

*A GUIDE TO CHANGED FUNCTIONALITY
BETWEEN ORACLE REPORTS 6i AND 10g*

An Oracle White Paper

August 2005

Introduction

INTRODUCTION

Successful businesses know that presenting their data in a timely and meaningful way provides a powerful advantage over the competition. To that end, businesses continue to need ever more powerful tools for producing high quality reports from the mass of disparate data sources kept in every major corporation today.

Oracle Reports has maintained its position as a premier enterprise reporting tool by keeping pace with rapid shifts in technology. As rapidly as technology has shifted, Oracle Reports has moved from character-based, to graphical-based, to client/server, and now to multi-tier Web-based development and deployment.

Oracle6*i* Reports was the last version of Oracle Reports to support both a character-mode and a client/server GUI reports viewer. With Oracle9*i* Reports, users moved to a Java 2 Enterprise Edition (J2EE) Web-based multi-tier environment. Oracle Reports 10*g* (9.0.4) and Oracle Reports 10*g* Release 2 (10.1.2) have improved the performance and scalability of Reports in the Web-based environment, and have also provided a host of new features.

It is recommended that Oracle6*i* Reports customers migrate to Oracle Reports 10*g* Release 2 (10.1.2) as soon as possible, and that Oracle9*i* Reports customers start planning their migration to Oracle Reports 10*g* Release 2 (10.1.2). Refer to [Metalink](#) for the latest information on the desupport dates for Oracle6*i* Reports and Oracle9*i* Reports.

OBJECTIVE AND SCOPE

The objective of this white paper is to provide one-stop information about the availability of features from earlier releases (6i, 9i, and 10g (9.0.4)) in Oracle Reports 10g Release 2 (10.1.2). In other words, this white paper is for you if you describe yourself as follows:

I am using Oracle6i Reports / Oracle9i Reports / Oracle Reports 10g (9.0.4). I would like to know which features are deprecated, obsolete, or changed in Oracle Reports 10g Release 2 (10.1.2), and how I can migrate existing reports that use these features.

The earlier version of this white paper, *Statement of Direction*, for Oracle Reports 10g (9.0.4) explained:

- Functionality that changed between Oracle6i Reports and Oracle9i Reports, and between Oracle9i Reports and Oracle Reports 10g (9.0.4).

In addition to that information, this white paper covers:

- The Oracle Reports 10g (9.0.4) functionality that changed in Oracle Reports 10g Release 2 (10.1.2)

As stated above, this white paper covers the *functionality* that was available in earlier releases, and is now deprecated, obsolete, or has changed its behavior. If you would like to know Oracle Corporation's *strategy* towards Oracle Reports, refer to the new document *Oracle Forms - Oracle Reports - Oracle Designer Statement of Direction* available on the Oracle Technology Network (OTN). If you want to take a look at the *new features* introduced in Oracle9i Reports, Oracle Reports 10g (9.0.4), and Oracle Reports 10g Release 2 (10.1.2), refer to the Oracle Technology Network (OTN).

SECTIONS

This white paper is divided into the following sections:

- [Deprecated Functionality](#)
- [Obsolete Functionality](#)
- [Changed Behavior and Naming](#)
- [Resources](#)

Here is a glossary of the terms used:

Obsolete functionality: Obsolete functionality is no longer available in Oracle Reports 10g Release 2 (10.1.2). By continuing to remove obsolete functionality and replacing it with new technology, Oracle Reports ensures that our customers always have the latest functionality they need for successful e-Business development and deployment.

Deprecated functionality: Deprecated functionality is at a point where it no longer contributes significantly to the usefulness of the product. As a result, it is no longer documented and its use is discouraged, as it will be made *obsolete* in a subsequent release of Oracle Reports. The deprecation phase is provided so that customers can gradually migrate to new functionality, before the deprecated functionality is made *obsolete*.

Deprecated Functionality

The following functionality is deprecated in Oracle Reports 10g Release 2 (10.1.2):

- Oracle Express data source and related options
- Reports Server clustering
- Import Portal component
- Backward compatibility with Oracle6i Reports clients
- Using keywords `SERVER` and `USERID` with `SRW.RUN_REPORT`

The following functionality was deprecated in earlier releases (Oracle9i Reports or Oracle Reports 10g (9.0.4)):

- User exits
- Selected executables
- Selected environment variables
- `SRW.SET_ATTR` built-in function
- Formatting-related SRW built-in parameters
- Tracing-related SRW built-in procedures and parameters
- Embedding an OLE2 object

EXPRESS DATA SOURCE AND RELATED OPTIONS

Oracle Express Server is an advanced calculation engine and data cache that delivers on-line analytical processing (OLAP) via a range of Web-based and client/server platforms. Based on a multidimensional data model, Express Server is optimized to support mathematical, financial, statistical, and time-series manipulations, as well as forecasting, modeling, and multi-user what-if scenarios.

The OLAP option to the Oracle Database is the successor to Oracle Express Server. The OLAP option fulfills the same purpose as Express Server and it provides the same general feature set. It does so by including Express Server's multidimensional calculation engine and data types directly in the context of the Oracle Database. In addition, the OLAP option to the Oracle Database provides several new features. Oracle Express customers are advised to migrate to the OLAP option of the Oracle Database.

Oracle Reports 10g Release 2 (10.1.2) provides an OLAP data source that can be used to create reports based on OLAP data. As a result, the Oracle Express data source is now deprecated. Additionally, any related options, such as the command line keyword `EXPRESS_SERVER`, are deprecated.

Action Required

Migrate your Express data to the OLAP option provided with the Oracle Database. Use the OLAP data source in Oracle Reports 10g Release 2 (10.1.2) to create your reports based on OLAP data.

REPORTS SERVER CLUSTERING

A cluster is a virtual grouping of servers into a community for the purpose of sharing the request-processing load efficiently across members of the cluster. Reports Server clustering provides the following benefits:

- Load balancing: The load is shared equally across multiple Reports Servers.
- Failover: If one member is shut down, the other members carry on managing the request load. There is no single-point-of-failure, where one machine's malfunction brings the whole system down.

