Oracle Forms Developer & Oracle Forms Server Release 6 (6.0 and 6*i*)

New Features Overview

March 2000

ORACLE

PRODUCT SUMMARY

Oracle Forms Developer is a productive development environment for building enterprise-class Internet database applications. Forms Developer provides a set of integrated builders that allow business developers to easily and quickly construct sophisticated database forms, charts, and business logic with minimal effort. Forms Developer's powerful declarative features enable business developers to create fully functional applications from database definitions with significantly reduced coding.

Oracle Forms Server is an application server that is optimized for deploying Forms Developer applications to the Internet. Oracle Forms Server delivers out-of-the-box functionality and native services to ensure that Forms Developer applications automatically scale and perform over any network, and to enable rich, extensible Java clients that are optimized for the Internet.

This paper describes many of the new features in Oracle Forms Developer and Oracle Forms Server Release 6, the latest major release of these products.

DEPLOYING INTERNET APPLICATIONS THAT SCALE AUTOMATICALLY

Oracle Forms Server Release 6 provides many new features to optimize scalability, minimize network traffic, and support application configuration and management. New features include:

- HTTP/1.1 support
- SSL support
- Standard web listener integration (Oracle Forms Server is shipped with WebDB 2.2 http listener)
- Out-of-the-box installation
- Integration with Oracle Enterprise Manager
- Load management
- Memory optimizations
- Network traffic optimizations
- Connection Activity Logging
- Forms Runtime Diagnostics
- Integration with Oracle Trace

HTTP/1.1 and SSL Support

Oracle Forms Server Release 6*i* introduces Internet support with HTTP/1.1 and SSL. To deploy applications to the Internet, use the Forms Server in one of three modes: sockets, HTTP, or HTTPS. A socket is a standard programming interface to TCP/IP. The socket mode of deployment is efficient and simple to use. The Forms Server runs on a networked host machine and listens on a specified socket or port for connections from the client running on a user machine. For this method to work, the client and server machines must be able to communicate with one another directly on the network. It is, therefore, not possible to use a server-side proxy (necessary for use with a firewall and other security measures) in sockets mode.

With HTTP, communication is also accomplished through a "socket connection," but it is now an HTTP socket connection. The Forms Server listens for HTTP connections from a Java client rather than for proprietary connections via specified sockets. All internal messaging between the Forms Server and the Java client is encapsulated in HTTP packets. An HTTP socket connection makes it possible for sites to allow secure communication between clients and servers through a firewall.

HTTPS is an HTTP connection with the addition of a Secure Socket Layer (SSL) protocol. This provides server authentication, secure data transmission, and data integrity. To enable an HTTPS connection, you must obtain a certificate from a Certification Authority (CA) and store this certificate on the server side using Oracle Wallet Manager (installed with Forms Server).

Using HTTPS, you can deploy Forms applications on the Internet securely with 128-bit encryption for domestic versions and 40 bit encryption for export versions.

Standard web listener integration

Oracle Forms Server Release 6 is now shipped with a standard open interface which allows the deployment of your applications with any web listener. This component enables the server to :

- Generate the initial HTML page on the fly
- Accept parameters through the URL
- Detect if the client is using Oracle JInitiator or a native JVM, and generate the right tags accordingly
- Dynamically balance the load across systems

After installing this interface, all you have to do is specify your custom configuration parameters under a named section in a configuration file (formsweb.cfg) that is provided with Oracle Forms Server. Set values for such parameters as load balancing, connect string, forms module, and the like. When you want to run the application, simply specify your named configuration in a query string in the application's URL (config=<section name>) and the parameters are automatically merged with an HTML template (one template for JInitiator, one template for the <APPLET> tag).

Use this new deployment method with any web listener, including Oracle Application Server and Oracle WebDB listener.

