Oracle® Forms Developer and Oracle Forms Services 6*i* (Patch 8)

Release Notes

Release 6.0.8.17 for Solaris

September 2001

Part No. A95267-01

Purpose Of This Document

This document notes the differences between Oracle Forms Developer and the Oracle Forms Server and their documented functionality.

Server Licensing

If you want to deploy your Oracle Forms or Oracle Graphics to the Internet or an intranet, then you need the Oracle Forms Server and its associated deployment license.

The Oracle Forms Server is included on the general Oracle Forms Developer CD-ROM for demonstration and test purposes, but is licensed separately. Contact your Oracle sales representative to get additional information on Oracle Forms Server pricing.

General Considerations

The following are general considerations for this release.

Name and Packaging Changes

Previously, Oracle Developer was two separate products: Oracle Forms Developer and Oracle Reports Developer, and Oracle Developer Server was two separate products: Oracle Forms Server and Oracle Reports Server.

In this release, the names Oracle Developer and Oracle Developer Server are obsolete. Oracle Forms Server is now called Oracle Forms Services and Oracle Reports Server is now called Oracle Reports Services.

ORACLE®

Release Numbering

In this release of Oracle Forms Developer 6*i*, most of the major components (for example, Oracle Forms Builder and Oracle Graphics Builder) have a 6.0.8 version number. Most of the subcomponents (Toolkit, for example) have a 6.0.5 version number. These are the appropriate versions of the subcomponents for this initial release.

RSF Components Include Additional Bug Fixes

Release 6*i* includes some RSF components that contain code fixes beyond those in the formal, numbered patches to the component. (These fixes or patch-levels of a component are sometimes referred to as one-offs.)

Specifically, Release 6*i* includes fixes for bugs 1063571, 1063104, 1028960, 1049171, and 1040536. These bug fixes affect the following components: SQLNET, RDMBMS, NLS, and PL/SQL.

Known Limitations

The following are known limitations in this release.

Co-existence in the same $ORACLE_HOME with Previous Releases

Oracle Developer 1.6.1 and Oracle Forms Developer 6*i* can co-exist in the same $ORACLE_HOME.

Oracle Developer 2.1 and Oracle Forms Developer 6*i* can co-exist in the same $ORACLE_HOME.

Oracle Developer 1.6.1 and Oracle Developer 2.1 *cannot* co-exist in the same $ORACLE_HOME.

Oracle Developer 6.0 and Oracle Forms Developer 6*i cannot* co-exist in the same $ORACLE_HOME.

Non-Alphanumeric Characters in Database Objects Names

Oracle Forms Developer 6*i* cannot support tables or columns names that contain non-alphanumeric ASCII characters.

Motif Patch on Solaris

Motif runtime library patch 103461-07 is the minimum patch level for Oracle Forms Developer and Oracle Forms Server 6*i* on Solaris 2.5.1.

Motif runtime library patch 105284-20 is the minimum patch level for Oracle Forms Developer and Oracle Forms Server 6*i* on Solaris 2.6.

Motif runtime library patch 107081-19 is the minimum patch level for Oracle Forms Developer and Oracle Forms Server 6*i* on Solaris 7.

Precompilers Compatible With This Release

If you are developing user-exits for Oracle Forms Developer 6*i* using the Oracle Precompilers, then use the versions of the Precompilers that are bundled with the Oracle 8.0.6 release.

ORAINFONAV_DOCPATH Environment Variable on UNIX

On UNIX platforms, the ORAINFONAV_DOCPATH environment variable should be set to the location of the online manuals. The default value for this environment variable is set to the location of the American English version of the online manuals (`$ORACLE_HOME/doc60/admin/manuals/US`). If you are interested in the Japanese version of the online manuals, you should set the ORAINFODOC_DOCPATH environment variable to: `$ORACLE_HOME/doc60/admin/manuals/JA.`

Oracle File Packager Not Included

The Oracle File Packager (mentioned in the Oracle Forms Developer documentation) is not included as part of this 6*i* product.

Sequence of Installing in the Same ORACLE_HOME

If you choose to install both Oracle Forms Developer 6*i* and the Oracle Database Server 8.0.6 in the same ORACLE_HOME, then you should install the Database Server first, and then install Oracle Forms Developer. This sequence will prevent library linking conflicts.

Path Requirement for Multiple ORACLE_HOMEs

Multiple ORACLE_HOMEs is not completely supported, but a future patch release will enable this support.

In order for the Oracle Forms CGIs (ifcgi60.exe or rwcgi60.exe respectively) to run successfully on NT when there are multiple ORACLE_HOMEs, it is imperative that the `%ORACLE_HOME%\bin` directory for Forms be in the machine's PATH environment setting, ahead of any other `<ORACLE_HOME>\bin` directories.

This path setting has to have been in effect when the machine was last booted, in order for it to be seen by the CGI (assuming that whatever Web listener the user is using is being run as a NT service, which is usually the case).

If these conditions are not met, then there are two possible consequences:

1. The CGIs might fail to run because one or more DLLs on which they depend, such as core40.dll, cannot be found. This occurs if the `%ORACLE_HOME%\bin` directory is not present in the PATH.

2. The CGIs might crash because an incompatible version of core40.dll is being used. This can occur in the case of multiple ORACLE_HOMEs if an

ORACLE_HOME other than that into which Oracle Forms or Oracle Reports was installed appears in the PATH before the Oracle Forms or Oracle Reports ORACLE_HOME).

Of these, the first situation should not normally be a problem because the installation process always adds `%ORACLE_HOME%\bin` to the PATH and instructs the user to reboot the machine if any of the CGIs was installed.

The second situation is known to occur if you install Oracle Forms into one ORACLE_HOME, then install OEM into a separate ORACLE_HOME (which is in fact mandatory). This second ORACLE_HOMEs bin directory is added into the machine's PATH environment setting, by the installation process, ahead of the first one (for example `c:\orant2\bin;c:\orant1\bin;c:\winNT\system32`). Once the user reboots, this causes the Oracle Forms CGI to fail.

Workarounds:

1. Install OEM (into its own separate ORACLE_HOME) before installing Oracle Forms (into a different ORACLE_HOME). Then reboot. This works because the last ORACLE_HOME installed into is left as the DEFAULT ORACLE_HOME.

