration Parameters"

- Section 4.2.5.3, "Trace Configuration Parameters"

- Section 4.2.5.4, "Plug-in Configuration Parameters"

- Section 4.2.5.5, "HTML Page Configuration Parameters"

- Section 4.2.5.6, "Applet Configuration Parameters"

- Section 4.2.5.7, "Advanced Configuration Parameters"

- Section 4.2.5.8, "List of Parameters that Cannot be Specified in the URL"

### 4.2.5.1  Basic Configuration Parameters

These basic parameters control the behavior of the Forms servlet. These parameters are described in Table 4–7:

*Table 4–7    Basic Configuration Parameters*

| Parameter | Required/ Optional | Parameter Value and Description |
|---|---|---|
| envFile | Required | Specifies the name of the environment configuration file. Default value from `formsweb.cfg` is `default.env`. |
| form | Required | Specifies the name of the top level Forms module (fmx file) to run. Default value from `formsweb.cfg` is `test.fmx`. This parameter is a runform parameter. |

*Table 4–7   (Cont.)  Basic Configuration Parameters*

| Parameter | Required/ Optional | Parameter Value and Description |
|---|---|---|
| height | Required | Specifies the height of the form applet, in pixels.<br><br>Default value from `formsweb.cfg` is 600.<br><br>You can also specify the value of the height in percentage. This value is relative to the sixe of the content area of the browser. The value should not exceed 100% or be less than 1%. To use a percentage value, the numeric value must be followed by a percentage (%) sign as shown in the following example:<br><br>HEIGHT= 75%<br><br>This example means that the applet height is 75% of the size of the browser's content area. |
| userid | Optional | Login string. For example: `scott/tiger@ORADB`. This parameter is a runform parameter. |
| width | Required | Specifies the width of the form applet, in pixels.<br><br>Default value from `formsweb.cfg` is 750.<br><br>You can also specify the value of the width in percentage. This value is relative to the sixe of the content area of the browser. The value should not exceed 100% or be less than 1%. To use a percentage value, the numeric value must be followed by a percentage (%) sign as shown in the following example:<br><br>WIDTH= 75%<br><br>This example means that the applet width is 75% of the size of the browser's content area. |

### 4.2.5.2  Single Sign-On Configuration Parameters

*Table 4–8    Single Sign-On Configuration Parameters*

| Parameter | Required / Optional | Parameter Value and Description |
|---|---|---|
| ssoCancelUrl | Optional | Specifies the Cancel URL for the dynamic resource creation page. |
| ssoDynamicResourceCreate | Optional | Specifies whether dynamic resource creation is enabled if the resource is not yet created in the OID.<br><br>Default value is `true`. |
| ssoErrorUrl | Optional | Specifies the URL to redirect to if `ssoDynamicResourceCreate` is set to `false`. |

*Table 4–8 (Cont.) Single Sign-On Configuration Parameters*

| Parameter | Required / Optional | Parameter Value and Description |
|---|---|---|
| ssoMode | Optional | Specifies whether the URL is protected in which case, mod_osso or webgate is given control for authentication or continue in the FormsServlet if not. Set it to true or mod_osso or webgate in an application-specific section to enable Single Sign-On for that application. Default value is false. |
| ssoProxyConnect | Optional | Specifies whether session should operate in proxy user support or not. Set ssoProxyConnect to yes to enable for particular application. Default value is no. This parameter is a sub-argument for otherparams. |

### 4.2.5.3 Trace Configuration Parameters

*Table 4–9 Trace Configuration Parameters*

| Parameter | Required/ Optional | Parameter Value and Description |
|---|---|---|
| debug | Optional | Allows running in debug mode. Default value is No. This parameter is a runform parameter. |
| EndUserMonitoringEnabled | Optional | Indicates whether End User Monitoring integration is enabled. Default value is false. |
| EndUserMonitoringURL | Optional | Indicates where to record End User Monitoring data. |
| host | Optional | Specifies the host for the debugging session. This parameter should be used for debugging purposes only. It identifies the host on which the forms engine process is started. This parameter is a runform parameter. |
| log | Optional | Supports tracing and logging. The value of this parameter, if set, is the file name of the trace log file. This parameter is a sub-argument for otherparams. |
| port | Optional | Port to use for debugging. This parameter should be used for debugging purposes only. The value of this parameter identifies the port on which the forms engine process is listening. If not specified, the default value is 9000. This parameter is ignored if serverURL has been specified. This parameter is a runform parameter. |

*Table 4–9  (Cont.)  Trace Configuration Parameters*

| Parameter | Required/ Optional | Parameter Value and Description |
|---|---|---|
| record | Optional | Supports tracing and logging.<br><br>This parameter is a sub-argument for otherparams. |
| tracegroup | Optional | Supports tracing and logging.<br><br>This parameter is a sub-argument for otherparams. |

#### 4.2.5.4  Plug-in Configuration Parameters

These parameters are for use with Oracle Java Plug-in.

*Table 4–10    Oracle Java Plug-in Configuration Parameters*

| Parameter | Required/ Optional | Parameter Value and Description |
|---|---|---|
| archive | Optional | Comma-delimited list of archive files that are used or downloaded to the client. For each file, include the file name if the file is in the codebase directory, or include the virtual path and file name.<br><br>Default value for formsweb.cfg is frmall.jar. |
| codebase | Required | Virtual directory you define to point to the physical directory ORACLE_HOME/forms/java, where, by default, the applet JAR files are downloaded from.<br><br>Default value from formsweb.cfg is /forms/java. |
| imageBase | Optional | Indicates where icon files are stored.  Legal values:<br><br>■ codeBase, which indicates that the icon search path is relative to the directory that contains the Java classes. Use this value if you store your icons in a JAR file (recommended).<br><br>■ documentBase, which indicates that the icon search path is relative to the Forms webapp's directory. The Forms webapp's directory is located at $DOMAIN_HOME/servers/WLS_FORMS/tmp/_WL_user/formsapp_11.1.2/<random string>/war.<br><br>Default value from formsweb.cfg is codeBase. If no value is specified, then the value of documentBase is used. |
| jpi_classid | Required | Oracle Java Plug-in class ID. formsweb.cfg specifies an appropriate value.The default value is clsid:CAFEEFAC-0016-0000-0012-ABCDEFFEDCBA. |

*Table 4–10   (Cont.)  Oracle Java Plug-in Configuration Parameters*

| Parameter | Required/ Optional | Parameter Value and Description |
|---|---|---|
| `jpi_codebase` | Required | Oracle Java Plug-in codebase setting. `formsweb.cfg` specifies an appropriate value. |
| `jpi_download_page` | Required | Oracle Java Plug-in download page. `formsweb.cfg` specifies an appropriate value. |
| `jpi_mimetype` | Required | Parameter related to version of Java Plug-in. `formsweb.cfg` specifies an appropriate value. |

### 4.2.5.5  HTML Page Configuration Parameters

*Table 4–11    HTML Page Configuration Parameters*

| Parameter | Required/ Optional | Parameter Value and Description |
|---|---|---|
| `baseHTML` | Required | Used as the base HTML file, if the client browser is not on MS Windows and/or does not support the <OBJECT> tag.<br><br>Default value from `formsweb.cfg` is `base.htm`. |
| `baseHTMLjpi` | Required | Physical path to HTML file that contains Java Plug-in tags. Used as the base HTML file if the client browser is on MS Windows and supports the <OBJECT> tag.<br><br>Default value from `formsweb.cfg` is `basejpi.htm`. |
| `HTMLafterForm` | Optional | HTML content to add to the page below the area where the Forms application is displayed. |
| `HTMLbeforeForm` | Optional | HTML content to add to the page above the area where the Forms application is displayed. |
| `HTMLbodyAttrs` | Optional | Attributes for the <BODY> tag of the HTML page. |
| `pageTitle` | Optional | HTML page title, attributes for the BODY tag, and HTML to add before and after the form.<br><br>Default value from `formsweb.cfg` is `Oracle Fusion Middleware Forms Services`. |

### 4.2.5.6  Applet Configuration Parameters

These parameters are specified in the baseHTML file as values for object or applet parameters. They describe the visual behavior and appearance of the applet.

*Table 4–12    Applet or Object Configuration Parameters*

| Parameter | Required/ Optional | Parameter Value and Description |
|---|---|---|
| background | Optional | Specifies the image file that should appear in the background. Set to NO for no background. Leave empty to use the default background. |
| colorScheme | Optional | Determines the application's color scheme. Legal values: Teal, Titanium, Red, Khaki, Blue, BLAF, SWAN, Olive, or Purple. Default value from formsweb.cfg is teal.<br><br>**Note**: colorScheme is ignored if LookAndFeel is set to Generic. |
| logo | Optional | Specifies the image file that should appear at the Forms menu bar. Set to NO for no logo. Leave empty to use the default Oracle logo. |
| lookAndFeel | Optional | Determines the applications look-and-feel. Legal values: Oracle or Generic (Windows look-and-feel).<br><br>Default value from formsweb.cfg is Oracle. |
| separateFrame | Optional | Determines whether the applet appears within a separate window. Legal values: true or false (default). |
| splashScreen | Optional | Specifies the image file that should appear before the applet appears. Set to NO for no splash. Leave empty to use the default splash image.<br><br>To set the parameter include the file name (for example, myfile.gif) or the virtual path and file name (for example, images/myfile.gif). |
| guiMode | Optional | This parameter determines the visibility of the default windows menu bar and the Windows title bar.<br><br>Possible values: 0,1,2,3.<br><br>Default value is 0. At the default value, the default Windows menu bar *and* the Windows title bar are visible.<br><br>**Note**: This parameter is applicable for a menubar only when no menu is specified for a form in the Forms Builder; if there is any menu associated with the form, then this parameter is not applicable. In case of window-bars, this parameter is applicable even if there is a menu specified for that form in the Forms Builder.<br><br>For more information about guiMode, see Section 3.4.3, "guiMode Configuration Parameter". |

## 4.2.5.7  Advanced Configuration Parameters

*Table 4–13    Advanced Configuration Parameters*

| Parameter | Required/ Optional | Parameter Value and Description |
|---|---|---|
| allowAlertClipboard | Optional | Forms applet parameter.<br><br>Default value is true. |
| allowNewConnections | Optional | Determines whether new Forms sessions are allowed. This is also used by the Forms Home page in Fusion Middleware Control to show the current Forms status.<br><br>Default value is true. |
| applet_name | Optional | Configuration for JavaScript integration. This is  name of the Forms applet that can be used to refer to it from a JavaScript code. |
| array | Optional | Set this parameter to no to suppress array processing.  This causes Forms to send only a single row at a time to the database for an INSERT, UPDATE, or DELETE, and it causes the database to return only a single row of query results at a time.  This usually results in the first retrieved record displaying faster, but the total time to display all rows in the query result is longer.<br><br>Default value if not specified is yes. This parameter is a sub-argument for otherparams. |
| buffer_records | Optional | Set this parameter to yes to set the number of records buffered in memory to the number of rows displayed, plus 3 (for each block).  This saves Forms Runtime memory, but may slow down processing because of increased disk I/O. Sub argument for otherparams.<br><br>Default value if not specified is no. This parameter is a sub-argument for otherparams. |
| clientDPI | Optional | Specifies the dots per inch (DPI) and overrides the DPI setting returned by the JVM, allowing you to manage varying DPI settings per platform. Oracle recommends that you use an integer between 50 and 200. |
| connectionDisallowedURL | Optional | This is the URL shown in the HTML page that is not allowed to start a session. |
| cursorBlinkRate | Optional | To modify the cursor blink rate, or disable blinking, set the client parameter cursorBlinkRate as follows: <PARAM NAME="cursorBlinkRate" VALUE="1000">.<br><br>The default is 600 milliseconds: the cursor completes one full blink every 1.2 seconds (1200 ms). A value of zero disables the blinking and the cursor remains visible all the time. |

**Table 4–13   (Cont.)  Advanced Configuration Parameters**

| Parameter | Required/ Optional | Parameter Value and Description |
|---|---|---|
| debug_messages | Optional | Set this parameter to `yes` to cause Forms to display ongoing messages about trigger execution while the form runs. |
| | | Default value if not specified is `no`. This parameter is a sub-argument for otherparams. |
| defaultcharset | Optional | Specifies the character set to be used in servlet requests and responses. Defaults to ISO-8859-1 (also known as Latin-1). Ignored if the servlet request specifies a character set (for example, in the content-type header of a POST). The values of this parameter may be specified either as an  IANA character set name (for example, SHIFT_JIS) or as an Oracle character set name (for example, JA16SJIS). It should match the character set specified in the NLS_LANG environment variable, and it should also be a character set that the browser can display. Also, if the browser allows multibyte characters to be entered directly into a URL, for example, using the IME, as opposed to URL escape sequences, and to allow end users to do this, then the value of this parameter should match the character set that the browser uses to convert the entered characters into byte sequences. |
| | | **Note**: If your configuration file contains configuration sections with names that contain characters other than 7-bit ASCII characters, then the following rules apply. If a config parameter is specified in a URL or in the body of a POST request with no specified character set, and the value contains non-7-bit ASCII characters, then the value is interpreted using a character set named in the defaultcharset parameter. However, only the language-dependent default section and the language-independent default section of the configuration file is searched for the defaultcharset parameter. No other configuration section is searched because the name is not yet known. |
| digitSubstitution | Optional | Determines the BIDI digitSubstitution. Permissible values are `none`, `national`, and `context`. Default value is context. |
| disableMDIScrollbars | Optional | Set this parameter to `true` to disable horizontal and vertical scrollbars in the Forms main applet window. |
| | | You can also add this parameter in `basejpi.html`, in the OBJECT tag: |
| | | `<PARAM NAME="disableMDIscrollbars"` `VALUE="%disableMDIScrollbars%">`. |
| | | In the tag `<EMBED SRC>` add |
| | | `disableMDIScrollbars="%disableMDIScrollbars%"`. |
| | | Default value if not specified is `false`. |

*Table 4–13   (Cont.)  Advanced Configuration Parameters*

| Parameter | Required/ Optional | Parameter Value and Description |
|---|---|---|
| `disableValidateClipboard` | Optional | Forms applet parameter.<br><br>Default value is `false`. |
| `enableJavascriptEvent` | Optional | Configuration for JavaScript integration.<br><br>Default value is `true`. |
| `escapeparams` | Optional | Set this parameter to `false` for runform to treat special characters in runform parameters as it did in releases before 9.0.4. This parameter is a Forms run-time argument and specifies whether to escape certain special characters in values extracted from the URL for other run-time arguments.<br><br>Default value is `false`. |
| `formsMessageListener` | Optional | Forms applet parameter that specifies the class that the Forms client uses to enable recording of Forms messages for Tool Vendor Interface (TVI) / Intercept Server. |
| `heartBeat` | Optional | Use this parameter to set the frequency at which a client sends a packet to the server to indicate that it is still running. Define this integer value in minutes or in fractions of minutes, for example, 0.5 for 30 seconds. Default value, if not specified, is 2 minutes.<br><br>If the `heartBeat` is less than FORMS_TIMEOUT, the user's session is kept active, even if they are not actively using the form.<br><br>**Note**: It is not recommended to set the value of `heartbeat` greater than the value of `FORMS_TIMEOUT` because this will result in the termination of the user's existing session. If heartBeat is higher than the parameter session-timeout, then the value of session-timeout takes precedence over heartBeat. To increase the value of heartBeat, the value of session-timeout must be greater than heartBeat. For more information on this parameter, see "Session-timeout" in *Oracle Fusion Middleware Developing Web Applications, Servlets, and JSPs for Oracle WebLogic Server*. |
| `highContrast` | Optional | When `highContrast` is set to `true`, frame labels are black if foreground and background colors are not specified. Default value is `false`. |
| `HTMLdelimiter` | Optional | This parameter defines the delimiter for parameters in the base HTML files.<br><br>Default delimiter is %. |
| `JavaScriptBlocksHeartBeat` | Optional | Configuration variable that indicates if `HeartBeat` is blocked when a JavaScript call is a blocking call.<br><br>Default value is `false`. |

*Table 4–13   (Cont.)  Advanced Configuration Parameters*

| Parameter | Required/ Optional | Parameter Value and Description |
|---|---|---|
| `legacy_lifecycle` | Optional | Applet parameter for Oracle Java Plug-in. A value of true causes a running applet to be reused when requested.  This parameter also affects the contents of the initial page that is generated as the response from the Forms servlet, to ensure the reusability of the applet when `legacy_lifecycle` is set to true. When set to true, JavaScript must be enabled on the Java client. Default value is `false`. |
| `maxRuntimeProcesses` | Optional | This specifies the maximum allowable number of concurrent Forms run-time processes.  It should be set a value that reflects the customer's hardware configuration (and the portion that can be used by Forms applications).  A value of 0 (the default) indicates that there is no explicit limit.  This default is not recommended, because it leaves the system vulnerable to Denial of Service attacks. Default value if not specified is 0. |
| `networkRetries` | Optional | Number of times client should retry if a network failure occurs. Default value is 0. |
| `networkStats` | Optional | Set this parameter to `true` to enable the display of the aggregate statistics in the status bar. It also enables the display of round-trip statistics in the java console. This network statistics feature is enabled only when this parameter is defined in the baseHTML files. Default value is `false`. |
| `obr` | Optional | For internal use only. Default value is no. This parameter is a sub-argument for otherparams. |
| `otherparams` | Optional | This setting specifies command line parameters to pass to the Forms run-time process in addition to form and userid. This parameter is a runform parameter. Default value from `formsweb.cfg` is `obr=%obr% record=%record% tracegroup=%tracegroup% log=%log% term=%term% ssoProxyConnect=%ssoProxyConnect%` **Note**: Special syntax rules apply to this parameter when it is specified in a URL: a + may be used to separate multiple name=value pairs (see Section 3.3.4, "Specifying Special Characters in Values of Runform Parameters" for more information). For production environments, to provide better control over which runform parameters, end users can specify in a URL, include the otherparams parameter in the value of the restrictedURLparams parameter. |

*Table 4–13   (Cont.)  Advanced Configuration Parameters*

| Parameter | Required/ Optional | Parameter Value and Description |
|---|---|---|
| pingStats | Optional | Set the value to true to enable the pinging of the managed server by the java applet when Forms is being rendered. The ping result is, then, displayed on the java console.<br><br>This feature is enabled only when this parameter is defined in the baseHTML files.<br><br>Default value is false. |
| pingWait | Optional | This parameter indicates the maximum amount of time (in milliseconds) that the pingStats parameter must wait for in order to receive a response from the server.<br><br>This feature is enabled only when this parameter is defined in the baseHTML files.<br><br>Default value if not specified is 300. |
| prestartIncrement | Optional | The number of run-time processes to be created when the number of prestarted run-time processes is less than minRuntimes.<br><br>Default value if not specified is 1. |
| prestartInit | Optional | Number of the run-time processes that should be spawned initially.<br><br>Default value if not specified is 1. |
| prestartMin | Optional | Minimum number of run-time processes to exist in the pool. Default value if not specified is 0. |
| prestartRuntimes | Optional | Run-time prestarting or pooling is enabled only if true.<br><br>Default value if not specified is false. |
| prestartTimeout | Optional | Time in minutes after which all the prestarted processes of this pool (configuration section) is stopped. A run-time process is removed from the prestart pool after the client connection is made and thus is not stopped.<br><br>Default value if not specified is 0. |
| query_only | Optional | Set this parameter to yes to prevent the end user from inserting, updating, or deleting records.<br><br>Default value if not specified is no. This parameter is a sub-argument for otherparams. |
| quiet | Optional | Set this parameter to yes to prevent messages from producing an audible beep.<br><br>Default value if not specified is no. This parameter is a sub-argument for otherparams. |
| recordFileName | Optional | Forms applet parameter that specifies the name of file (for example, d:\temp\is) that stores the recorded Forms messages.<br><br>Default value if not specified is 'is' (without the quotes). |

*Table 4–13   (Cont.)  Advanced Configuration Parameters*

| Parameter | Required/ Optional | Parameter Value and Description |
|---|---|---|
| restrictedURLpara ms | Optional | Forms applet parameter. Specifies a comma-delimited list of parameters which is rejected if specified in a URL.<br><br>Default value from `formsweb.cfg` is "`pageTitle,HTMLbodyAttrs,HTMLbefor eForm,HTMLafterForm,log`". |
| restrictedURLchar s | Optional | Forms applet parameter. Specifies a comma-delimited characters that is restricted for use in the request URL's query string. |
| serverApp | Optional | Forms applet parameter.<br><br>Default value is `default`. |
| serverURL | Required | Determines the URL path to Forms Listener Servlet.<br><br>Default value is `/forms/lservlet`. |
| term | Optional | The full path of a custom key binding file (to be used instead of the standard `fmrweb` or `fmrweb_utf8` files).<br><br>This parameter is a sub-argument for otherparams. |

### 4.2.5.8  List of Parameters that Cannot be Specified in the URL

This section lists the parameters that can be specified only in the servlet configuration file (`formsweb.cfg`).  If any are specified in the URL, the value is ignored. In addition, any parameter that is listed in the value of the restrictedURLparams parameter is rejected if specified in the URL.

- `allowNewConnections`
- `baseHTML`
- `baseHTMLjpi`
- `connectionDisallowedURL`
- `defaultCharset`
- `envFile`
- `escapeparams`
- `HTMLdelimiter`
- `maxRuntimeProcesses`
- `prestartIncrement`
- `prestartInit`
- `prestartMin`
- `prestartRuntimes`
- `prestartTimeout`
- `restrictedURLparams`
- `restrictedURLchars`

- `serverURL`
- `ssoCancelURL`
- `ssoDynamicResourceCreate`
- `ssoErrorURL`
- `ssoMode`
- `workingDirectory`

## 4.3 Managing Environment Variables

Use the **Environment Configuration** page of Fusion Middleware Control to manage environment variables. From this page, you can add, edit, or delete environment variables as necessary.

The environment variables such as `PATH`, `ORACLE_INSTANCE`, `ORACLE_HOME`, and `FORMS_PATH` for the Forms run-time executable (`frmweb.exe` on Windows and `frmweb` on UNIX) are defined in `default.env`. The Forms listener servlet calls the executable and initializes it with the variable values provided in the environment file, which is found in the `$DOMAIN_HOME/config/fmwconfig/servers/WLS_FORMS/applications/formsapp_11.1.2/config` directory by default.

Any environment variable that is not defined in `default.env` is inherited from the Oracle WebLogic Managed Server. The environment file must be named in the `envFile` parameter in the Default section of the **Web Configuration** page.

A few things to keep in mind when customizing environment variables are:

- Environment variables may also be specified in the Windows registry. Values in the environment file override settings in the registry. If a variable is not set in the environment file, the registry value is used.
- You need administrator privileges to alter registry values.
- The server does not require restarting for configuration changes to take effect.
- Existing Forms processes are not affected by environment variables that were defined after they were started.
- Environment variables not set in the environment file or Windows registry are inherited from the environment of the parent process, which is the Oracle WebLogic Managed Server.

Table 4–14, " Default Environment Variables" describes important environment variables that are specified in `default.env`.

### 4.3.1 Managing Environment Configuration Files

**To access the Environment Configuration page:**

1. Start Fusion Middleware Control.

2. From the Fusion Middleware Control main page, click the link to the Oracle Forms Services instance that you want to configure.

3. From the Forms menu list, select **Environment Configuration.** The **Environment Configuration** page (Figure 4–4) is displayed.

*Figure 4–4   Environment Configuration page*
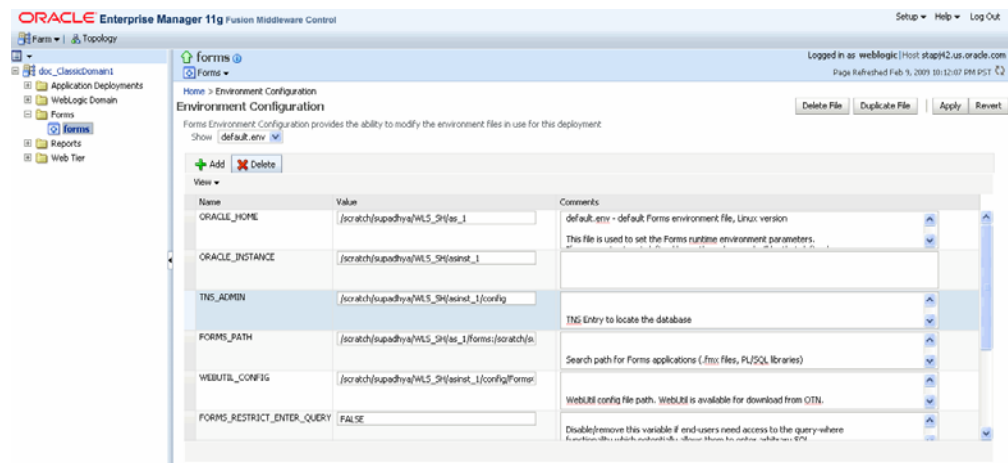


**To duplicate an environment configuration file:**

1.  From the **Environment Configuration** page, click Duplicate File.

    The Duplicate File dialog is displayed.

2.  Select the file which you want to duplicate and enter a unique name for the file.

3.  Click Duplicate to create the file.

**To delete an environment configuration file:**

1.  In the **Environment Configuration** page, from the **Show** menu list, select the environment configuration file you want to delete.

2.  Click **Delete File**.

    The Confirmation dialog is displayed.

3.  Click **Yes** to confirm the deletion.

> **Note:**   You cannot delete `default.env`. You can delete only user-defined environment configuration files.

**To view an environment configuration file:**

1.  In the **Environment Configuration** page, from the Show menu list, select the environment configuration file that you want to view.

2.  The parameters and their values are displayed.

## 4.3.2 Configuring Environment Variables

**To edit an environment variable:**

1.  In the **Environment Configuration** page, select the row of the parameter that contains the environment variable you want to edit.

2.  Enter the Value and Comments.

3.  Click **Apply** to save the changes or **Revert** to discard them.

**To add an environment variable:**

1. From the **Show** menu list, select the environment configuration file to which you want to add the variable.

2. Click **Add** to add a parameter.

   The Add dialog box is displayed.

3. Enter the Name, Value and Comments.

4. Click **Create**.

5. Click **Apply** to save the changes or **Revert** to discard them.

**To delete an environment variable:**

1. From the **Show** menu list, select the environment configuration file where you want to delete an environment variable.

2. Select the rows of the parameters you want to delete. You can delete more than one parameter at a time.

3. Click **Delete**.

4. Click **Apply** to save the changes or **Revert** to discard them.

### 4.3.3 Default Environment Variables

Table 4–14 provides the valid values and a description of some of the environment variables.

*Table 4–14    Default Environment Variables*

| Parameter | Valid Values | Description |
| --- | --- | --- |
| ORACLE_HOME | ORACLE_HOME (default) | Points to the base installation directory of any Oracle product. |
| ORACLE_INSTANCE | ORACLE_INSTANCE (default) | Contains all configuration files, repositories, log files, deployed applications, and temporary files. |
| PATH | ORACLE_HOME/bin (default) | Contains the executables for Oracle products. |
| FORMS_PATH | ORACLE_HOME/forms:ORACLE_INSTANCE/FormsComponent/forms (default) | Specifies the path that Oracle Forms searches when looking for a form, menu, or library to run.<br><br>For **Windows**, separate paths with a *semi-colon* (;).<br><br>For **UNIX**, separate paths with a *colon* (:). |
| FORMS_RESTRICT_ENTER_QUERY | TRUE (default) | Disable or remove this variable for end-users who need access to the query-where functionality which potentially allows them to enter arbitrary SQL statements when in enter-query mode. |
| TNS_ADMIN | ORACLE_INSTANCE/config | Specifies the path name to the TNS files such as TNSNAMES.ORA, SQLNET.ORA and so on. |

*Table 4–14   (Cont.)  Default Environment Variables*

| Parameter | Valid Values | Description |
|---|---|---|
| CLASSPATH | ORACLE_ HOME/jdk/bin/java | Specifies the Java class path, which is required for Forms using imported Java. |
| LD_LIBRARY_PATH | Set the `LD_LIBRARY_PATH` environment variable for the first time to `ORACLE_HOME/lib`.<br><br>You can reset `LD_LIBRARY_PATH` in the Bourne shell by entering:<br><br>`$ set LD_LIBRARY_ PATH=ORACLE_ HOME/lib:${LD_LIBRARY_ PATH}`<br><br>`$ export LD_LIBRARY_ PATH`<br><br>or in the C shell by entering:<br><br>`% setenv LD_LIBRARY_ PATH ORACLE_ HOME/lib:${LD_LIBRARY_ PATH}` | Oracle Forms Developer and Reports Developer products use dynamic, or shared, libraries. Therefore, you must set `LD_ LIBRARY_PATH` so that the dynamic linker can find the libraries. |
| WEBUTIL_CONFIG | ORACLE_ INSTANCE/config/FormsC omponent/forms/server/ webutil.cfg | |
| FORMS_MESSAGE_ ENCRYPTION | TRUE | Possible values are `TRUE` or `FALSE`. Use this environment variable to turn off or on the proprietary obfuscation applied to Forms messages when using HTTP mode. By default, communication is obfuscated. |
| LD_PRELOAD | <JDK_ HOME>/jre/lib/i386/lib jsig.so | Specifies the location of the library `libjsig.so`. This library is used for the signal-chaining facility offered by JVM 1.5. The signal-chaining facility enables an application to link and load the shared library `libjsig.so` before the system libraries. Ensure this is set for Forms and Reports integration on UNIX/Linux.<br><br>**Note**: If there are multiple environment files, ensure that `LD_PRELOAD` has the same settings as in `default.env`. |
| FORMS_PLSQL_BHVR_ COMMON_SQL | To enable the feature, set the `FORMS_PLSQL_BHVR_ COMMON_SQL` environment variable to true or 1.<br><br>To disable the feature, set the environment variable value to false or 0. | If this variable is set, PL/SQL uses a common SQL parser (that is, the one in RDBMS SQL engine) for compiling SQL code rather than the separate one built in to PL/SQL used for compiling static SQL. |

*Table 4–14   (Cont.)  Default Environment Variables*

| Parameter | Valid Values | Description |
| --- | --- | --- |
| FORMS_MODULE_PATH | A list of paths, separated by colons on UNIX, or seperated by semi-colons on Windows. | Setting this environment variable to a non-empty value restricts the directories from which Forms applications may be launched.  The significant effects are as follows: |
| | | ■  The initial Form must specify a path that appears either in the value of FORMS_PATH, ORACLE_PATH, FORMS_MODULE_PATH, or is a subdirectory (without any references to the parent directory) of a path in the value of FORMS_MODULE_PATH. |
| | | If no such match is found, an error message "FRM-40010: Cannot read form" is displayed. |
| | | ■  All Forms, menus, and libraries (.fmx, .mmx, .plx, and .pll files) specified by the application must include a path. |
| | | For example, forms specified in a CALL_FORM, NEW_FORM, or OPEN_FORM, as well as the initial form must include a path. |
| | | The current working directory is not  searched. |

> **Note:**   On Windows, Oracle Forms Services reads Oracle environment settings from the Windows Registry unless they are set as environment variables.

## 4.4  Managing User Sessions

Administrators can manage user sessions, and related features such as monitoring, debugging and tracing using Fusion Middleware Control.

A user session starts when the frmweb process starts. Use the Forms User Sessions pages to monitor and trace the Forms sessions within a Forms Instance. The Forms User Sessions page is accessed from the Forms menu list by selecting **User Sessions**.

**To view Forms user sessions:**

1.  Start Fusion Middleware Control.

2.  From the Forms menu list, select **User Sessions**.

    The **User Sessions** page (Figure 4–5) is displayed.

*Figure 4–5   User Sessions page*
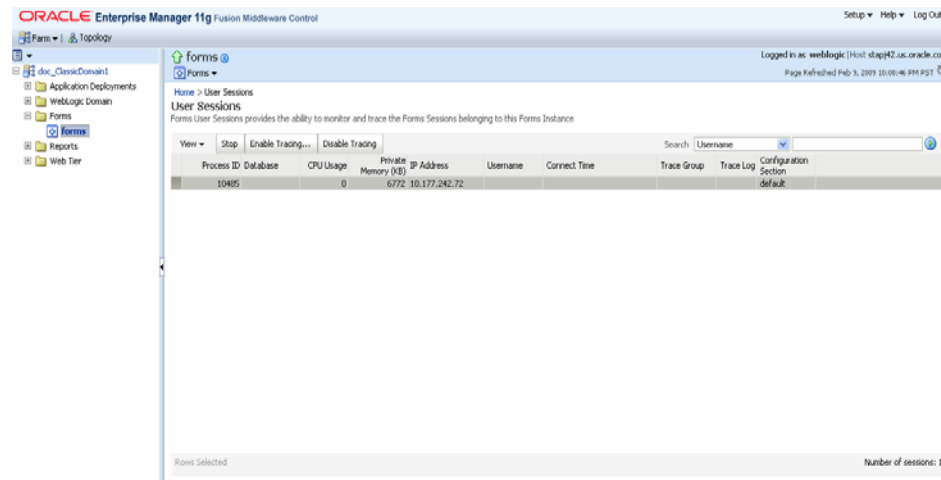


3.   Table 4–15 describes the fields on the User Sessions page.

*Table 4–15    User Sessions Page*

| Field | Description |
| --- | --- |
| Process ID | The process ID of the user session. |
| Database | The database name used by the Forms application for the user session. Click the Database name to view the Database Sessions page. |
| CPU Usage | The percentage of CPU used by the run-time process. |
| Private Memory (KB) | The memory used by the run-time process. On Linux platforms, private memory is not the actual private memory but indicates the Resident Set Size (RSS). |
| IP Address | The IP address of the client computer used to connect to Forms Services. |
| Username | Database user name. |
| Connect Time | The time when the user connected to Forms Services. If the client connection time and client IP are empty, the session is a prestarted session, which is not yet connected to any client. |
| Trace Group | The trace group used for tracing the user session. When tracing is enabled, this column shows the trace group name or the events being traced. The events are displayed if the events of the trace group that was enabled for the session have been later modified in the trace configuration.<br><br>Note that the Trace group name that is displayed may not be indicate the accurate events being traced if built-ins are used to control the tracing. |
| Trace Log | Displays the trace log if one exists for the user session. |
| Configuration Section | Indicates the configuration section used by the Forms application. |
| Form Name | Indicates the module name of the form application. |
| CPU Time | Indicates total CPU time used by forms sessions since Connect time. |

**To enable new Forms user sessions:**

By default, new Forms user sessions are enabled. You can disable them by using Fusion Middleware Control to set the `allowNewConnections` parameter to false.

1. Start Fusion Middleware Control.

2. From the Forms menu, select **Web Configuration**.

3. Select the default configuration section. allowNewConnections cannot be overridden in named sections.

4. In the Sections region, find and edit the value for the `allowNewConnections` parameter. A value of `true` (default) enables new user sessions, whereas `false` disables them.

5. Click **Apply** to save the changes.

**To disable new Forms user sessions:**

1. Start Fusion Middleware Control.

2. From the Forms menu, select **Web Configuration**.

3. Select the default configuration section. allowNewConnections cannot be overridden in named sections.

4. In the Sections region, find and edit the value for the `allowNewConnections` parameter. A value of `true` (default) enables new user sessions, whereas `false` disables them.

5. Click **Apply** to save the changes.

When new user sessions are disabled, attempted connections are directed to a URL identified by the `formsweb.cfg` parameter `connectionDisallowedURL` (in the default section). You must specify a complete and valid URL as the value.

If `connectionDisallowedURL` is not specified, then the following message is displayed in the browser:

```
The Forms servlet will not allow new connections. Please contact
your System Administrator.
```

When you disable new user sessions, existing forms sessions are unaffected and the Oracle WebLogic Managed Server instance remains up.

**To enable tracing for a Forms user sessions:**

1. Start Fusion Middleware Control.

2. In the User Sessions page, select the row that has the user session for which you want to enable tracing.

3. Select Enable Tracing.

4. From the Select Trace Group list, select an available trace group and click **OK**.

**To disable tracing for a Forms user sessions:**

1. In the User Sessions page, select the row that has the user session for which you want to disable tracing.

2. Click **Disable Tracing**.

3. Click **OK**. The Disable Tracing dialog is dismissed and tracing is now stopped for the selected Forms user session.

**To terminate a Forms user session:**

1. Select the link to the Forms Services instance that has the user session to be terminated.

2. From the Forms menu, select **User Sessions**.

3. Click the row of the user session to be deleted.

4. Click **Stop**.

5. The Confirmation dialog is displayed.

6. Click **Yes**.

   The user session is deleted and the Runform instance is terminated.

**To view trace logs of a Forms user sessions:**

1. From the Forms menu, select **User Sessions**.

2. For a user session that is active, click **View Trace Log** in the **Trace Log** column. Log in to view the trace file.

**To search for a Forms user sessions:**

1. From the Forms menu, select **User Sessions**.

2. Select the column name in which you want to search.

3. Enter the search string.

4. Click the blue arrow to search. The search results are displayed.

**To sort the list of Forms user sessions:**

1. From the Forms menu, select **User Sessions**.

2. Move the mouse over the column.

3. Click the up or down arrow to sort in ascending or descending order. The page is refreshed showing the sorted user sessions. You can sort in order of all columns except Trace Logs.

**To customize your view of Forms user sessions:**

1. From the User Sessions page, click View.

2. From the View menu, you can:

   ■ Select **Show All** to view all columns.

   ■ Select specific columns you want displayed.

   ■ Select **Reorder Columns** to organize the order of display of the columns.

   ■ Select **Show More Columns** to hide or display specific columns.

**To view database sessions for a Forms user session:**

1. From the Forms menu, select **User Sessions**.

2. Click the Database name in the Database column.

   Log in to view the Database Sessions page (Figure 4–6). You need Database Administrator privileges to log in to Database Sessions page.

*Figure 4–6   Database Sessions Page*
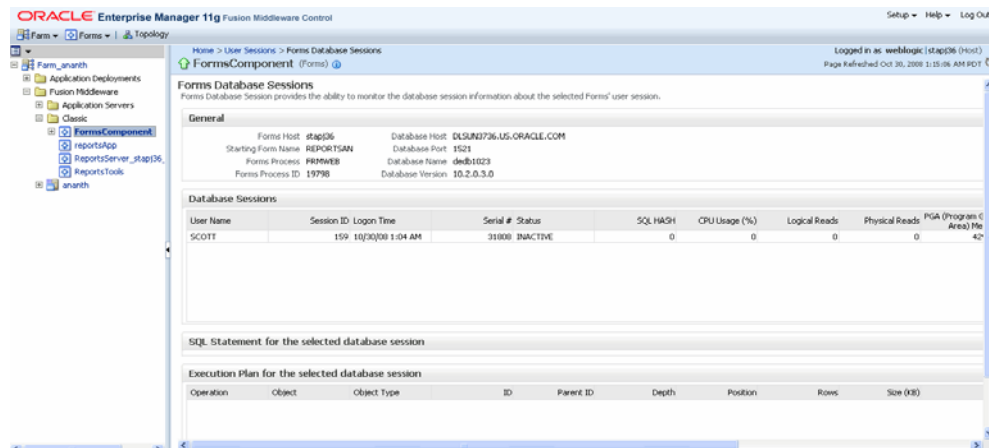


3.  Table 4–16, Table 4–17, and Table 4–18 describe the information displayed in the Database Sessions page.

*Table 4–16    Database Sessions Page*

| Field | Description |
| --- | --- |
| Username | Database username used for connection to the database. |
| Session ID | Database session identifier. |
| Logon Time | Date and time when user logged on to the session. |
| Serial # | Session serial number. Used to uniquely identify a session's objects. Guarantees that session-level commands are applied to the correct session objects if the session ends and another session begins with the same session ID. |
| Status | Indicates whether the session is active or not. |
| SQL HASH | Used to identify the SQL statement executed |
| CPU Usage (%) | CPU Usage (in percentage) on the Database system for the given session. |
| Logical Reads | Number of Logical Reads for the given session. |
| Physical Reads | Number of Physical Reads for the given session. |
| PGA (Program Global Area) Memory | Size of PGA (Program Global Area) Memory after an interval. |

*Table 4–17    Details of Selected Database Session*

| Field | Description |
| --- | --- |
| SQL Statement for the selected Database Session | Displays the most recent SQL statement. |

*Table 4–18    Execution Plan for the Selected Database Session*

| Field | Description |
| --- | --- |
| Operation | Name of the internal operation performed in the execution step (for example, TABLE ACCESS). |
| Object | Name of the table or index. |

*Table 4–18 (Cont.) Execution Plan for the Selected Database Session*

| Field | Description |
| --- | --- |
| Object Type | Type of the object. |
| ID | A number assigned to each step in the execution plan. |
| Parent ID | ID of the next execution step that operates on the output of the current step. |
| Depth | Depth (or level) of the operation in the tree. It is not necessary to issue a CONNECT BY statement to get the level information, which is generally used to indent the rows from the PLAN_TABLE table. The root operation (statement) is level 0. |
| Position | Order of processing for all operations that have the same PARENT_ID. |
| Rows | Estimate, by the cost-based optimizer, of the number of rows produced by the operation. |
| Size (KB) | Estimate, by the cost-based optimizer, of the number of bytes produced by the operation. |
| Cost | Cost of the operation as estimated by the optimizer's cost-based approach. For statements that use the rule-based approach, this column is null. |
| Time (sec) | Elapsed time (in seconds) of the operation as estimated by the optimizer's cost-based approach. For statements that use the rule-based approach, this column is null. |
| CPU Cost | CPU cost of the operation as estimated by the optimizer's cost-based approach. For statements that use the rule-based approach, this column is null. |
| I/O Cost | I/O cost of the operation as estimated by the optimizer's cost-based approach. For statements that use the rule-based approach, this column is null. |

## 4.5 Managing URL Security for Applications

Oracle Forms applications are web-deployed solutions that users access through a browser. Oracle Forms architecture allows Forms developers two ways to choose and configure how a Forms application runs. One option is to set the parameter and the value in the URL. The second option is to set the parameter and its value(s) in the configuration file, that is, `formsweb.cfg`. The parameter that is set in the `formsweb.cfg` can be overridden by the parameter set in the URL.

A Forms administrator can override this default behavior, and give the Forms administrator full control over what parameter can be used in the URL.

Here are two scenarios to consider when deciding which parameters to allow or not allow in a URL. The first scenario is when an administrator just wants to restrict the usage of the USERID parameter in the URL that forces the end-user to always log in using the default login window. The second scenario is when an administrator disables all parameters except a few, such as `CONFIG=MyApp` in a URL.

The parameter `restrictedURLparams` allows flexibility for the Forms administrator to consider any URL-accessible parameter in the `formsweb.cfg` file as restricted to a user. An administrator can specify this parameter in a named configuration section to override the one specified in the default configuration section. The `restrictedURLparams` parameter itself cannot be set in the URL.

By design, command line arguments passed in a URL always override similar definitions in the formsweb.cfg.

In this example, the `userid` is defined as `scott/tiger` and `debug` is set to `false`. An application that is configured to connect to the database as `scott/tiger` can connect as a different user with the `userid` parameter added as a URL parameter. To prevent this, the `userid` parameter is defined in the `restrictedURLparams` as shown in Figure 4–7, "Defining the restrictedURLparams Parameter".

*Figure 4–7   Defining the restrictedURLparams Parameter*



Similarly, an administrator can use the `restrictedURLparams` parameter to redirect a user to a page which lists the restricted parameters that were used.

## 4.5.1  Securing the Oracle Forms Test Form

The test form runs when you access an Oracle Forms URL but do not specify an application to run. For example, normally you call an Oracle Forms application with the following syntax:

```
http://<host>:<port>/forms/frmservlet?config=myApp
```

The Forms servlet locates `[myApp]` in the formsweb.cfg file and launches that application. However, when no application is specified, for example:

```
http://<host>:<port>/forms/frmservlet
```

The Forms servlet uses the settings in the default section of the formsweb.cfg file. These settings are located under `[default]` in the Forms Configuration file (anytime an application does not override any of these settings, the defaults are used). The default section has the following setting:

```
form=test.fmx
```

This is the test form which enables you to test your Oracle Forms Services installation and configuration. Thus if you do not specify an application, Forms launches the test.fmx file. You could change this to:

```
form=
```

And the form does not run. However, this is not optimal; the Forms servlet still sends the dynamically generated HTML file to the client, from which a curious user could obtain information. The optimally secure solution is to redirect requests to an informational HTML page that is presented to the client instead. Some parameters in the `formsweb.cfg` file must be changed.

Here are the parameters to change, along with their default values when you install Oracle Forms Services:

```
# System parameter: default base HTML file
baseHTML=base.htm
# System parameter: base HTML file for use with Sun's Java Plug-In
```

```
baseHTMLjpi=basejpi.htm
```

These parameters are templates for the HTML information that are sent to the client. Create an informational HTML page and have these variables point to that instead. For example, in the `$ORACLE_ INSTANCE/config/FormsComponent/forms/server` directory, create a simple HTML page called `forbidden.html` with the following content:

```
<html>
  <head>
    <title>Forbidden</title>
  </head>
  <body>
   <h1>Forbidden!</h1>
    <h2>You may not access this Forms application.</h2>
  </body>
</html>
```

> **Note:** This message page displayed as a result of redirecting of client information is different from the page that the Web server returns when the requested content has restricted permissions on it.

Next, modify the `formsweb.cfg` parameters by commenting out or modifying the original parameters:

```
# System parameter: default base HTML file
#baseHTML=base.htm
baseHTML=forbidden.html
# System parameter: base HTML file for use with Sun's Java Plug-In
#baseHTMLjpi=basejpi.htm
baseHTMLjpi=forbidden.html
# System parameter: base HTML file for use with Microsoft Internet Explorer
# (when using the native JVM)
```

When a user enters the URL

```
http://<host>:<port>/forms/frmservlet
```

the customized Web page is presented. Of course, you can customize `forbidden.html`, including its contents, its filename, and its location if you make the corresponding changes to these parameters in the `formsweb.cfg` file. Administrators can put any information, such as warnings, errors, time stamps, IP logging, or contact information in this information Web page with minimal impact on the server configuration.

> **Note:** Overriding the base HTML template entries in the default section of `formsweb.cfg` requires that you add the same entries pointing to the original values (or some other valid HTML file) in your application-specific named configuration:
>
> ```
> [myApp]
> form=myApplication.fmx
> lookandfeel=oracle
> baseHTML=base.htm
> baseHTMLjpi=basejpi.htm
> ```
>
> If you do not specify these base HTML values, and when a user runs an application, the `forbidden.html` page is displayed because the application-specific configuration section has not overridden the default values.

## 4.6  Creating Your Own Template HTML Files

Consider creating your own HTML file templates (by modifying the templates provided by Oracle). By doing this, you can hard-code standard Forms parameters and parameter values into the template. Your template can include standard text, a browser window title, or images (such as a company logo) that would appear on the first Web page users see when they run Web-enabled forms. Adding standard parameters, values, and additional text or images reduces the amount of work required to customize the template for a specific application. To add text, images, or a window title, you must include the appropriate tags in the template HTML file.

See Chapter 3.3.4, "Specifying Special Characters in Values of Runform Parameters" for information about coding the `serverArgs` applet parameter.

Any user-added customized configuration files (such as user client registry files or user key binding files or multiple environment files) must be copied to the same directory as the corresponding default configuration file.

For example, if the user has created a French environment configuration file `default_fr.env`, then it must be placed in the `$DOMAIN_HOME/config/fmwconfig/servers/WLS_FORMS/applications/formsapp_11.1.2/config` directory.

### 4.6.1  Variable References in Template HTML Files

When a variable reference occurs within a string delimited by quotes or apostrophes (for example, the value of an applet parameter), then when the value of the variable is substituted for the variable reference, HTML metacharacters ('&', '<', '>', quote, and apostrophe) are replaced by HTML escape sequences.

This sequence is *not* done for variable references outside delimited strings. Therefore, such variables should be specified in the `restrictedURLparams` system default configuration parameter, for security reasons.

> **Note:** To modify the cursor blink rate, or disable blinking, set the client parameter `cursorBlinkRate` as follows.
>
> ```
> <PARAM NAME="cursorBlinkRate" VALUE="1000">
> ```
>
> The default is 600 milliseconds: the cursor completes one full blink every 1.2 seconds (1200 ms).
>
> A value of zero disables the blinking and the cursor remains visible all the time.

## 4.7 Deploying Fonts, Icons, and Images Used by Forms Services

This section explains how to specify the default location and search paths for fonts, icons, and images in Registry.dat. To look at a sample of the default Registry.dat file, see Section C.8.1, "Registry.dat".

### 4.7.1 Managing Registry.dat with Fusion Middleware Control

Use Fusion Middleware Control to change, add, or delete parameters from `Registry.dat`.

**To access the Fonts and Icon Mapping page:**

1. Start Fusion Middleware Control.

2. From the Forms menu list, select **Font and Icon Mapping**.

   The Font and Icon Mapping page (Figure 4–8) is displayed.

*Figure 4–8   Font and Icon Mapping Page*
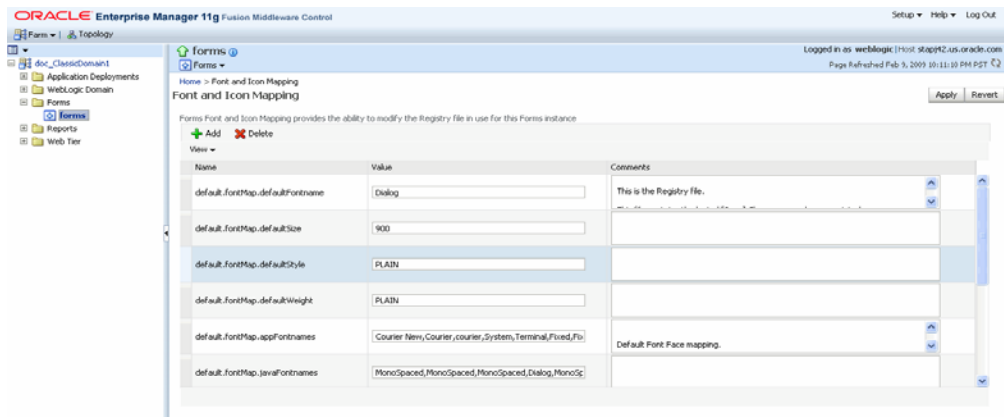


**To edit a Registry.dat parameter value:**

1. Start Fusion Middleware Control.

2. From the Forms menu list, select **Font and Icon Mapping**.

3. Select the row containing the parameter to modify and change the value(s) for it in the **Value** text field.

4. Click **Apply** to save the changes.

**To add a Registry.dat parameter and its value:**

1. From the Forms menu list, select **Font and Icon Mapping**.

2. Click Add.

   The Add dialog appears.

3. Enter the name, value, and comments for this parameter.

4. Click **Create**.

5. Click **Apply** to save or **Revert** to discard the changes.

**To delete a Registry.dat parameter and its value:**

1. From the Forms menu list, select **Font and Icon Mapping**.

2. Select the row containing the parameter to delete and click **Delete**.

3. The parameter is deleted.

4. Click **Apply** to save or **Revert** to discard the changes.

## 4.7.2 Managing Application Fonts

Using Fusion Middleware Control, you can also change the default font and font settings by the Registry.dat file. All font names are Java Font names. Each of these parameters represents the default property to use when none is specified.

**To change the font settings for a deployed application:**

1. Start Fusion Middleware Control.

2. From the Forms menu list, select **Font and Icon Mapping**.

3. Change any of the settings to reflect your desired font setting, based on Table 4–19:

*Table 4–19    Default Font Values*

| Font Name | Default Value |
|---|---|
| default.fontMap.defaultFontname | Dialog<br><br>Represents the default Java fontName. |
| default.fontMap.defaultSize | 900<br><br>Represents the default fontSize.  Note that the size is multiplied by 100 (for example, a 10pt font has a size of 1000). |
| default.fontMap.defaultStyle | PLAIN<br><br>Represents the default fontStyle, PLAIN or *ITALIC*. |

*Table 4–19   (Cont.)  Default Font Values*

| Font Name | Default Value |
|---|---|
| default.fontMap.defaultWeight | PLAIN |
| | Represents the default fontWeight, PLAIN or **BOLD**. |
| default.fontMap.appFontnames | Courier New,Courier,courier,System,Terminal,Fixedsys,Times,Times New Roman,MS Sans Serif,Arial |
| | Default Font Face mapping. Represents a comma delimited list of application font names. |
| | The number of entries in the appFontname list should match the number in the javaFontname list. The elements of the list are comma separated and *all* characters are taken literally; leading and trailing spaces are stripped from Face names. |
| | Note that this file uses the Java 1.1 font names to handle the NLS Plane. |
| default.fontMap.javaFontnames | MonoSpaced,MonoSpaced,MonoSpaced,Dialog,MonoSpaced,Dialog,Dialog,Serif,Serif,Dialog,SansSerif |
| | Represents a comma delimited list of Java font names. |

For example, to change your default font to Times New Roman, replace **Dialog** with **Times New Roman**.

You can change the default font face mappings:

```
default.fontMap.appFontnames=Courier New,Courier,
courier,System,Terminal,Fixed,Fixedsys,Times,Times New Roman,
MS Sans Serif,Arial
default.fontMap.javaFontnames=MonoSpaced,MonoSpaced,MonoSpaced,Dialog,
MonoSpaced,Dialog,Dialog,Serif,Serif,Dialog,SansSerif
```

4.  Click **Apply** to save the changes.

Some fonts on Windows are not supported in Java. For this reason you can specify (map) Java-supported fonts that appear when a non-supported font is encountered. In the previous sample, each font in default.fontMap.appFontnames corresponds to a font in default.fontMap.javaFontnames.

### 4.7.3  Deploying Application Icons

When deploying an Oracle Forms application, the icon files used must be in a Web-enabled format, such as JPG or GIF (GIF is the default format).

By default, the icons are found relative to the `DocumentBase` directory. That is, `DocumentBase` looks for images in the directory relative to the base directory of the application start HTML file. As the start HTML file is dynamically rendered by the Forms servlet, the Forms `webapp's` directory becomes the document base. The Forms `webapp's` directory is located at `$DOMAIN_HOME/servers/WLS_FORMS/tmp/_WL_ user/formsapp_11.1.2/<random string>/war`.

For example, if an application defines the icon location for a button with `myapp/<iconname>`, then the icon is looked up in the directory `forms/myapp`.

To change the default location, set the `imageBase` parameter to `codebase` in the Web Configuration page of Enterprise Manager Fusion Middleware Control. Alternatively,

you can change the `default.icons.iconpath` value of the Registry.dat file in the `$DOMAIN_HOME/config/fmwconfig/servers/WLS_ FORMS/applications/formsapp_ 11.1.2/config/forms/registry/oracle/forms/registry` directory.

Setting the imageBase parameter to `codebase` enables Oracle Forms to search the `forms/java` directory for the icon files. Use this setting if your images are stored in a Java archive file. Changing the image location in the Registry.dat configuration file is useful to store images in a central location independent of any application and independent of the Oracle Forms installation.

### 4.7.3.1  Storing Icons in a Java Archive File

If an application uses a lot of custom icon images, it is recommended you store icons in a Java archive file and set the `imageBase` value to `codebase`. The icon files can be zipped to a Java archive using the Jar command of any Java Software Development Kit (Java SDK).

For example, the command `jar -cvf myico.jar *.gif` packages all files with the extension `.gif` into an archive file with the name myico.jar.

In order for Oracle Forms to access the icon files stored in this archive, the archive must be stored into the `forms/java` directory. Also, the name of the archive file must be part of the archive tag used in the custom application section of the `formsweb.cfg` file. Now, when the initial application starts, the icon files are downloaded and permanently stored on the client until the archive file is changed.

> **Note:**  Oracle Forms default icons (for example, icons present in the default smart icon bar) do not require deployment, as they are part of the `frmall.jar` file.

### 4.7.3.2  Adding, Modifying, and Deleting Icon Mappings

Use Fusion Middleware Control to add icon changes to the Registry.dat file used by your application.

**To add icon mappings:**

1. Start Fusion Middleware Control.

2. From the Forms menu, select **Font and Icon Mapping**.

3. Click **Add**.

   The Add dialog appears.

4. Enter the name, value, and an optional comment.

5. Click **Create** to create the mapping.

   The mapping is added to the list.

6. Click **Apply** to save the changes.

**To modify icon mappings:**

1. From the Font and Icon Mapping region, select the mapping you want to modify.

2. Change the name and value of the mapping. For example,

   - Modify the `iconpath` parameter specifying your icon location:

     `default.icons.iconpath=/mydir`

(for an absolute path)

or

```
default.icons.iconpath=mydir
```

(for a relative path, starting from the `DocumentBase` Directory)

- Modify the `iconextension` parameter:

```
default.icons.iconextension=gif
```

or

```
default.icons.iconextension=jpg
```

3. Click **Apply** to save and activate the changes.

**To delete an icon mapping:**

1. From the Font and Icon Mapping region, select the mapping you want to delete.

2. Click **Delete**.

3. The selected icon mapping is deleted.

4. Click **Apply** to save or **Revert** to discard the changes.

**To reference the application file:**

- In a specific named configuration section in the `formsweb.cfg` file, modify the value of the `serverApp` parameter and set the value to the location and name of your application file.

For example:

```
[my_app]
ServerApp=/appfile/myapp
```

(for an absolute path)

or

```
[my_app]
ServerApp=appfile/myapp
```

(for a relative path, relative to the `CodeBase` directory)

Table 4–20 describes the correct locations where to place your application icons:

*Table 4–20    Icon Location Guide*

| Icon Location | When | How |
|---|---|---|
| DocumentBase | Default. Applications with few or no custom icons. | Store icons in forms `webapp`'s directory or in a directory relative to it. The forms `webapp`'s directory is located at `$DOMAIN_HOME/servers/WLS_FORMS/tmp/_WL_user/formsapp_11.1.2/<random string>/war`. |
| Java Archives | Applications that use many custom icons. | Set `ImageBase` to `codebase`, create Java archive file for icons, and add archive file to the archive parameter in `formsweb.cfg`. |

*Table 4–20   (Cont.)  Icon Location Guide*

| Icon Location | When | How |
|---|---|---|
| Registry.dat | Applications with custom icons that are stored in a different location as the Oracle Forms install (can be another server).<br><br>Useful to make other changes to the Registry.dat file such as font mapping. | Copy Registry.dat and change ServerApp parameter in `formsweb.cfg`. |

## 4.7.4  Splash screen and Background Images

When you deploy your applications, you have the ability to specify a splash screen image (displayed during the connection) and a background  image file.

Those images are defined in the HTML file or you can use the **Web Configuration** page in Enterprise Manager:

```
<PARAM NAME="splashScreen" VALUE="splash.gif">
```

```
<PARAM NAME="background" VALUE="back.gif">
```

The default location for the splash screen and background image files is in the DocumentBase directory containing the baseHTML file.

> **Note:**   Image formats for splash screens and icons are the standard formats that are supported by `java.awt.Image`. For more information on `java.awt.Image`, refer to the Java Advanced Imaging (JAI) API  at
> `http://www.oracle.com/technetwork/java/index.html`.

## 4.7.5  Custom Jar Files Containing Icons and Images

Each time you use an icon or an image (for a splash screen or background), an HTTP request is sent to the Web server. To reduce the HTTP round-trips between the client and the server, you have the ability to store your icons and images in a Java archive (Jar) file. Using this technique, only one HTTP round-trip is necessary to download the Jar file.

### 4.7.5.1  Creating a Jar File for Images

The Java SDK comes with an executable called *jar*. This utility enables you to store files inside a Java archive. For more information, see
`http://www.oracle.com/technetwork/java/index.html`.

For example:

```
jar -cvf myico.jar Splash.gif Back.gif icon1.gif
```

This command stores three files (`Splash.gif`, `Back.gif`, `icon1.gif`) in a single Jar file called `myico.jar`.

### 4.7.5.2 Using Files Within the Jar File

The default search path for the icons and images is relative to the documentBase. However, when you want to use a Jar file to store those files, the search path must be relative to the `codebase` directory, the directory which contains the Java applet.

To use a Jar file to store icons and images, you must specify that the search path is relative to `codebase` using the imageBase parameter in the `formsweb.cfg` file or HTML file.

This parameter accepts two different values:

- **documentBase** The search path is relative to the documentBase directory. If no value is specified for imageBase, then the value of documentBase is used.

- **codeBase** The search path is relative to the codeBase directory, which gives the ability to use Jar files.

In this example, we use a JAR file containing the icons and we specify that the search should be relative to codeBase. If the parameter `imageBase` is not set, the search is relative to `documentBase` and the icons are not retrieved from the Jar file.

For example (`formsweb.cfg`):

```
archive=frmall.jar, icons.jar

imageBase=codeBase
```

## 4.7.6 Search Path for Icons and Images

The icons and images search path depends on:

- What you specify in your custom application file (for the icons).

- What you specified in the `splashScreen` and `background` parameters of your default Forms configuration file or HTML file (for the images).

- What you specify in the `imageBase` parameter in the **Web Configuration** page of Fusion Middleware Control for the file or HTML file (for both icons and images).

Forms Services searches for the icons depending on what you specify. This example assumes:

- *host* is the computer name.

- `DocumentBase` is the URL pointing to the HTML file.

- codebase is the URL pointing to the location of the starting class file (as specified in the `formsweb.cfg` file or HTML file).

- *mydir* is the URL pointing to your icons or images directory.

### 4.7.6.1 DocumentBase

The default search paths for icons and images are relative to the `DocumentBase`. In this case, do not specify the imageBase parameter:

*Table 4–21   Search Paths for Icons*

| Location Specified | Search path used by Forms Services |
|---|---|
| default | http://host/documentbase |
| `iconpath=mydir`<br><br>(specified in your application file) | `http://host/documentbase/mydir`<br><br>(relative path) |

*Table 4–21   (Cont.)  Search Paths for Icons*

| Location Specified | Search path used by Forms Services |
|---|---|
| `iconpath=/mydir`<br><br>(specified in your application file) | `http://host/mydir`<br><br>(absolute path) |

*Table 4–22    Search Paths for Images*

| Location Specified | Search Path Used by Forms Services |
|---|---|
| file.gif (specified, for example, in formsweb.cfg as splashscreen=file.cfg) | http://host/documentbase/file.gif |
| mydir/file.gif | `http://host/documentbase/mydir/file.gif`<br><br>(relative path) |
| /mydir/file.gif | `http://host/mydir/file.gif`<br><br>(absolute path) |

### 4.7.6.2  codebase

Use the `imageBase=codebase` parameter to enable the search of the icons (Table 4–23) and images (Table 4–24) in a Jar file:

*Table 4–23    Icon Search Paths Used by Forms Services*

| Location Specified | Search Path Used by Forms Services |
|---|---|
| default | `http://host/codebase` or root of the Jar file |
| `iconpath=mydir`<br><br>(specified in your application file) | `http://host/codebase/mydir` or in the mydir directory in the Jar file<br><br>(relative path) |
| `iconpath=/mydir`<br><br>(specified in your application file) | `http://host/mydir`<br><br>(absolute path)<br>No Jar file is used. |

*Table 4–24    Image Search Paths Used by Forms Services*

| Location Specified | Search Path Used by Forms Services |
|---|---|
| file.gif | `http://host/codebase/file.gif` or root of the Jar file |
| `mydir/file.gif`<br><br>(specified in your HTML file) | `http://host/codebase/mydir/file.gif` or in the mydir directory in the Jar file<br><br>(relative path) |
| `/mydir/file.gif`<br><br>(specified in your HTML file) | `http://host/mydir/file.gif`<br><br>(absolute path)<br>No Jar file is used. |

## 4.8  Enabling Language Detection

Oracle Forms architecture supports deployment in multiple languages. The purpose of this feature is to automatically select the appropriate configuration to match a user's preferred language. In this way, all users can run Oracle Forms applications using the same URL, yet have the application run in their preferred language. As Oracle Forms

Services do not provide an integrated translation tool, you must have translated application source files.

### 4.8.1 Specifying Language Detection

For each configuration section in the **Web Configuration** page, you can create language-specific sections with names like `<config_name>.<language-code>`. For example, if you created a configuration section "hr", and wanted to create French and Chinese languages, your configuration section might look like the following:

```
[hr]
lookAndFeel=oracle
width=600
height=500
envFile=default.env
workingDirectory=/private/apps/hr
[hr.fr]
envFile=french.env
workingDirectory=/private/apps/hr/french
[hr.zh]
envFile=chinese.env
workingDirectory=/private/apps/hr/chinese
```

### 4.8.2 Inline IME Support

Inline IME support enables Forms Web applications to properly display the composing text in which each character may not be directly represented by a single keystroke (for example, Asian characters) near the insertion cursor (so called inline, or on-the-spot). It is enabled by default. To disable, set the applet parameter "inlineIME" to "false" in the baseHTML file:

```
<HTML>
<!-- FILE: base.htm (Oracle Forms) -->
 <BODY>
 ...
 <OBJECT classid=...
>
<PARAM NAME="inlineIME" VALUE="false">
<EMBED SRC="" ...
inlineIME="false"
>
...
.
</BODY>
</HTML>
```

For more information about using baseHTML, see Appendix C.4, "base.htm and basejpi.htm Files".

### 4.8.3 How Language Detection Works

When the Forms servlet receives a request for a particular configuration (for example, `http://myserv/servlet/frmservlet?config=hr`) it gets the client language setting from the request header `"accept-language"`. This gives a list of languages in order of preference. For example, accept-language: de, fr, en_us means the order of

preference is German, French, then US English. The servlet looks for a language-specific configuration section matching the first language. If one is not found, it looks for the next and so on. If no language-specific configuration is found, it uses the base configuration.

When the Forms servlet receives a request with no particular configuration specified (with no `"config="` URL parameter, for example, `http://myserv/servlet/frmservlet`), it looks for a language-specific section in the default section matching the first language (for example, [.fr]).

### 4.8.3.1  Multi-Level Inheritance

For ease of use, to avoid duplication of common values across all language-specific variants of a given base configuration, only parameters which are language-specific to be defined in the language-specific sections are allowed. Four levels of inheritance are now supported:

1.  If a particular configuration is requested, using a URL query parameter like `config=myconfig`, the value for each parameter is looked for in the langage-specific configuration section which best matches the user's browser language settings (for example in section [myconfig.fr]),

2.  Then, if not found, the value is looked for in the base configuration section ([myconfig],

3.  Then, failing that, in the language-specific default section (for example, [.fr]),

4.  And finally in the default section.

Typically, the parameters which are most likely to vary from one language to another are `"workingDirectory"` and `"envFile"`. Using a different `envFile` setting for each language lets you have different values of `NLS_LANG` (to allow for different character sets, date and number formats) and `FORMS_PATH` (to pick up language-specific fmx files). Using different `workingDirectory` settings provides another way to pick up language-specific .fmx files.

# 4.9  Enabling Key Mappings

A key binding connects a key to an application function. When you bind a key to a function, the program performs that function when you type that keystroke. You define key bindings in the `fmrweb.res` file in the `$ORACLE_INSTANCE/config/FormsComponent/forms/admin/resource/<lang>` directory in UNIX, for example `$ORACLE_INSTANCE/config/FormsComponent/forms/admin/resource/US`. For Windows, the location is `ORACLE_INSTANCE\config\FormsComponent\forms`.

By defining key bindings, you can integrate a variety of keyboards to make an application feel similar on each of them. On some platforms not all keys are able to be re-mapped. For example, on Microsoft Windows, because keys are defined in the Windows keyboard device driver, certain keys cannot be re-mapped. Key combinations integral to Windows, such as Alt-F4 (Close Window) and F1 (Help) cannot be re-mapped. As a general rule, keys which are part of the "extended" keyboard also cannot be re-mapped. These keys include the number pad, gray arrow and editing keys, Print Screen, Scroll Lock, and Pause.

> **Note:** If running with different `NLS_LANG` settings, for example, `NLS_LANG=GERMAN_GERMANY=WE8ISO8859P1`, a different resource file, `fmrwebd.res`, is used. There is a resource file for each supported language. To override this, pass parameter `term=fullpath\filename.res` to the Oracle Forms Runtime process.

It is possible to pass this parameter directly within the URL. For example:

```
http://hostname:port/forms/frmservlet?Form=test.fmx&term=fullpat
h/filename.res
```

You can also set this parameter in the `formsweb.cfg` file, for example:

```
otherParams=term=fullpath\filename.res
```

### 4.9.1 Customizing fmrweb.res

`fmrweb.res` is a text file which can edited with a text editor such as vi in UNIX or Notepad or Wordpad on Windows. Unlike Oracle 6*i* Forms, Oracle Terminal editor is no longer required. The text file is self-documented.

> **Note:** The customization is limited, particularly compared to character mode forms. You *cannot* edit `fmrweb.res` with Oracle Enterprise Manager Fusion Middleware Control.

#### 4.9.1.1 Example change: Swapping Enter and Execute Mappings

In the section marked `USER-READABLE STRINGS`, find the entries with

```
122 : 0 : "F11" : 76 : "Enter Query"
122 : 2 : "Ctrl+F11" : 77 : "Execute Query"
```
and change them to:

```
122 : 2 : "Ctrl+F11" : 76 : "Enter Query"
122 : 0 : "F11" : 77 : "Execute Query"
```

> **Note:** By default `fmrweb.res` does *not* reflect the Microsoft Windows client/server keyboard mappings. It reflects the key mapping if running client/server on UNIX X-Windows/Motif.

A file called `fmrpcweb.res` has also been provided which gives the Microsoft Windows client/server keyboard mappings. To use this file, rename `fmrpcweb.res` to `fmrweb_orig.res`, and copy `fmrpcweb.res` to `fmrweb.res`. Alternatively, use the term parameter as described above.

#### 4.9.1.2 Exceptions/ Special Key Mappings

The following examples show special key mappings:

- Section 4.9.1.2.1, "Mapping F2"
- Section 4.9.1.2.2, "Mapping for ENTER to Fire KEY-ENTER-TRIGGER"
- Section 4.9.1.2.3, "Mapping Number Keys"
- Section 4.9.1.2.4, "Mapping for ESC Key to exit out of a Web Form"

### 4.9.1.2.1 Mapping F2

To map F2, change the default entry for F2, "List Tab Pages", to another key. Here is an example of the default entry:

```
113: 0 : "F2" : 95 : "List Tab Pages"
```

This must be explicitly changed to another key mapping such as the following:

```
113: 8 : "F2" : 95 : "List Tab Pages"
```

To map the F2 function to the F2 key, comment out the lines that begin with "113 : 0" and "113 : 8" with a # symbol and add the following lines to the bottom of the resource file:

```
113: 0 : "F2" : 84 : "Function 2"
113: 8 : " " : 95 : " "
```

Since a new function has been added which uses F2 by default, it is necessary to explicitly map this new function to something else to map the F2 key. This function was added to allow for keyboard navigation between the tab canvas pages and it defaults to F2. Even if it is commented out and not assigned to F2, the F2 key cannot be mapped unless this function, Forms Function Number 95, is mapped to another key.

### 4.9.1.2.2 Mapping for ENTER to Fire KEY-ENTER-TRIGGER

By default, whether deploying client/server or over the Web pressing the ENTER key takes the cursor to the next navigable item in the block. To override this default behavior it is necessary to modify the forms resource file to revise the key mapping details.

Modify `fmrweb.res` and change the Forms Function Number (FFN) from 27 to 75 for the Return Key. The line should be changed to the following:

```
10 : 0 : "Return" : 75 : "Return"
```

By default, the line is displayed with an FFN of 27 and looks as follows:

```
10 : 0 : "Return" : 27 : "Return"
```

This line should NOT fire the Key-Enter trigger since the Return or Enter key is actually returning the Return function represented by the FFN of 27. The FFN of 75 represents the Enter function and fires the Key-Enter trigger.

### 4.9.1.2.3 Mapping Number Keys

The objective is to map CTRL+<number> keys in `fmrweb.res` for numbers 0 to 9 and there are no Java Function keys mentioned for the numbers in `fmrweb.res`. Perform the following steps along with an example that shows the steps needed to map CTRL+1 to 'Next Record':

1. List the Java function key numbers that could be implemented in `fmrweb.res` file for the Key Mapping. For example:

   ```
   public static final int VK_1 = 0x31;
   ```

2. The hexadecimal values have to be converted to their decimal equivalents before their use in `fmrweb.res`.

   In step (1), 0x31 is a hexadecimal value that has to be converted to its decimal equivalent. (Note:1019580.6). For example,

   ```
   SQL> select hextodec('31') from dual;
   HEXTODEC('31')
   --------------
   ```

```
49
```

3. Use this decimal value for mapping the number key 1 in `fmrweb.res`. For example, CTRL+1 can be mapped to 'Next Record' as:

```
49 : 2 : "CTRL+1" : 67 : "Next Record"
```

#### 4.9.1.2.4   Mapping for ESC Key to exit out of a Web Form

1. Make a backup copy of `fmrweb.res`.

2. Open the `fmrweb.res` file present in the path `ORACLE_HOME/FORMS` and add the following entry in it:

```
27 : 0 : "Esc" : 32 : "Exit"
```

3. Ensure that you comment or delete the old entry

```
#115 : 0 : "F4" : 32 : "Exit"
```

The first number (115) might differ on different versions or platforms. When you run the Web Form and press the ESC key, then the Form exits.

# 5

# Using Oracle Forms Services with the HTTP Listener and Oracle WebLogic Server

Oracle WebLogic Server is a scalable, enterprise-ready Java EE application server. It implements the full range of Java EE technologies, and provides many more additional features such as advanced management, clustering, and Web services. It forms the core of the Oracle Fusion Middleware platform, and provides a stable framework for building scalable, highly available, and secure applications.

This chapter contains the following sections:

- Section 5.1, "About the Oracle WebLogic Managed Server"
- Section 5.2, "Working with Forms Managed Server"
- Section 5.3, "Performance/Scalability Tuning"
- Section 5.4, "Load Balancing Oracle WebLogic Server"
- Section 5.5, "Using HTTPS with the Forms Listener Servlet"
- Section 5.6, "Using an Authenticating Proxy to Run Oracle Forms Applications"
- Section 5.7, "Oracle Forms Services and SSL"
- Section 5.8, "Enabling SSL with a Load Balancing Router"

## 5.1 About the Oracle WebLogic Managed Server

Managed Servers host business applications, application components, Web services, and their associated resources. To optimize performance, managed servers maintain a read-only copy of the domain's configuration document. When a managed server starts up, it connects to the domain's administration server to synchronize its configuration document with the document that the administration server maintains.

Oracle Fusion Middleware system components (such as SOA, WebCenter, and Identity Management components), as well as customer-deployed applications, are deployed to managed servers in the domain.

During configuration, some managed servers are created specifically to host the Oracle Fusion Middleware system components (for example, wls_soa, wls_portal, and wls_forms).

Figure 5–1 shows a simple scenario of the Oracle WebLogic Managed Server. In the left side of the image, the Forms servlet renders the start HTML file and provides the information about the Forms Listener servlet to the client. An HTTP request is then received by the Oracle HTTP Server Listener, which passes it off to the Forms Listener servlet running inside Oracle WebLogic Managed Server, in the right side of the

image. The Forms Listener servlet establishes a runtime process and is responsible for on-going communication between the client browser and the runtime process. As more users request Oracle Forms sessions, the requests are received by the Oracle HTTP Server Listener. The HTTP Listener again passes them off to the Forms Listener servlet, which establishes more runtime processes. The Forms Listener servlet can handle many Forms runtime sessions simultaneously. While there is, of course, a limit to the number of concurrent users, the architecture presents a number of opportunities for tuning and configuration to achieve better performance (see the next section).

*Figure 5–1  Oracle WebLogic Managed Server and Forms Services*
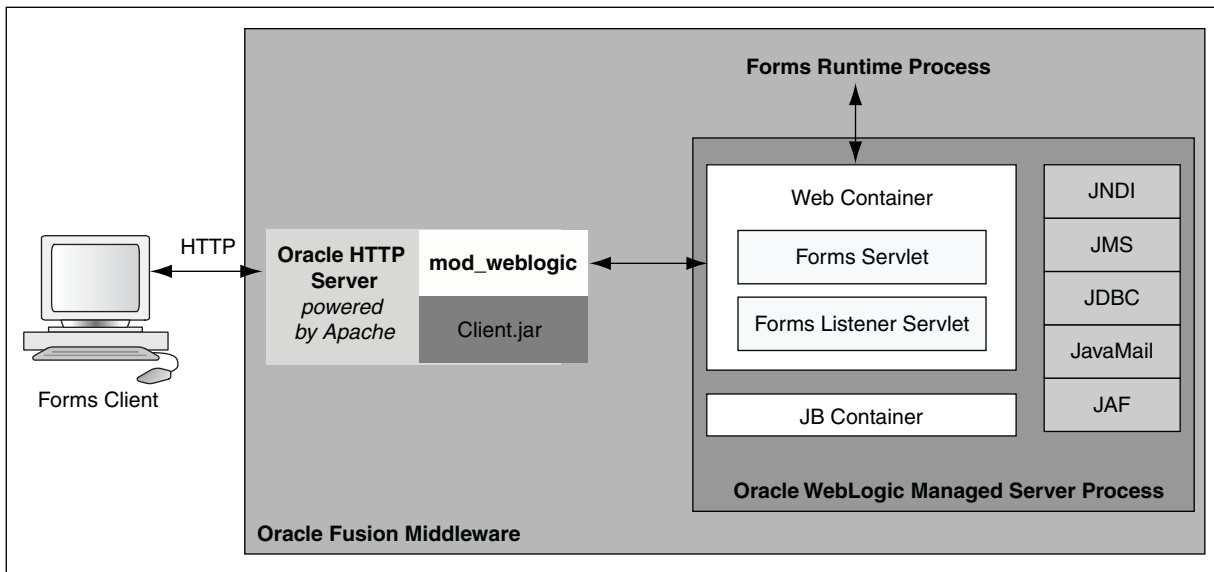


## 5.2 Working with Forms Managed Server

By default (out-of-the-box installation), the Forms Services Java EE application (`formsapp.ear`) is deployed on Forms Managed Server (`WLS_FORMS`). You can manage `WLS_FORMS` and `formsapp.ear` using Oracle WebLogic Administration Console or Oracle Enterprise Manager Fusion Middleware Control. Refer to the following topics for more information:

- Starting and Stopping Forms Managed Server: For more information, refer to "Overview of Starting and Stopping Procedures" in *Oracle Fusion Middleware Administrator's Guide*.

- Deploying Forms Application to Forms Managed Server: For more information, refer to "Install an Enterprise application" in WebLogic Administration Console Online Help. For information on deploying, undeploying, and redeploying applications, see "Deploying Applications" in *Oracle Fusion Middleware Administrator's Guide*.

- Custom deployment of Forms Java EE application: For more information, refer to Section 5.2.1, "Custom Deployment of Forms Java EE Application".

- Expanding Forms Managed Server Clusters: For more information, refer to Section 5.2.2, "Expanding Forms Managed Server Clusters".

- Managing Cloned Managed Servers: For more information on using Fusion Middleware Control to manage cloned managed servers, see Section 5.2.3, "Registering Forms Java EE Applications."

- Modifying `weblogic.xml`, `web.xml`, `application.xml` and `weblogic-application.xml` post deployment: For more information, refer to Section 5.2.4, "Modification of Forms J2EE Application Deployment Descriptors".

- Starting Forms Managed Server as a Windows Service: For more information, refer to "Setting Up a WebLogic Server Instance as a Windows Service" in *Oracle Fusion Middleware Managing Server Startup and Shutdown for Oracle WebLogic Server*.

