

# Oracle Database 11g Application Development

*An Oracle White Paper*  
*June 2007*

# Oracle Database 11g Application Development

APPLICATION DEVELOPMENT CHALLENGES .....	4
SHORTEN DEVELOPMENT TIME TO MARKET .....	4
Oracle SQL Developer .....	5
Oracle SQL Developer: Creating and Editing Objects .....	5
Oracle SQL Developer: SQL Worksheet .....	5
Oracle SQL Developer: PLSQL Editing and Debugging.....	5
Oracle SQL Developer: Reporting.....	6
Oracle SQL Developer: Migration Workbench.....	6
Oracle SQL Developer: Oracle Application Express 3.0.1 .....	6
Oracle Application Express.....	7
Oracle Application Express: PDF Printing.....	7
Oracle Application Express: Microsoft Access Migration.....	8
Oracle Application Express: Flash Charts.....	8
Oracle Application Express: Drag & Drop Item Layout.....	8
Oracle Application Express: New Item Types .....	8
Oracle Application Express: Supporting Objects .....	9
Oracle Application Express: Page and Region Caching.....	9
Oracle Application Express: Other Enhancements .....	9
Cross Language Features .....	10
Client Result Cache .....	10
Server Result Cache .....	10
Secure LOBs and Prefetching of LOB Data.....	10
.NET.....	11
.NET: Visual Studio 2005 Integration .....	12
.NET: A Fully Integrated PL/SQL Debugger.....	12
.NET: 64-bit Support .....	12
.NET: ADO.NET 2.0 Support.....	12
.NET: Performance Enhancements .....	13
PHP.....	13
PHP: Real Application Clusters .....	13
PHP: Result Caching .....	13
PHP: Prefetching Rows.....	14
PL/SQL .....	14
PL/SQL: Real Native Compilation .....	14
PL/SQL: Function Result Cache.....	14
PL/SQL: Fine Grained Dependency Tracking .....	15
PL/SQL: Sequences.....	15

PL/SQL: PL/Scope.....	15
PL/SQL: Hierarchical Profiler.....	16
Java.....	16
Java: OracleJVM Performance, JIT Compiler .....	16
Java: Productivity and Portability through Java Standards.....	16
Java: JDBC 4.0.....	17
Java: OracleJVM Compliance with Java 5 .....	17
Java: RowSet (JSR-114) Comes to Server-side JDBC.....	17
Java: Exposing Oracle to Java .....	17
Java: Prefetch in 1st Roundtrip .....	17
Java: Native AQ Protocol .....	17
Java: Database/Query Change Notification.....	17
Java: Server and Client Result Cache.....	18
Java: Security .....	18
Java: Oracle JVM Ease of Use.....	18
Java: Manageability.....	18
Oracle Call Interface (OCI).....	18
Database Resident Connection Pooling .....	18
Implicit Fetching of ROWIDs .....	19
Binding and Defining Multiple Buffers .....	19
Allocating an Array of Descriptors.....	19
CONCLUSION .....	19

# Oracle Database 11g Application Development

## APPLICATION DEVELOPMENT CHALLENGES

The modern IT organization is faced with a multitude of application development languages and tools to choose from. In some cases multiple application development products are used within a single organization. Organizations that have grown through mergers and acquisitions are faced with the difficult problem of managing and integrating a large collection of competing application development technologies that were used in predecessor organizations. Technology enables business innovation and those organizations that are able to quickly deliver applications gain competitive advantage.

## SHORTEN DEVELOPMENT TIME TO MARKET

Oracle Database 11g provides a single integrated platform offering high performance and scalability for the most popular technologies used by application developers today. Oracle Database 11g adds significant new capabilities to all the major application development environments, enabling you to increase developer productivity and shorten time to market.

Application developers have a choice of languages for creating Oracle database-centric applications including C, C++, Java, COBOL, PL/SQL, Visual Basic, C# and PHP. Developers can choose the language in which they are most proficient or one that is most suitable for a specific task. For example an application might use Java on the server to create dynamic Web pages, PL/SQL to implement stored procedures in the database, and C++ to implement computationally intensive logic in the middle tier. Oracle Database 11g offers customers the power and performance of the industry's leading database delivered across a wide variety of the most popular application development technologies.

This paper provides an overview of new application development features in 11g for the following products and technologies.

- Oracle SQL Developer
- Oracle Application Express
- Cross Language Features
- .NET
- PHP

Oracle Database 11g supports a wide variety of application development technology with tools that will improve the performance of your applications and the productivity of your developers.

- PL/SQL
- Java
- OCI

A powerful and free database development tool.