2. Use the ORACLE HOME SELECTOR to ensure that the ORACLE_HOME where Oracle Forms was installed is the DEFAULT ORACLE_HOME. The "ORACLE HOME SELECTOR" may be found in the Oracle for Windows NT program folder).

3. Manually correct the PATH setting and reboot.

Misleading Installation Message: Acrobat Reader Available

After installing components from the asset manager, you might see the following message:

```
Installation Successful. Some of the products you have installed
require the Adobe Acrobat Reader. If you do not already have this
installed on your machine, you should select it from the 'Products
available' list.
```

However, the Acrobat Reader is not in this particular list.

Acrobat can be obtained from http://www.adobe.com.

Misleading Installation Message: Folder Moved

When Release 6*i* is installed, the following message may appear:

```
Oracle Developer 6.0 D:\winnt\Profiles\All Users\Start
Menu\Programs\oracle\Oracle Developer 6.0 is not accessible. The
folder was moved or removed.
```

Ignore this message. Press **OK**. There is no impact on the product functionality.

Database Admin Scripts Require OTB

The Database Admin build and drop scripts will fail if they do not find the Oracle Translation Builder SQL scripts. This problem can be avoided by first explicitly installing the Oracle Translation Builder from the product CD.

Additional Step Required for Installation on UNIX

In order for the Oracle Forms Developer help system to work properly on a UNIX system, you must set the ORACLE_AUTOREG variable to the value `$ORACLE_HOME/guicommon6/tk60/admin`. (The help system requires the autoprefs.oar file, which is the toolkit automation registry.)

TEMPLATES Directory Is Missing

In the manual *Oracle Forms Developer and Oracle Reports Developer: Guidelines for Building Applications*, references are made to TEMPLATES. However, the TEMPLATES directory is not included with the product.

Use a different method to invoke the Oracle Installer.

Forms Listener Requirement

The Oracle HTTP Listener (powered by Apache) is the recommended listener for Oracle Forms Server.

If you are going to use the WebDB listener with the Oracle Forms Server, WebDB 2.2 or higher is required.

Problem Connecting to Oracle 8.1.6 Server

Problem: You might not be able to connect to the 8.1.6 database from any of the Oracle Forms Developer products using operating system authentication. (This problem is due to bug 1139334 in the database server.)

Workaround: This issue is resolved in the 8.1.6.1 and 8.1.7 database server releases.

Problem Connecting to Oracle 8.0.6 Server for Oracle Forms Graphics

Problem: Forms-Graphics integration does not work correctly when the connection to a local Oracle 8.0.6 database server is made with only a userid and password (for example, scott/tiger).

Workaround: Define the local database in the tnsnames.ora file. Then provide a complete connect string when connecting to the database. (For example, scott/tiger@local806.)

Custom Install of Wallet Manager Requires Net8 Assistant

Wallet Manager requires Net8 Assistant in order to launch successfully.

In the typical, default install of Forms Developer, both these components will be installed for you.

However, if you choose a custom install, where you select specific components, and you install Wallet Manager, then you also need to install Net8 Assistant.

Integration with Oracle Repository

Oracle Forms Developer can be integrated with Oracle Repository.

This 6*i* release is shipped with a d2sc plug-in (similar to those provided for PVCS, Clearcase, and other products) that allows integration with Repository for source control management. Users will be able to check in and check out FMBs, MMBs, and so forth, and can then use Repository's dependency tracking and other advanced functionality. (See the Repository documentation for more information.)

This initial feature establishes the potential for enhanced levels of integration in future Forms Developer releases.

Client Platform Certifications

For the latest information regarding supported client platforms and their support level, consult the current version of the Client Platform SOD, available from the Oracle Technology Network (http://technet.oracle.com) website or your Oracle support representative.

Use Caution If Deinstalling Intersolve Drivers

Do not deinstall Intersolve drivers (ODBC) from an Oracle_Home if other products on another Oracle_Home on that machine use them. Deinstalling the drivers from one Oracle_Home will remove them from the entire system.

Forms Considerations

The following are Forms considerations for this release.

What is New in Forms6*i* Patch 8?

Support envFile and workingDirectory parameters in formsweb.cfg. The environment configuration file and the current working directory for the runtime process can be set in the formsweb.cfg file (the FormsServlet configuration file). This will make it easier to have multiple configurations with different working directories and different sets of environment variables.

For more information, see CONFIGURATION ENHANCEMENTS in *Forms6i Patch 8: Oracle Forms Listener Servlet for Deployment of Forms on the Internet*.

Look for baseHTML files and envFile in same directory as formsweb.cfg. The values of the baseHTML and envFile parameters can be given as simple file names with no path (for example, envFile=default.env). In cases where the path is not identified in the name, Forms will look for the baseHTML and envFile in the same directory as the Forms Servlet configuration file (formsweb.cfg).

For more information, see CONFIGURATION ENHANCEMENTS in *Forms6i Patch 8: Oracle Forms Listener Servlet for Deployment of Forms on the Internet*.

Alteration in the default formsweb.cfg file. The following values in the default Forms Servlet configuration file (formsweb.cfg), which is installed when Forms is first installed, will be altered:

```
baseHTML=base.htm
baseHTMLJInitiator=basejini.htm
baseHTMLIE=baseie.htm
envFile=default.env
workingDirectory=
```

For more information, see CONFIGURATION ENHANCEMENTS in *Forms6i Patch 8: Oracle Forms Listener Servlet for Deployment of Forms on the Internet*.

Avoid need to configure PATH (or LD_LIBRARY_PATH) for the servlet engine. Particular PATH or LD_LIBRARY_PATH settings will not be required when starting the servlet engine.

For more information, see CONFIGURATION ENHANCEMENTS in *Forms6i Patch 8: Oracle Forms Listener Servlet for Deployment of Forms on the Internet*.

FormsServlet baseHTML initialization parameters no longer required. The baseHTML, baseHTMLie, and baseHTMLJInitiator Forms Servlet parameters are no longer required. Instead, they should be specified in the default (initial) section of the Forms Servlet configuration file (formsweb.cfg).