## 5.2.1 Custom Deployment of Forms Java EE Application

To create a custom managed server and deploy Forms application on it, perform the following steps:

### 5.2.1.1 Prerequisite Steps

1. Set the following environment variables to the paths specified:

   - `MW_HOME`: Set this variable to point to the Oracle Middleware Home location (for more information, see "A.9 Specify Installation Location Screen" in *Oracle Fusion Middleware Installation Guide for Oracle Portal, Forms, Reports and Discoverer*).

   - `ORACLE_HOME`: Set this variable with the absolute path of the Oracle Home directory. For more information, see "A.9 Specify Installation Location Screen" in *Oracle Fusion Middleware Installation Guide for Oracle Portal, Forms, Reports and Discoverer*.

   - `DOMAIN_HOME`: Set this variable with the location of the folder created by Oracle WebLogic Server for the domain specified in "A.7 Select Domain Screen" in Oracle Fusion Middleware Installation Guide for Oracle Portal, Forms, Reports and Discoverer.

2. Specify the JDK path in the system path.

   Enter the path to the Java executable. For example on UNIX operating systems, enter`$MW_HOME/jdk<version>/bin` in the system path (on Windows operating systems, the path is `%MW_HOME%\jdk<version>\bin`).

3. Create a managed server, for example, `WLS_FORMS_CUSTOM_APP`, as part of the same cluster as the default managed server (`WLS_FORMS`).

   For more information on adding a managed server, refer to "Adding Additional Managed Servers to a Domain" in *Oracle Fusion Middleware Administrator's Guide*.

4. Specify the following properties of the managed server using the WebLogic Administration Console.

   - Classpath: Specify the value: `<ORACLE_HOME>/opmn/lib/optic.jar` (on Windows operating systems: `<ORACLE_HOME>\opmn\lib\optic.jar`). Replace `<ORACLE_HOME>` with the absolute path.

   - Arguments: Specify the following values:

     ```
     Dclassic.oracle.home=<ORACLE_HOME> -
     Doracle.instance=<ORACLE_INSTANCE> -
     Doracle.instance.name=<ORACLE_INSTANCE_NAME> -Doracle.forms.weblogic=1
     ```

     Make sure all the entries are in a single line (without any carriage returns). Replace *<ORACLE_HOME>*, *<ORACLE_INSTANCE>* with the absolute paths. Replace *<ORACLE_INSTANCE_NAME>* with the name of the Oracle Instance (default name asinst_1).

For more information, refer to "Server Start" in *Oracle WebLogic Administration Console Help*.

5. Perform the following steps to create a folder structure in `ORACLE_HOME`:

   a. On UNIX operating systems, create a new folder for the custom application.

      For example, create `customapp` as follows:

      ```
      mkdir -p $ORACLE_HOME/customapp
      ```

      Create a Java folder in customapp and create a symbolic link for the folder as follows:

      For example:

      ```
      cd $ORACLE_HOME/customapp

      ln -s $ORACLE_HOME/forms/java $ORACLE_HOME/customapp/java
      ```

      Copy the application files to the new folder.

      For example:

      ```
      cp -rpf $ORACLE_HOME/forms/j2ee $ORACLE_HOME/customapp/
      ```

   b. On Windows operating systems, use the following commands to create a folder structure under ORACLE_HOME directory:

      ```
      mkdir %ORACLE_HOME%\customapp\java

      mkdir %ORACLE_HOME%\customapp\j2ee

      cd %ORACLE_HOME%\customapp

      xcopy /S /E %ORACLE_HOME%\forms\java %ORACLE_
      HOME%\customapp\java

      xcopy /S /E %ORACLE_HOME%\forms\j2ee %ORACLE_
      HOME%\customapp\j2ee
      ```

### 5.2.1.2 Override the Default Servlet Alias and the Context Root

1. Extract the EAR file.

   For example, on UNIX operating systems:

   ```
   cd $ORACLE_HOME/customapp/j2ee

   jar xvf formsapp.ear
   ```

   On Windows operating systems:

   ```
   cd %ORACLE_HOME%\customapp\j2ee

   jar xvf formsapp.ear
   ```

2. Extract the WAR file.

   For example, on UNIX operating systems:

   ```
   mkdir -p $ORACLE_HOME/customapp/j2ee/warfile

   cd $ORACLE_HOME/customapp/j2ee/warfile

   jar xvf $ORACLE_HOME/customapp/j2ee/formsweb.war
   ```

   On Windows operating systems:

   ```
   mkdir %ORACLE_HOME%\customapp\j2ee\warfile
   ```

```
cd %ORACLE_HOME%\customapp\j2ee\warfile
```

```
jar xvf %ORACLE_HOME%\customapp\j2ee\formsweb.war
```

3. Override the servlet alias in web.xml deployment descriptor that is located in the WEB-INF folder.

   For example, on UNIX operating systems:

   ```
   cd $ORACLE_HOME/customapp/j2ee/warfile/WEB-INF
   ```

   On Windows operating systems:

   ```
   cd %ORACLE_HOME%\customapp\j2ee\warfile\WEB-INF
   ```

   Edit web.xml in an editor and replace frmservlet with customservlet (entries under tags <Servlet-Name>, <url-pattern>, <welcome-file>).

4. Repackage the WAR file.

   For example, on UNIX operating systems:

   ```
   cd $ORACLE_HOME/customapp/j2ee/warfile
   ```

   ```
   jar cvfM formsweb.war ./*
   ```

   ```
   mv formsweb.war $ORACLE_HOME/customapp/j2ee/
   ```

   On Windows operating systems:

   ```
   cd %ORACLE_HOME%\customapp\j2ee\warfile
   ```

   ```
   jar cvfM formsweb.war .\*
   ```

   ```
   copy formsweb.war %ORACLE_HOME%\customapp\j2ee\
   ```

   ```
   del formsweb.war
   ```

5. Override the application context root in application.xml deployment descriptor that is located in the META-INF folder.

   For example, on UNIX operating systems:

   ```
   cd $ORACLE_HOME/customapp/j2ee/META-INF
   ```

   On Windows operating systems:

   ```
   cd %ORACLE_HOME%\customapp\j2ee\META-INF
   ```

   Edit application.xml, change context-root to customapp.

6. Modify the codebase and serverURL entries in formsweb.cfg.

   For example, on UNIX operating systems:

   ```
   cd $ORACLE_HOME/customapp/j2ee/config
   ```

   On Windows operating systems:

   ```
   cd %ORACLE_HOME%\customapp\j2ee\config
   ```

   Edit `formsweb.cfg` and change the context-root entries in serverURL and codebase parameters.

   For example,

   Change `serverURL=/forms/lservlet` to `serverURL=/customapp/lservlet`.

   Change codebase from `/forms/java` to `customapp/java`.

7. Repackage the EAR file.

For example, on UNIX operating systems:

```
cd $ORACLE_HOME/customapp/j2ee
```

```
jar cvfM customapp.ear META-INF/MANIFEST.MF APP-INF/*
config/* formsweb.war META-INF/*
```

On Windows operating systems:

```
cd %ORACLE_HOME%\customapp\j2ee
```

```
jar cvfM customapp.ear META-INF\MANIFEST.MF APP-INF\*
config\* formsweb.war META-INF\*
```

8. Clean the extracted EAR file contents. On UNIX operating systems:

```
rm -rf META-INF APP-INF config META-INF formsweb.war
```

On Windows operating systems:

```
RMDIR META-INF APP-INF config META-INF /s /q
```

```
DEL formsweb.war
```

### 5.2.1.3 Create the Deployment Plan

1. Create a folder in customapp named 11.1.2.

   For example, on UNIX operating systems:

   ```
   mkdir -p $DOMAIN_HOME/deploymentplans/customapp/11.1.2
   ```

   On Windows operating systems,

   ```
   mkdir %DOMAIN_HOME%\deploymentplans\customapp\11.1.2
   ```

2. Copy the following entries to a file `$DOMAIN_HOME/deploymentplans/customapp/11.1.2/plan.xml` (on Windows operating systems, `%DOMAIN_HOME%\deploymentplans\customapp\11.1.2\plan.xml`). Example 5–1 describes a deployment plan with application name of `customapp` and managed server name of `WLS_FORMS_CUSTOM_APP`. Ensure you make the following changes:

   - Replace the custom application name, location of EAR file, and managed server with the names and locations in your environment.

   - Replace *<DOMAIN_HOME>*, *<ORACLE_HOME>* with the absolute paths.

***Example 5–1   Example of Deployment Plan***

```
<?xml version='1.0' encoding='UTF-8'?>
<deployment-plan xmlns="http://xmlns.oracle.com/weblogic/deployment-plan"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xsi:schemaLocation="http://xmlns.oracle.com/weblogic/deployment-plan
 http://xmlns.oracle.com/weblogic/deployment-plan/1.0/deployment-plan.xsd"
 global-variables="false">
  <application-name>customapp</application-name>
  <variable-definition>
    <variable>
      <name>vd-<ORACLE_HOME>/customapp</name>
      <value><ORACLE_HOME>/customapp</value>
    </variable>
    <variable>
      <name>vd-<DOMAIN_HOME>/config/fmwconfig/servers/WLS_FORMS_CUSTOM_
APP/applications/customapp_11.1.2/config/customapp</name>
```

```
        <value><DOMAIN_HOME>/config/fmwconfig/servers/WLS_FORMS_CUSTOM_
APP/applications/customapp_11.1.2/config/customapp</value>
      </variable>
    </variable-definition>
  <module-override>
    <module-name>customapp.ear</module-name>
    <module-type>ear</module-type>
    <module-descriptor external="false">
      <root-element>weblogic-application</root-element>
      <uri>META-INF/weblogic-application.xml</uri>
    </module-descriptor>
    <module-descriptor external="false">
      <root-element>application</root-element>
      <uri>META-INF/application.xml</uri>
    </module-descriptor>
    <module-descriptor external="true">
      <root-element>wldf-resource</root-element>
      <uri>META-INF/weblogic-diagnostics.xml</uri>
    </module-descriptor>
  </module-override>
  <module-override>
    <module-name>formsweb.war</module-name>
    <module-type>war</module-type>
    <module-descriptor external="false">
      <root-element>weblogic-web-app</root-element>
      <uri>WEB-INF/weblogic.xml</uri>
      <variable-assignment>
        <name>vd-<ORACLE_HOME>/customapp</name>

<xpath>/weblogic-web-app/virtual-directory-mapping/[url-pattern="java/*"]/l
ocal-path</xpath>
      </variable-assignment>
      <variable-assignment>
        <name>vd-<ORACLE_HOME>/customapp</name>

<xpath>/weblogic-web-app/virtual-directory-mapping/[url-pattern="webutil/*"
]/local-path</xpath>
      </variable-assignment>
      <variable-assignment>
        <name>vd-<DOMAIN_HOME>/config/fmwconfig/servers/WLS_FORMS_CUSTOM_
APP/applications/customapp_11.1.2/config/customapp</name>

<xpath>/weblogic-web-app/virtual-directory-mapping/[url-pattern="registry/*
"]/local-path</xpath>
      </variable-assignment>
    </module-descriptor>
    <module-descriptor external="false">
      <root-element>web-app</root-element>
      <uri>WEB-INF/web.xml</uri>
    </module-descriptor>
  </module-override>
</deployment-plan>
```

### 5.2.1.4 Deploy the Custom EAR file

Deploy the custom EAR file using WebLogic Scripting Tool (WLST) commands. For example, on UNIX operating systems:

```
$MW_HOME/oracle_common/common/bin/wlst.sh
```

On Windows operating systems: `%MW_HOME%\oracle_ common\common\bin\wlst.cmd`

Use the WLST `deploy` command to deploy the application:

```
wls:/offline> connect('weblogic','welcome1')

wls:/ClassicDomain/serverConfig> deploy('customapp', '<ORACLE_
HOME>/customapp/j2ee/customapp.ear',  'WLS_FORMS_CUSTOM_APP',
'nostage','<DOMAIN_
HOME>/deploymentplans/customapp/11.1.2/plan.xml')
```

Be sure to make the following changes in the command:

- Replace customapp with actual context root.

- Replace *<DOMAIN_HOME>*, *<ORACLE_HOME>* with the absolute paths.

### 5.2.1.5 Post-Patching Tasks

After applying Oracle Fusion Middleware 11*g*R2 Patch Sets, perform the following steps for custom deployments:

1. Copy the Forms J2EE application files to the customapp directory. On Unix operating systems:

   ```
   cp -rpf $ORACLE_HOME/forms/j2ee/* $ORACLE_
   HOME/customapp/j2ee/*
   ```

   On Windows operating systems:

   ```
   xcopy /S /E %ORACLE_HOME%\forms\java %ORACLE_
   HOME%\customapp\java
   ```

   ```
   xcopy /S /E %ORACLE_HOME%\forms\j2ee %ORACLE_
   HOME%\customapp\j2ee
   ```

2. Repeat the steps in "Override the Default Servlet Alias and the Context Root".

3. Restart the custom managed server.

### 5.2.1.6 Test the Custom Deployment

Test the deployment using the URL: `http://<Host>:<Port Number>/<context root>/<servlet name>`.

For the example in this section, the URL would be `http://<Host>:<Port Number>/customapp/customservlet`.

## 5.2.2 Expanding Forms Managed Server Clusters

To improve the scalability and performance of Forms deployments on high-end machines (multiprocessor and high-memory configuration machines), expand the Forms Managed Server cluster (`cluster_forms`). Perform the following manual steps to expand the Forms Managed Server cluster:

1. Perform the following steps to add a new Managed Server to the cluster (`cluster_forms`):

   a. Using the Oracle WebLogic Server Administration Console, you can choose to either clone the default Forms Managed Server (`WLS_FORMS`) or create a new Managed Server (for example, `WLS_FORMS_1`, with port number 9010).

For more information on using Fusion Middleware Control to manage the new or cloned managed server, see Section 5.2.3, "Registering Forms Java EE Applications".

    **b.** In the Server Properties page, add the newly created Managed Server to the Forms cluster `cluster_forms`.

    **c.** In the General Tab, assign a port number to the Managed Server.

    **d.** Assign a machine to the Managed Server.

**2.** Perform the following steps to edit the configuration of the new managed server:

    **a.** Using the Oracle WebLogic Server Administration Console, in the Server Start Tab, set the following Server Start properties.

    **b.** Add the following system properties without any carriage returns to the arguments:

```
-Dclassic.oracle.home=<ORACLE_HOME
location>-Doracle.instance=<ORACLE_INSTANCE
location>-Doracle.instance.name=<ORACLE_INSTANCE Name>

-Doracle.forms.weblogic=1
```

    **c.** Add the following to the CLASSPATH: `<ORACLE_HOME>/opmn/lib/optic.jar:<FMW_HOME>/oracle_common/modules/oracle.ldap_11.1.1/ldapjclnt11.jar:<FMW_HOME>/oracle_common/jlib/rcucommon.jar`

**3.** Activate the changes and start the new Managed Server.

**4.** Add the new Managed Server's host and port information to the WebLogicCluster entry in `forms.conf`.

```
<Location /forms>

    SetHandler weblogic-handler

    WebLogicCluster <HostName>:9001, <HostName>:9010

    DynamicServerList OFF

</Location>
```

**5.** Restart OHS.

## 5.2.3 Registering Forms Java EE Applications

To use Fusion Middleware Control to manage the new or cloned managed servers under the default Forms WLS cluster, you register the Forms Java EE applications.

Perform the following steps to register the Forms Java EE applications:

**1.** Create a sample WLST script as shown in Example 5–2. In this example, the script is named formsappRegistration.py.

***Example 5–2   Sample WLST Script***

```
#
# formsappRegistration.py
# Workaround script to register/unregister Forms J2EE application Mbean
# as a member of Forms System Component Mbean
#
from javax.management import ObjectName, Attribute
from jarray import array
```

```
import getopt, sys
#
# function prints the usage
#
def usage():
    message =
"--------------------------------------------------------------------------------"
 + \
                "\nUsage : " + \
                "\n $FMW_HOME/oracle_common/common/bin/wlst.sh " + sys.argv[0] + "
 --adminServerName=<admin server name> --asinstName=<Oracle Instance name>
 --managedServer=<newly added Forms managed server name>  --formsappName=<forms
 J2EE application name> -o <option> " + \
                "\n \n valid options - registerApp or unregisterApp" + \
                "\n \nexamples:" + \
                "\n  $FMW_HOME/oracle_common/common/bin/wlst.sh " + sys.argv[0] +"
 --adminServerName=AdminServer --asinstName=asinst_1 --managedServer=WLS_FORMS1
 --formsappName=formsapp -o registerApp " + \
                "\n $FMW_HOME/oracle_common/common/bin/wlst.sh " + sys.argv[0] + "
 --adminServerName=AdminServer --asinstName=asinst_1 --managedServer=WLS_FORMS1
 --formsappName=formsapp -o unregisterApp " + \

"\n------------------------------------------------------------------------------
-"
    print message
#
# getFormsCompMbeanObjectName - function to generate the Forms System Component
#                                                   Mbean ObjectName.
#
def getFormsCompMbeanObjectName(asInstName, adminServerName):
    frmCompONameString = "oracle.as.management.mbeans.register:" \
                        + "Location="+ adminServerName +
",type=SystemComponent,name=/" \
                        + asinstName +"/forms,instance=" + asinstName \
                        + ",component=forms,EMTargetType=oracle_forms";
    print frmCompONameString
    frmCompOName = ObjectName(frmCompONameString)
    return frmCompOName
#
# getFormsAppMbeanObjectName - function to generate the Forms J2EE application
#                                                   ObjectName.
#
def getFormsAppMbeanObjectName(appName, managedServer):
    frmappONameString = "com.bea:Name="+formsappName+ "#11.1.2,Location=" +\
                        managedServer+ ",Type=AppDeployment"
    frmappOName = ObjectName(frmappONameString)
    return frmappOName
#
# doesMemberExist - utility function to check if app is already registered as a
 member
#
def doesMemberExist(member, list):
    for item in list:
        if item == member:
            return 1
    return None
#
#  registerFormsApp - registers Forms J2EE application Mbean as a member of
#                          Forms System Component Mbean
#
```

```
def registerFormsApp(formsCompMbean, frmappMbean):
    domainRuntime()
    membersArray = mbs.getAttribute(formsCompMbean,"Members")
    membersList = membersArray.tolist()

    if membersList == []:
        print "Members list is empty"
    else:
        print "Members list is not empty"

        if doesMemberExist(frmappMbean, membersList):
            print "Member already registered, skipping registration"
        else:
            print "Member is not found, append it to the members list"
            membersList.append(frmappMbean)
            membersArray = array(membersList, ObjectName)
            membersAttrib = Attribute("Members",membersArray)
            mbs.setAttribute(formsCompMbean, membersAttrib)
#
#  unregisterFormsApp - unregisters Forms J2EE application Mbean as a member of
#                             Forms System Component Mbean
#
def unregisterFormsApp(formsCompMbean, frmappMbean):
    domainRuntime()
    membersArray = mbs.getAttribute(formsCompMbean,"Members")
    membersList = membersArray.tolist()

    if membersList == []:
        print "Members list is empty"
    else:
        print "Members list is not empty"

        if doesMemberExist(frmappMbean, membersList):
            print "Found the Member, removing it."
            membersList.remove(frmappMbean)
            membersArray = array(membersList, ObjectName)
            membersAttrib = Attribute("Members",membersArray)
            mbs.setAttribute(formsCompMbean, membersAttrib)
        else:
            print "Member not found, skipping unregister"
#
# execution starts here
#

if len(sys.argv) != 7 :
    print "invalid arguments passed to the script"
    usage()
    sys.exit(0)

# trim the first argument which is the name of the script

args = sys.argv[1:7]
optlist, args = getopt.getopt(args,'o', [
'adminServerName=','asinstName=','managedServer=','formsappName='])
options = dict(optlist)

adminServerName = options["--adminServerName"]
asinstName = options["--asinstName"]
managedServer = options["--managedServer"]
formsappName  = options["--formsappName"]
```

Using Oracle Forms Services with the HTTP Listener and Oracle WebLogic Server   **5-11**

```
if adminServerName == [] or \
   managedServer == [] or formsappName == [] or not args:
    print "invalid arguments passed to the script "
    usage()
    sys.exit(0)

argument = args[0]
print "enter the WLST connection paramters ..."
connect()

frmcompMbean = getFormsCompMbeanObjectName(asinstName,adminServerName)
print frmcompMbean

frmappMbean = getFormsAppMbeanObjectName(formsappName,managedServer)
print frmappMbean

if argument == "registerApp":
    print "registering Forms J2EE application " + formsappName
    registerFormsApp(frmcompMbean,frmappMbean)
elif argument == "unregisterApp":
    print "unregistering Forms J2EE application " + formsappName
    unregisterFormsApp(frmcompMbean,frmappMbean)

else:
    print "invalid option passed to the scripts ..."
    usage()

disconnect()
print "done... "
```

**2.** Execute the script. You can use the help argument for more information as shown in Example 5–3.

### Example 5–3   Sample Script Execution

```
$FMW_HOME/oracle_common/common/bin/wlst.sh formsappRegistration.py help


--------------------------------------------------------------------------------
Usage :
 $FMW_HOME/oracle_common/common/bin/wlst.sh formsappRegistration.py
 --adminServerName=<admin server name> --asinstName=<Oracle Instance name>
 --managedServer=<newly added Forms managed server name>  --formsappName=<forms
 J2EE application name> -o <option>

 valid options - registerApp or unregisterApp

examples:
  $FMW_HOME/oracle_common/common/bin/wlst.sh formsappRegistration.py
 --adminServerName=AdminServer --asinstName=asinst_1 --managedServer=WLS_FORMS1
 --formsappName=formsapp -o registerApp
 $FMW_HOME/oracle_common/common/bin/wlst.sh formsappRegistration.py
 --adminServerName=AdminServer --asinstName=asinst_1 --managedServer=WLS_FORMS1
 --formsappName=formsapp -o unregisterApp
--------------------------------------------------------------------------------
```

**3.** When prompted, enter the administration server username, password, and connection information.

**4.** Accept the default server URL.

**5.** Example 5–4 shows a sample of the execution and results of server registration.

*Example 5–4   Sample Execution and Results*

```
$FMW_HOME/oracle_common/common/bin ( ) -> ./wlst.sh formsappRegistration.py -
adminServerName=AdminServer --asinstName=asinst_1 --managedServer=WLS_FORMS1 -
formsappName=formsapp -o registerApp

CLASSPATH=. . . . . .
. . . . . . . . . . .
. . . . . . . . . . .
Your environment has been set.

Initializing WebLogic Scripting Tool (WLST) ...

Welcome to WebLogic Server Administration Scripting Shell

Type help() for help on available commands

enter the WLST connection paramters ...
Please enter your username :weblogic
Please enter your password :
Please enter your server URL [t3://localhost:7001] :
Connecting to t3://localhost:7001 with userid weblogic ...
Successfully connected to Admin Server 'AdminServer' that belongs to domain
'ClassicDomain'.

Warning: An insecure protocol was used to connect to the
server. To ensure on-the-wire security, the SSL port or
Admin port should be used instead.

registering Forms J2EE application formsapp
Location changed to domainRuntime tree. This is a read-only tree with DomainMBean
as the root.
For more help, use help(domainRuntime)

Members list is not empty
Member is not found, append it to the members list
Disconnected from weblogic server: AdminServer
done...
```

## 5.2.4  Modification of Forms J2EE Application Deployment Descriptors

Post-deployment, Forms J2EE application deployment descriptors (`weblogic.xml`, `web.xml`, `application.xml` and `weblogic-application.xml`) cannot be modified in Oracle WebLogic Server.

As a workaround, perform the following steps to customize the Forms J2EE application deployment descriptors and redeploy the application:

**1.** Back up the default formsapp deployment plan, `$DOMAIN_HOME/deploymentplans/formsapp/11.1.2/plan.xml`.

**2.** Add the deployment descriptors customizations to the  Forms J2EE application's deployment plan. See the "Modifying the Deployment Plan" for an example.

> **Note:**   For more information on updating the deployment plan, refer to the *Oracle Fusion Middleware Deploying Applications to Oracle WebLogic Server.*

3. Using the WebLogic Administration Console, update the forms application (redeploy) and select the option **Update this application in place with new deployment plan changes**.

4. Restart the Forms J2EE application using the WebLogic Administration Console.

**Modifying the Deployment Plan**

In this example, the deployment plan is modified to override the Forms Servlet testMode parameter and set it to true. To modify the deployment plan, perform the following steps:

1. Enter the following commands:

```
mkdir -p $CLASSIC_ORACLE_HOME/forms/j2ee/backup
cd $CLASSIC_ORACLE_HOME/forms/j2ee
cp $DOMAIN_HOME/deploymentplans/formsapp/11.1.2/plan.xml backup/
vi $DOMAIN_HOME/deploymentplans/formsapp/11.1.2/plan.xml
```

2. Modify the deployment plan. The following is a sample of the deployment plan with the added entries highlighted in bold:

```
<?xml version='1.0' encoding='UTF-8'?>
<deployment-plan xmlns="http://xmlns.oracle.com/weblogic/deployment-plan"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://xmlns.oracle.com/weblogic/deployment-plan
http://xmlns.oracle.com/weblogic/deployment-plan/1.0/deployment-plan.xsd"
global-variables="false">
  <application-name>formsapp</application-name>
  <variable-definition>
    <variable>
      <name>vd-/scratch/t_work/Oracle/Middleware/as_1/forms</name>
      <value>/scratch/t_work/Oracle/Middleware/as_1/forms</value>
    </variable>
    <variable>
      <name>vd-/scratch/t_work/Oracle/Middleware/user_
projects/domains/ClassicDomain/config/fmwconfig/servers/WLS_
FORMS/applications/formsapp_11.1.2/config/forms</name>
      <value>/scratch/t_work/Oracle/Middleware/user_
projects/domains/ClassicDomain/config/fmwconfig/servers/WLS_
FORMS/applications/formsapp_11.1.2/config/forms</value>
    </variable>
    <variable>
      <name>FormsServlet_InitParam_testMode</name>
      <value>true</value>
    </variable>
  </variable-definition>
  <module-override>
    <module-name>formsapp.ear</module-name>
    <module-type>ear</module-type>
    <module-descriptor external="false">
      <root-element>weblogic-application</root-element>
      <uri>META-INF/weblogic-application.xml</uri>
    </module-descriptor>
    <module-descriptor external="false">
      <root-element>application</root-element>
      <uri>META-INF/application.xml</uri>
    </module-descriptor>
    <module-descriptor external="true">
      <root-element>wldf-resource</root-element>
      <uri>META-INF/weblogic-diagnostics.xml</uri>
    </module-descriptor>
```

```
      </module-override>
      <module-override>
        <module-name>formsweb.war</module-name>
        <module-type>war</module-type>
        <module-descriptor external="false">
          <root-element>weblogic-web-app</root-element>
          <uri>WEB-INF/weblogic.xml</uri>
          <variable-assignment>
            <name>vd-/scratch/t_work/Oracle/Middleware/as_1/forms</name>
<xpath>/weblogic-web-app/virtual-directory-mapping/[url-pattern="java/*"]/local
-path</xpath>
          </variable-assignment>
          <variable-assignment>
            <name>vd-/scratch/t_work/Oracle/Middleware/as_1/forms</name>
<xpath>/weblogic-web-app/virtual-directory-mapping/[url-pattern="webutil/*"]/lo
cal-path</xpath>
          </variable-assignment>
          <variable-assignment>
            <name>vd-/scratch/t_work/Oracle/Middleware/user_
projects/domains/ClassicDomain/config/fmwconfig/servers/WLS_
FORMS/applications/formsapp_11.1.2/config/forms</name>
<xpath>/weblogic-web-app/virtual-directory-mapping/[url-pattern="registry/*"]/l
ocal-path</xpath>
          </variable-assignment>
        </module-descriptor>
        <module-descriptor external="false">
          <root-element>web-app</root-element>
          <uri>WEB-INF/web.xml</uri>
          <variable-assignment>
            <name>FormsServlet_InitParam_testMode</name>
<xpath>/web-app/servlet/[servlet-name="frmservlet"]/init-param/[param-name="tes
tMode"]/param-value</xpath>
          </variable-assignment>
        </module-descriptor>
      </module-override>
</deployment-plan>
```

**3.** Restart the Forms J2EE application using the WebLogic Administration Console.

## 5.3 Performance/Scalability Tuning

The steps for tuning the Forms Listener servlet are similar to steps for tuning any high throughput servlet application. You have to take into account resource management and user needs for optimal tuning of your particular Forms Services configuration. For more information, see *Oracle Fusion Middleware Performance Guide* available on OTN at http://www.oracle.com/technetwork/indexes/documentation/index.ht ml.

### 5.3.1 Limit the number of HTTPD processes

To control spawning HTTPD processes (which is memory consuming) set the KeepAlive directive in the Oracle HTTP Listener configuration file (httpd.conf):
KeepAlive Off

KeepAlive specifies whether or not to allow persistent connections (more than one request per connection). If you must use KeepAlive On, for example, for another application, make sure that KeepAliveTimeout is set to a low number for example, 15 seconds, which is the default. The KeepAlive setting is used to maintain a persistent

connection between the client (Browser) and the OHS server. It does not have anything to do with the OHS to Oracle WebLogic Server connection.

### 5.3.2 Set the MaxClients Directive to a High value

You can let the HTTP Listener determine when to create more HTTPD processes. Therefore, set the `MaxClients` directive to a high value in the configuration file (`httpd.conf`). However, you need to consider the memory available on the system when setting this parameter.

`MaxClients=256` indicates that the listener can create up to 256 HTTPD processes to handle concurrent requests.

If your HTTP requests come in bursts, and you want to reduce the time to start the necessary HTTPD processes, you can set `MinSpareServers` and `MaxSpareServers` (in `httpd.conf`) to have an appropriate number of processes ready. However, the default values of 5 and 10 respectively are sufficient for most sites.

## 5.4 Load Balancing Oracle WebLogic Server

The Forms Listener servlet architecture allows you to load balance the system using any of the standard HTTP load balancing techniques available.

The Oracle HTTP Server Listener provides a load balancing mechanism that allows you to run multiple WebLogic instances on the same host as the HTTP process, on multiple, different hosts, or on any combination of hosts. The HTTP Listener then routes HTTP requests to Oracle WebLogic Managed Server instances.

The following scenarios are just a few of the possible combinations available and are intended to show you some of the possibilities. The best choice for your site will depend on many factors.

For a complete description of this feature, refer to the *Oracle Fusion Middleware Performance Guide* (available on OTN at http://www.oracle.com/technetwork/indexes/documentation/index.html).

The following images illustrate four possible deployment scenarios:

- Figure 5–2 shows the Oracle HTTP Server balancing incoming requests between multiple Oracle WebLogic Managed Servers  on the same host as the Oracle HTTP Listener.

- Figure 5–3 shows the Oracle HTTP Server balancing incoming requests between multiple Oracle WebLogic Managed Servers on a different host to the Oracle HTTP Listener.

- Figure 5–4: shows the Oracle HTTP Server balancing incoming requests between multiple Oracle WebLogic Managed Servers on multiple different hosts and multiple different hosts each running an Oracle HTTP Listener.

- Figure 5–5: shows the Oracle HTTP Server balancing incoming requests between multiple Oracle WebLogic Managed Servers on a single host but with multiple different hosts each running an Oracle HTTP Listener.

*Figure 5–2   Multiple Oracle WebLogic Servers on the same host as the Oracle HTTP Listener*
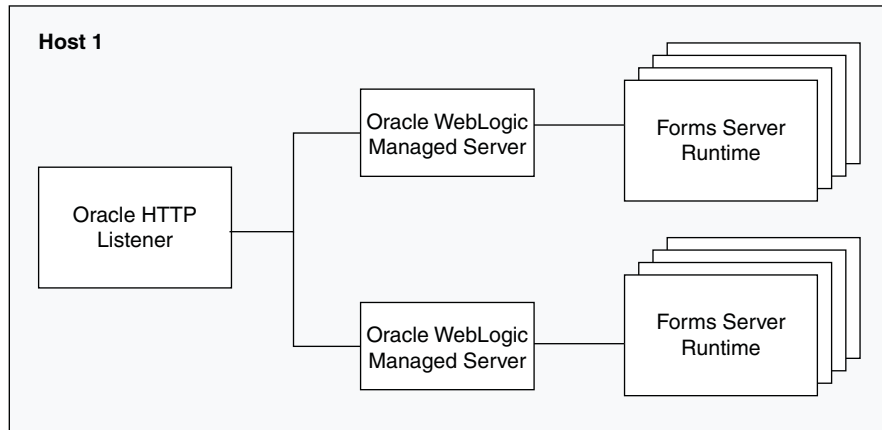


*Figure 5–3   Multiple Oracle WebLogic Servers on a different host to the Oracle HTTP Listener*
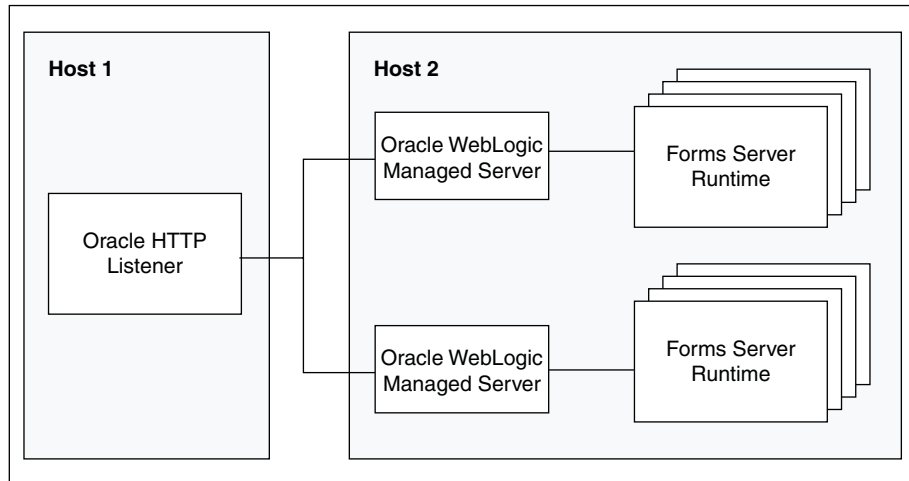
**Figure 5–4   Multiple Oracle WebLogic Servers and multiple Oracle HTTP Listeners on different hosts**
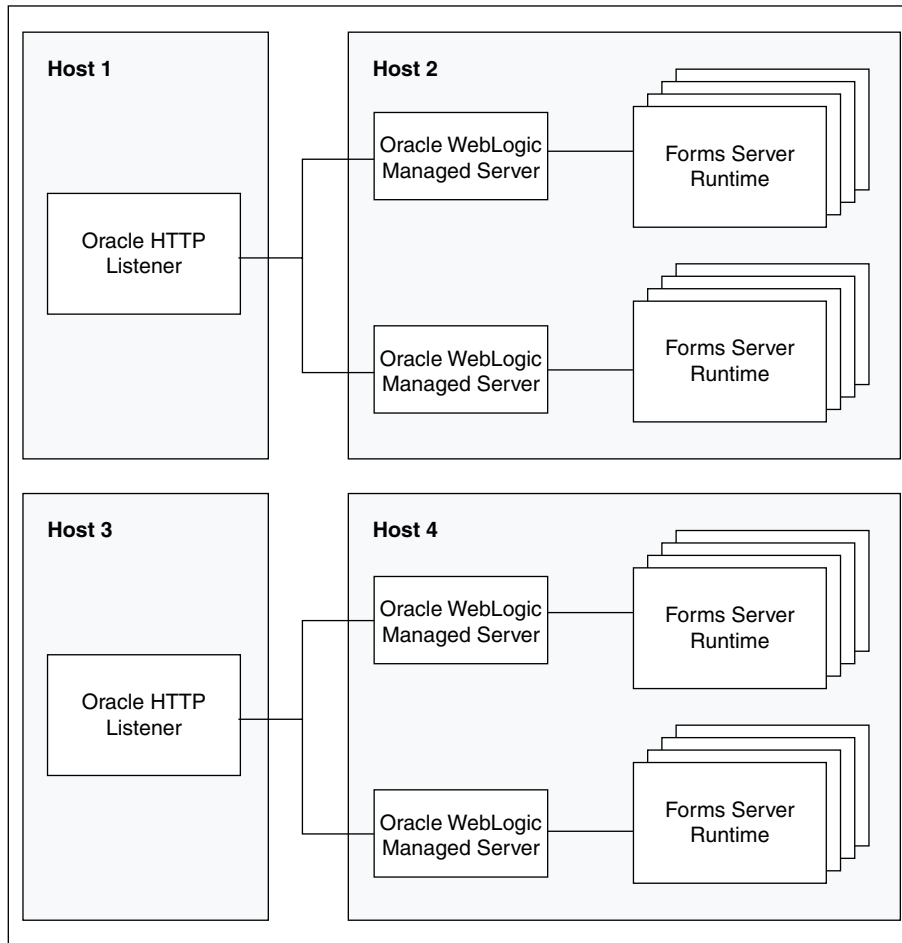
*Figure 5–5   Multiple Oracle HTTP Listeners on different hosts with multiple Oracle WebLogic Servers on one host*
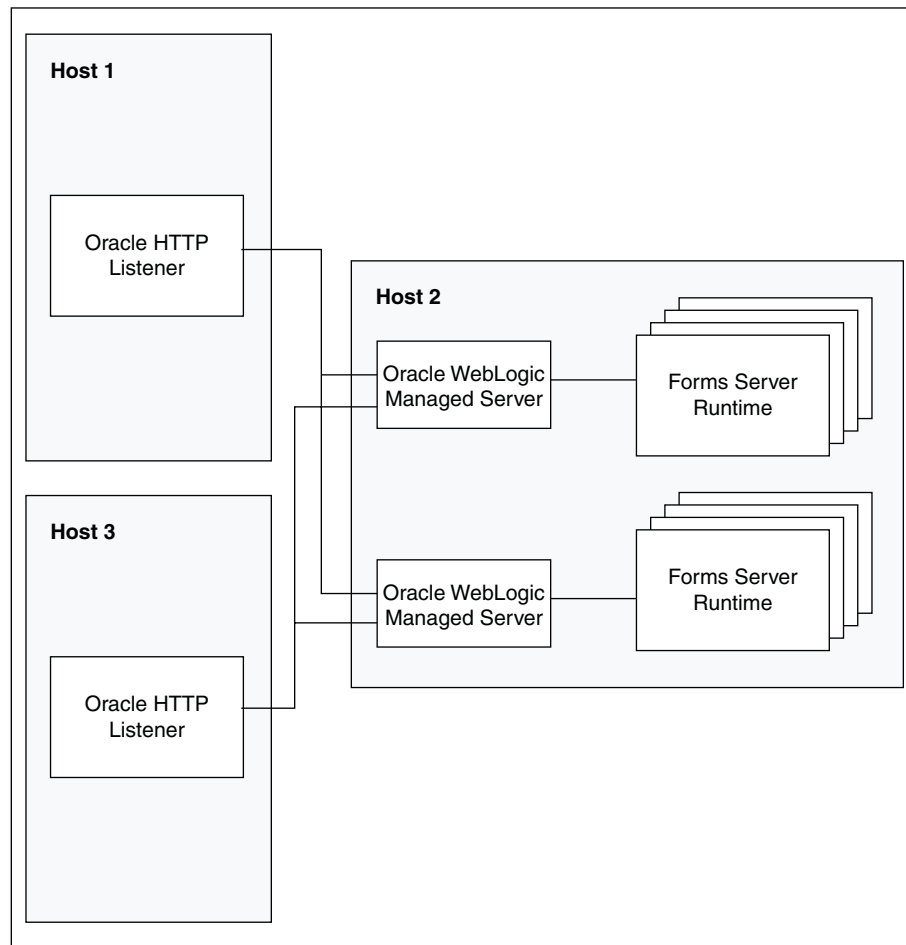


For more information about tuning and optimizing Forms Services with the HTTP Listener and Oracle WebLogic Server, see *Oracle Fusion Middleware Performance Guide*, available on Oracle Technology Network (OTN) at
http://www.oracle.com/technetwork/indexes/documentation/index.html.

## 5.5  Using HTTPS with the Forms Listener Servlet

Using HTTPS with Oracle Forms is no different than using HTTPS with any other Web-based application. HTTPS requires the use of digital certificates (for example, VeriSign). Because Forms Services servlets are accessed via your Web server, you do not need to purchase special certificates for communications between the Oracle Forms client and the server. You only need to purchase a certificate for your Web server from a recognized certificate authority.

## 5.6  Using an Authenticating Proxy to Run Oracle Forms Applications

The default configuration as set up by the Oracle Fusion Middleware installation process supports authenticating proxies. An authenticating proxy is one that requires the user to supply a username and password in order to access the destination server where the application is running. Typically, authenticating proxies set a cookie to

detect whether the user has logged on (or been authenticated). The cookie is sent in all subsequent network requests to avoid further logon prompts.

The codebase and server URL values that are set up by the Oracle WebLogic Server installation process include `$ORACLE_HOME/forms/java` and `/forms/lservlet`. As these are under the document base of the page (`$ORACLE_HOME/forms`), authenticating proxies will work.

## 5.7 Oracle Forms Services and SSL

To run Oracle Forms Services applications in SSL mode:

- Create a Wallet to manage certificates.

- Enable the HTTPS port in Oracle HTTP Server. By default, Oracle HTTP Server has one SSL Port enabled (8890).

- Enable Web Cache to accept HTTPS connections from Oracle HTTP Server.

For more information on the above topics, see the section "SSL Configuration in Oracle Fusion Middleware" in the *Oracle Fusion Middleware Administrator's Guide*.

> **Note:** When you change the Oracle Web Cache port using Enterprise Manager, regenerate the `osso.conf` and copy the generated `osso.conf` file to `$ORACLE_INSTANCE/config/OHS/<OHS_INSTANCE>/moduleconf` directory. Restart the Oracle HTTP Server and Oracle Web Cache for the changes to take effect.

## 5.8 Enabling SSL with a Load Balancing Router

Running a Forms application that uses an HTTPS port requires a certificate to be imported. If Oracle Forms is behind a load balancing router, and SSL terminates at it, you need to import the certificate from the load balancing router.

**To enable SSL with your Forms applications over a load balancing router:**

1. Start a Web browser and enter the Forms application HTTPS URL containing the fully qualified host name (including port number if required) used by your own Oracle installation. For example:
   `https://example.com:443/forms/frmservlet`

   The Security Alert dialog box is displayed.

2. Click **View Certificate**.

3. Click the **Details** tab in the Certificate dialog.

4. Click **Copy to File...**

5. In the Welcome page of the Certificate Export Wizard, click **Next**.

6. In the Export File Format page, select **Base-64 encoded X.509 (.CER)**, then click Next.

7. Enter a file name such as `c:\temp\forms`, then click Next.

8. Click **Finish**.

   A message appears saying that the export was successful.

9. Click **OK**.

10. Close the Certificate Export Wizard, but keep the Security Alert dialog open.

11. Import the security certificate file that you saved earlier into the certificate store of the JVM you are using. For more information, see the next section.

12. At the Security Alert dialog, click Yes to accept the security certificate and start the Forms application.

**Importing the certificate into Java Plugin**

1. On the client machine, open the Control Panel.

2. Open Java.

3. Navigate to Securities tab.

4. Click Certificate.

5. Import the certificate that was exported in the previous section.

6. Click Apply.

# 6

# Oracle Forms and JavaScript Integration

This chapter contains the following sections:

## 6.1 About Oracle Forms Calling External Events

In previous releases of Oracle Forms, you had to implement OLE and DDE to interact with a limited number of event types outside of Forms. In later versions, Forms offered `web.show_document` and Java integration to interface with external application sources. But in terms of calling out to the Web page where Forms is displayed, there was no easy solution. It was also not possible to call from the Web page into Forms, perhaps to update a value acquired from an HTML form.

In Oracle Forms 11*g*, JavaScript integration provides the ability to have JavaScript events call into Forms, or have Forms execute JavaScript events. Figure 6–1 shows how JavaScript and Oracle Forms work together. In the left side of the image, JavaScript is executed in the page in which the Forms applet is hosted. Oracle Forms now has the capability to call JavaScript functions using native built-ins. Also, JavaScript functions can now trigger a Oracle Forms trigger by using a new API that has been provided.

*Figure 6–1   Oracle Forms and JavaScript*
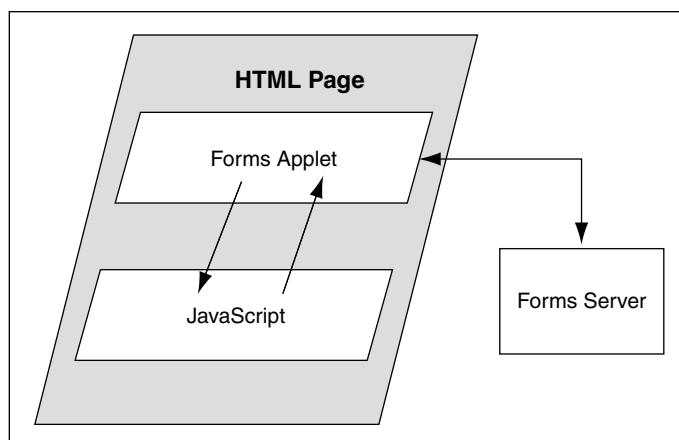
Two new calls are available in the `web` Built-in package:

- `web.javascript_eval_expr`

- `web.javascript_eval_function`

The first call `web.javascript_eval_expr` is a procedure which takes two arguments: an expression and a target, both of data type `varchar2`. This legal JavaScript expression is interpreted in the Web page in which the Forms applet is embedded. The expression can be a call to a function that is defined in the target page or any valid JavaScript expression that can be executed on the target page, for example, `document.bgColor='red'`. The expression is executed, using LiveConnect's JSObject.eval() method, in the context of the page or frame that is named in the target argument. If the target argument is null, then it is executed in the page or frame in which the Forms applet is embedded.

The second call, `web.javascript_eval_function` is a function and returns a varchar2 value. Both `web.javascript_eval_expr` and `web.javascript_eval_function` have the same functionality except that `javascript_eval_expr` does not send any return value from the Forms client to the Forms Services. If your application does not need a return value, use `web.javascript_eval_expr`. The additional network trip that is required to carry the return value from the Forms client to the Forms Services is eliminated.

To set the value of an HTML text item with the ID `outside_field_id` to the value of the Forms field called `inside`, you could write this PL/SQL code:

```
web.javascript_eval_expr('
document.getElementById("outside_field_id").value='
||:inside
);
```

Note that the PL/SQL string must use single quotes while JavaScript is flexible enough to use single or double quotes. Using double quotes inside the expression works without having to use escape sequences. You could also write a function in the Web page:

```
<SCRIPT>
    function set_field(field_id, myvalue){
        document.getElementById(field_id).value=myvalue;
    };
</SCRIPT>
```

To get the value of the outside field and assign it to the inside field, you could write the following PL/SQL code:

```
:inside:=web.javascript_eval_function('
    document.getElementById("outside_field_id").value
    ');
```

### 6.1.1 Why Call Events Outside of Oracle Forms?

In Oracle Forms 11*g*, the newly added JavaScript functionality allows you to integrate Forms with HTML-based application technologies in the Web browser. For example you can use JavaScript integration when the Forms-based application is required to integrate on the page with new functionality based on an HTML front end.

## 6.2 About JavaScript Events Calling into Oracle Forms

You can also allow JavaScript calls into Oracle Forms by using JavaScript in the Web page that hosts the Forms applet. There is new functionality available on the embedded Forms object in the DOM (Document Object Model) tree. You use JavaScript to do:

```
document.forms_applet.raiseEvent(event_name, payload);
```

The assumption here is that you have set the ID configuration variable to `forms_applet`.

When the surrounding Web page executes this JavaScript code, Oracle Forms fires a new type of trigger called `WHEN-CUSTOM-JAVASCRIPT-EVENT`. In this trigger there are only two valid system variables: `system.javascript_event_value` and `system.javascript_event_name`. These variables contain the payload and event name that were passed into Forms through the `raiseEvent` method. On calling the `raiseEvent` method, a trigger named `WHEN-CUSTOM-JAVASCRIPT-EVENT` is fired on the server side.

```
declare
    event_val varchar2(300):= :system.javascript_event_value;
begin
    if (:system.javascript_event_name='show') then
        handleShowEvent(event_val);
    elsif(:system.javascript_event_name='grab') then
  handleGrabEvent(event_val);
    else
        null;
    end if;
end;
```

This PL/SQL code recognizes two events: 'show' and 'grab'. Any other name is ignored.

### 6.2.1 Why Let Events Call into Oracle Forms?

You can synchronize an HTML based application, whether it is Java-based or otherwise, with a Forms-based application in the same hosting Web page. For example, you can use the HTML-based application to query data and use Forms to update it if, and only if, the user has the correct access privileges.

## 6.3 Integrating JavaScript and Oracle Forms

This section describes an example for integrating JavaScript in Oracle Forms application. To integrate JavaScript in Oracle Forms applications, perform the following steps:

1. Build a Forms application using the JavaScript events as described in Section 6.1, "About Oracle Forms Calling External Events" and Section 6.2, "About JavaScript Events Calling into Oracle Forms". Use the `:system.javascript_event_name` and `:system.javascript_event_value` in the `WHEN-CUSTOM-JAVASCRIPT-EVENT` trigger. Compile the module. For more information, refer to the Forms Builder Online Help.

2. Create an html file (for example, `test.html`) that the Forms servlet will use as a template when generating the HTML page used to start an Oracle Forms

application. Copy the file to the Forms configuration directory: `$ORACLE_INSTANCE/config/FormsComponent/forms/server`

3. Copy any required images, html files, JavaScript files, and css files to the following directory: $DOMAIN_HOME/servers/WLS_FORMS/tmp/_WL_user/formsapp_11.1.2/<random_string2>/war/

4. Create an html file that uses the JavaScripts (for example, `js.html`) and invokes the servlet URL.

5. Using Enterprise Manager, create a new configuration section or modify an existing one and enable `enableJavascriptEvent`. Set baseHTMLjpi to `test.html`.

6. Using Enterprise Manager, edit the `default.env` file and add the directory where you saved the forms application to the environment variable `FORMS_PATH`.

7. Run the application by using the URL in your browser: `http://<localhost>:9001/forms/js.html`

# 6.4  Configuration of formsweb.cfg

The administrator of the Forms application can enable or disable JavaScript integration by setting the parameter `enableJavascriptEvent` in `formsweb.cfg` to "true" or "false". If `enableJavascriptEvent` is not set to true, then calls from JavaScript would be ignored. The applet_name parameter must be set to the value that is used by the HTML developer to reference the forms applet via document.<applet_name>.

The administrator can also set `JavaScriptBlocksHeartBeat` (default value is false) in `formsweb.cfg` to true. This blocks Form's `HEARTBEAT` during the time JavaScript is executed. If the JavaScript calls complete execution before the FORMS_TIMEOUT period, setting JavaScriptBlocksHeartBeat to true provides an increase in performance by avoiding additional network messages.

Note that if JavaScriptBlocksHeartBeat is set to true, Forms would abnormally terminate if the time taken for executing a JavaScript is more than FORMS_TIMEOUT.

# 6.5  Configuration of Environment Variables

An environment variable called `FORMS_ALLOW_JAVASCRIPT_EVENTS` in `default.env` is also used to enable or disable JavaScript integration. By default, the value of the variable is true. If this is set to false, then JavaScript integration is not enabled for any Forms aplication that uses that instance of `default.env`, no matter what value is set for `enableJavascriptEvent` in `formsweb.cfg`.

# 7

# Enhanced Java Support

This chapter contains the following sections:

- Section 7.1, "Overview"
- Section 7.2, "About Custom Item Event Triggers"

## 7.1 Overview

Oracle Forms provides Java classes that define the appearance and behavior of standard user interface components such as buttons, text areas, radio groups, list items, and so on. A Forms pluggable Java component (PJC) can be thought of as an extension of the default Forms client component. When you create a PJC, you write your own Java code to extend the functionality of any of the provided default classes.

### 7.1.1 Dispatching Events from Forms Developer

In addition to extending the standard Forms user interface components, you can also create a PJC that includes Java Swing user interface components in your form. A pluggable Java component extends a class provided by Forms, that is, `oracle.forms.ui.VBean`, and lives in the Bean Area as seen on the Forms canvas. The Bean Area does not have its own user interface, but rather is a container. On the layout editor or on a canvas, you see only an empty rectangle until you associate an implementation class with it and add some user interface components.

In earlier releases of Oracle Forms, Forms user interface components implemented the IView interface. However, it did not have any special method to add or remove CustomListener from the pluggable Java component or the view. In Oracle Forms 11*g*, you can add or remove CustomListener in the IView interface.

### 7.1.2 Dispatching Events to Forms Services

Oracle Forms 11*g* makes it easier to dispatch CustomEvent along with parameters and payloads. Since JavaBean classes do this by exposing the public method `dispatchCustomEvent`, you need to add the same method for your PJC. You call the `dispatchCustomEvent` method from the PJC to dispatch the CustomEvent.

Since CustomEvent is usually associated with parameters, Forms provides a way to add them. In a JavaBean, you can use the `getHandler().setProperty()` method to set the parameters. Users must be able to do the same for PJC. For more information, see Section 7.2.2, "About the Custom Item Event Trigger at Runtime".

## 7.2  About Custom Item Event Triggers

In Oracle Forms 11*g*, you can add the WHEN-CUSTOM-ITEM-EVENT trigger to items at design time and code the pluggable Java components so that the trigger can be fired at runtime. This trigger fires whenever a JavaBean custom component in the form causes the occurrence of an event. You can use a WHEN-CUSTOM-ITEM-EVENT trigger to respond to a selection or change of value of a custom component. The system variable SYSTEM.CUSTOM_ITEM_EVENT_PARAMETERS stores a parameter name that contains the supplementary arguments for an event that is fired by a custom control. Control event names are case sensitive.

### 7.2.1  Adding the When-Custom-Item-Event Trigger at Design Time

The most common way of adding a trigger to an item is by clicking the Create button in the Object Navigator toolbar in Oracle Forms Developer, while the focus is on the Trigger node, or by pressing the corresponding shortcut key. Forms Developer presents to you a list of available triggers at that level or for that item.

Another way of adding some of the commonly used triggers is by right-clicking the trigger node of the item in the Object Navigator. Then, select one of the triggers listed in the smart Triggers menu.

For more information on working with triggers, see the Oracle Forms Developer online help.

### 7.2.2  About the Custom Item Event Trigger at Runtime

In Oracle Forms 11*g*, pluggable Java components can raise the WHEN-CUSTOM-ITEM-EVENT trigger. This enhanced trigger provides greater control over the content of the communication between the client and server.

The Forms client dispatches CustomEvent through the pluggable Java component, which fires the WHEN-CUSTOM-ITEM-EVENT trigger on the Forms Services. The WHEN-CUSTOM-ITEM-EVENT trigger provides a simple way to retrieve the event name and parameter values that are passed from the client pluggable Java component through CustomEvent. The event name is stored in SYSTEM.CUSTOM_ITEM_EVENT; parameters (name and value) are stored in SYSTEM.CUSTOM_ITEM_EVENT_PARAMETERS.

The Forms Built-in get_parameter_attr is used to retrieve the values and different parameters from SYSTEM.CUSTOM_ITEM_EVENT_PARAMETERS. The supported datatype for the values or payloads that are returned from get_parameter_attr is a VARCHAR2 string.

### 7.2.3  Example: A Java class for a Push Button

In this example, a Java class is created for a push button that enables selecting a client file using the File Open option and returns the path to the server.

1. Create a Java class for a push button with simple PJC code such as:

```
 // MyButtonPJC.java
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import javax.swing.JFileChooser;
import oracle.forms.ui.CustomEvent;
import oracle.forms.ui.VButton;
import oracle.forms.properties.ID;
public class MyButtonPJC extends VButton implements ActionListener
```

```
 {
   private static final ID CLIENT_SELECTED_FILE = ID.registerProperty("CLIENT_
 SELECTED_FILE");
   public MyButtonPJC()
   {
     addActionListener(this);
   }
   public void actionPerformed(ActionEvent event)
   {
     JFileChooser fc = new JFileChooser();
     if(fc.showOpenDialog(getHandler().getApplet()) == JFileChooser.APPROVE_
 OPTION)
     {
       CustomEvent ce = new CustomEvent(getHandler(), "MyButtonPJC_Event");
       ce.setProperty(CLIENT_SELECTED_FILE,
 fc.getSelectedFile().getAbsolutePath());
       this.dispatchCustomEvent(ce);
     }
   }
   public void destroy()
   {
     removeActionListener(this);
     super.destroy();
   }
 }
```

2. Ensure CLASSPATH variable is defined in the environment and `$ORACLE_HOME/forms/java/frmall.jar` is added to it.

3. Compile the Java class. For ease of creating the jar later, place the output class files in a separate directory by using the `-d <output-directory>` option of the javac (java compiler).

4. Navigate to the output directory and create a jar file, for example, MyButtonPJC.jar, containing the generated class files by using the command

   `jar cvf <jar-file-path> *`

5. MyButtonPJC.jar needs to be signed before deploying in Forms applet. You can use `sign_webutil.sh` (sign_webutil.bat in Windows) that is available in the directory `$ORACLE_INSTANCE\bin` to sign the jar file. For more information, see Forms Builder Online Help.

6. Copy MyButtonPJC.jar to `$ORACLE_HOME/forms/java` directory.

7. Add the path of MyButtonPJC.jar to the FORMS_BUILDER_CLASSPATH. This makes the class files in that jar available in Forms Builder.

8. Add the push button on the layout in the Forms application.

9. In Property Palette of the push button, set MyButtonPJC as the implementation class.

10. Add `WHEN-CUSTOM-ITEM-EVENT` trigger to the push button.

11. Add the following PL/SQL code to the `WHEN-CUSTOM-ITEM-EVENT` trigger of the push button. This code handles the CustomEvent dispatched by the PJC and then extracts the parameters in the event.

```
 declare
    filePath VARCHAR2(1024);
    dataType      PLS_INTEGER;
 begin
    Message('Custom Event Name='||:SYSTEM.CUSTOM_ITEM_EVENT);
```

```
    get_parameter_attr(:SYSTEM.CUSTOM_ITEM_EVENT_PARAMETERS,'CLIENT_SELECTED_
FILE',dataType, filePath);
    Message('The selected client file path is '|| filePath);
  end;
```

**12.** Add MyButtonPJC.jar to the list of comma-separated jars (only jar file name, not the full path) in the `archive` parameter in Forms configuration file (`formsweb.cfg`). This ensures that the jar file is loaded in Forms applet on the client side.

# 8

# Working with Server Events

This chapter contains the following:

## 8.1 About Oracle Forms and Server Events

With the exception of timers, most events in Oracle Forms occur from some kind of user interaction. In previous versions of Oracle Forms, there was no easy support to receive an external event if it could not be bound to the Form's graphical user interface. Forms clients had to use techniques such as polling through a great deal of coding to respond to these events to deal with external events that it did not initiate.

With Oracle Forms 11*g* and Oracle Database, you can handle external events, such as asynchronous events, by using the database queue. Note that in order to work with database queues in Oracle Forms 11*g* you must be using Oracle Database 10*g* Release 2 or later. Oracle Streams Advanced Queuing (AQ), an asynchronous queuing feature, enables messages to be exchanged between different programs. AQ functionality is implemented by using interfaces such as DBMS_AQ, DBMS_AQADM, and DBMS_AQELM, which are PL/SQL packages. For more information about Advanced Queuing, see the *Oracle Streams Advanced Queuing User's Guide* at http://www.oracle.com/technetwork/indexes/documentation/index.html.

In general, the steps required to integrate events and database queues are:

**Database**

- Create a queue table: Define the administration and access privileges (AQ_ADMINISTRATOR_ROLE, AQ_USER_ROLE) for a user to set up advanced queuing. Define the object type for the payload and the payload of a message that uses the object type. Using the payload, define the queue table.

- Create a queue: Define the queue for the queue table. A queue table can hold multiple queues with the same payload type.

- Start the queue: Enable enqueue/dequeue on the queue.

- Enqueue a message: Write messages to the queue using the DBMS_AQ.ENQUEUE procedure.

**Forms Builder**

- Create an event object: Create a new event in the Events node in the Object Navigator in the Forms Builder.

- Subscribe the event object to the queue: The name of the queue is specified in the Subscription Name property.

- Code necessary notification: Write the event handling function, which is queued up for execution by Forms and is executed when the server receives a request from the client. Write the trigger code for the When-Event-Raised trigger that is attached to the Event node.

**Forms Services**

- Run the form and register the subscription

- Invoke the When-Event-Raised trigger upon event notification

In earlier versions of Forms, handling external events was only possible through custom programming, usually done in Java with the help of Forms' Java Bean support. In Oracle Forms 11*g* it is possible to call into Forms from any technology that can interface with Advanced Queuing (AQ), for example Java Messaging (JMS).

Figure 8–1 shows the flow of events that take advantage of the improved integration of the different components your application might work with. In the left side of the image, the Oracle Forms has two-way communication with the AQ functionality of Oracle Database. In the center of the image, the AQ function of Oracle Database also has two-way communication with the possible outside events that can trigger internal Forms events. In the right side of the image, these external events can include technologies such as files with dynamic content, Web services, mail, JMS, or database content that interact with BPEL processes which in turn interact with AQ. BPEL, however, is not necessary. JMS, as an example, can interact with AQ directly without having to go through BPEL.

> **Note:** Third party tools such as antivirus and security software may prevent Advanced Queuing from working correctly with Oracle Forms. As a workaround, turn off any third party security tools.

*Figure 8–1 Oracle Forms Handles Outside Events with Advanced Queueing in Oracle Database*
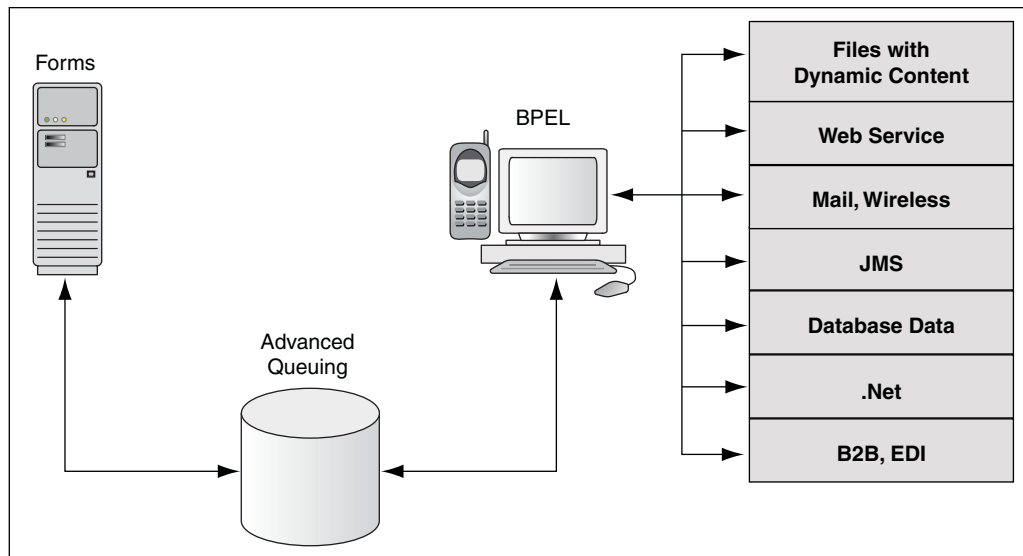


## 8.2 Creating Events

Oracle Forms Developer provides a declarative environment for creating and managing event objects. For known external events, Forms Developer provides a list of available events that can be subscribed to. The property of the event object can be set at runtime or at design time. The ability to end a subscription to a particular external event is also provided through a dynamic setting of the event object property.

Most of the new event functionality is also available through standard Oracle interfaces. Both client and server-side PL/SQL provide all the necessary functionality to create, subscribe, and publish a database event. Oracle Forms provides a declarative and user-friendly way of registering a database event. Oracle Forms provides a standard way of responding to the event by hiding most of the complexity from end-users.

## 8.3 Subscribing to Events

The Forms Services gets notified when events it has registered interest in are added to the event queue. Registration is done either when the runtime starts up or when connecting to the database, depending on the type of the event. For database events, the type of the event queue (persistent or non-persistent) is also saved as part of the event creation.

## 8.4 Event Propagation

Figure 8–2 shows a situation where a Forms client is idle. Since Oracle Forms is driven by the HTTP protocol, which is a request/response protocol only, nothing can change on the client if the client is idle. A new applet property MaxEventWait, expressed in milliseconds, governs how long the application should wait before checking for an event. In other words, you can specify how often the client should send a request to the server, thus causing the execution of the PL/SQL that is specified as a response to an event.

Note, however, that, on the server-side, Forms Services receives all the events without polling. However, the server does not start running the WHEN_EVENT_RAISED triggers

until it receives the notification from the Forms Client (because of the HTTP request/reply paradigm of the Forms Client and hence the need for the `MaxEventWait` property).

**Figure 8–2   Notification flow with idle or active clients**
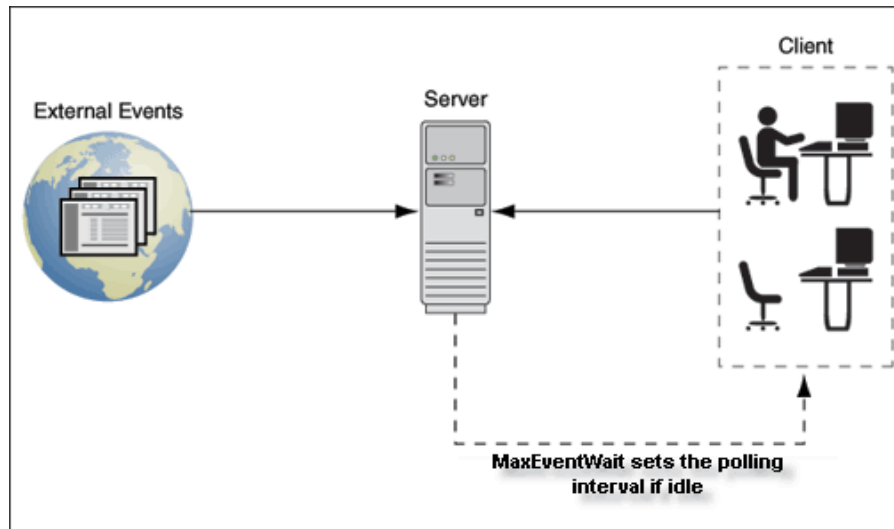


## 8.4.1  About the When-Event-Raised Trigger

Oracle Forms responds to or fires a trigger in response to a variety of events. For both Forms Developer and internal events, Forms provides entry points in terms of triggers so that an application developer can associate and execute some code in response to an event.

For example, a defined trigger is attached to a specific object in a form. The object to which a trigger is attached defines the *scope* of the trigger. For example, the `WHEN-BUTTON-PRESSED` trigger corresponds to the Button Pressed event which occurs when an operator selects a button. The name of the trigger establishes the association between the event and the trigger code. When a user clicks on a button, Forms responds by executing the code in the `WHEN-BUTTON-PRESSED` trigger.

This new event object has a corresponding trigger defined at the event object level. The `WHEN-EVENT-RAISED` trigger fires in response to the occurrence of a database event for which it has a subscription. The firing of the new trigger is similar to the internal processing of triggers. However, the source of the event is, in this case, an external event such as a database event (firing as a result of an operation) and not the result of any user interaction with forms or as a result of an internal form processing.

## 8.4.2  About Trigger Definition Level and Scope

Oracle Forms triggers are usually attached to a specific object, such as an item, block, or Form. The object to which a trigger is attached determines the trigger's *definition level* in the object hierarchy. A trigger's definition level determines the trigger's *scope*. The scope of a trigger is its domain within the Forms object hierarchy, and determines where an event must occur for the trigger to respond to it. Although the `WHEN-EVENT-RAISED` trigger is attached to an event object, it has an application level scope because of the nature of the server-centric events. When the event notification is invoked as a result of an asynchronous callback mechanism for registered database events, any number of forms running within that application and with a subscription

for that event receive the notification. This alleviates the need for the application developer to code complex logic to deal with the event.

There is also a Form-level scope so that the event will only be handled if the application is running the specific form from where the event is defined.

## 8.5 Publishing Database Events

You use the standard PL/SQL interface for publishing a database event from Forms. For example, you can publish the SalaryExceed event by calling the enqueue interface and providing all the necessary arguments. You can also call a stored procedure to perform this task.

The following program unit can be called from a WHEN-BUTTON-PRESSED trigger by passing the queue name. Depending on how you have defined the queue in the database, a commit might or might not be necessary to actually publish the event. The following sample code will not actually publish the event since there is no commit issued.

```
Declare
  msgprop       dbms_aq.message_properties_t;
  enqopt        dbms_aq.enqueue_options_t;
  enq_msgid     raw(16);
  payload       raw(10);
  correlation   varchar2(60);
begin
    payload := hextoraw('123');
    correlation := 'Jones';
    enqopt.visibility := dbms_aq.IMMEDIATE;
    msgprop.correlation := correlation;
    DBMS_AQ.ENQUEUE( queue, enqopt, msgprop, payload, enq_msgid);
end;
```

For more information about database events, see *Oracle Database PL / SQL Reference*.

## 8.6 About Application Integration Between Forms

Many enterprise applications are made of a large number of forms which are defined to perform specific tasks such as purchasing, accounting, and sales force management. These applications may also interact with other non-Forms based applications as part of performing a task. The need to provide an integration model where an enterprise can easily integrate its applications (including passing data) with those of its partners, suppliers, and distributors is extremely important.

In previous releases, Oracle Forms attempted to integrate loosely coupled applications through mechanisms ranging from using user_exit calls and some polling via timers to using pluggable Java components. These methods are all useful in some limited circumstances, but they do not provide a formal infrastructure for enterprise application integration.

Apart from the deployment concerns and performance issues, the main reason why these methods do not fully integrate applications is that the integration is only provided through Forms Developer as almost all events are bound to Forms visual components. Also, the communication with the Forms Services is always initiated by the Forms client via a request-reply model.

To provide better support for application integration, Oracle Forms 11*g* supports synchronous and asynchronous server-centric events.

### 8.6.1  About Synchronous Communication

Synchronous communication follows a request-reply paradigm, where a program sends a request to another program and waits until the reply arrives. HTTP follows this paradigm. This model of communication (also called online or connected) is suitable for programs that need to get the reply before they can proceed with their work. Traditional client-server architectures are based on this model. Earlier releases of Oracle Forms client-server architecture is also an example of this model. One of the drawbacks of the synchronous model of communication is that all the programs must be available and running for the application to work. In the event of network or machine failure, programs cease to function. For example, if the Forms Services dies, the Forms client ceases to function as well. The synchronous communication model is also in use when the Forms Services interacts with other systems such as PL/SQL or the database. The Forms system would be blocked waiting for the current operation to end before continuing with its work. Another drawback of synchronous communication is that the calling program has to wait for a response and unexpected events cannot be handled without first polling for them.

### 8.6.2  About Asynchronous Communication

Asynchronous communication is when a user or form places a request in a queue and then proceeds with its work without waiting for a reply or when an asynchronous event is received without any initial request. Programs in the role of consumers retrieve requests from the queue and act on them. This model is well-suited for applications that can continue with their work after placing a request in the queue because they are not blocked waiting for a reply. It is also suited to applications that can continue with their work until there is a message to retrieve.

Oracle Forms 11*g* supports asynchronous communication with the help of database events. A thin queuing mechanism provides the mechanism for asynchronous events. The queue is checked for messages once there are no more current operations to be performed.

For example, an application might require data to be entered or an operation executed at a later time, after specific conditions are met. The recipient program retrieves the request from the queue and acts on it.

### 8.6.3  Configuring Asynchronous Communication

Oracle Forms uses a polling technique at the application level. The client polls the server for an update after specified intervals of time. The frequency of polling can be modified using the parameters - `MaxEventWait` and `HEARTBEAT`. A higher frequency of polling may ensure that a client polls the server more frequently for updates; however, this may result in consumption of considerable resources.

The frequency value for polling is set in `formsweb.cfg`. The value assigned to this constant is in milliseconds and is a positive number.

In the absence of the configuration file setting, the current Oracle Forms `HEARTBEAT` setting is used. However, special attention and care should be made with regards setting and using of `MaxEventWait`. In a default setting where `MaxEventWait` is not set, the `HEARTBEAT` mechanism is used for polling.  The default delay when the HEARTBEAT mechanism is used is two minutes. You can set the `MaxEventWait` (which is in milliseconds) to a value smaller than the `HEARTBEAT` for faster response.

For more information on configuring these parameters using the Enterprise Manager, see  Chapter 4.2.4, "Managing Parameters".

# 9

# Using Forms Services with Oracle Single Sign-On

This chapter contains the following sections:

## 9.1 Overview

In addition to working with Oracle Single Sign-On Server 10*g* (OSSO), Oracle Forms Services applications can now run in a Single Sign-on environment using Oracle Access Manager 11*g* (OAM) and Oracle Internet Directory (OID) to eliminate the need for additional or different logins to access many applications during the same user session.

Oracle Forms Services applications in Oracle FMW 11*g* Release 2 can be protected by one of the following authentication servers:

- Oracle Access Manager (OAM) 11*g*

- Oracle Single Sign-On Server (OSSO) 10*g*

During the installation of Forms and Reports 11*g* Release 2, users can choose to authenticate their Forms Applications using one of these authentication servers. It is required that these authentication servers are configured to use Oracle Internet directory as the backend Identity Store. Authentication servers are designed to work in Web environments where multiple Web-based applications are accessible from a browser. Without an authentication server, each user must maintain a separate identity and password for each application they access. Maintaining multiple accounts and passwords for each user is unsecure and expensive.

Oracle Access Manager 11*g* is a Java Platform, Enterprise Edition (Java EE)-based enterprise-level security application that provides restricted access to confidential information and centralized authentication and authorization services. Oracle Access Manager 11*g*, a component of Oracle Fusion Middleware 11*g*, is a Single Sign-On solution for authentication and authorization.

Authentication servers enable an application to authenticate users by means of a shared authentication token or authentication authority. That means that a user authenticated for one application is automatically authenticated for all other applications within the same authentication domain.

Forms applications use a single sign-on solution only for obtaining database connection information from Oracle Internet Directory. Once the database information is obtained, interaction with the authentication server no longer occurs. Exiting a Forms application does not perform a single sign-on logout. Conversely, logging out of a single sign-on session does not terminate an active Forms session. The database session exists until the Forms Runtime (for example, frmweb.exe) on the server terminates, usually by explicitly exiting the form.

Authentication servers can be used to authenticate other applications that are not Oracle products, for example, custom-built Java EE applications.

Oracle Forms Services provides out-of-the box support for single sign-on for as many Forms applications as run by the server instance with no additional coding required in the Forms application.

> **Note:** Refer to the Section 3.4.2, "Forms Single Sign-On on Mozilla 3.x" for more information on browser support for Forms and single sign-on.

> **Note:** Oracle Forms Services applications runs in a single sign-on environment using the following OID and authentication server combinations:
>
> - Oracle Internet Directory 10*g* (10.1.2.3) with Oracle Single Sign-On 10*g* (10.1.2.3)
>
> - Oracle Internet Directory 10*g* (10.1.4.3) with Oracle Single Sign-On 10*g* (10.1.4.3)
>
> - Oracle Internet Directory 11*g* (11.1.1) with Oracle Single Sign-On 10*g* (10.1.4.3)
>
> - Oracle Internet Directory 11*g* (11.1.1) with Oracle Access Manager 11*g* (11.1.1.5)
>
> For more information about single sign-on, see *Oracle Application Server Single Sign-On Administrator's Guide*.
>
> For more information about Oracle Access Manager, see *Oracle Fusion Middleware Administrator's Guide for Oracle Access Manager*
>
> For more information about Oracle Internet Directory, see *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*.

### 9.1.1 Single Sign-On Components used by Oracle Forms

There are various Single Sign-On components in Oracle Fusion Middleware that are involved when running Forms applications in single sign-on mode with an authentication server. Figure 9–1 describes the high level overview of the various components involved in the single sign-on deployment setup of Forms Services.

**Figure 9–1   Components involved in the Single Sign-On Deployment Setup of Forms Services**
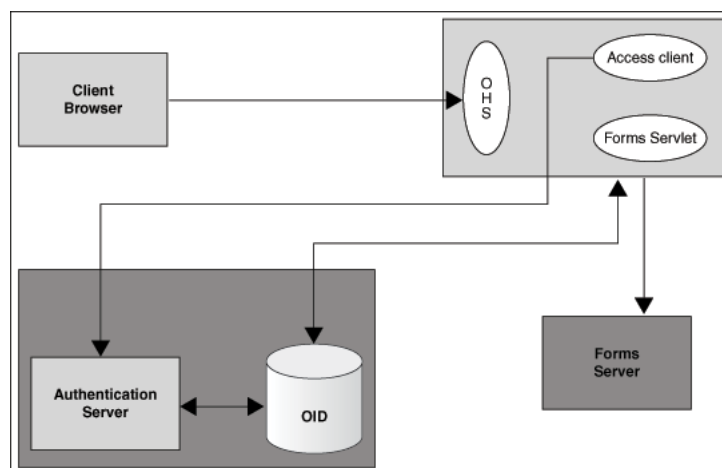


Following is the description of the components mentioned in the above figure:

- **Authentication Server**

  - Oracle Access Manager (OAM Server) - It is an Oracle FMW 11*g* authentication server that provides  a full range of security functions that include Web  single sign-on, authentication and authorization. When running Forms Services, it  uses Oracle Internet Directory as the Identity Store. Oracle Access Manager  can use either `mod_osso` or `webgate` as the access client configured with Oracle HTTP Server.

  - Oracle Single Sign-On Server (OSSO Server) - It is an OracleAS 10*g* authentication server. It uses Oracle Internet Directory as the Identity Store. Oracle Single Sign-On Server uses `mod_osso` as the access client configured with Oracle HTTP Server.

- **Access Client**

  - `webgate` - WebGate provides single sign-on support. It intercepts incoming HTTP requests and forwards them to the Access Server for authentication. Oracle Forms Services and Oracle Reports Services can use `webgate` as an access client with OAM server.

  - `mod_osso` - The HTTP module `mod_osso` simplifies the authentication process by serving as a partner application to the OAM server, rendering authentication transparent for applications. Oracle Forms Services and Oracle Reports Services can use `mod_osso` to register as partner applications with the OAM Server. `mod_osso` is also used as an access client with Oracle Single Sign-On server (OSSO).

- **Identity Store**

  - Oracle Internet Directory (OID) is an LDAP server that is used as the Identity store by the authentication server and the Forms applications. An LDAP server is a special database that is optimized for read access.

- **Forms Servlet** - The Oracle Forms Services component that accepts the initial user request to start a Forms application. The Forms servlet detects if an application requires authentication, directs the request to the authentication server and accesses the Oracle Internet Directory to obtain the database connect information.