However, since modern IT environments require high availability of the entire IT infrastructure, it is better to have a centralized high availability solution at the level of the application server, instead of a separate clustering solution for each component.

Oracle Application Server High Availability provides the industry's most reliable, resilient, and fault-tolerant application server platform. Oracle Reports' integration with OracleAS High Availability makes sure that your enterprise-reporting environment is extremely reliable and fault-tolerant. Since using OracleAS High Availability provides a centralized clustering mechanism and several cutting-edge features, Oracle Reports clustering is now deprecated.

Action Required

If you are using Reports Server clustering, switch to OracleAS High Availability. Refer to *Oracle Application Server Enterprise Deployment Guide* for information on how to configure your reporting environment to use OracleAS High Availability.

If you are using an OracleAS Forms Services application from prior releases that includes a Reports Server cluster name, this application will fail to bind to the Reports Server cluster it references. To resolve this issue, use the new Reports Servlet property `REPORTS_SERVERMAP`, which enables you to map a cluster name to a Reports Server name. For more information on this property, refer to the chapter *Configuring Oracle Reports Services available in Oracle Application Server Reports Services Publishing Reports to the Web*.

IMPORT PORTAL COMPONENT

You can create simple reports in Oracle Portal using its browser-based wizard. This wizard does not offer all the powerful capabilities available inside Reports Builder, but allows you to develop and deploy simple reports. If you need to apply advanced formatting to these reports, you can import those reports into Reports Builder by choosing **File > Import Portal Component**.

However, since it is possible to create simple as well as complex reports inside Reports Builder and then deploy them in Oracle Portal, it is not necessary to use the browser-based wizard of Oracle Portal. As a result, the functionality to import reports developed with the Oracle Portal wizard into Reports Builder is deprecated in Oracle Reports 10g Release 2 (10.1.2).

Action Required

Instead of using the browser-based wizard, develop reports in Reports Builder, and deploy them in Oracle Portal.

BACKWARD COMPATIBILITY WITH ORACLE6i REPORTS CLIENTS

Oracle9i Reports and Oracle Reports 10g (9.0.4) allowed backward compatibility with Oracle6i Reports clients if the `compatible` element was configured appropriately in the Reports Server configuration file. Specifically, when the `compatible` element was set, Oracle6i Reports client requests were forwarded to the Reports Server. This backward compatibility feature, and thus the `compatible` element, is deprecated in Oracle Reports 10g Release 2 (10.1.2). Refer to *Oracle Application Server Reports Services Publishing Reports to the Web* for more information on the `compatible` element.

Action Required

Switch to using Oracle Reports 10g Release 2 (10.1.2) clients.

USING KEYWORDS SERVER AND USERID WITH SRW.RUN_REPORT

In Oracle Reports 10g Release 2 (10.1.2), the use of keywords `SERVER` and `USERID` with `SRW.RUN_REPORT` is deprecated. Reports created in prior releases that use these keywords with `SRW.RUN_REPORT` will continue to run in Oracle Reports 10g Release 2 (10.1.2), but the values specified in these keywords will be ignored. If you need these keyword values to be honored, set the environment variable `REPORTS_SRWRUN_TO_SERVER=YES`.

In general, it is recommended not to use the keywords `SERVER` and `USERID` with `SRW.RUN_REPORT`, because the values for these keywords should be inherited from the parent report request.

Action Required

In existing reports, it is recommended that you remove the keywords `SERVER` and `USERID` in `SRW.RUN_REPORT` commands. If you need to use these keywords with `SRW.RUN_REPORT`, set the environment variable `REPORTS_SRWRUN_TO_SERVER=YES`. However, note that this environment variable is available only for backward compatibility, and may not be available in later releases.

USER EXITS

A user exit is a call to an external function executed within Oracle Reports. User exits provide a way to pass control (and possibly arguments) from Oracle Reports to another Oracle product or 3GL program, and then return control (and possibly arguments) back to Oracle Reports. However, user exits require you to re-make the executables to link in all the user exits. This makes troubleshooting and upgrades to new releases of the product difficult.

In Oracle Reports 10g Release 2 (10.1.2), you can call Java methods using the `ORA_JAVA` package and the Java Importer. This reduces the need to have user exits in a report and allows for a more open and portable deployment. You may also use the `ORA_FFI` package, which provides a foreign function interface for invoking C functions in a dynamic library. With the availability of these new built-ins, the use of user exits is deprecated in Oracle Reports 10g Release 2 (10.1.2), though makefiles will still be supplied to permit you to continue to work with existing user exits.

Action Required

If you are using user exits, switch to using either the ORA_JAVA package or the Java Importer utility. Refer to *Oracle Reports online Help* for more information on how to use the Java Importer.

Refer to Table 1 to replace deprecated SRW user exit built-in procedures with new functionality.

Table 1: *Deprecated user exit built-in procedures*

Built-in Procedure	Action Required
SRW.UNKNOWN_USER_EXIT	Replace with ORA_FFI, or ORA_JAVA calls.
SRW.USER_EXIT	Replace with ORA_FFI, or ORA_JAVA calls.
SRW.USER_EXIT20	Replace with ORA_FFI, or ORA_JAVA calls.
SRW.USER_EXIT_FAILURE	Replace with ORA_FFI, or ORA_JAVA calls.

SELECTED EXECUTABLES

In Oracle6i Reports, `rwcgi60` was the Reports Web CGI (Common Gateway Interface) executable that provided a connection between a Web Server and the Reports Server, allowing you to run reports dynamically from your Web browser.

Oracle Reports 10g Release 2 (10.1.2) provides you with two executables that perform the same function as `rwcgi60`: `rwcgi.exe` (on Win32) and `rwcgi` (on UNIX). However, all of these CGI executables are deprecated. Instead, use standard Java 2 Enterprise Edition (J2EE) methods of making a Web connection: use either a servlet URL or a JSP URL.