For more information, see CONFIGURATION ENHANCEMENTS in *Forms6i Patch 8: Oracle Forms Listener Servlet for Deployment of Forms on the Internet*.

Simplified servlet configuration requirements. Modified, simpler configuration process.

For more information, see BASIC CONFIGURATION and ADVANCED CONFIGURATION in *Forms6i Patch 8: Oracle Forms Listener Servlet for Deployment of Forms on the Internet*.

What was New in Forms6*i* Patch 7?

The following changes were made for Forms6*i* Patch 7.

Hide User/Password. If the user id is specified in the configuration file (formsweb.cfg), the HTML page generated by a Forms session using the Forms Servlet (or CGI) will no longer display the userid value, including the password, in the HTML source.

Prior to Forms6i Patch 7, if the user id is specified in the configuration file (formsweb.cfg), a Forms session using the Forms Servlet (or CGI) generates a page where the entire userid value, including the password, is visible in the HTML source. This visibility created an obvious security issue. The only alternative was to not specify a userid value or password, in which case Forms would prompt the user for the userid or password.

With Patch 7, the userid parameter value is not included in the HTML generated by the Forms Servlet, provided the Forms Servlet is used in conjunction with the Listener Servlet. It will not work when using the Forms CGI or static HTML pages or when using the Forms Listener.

Fixed Bug Enhancement - Retry Response. The following enhancement to the Listener Servlet and Forms client improves system throughput in situations of high load.

Previously, when the Apache (iAS) Web listener received a request, it passed the request to a faulty servlet engine (JServ process) if the intended servlet engine was busy. This occurrence, called session migration, caused the Forms session to terminate abruptly, causing possible loss of data.

With Patch 7, instead of the listener servlet sending a request to a faulty servlet engine when the intended one is busy, the listener servlet will detect a session migration condition and send a "session migrated" retry response back to the client. In some cases, the JServ process is not busy, but rather has stopped running. In either case, if the JServ process is busy or has stopped running, the client will wait 500 milliseconds then attempt to resend the request, along with the request data. The client will retry up to 10 times before reporting a fatal error and aborting the Forms session.

You must do the following for this enhancement to work:

•Specify the user/password@database using a parameter called "userid" (not case-sensitive). This is already done if you are using the default base HTML

files, which are provided when Forms is installed. They contain syntax like userid=%userid%.

•Use the FormsServlet and ListenerServlet.

•Specify the serverURL value using a parameter called "serverURL" (not case-sensitive). Again, this is already done if you are using the configuration files provided with Forms.

Enhanced Single Sign-On (SSO) Support. Forms Services applications running in a single sign-on environment (using Oracle Login Server) require a special logon format that consists of name-value pairs separated by ampersands (&). Prior to patch 7, users had to adapt to this special logon format by modifying the base html files, which were read by the Forms Servlet.

With patch 7, the default Forms userid format can be changed by only modifying the formsweb.cfg file.

To use enhanced single sign-on support:

1. Change the logon mask to a format like `username=value&password=value&database=value`, by specifying the following entry in the formsweb.cfg file: `userid=%user%/%password%@%database%`. This can be done in a separate configuration section. For example:

   ```
   [myapp_sso]
   ```

   ```
   userid=%user%/%password%@%database%
   ```

   Note: The names between % (user, password, and database) can be changed as long as they match what is later used in the URL.

2. Go to Logon Server Adminstration page, which is accessible from Oracle Portal. Configure your Forms application as a new external application. Make sure you specify:

   •The URL used to invoke the application (for example, http://myserver/servlet/f60servlet).

   •The field names of the logon parameters used to give the username and password.

   •Extra logon parameter (displayed to the user or with a suitable default value), such as the configuration parameter and the database connect information. The name of the database connect parameter is whatever you specify in the formsweb.cfg file.

   For example, to define a name "Source" for the database connect parameter, specify the database connect parameter in the formsweb.cfg file to read: userid=%name%/%password%@%Source% .

• The "Type of Authentication Used" as POST. This means logon parameters will be passed to the Forms Servlet as URL parameters using the POST method, so they will not be visible to users in the browser's location.

What was New in Forms6*i* Patch 6?

The following changes were made for Forms6*i* Patch 6.

Enhanced Network Resilience. Currently in Oracle Forms 6*i* patch 5, if a network failure lasts more than a few seconds, then the Oracle Forms session is lost and you receive the following error message, which results in the loss of any unsaved information:

"FRM-92100: Your connection to the Server has been interrupted. This may be the result of a network error or a failure on the Server. You will need to reestablish your session."

For patch 6 (as a fix to bug 1762055), when using the listner servlet, the client has the ability to distinguish network failures from server (Oracle Forms runtime) failures, and in the case of a network failure, attempts to reestablish the network connection. The client tries to reconnect as many times as you specify in the <networkRetries> parameter, and for each attempt, a message such as the following is displayed in the right-hand side of the status bar:

"2/30"

where 2 indicates the second attempt and 30 is the <networkRetries> value.

If the client is unable to talk to the server after retrying the number of times specified in the <networkRetries> parameter, it aborts the Oracle Forms session and displays the following message:

"FRM-92102: A network error has occurred. The Forms Client has attempted to reestablish its connection to the Server <number> times without success. Please check the network connection and try again later."

If there is a failure in the server (that is, the Forms runtime process unexpectedly terminates or someone has stopped it), the client immediately aborts the Oracle Forms session and displays the following message:

"FRM-92101: A failure has occurred on the Server. You will need to reestablish your session."

Using the Enhanced Network Resilience Feature

You should edit your base HTML files to include the new "networkRetries" applet parameter. You should then add this parameter to the default section of the formsweb.cfg file, by adding a line like the following:

```
networkRetries=30
```

If you do not take these steps, a default value of 0 is used, causing the client to behave as before (that is, not reattempt connection) in the case of a network failure. However, you can edit the formsweb.cfg file to change this setting to retry more or less times, according to your particular network characteristics and needs.

Avoid Use of Fixed Port Numbers. Previously, as a temporary workaround, the Oracle Forms Listner Servlet communicated with the Oracle Forms Server Runtime processes using fixed port numbers. As of Oracle Forms6*i* patch 6, the maxPorts and startPort initialization parameters are obsolete, as the issue of port numbers has been resolved.

