

# Oracle Database 11g en Windows: Desarrollo e Implementación

*Informe Técnico de Oracle  
Septiembre de 2007*

# Oracle Database 11g en Windows: Desarrollo e Implementación

Informe Ejecutivo.....	3
Oracle en Windows.....	3
Desarrollo de Aplicaciones .NET.....	4
Oracle Developer Tools para Visual Studio .NET.....	5
Oracle Explorer.....	5
Diseñadores y Wizards.....	5
Generación Automática de Códigos .NET.....	6
Editor PL/SQL.....	6
Depurador PL/SQL Totalmente Integrado.....	6
Prueba de Procedimientos Almacenados.....	6
Oracle Data Window.....	7
SQL Query Window.....	7
Ayuda Online Integrada.....	7
Oracle Data Provider para .NET.....	7
Desempeño.....	8
Desempeño – Nuevo en ODP.NET 11g.....	10
Notificación de Cambios en la Base de Datos.....	11
Real Application Clusters (RAC).....	12
Características XML.....	13
ADO.NET 2.0.....	14
Tipos Nativos de Oracle.....	15
Otras Características Importantes.....	15
Oracle Database Extensions para .NET.....	16
Implementación de Aplicaciones.....	16
Directorio.....	16
Autenticación Nativa y Active Directory.....	17
Oracle Net Naming con Active Directory.....	17
Características de Active Directory – Nuevo en Oracle Database 11g.....	18
Seguridad.....	18
Oracle Wallets en el Registro Windows.....	19
Integración con Microsoft Certificate Store.....	19
Características de Seguridad de Windows – Nuevo en Oracle Database 11g.....	19
Backup y recuperación con el Servicio de Copias de Respaldo – Nuevo en Oracle Database 11g.....	19
Alta Disponibilidad con Oracle Fail Safe.....	20
Capacidad de Administración Standby Lógica/Física y de múltiples clusters.....	21
Conclusión.....	21

# Oracle Database 11g en Windows: Desarrollo e Implementación

## INFORME EJECUTIVO

Oracle Database 11g brinda una solución de base de datos integral para la plataforma Windows a fin de lograr que el desarrollo y la implementación de Oracle en Windows y .NET sean más fáciles, rápidos y económicos.

Oracle Developer Tools para Visual Studio .NET es un poderoso “complemento” para Microsoft Visual Studio 2005 y Visual Studio .NET 2003 que permite a los desarrolladores .NET escribir las aplicaciones de Oracle de manera más fácil y rápida. Está estrechamente integrado con Visual Studio e incluye características avanzadas, como por ejemplo diseñadores, la generación automática de códigos .NET y un depurador y editor PL/SQL totalmente integrados.

Los desarrolladores .NET pueden acceder a la base de datos Oracle por medio de Oracle Data Provider para .NET (ODP.NET). ODP.NET es un proveedor nativo de datos .NET que ofrece el mejor desempeño y la mayor parte de la funcionalidad de base de datos Oracle disponible desde un entorno .NET. En las versiones más recientes, ODP.NET respalda un Entorno .NET de 64 bits (Windows x64 e Itanium), desempeño más rápido para las consultas y la recuperación de LOB, y transacciones locales para System.Transactions.

Como parte de Oracle Database 11g, Oracle Database Extensions para .NET es una opción de base de datos para implementar procedimientos almacenados y funciones escritas en un lenguaje administrado por .NET en la base de datos Oracle sobre Windows. Un procedimiento almacenado en .NET puede invocarse desde un código .NET; desde una sentencia SQL o un *trigger*; desde otro procedimiento almacenado en .NET, PL/SQL o Java; o desde cualquier otro lugar donde se permita invocar una función o procedimiento almacenado.