## 9.1.2 Authentication Flow

Figure 9–2 describes the authentication flow of authentication server support in Oracle Forms the first time the user requests an application URL that is protected by authentication server:

*Figure 9–2 Authentication Flow for First Time Client Request*
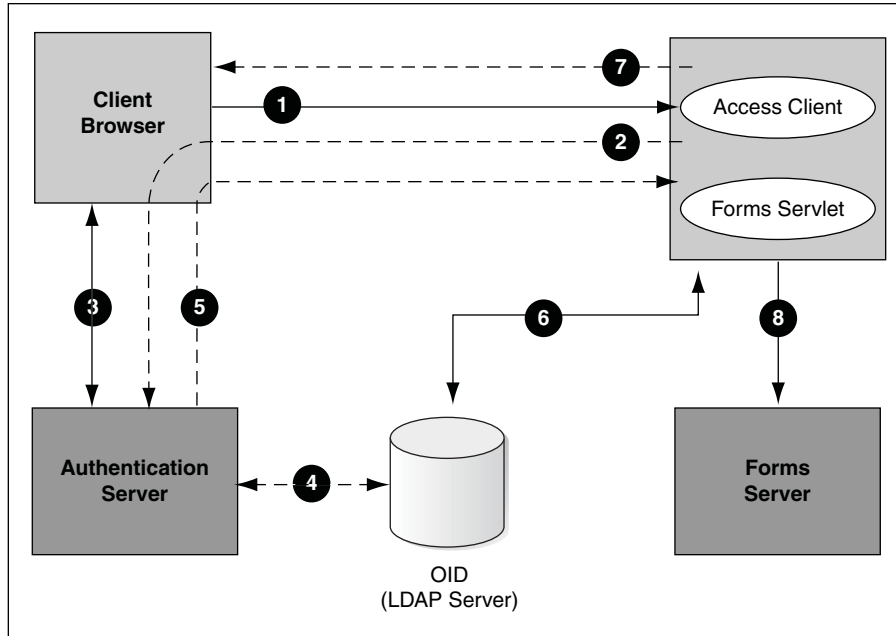


1. The user requests a Forms URL similar to
   `http(s)://<hostname>:<port>/forms/frmservlet?config=<application>&...`

   > **Note:** Use the HTTP or Web Cache port number in the Forms URL for Forms applications that use single sign-on. The Forms URL is similar to `http://<host name>:<http port>/forms/frmservlet?config=ssoapp` where ssoapp is the name of the section in forms configuration file with single sign-on (`ssoMode`) enabled.

2. The Forms servlet redirects the user to the authentication server login page.

3. The user provides user name and password through the login form.

4. The password is verified through Oracle Internet Directory (LDAP Server).

5. The user is redirected to the URL with `sso_userid` information.

6. The Forms servlet retrieves the database credentials from Oracle Internet Directory.

7. The Forms servlet sets the `sso_userid` parameter in the Runform session and permits the applet to connect to the Forms listener servlet.

8. The Forms servlet starts the Forms server.

Figure 9–3 describes the authentication flow of single sign-on support in Oracle Forms Services when a user, authenticated through another partner application, requests an application that is protected by authentication server.

*Figure 9–3 Authentication Flow for Subsequent Client Requests*
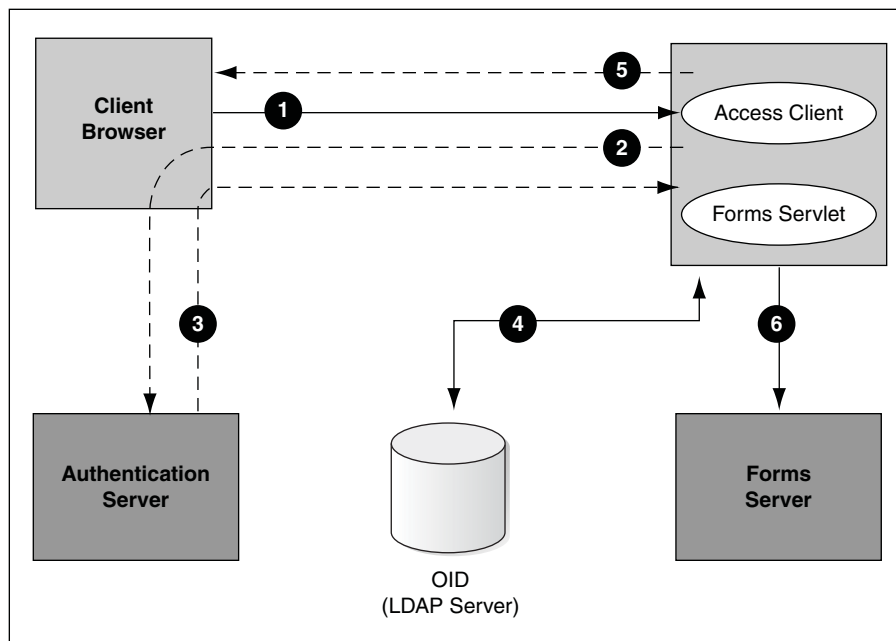


1. The user requests the Forms URL.

2. The Forms servlet redirects the user to the authentication server and its login page.

3. The user is redirected to the URL with the sso_userid information.

4. The Forms servlet retrieves the database credentials from Oracle Internet Directory.

5. The Forms servlet sets the sso_userid parameter in the Runform session and the applet connects to the Forms listener servlet.

6. The Forms servlet starts the Forms server.

## 9.2  Setup Process

The user can enable Single Sign-On for Forms application either during installation or postinstallation. This section discusses the following scenarios:

- Enabling Single Sign-On for Forms Application during Installation
- Enabling Single Sign-On for Forms Application Postinstallation

### 9.2.1  Enabling Single Sign-On for Forms Application during Installation

If the user selects Application Identity Store and an authentication server during the installation of Oracle Forms and Reports 11*g*R2, then the Forms applications will be configured to be authenticated by Oracle AS Authentication Server. The flowchart in Figure 9–4 describes the steps to enable SSO authentication for Forms applications.

*Figure 9–4   Enabling Single Sign-On for Forms Application during Installation*
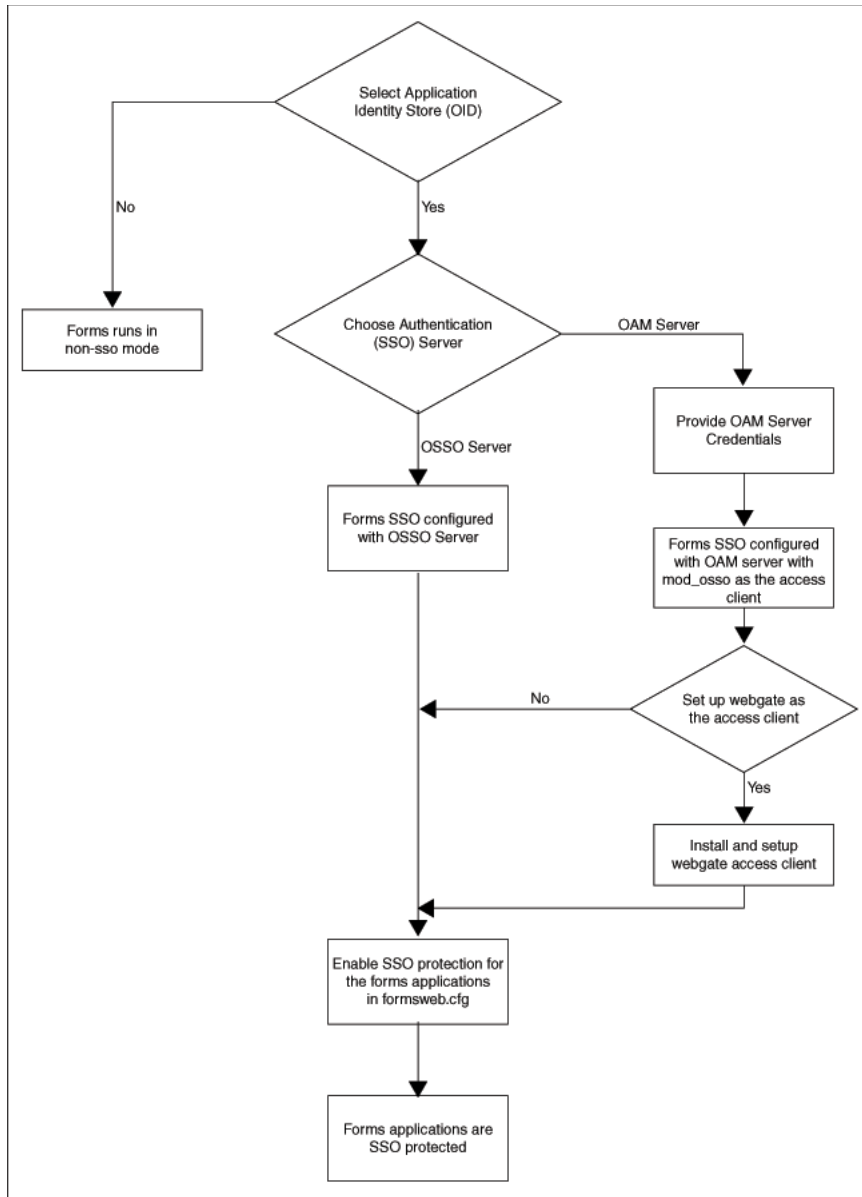


The steps depicted in the flowchart are described in details in Table 9–1:

*Table 9–1    Tasks to Enable Single Sign-On for Forms Application during installation*

| Tasks | Options | Description | Comments |
|---|---|---|---|
| **Task 1: Select an Application Identity Store (OID)** | No | User chooses not to configure Forms with Single Sign-On authentication | |
| | Yes | User chooses to configure Forms with Single Sign-On authentication. User has to provide the OID access details in the install screen. In the subsequent install screen, the user will be asked to choose the SSO server | For detailed steps for selecting an Application Identity Store, see *Flowchart of Oracle Forms and Reports Installation and Configuration Screens* in *Oracle Fusion Middleware Installation Guide for Oracle Forms and Reports*. |
| **Task2: Select an Authentication (SSO) server** | Oracle Single Sign-On Server (OSSO) | User selects Oracle AS 10*g* Oracle Single Sign On Server (OSSO) as the authentication/SSO server. No additional credentials required here | For detailed steps for Selecting an Authentication server, see *Flowchart of Oracle Forms and Reports Installation and Configuration Screens* in *Oracle Fusion Middleware Installation Guide for Oracle Forms and Reports*. |
| | OAM Server | User selects Oracle Access Manager (OAM Server) as the authentication/SSO server. User needs to provide OAM server Administrator Credentials | For detailed steps for Selecting an Authentication server, see *Flowchart of Oracle Forms and Reports Installation and Configuration Screens* in *Oracle Fusion Middleware Installation Guide for Oracle Forms and Reports*. |
| **Task3: Setup Webgate Access Client** | No | User chooses to configure Forms application with OAM authentication server in the out of the box setup. `mod_osso` is setup as the access client by default. In this case, no additional steps are required. | |
| | Yes | User chooses to configure Forms application with OAM authentication server with webgate as the access client. The user must install and configure Webgate manually. | For detailed steps for setting up Webgate Access Client, see Section 9.7.4, "Installing and Configuring Webgate with OAM". |
| **Task4: Enable SSO for Forms applications in formsweb.cfg** | This task is mandatory. | After having registered the access client with the authentication server, the user must enable SSO for Forms applications. | For detailed steps for enabling SSO for Forms applications in formsweb.cfg, see Section 9.4, "Protecting Forms applications with Single Sign-On". |

> **Note:** For more information about enabling Single Sign-On for Forms application during installation, see *Oracle Fusion Middleware Installation Guide for Oracle Forms and Reports*.

## 9.2.2 Enabling Single Sign-On for Forms Application Postinstallation

If the user does not select Application Identity store during the installation of Oracle Forms and Reports 11*g*R2, then the Forms application does not get authenticated by the authentication server. However, the user has the choice to enable single sign-on authentication for Forms application postinstallation. The flowchart in Figure 9–5 describes the steps to enable SSO for Forms application postinstallation.

*Figure 9–5   Enabling SSO for Forms Application Postinstallation*
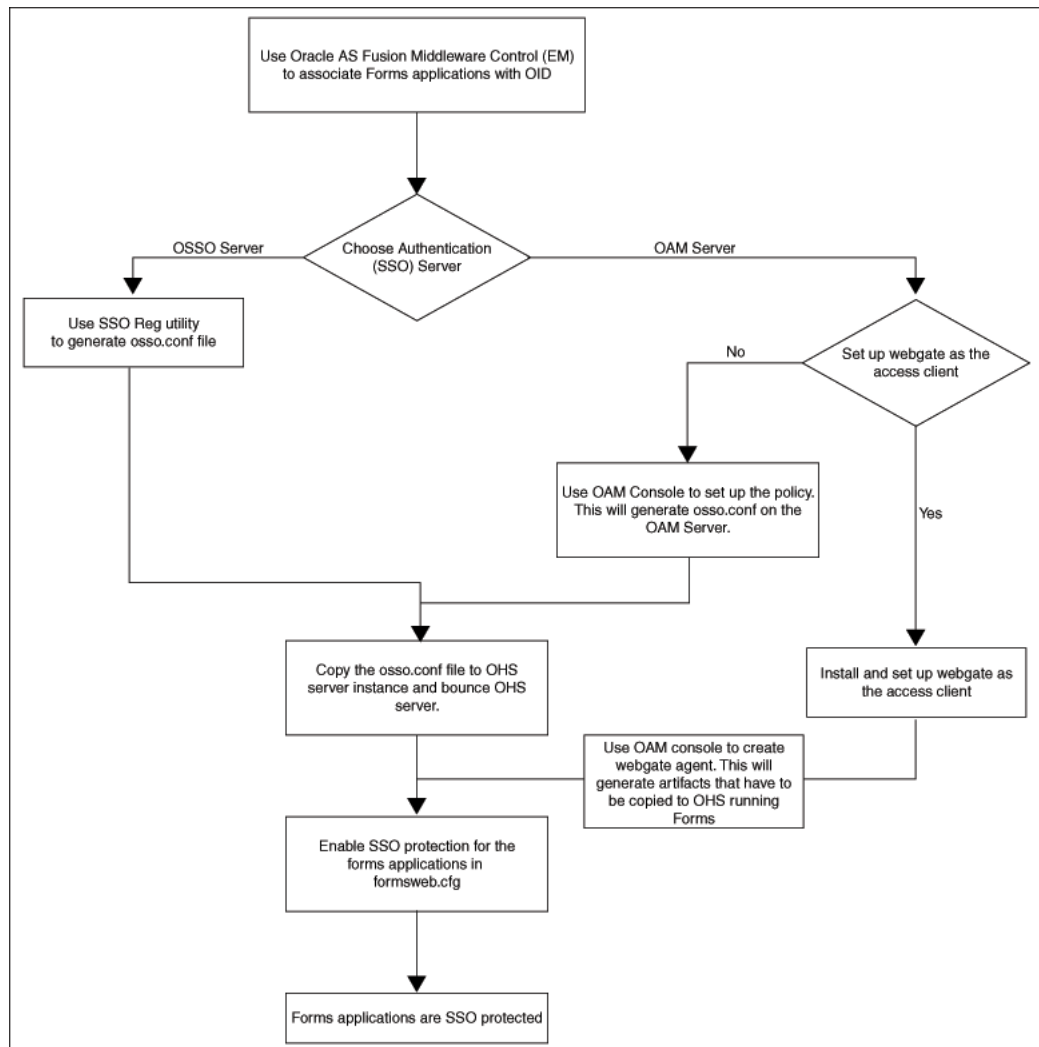


The steps depicted in the flowchart are described in details in Table 9–2:

*Table 9–2    Tasks to Enable Single Sign-On for Forms Application Postinstallation*

| Tasks | Options | Description | Comments |
|---|---|---|---|
| **Task 1: Use Fusion Middleware Control (EM) to associate Forms applications with OID** | | User chooses to associate Forms application with Oracle Internet Directory. | For detailed steps for selecting an Application Identity Store, see Section 9.7.1, "Configuring Forms J2EE application with Oracle Internet Directory". |
| **Task2: Select an Authentication (SSO) server** | Oracle Single Sign-On Server (OSSO) | User has selected Oracle AS 10*g* Oracle Single Sign On Server (OSSO) as the authentication server. | If you already have an Oracle Single Sign-On (OSSO) 10*g* server installed and running, you can use that. If not, you can installed Oracle Access Manager 11*g*.For detailed steps for installing OAM 11*g*, see *Oracle Fusion Middleware Installation Guide for Oracle Forms and Reports*. |
| | OAM Server | User has selected Oracle Access Manager (OAM Server) as the authentication server. | For detailed steps for installing OAM 11*g*, see *Oracle Fusion Middleware Installation Guide for Oracle Forms and Reports*. |
| **Task 3: Generate and apply the osso.conf file** | Oracle Single Sign-On Server (OSSO) | User has selected Oracle AS 10*g* Oracle Single Sign-On Server (OSSO) as the authentication server. User must generate the `osso.conf` file using `ssoreg.sh` from the OSSO server home. | For detailed steps for generating the `osso.conf` file on the authentication server, see Section 9.7.2, "Generating the Access Client File". |
| | OAM Server | User has selected Oracle Access Manager (OAM Server) as the authentication server. User must generate the `osso.conf` file on the OAM server using the OAM console. | |

*Table 9–2   (Cont.) Tasks to Enable Single Sign-On for Forms Application Postinstallation*

| Tasks | Options | Description | Comments |
|---|---|---|---|
| **Task 4: Set up Webgate Access Client** | No | In this case, no additional steps are required. mod_osso is set up as the access client. | |
| | Yes | The user chose to configure Forms application with OAM authentication server with webgate as the access client. The user must install and configure Webgate manually. | For detailed steps for setting up webgate access client, see Section 9.7.4, "Installing and Configuring Webgate with OAM". |
| **Task 5: Enable SSO for Forms applications in formsweb.cfg** | This task is mandatory. | After having registered the Access client with the authentication server, the user must enable SSO for Forms applications. | For detailed steps for enabling SSO for Forms applications in formsweb.cfg, see Section 9.4, "Protecting Forms applications with Single Sign-On". |

# 9.3 Forms Services Features with Authentication Server Protection

The following features and enhancements are available with this release of Oracle Forms Services:

- Section 9.3.1, "Dynamic Resource Creation"
- Section 9.3.2, "Support for Dynamic Directives"
- Section 9.3.3, "Support for Database Password Expiration"

## 9.3.1 Dynamic Resource Creation

In single-sign on mode, when a user tries to connect to a Forms application, the user is authenticated by mod_osso or webgate in combination with an authentication server and Oracle Internet Directory. Once the user is authenticated, the user is directed to the Forms servlet which takes the user's request information containing the single sign-on user name. The user name and the application name build a unique pair that identifies the user's resource information for this application in Oracle Internet Directory.

When an authorized Forms user has neither the resource for a particular application that is being requested nor a default resource in Oracle Internet Directory, then the user is redirected to DAS or the Forms RAD Servlet for the creation of the Resource Access Descriptor. After creating the resource, the user is redirected to the original Forms request URL.

The way Forms Services handles the missing resource information can be customized by the application or Forms Services administrator. The following options are available:

- Allow dynamic resource creation (default)
- Redirect the user to a pre-defined URL as specified by the ssoErrorUrl parameter
- Display the Forms error message

The redirection URL is provided by the system administrator in the Forms configuration files and should be either absolute or relative.

### 9.3.2 Support for Dynamic Directives

Enforcing single sign-on in Forms is done within the `formsweb.cfg` file. The single sign-on parameter, `ssoMode`, when set to a valid value other than `FALSE`, indicates that the application requires authentication by authentication server.

This parameter allows a Forms Services instance to handle both application types, those that rely or do not rely on single sign-on for retrieving the database password. Because single sign-on is configured in the `formsweb.cfg` file, Enterprise Manager Fusion Middleware Control can be used to manage this aspect of authentication.

### 9.3.3 Support for Database Password Expiration

In previous releases of Oracle Forms, changing a database password would be successful, but the changes (including expirations) would not propagate to Oracle Internet Directory.

In Oracle Forms Services 11*g*, if the database password has expired and the *Forms Services* application, running in single sign-on mode, is used to renew it, the new password entered by the user is used to update the Resource Access Descriptor (RAD) in Oracle Internet Directory for this application. This feature ensures that authenticating a Forms user via authentication server with Forms continues to work even when the user's database password has changed. However, if password changes are made in SQL*Plus, and not in Oracle Forms, the database connect string is not updated in Oracle Internet Directory.

## 9.4 Protecting Forms applications with Single Sign-On

Oracle Forms applications are configured using a central configuration file, the `formsweb.cfg` file in the `$DOMAIN_HOME/config/fmwconfig/servers/WLS_FORMS/applications/formsapp_11.1.2/config` directory. The recommended method of managing `formsweb.cfg` file is using Fusion Middleware Control.

The following parameters defined in Oracle Forms Services configuration file `formsweb.cfg` should be used by the users to enable Single Sign-On in individual or collective Forms applications. It is recommended that this file should be managed using the Fusion Middleware Control.

*Table 9–3   Parameters used to enable single Sign-On*

| Parameter Name | Valid values | Default Value |
| --- | --- | --- |
| ssoMode | mod_osso | false |
| | true | |
| | webgate | |
| | false | |
| ssoProxyConnect | yes | yes |
| | no | |
| ssoDynamicResourceCreate | true | true |
| | false | |
| ssoErrorUrl | String URL | |
| ssoCancelUrl | String URL | |

> **Note:** A detailed description of these parameters along with their possible values are discussed below.

These Oracle Forms parameters in the `formsweb.cfg` file are set in the **User Parameter** section, which define the behavior for all Forms applications run by the server. These parameters can also be set in a **Named Configuration**, which define the settings for a particular application only. A single sign-on parameter set in a Named Configuration section overrides the same parameter set in the **User Parameter** section.

**To enable single sign-on for an application:**

1. Start Fusion Middleware Control.

2. Select **Web Configuration** from the **Forms** menu.

3. Select the row that lists the configuration section for your application.

4. In the Section region, select **sso** in the **Show** drop down list.

5. In the Section region, select the row containing `ssoMode`.

6. In the **Value** field, enter `mod_osso` or `webgate` or `TRUE`.

7. Click **Apply** to update the `formsweb.cfg` file.

   Single sign-on is now enabled for the selected application.

**To disable single sign-on for an application:**

1. Select **Web Configuration** from the **Forms** menu.

2. Select the row that lists the configuration section for your application.

3. In the Section region, select **sso** in the **Show** drop down list.

4. In the Section region, select the row containing `ssoMode`.

5. In the **Value** column, enter `FALSE`.

6. Click **Apply**.

   Single sign-on is now disabled for the selected application.

## 9.4.1 ssoMode

The `ssoMode` parameter enables a Forms Services application to connect to an authentication server. Following are the values that the single sign-on parameter, `ssoMode` can assume:

- `ssoMode`, when set to `TRUE` or `mod_osso` indicates that the application requires authentication by OAM Server or OracleAS Single Sign-On Server.

- `ssoMode`, when set to `webgate` indicates that the application requires authentication by OAM server using webgate as the access client. Webgate must be manually installed and configured.

- `ssoMode`, when set to `FALSE` indicates that the application does not require authentication with an authentication server.

By default, Oracle Forms applications are not configured to run in single sign-on mode. The ssoMode parameter can be set in two places in the formsweb.cfg file:

- By setting `ssoMode` in the default section of `formsweb.cfg` with a value of `true` or `mod_osso` or `webgate` which allows all applications to run in single sign-on mode by this Forms Services instance

- By setting the `ssoMode` parameter in a named configuration of an Oracle Forms application which enables or disables single sign-on only for this particular application, for example:

```
[myApp]
form=myFmx
ssoMode=true
```

### 9.4.2 ssoProxyConnect

The `ssoProxyConnect` parameter enables a user to control when Oracle Forms should use a proxy connection to the database and when it should not. The `ssoProxyConnect` parameter can be set in two ways:

- By setting `ssoProxyConnect` in the default section of `formsweb.cfg` with a value of `yes` which allows all applications to run in single sign-on mode by this Forms Services instance

- By passing the `ssoProxyConnect` parameter in the URL at runtime, for example `http://<host>:<port>/?config=myapp&......&ssoProxyConnect=yes`

### 9.4.3 ssoDynamicResourceCreate

The `ssoDynamicResourceCreate` parameter is set to `true` by default which allows the user to create a Resource Access Descriptor (RAD) entry in Oracle Internet Directory to run the application if this resource entry does not exist.

Allowing dynamic resource creation simplifies Oracle Internet Directory administration because there is no longer the need for an administrator to create user RAD information in advance. The `ssoDynamicResourceCreate` parameter can be set as a system parameter in the `formsweb.cfg` file or as a parameter of a named configuration. Because the default is set to `true`, this parameter may be used in a named configuration for a specific application to handle a missing RAD entry differently from the default.

Note that enabling an application for single sign-on with the value of the `ssoDynamicResourceCreate` parameter set to `false`, while not specifying a value for the `ssoErrorURL`, causes Oracle Forms to show an error message if no RAD resource exists for the authenticated user and this application.

Since not all administrators want their users to create resources for themselves (and potentially raising issues with Oracle Internet Directory), these parameters allow administrators to control Oracle Internet Directory resource creation. Although the default behavior is to direct users to an HTML form that allows them to create the resource, the administrator can change the setting and redirect the user to a custom URL.

For the configuration section for the Forms application, you need to set these parameters:

```
[myApp]
form=myFmx
ssoMode=true
ssoDynamicResourceCreate=false
```

For information about setting these parameters through Enterprise Manager Fusion Middleware Control, see Section 4.2.4, "Managing Parameters".

### 9.4.4 ssoErrorURL

The `ssoErrorURL` parameter allows an administrator to specify a redirection URL that handles the case where a user RAD entry is missing for a particular application. This parameter has effect only if the `ssoDynamicResourceCreate` parameter is set to `false`, which disables the dynamic resource creation behavior. The `ssoErrorURL` parameter can be defined in the default section and as a parameter in a named configuration section. The URL can be of any kind of application, a static HTML file, or a custom Servlet (JSP) application handling the RAD creation, as in the example below.

```
[myApp]
form=myFmx
ssoMode=true
ssoDynamicResourceCreate=false
ssoErrorURL=http://example.com:7779/servlet/handleCustomRADcreation.jsp
…
```

### 9.4.5 ssoCancelUrl

The `ssoCancelURL` parameter is used in combination with the dynamic RAD creation feature (`ssoDynamicResourceCreate= true`) and defines the URL that a user is redirected to if the user presses the cancel button in the HTML form that is used to dynamically create the RAD entry for the requested application.

### 9.4.6 Accessing Single Sign-on Information From Forms

Optionally, if you need to work with authentication server to authenticate information in a Forms application, the GET_APPLICATION_PROPERTY() Built-in can be used to retrieve the following login information: single sign-on user ID, the user distinguished name (dn), and the subscriber distinguished name (subscriber dn)

```
authenticated_username := get_application_property(SSO_USERID);
userDistinguishedName := get_application_property(SSO_USRDN);
subscriberName := get_application_property(SSO_SUBDN);
config := get_application_property(CONFIG).
```

The Forms application developer can obtain the SSO information such as single sign-on user ID, subscriber distinguished name (subscriber dn), and user distinguished name (dn) in SSO mode with either OracleAS Single Sign-On server or Oracle Access Manager when using `mod_osso` or `webgate` as the access client.

> **Note:** `config` can be obtained even in non-SSO mode.

## 9.5 Integrating Oracle Forms and Reports

Oracle Reports is installed with authentication server enabled.

The best practice for Oracle Forms applications calling integrated Oracle Reports is to use the Oracle Forms Built-in, `RUN_REPORT_OBJECT`.

When requesting a report from an SSO-enabled Oracle Forms application, the authenticated user's SSO identity is implicitly passed to the Reports Server with each

call to RUN_REPORT_OBJECT built-in. The SSO identity is used to authenticate the user to the Reports Server for further authorization checking, if required.

A Forms application running in non-SSO mode can run a report on a SSO-secured Reports Server, but fails if the Reports Server requires authorization. Also, users must provide their SSO credentials when retrieving the Reports output on the Web.

For more information on enabling single sign-on in Forms, see Section 9.4, "Protecting Forms applications with Single Sign-On".

For more information on configuring single sign-on in Reports, refer to *Oracle Fusion Middleware Publishing Reports to the Web with Oracle Reports Services*.

For more information about integrating Oracle Forms and Oracle Reports, see the white paper *Integrating Oracle Forms 11g and Oracle Reports 11g* at http://www.oracle.com/technetwork/developer-tools/forms/overview/index.html.

### 9.5.1 Forms and Reports Integration in non-SSO mode

Prior to 11*g* Release 1 (11.1.1), Oracle Reports generated sequential job IDs, making it easy to predict the job ID. This meant that unauthorized or malicious users could potentially view the job output using GETJOBID through rwservlet to obtain job output that belongs to another user. In 11*g*, Oracle Reports can generate random and non-sequential job IDs to make it impossible to predict the job ID for a particular job. This feature must be enabled postinstallation. Only the user who runs a report from Oracle Forms Services is able to see its output. Other users should not be able to see the report output as job IDs are random non-sequential numbers.

For more information about generating random and non-sequential job IDs, see *Oracle Fusion Middleware Publishing Reports to the Web with Oracle Reports Services.*

For a non-secure Reports Server, the user ID and password for administrators can be set in the identifier element of the Reports Server configuration file.

For more information about configuring the access levels for the users, refer to *Oracle Fusion Middleware Publishing Reports to the Web with Oracle Reports Services*.

### 9.5.2 Using Multiple Reports Server Clusters in Oracle Forms Services

If your Oracle Forms application from a prior release uses multiple Reports Server cluster names, you can map each of those cluster names to a different Reports Server. In Oracle Reports 11*g* Release 1 (11.1.1), Reports Server clustering was deprecated. An Oracle Forms application from prior releases that includes a Reports Server cluster name will fail to bind to the Reports Server cluster it references.

To resolve this issue, the reports_servermap element maps a cluster name to a Reports Server name. This avoids the necessity to change the cluster name in all Oracle Forms applications.

An Oracle Forms application can call Oracle Reports in the following ways:

- Using RUN_REPORT_OBJECT: If the call specifies a Reports Server cluster name instead of a Reports Server name, the reports_servermap environment variable must be set in the Oracle Forms Services default.env file. If your Oracle Forms application uses multiple Reports Server cluster names, you can map each of those cluster names to a different Reports Server using reports_servermap in rwservlet.properties, as follows:

  ```
  <reports_servermap>
  ```

```
cluster1:repserver1;cluster2:repserver2;cluster3:repserver3
```

```
</reports_servermap>
```

For example, if your Oracle Forms application includes 3 clusters with names `dev_cluster`, `prd_cluster`, and `qa_cluster` in 10.1.2, you can map these cluster names to respective server names in later releases, as follows:

```
<reports_servermap>
```

```
dev_cluster:dev_server;prd_cluster:prd_server;qa_cluster:qa_
server
```

```
</reports_servermap>
```

- Using `WEB.SHOW_DOCUMENT`: In this case, the request is submitted to rwservlet. If the call specifies a Reports Server cluster name instead of a Reports Server name, the `reports_servermap` element must be set in the `rwservlet.properties` file. For example:

```
<reports_servermap>
```

```
cluster:repserver
```

```
</reports_servermap>
```

For more information, see *Oracle Fusion Middleware Publishing Reports to the Web with Oracle Reports Services*.

### 9.5.3 Integrating Forms and Reports Installed in Different Instances

In 11*g*, Forms and Reports can be configured separately in different instances. If you chose to install Forms and Reports in different Oracle instances, and later require Forms and Reports integration, you need to manually configure files required to establish communication with Reports Servers. For more information, see *Oracle Fusion Middleware Publishing Reports to the Web with Oracle Reports Services*.

## 9.6 Enabling and Configuring Proxy Users

This section contains the following:

- Section 9.6.1, "Proxy User Overview"
- Section 9.6.2, "Enabling Proxy User Connections"
- Section 9.6.3, "Enabling SSO for Proxy Users"
- Section 9.6.4, "Accessing the Forms Application"
- Section 9.6.5, "Changes in Forms Built-ins"
- Section 9.6.6, "Reports Integration with Proxy Users"

### 9.6.1 Proxy User Overview

Many large applications, including Oracle's own E-Business Suite, use a single username for all connections. This makes it possible to manage users in a way that often suits large companies better but it creates a problem with auditing. All inserts, updates and removals of records appear, from the database's perspective, to have been done by a single user. To restore auditing, the application developers must write and implement customized auditing code in the database that requires a user name to be passed to the database from the application. This step not only takes development

time, but also duplicates functionality that is already implemented in the Oracle Database.

The second issue is security. If that single user access is ever compromised, the compromised user will have access to the entire application schema.

To address these two issues, Oracle Database supports proxy user authentication, which allows a client user to connect to the database through an application server, as a proxy user.

Figure 9–6 describes the authentication of a Forms proxy user.

*Figure 9–6   Proxy User Authentication*
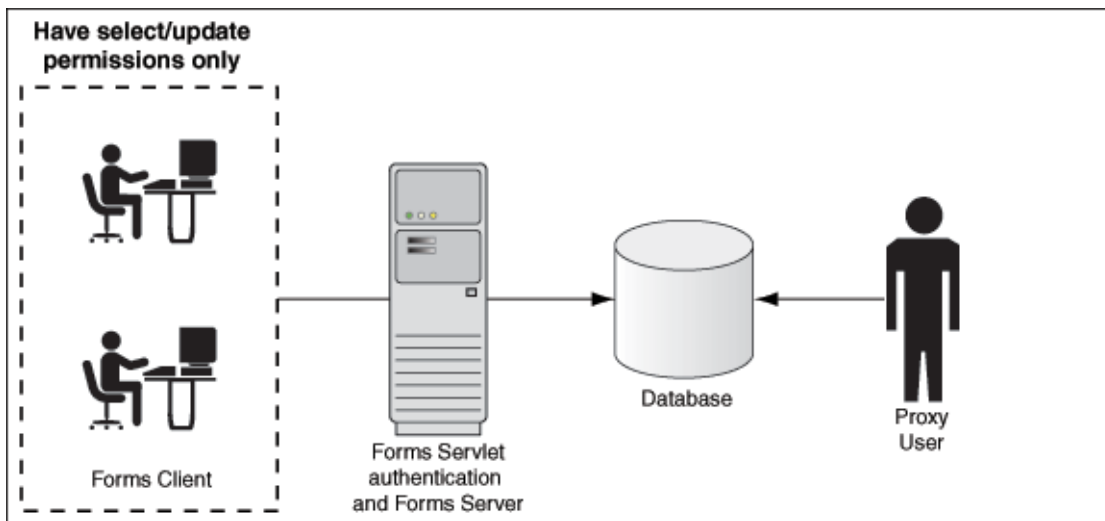


- Oracle Forms authenticates the user through Oracle Internet Directory or LDAP, as shown in the center of the image.

- Forms then connects as the proxy user with or without a password, passing in the real username from the Oracle Internet Directory repository.

- Typically, the proxy user is configured with least set of privileges. In the following procedure, the proxy user has "connect" and "create session" privileges.

- The database accepts the `create` session action for the proxy user and uses the real username in audits and access control.

- The Oracle Internet Directory user cannot connect to the database independently without configuration of the proxy user account.

- The proxy user account isolates the client from direct SQL*Plus connections.

## 9.6.2 Enabling Proxy User Connections

To use a proxy support in Forms, you first need to create a proxy user. In this example, the proxy user is called `midtier`:

1. Create a proxy user in the database.

   ```
   SQL> CREATE USER midtier IDENTIFIED BY midtierPW;
   ```

2. Assign connect and create session privileges to midtier:

```
SQL> GRANT CONNECT,CREATE SESSION TO midtier;
```

At this point, this proxy user has connect and create session privileges and has no grants on any of the user schemas.

3.  Create a database user which has one-to-one mapping with a SSO username (that is, if `appuser` is the SSO username create database user `appuser`).

    ```
    SQL> CREATE USER appuser IDENTIFIED BY appuserPW;
    ```

4.  Assign create session privileges to appuser.

    ```
    SQL> GRANT CREATE SESSION TO appuser;
    ```

5.  To make it possible to connect through the midtier user you need to alter the database user:

    ```
    SQL> ALTER USER appuser GRANT CONNECT THROUGH midtier;
    ```

    The user `appuser` can now connect through the midtier account.

    Alternatively, you can define the roles that the proxy user can connect to the database as

    ```
    SQL> ALTER USER appuser GRANT CONNECT THROUGH midtier WITH ROLE <role_name>;
    ```

    Repeat Step 3 and 4 for all database users who need to use the proxy user account.

It is also possible to set up the database users in Oracle Internet Directory with the help of the database functionality called Enterprise User Security. If you choose this method, the proxy user is the only user defined in the database and the additional benefit of easy administration is gained. For more information on using Enterprise User Security, refer to the *Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory 11g Release 1 (11.1.1).*

The application user's password is not presented to the database; only the user name and the proxy user's user name and password. Forms, with the help of OCI calls, issues the equivalent of:

```
SQL> connect midtier[appuser]/midtierPW@databaseTnsName
```

For example, suppose your application always connects to the database using midtier. This midtier now informs the database that the actual user is `appuser`. Without using proxy users, the SQL command `select USER from DUAL` would return midtier, but, using proxy users, this query returns `appuser`. This essentially tells the database to trust that the user is authenticated elsewhere and to let the user connect without a password and to grant the connect role.

> **Note:**
>
> - In the Step 3 of the above procedure, the database users are typically configured to have a subset of permissions granted to a schema. For example, appuser is granted `CREATE` permissions to the schema `app_schema` with the SQL command:
>
>   ```
>   SQL> GRANT CREATE ON SCHEMA app_schema TO appuser
>   ```
>   Thus, the appuser is restricted to perform only a set of actions in proxy user mode.
>
> - When the database user (for example, appuser) is connected in proxy mode, user actions of the database users are audited rather than that of the proxy user. For more information on user action auditing, refer to the Oracle Database documentation at http://www.oracle.com/technetwork/indexes/documentation/index.html.

### 9.6.3 Enabling SSO for Proxy Users

Create a configuration section in `formweb.cfg` for single sign-on (for example, ssoapp) and set `SSOProxyConnect` to `yes` and `ssoMode` to `true` or `mod_osso` or `webgate`.

The username and password that is used for the proxy connection is defined in the RAD entry in Oracle Internet Directory for the user that is logging on. If `ssoProxyConnect=yes`, the connect string equivalent issued by Forms is in effect:

```
SQL> connect RADUsername[appuserName]/RADPassword@databaseTnsName
```

### 9.6.4 Accessing the Forms Application

After enabling proxy user connections and single sign-on, perform the following steps to access the forms applications:

1. Run the forms application with the URL `http://<host name>:<http port>/forms/frmservlet?config=ssoapp` where `ssoapp` is the name of the configuration section with single sign-on (`ssoMode`) is enabled.

2. Use the single sign-on user name and password to log in (in this example given in Section 9.6.2, "Enabling Proxy User Connections", the single sign-on username is `appuser` and password is `appuserPW`).

### 9.6.5 Changes in Forms Built-ins

The Built-in `get_application_property` now takes a new parameter called `IS_PROXY_CONNECTION` (a Boolean). When this parameter is supplied, the call returns `true` if the form is running in proxy user mode, `false` otherwise.

### 9.6.6 Reports Integration with Proxy Users

The integration with Reports is maintained when a proxy user is used in Forms. The Oracle Reports administrator has to set up a proxy user. Ensure that the following configuration has been completed in the Reports configuration files.

In `rwserver.conf`, enter the Forms configuration section name (`frm_config_name`) and database SID name that is configured for proxy user support (dbname).

```
<dbProxyConnKeys>

    <dbProxyKey name="frm_config_name"  database="dbname"/>

</dbProxyConnKeys>
```

In `rwservlet.properties`, ensure that Proxy mode is enabled.

```
<enabledbproxy>yes</enabledbproxy>
```

For more information about Reports configuration files, see *Oracle Fusion Middleware Publishing Reports to the Web with Oracle Reports Services*.

## 9.7 Postinstallation Configuration

This section describes some postinstallation steps. These steps are optional and the user may be required to perform these steps depending on the choices that the user has made in Section 9.2, "Setup Process". This section includes the following topics:

- Configuring Forms J2EE application with Oracle Internet Directory
- Generating the Access Client File
- Enabling mod_osso in the OHS directives configuration
- Installing and Configuring Webgate with OAM

### 9.7.1 Configuring Forms J2EE application with Oracle Internet Directory

The users connecting through a Forms application as proxy users must also be defined in authentication server and Oracle Internet Directory. Oracle Forms authenticates the user via authentication server (using authentication server with Forms is a requirement when employing a proxy user). Oracle Forms then connects to the database as the proxy user with a username and password that is in the RAD for the Oracle Internet Directory entry for the application user.

For more information on Oracle Forms and Identity Management integration, see Section 11.1.4, "Leveraging Oracle Identity Management Infrastructure."

---

**Note:** When you change the Oracle Web Cache port using Enterprise Manager, regenerate the `osso.conf` and copy the generated `osso.conf` file to `$ORACLE_INSTANCE/config/OHS/<OHS_INSTANCE>/moduleconf` directory. Restart the Oracle HTTP Server and Oracle Web Cache for the changes to take effect.

---

**To access the Associate/Disassociate OID page:**

1. Start Enterprise Manager.

2. Navigate to the Forms **Home** page.

3. From the Forms menu, select **Associate/Disassociate OID**.

   The **Associate/Disassociate OID** page is displayed.

*Figure 9–7  Associate/Disassociate OID*
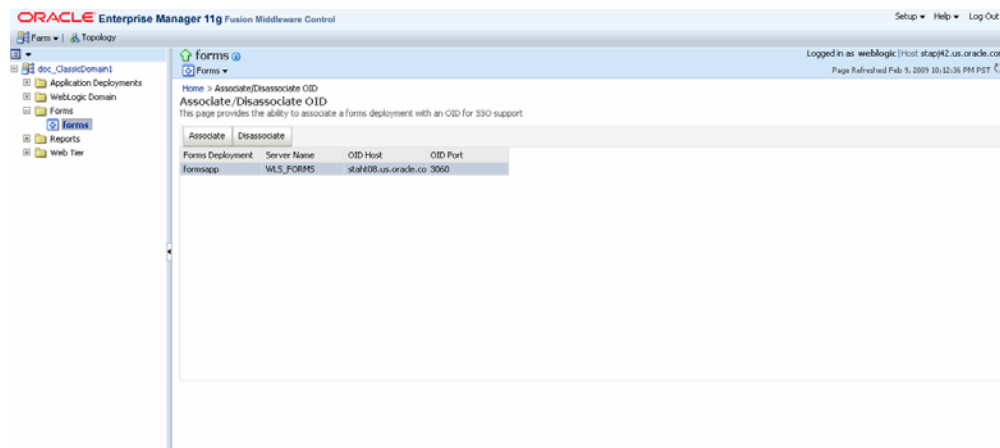


## To Associate OID Host with a Forms Application

1. To associate an Oracle Internet Directory host with a Forms application for the first time, from the **Associate/Disassociate OID** page, select the Forms application. Click **Associate**.

   The Associate dialog appears.

2. Enter the Oracle Internet Directory Host details as described in Table 9–4, " Oracle Internet Directory Host Details".

3. Click **Associate**.

   The **Associate/Disassociate OID** page reappears.

*Table 9–4    Oracle Internet Directory Host Details*

| Parameter | Description |
| --- | --- |
| OID Host | Select the Oracle Internet Directory Host from the list or select New Oracle Internet Directory (OID) host to add new host details. |
| New OID host | Host name of the Oracle Internet Directory server. This field is enabled if you have selected to add new Oracle Internet Directory (OID) Host. |
| New OID Port | Port number on which Oracle Internet Directory is listening. This field is enabled if you have selected to add new Oracle Internet Directory Host. |
| Username | Oracle Internet Directory Administrator username |
| Password | Oracle Internet Directory Administrator password |
| Use SSL Port | Select this box if the connection to the Oracle Internet Directory Host should use SSL (in which case the port number provided should be the SSL port). |

4. Generate and apply the access client file as described in Section 9.7.2, "Generating the Access Client File,".

## To Disassociate OID Host from a Forms Application

1. From the **Associate/Disassociate OID** page, select the Forms application. Click **Disassociate**.

A confirmation box appears.

2. Click **Yes**.

The Oracle Internet Directory host is disassociated from the Forms application.

3. Restart the Oracle WebLogic Managed Server and the front-end OHS for the changes to take effect.

To prevent users from being inadvertently disconnected from active forms sessions, ensure you choose to restart Oracle WebLogic Managed Server and the front-end OHS at a convenient time when users are not running any forms sessions.

**To re-associate an OID Host with a Forms Application**

1. From the **Associate/Disassociate OID** page, select the Forms application. Click **Disassociate**.

2. From the **Associate/Disassociate OID** page, select the Forms application. Click **Associate**.

Enter the Oracle Internet Directory Host details as described in Table 9–4, " Oracle Internet Directory Host Details".

3. Generate and apply the access client file as described in Section 9.7.2, "Generating the Access Client File,".

## 9.7.2 Generating the Access Client File

The access client file must be generated for the authentication servers. The procedure for generating the access client file are different depending on whether you are using OracleAS Single Sign-On 10*g* or Oracle Access Manager 11*g*.

**Generating the osso.conf file for the OracleAS Single Sign-On Server 10g**

Perform the following steps to generate the `osso.conf` file for the OSSO Server:

1. Run the `ssoreg.sh` script located at *ORACLE_HOME*/sso/bin on the authentication server.

```
ORACLE_HOME/sso/bin/ssoreg.sh
-oracle_home_path <ORACLE_HOME>
-site_name www.example.com
-config_mod_osso TRUE
-mod_osso_url http://www.oidtierexample.com:7777
-config_file osso.conf
-remote_midtier
```

> **Note:** On Windows, run the `ssoreg.bat` file.

2. Copy the generated `osso.conf` file to *ORACLE_INSTANCE*/config/OHS/<OHS_INSTANCE>. For more information, see *Oracle Application Server Single Sign-On Administrator's Guide*.

3. Restart Oracle WebLogic Managed Server (WLS_FORMS) and the front-end OHS for the changes to take effect.

To prevent users from being inadvertently disconnected from active forms sessions, ensure you choose to restart Oracle WebLogic Managed Server and the

front-end OHS at a convenient time when users are not running any forms sessions.

**Generating the osso.conf file for the Oracle Access Manager**

Perform the following steps to generate the `osso.conf` file for the OAM Server using the OAM console:

1. Log in to the OAM console.

2. Navigate to the **System Configuration** tab. Select **Agents** and navigate to the **OSSO Agents** node. Click **Create**.

3. Provide all the details such as the Base URL. Ensure that the **Auto Create Policies** check box is checked.

4. Click **Apply**.

   The `osso.conf` file is generated for the OAM server. The location of the file is mentioned in the OAM console.

5. Copy the generated `osso.conf` file to `ORACLE_INSTANCE`/config/OHS/<OHS_ INSTANCE>.

6. Restart OHS for the changes to take effect.

   To prevent users from being inadvertently disconnected from active forms sessions, ensure you choose to restart Oracle WebLogic Managed Server and the front-end OHS at a convenient time when users are not running any forms sessions.

   For more information about generating the osso.conf file using the OAM console, see *Registering and Managing OSSO Agents Using the Administration Console* in *Oracle Fusion Middleware Administrator's Guide for Oracle Access Manager*.

## 9.7.3 Enabling mod_osso in the OHS directives configuration

You must perform this task if you have chosen to install and configure Forms in non-SSO mode and want to enable SSO later. To be able to enable SSO later, you must register `mod_osso` as a partner application in OHS with authentication server. To achieve this, perform the following steps:

1. Follow the steps to generate and copy the `osso.conf` file as described in Section 9.7.2, "Generating the Access Client File,".

2. Create a `mod_osso.conf` file in the `ORACLE_INSTANCE`/config/OHS/<OHS_ INSTANCE>/moduleconf directory. The file looks similar to the following example:

```
LoadModule osso_module ${ORACLE_HOME}/ohs/modules/mod_osso.so
<IfModule mod_osso.c>
OssoIpCheck off
OssoSecureCookies off
OssoIdleTimeout off
OssoConfigFile osso.conf
#
# Insert Protected Resources: (see Notes below for
# how to protect resources)
#
#_____-
#
# Notes
#
```

```
#_____-
#
# 1. Here's what you need to add to protect a resource,
#    e.g. <ApacheServerRoot>/htdocs/private:
#
<Location /private>
require valid-user
AuthType Osso
</Location>

</IfModule>

#
# If you would like to have short hostnames redirected to
# fully qualified hostnames to allow clients that need
# authentication via mod_osso to be able to enter short
# hostnames into their browsers uncomment out the following
# lines
#
#PerlModule Apache::ShortHostnameRedirect
#PerlHeaderParserHandler Apache::ShortHostnameRedirect
```

3. Add the following lines to the beginning of the `forms.conf` file:

```
<IfModule !mod_osso.c>
LoadModule osso_module  ${ORACLE_HOME}/ohs/modules/mod_osso.so
</IfModule>
<IfModule mod_osso.c>
OssoHTTPOnly off
</IfModule>
```

4. Associate the OID Host using Enterprise Manager as described in To Associate OID Host with a Forms Application in Section 9.7.1, "Configuring Forms J2EE application with Oracle Internet Directory".

5. Restart the Oracle WebLogic Managed Server (WLS_FORMS) and the front-end OHS for the changes to take effect.

## 9.7.4 Installing and Configuring Webgate with OAM

For webgate to work with Oracle Access Manager 11*g*, you must install and configure webgate manually. For information about installing and configuring webgate as the access client, see *Installing and Configuring Oracle HTTP Server 11g Webgate for OAM* in *Oracle Fusion Middleware Installation Guide for Oracle Identity Managment*.

Postinstallation, you must register webgate with OAM 11*g* so that webgate can directly communicate with Oracle Access Manager 11*g* services. To register webgate with OAM 11*g*, perform the following steps:

1. Create a webgate 11*g* agent by using either RREG tool or through OAM console.

   While creating the webgate agent, you must add the following URL to the Protected Resource List:

   ```
   /forms/frmservlet?*oamMode=true*
   ```

   Add "/" and "/.../" to the Public Resource List.

2. Copy `ObAccessClient.xml` and `cwallet.sso` to the webgate instance directory of the relevant OHS as shown in the following example:

   ```
   cp <OAM_DOMAIN_HOME>/output/<Agent_Name>/*.xml <WEBGATE_
   ```

```
INSTANCE>/webgate/config
```

For information about registering webgate as an agent by using either OAM console or RREG tool, see *Register the New Webgate Agent* in *Oracle Fusion Middleware Installation Guide for Oracle Identity Managment*

> **Note:**   After installing and configuring webgate access client to work with Oracle Access Manager 11*g*, when the user accesses Forms application for the first time with webgate as the access client, the user will see a java exception error. As a workaround to this issue, the user must disable the HTTPOnly parameter. To achieve this, perform the following steps:
>
> 1. Log in to the OAM Administration Console.
> 2. Select **Authentication Schemes** and navigate to **LDAPScheme**.
> 3. Set the `ssoCookie` parameter value to *disablehttponly*.
> 4. Click **Apply**.

# 10

# Configuring and Managing Java Virtual Machines

This chapter contains the following sections:

## 10.1 Why Use Java Virtual Machine Pooling?

When a Forms application calls out to Java, a JVM is attached to each Forms process the first time the process makes a call. This JVM remains attached to each process for the remainder of the processes' lives, even though any individual process may never call out to Java again, potentially causing resource contention. JVM pooling makes provisions for sharing a limited number of JVMs among all participating Forms processes. Even though all Forms processes might at one point call out to Java, if only a subset of these call out to Java at any given point in time, only as many JVMs as are necessary at peak usage, need be started. Using JVM pooling brings the potential to significantly reduce resource usage for a Forms installation that calls out to Java.

When a Forms runtime process needs to execute Java, it sends a message to the Java Virtual Machine (JVM) that is contained in the JVM controller. The JVM creates a new thread for that Forms runtime process. The JVM then continues to listen for the next new request from a different Forms runtime process while the newly created thread processes the request and sends the results back to the Forms runtime process. For the life of this Forms session, the Forms runtime process communicates directly with that thread.

Java Virtual Machine pooling is a separate process that contains the JVM controller. With JVM pooling, the JVM runs outside of the Forms runtime process. The JVM can

also be shared by multiple Forms runtime processes. The JVM controller process is not a JVM itself, but a container that contains a JVM in a similar way that the Forms Runtime process contains an in-process JVM. Using JVM pooling is optional. Administrators can choose to not use JVM pooling and have the JVM contained in the Forms runtime process.

Java Virtual Machine (JVM) pooling works in conjunction with the Java Importer. It also works with Forms' ability to call out to Reports. The Java Importer allows developers at design time to reference Java classes from PL/SQL within the Forms Builder. At runtime, Forms uses a Java Virtual Machine (JVM) to execute Java code. In earlier versions of Oracle Forms, each Forms session that used the Java Importer had its own JVM instance to execute Java code. In this model, each JVM consumes memory on the server, and if there are many concurrent users, the amount of memory consumed by the multiple JVM processes becomes significant.

For more information on the Java Importer, see the Oracle Forms Developer online help.

When you enable JVM pooling, administrators can consolidate the number of running JVM instances so that the Forms sessions can share JVMs rather than each one having its own instance. The result is a large reduction in memory consumption, thus freeing up more resources on your server.

You also need to consider JVM pooling in application design and deployment. For more information, see Chapter 10.5, "Design-time Considerations".

### 10.1.1  JVM Pooling in Forms and Reports Integration

In 10*g*, Forms Runtime process creates a separate JVM before calling Reports and Reports uses this JVM to execute the java methods. This JVM is part of the Forms Runtime process. In 10*g*, the JVM pooling feature is used only by the Java Importer. However, in 11*g*, with JVM pooling enabled, Oracle Forms Services uses a shared JVM controller for Oracle Reports requests.

Instead of each Forms Runtime process having its own instance of the JVM, JVMs can be shared by multiple Forms Runtime processes. With JVM pooling, a process called JVM controller is available which houses the JVM. Forms Runtime processes can share this JVM. This would result in a large reduction of memory consumption, freeing more resources on the server.

A form can be configured to use a specific JVM controller using the jvmcontroller parameter. The jvmcontroller parameter indicates to the Forms Runtime process which JVM controller to use. This can be set in the Forms Configuration File, `formsweb.cfg`. Alternatively, this information can also be passed as a parameter in the URL for invoking the Forms Application. The parameters that need to be used during startup of the jvmcontroller have to be specified in the JVM controller's configuration file, `jvmcontrollers.cfg`.

For more information on using JVM pooling for Reports integration, see Section 10.10, "Integrating Forms and Reports".

## 10.2  About Child Java Virtual Machine Processes

Since each Forms runtime process has its own thread within the JVM, there is concurrency. If the JVM reaches a specified number of concurrent requests, it will spawn a *child* JVM to share the load. Moreover, it's possible to have multiple JVM controllers, each of which may have multiple child JVMs.

For example, different Forms applications may want to use different JVMs with different options or classpaths. You can specify which JVM controller and Forms application should be used in the named sections of the Forms configuration file (`formsweb.cfg`). See Section 10.8.6, "Forms Configuration File Settings" for more information.

Figure 10–1 shows an example of what an environment might look like using JVM pooling. There are two JVM controllers: the first one is using only its in-process JVM, the second one is using three JVMs.

*Figure 10–1   Multiple JVM Controllers with Child Processes*
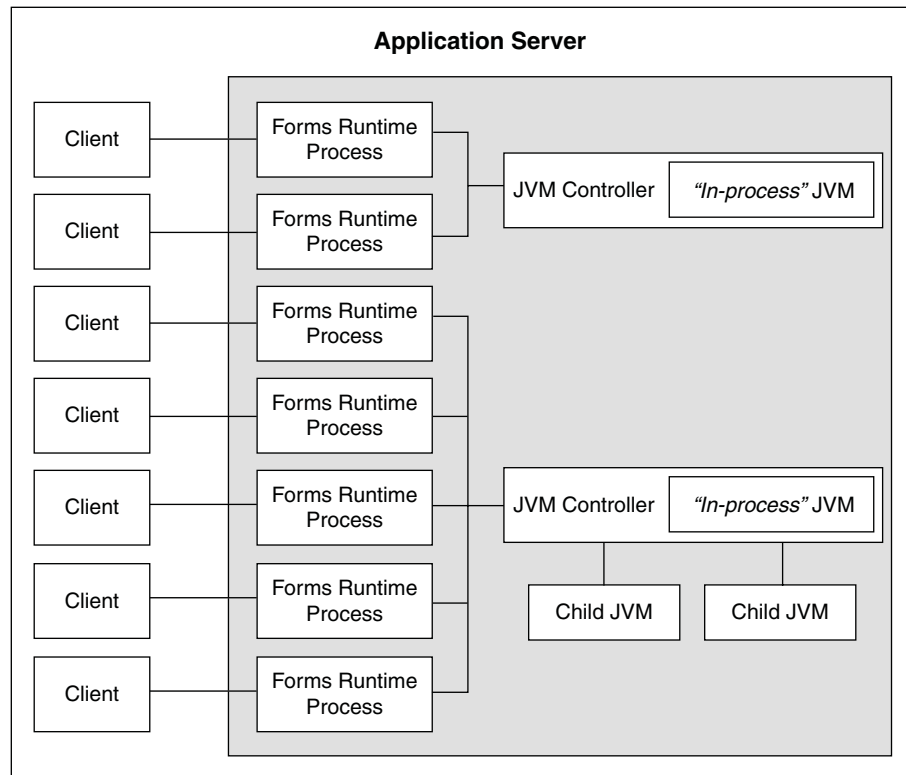


Although it's not shown in Figure 10–1, each JVM controller has a unique name which is used in starting and stopping, or for referencing in the Forms configuration file.

Figure 10–1 is conceptual only in that it shows different Forms applications using different JVM controllers. However, the Forms runtime process does not communicate with the JVM controller, but directly with one of the available JVMs. Therefore, the first two clients in the diagram can only use the in-process JVM; the rest have three available JVMs to work with.

When the performance of a JVM degrades significantly, it probably means it is servicing too many requests. In that case, it is possible to have multiple "child" JVMs for the same JVM controller which get created dynamically as needed.

The JVM parameter `maxsessions` specifies how many Forms runtime processes are allowed to attach to a JVM before a new child JVM is created. When a child JVM is started, it inherits the same parameters as the JVM controller.

If any JVM has `maxsessions` connections, it does not take any request from new Forms runtime processes. When a new Forms runtime process first attempts to execute Java code, it attaches to a JVM that is available, that is, has fewer than `maxsessions`

connections. The method of choosing the JVM is entirely arbitrary; there is no load balancing or round-robin algorithm.

If a JVM reaches maxsessions connections, but another JVM has not, no new JVM is created. If all JVMs have simultaneously reached maxsessions connections, another child JVM is created, and so on.

Child JVMs are not automatically removed when the load is reduced. So if you want to remove some child JVMs, the JVM controller must be stopped, which also stops all child JVMs. Then the JVM controller can be restarted.

The scope of a child JVM is within the context of a JVM controller namespace. For example, if you have two JVM controllers, `ordersJVM` and `hrJVM`, ordersJVM and its child JVMs do not affect – nor are affected by – `hrJVM` or its child JVMs.

### 10.2.1  Child JVM Example

Suppose the JVM controller called ordersJVM has `maxsessions=50`. Each Orders application that runs sends requests to ordersJVM. Each time a new Forms runtime process sends a request to ordersJVM, a new thread is created that communicates with the Forms runtime process. The JVM controller then returns to listening for new requests. As users end their sessions, the threads in the JVM are also terminated.

When the ordersJVM controller receives the 50th concurrent request (not necessarily the first 50 users because some of them may have quit before the later users started) it will spawn a child JVM. Since it inherits its parent's settings, `maxsessions` for this child JVM will also be `50`. At this stage, the JVM controller has 50 connections, and the child JVM has none.

As new users start this Oracle Forms application and execute Java code, the Forms runtime process attaches to a JVM that is listening within the JVM controller namespace. Since the JVM controller has 50 connections, it is unavailable and the child JVM receives the request. Later, when the parent JVM controller has fewer connections because some users have quit their applications, it is available to receive new requests as long as it has not reached `maxsessions` connections.

While all this is going on, the `hrJVM` is operating independently. Overflow connections from `ordersJVM` will not connect to `hrJVM`, only to child JVMs of `ordersJVM`.

## 10.3  About Multiple JVM Controllers

The JVM pooling architecture allows you to have multiple JVM controllers, each of which may have child JVMs. You would use multiple JVM controllers if:

- You want each application to have its own JVM controller so that it can be started and stopped independently of others.

- Different applications require different settings. For example, you may not want to mix classpaths or JVM settings between different controllers.

- You want to monitor resource usage of the JVM controllers from Fusion Middleware Control. If different JVM controllers are used by different applications and/or groups of users, you can determine how resources are being consumed by your Java Importer code.

- You have multiple development, test, or production environments on the same computer.

- You do not want different applications to share static data.

## 10.4  JVM Pooling Usage Examples

Consider, for example, an Oracle Forms application that has a user interface button. When a user presses the button, Oracle Forms takes the value from a field on the screen, and passes it to Java (using the Java Importer feature) to do some complex calculation which cannot be done in PL/SQL. The result is then returned and displayed in a field in the Form. One JVM process is running to execute this Forms session.

Figure 10–2 shows how this Oracle Forms session has its own  **in-process JVM** because JVM pooling is not enabled. In the left side of the image, there are multiple clients running their own Forms session. In the center of the image, each client makes a call to its own Forms Runtime process, which contains its own JVM process.

*Figure 10–2   Forms Runtime with no JVM Pooling*
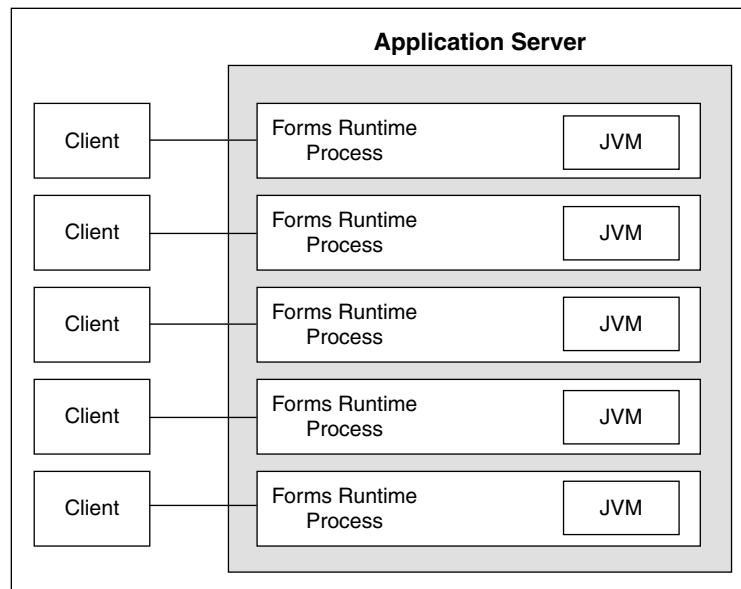


Figure 10–3 shows the Forms Runtime processes sharing a single JVM process when JVM pooling is enabled, as shown in the right side of the image.

*Figure 10–3   Forms Runtime with JVM Pooling Enabled*
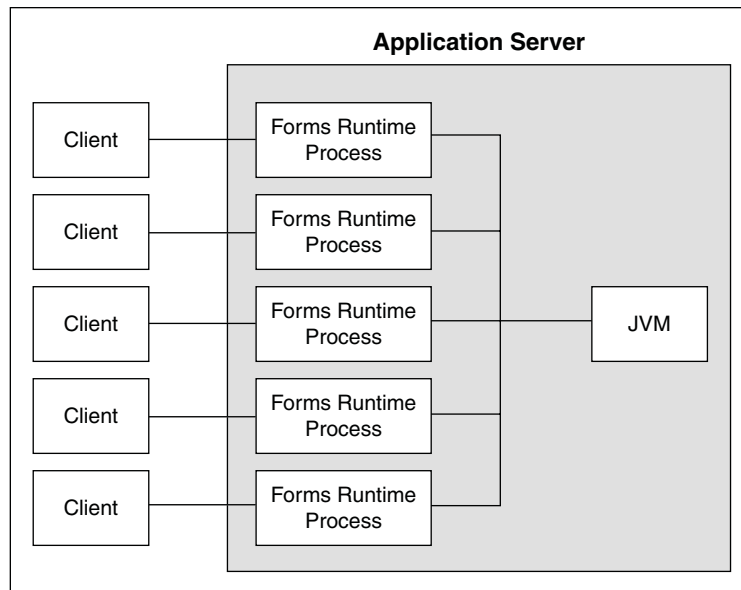


In this example, five clients working in the same application through their own runtime processes are using a pooled JVM process instead of each Forms Runtime process spawning its own JVM instance. This can be a significant savings in memory usage and system resources.

## 10.5 Design-time Considerations

This section contains the following:

- Section 10.5.1, "Re-importing Your Java Code"
- Section 10.5.2, "About Sharing Static Variables Across Multiple JVMs"

### 10.5.1 Re-importing Your Java Code

If you used the Java Importer feature of Oracle Forms prior to the availability of JVM Pooling, you will need to reimport your Java classes before using JVM pooling. When you originally imported your Java classes, PL/SQL wrappers for the Java classes were generated, which you can see in the Program Units that were created in your Form. However, the PL/SQL wrappers that are generated by the Java Importer to utilize JVM pooling are different.

From Oracle Forms Services 10*g* and later, the Java Importer generates the "new" PL/SQL wrappers. If you want to use the Java Importer, but do not wish to take advantage of JVM pooling, the in-process JVM will work with the new PL/SQL wrappers. It will also continue to work with the older-style PL/SQL wrappers.

### 10.5.2 About Sharing Static Variables Across Multiple JVMs

One advantage of JVM pooling is the ability to share data between instances of a class by using static variables. However, static variables will be shared between instances of the same class within a JVM, but not across JVMs. You will need to plan accordingly.

For example, suppose your loan class has a static variable called `interestRate` because all instances use the same interest rate in calculations. If you are using only

one JVM, and one of the instances of your loan class changes `interestRate`, all of the other instances will be affected (which is what you want).

However, if the JVM controller has one or more child JVMs, there may be at least two JVMs. If interestRate changes in one JVM, the loan instances in the other JVMs won't see this new value. For more information about managing child JVMs, see Section 10.2, "About Child Java Virtual Machine Processes". Prior to JVM pooling, if you changed interestRate it would not affect any other instances because each Oracle Forms Runtime process had its own in-process JVM.

If you rely on static variables to share information between instances of your class, ensure that no child JVM is spawned by setting `maxsessions` to `65535`.

## 10.6 Overview of JVM Configuration

To configure JVM using Fusion Middleware Control, perform the following steps:

1.  Using Fusion Middleware Control, add a new configuration section or modify an existing section in `formsweb.cfg` to enable or disable use of JVM controller for applications. For more information, refer to Section 10.8.6, "Forms Configuration File Settings".

2.  Ensure CLASSPATH is updated in `default.env` or in `jvmcontrollers.cfg`.

3.  Using Fusion Middleware Control, configure the JVM parameters. For more information, refer to Section 10.8.3, "Managing Parameters".

4.  Start the JVM controller. For more information, refer to Section 10.8.5, "Starting and Stopping JVM Controllers with Fusion Middleware Control".

## 10.7 Managing JVM Controllers from the Command Line

If you manage JVM controllers from the command line, you must know the options to start and stop them, as well as specify the environment. You can only access the JVM controllers on the same computer from which they are running.

> **Note:** The mechanics for controlling the JVM controller as described in this chapter are mostly relevant at the command line. It is easier to use Fusion Middleware Control with its user-friendly screens and online help. Fusion Middleware Control users are still urged to read through the following information, however, to understand what the different fields and options mean, and how the JVM controller works.

### 10.7.1 JVM Controller Command Examples

This section describes examples of JVM controller commands. For a detailed explanation on the example, see Section 10.8.7, "Startup Example."

- `dejvm -start jvmcontroller=hrJVM`

  Starts a JVM controller with ID hrJVM. The controller name hrJVM is defined as a named section in the configuration file. Therefore, JVM options and classpath parameters are taken from the configuration file. `maxsessions` is `50` as defined in the Default section, and other parameters take their default values.

- `dejvm -start jvmcontroller=myJVM`

  Starts a JVM controller with ID is myJVM. Since no option was specified, and there is no named section in `jvmcontrollers.cfg`, the JVM options parameter is

"`-Xms512m -Xmx1024m`" and `maxsessions=50` as set in the Default section. The other parameters take on their default values. For instance, the CLASSPATH value is the system CLASSPATH.

- `dejvm -start jvmcontroller=hrJVM jvmoptions="-Xms128m -Xmx256m" maxsessions=75`

  Sets the classpath to `/myJava/hrClasses` as defined in the named section. JVM options are "`-Xms128m -Xmx256m`" because the command line overrides the `jvmcontrollers.cfg` file. Similarly, `maxsessions` is `75`. All other parameters take on their default values.

- `dejvm -start jvmcontroller=myJVM maxsessions=100 classpath=/myJava/myClasses;/moreJava/moreClasses`

  The controller has `jvmoptions="-Xms512m -Xmx1024m"` as defined in the default section of `jvmcontrollers.cfg`. `maxsessions` is 100 which overrides the default section, and classpath is `/myJava/myClasses;/moreJava/moreClasses`. All other parameters take on their default values.

- `dejvm -stop jvmcontroller=hrJVM`

  Stops the hrJVM controller. It must already be started for you to issue this command successfully.

## 10.7.2  Command Restrictions

Keep these command restrictions in mind:

- The commands are case sensitive.

- You can only issue one command at a time to a JVM controller.

- You can only issue a command to one JVM controller at a time.

The available commands for the JVM controller (or the dejvm process) are specified in Table 10–1. If you are using Enterprise Manager, there are screens that have an interface for issuing these commands. If you are using the command line, you may not be able to manage the JVM controller using the Enterprise Manager.

## 10.7.3  Start Command Parameters

Table 10–1 describes the JVM parameters used to start the JVM from the command line.

*Table 10–1    JVM Parameters*

| Parameter | Description |
| --- | --- |
| jvmcontroller | Enter a name for this JVM. This name must contain a legal Oracle identifier that starts with a letter and contains an alphanumeric character, '_', '$' or '#' . An Oracle identifier has a maximum length of 30 bytes. |
| | Hint: You may want to enter a name based on the application that will be accessing it. You cannot change the name of this JVM controller later. |
| maxsessions | Specifies the maximum number of concurrent Oracle Forms sessions this JVM will serve before a new JVM is spawned. This value will override any set for the default JVM controller. |

*Table 10–1   (Cont.)  JVM Parameters*

| Parameter | Description |
| --- | --- |
| classpath | When you specify a classpath, it will override the system classpath or any classpath specified in your environment or any classpath set for the default JVM controller. |
| jvmoptions | Enter any valid options to pass to the JVM. This value will override any set for the default JVM controller. Refer to the Oracle Java documentation for a list of valid JVM startup options. |
| logdir | Leave Log Directory blank to use the log location for the default JVM controller. If any other directory is set, the log file may not be accessible through Enterprise Manager. |
| logging | On, or Off. |

## 10.8  Managing JVM Pooling from Fusion Middleware Control

Fusion Middleware Control provides a Web-based environment to manage all available JVM pooling options. It also lists all JVM controllers in your environment and allows you to (remotely) manage them. For example, you can start and stop JVM controllers; add new ones; or reconfigure existing ones. In addition, Fusion Middleware Control also provides metric information such as resources (memory and CPU) that are consumed by JVM controllers, number of Forms connected, total JVMs, and so on.

While the Forms runtime process interacts directly with a JVMs, the JVM controller manages the JVM, such as starting and stopping a JVM, or getting the state of one, etc. For example, when an administrator stops the JVM controller, the JVM controller ensures that all child JVMs are terminated. You use Fusion Middleware Control to manage the JVM controller.

The JVM controller can be started in three ways:

- From Fusion Middleware Control

- When a Forms application that is bound to an existing JVM controller requests that the controller start up

- From the command line

Fusion Middleware Control reads the JVM controller configuration file. It works in a similar way to the Forms configuration file (`formsweb.cfg`) in that it contains name-value pairs, has a default section, and has named sections. The parameters contained in `jvmcontrollers.cfg` correspond to the start parameters of the JVM controller.

> **Note:**   You cannot change the location or name of the JVM controllers configuration file.

When you start a JVM controller, it takes its settings from the configuration file. You may specify none, some, or all options in this file, both in the default section and in named sections.

Use the JVM Configuration and JVM Controller pages in Fusion Middleware Control to manage JVM pooling tasks:

- Section 10.8.1, "Common Tasks in the JVM Configuration Page"

- Section 10.8.2, "Managing JVM Configuration Sections"

- Section 10.8.3, "Managing Parameters"

- Section 10.8.4, "JVM Configuration Parameters and Default Values"

- Section 10.8.5, "Starting and Stopping JVM Controllers with Fusion Middleware Control"

- Section 10.8.6, "Forms Configuration File Settings"

- Section 10.8.7, "Startup Example"

## 10.8.1 Common Tasks in the JVM Configuration Page

This section describes the common tasks that you can do to edit configuration with the sections of a JVM configuration file and their parameters.

Table 10–2 describes the tasks you can do with the configuration sections within a JVM configuration file:

*Table 10–2    Tasks for Working with Configuration Sections*

| Task | Description | Comment |
|------|-------------|---------|
| Create Like | Creates a copy of a configuration section. | Use to create a configuration section based on the parameters of an existing configuration section. |
| Edit | Opens the Edit Description dialog. | Allows editing the text description of a configuration section. |
| Delete | Opens the Confirmation dialog when deleting a configuration section. | Irrevocably deletes a configuration section and its contents when you press Delete in the Confirmation dialog. |
| Create | Opens the Create Section dialog. | Creates a new configuration section. You must supply a required name and an optional description for it. |

Table 10–3 describes the tasks that you can do to modify the parameters within a named configuration section:

*Table 10–3    Tasks for Working with Parameters in a Named Configuration Section*

| Task | Description | Comment |
|------|-------------|---------|
| Revert | Allows you to revert back to the previous version of the configuration section. | Does not allow you to revert individual changes in a configuration section. |
| Apply | Applies and activates all changes made to parameters in a configuration section. | Once applied, you cannot revert changes to individual parameters. |
| Add | Opens the Add Parameter dialog. | Add a parameter to a configuration section based on a mandatory name and an optional value and description. |

*Table 10–3   (Cont.)  Tasks for Working with Parameters in a Named Configuration Section*

| Task | Description | Comment |
|------|-------------|---------|
| Delete | Deletes a parameter. | Use Apply to save changes or Revert to discard them. Once applied, you cannot revert changes to individual parameters. |

## 10.8.2  Managing JVM Configuration Sections

This section describes creating, editing, duplicating, and deleting named JVM configuration sections.

### 10.8.2.1  Accessing the JVM Configuration Page

**To access the JVM configuration page:**

1. Start the Enterprise Manager Fusion Middleware Control.

2. From the Fusion Middleware Control main page, click the link to the Forms Services instance that you want to configure.

3. From the Forms menu list, select the **JVM Configuration** menu item.

   The JVM Configuration page (Figure 10–4) is displayed.

*Figure 10–4   JVM Configuration Page*
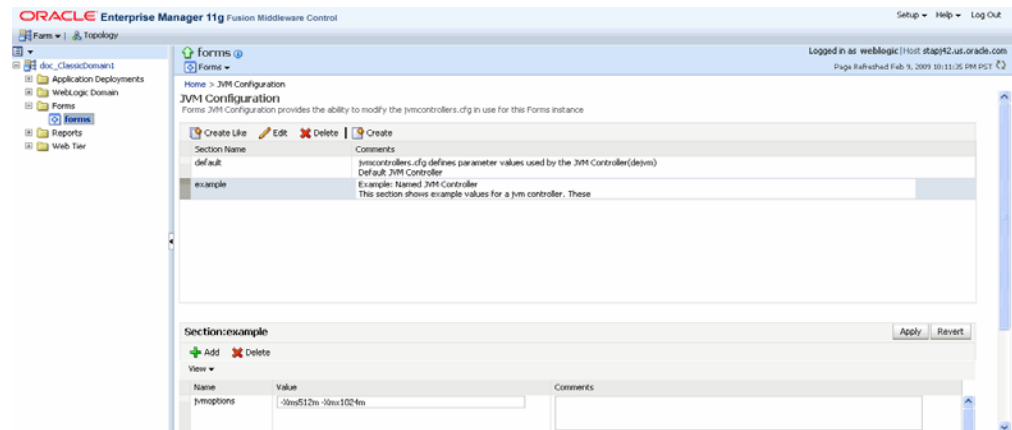


### 10.8.2.2  Creating a New Configuration Section

You can create new configuration sections in `jvmcontrollers.cfg` from the **JVM Configuration** page of Fusion Middleware Control. These configurations can be requested in the end-user's query string of the URL that is used to run a form.

**To create a new configuration section:**

1. From the Fusion Middleware Control main page, click the link to the Forms Services instance that you want to configure.

2. From the Forms menu list, select **JVM Configuration**.

3. Click **Create**.

   The **Create** dialog appears.

4. Enter a name and description for your new configuration section and click **Create**.

   The new configuration section is added.

### 10.8.2.3 Editing a Named Configuration Description

You can edit the description (comments) for a named configuration from the **JVM Configuration** page.

**To edit a named configuration description:**

1. In the **JVM Configuration** region, select the row containing the named configuration for which you want to edit the description.

2. Click **Edit**.

3. The **Edit Description** dialog appears.

4. Enter the description in the Comments field.

5. Click **Save**.

   The **Edit Description** dialog box is dismissed, and your changes are saved and displayed.

### 10.8.2.4 Duplicating a Named Configuration

You can make a copy of a named configuration for backup purposes, or create new configuration sections from existing configuration sections.

**To duplicate a named configuration:**

1. In the **JVM Configuration** region, select **Create Like**.

2. In the Create Like dialog, from the **Section to Duplicate** menu, select the name of an existing configuration section you want to duplicate.

3. In the **New Section Name** field, enter a name for the new configuration section. The name for the new configuration section must be unique.

4. Click **Create**.

   A new section with exactly the same parameters, parameter values and comments of the section you are duplicating is created.

### 10.8.2.5 Deleting a Named Configuration

When you delete a named configuration section, you delete *all* the information within it. If you only want to delete specific parameters, see Section 10.8.3, "Managing Parameters".

**To delete a named configuration:**

1. From the **JVM Configuration** region, select the row of the configuration section you want to delete.

2. Click **Delete**.

   The **Confirmation** dialog appears.

3. Click **Delete**.

   The configuration section is deleted.

   Oracle Enterprise Manager returns to the **JVM Configuration** page and displays the remaining configurations.

> **Note:** You cannot delete the Default configuration section.

## 10.8.3 Managing Parameters

Use Fusion Middleware Control to manage parameters within a named configuration. You can add, edit, or delete parameters using Fusion Middleware Control.

**To edit a parameter in a configuration section:**

1. From the **JVM Configuration** region, select the row of the configuration section that contains the parameter(s) you want to edit.

2. Select the row of the parameter you want to edit. Enter the Value and Comments.

3. Click **Apply** to save the changes or **Revert** to discard them.

**To add a parameter to a configuration section:**

1. In Fusion Middleware Control, from the **JVM Configuration** region, select the configuration section row for which you want to add a parameter.

2. Click Add to add a new parameter.

   The Add dialog box is displayed.

3. Enter the Name, Value and Comments for the parameter.

4. Click **Create** to add the parameter.

5. Click **Apply** to save the changes or **Revert** to discard them.

**To delete a parameter in a configuration section:**

1. In Fusion Middleware Control, from the **JVM Configuration** region, select the configuration section from which you want to delete a parameter.