Improved Performance When Running Forms Applications Under JInitator in HTTPS Mode Using the Listner Servlet. Previously, JInitiator's HTTPS implementation did not use the HTTP keep-alive option. Now that keep-alive has been implemented (available in JInitiator 1.1.8.11 and higher), the client does not have to reconnect every time it makes a URL request. Consequently, this eliminates the need for an SSL handshake every time the thin client communicates with the server.

For more information regarding Oracle Forms6*i* patch 6, see the Oracle Forms Developer Release 6*i* Patch 6 White Paper available on OTN (`http://otn.oracle.com/products/forms`).

Introducing the Forms Listener Servlet

Previous releases of Forms use the Forms Listener to manage the initial connection from a client. However, the Forms Listener has a shortcoming when used for HTTP or HTTPS deployment of Forms applications over the Internet in that the machine and port on which the Forms Listener is running are exposed at the firewall, which creates a potential security risk.

Since patch 4, Oracle Forms 6*i* includes the new Oracle Forms Listener Servlet, which improves upon the functionality of the Oracle Forms Listener by providing more secure connections when deploying over the Internet because it does not have to run on a machine that is exposed at the firewall.

Note: It is recommended that you use the Forms Listener Servlet when deploying applications using HTTP and HTTPS. The Forms Listener is still available for direct socket connections, and still supports HTTP and HTTPS connections.

Prerequisites *i*AS is required.

What is the Forms Listener Servlet? The Forms Listener Servlet is a Java servlet that runs on a web server equipped with a servlet engine, such as the Oracle9*i* Application Server.

The Forms Listener Servlet manages:

- The creation of a Forms Server Runtime process for each client

- Network communications between the client and its associated Forms Server Runtime process

Why Should I Use the Forms Listener Servlet?   The Forms Listener Servlet was designed to allow a more robust and standard deployment of Forms applications on the Internet. When compared to the Forms Listener, the Forms Listener Servlet provides the following benefits:

- **Broader range of firewalls and proxies supported.** Because the client browser now communicates with the web server, this architecture supports any firewall or proxy that can work with a standard servlet using servlet sessions.

- **No protocol restriction (HTTP/1.1 or HTTP/1.0).** Although the use of HTTP/1.1-compliant proxies provides better performance, this architecture works well with HTTP/1.0-compliant proxies, too.

- **No extra process to manage.** Because this architecture eliminates the need for the Forms Listener process, the administrative tasks to start and stop the Forms Listener process are also no longer required.

- **No specific certificate to purchase/manage for SSL deployment.** In the case of deployment using SSL (secure sockets layer), the HTTPS connection occurs between the client browser and web server. Therefore, there are no specific security configuration requirements at the Forms Server level.

- **Standard load balancing support.** This architecture allows you to use standard load balancing techniques, such as hardware-based load balancing, reverse proxy, and standard Apache Jserv load balancing.

- **Internet Explorer 5.0 with native JVM support.** In addition to working with Oracle JInitiator, this architecture supports the use of Internet Explorer 5.0 with native Microsoft JVM for Internet deployment using HTTP and HTTPS connection modes.

Where Can I Get More Information?   For a detailed description of the Forms Listener Servlet, as well as installation and configuration information, see the white paper titled *Oracle9iAS Forms Services, Forms6i Patch 6: Forms Listener Servlet for Deployment of Forms on the Internet* May 2001.

Check the Oracle Technology Network web site at `http://otn.oracle.com` for updates to the white paper.

Effect of Fill Patterns on Performance

Assigning a fill pattern to a large Forms object may degrade runtime performance on the web. Objects are filled in 64x64 pixel increments. Thus, objects larger than

64x64 require multiple operations. (For example, painting a window of size 800x600 would require 120 such operations.)

Recommendations: Avoid using fill patterns for large objects unless doing so is a significant user requirement. Also, minimize the number of different patterns. (Each pattern used requires 4K of memory.)

Undocumented Error Messages (FRM-99999)

Any Forms error message not documented in the online help will display a generic message:

FRM-99999 Error FRM-nnnnn occurred, please consult the release notes.

The details of these error messages are documented in the following subsections.

Error Messages 1412 and 1413

FRM-99999, Error 1412.	An attempt was made to set scrollbar position on a block that has no scrollbar.
FRM-99999, Error 1413.	An attempt was made to get scrollbar position on a block that has no scrollbar.

Error Messages 10905-10909

FRM-10905: The following return values are invalid.

Cause: The invalid return values must be corrected before you can leave this page of the wizard.

Action: Enter a valid return value. Return values must be one of the following:

1. A fully qualified Item name (<BLOCK_NAME>.<ITEM_NAME>).

2. A Form Parameter (PARAMETER.<PARAMETER_NAME>).

3. A global PL/SQL variable (GLOBAL.<VARIABLE_NAME>).

FRM-10906: One or more LOV columns has a negative width.

Cause: One or more of the columns in the table has a negative width.

Action: Ensure that no columns in the table have a negative width.

FRM-10907: The LOV size or position has a negative value.

Cause: One or more of the attributes of the LOV size and position is negative.

Action: Ensure that none of the LOV size or position attributes is negative.

FRM-10908: Number of rows retrieved is less than or equal to zero.

Cause: The number of rows retrieved is less than or equal to zero.

Action: Enter a value greater than zero for the number of records retrieved.

FRM-10909: Old Style LOVs cannot be modified using the LOV wizard.

Cause: The LOV Wizard was invoked on an Old-Style LOV.

Action: Create a new LOV based on a Record Group.

Error Messages 13009-13010

FRM-13009: The JavaBean does not implement the IView interface.

Cause: For Forms to use this JavaBean, it must implement Oracle Forms IView interface.

Action: Implement Oracle Forms oracle.forms.ui.IView Interface.

FRM-18010: An unknown error occurred when attempting to process the JavaBean.

Cause: For Forms to use this JavaBean, it must be able to locate the specified JavaBean and instantiate it.

Action: Ensure that the specified JavaBean and supporting classes are installed correctly. Retry the JavaBean after restarting Form Builder.