Oracle Database 11g en Windows incluye nuevas características para Active Directory y el Servicio de Copias de Respaldo (*Volume Shadow Copy*). Oracle se integra con la seguridad nativa de Windows y Active Directory, permitiéndole a Oracle operar en entornos de directorio y sistemas operativos heterogéneos. Las herramientas de configuración y de la base de datos Oracle pueden utilizar las credenciales de ingreso de un usuario Windows para conectarse a Active Directory sin tener que reingresar las credenciales de autenticación. Asimismo, configurar la autenticación de Kerberos para los usuarios Oracle en Active Directory ahora es mucho más fácil y flexible.

El nuevo escritor del Servicio de Copias de Respaldo (VSS) permite a las bases de datos Oracle participar en la recuperación y el backup iniciado por VSS en Windows Server 2003 y versiones posteriores. Los administradores pueden, con total seguridad, realizar el backup y la recuperación de los datos Oracle, junto con proveedores y solicitantes conocidos de VSS.

Oracle Fail Safe ofrece soluciones de alta disponibilidad para el software Oracle en uno o más clusters Windows, con una consola de administración fácil de usar.

## ORACLE EN WINDOWS

Al ser el primero en haber adquirido una base de datos relacional sobre Windows NT en 1993, Oracle se comprometió a ofrecer las mejores soluciones de administración de la información disponibles para la plataforma. Desde ese entonces, la tecnología de Oracle se ha adaptado a los cambios más recientes en los conocimientos informáticos de Windows desde las primeras soluciones cliente/servidor hasta las aplicaciones de Internet, y ahora la tecnología grid. Actualmente, Oracle continúa siendo líder en la plataforma Windows a través de su base de datos de primera clase y su estrecha integración con las tecnologías subyacentes del sistema operativo y el Entorno .NET. Oracle tiene más de veinticinco años de experiencia en administración de datos y es el proveedor líder, con la tecnología, la experiencia y los servicios necesarios para implementar aplicaciones de negocio críticas sobre Windows—ya sea en una Intranet corporativa o en Internet.

Oracle ofrece varios productos para la programación .NET. Oracle Developer Tools para Visual Studio .NET es un plug-in para utilizar en el momento de diseño. Oracle Data Provider para .NET permite el acceso a datos en todas las características de base de datos clave de Oracle. Oracle Database Extensions brinda soporte para los procedimientos .NET almacenados.

Oracle Database 11g para Windows brinda todas las características necesarias para la administración de datos, ya sea que se utilice para la implementación en toda la empresa o en un solo departamento. Permite a los usuarios aprovechar las ventajas de facilidad de uso y costo que ofrece Windows, mientras brinda la escalabilidad, confiabilidad y el desempeño tradicionalmente disponibles con Oracle. Esta nueva base de datos Oracle contiene mejoras adicionales para la integración de las aplicaciones en la plataforma Windows.

Dos de estas áreas de atención son el desarrollo y la implementación de aplicaciones sobre Windows. Oracle Database 11g ha hecho grandes avances por mejorar el desempeño, la facilidad de uso y las capacidades de administración, tanto para los desarrolladores como para los administradores de Windows.

Uno de los mayores beneficios de utilizar los productos Oracle es el soporte de múltiples estándares de programación. Al soportar aplicaciones Java, .NET, PHP y C/C++, Oracle garantiza que todos los desarrolladores puedan utilizar las características avanzadas de la base de datos Oracle, ofreciendo así una verdadera flexibilidad para las empresas de desarrollo. Cada uno de los *drivers* para el acceso a datos Oracle está diseñado para maximizar el desempeño y tener acceso a las características más recientes de base de datos. Este informe ejecutivo se focaliza especialmente en el desarrollo .NET con la base de datos Oracle y la integración del servidor de la base de datos Oracle con Windows.