2. Select the row that contains the parameter you want to delete.

3. Click **Delete**.

4. Click **Apply** to save the changes or **Revert** to discard them.

## 10.8.4 JVM Configuration Parameters and Default Values

Table 10–4 describes the JVM configuration parameters and their default values.

*Table 10–4    JVM Configuration Parameters*

| Parameter | Description | Default Value |
|-----------|-------------|---------------|
| Maximum Sessions per JVM | Specifies the maximum number of concurrent Oracle Forms sessions the default JVM will serve before a new JVM is spawned. | 65535 |
| Classpath | When you specify a classpath, it will override the system classpath or any classpath specified in your environment. | `$ORACLE_HOME/jdk/bin/java` |
| JVM Options | Enter any valid options to pass to the JVM. Refer to the Oracle Java documentation for a list of valid JVM startup parameters. | Null |

**Table 10–4 (Cont.) JVM Configuration Parameters**

| Parameter | Description | Default Value |
|---|---|---|
| Log Directory | Leave Log Directory blank to use the log location for the default JVM controller. If any other directory is set, the log file cannot be viewed through Enterprise Manager. | `$ORACLE_ INSTANCE/FRCompone nt/frcommon/tools/ jvm/log` |
| Logging | Specifies whether logging is enabled or not. Valid values: On, Off. | `On` |
| Comment | Add any comments about this default JVM in this text area. | `Null` |

## 10.8.5 Starting and Stopping JVM Controllers with Fusion Middleware Control

Fusion Middleware Control is the recommended tool for managing Oracle Forms Services, such as starting, stopping, and restarting a JVM controller.

If a JVM controller is down, you can start it. If a JVM controller is already running, you can restart it without first having to manually stop it. Fusion Middleware Control does this step for you.

> **Note:** Ensure that users have stopped the forms sessions that are using the JVM controller before you stop or restart the JVM. Users may want to restart sessions when the JVM is restarted.

**To access the JVM Controller page:**

1. Start the Enterprise Manager Fusion Middleware Control.

2. From the Forms home page, select **JVM Controller**s.

   The **JVM Controllers** page () is displayed.

**Figure 10–5 JVM Controller Page**
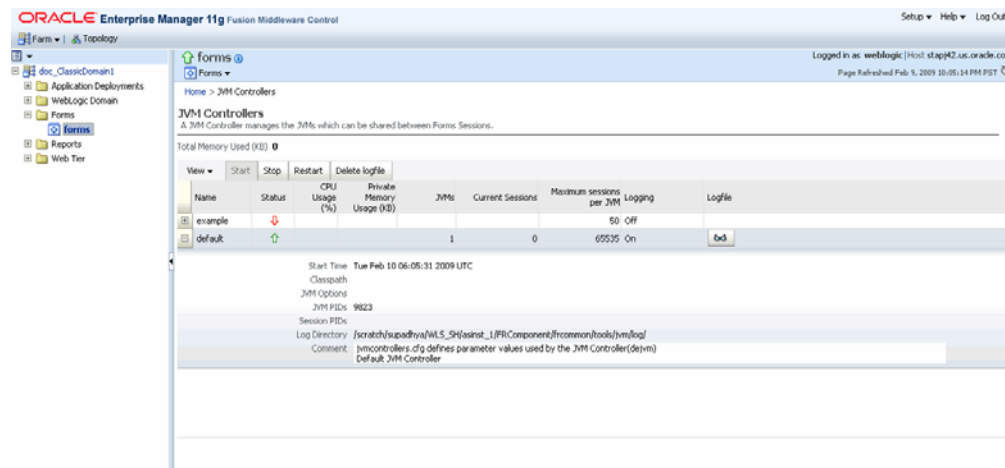


**To start a JVM controller that is not running:**

1. From the Forms menu, select **JVM Controllers**.

   The **JVM Controllers** page is displayed.

2. Select the JVM controller that you want to start. A JVM that is not running is indicated by a red, down arrow.

3. Click **Start**.

   When the JVM controller has started, a green, up arrow (Figure 10–5) is displayed in the Status.

**To restart a running JVM controller:**

1. From the Forms menu, select **JVM Controllers**.

   The **JVM Controllers** page is displayed.

2. Select the JVM controller to be restarted.

3. Click **Restart**.

4. Click **Yes** on the Confirmation dialog.

   The **JVM Controller page** reappears.

   When the JVM controller has restarted, a green, up arrow is displayed in the Status.

**To stop a JVM Controller**

1. From the Forms menu, select **JVM Controllers**.

   The **JVM Controllers** page is displayed.

2. Select the running JVM controller that you want to stop, indicated by a green, up arrow.

3. Click **Stop**.

4. Click **Yes** on the Confirmation dialog.

   When the JVM controller has been stopped, a red, down arrow (Figure 10–5) is displayed in the Status.

**To view additional details of a JVM Controller**

1. From the Forms menu, select **JVM Controllers**.

   The **JVM Controllers** page is displayed.

2. Click the plus symbol next to the JVM controller. The row is expanded to display additional details (Figure 10–5) of the JVM controller.

## 10.8.6 Forms Configuration File Settings

This section describes the JVM pooling parameters that are used in the Forms configuration file (`formsweb.cfg`) to enable or disable use of JVM controller for applications. The parameter names are not case-sensitive. You can use Fusion Middleware Control to administer the Forms configuration file.

Table 10–5, " Oracle Forms JVM Controller Startup Parameters" describes the startup options that you specify in the `formsweb.cfg` file.

For more information on modifying the parameters in `formsweb.cfg`, see Section 4.2.4, "Managing Parameters".

*Table 10–5    Oracle Forms JVM Controller Startup Parameters*

| Parameter | Description |
| --- | --- |
| jvmcontroller | Valid values: name of jvmcontroller. In addition, you can specify no JVM by leaving it blank. |
| | Default value: none |
| | **Note**: In order to specify this parameter in formsweb.cfg, you must first specify this parameter in otherparams in the form jvmcontroller=%jvmcontroller%. For more information on otherparams, see Table 4–13, " Advanced Configuration Parameters". |
| | This parameter can be set globally in the default section, or any application section can choose to override it. This tells the Forms runtime process which JVM controller to use. It corresponds to the jvmcontroller parameter for the dejvm executable. |
| | If jvmcontroller does not have a value (jvmcontroller=), then the Forms runtime process will start its own in-process JVM, which means that the Java Importer uses pre-10*g* behavior. |
| allowJVMControllerAutoStart | Valid values: true, false |
| | Default value: true |
| | This parameter enables Oracle Forms to run the JVM controller if Forms is configured to use the JVM controller which is not already running. |

## 10.8.7  Startup Example

This example illustrates an environment of multiple JVMs for multiple applications.

As shown in Table 10–6, formsweb.cfg is configured with four configuration sections.

*Table 10–6    Multiple JVMs for Multiple Applications*

| Named Configuration Section | JVM Configuration |
| --- | --- |
| default | jvmcontroller=commonJVM |
| ordersApp | None |
| hrApp | jvmcontroller=hrJVM |
| salesApp | jvmcontroller= |

If a user starts an ordersApp application, and the application executes Java code, the Forms runtime process will route the request to the JVM controller named commonJVM. Because the [ordersApp]  application section does not specify which JVM controller to use, the Forms runtime process uses the global one. If the JVM controller is not started, it will be dynamically started. If a second user starts the same application, it too will attach to commonJVM.

When a user starts an hrApp application and it executes Java code, the Forms runtime process sends the request to the JVM controller named hrJVM because the [hrApp] application section overrides the global setting. If the JVM controller is not started, it will be dynamically started. When a second user starts the same application, it too will attach to hrJVM.

When a user starts a `salesApp` application and it executes Java code, the Forms runtime process starts an in-process JVM in the same way the Java Importer works without JVM pooling. When a second user starts the same application, the application will get their own in-process JVM, thus consuming more memory, as shown in Figure 10–6:

*Figure 10–6   Multiple JVMs for multiple applications*
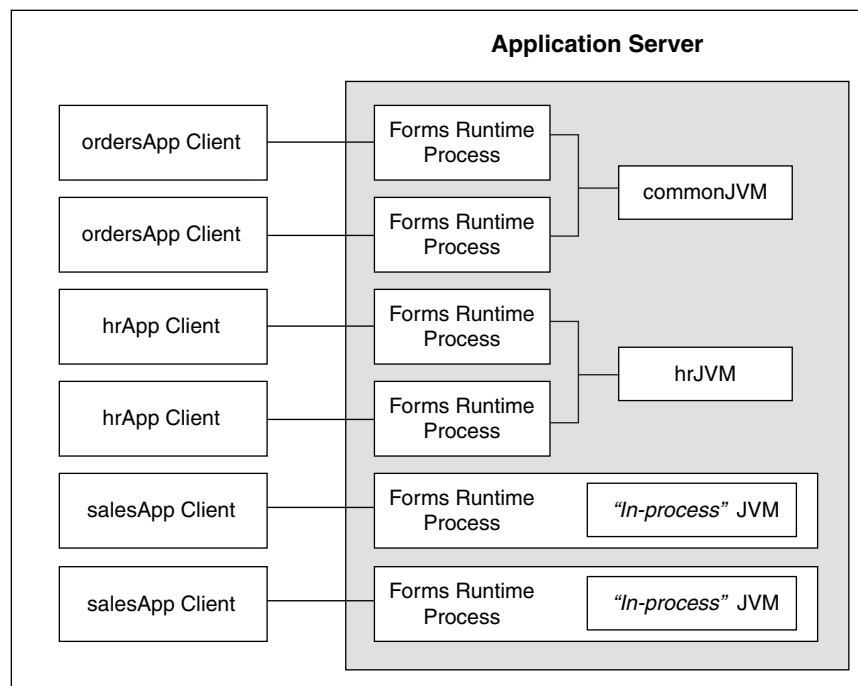


## 10.9   JVM Controller Logging

When logging is enabled, the JVM controller logs certain information to the log file:

- The values of the JVM parameters (maxsessions, classpath, and so on);

- When a JVM controller starts and stops;

- When a child JVM is spawned;

- When an Forms runtime process starts a new connection, along with its process ID

  This is useful for knowing which Forms runtime processes are connected to which JVM controller for diagnostics or administration;

- When an Forms runtime process session ends and disconnects from the JVM.

This section contains the following:

- Section 10.9.1, "Specifying JVM Default Logging Properties"

- Section 10.9.2, "Specifying the JVM Log Directory Location"

- Section 10.9.3, "Accessing Log Files"

- Section 10.9.4, "Deleting a Log File for a JVM Controller"

### 10.9.1  Specifying JVM Default Logging Properties

Use Fusion Middleware Control to manage the properties for JVM controller logging.

1. In the **JVM Configuration** page, select the the JVM configuration section.

2. For the Logging parameter, enter **On** or **Off**.

3. Click **Apply**.

## 10.9.2 Specifying the JVM Log Directory Location

You can specify the log file directory in the JVM controller. You can also specify the default JVM controller log file location for other JVM controllers to use.

**To specify the log file directory location:**

1. Create a JVM controller. For more information, see Section 10.8.2.2, "Creating a New Configuration Section" or Section 10.8.2.4, "Duplicating a Named Configuration".

2. Add the **Log Directory** parameter. For more information, see Section 10.8.3, "Managing Parameters."

   If you have duplicated a named configuration section that has **Log Directory** parameter defined in it, you can edit the existing parameter as given in the Section 10.8.3, "Managing Parameters."

3. Click **Apply** to save the changes.

   The **JVM Configuration** page reappears.

## 10.9.3 Accessing Log Files

When the log file exists, an icon is displayed in the Logfile column.

**To access a log file:**

- Click the Log File link in the Logfile column that is available for that JVM controller.

   The Log File page appears and displays the log information.

## 10.9.4 Deleting a Log File for a JVM Controller

Use Fusion Middleware Control to delete log files.

**To delete a log file for a JVM controller:**

1. From the **JVM Controllers** page, select the the target JVM.

2. Click **Delete Logfile**.

   The Delete Confirmation dialog appears.

3. Click **Delete**.

   The logfile is deleted and the **JVM Controllers** page reappears.

---

**Note:**   If you delete a log file of a JVM that is running, the log file will be available again when the JVM is restarted. Logging is possible only when the JVM is restarted.

---

## 10.10  Integrating Forms and Reports

JVM Controller (`dejvm`) is used for Reports integration in Forms. All requests related to Reports such as running a report on Reports Server, getting the status of a Report, getting Reports output, or cancelling the job submitted to Reports Server are routed to the dejvm for dejvm-enabled runform to make calls to Reports.

To use `dejvm` for reports integration, perform the following steps. These settings are not required when Oracle Forms makes Reports call directly.

**To use dejvm for Reports integration:**

1. Enable JVM pooling in `formsweb.cfg`. For more information, see Section 10.8.6, "Forms Configuration File Settings".

2. Two additional `.jar` files are required by `dejvm` for Reports integration. Set the classpath in `jvmcontrollers.cfg` to include these jars: `zrcclient.jar` (`$ORACLE_HOME/jlib/zrclient.jar`) and `rwrun.jar` (`$ORACLE_HOME/reports/jlib/rwrun.jar`).

---

**Note:**  If you want the Reports Server name to be returned when running a report using JVM controller, then the `REPORTS_SERVERMAP` environment variable must be defined in `jvmcontrollers.cfg` as shown below:

```
[myjvm]
jvmoptions=-DREPORTS_SERVERMAP=<value> <other-jvmoption-parameters>
```

---

## 10.11  JVM Pooling Error Messages

**PDE-JM001: Unable to communicate with the JVM Controller: <jvm_name>.**

**Cause:** Failed to start the JVM controller or connect to an existing JVM controller.

**Action:** Notify your administrator.

# 11

# Forms Services Security Overview

The ability to control user access to Web content and to protect your site against people breaking into your system is critical. This chapter describes the architecture and configuration of security for Oracle Forms Services:

- Section 11.1, "Forms Services Single Sign-On"

- Section 11.2, "Configuring Oracle Forms Services Security"

> **See Also:** For additional information about security, refer to the following documents:
>
> - *Oracle Fusion Middleware Security Overview* provides an overview of Oracle Fusion Middleware security and its core functionality.
>
> - *Oracle Fusion Middleware Getting Started with Oracle Identity Management* provides guidance for administrators of the Oracle security infrastructure.

## 11.1 Forms Services Single Sign-On

Single Sign-on in Oracle Forms Services is available through `mod_osso` or `webgate`, Oracle modules for the Oracle HTTP Server. `mod_osso` authenticates a user against Oracle Single Sign-On Server or Oracle Access Manager (OAM) 11*g*, which in turn uses Oracle Internet Directory as a user repository, before further passing the Forms application request to the Forms servlet. The `webgate` access client authenticates a user against Oracle Access Manager (OAM) 11*g*.

Forms applications expect a database connect string to be passed along with the application request, otherwise a logon dialog is shown. To retrieve the database connect information in the Single Sign-On environment, the Forms servlet queries Oracle Internet Directory for the value of the combined unique key that is constructed from the user's single sign-on server name, the authenticated user name, and the name of the application that the user is requesting to start.

Resource Access Descriptors (RAD) are entries in Oracle Internet Directory that are defined for each user and application which contain the required database connect information. The Forms servlet reads the database connect information from the RAD and passes it along with the command line that starts the Forms Web application. Although the Forms authentication is still database-centric, `mod_osso` or `webgate` and the Forms servlet are now integrated in a Web-based authentication server environment.

### 11.1.1 Classes of Users and Their Privileges

Historically, Forms applications use the database to authenticate application users. To use Oracle Forms Services with Single Sign-On (SSO), the user account and its connect information must be available in Oracle Internet Directory. Oracle Internet Directory provides several ways of provisioning user data, using PL/SQL, Java or the Oracle Delegated Administration Services. Oracle Delegated Administration Services is a Web-based user interface for Oracle Single Sign-On Server users and delegated administrators to administer self-service data in Oracle Internet Directory for which they are authorized.

Once a user account is created in Oracle Internet Directory, the Resource Access Descriptors (RAD) entries can be created dynamically the first time that a user requests a Forms application, assuming the user knows about the database connect information required for this application.

Another option is to use the RAD entries that can be created using Oracle Delegated Administration Services. The default RAD entries are accessible for all users that are authenticated through Oracle Single Sign-On Server. Use the default RAD if all users share the same database connect information when running a particular Forms application on the Web. This way, users are authenticated individually by their Oracle Single Sign-On Server credentials; however, all users share a common database connect (information) for the application defined by a default RAD entry.

#### 11.1.1.1 Default Single Sign-On Behavior for User Accounts

By default, the authentication server is enabled and no proxy user is involved. Oracle Forms users need to authenticate with an authentication server, retrieve Resource Access Descriptors from the identity store (which is usually Oracle Internet Directory) and use these credentials to connect to the database.

#### 11.1.1.2 Users Using Database Proxy Functionality

There is a new Single Sign-On parameter, `ssoProxyConnect`. Setting this to `true` allows users to connect as proxy users. The user is then required to authenticate with an authentication server; a Resource Access Descriptor is configured which holds the proxy user's username and password. There is additional database configuration that needs to be implemented by the database administrator to allow for proxy connections.

### 11.1.2 Resources That Are Protected

When you enable Single Sign-On for your Forms applications, you can secure your Forms applications with these features:

#### 11.1.2.1 Dynamic Directives

The dynamic `mod_osso` directive runs Single Sign-On protected Forms applications. This directive can optionally be used to run non-protected Forms applications from the same Oracle Forms Services instance. These applications use the same configuration files and Forms servlet. Single sign-on is enabled for applications by a Single Sign-On parameter in the application definition of the `formsweb.cfg` configuration file.

#### 11.1.2.2 Dynamic Resource Creation in Oracle Internet Directory

In some previous releases of Oracle Forms Services, if no resource access descriptor (RAD) definition was found for a specific application and user, an error message was displayed which locked out the user from running that Forms application, despite having authentication to do so. In this release of Oracle Forms Services, you can now

configure Oracle Forms Services to allow users to create the RAD for this application on the fly if it does not exist. The funtionality to redirect to DAS pages is achieved with the single sign-on parameter `ssoDynamicResourceCreate`.

### 11.1.2.3 Database Password Expiration when Using Single Sign-On

In some previous releases of Oracle Forms Services, the RAD information in Oracle Internet Directory was not updated if the database password had expired, and users then renewed them when connecting to a Forms application. In this release, Oracle Forms Services automatically updates the RAD information in Oracle Internet Directory whenever a database password is updated through Forms. There is no extra configuration necessary to enable this feature in Oracle Forms Services.

## 11.1.3 Authentication and Access Enforcement

For detailed information about the authentication flow in Single Sign-On support in Oracle Forms Services, such as when the first time the user requests an Oracle Forms Services URL, or from a partner application, see Section 9.1.2, "Authentication Flow".

## 11.1.4 Leveraging Oracle Identity Management Infrastructure

Oracle Forms Services has tighter integration with Oracle Internet Directory with minimal configuration. When you configure an authentication server for your Forms applications, Oracle Forms Services handles much of the configuration and interaction with Oracle Internet Directory.

With the absence of Repository API in 11*g*, Oracle Forms and Identity Management integration involves the registration of Forms application identity at the time of deployment when a relationship is established between Forms and the Oracle Internet Directory (OID) host. This process is known as associating with the Oracle Internet Directory. Related information such as Forms Distinguished Name (formsDN) and the password are stored in Credential Storage Framework (CSF).  At run time, a JNDI connection is made to Oracle Internet Directory after extracting the required information from CSF. Oracle Forms and Identity Management integration involves the following:

- Integration at bootstrap: The Forms application entity (and Distinguished Name) with a password is created in Oracle Internet Directory.

- Integration at run time: Previously, the connection to Oracle Internet Directory used Repository API. In 11*g*, a JNDI call is used to directly connect to Oracle Internet Directory.

For more information about associating and disassociating Oracle Internet Directory, see Section 9.7, "Postinstallation Configuration."

# 11.2 Configuring Oracle Forms Services Security

Configuring security for Oracle Forms Services is done through Oracle Fusion Middleware Control. Online help is available for each screen. For more information, see Chapter 4, "Configuring and Managing Forms Services" and Chapter 9, "Using Forms Services with Oracle Single Sign-On".

### 11.2.1 Configuring Oracle Identity Management Options for Oracle Forms

Oracle Forms Services can be configured to create resources dynamically in Oracle Internet Directory, or have a user with no Oracle Internet Directory resource use a common resource.

For more information, see Chapter 9, "Using Forms Services with Oracle Single Sign-On".

### 11.2.2 Configuring Oracle Forms Options for Oracle Fusion Middleware Security Framework

For more detailed information about configuring and securing Oracle Forms, see the following chapters:

- Chapter 3, "Basics of Deploying Oracle Forms Applications"
- Chapter 4, "Configuring and Managing Forms Services"
- Chapter 9, "Using Forms Services with Oracle Single Sign-On"
- Chapter 12, "Tracing and Diagnostics"

### 11.2.3 Securing RADs

To increase the security of RADs and prevent them from being viewable by the OID administrator, perform the following steps:

1. Copy the contents enclosed by `---aci-change.ldif---` into the file `aci-change.ldif`

   ```
   ---aci-change.ldif---
   dn: cn=Extended Properties,%s_OracleContextDN%
   changetype: modify
   delete: orclaci
   orclaci: access to attr=(orclUserIDAttribute,orclPasswordAttribute) by
   guidattr=(orclOwnerGUID)(read,search,compare,write) by
   dnattr=(orclresourceviewers) (read,search, compare, write) by
   groupattr=(orclresourceviewers) (read,search, write) by * (none)
   -
   add: orclaci
   orclaci: access to attr=(orclUserIDAttribute,orclPasswordAttribute)
   DenyGroupOverride by guidattr=(orclOwnerGUID)(read,search,compare,write) by
   dnattr=(orclresourceviewers) (read,search, compare, write) by
   groupattr=(orclresourceviewers) (read,search, write) by * (none)
   ---aci-change.ldif---
   ```

   > **Note:** In `aci-change.ldif`, the line beginning with `orclaci: access to attr=` is a single line ending with `by * (none)` and should not have any line breaks in the middle.

2. In the LDIF file, replace `%s_OracleContextDN%` with the distinguished name (DN) of the realm-specific Oracle Context.

   For example, if the DN in the deployment is `dc=acme,dc=com`, then the realm-specific Oracle Context is `cn=OracleContext,dc=acme,dc=com`.

3. Execute the following command on the OID tier:

```
ldapmodify -p <port> -h <host> -D cn=orcladmin -q -v -f
aci-change.ldif
```

4. When this command is run, it will prompt for the `cn=orcladmin` password since the password is not included as a command-line parameter.

To undo these changes, issue the same command (subject to the notes as above), but using the following contents in the `.ldif` file:

```
---aci-revert.ldif---
dn: cn=Extended Properties,%s_OracleContextDN%
changetype: modify
delete: orclaci
orclaci: access to attr=(orclUserIDAttribute,orclPasswordAttribute)
DenyGroupOverride by guidattr=(orclOwnerGUID)(read,search,compare,write) by
dnattr=(orclresourceviewers) (read,search, compare, write) by
groupattr=(orclresourceviewers) (read,search, write) by * (none)
-
add: orclaci
orclaci: access to attr=(orclUserIDAttribute,orclPasswordAttribute) by
guidattr=(orclOwnerGUID)(read,search,compare,write) by
dnattr=(orclresourceviewers) (read,search, compare, write) by
groupattr=(orclresourceviewers) (read,search, write) by * (none)
---aci-revert.ldif---
```

# 12

# Tracing and Diagnostics

This chapter contains the following sections:

## 12.1 About Forms Trace

Forms Trace allows you to record information about a precisely defined part of forms functionality or a class of user actions. This is accomplished by defining events for which you want to collect trace information. For example, you can record information about trigger execution, mouse-clicks, or both. From the Enterprise Manager Fusion Middleware Control, you can use trace output to diagnose performance and other problems with Oracle Forms applications.

Forms Trace replaces the functionality that was provided with Forms Runtime Diagnostics (FRD) and Performance Event Collection Services (PECS), which were available in earlier releases of Oracle Forms. Forms Trace allows you to trace the execution path through a form, for example, the steps the user took while using the form.

### 12.1.1 What Is the Difference between Tracing and Debugging?

You use Forms debugging to find out what happens when a user presses a button. Debugging allows a remote developer to connect to an existing Forms user session and to trace the user actions as the application runs or to debug on a local machine. Forms Trace provides information about the timing of specific events. Oracle Support uses tracing to isolate and analyze issues. For example, you use Forms trace to find out which query takes the longest time to execute, or which trigger causes performance issues with Oracle Forms.

## 12.2 Enabling and Configuring Forms Trace

An *event* is something that happens inside Oracle Forms as a direct or indirect result of a user action. An example is when a user presses a button that executes a query. An

*event set* specifies a group of events that you can trace simply by specifying the event set name rather than each event number individually when you start the trace.

Use the **Trace Configuration** selection in the **Forms** menu of Oracle Enterprise Manager page to define the events that you want to trace. This page manages all changes in the `ftrace.cfg` file for you.

Keep these items in mind when working with Forms Trace:

- If you first switch off trace, and then switch it on again with new settings, then trace is enabled with the new trace group.

- In order to trace Forms Processes on Windows, the Process Manager Service needs to have the check box "Allow service to interact with the desktop" selected. When this is not set, attempting to switch on Trace will result in the error: `oracle.sysman.emSDK.emd.comm.RemoteOperationException. Check the User Name and Password.`

- Backup the `ftrace.cfg` and `default.env` files before editing them with Fusion Middleware Control.

- As with most Web applications, it is easy to lose unsaved changes by switching pages. Be sure to save any changes you make through Fusion Middleware Control to Forms configuration, trace, or environment files before proceeding to other pages.

  The length of time it takes for changes to be saved is affected by the number of lines you have changed. For example, an additional fifty lines of comments will take longer to save than just deleting a single entry.

See Section 12.5, "List of Traceable Events" for a list of events and their corresponding event numbers.

## 12.2.1 Configuring Forms Trace

**To access the Trace Configuration page:**

1. Start Fusion Middleware Control.

2. From the Fusion Middleware Control main page, click the link to the Oracle Forms Services instance that you want to configure.

3. From the Forms menu list, select **Trace Configuration**.

   The **Trace Configuration** page (Figure 12–1) is displayed.

*Figure 12–1   Trace Configuration Page*
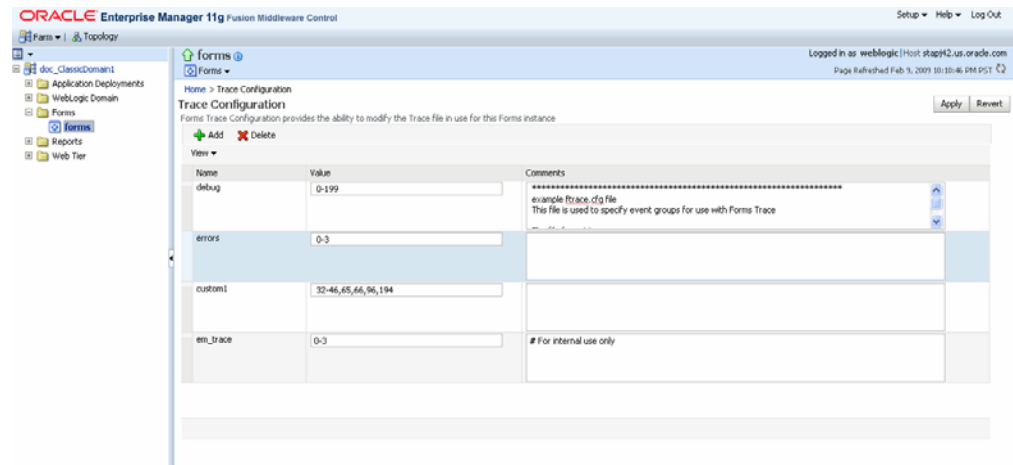


**To create a new trace group:**

1. From the Fusion Middleware Control main page, click the link to the Oracle Forms Services instance that you want to configure.

2. From the Forms menu list, select **Trace Configuration**.

   The **Trace Configuration** page is displayed.

3. Click **Add**.

   The **Add** dialog is displayed.

4. Enter the information for the new trace group:

   **Name**: Enter a name for the trace group.

   **Value:** See Table 12–2 for the values of traceable events.

   **Comment** : Enter a comment.

   - The trace group name must not contain spaces. For example, a_b_c is an acceptable trace group name.

   - There must be a comma between each event number you specify in the Value. For example, 65,66,96,194 is an acceptable value.

   - You can use a range of numbers. For example, 32-46 is an acceptable range.

5. Click **Add**.

   The new trace group is added.

6. Click **Apply** to save the changes, or **Revert** to discard them.

**To delete a trace group:**

1. In the **Trace Configuration** page, select the group you want to delete.

2. Click **Delete**.

   The trace group is deleted and the **Trace Configuration** page reappears.

3. Click **Apply** to save the changes, or **Revert** to discard them.

**To edit an existing trace group:**

1. In the **Trace Configuration** page, select the group you want to edit.

2. Enter the value and description for the trace group.

3. Click **Apply** to save the changes, or **Revert** to discard them.

## 12.2.2 Specifying URL Parameter Options

The following command line parameters are used to configure Forms Trace:

```
Record =
Tracegroup =
Log = <filename>
```

Table 12–1 describes the parameter values:

*Table 12–1    Forms Trace Command Line Parameters*

| Parameter | Values | Description |
|---|---|---|
| Record | forms | Enables Forms Trace. |
| Tracegroup | Name, event number, or event range | Indicates which events should be recorded and logged. |
| | | ■ If Tracegroup is not specified, only error and Startup messages are collected. |
| | | ■ Tracegroup is ignored if Forms Trace is not switched on at the command line. |
| | | ■ You can create a named set of events using the Tracegroup keyword, for example |
| | | Tracegroup=<keyword>, where <keyword> is specified in ftrace.cfg (for example, Tracegroup=MyEvents). |
| | | This lets you log the events in the named set MyEvents. |
| | | ■ You can log all events in a specified range using the Tracegroup keyword, for example |
| | | Tracegroup = 0-3 |
| | | This lets you log all events in the range defined by 0 <= event <=3. |
| | | ■ You can log individual events using the Tracegroup keyword, for example |
| | | Tracegroup = 34,67 |
| | | ■ You can combine event sets using the Tracegroup keyword, for example |
| | | Tracegroup = 0-3,34,67,SQLInfo |

## 12.3  Starting and Stopping Forms Trace

You start a trace by specifying trace entries in the URL or from Fusion Middleware Control. Entries should include the grouping of events to collect and the trace file name. Trace collection starts when the form executes.

The following are sample URLs to start a trace:

```
http://example.com/forms/frmservlet?form=cxl&record=forms&tracegroup=0-199
http://example.com/forms/frmservlet?form=cxl&record=forms&tracegroup=mysql
```

**To start tracing a session from Fusion Middleware Control:**

1. From the Forms menu, select **User Sessions**.

   The **User Sessions** page appears.

2. Select the row containing the Forms user session for which you want to enable tracing.

3. Click **Enable Tracing**.

   The Enable Tracing dialog appears.

4. From the Select Trace Group list, select an available trace group and click **OK**.

   The Enable Tracing dialog is dismissed and tracing is now enabled for the selected Forms user session.

**To stop tracing a session from Fusion Middleware Control:**

1. From the Forms menu, select **User Sessions**.

   The User Sessions page appears.

2. Select the row containing the Forms user session for which you want to disable tracing.

3. Click **Disable Tracing**.

   The Disable Tracing dialog is displayed.

4. Click **OK**.

   The Disable Tracing dialog is dismissed and tracing is now stopped for the selected Forms user session.

**To switch between trace groups for a session:**

1. Select the row containing the Forms user session for which you want to change the trace group.

2. Click **Enable Tracing**.

   The Enable Tracing dialog is displayed.

3. From the Select Trace Group list, select the new trace group and click **OK**.

   The Enable Tracing dialog is dismissed. Refresh the page.

## 12.4 Viewing Forms Trace Output

Only administrators or a user belonging to administrators' group can view trace log files. Once the user has logged in, he or she does not have to log in again in the same browser session to view trace log files for different sessions.

Trace data is stored in a binary file with a `*.trc` extension. The default location of the trace log is `$ORACLE_INSTANCE/FormsComponent/forms/trace/forms_pid.trc` where pid is the process ID of the user session. If you are not using Enterprise Manager Fusion Middleware Control, you need to use the Translate utility. For more information on running the Translate Utility, see the next section Section 12.4.1, "Running the Translate Utility."

**To view trace data:**

1. From the Forms menu in Fusion Middleware Control, select the **User Sessions** menu item.

2. Select a User Session row and click **Trace Log** to see the contents of the trace log.

3. Log in to view the trace file.

### 12.4.1 Running the Translate Utility

The Translate utility converts trace data to XML, HTML, or text formats. You need to specify an additional parameter "OutputClass" which has three legal values: "WriteOutTEXT", "WriteOutXML" and "WriteOutHTML". If you do not specify the `outputclass`, the output file is in text format. These values are case-sensitive.

> **Note:** To use the Translate Utility:
>
> 1. Set the PATH variable to include the path to the directory containing the Java executable.
>
> 2. Set the CLASSPATH variable to include the path to frmxlate.jar.

**To convert trace data to Text format:**

- At the command line, enter:

  ```
  java oracle.forms.diagnostics.Xlate datafile=a.trc
  outputfile=myfile.txt outputclass=WriteOutTEXT
  ```

  This creates a file called myfile.txt in text format.

**To convert trace data to HTML format:**

- At the command line, enter:

  ```
  java oracle.forms.diagnostics.Xlate datafile=a.trc
  outputfile=myfile.html outputclass=WriteOutHTML
  ```

  This creates a file called myfile.html in HTML format.

**To convert trace data to XML format:**

- To create `myfile.xml`, at the command line, enter:

  ```
  java oracle.forms.diagnostics.Xlate datafile=a.trc
  outputfile=myfile.xml outputclass=WriteOutXML
  ```

  This creates a file called myfile.xml in XML format.

## 12.5 List of Traceable Events

Table 12–2, " List of Traceable Events" lists the events that can be defined for tracing. In future releases of Forms, more events may be added to this list.

Event types are as follows:

- Point event: An event that happens in Oracle Forms as the result of a user action or internal signal for which there is no discernible duration, for example, displaying an error message on the status line. Each instance of this event type creates one entry in the log file.

- Duration event: An event with a start and end, for example, a trigger. Each instance of this event type creates a pair of entries in the log file (a start and end event).

■ Built-in event: An event associated with a built-in. Each instance of this event type provides a greater quantity of information about the event (for example, argument values).

***Table 12–2    List of Traceable Events***

| Event Number | Definition | Type |
| --- | --- | --- |
| 0 | Abnormal Error | point |
| 1 | Error during open form | point |
| 2 | Forms Died Error | point |
| 3 | Error messages on the status bar | point |
| 4-31 | Reserved for future use | NA |
| 32 | Startup | point |
| 33 | Menu | point |
| 34 | Key | point |
| 35 | Click | point |
| 36 | Double-click | point |
| 37 | Value | point |
| 38 | Scroll | point |
| 39 | LOV Selection | point |
| 40 | not used | not used |
| 41 | Window Close | point |
| 42 | Window Activate | point |
| 43 | Window Deactivate | point |
| 44 | Window Resize | point |
| 45 | Tab Page | point |
| 46 | Timer | point |
| 47 | DB Event | point |
| 48 | Reserved for future use | NA |
| 49-63 | Reserved for future use | NA |
| 64 | Form (Start & End) | duration |
| 65 | Program Unit (Start & End) | duration |
| 66 | Trigger (Start & End) | duration |
| 67 | LOV (Start & End) | duration |
| 68 | Opening a Editor | point |
| 69 | Canvas | point |
| 70 | Alert | duration |
| 71 | GetFile | point |
| 72-95 | Reserved for future use | NA |
| 96 | Builtin (Start & End) | builtin |

*Table 12–2   (Cont.)  List of Traceable Events*

| Event Number | Definition | Type |
|---|---|---|
| 97 | User Exit (Start & End) | duration |
| 98 | SQL (Start & End) | duration |
| 99 | MenuCreate (Start & End) | duration |
| 100 | DB PU (Start & End) | duration |
| 101 | Execute Query | duration |
| 102-127 | Reserved for future use | NA |
| 128 | Client Connect | point |
| 129 | Client Handshake | point |
| 130 | Heartbeat | point |
| 131 | HTTP Reconnect | point |
| 132 | Socket (Start & End) | duration |
| 133 | HTTP (Start & End) | duration |
| 134 | SSL (Start & End) | duration |
| 135 | DB Processing (Start & End) | duration |
| 136 | DB Logon (Start & End) | duration |
| 137 | DB Logoff (Start & End) | duration |
| 138-159 | Reserved for future use | NA |
| 160-168 | Reserved for internal use | NA |
| 169-191 | Reserved for future use | NA |
| 192* | Environment Dump | N/A |
| 193* | State Delta | N/A |
| 194* | Builtin Arguments | N/A |
| 195* | UserExit Arguments | N/A |
| 196* | Program Unit Arguments | N/A |
| 256 and higher | User defined | NA |
| 1024 and higher | Reserved for internal use | NA |

* These event numbers do not have a TYPE because they are not really events, but rather details for events. For example, the State Delta is something you can choose to see - it is triggered by a real action or event.

## 12.5.1  List of Event Details

The following tables list event details that can be defined for tracing:

- Table 12–3, " User Action Event Details"

- Table 12–4, " Forms Services Event Details"

- Table 12–5, " Detailed Events"

> **Note:** Event names are case sensitive.

### 12.5.1.1  User Action Events

*Table 12–3    User Action Event Details*

| Action | Details | Number |
|---|---|---|
| Menu Selection | Menu Name, Selection | 33 |
| Key | Key Pressed, Form, Block, Item | 34 |
| Click | Mouse/Key, Form, Block, Item | 35 |
| DoubleClick | Form, Block, Item | 36 |
| Value | Form, Block, Item | 37 |
| Scroll | Form, Up, Down, Page, Row | 38 |
| LOV Selection | LOV Name, Selection Item | 39 |
| Alert | AlertName, Selection | 40 |
| Tab | Form | 45 |
| DB Event | Queue Name | 47 |
| Window Activate, Deactivate,Close, Resize | WindowName, FormName, Size | 41,42,43,44 |

### 12.5.1.2  Forms Services Events

*Table 12–4    Forms Services Event Details*

| Event Name | Details | Number |
|---|---|---|
| Form | Form ID, Name, Path, Attached Libraries, Attached Menus | 64 |
| Program Unit | Program Unit Name, FormID | 65 |
| Trigger | TriggerName, FormName, BlockName, ItemName, FormID | 66 |
| LOV | LOV name, FormId | 67 |
| Editor | FormId , Editor Name | 68 |
| Canvas | FormId , Canvas Name | 69 |

### 12.5.1.3  Detailed Events

*Table 12–5    Detailed Events*

| Event Name | Details | Number |
|---|---|---|
| Builtin | BuiltinName, FormId | 96 |
| User Exit | UserExitName, FormId | 97 |
| MenuCreate | MenuName, FormID | 99 |
| PLSQL | PLSQLSTmt, FormID | 100 |

*Table 12–5   (Cont.)  Detailed Events*

| Event Name | Details | Number |
|---|---|---|
| ExecQuery | Block Name | 101 |

### 12.5.1.4 Three-Tier Events

*Table 12–6    Three-Tier Event Details*

| Event Name | Details | Number |
|---|---|---|
| Client Connect | Timestamp | 128 |
| Client Handshake | Timestamp | 129 |
| Heartbeat | Timestamp | 130 |
| HTTP Reconnect | NA | 131 |
| Socket | FormId, Packets, Bytes | 132 |
| HTTP | FormId, Packets, Bytes | 133 |
| HTTPS | FormId, Packets, Bytes | 134 |
| DB Processing | FormId, Statement | 135 |
| DB Logon | FormId | 136 |
| DB Logoff | FormId | 137 |

### 12.5.1.5 Miscellaneous Events

*Table 12–7    Miscellaneous Event Details*

| Event Name | Details | Number |
|---|---|---|
| Environment Dump | Selected environment information | 192 |
| State Delta | Changes to internal state caused by last action/event | 193 |
| Builtin Args | Argument values to a builtin | 194 |
| Userexit args | Arguments passed to a userexit | 195 |
| Procedure Args | Arguments (in \| out) passed to a procedure. | 196 |

## 12.6  Taking Advantage of Oracle Diagnostics and Logging Tools

Oracle Diagnostics and Logging (ODL) is a feature of Oracle Fusion Middleware that enables administrators to keep a record of all Oracle Forms sessions, monitor Oracle Forms-related network traffic, and debug site configuration problems. Some of the features of Oracle Diagnostics and Logging available to Forms Services include:

- Recording of all Oracle Forms sessions, including session start and end times, and the user's IP address and host name (session-level logging)

- Monitoring of Oracle Forms-related network traffic and performance (session-performance and request-performance-level logging)

- Generating debugging information for site configuration issues (debug-level logging)

- Logging handled through Fusion Middleware Control

- Correlating events in these log files with events in the database

- Automatic handling of log file rotation.

- Handling of log size restriction by the mechanism rather than by OS level scripts as was done previously

These sections on the servlet logging tools contain the following:

- Section 12.6.1, "Enabling Oracle Diagnostics and Logging"

- Section 12.6.4, "Location of Log Files"

- Section 12.6.5, "Example Output for Each Level of Servlet Logging"

## 12.6.1  Enabling Oracle Diagnostics and Logging

When you turn on logging, the Listener Servlet writes log messages to the servlet log file. Examples of output for the various levels of logging are in Section 12.6.5, "Example Output for Each Level of Servlet Logging".

Table 12–8 describes the supported logging capabilities. If no string is appended to serverURL, then default logging is supported. To start other loggers, they must be specified in serverURL as described in the next section.

*Table 12–8    Supported logging capabilities*

| String appended to serverURL client parameter | Description of logging |
|---|---|
| (none) | During Forms servlet initialization, a message is written to the log file stating the name and path of the configuration file being used. Messages of levels higher and equal to the log level set for the default logger in logging.xml are logged. Default Value is set to NOTIFICATION:1 and levels NOTIFICATION:1, WARNING:1, ERROR:1 and INTERNAL_ERROR are logged. |
| /session | Log messages are written whenever a Forms session starts or ends. These give the host name and IP address of the client (the computer on which the user's Web browser is running), the runtime process id, and a unique internal session id number. |
| /sessionperf | Performance summary statistics are included with the session end message. |
| /perf | A performance message is written for every request from the client. |
| /debug | Full debug messages. Other debug messages are written in addition to the messages mentioned above. This logging level is verbose and is intended mainly for debugging and support purposes. |

### 12.6.1.1  Specifying Logging

To specify logging for all users, change the serverURL entry in the default section in the **Web Configuration** page to the following:

```
serverURL=/forms/lservlet/<string>
```

where <string> specifies the logging capability as defined in Table 12–8. If no string is provided, the default logging For example, if you want to start session-level logging, modify the serverURL as follows:

```
serverURL=/forms/lservlet/session
```

### 12.6.1.2 Specifying Logging Levels Using Fusion Middleware Control

To set the log levels for Forms servlet logging using Fusion Middleware Control, perform the following:

1. From the Fusion Middleware Control, select the managed server (for example WLS_FORMS).

2. From the WebLogic Server menu, select Logs, then Log Configuration.

3. In the Logger Name field, expand Root Logger. Expand each of the following: oracle, oracle.forms. The Logger name defined in serverURL as described in Section 12.6.1.1, "Specifying Logging" is displayed, for example (oracle.forms.servlet.debug).

4. Choose the Log level as required from the list in the Oracle Diagnostic Logging Level field. Refer to Table 12–9 for the mapping of the internal Forms log level to the Java levels.

*Table 12–9 Oracle Diagnostic Logging Levels*

| Internal Forms Log Levels | Java Log Levels |
|---|---|
| DEBUG | TRACE:32 |
| REQUEST_PERFORMANCE | TRACE:16 |
| SESSION_PERFORMANCE | TRACE:1 |
| SESSION_START_END | NOTIFICATION:16 |
| NOTIFICATION | NOTIFICATION:1 |
| WARNING | WARNING:1 |
| ERROR | ERROR:1 |
| INTERNAL_ERROR | INTERNAL_ERROR |

This configuration modifies the logging.xml file for the managed server.

### 12.6.1.3 Specifying Full Diagnostics in the URL that Invokes the Forms Servlet

To start full diagnostics, specify the parameter serverURL in `formsweb.cfg` as follows:

```
serverURL=/forms/lservlet/debug
```

Start the Oracle Forms application using a URL as follows:

```
http://example.com/forms/frmservlet/debug?
```

## 12.6.2 Viewing Diagnostics Logs

You view the contents of diagnostics logs from Fusion Middleware Control.

**To view the contents of diagnostics logs:**

1. From the Forms menu, select Home.

   The Fusion Middleware Control home page is displayed.

2. In the Forms Deployment region, scroll to the Servlet Logs column.

3. Click the corresponding Logs link for the target deployed application.

   The Log Messages page is displayed.

### 12.6.3 Using the Servlet Page

From the Forms menu, select Monitoring and then Servlet Logs. Use this page to search, sort, view, download, and export collected server diagnostics logs.

For additional information on managing and viewing the log files, refer to the *Oracle Fusion Middleware Administrator's Guide*.

### 12.6.4 Location of Log Files

The default servlet log file is named `formsapp-diagnostic.log`. It is written to the `WLS_FORMS/logs` directory of the Oracle WebLogic Managed Server to which Forms is deployed.

In Oracle Forms Services, the full path is:

```
$DOMAIN_HOME/servers/WLS_FORMS/logs/<application
name>-diagnostic.log
```

The trace logs are stored in files named `forms_pid.trc` by default, where `pid` is the process ID of the user session. The default location of the trace log is:

```
$ORACLE_INSTANCE/FormsComponent/forms/trace/forms_pid.trc
```

Use the Translate Utility described in Section 12.4.1, "Running the Translate Utility" to view them.

### 12.6.5 Example Output for Each Level of Servlet Logging

The following are examples of the type of output you get when you use the following levels of logging:

- (none)

- /session

- /sessionperf

- /perf

- /debug

#### 12.6.5.1 (none)

```
[2008-09-10T06:58:47.106-07:00] [WLS_FORMS] [NOTIFICATION] [FRM-93100]
 [oracle.forms.servlet] [tid: 11] [ecid: 0000HlCYKnmD4i8nvgy0V118lx4u000000,0]
 [APP: formsapp] [arg:
configFileName:     <configfilename>
testMode:           false] Initializing the Forms Servlet.  Initialization
 parameters are:[[
    configFileName:    <configfilename>
    testMode:
false
]]
[2008-09-10T06:58:53.517-07:00] [WLS_FORMS] [NOTIFICATION] [FRM-93180]
 [oracle.forms.servlet] [tid: 11] [ecid: 0000HlCZfTDD4i8nvgy0V118lx4u000003,0]
 [APP: formsapp] [arg:
envFile:          null
WorkingDirectory: null
executable:       null
WaitTime:         500
MaxBlockTime:     1000]
Initializing ListenerServlet.  Initialization parameters
```

```
are:[[
    envFile:          null
    WorkingDirectory:  null
    executable:        null
    WaitTime:          500
    MaxBlockTime:      1000
]]
```

### 12.6.5.2  /session

```
[2008-09-11T07:35:01.507-07:00] [WLS_FORMS] [NOTIFICATION:16] [FRM-93251]
[oracle.forms.servlet.session] [tid: 14] [ecid:
0000HlHpYGDD4i8nvgy0V118mFuv00000V,0] [SRC_CLASS:
oracle.forms.servlet.RunformSession] [APP: formsapp] [SRC_METHOD: <init>] [FORMS
SESSION_ID: ..8] [arg: supadhya-pc1] [arg: 10.177.254.46] Runtime session started
for client <pc1> (IP address <ip address>).
2008-09-11T07:35:01.798-07:00] [WLS_FORMS] [NOTIFICATION:16] [FRM-93548]
[oracle.forms.servlet.session] [tid: 14] [ecid:
0000HlHpYGDD4i8nvgy0V118mFuv00000V,0] [SRC_CLASS:
oracle.forms.servlet.RunformProcess] [APP: formsapp] [SRC_METHOD: connect] [FORMS
SESSION_ID: ..8] [arg: 7765] Runtime process ID is 7765.
2008-09-11T07:38:11.372-07:00] [WLS_FORMS] [NOTIFICATION:16] [FRM-93252]
[oracle.forms.servlet.session] [tid: 14] [ecid:
0000HlHpYGDD4i8nvgy0V118mFuv00000V,0] [SRC_CLASS:
oracle.forms.servlet.RunformSession] [APP: formsapp] [SRC_METHOD: stop] [FORMS
SESSION_ID: ..8] Forms session ended.
```

### 12.6.5.3  /sessionperf

```
[2008-09-11T07:40:25.923-07:00] [WLS_FORMS] [NOTIFICATION:16] [FRM-93251]
[oracle.forms.servlet.sessionperf] [tid: 17] [ecid:
0000HlHqlS9D4i8nvgy0V118mFuv00000Y,0] [SRC_CLASS:
oracle.forms.servlet.RunformSession] [APP: formsapp] [SRC_METHOD: <init>] [FORMS
SESSION_ID: ..9] [arg: <pc1>] [arg: 10.177.254.46] Runtime session started
for client <pc1> (IP address 10.177.254.46).
2008-09-11T07:40:26.223-07:00] [WLS_FORMS] [NOTIFICATION:16] [FRM-93548]
 [oracle.forms.servlet.sessionperf] [tid: 17] [ecid:
0000HlHqlS9D4i8nvgy0V118mFuv00000Y,0] [SRC_CLASS:
oracle.forms.servlet.RunformProcess] [APP: formsapp] [SRC_METHOD: connect] [FORMS
SESSION_ID: ..9] [arg: 8023] Runtime process ID is 8023.
2008-09-11T07:40:43.593-07:00] [WLS_FORMS] [NOTIFICATION:16] [FRM-93252]
[oracle.forms.servlet.sessionperf] [tid: 17] [ecid:
0000HlHqlS9D4i8nvgy0V118mFuv00000Y,0] [SRC_CLASS:
oracle.forms.servlet.RunformSession] [APP: formsapp] [SRC_METHOD: stop] [FORMS
SESSION_ID: ..9] Forms session ended.
[2008-09-11T07:40:43.594-07:00] [WLS_FORMS] [TRACE] [FRM-93710]
[oracle.forms.servlet.sessionperf] [tid: 17] [ecid:
0000HlHqlS9D4i8nvgy0V118mFuv00000Y,0] [SRC_CLASS:
oracle.forms.servlet.RunformSession] [APP: formsapp] [SRC_METHOD: stop] [FORMS
SESSION_ID: ..9] [arg: 1.557] [arg: 6] [arg: 0] [arg: 1.000] [arg: 0.259] [arg:
5106] [arg: 352] Total duration of network exchanges is 1.557.[[
Total number of network exchanges is 6 (0 long ones over 1.000 sec).
Average time for one network exchange (excluding long ones) is 0.259.
Total number of bytes sent is 5106.
Total number of bytes received is 352.
]]
```

### 12.6.5.4  /perf

```
[2008-09-11T07:42:46.560-07:00] [WLS_FORMS] [NOTIFICATION:16] [FRM-93251]
```

```
[oracle.forms.servlet.perf] [tid: 14] [ecid: 0000HlHrJmWD4i8nvgy0V118mFuv00000^,0]
[SRC_CLASS: oracle.forms.servlet.RunformSession] [APP: formsapp] [SRC_METHOD:
<init>] [FORMS_SESSION_ID: ..10] [arg: <pc1>] [arg: 10.177.254.46] Runtime
session started for client <pc1> (IP address <ip address>).
[2008-09-11T07:42:46.854-07:00] [WLS_FORMS] [NOTIFICATION:16] [FRM-93548]
[oracle.forms.servlet.perf] [tid: 17] [ecid: 0000HlHqlS9D4i8nvgy0V118mFuv00000Y,0]
[SRC_CLASS: oracle.forms.servlet.RunformProcess] [APP: formsapp] [SRC_METHOD:
connect] [FORMS_SESSION_ID: ..10] [arg: 8149] Runtime process ID is 8149.
[2008-09-11T07:42:46.865-07:00] [WLS_FORMS] [TRACE:16] [FRM-93700]
[oracle.forms.servlet.perf] [tid: 17] [ecid: 0000HlHqlS9D4i8nvgy0V118mFuv00000Y,0]
[SRC_CLASS: oracle.forms.servlet.ListenerServlet] [APP: formsapp] [SRC_METHOD:
doPost] [FORMS_SESSION_ID: ..10] [arg: 0.011] [arg: 8] [arg: 8] [arg: null]
Request duration is 0.011 seconds.  Request size is 8 bytes; response size is 8
bytes.
[2008-09-11T07:42:47.921-07:00] [WLS_FORMS] [TRACE:16] [FRM-93700]
[oracle.forms.servlet.perf] [tid: 17] [ecid: 0000HlHqlS9D4i8nvgy0V118mFuv00000Y,0]
[SRC_CLASS: oracle.forms.servlet.ListenerServlet] [APP: formsapp] [SRC_METHOD:
doPost] [FORMS_SESSION_ID: ..10] [arg: 0.438] [arg: 272] [arg: 5022] [arg: null]
Request duration is 0.438 seconds.  Request size is 272 bytes; response size is
5022 bytes.
```

### 12.6.5.5  /debug

```
[2009-02-11T14:39:03.016+00:00] [WLS_FORMS] [NOTIFICATION:16] [FRM-93250]
 [oracle.forms.servlet] [tid: [ACTIVE].ExecuteThread: '2' for queue:
 'weblogic.kernel.Default (self-tuning)'] [userId: <anonymous>] [ecid: 0000Hx
_lhDcD4i8nvgy0V119Xz350000HZ,0] [APP: formsapp#11.1.2] Forms session started.
[2009-02-11T14:39:03.017+00:00] [WLS_FORMS] [TRACE:32] [FRM-94200]
 [oracle.forms.servlet] [tid: [ACTIVE].ExecuteThread: '2' for queue:
'weblogic.kernel.Default (self-tuning)'] [userId: <anonymous>] [ecid: 0000Hx
_lhDcD4i8nvgy0V119Xz350000HZ,0] [SRC_CLASS: oracle.forms.servlet.FormsServlet]
 [APP: formsapp#11.1.2] [SRC_METHOD: doRequest] [FORMS_SESSION_ID: ..43] [arg:
 GET] [arg:
cmd:             frmservlet
config:          null
requestCharset:  null
QueryString:     null
Content-Type:    null
Accept-Charset:  null
responseCharset: null] FormsServlet receiving GET request.  Details:[[
    cmd:             frmservlet
    config:          null
    requestCharset:  null
    QueryString:     null
    Content-Type:    null
    Accept-Charset:  null
    responseCharset: null
]]
[2009-02-11T14:39:03.017+00:00] [WLS_FORMS] [TRACE:32] [FRM-94281]
 [oracle.forms.servlet] [tid: [ACTIVE].ExecuteThread: '2' for queue:
 'weblogic.kernel.Default (self-tuning)'] [userId: <anonymous>] [ecid: 0000Hx
_lhDcD4i8nvgy0V119Xz350000HZ,0] [SRC_CLASS: oracle.forms.servlet.ListenerServlet]
 [APP: formsapp#11.1.2] [SRC_METHOD: printSessionDetails] [FORMS_SESSION_ID: ..43]
 No current servlet session ID.
[2009-02-11T14:39:03.017+00:00] [WLS_FORMS] [TRACE:32] [FRM-94170]
 [oracle.forms.servlet] [tid: [ACTIVE].ExecuteThread: '2' for queue:
 'weblogic.kernel.Default (self-tuning)'] [userId: <anonymous>] [ecid: 0000Hx
_lhDcD4i8nvgy0V119Xz350000HZ,0] [SRC_CLASS: oracle.forms.servlet.FormsServlet]
 [APP: formsapp#11.1.2] [SRC_METHOD: findFile] [FORMS_SESSION_ID: ..43] [arg:
 basejpi.htm] [arg: <config folder>] File basejpi.htm is missing from the
```

```
                    current directory, looking in <config folder>
         [2009-02-11T14:39:21.460+00:00] [WLS_FORMS] [TRACE:32] [FRM-94200]
          [oracle.forms.servlet] [tid: [ACTIVE].ExecuteThread: '2' for queue:
          'weblogic.kernel.Default (self-tuning)'] [userId: <anonymous>] [ecid: 0000Hx
         _llhoD4i8nvgy0V119Xz350000Hd,0] [SRC_CLASS: oracle.forms.servlet.FormsServlet]
          [APP: formsapp#11.1.2] [SRC_METHOD: doRequest] [FORMS_SESSION_ID: ..43] [arg:
          GET] [arg:
         cmd:               startsession
         config:            null
         requestCharset:    null
         QueryString:
         ifsessid=..43&acceptLanguage=en-us&ifcmd=startsession&iflocale=en-US
             Content-Type:       null
         Accept-Charset:    null
         responseCharset:   null]
         FormsServlet receiving GET request.  Details:[[
             cmd:               startsession
             config:            null
             requestCharset:    null
             QueryString:
         ifsessid=..43&acceptLanguage=en-us&ifcmd=startsession&iflocale=en-US
             Content-Type:      null
             Accept-Charset:    null
             responseCharset:   null
         ]]


         .
         .
         .
         .

         [2009-02-11T14:39:21.716+00:00] [WLS_FORMS] [TRACE:32] [FRM-94201]
          [oracle.forms.servlet] [tid: [ACTIVE].ExecuteThread: '2' for queue:
          'weblogic.kernel.Default (self-tuning)'] [userId: <anonymous>] [ecid: 0000Hx
         _llloD4i8nvgy0V119Xz350000Hf,0] [SRC_CLASS: oracle.forms.servlet.ListenerServlet]
         [APP: formsapp#11.1.2] [SRC_METHOD: doGet] [FORMS_SESSION_ID: ..43] [arg: GET]
          [arg:
         cmd:               getinfo
         QueryString:       ifcmd=getinfo&ifhost=supadhya-pc1&ifip=10.177.254.239]
          ListenerServlet receiving GET request.  Details:[[
             cmd:               getinfo
             QueryString:       ifcmd=getinfo&ifhost=supadhya-pc1&ifip=10.177.254.239
         ]]

         [2009-02-11T14:39:21.717+00:00] [WLS_FORMS] [TRACE:32] [FRM-94282]
          [oracle.forms.servlet] [tid: [ACTIVE].ExecuteThread: '2' for queue:
          'weblogic.kernel.Default (self-tuning)'] [userId: <anonymous>] [ecid: 0000Hx
         _llloD4i8nvgy0V119Xz350000Hf,0] [SRC_CLASS: oracle.forms.servlet.ListenerServlet]
          [APP: formsapp#11.1.2] [SRC_METHOD: printSessionDetails] [FORMS_SESSION_ID: ..43]
          [arg:
          HyLhJSjZ85F5GWbZLDgwp1MY02FK5tC6yVDP1LylbCvgmv9y3CfK!126690176!1234363161461]
          Existing servlet session, ID =
          HyLhJSjZ85F5GWbZLDgwp1MY02FK5tC6yVDP1LylbCvgmv9y3CfK!126690176!1234363161461
         [2009-02-11T14:39:21.717+00:00] [WLS_FORMS] [TRACE:32] [FRM-94286]
          [oracle.forms.servlet] [tid: [ACTIVE].ExecuteThread: '2' for queue:
         'weblogic.kernel.Default (self-tuning)'] [userId: <anonymous>] [ecid: 0000Hx
         _llloD4i8nvgy0V119Xz350000Hf,0] [SRC_CLASS: oracle.forms.servlet.ListenerServlet]
          [APP: formsapp#11.1.2] [SRC_METHOD: printSessionDetails] [FORMS_SESSION_ID: ..43]
          Session ID is not from cookie.
```

```
[2009-02-11T14:39:21.717+00:00] [WLS_FORMS] [TRACE:32] [FRM-94430]
 [oracle.forms.servlet] [tid: [ACTIVE].ExecuteThread: '2' for queue:
 'weblogic.kernel.Default (self-tuning)'] [userId: <anonymous>] [ecid: 0000Hx
_llloD4i8nvgy0V119Xz350000Hf,0] [SRC_CLASS: oracle.forms.servlet.RunformSession]
 [APP: formsapp#11.1.2] [SRC_METHOD: <init>] [FORMS_SESSION_ID: ..43] Trying to
 get a prestarted process.
[2009-02-11T14:39:21.717+00:00] [WLS_FORMS] [TRACE:32] [FRM-94432]
 [oracle.forms.servlet] [tid: [ACTIVE].ExecuteThread: '2' for queue:
 'weblogic.kernel.Default (self-tuning)'] [userId: <anonymous>] [ecid: 0000Hx
_llloD4i8nvgy0V119Xz350000Hf,0] [SRC_CLASS: oracle.forms.servlet.RunformSession]
 [APP: formsapp#11.1.2] [SRC_METHOD: <init>] [FORMS_SESSION_ID: ..43] Prestarted
 process is not available.
[2009-02-11T14:39:21.718+00:00] [WLS_FORMS] [TRACE:32] [FRM-94522]
 [oracle.forms.servlet] [tid: [ACTIVE].ExecuteThread: '2' for queue:
 'weblogic.kernel.Default (self-tuning)'] [userId: <anonymous>] [ecid: 0000Hx
_llloD4i8nvgy0V119Xz350000Hf,0] [SRC_CLASS: oracle.forms.servlet.RunformSession]
 [APP: formsapp#11.1.2] [SRC_METHOD: <init>] [FORMS_SESSION_ID: ..43] [arg: null]
 Creating new runtime process using default executable.
[2009-02-11T14:39:21.718+00:00] [WLS_FORMS] [TRACE:32] [FRM-94532]
 [oracle.forms.servlet] [tid: [ACTIVE].ExecuteThread: '2' for queue:
 'weblogic.kernel.Default (self-tuning)'] [userId: <anonymous>] [ecid: 0000Hx
_llloD4i8nvgy0V119Xz350000Hf,0] [SRC_CLASS: oracle.forms.servlet.RunformProcess]
 [APP: formsapp#11.1.2] [SRC_METHOD: startProcess] [FORMS_SESSION_ID: ..43] [arg:
 frmweb webfile=HTTP-0,default] RunformProcess.startProcess():  executing frmweb
 webfile=HTTP-0,default

.
.
.
.
```

# 13

# Upgrading to Oracle Forms Services 11*g*

This chapter describes the upgrade process from Forms *6i*. For information about changed or obsolete features, see the *Oracle Forms Upgrading Oracle Forms 6i to Oracle Forms 11g Guide*.

This chapter contains the following sections:

- Section 13.1, "Oracle Forms Services Upgrade Items"
- Section 13.2, "Oracle Forms Services Upgrade Tasks"
- Section 13.3, "Validating the Oracle Forms Services Upgrade"

For upgrading from Oracle Forms 10*g* and prior releases, you can use the Upgrade Assistant. Refer to the following documents for more information.

- *Oracle Fusion Middleware Upgrade Planning Guide*
- *Oracle Fusion Middleware Upgrade Guide for Oracle Portal, Forms, Reports, and Discoverer*

## 13.1 Oracle Forms Services Upgrade Items

Table 13–1 describes the items that are upgraded. These items include files, executables, or settings that you must add, change, delete, or replace in the Oracle Forms Services installation.

*Table 13–1    Oracle Forms Services Upgrade Items*

| Upgrade Item | Location in 6i Oracle home | Location in 11*g* (11.1.2) Oracle home | Description and Notes |
|---|---|---|---|
| Oracle HTTP Server configuration file: `6iserver.conf` (upgrades to forms.conf) | `6iserver/conf/` | `$ORACLE_INSTANCE/config/OHS/<OHS Instance>/moduleconf/forms.conf` | Contains virtual path mappings. |
| Servlet environment file: `default.env` | `6iserver/forms60/server` | `$DOMAIN_HOME/config/fmwconfig/servers/WLS_FORMS/applications/formsapp_11.1.2/config/default.env` | Contains environment variables settings for the Forms servlet Runtime Process. |
| Configuration files with Forms servlet alias: `jserv.properties` (upgrades to web.xml) | `/Apache/jserv/conf` | `$DOMAIN_HOME/servers/WLS_FORMS/tmp/_WL_user/formsapp_11.1.2/<random_string>/war/WEB-INF` | Contains Forms servlet aliases. |
| Application configuration file: `formsweb.cfg` | `6iserver/forms60/server` | `$DOMAIN_HOME/config/fmwconfig/servers/WLS_FORMS/applications/formsapp_11.1.2/config/formsweb.cfg` | Contains Forms Services application configuration information. |
| Forms servlet template html files: (`*.htm`, `*.html`) | `6iserver/forms60/server` | `$ORACLE_INSTANCE/config/FormsComponent/forms/server/` | Default and user defined Forms servlet template HTML files. |
| Forms application modules (`fmb`/`fmx` files) | | | Forms modules (`fmb` and `fmx` files) deployed to Oracle 6i Forms Services must be upgraded to be deployed to Oracle Forms Services. Note that when you upgrade to 11*g*, you need to update the `FORMS_PATH` variable with the location of the `fmx` file. For more information on `FORMS_PATH`, see Section 4.3, "Managing Environment Variables". |

## 13.2  Oracle Forms Services Upgrade Tasks

This section explains how to perform the Oracle Forms Services upgrade. It is divided into the following sub-sections:

- Section 13.2.1, "Upgrade Recommendations and Troubleshooting Tips" on page 13-3

- Section 13.2.2, "Upgrading Oracle Forms Services Application Modules" on page 13-3

- Section 13.2.3, "Upgrading Common Gateway Interface (CGI) to the Oracle Forms Servlet" on page 13-4

- Section 13.2.4, "Upgrading Static HTML Start Files to Generic Application HTML Start Files" on page 13-5

- Section 13.2.5, "Upgrading the Forms 6i Listener to the Forms Listener Servlet" on page 13-8

### 13.2.1 Upgrade Recommendations and Troubleshooting Tips

Consider the following recommendations and considerations while upgrading Forms applications:

- Keep the Oracle6*i* Forms Services installation available until applications are successfully deployed and tested.

- Back up and secure all files, then upgrade the source files.

- Replace `Run_Product` calls to integrated Reports with `Run_Report_Object` calls to Oracle Reports (or use the PL/SQL conversion utility, Forms Migration Assistant in Oracle Forms).

- Install Oracle Fusion Middleware and configure the `formsweb.cfg` file with the information used by your applications.

- Copy the environment files used by the applications to the same relative directory.

- Copy the upgraded Oracle Forms application module files to the computer on which Oracle WebLogic Server is installed, if it is not the same computer.

- After starting Oracle WebLogic Server, access the Forms Services Listener Servlet test page with this URL (default port 8888):

  `http://<hostname>:<port>/forms/frmservlet?form=test.fmx`

- Verify that any application settings are added to the `formsweb.cfg` file and that the environment variable `Forms_Path` contains the directory of the application modules.

- Verify that you can connect to the database using SQL*Plus.

- Use the following URL to invoke upgraded applications:

  `http://<hostname>:<port>/forms/frmservlet?config=<your application name>`

### 13.2.2 Upgrading Oracle Forms Services Application Modules

This section provides instructions for upgrading from Forms Application Modules (`fmb` files) that were deployed in Oracle 6*i* Forms Services. Follow these steps to upgrade Forms Application Modules (`fmb` files) deployed in Oracle 6*i* Forms Services to an Oracle Forms Services installation.

1. Copy the Forms application files to a new directory.

2. Optionally, use the Forms Migration Assistant to upgrade the Forms Application Modules (`.fmb` files), Forms menu modules (`.mmb` files), and the Library modules (`.pll` files).

3. Use the Forms Compiler (`frmcmp.sh` on Unix or `frmcmp.exe` on Windows) to regenerate the Forms Application executable files (`fmx`, `mmx`, and `plx` files).

For more information, see Oracle Forms Upgrading Oracle Forms 6*i* to Oracle Forms 11*g* at:

http://www.oracle.com/technetwork/developer-tools/forms/document
ation/index.html.

### 13.2.3 Upgrading Common Gateway Interface (CGI) to the Oracle Forms Servlet

This section provides instructions to upgrade Forms CGI to the Forms servlet deployment. Follow these steps if you are using the Oracle 6i Forms Services Common Gateway Interface to dynamically render the Forms Applet start HTML file for applications.

CGI deployment for Forms applications was introduced in Oracle Forms Services Release 6i to enable the Forms Applet Start HTML file to render dynamically. Forms CGI uses the `formsweb.cfg` configuration file and an HTML template to create the start HTML file for an application. The CGI interface is configured by an entry in the Forms HTTP configuration file `6iserver.conf` (it is referenced by an `Include` directive in the Oracle HTTP Server `oracle_apache.conf` file), which contains a `ScriptAlias` directive identifying `dev60cgi` for the directory structure containing the `ifcgi60.exe` file.

The Forms servlet renders the HTML in the same manner as the CGI, but also provides an automatic browser type detection. The Forms servlet is configured when you install Oracle Forms Services, and is named `frmservlet`.

To access the Forms servlet, request the URL:

`http://<hostname>:<port>/forms/frmservlet`

This URL is similar to the URL used with the CGI Interface in Oracle 6i Forms Services. To call an application configured as `myapp` in the custom configuration section of the `formsweb.cfg` file, request the URL:

`http://<hostname>:<port>/forms/frmservlet?config=myapp`

The Forms servlet is automatically configured during installation. The installer creates a virtual path `/forms/` pointing to the Oracle Forms Services configuration, `formsapp` and `formsweb`.

Follow these steps to upgrade an Oracle 6i Forms Services Release 6i CGI environment to an Oracle Forms Services servlet environment:

1. Copy all of the application-specific configurations from `<source_OH>/Forms60/Server/formsweb.cfg` and append them to `<destination_Domain_Dir>/config/fmwconfig/servers/WLS_FORMS/applications/formsapp_11.1.2/config/formsweb.cfg`.

   > **Note:** Do not copy and replace the entire `formsweb.cfg` file in `<source_OH>` to `<destination_Domain_Dir>`. The file in Release 6i is different from the Oracle Forms Services file. Copy only the application configuration to `<destination_Domain_Dir>/config/fmwconfig/servers/WLS_FORMS/applications/formsapp_11.1.2/config/formsweb.cfg`.

2. Configure `Forms_Path` in the `default.env` file to point to the upgraded Oracle Forms Services application modules.

> **Note:** You can create a new environment file by copying
> `default.env`, modifying it for use with a particular application, and
> adding `envFile=<created environment file>` to the custom
> application section in the `formsweb.cfg` file.

3. If you changed the Oracle 6*i* Forms HTML template files, then make the same changes to the Oracle Forms Services HTML template files.

> **Note:** You must make these changes in `basejpi.htm` rather than
> `basejini.htm` because the servlet supports the Oracle Java plug-in.

## 13.2.4 Upgrading Static HTML Start Files to Generic Application HTML Start Files

Each application deployed to Oracle Forms Services has a custom application definition, configured in the `formsweb.cfg` configuration file. It automatically inherits the general system settings, such as the names and locations of the base HTML template files.

The name of the custom application definition becomes part of the Forms application URL. The following custom settings define two different applications:

```
[MyHR_app]
serverURL=/forms/lservlet
Form = hr_main.fmx
lookAndFeel=oracle
Otherparams=myParam1=12
Userid=scott/tiger@orcl
```

The following URL invokes this application:

`http://<hostname>:<port>/forms/frmservlet?config=MyHR_app`

Another custom application definition might look like this:

```
[booking_app]
ServerURL=/forms/lservlet
Form = book.fmx
lookAndFeel=oracle
Otherparams=
Userid=
```

The following URL invokes this application:

`http://<hostname>:<port>/forms/frmservlet?config=booking_app`

For each static HTML file, you must create a custom application definition. Part of the static HTML file is the `archive` parameter directive, specifying at least the `frmall.jar` file in Oracle Forms Services. If you added a custom archive file, then the `archive` parameter directive would resemble the following: `Archive=frmall.jar,custom.jar`. Using the Forms servlet and the `formsweb.cfg` file, the archive settings are defined under the User Parameter section. All custom application settings inherit these values, so you don't have to explicitly set this parameter, unless you add a `custom.jar` file as required by an application.

If `custom.jar` was added, then you can add the following lines to the custom application definition. The example below assumes that you are using another VM.

```
[booking_app]
```

```
archive=frmall.jar, custom.jar
ServerURL=/forms/lservlet
Form = book.fmx
lookAndFeel=oracle
Otherparams=
Userid=
```

Follow these steps to upgrade applications:

1. Edit the `default.env` file, adding the location of the Oracle Forms Services application modules to the `Forms_Path`.

2. Edit the `formsweb.cfg` file, appending a custom application section for each static HTML application that you want to replace.

3. Name each custom application section, using a name that contains no spaces and is enclosed in square brackets, for example: `[booking_app]`, `[MyHR_app]`.

4. Start the application using this URL:

   `http://<hostname>:<port>/forms/frmservlet?config=<name>`

### 13.2.4.1  Using Static HTML Files with Oracle Forms Services

If you need to, you can continue to use static HTML files in Oracle Forms Services. However, with static HTML files, some features (Single Sign-On) are not available for use by Forms applications.

The Forms Listener servlet by default points to `/forms/lservlet` after installation. To use static HTML files in Oracle Forms Services, you must modify each static start HTML file to include a value for the `serverURL` parameter. The `serverPort` and `serverHost` parameters are no longer used, and can be left undefined.

Follow these steps to use static HTML files with Oracle Forms Services:

1. Configure `Forms_Path` in the `default.env` file to point to the upgraded Oracle Forms Services application modules.

2. Create virtual directories in the `$ORACLE_INSTANCE/config/OHS/<OHS Instance>/moduleconf/forms.conf` file to point to the location of the static HTML start files.

3. Modify the application start HTML files as follows:

   a. Add the `serverURL` value `/forms/lservlet`.

4. Change the `codebase` parameter to `forms/java`.

5. Navigate to `$DOMAIN_HOME/deploymentplans/formsapp/11.1.2/plan.xml`.

6. Modify the deployment plan as described in Section 5.2.4, "Modification of Forms J2EE Application Deployment Descriptors" by adding entries highlighted in bold as shown in the following sample:

```
<?xml version='1.0' encoding='UTF-8'?>
<deployment-plan xmlns="http://xmlns.oracle.com/weblogic/deployment-plan"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://xmlns.oracle.com/weblogic/deployment-plan
http://xmlns.oracle.com/weblogic/deployment-plan/1.0/deployment-plan.xsd"
global-variables="false">
  <application-name>formsapp</application-name>
  <variable-definition>
    <variable>
```

```
      <name>vd-/scratch/t_work/Oracle/Middleware/as_1/forms</name>
      <value>/scratch/t_work/Oracle/Middleware/as_1/forms</value>
    </variable>
    <variable>
      <name>vd-/scratch/t_work/Oracle/Middleware/user_
projects/domains/ClassicDomain/config/fmwconfig/servers/WLS_
FORMS/applications/formsapp_11.1.2/config/forms</name>
      <value>/scratch/t_work/Oracle/Middleware/user_
projects/domains/ClassicDomain/config/fmwconfig/servers/WLS_
FORMS/applications/formsapp_11.1.2/config/forms</value>
    </variable>
    <variable>
       <name>FormsServlet_InitParam_testMode</name>
       <value>true</value>
    </variable>
    <variable>
       <name>lServlet_InitParam_envFile</name>
       <value>complete path to default.env</value>
    </variable>
  </variable-definition>
  <module-override>
    <module-name>formsapp.ear</module-name>
    <module-type>ear</module-type>
    <module-descriptor external="false">
      <root-element>weblogic-application</root-element>
      <uri>META-INF/weblogic-application.xml</uri>
    </module-descriptor>
    <module-descriptor external="false">
      <root-element>application</root-element>
      <uri>META-INF/application.xml</uri>
    </module-descriptor>
    <module-descriptor external="true">
      <root-element>wldf-resource</root-element>
      <uri>META-INF/weblogic-diagnostics.xml</uri>
    </module-descriptor>
  </module-override>
  <module-override>
    <module-name>formsweb.war</module-name>
    <module-type>war</module-type>
    <module-descriptor external="false">
      <root-element>weblogic-web-app</root-element>
      <uri>WEB-INF/weblogic.xml</uri>
      <variable-assignment>
        <name>vd-/scratch/t_work/Oracle/Middleware/as_1/forms</name>
<xpath>/weblogic-web-app/virtual-directory-mapping/[url-pattern="java/*"]/local
-path</xpath>
      </variable-assignment>
      <variable-assignment>
        <name>vd-/scratch/t_work/Oracle/Middleware/as_1/forms</name>
<xpath>/weblogic-web-app/virtual-directory-mapping/[url-pattern="webutil/*"]/lo
cal-path</xpath>
      </variable-assignment>
      <variable-assignment>
        <name>vd-/scratch/t_work/Oracle/Middleware/user_
projects/domains/ClassicDomain/config/fmwconfig/servers/WLS_
FORMS/applications/formsapp_11.1.2/config/forms</name>
<xpath>/weblogic-web-app/virtual-directory-mapping/[url-pattern="registry/*"]/l
ocal-path</xpath>
      </variable-assignment>
    </module-descriptor>
```

```
                    <module-descriptor external="false">
                      <root-element>web-app</root-element>
                      <uri>WEB-INF/web.xml</uri>
                      <variable-assignment>
                        <name>lServlet_InitParam_envFile</name>
          <xpath>/web-app/servlet/[servlet-name="lservlet"]/init-param/[param-name="envFi
          le"]/param-value</xpath>
                      </variable-assignment>
                    </module-descriptor>
                  </module-override>
                </deployment-plan>
```

## 13.2.5  Upgrading the Forms 6*i* Listener to the Forms Listener Servlet

The Forms 6*i* Listener was a C program that starts a Forms runtime process on behalf of an incoming Forms Web request. The Forms Web runtime process was then directly accessed by the Forms client applet, using a direct socket or an HTTP socket connection. The Forms Listener was then no longer involved in the application Web client-server communication process, and was free to handle other incoming Web requests.

The Forms Listener servlet, a Java program, also takes incoming Web requests for a Forms application and starts the Forms Web runtime process. Unlike the Forms 6*i* Listener, the Forms Listener servlet remains between the Forms application applet-server communication.

While the Forms 6*i* Listener listened on a specific port (by default, 9000), the Forms servlet does not need an extra port, and is accessed by the HTTP listener port. The Forms Listener servlet was introduced in the Forms 6*i* patch 4, and is the only listener supported in Forms Services.

The Forms Listener servlet is automatically configured during the installation. The installer creates a virtual path /forms/ pointing to the Oracle Forms Services configuration, formsapp and formsweb.

To access the Forms Listener servlet test form, request the following URL:

http://<*hostname*>:<*port*>/forms/frmservlet?form=test.fmx

Ability to access this page means that the Forms Listener servlet is configured and ready to use. frmservlet is the access name configured for the Forms servlet during installation. The name of the Listener Servlet is lservlet.

If the Forms Listener servlet is accessed with the Forms servlet, then only the custom application settings from the Forms60/server/formsweb.cfg file need to be appended to the formsweb.cfg file. All application configurations automatically inherit the serverURL parameter value /forms/lservlet from the global system parameter settings.

To change a Forms application deployment from the Forms Listener architecture to the Listener Servlet architecture, you need only supply a value for the serverURL parameter in the formsweb.cfg file. During installation, this parameter is set to /forms/lservlet.