Error Messages 18114-18121

FRM-18114: FORMS60_JAVADIR not set.

Cause: For Web Preview from the Builder to work the Registry variable FORMS60_JAVADIR must point to the location that contains the Forms Java files. This variable should have been set by the Oracle Installer when Oracle Forms Developer was installed. A typical value for this variable is c:\orant\forms60\java.

Action: Create or update the registry variable on NT, FORMS60_JAVADIR, and set its value to the location that contains the Forms Java files.

FRM-18115: CLASSPATH variable not set.

Cause: For forms to run on the Web the environment variable CLASSPATH must point to the location that contains a valid Java installation. This variable should have been set by the Oracle Installer when Oracle Forms Developer was installed.

Action: Create or update the environment variable CLASSPATH and set its value to the location that contains a valid Java installation.

FRM-18116: The CLASSPATH does not contain a reference to Forms.

Cause: For Forms Server to work, the environment variable CLASSPATH must include an entry that points to a location containing the Java files required by Oracle Forms. An entry should have been added to the existing CLASSPATH variable by the Oracle Installer when Oracle Forms Developer was installed. A typical value for this variable is c:\orant\forms60\java.

Action: Create or update the environment variable CLASSPATH and set its value to the location that contains the Java files required by Oracle Forms.

FRM-18117: The preferences contains a reference to a non-existent HTML file.

Cause: In the 'Runtime' Preferences dialog an HTML file has been specified, but that HTML file does not exist, or the location specified does not exist.

Action: In the 'Runtime' Preferences dialog either remove the reference to the offending HTML file, so that the default HTML file is used, or specify an HTML file that exists. Alternatively place the HTML file in the location specified.

FRM-18118: Javai.DLL does not exist.

Cause: For Forms Server to work in the Microsoft Windows environment, e.g. Windows NT, the DLL 'javai.dll' must exist and be in the %ORACLE_JDK%\bin directory, where %ORACLE_JDK% contains a valid Java installation.

Action: Determine whether javai.dll exists in the %ORACLE_JDK%\bin directory, and if necessary, reinstall the Oracle JDK.

FRM-18119: ORACLE_JDK variable not set

Cause: For Forms Server to work, the environment variable ORACLE_JDK must point to the location that contains a valid Java installation. This variable should have been set by the Oracle Installer when Oracle Forms was installed.

Action: Create or update the environment variable ORACLE_JDK and set its value to the location that contains a valid Java installation.

FRM-18120: libjava.so does not exist.

Cause: For Forms Server to work in the Solaris environment, a valid JDK installation must exist and be in the path.

Action: Determine whether a valid JDK exists in the path and if necessary, reinstall the JDK.

FRM-18121: The JavaBean does not implement the IView interface.

Cause: For Forms to use this JavaBean, it must implement Oracle Forms IView interface.

Action: Implement Oracle Forms oracle.forms.ui.IView Interface.

Forms Environment Variable Added

A new environment variable is available:

FORMS60_WEB_CONFIG_FILE

You can use it to specify the full path and name of the Forms CGI configuration file.

For example: FORMS60_WEB_CONFIG_FILE=c:\temp\myfile.cfg

The default value is <ORACLE_HOME>\forms60\server\formsweb.cfg

Java Keybindings API Provided

To provide consistent, configurable, and performant keybindings in Forms, a keybinding API is provided. This API first downloads keybindings to the client. The API can then query the key sequence and return the corresponding form action or query the form action and return the corresponding key sequence. See the following subsections for details.

FormAction.java

Standard form actions are defined in FormAction.java. These form actions are static instances of the class FormAction. For example, the Next Field form action can be addressed as FormAction.FA_NEXT_FIELD.

The constants for the various form actions in FormAction.java are obtained from frmweb.res. The following shows the beginning of the keybinding constants in FormAction.java.

```
public class FormAction extends Property
{
    /*
    ** The constants (or id's) for various forms actions have
    ** been obtained from frmweb.res. These id's will not change.
    ** New forms actions will be assigned new id's.
    */
    public static final FormAction FA_NEXT_FIELD      = new
                                                        FormAction(1);
    public static final FormAction FA_PREVIOUS_FIELD  = new
                                                        FormAction(2);
    public static final FormAction FA_CLEAR_FIELD     = new
                                                        FormAction(3);

    .
    .
    .
```

The following is the list of all the static constants for keybindings included in the FormAction.java class.

```
FA_NEXT_FIELD
FA_PREVIOUS_FIELD
FA_CLEAR_FIELD
FA_UP
FA_DOWN
FA_SCROLL_UP
FA_SCROLL_DOWN
FA_EDIT
FA_RETURN
FA_LIST_OF_VALUES
```

```
FA_HELP
FA_EXIT
FA_SHOW_KEYS
FA_COMMIT
FA_NEXT_PRIMARY_KEY
FA_CLEAR_RECORD
FA_DELETE_RECORD
FA_DUPLICATE_RECORD
FA_INSERT_RECORD
FA_NEXT_SET_OF_RECORDS
FA_NEXT_RECORD
FA_PREVIOUS_RECORD
FA_CLEAR_BLOCK
FA_BLOCK_MENU
FA_NEXT_BLOCK
FA_PREVIOUS_BLOCK
FA_DUPLICATE_FIELD
FA_CLEAR_FORM
FA_ENTER_QUERY
FA_EXECUTE_QUERY
FA_DISPLAY_ERROR
FA_PRINT
FA_COUNT_QUERY
FA_UPDATE_RECORD
FA_FUNCTION_0
FA_FUNCTION_1
FA_FUNCTION_2
FA_FUNCTION_3
FA_FUNCTION_4
FA_FUNCTION_5
FA_FUNCTION_6
FA_FUNCTION_7
FA_FUNCTION_8
FA_FUNCTION_9
FA_LIST_TAB_PAGES
```

KeyBinder.java -- Using the API

Use the KeyBinder API to download keybindings and get a requested form action
or key sequence. The following is an overview of the process:

1. Execute downloadKeyBindings() in your start-up code.

2. Use isKeyBindingsAvailable() to test if the keybindings are available on the
    client.

3. If the keybindings are available, use getKeySequence(FormAction action) to
    get the key sequence for the requested form action,

    or

use getFormAction(KeyEvent event) to get the form action for the requested key sequence.