## DESARROLLO DE APLICACIONES .NET

Los desarrolladores de .NET de Oracle por lo general deben realizar varias tareas de base de datos, como modificar la tabla de base de datos o ver los diseños, actualizar los datos y editar y depurar procedimientos almacenados PL/SQL. Oracle Developer Tools para Visual Studio .NET (ODT) facilita estas tareas. ODT constituye un “complemento” estrechamente integrado para Microsoft Visual Studio 2005 y Visual Studio .NET 2003, que incluye características avanzadas, como diseñadores y *wizards*, la generación automática de códigos .NET, y un depurador y editor PL/SQL totalmente integrado.

Oracle ofrece soporte para el acceso a los datos ADO.NET por medio de ODP.NET. ODP.NET es un proveedor nativo de acceso a datos .NET para bases de datos Oracle. Puede ser utilizado desde cualquier lenguaje .NET, con inclusión de C# .NET, Visual Basic .NET y ASP.NET, y desde los servidores de aplicaciones .NET o los procedimientos almacenados .NET. ODP.NET ofrece el mejor desempeño, y acceso a más características de la base de datos Oracle que cualquier otro proveedor de datos. ODP.NET fue diseñado específicamente para permitir a los desarrolladores .NET maximizar las capacidades de la base de datos Oracle.

ODP.NET 11g incorpora características de desempeño que utilizan tanto la nueva funcionalidad disponible en Oracle Database 11g como la funcionalidad existente en la base de datos Oracle. Como tal, los desarrolladores se beneficiarán utilizando la última versión de ODP.NET, ya sea que quieran implementar las nuevas aplicaciones de base de datos o mejorar las aplicaciones tradicionales.

Oracle Database Extensions para .NET es una opción de base de datos que le permite implementar procedimientos almacenados y funciones escritas en un lenguaje administrado en .NET, como C# o VB.NET, en el servidor de base de datos sobre Windows. Un procedimiento almacenado en .NET puede invocarse desde un código .NET; desde una sentencia SQL o un *trigger*; desde otro procedimiento almacenado en .NET, PL/SQL o Java; o desde cualquier otro lugar donde se permita invocar una función o procedimiento almacenado.

### **Oracle Developer Tools para Visual Studio .NET**

Oracle Developer Tools para Visual Studio .NET es un poderoso “complemento” para Visual Studio .NET que permite a los desarrolladores .NET escribir aplicaciones Oracle de manera más fácil y rápida. Ahora disponible como descarga gratuita en Oracle Technology Network, se encuentra fuertemente integrado con Visual Studio 2005 y Visual Studio .NET 2003 e incluye características avanzadas como diseñadores, la generación automática de códigos .NET y un depurador/editor PL/SQL.

**Oracle Developer Tools para Visual Studio .NET permite a los desarrolladores .NET escribir aplicaciones Oracle de manera fácil y rápida.**

Todas estas características fueron diseñadas con el objetivo de hacer que el desarrollo de Oracle en Windows sea lo más intuitivo y fácil posible. Los desarrolladores habituados al Servidor SQL lo encontrarán familiar e intuitivo, lo cual minimiza cualquier curva de aprendizaje. Y para aquellos que son nuevos en el desarrollo Oracle, esta es una manera fácil de aprender.

Las características más importantes incluyen:

- Oracle Explorer – Navegue y altere el esquema Oracle por medio de un control de árbol
- Diseñadores y *Wizards*– por ej., Diseñador de Tabla – facilitan las tareas de base de datos
- Generación Automática de Códigos – Función *drag and drop* para crear el código de trabajo.
- Editor y depurador PL/SQL – Edite y depure los *triggers*, paquetes, funciones y procedimientos PL/SQL almacenados
- Prueba de Procedimientos Almacenados – Ejecute funciones y procedimientos almacenados
- Oracle Data Window – Visualice y edite sus datos Oracle
- SQL Query Window – Ejecute cualquier script o sentencia SQL ad-hoc
- Sistema de Ayuda Integrado – Manuales de Referencia de Errores y SQL, PL/SQL
- Wizard de Implementación .NET – Implementa fácilmente los Procedimientos .NET almacenados