Action Required

If you are using any of the CGI executables, use either a servlet or a JSP URL instead. Refer to *Oracle Application Server Reports Services Publishing Reports to the Web* for more information on how to submit requests to the Reports Server.

SELECTED ENVIRONMENT VARIABLES

Some environment variables have been deprecated either because the basic functionality which these environment variables controlled has been deprecated, or because newer ways of setting the values are available.

Action Required

Refer to Table 2 to replace deprecated environment variables with new functionality.

Table 2: *Deprecated environment variables*

Environment Variable	Action required
REPORTS_CGIDIAGBODYTAGS REPORTS_CGIDIAGHEADTAGS REPORTS_CGIHELP REPORTS_CGIMAP REPORTS_CGINODIAG	CGI functionality is deprecated. As a result, these environment variables are no longer needed. Refer to the section Deprecated Executables for a description of CGI functionality, and the action required.

Environment Variable	Action required
REPORTS_COOKIE_EXPIRE	This environment variable specifies the lifetime of a cookie within a given Reports Server session. This cookie is set by <code>rwServlet</code> so that users need to authenticate only once per session. In Oracle Reports 10g Release 2 (10.1.2), by default the security mechanism is handled by Oracle Single Sign-On, and there is no need for <code>rwServlet</code> to set a cookie. As a result, this environment variable is not needed.
REPORTS_DB_AUTH REPORTS_SYS_AUTH	REPORTS_DB_AUTH specifies the HTML template used to show the database authentication page to the user. REPORTS_SYS_AUTH specifies the HTML template used to show the Reports Server authentication page. In Oracle Reports 10g Release 2 (10.1.2), Oracle Single Sign-On shows its own authentication page. As a result, these environment variables are not needed.
REPORTS_ENCRYPTION_KEY	This environment variable specifies the encryption key used to encrypt the user name and password. This was needed in Oracle6i Reports because the username and password were stored in the Reports Server configuration file. Since Oracle9i Reports and Oracle Reports 10g store the username and password in Oracle Internet Directory (OID), this environment variable is no longer needed.
REPORTS_SERVER	This environment variable specifies the default Reports Server for Web Cartridge or Web CGI requests. In Oracle Reports 10g Release 2 (10.1.2), you should specify the default server (in-process server) name in the <code>rwServlet.properties</code> file.
REPORTS_SSLPORT	This environment variable specifies the port number when using SSL while using CGI. Since CGI is deprecated in Oracle Reports 10g Release 2 (10.1.2), this environment variable is no longer needed. When using servlet or JSP URL over SSL, the Oracle HTTP Server's SSL port is used. So you need to configure the SSL port in Oracle HTTP Server. See <i>Oracle HTTP Server Administrator's Guide</i> for more information.

SRW.SET_ATTR BUILT-IN FUNCTION

The `SRW.SET_ATTR` built-in function was originally designed to save time and resources by allowing report developers to set attributes like font face, foreground and background colors, border width, and so on in a *single* built-in function. However, each of these attributes has evolved into a *separate* built-in procedure (for example, `SRW.SET_FONT_FACE`, `SRW.SET_BORDER_WIDTH`). As a result, `SRW.SET_ATTR` is now deprecated, and it is recommended that you use the new built-in procedures.

For example, using `SRW.SET_ATTR`, your code might look like this:

```

If :sal > 2000 then
srw.attr.mask      :=      srw.face_attr      +
                          srw.sz_attr        +
                          srw.weight_attr     +
                          srw.style_attr      +
                          srw.gcolor_attr;

srw.attr.face      :=      'times';
srw.attr.sz        :=      18;
srw.attr.weight    :=      srw.bold_weight;
srw.attr.style     :=      srw.underline_style;
srw.attr.gcolor    :=      'blue';
srw.set_attr (0, srw.attr);
end if;
RETURN (TRUE);

```

You can change your code to perform the same function as above by using the new built-in procedures, as follows:

```

If :sal > 2000 then
srw.set_font_face('times')
srw.set_font_size(18)
srw.set_font_weight(srw.bold_weight)
srw.set_font_style(srw.underline_style)
srw.set_text_color('blue')
end if;
RETURN (TRUE);

```

Action Required

Refer to Table 3 to replace deprecated `SRW.SET_ATTR` functionality and attributes with new SRW built-in procedures in the PL/SQL code in your report.

Table 3: Deprecated SRW.SET_ATTR functionality

Deprecated SRW.SET_ATTR functionality	Attribute affected	Action Required
srw.action_attr	srw.attr.action	Use srw.set_pdf_action
srw.aftcode_attr	srw.attr.aftcode	Use srw.set_after_printing_code
srw.aftform_escape_attr	srw.attr.aftform_escape	Use srw.set_after_form_html
srw.aftpage_escape_attr	srw.attr.aftpage_escape	Use srw.set_after_page_html
srw.aftreport_escape_attr	srw.attr.aftreport	Use srw.set_after_report_html
srw.bbcolor_attr	srw.attr.bbcolor	Use srw.set_background_border_color
srw.befcode_attr	srw.attr.befcode	Use srw.set_before_printing_code
srw.befform_escape_attr	srw.attr.befform_escape	Use srw.set_before_form_html