Use the following KeyBinder.java API methods:

```
public synchronized static void downloadKeyBindings()
```

If the keybindings have not been created on the client, this method requests that the server send the bindings. This is not a blocking call, because we do not want to incur a round-trip for getting keybindings from the server. The next time a round- trip occurs, the server will send the necessary information.

```
public synchronized static Hashtable getKeyBindings()
```

This method returns a cloned table of all the keybindings.

```
public synchronized static boolean isKeyBindingsAvailable()
```

This method checks if the keybindings are available on the client.

```
public synchronized static KeyEvent getKeySequence(FormAction action)
```

This method gets the key sequence corresponding to a requested form action.

```
public synchronized static FormAction getFormAction(KeyEvent event)
```

This method gets the form action corresponding to a requested key sequence.

Example of KeyBinder.java API

The following example shows programming pertinent to the use of the KeyBinder.java API to get the form action from the keys pressed.

```
KeyBinder.downloadKeyBindings(); // get bindings from server
```

Then in a text field's processKeyEvent(KeyEvent e) implementation, this handles events:

```
// If we have key bindings, see whether this event maps to
// one that we care about.
// If it does, record which event it is with the name of
// the action that goes with it.
// Many actions can be invoked by buttons, key presses,
//   and/or menu selections...
if (KeyBinder.isKeyBindingsAvailable())
{
String       actionName;
FormAction   fact = KeyBinder.getFormAction(e);
    if (fact == (FormAction) null)
    {
    // if no binding found, issue message.
    system.out.printer("No binding found.");
```

```
          }
          else if (fact == FormAction.FA_LIST_OF_VALUES)
          actionName = "LOV";
          else if (fact == FormAction.FA_EDIT)
          actionName = "FLDEDIT";
          else if (fact == FormAction.FA_EXIT)
          actionName = "CANCEL";
          else
              {
              }
          .
          .
          .
```

On Windows, Forms Server and Web Previewer Require NT

The Forms Server component will not run on Windows 95 or 98. It runs only on
the NT version of Windows. Also, because the Forms web-previewing functionality
requires the Forms Server, the Web Previewer is available only on Windows NT
(not on Windows 95 or 98).

Server Does Not Work If Host and Machine Name Differ

Problem:
If the TCP/IP host name and the Windows machine name are different, the Forms
Server Service will not work.

Workaround:
Re-enter your host name by following the path HKEY_LOCAL_MACHINE ->
SYSTEM -> CurrentControlSet ->Services -> OracleFormsServer-Forms60Server.
Then double-click on 'Host' and enter your host name.

Requirements for Using Run_Product on the Web

If you use Run_Product in a form to run a report and want to display the output in
the browser window, it is necessary to do the following: On Windows NT, remove
the TMP user environment variable. On Unix, remove the TMPDIR environment
variable.

Database Commands Available on Separate Start Menu

In this release, the database commands Build, Drop, Grant, and Revoke are listed
on their own Start Menu, entitled "Oracle Forms 6*i* Admin," which is separate from
the Start Menu for the main Forms product.

Using Index-Only Tables

In order to use an index-only table in Forms, the block's Key Mode should be set to either Updateable or Non-Updateable. Also, a Primary Key should be nominated.

Forms Server Needs to Access the Wallet File

If you register the Forms Server as an NT service and configure it with https as the connect mode, launching Forms applets may fail because the service is not able to access the wallet file on the server.

To work around this problem, you should either run the Forms Server from the command line (that is, not as a service), or, if you do want to run the Forms Server as a service, do the following:

1. In the services dialog, select the Forms Server service and click the Startup button.

2. In the "Log On As:" section, select the "This Account" radio button.

3. In the "This Account" field, enter the account name under which Forms was installed, and fill in the password information for that account.

4. Click the OK button and restart the Forms Server service.

Netscape Users Prompted to Save Report HTML File

The installation process for Forms Server creates a registry entry named FORMS60_REPFORMAT with value of "HTML" (upper case).

If a user is running a Forms applet in a Netscape browser that launches a report, rather than having the report open in the browser window, the user will be prompted to save the report HTML file.

You can avoid this situation by setting the FORMS60_REPFORMAT registry entry to "html" (lower case).

Forms-Reports Integration Requires that TMPDIR Not Be Set

In order for Forms-Reports integration to work on the web on a Unix system, the variable TMPDIR must not be set. Unset this environment variable before starting the Forms listener.

Changing Font Causes Problem in Forms

Problem:
When using Form Builder, if you draw a text object and subsequently change its font, then the copy, paste, and delete keys will stop working.

Workaround:
None.

Using Oracle Trace Collection

Oracle Trace Collection is a beta feature in Forms Server Release 6*i*.

CAB Support in IE and Java Security

The Forms 6*i* CAB file (f60all.cab) supports default Java security settings (high, medium, and low) in the browser. Because it provides the most secure environment for Java code execution, the high security setting is recommended. If using Custom security settings, the Unsigned content flag must be set to either enabled or run in sandbox.

Java Importer

The Java Importer does not function when the web preview mode of the builder is used. The Java Importer functions as documented in client/server runtime mode and in a deployment environment with Oracle Forms Server. The Java Importer requires a separate, manual installation of JDK 1.2.2 before use.

OEM and Forms Internal Job

In the Oracle Enterprise Manager (OEM) console, when creating a job under the Developer server node, there may be items listed in the Tasks tab that are labeled "Forms Internal Job." These are Forms internal tasks and must not be scheduled in an OEM job.

Load Balancing Server Trace Log

A load balancing server trace log file is created when the load balancing server--d2ls60--is started with a trace level greater than 0. The default name for the log file is d2lslog.txt. When the d2ls60 load balancing server is started from the command line, the default location for the d2lslog.txt log file is the directory from which the load balancing server was started. If the load balancing server is started as an NT service, the default location for the d2lslog.txt log file is <WINNT>\system32.

Out of System Resources: file descriptors (Solaris)

On Solaris, the f60srvm process can spawn a maximum number of f60webmx processes that is 5 fewer than the system's descriptors value setting. When f60srvm cannot spawn a new f60webmx process because of the descriptor limit, the following error is thrown:

Out of System Resources: file descriptors, [errno=nn].