### **Oracle Explorer**

Oracle Explorer es un control de árbol que permite visualizar la estructura del esquema Oracle. Aquí se incluyen todos los tipos del esquema Oracle. Los metadatos como los tipos de datos de columnas o el tipo de parámetros de los procedimientos almacenados están disponibles en el “panel de propiedades” de Visual Studio .NET al seleccionar un objeto de esquema. Un menú de contexto para cada objeto de esquema ofrece características adicionales: Se pueden generar scripts SQL para los objetos de esquemas, proporcionar filtros para limitar lo que se ve, y se

pueden activar diseñadores y *wizards* para alterar el esquema. Por ejemplo, mientras se visualiza la estructura de una tabla, se puede utilizar el menú de contexto para activar “Oracle TableDesigner” y así modificar el diseño de una tabla.

#### **Diseñadores y Wizards**

Oracle Developer Tools ofrece una variedad de diseñadores y *wizards* con el objeto de brindar asistencia paso a paso para crear o alterar los objetos de base de datos como las tablas, visualizaciones, los procedimientos almacenados, las funciones almacenadas, los paquetes PL/SQL, las secuencias, los índices, las restricciones, los *triggers*, sinónimos y mucho más. Por ejemplo, el diseñador de tabla hace que el hecho de consultar tipos de datos y memorizar la sintaxis SQL sea algo del pasado. En el caso del diseñador de tabla, el usuario simplemente brinda el nombre de la tabla y el nombre de las columnas y selecciona el tipo de datos adecuado desde un casillero desplegable. Luego, al presionar el botón “guardar”, la tabla debería crearse o alterarse. Un objeto de esquema recientemente creado aparecerá inmediatamente en el explorador Oracle.

Cualquier SQL generado por el diseñador para crear o modificar un objeto de esquema se despliega tanto en la ventana “Vista Preliminar” (antes de producirse el cambio) como en la ventana de Oracle Database Output (después de producirse el cambio).

Utilizar la función *drag and drop* para mover el objeto de un esquema desde Oracle Explorer hasta una planilla de aplicaciones da como resultado un código .NET automáticamente generado para las operaciones SELECCIONAR, ACTUALIZAR, INSERTAR y ELIMINAR en ese objeto de esquema.

#### **Generación Automática de Códigos .NET**

Utilizar la función *drag and drop* para mover el objeto de un esquema desde Oracle Explorer hasta una planilla de aplicaciones da como resultado un código .NET automáticamente generado para las operaciones SELECCIONAR, ACTUALIZAR, INSERTAR y ELIMINAR en ese objeto de esquema. El código utiliza la clase OracleDataAdapter provista por Oracle Data Provider para .NET, el sólido proveedor de datos .NET de Oracle. DataAdapter puede utilizarse para conectar elementos UI (como un DataGrid) en la planilla de aplicación para la base de datos Oracle, con una cantidad mínima de códigos adicionales. Este tipo de uso de DataAdapter es estándar y resultará muy familiar para aquellos que tengan experiencia con otras bases de datos.

El usuario también tiene la opción de generar un código para un grupo de datos tipeados y no tipeados sobre la base de una visualización o tabla de Oracle. Estos grupos de datos pueden luego vincularse como fuente de datos a los elementos UI o pueden utilizarse de otras maneras estándar en Visual Studio .NET, como sucede con el diseñador Visual Studio XML Schema.

#### **Editor PL/SQL**

El editor PL/SQL ofrece un entorno de desarrollo fuertemente integrado para los desarrolladores Oracle y los libera de tener que dejar Visual Studio .NET cuando se requiere el desarrollo de un procedimiento almacenado. Las características de Standard Visual Studio incluyen la coloración de la sintaxis para mejorar la capacidad de lectura y regiones desplegables para ocultar procedimientos o funciones que son parte de un paquete muy grande o complejo –esto facilita la focalización en el procedimiento o la función sobre la que se está trabajando. Asimismo, a medida que se ingresan sentencias SQL o PL/SQL, aparece un listado desplegable de tablas o columnas para autocompletar la sentencia.