Out-of-the-box configuration

During installation, Oracle Forms Server Release 6 is automatically configured for you. The new Oracle installer provides a series of options where you specify the type of architecture you want to configure (single machine, primary node with load balancing without Forms Server, primary node with Load balancing with Forms Server, secondary node with load balancing). Once you specify your preferences, the installer does the rest. At the end of this process, the installer produces a text-based report that summarizes your configuration and provides instructions for any follow-up actions that may be required.

The installer also installs and configures WebDB listener 2.2 for you. Or, if you prefer, you can install any web server you choose.

Installation includes automatic generation of WebDB listener virtual directory mappings, environment variables, and configuration files.

6i   **Administration and Monitoring with Oracle Enterprise Manager**

To provide administration and management services, Oracle has integrated the Oracle Enterprise Manager (OEM) with the Forms Server. OEM provides an open, comprehensive framework and rich tool set to manage and administer the complete Oracle environment from an easy-to-use, centralized console. Integration with OEM enables system administrators to monitor and administer many Forms Servers from one location.

The OEM Intelligent Agent automatically "discovers" Forms Server components and enables centralized management of them from the OEM console's Navigator tree. With OEM, system administrators can start-up and shut-down the Forms Servers and monitor availability, memory usage, CPU usage, and user processes.
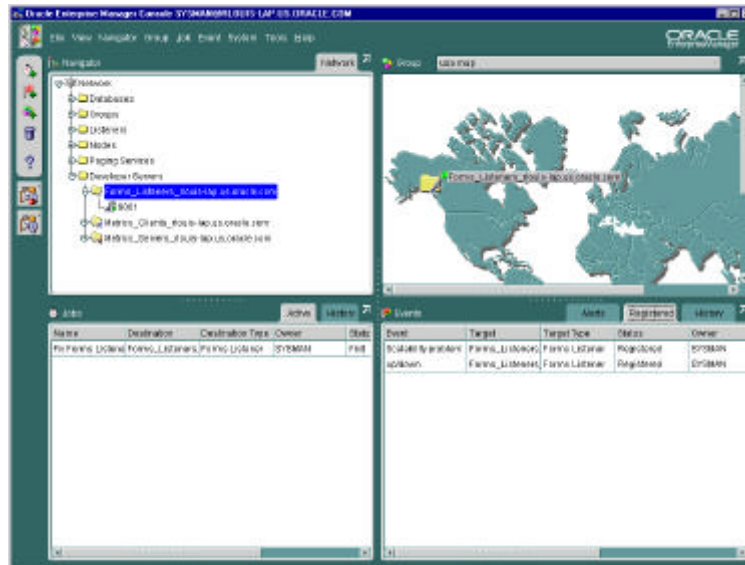

Figure 1 : OEM administration console

Load Management

Oracle Forms Server load management supports a distributed environment, with no single point of failure, and promotes the most efficient use of hardware and system resources. It enables system administrators to control and manage the user load on their application servers. With the Forms Server, systems administrators can dynamically balance the load across systems, allocating resources based on a configurable least-loaded-host algorithm.

Memory Optimizations

Oracle Forms Server Release 6 delivers significant memory optimizations that reduce the overall processing load on the server. It does this through improved dynamic link library sharing under Windows NT, improved middle-tier record caching, and enhancements to the messaging layer.

> *Note: Refer to the White Paper "Oracle Forms Server 6 - A Capacity Planning Guide," for more details. White papers are available from the Oracle Technology Network (http://technet.oracle.com) or the Oracle web site (http://www.oracle.com).*

Network Traffic Optimizations

A number of network traffic optimizations have been made. These include:
- Message diffing
- Smart event bundling
- Message optimizations
- Lightweight menu-bar

With message diffing, when sets of similar messages are sent across the network, the second and subsequent messages include only the differences from the previous message. This significantly reduces network traffic. Smart event bundling "bundles" all of the events that are triggered during a navigation between two objects and delivers them as a single packet to the server for processing. This reduces the number of network round-trips. New message optimizations between the client and the middle-tier have shown network traffic reductions of up to 25 percent over earlier version (Release 5.0). The new lightweight menu-bar for the Web client reduces processing and screen redrawing associated with menus.