Follow these steps to upgrade to the Forms Listener servlet:

1. Copy the Forms application files to a new directory and upgrade them to Oracle Forms Services modules as described in

2. Edit the `forms/server/default.env` file to add the location of the upgraded Forms application modules to the `Forms_Path` variable.

3. Copy all of the custom application settings from `<source_OH>/Forms60/Server/formsweb.cfg` and append them to `<destination_Domain_Dir>/config/fmwconfig/servers/WLS_FORMS/applications/formsapp_11.1.2/config/formsweb.cfg`.

4. If an application requires its own environment file, then instead of defining a separate servlet alias for the Listener Servlet, set the `envFile` parameter in the custom application definition section in `<destination_Domain_Dir>/config/fmwconfig/servers/WLS_FORMS/applications/formsapp_11.1.2/config/formsweb.cfg` to point to the new environment file. For example:

   ```
   envFile=myEnvFile.env
   ```

   where `myEnvFile.env` is located in the `forms/server` directory.

5. If you changed the Oracle 6*i* Forms Services HTML template files, then make the same changes to the Oracle Forms Services HTML template files.

   > **Note:** If you need to change the underlying HTML files, you should make a copy of the provided template files before editing them. Save the edited HTML files under a different name, and leave the default templates provided with the installation unchanged. This prevents overwriting of your customized HTML template files when patch sets are applied to the application.
   >
   > To use your own template files with applications, use these parameters in the system section, or one of your custom application definitions: `baseHTML=<your base template>.htm`

6. Start the application with this URL:

   ```
   http://<hostname>:<port>/forms/frmservlet?
   config=<application>
   ```

## 13.2.6 Upgrading the Forms Listener Servlet Architecture to Oracle Forms Services

In Oracle9*i*AS Forms Services Release 6*i*, the Listener Servlet, if not aliased, is accessed by the `oracle.forms.servlet.ListenerServlet`. The Listener Servlet configuration exists in the `jserv.properties` file and the `zone.properties` file.

In Oracle Forms Services, the Forms Listener servlet is the same except for the servlet names, which are `frmservlet` and `lservlet`, and the servlet container. The configuration is performed during installation. The Listener Servlet configuration in Oracle WebLogic Managed Server is stored in `$DOMAIN_HOME/servers/WLS_FORMS/tmp/_WL_user/formsapp_11.1.2/<random_string>/war/WEB-INF/web.xml`. Some initialization parameters, like the `envFile` parameter, need no longer be configured with the servlet engine, because they are moved to the `formsweb.cfg` file.

The Forms Listener servlet is automatically configured during the Oracle WebLogic Server installation. The installer creates a virtual path `/forms/` pointing to the Oracle Forms Services configuration, `formsapp` and `formsweb`.

To access the Forms Listener servlet test form, request the following URL:

```
http://<hostname>:<port>/forms/frmservlet?form=test.fmx
```

Ability to access this page means that the Forms Listener servlet is configured and ready to use. `frmservlet` is the access name configured for the Forms servlet during installation. The name of the Listener Servlet is `lservlet`.

Follow these steps to upgrade the Listener Servlet architecture to Oracle Forms Services:

1.  Copy the Forms application files to a new directory and upgrade them to Oracle Forms Services modules.

2.  Edit the `forms/server/default.env` file, adding the location of the upgraded Forms application modules to the `Forms_Path` variable.

3.  Copy all of the custom application settings from `<source_OH>/Forms60/Server/formsweb.cfg` and append them to `<destination_Domain_Dir>/config/fmwconfig/servers/WLS_FORMS/applications/formsapp_11.1.2/config/formsweb.cfg`.

4.  If an application requires its own environment file, then instead of defining a servlet alias for the Listener Servlet, set the envFile parameter in the custom application definition section in `<destination_Domain_Dir>/config/fmwconfig/servers/WLS_FORMS/applications/formsapp_11.1.2/config/formsweb.cfg` to point to the new environment file. For example:

    ```
    envFile=myEnvFile.env
    ```

    where `myEnvFile.env` is located in the `forms/server` directory.

5.  If you changed the Forms Services Release 6*i* HTML template files, then make the same changes to the Oracle Forms Services HTML template files.

    ---
    **Note:** If you need to change the underlying HTML files, you should make a copy of the provided template files before editing them. Save the edited HTML files under a different name, and leave the default templates provided with the installation unchanged. This prevents overwriting of your customized HTML template files when patch sets are applied to the application.

    To use your own template files with applications, use these parameters in the system section, or one of your custom application definitions: baseHTML=<your base template>.htm

    ---

6.  Start the application with this URL:

    ```
    http://<hostname>:<port>/forms/frmservlet?config=<application>
    ```

### 13.2.7 Upgrading Load Balancing

The method of upgrading the load balancing in Forms Services 6*i* depends on the deployment method used.

- With the Forms 6*i* listener, the Metrics Server (a separate process) performs load balancing.

- With the Forms 6*i* servlet, load balancing is configured with the JServ servlet engine, using round robin load balancing among JServ engines.

- In Oracle Forms Services, load balancing is managed by Oracle WebLogic Managed Server process. It binds Web requests to the servlet container processing the Forms servlet and the Forms Listener servlet.

## 13.2.8 Usage Notes

This section contains hints and tips that may be useful in the upgrade.

### 13.2.8.1 Deploying Icon Images with the Forms Servlet

Using static HTML start files in Forms Services Release 6*i* allowed storage of images in a location relative to the start HTML file. The Forms servlet in Oracle Forms Services does not support this.

The alternative is to use the `imagebase` parameter with the value of `codebase` as the location for the icon images used by applications. The `codebase` value refers to the `forms/java` directory, which contains all of the Forms client Java archive files. For performance reasons, it is not a good idea to store images here.

Instead, you should bundle the icons into a separate archive file, which improves performance because archives are cached permanently on the client. Follow these steps to create this archive file.

1. Verify that the `jar` command succeeds. If it does not, then you need to ensure that there is a JDK installed on your system with a correct PATH environment variable entry (pointing to the `JDK_HOME/bin` directory).

2. Navigate to the directory containing the application images and issue the command:

   **`jar -cvf <application>_images.jar *.<extension>`**

   where:

   - *`application`* is the name of the application

   - *`extension`* is the extension of the image file (for example,.gif)

   A jar file, `<application>_images.jar`, is created in the current directory.

3. Copy `<application>_images.jar` to the `forms/java` directory.

4. Edit the `formsweb.cfg` file, adding the `imageBase=codebase` parameter to the custom application section for the application.

5. Add the `<application>_images.jar` file to the archive path used by the application by adding the following line to the custom application section:

   `archive=frmall.jar,<application>_images.jar`

See Section 4.7, "Deploying Fonts, Icons, and Images Used by Forms Services" for more information on deploying custom icon files with Oracle Forms Services.

### 13.2.8.2 Upgrading Integrated Calls to Oracle Forms to use Oracle Reports

Integrated calls to Oracle Reports in Forms are no longer handled by a client-side background engine. Oracle Forms Services requires that applications use the `RUN_REPORT_OBJECT` built-in, calling Oracle Reports to process integrated reports. Oracle Reports is set up as part of the Business Intelligence and Forms installation.

Follow these steps to upgrade the call:

1. Change all occurrences of `RUN_PRODUCT (Reports,...)` to the equivalent call using `RUN_REPORT_OBJECT()`.

2. Add the location of the application's Reports modules to use the `Reports_Path` of Oracle Reports.

3. Change `RUN_REPORT_OBJECT` to reference Oracle Reports.

   For more information, see *Oracle Fusion Middleware Publishing Reports to the Web with Oracle Reports Services*.

### 13.2.8.3  Creating Forms Listener Servlet Alias Names

In Forms Services Release 6*i*, before patch 8, it was necessary to create alias names for the Forms servlet in the `$ORACLE_HOME/Apache/Apache/JServ/conf/zone.properties` file in order to use individual environment files for different applications. The Forms servlet in Oracle Forms Services does not require this. You can set the environment file name in the `formsweb.cfg` file using the `envFile` parameter, shown below:

`envFile=myApp.env`

Alias names for the Forms servlet are no longer created in `$ORACLE_HOME/Apache/Apache/JServ/conf/zone.properties`. Instead, they are created in `$DOMAIN_HOME/servers/WLS_FORMS/tmp/_WL_user/formsapp_11.1.2/<random_string>/war/WEB-INF/web.xml`.

To create the alias names, copy the content between the `<servlet>` and `</servlet>` tags and change the servlet's name. To create a URL mapping for the new servlet alias name, add the following to the file:

```
<servlet-mapping>
<servlet-name>new servlet name</servlet-name>
<url-pattern>/new url name*</url-pattern>
</servlet-mapping>
```

### 13.2.8.4  Accessing the Listener Servlet Administration Page

You can display a test page for the Listener Servlet in Oracle9*i*AS Forms Services Release 6*i* by accessing the following URL:

`http://<hostname>:<port>/servlet/`
`oracle.forms.servlet.ListenerServlet`

The information displayed depends on the value of the initialization parameter `TestMode`. This parameter is set in the `<source_OH>/Apache/Apache/JServ/conf/zone.properties` file.

You can display the test page for Oracle Forms Services with the following URL:

`http://<hostname>:<port>/forms/frmservlet/admin`

The information displayed depends on the value of the initialization parameter `TestMode`. This parameter is set in the `$DOMAIN_HOME/servers/WLS_FORMS/tmp/_WL_user/formsapp_11.1.2/<random_string>/war/WEB-INF/web.xml` file. An example is shown below:

```
<init-param>
<!-- Display sensitive options on the /admin page ? -->
    <param-name>TestMode</param-name>
    <param-value>true</param-value>
</init-param>
```

## 13.3 Validating the Oracle Forms Services Upgrade

After you complete the upgrade tasks, ensure that the upgraded version of the Oracle Forms Services is working as expected. You must devise and perform specific tests for applications and configuration elements that are unique to your site. Compare the performance and characteristics of each application in the source and destination installations.

In Oracle9iAS Release 1 (1.0.2.2.x), the forms application URL is typically:

```
http://<hostname>:<port>/servlet/<forms servlet alias>?<forms application name>
```

In Oracle Forms 11*g,* the forms application URL is typically:

```
http://<hostname>:<port>/forms/<forms servlet alias>?form=<forms application name>
```

# 14

# Performance Tuning Considerations

This chapter contains the following sections:

- Section 14.1, "Built-in Optimization Features of Forms Services"
- Section 14.2, "Tuning Oracle Forms Services Applications"
- Section 14.3, "Web Cache and Forms Integration"

Tuning the connection between Oracle Forms Services and the Oracle Database Server is beyond the scope of this chapter.

## 14.1 Built-in Optimization Features of Forms Services

The Oracle Forms Services and Java client include several optimizations that fit broadly into the following categories:

- Section 14.1.1, "Monitoring Forms Services"
- Section 14.1.2, "Forms Services Web Runtime Pooling"
- Section 14.1.3, "Minimizing Client Resource Requirements"
- Section 14.1.4, "Minimizing Forms Services Resource Requirements"
- Section 14.1.5, "Minimizing Network Usage"
- Section 14.1.6, "Maximizing the Efficiency of Packets Sent Over the Network"
- Section 14.1.7, "Rendering Application Displays Efficiently on the Client"

### 14.1.1 Monitoring Forms Services

Use Fusion Middleware Control to monitor Oracle Forms Services and review metrics information, including:

- Forms Services Instances
- Events
- User Sessions
- Forms Trace

#### 14.1.1.1 Monitoring Forms Services Instances

Use the Forms Home page to monitor metrics for a Forms Services instance.

1. Start Enterprise Manager Fusion Middleware Control.

**2.** From the Enterprise Manager Fusion Middleware Control main page, select the link to the Forms Services instance that you want to monitor.

The Forms Home page for the Forms Services instance displays the following:

- Status of Forms application instance (up, down, unknown)
- URL of the Forms Services instance being monitored
- Number of Forms sessions

Additionally, you can navigate to the following detail pages:

- Performance Summary
- Servlet Logs
- Session Details
- Web Configuration
- Environment Configuration
- Trace Configuration
- User Sessions
- JVM Configuration
- JVM Controllers

  In the Performance Summary page, you can add charts for other Forms metrics to the page dynamically by using the Show Metric Palette. You can also overlay metrics to compare them. For example, drag and drop Private Memory consumed by two JVM Controllers into one chart to compare them. For more information, see the Oracle Fusion Middleware Performance Guide.

### 14.1.1.2 Monitoring Forms Events

Use the Enterprise Manager Fusion Middleware Control to enable tracing for all events or specific ones. See Table 14–1 for a list of tasks you can perform on this page.

*Table 14–1   Monitoring Forms Events*

| Task | See Section |
| --- | --- |
| Monitoring metrics for user sessions | "To view Forms user sessions:" |
| Sorting metrics information | "To sort the list of Forms user sessions:" |
| Searching for metrics information | "To search for a Forms user sessions:" |

## 14.1.2 Forms Services Web Runtime Pooling

Forms Runtime Pooling (or Forms Runtime prestart) enables the startup of a configurable number of application runtime engines prior to their usage. Runtime Pooling provides quick connections at server peak times, which shortens the server-side application startup time. Runtime pooling is useful for situations where server configurations have a small window in which many users connect to a Forms application. All prestarted runtime engines run in the same environment serving the same application.

### 14.1.2.1 Configuring Prestart Parameters

Use Enterprise Manager Fusion Middleware Control to configure runtime pooling for Forms Services with the following parameters as described in Table 14–2:

*Table 14–2    Forms Runtime Pooling Parameters*

| Parameter Name | Data type | Description | Default Value |
| --- | --- | --- | --- |
| prestartRuntimes | boolean | Runtime pre starting or pooling is enabled only if true | false |
| prestartInit | integer | Number of the runtime processes that should be spawned initially | 1 |
| prestartTimeout | integer | Time in minutes after which all the prestarted processes of this pool (configuration section) will be stopped. A runtime process is removed from the prestart pool once client connection is made and thus will not be stopped. | 0 (When set to zero the timer never starts) |
| prestartMin | integer | Minimum number of runtime processes to exist in the pool. | 0 |
| prestartIncrement | integer | The number of runtime processes to be created when the number of prestarted runtime processes is less than minRuntimes. | 0 |

Note that prestartMin defines the minimum number of pre-started runtimes that must exist at any time while runtime pooling is still active for a specific application. The minimum value must be less than or equal to what's defined for the prestartInit parameter. The prestartMin parameter can be modified at any time and does not require the application server to be restarted. The new entries will be picked up when a client requests a connection to a pre-started runtime process and the prestarted runtime processes have not timed out. Once they have timed out, an application uses default behavior and a minimum threshold is not maintained.

Each configuration section can specify values for these parameter. If the prestartRuntimes = true entry is found, but there is no associating prestart parameter, then default values are used.

In a load balanced system that has multiple instances of Oracle WebLogic Managed Server, the various values provided for the above parameters are on a per JVM basis, and not the total for the application.

### 14.1.2.2 Starting Runtime Pooling

An Administrator can configure specific application(s), from the Enterprise Manager Fusion Middleware Control, to enable Runtime Pooling. On the startup of the application server (Oracle WebLogic Managed Server), the configured number of Forms Runtime processes are pre-started for each application.

In the initialization phase of the Forms servlet, the configuration file (`formsweb.cfg`) is read and the server pre-starts the applications which have the `prestartRuntimes` parameter enabled.

### 14.1.2.3 Scheduling Runtime Pooling

Scheduling Runtime Pooling (or Scheduling Runtime Prestart) is a feature that enables you to schedule the prestart of Forms Runtime engines. In addition to managing the startup of a configurable number of Forms Runtime engines prior to their usage, Oracle Forms now allows you to schedule the prestarting of Forms Runtime processes on a more flexible basis, at any appropriate time. You can schedule a Forms Runtime prestart, view existing schedules, delete any existing schedule, and export and import a schedule using the Enterprise Manager Fusion Middleware Control.

**Creating a Prestart Schedule**

To create a prestart schedule, perform the following steps:

1. From the Forms Menu, select **Schedule Prestart**.

   The **Schedule Prestart** page is displayed.

2. From the **Scheduled Jobs** region, click **Create**.

   Figure 14–1 shows the page that is displayed for scheduling a prestart.

**Figure 14–1  Scheduling Forms Runtime Prestart**



3. In **Job Name**, enter a name for the schedule.

   Maximum length of the job name must not exceed 100 characters. The name must not contain any special characters such as ampersand (&).

4. From the **Configuration Section** list, choose a configuration type.

   This list contains a logical set of parameters.

5. In **Prestart Init**, enter a numerical value for the number of runtime processes that must be spawned initially. Ensure that the value is greater than or equal to 1.

6. In **Prestart Timeout**, enter a numerical value for the time in minutes after which the unused prestart process will be stopped. If this value is set to zero, the timer never starts and thus the processes do not time out.

7. From the **Schedule Type** options, select the appropriate schedule type.

   Following are the three types of schedules that you can set:

   - **One Time(Delay based)**: Select this option if you want to schedule a single occurence prestart based on the initial delay. Initial delay is a time based parameter that specifies the number of hours or minutes after which the prestart will begin. If you  select this option, you must enter the initial delay time (in hours or minutes) in the **Initial delay** field that appears below the schedule type.

   - **One Time(Date based)**: Select this option if you want to schedule a single occurence prestart based on a date. If you select this option, you must enter the date and time in the **Start Date** field that appears below the Schedule type.

   - **Repeating**: Select this option if you want to schedule a repeat prestart.  From the **Frequency** list, you can select one of the following options:

     - **Repeat date and interval**: If you select this option, you must specify the start date and interval after which you want the prestart to repeat.

     - **Repeat initial delay and interval**: If you select this option, you must specify the initial delay and interval after which you want the prestart to repeat.

8. Click **Submit**.

   You can click **Show Jobs** to view all the prestart schedules that are created.

### Deleting a Prestart Schedule

To delete a prestart schedule, perform the following steps:

1. From the Forms Menu, select **Schedule Prestart**.

   The **Schedule Prestart** page is displayed.

2. From the **Scheduled Jobs** region, select the row of the scheduled prestart that you want to delete.

   You can select multiple rows to delete at the same time.

3. Click **Delete**.

   The selected prestart schedule is deleted.

### Exporting and Importing Prestart Schedules

This utility allows you to export all existing schedules from a particular managed server and import it to some other managed server. This feature does not allow the users to export schedules selectively; the user must export all existing schedules. To export and import prestart schedules, perform the following steps:

1. From the Forms Menu, select **Schedule Prestart**.

   The **Schedule Prestart** page is displayed.

2. From the **Scheduled Jobs** region, click **Export**.

   A dialog box appears.

3. Enter a new file name in the dialog box.

   The file name must contain `.xml` extension.

4. Click **Ok**.

   This will create an xml file that contains the attributes of all prestart schedules. Depending on your browser settings, you will either be prompted to choose a location to save the file in, or the file will be saved at a default location on you local machine.

5. To import the schedules (that you have exported in the above steps) to some other managed server, go to the **Schedule Prestart** page of that server, and click **Import**.

   A dialog box appears.

6. Enter the name of the file that you created while exporting the schedules. You can also click **Browse** on the dialog box and upload the `xml` file from you local machine.

7. Click **Ok**.

   The imported schedules will now be listed in the **Scheduled Jobs** region.

---

> **Note:** If there is an error in any schedule entry in the `xml` file, the server skips that particular schedule and imports the next schedule entry. If there are any expired schedules in the `xml` file, they are ignored and not imported.

---

### 14.1.3  Minimizing Client Resource Requirements

The Java client is primarily responsible for rendering the application display. It has no embedded application logic. Once loaded, a Java client can display multiple forms simultaneously. Using a generic Java client for all Oracle Forms applications requires fewer resources on the client when compared to having a customized Java client for each application.

The Java client is structured around many Java classes. These classes are grouped into functional subcomponents, such as displaying the splash screen, communicating with the network, and changing the look-and-feel. Functional subcomponents allow the Forms Developer and the Java Virtual Machine (JVM) to load functionality as it is needed, rather than downloading all of the functionality classes at once.

### 14.1.4  Minimizing Forms Services Resource Requirements

When a form definition is loaded from an FMX file, the profile of the executing process can be summarized as:

- Encoded Program Units
- Boilerplate Objects/Images
- Data Segments

Of these, only the data segments section is unique to a given instance of an application. The encoded program units and boilerplate objects/images are common to all application users. Forms Services maps the shared components into physical memory, and then shares them between all processes accessing the same FMX file.

The first user to load a given FMX file will use the full memory requirement for that form. However, subsequent users will have a greatly reduced memory requirement, which is dependent only on the extent of local data. This method of mapping shared components reduces the average memory required per user for a given application.

## 14.1.5 Minimizing Network Usage

Bandwidth is a valuable resource, and the general growth of Internet computing puts an ever increasing strain on the infrastructure. Therefore, it is critical that applications use the network's capacity sparingly.

Oracle Forms Services communicates with the Java client using metadata messages. Metadata messages are a collection of name-value pairs that tell the client which object to act upon and how. By sending only parameters to generic objects on the Java client, there is approximately 90-percent less traffic (when compared to sending new code to achieve the same effect).

Oracle Forms Services intelligently condenses the data stream in three ways:

- When sets of similar messages (collections of name-value pairs) are sent, the second and subsequent messages include only the differences from the previous message. This results in significant reductions in network traffic. This process is called *message diff-ing*.

- When the same string is to be repeated on the client display (for example, when displaying multiple rows of data with the same company name), Oracle Forms Services sends the string only once, and then references the string in subsequent messages. Passing strings by reference increases bandwidth efficiency.

- Data types are transmitted in the lowest number of bytes required for their value.

## 14.1.6 Maximizing the Efficiency of Packets Sent Over the Network

The extensive use of triggers within the Forms Developer model is a strength, but they can increase the effect of latency by requiring a network round trip for each trigger. Latency can be the most significant factor that influences the responsiveness of an application. Note that latency is not the same as network speed. Network speed involves a measure of the bits that can be transported per time unit whereas latency is the time taken for one bit to travel from one end-point to the other. One of the best ways to reduce the effects of latency is to minimize the number of network packets sent during a conversation between the Java client and the Forms Services.

Oracle Forms Services implements event bundling by grouping trigger events together through Event Bundling. Event Bundling gathers all of the events triggered while navigating between the two objects, and delivers them as a single packet to Oracle Forms Services for processing.

For example, when a user navigates from item A to item B (such as when tabbing from one entry field to another), a range of pre- and post-triggers may fire, each of which requires processing on the Forms Services. When navigation involves traversing many objects (such as when a mouse click is on a distant object), Event Bundling gathers all events from all of the objects that were traversed, and delivers the group to Oracle Forms Services as a single network message.

## 14.1.7 Rendering Application Displays Efficiently on the Client

All boilerplate objects in a given form are part of a Virtual Graphics System (VGS) tree. VGS is the graphical subcomponent that is common to all Forms Developer products. VGS tree objects are described using attributes such as coordinates, colors, line width,

and font. When sending a VGS tree for an object to the Java client, the only attributes that are sent are those that differ from the defaults for the given object type.

Images are transmitted and stored as compressed JPEG images. This reduces both network overhead and client memory requirements.

Minimizing resources includes minimizing the memory overhead of the client and server processes. Optimal use of the network requires that bandwidth be kept to a minimum and that the number of packets used to communicate between the client and Oracle Forms Services be minimized in order to contain the latency effects of the network.

## 14.2 Tuning Oracle Forms Services Applications

An application developer can take steps to ensure that maximum benefits are gained from Forms Services' built-in architectural optimizations. The remainder of this chapter discusses key performance issues that affect many applications and how developers can improve performance by tuning applications to exploit Forms Services features.

### 14.2.1 Location of the Oracle Forms Services with Respect to the Data Server

The Forms Java client is only responsible to display the GUI objects. All of the Oracle Forms logic runs in Oracle Forms Services, on the middle tier. This includes inserting or updating the data to the database, querying data from the database, executing stored procedures on the database, and so on. Therefore, it is important to have a high-speed connection (high bandwidth and not low latency) between the application server and the database server.

All of this interaction takes place without any communication to the Forms Java client. Only when there is a change on the screen is there any traffic between the client and Forms Services. This allows Oracle Forms applications to run across slower networks (high latency networks), such as with modems or satellites.

The configuration in Figure 14–2, displays how Forms Services and the database server are co-located in a data center.

*Figure 14–2  Co-Locating the OracleAS Forms Services and Database Server*
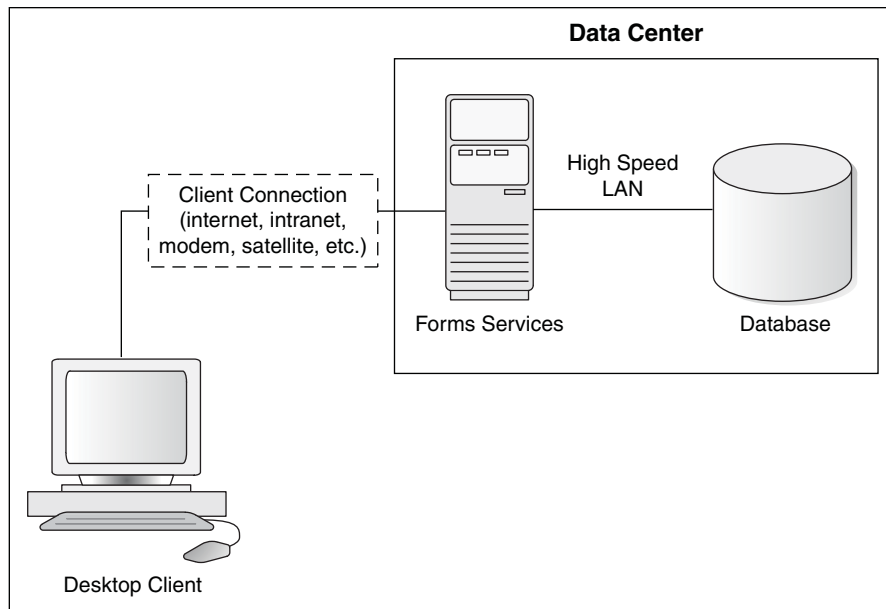


## 14.2.2  Minimizing the Application Startup Time

First impressions are important, and a key criterion for any user is the time it takes to load an application. Startup time is regarded as overhead. It also sets an expectation of future performance. When a business uses thin-client technologies, the required additional overhead of loading client code may have a negative impact on users. Therefore, it is important to minimize load time wherever possible.

After requesting an Oracle Forms application, several steps must be completed before the application is ready for use:

1.  Invoke Java Virtual Machine (JVM).

2.  Load all initial Java client classes, and authenticate security of classes.

3.  Display splash screen.

4.  Initialize form:

    a.  Load additional Java classes, as required.

    b.  Authenticate security of classes.

    c.  Render boilerplate objects and images.

    d.  Render all elements on the initial screen.

5.  Remove splash screen.

6.  Form is ready for use.

An application developer has little influence on the time it takes to launch the JVM. However, the Java deployment model and the structure of the Oracle Forms Developer Java client allow the developer to decide which Java classes to load and how. This, in turn, minimizes the load time required for Java classes.

The Java client requires a core set of classes for basic functionality (such as opening a window) and additional classes for specific display objects (such as LOV items). These

classes must initially reside on the server, but the following techniques can be used to improve the time it takes to load these classes into the client's JVM:

- Using Java Files
- Using Caching

### 14.2.2.1 Using Java Files

Java provides the Java Archive (Jar) mechanism to create files that allow classes to be grouped together and then compressed (zipped) for efficient delivery across the network to the client. Once used on the client, the files are cached for future use.

It is also possible to double jar a file. This saves about 700k when done with `frmall.jar`. For Oracle's plugin, the resulting file must have a suffix of `jarjar`.

The following sections describe the pre-configured Jar files that Oracle Forms Services provides to support typical deployment scenarios.

### 14.2.2.2 Using Oracle's Java Plug-in

frmall.jar includes all required classes for running with the Java Plug-in.

To specify one or more Jar files, use the archive setting in the named configuration section of the Forms Configuration file (`formsweb.cfg`). For example,

```
[MyApp]
archive=frmall.jar
```

### 14.2.2.3 Using Caching

Oracle's Java Plug-in supports the caching of Jar files for Oracle Forms Services. When the JVM references a class, it first checks the local client cache to see if the class exists in a pre-cached Jar file. If the class exists in cache, JVM checks the server to see if there is a more current version of the Jar file. If there isn't, the class is loaded from the local cache rather than from across the network.

Be sure that the cache is of proper size to maximize its effectiveness. Too small a cache size may cause valid Jar files to be overwritten, thereby requiring that another Jar file be downloaded when the application is run again. The default cache size is 20MB. This size should be compared with the size of the cache contents after successfully running the application.

Jar files are cached relative to the host from which they were loaded. This has implications in a load-balancing architecture where identical Jar files from different servers can fill the cache. By having Jar files in a central location and by having them referenced for each server in the load-balancing configuration, the developer can ensure that only one copy of each Jar file is maintained in the client's cache. A consequence of this technique is that certain classes within the Jar file must be signed to enable connections back to servers other than the one from which they were loaded. The Oracle-supplied Jar files already pre-sign the classes.

## 14.2.3 Reducing the Required Network Bandwidth

The developer can design the application to maximize the data stream compression, called message-diffing, that Forms automatically performs. This means that forms sends along data stream compression by using message diff-ing, which sends along only the information that differs from one message to another. The following steps can be taken to reduce the differences between messages:

- **Promote similarities between objects.** Using similar objects improves *message diff-ing* effectiveness (in addition to being more visually appealing to the user). The following steps encourage consistency between objects:

  - Accept default values for properties, and change only those attributes needed for the object.

  - Use Smart Classes to describe groups of objects.

  - Lock the look-and-feel into a small number of visual attributes.

- **Reduce the use of boilerplate text.** As a developer, you should use the PROMPT item property rather than boilerplate text wherever applicable. Forms Developer 6.0 and higher includes the Associate Prompt feature, which allows boilerplate text to be re-designated as the prompt for a given item.

- **Reduce the use of boilerplate items (such as arcs, circles, and polygons).** All boilerplate items for a given Form are loaded at Form initialization. Boilerplate items take time to load and use resources on the client whether they are displayed or not. Common boilerplate items, namely rectangles and lines, are optimized. Therefore, restricting the application to these basic boilerplate items reduces network bandwidth and client resources while improving startup times.

- **Keep navigation to a minimum.** An Event Bundle is sent each time a navigation event finishes, whether the navigation extends over two objects or many more. Design Forms that do not require the user to navigate through fields when default values are being accepted. A Form should encourage the user to quickly exit once the Form is complete, which causes all additional navigation events to fire as one Event Bundle.

- **Reduce the time to draw the initial screen.** Once the Java client has loaded the required classes, it must load and initialize all of the objects to be displayed before it can display the initial screen. By keeping the number of items to a minimum, the initial screen is populated and displayed to the user more promptly. Techniques that reduce the time to draw the initial screen include:

  - Providing a login screen for the application with a restricted set of objects (such as a title, small logo, username, and password).

  - On the Form's initial display, hiding elements not immediately required. Use the canvas properties:

    ```
    RAISE ON ENTRY = YES  (Canvas only)
    ```

    ```
    VISIBLE = NO
    ```

    Pay attention to TAB canvases that consist of several sheets where only one will ever be displayed. For responsive switching between tabs, all items for all sheets on the canvas are loaded, including those that are hidden behind the initial tab. Consequently, the time taken to load and initialize a TAB canvas is related to all objects on the canvas and not just to those initially visible.

    **Tip:**   When using Tab canvases, use stacked canvases and display the right canvas in the when-tab-page-changed trigger. Remember to set the properties `RAISE ON ENTRY = YES` and `VISIBLE = NO` for all the canvases not displayed in the first screen.

- **Disable MENU_BUFFERING.** By default, MENU_BUFFERING is set to True. This means that changes to a menu are buffered for a future "synchronize" event when the altered menu is re-transmitted in full. (Most applications make either many simultaneous changes to a menu or none at all. Therefore, sending the entire

menu at once is the most efficient method of updating the menu on the client.) However, a given application may make only minimal changes to a menu. In this case, it may be more efficient to send each change as it happens. You can achieve this using the statement:

```
Set_Application_Property (MENU_BUFFERING, 'false');
```

Menu buffering applies only to the menu properties of LABEL, ICON, VISIBLE, and CHECKED. An ENABLE/DISABLE event is always sent and does not entail the retransmission of an entire menu.

## 14.2.4 Other Techniques to Improve Performance

The following techniques may further reduce the resources required to execute an application:

- **Examine timers and replace with JavaBeans.** When a timer fires, an asynchronous event is generated. There may not be other events in the queue to bundle with this event. Although a timer is only a few bytes in size, a timer firing every second generates 60 network trips a minute and almost 30,000 packets in a typical working day. Many timers are used to provide clocks or animation. Replace these components with self-contained JavaBeans that achieve the same effect without requiring the intervention of Forms Services and the network.

- **Consider localizing the validation of input items.** It is common practice to process input to an item using a When-Validate-Item trigger. The trigger itself is processed on the Forms Services. You should consider using pluggable Java components to replace the default functionality of standard client items, such as text boxes. Then, validation of items, such as date or max/min values, are contained within the item. This technique opens up opportunities for more complex, application-specific validation like automatic formatting of input, such as telephone numbers with the format (XXX) XXX-XXXX.

- **Reduce the application to many smaller forms, rather than one large form.** By providing a fine-grained application, the user's navigation defines which objects are loaded and initialized from the Forms Services. With large Forms, the danger is that the application is delayed while objects are initialized, many of which may never be referenced. When chaining Forms together, consider using the built-ins OPEN_FORM and NEW_FORM:

  - With OPEN_FORM, the calling Form is left open on the client and the server, so that the additional Form on both the client and the server consumes more memory. However, if the Form is already in use by another user, then the increase in server memory is limited to just the data segments. When the user returns to the initial Form, it already resides in local memory and requires no additional network traffic to redisplay.

  - With NEW_FORM, the calling Form is closed on the client and the server, and all object properties are destroyed. Consequently, it consumes less memory on the server and client. Returning to the initial Form requires that it be downloaded again to the client, which requires network resources and startup time delays. Use OPEN_FORM to display the next Form in an application unless it is unlikely that the initial form will be called again (such as a login form).

- **Avoid unnecessary graphics and images.** Wherever possible, reduce the number of image items and background images displayed in your applications. Each time an image is displayed to application users, the image must be downloaded from

the application server to the user's Web browser. To display a company logo with your Web application, include the image in the HTML file that downloads at application startup. Do this instead of including it as a background image in the application. As a background image, it must be retrieved from the database or file system and downloaded repeatedly to users' computers.

## 14.3 Web Cache and Forms Integration

Oracle Web Cache can be used as a load balancer with Oracle Forms applications. A Forms client needs to be able to communicate with the same instance of the server process for the duration of a session. Since Forms applications are stateful, Web Cache must be configured for stateful load balancing using its session binding feature.

**Figure 14–3    Web Cache Load Balancing**
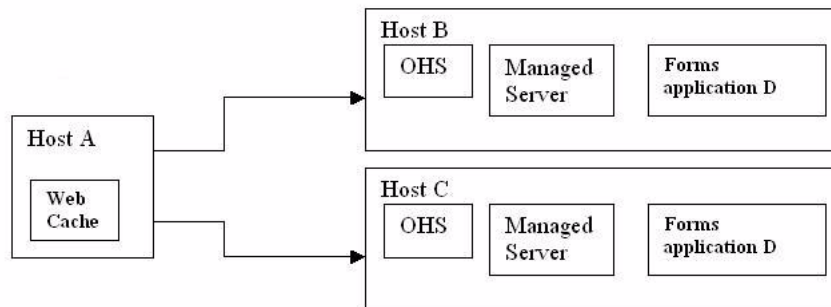


Figure 14–3 assumes a setup where a single Web Cache instance is load balancing two Application Server tiers. This can be described as following:

1. Oracle Web Cache instance running on Host A

2. Oracle HTTP Server instance and Oracle WebLogic Managed Server on Host B running Oracle Forms application D

3. Oracle HTTP Server instance and Oracle WebLogic Managed Server on Host C running Oracle Forms application D

> **Note:**    There can be more Oracle HTTP Server/Oracle WebLogic Managed Server, but only two instance pairs are described here for purpose of simplification. The Oracle HTTP Server/Oracle WebLogic Managed Server is not configured for clustering/active failover because Oracle Forms applications cannot take advantage of Oracle WebLogic Server active failover.

**Prerequisites**

1. Modify the Forms J2EE application deployment descriptor (`weblogic.xml`) to override the default session tracking entries and to enable the cookies for session tracking.  For more information, refer to Section 5.2.4, "Modification of Forms J2EE Application Deployment Descriptors".

   Modify the deployment plan. The following is a sample of the deployment plan with the added entries highlighted in bold:

```
<?xml version='1.0' encoding='UTF-8'?>
<deployment-plan xmlns="http://xmlns.oracle.com/weblogic/deployment-plan"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://xmlns.oracle.com/weblogic/deployment-plan
http://xmlns.oracle.com/weblogic/deployment-plan/1.0/deployment-plan.xsd"
global-variables="false">
  <application-name>formsapp</application-name>
  <variable-definition>
    <variable>
      <name>vd-/scratch/t_work/Oracle/Middleware/as_1/forms</name>
      <value>/scratch/t_work/Oracle/Middleware/as_1/forms</value>
    </variable>
    <variable>
      <name>vd-/scratch/t_work/Oracle/Middleware/user_
projects/domains/ClassicDomain/config/fmwconfig/servers/WLS_
FORMS/applications/formsapp_11.1.1/config/forms</name>
      <value>/scratch/t_work/Oracle/Middleware/user_
projects/domains/ClassicDomain/config/fmwconfig/servers/WLS_
FORMS/applications/formsapp_11.1.1/config/forms</value>
    </variable>
    <variable>
      <name>Frmapp_url-rewriting-enabled_variable</name>
      <value>false</value>
    </variable>
    <variable>
      <name>Frmapp_cookies-enabled_variable</name>
      <value>true</value>
    </variable>
    <variable>
      <name>Frmapp_cookie-http-only_variable</name>
      <value>false</value>
    </variable>
  </variable-definition>
  <module-override>
    <module-name>formsapp.ear</module-name>
    <module-type>ear</module-type>
    <module-descriptor external="false">
      <root-element>weblogic-application</root-element>
      <uri>META-INF/weblogic-application.xml</uri>
    </module-descriptor>
    <module-descriptor external="false">
      <root-element>application</root-element>
      <uri>META-INF/application.xml</uri>
    </module-descriptor>
    <module-descriptor external="true">
      <root-element>wldf-resource</root-element>
      <uri>META-INF/weblogic-diagnostics.xml</uri>
    </module-descriptor>
  </module-override>
  <module-override>
    <module-name>formsweb.war</module-name>
    <module-type>war</module-type>
    <module-descriptor external="false">
      <root-element>weblogic-web-app</root-element>
      <uri>WEB-INF/weblogic.xml</uri>
      <variable-assignment>
        <name>vd-/scratch/t_work/Oracle/Middleware/as_1/forms</name>
<xpath>/weblogic-web-app/virtual-directory-mapping/[url-pattern="java/*"]/local
-path</xpath>
      </variable-assignment>
```

```
            <variable-assignment>
              <name>vd-/scratch/t_work/Oracle/Middleware/as_1/forms</name>
<xpath>/weblogic-web-app/virtual-directory-mapping/[url-pattern="webutil/*"]/lo
cal-path</xpath>
            </variable-assignment>
            <variable-assignment>
              <name>vd-/scratch/t_work/Oracle/Middleware/user_
projects/domains/ClassicDomain/config/fmwconfig/servers/WLS_
FORMS/applications/formsapp_11.1.1/config/forms</name>
<xpath>/weblogic-web-app/virtual-directory-mapping/[url-pattern="registry/*"]/l
ocal-path</xpath>
            </variable-assignment>
            <variable-assignment>
              <name>Frmapp_url-rewriting-enabled_variable</name>
              <xpath>/weblogic-web-app/session-descriptor/url-rewriting-enabled
</xpath>
            </variable-assignment>
            <variable-assignment>
              <name>Frmapp_cookies-enabled_variable</name>
              <xpath>/weblogic-web-app/session-descriptor/cookies-enabled</xpath>
            </variable-assignment>
            <variable-assignment>
              <name>Frmapp_cookie-http-only_variable</name>
             <xpath>/weblogic-web-app/session-descriptor/cookie-http-only</xpath>
            </variable-assignment>
        </module-descriptor>
        <module-descriptor external="false">
            <root-element>web-app</root-element>
            <uri>WEB-INF/web.xml</uri>
        </module-descriptor>
      </module-override>
</deployment-plan>
```

2. Add the virtual host directives to `forms.conf` as shown below:

```
<VirtualHost *:8888>
      ServerName hostA:8090
      UseCanonicalName On
      CookieTracking On
      WLCookieName cookieName
#
# virtual mapping for the /forms/html mapping.
#
RewriteEngine on
RewriteRule ^/forms/html/(..*) /workaroundhtml/$1 [PT]
AliasMatch ^/workaroundhtml/(..*) ". . . . . .
/config/FormsComponent/forms/html/$1"

<Location /forms>
        SetHandler weblogic-handler
        WebLogicCluster    HostB:9001
        DynamicServerList OFF
</Location>
</VirtualHost>
```

> **Note:** The `ServerName` entry must have the Web Cache hostname and port number. The value of the `WLCookieName` entry must be unique across hosts. For example, `cookieHostA`, `cookieHostB`, and so on. The port number 8888 in the `VirtualHost` directive corresponds to the OHS HTTP port. You can use `$ORACLE_INSTANCE/bin/opmnctl status -l` to obtain the actual OHS HTTP port value.

3. The Application Server hosts (Host B and Host C) must have the same FMW patch set version.

4. Ensure that the Forms configuration files are synchronized across the Application Server hosts. This means that you must create matching entries in the Forms configuration files (`formsweb.cfg`, `default.env`, `Registry.dat`, and so on) across all the Application Server hosts.

**To Configure Session Binding in Web Cache:**

1. Set up Web Cache running on Host A as described in *Configuring Oracle Web Cache as a Software Load Balancer* in *Oracle Fusion Middleware Administrator's Guide for Oracle Web Cache*.

2. Add the Application servers, Host B and Host C as Origin Servers as described in *Specify Origin Server Settings* in *Oracle Fusion Middleware Administrator's Guide for Oracle Web Cache*.

   Ensure to specify the URL as `/forms/lservlet` in the **Ping URL** field.

3. Create ordered mappings of sites to origin servers, Host B and Host C as described in *Map Site Definitions to Origin Servers* in *Oracle Fusion Middleware Administrator's Guide for Oracle Web Cache*.

4. Configure session binding in Web Cache as described in *Configuring Session Binding* in *Oracle Fusion Middleware Administrator's Guide for Oracle Web Cache*.

   Ensure to select the **Cookie based session binding with any Set-Cookie** option.

**To test the setup:**

1. Using a browser, point it to the Web Cache host and access Oracle Forms application D. Ensure that the application works as expected. Keep the browser window open.

2. Use the OHS access logs to identify the Oracle HTTP Server/Oracle WebLogic Managed Server that handled the requests. For example, assume this is Host B and shut down the Oracle HTTP Server/WebLogic Managed Server on that host. Now only the Oracle HTTP Server/WebLogic Managed Server running on Host C will be accessible.

3. Using the same browser that is running the Oracle Forms client, access Oracle Forms application D again. The request will fail, and the Forms client will lose its session. Note that Oracle Forms session state is not replicated among Oracle WebLogic Managed Server.

4. Next, clear the browser cookies and open a browser window. Point it to the Web Cache host and access Oracle Forms application D. Web Cache will direct the requests to the remaining Oracle HTTP Server/WebLogic Managed Server running on Host C. Ensure that the application works as expected.

5. Restart the Oracle HTTP Server/WebLogic Managed Server on Host B. Using a browser, log on to the Web Cache Manager.

6. In the Oracle Enterprise Manager navigator panel, select **Web Cache** and navigate to the **Home** page.

7. In the **Web Cache** Home page, ensure that Host B listed under Origin Servers is checked.

> **Note:** For information about load balancing scenarios and setting up clustered Web Cache, see *Oracle Fusion Middleware High Availability Guide*. For more information about Web Cache, see *Oracle Fusion Middleware Web Cache Administrator's Guide*.

# 15

# Forms Diagnostics Agent

Forms Diagnostics Agent or Forms Metrics Agent enables the user to analyze various performance-related information about Forms applications running in your environment. This agent accesses the metrics data (available in DMS) at regular time intervals and populates the database tables. This process allows the user to access the data collected as historical data.

Forms Diagnostics agent works only with Oracle Forms 11*g*R2. The deployment of this application is optional. The agent application provides an interactive interface where the user can specify the frequency of data collection and also control the starting and stopping of data collection. This can be achieved by performing the following tasks:

- Install and Configure Oracle Forms 11gR2
- Setting up the Database Schema
- Setting up a Data Source in WebLogic
- Deploying Forms Diagnostics Agent
- Managing the Data Collection
- Using the Agent Application
- Limitations of the Agent Application

## 15.1 Install and Configure Oracle Forms 11gR2

Since Forms Diagnostics Agent works only with Oracle forms 11*g* Release 2, you must install and configure this version.

For information about installing and configuring Oracle Forms 11*g*R2, see *Oracle Fusion Middleware Installation Guide for Oracle Forms and Reports*.

## 15.2 Setting up the Database Schema

In order to set up DB schema for Forms Diagnostics Agent, you must create a user and schema in the database. The user can choose a database instance of their choice.  There is no special database that is installed with Forms or the diagnostic agent.

**Create a User in Database**
To create a user in the database, perform the following steps:

> **Note:** Before creating a user in the database, ensure that the user name provided by you is new and does not already exist. This is because the .sql script (used to create the user in the database) overwrites the user (user name provided during the creation) with the new user.

1. Log in to the database as sysdba as shown below:

   sqlplus sys/<sys-password>@<DB> as sysdba

2. Run the following script:

   @ORACLE_HOME/forms/forms_create_diagnostics_user.sql.

3. The user must enter the userID and password.

   The user is thus created.

**Create a Schema in Database**

To create a schema in the database, perform the following steps:

> **Note:** Before creating a schema in the database, ensure that the user name provided by you is new and does not already exist. This is because the .sql script (used to create the schema in the database) overwrites the schema (user name provided during the creation) with the new schema.

1. Log in to the database as the user that you created in the above steps:

   sqlplus <user>/<password>@<DB>

2. Run the following script:

   @ORACLE_HOME/forms/forms_create_diagnostics_schema.sql.

3. The schema is thus created.

## 15.3 Setting up a Data Source in WebLogic

After setting up the database to work with the Forms Diagnostics Agent, you must set up a data source using the Weblogic console. To achieve this, perform the following steps:

1. Log into the WebLogic console.

2. In the left navigation panel, select **Services** and navigate to **Data Sources**.

   Click **Lock and Edit** in the Change Center window to make changes.

3. In the **Summary of JDBC Data Sources** page, click **Configuration**.

   In the **Data Sources** table, click **New** and select **Generic Data Sources** from the list.

4. Enter the values for the following parameters:

   **name for the JDBC Data Sources**

   The user can enter any name.

   **JNDI name**

oracle/forms/agentDS

**Database Type**

Choose type of database that you used to create user and schema in the previous steps.

Click **Next**. The **Create a new JDBC data sources** page appears.

5. Select **Database driver** from the list of drivers available for the type of database you have selected. Click **Next**.

6. Enter the values for the following parameters:

    **Database Name**

    **Host Name**

    **Port**

    **Database User Name**

    Enter the user name that you used while creating a user in the database in the steps above.

    **Password**

    Enter the password that you used while creating a user in the database in the steps above.

    Click **Next**.

7. In the next page, click **Test Configurations** at the top left corner to check if the database has been configured successfully.

    Click **Next**.

8. Select **Admin Server** as a target to deploy the data source.

    Click **Finish**.

9. Click **Activate Changes** in the Change Center window to save changes.

    You have now set up a JDBC data source.

## 15.4 Deploying Forms Diagnostics Agent

After setting up a data source in Weblogic, Forms Diagnostics agent must be deployed to the Weblogic Admin Server. This can be achieved by performing the following steps in the Weblogic console:

1. Log into the Weblogic Console.

2. In the left navigation panel, select **Deployments**.

    Click **Lock and Edit** in the Change Center window to make changes.

3. In the Summary of Deployments page, click **Install**.

    The **Install Application Assistant** page appears.

4. Enter the path of the .war file as shown below:

    ```
    ORACLE_HOME/forms/j2ee
    ```

    This is the location of the `formsagentapp.war` file.

5. Select the `formsagentapp.war` file. Click **Next**.

The **Choose Targeting Style** page appears.

6. Select **Install this deployment as an application**.

7. Select **Admin Server** as a target to deploy Forms Diagnostics Agent. Click **Next**.

8. Leave the optional settings at their default values and click **Finish**.

   Click **Activate Changes** in the Change Center window to save changes. The Forms Diagnostics agent has been successfully deployed to the Weblogic Admin Server.

   To start the application, select **formsagentapp** from the list of deployed applications and Click **Start**.

## 15.5 Managing the Data Collection

The agent application allows the users to manage data collection using an interface. The user can specify the frequency of data collection and control the starting or stopping of data collection. This can be achieved through the following steps:

1. Log in to the agent console by using the following url:

   ```
   http://<host>:<admin port>/formsagent/AgentConsole.jsp
   ```

2. Enter the user ID and password.

   Any user with administrator's privileges can log in to the console.

3. Enter a value for the **Frequency of Data Collection**. This parameter is the time difference between two consecutive data collections.

   The default value is 10 minutes. The minimum value should be one minute.

4. Click **Start**.

   You will see a message indicating that the Forms Diagnostics Agent is running. Click **Stop** whenever you want to stop the collection of metrics by the agent.

## 15.6 Using the Agent Application

When the user prompts the agent application to start the collection of metrics, the agent collects the metrics from DMS and populates the database tables. The user can access this collected metrics in the database tables. To bring this collected metrics into use, the user can create a frontend application which will be able to read this data and analyze the historical performance of Forms applications running in your environment by preparing charts, graphs, etc.

The primary and foreign key in each table has been mentioned in the respective tables below. The following are the database tables that get popluated during the collection of metrics by the Forms Diagnostics agent:

*Table 15–1    ADMIN_SERVER Database Table*

| Serial Number | Column Name | Sample Value | Description |
|---|---|---|---|
| 01 | AGENT_ID | 1 | AGENT_ID is the primary key in the ADMIN_SERVER database table. |
| | | | ID of the agent application. Any integer value beginning with 1 |

*Table 15–1 (Cont.) ADMIN_SERVER Database Table*

| Serial Number | Column Name | Sample Value | Description |
| --- | --- | --- | --- |
| 02 | ADMIN_HOSTNAME | myhost.mydomain.com | Name of the machine where admin server is deployed |
| 03 | ADMIN_PORT | 7001 | Port of the admin server |

*Table 15–2 AGENT Database Table*

| Serial Number | Column Name | Sample Value | Description |
| --- | --- | --- | --- |
| 01 | AGENT_STATUS | Running | Status of the Agent |
| 02 | REAL_TIME | 2009.07.23 at 17:11:41 | Date and time when status is recorded<br><br>All time entries are in UTC/GMT |
| 03 | SEQUENCE_ID | 205 | ID created by the agent each time the agent status is recorded |
| 04 | FREQUENCY | 40 | Time difference (in minutes) between two consecutive data collections |
| 05 | AGENT_ID | 1 | AGENT_ID is a foreign key in this database table. It refers to AGENT_ID in the ADMIN_SERVER database table<br><br>ID of the agent application. Any integer value beginning with 1 |

*Table 15–3 FRM_DB Database Table*

| Serial Number | Column Name | Sample Value | Description |
| --- | --- | --- | --- |
| 01 | FRM_DB_ID | 8769 | FRM_DB_ID is the primary key in the FRM_DB database table.<br><br>ID assigned to the database that is used by the Forms application |
| 02 | DB_NAME | v11g | The database to which frmweb is connected. If it is connected to v11g, the this field will display v11g. This can be NULL when frmweb is not connected to any database |

**Table 15–3 (Cont.) FRM_DB Database Table**

| Serial Number | Column Name | Sample Value | Description |
|---|---|---|---|
| 03 | TNS_ENTRY | (DESCRIPTION = (ADDRESS_LIST = (ADDRESS = (PROTOCOL = TCP)(HOST = sample.host.com)(PORT = 1521))) (CONNECT_DATA = (SERVICE_NAME = v11g))) | TNS entry of the database to which frmweb is connected |
| 04 | USER_NAME | scott | The database user who has logged in |

**Table 15–4 FRM_DB_LOGIN Database Table**

| Serial Number | Column Name | Sample Value | Description |
|---|---|---|---|
| 01 | FRM_DB_LOGIN_ID | 345 | FRM_DB_LOGIN_ID is the primary key in the FRM_DB_LOGIN database table. ID assigned to each row in the table |
| 02 | FRM_RUNTIME_ID | 107 | FRM_RUNTIME_ID is a foreign key in this database table. It refers to FRM_RUNTIME_ID in the FRM_RUNTIME database table |
| 03 | FRM_DB_ID | 1025 | FRM_DB_ID is a foreign key in this database table. It refers to FRM_DB_ID in the FRM_DB database table. ID assigned to the database that is used by the Forms application |
| 04 | REAL_TIME | 2009.07.23 at 17:11:41 | Date and time when status is recorded and metric data collected to update the table. All time entries are in UTC/GMT |

**Table 15–5 FRM_RUNTIME Database Table**

| Serial Number | Column Name | Sample Value | Description |
|---|---|---|---|
| 01 | FRM_RUNTIME_ID | 107 | FRM_RUNTIME_ID is a primary key in the FRM_RUNTIME database table. ID that identifies the Forms process |

*Table 15–5   (Cont.) FRM_RUNTIME Database Table*

| Serial Number | Column Name | Sample Value | Description |
|---|---|---|---|
| 02 | WLS_APP_ID | 5005 | WLS_APP_ID is a foreign key in this database table. It refers to WLS_APP_ID in the WLS_APP database table |
| | | | Determines the Forms application in the WLS_APP table |
| 03 | FRM_USER_ID | 1310 | FRM_USER_ID is a foreign key in this database table. It refers to FRM_USER_ID in the FRM_USER database table |
| | | | ID assigned to the Forms client or client instance |
| 04 | CONFIG_VALUE | | Configuration section name from `formsweb.cfg` |
| 05 | CONNECT_TIME | 2009.07.23 at 17:11:41 | Date and time when frmweb is spawned |
| | | | All time entries are in UTC/GMT |
| 06 | DISCONNECT_TIME | 2009.07.23 at 18:15:43 | Date and time when frmweb terminates |
| | | | All time entries are in UTC/GMT |
| 07 | STARTING_FORM_NAME | emp | Name of starting form |
| 08 | PROCESS_ID | 8020 | Process Id of frmweb on the middle tier machine |
| 09 | FRM_STATUS | Running / Exited | Status of frmweb |
| 10 | FRM_CPU_TIME_ON_EXIT | 256 | CPU time on exit of frmweb |
| 11 | FRM_PRIVATE_MEMORY_ON_EXIT | 6385 | Memory used by the Forms process at the time of exit |
| 12 | FRM_EXIT_CODE | | |

*Table 15–6   FRM_TRACE Database Table*

| Serial Number | Column Name | Sample Value | Description |
|---|---|---|---|
| 01 | FRM_TRACE_ID | 2679 | FRM_TRACE_ID is the primary key in this database table. |
| | | | ID assigned to the rows in the FRM_TRACE database table |
| 02 | TRACE_FILE | forms_1055.trc | Name of the trace file when `ftrace` is enabled. It can be NULL if `ftrace` is disabled |
| 03 | TRACING | mytrace | The name of the trace group selected by the application |

*Table 15–7    FRM_TRACE_USE Database Table*

| Serial Number | Column Name | Sample Value | Description |
|---|---|---|---|
| 01 | FRM_TRACE_USE_ID | 1906 | FRM_TRACE_USE_ID is the primary key in this database table. |
| | | | ID assigned to the rows in the database table |
| 02 | FRM_RUNTIME_ID | 107 | FRM_RUNTIME_ID is a foreign key in this database table. It refers to FRM_RUNTIME_ID in the FRM_RUNTIME database table |
| | | | ID that identifies the Forms process |
| 03 | FRM_TRACE_ID | 2679 | FRM_TRACE_ID is a foreign key in this database table. It refers to FRM_TRACE_ID in the FRM_TRACE database table |
| | | | ID assigned to the row in the database table |
| 04 | REAL_TIME | 2009.07.23 at 17:11:41 | Date and time when status is recorded and metric data collected to update the table |
| | | | All time entries are in UTC/GMT |

*Table 15–8    FRM_USER Database Table*

| Serial Number | Column Name | Sample Value | Description |
|---|---|---|---|
| 01 | FRM_USER_ID | 1310 | FRM_USER_ID is the primary key in this database table. |
| | | | ID assigned to the Forms client or client instance |
| 02 | CLIENT_IP | 255.255.255.255 | IP address of client machine from where the browser was launched and through which the user connected to the middle tier |
| 03 | SSO_USERID | fname.lname@myapp.com | Single Sign-On ID of the user who logged in. |

*Table 15–9   HISTORY Database Table*

| Serial Number | Column Name | Sample Value | Description |
|---|---|---|---|
| 01 | FRM_RUNTIME_ID | 107 | FRM_RUNTIME_ID is a foreign key in this database table. It refers to FRM_RUNTIME_ID in the FRM_RUNTIME database table |
| | | | ID that identifies the Forms process |
| 02 | REAL_TIME | 2009.07.23 at 17:11:41 | Date and time when the snapshot is taken |
| | | | All time entries are in UTC/GMT |
| 03 | SEQUENCE_ID | 205 | ID created by the agent each time agent status is recorded. |
| 04 | FRM_BYTES_SENT | 400 | Number of bytes sent from the server to the client for this process so far |
| 05 | FRM_BYTES_SENT_DELTA | 37 | Difference in the number of bytes sent from the server to the client for this process since the previous reading of the agent was taken |
| 06 | FRM_BYTES_RECEIVED | 200 | Number of bytes sent from the client to the sever for this process so far |
| 07 | FRM_BYTES_RECEIVED_DELTA | 23 | Difference in the number of bytes sent from the client to the server for this process since the previous reading of the agent was taken |
| 08 | FRM_NETWORK_ROUND_TRIPS | 30 | Number of network round trips between the client and the server for this process so far |
| 09 | FRM_NETWORK_ROUND_TRIPS_DELTA | 3 | Difference in the number of network roundtrips between the client and the server for this process since the previous reading of the agent was taken |
| 10 | FRM_CPU_TIME | 230 | Total processing time taken by frmweb (in milliseconds) for this process so far |
| 11 | FRM_CPU_TIME_DELTA | 47 | Difference in the value of FRM_CPU_TIME since the previous reading of the agent was taken |
| 12 | FRM_PRIVATE_MEMORY | 7998 | Memory used by the Forms process at the time when snapshot was taken. |
| 13 | ITERATION | 50 | Number of times data is collected into the database table. |

*Table 15–10    WLS_APP Database Table*

| Serial Number | Column Name | Sample Value | Description |
|---|---|---|---|
| 01 | WLS_APP_ID | 5005 | WLS_APP_ID is the primary key in this database table. |
| | | | Determines the Forms application in the WLS_APP table |
| 02 | SERVER_TYPE | MANAGED | Type of the server (For example, MANAGED or ADMIN) |
| 03 | SERVER_NAME | WLS_FORMS | Name of the server |
| 04 | DEPLOYED_APPLN_ NAME | formsapp | Forms Application name |
| 05 | FORMS_HOSTNAME | host52.example.com | Middle tier machine on which Forms runtime is running |
| 06 | INSTANCE_HOME_ NAME | asinst_1 | Name of the FMW instance home, where Forms runtime is deployed |
| 07 | CLUSTER_NAME | cluster_xyz | Name of the cluster where the Forms application is deployed |
| 08 | AGENT_ID | 1 | AGENT_ID is a foreign key in this database table. It refers to AGENT_ID in the ADMIN_ SERVER database table. |
| | | | ID of the agent application. Any integer value beginning with 1 |

## 15.7  Limitations of the Agent Application

Forms Diagnostics agent has certain limitations on its deployment and usage. They are as follows:

- The deployment of the Forms Diagnostics Agent application is optional. In case you want to analyze performance-related information about Forms applications, you must deploy Forms Diagnostics Agent manually postinstallation.

- The agent application must be deployed to the Admin Server only. The agent application collects information about all Forms sessions that are running in the WLS domain of the Admin Server.

- For the agent to be able to collect information about Forms applications, opmn must be up and running on all AS instances.

- For the agent to be able to access the metrics data (available in DMS), the DMS application must be up and running.

- The schema is designed to be functional only on one domain at any given time. You cannot use the same schema for multiple agents (running in separate domains).

- Do not set the frequency of data collection to a small value. Setting the frequency of data collection to a small value slows down the production environment and causes excessive, needless data collection.

# A

# Troubleshooting Oracle Forms Services

This chapter contains the following:

This chapter provides information to help you resolve problems that might occur when you run an application over the Web using Oracle Forms. It contains an outline of common causes for errors, the method you can use to verify your installation, and the tools and techniques provided to diagnose problems.

This chapter is also a subset of the whitepaper *Oracle Forms Diagnostic Techniques* that can be found at
http://www.oracle.com/technetwork/developer-tools/forms/overview/index.html.

## A.1 Verifying The Installation

If there is something wrong with the installation, then it will result in faulty configuration and Oracle Forms will not run correctly. After the Oracle Universal Installer indicates that Fusion Middleware Control was successfully installed, you can verify whether Oracle Forms Services is correctly configured or not. You can use these tools:

### A.1.1 Use The Web Form Tester

The Web Form Tester is available with your Oracle Fusion Middleware installation. To verify whether the Oracle installation and configuration of Forms Services is correct, run the Web Form Tester on the middle tier. The following is an example of how this can be done on a Windows computer.

1. Start the Admin server for the WebLogic Server domain by selecting **Start** |
   **Program Files** | **Oracle WebLogic Server** | **User Projects** | **Domain** | **Start
   Admin Server for WLS Domain**, if it is not already started.

2. If the managed server is not up, perform the following steps:

   1. Start the node manager by selecting **Start** | **Program Files** | **Oracle WebLogic**
      | **WebLogic Server 11gR1** | **Tools** | **Node Manager**, if it is not already
      started.

   2. Start Forms Services from the WebLogic Administrator Console.

3. Open an instance of the browser by typing `<ORACLE_
   HOME>/tools/web/html/runform.htm` for the URL and press ENTER.
   Replace ORACLE_HOME with your actual Oracle home for Oracle Fusion
   Middleware.

4. Alternatively, you can run the Web Form Tester by selecting **Start** | **Program Files**
   | **<Oracle_Home>** | **Forms Services** | **Run a Form on the Web** from the
   Windows Start menu for Oracle Fusion Middleware.

5. Enter the Web port and click the **Run Form** button. See Section A.1.2, "Find Port
   Information" to learn how to find out the Web port.

6. If the installation of Oracle Fusion Middleware is correct, you will see a success
   message in the Web browser. Also, it can be tested from a client computer whether
   the basic Forms setup in Oracle Fusion Middleware on the middle tier is installed
   correctly or not by the installer. You can run the test form from any client
   computer by running it from the browser with the URL
   `http://example.com:NNNN/forms/frmservlet?form=test.fmx`.

## A.1.2 Find Port Information

When in doubt or you need to know what port numbers to use to run Forms after
installation, you can look at port information in the file `<ORACLE_
HOME>/install/portlist.ini`. Use the appropriate port numbers for your
installation.

# A.2 Diagnosing FRM-XXXXX Errors

Use these tools to diagnose and resolve FRM-XXXXX errors:

- Section A.2.1, "The Oracle Forms Applet"

## A.2.1 The Oracle Forms Applet

The brief message about the FRM error should help in identifying the basic cause of
the problem. Often, everything required to identify the cause an FRM error is
contained in the error reported by the Forms applet. When a FRM error is raised, the
error dialog will have a **Details** button. Pressing the 'Details' button will show the
current Java stack. The exact stack is tied to the root cause and the version of Oracle
Forms. This is due to the differing package structure used for the applet class files in
the different releases.

# A.3 Diagnosing Server Crashes with Stack Traces

This section contains the following:

- Section A.3.1, "About Stack Traces"

- Section A.3.2, "Configuring and Using Stack Traces"

If the Forms web runtime terminates unexpectedly, then it writes a stack trace to the directory `$ORACLE_INSTANCE/FormsComponent/forms/trace`. The filename will have the format `<forms_runtime_process>_dump_<process id>`.The dump file contains a stack trace of the running process, and shows the last successful operation performed by Forms. This core file can be used to assemble a stack trace with symbol names using GNU Debugger, dbx or similar debugging tool on the machine where the dump occurred.

## A.3.1 About Stack Traces

A stack trace is useful for two reasons:

- The information in the stack can be used to identify a known issue. It is not 100% reliable, but an identical stack trace is a good indicator of a matching problem. Even if it is not the same, there may be a workaround or patch for an existing bug that can be tested.

- If the problem is not a known bug, then the stack may provide valuable information to assist development efforts to pinpoint the cause.

## A.3.2 Configuring and Using Stack Traces

This section contains the following:

- Section A.3.2.1, "Verifying the Environment"
- Section A.3.2.2, "Understanding UNIX Stack Traces"
- Section A.3.2.3, "Understanding Windows Stack Traces"

### A.3.2.1 Verifying the Environment

In order to test stack tracing on UNIX or Windows you can set the environment variable `FORMS_DELIBERATECRASH`. As the name suggests, setting this will cause the forms runtime process to crash. Oracle Forms currently recognizes two settings: `1` and `2`. If `FORMS_DELIBERATECRASH` is set to `1` then forms will crash at runtime whenever the BELL Built-in is executed. If it is set to `2` then forms will crash at runtime whenever a when-button-pressed trigger is fired. This environment variable can be set in the environment (for example, `default.env`) file.

### A.3.2.2 Understanding UNIX Stack Traces

In a UNIX stack trace, the top two functions `siehjmpterm()` and `sigacthandler()` are the signal handling code - these functions will often be present in the stack trace. To see the function the program was in when the error occurred you need to read further down the stack.

If you set `FORMS_CATCHTERM=0` the two functions do not show up in the dump file. The stack trace is displayed without the crash handling symbols.

### A.3.2.3 Understanding Windows Stack Traces

Stack tracing works differently on UNIX and on Windows. The symbol information is contained inside the executable files and shared libraries on Unix. On Windows this information is stripped out at link time and is in the form of binary `.sym` files. There should be one `.sym` file for every Oracle Forms executable or DLL. The `.sym` files are installed by default. On Windows the files are located in the `ORACLE_HOME\bin` directory. The mechanism on Windows platforms is such that in the event of a crash

the Forms runtime process reads all the .sym files that correspond to the forms executable files loaded into memory. It then uses the information in the .sym files to lookup the symbol name.

# A.4 Diagnosing Client Crashes

This section contains the following:

- Section A.4.1, "About Diagnosing Client Crashes"
- Section A.4.2, "Diagnosing Hanging Applications"

## A.4.1 About Diagnosing Client Crashes

If the Forms applet disappears unexpectedly, accompanied by a dialog indicating a fatal error, then the Forms applet has crashed. On Windows, a crash will result in the operating system raising an 'illegal operation' dialog, or may cause the "Not responding" flag in Task Manager.To verify the crash, check for a stack trace file on the client. If the client has crashed then a file with the .rpt extension will be created in the same directory as the executable. The root of the filename will be the name of the executable.

Sometimes the applet may appear to have crashed, but no corresponding .rpt file can be found. In this case it is likely that the Oracle Forms has unexpectedly disconnected from the client. The applet will still be running, but it has shutdown all the Forms windows, giving the appearance of a client crash.

## A.4.2 Diagnosing Hanging Applications

If the client appears to hang then it is important to verify that the server process is still alive. If the server process has not crashed, but the client no longer appears to respond to user interaction then the application is said to be hanging.

In such cases a thread dump can point to the deadlock.  A thread dump can be obtained by pressing t in the Java console. This displays a list of all the threads running in the client JVM.

The information contained in the dump file is extremely useful to Oracle development, and should be included in any bug filed to report the problem.

### A.4.2.1 Causes of Hanging Applications

One cause could be a mismatch between the Java class files and the Oracle Forms version. Communication between the applet and the Forms runtime process is based on message ID. If these message ID's are out of sync, then the applet may not understand an instruction from the server, and vice versa. If you are using Jar files, then try with the <ARCHIVE> tag removed. If the problem persists then pull the correct class files off the installation/patch CD by hand.

Another cause is that the Forms Runtime process may have died. Check if the Forms Runtime process on the server is still alive. Check that the FORMS_TIMEOUT parameter is set. It defines how long the server should wait for a ping from the Oracle Forms client, only cleaning up the runtime process when there has been no activity from the Forms client for the specified time. The client sends out a HEARTBEAT every two minutes by default. If FORMS_TIMEOUT is set to two minutes or longer, the server will stay up as long as it hears a HEARTBEAT from the client. Set to shorter than the HEARTBEAT interval, it will shut down after the interval specified in FORMS_ TIMEOUT. You can set the interval by setting the HEARTBEAT applet parameter in

`formsweb.cfg`. For more information, see Section 8.6.3, "Configuring Asynchronous Communication." Although this is primarily intended to prevent orphaned server processes, it can also prevent the unwanted premature cleanup of server processes.

## A.5 Forms Trace and Servlet Logging Tools

Forms Trace and Servlet Logging are two more tools to use in troubleshooting your Oracle Forms Environment. For more information on configuring and using Forms Trace, see Chapter 12.1, "About Forms Trace" and Chapter 12.6, "Taking Advantage of Oracle Diagnostics and Logging Tools".

## A.6 Resolving Memory Problems

This section contains the following:

- Section A.6.1, "How Java Uses Memory"
- Section A.6.2, "Setting the Initial Java Heap"
- Section A.6.3, "About Memory Leaks"
- Section A.6.4, "Improving Performance with Caching"

### A.6.1 How Java Uses Memory

Like all software programs, a Java applet uses memory. For Java, the language specification requires a 'garbage collector', which is in an internal memory manager for the Java Virtual Machine (JVM). When a Java program needs memory, it requests this memory from the JVM. If there is no memory left, then the JVM will attempt to free some memory by using the garbage collector. The garbage collector will try to release memory that is no longer required to run the program back to the JVM. If there is still insufficient memory to perform the required task then the JVM will attempt to get more memory from the operating system. If that memory allocation fails, then the Java program will be unable to continue.

### A.6.2 Setting the Initial Java Heap

You can specify the initial Java Heap (the memory used by the JVM) for your application through Fusion Middleware Control. For the client, you can change the setting in the Java control panel after you've installed the Oracle Java Plug-in.

> **Note:** The JVM will only use the memory it is told it is allowed to use. Even if you have memory available with the operating system, the JVM will not use it if told not to.

### A.6.3 About Memory Leaks

A *memory leak* is an error in a program's dynamic-store allocation logic that causes it to fail to reclaim discarded memory, leading to eventual collapse due to memory exhaustion.

For example, when a program runs it may need to allocate some memory to perform a particular task. If the program has finished with that memory and no longer has any use for it, but fails to make that memory available to other programs running on the computer, then it is said to have leaked the memory.

A typical method used to spot memory leaks is to repeat a series of steps, and observe the memory in use by the application - if the memory usage continues to rise with each iteration, then the assumption is often that the program has a memory leak.

However, some complex applications may choose to retain control of memory it has previously allocated so that it can reuse it at a later point - memory allocation can be an expensive operation, and if the program expects that it will need more memory later it may be more efficient to keep the unused memory available for reuse.

### A.6.3.1  Memory Leaks in Java

The Java language specification demands that the JVM has a garbage collector. In Java, the programmer allocates memory by creating a new object. There is no way to de-allocate that memory. Periodically the garbage collector sweeps through the memory allocated to the program, and determines which objects it can safely destroy, therefore releasing the memory. To determine which objects it can safely destroy, the garbage collector uses a 'mark and sweep' algorithm. The garbage collector scans the dynamically allocated memory for objects, marking those which still have active references to them.

After all possible paths to objects have been investigated, unmarked objects that are known to be no longer needed can be garbage collected. A common myth with Java programming is that the presence of a garbage collector means that there can be no memory leaks. This is not true because the garbage collector simply marks those objects, which have active references, and destroys those that do not. It is possible to have an active reference to an object that is no longer needed. This is a memory leak in Java. The solution to the leak is to destroy the references to the object once it is no longer needed so that the garbage collector can identify it as safe to destroy. If a memory leak exists in a Java program, then calling the garbage collector more frequently will not help.

To complicate matters further, the JVM may choose not to release unused memory back to the operating system. In the real world this seldom matters, as most programs will typically require more memory at some point in the near future and can reuse the free memory in the JVM. However, it is worth bearing in mind that not all the memory allocated to the JVM will be in use by the program running in the JVM.

### A.6.3.2  Identifying Memory Leaks

Typically, if a growth in memory usage is observed each time a particular series of operations is performed, then it is a memory leak. The ideal proof is to:

1.  Get the form into an initial base state, and record the memory usage,

2.  Perform a series of steps to illustrate the problem,

3.  Return to the initial base state, and record the memory usage.

By repeating steps 2 and 3, it is possible to determine whether there is a steady memory leak or not. If the growth in memory is small over a large number of iterations, then it may not be a leak at all; it could be that the JVM is retaining unused memory, or the garbage collector is not activating as frequently as expected.

## A.6.4  Improving Performance with Caching

When any Java program runs, the Java Virtual Machine needs to load class files. When running over the Internet, the time taken to download a class file each time the program runs can lead to performance problems. In order to solve this download problem, the JDK supports Java Archive (Jar) files. A Jar file is simply a collection of

class files bundled into one compressed file. Typically, the size of the Jar file will be much smaller than the combined size of the class files it contains.

When the JVM first references a class, it checks the local computer to see if any of the previously cached Jar files contain this class. If the class does exist in one of the pre-cached Jar files, then the JVM checks to see if there is a newer version of this Jar file on the application server. If there is a newer Jar file available then the new copy of the Jar file is downloaded to the client cache. If the cached Jar file is up to date, then the class file is loaded from the cached Jar file rather than from over the network.

Caching is important because if the application Jar files do not change, then after the application has run once, and all the Jar files required have been cached on the client, then subsequent invocations of the application will always load the classes from the local cached copies. This can lead to significant performance improvements in the startup time for the application. If new classes are needed to run a specific part of the application, these will be downloaded as required.

## A.7 Troubleshooting Tips

The following troubleshooting list will help you deal with complex issues, but it is not a definitive guide to problem solving or a guaranteed set of solutions to your Oracle Forms environment.

### Be methodical

Do not immediately leap to the area you believe to be the cause based on a hunch, or a guess - make sure you eliminate the other possibilities first. An easy trap to fall into is that of spending long periods of time trying to find evidence to support your theory, rather than concentrating on what the evidence shows. Do not overlook the trivial or the obvious.

### Divide the problem into sections

- Chop the problem into manageable sections - this helps eliminate whole areas from investigation. As you investigate an area and satisfy yourself that the problem does not lie there, you can proceed to the next section. An approach to diagnosing a problem that is often successful is to reduce it to its essential parts. This will be important if you need to discuss the problem with Oracle Support Services to obtain a solution.

- Define what happens, when it happens, how often it happens. Of equal importance is, understanding what does not happen, when it does not happen etc. For example, if a group of users in the same building all get the problem, and it always happens between 9 and 10am, it is just as important to know that it never reproduces in another building, or after 10pm. Perhaps the users only use a particular Form between 9 and 10, or the load on the system is highest between 9 and 10am.

### Read the error messages.

It sounds obvious, but often the solution information is within the error text. This document will help you understand the error messages, and help identify what action to take.

### Make sure you can reproduce the problem, if possible

If you can reproduce the problem yourself, you may notice some behavior that the end user never spotted - perhaps it had always happened, so they simply assumed it was

meant to happen. If you can reproduce the problem then you have already started the first step to resolve it.

**Make sure you understand the tools you are trying to use**

If you decide to use a diagnostic tool, make sure you know how to use it, and how to interpret the data it produces. Time spent in investigating the usage of a tool before the problem happens is time well invested. Make time to learn the tool as well.

## A.8 Need More Help?

In case the information in the previous sections was not sufficient, you can find more solutions on My Oracle Support (formerly Oracle*MetaLink*), http://support.oracle.com. If you do not find a solution for your problem, log a service request.

> **See Also:**
>
> ■ *Oracle Fusion Middleware Release Notes*, available on the Oracle Technology Network:
> http://www.oracle.com/technetwork/developer-tools /forms/overview/index.html.

# B

# Configuring Java Plug-ins

This section describes the use of Oracle's Java Plug-in as a Web browser plug-in. Oracle Java Plug-in enables users to run Oracle Forms applications using Mozilla Firefox or Internet Explorer. It provides the ability to specify the use of a specific Java Virtual Machine (JVM) on the client. For more information, see the white paper "Using Sun's Java Plug-in" at
http://www.oracle.com/technetwork/developer-tools/forms/overview/index.html.

## B.1 Supported Configurations

Oracle supports the Java Plug-in. For more information, see the Java Plug-in Documentation at
http://www.oracle.com/technetwork/java/index-137617.html.

## B.2 Legacy Lifecycle Behavior And Configuration Requirements

In JDK 1.4.1 and later, the Java Plug-in supports the `LEGACY_LIFECYCLE` applet parameter. When this parameter is set to true, a running applet is not destroyed when the user navigates away from a page. Furthermore, when the user navigates back to the page, the running applet is resumed unless:

- The browser must re-issue the request for the applet definition, and

- The response to that request produces an applet definition that differs from the applet definition that was returned by the original request.

### B.2.1 Configuration Requirements

To use the `LEGACY_LIFECYCLE` feature for certain configurations, add `LEGACY_LIFECYCLE=true` parameter to the relevant configuration sections, such as in `formsweb.cfg`.

Alternatively, `legacy_lifecycle=true` can be specified on the URL that is used to launch a Forms application. This technique is useful primarily during application development.

In addition, JavaScript must be enabled in the browser from which the Forms application (that specifies `legacy_lifecycle=true`) is launched.

The HTML files must also adhere to certain guidelines. The base HTML files that are shipped with the product already adhere to the required guidelines. However, users who write their own base HTML files must ensure that such files adhere to the following guidelines:

1. The base HTML file must define the serverURL attribute to the value of the serverURL variable (serverURL="%serverURL%"), in the COMMENT node that has the ID forms_plugin_info.

2. The base HTML file must define the serverURL applet parameter, and its value must be the value of the appletServerURL variable. (Prior to Forms 11*g*, it was set to the value of the serverURL variable). This can be accomplished by including

   ```
   <PARAM NAME="serverURL" VALUE="%appletServerURL%">
   ```

   and

   ```
   serverURL="%appletServerURL%"
   ```

   in the OBJECT definition and the EMBED comment in user-written base HTML files. Note that the appletServerURL variable should not be set in a configuration file. (If it is, the value is ignored.) Instead, Forms computes its value automatically: if legacy_lifecycle=true (in the configuration file or in the initial URL), then the appletServerURL variable evaluates to "?", which causes Forms to look for the serverURL attribute of the COMMENT node (see above). Otherwise, the appletServerURL evaluates to the value of the serverURL variable.