Deprecated SRW.SET_ATTR functionality	Attribute affected	Action Required
srw.befpage_escape_attr	srw.attr.befpage_escape	Use srw.set_before_page_html
srw.befreport_escape_attr	srw.attr.befreport_escape	Use srw.set_before_report_html
srw.bfcolor_attr	srw.attr.bfcolor	Use srw.set_background_fill_color
srw.bookmark_attr	srw.attr.bookmark	Use srw.set_bookmark
srw.borderwidth_attr	srw.attr.borderwidth	Use srw.set_border_width
srw.borderpatt_attr	srw.attr.borderpatt	Use srw.set_border_pattern
srw.face_attr	srw.attr.face	Use srw.set_font_face
srw.fborder_attr	srw.attr.fborder	Use srw.set_foreground_border_color
srw.ffcolor_attr	srw.attr.ffcolor	Use srw.set_foreground_fill_color
srw.fillpatt_attr	srw.attr.fillpatt	Use srw.set_fill_pattern
srw.formatmask_attr	srw.attr.formatmask	Use srw.set_format_mask
srw.gcolor_attr	srw.attr.gcolor	Use srw.set_text_color
srw.gspacing_attr	srw.attr.gspacing	Use srw.set_custom_spacing or srw.set_spacing
srw.hjust_attr	srw.attr.hjust	Use srw.set_justification
srw.hyperlink_attr	srw.attr.hyperlink	Use srw.set_hyperlink
srw.linktag_attr	srw.attr.linktag	Use srw.set_linktag
srw.printer_intray_attr	srw.attr.printer_intray	Use srw.set_printer_tray
srw.plain_style	srw.attr.plain_style	Use srw.set_font_style(srw.plain_style)
srw.italic_style	srw.attr.italic_style	Use srw.set_font_style(srw.italic_style)
srw.oblique_style	srw.attr.oblique_style	No longer applicable
srw.underline_style	srw.attr.underline_style	Use srw.set_font_style(srw.underline_style)
srw.outline_style	srw.attr.outline_style	No longer applicable
srw.shadow_style	srw.attr.shadow_style	No longer applicable

Deprecated SRW.SET_ATTR functionality	Attribute affected	Action Required
srw.inverted_style	srw.attr.inverted_style	No longer applicable
srw.overstrike_style	srw.attr.overstrike_style	No longer applicable
srw.blink_style	srw.attr.blink_style	No longer applicable
srw.sz_attr	srw.attr.sz	Use srw.set_font_size(size)
srw.plain_texta	srw.attr.plain_texta	Use srw.set_charmode_text(srw.plain_texta)
srw.reverse_texta	srw.attr.reverse_texta	Remove srw.set_attr built-in
srw.bold_texta	srw.attr.bold_texta	Use srw.set_charmode_text(srw.bold_texta)
srw.reversebold_texta	srw.attr.reversebold_texta	No longer applicable
srw.underline_texta	srw.attr.underline_texta	Use srw.set_charmode_text(srw.underline_texta)
srw.underlinereverse_texta	srw.attr.underlinereverse_texta	No longer applicable
srw.underlinebold_texta	srw.attr.underlinebold_texta	No longer applicable
srw.reverseboldunderline_texta	srw.attr.reverseboldunderline_texta	No longer applicable
srw.ultralight_weight	srw.attr.ultralight_weight	No longer applicable
srw.extralight_weight	srw.attr.extralight_weight	No longer applicable
srw.light_weight	srw.attr.light_weight	No longer applicable
srw.demilight_weight	srw.attr.demilight_weight	No longer applicable
srw.medium_weight	srw.attr.medium_weight	Use srw.set_font_weight(srw.medium_weight)
srw.demibold_weight	srw.attr.demibold_weight	No longer applicable
srw.bold_weight	srw.attr.bold_weight	Use srw.set_font_weight(srw.bold_weight)
srw.extrabold_weight	srw.attr.extrabold_weight	No longer applicable

FORMATTING-RELATED SRW BUILT-IN PARAMETERS

The formatting-related SRW built-in parameters that were applicable only to client-server environments (for example, `SRW.BLINK_STYLE`) are deprecated in Oracle Reports 10g Release 2 (10.1.2). Since they are not applicable to a Web environment, there is no replacement for such built-in parameters.

Action Required

Refer to Table 4 to remove formatting-related SRW built-in parameters from the PL/SQL code in your report.

Table 4: Depreciated formatting-related SRW built-in parameters

Deprecated parameter	Built-in parameter affected	Description
<code>srw.oblique_style</code>	<code>srw.set_font_style</code>	Applicable for client-server only Note: Font styles to remain: <code>srw.underline_style</code> <code>srw.plain_style</code> <code>srw.italic_style</code>
<code>srw.outline_style</code>	<code>srw.set_font_style</code>	Applicable for client/server only
<code>srw.shadow_style</code>	<code>srw.set_font_style</code>	Applicable for client/server only
<code>srw.inverted_style</code>	<code>srw.set_font_style</code>	Applicable for client/server only
<code>srw.blink_style</code>	<code>srw.set_font_style</code>	Applicable for client/server only
<code>srw.ultralight_weight</code>	<code>srw.set_font_weight</code>	Applicable for client/server only Note: Font weights to remain: <code>srw.medium_weight</code> <code>srw.bold_weight</code>
<code>srw.extralight_weight</code>	<code>srw.set_font_weight</code>	Applicable for client/server only
<code>srw.light_weight</code>	<code>srw.set_font_weight</code>	Applicable for client/server only
<code>srw.demilight_weight</code>	<code>srw.set_font_weight</code>	Applicable for client/server only
<code>srw.demibold_weight</code>	<code>srw.set_font_weight</code>	Applicable for client/server only
<code>srw.extrabold_weight</code>	<code>srw.set_font_weight</code>	Applicable for client/server only

TRACING-RELATED SRW BUILT-IN PROCEDURES AND PARAMETERS

Before Oracle Reports 10g Release 2 (10.1.2), it was possible to specify trace options during the report execution using SRW built-in procedures. For example, you could turn on tracing in one of the report triggers, and turn it off at a later point in the report execution using SRW built-ins. Within the command that turned on the tracing, you could specify various trace options using SRW trace parameters. These SRW built-ins and the related SRW parameters are deprecated in Oracle Reports 10g Release 2 (10.1.2).

Action Required

Refer to Table 5 to remove deprecated tracing-related SRW built-in procedures in the PL/SQL code in your report. Instead, specify the trace options in the Reports Server configuration file or on the command line using the `TRACEFILE`, `TRACEMODE`, and `TRACEOPTS` keywords. Refer to *Oracle Application Server Reports Services Publishing Reports to the Web* for more information on tracing.