> *Note: Refer to the White Paper "Oracle Developer Server: How to Tune for the Deployment of Internet Applications" for more details. White papers are available from the Oracle Technology Network (http://technet.oracle.com) or the Oracle web site (http://www.oracle.com).*

Connection Activity Logging

To facilitate more effective problem diagnosis and resolution, Oracle has added connection activity logging to the Forms Server. Forms Server logging messages are written to a log file that you specify. The log file includes information about server start-up, connection requests, connection establishment, disconnection, and abnormal terminations. It also records IP addresses, port numbers, and process ID information to enable the server administrator to determine which processes belong to which users. Connection activity logging has a very low overhead and provides significant value for problem resolution.

Forms Runtime Diagnostics

Forms Runtime Diagnostics (FRD) is a runtime, event-based logging system designed to help you debug your application. When a form is run with FRD enabled, a combination of external user-application interactions and internal Forms processing events are written in chronological order to a log file. These events can be analyzed to determine user actions and corresponding system responses, which aid in problem diagnosis and issue resolution.

Integration with Oracle Trace

Oracle Trace is a utility that enables the collection of performance data. It is shipped with Oracle Forms Release 6 as part of Oracle Enterprise Manager's Diagnostic Pack. Using this feature you can generate trace files during an application's execution and analyze the results with Oracle Trace. For each Forms session, you can analyze the execution time, the CPU consumption, and the network traffic for the whole session. Use Oracle Trace to perform a drill-down analysis of the performance of a specific form, a particular trigger in a form, and even a built-in. With Oracle Trace integration, you can easily detect a possible performance problem and determine its origin.

Figure 2 : Oracle Trace viewer shows you the performance of each built-in used in the application

DELIVERING SOLUTIONS IN WEB-TIME

To respond to changing business requirements, development teams must be able to rapidly deliver and modify applications. The new features in Oracle Forms enable development teams to get up to speed quickly and respond rapidly to changing user requirements. The new features leverage the strengths of Java, promote re-use, and enable business developers to build rich, extensible user interfaces and complex business logic with minimal effort.

JavaBeans and Pluggable Java Components

As Oracle Forms Developer and Oracle Forms Server evolve, Oracle is phasing in plans to increase the product architecture's openness to Java. Release 6 marks completion of the first phase. In this release, Oracle Forms Developer and Oracle Forms Server have enabled business developers to extend the client Java Applet using JavaBeans and Pluggable Java Components (PJCs). Use Oracle JDeveloper to build these components, or, if you prefer, use a third-party 3GL Java development environment.

Use JavaBeans to easily extend and customize a form's functionality. JavaBeans can be integrated into a Forms Developer application in much the same way that OCX controls could be embedded with the previous releases. Embed JavaBeans via the Layout Editor, then specify their Implementation Class in the Property Palette.

To provide greater flexibility and control over an application's user interface, Release 6 supports PJCs. PJCs enable you to extend, customize, or override any native Forms widget. Each widget has an associated Implementation Class property where you can specify the desired corresponding Java class. For example, you can extend or replace the standard Forms push-button with a custom button, without losing the business logic that was attached to the original push-button.