## Oracle SQL Developer

Oracle SQL Developer is a free graphical tool that enhances productivity and simplifies database development tasks. Designed for Oracle Database developers, Oracle SQL Developer simplifies development cycles and reduces the need to buy third-party tools for developing and debugging SQL and PL/SQL code. Using Oracle SQL Developer, users can browse, create and modify database objects, run SQL statements, edit and debug PL/SQL and can run reports from an extensive list of predefined reports or create their own. The introduction of this tool underscores Oracle's commitment to improving the productivity and supporting the needs of the database developer community. Oracle SQL Developer 1.1.3 ships with Oracle Database 11g. SQL Developer 1.2 is available for download at the SQL Developer Web site. For more information about Oracle SQL Developer, go to: [http://www.oracle.com/technology/products/database/sql\\_developer](http://www.oracle.com/technology/products/database/sql_developer)

### Oracle SQL Developer: Creating and Editing Objects

Users can connect to any target Oracle database schema using standard Oracle database authentication. Once connected, they can create, edit and update database objects. As new objects are created or existing objects edited, the DDL for those adjustments is available for review. An Export DDL option is available for users wanting to create the full DDL for one or more objects in the schema.

### Oracle SQL Developer: SQL Worksheet

The SQL Worksheet supports the creation of SQL, PL/SQL and SQL \*Plus commands. These can be run individually or consecutively. A SQL History option makes recalling previous commands easy, while the Explain Plan option allows users to see the execution plan for selected statements.

### Oracle SQL Developer: PLSQL Editing and Debugging

Using this robust editing environment, users can create and edit PL/SQL, take advantage of the code formatting, add bookmarks and use code insight. When it comes to debugging PL/SQL, breakpoints, smart data, a debugger stack and watches are all available. These features allow the user to set a break point and run and test the code, supplying alternate data at runtime while debugging. Creating PL/SQL in the editor or using the SQL Worksheet is made easier by the availability of snippets. Snippets are code fragments, such as SQL functions, Optimizer hints or miscellaneous PL/SQL programming techniques, which users can drag onto the PL/SQL Editor or the SQL Worksheet.

### **Oracle SQL Developer: Reporting**

SQL Developer provides a number of predefined reports about the database and its objects. Any report can be pushed over to a SQL Worksheet and manipulated further. SQL Developer allows users to create, save and share user-defined reports for repeated use. Report styles include charts and master-detail reports.

### **Oracle SQL Developer: Migration Workbench**

**Eases migrations of third-party databases to Oracle.**

Oracle SQL Developer Migration Workbench was added to Oracle SQL Developer 1.2 and includes the following features:

- Full integration with Oracle SQL Developer means that users have one tool to browse third-party databases (MySQL, Microsoft SQL Server and Microsoft Access), migrate selected objects to Oracle and then work with them.
- Quick Migration Wizard offers least privilege migration i.e. No DBA privilege is required.
- Step driven migration offers control at each stage of the migration process.
- Fine grain migration support provides users with the ability to select specific objects for migration.
- Complex object migration supports stored procedures, triggers and views.
- Translation Scratch Editor offers single statement migration.
- Translation Difference Viewer provides side by side comparison by matching statements and blocks.

Please note: Oracle Migration Workbench (a separate and standalone product) can be used to migrate Sybase, Informix, and DB2 databases to Oracle databases.

### **Oracle SQL Developer: Oracle Application Express 3.0.1**

Oracle SQL Developer 1.2 supports Oracle Application Express 3.0.1 objects and includes the following features:

- Connect to and browse all Oracle APEX applications associated with a database schema (at application and page level).
- Run a number of reports that allow users to report on Workspaces, Applications, Pages and Schemas.
- Export and Import Oracle APEX applications using a context menu.

**A unique, free development tool for building Web applications.**

## **Oracle Application Express**

Oracle Application Express (Oracle APEX), formerly called HTML DB, is a rapid Web application development tool for the Oracle Database. Using only a Web browser and limited programming experience, you can develop and deploy professional applications that are both fast and secure. No client software is required to develop, deploy or run Oracle Application Express applications. Oracle Application Express is built for the Web and combines the qualities of a personal database (i.e., productivity, ease of use and flexibility) with the qualities of an enterprise database (i.e., security, integrity, scalability and availability).

Oracle Application Express provides three primary tools:

- Application Builder - to create dynamic database driven Web applications.
- SQL Workshop - to browse database objects, run ad-hoc SQL queries, as well as a graphical query builder.
- Utilities - to load and unload data from both flat files and spreadsheets.