Cuando se compila PL/SQL y se encuentran los errores, éstos se enumeran en la lista de tareas de Visual Studio .NET. Hacer un click sobre ellos llevará al desarrollador a la línea de código que falló. Al presionar la tecla de ayuda sensible al contexto, se abrirá el manual de mensajes de errores para el código de error en cuestión.

### Depurador PL/SQL Totalmente Integrado

Una característica incorporada en ODT 10.2.0.2 es un depurador PL/SQL totalmente integrado con Visual Studio. Usted ahora puede depurar el código PL/SQL dentro de los *triggers*, las funciones y los procedimientos almacenados exactamente de la misma manera que depura VB.NET o el código C#. Puede ingresar en el código PL/SQL, visualizar y modificar los valores variables, mirar las llamadas, establecer puntos de ruptura y mucho más.

También se puede ingresar desde el código de aplicación C# o VB.NET directamente hasta el código PL/SQL y volver a salir. Por ejemplo, si su código C# invoca una función o procedimiento PL/SQL utilizando Oracle Data Provider para .NET, usted puede depurar el código C#, y luego ingresar en el código PL/SQL para continuar la depuración. Puede examinar todos los valores de parámetro transmitidos en la función o procedimiento almacenado, con inclusión de los tipos de datos complejos y las matrices dentro del depurador PL/SQL. Cuando se completa la ejecución del código PL/SQL, puede regresar al código C# para continuar con la depuración.

También puede establecer puntos de ruptura y depurar *triggers*, funciones y procedimientos PL/SQL almacenados invocados por cualquier aplicación que se encuentre en cualquier plataforma sin tener que modificar ni reconstruir esa aplicación. Por ejemplo, una aplicación PHP que se ejecuta en Linux invoca un FOO de procedimiento PL/SQL almacenado y transmite una matriz de valores. Después de que establece una variable en el entorno de la aplicación PHP, usted puede establecer un punto de ruptura en FOO dentro de Visual Studio y comenzar la depuración cuando se ejecuta FOO.

Ahora, con el depurador PL/SQL integrado de ODT, usted puede ingresar en el código PL/SQL, visualizar y modificar los valores variables, observar las llamadas, establecer puntos de ruptura y mucho más. Puede entrar desde su aplicación C# o VB.NET directamente al código PL/SQL y volver a salir.

### Prueba de Procedimientos Almacenados

La función y los procedimientos almacenados pueden probarse rápidamente por medio del menú de contexto “Ejecutar” en Oracle Explorer. Se requieren los parámetros de entrada y luego se ejecuta el procedimiento. Los parámetros de salida se brindan en un formato fácil de leer. Los tipos de salida complejos, como REF CURSOR, producen un enlace que, al hacer un click, abrirá un grid con los datos complejos.

### Oracle Data Window

Oracle Data Window despliega tablas o visualiza datos en un grid fácil de leer y permite a los desarrolladores insertar, actualizar o eliminar los datos sin tener que dejar el entorno de desarrollo Visual Studio .NET. Para las tablas grandes, el usuario puede proporcionar el número de fila para saltar a una fila particular.

### SQL Query Window

Las sentencias SQL ad hoc, como aquellas que contienen las funciones SELECCIONAR, CREAR, ALTERAR, etc., pueden ejecutarse desde SQL Query Window. Si el *output* de la sentencia SQL es una tabla o una visualización, al usuario se le da la opción de *output* grid o de texto. Se pueden resaltar múltiples sentencias y ejecutarlas como un grupo. El operador de script “@” puede utilizarse para ejecutar scripts SQL.

También se puede aplicar la función drag and drop para mover los objetos de esquema desde Oracle Explorer hasta la superficie de SQL Query Window para generar automáticamente el SQL adecuado.