Table 5: Deprecated tracing-related SRW built-in procedures

Deprecated SRW built-in procedure	Description
<code>srw.trace_start</code>	Was used to begin the tracing
<code>srw.traceopts.mask</code>	Was used to specify the tracing options while starting the trace
<code>srw.trace_add_option</code>	Was used to add trace options
<code>srw.trace_rem_option</code>	Was used to remove trace options
<code>srw.trace_end</code>	Was used to end the tracing

Since the above SRW built-in procedures are deprecated, the following tracing-related SRW parameters used to specify the trace options are also deprecated:

```
srw.trace_all  srw.trace_app  srw.trace_brk  srw.trace_dst
srw.trace_err  srw.trace_pls  srw.trace_prf  srw.trace_sql
```

EMBEDDING AN OLE2 OBJECT

Oracle6i Reports allowed users to embed an Object Linking and Embedding (OLE2) object in the report. For example, OLE2 objects could be used to insert a company logo, a Microsoft Word document, or a Microsoft Excel file in your report. This functionality is deprecated in Oracle Reports 10g Release 2 (10.1.2) because it is only applicable to client-server environments. As a result, the related command line keyword `CONTAINSOLE` is also deprecated. This command line keyword specifies whether the program units or attached libraries for the report contain any OLE calls.

Action Required

Remove the OLE2 objects from your reports. Since this functionality is not applicable to Web environments, there is no direct replacement available. However, you can mimic the same functionality in the following ways:

- Display images and text from external files using the File Link tool available in the Paper Layout tool palette.
- Use the Read from File property to retrieve column information from an external file.
- If you have access to a Java plug-in that can read and embed an object (for example, a Microsoft Word document or Microsoft Excel file), you can use the plug-in inside Oracle Reports using the Java Importer utility.

Refer to the *Oracle Reports online Help* for more information on the File Link tool, the Read from File property, and the Java Importer utility.

Obsolete Functionality

The following functionality is obsolete in Oracle Reports 10g Release 2 (10.1.2):

- Running Reports Server as a Windows service

The following functionality was obsoleted in a prior release (that is, Oracle9i Reports or Oracle Reports 10g (9.0.4)):

- Character-mode Runtime Viewer
- Graphical user interface for reports runtime
- Oracle Forms built-in RUN_PRODUCT for submitting report requests
- Oracle Graphics
- Selected executables
- Selected command line keywords
- Selected system parameters
- Miscellaneous Items

RUNNING REPORTS SERVER AS A WINDOWS SERVICE

In releases prior to Oracle Reports 10g Release 2 (10.1.2), Reports Server could be run on the Windows platform as a Windows service with the following command:

```
rwserver -install server_name
```

This functionality is no longer available in Oracle Reports 10g Release 2 (10.1.2). As a result, the related command line keywords `INSTALL` and `UNINSTALL` are also obsolete.

In Oracle Reports 10g Release 2 (10.1.2), you must start, shut down, monitor, and manage the Reports Server through the Oracle Process Manager and Notification Server (OPMN) and Oracle Enterprise Manager. OPMN provides a centralized mechanism for initializing, maintaining, and shutting down your Oracle Application Server components, including the Reports Server. Oracle Enterprise Manager, included with the Oracle Application Server, provides managing and monitoring services to OracleAS Reports Services. You can conveniently monitor the Reports Servers through Oracle Enterprise Manager and, if the process crashes for any reason, OPMN restarts Reports Server for you automatically. On the Windows platform, OPMN itself will be run as a Windows service.

Action Required

Instead of running the Reports Server as a Windows service, configure and run the Reports Server through OPMN. Refer to *Oracle Application Server Reports Services Publishing Reports to the Web* for more information.

CHARACTER-MODE RUNTIME VIEWER

Releases prior to Oracle9i Reports included the special character-mode runtime user interface executable (`rwrun60c`), which was specific to the UNIX operating system. This executable is no longer shipped with Oracle Reports.

Action Required

Character-mode reports can still be designed in Reports Builder 10g (10.1.2), and character-mode output produced with Oracle Reports 10g (10.1.2), using the command line option `MODE=CHARACTER`.

GRAPHICAL USER INTERFACE FOR REPORTS RUNTIME

In releases prior to Oracle9i Reports, the runtime executable (`rwrun.exe`) displayed the screen preview or the Parameter Form. This functionality is obsolete. As a result, for the command line keyword `DESTYPE` values of `PREVIEW` and `SCREEN` are no longer supported. You can still use `rwrun.exe` to send the report output to supported output destinations like file, printer, and email. Additionally, if the command line option `PARAMFORM=YES` is used with `rwrun.exe`, it will be ignored. If the report contains parameters, they must be passed on the command line.

In Oracle Reports 10g Release 2 (10.1.2), all reports are run through the Reports Server, and you can no longer submit requests directly to the Reports Engine. This allows you to take full advantage of the Reports Server for security and distribution functionality. It also allows for a more streamlined architecture and consistency within the product.

Action Required

Perform the following steps:

1. When running the report using `rwrun.exe`, pass the parameters in the command line. When running the report on the Web using `rwervlet`, either pass the parameters in the command line, or use the command line option `PARAMFORM=YES` to display the Parameter Form.
2. To view the paper layout output, run the report on the Web, and use the command line option `DESTYPE=CACHE`. This will display the report output in the browser. For a list of all supported output destinations, refer to *Oracle Application Server Reports Services Publishing Reports to the Web*.

Note: In Reports Builder (`rwbuilder`), you can still set the `DESTYPE` system parameter to `SCREEN` or `PREVIEW` to format a report to display screen fonts or printer fonts in the Reports Builder user interface.

ORACLE FORMS BUILT-IN RUN_PRODUCT FOR SUBMITTING REPORT REQUESTS

The Oracle Forms built-in `RUN_PRODUCT` is replaced by the new built-in `RUN_REPORT_OBJECT`.