With Oracle Application Express you can easily build applications that report on database data. Reports can be hypertext linked with other reports allowing users to navigate through database data in the same way they navigate Web sites. Columns in reports can be easily linked to other reports, charts and data entry forms. An extensive charting engine allows SQL queries to be represented graphically. Oracle Application Express is also very adept at editing database data and supports a large number of declarative form controls including radio groups, checkboxes, select lists, shuttles, text editors and date pickers.

Due to its ease of use, Oracle Application Express is an ideal tool for quickly building opportunistic and departmental applications. Traditionally, prototypes and small applications have been built using desktop databases. These databases are difficult to manage when propagated across many desktop computers. They can also be inappropriate for use with sensitive information and they are typically not Web friendly. The browser-based design time interface, declarative programming framework and simple wizards make Oracle Application Express a natural replacement for multi-user desktop database applications such as Microsoft Access.

SQL-savvy application developers with little or no Web development experience can easily create database applications with Oracle Application Express. You don't need to learn scripting languages or complex deployment frameworks, you simply write a few queries and choose from the provided set of user interface themes and form controls to create highly professional, secure, and scalable applications. For more information about Oracle Application Express, go to:

[http://www.oracle.com/technology/products/database/application\\_express](http://www.oracle.com/technology/products/database/application_express)

### **Oracle Application Express: PDF Printing**

You can export a report region to PDF - essentially, printing a report. Integration with Oracle Business Intelligence (BI) Publisher enables “high fidelity” reports to be produced. BI Publisher provides a Microsoft Word plug-in to develop complex

report templates that can incorporate multiple tables (e.g., master – detail), charts, and other parameters passed from Oracle Application Express.

**Migrate Microsoft Access applications and databases to Oracle to consolidate, control and centrally manage important data.**

#### **Oracle Application Express: Microsoft Access Migration**

Microsoft Access Migration enables you to generate an Oracle Application Express application based upon a Microsoft Access application. The migration process involves the following steps:

1. Use the Exporter tool to export metadata from Microsoft Access.
2. Use Oracle SQL Developer Migration Workbench (included in Oracle SQL Developer 1.2) to migrate a Microsoft Access database to an Oracle database.
3. Create an Oracle Application Express workspace and then a migration project.
4. Validate and update the retrieved objects.
5. Generate an Oracle Application Express application.

#### **Oracle Application Express: Flash Charts**

Applications can be built to use Adobe’s Flash Player browser plug-in to display charts. There are 18 types of Flash Charts supported. During the creation process you can preview the selected type and set most attributes for your resulting chart. In addition to creating new Flash Charts, you can also convert your existing SVG Charts to Flash Charts. Flash Charts enable far greater controls on the user interface and also includes features such as asynchronous updates.

#### **Oracle Application Express: Drag & Drop Item Layout**

With the new Drag and Drop Item Layout feature, you can reorder items within the current region, change select attributes (item name, label, and type within the current item-type range), delete items and quickly create new items. You can place items to the left or right of an existing item and can insert a new row to quickly add items in the middle of an existing form. To remove an item, simply drag it into the recycle bucket at the bottom.

#### **Oracle Application Express: New Item Types**

Several new items types have been added. These include:

- Shuttle
- HTML Editor Minimal - available under Textarea
- HTML Editor Standard - available under Textarea
- Popup Color Picker - available under Popup List of Values
- Date Picker (use format mask)

### **Oracle Application Express: Supporting Objects**

The Supporting Objects feature introduced in Oracle Application Express 2.2 has been enhanced to allow for the definition of upgrade scripts. You can use this feature for distributing upgrades. We have also added the ability to include your Access Control Table definitions with your installation scripts.

### **Oracle Application Express: Page and Region Caching**

This allows you to write parts of your application to a cache to improve performance. The cache attributes are set on the Page and Region attributes pages. This is great for static pages and regions such as lists that do not have any conditions and regions containing HTML text.

### **Oracle Application Express: Other Enhancements**

There are several other enhancements introduced in this release including:

- Calendar – Default Calendars now include Monthly and Daily views.
- Shared components – Additional abilities copying and re-parenting Lists and Breadcrumbs, etc.
- Item Finder – Searching of Cascading Style Sheets (CSS) and Images.
- Multi-Delete – Ability to delete multiple Buttons, Processes, Computations, and Validations.
- Navigation – New icons were added to navigate to Shared Components and Application Reports.
- Application & Schema Comparison – Identify differences between two applications, or objects in two different schemas.
- Application Builder Defaults – Define workspace preferences for Tabs, Authentication, Themes and Globalization to enhance consistency.
- Developer Preferences – Define your preferences for view mode within the Application Builder.
- Support for Public Pages – Allow a Session ID of zero for public applications to assist with search engines and bookmarks.
- SQL Workshop – Enhanced navigation and improved schema access.
- New Password and Account Controls – Specify rules for password expiration, strength (minimum characters, etc.), and locking.
- Workspace Management – Define sizes for workspaces, request an email of Workspace names, and review a log of login attempts.
- New Look for Managing Users – Simplified navigation and enhanced icons.