> *Note: Refer to the White Paper "Using Java Components in Oracle Forms Applications" for more details. White Papers are available from the Oracle Technology Network (http://technet.oracle.com) or the Oracle web site (http://www.oracle.com)*
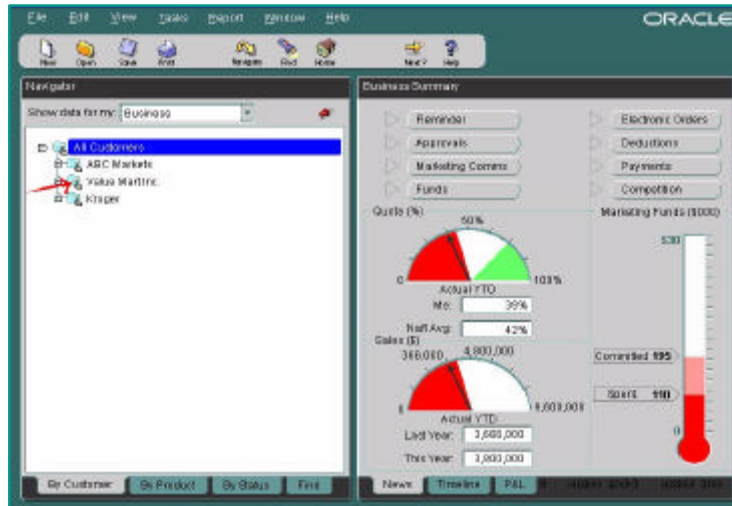
Figure 3 : Java Bean and Pluggable Java Component integration example

Hierarchical Tree Control

Hierarchical Tree Control (the Navigator) has long been a feature of the Form Builder component of Oracle Forms Developer. In Release 6, we added it to Forms Runtime. Display hierarchical information in this easily visualized format. Populate a hierarchical tree with values contained in a Record Group or Query. At runtime, programmatically add, remove, modify, or evaluate elements in a hierarchical tree. Seamlessly deploy this control to the Internet with Oracle Forms Server.

Figure 4: A sample application using the native Hierarchical Tree Control to display data in the form of a standard navigator.

Pop-up Menu Enhancements

We enhanced pop-up menus in Oracle Forms Developer Release 6 to provide support for images, check boxes, and radio buttons. Add images and other controls to pop-up menus to provide consistency across applications and help users quickly familiarize themselves with application functionality through these visual enhancements.

Figure 5: Oracle Forms Developer Release 6 supports icons in the pop-up menus.

New Look and Feel

Oracle Forms Developer includes a new user interface (UI). Designed by Oracle Corporation's usability experts, the new UI sets a higher standard for Java applications, exceeding even the quality of the Windows desktop. It adds unprecedented richness to Web-deployed applications, including:

- Animated controls in user dialogs
- Enhanced user controls, including rounded buttons and scroll bars
- Highlighting for drop-down lists and tab-controls
- Support for multiple document interface (MDI)

The new UI includes a choice of color schemes optimized for deployment with Oracle Forms Server.
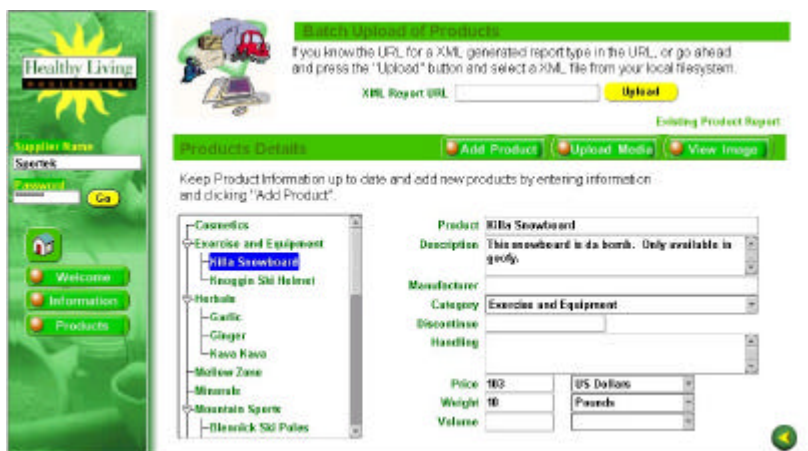

Figure 6: The new look and feel sets a higher standard for web applications.

PL/SQL Editor Enhancements

Oracle has enhanced the PL/SQL Editor for the Windows platform. The new editor supports automatic indenting; color syntax highlighting for keywords, comments, and strings; and column and line selection for selecting columns of text and lines of source code. The new editor provides drag-and-drop text manipulation, multiple split views, printing, and unlimited undo/redo (which lets you undo or redo all edit actions up to the last save operation).