Action Required

Remove `RUN_PRODUCT` from your Oracle Forms code. Instead, use the new built-in `RUN_REPORT_OBJECT` for submitting report requests. For more information, refer

to the white paper *Oracle Application Server 10g - Integrating Oracle Reports in Oracle Forms Services* available on the Oracle Technology Network (OTN).

ORACLE GRAPHICS

In releases prior to Oracle9i Reports, Oracle Graphics was used to create charts in reports. Oracle Graphics and its OGD image format are no longer supported. Reports built with previous versions of Oracle Reports containing Oracle Graphics charts will continue to run in Oracle Reports 10g Release 2 (10.1.2) if the Oracle Graphics 6i runtime is installed on the same machine in a separate ORACLE_HOME. However, this is not a supported configuration, and Oracle will not fix bugs that result from this configuration.

Instead, use the Graph Wizard, which offers a much wider selection of graph types, as well as more control over the look and feel of the graph from within Reports Builder. It is strongly recommended that all Oracle Graphics objects be re-created using the Graph Wizard. Due to inherent differences in the underlying technology, there is no migration path or conversion mechanism available to convert Oracle Graphics charts to the Graph Wizard format.

Action Required

Re-create all Oracle Graphics 6i objects using the Graph Wizard. For more information on using the Graph Wizard, refer to the *Oracle Reports online Help* and the *Graphing FAQ* available on the Oracle Technology Network (OTN).

SELECTED EXECUTABLES

Executables that were applicable to only client-server environments have been rendered obsolete beginning with Oracle9i Reports. In addition, some executables have been replaced with new executables that provide more comprehensive functionality.

Action Required

Refer to Table 6 to replace the use of obsolete executables with new functionality.

Table 6: Obsolete executables

Obsolete Executable	Description	Action Required
rwrbe60.exe (Win32) and rwrbe60 (UNIX)	Reports Background Engine. Reports requests could be directly submitted to the background engine. Since Oracle9i Reports, all reports are executed through the Reports Server. There are no more direct calls to the Reports Engine. As a result, the background engine is no longer needed.	Submit requests to the Reports Server.
rwisv60.exe (Win32)	This executable was used to submit reports to the Reports Background Engine for execution. Since the background engine is no longer available, this executable is not needed.	Submit requests to the Reports Server.
rrows60.dll (Win32) and rrows60.so	Oracle Application Server cartridge. This was used to	Use servlet or JSP URL to submit reports requests on

Obsolete Executable	Description	Action Required
(UNIX)	submit requests to the Reports Server via the Web. Since the same functionality is available via J2EE-standard Servlet and JSP URLs, the cartridge is no longer needed.	the Web.
obe60.exe (Win32)	Query Builder. The Query Builder is available from within Reports Builder. Since it is no longer available as a standalone tool, this executable is not needed.	Use Query Builder from within the Reports Builder.
gorun60.exe (Win32) and g60runm (UNIX)	Oracle Graphics Runtime	Oracle Graphics is no longer available since Oracle9i Reports. Use the Graph Wizard in Reports Builder.
gobld60.exe (Win32) and g60desm (UNIX)	Oracle Graphics Builder	
gobat60.exe (Win32) and g60batm (UNIX)	Oracle Graphics Batch	

SELECTED COMMAND LINE KEYWORDS

There are a number of command line keywords that are obsolete in Oracle Reports 10g Release 2 (10.1.2). The functionality provided by these command line keywords is either now provided by new keywords or environment variables that give you more comprehensive options, or they are no longer needed.

Action Required

Refer to Table 7 to replace the use of obsolete command line keywords with new functionality.

Table 7: Obsolete command line keywords

Obsolete Keyword	Action required
INSTALL UNINSTALL	Since Oracle Reports can no longer be run as a Windows service, these keywords are not needed. See the section Running Reports Server as a Windows Service for action required.
CURRENCY - The currency character to be used in number formats.	Set the NLS_CURRENCY environment variable. If not set, the default value is based on the language derived from the NLS_LANG environment variable.
THOUSANDS - The thousands character to be used in number formats.	Set the NLS_NUMERIC_CHARACTERS environment variable for both the thousands separator and the decimal character. If not set, the default value is based on the language derived from the NLS_LANG environment variable.
DECIMAL - The decimal character to be used in number formats.	Set the NLS_NUMERIC_CHARACTERS environment variable for both the thousands separator and the decimal character. If not set, the default value is based on the language derived from the NLS_LANG environment variable.

PROFILE - Name of file that stores performance statistics.	Use TRACEOPTS=TRACE_PRF to write the performance statistics in the trace file. See the <i>Oracle Reports Online Help</i> for complete list of options.
LOGFILE - Name of file that stores status and error output.	Use a third party application to capture screen print.
KEYIN - Name of a keystroke file that executes at runtime.	Obsolete function with removal of client/server and character mode GUI.
KEYOUT - Name of a keystroke file that records at runtime.	Obsolete function with removal of client/server and character mode GUI.
UPGRADE_PLSQL - Whether rwconverter should upgrade the PL/SQL code in the report to the latest version required by Oracle Reports Developer.	In Oracle Reports 10g Release 2 (10.1.2), rwconverter no longer converts the PL/SQL code in the report to the latest version. You need to manually change the PL/SQL code in your reports to make sure that it compiles with the latest version required by Oracle Reports Developer.

NOTE: The command line keyword ERRFILE was included in the obsolete command line keywords list in the past. This keyword has been removed from the obsolete command line keywords list in order to provide compatibility with Oracle6i Reports scenarios where it was possible for users to get the error message file on the client using this keyword. As a result, this keyword will continue to be supported for the following Oracle Reports executables: RWBUILDER, RWCLIENT, and RWRUN.

SELECTED SYSTEM PARAMETERS

Due to changes in how some functionality is configured, the system parameters associated with such functionality are obsolete. Instead, use the new methods of configuration such as setting the appropriate environment variable or using command line keywords.

Action Required

Refer to Table 8 to replace the use of obsolete system parameters with new functionality.