## Cross Language Features

The following application development features are available across multiple programming languages. Some of the features are implemented within the database server. Client-side features are provided by OCI, Oracle's native C programming interface. They are also available to other programming language interfaces such as .NET, PHP, JDBC type 2 driver (JDBC-OCI), precompilers and OCCI all of which are built on top of OCI.

### Client Result Cache

This feature enables client-side caching of query results in client memory. The caching of result sets is completely transparent to the application, and the cache of the result set data is internally kept consistent with any session or database changes that affect the result set.

Because retrieving results locally from the client memory is much faster than making a database call, processing the query, and retrieving results from the disk, frequently executed queries experience a significant performance improvement when their results are cached. Utilization of the Client Result Cache also reduces the server CPU that would have otherwise been consumed for processing the query, thereby improving server scalability as well. Internal testing of a standard benchmark shows response time improvements of up to 22% and 6.5 times reduction in server CPU consumption.

Client Result Cache can be enabled by setting the `CLIENT_RESULT_CACHE_SIZE` initialization parameter. Users can annotate a query with the `result_cache` hint to indicate that results are to be stored in the result cache.

### Server Result Cache

Results of queries can also be cached in the server result cache (within the shared pool). Users can annotate a query with the `result_cache` hint to indicate that results are to be stored in the result cache.

You can set the `RESULT_CACHE_MODE` initialization parameter to control whether the server result cache is used for all queries (when possible), or only for queries that are annotated.

### Secure LOBs and Prefetching of LOB Data

Several enhancements have been made to reduce the storage requirements for LOB data and increase performance in accessing it.

For Oracle SecureFiles (LOBs with `STORE` as `SECUREFILE` option), one can specify the SQL parameter `DEDUPLICATE` in `CREATE TABLE` and `ALTER TABLE` statements. This enables the application to specify that LOB data that are identical in two or more rows in LOB column will all share the same data block, thus saving disk space. Similarly, the parameter `COMPRESS` turns on LOB

**Significant performance gains are possible for queries that are invoked frequently with data that changes slowly.**

compression. The parameter ENCRYPT turns on LOB encryption and optionally selects an encryption algorithm. Each LOB column can have its own encryption specification, independent of the encryption of other LOB or non-LOB columns.

To improve access of smaller LOBs, LOB data can be prefetched and cached while also fetching the locator.

## **.NET**

### **Tight integration with Visual Studio.**

The Oracle Developer Tools for Visual Studio .NET (ODT) is a tightly integrated "Add-in" for Microsoft Visual Studio. ODT is free and it is available for both Visual Studio 2005 and Visual Studio .NET 2003. ODT makes developing .NET code for Oracle easy and fast and allows developers to stay in Visual Studio for the entire development lifecycle. ODT provides the ability to browse and edit Oracle schema objects using integrated visual designers and to automatically generate .NET code by simply dragging and dropping. Developers can easily modify table data, execute Oracle SQL statements, edit and debug PL/SQL code and develop and deploy .NET stored procedures. The integrated context sensitive online help, including the Oracle SQL and PL/SQL Users Guides puts the Oracle documentation at your fingertips. For more information about Oracle Developer Tools for Visual Studio .NET, go to:

<http://www.oracle.com/technology/tech/dotnet/tools>

### **Oracle Data Provider for .NET brings the power of Oracle Database 11g to .NET applications.**

The Oracle Data Provider for .NET (ODP.NET) features optimized data access to Oracle databases from a .NET environment. ODP.NET allows developers to take advantage of advanced Oracle Database functionality, including Real Application Clusters, performance optimizations, XML DB, and advanced security features. ODP.NET gives programmers better performance, flexibility, and feature choice for their .NET applications. With it, developers can use .NET, but not sacrifice the powerful data management capabilities that Oracle provides. ODP.NET natively supports .NET Framework 1.0 and higher, including the .NET Framework for 32-bit Windows, Windows x64 (AMD64 and Intel EM64T), and 64-bit Windows for Intel Itanium. For more information about Oracle Data Provider for .NET, go to:

<http://www.oracle.com/technology/tech/windows/odpnet>

The Oracle Database Extensions for .NET is a feature of Oracle Database 11g on Windows that makes it easy to develop, deploy, and run stored procedures and functions written in a .NET managed language such as C# or VB.NET. .NET Stored procedures or functions are developed using Microsoft Visual Studio and deployed using the tightly integrated .NET Deployment Wizard which is a feature of the Oracle Developer Tools for Visual Studio .NET. After deployment, a .NET stored procedure can be called from within .NET application code, from SQL or PL/SQL, from another .NET, PL/SQL or Java stored procedure, from a trigger, or from anywhere else a stored procedure or function call is allowed. For more information about Oracle Database Extensions for .NET, go to:

<http://www.oracle.com/technology/tech/dotnet/ode>

### **.NET: Visual Studio 2005 Integration**

The Oracle Developer Tools for Visual Studio .NET is fully integrated with Visual Studio 2005. A tree control makes it easy to browse database schema objects. Each schema object node offers context-sensitive menus that allow operations on that node. For example, a table node offers a “Retrieve Data” menu item, which opens up the Oracle Data Window. It also offers a “Design” menu item that opens up the Oracle Table Designer for easy creation or modification of an Oracle table. Similarly, other Oracle schema types have item-specific menus. The Visual Studio Properties Window shows metadata for any Oracle Database schema node that is currently selected. For example, selecting a table column node will show the Oracle data type of that column in the Properties Window.

For rapid application development, ODT is also fully integrated with Visual Studio’s automatic code generation features. Dragging and dropping onto the design surface immediately creates runnable .NET code which accesses an Oracle Database. This code can be easily and visually wired up to widgets on the design surface (such as a Gridview control) resulting in a runnable application with very little coding required.

### **.NET: A Fully Integrated PL/SQL Debugger**

The fully integrated PL/SQL Editor and Debugger allows you to leverage all of your favorite Visual Studio debugging features from within PL/SQL code. You can even seamlessly step from your .NET code into your PL/SQL stored procedure code and back out again!

### **.NET: 64-bit Support**

ODP.NET supports 64-bit computing for Windows x64 and Windows Itanium resulting in more scalable .NET applications.

### **.NET: ADO.NET 2.0 Support**

Microsoft ADO.NET 2.0 support, including:

- Provider factory classes and base classes simplify data access code to multiple data sources with a provider generic API.
- Connection string builder makes creating connections strings less error-prone and easier to manage.
- Local and distributed transactions in System.Transactions are supported.
- Batch processing enables multiple row changes in a single database round trip using the OracleDataAdapter.Update method.
- Schema discovery permits application developers to find and return database schema information, such as tables and stored procedures.
- Data source enumerator enables an application to generically obtain a collection of the Oracle data sources that are available.

### **.NET: Performance Enhancements**

ODP.NET 11g introduces new performance enhancements, including:

- Using statement caching, ODP.NET parameter contexts are now cached automatically, providing faster performance when the same statement is repeatedly executed.
- Using Oracle Database 11g, ODP.NET users will experience faster LOB retrieval performance (as described in the Cross Language Features section).
- Using Client Result Cache (as described in the Cross Language Features section), ODP.NET applications executing frequently run queries will experience a significant performance improvement.

### **PHP**

**Oracle Database 11g for highly scalable  
PHP applications.**

PHP is a popular, interpreted scripting language commonly used for Web applications. PHP is used for rapidly developing applications both big and small, and is useful for creating Web 2.0 applications. It powers over twenty million Web sites on the Internet and has a large user community. PHP runs on many operating systems and Web servers.

PHP is increasingly seen in mission-critical applications that require large, scalable databases. Oracle has partnered with the open source community to create the OCI8 extension which is a stable, high-performance PHP database driver that is fully integrated with Oracle Database. Using PHP with Oracle Database, you can query and update data, execute stored procedures and functions, load images, and easily build scalable, global applications. The new features of Oracle Database 11g give PHP developers even more reason to develop high performance, scalable and easily managed applications using Oracle Database. For more information about Oracle and PHP, go to:

<http://www.oracle.com/technology/tech/php>

### **PHP: Real Application Clusters**

PHP applications can greatly benefit from the scalability and high availability of Oracle's Real Application Clusters (RAC). RAC is a shared-disk implementation of Oracle Database that allows multiple instances to simultaneously access the same database. RAC's unique ability to run and scale real application workloads on a cluster of servers is an essential foundation of Enterprise Grids.

### **PHP: Result Caching**

Oracle Database 11g includes result caching. Result caches greatly speed the repeated execution of queries (see Cross Language Features section) and function calls (see PL/SQL section) that access read-only or read-mostly data.