3. The base HTML file must define the legacy_lifecycle applet parameter, and the value must not be hard-coded: it must match the value of the legacy_lifecycle variable. That is because in Forms 11*g*, the variable also affects the value of the appletServerURL variable (as explained above). This can be accomplished by including

   ```
   <PARAM NAME="legacy_lifecycle" VALUE="%legacy_lifecycle%">
   ```

   and

   ```
   legacy_lifecycle="%legacy_lifecycle%"
   ```

   in the OBJECT definition and the EMBED comment in user-written base HTML files.

# C

# Locations and Samples of Configuration Files

This section includes a list of configuration files and their default locations. This section also includes samples of the default configuration files that are installed on the system. Some default values such as locations and paths may vary.

- Section C.1, "Locations of Forms Configuration Files"

- Section C.2, "Default formsweb.cfg"

- Section C.3, "Platform Specific default.env Files"

- Section C.4, "base.htm and basejpi.htm Files"

- Section C.5, "web.xml"

- Section C.6, "weblogic.xml"

- Section C.7, "forms.conf"

- Section C.8, "Registry.dat"

- Section C.9, "Default jvmcontroller.cfg"

- Section C.10, "Default webutil.cfg"

- Section C.11, "Default webutilbase.htm"

- Section C.12, "Default webutiljpi.htm"

## C.1 Locations of Forms Configuration Files

Table C–1 lists the default locations of Forms configuration files on UNIX. The location of these files in Windows is similar.

*Table C–1    List of Files and their Locations in Release 11.1.2.0.0*

| File Name | Location in Release 11.1.2.0.0 |
|---|---|
| `formsweb.cfg` | `$DOMAIN_HOME/config/fmwconfig/servers/WLS_FORMS/applications/formsapp_11.1.2/config` |
| `default.env` | `$DOMAIN_HOME/config/fmwconfig/servers/WLS_FORMS/applications/formsapp_11.1.2/config` |

*Table C–1   (Cont.)  List of Files and their Locations in Release 11.1.2.0.0*

| File Name | Location in Release 11.1.2.0.0 |
|---|---|
| base.htm | $ORACLE_ INSTANCE/config/FormsComponent/forms/ server |
| basejpi.htm | $ORACLE_ INSTANCE/config/FormsComponent/forms/ server |
| webutilbase.htm | $ORACLE_ INSTANCE/config/FormsComponent/forms/ server |
| webutiljpi.htm | $ORACLE_ INSTANCE/config/FormsComponent/forms/ server |
| ftrace.cfg | $ORACLE_ INSTANCE/config/FormsComponent/forms/ server |
| web.xml | $DOMAIN_HOME/servers/WLS_FORMS/tmp/_ WL_user/formsapp_11.1.2/<random_ string>/war/WEB-INF |
| weblogic.xml | $DOMAIN_HOME/servers/WLS_FORMS/tmp/_ WL_user/formsapp_11.1.2/<random_ string>/war/WEB-INF |
| forms.conf | $ORACLE_INSTANCE/config/OHS/<OHS INSTANCE NAME>/moduleconf |
| jvmcontroller.cfg | $ORACLE_ INSTANCE/config/FRComponent/frcommon/ tools/jvm/ |
| webutil.cfg | $ORACLE_ INSTANCE/config/FormsComponent/forms/ server/ |
| Registry.dat | $DOMAIN_ HOME/config/fmwconfig/servers/WLS_ FORMS/applications/formsapp_ 11.1.2/config/forms/registry/oracle/f orms/registry |

# C.2  Default formsweb.cfg

A sample of the default `formsweb.cfg` file contains the following:

```
#formsweb.cfg defines parameter values used by the FormsServlet
# formsweb.cfg defines parameter values used by the FormsServlet (frmservlet)
# This section defines the Default settings.  Any of them may be overridden in the
# following Named Configuration sections.  If they are not overridden, then the
# values here will be used.
# The default settings comprise two types of parameters: System parameters,
# which cannot be overridden in the URL, and User Parameters, which can.
# Parameters which are not marked as System parameters are User parameters.
# SYSTEM PARAMETERS
# -----------------
# These have fixed names and give information required by the Forms
# Servlet in order to function.  They cannot be specified in the URL query
# string.  But they can be overridden in a named configuration (see below).
# Some parameters specify file names: if the full path is not given,
```

```
# they are assumed to be in the same directory as this file.  If a path
# is given, then it should be a physical path, not a URL.
# USER PARAMETERS
# ---------------
# These match variables (e.g. %form%) in the baseHTML file. Their values
# may be overridden by specifying them in the URL query string
# (e.g. "http://myhost.example.com/forms/frmservlet?form=myform&width=700")
# or by overriding them in a specific, named configuration (see below)
[default]
# System parameter: default base HTML file
baseHTML=base.htm
# System parameter: base HTML file for use with Sun's Java Plug-In
baseHTMLjpi=basejpi.htm
# System parameter: delimiter for parameters in the base HTML files
HTMLdelimiter=%
# System parameter: file setting environment variables for the Forms runtime
processes
envFile=default.env

# Forms runtime argument: whether to escape certain special characters
# in values extracted from the URL for other runtime arguments
escapeparams=true
# Forms runtime argument: which form module to run
form=test.fmx
# Forms runtime argument: database connection details
userid=
# Forms runtime argument: whether to run in debug mode
debug=no
# Forms runtime argument: host for debugging
host=
# Forms runtime argument: port for debugging
port=
# Forms runtime argument: BIDI digitSubstitution
digitSubstitution=context
# Other Forms runtime arguments: grouped together as one parameter.
# These settings support running and debugging a form from the Builder:
otherparams=obr=%obr% record=%record% tracegroup=%tracegroup% log=%log%
term=%term% ssoProxyConnect=%ssoProxyConnect%
# Sub argument for otherparams
obr=no
# Sub argument for otherparams
record=
# Sub argument for otherparams
tracegroup=
# Sub argument for otherparams
log=
# Sub argument for otherparams
term=

# HTML page title
pageTitle=Oracle Fusion Middleware Forms Services
# HTML attributes for the BODY tag
HTMLbodyAttrs=
# HTML to add before the form
HTMLbeforeForm=
# HTML to add after the form
HTMLafterForm=

# Forms applet parameter: URL path to Forms ListenerServlet
serverURL=/forms/lservlet
```

```
# Forms applet parameter
codebase=/forms/java
# Forms applet parameter
imageBase=codebase
# Forms applet parameter
width=750
# Forms applet parameter
height=600
# Forms applet parameter
separateFrame=false
# Forms applet parameter
splashScreen=
# Forms applet parameter
allowAlertClipboard=true
# Forms applet parameter
disableValidateClipboard=false
# Forms applet parameter
highContrast=false
# Forms applet parameter
background=
# Forms applet parameter
lookAndFeel=Oracle
# Forms applet parameter
colorScheme=teal
# Forms applet parameter
logo=
# Forms applet parameter
restrictedURLparams=pageTitle,HTMLbodyAttrs,HTMLbeforeForm,HTMLafterForm,log
# Forms applet parameter
formsMessageListener=
# Forms applet parameter
recordFileName=
# Forms applet parameter
serverApp=default
# Forms applet archive setting for other clients (Sun Java Plugin, Appletviewer,
 etc)
archive=frmall.jar
# Number of times client should retry if a network failure occurs.  You should
# only change this after reading the documentation.
networkRetries=0
# Page displayed to users to allow them to download Sun's Java Plugin.
# Sun's Java Plugin is typically used for non-Windows clients.
# (NOTE: you should check this page and possibly change the settings)
jpi_download_page=http://java.sun.com/products/archive/j2se/6u12/index.html
# Parameter related to the version of the Java Plugin
jpi_classid=clsid:CAFEEFAC-0016-0000-0012-ABCDEFFEDCBA
# Parameter related to the version of the Java Plugin
jpi_
codebase=http://java.sun.com/update/1.6.0/jinstall-6-windows-i586.cab#Version=1,6,
0,12
# Parameter related to the version of the Java Plugin
jpi_mimetype=application/x-java-applet;jpi-version=1.6.0_12
# Applet parameter for Sun's Java Plugin
legacy_lifecycle=false
# Single Sign-On OID configuration parameter: indicates whether we allow
# dynamic resource creation if the resource is not yet created in the OID.
ssoDynamicResourceCreate=true
# Single Sign-On parameter: URL to redirect to if ssoDynamicResourceCreate=false
ssoErrorUrl=
# Single Sign-On parameter: Cancel URL for the dynamic resource creation DAS page.
```

```
ssoCancelUrl=
# Single Sign-On parameter: indicates whether the url is protected in which
# case mod_osso will be given control for authentication or continue in
# the FormsServlet if not. It is false by default. Set it to true in an
# application-specific section to enable Single Sign-On for that application.
ssoMode=false
# Single Sign-On parameter: indicates whether session should operate in proxy
# user support or not. Specify ssoProxyConnect=yes to enable for particular
 application.
ssoProxyConnect=no
# The parameter allow_debug determines whether debugging is permitted.
# Administrators should set allow_debug to "true" if servlet
# debugging is required, or to provide access to the Forms Trace Xlate utility.
# Otherwise these activities will not be allowed (for security reasons).
allow_debug=false
# Parameter which determines whether new Forms sessions are allowed.
# This is also read by the Forms EM Overview page to show the
# current Forms status.
allowNewConnections=true
# EndUserMonitoring
# EndUserMonitoringEnabled parameter
# Indicates whether EUM/Chronos integration is enabled
EndUserMonitoringEnabled=false
# EndUserMonitoringURL
# indicates where to record EUM/Chronos data
EndUserMonitoringURL=
# Config for javascript integration
applet_name=
enableJavascriptEvent=true
# Config variable that will indicate if heartbeat will
# be blocked when a javascript call is a blocking call.
# The default value if false, i.e heart beat will not be
# blocked for any javascript calls.
JavaScriptBlocksHeartBeat=false
# Example Named Configuration Section
# Example 1: configuration to run forms in a separate browser window with
# "generic" look and feel (include "config=sepwin" in the URL)
# You may define your own specific, named configurations (sets of parameters)
# by adding special sections as illustrated in the following examples.
# Note that you need only specify the parameters you want to change.  The
# default values (defined above) will be used for all other parameters.
# Use of a specific configuration can be requested by including the text
# "config=<your_config_name>" in the query string of the URL used to run
# a form.  For example, to use the sepwin configuration, your could issue
# a URL like "http://myhost.example.com/forms/frmservlet?config=sepwin".
[sepwin]
separateFrame=True
lookandfeel=Generic
# Example Named Configuration Section
# Example 2: configuration running the Forms ListenerServlet in debug mode
# (debug messages will be written to the servlet engine's log file).
[debug]
serverURL=/forms/lservlet/debug
# Sample configuration for deployingWebUtil. Note that WebUtil is
# only installed with the Forms Builder and is also available for download
# from OTN.
[webutil]
WebUtilArchive=frmwebutil.jar,jacob.jar
WebUtilLogging=off
WebUtilLoggingDetail=normal
```

```
WebUtilErrorMode=Alert
WebUtilDispatchMonitorInterval=5
WebUtilTrustInternal=true
WebUtilMaxTransferSize=16384
baseHTML=webutilbase.htm
baseHTMLjpi=webutiljpi.htm
archive=frmall.jar
lookAndFeel=oracle
```

# C.3 Platform Specific default.env Files

There are two platform specific versions of default.env:

- Default default.env File for Windows

- Default default.env File for UNIX and Linux

## C.3.1 Default default.env File for Windows

```
# default.env - default Forms environment file, Windows version
#
# This file is used to set the Forms runtime environment parameters.
# If a parameter is not defined here, the value used will be that defined
# in the environment in which the WLS Managed Server was started.
#
# NOTES
# Configuration assistant will replace all the macro's with
#    the actual values.
#
  ORACLE_HOME=D:\Oracle2\Middleware\as_2
  ORACLE_INSTANCE=D:\Oracle2\Middleware\asinst_2

  #
  # TNS Entry to locate the database
  #
  TNS_ADMIN=D:\Oracle2\Middleware\asinst_2\config

  #
  # Search path for Forms applications (.fmx files, PL/SQL libraries)
  # If you need to include more than one directory, they should be semi-colon
  # separated (e.g. c:\test\dir1;c:\test\dir2)
  #
  FORMS_PATH=D:\Oracle2\Middleware\as_2\forms;D:\Oracle2\Middleware\asinst
_2\FormsComponent\forms

  # webutil config file path
  WEBUTIL_CONFIG=D:\Oracle2\Middleware\asinst
_2\config\FormsComponent\forms\server\webutil.cfg

  # Disable/remove this variable if end-users need access to the query-where
  # functionality which potentially allows them to enter arbitrary SQL
  # statements when in enter-query mode.
  FORMS_RESTRICT_ENTER_QUERY=TRUE

  #
  # The PATH setting is required in order to pick up the JVM (jvm.dll and
  # java.exe). Since PATH is being set, it needs to also include
  # D:\Oracle2\Middleware\as_2\bin so relevant files are correctly found.
  #
```

```
    PATH=D:\Oracle2\Middleware\as_2\bin;D:\Oracle2\Middleware\as
2\jdk\jre\bin\client;D:\Oracle2\Middleware\as_2\jdk\bin

  #
  # Settings for Forms tracing and logging
  # ---------------------------------------------
  # Note: By default tracing and logging directory is
  # %ORACLE_INSTANCE%\FormsComponent\forms\trace
  # To change the trace directory this entry has to be uncommented and set to
  # desired directory for tracing and logging

    #FORMS_TRACE_DIR=%ORACLE_INSTANCE%\FormsComponent\forms\trace

  #
  # Settings for Javascript events
  # ---------------------------------------------
  # Note: If this variable is set to false then the triggers and
  # built-ins associated with javascript events are disabled

    #FORMS_ALLOW_JAVASCRIPT_EVENTS=

  #
  # System settings
  # ---------------
  # You should not normally need to modify these settings
  #
  FORMS=D:\Oracle2\Middleware\as_2\forms

  #
  # Java class path
  #    This is required for the Forms debugger
  #    You can append your own Java code here)
  # frmsrv.jar and ldapjclnt11.jar are required for
  # the password expiry feature to work(#2213140).
  #
  CLASSPATH=D:\Oracle2\Middleware\as
_2\forms\j2ee\frmsrv.jar;D:\Oracle2\Middleware\as
_2\jlib\ldapjclnt11.jar;D:\Oracle2\Middleware\as
_2\jlib\debugger.jar;D:\Oracle2\Middleware\as
_2\jlib\ewt3.jar;D:\Oracle2\Middleware\as
_2\jlib\share.jar;D:\Oracle2\Middleware\as
_2\jlib\utj.jar;D:\Oracle2\Middleware\as
_2\jlib\zrclient.jar;D:\Oracle2\Middleware\as
_2\reports\jlib\rwrun.jar;D:\Oracle2\Middleware\as
_2\forms\java\frmwebutil.jar;D:\Oracle2\Middleware\as_2/jlib/start
_dejvm.jar;D:\Oracle2\Middleware\as_2\opmn\lib\optic.jar
```

## C.3.2  Default default.env File for UNIX and Linux

```
# default.env - default Forms environment file, Linux version
#
# This file is used to set the Forms runtime environment parameters.
# If a parameter is not defined here, the value used will be that defined
# in the environment in which the WLS Managed Server was started.
#
# NOTES
#    Configuration assitant will replace all the macro's with
#    the actual values.
#
#
```

```
#
  ORACLE_HOME=/as_1
  ORACLE_INSTANCE=/asinst_1

  #
  # TNS Entry to locate the database
  #
  TNS_ADMIN=/asinst_1/config

  #
  # Search path for Forms applications (.fmx files, PL/SQL libraries)
  #
  FORMS_PATH=/as_1/forms:/asinst_1/FormsComponent/forms

  #
  # WebUtil config file path. WebUtil is available for download from OTN.
  #
  WEBUTIL_CONFIG=/asinst_1/config/FormsComponent/forms/server/webutil.cfg

  # Disable/remove this variable if end-users need access to the query-where
  # functionality which potentially allows them to enter arbitrary SQL
  # statements when in enter-query mode.
  FORMS_RESTRICT_ENTER_QUERY=TRUE

  # Java class path
  #   This is required for the Forms debugger
  #   You can append your own Java code here)
  # frmsrv.jar and ldapjclnt11.jar are required for
  # the password expiry feature to work(#2213140).
  #
  CLASSPATH=/as
_1/forms/j2ee/frmsrv.jar:/as
_1/jlib/ldapjclnt11.jar:/as
_1/jlib/debugger.jar:/as
_1/jlib/ewt3.jar:/as_1/jlib/share.jar:/as
_1/jlib/utj.jar:/as
_1/jlib/zrclient.jar:/as
_1/reports/jlib/rwrun.jar:/as
_1/forms/java/frmwebutil.jar:/as_1/jlib/start
_dejvm.jar:/as_1/opmn/lib/optic.jar
#

  # The PATH setting is not required for frmweb if the Forms executables are
  # in <ORACLE_HOME>/bin. JDK/bin is also required for dejvm to be
  # auto-started by frmweb.
  #
  PATH=/scratch/cls0223/bea/as_1/bin:/scratch/cls0223/bea/as_1/jdk/bin

  #
  # Settings for Reports
  # -------------------------------
  # NOTE: This setting is only needed if Reports applications
 # are called from Forms applications
  # However, because of bug 2336698 where a report is started from
  # a forms debugger session with an already running JVM, then
  # the report's class path should also be included in the forms
  # class path.
  # We no longer need to set REPORTS_CLASSPATH as forms will
  # always start the JVM before calling reports.
```

```
    #
    # Settings for Forms tracing and logging
    # ----------------------------------------------
    # Note: By default tracing and logging directory is
    # $ORACLE_INSTANCE/FormsComponent/forms/trace
    # To change the trace directory this entry has to be uncommented and set to
    # desired directory for tracing and logging

      #FORMS_TRACE_DIR=/scratch/cls0223/asinst_1/FormsComponent/forms/trace


    #
    # Settings for Javascript events
    # ----------------------------------------------
    # Note: If this variable is set to false then the triggers and
    # built-ins associated with javascript events are disabled

      #FORMS_ALLOW_JAVASCRIPT_EVENTS=



    #
    # System settings
    # ---------------
    # You should not normally need to modify these settings
    #
      #
      # Path for shared library objects
      #   This is highly platform (if not machine) specific !  At install time
      #   <percent>LD_LIBRARY_PATH<percent> should be replaced with the
      #   actual value of the LD_LIBRARY_PATH environment variable (at install
      #   time).  That should ensure we have the paths for such necessities as
      #   the motif and X11 libraries.
      # Explanations:
      #   - Reports needs the path for libjava.so
      #     (.../jre/lib/sparc)
      #   - Forms needs two paths to the jre, for libjvm.so and libhpi.so
      #   - In JDK 1.4.1 the location of libjvm.so is lib/sparc (there is no
      #     classic directory) so we do not include the .../classic directory
      #     below. There are other versions of libjvm.so (in directories server,
      #     client and hotspot) but we will use the version in lib/sparc for now.
      #
      LD_LIBRARY_PATH=/bea/as_1/lib:/bea/as
_1/jdk/jre/lib/i386:/bea/as
_1/jdk/jre/lib/i386/server:/bea/as_1/jdk/jre/lib/i386/native
_threads
      #
      # Setting to take care of signal-chaining facility offered by JVM 1.5
      # Without this Forms/Reports integration could have issues on Unix/Linux
      #
      LD_PRELOAD=/as_1/jdk/jre/lib/i386/libjsig.so
```

## C.4 base.htm and basejpi.htm Files

Two baseHTML files are created for your system by the Oracle Universal Installer
during Forms installation and configuration. **In most cases, you will not need to
modify these files.** If you do need to modify these files, you should create your own
versions and reference them from the `formsweb.cfg` file. The default files may be
overridden by a patch installation.

When a user first starts an Oracle Forms application (by clicking a link to the application's URL), a baseHTML file is read by Forms servlet.

Any variables (*%variablename%*) in the baseHTML file are replaced with the appropriate parameter values specified in the `formsweb.cfg` file described in Section 4.2, "Configuring Forms Services", and from query parameters in the URL request (if any). Query parameter values override the values in the `formsweb.cfg` file.

Then, the baseHTML file is downloaded to the user's Web browser.

The following baseHTML starter files are available in the `$ORACLE_INSTANCE/config/FormsComponent/forms/server/` directory:

- **basejpi.htm**: This is the baseHTML file for Java Plug-in. The Forms servlet uses this default file if the client browser is on Windows.

- **base.htm**: This is a baseHTML file containing the APPLET tags required to run the Forms applet in the AppletViewer, or in any Web browser certified by Oracle with a native JVM that is certified with Oracle Forms. See Default base.htm File for an example.

**To create a new baseHTML file:**

1. Copy the basejpi.htm, or base.htm starter file, which is located in the `$ORACLE_INSTANCE/config/FormsComponent/forms/server/` directory.

2. Rename the file (for example, `order.htm`).

3. Add or modify any text that is visible to the user (for example, text contained within <TITLE> and <BODY> tags).

4. Modify the parameters as needed. It is recommended that you use variables in the baseHTML file, and specify the actual values in the `formsweb.cfg` file, as described in `formsweb.cfg`.

   The baseHTML tags can also be set in the specific named configuration section, overwriting the system default value. This is recommended if an individual custom baseHTML template needs to be used. However, if a custom template is used for all applications, then it is recommended you change the default configuration section in the `formsweb.cfg` file.

5. Place the new baseHTML file in the `$ORACLE_INSTANCE/config/FormsComponent/forms/server/` directory, update the baseHTML, baseHTMLjpi parameter in the `formsweb.cfg` file to point to the new baseHTML files.

## C.4.1 Parameters and variables in the baseHTML file

If you do not want to use a parameter tag that is provided in the base.htm or basejpi.htm file, delete it from the file.

Oracle recommends that you specify the rest of the parameter values as variables (*%variablename%*) in the baseHTML file. For example:

```
<PARAM NAME="logo"  VALUE="%logo%">
```

Then, specify the actual parameter values in the `formsweb.cfg` file. All variables are replaced with the appropriate parameter values at runtime.

### C.4.1.1 Usage Notes

- You can use a variable value anywhere in the baseHTML file. Variables are specified as a name enclosed in a special delimiter (the default delimiter is %). For example, you could have the following line in your HTML file:

  ```
  ARCHIVE="%Archive%"
  ```

  You must then assign a value to `%Archive%` either in the formsweb.cfg file or in the URL query string.

- All variables must receive values at runtime. If a variable does not receive a value, Forms Services cannot build a proper HTML file to pass back to the user's Web browser, resulting in an error.

- To streamline performance, use only one Web server as a source for Jar file downloads. This will prevent multiple downloads of the same files from different servers.

## C.4.2 Default base.htm File

```
<HTML>
<!-- FILE: base.htm (Oracle Forms) -->
<!--                              -->
<!-- This is the default base HTML file for running a form on
<!--the web using a generic APPLET tag to include   -->
<!-- Forms applet.-->
<!--                              -->
<!-- IMPORTANT NOTES:           -->
<!-- Default values for all the variables which appear
<!-- below (enclosed in percent characters) are defined-->
<!-- in the servlet configuration file (formsweb.cfg).   -->
<!--  It is preferable to make changes in that file where   -->
<!--  possible, rather than this one.      -->

<!-- This file will be REPLACED if you reinstall
<!--Oracle Forms, so you are advised to make your own  -->
<!--  version if you want to make want to make any  -->
<!--  modifications.  You should then set the       -->
<!-- baseHTML parameter in the Forms Servlet
<!--configuration file (formsweb.cfg) to point to     -->
<!--  your new file instead of this one.  -->
<HEAD><TITLE>%pageTitle%</TITLE></HEAD>
<BODY %HTMLbodyAttrs%>
%HTMLbeforeForm%
<COMMENT id="forms_plugin_info"
        serverURL="%serverURL%"
        appcodebase="%codebase%"
        apparchive="%archive%"
        appheight="%Height%"
        appwidth="%Width%"
        appname="%applet_name%">
</COMMENT>
<!-- Forms applet definition (start) -->
<NOSCRIPT>
<APPLET CODEBASE="%codebase%"
        CODE="oracle.forms.engine.Main"
        ARCHIVE="%archive%"
        WIDTH="%Width%"
        HEIGHT="%Height%"
```

```
              NAME="%applet_name%" MAYSCRIPT>
</NOSCRIPT>
<SCRIPT LANGUAGE="JavaScript" SRC="/forms/frmjscript/forms_base_ie.js">
</SCRIPT>
<PARAM NAME="serverURL" VALUE="%appletServerURL%">
<PARAM NAME="networkRetries" VALUE="%networkRetries%">
<PARAM NAME="serverArgs"
VALUE="%escapeParams% module=%form% userid=%userid% debug=%debug% host=%host%
port=%port% %otherParams%">
<PARAM NAME="separateFrame" VALUE="%separateFrame%">
<PARAM NAME="splashScreen"  VALUE="%splashScreen%">
<PARAM NAME="background"  VALUE="%background%">
<PARAM NAME="lookAndFeel"  VALUE="%lookAndFeel%">
<PARAM NAME="colorScheme"  VALUE="%colorScheme%">
<PARAM NAME="serverApp" VALUE="%serverApp%">
<PARAM NAME="logo" VALUE="%logo%">
<PARAM NAME="imageBase" VALUE="%imageBase%">
<PARAM NAME="formsMessageListener" VALUE="%formsMessageListener%">
<PARAM NAME="recordFileName" VALUE="%recordFileName%">
<PARAM NAME="EndUserMonitoringEnabled" VALUE="%EndUserMonitoringEnabled%">
<PARAM NAME="EndUserMonitoringURL" VALUE="%EndUserMonitoringURL%">
<PARAM NAME="heartbeat" VALUE="%heartbeat%">
<PARAM NAME="MaxEventWait" VALUE="%MaxEventWait%">
<PARAM NAME="allowAlertClipboard" VALUE="%allowAlertClipboard%">
<PARAM NAME="disableValidateClipboard" VALUE="%disableValidateClipboard%">
<PARAM NAME="enableJavascriptEvent" VALUE="%enableJavascriptEvent%">
<PARAM NAME="digitSubstitution" VALUE="%digitSubstitution%">
<PARAM NAME="legacy_lifecycle" VALUE="%legacy_lifecycle%">
<PARAM NAME="JavaScriptBlocksHeartBeat" VALUE="%JavaScriptBlocksHeartBeat%">
<PARAM NAME="highContrast" VALUE="%highContrast%">
</Applet>
<!--Forms applet deinition (end) -->
&HTMLafterForm%
</BODY>
</HTML>
```

## C.4.3  Default basejpi.htm File

```
<HTML>
<!-- FILE: basejpi.htm (Oracle Forms)   -->
<!--                                     -->
<!-- This is the default base HTML file for running
<!--a form on the  web using the JDK Java Plugin.   -->
<!--  This is used for example when       -->
<!-- running with Netscape on Unix.      -->
<!-- IMPORTANT NOTES:                    -->
<!-- Default values for all the variables which -->
<!--appear below (enclosed in percent characters)-->
<!-- are defined in the servlet configuration file  -->
<!-- (formsweb.cfg). It is preferable to make      -->
<!-- changes in that file where possible, rather than
<!--this one.        -->
<!--                                     -->
<!-- This file will be REPLACED if you reinstall
<!--Oracle Forms, so      -->
<!-- you are advised to create your own version if
<!--you want to make    -->
<!-- any modifications.  You should then set the
<!--baseHTMLjpi         -->
<!-- parameter in the Forms Servlet configuration file
```

```
<!--(formsweb.cfg) -->
<!-- to point to your new file instead of this one. -->
<HEAD>
<TITLE>%pageTitle%</TITLE>
</HEAD>
<BODY %HTMLbodyAttrs%>
%HTMLbeforeForm%
<COMMENT id="forms_plugin_info"
         serverURL="%serverURL%"
         plug_ver="%jpi_classid%"
         appheight="%Height%"
         appwidth="%Width%"
         appcodebase="%jpi_codebase%"
         appname="%applet_name%">
</COMMENT>
<!-- Forms applet definition (start) -->
<NOSCRIPT>
<OBJECT classid="%jpi_classid%"
         codebase="%jpi_codebase%"
         WIDTH="%Width%"
         HEIGHT="%Height%"
         HSPACE="0"
         VSPACE="0"
         ID="%applet_name%">
</NOSCRIPT>
<SCRIPT LANGUAGE="JavaScript" SRC="/forms/frmjscript/forms_ie.js">
</SCRIPT>
<PARAM NAME="TYPE"        VALUE="%jpi_mimetype%">
<PARAM NAME="CODEBASE"    VALUE="%codebase%">
<PARAM NAME="CODE"        VALUE="oracle.forms.engine.Main" >
<PARAM NAME="ARCHIVE"     VALUE="%archive%" >
<PARAM NAME="serverURL" VALUE="%appletServerURL%">
<PARAM NAME="networkRetries" VALUE="%networkRetries%">
<PARAM NAME="serverArgs"
      VALUE="%escapeParams% module=%form% userid=%userid% debug=%debug%
host=%host% port=%port% %otherParams%">
<PARAM NAME="separateFrame" VALUE="%separateFrame%">
<PARAM NAME="splashScreen"  VALUE="%splashScreen%">
<PARAM NAME="background"  VALUE="%background%">
<PARAM NAME="lookAndFeel"  VALUE="%lookAndFeel%">
<PARAM NAME="colorScheme"  VALUE="%colorScheme%">
<PARAM NAME="serverApp" VALUE="%serverApp%">
<PARAM NAME="logo" VALUE="%logo%">
<PARAM NAME="imageBase" VALUE="%imageBase%">
<PARAM NAME="formsMessageListener" VALUE="%formsMessageListener%">
<PARAM NAME="recordFileName" VALUE="%recordFileName%">
<PARAM NAME="EndUserMonitoringEnabled" VALUE="%EndUserMonitoringEnabled%">
<PARAM NAME="EndUserMonitoringURL" VALUE="%EndUserMonitoringURL%">
<PARAM NAME="heartBeat" VALUE="%heartBeat%">
<PARAM NAME="MaxEventWait" VALUE="%MaxEventWait%">
<PARAM NAME="allowAlertClipboard" VALUE="%allowAlertClipboard%">
<PARAM NAME="disableValidateClipboard" VALUE="%disableValidateClipboard%">
<PARAM NAME="enableJavascriptEvent" VALUE="%enableJavascriptEvent%">
<PARAM NAME="MAYSCRIPT" VALUE="%enableJavascriptEvent%">
<PARAM NAME="digitSubstitution" VALUE="%digitSubstitution%">
<PARAM NAME="legacy_lifecycle" VALUE="%legacy_lifecycle%">
<PARAM NAME="JavaScriptBlocksHeartBeat" VALUE="%JavaScriptBlocksHeartBeat%">
<PARAM NAME="highContrast" VALUE="%highContrast%">
<COMMENT>
<EMBED SRC="" PLUGINSPAGE="%jpi_download_page%"
```

```
                    TYPE="%jpi_mimetype%"
                    java_codebase="%codebase%"
                    java_code="oracle.forms.engine.Main"
                    java_archive="%archive%"
                    WIDTH="%Width%"
                    HEIGHT="%Height%"
                    HSPACE="0
                    VSPACE="0"
                    NAME="%applet_name%"
serverURL="%appletServerURL%"
                    networkRetries="%networkRetries%"
                    serverArgs="%escapeParams% module=%form% userid=%userid% debug=%debug%
host=%host% port=%port% %otherparams%"
                    separateFrame="%separateFrame%"
                    splashScreen="%splashScreen%"
                    background="%background%"
                    lookAndFeel="%lookAndFeel%"
                    colorScheme="%colorScheme%"
                    serverApp="%serverApp%"
                    logo="%logo%"
                    imageBase="%imageBase%"
                    recordFileName="%recordFileName%"
                    EndUserMonitoringEnabled="%EndUserMonitoringEnabled%"
                    EndUserMonitoringURL="%EndUserMonitoringURL%"
                    heartBeat="%heartBeat%"
                    MaxEventWait="%MaxEventWait%"
                    disableValidateClipboard="%disableValidateClipboard%"
allowAlertClipboard="%allowAlertClipboard%"
                    enableJavascriptEvent="%enableJavascriptEvent%"
                    MAYSCRIPT="%enableJavascriptEvent%"
                    digitSubstitution="%digitSubstitution%"
                    legacy_lifecycle="%legacy_lifecycle%"
                    JavaScriptBlocksHeartBeat="%JavaScriptBlocksHeartBeat%"
                    highContrast="%highContrast%"
<NOEMBED>
</COMMENT>
</NOEMBED>
</EMBED>
</OBJECT>
<!-- Forms applet definition (end) -->
%HTMLafterForm%
</BODY>
</HTML>
```

## C.5  web.xml

The web.xml file is the web application deployment descriptor file for forms Java EE application. This file is located at $DOMAIN_HOME/servers/WLS_FORMS/tmp/_WL_user/formsapp_11.1.2/<random_string>/war/WEB-INF/. Advanced users might want to edit the web.xml file to:

■   Enable extra testing options.

  If you are having difficulty running Oracle Forms in your  Oracle Fusion Middleware installation, it can be useful to enable certain test options which are not usually enabled for security reasons. To use these options, edit the web.xml file to set the testMode frmservlet parameter to true. Then restart the Web server (or Oracle WebLogic Managed Server). The additional options are then visible on the

Forms servlet administration page (which can be accessed at a URL like http://<your_web_server_hostname>:<port>/forms/frmservlet/admin).

- Run Oracle Forms using static HTML pages (rather than the Forms servlet).

When Oracle Forms applications are run using a method other than the Forms servlet (for example, static HTML pages, or JSPs), parameter settings in the formsweb.cfg file are not used. You may therefore need to define servlet parameters for the Listener Servlet, such as workingDirectory and envFile (specifying the current working directory for the Forms runtime processes, and the file containing environment settings to be used).

Servlet mappings are defined in web.xml. Table C–2 describes some of the servlet mappings.

*Table C–2   web.xml Servlet Mappings*

| URL Path | Type | Maps to | Purpose |
|----------|------|---------|---------|
| /forms/frmservlet | Servlet mount point | Forms servlet | Generate HTML page to run a form |
| /forms/lservlet | Servlet mount point | Forms Listener servlet | Handles message traffic from the Forms applet |

## C.5.1  Default web.xml File

```
<?xml version='1.0' encoding='UTF-8'?>
<web-app xmlns="http://java.sun.com/xml/ns/javaee">
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <servlet>
    <servlet-name>frmservlet</servlet-name>
    <servlet-class>oracle.forms.servlet.FormsServlet</servlet-class>
    <init-param>
       <!-- Turn on or off sensitive options on the frmservlet/admin page.
            For security reasons this should be set to false for
            production sites.
       -->
       <param-name>testMode</param-name>
       <param-value>false</param-value>
    </init-param>
    <load-on-startup>1</load-on-startup>
  </servlet>
  <servlet>
    <servlet-name>lservlet</servlet-name>
    <servlet-class>oracle.forms.servlet.ListenerServlet</servlet-class>
  </servlet>
  <servlet-mapping>
    <servlet-name>frmservlet</servlet-name>
    <url-pattern>/frmservlet/*</url-pattern>
  </servlet-mapping>
  <servlet-mapping>
    <servlet-name>lservlet</servlet-name>
    <url-pattern>/lservlet/*</url-pattern>
  </servlet-mapping>
  <!-- add mime mapping for the java scripts -->
  <mime-mapping>
     <extension>js</extension>
      <mime-type>application/x-javascript</mime-type>
  </mime-mapping>
```

```
    <welcome-file-list>
      <welcome-file>lservlet</welcome-file>
      <welcome-file>frmservlet</welcome-file>
    </welcome-file-list>
    <listener>

<listener-class>oracle.forms.config.mbeans.FormsappLifeCycleCallBack</listener-cla
ss>
    </listener>
    <!--  Define security constraints to limit access to the defined url to a
    particular role. Logical roles are defined in web.xml and these roles are
mapped
    to actual roles(principal roles) in weblogic.xml
    -->
    <security-constraint>
        <web-resource-collection>
            <web-resource-name>TraceLog</web-resource-name>
            <url-pattern>/frmservlet/trace/*</url-pattern>
            <http-method>GET</http-method>
        </web-resource-collection>
        <auth-constraint>
            <description>Admin users only</description>
            <role-name>formsadmin</role-name>
        </auth-constraint>
    </security-constraint>

    <login-config>
      <auth-method>BASIC</auth-method>
      <realm-name>WebApp</realm-name>
    </login-config>
    <security-role>
          <description>admin role</description>
          <role-name>formsadmin</role-name>
     </security-role>

</web-app>
```

# C.6  weblogic.xml

The `weblogic.xml` is the web application deployment descriptor file. This file is
located at `$DOMAIN_HOME/servers/WLS_FORMS/tmp/_WL_user/formsapp_`
`11.1.2/<random_string>/war/WEB-INF`.

```
<?xml version='1.0' encoding='UTF-8'?>
<weblogic-web-app xmlns="http://www.bea.com/ns/weblogic/90"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<session-descriptor>
   <timeout-secs>7200</timeout-secs>
   <invalidation-interval-secs>120</invalidation-interval-secs>
   <debug-enabled>false</debug-enabled>
   <id-length>52</id-length>
   <tracking-enabled>true</tracking-enabled>
   <cache-size>1024</cache-size>
   <max-in-memory-sessions>-1</max-in-memory-sessions>
   <cookies-enabled>false</cookies-enabled>
</session-descriptor>

<!--logical roles defined in web.xml are mapped to the real users below -->
<security-role-assignment>
    <role-name>formsadmin</role-name>
```

```
                <principal-name>Administrators</principal-name>
        </security-role-assignment>
        </weblogic-web-app>
```

## C.7  forms.conf

Prior to 11*g*, virtual path mappings were defined in forms.conf. In 11*g*, `forms.conf` defines WebLogic handler mappings for the Managed Server where the Forms Services applications are deployed.  For more information, see the Section 3.2.3, "Oracle HTTP Listener Configuration File." The location of the file is `$ORACLE_INSTANCE/config/OHS/<OHS INSTANCE NAME>/moduleconf`.

---

> **Note:**   When including any user-defined `aliasMatch` with the prefix `/forms/` in `forms.conf`, add the directive `WLExcludePathOrMimeType`. For example, in Linux, when defining the `aliasMatch` for `/forms/usericons` in `forms.conf`, the directive `WLExcludePathOrMimeType` is defined as following:
>
> ```
> AliasMatch /forms/usericons/(..*) "/home/userx/myicons/$1"
> WLExcludePathOrMimeType /forms/usericons/
> ```

---

### C.7.1  Default forms.conf

```
# Name
#    forms.conf  - Forms component Apache directives configuration file.

# Purpose
#    It should include the weblogic managed server (routing) directives for
#    the servers where Forms applications are deployed and other miscellaneous
#    Forms component OHS directives.
#
#
# Remarks
#    This file is included with the OHS configuration under
#    $OI/config/OHS/<OHS Node Name>/moduleconf sub-directory.
#
#
<IfModule !mod_osso.c>
        LoadModule osso_module  ${ORACLE_HOME}/ohs/modules/mod_osso.so
</IfModule>
<IfModule mod_osso.c>
        OssoHTTPOnly off
</IfModule>
<Location /forms>
        SetHandler weblogic-handler
        WebLogicCluster    dadvma0190.example.com:9001
        DynamicServerList OFF
</Location>
#
# virtual mapping for the /forms/html mapping.
#
RewriteEngine on
RewriteRule ^/forms/html/(..*) /workaroundhtml/$1 [PT]
AliasMatch ^/workaroundhtml/(..*)
"/scratch/fmw/ps1/rc3/asinst_2/config/FormsComponent/forms/html/$1"
```

# C.8  Registry.dat

Location: `$DOMAIN_HOME/config/fmwconfig/servers/WLS_ FORMS/applications/formsapp_ 11.1.2/config/forms/registry/oracle/forms/registry`

This file enables you to change the default font, font mappings, and icons that Forms Services uses.

## C.8.1  Registry.dat

```
# This is the Registry file.
#
# This file contains the logical [Java] Class name and an associated
# [numerical] identifier that will be used to refer to objects of the
# class in order to reduce the amount of information that needs to be
# repeatedly transmitted to the client.
#
# This file is of the Form understood by java.util.Properties (for now)
#
# The System Level sound file is relative to the CODEBASE
#
# The oracle classes which used to be defined here have now been moved to
# within the code.
#
# #
# Defaults for the Font details, all names are Java Font names.  Each of
# these parameters represents the default property to use when none is
# specified.
# defaultFontname represents the default Java fontName.
# defaultSize     represents the default fontSize.  Note that the size is
#                 multiplied by 100 (e.g. a 10pt font has a size of 1000).
# defaultStyle    represents the default fontStyle, PLAIN or ITALIC.
# defaultWeight   represents the default fontWeight, PLAIN or BOLD.
#
default.fontMap.defaultFontname=Dialog
default.fontMap.defaultSize=900
default.fontMap.defaultStyle=PLAIN
default.fontMap.defaultWeight=PLAIN
#
#
# Default Font Face mapping.
#
# appFontname  represents a comma delimited list of Application Font Names.
# javaFontname represents a comma delimited list of Java Font Names.
#
# The number of entries in the appFontname list should match the number in
# the javaFontname list.  The elements of the list are comma separated and
# *all* characters are taken literally, leading and trailing spaces are
# stripped from Face names.
#
# Note that this file uses the Java 1.1 Font names in order to be able to
# handle the NLS Plane (BUG #431051)
#
default.fontMap.appFontnames=Courier
New,Courier,courier,System,Terminal,Fixed,Fixedsys,Times,Times New Roman,MS Sans
Serif,Arial
default.fontMap.javaFontnames=MonoSpaced,MonoSpaced,MonoSpaced,Dialog,MonoSpaced,D
ialog,Dialog,Serif,Serif,Dialog,SansSerif
# The Application Level icon files are relative to the DOCUMENTBASE
```

```
#    example: icons/
# or an absolute URL.
#    example: http://www.example.net/~luser/d2k_project/
#
default.icons.iconpath=
default.icons.iconextension=gif


#
# Application level settings to control UI features
#
app.ui.lovButtons=false
app.ui.requiredFieldVA=false
#    The background color is specified as an RGB triple.
app.ui.requiredFieldVABGColor=255,0,0
```

## C.9 Default jvmcontroller.cfg

A Forms application can be configured to use a specific JVM controller using the jvmcontroller parameter. This parameter is specified in formsweb.cfg. The parameters that are used by the JVM controller are specified in the JVM controller's configuration file, jvmcontrollers.cfg. This file is located at $ORACLE_INSTANCE/config/FRComponent/frcommon/tools/jvm/.

```
# jvmcontrollers.cfg defines parameter values used by the JVM Controller(dejvm)

# Default JVM Controller
# This section defines the default settings. Any of them may be overridden
# in the following Named JVM Controller sections.  If they are not overridden,
# then the values here will be used.
[default]


# Example: Named JVM Controller
# This section shows example values for a jvm controller. These
# values overrides any values defined for the default controller.
[example]
jvmoptions=-Xms512m -Xmx1024m

# Classpath settings given here is an example only. This should be
# modified to include the required jar files and should be set in
# platform specific manner.
classpath=/myapps/common/jars/common.jar:/myapps/anapp/jars/anapp.jar
maxsessions=50
logdir=/myapps/anapp/log
logging=off
```

## C.10 Default webutil.cfg

The webutil.cfg file is one of the files used to configure WebUtil at run time. For more information on the file, see Section 3.2.6, "WebUtil Configuration Files." For information about using WebUtil at design time, see the Oracle Forms Developer Help. This file is located at $ORACLE_INSTANCE/config/FormsComponent/forms/server/.

```
# ------------------------------------------------------------------------
# webutil.cfg - WebUtil default configuration file
```

```
# ----------------------------------------------------------------------------
# This file provides all of the configuration settings for webutil. These are
# divided into the following sections:
# 1. Logging Options
# 2. Installation Options
# 3. File Upload and Download Options
# 1. Server Side Logging Options for logging errors and log messages
# You must set logging.enabled to true to allow mid-tier logging. Without this
# mid-tier logging will not take place no matter what PL/SQL or URL options
# are supplied to switch it on. Once logging is enabled the other settings come
# into play.
#
# Details
# -------
#  logging.file       : Defines the file name and location of the log file.
#                       Note that WebUtil does no log file management. You may
#                       need to manually clean this file up from time to time.
#  logging.enabled    : Can be TRUE or FALSE
#  logging.errorsonly : Can be TRUE or FALSE. Setting to true will ensure that
 #                        only errors and not normal informational log messages
#                       are written to the log file. For product use this would
#                       normally be set to TRUE
#  logging.connections: Can be TRUE or FALSE. Setting to true will cause each
#                       connection from a client using WebUtil to write into
#                       the log as it sets up.
logging.file=
logging.enabled=FALSE
logging.errorsonly=FALSE
logging.connections=FALSE
# 2. Installation Options
# WebUtil needs to download some files to the client in order to perform
# certain integration operations such as OLE or Registry Access. These files
# are downloaded to the client when webutil is being run for the first time.
# Subsequent runs will use the cached dll. You have to define the location of
# these files on the server and the location on the client.
#
# Here is a table that shows various numeric constants and their meanings. They
# are used in the install options configurations. These constants are defined
# in WEBUTIL_CORE package spec. Note that the Client OS Processor Architecture
# is typically the architecture of the JVM that will be running on the client,
# not necessarily same as the actual client OS architecture.
#
# <os> -   Client OS Family          <arch> -   Client OS Processor Architecture
# 0   -   Windows                    0   -   32 bit architecture
# 1   -   Linux                      1   -   64 bit architecture
# 2   -   SOLARIS
# 3   -   HP
# 4   -   AIX
# 5   -   Mac
# 9   -   Unknown
#
# <package> -    Package Name
#     7    -    WEBUTIL_OLE2
#     9    -    WEBUTIL_C_API
# We don't yet have downloads for other packages. If there is a need, refer
# WEBUTIL_CORE package spec for the constants of other packages
#
# Details
# -------
#  NOTE: For all settings <arch> field is optional. Without this field, the
```

```
#     setting will be applicable for both 32 bit and 64 bit client OS (JVM)
#     architectures.
#
#  install.syslib.location.<os>.<arch>  :
#    The virtual path to the directory holding the webutil library files on the
#    server side. This must either be an absolute URL or a URL that is relative
#    to the documentbase.
#    NOTE: <os> field is optional for this setting. In the absence of a setting
#    that has the matching client OS and ARCH, it will use
#    install.syslib.location. <arch> cannot be specified if <os> is omitted.
#
#  install.syslib.location.client.<os>.<arch>  :
#    The path to the directory on the client machine where webutil library
#    files will be downloaded. This must either be an absolute path or a path
#    that is relative to client user profile or HOME. Directory will be created
#    if necessary along with other required parent directories. If the path is
#    not set, it will be treated as a special case where libraries will be
#    downloaded to client JRE\bin (windows) or JRE/lib (unix). If this
#    directory is changed, all the libraries will be redownloaded again.
#
#    Please keep in mind that other Forms server could also have a similar
#    location setting and thus libraries could be overwritten when the same
#    client runs webutil from different forms server. It is therefore
#    advisable that the location has a subdirectory that can be unique to your
#    server, such as server host name. You could use $(SERVERHOST) in the
#    location. This string will be replaced with the actual server host name
#
#  install.syslib.<os>.<arch>.<package>.<n> :
#    The name(s) of the libraries required for particular webutil beans. The
#    format of this is name|size|version|showDownloadDialog. Multiple libraries
#    can be downloaded per package. But ensure that the <n> values are
#    consecutive and start at 1 for a given os, arch and package set.

install.syslib.location=/webutil
install.syslib.location.0.0=/webutil/win32
install.syslib.location.0.1=/webutil/win64
# Add/change the following if you want to specify a different client location
# where the syslib libraries can be downloaded.
# Format:
# install.syslib.location.client.<os>.<arch>=<location>
install.syslib.location.client.0.0=webutil\syslib\$(SERVERHOST)\win32
install.syslib.location.client.0.1=webutil\syslib\$(SERVERHOST)\win64
# Change size and version if necessary, like when upgrading the library.
# Normally this would not be required since most of these libraries come with
# install itself. Jacob however does not come with install
# Format:
# install.syslib.<os>.<arch>.<package>.<n>=name|size|version|showDownloadDialog
install.syslib.0.0.7.1=jacob-1.14.3-x86.dll|102400|1.14.3|true
install.syslib.0.1.7.1=jacob-1.14.3-x64.dll|117760|1.14.3|true
install.syslib.0.0.9.1=JNIsharedstubs.dll|45056|1.0|true
install.syslib.0.1.9.1=JNIsharedstubs.dll|58368|1.0|true
install.syslib.0.0.9.2=d2kwut60.dll|81920|1.0|true
install.syslib.0.1.9.2=d2kwut60.dll|102400|1.0|true
# You can also add your own libraries in here, e.g.
#install.syslib.0.0.user.1=testwebutil.dll|204872|1.0|true
#install.syslib.0.0.user.1=ffisamp.dll|40960|1.0|true
#install.syslib.0.1.user.1=ffisamp.dll|35328|1.0|true
# 3. Upload / Download Options
# For the file upload and download options you can define the default locations
# on the server that webutil can use as a work area. Optionally you can switch
```

```
                    # upload and download off
                    # Details
                    # -------
                    #  transfer.database.enabled  : Can be TRUE or FALSE - allows you to enable or
                    #                                 disable upload and download from the database
                    #                                 server.
                    #  transfer.appsrv.enabled     : Can be TRUE or FALSE - allows you to enable or
                    #                                 disable upload and download from the
                    #                                 application server.
                    #  transfer.appsrv.workAreaRoot: The root of the location in which WebUtil can
                    #                                 store temporary files uploaded from the client.
                    #                                 If no location is specified, application server
                    #                                 user_home/temp will be assumed.
                    #                                 This location is always readable and writable
                    #                                 no matter what the settings in
                    #                                 transfer.appsrv.* are. This setting is
                    #                                 required if you need the Client side
                    #                                 READ/WRITE_IMAGE_FILE procedures.
                    #  transfer.appsrv.accessControl:Can be TRUE or FALSE - allows you to indicate
                    #                                 that uploads and downloads can only occur from
                    #                                 the directories named in the
                    #                                 transfer.appsrv.read.n and
                    #                                 transfer.appsrv.write.n entries and their
                    #                                 subdirectories. If this setting is FALSE,
                    #                                 transfers can happen anywhere.
                    #  transfer.appsrv.read.<n>:   List of directory names that downloads can read
                    #                                 from.
                    #  transfer.appsrv.write.<n>:  List of directory names that uploads can write
                    #                                 to.
                    #NOTE: By default the file transfer is disabled as a security measure
                    transfer.database.enabled=FALSE
                    transfer.appsrv.enabled=FALSE
                    transfer.appsrv.workAreaRoot=
                    transfer.appsrv.accessControl=TRUE
                    #List transfer.appsrv.read.<n> directories
                    transfer.appsrv.read.1=c:\temp
                    #List transfer.appsrv.write.<n> directories
                    transfer.appsrv.write.1=c:\temp
                    # 4. Others
                    # Details
                    # -------
                    # BlockAllowHeartBeat     : To continue the heart beat communication with the
                    #                             server when set to TRUE. By default the value is
                    #                             set to False. When False there would not be heart
                    #                             beat communication in blocking mode.
                    BlockAllowHeartBeat=False
```

## C.11  Default **webutilbase.htm**

This file is located at $ORACLE_
INSTANCE/config/FormsComponent/forms/server/

This template .htm file is used in the WebUtil application section.

```
<HTML>
<!-- FILE: webutilbase.htm (Oracle Forms)                      -->
<!--                                                           -->
<!-- This is the default base HTML file for running a form on the  -->
<!-- web using a generic APPLET tag to include Forms applet.       -->
<!-- and a certificate regsitration applet for the WebUtil utility -->
```

```
<!--                                                     -->
<!-- IMPORTANT NOTES:                                    -->
<!-- Default values for all the variables which appear below    -->
<!-- (enclosed in percent characters) are defined in the servlet -->
<!-- configuration file (formsweb.cfg). It is preferable to make -->
<!-- changes in that file where possible, rather than this one.  -->
<!--                                                     -->
<!-- This file uses several extra tags that are not present in the -->
<!-- default template files.  You should ensure that these are   -->
<!-- present in the configuration that uses this template        -->
<!-- The extra substitution Tags are:                            -->
<!--  %webUtilArchive% = jar file containing the WebUtil code    -->
<!--                    (by default this should be frmwebutil.jar) -->
<!--  %WebUtilLogging% = Defines the current logging mode.       -->
<!--                    Valid values: off|on|console|server|all  -->
<!--                    (on == console)                          -->
<!--  %WebUtilLoggingDetail% = Specifies the level of error logging.-->
<!--                    Valid values: normal|detailed            -->
<!--  %WebUtilErrorMode% = Should errors be displayed in an alert -->
<!--                    as well as the programmer defined        -->
<!--                    locations                                -->
<!--                    Valid values: console|server|alert|all   -->
<!--  %WebUtilDispatchMonitorInterval% = Counts in second to     -->
<!--                    indicate how often the monitor thread    -->
<!--                    checks to see if the Forms session is still-->
<!--                    alive. Used with the WebUtil_Session      -->
<!--                    package.                                  -->
<!--  %WebUtilTrustInternal% = Should intranet without domain suffix-->
<!--                    be trusted.                              -->
<!--                    Valid values: true|yes|false|no          -->
<!--  %WebUtilMaxTransferSize% = Size in bytes of file transfer  -->
<!--                    segments. Default and maximum allowed is  -->
<!--                    16384, i.e. 16K.                         -->

<HEAD><TITLE>%pageTitle% - WebUtil</TITLE></HEAD>

<BODY %HTMLbodyAttrs%>
%HTMLbeforeForm%

<!-- Registration applet definition (start) -->
<APPLET CODEBASE="%codebase%"
        CODE="oracle.forms.webutil.common.RegisterWebUtil"
        ARCHIVE="%webUtilArchive%"
        WIDTH="0"
        HEIGHT="0"
        HSPACE="0"
        VSPACE="0">

</APPLET>
<!-- Registration applet definition (end) -->

<COMMENT id="forms_plugin_info"
        serverURL="%serverURL%"
        appcodebase="%codebase%"
        apparchive="%archive%,%webUtilArchive%"
        appheight="%Height%"
        appwidth="%Width%"
        appname="%applet_name%">
</COMMENT>
```

```
                    <!-- Forms applet definition (start) -->
                    <NOSCRIPT>
                    <APPLET CODEBASE="%codebase%"
                            CODE="oracle.forms.engine.Main"
                            ARCHIVE="%archive%,%webUtilArchive%"
                            WIDTH="%Width%"
                            HEIGHT="%Height%"
                            NAME="%applet_name%" MAYSCRIPT>
                    </NOSCRIPT>
                    <SCRIPT LANGUAGE="JavaScript" SRC="/forms/frmjscript/forms_base_ie.js"></SCRIPT>
                    <PARAM NAME="serverURL" VALUE="%appletServerURL%">
                    <PARAM NAME="networkRetries" VALUE="%networkRetries%">
                    <PARAM NAME="serverArgs"
                          VALUE="%escapeParams% module=%form% userid=%userid% debug=%debug%
                    host=%host% port=%port% %otherParams%">
                    <PARAM NAME="separateFrame" VALUE="%separateFrame%">
                    <PARAM NAME="splashScreen"  VALUE="%splashScreen%">
                    <PARAM NAME="background"  VALUE="%background%">
                    <PARAM NAME="lookAndFeel"  VALUE="%lookAndFeel%">
                    <PARAM NAME="colorScheme"  VALUE="%colorScheme%">
                    <PARAM NAME="serverApp" VALUE="%serverApp%">
                    <PARAM NAME="logo" VALUE="%logo%">
                    <PARAM NAME="imageBase" VALUE="%imageBase%">
                    <PARAM NAME="formsMessageListener" VALUE="%formsMessageListener%">
                    <PARAM NAME="recordFileName" VALUE="%recordFileName%">
                    <PARAM NAME="EndUserMonitoringEnabled" VALUE="%EndUserMonitoringEnabled%">
                    <PARAM NAME="EndUserMonitoringURL" VALUE="%EndUserMonitoringURL%">
                    <PARAM NAME="heartbeat" VALUE="%heartbeat%">
                    <PARAM NAME="heartBeat" VALUE="%heartBeat%">
                    <PARAM NAME="MaxEventWait" VALUE="%MaxEventWait%">
                    <PARAM NAME="allowAlertClipboard" VALUE="%allowAlertClipboard%">
                    <PARAM NAME="disableValidateClipboard" VALUE="%disableValidateClipboard%">
                    <PARAM NAME="enableJavascriptEvent" VALUE="%enableJavascriptEvent%">
                    <PARAM NAME="digitSubstitution" VALUE="%digitSubstitution%">
                    <PARAM NAME="legacy_lifecycle" VALUE="%legacy_lifecycle%">
                    <PARAM NAME="JavaScriptBlocksHeartBeat" VALUE="%JavaScriptBlocksHeartBeat%">
                    <PARAM NAME="highContrast" VALUE="%highContrast%">
                    <PARAM NAME="disableMDIScrollbars" VALUE="%disableMDIScrollbars%">
                    <PARAM NAME="clientDPI" VALUE="%clientDPI%">
                    <!-- Params specific to webutil -->
                    <PARAM NAME="WebUtilLogging" VALUE="%WebUtilLogging%">
                    <PARAM NAME="WebUtilLoggingDetail" VALUE="%WebUtilLoggingDetail%">
                    <PARAM NAME="WebUtilErrormode" VALUE="%WebUtilErrorMode%">
                    <PARAM NAME="WebUtilDispatchMonitorInterval"
                    VALUE="%WebUtilDispatchMonitorInterval%">
                    <PARAM NAME="WebUtilTrustInternal" VALUE="%WebUtilTrustInternal%">
                    <PARAM NAME="WebUtilMaxTransferSize" VALUE="%WebUtilMaxTransferSize%">
                    </APPLET>
                    <!-- Forms applet definition (end) -->
                    %HTMLafterForm%
                    </BODY>
                    </HTML>
```

## C.12  Default webutiljpi.htm

This file is located at `$ORACLE_INSTANCE/config/FormsComponent/forms/server/`

This template `.htm` file is used in the WebUtil application section.

```
<HTML>
<!-- FILE: webutiljpi.htm (Oracle Forms)                       -->
<!--                                                           -->
<!-- This is the default base HTML file for running a form on the -->
<!-- web using the JDK Java Plugin. This is used for example when  -->
<!-- running with Netscape on Unix.                            -->
<!-- and a certificate regsitration applet for the WebUtil utility -->
<!--                                                           -->
<!-- IMPORTANT NOTES:                                          -->
<!-- Default values for all the variables which appear below   -->
<!-- (enclosed in percent characters) are defined in the servlet -->
<!-- configuration file (formsweb.cfg). It is preferable to make -->
<!-- changes in that file where possible, rather than this one. -->
<!--                                                           -->
<!-- This file uses several extra tags that are not present in the -->
<!-- default template files.  You should ensure that these are -->
<!-- present in the configuration that uses this template      -->
<!-- The extra substitution Tags are:                          -->
<!-- %webUtilArchive% = jar file containing the WebUtil code    -->
<!--                    (by default this should be frmwebutil.jar) -->
<!-- %WebUtilLogging% = Defines the current logging mode.      -->
<!--                    Valid values: off|on|console|server|all -->
<!--                    (on == console)                        -->
<!-- %WebUtilLoggingDetail% = Specifies the level of error logging.-->
<!--                    Valid values: normal|detailed          -->
<!-- %WebUtilErrorMode% = Should errors be displayed in an alert -->
<!--                    as well as the programmer defined      -->
<!--                    locations                              -->
<!--                    Valid values: console|server|alert|all -->
<!-- %WebUtilDispatchMonitorInterval% = Counts in second to    -->
<!--                    indicate how often the monitor thread  -->
<!--                    checks to see if the Forms session is still-->
<!--                    alive. Used with the WebUtil_Session    -->
<!--                    package.                                -->
<!-- %WebUtilTrustInternal% = Should intranet without domain suffix-->
<!--                    be trusted.                            -->
<!--                    Valid values: true|yes|false|no        -->
<!-- %WebUtilMaxTransferSize% = Size in bytes of file transfer  -->
<!--                    segments. Default and maximum allowed is -->
<!--                    16384, i.e. 16K.                        -->

<HEAD><TITLE>%pageTitle% - WebUtil</TITLE></HEAD>

<BODY %HTMLbodyAttrs%>
%HTMLbeforeForm%

<!-- Registration applet definition (start) -->
<OBJECT classid="%jpi_classid%"
        codebase="%jpi_codebase%"
        WIDTH="0"
        HEIGHT="0"
        HSPACE="0"
        VSPACE="0">
<PARAM NAME="TYPE"      VALUE="%jpi_mimetype%">
<PARAM NAME="CODEBASE"  VALUE="%codebase%">
<PARAM NAME="CODE"      VALUE="oracle.forms.webutil.common.RegisterWebUtil" >
<PARAM NAME="ARCHIVE"   VALUE="%webUtilArchive%" >
<COMMENT>
<EMBED SRC="" PLUGINSPAGE="%jpi_download_page%"
        TYPE="%jpi_mimetype%"
```

```
                            java_codebase="%codebase%"
                            java_code="oracle.forms.webutil.common.RegisterWebUtil"
                            java_archive="%webUtilArchive%"
                            WIDTH="1"
                            HEIGHT="1"
                            HSPACE="0"
                            VSPACE="0"
            >
            <NOEMBED>
            </COMMENT>
            </NOEMBED></EMBED>
            </OBJECT>
            <!-- Registration applet definition (end) -->

            <COMMENT id="forms_plugin_info"
                        serverURL="%serverURL%"
                        plug_ver="%jpi_classid%"
                        appheight="%Height%"
                        appwidth="%Width%"
                        appcodebase="%jpi_codebase%"
                        appname="%applet_name%">
            </COMMENT>

            <!-- Forms applet definition (start) -->
            <NOSCRIPT>
            <OBJECT classid="%jpi_classid%"
                        codebase="%jpi_codebase%"
                        WIDTH="%Width%"
                        HEIGHT="%Height%"
                        HSPACE="0"
                        VSPACE="0"
                        ID="%applet_name%">
            </NOSCRIPT>
            <SCRIPT LANGUAGE="JavaScript" SRC="/forms/frmjscript/forms_ie.js"></SCRIPT>
            <PARAM NAME="TYPE"        VALUE="%jpi_mimetype%">
            <PARAM NAME="CODEBASE"    VALUE="%codebase%">
            <PARAM NAME="CODE"        VALUE="oracle.forms.engine.Main" >
            <PARAM NAME="ARCHIVE"     VALUE="%archive%,%webUtilArchive%" >

            <PARAM NAME="serverURL" VALUE="%appletServerURL%">
            <PARAM NAME="networkRetries" VALUE="%networkRetries%">
            <PARAM NAME="serverArgs"
                    VALUE="%escapeParams% module=%form% userid=%userid% debug=%debug%
             host=%host% port=%port% %otherParams%">
            <PARAM NAME="separateFrame" VALUE="%separateFrame%">
            <PARAM NAME="splashScreen"  VALUE="%splashScreen%">
            <PARAM NAME="background"  VALUE="%background%">
            <PARAM NAME="lookAndFeel"  VALUE="%lookAndFeel%">
            <PARAM NAME="colorScheme"  VALUE="%colorScheme%">
            <PARAM NAME="serverApp" VALUE="%serverApp%">
            <PARAM NAME="logo" VALUE="%logo%">
            <PARAM NAME="imageBase" VALUE="%imageBase%">
            <PARAM NAME="formsMessageListener" VALUE="%formsMessageListener%">
            <PARAM NAME="recordFileName" VALUE="%recordFileName%">
            <PARAM NAME="EndUserMonitoringEnabled" VALUE="%EndUserMonitoringEnabled%">
            <PARAM NAME="EndUserMonitoringURL" VALUE="%EndUserMonitoringURL%">
            <PARAM NAME="heartBeat" VALUE="%heartBeat%">
            <PARAM NAME="MaxEventWait" VALUE="%MaxEventWait%">
            <PARAM NAME="allowAlertClipboard" VALUE="%allowAlertClipboard%">
            <PARAM NAME="disableValidateClipboard" VALUE="%disableValidateClipboard%">
```

```
<PARAM NAME="enableJavascriptEvent" VALUE="%enableJavascriptEvent%">
<PARAM NAME="MAYSCRIPT" VALUE="%enableJavascriptEvent%">
<PARAM NAME="digitSubstitution" VALUE="%digitSubstitution%">
<PARAM NAME="legacy_lifecycle" VALUE="%legacy_lifecycle%">
<PARAM NAME="JavaScriptBlocksHeartBeat" VALUE="%JavaScriptBlocksHeartBeat%">
<PARAM NAME="highContrast" VALUE="%highContrast%">
<PARAM NAME="disableMDIScrollbars" VALUE="%disableMDIScrollbars%">
<PARAM NAME="clientDPI" VALUE="%clientDPI%">
<!-- Params specific to webutil -->
<PARAM NAME="WebUtilLogging" VALUE="%WebUtilLogging%">
<PARAM NAME="WebUtilLoggingDetail" VALUE="%WebUtilLoggingDetail%">
<PARAM NAME="WebUtilErrorMode" VALUE="%WebUtilErrorMode%">
<PARAM NAME="WebUtilDispatchMonitorInterval"
 VALUE="%WebUtilDispatchMonitorInterval%">
<PARAM NAME="WebUtilTrustInternal" VALUE="%WebUtilTrustInternal%">
<PARAM NAME="WebUtilMaxTransferSize" VALUE="%WebUtilMaxTransferSize%">
<COMMENT>
<EMBED SRC="" PLUGINSPAGE="%jpi_download_page%"
        TYPE="%jpi_mimetype%"
       java_codebase="%codebase%"
        java_code="oracle.forms.engine.Main"
       java_archive="%archive%,%webUtilArchive%"
        WIDTH="%Width%"
        HEIGHT="%Height%"
        HSPACE="0"
        VSPACE="0"
        NAME="%applet_name%"
        serverURL="%appletServerURL%"
        networkRetries="%networkRetries%"
        serverArgs="%escapeParams% module=%form% userid=%userid% debug=%debug%
 host=%host% port=%port% %otherparams%"
        separateFrame="%separateFrame%"
        splashScreen="%splashScreen%"
        background="%background%"
        lookAndFeel="%lookAndFeel%"
        colorScheme="%colorScheme%"
        serverApp="%serverApp%"
        logo="%logo%"
        imageBase="%imageBase%"
        recordFileName="%recordFileName%"
        EndUserMonitoringEnabled="%EndUserMonitoringEnabled%"
        EndUserMonitoringURL="%EndUserMonitoringURL%"
        heartBeat="%heartBeat%"
        MaxEventWait="%MaxEventWait%"
        allowAlertClipboard" VALUE="%allowAlertClipboard%"
        disableValidateClipboard="%disableValidateClipboard%"
        enableJavascriptEvent="%enableJavascriptEvent%"
        MAYSCRIPT="%enableJavascriptEvent%"
        digitSubstitution="%digitSubstitution%"
        legacy_lifecycle="%legacy_lifecycle%"
        JavaScriptBlocksHeartBeat="%JavaScriptBlocksHeartBeat%"
        highContrast="%highContrast%"
        disableMDIScrollbars="%disableMDIScrollbars%"
        clientDPI="%clientDPI%"
        WebUtilLogging="%WebUtilLogging%"
        WebUtilLoggingDetail="%WebUtilLoggingDetail%"
        WebUtilErrormode="%WebUtilErrorMode%"
        WebUtilDispatchMonitorInterval="%WebUtilDispatchMonitorInterval%"
        WebUtilTrustInternal="%WebUtilTrustInternal%"
        WebUtilMaxTransferSize="%WebUtilMaxTransferSize%"
```

```
>
<NOEMBED>
</COMMENT>
</NOEMBED></EMBED>
</OBJECT>
<!-- Forms applet definition (end) -->
%HTMLafterForm%
</BODY>
</HTML>
```

# D

# Forms Error Messages

**FRM-10200: Illegal function in this context.**

    **Cause:** You pressed a key that is not valid in this context.

    **Action:** Press [Show Keys] to view a list of valid function keys.

    **Level:** 25

    **Trigger:** None

**FRM-10201: No parameters needed.**

    **Cause:** You pressed [Enter Application Parameters] or [Enter Menu Parameters], but none are required in this context.

    **Action:** No action required.

    **Level:** 25

    **Trigger:** None

**FRM-10202: Menus are nested too deeply.**

    **Cause:** You tried to select an item that would nest menus more than 10 deep.

    **Action:** Press [Main Menu] to return to the main menu, then navigate to the menu of your choice.

    **Level:** 25

    **Trigger:** None

**FRM-10203: Selected item is not in this menu.**

    **Cause:** In a full-screen menu, you entered a number that exceeds the maximum number of menu items.

    **Action:** Choose an item that is on this menu.

    **Level:** 25

    **Trigger:** None

**FRM-10204: No command defined for the selected background item.**

    **Cause:** You pressed [Background Menu n], where n was greater than the maximum number on the background menu.

    **Action:** No action required. Press [Show Background Menu] to see the valid background menu items.

    **Level:** 25

    **Trigger:** None

**FRM-10205: Menu %s not found.**

**Cause:** In the choice field of a full-screen menu, you entered a menu name that does not exist in this application or is not found in the library.

**Action:** No action is required if the menu does not exist in the application. If it does, recompile the library.

**Level:** 25

**Trigger:** None

**FRM-10206: memory allocation failure**

**Cause:** A memory allocation failed when Forms Runtime attempted a menu operation.

**Action:** Try executing the application when the system is less heavily loaded. If the problem persists, contact Oracle Support Services.

**Level:** 25

**Trigger:** None

**FRM-10207: No background menu present.**

**Cause:** You pressed [Show Background Menu], but no background menu exists.

**Action:** No action required.

**Level:** 25

**Trigger:** None

**FRM-10208: Parameter %s not found.**

**Cause:** A menu item referenced an undefined parameter.

**Action:** Contact your DBA.

**Level:** 25

**Trigger:** None

**FRM-10209: No next menu from background in this context.**

**Cause:** The application attempted to navigate to a named menu from the background menu.

**Action:** No action required.

**Level:** 25

**Trigger:** None

**FRM-10210: Response required.**

**Cause:** You did not enter a required parameter, or you left the choice field blank in a full-screen menu.

**Action:** Make an entry.

**Level:** 25

**Trigger:** None

**FRM-10211: Field must be filled completely.**

**Cause:** You partially entered a parameter that must be entered completely.

**Action:** Enter enough data to completely fill the field.

**Level:** 25

**Trigger:** None

**FRM-10212: Login failed for this username and password.**

    **Cause:** You specified an illegal username and password.

    **Action:** Check the username and password and try again.

    **Level:** 25

    **Trigger:** None

**FRM-10213: Login procedure terminated.**

    **Cause:** You failed to logon to ORACLE three times in a row.

    **Action:** No action required. If you are a valid user, check your user name and password.

    **Level:** 25

    **Trigger:** None

**FRM-10214: No authorization to run any application.**

    **Cause:** You are not a valid user of any module in Oracle Forms.

    **Action:** No action required. If you think that you should be a valid user, ask your DBA to grant you access to the module you wish to run.

    **Level:** 25

    **Trigger:** None

**FRM-10215: No help available.**

    **Cause:** You pressed [Help], but none is available for this item.

    **Action:** No action required.

    **Level:** 25

    **Trigger:** None

**FRM-10216: Failed to spawn a command to the operating system.**

    **Cause:** The operating system could not spawn a sub-process.

    **Action:** Refer to the error message that the operating system issued.

    **Level:** 25

    **Trigger:** None

**FRM-10217: No authorization for any item in selected menu.**

    **Cause:** You tried to move to a menu that has no items you can access.

    **Action:** Check the menu name you entered and try again.

    **Level:** 25

    **Trigger:** None

**FRM-10218: Error for menu %s.**

    **Cause:** Oracle Forms could not read the library information for this menu, or an invalid menu name was specified.

    **Action:** Recompile the library or correct the menu name.

    **Level:** 25

    **Trigger:** None

**FRM-10219: Item number is invalid.**

> **Cause:** In a full-screen menu, you entered an invalid number in the choice field.

> **Action:** Check the item number and re-enter it.

> **Level:** 25

> **Trigger:** None

**FRM-10220: No detailed help available for this item.**

> **Cause:** You pressed [Help], but none is available for this menu item.

> **Action:** No action required.

> **Level:** 25

> **Trigger:** None

**FRM-10221: Cannot read file %s.**

> **Cause:** Either file privileges are set incorrectly, or the library you tried to open is invalid.

> **Action:** Recompile the application library and try again.

> **Level:** 25

> **Trigger:** None

**FRM-10222: Menu %s was created by an old version of the Form Compiler.**

> **Cause:** You are using a newer version of Oracle Forms than the one that created this menu module.

> **Action:** Recompile the menu module and re-execute the command.

> **Level:** 25

> **Trigger:** None

**FRM-10223: Application parameter module does not exist.**

> **Cause:** The parameter information could not be located in the library. This may be due to a library file that is invalid, or one that contains a different application.

> **Action:** Recompile the application library and try again. If this is unsuccessful, contact your DBA.

> **Level:** 25

> **Trigger:** None

**FRM-10224: Application bind variable module does not exist.**

> **Cause:** The bind variable information could not be located in the library. This may be due to an invalid library file.

> **Action:** Recompile the application library and try again. If this is unsuccessful, contact your DBA.

> **Level:** 25

> **Trigger:** None

**FRM-10225: Could not read parameter data.**

> **Cause:** The application library is invalid.

> **Action:** Recompile the application library and try again. If this is unsuccessful, contact your DBA.

**Level:** 25

**Trigger:** None

**FRM-10226: Could not read bind variable data.**

    **Cause:**  The application library is invalid.

    **Action:**  Recompile the application library and try again. If this is unsuccessful, contact your DBA.

    **Level:** 25

    **Trigger:** None

**FRM-10227: Too many menu parameters.**

    **Cause:**  The application contains more menu parameters than can be used on your operating system.

    **Action:**  Revise and recompile the application, or contact your DBA.

    **Level:** 25

    **Trigger:** None

**FRM-10228: Could not read help text.**

    **Cause:**  The application library is invalid.

    **Action:**  Recompile the application library and try again. If this is unsuccessful, contact your DBA.

    **Level:** 25

    **Trigger:** None

**FRM-10229: Could not close file %s.**

    **Cause:**  Operating system error or internal error.

    **Action:**  Contact your DBA.

    **Level:** 25

    **Trigger:** None

**FRM-10230: Application procedure module does not exist.**

    **Cause:**  The procedure information could not be located in the library. This may be due to an invalid library file.

    **Action:**  Recompile the application library and try again. If this is unsuccessful, contact your DBA.

    **Level:** 25

    **Trigger:** None

**FRM-10231: Could not read procedure data.**

    **Cause:**  The application library is invalid.

    **Action:**  Recompile the application library and try again. If this is unsuccessful, contact your DBA.

    **Level:** 25

    **Trigger:** None

**FRM-10233: Navigational procedures/macros not valid in current menu style.**

**Cause:** You tried to use the full-screen, navigational packaged procedures, or macros in the pull-down or menu bar display style.

**Action:** Notify your DBA.

**Level:** 25

**Trigger:** None

**FRM-10234: Semicolon missing in macro statement.**

**Cause:** The command line specified for this item has a syntax error.

**Action:** Notify your DBA.

**Level:** 25

**Trigger:** None

**FRM-10235: Macro %s not found.**

**Cause:** The menu designer specified an undefined macro to be executed.

**Action:** Notify your DBA.

**Level:** 25

**Trigger:** None

**FRM-10236: No procedure/macro specified.**

**Cause:** The menu designer has specified a blank command.

**Action:** Notify your DBA.

**Level:** 25

**Trigger:** None

**FRM-10237: Argument(s) not allowed for this procedure/macro.**

**Cause:** The menu designer specified an argument to a command that does not take arguments.

**Action:** Notify your DBA.

**Level:** 25

**Trigger:** None

**FRM-10238: Error executing %s. Check argument(s).**

**Cause:** The menu designer specified an argument to a command that does not take arguments.

**Action:** Notify your DBA.

**Level:** 25

**Trigger:** None

**FRM-10239: Cannot read form by that name.**

**Cause:** Oracle Forms tried to read a form that does not exist in the current directory.

**Action:** Notify your DBA.

**Level:** 25

**Trigger:** None

**FRM-10240: Form name not specified.**

**Cause:**  Forms Runtime command did not give the name of a form to execute.

**Action:**  Notify your DBA.

**Level:** 25

**Trigger:** None

## FRM-10241: Illegal operation when the Form Builder is active.

**Cause:**  The menu designer specified a Built-in or macro that cannot be used when the Form Builder calls Forms Runtime.

**Action:**  Notify your DBA.

**Level:** 25

**Trigger:** None

## FRM-10242: Cannot call linked-in Forms from Oracle Forms.

**Cause:**  The menu designer specified a call to linked-in Forms from within Oracle Forms.

**Action:**  Notify your DBA.

**Level:** 25

**Trigger:** None

## FRM-10243: Error occurred during invocation of Oracle Forms.

**Cause:**  A call to Forms Runtime failed.

**Action:**  Notify your DBA.

**Level:** 25

**Trigger:** None

## FRM-10244: Application %s does not exist.

**Cause:**  The application name you specified does not exist in the database, or you do not have access privileges to it.

**Action:**  Check the application name and try again, or contact your DBA.

**Level:** 25

**Trigger:** None

## FRM-10245: Already on first item.

**Cause:**  You pressed [Previous Item] from the first item in the parameter form.

**Action:**  No action required. You cannot go to an item prior to the first item in a parameter form.

**Level:** 25

**Trigger:** None

## FRM-10246: Error executing packaged procedure - inactive form.

**Cause:**  The menu designer specified a built-in that cannot be executed in the current context.

**Action:**  Notify your DBA.

**Level:** 25

**Trigger:** None

**FRM-10247: No active items in root menu of application.**

    **Cause:** You tried to open an application but its root menu has no items that you can access. The root menu is either the application's main menu or another menu specified when Oracle Forms called Forms Designer.

    **Action:** Notify your DBA.

    **Level:** 25

    **Trigger:** None

**FRM-10248: No direct menu selection allowed when using a root menu.**

    **Cause:** The root menu is not the module's main menu, because Oracle Forms specified another root menu when calling Forms Designer.

    **Action:** No action required. You can only use direct menu selection when the module's main menu is the root menu.

    **Level:** 25

    **Trigger:** None

**FRM-10249: No authorization to run application %s.**

    **Cause:** You are not a valid user of the application you tried to run.

    **Action:** No action required. If you think you should be a valid user, ask your DBA to grant you access privileges to the application.

    **Level:** 25

    **Trigger:** None

**FRM-10250: Error initializing Forms Runtime application.**

    **Cause:** You did not name the module properly.

    **Action:** Check the module name and enter it correctly.

    **Level:** 25

    **Trigger:** None

**FRM-10251: Unsupported command type 4 switch used (-e,-i,-r,-w).**

    **Cause:** This menu option attempted to run a form, but specified a command line argument for Forms Runtime which is invalid when running a form from a menu.