Table 8: Obsolete system parameters

System Parameter	Action required
CURRENCY - The currency character to be used in number formats.	Set the NLS_CURRENCY environment variable. If not set, the default value is based on the language derived from the NLS_LANG variable.
THOUSANDS - The thousands character to be used in number formats.	Set the NLS_NUMERIC_CHARACTERS environment variable for both the thousands separator and the decimal character. If not set, the default value is based on the language derived from the NLS_LANG variable.
DECIMAL - The decimal character to be used in number formats.	Set the NLS_NUMERIC_CHARACTERS environment variable for both the thousands separator and the decimal character. If not set, the default value is based on the language derived from the NLS_LANG variable.

System Parameter	Action required
BACKGROUND	Use BACKGROUND as a command line keyword.

MISCELLANEOUS ITEMS

Since Oracle9i Reports, some functionality that is not applicable to the Web environment or has been replaced by newer functionality is no longer available.

Action required

Refer to Table 9 to replace the use of miscellaneous obsolete functionality with new functionality.

Table 9: Other obsolete functionalities

Obsolete Functionality	Executable affected	Description and Action Required
Ability to create a button in layout model.	rwbuilder.exe and rwrwn.exe	Buttons were used in client/server user interface (for example, to launch drill-down reports). Buttons are not applicable to the Web environment, and you must use hyperlinks instead. If you open existing 6i reports containing buttons, the buttons are displayed as simple text objects.
Define an external query (File>New>External Query)	rwbuilder.exe	The files created by this action contain stand-alone SQL queries, and are stored in the file system. Instead of creating external files, you either need to define the query in the report's data model, or <i>import</i> a query from an external file.
PVCS, Clearcase for source control	rwbuilder.exe	Instead of PVCS or Clearcase, use the Oracle Software Configuration Manager for source control. Oracle Reports provides built-in integration with Oracle Software Configuration Manager.
Ability to open reports from or save reports to the database	rwbuilder.exe	Reports can no longer be saved to/opened from the database. The report definition files can be saved either to the file system or into the Oracle Software Configuration Manager. If you currently have reports stored in the database, you will need to open them in a previous version of Oracle Reports and save them to the file system.

Obsolete Functionality	Executable affected	Description and Action Required
Oracle Reports Call Interface		The Oracle Reports Call Interface was used to call Oracle Reports from within a 3GL program (written in languages like C, COBOL, or FORTRAN). Since Oracle Reports has moved to the industry-standard J2EE technology, this functionality is no longer supported. You should call the Reports Server via standard mechanisms like a URL (servlet or JSP), Web services, and so on. Refer to <i>Oracle Application Server Reports Services Publishing Reports to the Web</i> for more information on how to submit requests to the Reports Server.
Report on reports stored in database	rwbuilder.exe	Two report definition files provided with Oracle Reports, <code>srwdoc1b.rdf</code> and <code>srwdocpb.rdf</code> , could be run against RDFs stored in the database to produce information on the stored reports' metadata. Since it is no longer possible to store reports in the database, this functionality is not needed.
MAPI e-mail protocol	rwsrvr.exe	Use the Internet-standard SMTP (Simple Mail Transfer Protocol) to send emails.
Oracle drawing format (*.odf) and Oracle6i Graphics image format (*.ogd)	rwbuilder.exe	These image formats were associated with Oracle Graphics. Since Oracle Graphics is no longer supported, files with *.odf and *.ogd extensions are not supported.
ActiveX control (or OCX) interface to Reports	rwsxa60.ocx rwsxu60.ocx	Since Oracle Reports has moved to the industry-standard J2EE technology, the ActiveX control (or OCX) interface to the Reports Server is no longer supported. You should communicate with Reports Server via standard mechanisms like a URL (servlet or JSP), and Web services. Refer to <i>Oracle Application Server Reports Services Publishing Reports to the Web</i> for more information on how to submit requests to the Reports Server.
Web Wizard	rwbuilder.exe	The Web Wizard is no longer available in Reports Builder. Instead, use the Insert Bookmarks dialog box. For more information, refer to the <i>Oracle Reports online Help</i> .

Changed Behavior and Naming

With improvements in functionality offered by Oracle Reports with every new release, it is necessary that the names of some of the properties, executables, and interfaces change in keeping with new and enhanced functionality. For example, since the Parameter Form is applicable only to the paper layout, and not to the Web layout, the previously named “Parameter Form view” in Reports Builder is now called “Paper Parameter Form view”.

This section lists changes to:

- Behavior
 - Changed behavior of command line keyword `BACKGROUND`
 - Changed behavior of command line keyword `BATCH`
- Naming
 - Executable names
 - Environment variable names
 - Reports Builder user interface names

Note that since you simply need to start using the new names in Oracle Reports 10g Release 2 (10.1.2), there is no **Action Required** description under those sub-sections that explain name changes.

CHANGED BEHAVIOR OF COMMAND LINE KEYWORD BACKGROUND

In Oracle Reports 10g Release 2 (10.1.2), the command line keyword `BACKGROUND` specifies whether a report on the server should be run synchronously (`NO`) or asynchronously (`YES`). It can be used only with the executables `rwclient`, `rwervlet`, and `rwcgi`.

Action Required

Make sure that your use of the command line keyword `BACKGROUND` is consistent with Table 10. Refer to *Oracle Application Server Reports Services Publishing Reports to the Web* for more information.