To increase the maximum number of f60webmx processes that can be spawned, increase the number of descriptors.

LD_LIBRARY_PATH (Solaris)

Append the following to the end of LD_LIBRARY_PATH environment variable before installing Forms and Reports 6*i*:

$ORACLE_HOME/6iserver/network/jre11/lib/sparc/native_threads

If this value is not set, Reports linking may fail during installation.

Note: On HP-UX, replace

```
$ORACLE_HOME/6iserver/network/jre11/lib/sparc/native_threads
```

with

```
$ORACLE_HOME/6iserver/network/jre11/pa_risk/native_threads
```

New FORMS60_TRACE_PATH environment variable

The FORMS60_TRACE_PATH environment variable specifies the location of dump files produced as the result of a crash of any of the Forms runtime executables. The dump files contain diagnostic information about events at the time the process crashed.

To set in UNIX:

```
setenv FORMS60_TRACE_PATH /tmp
```

To set in NT:

```
set FORMS60_TRACE_PATH=c:\temp
```

If FORMS60_TRACE_PATH is not set and a crash occurs, Forms will attempt to place the dump file in the directory from which the executable was called.

If you are using the Forms server or servlet, then setting this environment variable is recommended. If you are using the Forms servlet with iAS, set this environment variable in the jserv.properties file or the default.env file.

Note: If Forms server logging is switched on, diagnostic information is included in the forms server log file rather than in directory specified by FORMS60_TRACE_PATH.

Forms/OEM Extension Issues

Fixes for the following bugs are available on ARU/Metalink:

Bug number	Description
1286040	NLS: MULTIBYTE CHARACTERS IN FORMS LISTENER PROCESS LOG ARE GARBAGED

Bug number	Description
1544477	IAS V1.0.2.1.0 M1A - EVENT DEREGISTERING GIVES ERROR VD-1525
1549369	CANNOT DEREGISTER FORMS LISTENER UPDOWN EVENT ...
1554211	FORMS EVENT FAILS WHEN DB IS 8.1.7
1562887	MULTIPLE SIDS WILL NOT MAKE THE FORMS LISTENER STARTS FROM OEM
1383239	OEM: STARTUP JOB RETURNS "COMPLETED" STATUS IF IT FAILS
1479367	NMIFORMS.TCL IS LOOKING FOR AN ORACLE_ HOME ENVIRONMENT VARIABLE IN A WRONG WAY

Running Forms Applications on the Web Using an Authenticating Proxy

In Forms 6i Patch 5, support was added to run Forms applications on the web using an authenticating proxy. 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 detect whether the user has logged on (i.e. been authenticated) by setting a cookie. The cookie is sent in all subsequent network requests to avoid further logon prompts.

To run Forms applications using an authenticating proxy, Forms 6i patch 5 (or later) must be installed, and you must be running the Listener Servlet (rather than the Forms Listener).

If users are running Internet Explorer with the native Microsoft Java VM or Internet Explorer with JInitiator, then no other configuration is required.

However, if users are running Netscape with JInitiator, then you need to perform additional configuration steps. These steps are necessary to ensure that the authentication cookie gets sent with all requests to the server. The basic requirement is that every URL that JInitiator has to access (those for the jar files AND that for the Listener Servlet) MUST be under the document base of the HTML page. This is achieved by using the Forms Servlet to generate the page and by invoking the Listener Servlet under the /forms60java path by mapping a file extension to it. The Listener Servlet is accessed under that path by mapping /forms60java/servlet to the servlet zone.

If you have users running Netscape with JInitiator, do the following:

Note: The following steps assume the web server and servlet engine are Apache and JServ (as supplied with Oracle iAS), and that the Forms Servlet is running using the servlet alias "f60servlet".

1. Stop Apache/JServ.

2. Edit the jserv.conf file, and add the following lines (after the existing ApJServMount lines):

```
ApJServMount /forms60java/servlet /root
ApJServAction .f60 /servlet/f60servlet
```

3. Edit the formsweb.cfg file, and use the following serverURL setting under the config section that is being used (or alter the default setting):

```
serverURL=/forms60java/servlet/oracle.forms.servlet.ListenerServlet
```

4. Restart Apache/JServ.

5. Access the Forms application (the page where the form runs) using a URL like:

```
https://theserver.thedomain.com/forms60java/aname.f60?config=myconfi
g
```

where aname can be any name (for example, forms or fred). Because the file name ends in ".f60" this request is routed to the Forms Servlet (f60servlet).

Note: You do not have to use https, as in the example above. You can also use http.

6. Log on to the authenticating proxy when prompted.

Open Bugs

Fixes for the following bugs were scheduled for but not included in this patch release. Check the bug database for more information.

1643859

CHART ITEM LABELS ARE NOT DISPLAYED COMPLETELY

1647725

DEV6I-P4:WEB:STATUS BAR HAS COUPLE OF ".."s AND FOCUSING ON IT SHOWS EMPTY TOOL TIP

1647552

DEV6I-P4:WEB:IAS:CANNOT WRITE IMAGE BACK TO THE FILE SYSTEM

1645907

WEBFORMS:SERVLET:IAS SHIPHOME:NOT ABLE TO CONNECT TO DATABASE AFTER DISCONNETION

National Language Support

Known Problems for All Languages

Form Builder User Interface May Not Be Fully Translated

Customers who install the Forms Builder with a language other than English may find that the user interface is a mixture of local language and English. The Forms Builder is not translated fully for some languages.

Strings displayed in Dialogs and Menus are not translated.

For these languages, please use the English language interface.

To do this, set DEVELOPER_NLS_LANG=AMERICAN_AMERICA.<charset> USER_NLS_LANG=< Language>_<Territory>.<charset> on the workstation that is running the Builder.

Substitute < Language> with the Language you want to run your form in. Substitute < Territory> with the Territory you want to run your form in. Substitute for <charset> the Oracle character set that you wish to use.