    **Action:** Notify your DBA.

    **Level:** 25

    **Trigger:** None

**FRM-10252: Unknown command type 4 switch used.**

    **Cause:** This menu option attempted to run a form, but specified an unknown command line argument for Forms Runtime.

    **Action:** Notify your DBA.

    **Level:** 25

    **Trigger:** None

**FRM-10253: File name must be entered.**

    **Cause:** You have not entered a name (or you have deleted a name) for the file.

    **Action:** You must enter a file name.

**Level:** 25

**Trigger:** None

**FRM-10254: Cannot open file for screen shot.**

**Cause:** The operating system could not open a file (e.g. permission problems, lack of disk space).

**Action:** Resolve the operating system condition that caused the error.

**Level:** 25

**Trigger:** None

**FRM-10255: Error occurred during printing of screen shot.**

**Cause:** The operating system had trouble with a file.

**Action:** Resolve the operating system condition that caused the error.

**Level:** 25

**Trigger:** None

**FRM-10256: User is not authorized to run Oracle Forms Menu.**

**Cause:** You are not enrolled in Oracle Forms. You do not have SELECT permission on the Oracle Forms base tables.

**Action:** Notify your DBA.

**Level:** 25

**Trigger:** None

**FRM-10257: User is not authorized to select specified option.**

**Cause:** You tried to select a menu item to which you do not have access.

**Action:** Choose another item or notify your DBA.

**Level:** 25

**Trigger:** None

**FRM-10258: Specified menu is already active.**

**Cause:** You tried to navigate to the current menu.

**Action:** No action required.

**Level:** 25

**Trigger:** None

**FRM-10259: Invalid null argument to packaged procedure or function.**

**Cause:** You did not specify an argument to a built-in, or the argument is invalid.

**Action:** Check the online Help for the proper built-in syntax.

**Level:** 25

**Trigger:** None

**FRM-10260: No active items in selected menu.**

**Cause:** You do not have access privileges for any items in this menu.

**Action:** Contact your DBA for access privileges if you think you should have access to the items on this menu.

**Level:** 25

**Trigger:** None

**FRM-10261: Menu %s was created by a new version of the Form Compiler.**

**Cause:** You are using an old version of Forms Runtime with a new version of the Form Compiler.

**Action:** Upgrade to new version of Forms Runtime.

**Level:** 25

**Trigger:** None

**FRM-10262: Cannot put radio items, check boxes, or separators in menu bar.**

**Cause:** You attempted to put radio items, check boxes, or separators in menu bar.

**Action:** Put these items in a submenu.

**Level:** 25

**Trigger:** None

**FRM-10263: Cannot find icon file for iconic menu item.**

**Cause:** No directory name for this icon.

**Action:** Contact the person who created the menu application.

**Level:** 25

**Trigger:** None

**FRM-10264: Specified menu item does not exist.**

**Cause:** You specified a menu item that does not exist in the form.

**Action:** Try retyping the name or choose another item name.

**Level:** 25

**Trigger:** None

**FRM-10265: Library was created by an old version of the Form Compiler.**

**Cause:** Oracle Forms cannot use library.

**Action:** Recompile the library with the current version of the Form Compiler.

**Level:** 25

**Trigger:** None

**FRM-10266: Library was created by a new version of the Form Compiler.**

**Cause:** Oracle Forms cannot use library.

**Action:** Recompile the library with current version of the Form Compiler.

**Level:** 25

**Trigger:** None

**FRM-10267: Help type magic menu item must be placed on top-level menu.**

**Cause:** You placed a help magic menu item on a submenu.

**Action:** Move the help magic menu item to the top-level menu (main menu).

**Level:** 25

**Trigger:** None

**FRM-10268: Error: Program unit %s in library %s is uncompiled.**

**Cause:** You called an uncompiled program unit from a library.

**Action:** Follow the PL/SQL program error.

**Level:** 25

**Trigger:** None

### FRM-10269: Warning! Program unit %s in library %s is uncompiled.

**Cause:** In debug Forms Runtime, you called an uncompiled program unit in a library.

**Action:** This is just a warning. Forms Runtime will attempt to compile and run the program unit.

**Level:** 25

**Trigger:** None

### FRM-10270: Cannot attach library %s while opening menu %s.

**Cause:** The specified library file is attached to the given menu, but cannot be located in the search path for PL/SQL libraries.

**Action:** Make sure the library file can be located before attempting to run with the specified menu again. For example, have it in the working directory.

**Level:** 25

**Trigger:** None

### FRM-21011: PL/SQL unhandled exception %s.

**Cause:** An unhandled exception occurred while executing a menu trigger.

**Action:** Examine the text of the exception in this message. If this indicates a cause, correct it. If the problem persists, contact Oracle Support Services.

**Level:** 25

**Trigger:** None

### FRM-40007: Invalid user identifier or password. Re-enter.

**Cause:** You entered an incorrect ORACLE username or password.

**Action:** Retype your username and password properly.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-40010: Cannot read form %s.

**Cause:** One of the following:

1. You entered a nonexistent form name.

2. You typed an incomplete path.

3. You do not have the proper privileges to run the form.

4. You do not have a compiled copy of the form.

**Action:** Retype the form name correctly, provide the proper path name, contact your system administrator, or compile the form.

**Level:** 5

**Trigger:** ON-ERROR

**FRM-40011: Form was created by an old version of Oracle Forms.**

**Cause:** The .FMB file was created with an old and incompatible version of the Form Compiler.

**Action:** Recompile the form or relink Generate.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40012: Form was created by a new version of Oracle Forms.**

**Cause:** The .FMB file was created by a new and incompatible version of the Form Compiler.

**Action:** Recompile the form.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40013: Program Error: error occurred while reading form.**

**Cause:** An internal error occurred while Oracle Forms was trying to read the .FMB file.

**Action:** Recompile the form.

**Level:** 99

**Trigger:** None

**FRM-40014: Not enough memory to load the form.**

**Cause:** Internal error. Your computer does not have enough memory to run the form.

**Action:** The designer might be able to modify the form so that it will run. If that is not feasible, your installation must make more memory available, either by modifying the operating system parameters or by adding more memory to the computer.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40015: Unexpected end of file reading form.**

**Cause:** The form was fragmented or incomplete.

**Action:** Recompile the form.

**Level:** 99

**Trigger:** None

**FRM-40019: Unknown screen number to display.**

**Cause:** An internal error occurred.

**Action:** If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** None

**FRM-40020: Page %d too small for this form.**

**Cause:** Application design error. An item is positioned off the page.

**Action:** Ensure that all items that are associated with the given page fit completely on that page. You can reposition the items or resize the page.

**Level:** 99

**Trigger:** None

**FRM-40021: Item in form file is too large.**

    **Cause:** An internal error occurred while Oracle Forms was reading the form.

    **Action:** If the problem persists, contact Oracle Support Services.

    **Level:** 99

    **Trigger:** None

**FRM-40023: Error creating record manager context.**

    **Cause:** Oracle Forms could not initialize its internal record manager.

    **Action:** If the problem persists, contact Oracle Support Services.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-40024: Out of memory.**

    **Cause:** Internal error. Your computer does not have enough memory to run the form.

    **Action:** The designer might be able to modify the form so that it will run. If that is not feasible, your installation must make more memory available, either by modifying the operating system parameters or by adding more memory to the computer.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-40025: Cannot suppress screen output without file input.**

    **Cause:** You tried to run a form on the command line using incompatible preferences. The output_file preference works only in conjunction with the keyin preference.

    **Action:** Retype the command to include both the output_file and keyin preferences.

    **Level:** 99

    **Trigger:** None

**FRM-40026: Error opening key script file.**

    **Cause:** Oracle Forms cannot open the file you specified with the keyin preference.

    **Action:** Make sure the file exists and the file protections are set properly. Or create a file with the keyin preference.

    **Level:** 99

    **Trigger:** None

**FRM-40027: Error opening display spool file.**

    **Cause:** Operating system error. Oracle Forms cannot open a file specified with the output_file preference because there is insufficient disk space or because you have specified an incorrect filename.

    **Action:** Contact your system administrator.

    **Level:** 99

**Trigger:** None

**FRM-40028: Error opening message file %s%s%s.MSB.**
    **Cause:** Oracle Forms cannot find the message file.
    **Action:** Make sure the message file exists and the appropriate path is set.

    **Level:** 99

    **Trigger:** None

**FRM-40029: Already logged on. Must logout before changing connections.**
    **Cause:** The Login() Built-in was issued while already logged on.
    **Action:** Use the Logout() Built-in first.

    **Level:** 25

    **Trigger:** ON-ERROR

**FRM-40030: File %s is not a Forms file.**
    **Cause:** The file specified on the command line was not a valid Oracle Forms file.
    **Action:** Re-enter Forms Runtime startup command with the name of a valid file.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-40031: File %s is not a Forms Runtime file.**
    **Cause:** The file specified on the command line is not a Forms Runtime (.FMX) file.
    **Action:** Re-enter a valid Forms Runtime (.FMX) file.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-40032: Internal Error: file %s contains an improper chunk size.**
    **Cause:** Internal error. File was compiled incorrectly or is corrupted.
    **Action:** Recompile your file.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-40033: Internal Error: file %s contains a bad chunk table.**
    **Cause:** Internal error. File was compiled incorrectly or is corrupted.
    **Action:** Recompile your file.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-40034: Cannot attach the library file.**
    **Cause:** Oracle Forms was unable to find the specified library file.
    **Action:** Exit Forms Runtime and try again.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-40036: Library was created by a new version of Oracle Forms.**
    **Cause:** Oracle Forms unable to use library.

**Action:** Recompile the library with current version of Oracle Forms.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40037: Library was created by an old version of Oracle Forms.**

**Cause:** Oracle Forms unable to use library.

**Action:** Recompile the library with current version of Oracle Forms.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40039: Cannot attach library %s while opening form %s.**

**Cause:** The given library is attached to the form but cannot be located in the search path for PL/SQL libraries.

**Action:** Make sure that the given library can be found and that it has read permissions set.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40040: Cannot perform proxy connection.**

**Cause:** Database privileges for proxying user may not be configured on the database side or database account for SSO user not created.

**Action:** Make sure database is appropriately configured for making proxy connection.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40041: Form %s requires a UTF8 character set.**

**Cause:** An attempt was made to execute the specified form, but it contained one or more items whose datatype was NCHAR, and the NLS_LANG environment variable did not specify the UTF8 or AL32UTF8 character set.

**Action:** Set the NLS_LANG environment variable to a value which specifies the UTF8 or AL32UTF8 character set, and restart the application.

**Level:** 25

**Trigger:** ON-ERROR

**FRM-40100: At first record.**

**Cause:** You pressed [Previous Record] when the cursor was at the first record.

**Action:** No action is necessary.

**Level:** 5

**Trigger:** ON-ERROR

**FRM-40101: Cannot position to a key item. None are navigable.**

**Cause:** You pressed [Next Primary Key Item], but there are no enterable primary key items in this block.

**Action:** Use [Next Item] for navigation rather than [Next Primary Key Item].

**Level:** 10

**Trigger:** ON-ERROR

**FRM-40102: Record must be entered or deleted first.**

    **Cause:** You pressed [Next Record] or [Down] in a context where it is meaningless. Either:

    1. The last record in a block is the current record.

    2. The block is empty.

    3. You are in a new record in the middle of the block created by pressing [Insert Record].

    **Action:** No action is necessary.

    **Level:** 5

    **Trigger:** ON-ERROR

**FRM-40103: Cannot position to a key item. None are queryable.**

    **Cause:** You tried to use [Next Primary Key Item], but none of the primary key items in the block allow you to enter query criteria.

    **Action:** No action is necessary.

    **Level:** 10

    **Trigger:** ON-ERROR

**FRM-40104: No such block: %s.**

    **Cause:** Runtime error. A GO_BLOCK statement references a nonexistent block.

    **Action:** Correct the statement.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-40105: Unable to resolve reference to item %s.**

    **Cause:** Runtime error. A GO_ITEM statement references a nonexistent item.

    **Action:** Correct the statement.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-40106: No navigable items in destination block.**

    **Cause:** Runtime error. A GO_BLOCK statement references a block with no enterable items.

    **Action:** Remove the statement or make at least one item in the block enterable.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-40107: Cannot navigate to non-displayed item %s.**

    **Cause:** Runtime error. A GO_ITEM statement references a non-displayed item.

    **Action:** Remove the statement or turn on the Displayed Property for the indicated item.

    **Level:** 20

    **Trigger:** ON-ERROR

**FRM-40108: No such form: %s.**

> **Cause:**  You attempted to get/set properties of a nonexistent or unloaded form.

> **Action:**  Use a valid form name.

> **Level:** 99

> **Trigger:** ON-ERROR

**FRM-40109: Cannot navigate out of current block in enter-query mode.**

> **Cause:**  You attempted to navigate out of the current block during enter-query mode.

> **Action:**  No action is necessary. You cannot navigate out of the current block or record during enter-query mode.

> **Level:** 99

> **Trigger:** None

**FRM-40110: At first block.**

> **Cause:**  You attempted [Previous Block] when at the first block.

> **Action:**  None. Consider an alternative method of navigation.

> **Level:** 5

> **Trigger:** ON-ERROR

**FRM-40111: At last block.**

> **Cause:**  You attempted [Next Block] when at the last block.

> **Action:**  None. Consider an alternative method of navigation.

> **Level:** 5

> **Trigger:** ON-ERROR

**FRM-40112: Attempted go_item to non enabled item %s:%s.**

> **Cause:**  You attempted to issue a go_item to a non enabled item.

> **Action:**  None. Consider an alternative method of navigation.

> **Level:** 99

> **Trigger:** ON-ERROR

**FRM-40200: Field is protected against update.**

> **Cause:**  You tried to update a field that does not allow updates.

> **Action:**  No action is necessary. You cannot update this field in this form.

> **Level:** 15

> **Trigger:** ON-ERROR

**FRM-40201: Field is full. Can't insert character.**

> **Cause:**  Oracle Forms is in insert mode, and the current field is full.

> **Action:**  Delete a character to make room for the new character or press [Insert/Replace] to activate replace mode.

> **Level:** 99

> **Trigger:** None

**FRM-40202: Field must be entered.**

**Cause:** You have not entered a value (or you have deleted a value) in a field that requires data input.

**Action:** You must enter a value in this field.

**Level:** 15

**Trigger:** ON-ERROR

**FRM-40203: Field must be entered completely.**

**Cause:** You have not entered a complete value (or you have deleted part of a value) in a field that has a fixed length requirement.

**Action:** Enter a complete value (one that extends to the end of the field).

**Level:** 15

**Trigger:** ON-ERROR

**FRM-40204: Cursor is at beginning of field value.**

**Cause:** You tried to delete a character before the first character position of the field.

**Action:** Use [Delete Character] to delete the character that the cursor is on.

**Level:** 10

**Trigger:** ON-ERROR

**FRM-40205: Cursor is beyond the current field value.**

**Cause:** On a block mode terminal, you positioned the cursor out of a field.

**Action:** Move the cursor into the field and try the entry again.

**Level:** 99

**Trigger:** None

**FRM-40206: Previous character is currently hidden.**

**Cause:** You tried to delete a character that is off the screen.

**Action:** Scroll the character you want to delete into view using the arrow keys or [Scroll Left] and [Scroll Right].

**Level:** 10

**Trigger:** ON-ERROR

**FRM-40207: Must be in range %.30s to %.30s.**

**Cause:** You entered a value not in the valid item range.

**Action:** Enter a value in the range shown.

**Level:** 99

**Trigger:** None

**FRM-40208: Form running in query-only mode. Cannot change database fields.**

**Cause:** You entered a value on a query-only form.

**Action:** Do not enter values on this form. You can execute queries and view data, but you cannot alter existing data or enter new data.

**Level:** 15

**Trigger:** ON-ERROR

**FRM-40209: Field must be of form %s.**

    **Cause:** The value that you entered did not match the format mask on the field.

    **Action:** Retry with a field value that matches the format mask.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-40210: Search string not found.**

    **Cause:** Search string does not exist in the module.

    **Action:** Check your search string to make sure it is accurate or try another search string.

    **Level:** 99

    **Trigger:** ON-MESSAGE

**FRM-40211: Warning! Newlines may be stripped from this field.**

    **Cause:** You attempted to assign data with newlines to a single-line text field.

    **Action:** Assign to a multi-line text field if you need the newlines.

    **Level:** 5

    **Trigger:** ON-MESSAGE

**FRM-40212: Invalid value for field %s.**

    **Cause:** Caused by one of the following:

    1. The value is not of the proper data type.

    2. The value does not match any of the list of acceptable values.

    3. For a text field, the value does not match the specified range.

    **Action:** Retry with another value.

    **Level:** 20

    **Trigger:** ON-ERROR

**FRM-40213: Cannot Copy_Region/Cut_Region; region not selected.**

    **Cause:** Region not selected.

    **Action:** Select a region and try again.

    **Level:** 5

    **Trigger:** ON-ERROR

**FRM-40214: Cannot open the clipboard for the copy/cut.**

    **Cause:** Clipboard unavailable.

    **Action:** Platform specific.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-40215: Cannot write to the clipboard.**

    **Cause:** Clipboard unavailable.

    **Action:** Platform specific.

    **Level:** 99

**Trigger:** ON-ERROR

**FRM-40216: Cannot open the clipboard for the paste.**
    **Cause:** Clipboard unavailable.
    **Action:** Platform specific.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-40217: Cannot get the data size from the clipboard.**
    **Cause:** Invalid data.
    **Action:** Platform specific.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-40218: Cannot read from the clipboard.**
    **Cause:** Invalid data.
    **Action:** Platform specific.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-40219: Cannot format the data read from the clipboard.**
    **Cause:** Invalid data.
    **Action:** Platform specific.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-40220: Cannot paste from the clipboard; value too long.**
    **Cause:** Invalid data.
    **Action:** Platform specific.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-40221: Cannot Paste_Region; region not selected.**
    **Cause:** Paste_Region was invoked while no portion of the image item was selected.
    **Action:** Select a region of the image item prior to calling Paste_Region.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-40222: Disabled item '%s.%s' failed validation.**
    **Cause:** Probable application design error. Forms determined that the item contains an invalid value, but it cannot give focus to the item because it is disabled. This could happen because either:

    The application programmatically assigned an invalid valid to the item.

    The application programmatically disabled the item after the end user entered an invalid valid in the item (and before the item was validated).

**Action:** Correct the application logic.

**Level:** 25

**Trigger:** ON-ERROR

**FRM-40223: Field contains an invalid string for security purposes.**

**Cause:** Field contains a string that could be a potential security violation.

**Action:** Remove the offending string.

**Level:** 15

**Trigger:** ON-ERROR

**FRM-40301: Query caused no records to be retrieved. Re-enter.**

**Cause:** No records matched the query criteria. Still in Enter Query mode.

**Action:** Either adjust the query criteria or press [Exit/Cancel] to leave Enter Query mode.

**Level:** 99

**Trigger:** ON-MESSAGE

**FRM-40302: Cannot enter a query. No fields are queryable.**

**Cause:** You pressed [Enter Query] while the cursor was in a block with no queryable fields.

**Action:** No action is necessary.

**Level:** 15

**Trigger:** ON-ERROR

**FRM-40303: No base table fields in the block.**

**Cause:** One of the blocks in the current module has no base table fields.

**Action:** No action is necessary.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40350: Query caused no records to be retrieved.**

**Cause:** The current query fetched no records from the table. The table is empty, or it contains no records that meet the query's search criteria.

**Action:** No action is necessary.

**Level:** 5

**Trigger:** ON-MESSAGE

**FRM-40352: Last record of query retrieved.**

**Cause:** You pressed [Down], [Next Record], [Next Set of Records], or [Scroll Down] after all records had been retrieved.

**Action:** No action is necessary.

**Level:** 5

**Trigger:** ON-MESSAGE

**FRM-40353: Query cancelled.**

**Cause:** You pressed [Exit/Cancel] in Enter Query mode, or you pressed CTRL-C (or its equivalent) while Oracle Forms was fetching rows from the database.

**Action:** No action is necessary.

**Level:** 5 when the query was canceled by CTRL-C; 10 otherwise

**Trigger:** ON-MESSAGE

**FRM-40355: Query will retrieve 1 record.**

**Cause:** You pressed [Count Query Hits]. If you now press [Execute Query], the number of records will be retrieved.

**Action:** No action is necessary.

**Level:** 25

**Trigger:** ON-MESSAGE

**FRM-40356: Invalid number in example record. Query not issued.**

**Cause:** In Enter Query mode, you entered an invalid number in the example record.

**Action:** Correct the entry and retry the query.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40357: Invalid string in example record. Query not issued.**

**Cause:** In query mode, you entered an invalid ALPHA or CHAR value in the example record.

**Action:** Correct the entry and retry the query.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40358: Invalid date in example record. Query not issued.**

**Cause:** In Enter Query mode, you entered an invalid DATE in the example record.

**Action:** Correct the entry and retry the query.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40359: Invalid date or time in example record. Query not issued.**

**Cause:** In Enter Query mode, you entered an invalid JDATE, EDATE, or TIME value in the example record.

**Action:** Correct the entry and retry the query.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40360: Cannot query records here.**

**Cause:** You attempted to query a block that does not allow queries.

**Action:** Do not attempt to query this block.

**Level:** 10

**Trigger:** ON-ERROR

**FRM-40361: Query operation not support for TIME data type.**

**Cause:** You made a query with a % "like" operator in a time field, which is not supported.

**Action:** Try to restate your query without a time data type.

**Level:** 10

**Trigger:** ON-ERROR

**FRM-40364: The data type of item '%s' does not match the corresponding column in the stored procedure.**

**Cause:** The data type of the item is different from the data type of the corresponding column in the stored procedure.

**Action:** Make the data type of the item in the block and the column in the stored procedure the same.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-40367: Invalid criteria in field %s in example record.**

**Cause:** Only simple clauses are allowed in restricted enter query mode.

**Action:** Re-enter the criteria.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40400: Transaction complete: %d records applied and saved.**

**Cause:** Save complete.

**Action:** No action is necessary.

**Level:** 5

**Trigger:** ON-MESSAGE

**FRM-40401: No changes to save.**

**Cause:** No records were added or modified since the last apply or save. Caution: Unapplied database changes that were made through explicit sql (DML) are still applied, even when this message is displayed.

**Action:** No action is necessary.

**Level:** 5

**Trigger:** ON-ERROR

**FRM-40402: Save cancelled.**

**Cause:** You pressed CTRL-C (or the equivalent) while waiting for a lock.

**Action:** No action is necessary.

**Level:** 10

**Trigger:** ON-MESSAGE

**FRM-40403: A calling form has unapplied changes. Save not allowed.**

**Cause:** A calling form has unapplied changes.

**Action:** Apply the changes or return to the calling form and retry the save.

**Level:** 15

**Trigger:** ON-ERROR

**FRM-40404: Database apply complete: %d records applied.**

**Cause:** Apply complete.

**Action:** No action is necessary.

**Level:** 5

**Trigger:** ON-MESSAGE

**FRM-40405: No changes to apply.**

**Cause:** No records were added or modified since the last apply or save.

**Action:** No action is necessary.

**Level:** 5

**Trigger:** ON-ERROR

**FRM-40406: Transaction complete: %d records applied; all records saved.**

**Cause:** You finished an apply that recorded your changes and saved previously applied changes.

**Action:** No action is necessary.

**Level:** 5

**Trigger:** ON-MESSAGE

**FRM-40407: Transaction complete: applied records saved.**

**Cause:** You finished a save that saved previously applied changes.

**Action:** No action is necessary.

**Level:** 5

**Trigger:** ON-MESSAGE

**FRM-40408: database commit failure.**

**Cause:** A database commit failed.

**Action:** Examine integrity constraints on the database tables that were updated. If any were violated, redo the updates without violating the constraints. If necessary, do the updates and the commit in sqlplus, and see if it issues an ORA-nnnnn message that will identify the constraint that was violated.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40501: ORACLE error: unable to reserve record for update or delete.**

**Cause:** A fatal error occurred while trying to select the record for update.

**Action:** Pressing [Display Error] provides more information, if it is available. You can also try to update or delete this record later. If necessary, contact your DBA.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40502: ORACLE error: unable to read list of values.**

**Cause:** A fatal error occurred while trying to read a list of values.

**Action:** Contact your system administrator. If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40504: ORACLE error: unable to execute a %s trigger.**

**Cause:** A fatal error occurred while trying to execute a trigger.

**Action:** Contact your system administrator. If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40505: ORACLE error: unable to perform query.**

**Cause:** Processing error encountered. The table associated with the current block of the form might not exist, or your username might not have authority to perform the specified action on the table.

**Action:** Pressing [Display Error] provides more information, if it is available. You can also try to update or delete this record later. If necessary, contact your DBA.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40506: ORACLE error: unable to check for record uniqueness.**

**Cause:** Processing error encountered while checking a record's primary key items for uniqueness. The table associated with the current block of the form does not exist, or you do not have authority to access the table.

**Action:** Contact your DBA.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40507: ORACLE error: unable to fetch next query record.**

**Cause:** One of the following:

1. A fatal error occurred while trying to fetch the next query record.

2. If you are connected to an non-Oracle datasource through ODBC, the cursor loses its position in the result set after a commit.

**Action:** Requery if you are connected with a non-Oracle datasource. If not, contact your DBA.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40508: ORACLE error: unable to INSERT record.**

**Cause:** A fatal error occurred while trying to insert a record. The table associated with the current block of the form might not exist, your username might not have authority to perform the specified action on the table, or some other reason might have caused the fatal error.

**Action:** Contact your DBA.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40509: ORACLE error: unable to UPDATE record.**

**Cause:** A fatal error occurred while trying to update a record. The table associated with the current block of the form might not exist, your username might not have authority to perform the specified action on the table, or some other reason might have caused the fatal error.

**Action:** Contact your DBA.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40510: ORACLE error: unable to DELETE record.**

**Cause:** A fatal error occurred while trying to delete a record. The table associated with the current block of the form might not exist, your username might not have authority to perform the specified action on the table, or some other reason might have caused the fatal error.

**Action:** Contact your DBA.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40511: ORACLE error occurred while executing a %s trigger.**

**Cause:** A fatal error occurred while trying to execute a trigger. The table associated with the current block of the form might not exist, your username might not have authority to perform the specified action on the table, or some other reason might have caused the fatal error.

**Action:** Contact your DBA

**Level:** 99

**Trigger:** None

**FRM-40512: ORACLE error: unable to issue SAVEPOINT command.**

**Cause:** While attempting to call a new form or to commit, the issued SAVEPOINT command failed. This generally means that the module has run out of savepoints.

**Action:** Press [Display Error] to display the specific ORACLE error. You might be able to increase the maximum number of savepoints in the INIT.ORA file.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40513: ORACLE error: unable to get date/time from database.**

**Cause:** An error occurred while trying to resolve a database date/time initial value.

**Action:** Connect if you have not already done so. Verify database status.

**Level:** 10

**Trigger:** ON-ERROR

**FRM-40514: Operation requires a database connection.**

**Cause:** You tried to perform an database operation without connecting to the database.

**Action:** Connect to the database and retry.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-40515: ORACLE error: unable to open cursor.**

**Cause:** You reached the limit in the number of cursors you can open.

**Action:** Check the number of cursors you have open.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40600: Record has already been inserted.**

**Cause:** You attempted to insert or update a record, but uniqueness is enforced on the block's primary key items. The record, as inserted or updated, is not unique.

**Action:** Change the values in one or more primary key fields of the current record, making them unique. If the requirement of unique primary key fields creates difficulties, consider eliminating the constraint.

**Level:** 25

**Trigger:** ON-ERROR

**FRM-40602: Cannot insert into or update data in a view.**

**Cause:** You tried to modify the contents of a view in a manner that is not permitted.

**Action:** No action is necessary; you cannot perform the operation you have attempted.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-40603: Records no longer reserved for update. Re-query to make changes.**

**Cause:** You committed your modifications in a block where you had previously entered an ENTER_QUERY or EXECUTE_QUERY packaged procedure with the FOR_UPDATE parameter. This action released all locks on the records in this block.

**Action:** If you want to modify the block, you will need to re-query.

**Level:** 99

**Trigger:** ON-MESSAGE

**FRM-40652: Cannot lock table in shared update mode.**

**Cause:** Caused by one of the following:

1. You do not have access to this table.

2. Oracle Forms cannot lock the table in shared update mode.

**Action:** Contact your DBA.

**Level:** 15

**Trigger:** ON-ERROR

**FRM-40653: Record not reserved for update or delete. Try again later.**

**Cause:** You pressed CTRL-C (or the equivalent) to cancel. The operation that was attempting to update or delete the record was terminated.

**Action:** No action is necessary.

**Level:** 20

**Trigger:** ON-MESSAGE

**FRM-40654: Record has been updated by another user. Re-query to see change.**

**Cause:** Another user has updated this record since you performed a query and has changed at least one field in the record. Your actions have not changed the record in memory.

**Action:** You can update or delete this record now only if another user has restored the field values back to the way they were when you performed the query. Otherwise, you must re-query to fetch and display the new record into the form before you can update or delete it.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-40655: SQL error forced rollback: clear form and re-enter transaction.**

**Cause:** A deadlock or some other error has caused the current transaction to fail. Your changes were rolled back.

**Action:** Clear the form (or exit and re-enter the form) and re-enter the transaction. You might have to modify the form's design to prevent the error from recurring.

**Level:** 25

**Trigger:** ON-ERROR

**FRM-40656: Update cannot be made due to prior rollback. Clear the record.**

**Cause:** This record was already updated, but when you attempted to commit your changes, a serious error prevented this update or any further update or delete from being performed. The error might have occurred due to one of the following reasons:

1. A deadlock forced the loss of row locks.

2. The record had a database cluster key, and the previous attempt to update the record in the database was rolled back due to an error somewhere else in the form.

**Action:** You must clear this record before you can commit any other transactions in the form.

**Level:** 25

**Trigger:** ON-ERROR

**FRM-40657: Record changed or deleted by another user.**

**Cause:** Another user has deleted the record since the query was executed, or database access control does not allow the operation.

**Action:** You can clear this record from your screen, but you cannot update or delete it since it no longer exists in the database, or database access control does not allow the operation. Check database access control policy.

**Level:** 20

**Trigger:** ON-MESSAGE

**FRM-40659: Last row of query retrieved. Re-query to see remaining records.**

**Cause:** A FOR_UPDATE query has been closed by executing a commit. Because the query was open prior to the commit, there may be more records to retrieve.

**Action:** Re-query to see remaining records.

**Level:** 5

**Trigger:** ON-MESSAGE

**FRM-40700: No such trigger: %s.**

**Cause:** Application design error. The form attempted to execute a trigger that doesn't exist, causing a fatal error.

**Action:** Correct the reference to the trigger.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-40702: Cannot call form with changes to save**

**Cause:** You attempted to call another form with unsaved changes in the current form and savepoint mode off.

**Action:** Commit/post changes and then retry.

**Level:** 15

**Trigger:** ON-ERROR

**FRM-40703: Fetched field cannot be changed in query mode.**

**Cause:** You attempted to modify a fetched item in query mode.

**Action:** None required.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-40704: Illegal SQL statement in query-only mode**

**Cause:** Application design error. The form tried to execute a function that is illegal in a query-only form.

**Action:** You might need to redesign the form.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-40705: Illegal SQL statement in non-commit-time trigger.**

**Cause:** Application design error. The current trigger contains a SQL statement that is illegal for the trigger type.

**Action:** Rewrite the trigger text or use a different type of trigger.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-40714: Function illegal in this context.**

**Cause:** Application design error. The current trigger contains an illegal function code.

**Action:** Rewrite the trigger text or use a different type of trigger.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-40724: Missing selector in CASE statement.**

**Cause:** Application design error. The selector portion is missing in a CASE statement.

**Action:** Correct the statement.

**Level:** 99

**Trigger:** None

**FRM-40730: Invalid message suppress level--unchanged from %d.**

> **Cause:** Application design error. A trigger attempted to set the system message level to an invalid number.

> **Action:** Reset the SYSTEM.MESSAGE_LEVEL system variable to a valid number.

> **Level:** 99

> **Trigger:** ON-ERROR

**FRM-40732: Target of GOTO does not exist in this macro.**

> **Cause:** Application design error. The label referenced in PL/SQL does not exist.

> **Action:** Correct the statement.

> **Level:** 99

> **Trigger:** ON-ERROR

**FRM-40733: PL/SQL Built-in %s failed.**

> **Cause:** A fatal error occurred in Oracle Forms or in PL/SQL during trigger execution.

> **Action:** Examine application logic to see if the Built-in is invoked incorrectly. If the problem persists, contact Oracle Support Services.

> **Level:** 99

> **Trigger:** ON-ERROR

**FRM-40734: Internal Error: PL/SQL error occurred.**

> **Cause:** An internal error occurred in PL/SQL during trigger execution.

> **Action:** If the problem persists, contact Oracle Support Services.

> **Level:** 99

> **Trigger:** ON-ERROR

**FRM-40735: %s trigger raised unhandled exception %s.**

> **Cause:** Application design error. The current trigger raised an exception (other than FORM_TRIGGER_FAILURE), but it did not handle the exception.

> **Action:** Rewrite the trigger text to handle the exception.

> **Level:** 99

> **Trigger:** ON-ERROR

**FRM-40736: Cannot initialize PL/SQL.**

> **Cause:** An internal error occurred while initializing PL/SQL.

> **Action:** If the problem persists, contact Oracle Support Services.

> **Level:** 99

> **Trigger:** ON-ERROR

**FRM-40737: Illegal restricted procedure %s in %s trigger.**

> **Cause:** Application design error. A trigger tried to execute a restricted packaged procedure.

> **Action:** Remove the packaged procedure from the trigger text.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40738: Argument %d to builtin %s cannot be null.**

**Cause:** Application design error. No arguments were provided to the Built-in.

**Action:** Refer to the online Help for the correct usage of this Built-in.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40739: Full rollback not allowed in post-only form.**

**Cause:** Application design error. A trigger tried to issue a CLEAR_FORM packaged procedure with the FULL_ROLLBACK parameter in a post-only, called form.

**Action:** Remove the FULL_ROLLBACK parameter or ensure that the calling form does not have unposted changes when the call occurs.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40740: Procedure %s only allowed in an on-%s trigger.**

**Cause:** Application design error. A non-transactional trigger attempted to invoke a Built-in procedure that is restricted to a given trigger.

**Action:** Refer to the online Help for the correct usage of this procedure.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40741: Unable to locate record %d on block %s.**

**Cause:** You attempted to get or set record properties for an invalid record number for the given block.

**Action:** Verify your Get/Set record property parameters.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-40742: Illegal status conversion on record %d: %s to %s.**

**Cause:** Application design error. A call to SET_RECORD_PROPERTY attempted an illegal conversion between record statuses.

**Action:** Refer to SET_RECORD_PROPERTY in online Help for correct transitions.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40743: This operation with no base table requires the %s trigger.**

**Cause:** Application design error. Attempted a database operation (query, insert, update, etc.) on a non-base table block without the appropriate transactional trigger.

**Action:** Refer to online Help for the appropriate transactional trigger and then create the correct trigger.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-40744: Truncation of input value will occur if editor accepted.**

  **Cause:**  The editor's buffer is too small to accept the input.

  **Action:**  Change editors or enlarge the buffer.

  **Level:** 99

  **Trigger:** None

**FRM-40745: Output value of Built-in %s was truncated.**

  **Cause:**  Output variable is too small.

  **Action:**  Increase the size of the PL/SQL output variable.

  **Level:** 15

  **Trigger:** ON-ERROR

**FRM-40746: Cannot call Built-in %s from startup debugger window.**

  **Cause:**  Built-in is not accessible from the startup debugger window.

  **Action:**  Refer to the Oracle Forms Developer's Guide for a list of Built-ins that are not accessible from the startup debugger window.

  **Level:** 99

  **Trigger:** None

**FRM-40747: Cannot call Built-in %s from a debug trigger.**

  **Cause:**  Built-in is not accessible from the debug trigger.

  **Action:**  Refer to the Oracle Forms Developer's Guide for a list of Built-ins that are accessible from a debug trigger .

  **Level:** 99

  **Trigger:** None

**FRM-40748: Trigger %s terminated by reset command.**

  **Cause:**  You issued the reset command or you pressed the reset button in the debugger.

  **Action:**  If you want to navigate downward, go to the call stack.

  **Level:** 99

  **Trigger:** None

**FRM-40749: Invalid record status specified for record %d.**

  **Cause:**  Application design error. An attempt was made to set the Status Property of a record to an invalid value.

  **Action:**  The record's Status Property should be set to NEW_STATUS, QUERY_ STATUS, INSERT_STATUS, or CHANGED_STATUS.

  **Level:** 99

  **Trigger:** ON-ERROR

**FRM-40750: Record %d: Can't set status to QUERY or CHANGED in a control block.**

  **Cause:**  Application design error. An attempt was made to set the Status Property of a record in a control block to QUERY_STATUS or CHANGED_STATUS.

  **Action:**  The record's Status Property should be set to NEW_STATUS or INSERT_ STATUS.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40800: User exit %s does not exist.**

**Cause:** The form tried to invoke a user exit that does not exist. This could be caused by one of the following:

1. You are using the wrong version of Forms Runtime.

2. There could be an error in the form.

3. FORMS_USEREXITS is not set correctly.

**Action:** Make sure that the dynamic library which defines the user exit symbol is listed in FORMS_USEREXITS list.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-40801: memory allocation failure**

**Cause:** A memory allocation failed when Forms Runtime attempted to create an internal macro.

**Action:** Try executing the application when the system is less heavily loaded. If the problem persists, contact Oracle Support Services.

**Level:** 25

**Trigger:** ON-ERROR

**FRM-40808: Cannot execute HOST command. Error code = %s.**

**Cause:** Cannot execute a HOST statement because of an operating system error.

**Action:** Contact your system administrator.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40809: HOST command had error code = %s.**

**Cause:** The operating system command resulted in the above error code.

**Action:** Verify that you entered the command properly.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40811: Shell command had error.**

**Cause:** The operating system command resulted in the above error code.

**Action:** Verify that you entered the command properly.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40815: Variable GLOBAL.%s does not exist.**

**Cause:** Application design error. A trigger references a global variable that does not exist.

**Action:** Create the global variable or remove the reference.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-40816: Could not allocate memory for new symbol.**

**Cause:** A memory allocation failed when Forms Runtime attempted to access a new global variable.

**Action:** Try executing the application when the system is less heavily loaded. If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** None

**FRM-40817: Could not allocate memory for new value.**

**Cause:** A memory allocation failed when Forms Runtime attempted to access a new global variable.

**Action:** Try executing the application when the system is less heavily loaded. If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** None

**FRM-40818: System variable name not defined.**

**Cause:** You have tried to access a system variable that does not exist.

**Action:** Check the system variable name.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40819: System variable is not modifiable.**

**Cause:** You have tried to modify a system variable.

**Action:** You cannot modify system variables.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40820: Not enough memory to evaluate system variable.**

**Cause:** A memory allocation failed when Forms Runtime attempted to access a system variable.

**Action:** Try executing the application when the system is less heavily loaded. If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40828: CALL or CALLQRY with invalid variable reference.**

**Cause:** Application design error. A CALL or CALLQRY function code contains an invalid variable reference.

**Action:** Correct the statement.

**Level:** 99

**Trigger:** None

**FRM-40831: Truncation occurred: value too long for field %s.**

**Cause:** Application design error. A trigger, query, or user exit read a value into an item that is not long enough to hold the entire value. The item truncated the value.

**Action:** Increase the target item's item length to avoid truncation.

**Level:** 15

**Trigger:** ON-ERROR

### FRM-40832: Variable FORMS_USEREXITS not set. User exit %s did not execute.

**Cause:** FORMS_USEREXITS must be set for forms USER_EXIT Built-in to work.

**Action:** Set FORMS_USEREXITS to dynamic libraries, which define the user exit symbols. Searching is done in the order libraries are listed.

**Level:** 20

**Trigger:** ON-ERROR

### FRM-40833: Could not completely load the dynamic user exit libraries. User exit %s did not execute.

**Cause:** User exit symbol was not found and there were failures in opening some of the dynamic user exit libraries.

**Action:** Make sure that all the user exit libraries listed in FORMS_USEREXITS are correct and available.

**Level:** 20

**Trigger:** ON-ERROR

### FRM-40900: Unable to allocate record buffer. Clear form to continue.

**Cause:** Failed attempt to allocate memory for a fetched or new record. The remaining records that are queried are temporarily buffered on disk. This is an indication that no more temporary files can be written to the disk.

**Action:** Clear form and then attempt to continue. You may have to exit the form and then re-open. Also, make sure there is enough disk space and that you have write privileges to the disk drive and directory.

**Level:** 99

**Trigger:** ON-MESSAGE

### FRM-40901: Note: not enough memory to remember all or part of this query.

**Cause:** A memory allocation failed when Forms Runtime attempted to save a query.

**Action:** Try executing the application when the system is less heavily loaded. If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-40902: SQL statement too large.

**Cause:** Application design error. The form's design includes a SQL command that is more than 2048 characters long.

**Action:** Shorten the SQL command.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-40903: Cannot create output file.

**Cause:** You pressed [Print Screen], but screen contents could not be written to a file because of one of the following:

1. You have entered an illegal file name.

2. The operating system does not give you authority to create files.

3. The necessary disk or directory space is not available.

**Action:** Check the file name you have entered and correct it if necessary. If you need additional help, contact your system administrator.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-40904: Program error: unknown operation to be performed on record.
**Cause:** Internal error.

**Action:** If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-40905: Unable to buffer more records on disk.
**Cause:** Internal error.

**Action:** If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** None

### FRM-40906: FATAL ERROR: cannot write a buffered record to disk.
**Cause:** Internal error while trying to write a buffered record to the disk.

**Action:** If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-40907: FATAL ERROR: cannot read a buffered record from disk.
**Cause:** Internal error while trying to read a buffered record from the disk.

**Action:** If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-40908: RAM Internal Error: %s
**Cause:** An internal error occurred within the form's internal record manager.

**Action:** If the problem persists, contact Oracle Support Services.

**Level:** 20

**Trigger:** ON-ERROR

### FRM-40909: Internal Error: unknown error %d.
**Cause:** Internal error. Forms Runtime attempted to issue an unknown error message.

**Action:** If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** None

**FRM-40911: Record not created due to sequence number generation error.**

**Cause:** Internal error. Either the sequence number object does not exist, or the designer does not have privileges for the sequence number object, or some other fatal database error occurred.

**Action:** Contact your DBA. If your DBA cannot correct the problem, and the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40912: WHEN-NEW-RECORD trigger failed. Record not created.**

**Cause:** A runtime error occurred in a When-New-Record trigger that caused the trigger to fail. No new record was created.

**Action:** Contact your DBA. If your DBA cannot correct the problem, and the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40913: List of Values maximum exceeded. Some values are not displayed.**

**Cause:** Application design error. Unable to return all the records in the current list of values; the number exceeds the maximum limit.

**Action:** Specify no more than 32,767 records to be returned in a list of values.

**Level:** 25

**Trigger:** ON-ERROR

**FRM-40914: Memory allocation error: unable to complete transaction.**

**Cause:** A memory allocation failed while Forms Runtime attempted to complete a transaction.

**Action:** Try executing the application when the system is less heavily loaded. If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40915: Memory allocation error: unable to execute trigger %s.**

**Cause:** A memory allocation failed while Forms Runtime attempted to execute a trigger.

**Action:** Try executing the application when the system is less heavily loaded. If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40916: Memory allocation error: unable to execute query.**

**Cause:** A memory allocation failed while Forms Runtime attempted to execute a query.

**Action:** Try executing the application when the system is less heavily loaded. If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40917: Memory allocation error: unable to lock record.**

> **Cause:** A memory allocation failed while Forms Runtime attempted to lock a record.

> **Action:** Try executing the application when the system is less heavily loaded. If the problem persists, contact Oracle Support Services.

> **Level:** 99

> **Trigger:** ON-ERROR

**FRM-40919: Internal SQL statement execution error: %d.**

> **Cause:** Error in the SQL statement Oracle Forms has tried to execute.

> **Action:** Check the last SQL statement.

> **Level:** 25

> **Trigger:** ON-ERROR

**FRM-40920: Unable to create view: low on system resources.**

> **Cause:** Something in your environment or application has prevented view creation.

> **Action:** Try executing the application when the system is less heavily loaded. If the problem persists, contact Oracle Support Services.

> **Level:** 99

> **Trigger:** ON-ERROR

**FRM-40921: Could not create item: %s.**

> **Cause:** Something in your environment or application has prevented item creation.

> **Action:** Try executing the application when the system is less heavily loaded. If the problem persists, contact Oracle Support Services.

> **Level:** 99

> **Trigger:** ON-ERROR

**FRM-40922: An OLE error occurred: 0x%x.**

> **Cause:** A Built-in called an OLE or OLE-related function which failed.

> **Action:** You need to lookup the error number in an OLE manual for further details.

> **Level:** 99

> **Trigger:** ON-ERROR

**FRM-40923: OLE is not supported on this platform.**

> **Cause:** A Built-in that required OLE support was called, and your platform does not support OLE.

> **Action:** Don't call OLE-related Built-ins on platforms that don't support OLE.

> **Level:** 99

> **Trigger:** ON-ERROR

**FRM-40924: Invalid argument index specified.**

> **Cause:** An attempt was made to retrieve a value from the OLE-argument stack whose index was out of bounds for the size of the current OLE-argument stack

**Action:** Argument indices range from 1 to the number specified in the last call to FORMS_OLE.InitArgs()

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40925: No space initialized in OleArg for argument.**

**Cause:** An attempt was made to store too many arguments into the initialized OLE-argument stack.

**Action:** Make sure you specify enough space in your call to FORMS_OLE.InitArgs().

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40926: OLE Object is NULL.**

**Cause:** You cannot operate on a NULL OLE object.

**Action:** Do not attempt to call FORMS_OLE Built-ins with NULL OLE-objects.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40927: Variant is not an array.**

**Cause:** An attempt was made to access a variant as if it contained an array, and it did not.

**Action:** You can call FORMS_OLE.Get_Dims() to ensure that you have a variant with an array. For arrays, the return value for the function is greater than or equal to 1.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40928: Too many array indices specified.**

**Cause:** An attempt was made to access a variant that holds an array, but too many array indices were specified.

**Action:** You must use the correct number of array indices, the same number as returned by FORMS_OLE.GET_Dims().

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40929: Too few array indices specified.**

**Cause:** An attempt was made to access a variant that holds an array, but too few array indices were specified.

**Action:** You must use the correct number of array indices, the same number as returned by FORMS_OLE.GET_Dims().

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40930: Array index was non-numeric.**

**Cause:** An attempt was made to access a variant that holds an array, but the supplied array indices were non-numeric and not either ROW or COLUMN.

**Action:** The only valid array indices are numbers, which should be separated by columns. If you're fetching into a table from a variant, ROW and COLUMN can be used as placeholders for the row and column iterators during table construction.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40931: Cannot populate table because datatype is unsupported.**

**Cause:** An attempt to populate a table failed because one of its columns used an unsupported datatype.

**Action:** Restrict your column types to integers, numbers, strings, and dates.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40932: Cannot populate variant because table's datatype is unsupported.**

**Cause:** An attempt to populate a variant failed because one of the source table's columns used an unsupported datatype.

**Action:** Restrict your column types to integers, numbers, strings, and dates.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40933: Cannot populate table because datatype is incorrect.**

**Cause:** An attempt to populate a table failed because the block's datatype did not match the table's datatype.

**Action:** The table datatype must match the datatypes of the block's columns that are being retrieved.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40934: Cannot populate table because records are out of bounds.**

**Cause:** An attempt to populate a table failed because an illegal start or end record was specified.

**Action:** start_rec and end_rec parameters must fall between 1 and the number of retrievable records. end_rec may also be ALL_RECORDS.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-40935: Object does not exist locally.**

**Cause:** An attempt to release an object failed because the 'kill_persistent' parameter was set to FALSE, and no local object existed to release.

**Action:** Never release objects you don't own.

**Level:** 10

**Trigger:** ON-ERROR

**FRM-41000: This function is not currently available.**

**Cause:** You pressed an undefined function key.

**Action:** Press [Show Keys] to determine which function key you should have pressed.

**Level:** 5

**Trigger:** ON-ERROR

**FRM-41001: This function is not allowed on this device.**

    **Cause:** You tried to execute the Insert/Replace function.

    **Action:** No action is necessary.

**Level:** 5

**Trigger:** ON-ERROR

**FRM-41002: Please make a valid selection.**

    **Cause:** You entered an invalid selection number on the block menu; that block does not exist in this form.

    **Action:** Select an existing block.

**Level:** 10

**Trigger:** ON-ERROR

**FRM-41003: This function cannot be performed here.**

    **Cause:** You tried to perform a function that references a table, but current block does not correspond to any table.

    **Action:** No action is necessary. You cannot perform the requested function on this block.

**Level:** 10

**Trigger:** ON-ERROR

**FRM-41004: This function is not allowed in this mode.**

    **Cause:** You pressed a function key that does not work in this mode.

    **Action:** No action is necessary.

**Level:** 10

**Trigger:** ON-ERROR

**FRM-41005: Internal Error: function key not implemented.**

    **Cause:** You pressed a disabled function key.

    **Action:** No action is necessary. You cannot use the function key in the current context unless the form's definition is modified.

**Level:** 25

**Trigger:** ON-ERROR

**FRM-41007: Cursor not in a valid item. Function key was ignored.**

    **Cause:** You were not in a valid item when you pressed the function key.

    **Action:** Position the cursor inside the item and press the function key again.

**Level:** 10

**Trigger:** ON-ERROR

**FRM-41008: Undefined function key. Press %s for list of valid keys.**

    **Cause:** You pressed an undefined function key.

    **Action:** Press [Show Keys] to determine which function key you should have pressed.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41009: Function key not allowed. Press %s for list of valid keys.**

**Cause:** You pressed a function key that is not allowed in this environment.

**Action:** Press [Show Keys] to determine which function key you should have pressed.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41010: Cannot set attribute of the current item.**

**Cause:** Application design error. A SET_ITEM statement tried to turn off the Input Allowed Property for the current item.

**Action:** Eliminate the statement or rewrite the trigger.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41011: Undefined visual attribute.**

**Cause:** Application design error. A Built-in tried to set an undefined visual attribute.

**Action:** Correct the statement.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41012: Undefined item or variable reference.**

**Cause:** Application design error. A NAME_IN statement tried to reference a nonexistent item or variable.

**Action:** Correct the statement.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41013: Undefined property specified for item %s.%s.**

**Cause:** Application design error. A SET_ITEM_PROPERTY or SET_ITEM_ INSTANCE_PROPERTY Built-in specified an undefined property.

**Action:** Correct the statement.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41014: Cannot set property of null canvas item %s.%s.**

**Cause:** Application design error. A SET_ITEM_PROPERTY or SET_ITEM_ INSTANCE_PROPERTY Built-in tried to change some property of a NULL canvas item.

**Action:** Specify a canvas for the item, or remove the statement.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41015: Cannot set ENTERABLE Property of the current item %s.%s.**

**Cause:** Application design error. A SET_ITEM_PROPERTY Built-in tried to change the Enterable Property of the current item.

**Action:** Correct the statement.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41016: Cannot set DISPLAYED Property of the current item %s.%s.**

**Cause:** Application design error. A SET_ITEM_PROPERTY Built-in tried to change the Displayed Property of the current item.

**Action:** Correct the statement.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41017: Cannot set UPDATE ALLOWED Property of non-enabled item %s.%s.**

**Cause:** Application design error. A SET_ITEM_PROPERTY or SET_ITEM_ INSTANCE_PROPERTY Built-in tried to turn on the Update Allowed Property of a non-enterable item.

**Action:** To turn on the Update Allowed Property of an item you must also turn on the Input Allowed Property of the item.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41018: Cannot set UPDATE_NULL Property of non-enabled item %s.%s.**

**Cause:** Application design error. A SET_ITEM_PROPERTY Built-in tried to turn on the Update If Null Property of a non-enterable item.

**Action:** To turn on the Update If Null Property of an item you must also turn on the Input Allowed Property of the item.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41019: Cannot set REQUIRED Property of non-enabled item %s.%s.**

**Cause:** Application design error. A SET_ITEM_PROPERTY or SET_ITEM_ INSTANCE_PROPERTY Built-in tried to turn on the Required Property of a non-enterable item.

**Action:** To turn on the Required Property of an item you must also turn on the Input Allowed Property of the item.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41020: Cannot set ENTERABLE Property of non-displayed item %s.%s.**

**Cause:** Application design error. A SET_ITEM_PROPERTY Built-in tried to turn on the Enterable Property of a non-displayed item.

**Action:** To turn on the Input Allowed Property of an item you must also turn on the Displayed Property of the item.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41021: Cannot set QUERYABLE Property of non-displayed item %s.%s.**

**Cause:** Application design error. A SET_ITEM_PROPERTY Built-in tried to turn on the Query Allowed Property of a non-displayed item.

**Action:** To turn on the Query Allowed Property of an item you must also turn on the Displayed Property of the item.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41022: Cannot set REQUIRED Property of non-updateable item %s.%s.**

**Cause:** Application design error. A SET_ITEM_PROPERTY or SET_ITEM_INSTANCE_PROPERTY Built-in tried to turn on the Required Property of a non-updateable item.

**Action:** To turn on the Required Property of an item you must also turn on either the Update Allowed Property or the Update If Null Property of the item.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41023: Cannot set UPDATE ALLOWED Property of secure item %s.%s.**

**Cause:** Application design error. A SET_ITEM_PROPERTY or SET_ITEM_INSTANCE_PROPERTY Built-in tried to change the Update Allowed Property of a database item which the user does not have permission to update.

**Action:** Either correct the SET_ITEM statement or grant update permission on the column to the user.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41024: Cannot set UPDATE_NULL Property of secure item %s.%s.**

**Cause:** Application design error. A SET_ITEM_PROPERTY Built-in tried to change the Update If Null Property of a database item which the user does not have permission to update.

**Action:** Either correct the SET_ITEM statement or grant update permission on the column to the user.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41025: Page number %d does not exist.**

**Cause:** Application design error. Attempted an operation on a non-existent page.

**Action:** Check arguments to page related Built-ins.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41026: Field does not understand operation.**

**Cause:** You attempted to perform an operation that is invalid for the given item type.

**Action:** Do not attempt to perform the operation on an item to which the operation cannot be applied.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41027: Primary key must be defined for this block.**

**Cause:** There are no primary key items in the block and one of the following has happened:

1. You attempted to set Key_Mode of primary key option on for the block.

2. The Key_Mode is set to Automatic and the datasource to which your are connected does not support UNIQUE key mode.

**Action:** Specify one or more primary key items on the block.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41028: Invalid property.**

**Cause:** You passed an invalid property constant to a Get or Set property Built-in.

**Action:** Verify arguments.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41029: Invalid parameter.**

**Cause:** You attempted to set a form, block, item, or record property to an invalid value.

**Action:** Verify arguments to SET_FORM_PROPERTY, SET_BLOCK_PROPERTY, SET_ITEM_PROPERTY, or SET_RECORD_PROPERTY.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41030: Cannot reset ITEM_LENGTH of item %s.%s.**

**Cause:** Application design error. Attempted to change the length of a fixed length item.

**Action:** Statically declare the item to be of the maximum necessary length or change item type.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41031: Cannot reset ITEM_LENGTH greater than the allocated buffer.**

**Cause:** Application design error. Tried to reset ITEM_LENGTH greater than the allocated buffer.

**Action:** Increase item length in the form definition.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41032: Cannot set ENABLED Property of current item %s.%s.**

**Cause:** A call to SET_ITEM_PROPERTY attempted to set the Enabled Property of the current item.

**Action:** Either correct the call to SET_ITEM_PROPERTY or navigate to another item before setting the Enabled Property.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41033: Cannot set ENABLED Property of non-displayed item %s.%s.**

**Cause:** A call to SET_ITEM_PROPERTY attempted to set the Enabled Property of a non-displayed item.

**Action:** First navigate to the item, then set the Enabled Property with a call to SET_ITEM_PROPERTY.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41034: Cannot set NAVIGABLE Property of non-displayed item %s.%s.**

**Cause:** A call to SET_ITEM_PROPERTY attempted to set the Navigable Property of a non-displayed item.

**Action:** First navigate to the item, then set the Navigable Property with a call to SET_ITEM_PROPERTY.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41035: Cannot set NAVIGABLE Property of non-enabled item %s.%s.**

**Cause:** A call to SET_ITEM_PROPERTY attempted to set the Navigable Property of a non-enabled item.

**Action:** First set the Enabled Property of the item with a call to SET_ITEM_ PROPERTY. Then set the Navigable Property of the item with another call to SET_ ITEM_PROPERTY.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41036: Cannot modify a checkbox that does not allow querying.**

**Cause:** You attempted to modify a check box that does not allow querying.

**Action:** First set the Query Allowed Property to True, then the end user may shift and click the check box to enable or disable the item.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41037: Cannot modify a radio group that does not allow querying.**

**Cause:** You attempted to modify a radio group that does not allow querying.

**Action:** First set the Enabled Property to True with a call to SET_RADIO_ BUTTON_PROPERTY.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41038: Item %s is not a checkbox.**

**Cause:** A call to CHECKBOX_CHECKED was made to an item which was not a check box.

**Action:** Correct the call to CHECKBOX_CHECKED.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41039: Invalid Alert ID %d.**

   **Cause:**  An invalid ID was passed to a Built-in subprogram.

   **Action:**  Verify that a proper call to FIND_ALERT will be performed.

   **Level:** 99

   **Trigger:** ON-ERROR

**FRM-41040: Cannot find radio button: %s.**

   **Cause:**  An invalid ID or name was passed to a Built-in subprogram.

   **Action:**  Check the name or ID that you entered and try again.

   **Level:** 99

   **Trigger:** ON-ERROR

**FRM-41041: Cannot find form module: invalid ID.**

   **Cause:**  An invalid ID was passed to a Built-in subprogram.

   **Action:**  Verify that a proper call to FIND_FORM will be performed.

   **Level:** 99

   **Trigger:** ON-ERROR

**FRM-41042: No such property for Set_Item_Property.**

   **Cause:**  You attempted to set an invalid item property.

   **Action:**  Check the documentation for setting item properties and try again.

   **Level:** 99

   **Trigger:** ON-ERROR

**FRM-41043: Cannot find timer: invalid ID.**

   **Cause:**  An invalid ID was passed to a Built-in subprogram.

   **Action:**  Verify that a proper call to FIND_TIMER will be performed.

   **Level:** 99

   **Trigger:** ON-ERROR

**FRM-41044: Error deleting timer %s**

   **Cause:**  An error occurred while executing a DELETE_TIMER Built-in.

   **Action:**  Verify that your timer has been created correctly.

   **Level:** 99

   **Trigger:** ON-ERROR

**FRM-41045: Cannot find item: invalid ID.**

   **Cause:**  An invalid ID was passed to a Built-in subprogram.

   **Action:**  Verify that a proper call to FIND_ALERT will be performed.

   **Level:** 99

   **Trigger:** ON-ERROR

**FRM-41046: Invalid parameter used for Set_Item_Property.**

   **Cause:**  An invalid parameter was passed to SET_ITEM_PROPERTY.

   **Action:**  Verify the valid parameters for SET_ITEM_PROPERTY and try again.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41047: Cannot navigate out of current block in enter-query mode.**

**Cause:** An illegal attempt to navigate out of the current block when in Enter Query mode.

**Action:** Perform operation before entering query mode.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41048: Procedure %s is not valid in a %s trigger.**

**Cause:** The indicated procedure is not valid when called from the indicated trigger. The procedure may be a restricted procedure, which cannot be called from any trigger that fires during navigation.

**Action:** Correct the invalid trigger.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41049: You cannot delete this record.**

**Cause:** You attempted to delete a record on a block that does not allow deletes.

**Action:** Do not attempt to delete records in this block until you have set the Delete Allowed Property to True.

**Level:** 10

**Trigger:** ON-ERROR

**FRM-41050: You cannot update this record.**

**Cause:** You attempted to update a record on a block that does not allow updates.

**Action:** Do not attempt to update records in this block until you have set the Update Allowed Property to True.

**Level:** 10

**Trigger:** ON-ERROR

**FRM-41051: You cannot create records here.**

**Cause:** You attempted to create records on a block that does not allow inserts.

**Action:** Do not attempt to create and insert new records into this block until you have set the Insert Allowed Property to True.

**Level:** 10

**Trigger:** ON-ERROR

**FRM-41052: Cannot find Window: invalid ID.**

**Cause:** An invalid ID was passed to a Built-in subprogram.

**Action:** Verify that a proper call to FIND_WINDOW will be performed.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41053: Cannot find Canvas: invalid ID.**

**Cause:** An invalid ID was passed to a Built-in subprogram.

**Action:** Verify that a proper call to FIND_CANVAS will be performed.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41054: No such property for Get_Record_Property.**

**Cause:** You attempted to get a non-existent record property.

**Action:** Verify call to GET_RECORD_PROPERTY for valid property.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41055: No such property for Set_Record_Property.**

**Cause:** You attempted to set a non-existent record property.

**Action:** Verify call to SET_RECORD_PROPERTY for valid property.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41056: Cannot find Block: invalid ID.**

**Cause:** An invalid ID was passed to a Built-in subprogram.

**Action:** Verify that a proper call to FIND_BLOCK will be performed.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41057: No such property for Set_View_Property.**

**Cause:** You attempted to set a non-existent view property.

**Action:** Verify call to SET_VIEW_PROPERTY for valid property.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41058: No such property for Get_Item_Property.**

**Cause:** You attempted to get a non-existent item property.

**Action:** Verify call to GET_ITEM_PROPERTY for valid property.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41059: No such property for Set_Canvas_Property.**

**Cause:** You attempted to set a non-existent canvas property.

**Action:** Verify call to SET_CANVAS_PROPERTY for valid property.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41060: Cannot disable Primary Key Property of only key item.**

**Cause:** You attempted to turn off the Primary Key Property on the last primary key item on a block with one of the following:

1. The Primary Key Property.

2. Key mode of the primary key.

3. The database to which you are connected does not support UNIQUE key mode.

**Action:** Disable the block properties first.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41061: No such property for Get_Window_Property.**
**Cause:** You attempted to get a non-existent window property.
**Action:** Verify call to GET_WINDOW_PROPERTY for valid property.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41062: Cannot find Editor: invalid ID.**
**Cause:** An invalid ID was passed to a Built-in subprogram.
**Action:** Verify that a proper call to FIND_EDITOR will be performed.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41063: Cannot create Editor.**
**Cause:** Not enough memory available for Forms Runtime.
**Action:** Try calling SHOW_EDITOR again after closing some of your windows.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41064: Cannot create Timer %s: illegal identifier name.**
**Cause:** Illegal identifier name.
**Action:** Check legal syntax for naming timers.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41065: Cannot find Menu: invalid ID.**
**Cause:** An invalid ID was passed to a Built-in subprogram.
**Action:** Verify that a proper call to FIND_MENU will be performed.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41066: No such property for Get_Form_Property.**
**Cause:** You attempted to get a non-existent form property.
**Action:** Verify call to GET_FORM_PROPERTY for valid property.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41067: Cannot find Menu Item: invalid ID.**
**Cause:** An invalid ID was passed to a Built-in subprogram.
**Action:** Verify that a proper call to FIND_MENU_ITEM will be performed.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41068: Error in Set_Menu_Item_Property.**

    **Cause:** Invalid call to SET_MENU_ITEM_PROPERTY.

    **Action:** Verify valid parameters and try again.

    **Level:** 20

    **Trigger:** ON-ERROR

**FRM-41069: Error in Get_Menu_Item_Property.**

    **Cause:** Invalid call to GET_MENU_ITEM_PROPERTY.

    **Action:** Verify valid parameters and try again.

    **Level:** 20

    **Trigger:** ON-ERROR

**FRM-41070: Unknown property for Set_Menu_Item_Property.**

    **Cause:** You attempted to set a non-existent menu item property.

    **Action:** Verify call to SET_MENU_ITEM_PROPERTY for valid property.

    **Level:** 20

    **Trigger:** ON-ERROR

**FRM-41071: Unknown property for Get_Menu_Item_Property.**

    **Cause:** You attempted to get a non-existent menu item property.

    **Action:** Verify call to GET_MENU_ITEM_PROPERTY for valid property.

    **Level:** 20

    **Trigger:** ON-ERROR

**FRM-41072: Cannot create Group %s**

    **Cause:** Caused by one of the following

    1. Duplicate column names in SQL statement.

    2. Invalid record group name.

    3. Query is invalid.

    **Action:** Check the group name and/or correct the SQL statement.

    **Level:** 20

    **Trigger:** ON-ERROR

**FRM-41073: Cannot find Group: invalid ID.**

    **Cause:** An invalid ID was passed to a Built-in subprogram.

    **Action:** Verify that a proper call to FIND_GROUP will be performed.

    **Level:** 20

    **Trigger:** ON-ERROR

**FRM-41074: Cannot find Group or Column: invalid ID.**

    **Cause:** An invalid ID was passed to a Built-in subprogram.

    **Action:** Verify that a proper call to FIND_GROUP or FIND_COLUMN will be performed.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41075: Error deleting Group.**

**Cause:** The record group name or ID specified in the call to DELETE_GROUP is invalid, or the record group was not dynamically created.

**Action:** Check the record group name or ID, and make sure that the specified record group was dynamically created.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41076: Error populating Group.**

**Cause:** Query failed due to an invalid column or table name, or the query and group column structure do not match.

**Action:** Check the SQL SELECT statement in your call to POPULATE_GROUP_ WITH_QUERY.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41077: Error deleting Group Row(s).**

**Cause:** DELETE_GROUP_ROW cannot be used to delete records from a static record group, or you specified an invalid row number.

**Action:** Correct the call to DELETE_GROUP_ROW.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41078: Error resetting Group selection.**

**Cause:** Record group name or ID specified is invalid.

**Action:** Check the record group name or ID and try again.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41079: Error adding Group column.**

**Cause:** Caused by one of the following:

1. You cannot add columns to a group that already has rows.

2. The width of CHAR_COLUMN-typed columns cannot be less than the width of the corresponding database column.

3. You entered the name of a nonexistent or invalid record group.

4. You entered the name of a nonexistent or invalid column.

5. You entered a column type other than CHAR, NUMBER, or DATE.

**Action:** You can only add columns to a group after it is created with a call to CREATE_GROUP. If the group already has rows, delete the rows with DELETE_ GROUP_ROW, then add the column.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41080: Error adding Group row.**

    **Cause:** Caused by one of the following:

    1. You attempted to add rows to a group that is nonexistent or has no columns.

    2. You entered the name of a nonexistent record group.

    3. You provided a row number that is out of range or invalid.

    **Action:** Create the group and add columns first. Check the call to ADD_GROUP_ ROW to make sure that the record group name and row number are valid.

    **Level:** 20

    **Trigger:** ON-ERROR

**FRM-41081: Cannot move Item: invalid position.**

    **Cause:** You attempted to move the item to an invalid position on the canvas.

    **Action:** Make sure the coordinates you chose in your call to SET_ITEM_ PROPERTY are valid.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41082: Cannot resize item: position of item places it off of canvas.**

    **Cause:** The height and/or width you specified in your call to SET_ITEM_ PROPERTY is invalid, or the height and/or width you specified causes the item to extend off of the canvas.

    **Action:** Correct the call to SET_ITEM_PROPERTY.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41083: No such property for Set_Form_Property**

    **Cause:** You attempted to set a nonexistent form property.

    **Action:** Verify call to SET_FORM_PROPERTY for valid property.

    **Level:** 20

    **Trigger:** ON-ERROR

**FRM-41084: Error getting Group Cell.**

    **Cause:** Invalid call to GET_GROUP_CHAR_CELL, GET_GROUP_DATE_CELL, OR GET_GROUP_NUMBER_CELL.

    **Action:** Make sure the column type is of CHAR, DATE, or NUMBER, respectively. Check the validity of the row number and column name specified.

    **Level:** 20

    **Trigger:** ON-ERROR

**FRM-41085: Error getting Group Row count.**

    **Cause:** Invalid call to GET_GROUP_ROW_COUNT.

    **Action:** Check the record group name and try again.

    **Level:** 20

    **Trigger:** ON-ERROR

**FRM-41086: Error getting Group selection count.**

**Cause:** You specified an invalid record group name. Invalid call to GET_GROUP_ SELECTION_COUNT.

**Action:** Correct the call to GET_GROUP_SELECTION.

**Level:** 20

**Trigger:** ON-ERROR

### FRM-41087: Error getting Group selection.
**Cause:** You specified an invalid record group name or selection number. Invalid call to GET_GROUP_SELECTION.

**Action:** Correct the call to GET_GROUP_SELECTION.

**Level:** 20

**Trigger:** ON-ERROR

### FRM-41088: Cannot set Group selection.
**Cause:** You specified an invalid record group name, ID, or row number.

**Action:** Correct the call to SET_GROUP_SELECTION.

**Level:** 20

**Trigger:** ON-ERROR

### FRM-41089: Cannot move View: invalid position.
**Cause:** The x, y pair specified in the call to SET_VIEW_PROPERTY is invalid.

**Action:** Correct the call to SET_VIEW_PROPERTY by making sure that the position specified by your coordinates is on the canvas.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-41090: Invalid item type for go_item: %s.
**Cause:** You cannot navigate to the item.

**Action:** Check to make sure the item is a navigable item.

**Level:** 20

**Trigger:** ON-ERROR

### FRM-41091: Cannot find LOV: invalid ID.
**Cause:** An invalid ID was passed to a Built-in subprogram.

**Action:** Verify that a proper call to FIND_LOV will be performed.

**Level:** 20

**Trigger:** ON-ERROR

### FRM-41092: No records in block %s.
**Cause:** You attempted to place a value into an item on a block that has no records.

**Action:** Put records in the block first.

**Level:** 20

**Trigger:** ON-ERROR

### FRM-41093: Error setting item property: %s.

**Cause:** You specified Lock Record and the item was not a text item, or you specify Case Insensitive Query and the data type was not ALPHA or CHAR.

**Action:** In the case of Lock Record, make sure that the item is a text item. When specifying Case Insensitive Query, make sure that the data type is ALPHA or CHAR.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41094: No such property for Get_View_Property.**

**Cause:** You attempted to get a non-existent view property.

**Action:** Verify call to GET_VIEW_PROPERTY for valid property.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41095: No such property for Get_Canvas_Property.**

**Cause:** You attempted to get a non-existent canvas property.

**Action:** Verify call to GET_CANVAS_PROPERTY for valid property.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41096: Cannot resize View: invalid size.**

**Cause:** The x, y coordinates place the view off the canvas.

**Action:** Choose another x, y pair.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41097: Cannot resize Canvas: invalid size.**

**Cause:** The x, y coordinates place the view off the window.

**Action:** Choose another x, y pair.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41098: Cannot modify Display Position of a content view.**

**Cause:** The Display Position Property applies to a stacked canvas-view only.

**Action:** Correct the call to SET_VIEW_PROPERTY.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41099: Cannot modify Size of a content view.**

**Cause:** The size of a content view is dependent on window size. Only stacked view sizes may be modified using SET_VIEW_PROPERTY.

**Action:** Correct the call to SET_VIEW_PROPERTY.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41100: Cannot find relation %s.**

**Cause:** You attempted to get, set, or find using an invalid relation.

**Action:** Check call to Built-in for correct arguments.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-41101: No such property for Get_Relation_Property.

**Cause:** You attempted to get a non-existent relation property.

**Action:** Verify call to GET_RELATION_PROPERTY for valid property.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-41102: No such property for Set_Relation_Property.

**Cause:** You attempted to set a non-existent relation property.

**Action:** Verify call to SET_RELATION_PROPERTY for valid property.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-41103: No such property value for Set_Relation_Property.

**Cause:** Application design error. Improper relation property value passed to SET_RELATION_PROPERTY Built-in.

**Action:** Correct call to SET_RELATION_PROPERTY Built-in and retry.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-41104: Cannot find Relation: invalid ID.

**Cause:** An invalid ID was passed to a Built-in subprogram.

**Action:** Verify that a proper call to FIND_RELATION will be performed.

**Level:** 20

**Trigger:** ON-ERROR

### FRM-41105: You cannot query records without a saved parent record.

**Cause:** You attempted to query detail records without first creating a master record.

**Action:** Create a master record, and then query the detail records.

**Level:** 10

**Trigger:** ON-ERROR

### FRM-41106: You cannot create records without a parent record.

**Cause:** You attempted to create new detail records without first creating a master record.

**Action:** Create a master record, and then add the detail records.

**Level:** 10

**Trigger:** ON-ERROR

### FRM-41107: Master delete option for the relation is invalid.

**Cause:** An invalid query data source type or an invalid DML data target type is specified for the detail block.

**Action:** Verify that the detail block's query data source and the DML data targets are of type table.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41200: Integration error: invalid product.**

**Cause:** Invalid product name specified during integration.

**Action:** Check the integration parameters.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41201: Integration error: communication mode must be SYNCHRONOUS or ASYNCHRONOUS.**

**Cause:** Invalid communication mode specified in RUN_REPORT_OBJECT.

**Action:** Check the RUN_REPORT_OBJECT parameters and try again.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41202: Integration error: parameter list %s has no parameters.**

**Cause:** Parameter list has no arguments.

**Action:** Check the specified parameter list for parameters.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41203: Integration error: invalid parameter list ID.**

**Cause:** An invalid parameter list ID was passed.

**Action:** Check the parameter list ID name and try again.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41204: Integration error: memory allocation error.**

**Cause:** An internal error occurred.

**Action:** If the problem persists, contact Oracle Support Services.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41208: Integration error: execution mode must be BATCH or RUNTIME.**

**Cause:** Invalid execution mode specified in RUN_REPORT_OBJECT.

**Action:** Specify either BATCH or RUNTIME for the execmode parameter.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41209: Integration error: document location must be FILESYSTEM or DB.**

**Cause:** Invalid document location specified while integrating with another product.

**Action:** Specify either FILESYSTEM or DB for the location parameter.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41211: Integration error: SSL failure running another product.**

**Cause:** There is a problem detected when launching another product.

**Action:** Check the RUN_REPORT_OBJECT Built-in.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41212: Integration error: invalid communication mode for data exchange**

**Cause:** User specified an asynchronous RUN_REPORT_OBJECT.

**Action:** Change to a synchronous RUN_REPORT_OBJECT communication mode.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41213: Unable to connect to the Report server %s.**

**Cause:** There is a problem connecting to the specified Report server.

**Action:** Check the Report server and make sure it is up and running.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41214: Unable to run report.**

**Cause:** The report server was unable to run the specified report.

**Action:** Check the Report server and make sure it is up and running.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41215: Invalid server name or jobid.**

**Cause:** There is a problem decoding the return value from the Built-in run_report.

**Action:** The return value from the Built-in run_report should not be modified before being passed to another report Built-in.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41216: Unable to cancel job.**

**Cause:** There is a problem cancelling a report job.

**Action:** Check the Report server and make sure that the specified job exists.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41217: Unable to get report job status.**

**Cause:** There is a problem getting report status for a given report job.

**Action:** Check the Report server and make sure that the specified job exists.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41218: Unable to copy report output.**

**Cause:** There is a problem copying report output for a given report job.

**Action:** Check the Report server and make sure that the specified output file exists.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41219: Cannot find report: invalid ID.**

**Cause:** The user has specified an invalid report object name.

**Action:** Check the form and make sure that the report object exists.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41220: Failed to authenticate user.**

**Cause:** There was a failure in displaying the web report.

**Action:** Check if the user credentials are valid against identity store in use.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41221: Failed to connect to identity store.**

**Cause:** There was a failure in connecting to authentication service of identity store.

**Action:** Check if the authentication service is running.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41300: Invalid parameter used for Set_Radio_Button_Property.**

**Cause:** You specified a parameter that does not exist.

**Action:** Check the list of legal parameters.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41301: Invalid parameter used for Set_View_Property.**

**Cause:** You specified a parameter that does not exist.

**Action:** Check the list of legal parameters.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41302: Invalid parameter used for Set_Canvas_Property.**

**Cause:** You specified a parameter that does not exist.

**Action:** Check the list of legal parameters.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41303: No such property for Set_Window_Property.**
    **Cause:** You specified a property that does not exist.
    **Action:** Check the list of legal properties.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41304: No such property for Set_Block_Property.**
    **Cause:** You specified a property that does not exist.
    **Action:** Check the list of legal properties.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41305: No such property for Get_Block_Property.**
    **Cause:** You specified a property that does not exist.
    **Action:** Check the list of legal properties.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41306: Invalid parameter used for Set_Window_Property.**
    **Cause:** You specified a parameter that is not valid.
    **Action:** Check the list of valid parameters.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41307: Invalid parameter used for Set_Block_Property.**
    **Cause:** You specified a parameter that is not valid.
    **Action:** Check the list of valid parameters.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41308: Error unsetting Group selection.**
    **Cause:** You tried to deselect a record or a subset of records that was not selected or is not in the record group.
    **Action:** Check the records that are expected in the group.

    **Level:** 20

    **Trigger:** ON-ERROR

**FRM-41309: No such property for Get_Radio_Button_Property.**
    **Cause:** You specified a property that is invalid.
    **Action:** Check the list of valid properties.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41310: No such property for Set_Radio_Button_Property.**

**Cause:** You specified a property that is invalid.

**Action:** Check the list of valid properties.

**Level:** 99

**Trigger:** ON-ERROR

## FRM-41311: Invalid argument or argument ordering for %s.

**Cause:** You supplied an incorrect argument list.

**Action:** Check the list of valid arguments.

**Level:** 99

**Trigger:** ON-ERROR

## FRM-41312: Must have at least one writable item in block.

**Cause:** A block with the Insert Allowed Property or Update Allowed Property set to True must have at least one writable item. You attempted to make the only remaining base table item in the block not writable by setting either the Derived Column Property or Query Only Property to True.

**Action:** Set Insert Allowed or Update Allowed to False for the block, rather than setting Derived Column or Query Allowed to False for each item.