Table 10: Changed behavior of command line keyword *BACKGROUND*

Executable affected	BACKGROUND=YES	BACKGROUND=NO
<code>rwr</code>	The keyword <code>BACKGROUND</code> is obsolete with <code>rwr</code> and is ignored. Reason for obsolescence: In Oracle6i Reports, <code>BACKGROUND=NO</code> (default) displayed the <code>RWRUN60</code> interface (Runtime Previewer) for running the report interactively, while <code>BACKGROUND=YES</code> spawned another process to run the report and allowed the user to work on something else in the meantime. Since the <code>rwr60</code> interface is no longer available, the keyword <code>Background</code> does not serve any purpose with <code>rwr</code> .	
<code>rwclient</code>	Runs the job asynchronously.	Default. The client waits until the job is finished.
<code>rwservlet</code>	Runs the job asynchronously. (Note: In Oracle6i Reports, this was the default mode for <code>DESTYPES</code> printer and file)	Default. The client waits until the job is finished.
<code>rwsgi</code>	Same behavior as <code>rwservlet</code> . Note that <code>rwsgi</code> is available only for backward compatibility, and should be replaced by <code>rwservlet</code> .	
<code>rwserver</code>	Not applicable.	Not applicable.
<code>rwbuilder</code>	Not applicable. Causes a deprecated feature error, but continues working.	Not applicable.
<code>rwconverter</code>	Not applicable.	Not applicable.

CHANGED BEHAVIOR OF COMMAND LINE KEYWORD `BATCH`

In Oracle Reports 10g Release 2 (10.1.2), the command line option `BATCH=YES` specifies that the relevant executable should run in no-UI mode (that is, all terminal input and output should be suppressed). This command line keyword can be used only with the executables `rwserver` and `rwconverter`.

Action required

Make sure that your use of the command line keyword `BATCH` is consistent with Table 11. Refer to *Oracle Application Server Reports: Services Publishing Reports to the Web* for more information.

Table 11: Changed behavior of the command line keyword *BATCH*

Executable affected	Batch=Yes	Batch=No
<code>Rwr</code>	The keyword <code>BATCH</code> is obsolete with <code>rwr</code> . Reason for obsolescence: In Oracle6i Reports, <code>BATCH=NO</code> (default) displayed the <code>rwr60</code> interface (Runtime Previewer), while <code>BATCH=YES</code> suppressed it. Since <code>rwr60</code> interface is no longer available, the keyword <code>BATCH</code> does not serve any purpose with <code>rwr</code> .	
<code>Rwclient</code>	Not applicable.	Not applicable.
<code>rwservlet</code>	Not applicable.	Not applicable.

Executable affected	Batch=Yes	Batch=No
Rwcgi	Same behavior as <code>rwserverlet</code> . Note that <code>rwcgi</code> is available only for backward compatibility, and should be replaced by <code>rwserverlet</code> .	
<code>rwserver</code>	Reports Server dialog is not displayed.	Default. Reports Server dialog is displayed.
<code>rwbuilder</code>	Not applicable.	Not applicable.
<code>rwconverter</code>	Suppresses the Convert dialog box in order to convert reports/libraries without user intervention.	Default. The Convert dialog box is displayed.

EXECUTABLE NAMES

Refer to Table 12 for executable name changes from releases prior to Oracle9i Reports. Note that `xx` refers to the release number (for example, `rwblcxx` refers to `rwblc60` in Oracle6i Reports).

Table 12: Executable name changes

Component	Pre-9i	9i and 10g	
		Windows	UNIX
Reports Builder	<code>rwblcxx</code>	<code>rwbuilder.exe</code>	<code>rwbuilder</code>
Reports Runtime	<code>rwruncxx</code>	<code>rwrunc.exe</code>	<code>rwrunc</code>
Reports Server	<code>rwmtsxx</code>	<code>rwserver.exe</code>	<code>rwserver</code>
Proxy Server	-	<code>rwproxy.exe</code>	<code>rwproxy</code>
Reports Command Line Interface (Reports Client)	<code>rwclixx</code>	<code>rwclient.exe</code>	<code>rwclient</code>
Queue Manager	<code>rwrqmx</code>	<code>rwrqm.exe</code>	-
Queue Manager Unicode	<code>rwrquxx</code>	<code>rwrqu.exe</code>	-
Queue Viewer	<code>rwrqvxx</code>	-	<code>rwrqv</code>
Reports Converter	<code>rwconxx</code>	<code>rwconverter.exe</code>	<code>rwconverter</code>

ENVIRONMENT VARIABLE NAMES

In general, the release number has been dropped from the environment variable names. For example, `REPORTS60_PATH` is now `REPORTS_PATH`.

REPORTS BUILDER USER INTERFACE NAMES

The Report Editor in Reports Builder comprises the following views:

- Data Model view
- Paper Layout view (previously Layout Model view)
- Paper Design view (previously Live Previewer view)
- Paper Parameter Form view (previously Parameter Form view)
- Web Source view

Other user interface name changes in Reports Builder are:

- Previewer (previously named Runtime Previewer), displayed by choosing **File > Print Preview**.
- Property Inspector (previously named Property Palette).

ORACLE TECHNOLOGY NETWORK (OTN)

Here are some documents that will provide you more information about migrating from a previous release to Oracle Reports 10g Release 2 (10.1.2). These documents are available on OTN (<http://www.oracle.com/technology/products/reports>).

- Frequently Asked Questions on Migration
- Oracle Forms, Oracle Reports, Oracle Designer Statement of Direction
- Integrating Oracle Reports Services 10g in Oracle Forms Services 10g
- Oracle Application Server Reports Services Publishing Reports to the Web

ORACLE UNIVERSITY

Check with your local Oracle University representative on the availability of the new Instructor Led Training (ILT) course, *Oracle Reports Developer 10g: Move to the Web*. Also visit the [Oracle University Web site](#) for more information about this course and other Oracle University offerings.

ORACLE FUSION MIDDLEWARE

A Guide to Changed Functionality Between Oracle Reports 6i and 10g

August 2005

Author: Navneet Singh

Contributing Authors: Philipp Weckerle, Stewart Wilson, Rajesh Ramachandran, Rohit Marwaha, Ingrid Snedecor, Frank Rovitto, Ellen Gravina

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:

Phone: +1.650.506.7000

Fax: +1.650.506.7200

oracle.com

Copyright © 2005, Oracle. All rights reserved.

This document is provided for information purposes only and the contents hereof are subject to change without notice.

This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission. Oracle, JD Edwards, PeopleSoft, and Retek are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.