### Ayuda Online Integrada

Oracle Developer Tools para Visual Studio .NET incluye documentación clave de Oracle que ha sido convertida al formato de ayuda de Visual Studio. Asimismo, se ha incluido la Guía de Referencia de Lenguaje SQL, la Guía de Referencia y Usuarios PL/SQL, y los Manuales de Códigos y Mensajes de Error.

Los capítulos de estos manuales pueden leerse secuencialmente a través del panel de ayuda de Visual Studio .NET. Además, esta documentación ha sido integrada con la ayuda sensible al contexto de Visual Studio. Por ejemplo, mientras se escribe un procedimiento almacenado, el desarrollador puede resaltar una palabra clave SQL o PL/SQL, como “SELECCIONAR”, presionar la tecla de ayuda de contexto, y ser remitido automáticamente a la página adecuada en la Guía de Referencia de Lenguaje SQL. Además de estos manuales, la documentación de Oracle Developer Tools contiene capítulos introductorios y “revisiones detalladas” útiles, así como una guía de referencia.

## Oracle Data Provider para .NET

ODP.NET es un *driver* de acceso nativo a datos.NET para la base de datos Oracle. Debido a que fue desarrollado teniendo en cuenta las fuentes de datos específicas de Oracle, ODP.NET ha sido optimizado para alcanzar un rápido desempeño y acceder a la funcionalidad más reciente de la base de datos. Como tal, ODP.NET hace que utilizar las bases de datos Oracle sea más flexible, más rápido y más estable que muchas otras soluciones de acceso a datos. Comenzando con la versión 10.2.0.2, ODP.NET ha sido compatible con ADO.NET 1.x y 2.0. ODP.NET soporta todas las versiones actualmente disponibles del Entorno .NET, con inclusión del Entorno .NET 3.0.

Los demás proveedores de datos .NET para Oracle están diseñados para obtener portabilidad máxima en el acceso a distintas fuentes de datos. Lo que se pierde al utilizar estos proveedores es la capacidad de aprovechar las características inherentes de desempeño y administración de datos. Por ejemplo, una de las características exclusivas de ODP.NET es soportar los tipos de datos nativos de Oracle, como los Cursores REF y LOBs. Estos tipos de datos brindan mayor flexibilidad para la recuperación de datos que los tipos de datos .NET de sus contrapartes.

Oracle Data Provider para .NET brinda un rápido desempeño para el acceso a datos en las bases de datos Oracle. Soporta la funcionalidad de base de datos más reciente, con inclusión de la notificación de cambios en la base de datos, XML DB, los tipos de datos nativos de Oracle y Real Application Clusters.

Los desarrolladores pueden seleccionar los tipos de datos que más se ajusten a los requisitos de desempeño y uso de datos. Por ejemplo, los Cursores REF diferirán la recuperación del grupo de resultados hasta que los datos no hayan sido realmente leídos, una optimización de desempeño clave frecuentemente utilizada. Con los tipos de datos de las LOB de Oracle, una aplicación puede elegir recuperar todos los datos LOB en un grupo de resultados en un solo recorrido de la base de datos o diferir la recuperación LOB hasta algún momento posterior cuando los datos LOB sean realmente consumidos por el cliente. Luego, cuando la LOB se lea, el desarrollador podrá decidir qué cantidad de datos recuperar. Tal vez la aplicación solo pueda necesitar los últimos 100KB de un gran CLOB, en vez de todo el CLOB. Este es un ejemplo de cómo ODP.NET brinda un mayor control para el ajuste de desempeño y para aprovechar las características nativas de base de datos.

Con la llegada de ADO.NET 2.0, incluso la portabilidad de código ya no es un problema. Ahora, con las *factory classes* del proveedor de datos ADO.NET, los desarrolladores pueden crear un solo grupo de códigos que a su vez puede acceder a las bases de datos de múltiples proveedores de manera mucho más fácil. ODP.NET 11g brinda una enorme cantidad de características para la base de datos Oracle, con inclusión del ajuste de desempeño para el acceso a datos; las notificaciones de cambios en la base de datos; el grupo de conexiones para RAC y Data Guard; el soporte XML; el soporte de los tipos de datos nativos de Oracle; el soporte ADO.NET 2.0; y muchas otras características.