### **PHP: Prefetching Rows**

Database query results are returned in groups of rows, rather than one row at a time or all at once. Tuning the number of rows returned can have a dramatic improvement on the performance of PHP applications by reducing network round-trips, and making available the appropriate number of results for an application.

### **PL/SQL**

PL/SQL is an imperative 3GL that was designed specifically for the seamless processing of SQL commands. It provides specific syntax for this purpose and supports exactly the same datatypes as SQL. While it is available in other environments, this document focuses on PL/SQL that is stored and compiled in Oracle Database and that runs within the Oracle executable where it automatically inherits the portability of Oracle Database.

A best practice used by many Oracle customers is to have client code access Oracle Database only by calling PL/SQL subprograms. This approach not only reflects generic modular programming best practices (define a clean, functional API and hide the implementation); also, it typically drastically reduces network round trips between the client and Oracle Database. For more information about PL/SQL, go to:

[http://www.oracle.com/technology/tech/pl\\_sql](http://www.oracle.com/technology/tech/pl_sql)

### **PL/SQL: Real Native Compilation**

Prior to Oracle Database 11g it was possible to speed up PL/SQL procedures by compiling them into native code residing in shared libraries. These procedures were translated into C code, then compiled with a C compiler and linked into the Oracle process. In Oracle Database 11g, the need for a C compiler vanishes and PL/SQL source is directly converted into native code. Moreover, Oracle does the linking and loading so that file system directories are no longer needed. An individual developer can now compile program units for native execution without relying on any set-up on the part of the DBA. Compiling PL/SQL code natively is about twice as fast in Oracle Database 11g as it was in Oracle Database 10g. The run-time performance also improves noticeably.

**PL/SQL can now be compiled without the need to install a C compiler.**

### **PL/SQL: Function Result Cache**

A PL/SQL function is sometimes used to return the result of a computation whose inputs are one or several parameterized queries issued by the function. In some cases, these queries access data (for example, the catalog of wares in a shopping application) that changes very infrequently compared to the frequency of calling the function. You can include syntax in the source text of a PL/SQL function to request that its results be cached and, to ensure correctness, that the cache be purged when any of a list of tables experiences DML. The cache, like the SQL query result cache, is in the shared pool. The DBA manages and monitors both result caches using common APIs. The look-up key for the cache is the

combination of actual arguments with which the function is invoked. When a particular invocation of the result-cached function is a cache hit, then the function body is not executed; instead, the cached value is returned immediately.

### **PL/SQL: Fine Grained Dependency Tracking**

In previous releases, metadata recorded mutual dependencies between objects with the granularity of the whole object. For example, that PL/SQL unit P depends on PL/SQL unit Q or that view V depends on table T. This means that dependent objects were sometimes invalidated when there was no logical requirement to do so. For example, if view V depends only on columns C1, C2, and C3 in table T and a new column, C99, is added, the validity of view V is not logically affected. Nevertheless, in earlier releases, V was invalidated by the addition of column C99.

Oracle Database 11g records dependency metadata at a finer level of granularity so that the addition of C99 does not invalidate view V. Similarly, if procedure P depends only on elements E1 and E2 in package PKG, then if element E99 is added to PKG, procedure P is not invalidated. (In Oracle Database 10g, this change to PKG would invalidate procedure P.)

By reducing the consequential invalidation of dependent objects in response to changes in the objects they depend upon, application availability is dramatically increased. The benefit is felt both in the development environment and when a deployed application is patched or upgraded. The benefit occurs when an Oracle Database patchset is applied because changes to schema objects are required to be compatible and, therefore, will not now cause consequential invalidations.

### **PL/SQL: Sequences**

**Simplifies coding.**

The sequence generator provides a sequential series of numbers to applications. The sequence generator is especially useful in multi-user environments for generating unique sequential numbers such as an employee id without the overhead of disk I/O or transaction locking.

In the previous release of Oracle Database, when a PL/SQL program needed to get a value from a sequence, it used SQL. This is a usability irritation for PL/SQL programmers. In Oracle Database 11g, it is now possible to simply use the pseudocolumns CURRVAL and NEXTVAL in a PL/SQL expression.

### **PL/SQL: PL/Scope**

PL/Scope allows you to browse PL/SQL source code analogously to the way that Cscope (see <http://cscope.sourceforge.net>) allows you to browse C source code. You can search for and display all types of definitions, declarations, assignments and references in the PL/SQL source code. The PL/SQL compiler can optionally derive the metadata needed to support PL/Scope and store it in the database catalog. The metadata takes into account the nuances of the language, including scoping and overloading. You can generate reports (especially hyperlinked HTML

reports) with supplied report generators. PL/Scope supports increased developer productivity, especially for those who need to maintain someone else's code.