**Level:** 10

**Trigger:** ON-ERROR

## FRM-41313: No such property for Set_Alert_Property.

**Cause:** An invalid property has been specified for SET_ALERT_PROPERTY.

**Action:** Enter a valid alert property.

**Level:** 99

**Trigger:** ON-ERROR

## FRM-41314: Cannot set Insert Allowed Property of current item %s.%s

**Cause:** You attempted to set the Insert Allowed Property for a current item.

**Action:** The Insert Allowed Property is only valid on non-current items. Make sure the item is not current.

**Level:** 99

**Trigger:** ON-ERROR

## FRM-41315: Cannot set Insert Allowed Property of non-displayed item %s.%s

**Cause:** You tried to set Insert Allowed Property for a non-displayed item.

**Action:** The Insert Allowed Property is only valid on displayed items. Make sure the item is displayed.

**Level:** 99

**Trigger:** ON-ERROR

## FRM-41316: Cannot set Insert Allowed Property of disabled item %s.%s

**Cause:** You tried to set Insert Allowed Property for a disabled item.

**Action:** The Insert Allowed Property is only valid on enabled items. Make sure the item is enabled.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41317: Item is not a radio button %s**

    **Cause:** You tried to use a radio button Built-in with an item that is not a radio button.

    **Action:** Make sure the item is a radio button.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41318: Item %s is not a VBX item.**

    **Cause:** You tried to use a VBX Built-in with an item that is not a VBX item.

    **Action:** Make sure the item is a VBX item.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41319: Invalid property %s specified for VBX item %s.**

    **Cause:** You tried to get or set an invalid property for the specified VBX item.

    **Action:** Make sure the property is valid for the specified VBX item.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41320: Unable to get property %s for VBX item %s.**

    **Cause:** Could not get the valid property for the VBX item.

    **Action:** Check the list of legal properties.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41321: Unable to set property %s for VBX item %s.**

    **Cause:** Could not set the valid property for the VBX item.

    **Action:** Check the list of legal properties.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41322: Invalid event %s for VBX item %s.**

    **Cause:** You tried to get or set an invalid event for the specified VBX item.

    **Action:** Make sure the event is valid for the specified VBX item.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41323: Too many parameters for event %s for VBX item %s.**

    **Cause:** You specified too many parameters for the event name for the VBX item.

    **Action:** Make sure there is a valid number of parameters for the event.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41324: Too few parameters for event %s for VBX item %s.**

**Cause:** You specified too few parameters for the event name for the VBX item.

**Action:** Make sure there is a valid number of parameters for the event.

**Level:** 99

**Trigger:** ON-ERROR

## FRM-41325: VBX event parameter must be a string.

**Cause:** The VBX event parameter is not a string.

**Action:** Make sure the VBX event parameter is a string.

**Level:** 99

**Trigger:** ON-ERROR

## FRM-41326: Failed to deliver event %s to VBX item %s.

**Cause:** The VBX event failed.

**Action:** Make sure the event is valid for the specified VBX item.

**Level:** 99

**Trigger:** ON-ERROR

## FRM-41327: Failed to get default property for VBX item %s.

**Cause:** The VBX.GET_VALUE_PROPERTY Built-in failed.

**Action:** Make sure an initial value is assigned to the VB Control Value property.

**Level:** 99

**Trigger:** ON-ERROR

## FRM-41328: Failed to set default property for VBX item %s.

**Cause:** The VBX.SET_VALUE_PROPERTY Built-in failed.

**Action:** Make sure you are setting a valid value property.

**Level:** 99

**Trigger:** ON-ERROR

## FRM-41329: Item %s is not a List item.

**Cause:** You tried to add a list element to an item that is not a list.

**Action:** Make sure the item is a List item.

**Level:** 99

**Trigger:** ON-ERROR

## FRM-41330: Could not insert list element into %s.

**Cause:** You tried to insert an other values element when the block contained either queried or changed records.

**Action:** For more information, refer to help for restrictions on <a href="../builta_c/addliste.html">ADD_LIST_ELEMENT Built-in</a>.

**Level:** 99

**Trigger:** ON-ERROR

## FRM-41331: Could not delete element from %s.

**Cause:** Caused by one of the following:

You tried to delete the other values element when the block contained either queried or changed records.

You tried to delete an element from a list that does not contain an other values element when the block contained either queried or changed records.

**Action:** For more information, refer to help for restrictions on <a href="../builta_ c/clearlis.html">CLEAR_LIST</a> and <a href="../builtd_ f/dellsele.html">DELETE_LIST_ELEMENT</a>.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41332: List element index out of range.**
**Cause:** An invalid index (e.g. a negative number) was specified to the Add_List_ Element Built-in.
**Action:** Correct the index in the call to Add_List_Element.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41333: Cannot convert list element value.**
**Cause:** Could not resolve list element value to a string.
**Action:** Make sure list element is a string.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41334: Invalid record group for list population.**
**Cause:** You tried to populate a list from a record group that does not exist.
**Action:** Make sure the record group exists.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41335: Populate_List: invalid column type for column 1.**
**Cause:** The record group does not have a column of the same type.
**Action:** Make sure record group has a column of the same type.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41336: Populate_List: invalid column type for column 2.**
**Cause:** The record group does not have a column of the same type.
**Action:** Make sure record group has a column of the same type.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41337: Cannot populate the list from record group.**
**Cause:** The record group is invalid or the list item does not satisfy the requirements for deleting and adding elements.
**Action:** Make sure the record group is valid. For more information about deleting and adding list elements, refer to help for restrictions on <a href="../builtd_

f/dellsele.html">DELETE_LIST_ELEMENT</a> and <a href="../builta_
c/addliste.html">ADD_LIST_ELEMENT</a>.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41338: Cannot retrieve the list into record group.**

**Cause:** The record group is invalid.

**Action:** Make sure the record group is valid.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41339: Cannot clear the list.**

**Cause:** A memory allocation failed when Forms Runtime attempted to clear a list.

**Action:** Try executing the application when the system is less heavily loaded. If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41340: No such property or value for Set_Application_Property.**

**Cause:** You specified an invalid property and/or an invalid value for a property.

**Action:** Specify a valid property and/or a valid value.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41341: Invalid cursor shape %s specified.**

**Cause:** There is a predefined set of cursor types, and an invalid cursor type was specified.

**Action:** Specify a valid cursor type.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41342: Invalid parameter %s specified for VBX event %s.**

**Cause:** You specified an invalid parameter for a VBX event.

**Action:** Check the parameter type.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41343: Item %s is not an OLE object.**

**Cause:** Invalid item passed to OLE Built-in.

**Action:** Specify a valid OLE item.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41344: OLE object not defined for %s in the current record.**

**Cause:** An empty OLE container is defined.

**Action:** Define an OLE object to reside in the OLE container.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41345: Cannot find the verb %s for this server.**
    **Cause:** You specified an invalid OLE verb.
    **Action:** Specify a valid OLE verb.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41346: Cannot determine the verb count for OLE object %s.**
    **Cause:** Could not communicate with OLE server.
    **Action:** Re-install the OLE server.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41347: Invalid verb index for OLE object %s.**
    **Cause:** You provided an index that is greater than the verb count.
    **Action:** Check the index value.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41348: OLE server error: %s.**
    **Cause:** OLE server detects an error.
    **Action:** Try to resolve the error based on the message from the OLE server.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41349: OLE object %s cannot execute verb; verb id %d**
    **Cause:** OLE object does not recognize the verb.
    **Action:** Try to execute another verb.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41350: OLE object is currently not displayed.**
    **Cause:** You tried to close a server that is not running.
    **Action:** Ask if the server is active in a record that is not currently active.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41351: Cannot navigate out of current form.**
    **Cause:** You cannot navigate to an inactive form.
    **Action:** Check to make sure the form you are navigating to is active.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41352: Failed to create a new session.**

**Cause:** You attempted to open a new form with a new session.

**Action:** Check the database server.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-41353: Cannot start another call form.

**Cause:** You went to a peer form and performed a call form.

**Action:** Make sure you are not at a peer form when calling the form.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-41354: Cannot close form %s.

**Cause:** Unsuccessful attempt to close a form.

**Action:** Make sure the form is open.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-41355: Cannot navigate to form %s.

**Cause:** You cannot navigate to an inactive form.

**Action:** Check to make sure you are navigating to an active form.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-41356: Invalid method %s for VBX item %s.

**Cause:** You specified an invalid method name for the VBX item.

**Action:** Specify a valid method name for the VBX item.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-41357: Incorrect number of arguments to method %s for VBX item %s.

**Cause:** You specified an incorrect number of arguments to the method for the VBX item.

**Action:** Make sure the number of arguments is what the VBX item expects.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-41358: Method %s failed for VBX item %s.

**Cause:** You specified an invalid method name for the VBX item.

**Action:** Specify a valid method name for the VBX item.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-41359: The Open_Form session feature is not enabled. Cannot create new session.

**Cause:** You do not have the multiple sessioning feature enabled on the database.

**Action:** The Open_Form session feature is only available for use against a database with multiple sessioning enabled.

**Level:** 99

**Trigger:** ON-ERROR

## FRM-41360: Invalid value used in Set_Window_Property for window %s.

**Cause:** You are using an invalid value when attempting to set a window property.

**Action:** Specify a valid window property value.

**Level:** 99

**Trigger:** ON-ERROR

## FRM-41361: Cannot navigate out of current form in Enter-Query mode.

**Cause:** You are in Enter-Query mode and trying to navigate to another form when using Open Form.

**Action:** Exit Enter-Query mode and try again.

**Level:** 10

**Trigger:** ON-ERROR

## FRM-41362: No such property for Set_Alert_Button_Property.

**Cause:** You specified an invalid property for Set_Alert_Button_Property.

**Action:** Specify a valid property for Set_Alert_Button_Property.

**Level:** 99

**Trigger:** ON-ERROR

## FRM-41363: No such property for Set_LOV_Column_Property.

**Cause:** You specified an invalid property for Set_LOV_Column_Property.

**Action:** Specify a valid property for Set_LOV_Column_Property.

**Level:** 99

**Trigger:** ON-ERROR

## FRM-41364: Invalid column number specified for LOV %s.

**Cause:** You specified an invalid column number for the LOV.

**Action:** Specify a valid column number for the LOV.

**Level:** 99

**Trigger:** ON-ERROR

## FRM-41365: No such property for Set_TabPage_Property.

**Cause:** You specified an invalid property for Set_TabPage_Property.

**Action:** Specify a valid property for Set_TabPage_Property.

**Level:** 99

**Trigger:** ON-ERROR

## FRM-41366: No such property for Get_TabPage_Property.

**Cause:** You specified an invalid property parameter.

**Action:** Check the list of valid properties.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41367: Cannot find TabPage: invalid ID.**

    **Cause:** An invalid ID was passed to a Built-in subprogram.

    **Action:** Verify that a proper call to FIND_TABPAGE will be performed.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41368: Invalid parameter used for Set_TabPage_Property.**

    **Cause:** You specified a parameter that is not valid.

    **Action:** Check the list of valid parameters.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41369: Cannot insert a second record into a single-record block.**

    **Cause:** You (or the application) have attempted to insert a second record into a block whose Single Record Property is TRUE.

    **Action:** Don't attempt to insert a record into such a block.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41370: Cannot modify calculated item %s.%s.**

    **Cause:** Application design error. The application attempted to assign a value to a calculated item.

    **Action:** If the calculated item is a formula item, then its formula determines its value at all times. It may be appropriate to modify the formula. Or it may be appropriate to change the calculated item to a non-calculated control item whose value is set in various triggers.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41371: Cannot set INSERT_ALLOWED Property of calculated item %s.%s.**

    **Cause:** Application design error. A SET_ITEM_PROPERTY or SET_ITEM_ INSTANCE_PROPERTY Built-in attempted to set a calculated item's INSERT_ ALLOWED Property to TRUE.

    **Action:** The call to the Built-in must be modified or removed.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41372: Cannot set ITEM_IS_VALID Property of calculated item %s.%s.**

    **Cause:** Application design error. A SET_ITEM_PROPERTY Built-in attempted to set a calculated item's ITEM_IS_VALID Property to FALSE.

    **Action:** The call to the Built-in must be modified or removed.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41373: Cannot set LOCK_RECORD Property of calculated item %s.%s.**

**Cause:** Application design error. A SET_ITEM_PROPERTY Built-in attempted to set a calculated item's LOCK_RECORD Property to TRUE.

**Action:** The call to the Built-in must be modified or removed.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41374: Cannot set PRIMARY_KEY Property of calculated item %s.%s.**

**Cause:** Application design error. A SET_ITEM_PROPERTY Built-in attempted to set a calculated item's PRIMARY_KEY Property to TRUE.

**Action:** The call to the Built-in must be modified or removed.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41375: Cannot set QUERYABLE Property of calculated item %s.%s.**

**Cause:** Application design error. A SET_ITEM_PROPERTY Built-in attempted to set a calculated item's QUERYABLE Property to TRUE.

**Action:** The call to the Built-in must be modified or removed.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41376: Cannot set REQUIRED Property of calculated item %s.%s.**

**Cause:** Application design error. A SET_ITEM_PROPERTY or SET_ITEM_ INSTANCE_PROPERTY Built-in attempted to set a calculated item's REQUIRED Property to TRUE.

**Action:** The call to the Built-in must be modified or removed.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41377: Cannot set UPDATEABLE Property of calculated item %s.%s.**

**Cause:** Application design error. A SET_ITEM_PROPERTY or SET_ITEM_ INSTANCE_PROPERTY Built-in attempted to set a calculated item's UPDATEABLE Property to TRUE.

**Action:** The call to the Built-in must be modified or removed.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41378: Cannot set UPDATE_NULL Property of calculated item %s.%s.**

**Cause:** Application design error. A SET_ITEM_PROPERTY Built-in attempted to set a calculated item's UPDATE_NULL Property to TRUE.

**Action:** The call to the Built-in must be modified or removed.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41379: Cannot recalculate non-formula item %s.%s.**

**Cause:** Application design error. A RECALCULATE Built-in specified an item which is not a formula item.

**Action:** The call to the Built-in must be modified or removed.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41380: Cannot set the blocks query data source.**

**Cause:** The user attempt to change the block's data source dynamically has failed.

**Action:** Check the form and make sure that the specified block is not a control block and the block status is new.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41381: Cannot set the blocks DML data target source.**

**Cause:** The user attempt to change the block's DML data target dynamically has failed.

**Action:** Check the form and make sure that the specified block is not a control block and the block status is new.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41382: No such property for Get_Item_Instance_Property.**

**Cause:** Application design error. A GET_ITEM_INSTANCE_PROPERTY Built-in specified an invalid property.

**Action:** The property must be changed to BORDER_BEVEL, INSERT_ALLOWED, NAVIGABLE, REQUIRED, UPDATEABLE, or VISUAL_ATTRIBUTE, or else the call to the Built-in must be removed.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41383: No such property for Set_Item_Instance_Property.**

**Cause:** Application design error. A SET_ITEM_INSTANCE_PROPERTY Built-in specified an invalid property.

**Action:** The property must be changed to BORDER_BEVEL, INSERT_ALLOWED, NAVIGABLE, REQUIRED, UPDATEABLE, or VISUAL_ATTRIBUTE, or else the call to the Built-in must be removed.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41384: Invalid parameter used for Set_Item_Instance_Property.**

**Cause:** Application design error. A SET_ITEM_INSTANCE_PROPERTY Built-in specified an invalid value for a property.

**Action:** The call to the Built-in must be modified or removed.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41385: Maximum number of queried records exceeded.**

**Cause:** The user specified maximum number of records for a given block is reached.

**Action:** Check the forms block and form level properties.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-41386: Cannot set VISIBLE Property of tab page containing current item.

**Cause:** You tried to set the Visible Property for the tab page which contains the current item.

**Action:** The Visible Property is only valid for tab pages which don't contain the current item.

Navigate to an item on a different tab page or different canvas first.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-41387: Cannot set VISIBLE Property of last enterable tab page.

**Cause:** You tried to set the Visible Property for the only enterable tab page on the canvas.

**Action:** Make sure there is at least one other enterable tab page on the canvas before trying to set the Visible Property.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-41388: Cannot set ENABLED Property of tab page containing current item.

**Cause:** You tried to set the Enabled Property for the tab page which contains the current item.

**Action:** The property is only valid for tab pages which don't contain the current item.

Navigate to an item on a different tab page or different canvas first.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-41389: Cannot set ENABLED Property of last enterable tab page.

**Cause:** You tried to set the Enabled Property for the only enterable tab page on the canvas.

**Action:** Make sure there is at least one other enterable tab page on the canvas before trying to set the Enabled Property.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-41390: Cannot set REQUIRED Property of subordinate mirror item %s.%s.

**Cause:** Application design error. A SET_ITEM_PROPERTY or SET_ITEM_ INSTANCE_PROPERTY Built-in attempted to set the Required Property of a subordinate mirror item. The Required Property will be obtained from the master mirror item (the item specified by the Synchronize With Item Property).

**Action:** Set the Required Property of the master mirror item.

**Level:** 25

**Trigger:** ON-ERROR

**FRM-41391: Cannot find visual attribute: invalid ID.**

    **Cause:** An invalid ID was passed to a Built-in subprogram.

    **Action:** Verify that a proper call to Find_VA will be performed.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41392: No such property for Get_VA_Property.**

    **Cause:** You attempted to get a non-existent visual attribute property.

    **Action:** Verify call to Get_VA_Property for a valid property.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41393: No such property for Set_VA_Property.**

    **Cause:** You attempted to set an invalid visual attribute property.

    **Action:** Check the documentation for setting visual attribute properties and try again.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41394: Invalid parameter value used for Set_VA_Property.**

    **Cause:** You attempted to set an invalid value for a visual attribute property.

    **Action:** Check the documentation for setting visual attribute properties and try again.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41395: Invalid parameter used for Set_Report_Object_Property.**

    **Cause:** You specified a parameter that does not exist.

    **Action:** Check the list of legal parameters.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41396: No such property for Get/Set_Report_Object_Property.**

    **Cause:** You specified a property that does not exist.

    **Action:** Check the list of legal properties.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41402: Invalid type of visual attribute passed to Set_<object>_Property.**

    **Cause:** You attempted to set an object's visual attribute to a VA of the wrong type.

    **Action:** Verify VA types in the Builder and specify a valid VA for this object.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41403: Cannot set DEFAULT_WHERE: invalid value.**

    **Cause:** The user attempted to set the DEFAULT_WHERE to an invalid value.

**Action:** Check the value you chose in your call to SET_BLOCK_PROPERTY is valid.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-41411: SELECTED_RADIO_BUTTON property allowed only on a radio group.

**Cause:** The user attempted to obtain the SELECTED_RADIO_BUTTON property for an item which is not a rasio group.

**Action:** Check the value that was specified for the item.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-41412: Cannot set scrollbar position for specified block.

**Cause:** The user attempted to set a scrollbar position property for a block which has no scrollbar.

**Action:** Check the value that was specified for the block.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-41413: Cannot get scrollbar position for specified block.

**Cause:** The user attempted to get a scrollbar position property for a block which has no scrollbar.

**Action:** Check the value that was specified for the block.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-41414: Combo box item element %s is longer than Maximum Length.

**Cause:** The label for combo box item element %s is longer than Maximum Length.

**Action:** Reduce the number of characters in the element's label.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-41800: List of Values not available for this field.

**Cause:** You pressed [List], but the form does not provide a list of values for this field.

**Action:** No action is necessary.

**Level:** 10

**Trigger:** ON-MESSAGE

### FRM-41801: Last value retrieved.

**Cause:** You pressed [List] and then pressed [Next Item] after the last value in the list was displayed.

**Action:** Enter an item value or press [List] again to display the list of possible values.

**Level:** 10

**Trigger:** ON-MESSAGE

**FRM-41802: Duplicate record function allowed on new records only.**

**Cause:** You pressed [Duplicate Record], but the current record is the one that has been fetched from the database.

**Action:** No action is necessary. You can use [Duplicate Record] only when creating a new record.

**Level:** 10

**Trigger:** ON-ERROR

**FRM-41803: No previous record to copy value from.**

**Cause:** You pressed [Duplicate Item] or [Duplicate Record], but the current record is the first record in the block.

**Action:** No action is necessary. [Duplicate Item] and [Duplicate Record] are meaningless in this context.

**Level:** 10

**Trigger:** ON-ERROR

**FRM-41804: Variable was not entered: %.30s.**

**Cause:** Your response to the Query Where alert contained a placeholder not used in any of the query items.

**Action:** Correct the placeholder in your response, or define it in one of the query items. Then re-execute the query.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41805: Ambiguous item name: %s.**

**Cause:** Application design error. A call to a Built-in specified an ambiguous item name. (No block was specified, and more than one block contains an item of the specified name).

**Action:** Specify a block name (block.item).

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41806: Too many variables used.**

**Cause:** You used more than 25 substitution variables in your query.

**Action:** Reduce the number of substitution variables and re-query.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41809: Error initializing Menu.**

**Cause:** You tried to use the menu component from within Oracle Forms, and an internal error occurred.

**Action:** If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41810: Error creating menu.**

**Cause:** You tried to use the menu component from within Oracle Forms, and an internal error occurred.

**Action:** If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41811: Error removing menu.**

**Cause:** You tried to use Menus from within Oracle Forms, and an internal Menu error occurred.

**Action:** If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41812: Error resetting Menu.**

**Cause:** You tried to use the menu component from within Oracle Forms, and an internal error occurred.

**Action:** If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41813: Form exited by debug mode.**

**Cause:** You selected the Exit Oracle Forms Runtime option on the Break Processing menu.

**Action:** No action is necessary.

**Level:** 99

**Trigger:** None

**FRM-41814: Invalid page position.**

**Cause:** Application design error. A trigger tried to move or resize a view to a page that would cause all or part of the view to display off of the screen.

**Action:** Correct the statement.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41815: No such property for Get_LOV_Property.**

**Cause:** You attempted to get a nonexistent LOV property.

**Action:** Verify the valid LOV properties and try again.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41816: Attempt to create existing timer: %s.**

**Cause:** Attempted to create a timer that already exists.

**Action:** Delete or alter the existing timer before re-creating a new one.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41817: No such timer: %s.**

    **Cause:** You attempted to alter or delete a non-existent timer.

    **Action:** Check the Built-in for proper arguments.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41818: Toolkit failed to create timer %s :may be out of memory.**

    **Cause:** An internal error occurred while attempting to create a timer, possibly as a result of memory constraints.

    **Action:** Check and adjust memory quotas as necessary.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41819: Timers are not supported on this platform.**

    **Cause:** Illegal attempt to create a timer on a platform where timers are not supported.

    **Action:** None. A timer option is unavailable on your platform.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41820: Toolkit failed to delete timer: %s.**

    **Cause:** Internal error caused by timer failure.

    **Action:** If the problem persists, contact Oracle Support Services.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41821: Timer name too long: %s...**

    **Cause:** You attempted to create a timer with a name longer than 30 bytes.

    **Action:** Retry with a shorter name.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41822: Timer name may not be null string.**

    **Cause:** You attempted to create a timer with a null name.

    **Action:** Retry with a non-null name.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41823: Illegal timer interval for timer %s.**

    **Cause:** You attempted to create a timer with an interval less than 1 millisecond.

    **Action:** Recreate your timer with an interval of at least 1 millisecond.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-41824: Date/time operation failed for %s.**

**Cause:** An internal error occurred while attempting to resolve a date/time initial value for an item.

**Action:** If the problem persists, contact Oracle Support Services.

**Level:** 10

**Trigger:** ON-ERROR

**FRM-41825: No such property for Set_LOV_Property.**

**Cause:** You attempted to set a nonexistent LOV property.

**Action:** Verify the valid LOV properties and try again.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41826: Cannot replace group; columns don't match LOV.**

**Cause:** Cannot replace the list of values' current record group with a record group that is incompatible with the LOV column structure.

**Action:** Do not attempt to assign this record group to this LOV.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41827: Group does not exist.**

**Cause:** The group name or ID specified is invalid.

**Action:** Check the name or ID entered and try again.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41828: LOV does not exist.**

**Cause:** LOV name or ID specified is invalid.

**Action:** Check the name or ID entered and try again.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41829: Record not created.**

**Cause:** Application design error. The record failed to get its initial value.

**Action:** Contact the application designer.

**Level:** 20

**Trigger:** ON-ERROR

**FRM-41830: List of Values contains no entries.**

**Cause:** The record group underlying the LOV contains no records.

**Action:** Check to be sure that any criteria used to reduce a long list LOV did not eliminate all matches.

**Level:** 5

**Trigger:** ON-ERROR

**FRM-41832: Error: program unit %s in library %s is uncompiled.**

**Cause:** You called an uncompiled program unit from a library.

**Action:** Follow the PL/SQL program error.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41833: Warning! Program unit %s in library %s is uncompiled.**

**Cause:** You called an uncompiled program unit in a library when debug mode was specified.

**Action:** This is just a warning. Forms Runtime will attempt to compile and run the program unit.

**Level:** 99

**Trigger:** None

**FRM-41835: Canvas %s is not a tab canvas.**

**Cause:** You tried to perform a tab canvas specific operation on a canvas which is not a tab canvas.

**Action:** Make sure the canvas specified is a tab canvas.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41836: No tab page %s in canvas %s.**

**Cause:** You tried to perform an operation on a tab page which does not exist in the specified canvas.

**Action:** Make sure you specify a tab page which exists in the specified tab canvas.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41837: Error raising tab page %s.**

**Cause:** The specified tab page could not be brought to the top (made the current page of the tab canvas).

**Action:** Make sure the specified page is enabled, and not hidden.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41838: Unable to open temporary record buffer file %s**

**Cause:** Unable to open file used as temporary record buffer.

**Action:** Verify that the file system or directory in which the file resides exists and that you have permissions to read and write to it.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41839: Disk I/O error on temporary record buffer file %s**

**Cause:** An I/O error occurred on attempting to read or write a record to the temporary record buffer file.

**Action:** Verify that the file system or directory in which the file resides exists, and that you have permissions to read and write to it, and that it contains sufficient space.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41840: Insufficient main memory for record buffers**

**Cause:** Unable to allocate memory for a record being created in a block.

**Action:** Try restarting application when fewer programs are running concurrently, or on a machine with more memory. The design may need to be changed so that one or more blocks have smaller values for the Records Buffered Property.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41841: Use the debugger-enabled executable if specifying DEBUG=YES.**

**Cause:** You tried to use the debugger from an executable which doesn't include it.

**Action:** Run the other executable (name will vary with operating system), which includes the debugger.

**Level:** 99

**Trigger:** None

**FRM-41843: Invalid time zone region %s for ADJUST_TZ.**

**Cause:** A call to the ADJUST_TZ procedure specified an invalid 'from' or 'to' time zone region name.

**Action:** Specify a valid name. If the name is valid, you may need to ask your system administrator to install a larger time zone file.

**Level:** 25

**Trigger:** ON-ERROR

**FRM-41844: ADJUST_TZ could not convert date.**

**Cause:** A call to the ADJUST_TZ procedure specified valid 'from' and 'to' time zone region names, but nevertheless failed. This probably indicates that the date was too close to the boundary dates of Jan 1, 4712 BC or Dec 31, 9999 AD.

**Action:** Specify a valid date.

**Level:** 25

**Trigger:** ON-ERROR

**FRM-41845: Javascript events have been disabled.**

**Cause:** Either the environment variable FORMS_ALLOW_JAVASCRIPT_ EVENTS or the applet parameter enableJavaScriptEvent has been set to FALSE.

**Action:** Set client applet's parameter enableJavaScriptEvent and the server's environment variable FORMS_ALLOW_JAVASCRIPT_EVENTS to true. The default value of both these variables is true.

**Level:** 15

**Trigger:** ON-ERROR

**FRM-41846: Too many items in block %s.**

**Cause:** The specified block contains more than approximately 5000 items.

**Action:** Redesign the form.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-41900: Run aborted by fatal error.**
    **Cause:** Internal error.

    **Action:** If the problem persists, contact Oracle Support Services.

    **Level:** 99

    **Trigger:** None

**FRM-41901: Error: %d cursors were not closed.**
    **Cause:** Internal error.

    **Action:** If the problem persists, contact Oracle Support Services.

    **Level:** 99

    **Trigger:** None

**FRM-41902: Total cursors used %d.**
    **Cause:** This message appears when you run a form with the Statistics preference set to True.

    **Action:** No action is necessary.

    **Level:** 20

    **Trigger:** ON-MESSAGE

**FRM-41903: Run aborted by end of input file.**
    **Cause:** Internal error.

    **Action:** If the problem persists, contact Oracle Support Services.

    **Level:** 99

    **Trigger:** None

**FRM-42017: Module name must be specified.**
    **Cause:** You did not specify a module name.

    **Action:** Specify a module name.

    **Level:** 99

    **Trigger:** None

**FRM-42100: No errors encountered recently.**
    **Cause:** You pressed [Display Error], but no error has occurred recently.

    **Action:** No action is necessary.

    **Level:** 5

    **Trigger:** ON-MESSAGE

**FRM-42400: Performing event trigger %s.**
    **Cause:** This message is displayed during a trigger when debug mode is specified.

    **Action:** No action is necessary.

    **Level:** 99

    **Trigger:** ON-MESSAGE

**FRM-42401: Performing program trigger %s.**

**Cause:** This message is displayed during a trigger when debug mode is specified.

**Action:** No action is necessary.

**Level:** 99

**Trigger:** ON-MESSAGE

**FRM-42423: Cannot execute trigger %s: no compiled state**

**Cause:** This message is displayed when running a form in debug mode, if the compiled state of a trigger has been destroyed. This can happen if you apply a change to the trigger text and that change results in compilation errors.

**Action:** You can recompile the trigger in the debugger or exit the form and start it up again.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-42431: Unable to initialize debugger.**

**Cause:** An error occured while attempting to initialize debugger. This could be caused by one of the following:

1. The JVM failed to startup.

2. The Classpath does not include the debugger dependencies.

**Action:** Restart the JVM or modify the Classpath.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-42435: Remote Debugger: Specified port(s) are not available.**

**Cause:** Other processes are using the specified port(s).

**Action:** Try some other port.

**Level:** 99

**Trigger:** None

**FRM-47000: Cannot create Parameter List %s : internal error.**

**Cause:** Internal error.

**Action:** If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47001: Cannot create Parameter List %s : list with this name exists.**

**Cause:** The name you specified for the parameter list is already in use.

**Action:** Specify another name for the parameter list.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47002: Cannot create Parameter List : name must not be null.**

**Cause:** The parameter list name cannot be null.

**Action:** Specify a name for the parameter list.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47003: Cannot delete Parameter List : internal error.**
    **Cause:** Internal error.
    **Action:** If the problem persists, contact Oracle Support Services.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-47004: Cannot delete Parameter List : invalid ID.**
    **Cause:** Attempted to pass an invalid parameter list ID.
    **Action:** Check the list ID name and try again.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-47005: Cannot validate parameter %s : internal error.**
    **Cause:** Internal error.
    **Action:** If the problem persists, contact Oracle Support Services.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-47006: Cannot create Parameter List '%s' : illegal identifier name.**
    **Cause:** Illegal identifier name.
    **Action:** Check valid syntax for the identifier.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-47007: Cannot get parameter %s attributes from Parameter List : invalid list ID.**
    **Cause:** Specified an invalid parameter list ID.
    **Action:** Check the parameter list ID name and try again.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-47008: Cannot add parameter %s to Parameter List %s : internal error.**
    **Cause:** An internal error occurred.
    **Action:** If the problem persists, contact Oracle Support Services.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-47009: Cannot add parameter %s to Parameter List : invalid list ID.**
    **Cause:** You specified an invalid parameter list ID.
    **Action:** Check the parameter list ID name and try again.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-47010: Cannot add parameter to Parameter List %s : null key specified.**

**Cause:** No name specified for the parameter.

**Action:** Specify a parameter name key in your call to ADD_PARAMETER.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47011: Cannot add parameter %s to Parameter List %s : incorrect type specified.**

**Cause:** You specified an invalid parameter type.

**Action:** Specify either TEXT_PARAMETER or DATA_PARAMETER.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47012: Cannot add parameter %s to Parameter List %s : group %s does not exist.**

**Cause:** Record group name does not exist.

**Action:** Check the record group name and try again.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47013: Cannot add parameter %s to Parameter List %s : parameter with this name exists.**

**Cause:** Parameter with this name already exists.

**Action:** Specify another name for the parameter.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47014: Cannot delete parameter %s from Parameter List %s : internal error.**

**Cause:** Internal error.

**Action:** If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47015: Cannot delete parameter %s from Parameter List : invalid list ID.**

**Cause:** Specified an invalid parameter list ID.

**Action:** Check the list ID name and try again.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47016: Cannot delete parameter from Parameter List %s : null key specified.**

**Cause:** You did not specify a name for the parameter.

**Action:** Correct the call to DELETE_PARAMETER by supplying a parameter name.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47017: Cannot delete parameter %s from Parameter List %s : no such named parameter exists.**

**Cause:** Caused by one of the following:

1. You specified an invalid parameter list ID.

2. You specified a parameter name that does not exist.

**Action:** Check the parameter name and try again.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47018: Cannot set parameter %s attributes in Parameter List %s : internal error.**

**Cause:** Internal error: you specified an invalid parameter list ID.

**Action:** If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47019: Cannot set parameter %s attributes in Parameter List : invalid list ID.**

**Cause:** You specified an invalid parameter list ID.

**Action:** Check the parameter list ID name and try again.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47020: Cannot set parameter %s attributes in Parameter List %s : no such named parameter exists.**

**Cause:** Caused by one of the following:

1. You specified an invalid parameter list ID.

2. You specified a parameter name that does not exist.

**Action:** Check the parameter name and try again.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47021: No such parameter named %s exists in Parameter List %s.**

**Cause:** Caused by one of the following:

1. You specified an invalid parameter list ID.

2. You specified a parameter name that does not exist.

**Action:** Check the name and try again.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47022: Cannot create Parameter List %s : name is a reserved word.**

**Cause:** Caused by one of the following:

1. You specified an invalid parameter list ID.

2. You specified a name that is a reserved word.

**Action:** Specify another name for the parameter list.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47023: No such parameter named %s exists in form %s.**
**Cause:** Caused by one of the following:

1. You specified an invalid parameter list ID.

2. You specified a parameter name does not exist.

**Action:** Check the name and try again.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47024: Parameter %s type does not match definition in form %s.**
**Cause:** You specified a parameter type that does not match the definition in the form.

**Action:** Specify a parameter type that matches the definition in the form.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47025: Cannot get parameter %s attributes from Parameter List %s : internal error.**
**Cause:** Internal error: you specified an invalid parameter list ID.

**Action:** If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47026: Cannot get parameter %s attributes from Parameter List %s : no such named parameter exists.**
**Cause:** Caused by one of the following:

1. You specified an invalid parameter list ID.

2. You specified a parameter name does not exist.

**Action:** Check the name and try again.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47027: Cannot add parameter %s to Parameter List %s : invalid key specified .**
**Cause:** You specified an invalid parameter list ID.

**Action:** Check the name and try again.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47028: Cannot set parameter %s attribute in Parameter List %s : group %s does not exist.**
**Cause:** You specified an invalid parameter list ID.

**Action:** Check the name and try again.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47029: Invalid parameter list ID in form %s.**

**Cause:** Caused by one of the following:

1. You specified an invalid parameter list ID.

2. You specified a parameter name does not exist.

**Action:** Check the ID and try again.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47030: Value of parameter %s is too long for definition in form %s.**

**Cause:** You specified a parameter that is too long.

**Action:** Specify a parameter that is valid.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47031: Cannot set value of parameter %s in DEFAULT parameter list: invalid value specified.**

**Cause:** Application design error. A Built-in (such as SET_PARAMETER_ATTR) is attempting to set the value of a parameter which was defined when the form was designed, but the value specified is not legal for the parameter's datatype.

**Action:** The call to the Built-in must be modified or removed.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47032: Cannot set value of parameter %s in DEFAULT parameter list: internal error.**

**Cause:** An internal error while attempting to set the value of a parameter.

**Action:** If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47033: Cannot set value of read-only bind variable %s.**

**Cause:** Application design error. The application attempted to assign a value to a bind variable which cannot be programmtically modified.

**Action:** The assignment must be removed.

**Level:** 25

**Trigger:** ON-ERROR

**FRM-47100: Cannot read image file %s.**

**Cause:** Caused by one of the following:

1. You specified an invalid parameter list ID.

2. Oracle Forms was unable to find or open the file.

3. The data in the file is not in the specified format.

**Action:** Check the file name and file format and try again.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47101: Cannot write image file %s.**

**Cause:** Caused by one of the following:

1. You specified an invalid parameter list ID.

2. Oracle Forms was unable to find or open the file.

3. The data in the file is not in the specified format.

**Action:** Check the file name, and make sure you have write privileges.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47102: Cannot perform %s operation on images %s and %s.**

**Cause:** Caused by one of the following:

1. You specified an invalid parameter list ID.

2. This operation cannot be performed on color images.

**Action:** Check to see that both images are black and white.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47103: Cannot zoom image %s.**

**Cause:** Caused by one of the following:

1. You specified an invalid parameter list ID.

2. Internal multimedia error caused by trying to scale a null image or invalid image data.

**Action:** Check the image name that you want to zoom and try again.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47104: Invalid image type %s.**

**Cause:** Caused by one of the following:

1. You specified an invalid parameter list ID.

2. Data in the file name specified does not match the data type specified.

**Action:** Check the file name and try again.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47105: No image name specified.**

**Cause:** Caused by one of the following:

1. You specified an invalid parameter list ID.

2. You did not supply a name to the Built-in call.

**Action:** Refer to the documentation for the proper syntax for the Built-in in question.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47106: No image type specified.**
> **Cause:** Caused by one of the following:
>
> 1. You specified an invalid parameter list ID.
>
> 2. You did not specify an image type when calling READ_IMAGE_FILE or WRITE_IMAGE_FILE.
>
> **Action:** Supply an image type as an argument in your call to READ_IMAGE_FILE or WRITE_IMAGE_FILE.

> **Level:** 99

> **Trigger:** ON-ERROR

**FRM-47107: Invalid scaling factor %d for Image_Zoom.**
> **Cause:** Caused by one of the following:
>
> 1. You specified an invalid parameter list ID.
>
> 2. You specified an invalid scaling factor.
>
> **Action:** Correct the call to IMAGE_ZOOM.

> **Level:** 20

> **Trigger:** ON-ERROR

**FRM-47108: Item %s is not an image item.**
> **Cause:** Caused by one of the following:
>
> 1. You specified an invalid parameter list ID.
>
> 2. You attempted to perform an image operation on an item that is not an image item.
>
> **Action:** Check the item name and try again.

> **Level:** 99

> **Trigger:** ON-ERROR

**FRM-47109: Cannot locate image file %s.**
> **Cause:** Caused by one of the following:
>
> 1. You specified an invalid parameter list ID.
>
> 2. You specified a file that cannot be found or does not exist.
>
> **Action:** Verify that the file exists and the pathname is correct.

> **Level:** 20

> **Trigger:** ON-ERROR

**FRM-47110: No region was selected for Image_Zoom: %s**
> **Cause:** Caused by one of the following:
>
> 1. You specified an invalid parameter list ID.
>
> 2. You tried to zoom an image without selecting an image region.
>
> **Action:** Select an image region.

> **Level:** 20

**Trigger:** ON-ERROR

**FRM-47111: Cannot copy value to item: %s**

    **Cause:** Caused by one of the following:

    1. You specified an invalid parameter list ID.

    2. You tried to use the COPY Built-in on an image item.

    **Action:** You cannot use the COPY Built-in on an image item.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-47300: Item is not a hierarchical tree. (%s)**

    **Cause:** A hierarchical tree Built-in was invoked on a non-tree item.

    **Action:** Check item type and name.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-47301: Cannot add data as sibling to the tree root.**

    **Cause:** ADD_TREE_DATA or ADD_TREE_NODE attempted to add data as a sibling of the root.

    **Action:** Add data at a lower level in the tree.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-47302: Can only add data to tree as child or sibling.**

    **Cause:** ADD_TREE_DATA or ADD_TREE_NODE attempted to use an unknown offset_type value.

    **Action:** Only PARENT_OFFSET and SIBLING_OFFSET are allowed for the offset_type parameter.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-47303: ADD_TREE_DATA only accepts data from a group or query.**

    **Cause:** ADD_TREE_DATA attempted to use an unknown datasource value.

    **Action:** Only RECORD_GROUP and QUERY_TEXT are allowed for the datasource parameter.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-47304: Cannot delete the root node of a tree.**

    **Cause:** DELETE_TREE_NODE attempted to delete the root node.

    **Action:** Check if a node is the root (use ID_NULL) before trying to delete it.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-47305: Can only search a tree looking for label or value.**

    **Cause:** FIND_TREE_NODE attempted to use an unknown search_by parameter value.

**Action:** Set the search_by parameter to NODE_LABEL or NODE_VALUE.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47306: Search_type must be FIND_NEXT or FIND_NEXT_CHILD.**

**Cause:** FIND_TREE_NODE attempted to use an invalid search_type parameter value.

**Action:** Set the search_type parameter to FIND_NEXT or FIND_NEXT_CHILD.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47307: Cannot get the properties of the tree root node.**

**Cause:** GET_TREE_NODE_PARENT or GET_TREE_NODE_PROPERTY attempted to obtain information from the root node.

**Action:** Only invoke the Built-ins against valid nodes.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47308: Invalid property for GET or SET_TREE_NODE_PROPERTY.**

**Cause:** You passed an invalid property constant to GET or SET_TREE_NODE_ PROPERTY.

**Action:** Verify arguments.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47309: Invalid property for GET or SET_TREE_PROPERTY.**

**Cause:** You passed an invalid property constant to GET or SET_TREE_ PROPERTY.

**Action:** Verify arguments.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47310: Bad selection index for GET_TREE_SELECTION.**

**Cause:** Selection index must be in the range 1 .. number of selected nodes.

**Action:** Ensure that the selection is within the required range.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47311: Error populating record group.**

**Cause:** Invalid record group specified for POPULATE_GROUP_FROM_TREE.

**Action:** Check existence of the record group, and that it isn't statically defined.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47312: Internal error populating record group.**

**Cause:** Internal error during POPULATE_GROUP_FROM_TREE. Note that some rows may already have been added.

**Action:** Unable to add row to the record group.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-47313: Invalid query for the hierarchical tree.

**Cause:** Unable to create valid tree data from the specified query text.

**Action:** Check the query text for a valid number of columns and valid data types.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-47314: Cannot set the properties of the tree root node.

**Cause:** SET_TREE_NODE_PROPERTY attempted to set a property of the root node.

**Action:** Only invoke the Built-in against valid nodes.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-47315: Invalid parameter value for SET_TREE_NODE_PROPERTY.

**Cause:** Check the parameter values for the tree node property being set.

**Action:** Correct the parameter value passed to SET_TREE_NODE_PROPERTY.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-47316: Branch nodes with no children are not allowed.

**Cause:** Attempt to change the state of a node from a leaf node to a branch. This tree does not allow empty branch nodes.

**Action:** Either change the tree to allow empty branch nodes, or add children to the node.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-47317: Leaf nodes cannot have children.

**Cause:** Attempt to change the state of a node with children from a branch node to a leaf.

**Action:** Delete the children from the node before changing the node's state.

**Level:** 99

**Trigger:** ON-ERROR

### FRM-47318: Invalid parameter value for SET_TREE_PROPERTY.

**Cause:** Check the parameter values for the tree property being set.

**Action:** Correct the parameter value passed to SET_TREE_PROPERTY.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47319: Cannot select the tree root node.**

**Cause:** SET_TREE_SELECTION attempted to select the root node.

**Action:** Only invoke the Built-in against valid nodes.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47320: Bad selection type for SET_TREE_SELECTION.**

**Cause:** SET_TREE_SELECTION attempted to use an invalid selection_type parameter value.

**Action:** Set the selection_type parameter to SELECT_ON, SELECT_OFF, or SELECT_TOGGLE.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47321: Data used to populate tree is invalid.**

**Cause:** ADD_TREE_DATA attempted to use data of wrong format.

**Action:** Check number and type of columns in group or query.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47322: The specified tree data source is not a record group.**

**Cause:** The data source for the tree was declared a record group, but isn't.

**Action:** Check that the id or name specified is of an existing record group.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47323: No nodes are selected in the tree.**

**Cause:** Attempt was made to obtain a selected node when none are currently selected.

**Action:** Check for number of selected nodes before trying to retrieve one of them.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47324: Could not allocate memory for tree structures.**

**Cause:** Unable to allocate memory for internal tree structures. Tree destroyed.

**Action:** Try executing the application when the system is less heavily loaded. If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47325: Could not allocate memory for tree node.**

**Cause:** Unable to allocate memory for a tree node.

**Action:** Try executing the application when the system is less heavily loaded. If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47333: Could not set required state for tree node.**

**Cause:** Unable to change state for a tree node.

**Action:** Try executing the application when the system is less heavily loaded. If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47334: Could not allocate memory for tree node icon string.**

**Cause:** Unable to allocate memory for the name of a tree node icon.

**Action:** Try executing the application when the system is less heavily loaded. If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47335: Could not locate a tree node icon. (%s)**

**Cause:** Unable to find the desired icon in standard locations.

**Action:** Check that your tree node icons are located in the proper directories.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47336: Could not set tree node to the requested icon.**

**Cause:** Unable to set the requested tree node icon.

**Action:** Try executing the application when the system is less heavily loaded. If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47337: Tree node label cannot be null.**

**Cause:** Attempt was made to set a tree node's label to a null value.

**Action:** Set the label to a non-null value.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47338: Could not allocate memory for tree node label.**

**Cause:** Unable to allocate memory for a tree node label.

**Action:** Try executing the application when the system is less heavily loaded. If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47339: Could not set tree node to the requested label.**

**Cause:** Unable to set the requested tree node label.

**Action:** Try executing the application when the system is less heavily loaded. If the problem persists, contact Oracle Support Services.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47340: Could not allocate memory for tree node value.**

    **Cause:** Unable to allocate memory for a tree node value.

    **Action:** Try executing the application when the system is less heavily loaded. If the problem persists, contact Oracle Support Services.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-47341: There are too many nodes for the tree.**

    **Cause:** Only MAX-SIGNED-4-BYTE nodes, both current and deleted, are permitted in a tree.

    **Action:** Decrease the number of nodes placed in the tree. If constantly adding and removing nodes, you might need to clear and re-populate the tree.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-47342: Could not allocate memory for tree query text.**

    **Cause:** Unable to allocate memory for the tree query text.

    **Action:** Try executing the application when the system is less heavily loaded. If the problem persists, contact Oracle Support Services.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-47343: Invalid node ID specified for hierarchical tree item %s**

    **Cause:** A hierarchical tree built-in was invoked on a hierarchical tree item, but the node ID passed to the built-in was not valid for the tree item.

    **Action:** Specify a valid node ID.

    **Level:** 99

    **Trigger:** ON-ERROR

**FRM-47500: Failed to register database event %s**

    **Cause:** Attempt to register a database event failed

    **Action:** Check the event attributes on the database side

    **Level:** 5

    **Trigger:** ON-ERROR

**FRM-47501: Invalid event id**

    **Cause:** Invalid event id

    **Action:** Check the event id

    **Level:** 99

    **Trigger:** None

**FRM-47502: Invalid event property**

    **Cause:** Invalid event property

    **Action:** Check the event event property

**Level:** 5

**Trigger:** ON-ERROR

**FRM-47700: Failed to start the JVM.**

**Cause:** An error occured while attempting to start the inprocess JVM.

**Action:** Make sure that jvm libraries can be located by the runtime process

**Level:** 99

**Trigger:** ON-ERROR

**FRM-47800: Unable to communicate with the JVM Controller: %s.**

**Cause:** Unable to communicate with the JVM Controller

**Action:** JVM Controller, to which runform is connected, might be down. Contact your system administrator.

**Level:** 99

**Trigger:** ON-ERROR

**FRM-50000: Value is too long.**

**Cause:** You entered a value which contains too many bytes or characters for the item.

**Action:** Enter a shorter value.

**Level:** 15

**Trigger:** ON-ERROR

**FRM-50001: Acceptable characters are a-z, A-Z, and space.**

**Cause:** You entered an unacceptable character into the item.

**Action:** Enter a character from a-z, A-Z, or a space.

**Level:** 15

**Trigger:** ON-ERROR

**FRM-50002: Month must be between 1 and 12.**

**Cause:** You entered an invalid month value in a date field.

**Action:** Enter a month value from 1 (for January) to 12 (for December).

**Level:** 15

**Trigger:** ON-ERROR

**FRM-50003: Year must be in proper range.**

**Cause:** You entered a year that is not valid for the applicable format mask year element.

**Action:** Enter a valid year. For most format mask year elements, a number between 0 and 9999 is acceptable. For signed format mask year elements, a number between -4712 and -1 may also be specified.

**Level:** 15

**Trigger:** ON-ERROR

**FRM-50004: Day must be between 1 and last of month.**

**Cause:** You entered an invalid day.

**Action:** Enter a valid day. For April, for example, enter a number between 1 and 30.

**Level:** 15

**Trigger:** ON-ERROR

**FRM-50006: Legal characters are 0-9 + and -.**

**Cause:** You entered an unacceptable character in a number item.

**Action:** Enter a valid number. A valid number has digits 0 through 9. A number may be preceded by a plus (+) or minus (-) sign. If the message allows it, a number may contain one decimal point at any location, except before the sign.

**Level:** 15

**Trigger:** ON-ERROR

**FRM-50007: Too many digits after decimal point.**

**Cause:** You entered a number with 3 or more decimal digits after the decimal point in an item with the MONEY or RMONEY data type.

**Action:** Re-enter a valid number.

**Level:** 15

**Trigger:** ON-ERROR

**FRM-50009: Too many decimal points.**

**Cause:** You entered a number that contains two or more decimal points, or you have entered a number that contains a decimal point in an item that requires a whole (non-decimal) number.

**Action:** Enter a number with no more than one decimal point. If you have used only one decimal, remove the decimal and the decimal part of the number.

**Level:** 15

**Trigger:** ON-ERROR

**FRM-50010: Money format is [+-]9999999.99**

**Cause:** You entered an invalid value in a MONEY or RMONEY item.

**Action:** Enter a valid value. This value should have zero or dollar digits, followed by a decimal and two cents digits. The entire number can be preceded by a plus (+) or a minus (-) sign.

**Level:** 15

**Trigger:** ON-ERROR

**FRM-50011: Not a valid month name.**

**Cause:** You entered an invalid month name in a date field.

**Action:** Enter a valid month name. Oracle Forms recognizes the first three characters of a month name. For example, JAN stands for January, JUN for June.

**Level:** 15

**Trigger:** ON-ERROR

**FRM-50012: Date must be entered in a format like %s.**

**Cause:** You entered an invalid or incorrectly formatted date.

**Action:** Re-enter the date in the requested format.

**Level:** 15

**Trigger:** ON-ERROR

**FRM-50013: Plus or minus must be in first position.**
    **Cause:** You entered the plus or minus sign in the wrong position.

    **Action:** Retype with the plus or minus sign in the first position.

    **Level:** 15

    **Trigger:** ON-ERROR

**FRM-50014: Bad exponent.**
    **Cause:** You entered an exponent in an item that does not accept exponents.

    **Action:** Enter a value without an exponent.

    **Level:** 15

    **Trigger:** ON-ERROR

**FRM-50016: Legal characters are 0-9 - + E .**
    **Cause:** You entered an unacceptable character in a number item.

    **Action:** Enter a valid number. A valid number has digits 0 through 9. A number may be preceded by a plus (+) or minus (-) sign. If the message allows it, a number may contain one decimal point at any location, except before the sign. You can use an E to specify scientific notation.

    **Level:** 15

    **Trigger:** ON-ERROR

**FRM-50017: Hour must be between 0 and 23.**
    **Cause:** You entered an invalid hour.

    **Action:** Enter a valid hour. Oracle Forms records time on a 24-hour basis.

    **Level:** 15

    **Trigger:** ON-ERROR

**FRM-50018: Minutes must be between 00 and 59.**
    **Cause:** You entered an invalid minute value.

    **Action:** Enter a valid minute value.

    **Level:** 15

    **Trigger:** ON-ERROR

**FRM-50019: Seconds must be between 00 and 59.**
    **Cause:** You entered an invalid value.

    **Action:** Enter a value between 00 and 59.

    **Level:** 15

    **Trigger:** ON-ERROR

**FRM-50020: Missing exponent.**
    **Cause:** You failed to enter an exponent.

    **Action:** Enter an exponent.

    **Level:** 15

**Trigger:** ON-ERROR

**FRM-50021: Date must be entered in a format like %s.**

**Cause:** You entered an invalid or incorrectly formatted date.

**Action:** Re-enter the date in the requested format.

**Level:** 15

**Trigger:** ON-ERROR

**FRM-50022: Time must be entered in a format like %s.**

**Cause:** You entered an invalid or incorrectly formatted time.

**Action:** Re-enter the time in the requested format.

**Level:** 15

**Trigger:** ON-ERROR

**FRM-50023: Date must be entered in a format like %s.**

**Cause:** You entered an invalid or incorrectly formatted date.

**Action:** Re-enter the date in the requested format.

**Level:** 15

**Trigger:** ON-ERROR

**FRM-50024: Space are allowed in leading positions only.**

**Cause:** You entered spaces intermixed with data.

**Action:** Re-enter data with no spaces intermixed.

**Level:** 15

**Trigger:** ON-ERROR

**FRM-50025: Date/time must be entered in a format like %s.**

**Cause:** You entered an invalid or incorrectly formatted date and time.

**Action:** Re-enter the date and time in the requested format.

**Level:** 15

**Trigger:** ON-ERROR

**FRM-50026: Date must be entered in a format like %s.**

**Cause:** You entered an invalid or incorrectly formatted date.

**Action:** Re-enter the date in the requested format.

**Level:** 15

**Trigger:** ON-ERROR

**FRM-50027: Invalid format mask for given datatype.**

**Cause:** The format mask you assigned to a text item is incompatible with the data type of the text item.

**Action:** Assign a new format mask to the text item. For more information, refer to help on <a href="../../designing_forms/items/f500842.html">About Formatting Text Item Values with Format Masks</a>.

**Level:** 15

**Trigger:** ON-ERROR

**FRM-50028: Format mask not allowed for this datatype.**

    **Cause:** The data types LONG and IMAGE do not support a format mask.

    **Action:** Do not try to create a format mask for data types LONG or IMAGE.

    **Level:** 15

    **Trigger:** ON-ERROR

**FRM-50029: Too many digits preceding decimal point for scientific notation.**

    **Cause:** You specified a number using scientific notation, but used more than one digit preceding the decimal point.

    **Action:** Re-enter the number using scientific notation.

    **Level:** 15

    **Trigger:** ON-ERROR

**FRM-50045: Seconds past midnight conflilicts with hour.**

    **Cause:** You entered a time where the seconds past midnight component does not agree with the hour component.

    **Action:** Make sure the hour and seconds past midnight agree, or use a format mask without seconds past midnight.

    **Level:** 15

    **Trigger:** ON-ERROR

**FRM-50048: New passwords do not match. Please make them identical.**

    **Cause:** You entered different strings in 'New Password' and 'Retype New' fields.

    **Action:** Re-enter the values in (New and Retype) fields such that they identical.

    **Level:** 99

    **Trigger:** None

**FRM-91124: fatal error in runtime process: %s specified for FORMS_DECIMAL_ PREFIX. Should be zero or the empty string**

    **Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

    **Action:** Contact your system administrator.

    **Appears:** Java console, alert

    **Level:** 99

    **Trigger:** None

**FRM-91126: fatal error in runtime process: invalid value %s specified for environment variable %s**

    **Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

    **Action:** Contact your system administrator.

    **Appears:** Java console, alert

    **Level:** 99

    **Trigger:** None

**FRM-91127: fatal error in runtime process: invalid directory name specified for environment variable %s**

**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-91129: fatal error in runtime process: no value specified for required environment variable %s**

**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-91130: fatal error in runtime process: timezone file %s is missing**

**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-91131: fatal error in runtime process: timezone file %s not loaded - required memory unavailable**

**Cause:** The timezone file could not be loaded. This probably indicates insufficient swap space.

**Action:** Retry the application when the system is less heavily loaded.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-91132: fatal error in runtime process: invalid data in timezone file %s**

**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-91135: fatal error in runtime process: message file %s is missing**

**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

### FRM-92000: internal error: cannot access Java class

**Cause:** The Forms server requested a Java class by specifying a numeric "handlerClassId", and the Forms Java client found an entry for the specified handlerClassId in the registry. However, the Java class that was specified by the registry entry could not be accessed.

**Action:** Examine the stack trace that accompanies this message. If the stack trace indicates a possible cause (e.g. a configuration problem), correct it. If the problem persists, contact Oracle Support Services.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

### FRM-92010: Fatal error: serverArgs parameter is either not set or is blank.

**Cause:** The serverArgs applet parameter (which represents the command-line arguments that are passed to the frmweb executable) is either missing or has a blank value. The base HTML files that are shipped with the produce specify a valid value for the serverArgs parameter, so presumably that value has been modified, or else a non-standard base HTML file has been specified.

**Action:** The system administrator should ensure that the base HTML file correctly defines the serverArgs parameter.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

### FRM-92020: invalid URL %s sent to browser with target %s. full details: %s

**Cause:** The Forms application executed the web.showDocument built-in, and the applet did not define the clientBrowser parameter, which caused Forms to attempt to resolve the specified URL relative to the applet's document base. However, this attempt encountered a MalformedURLException. This is not a fatal error.

**Action:** Correct the URL that is passed to the web.showDocument built-in.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

### FRM-92030: internal error: no registry entry for handleClassId=%s

**Cause:** The Forms server requested a Java class by specifying a numeric "handlerClassId", but the Forms Java client could not find an entry for the specified handlerClassId in the registry.

**Action:** If the problem persists, contact Oracle Support Services.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-92040: internal error: cannot find Java class**

**Cause:** The Forms server requested a Java class by specifying a numeric "handlerClassId", and the Forms Java client found an entry for the specified handlerClassId in the registry. However, the Java class that was specified by the registry entry could not be found.

**Action:** Examine the stack trace that accompanies this message. If the stack trace indicates a possible cause (e.g. a configuration problem), correct it. If the problem persists, contact Oracle Support Services.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-92050: fatal error: cannot connect to the server: %s:%s**

**Cause:** An attempt was made to connect to the Forms server. The serverURL applet parameter was not specified, so Forms attempted to connect to the specified host machine, on the specified port. (These are derived from the serverHost and serverPort applet parameters, if specified). However, an unexpected Exception was encountered. This message appears when there is no message that gives a more specific reason for the connection failure.

**Action:** Examine the stack trace that accompanies this message. If the stack trace indicates a possible cause, correct it. If the problem persists, contact Oracle Support Services.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-92052: fatal error: cannot connect to the server at URL %s**

**Cause:** An attempt was made to connect to the Forms server, at the specified URL. (This is derived from the serverURL applet parameter). However, an unexpected Exception was encountered. This message appears when there is no message that gives a more specific reason for the connection failure.

**Action:** Examine the stack trace that accompanies this message. If the stack trace indicates a possible cause, correct it. If the problem persists, contact Oracle Support Services.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-92060: fatal error: cannot connect to the server: bad machine specification: %s:%s**

**Cause:** An attempt was made to connect to the Forms server. The serverURL applet parameter was not specified, so Forms attempted to connect to the specified host machine, on the specified port. (These are derived from the serverHost and serverPort applet parameters, if specified). However, the format of the host/port combination was invalid.

**Action:** Correct the syntax of the serverHost and/or the serverPort applet parameter, or specify a valid value for the serverURL applet parameter.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-92062: fatal error: cannot connect to the server: bad URL specification: %s**

    **Cause:** An attempt was made to connect to the Forms server, at the specified URL. (This is derived from the serverURL applet parameter). However, the URL was malformed.

    **Action:** Correct the syntax of the serverURL applet parameter.

    **Appears:** Java console, alert

    **Level:** 99

    **Trigger:** None

**FRM-92070: internal error: cannot instantiate Java class**

    **Cause:** The Forms server requested a Java class by specifying a numeric "handlerClassId", and the Forms Java client found an entry for the specified handlerClassId in the registry. However, the Java class that was specified by the registry entry could not be instantiated.

    **Action:** Examine the stack trace that accompanies this message. If the stack trace indicates a possible cause (e.g. a configuration problem), correct it. If the problem persists, contact Oracle Support Services.

    **Appears:** Java console, alert

    **Level:** 99

    **Trigger:** None

**FRM-92080: fatal error: cannot execute command: %s %s. full details: %s**

    **Cause:** The Forms application executed the web.showDocument built-in, and the applet defined the clientBrowser parameter, which caused Forms to attempt to start the specified external client browser. However, this attempt encountered an Exception.

    **Action:** If the "full details" messages indicates the problem, correct it. Otherwise, if the problem persists, contact Oracle Support Services.

    **Appears:** Java console, alert

    **Level:** 99

    **Trigger:** None

**FRM-92089: unexpected fatal error while initializing the applet's user interface**

    **Cause:** An unexpected Exception was encountered while attempting to initialize the applet's user interface.

    **Action:** Examine the stack trace that accompanies this message. If the stack trace indicates a possible cause, correct it. If the problem persists, contact Oracle Support Services.

    **Appears:** Java console, applet status window

    **Level:** 99

    **Trigger:** None

**FRM-92090: unexpected fatal error in client-side Java code during startup**

**Cause:** An unexpected Exception was encountered in client-side Java code during startup.

**Action:** Examine the stack trace that accompanies this message. If the stack trace indicates a possible cause, correct it. If the problem persists, contact Oracle Support Services.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

### FRM-92091: unexpected fatal error in client-side Java code

**Cause:** An unexpected Exception was encountered in client-side Java code (after startup).

**Action:** Examine the stack trace that accompanies this message. If the stack trace indicates a possible cause, correct it. If the problem persists, contact Oracle Support Services.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

### FRM-92102: A network error or server failure has occurred. The Forms client has attempted to reestablish its connection to the Server %s time(s) without success. You will need to restart your application.

**Cause:** The Forms Java client attempted to communicate with the Forms server. The indicated number of attempts (specified by the networkRetries applet parameter) were made, but each attempt encountered an unexpected Exception. This probably indicates a problem with the network, or with the application server that was hosting the Forms server, or with the server machine that was hosting the application server.

**Action:** Correct the network problem (if any), or restart the application server or reboot the server host machine (if necessary).

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

### FRM-92103: A network error or server failure has occurred. You will need to restart your application.

**Cause:** The Forms Java client attempted to communicate with the Forms server. The networkRetries applet parameter did not specify a positive value, so only a single attempt was made. This attempt encountered an unexpected Exception. This probably indicates a problem with the network, or with the application server that was hosting the Forms server, or with the server machine that was hosting the application server.

**Action:** Correct the network problem (if any), or restart the application server or reboot the server host machine (if necessary).

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-92104: A network error or server failure has occurred. The request was sent to the wrong application server (not the one which created the session). The Forms client has attempted to migrate the session %s time(s) without success. You will need to restart your application.**

**Cause:** The Forms Java client attempted to communicate with the Forms server. The indicated number of attempts were made, but on each attempt, the request was sent to the wrong application server (not the one which created the session). This probably indicates a problem with the network, or with the application server that was hosting the Forms server that initially created the session, or with the server machine that was hosting the application server.

**Action:** Correct the network problem (if any), or restart the application server or reboot the server host machine (if necessary).

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-92110: New passwords do not match. They must be identical. Password change failed.**

**Cause:** In the Change Password dialog, the new password and the retyped new password do not match.

**Action:** Retry the Change Password dialog, and correctly type and retype the new password.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-92120: Fatal error: registry file %s is missing.**

**Cause:** The Forms Java client was unable to read the registry file at the specified URL (on the Forms server machine).

**Action:** The system administrator should ensure that the registry file exists and is readable.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-92150: Fatal error: web client version is too new.**

**Cause:** The version of the client is newer than the version of the Server.

**Action:** The system administrator should ensure that the Forms product is correctly installed on the Forms server machine.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-92160: Fatal error: web client version is too old.**

**Cause:** The version of the client is older than the version of the Server.

**Action:** The system administrator should ensure that the Forms product is correctly installed on the Forms server machine.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-92180: Fatal error: JavaScript is unable to obtain the server URL. This can occur if legacy_lifecycle=true and JavaScript has been disabled. If so, you will need to reenable JavaScript, restart the browser, and restart your application.**

**Cause:** The serverURL applet parameter specified a value of "?", which indicates that the serverURL should be obtained from an element outside of the applet. (In a page that's generated from a standard base HTML file, this occurs when legacy_lifecycle=true in formsweb.cfg). Forms attempted to obtain the serverURL using JavaScript, but the attempt failed, probably because JavaScript has been disabled.

**Action:** Reenable JavaScript and restart the browser. If that does not solve the problem, examine the stack trace that accompanies this message. If the stack trace indicates a possible cause, correct it. If the problem persists, contact Oracle Support Services.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-92190: JavaScript is unable to evaluate expression.**

**Cause:** The Forms application executed the web.JavaScript_Eval_Expr built-in, but an invalid Javascript expression was specified as the first argument.

**Action:** Correct the Javascript expression that is passed to the web.JavaScript_Eval_Expr built-in.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-92192: Target %s for JavaScript evaluation does not exist.**

**Cause:** The Forms application executed the web.JavaScript_Eval_Expr built-in, but an invalid target was specified as the second argument.

**Action:** Correct the target that is passed to the web.JavaScript_Eval_Expr built-in.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-92210: invalid value %s for lookAndFeel applet parameter. Defaulting to %s.**

**Cause:** An invalid value was specified for the lookAndFeel applet parameter.

**Action:** The system administrator should ensure that a valid value was specified for the lookAndFeel applet parameter.

**Appears:** Java console, applet status window

**Level:** 99

**Trigger:** None

**FRM-92211: invalid value %s for colorScheme applet parameter. Forms will use the default colorScheme for the specified lookAndFeel.**

**Cause:** An invalid value was specified for the colorScheme applet parameter.

**Action:** The system administrator should ensure that a valid value was specified for the colorScheme applet parameter.

**Appears:** Java console, applet status window

**Level:** 99

**Trigger:** None

### FRM-92220: access to system clipboard denied

**Cause:** The system clipboard is locked by some other application. This generally indicates a problem with the other application. Note: If the allowAlertClipboard applet parameter is set to 'false', this message appears only on the Java console.

**Action:** Determine which application is locking the system clipboard, and terminate it (or terminate the action that's locking the clipboard). Then, if possible, correct the application so that it does not erroneously lock the system clipboard.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

### FRM-92410: EndUserMonitoring initialization has failed. Verify that %s (specified by the applet parameter %s) is a valid URL.

**Cause:** EndUserMonitoring initialization failed, probably due to an invalid URL specified by the specified applet parameter (typically the EndUserMonitoringURL parameter). The applet parameter was ignored; EndUserMonitoring was then disabled and execution continued.

**Action:** The system administrator should ensure that a valid URL was specified for the specified applet parameter. If that does not solve the problem, examine the stack trace that accompanies this message. If the stack trace indicates a possible cause, correct it. If the problem persists, contact Oracle Support Services.

**Appears:** Java console

**Level:** 99

**Trigger:** None

### FRM-92411: EndUserMonitoring has failed, and will be disabled.

**Cause:** EndUserMonitoring was enabled, but a subsequent attempt to send a message to the EndUserMonitoring monitor encountered an unexpected Exception. EndUserMonitoring was then disabled, and execution continued.

**Action:** Examine the stack trace that accompanies this message. If the stack trace indicates a possible cause, correct it. If the problem persists, contact Oracle Support Services.

**Appears:** Java console

**Level:** 99

**Trigger:** None

### FRM-92412: failure to send EndUserMonitoring data

**Cause:** EndUserMonitoring was enabled, but a subsequent attempt to send data to the EndUserMonitoring monitor was unsuccessful. Execution continued.

**Action:** If the problem persists, contact Oracle Support Services.

**Appears:** Java console

**Level:** 99

**Trigger:** None

### FRM-92420: could not find listener class %s

**Cause:** The class specified by the formsMessageListener applet parameter could not be found. The applet parameter was ignored, and execution continued.

**Action:** The system administrator should ensure that a valid class name was specified for the formsMessageListener applet parameter.

**Appears:** Java console

**Level:** 99

**Trigger:** None

### FRM-92421: could not instantiate listener class %s

**Cause:** The class specified by the formsMessageListener applet parameter could not be instantiated. The applet parameter was ignored, and execution continued.

**Action:** The system administrator should ensure that a valid class name was specified for the formsMessageListener applet parameter.

**Appears:** Java console

**Level:** 99

**Trigger:** None

### FRM-92422: could not initialize listener class %s

**Cause:** An unexpected Error was encountered while attempting to find and instantiate the class specified by the formsMessageListener applet parameter. The applet parameter was ignored, and execution continued.

**Action:** Examine the stack trace that accompanies this message. If the stack trace indicates a possible cause, correct it. If the problem persists, contact Oracle Support Services.

**Appears:** Java console

**Level:** 99

**Trigger:** None

### FRM-92430: warning: invalid value %s ignored for parameter %s - defaulting to %s

**Cause:** The specified applet parameter (typically the asyncEventDelay parameter) did not specify a valid decimal number. A default value (5 seconds) was substituted, and execution continued.

**Action:** The system administrator should ensure that a valid decimal number was specified for the specified applet parameter.

**Appears:** Java console

**Level:** 99

**Trigger:** None

### FRM-92440: Thread %s has been interrupted while waiting for a message from the server.

**Cause:** The specified thread (in the Forms Java client) was interrupted while waiting for a message from the Forms server. This message identifies the thread, for diagnostic purposes. A fatal error subsequently occurs.

**Action:** No action is required for this message. (But action may be required for the subsequent fatal error).

**Appears:** Java console

**Level:** 99

**Trigger:** None

**FRM-92450: Thread %s has been interrupted while waiting for a dialog to appear.**

**Cause:** The specified thread was starting a dialog, and was interrupted while waiting for a dialog to appear. The wait was restarted.

**Action:** No action is required.

**Appears:** Java console

**Level:** 99

**Trigger:** None

**FRM-92460: Thread %s has been interrupted while waiting for the LOV data fetching thread to die.**

**Cause:** The specified thread had been fetching data for an LOV, but the end user accepted or canceled the LOV, so the thread requested a graceful death. But it was interrupted while the request was underway.

**Action:** No action is required.

**Appears:** Java console

**Level:** 99

**Trigger:** None

**FRM-92470: unable to load image %s for image item**

**Cause:** A requested image could not be loaded. A default image (indicating that the load failed) was substituted, and execution continued."

**Action:** If the error message specifies the name of the image file, verify that it exists, and is readable, and is in a valid image format.

**Appears:** Java console

**Level:** 99

**Trigger:** None

**FRM-92471: unable to load image %s for iconic button item**

**Cause:** A requested image could not be loaded. A default image (indicating that the load failed) was substituted, and execution continued."

**Action:** If the error message specifies the name of the image file, verify that it exists, and is readable, and is in a valid image format.

**Appears:** Java console

**Level:** Warning

**Trigger:** None

**FRM-92480: Property %s: specified value has caused %s.**

**Cause:** An attempt to set the value of an item property failed because the value was not of the proper type. The property was left unchanged, and execution continued. This error most commonly occurs when the application executes the SET_CUSTOM_PROPERTY built-in.

**Action:** Correct the value that is being passed to the SET_CUSTOM_PROPERTY built-in.

**Appears:** Java console

**Level:** 99

**Trigger:** None

### FRM-92522: forcing use of the native HTTP implementation

**Cause:** The useURLConnection applet parameter was specified as 'true' or 'yes'.

**Action:** No action is required.

**Appears:** Java console

**Level:** 99

**Trigger:** None

### FRM-92523: forcing use of the Forms HTTP(S) implementation

**Cause:** The useURLConnection applet parameter was specified as 'false' or 'no'.

**Action:** No action is required.

**Appears:** Java console

**Level:** 99

**Trigger:** None

### FRM-92530: error closing socket: %s

**Cause:** An unexpected Exception was encountered while attempting to close a socket. Execution continued.

**Action:** No action is required.

**Appears:** Java console

**Level:** 99

**Trigger:** None

### FRM-92540: reallocating output buffer for native HTTP implementation

**Cause:** The native HTTP 'write' method was requested to write a block of data larger than its current output buffer. The buffer was reallocated, and execution continued.

**Action:** No action is required.

**Appears:** Java console

**Level:** 99

**Trigger:** None

### FRM-92550: negative content-length on a response to a GET to URL %s

**Cause:** The Forms Java client issued a GET request to the Forms servlet. The result was a response with negative content-length. The fatal error FRM-92052 will subsequently appear.

**Action:** No action is required for this message. (But action may be required for the subsequent fatal error).

**Appears:** Java console

**Level:** 99

**Trigger:** None

### FRM-92551: negative response (%s) to a read on behalf of a GET to URL %s

**Cause:** The Forms Java client issued a GET request to the Forms servlet. The result was a negative response. The fatal error FRM-92052 will subsequently appear.

**Action:** No action is required for this message. (But action may be required for the subsequent fatal error).

**Appears:** Java console

**Level:** 99

**Trigger:** None

### FRM-92572: Forms HTTP connect has failed - giving up after %s attempts.

**Cause:** A Forms HTTP connect failed. The fatal error FRM-92050 (or possibly FRM-92060) will subsequently appear.

**Action:** No action is required for this message. (But action may be required for the subsequent fatal error).

**Appears:** Java console

**Level:** 99

**Trigger:** None

### FRM-92574: Forms HTTP read has failed: %s

**Cause:** An unexpected Exception was encountered while attempting a native HTTP 'read'.

**Action:** Examine the stack trace that accompanies this message. If the stack trace indicates a possible cause, correct it. If the problem persists, contact Oracle Support Services.

**Appears:** Java console

**Level:** 99

**Trigger:** None

### FRM-93110: No Forms Servlet configuration file is specified.

**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

### FRM-93111: Cannot find Forms Servlet configuration file %s.

**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

### FRM-93112: error reading Forms Servlet configuration file %s

**Cause:**  A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:**  Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

### FRM-93114: Forms Servlet configuration file %s contains invalid data.

**Cause:**  A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:**  Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

### FRM-93115: error reading configuration file FormsOIDConfig.xml via Forms Servlet.

**Cause:**  A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:**  Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

### FRM-93121: Cannot find environment variable configuration file %s.

**Cause:**  A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:**  Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

### FRM-93122: error reading environment variable configuration file %s

**Cause:**  A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:**  Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

### FRM-93124: Environment variable configuration file %s contains invalid data.

**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

### FRM-93130: No base HTML file is specified.

**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

### FRM-93131: Cannot find base HTML file %s.

**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

### FRM-93132: error reading base HTML file %s

**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

### FRM-93150: The following restricted parameters cannot be specified in the URL: %s

**Cause:** The indicated parameters were specified by the end user (in a Forms Servlet URL).

**Action:** Remove the indicated parameters from the URL and resubmit.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

### FRM-93151: Restricted characters cannot be specified in the URL

**Cause:** Invalid characters were specified by the end user (in a Forms Servlet URL).

**Action:** Remove the invalid characters from the URL and resubmit.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-93154: The Forms Servlet is not allowing new connections.**

**Cause:** The system administrator has disabled new connections to the Forms Servlet.

**Action:** No action is required.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-93156: The Forms Servlet is unable to contact the Forms Load Balancing Server at %s:%s.**

**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-93218: fatal error reading client request content**

**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-93230: fatal error creating the HTTP session**

**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-93240: Multiple Forms applications cannot share an HTTP session.**

**Cause:** Either (1) the user selected File, then New, then Browser Window (or Ctrl+N) in Internet Explorer while the servlet session was being tracked using cookies, or else (2) the Forms application was configured incorrectly.

**Action:** In case (1), select File, then New, then Browser Window (or Ctrl+N) in an Internet Explorer window that is not running a Forms application, or start a second instance of Internet Explorer. Otherwise, contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-93301: Fatal authentication error: Unable to connect to Oracle Internet Directory.**

**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-93320: unable to obtain application entity credential from CSF**

**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-93330: Fatal authentication error: User does not have proper credentials configured in Oracle Internet Directory.**

**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-93340: Session requires SSO user authentication.**

**Cause:** The session was not SSO-authenticated or had expired.

**Action:** Re-login using SSO credentials.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-93364: Cannot dynamically create resource in Oracle Internet Directory: URL specifies invalid value "%s" for the config parameter.**

**Cause:** An attempt to dynamically create an SSO resource failed because the user specified a nonexistent configuration section.

**Action:** Specify the name of a valid configuration section in the base configuration file as the value of the config parameter.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-93365: error running Forms in SSO mode**

**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

### FRM-93500: unexpected error while attempting to create the runtime process

**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

### FRM-93520: Runtime process not created: Maximum permissible number of runtime processes is exceeded.

**Cause:** The number of currently executing runtime processes has reached the limit that was set by the system administrator.

**Action:** Retry the application when the system is less heavily loaded.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

### FRM-93531: cannot create runtime process: unable to switch to working directory

**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

### FRM-93535: cannot create runtime process: unable to execute startup command

**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

### FRM-93536: cannot create runtime process: unable to create temporary logging file

**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-93543: cannot connect to runtime process: unable to get I/O streams from newly created runtime process**
**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-93550: cannot connect to runtime process: no response from newly created runtime process**
**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-93552: cannot connect to runtime process: Newly created runtime process has terminated abnormally.**
**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-93553: cannot connect to runtime process: unable to establish a socket connection to newly created runtime process**
**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-93558: cannot connect to runtime process: error reading data from newly created runtime process**
**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-93600: unexpected error while attempting to communicate with the client or the runtime process**

**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-93618: fatal error reading data from runtime process**

**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-93628: fatal error writing data to runtime process**

**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-93652: The runtime process has terminated abnormally.**

**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-93830: Test cookie set. Details:%s**

**Cause:** This is the normal output of the setcookie and setcookiesess commands.

**Action:** No action is required.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-93832: Found test cookie: %s=%s**

**Cause:** The test cookie was found.

**Action:** No action is required.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-93834: Test cookie not found.**

    **Cause:** The test cookie was not found.

    **Action:** No action is required.

    **Appears:** Java console, alert

    **Level:** 99

    **Trigger:** None

**FRM-93840: Proctest: Warning: nProcs configuration parameter is nonnumeric - using 1.**

    **Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

    **Action:** Contact your system administrator.

    **Appears:** Java console, alert

    **Level:** 99

    **Trigger:** None

**FRM-93841: Proctest: Process test starting with %s processes.**

    **Cause:** The Proctest command started.

    **Action:** No action is required.

    **Appears:** Java console, alert

    **Level:** 99

    **Trigger:** None

**FRM-93842: Proctest: Creating process %s.**

    **Cause:** The Proctest command started creating a runtime process.

    **Action:** No action is required.

    **Appears:** Java console, alert

    **Level:** 99

    **Trigger:** None

**FRM-93843: Proctest: Failure to create process %s - aborting test.**

    **Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

    **Action:** Contact your system administrator.

    **Appears:** Java console, alert

    **Level:** 99

    **Trigger:** None

**FRM-93844: Proctest: Connecting to process %s.**

    **Cause:** The Proctest command started connecting to a newly created runtime process.

    **Action:** No action is required.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

### FRM-93845: Proctest: Failure to connect to process %s - aborting test.

**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

### FRM-93846: Proctest: unexpected error in process test

**Cause:** A fatal error occurred in the Forms server, which will require the attention of your system administrator.

**Action:** Contact your system administrator.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

### FRM-93847: Proctest: Number of processes started and connected to is %s.

**Cause:** The Proctest command created the requested number of runtime processes, and successfully connected to them.

**Action:** No action is required.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

### FRM-93848: Proctest: Stopping the %s runtime processes.

**Cause:** The Proctest command reached the point in its processing where it was about to stop the runtime processes that it had previously created.

**Action:** No action is required.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

### FRM-93849: Proctest: Process test complete.

**Cause:** The Proctest command completed.

**Action:** No action is required.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

### FRM-93860: em_result=%s|000

**Cause:** This is the normal output of the status command.

**Action:**  No action is required.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-93890: Trace: invalid Xlate parameter %s**
**Cause:**  The URL that contains the trace command also specifies a parameter that is not recognized by the Xlate utlity.
**Action:**  Correct the URL.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-93891: Trace: outputClass parameter value "%s" for Xlate cannot contain a ".".**
**Cause:**  The URL that contains the trace command specifies an invalid value for the outputClass parameter.
**Action:**  Correct the URL that specifies the invalid parameter value.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-93892: Trace: PID parameter value "%s" for Xlate cannot contain a "%s".**
**Cause:**  The URL that contains the trace command specifies an invalid value for the PID parameter.
**Action:**  Correct the URL that specifies the invalid parameter value.

**Appears:** Java console, alert

**Level:** 99

**Trigger:** None

**FRM-99999: Error %s occurred.**
**Cause:**  An error occurred; the error is documented in the release notes file.
**Action:**  See the release notes file.

**Level:** 25

**Trigger:** None

# Index