Euro Currency Symbol Has Limited Support
There is limited support in this release for the Euro currency symbol. To enable a limited use of the Euro symbol in this release, you must do the following:

1. If you are using Windows 95 or NT, update that operating Microsoft. (Windows 98 already contains the Euro support.)

2. Install Microsoft TrueType fonts that contain the Euro symbol.

3. Verify that your database and client NLS_LANG character sets both support the Euro symbol. If you will print the Euro symbol, Euro support on the printer is also required.

Microsoft has reserved hexcode 0x80 for the Euro symbol. To enter the Euro symbol on keyboards without an explicit Euro key, click the NumLock key to enable the Numeric Keypad, then click Alt + 128.

Some Wizard Buttons Have Untranslated Text

In the wizards on Unix systems, some buttons appear with their texts in English.

Restore Icons During Installation Creates Invalid Menus

Problem:
When using Software Asset Manager during a custom installation and selecting the Restore Icons button, invalid menus are created in 'Start -> Program'.

Workaround:

Reinstalling the 6*i* release will correctly reset the icons and menus.

Menu Action Not Available Via Keyboard After Print Cancel

Problem:

When accessing a form via keyboard commands, and selecting a print dialog and then cancelling that dialog several times, the print dialog could not be brought up again.

Workaround:

Avoid repeated cancelling.

FMRWEB.RES Must Be Configured Manually

The file "fmrweb.res" contains a keyboard map. This map is used by Oracle Forms Server to detect key strokes entered in the user's browser and map them to Forms actions. The map assumes that the keyboard is similar to a VT100 keyboard. If the user has a PC-style keyboard, we recommend that you copy the file "fmrpcweb.res" over the original "fmrweb.res". This will make the server assume that the keyboard is similar to a 101-key "PC" keyboard.

By default, the US versions of fmrweb.res and fmrpcweb.res are installed, regardless of language. In addition, for a particular language, the appropriate mapping files for that language's typical keyboard are installed. They are installed to the files "fmrweb<lang>.res" and "fmrpcweb<lang>.res", where <lang> is the Oracle language code. For instance, the Oracle language code for Latin American Spanish is "esa". To use the files for a particular language instead of the US versions, copy the desired file over the file "fmrweb.res".

Starting with Oracle Forms Server 6*i*, the files "fmrweb<lang>_utf8.res" and "fmrpcweb<lang>_utf8.res" are also installed. These contain the same key mappings as "fmrweb<lang>.res" and "fmrpcweb<lang>.res", but the files are encoded in the UTF8 character set. These files should be used if the server is using the UTF8 character set (UTF8 is the character set specified in NLS_LANG).

Information About NLS_LANG Parameter Values   The manual "Deploying Forms Applications to the Web" refers to a file named \bonus\nls\nlsdrl.wri. However, that file is no longer up-to-date, and is not supplied on the product CD.

For information about parameter values for NLS_LANG, consult the documentation for the Oracle8 8.0.6 server. The valid NLS_LANG parameter values are the same for Forms and for that database server.

Wallet Manager User Interface Is in English   When you install Wallet Manager from the Forms Developer CD, it will not have a translated user interface. The user interface will be in English. Oracle Wallet Manager translations can be obtained if required. Contact your Oracle representative for more details.

Known Problems for Double-Byte Languages

Editing with Single-Byte Font

Problem:
In any double-byte language implementation of the Builders, editing using a single-byte font face (such as Arial) causes characters to become distorted or unreadable. This occurs in any editing field.

Workaround:
Use double-byte fonts that display Roman script, instead of using the single-byte font.

Known Problems for Japanese

Cannot Save Modules If Character Set Is JA16EUC

Problem:
Cannot save modules in an Oracle Database if thecharacter set is JA16EUC.

Workaround:
Use the JA16SJIS character set instead.

Length Limit Moving From Windows to Solaris

Problem:
Cannot take objects with names over 30 bytes in length (using Hankaku-Katakana) from Windows to Solaris.

Workaround:
No workaround.

PL/SQL Editor Display Problems

Problem:
Characters entered on a single line are displayed on multiple lines, overlapping.

Workaround:
None.

UTF8 Limitation

Problem:
If NLS_LANG is set to American_America.UTF8, you can not create fmx files from fmb files that were created in JA16SJIS.

Workaround:
None.

Multibyte Characters Cannot Be Used for PL/SQL Library Name   Problem:
Trying to create a PL/SQL library name using a multibyte character set does not work correctly.

Workaround:
None.

Different prefs.ora Files May Be Needed

Problem:
If a customer selects Japanese installation, prefs.ora for the Japanese language in JA16SJIS encoding will be installed. This causes some problems for customers who develop their applications with other NLS_LANG settings such as:

- American_America.JA16SJIS (a)

     or

- Japanese_Japan.UTF8 (b)

Workarounds:
   a. (a) prefs.ora files for the American language are required. They need to be copied from the installation CD.

   b. (b) prefs.ora files in UTF8 encoding are required. Convert the prefs.ora files from JA16SJIS encoding to UTF8 encoding.

Message Texts from PL/SQL Interpreter Are Mixed

When running Procedure Builder, error messages (for example, ORA-04098) from the PL/SQL Interpreter are sometimes displayed in English and sometimes in Japanese.

Storage Requirement for Installing Cue Cards on Unix

If you choose to install the Cue Cards on a Unix system, both the Japanese tar files and the US tar files will be installed.

The total storage requirement for these tar files is approximately 275Mb. The Cue Cards are optional. If your storage space is limited, you may choose to not install the Cue Cards.

Known Problems for Arabic

Limitations for Displaying Charts on Solaris

Customers using the Solaris version of Forms Developer 6*i* who generate charts should note that the Solaris operating system has less support for fonts and locales than Windows NT. Specifically, Solaris does not have an official Arabic locale. Forms Developer 6*i* for Solaris has only very limited support for the Unicode locale.

As a result, charts generated on Solaris will not display text correctly if Arabic or Unicode is used. This will also happen with charts displayed on any Web Client that is accessing a Solaris-based server. This occurs because the chart is rendered into bitmap graphics on the server. If the server is Solaris-based, then Arabic and Unicode fonts are not available. Other text in forms, reports, and graphics is usually sent directly to the client and rendered in the client's locale.

The recommended workaround is to select a chart text font that is Western European and not Unicode.