The new global search and replace engine supports powerful expression-based searching for easily locating and modifying code across client-side program units.

Finally, the Syntax Palette not only helps you get up to speed rapidly on the tool, but also allows you to focus on specific sections of application logic within code blocks, greatly enhancing the construction of syntactically correct PL/SQL code.

LOV Wizard

The LOV Wizard guides you step-by-step through building a List of Values (LOV). As with all of the wizards in Oracle Forms Developer, the LOV Wizard is re-entrant; this means you can re-enter the wizard and make changes to previously completed actions. Through tight integration with Query Builder, the LOV Wizard allows you to specify database tables and columns with a few simple mouse-clicks. It also provides complete control over the layout by allowing you to specify the title and position of the LOV on the screen.

Web Preview

Developers will notice an important change in Form Builder that will simplify the testing of applications for Internet deployment. In Oracle Forms Developer Release 6, Form Builder includes a built-in Java Virtual Machine (JVM). The JVM allows you test applications running on the Web from directly within the Builder environment. Use Web Preview to go back and forth between development and runtime environments within the same interface.

Builder Enhancements

Developers will also notice some visual changes in the Builders. We've added "cool bars" in the Builder that conform to Windows-style flat toolbars. The new color, pattern, and font picker in the Property Palette makes selecting colors and styles much more intuitive.

Support for Oracle8 Objects

We have enhanced the re-entrant wizards in Oracle Forms Developer Release 6 to include support for Oracle8 objects. In Form Builder, this means you can base a block on an object table or a relational table with an object column. This includes support for REFs and declarative support from the Data Block Wizard for an LOV based on an REF item. Graphics Builder provides basic Oracle8 type support in charts and graphics displays. All of the components also benefit from the Oracle8 enhancements added to Query Builder that support the construction of queries containing Oracle8 data types, including REFs, BLOBs, and CLOBs, and the parsing of SQL statements and queries with Oracle8 expressions.
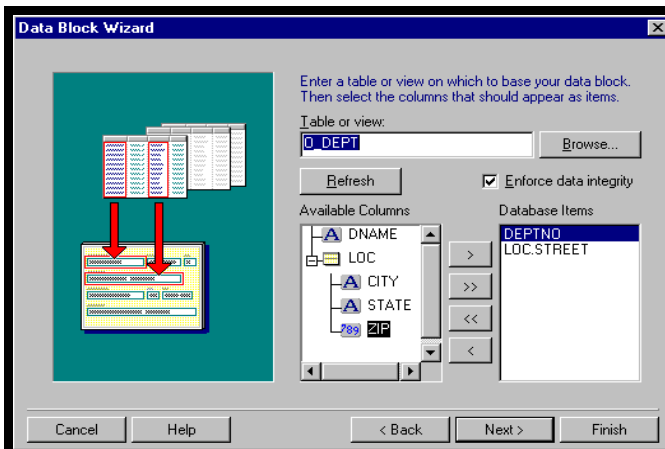

Figure 7: Oracle Forms Release 6 provides native declarative support for Oracle8 Objects.

Open Client Adapter

In Oracle Forms Server Release 6, enhancements to the EXEC_SQL package and Open Client Adapter (OCA) improve native support for foreign databases and multiple simultaneous connections. This package is now accessible as a native built-in from Forms.

CONCLUSION

In today's corporate environments, overwhelming demands are placed on information systems to readily adapt to changing business requirements and emerging technologies. Development teams are continually faced with demands for enhanced application functionality; improved user interfaces; and more complex, secure, and scalable deployment configurations. By leveraging the Internet platform through openness, inherent scalability, and high performance, Oracle Forms Developer and Oracle Forms Server Release 6 provide a unique environment for delivering mission-critical database applications that take advantage of each new wave of emerging technology.

ORACLE