#### **PL/SQL: Hierarchical Profiler**

The PL/SQL hierarchical profiler reports the dynamic execution profile of your PL/SQL program, organized by subprogram calls. It accounts for SQL and PL/SQL execution times separately. Each subprogram-level summary in the dynamic execution profile includes information such as number of calls to the subprogram, time spent in the subprogram itself, time spent in the subprogram's subtree (that is, in its descendent subprograms), and detailed parent-children information. You can browse the generated HTML reports in any browser. The browser's navigational capabilities, combined with well chosen links, provide a powerful way to analyze performance of large applications, improve application performance, and lower development costs.

#### **Java**

Oracle Database includes an embedded Java VM, which lets you run Java directly in the database. This feature is a response to customer requirements for portability and reuse. Java can be migrated to/from middle-tier (J2EE, POJOS, JDBC) to the database (Java stored procedures) and vice versa. Other important benefits Java brings include the ability to reuse the large collection of Java class libraries that are available and to leverage the Java skills of application developers.

JDBC is an industry-standard application programming interface (API) that lets you embed SQL statements in Java code. Each database vendor, such as Oracle, creates its JDBC implementation by implementing the interfaces of the standard `java.sql` package. For more information about Oracle Java, JDBC & Database Web Services, go to: [http://www.oracle.com/technology/tech/java/java\\_db/index.html](http://www.oracle.com/technology/tech/java/java_db/index.html)

#### **Java: OracleJVM Performance, JIT Compiler**

The OracleJVM furnishes a just-in-time compiler (JIT). The JIT dynamically and transparently produces native binaries from any Java classes existing in the database. The binary code are stored, avoiding recompilation. The JIT is enabled by default (out-of-the-box) and does not need a C compiler or further configuration. Internal testing using an industry standard benchmark shows an order of magnitude (10x) speed-up.

#### **Java: Productivity and Portability through Java Standards**

Productivity and application portability across platforms and vendors are the primary motivations for using Java. In this release, JDBC brings support for Java SE 6, JDBC 4.0 and JMX. The Java runtime embedded in the RDBMS brings support for Java SE 5, RowSet (JSR-114), and JMX.

**Large out-of-the-box performance boost  
for Java in the database.**

**Java: JDBC 4.0**

The Oracle Database 11g JDBC comes in two flavors: ojdbc5.jar for Java 5 (i.e., JDK 1.5) and ojdbc6.jar for Java 6 (i.e., JDK 1.6). The ojdbc6.jar supports the new JDBC 4.0 specification including: Connection and Statement Enhancements, Wrapper Interface, New Standard Datatypes, SQL 2003 National Character Set types, LOB Enhancements, Exception Hierarchy, and RowSet (JSR-114) Enhancements.

**Java: OracleJVM Compliance with Java 5**

OracleJVM allows reusing Java SE applications and libraries directly in the database resulting in significant productivity and performance gains. This release supports Java SE 5 (i.e., compatibility with JDK 1.5), out of the box.

**Java: RowSet (JSR-114) Comes to Server-side JDBC**

The server-side JDBC driver now supports the RowSet specification. Java/JDBC code that make use of JDBCRowSet, CachedRowSet, WebRowSet, JoinRowSet and FilteredRowSet can now run directly in the database.

**Java: Exposing Oracle to Java**

In this release the following features are exposed: SYS.ANYDATA, Sys:ANYTYPE, Secure Files, and Query change notification .

**Java: Prefetch in 1st Roundtrip**

For ad-hoc SQL queries, the new pre-fetch mechanism combines parse, execute, and fetch to retrieves the first batch of the results set in the first interaction with the database. Reliable internal tests show a 50% reduction in network roundtrip; 50% reduction in response time; 30% reduction of server and client CPU consumption. To put this reduction in perspective, a typical Web retail application issuing 1 million ad-hoc queries per day (or a determined period of time) will see a reduction of 1 million roundtrips during the same period.

**Java: Native AQ Protocol**

For JMS/AQ calls via JDBC, a new native AQ protocol (as opposed to a PL/SQL based interface) garners 40% up to 300% performance and a reduction of server CPU consumption.

**Java: Database/Query Change Notification**

JDBC-Thin and JDBC-OCI support for Database and Query Notification allows a JDBC thread subscribe to notification of changes in a query result set.

Middle-tier may use this feature to invalidate and refresh data caches. See more details on the APIs in the Oracle Database 11g JDBC doc.

### **Java: Server and Client Result Cache**

SQL result sets can be cached in the SGA on the server-side and automatically invalidated whenever related objects are changed. The JDBC-OCI driver furnishes a client-side counterpart of server result cache. It is synchronized with changes to the server cache. See the Cross Language Features section for more details.

### **Java: Security**

JDBC-Thin now supports the Oracle Advanced Security option – on par with JDBC-OCI, which already supports it -- including strong authentication (i.e., Kerberos, Radius, SSL) and support for new encryption and data integrity algorithms.

### **Java: Oracle JVM Ease of Use**

Limited ease of use is one of the obstacles to mass adoption of Java in the database. This release brings many features to simplify and improve the user experience including: like-JDK interface, output redirect, property interface, database resident JAR, two-tier duration of Java session state.

### **Java: Manageability**

Manageability is a key requirement in production environments for Java applications running against (i.e., JDBC, Java EE components) or within the Oracle Database (i.e., Java in the database). This release brings the following manageability features: MBean for JDBC Logging, programmatic startup and shutdown, JMX in OracleJVM, and OracleJVM Utilities Enhancements.

### **Oracle Call Interface (OCI)**

Oracle Call Interface provides high performance, powerful access and fine-grained control for application design. OCI is also the foundation on which .NET, PHP, and the JDBC type 2 driver is built on. See the Cross Language Features section for OCI features exposed through multiple languages. Currently, the following features are only available through OCI.

### **Database Resident Connection Pooling**

Database Resident Connection Pooling (DRCP) provides a connection pool in the database server for typical Web application usage scenarios where the application acquires a database connection, works on it for a relatively short duration, and then releases it. DRCP pools “dedicated” servers, which is the equivalent of a server foreground process and a database session combined and henceforth are referred to as the “pooled” servers. DRCP complements middle-tier connection pools that share connections among threads in a middle-tier process. In addition, DRCP enables sharing of database connection across middle-tier processes on the same middle-tier host and even across middle-tier hosts. This results in a significant reduction in key database resources needed to support a large number of client connections, thereby reducing the database tier memory footprint and boosting the

scalability of both middle-tiers and database tiers. Having a pool of readily available servers also has the additional benefit of reducing the cost of creating and tearing down client connections.

DRCP is especially beneficial for architectures with multi-process single threaded application servers that cannot do middle-tier connection pooling. The database can still scale to tens of thousands of simultaneous connections with DRCP.

Internal testing shows that 20 times more user connections can be supported by the database when using the DRCP feature.

#### **Implicit Fetching of ROWIDs**

A ROWID is a globally unique identifier for a row in a database. A ROWID is the fastest way to access a single row. Implicit fetching of ROWIDs in SELECT ... FOR UPDATE statements means that the ROWID is retrieved at the client side even if it is not one of the columns named in the SELECT statement. The position parameter of `OCIDefineByPos()` is set to zero (0) to access the ROWID.

#### **Binding and Defining Multiple Buffers**

You can specify non-contiguous buffers for use with a single bind or define call. Performance is improved because data stored/fetched at non-contiguous addresses does not need to be copied from/to a contiguous buffer. Also, this feature is useful for applications that cannot allocate large contiguous buffers for array bindings due to memory management concerns. With this feature, the application can split the array across a set of smaller non-contiguous buffers.

#### **Allocating an Array of Descriptors**

Array binds and defines allow an application to insert or fetch multiple rows in one round trip to the database. To facilitate binding/defining array of descriptors (such as those for LOBs, timezone etc.), OCI has added the `OCIArrayDescriptorAlloc()` call where you can allocate an array of descriptors in one OCI call.

### **CONCLUSION**

Application development organizations have a wide variety of programming languages and technologies to choose from. Oracle Database 11g delivers a highly productive and powerful set of application development tools supporting the most popular development technologies including Java, PHP and .NET. Compelling Oracle Database features (e.g., Real Application Clusters, XML DB, etc.) are available to developers through Oracle's programming language interfaces.

Oracle Application Express is a unique Web application development tool that is ideal for quickly building departmental and SMB applications. Oracle SQL Developer provides a graphical work environment for the Oracle Database that increases database developer productivity. Oracle SQL Developer Migration Workbench eases migrations from other databases to Oracle. PL/SQL and Java in the database provide architectural options to create database server-side code that

can increase the performance, security, maintainability and scalability of your applications.

Collectively these tools simplify your development tasks and enable your organization to reduce application development time to market.



Oracle Database 11g Application Development  
June 2007

Author: David Gambino

Contributing Authors: Sue Harper, Alison Holloway, Alex Keh, Srinath Krishnaswamy, Bryn Llewellyn, Kuassi Mensah, David Peake, Christian Shay

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](http://oracle.com)

Copyright © 2007, 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 is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.