### Desempeño

Uno de los diferenciadores clave de ODP.NET respecto de otros proveedores es su fácil desempeño y numerosas opciones de ajuste. De modo no visible, se han realizado muchas optimizaciones para garantizar el rápido acceso .NET a las fuentes de datos Oracle. Asimismo, ODP.NET tiene muchos parámetros ajustables para la recuperación de datos específicos y escenarios de actualización de datos. Muchas de



estas optimizaciones fueron desarrolladas para recuperar y manipular los tipos nativos de Oracle, como los Cursores REF y las LOB.

### **Grupo de Conexiones y Caché de Sentencias**

Una de las optimizaciones de desempeño más ampliamente utilizadas se relaciona con el grupo de conexiones, que es esencial para las aplicaciones con una gran cantidad de usuarios que se conectan y desconectan de la base de datos. ODP.NET crea un grupo de conexiones con parámetros ajustables que incluyen el ciclo de vida y la finalización de la conexión, los tamaños máximos y mínimos del grupo, y la cantidad de conexiones para aumentar o disminuir el grupo simultáneamente. Estos parámetros les dan a los desarrolladores un mayor control sobre cómo su aplicación debe manejar una gran población de usuarios y los cambios que se produzcan en esas poblaciones con el tiempo. Esto finalmente conduce a un mejor tiempo de respuesta de la aplicación y una mejor calidad de servicio para los usuarios finales.

Si una sentencia PL/SQL o consulta adicional se ejecuta múltiples veces, ODP.NET puede utilizar un caché de sentencias para acelerar la ejecución de la sentencia. Al poner en caché el cursor de servidor creado durante la ejecución de la sentencia inicial, el caché de sentencias elimina la necesidad de volver a analizar cada sentencia antes de las posteriores ejecuciones. Cada futura ejecución de sentencias vuelve a utilizar la información analizada guardada, y luego ejecuta la sentencia. Los datos del grupo de resultados no se colocan en caché, solo la información de las sentencias analizadas. ODP.NET aún recuperará los datos más recientes del servidor de base de datos. El caché de sentencias permite que estas consultas se ejecuten de manera más rápida.

**ODP.NET incluye numerosas características para el ajuste de desempeño a fin de optimizar los cambios y la recuperación de la información. Algunas de estas opciones de ajuste incluyen el hecho de utilizar grupos de conexión; caché de sentencias; tipos de datos LOB; e implementar matrices PL/SQL asociativas. ODP.NET de 64 bits es respaldado por Windows x64 y Windows Itanium.**

Cuando se implementa un caché de sentencias, las sentencias SQL o PL/SQL deberían utilizar parámetros en lugar de valores. Al hacer esto, se aprovechan las ventajas del caché de sentencias, ya que la información analizada de las sentencias parametrizadas puede volver a utilizarse incluso si el valor de los parámetros cambia en posteriores ejecuciones. Si en cambio se utilizaron valores literales y esos valores literales cambiaron, la información analizada no podría volver a utilizarse, y la base de datos necesitaría analizar la sentencia nuevamente.

Por defecto, ODP.NET colocará en caché las últimas diez sentencias ejecutadas. La cantidad de sentencias para colocar en caché y qué sentencias se colocarán en caché se puede configurar en el nivel de aplicación .NET o en el nivel de máquina.

### **Controlar el Volumen de la Búsqueda de Datos**

Para ajustar el desempeño de la recuperación de datos, ODP.NET puede especificar la cantidad fija de datos que deberán devolverse en cada recorrido completo de la base de datos. Muchas veces, un desarrollador puede no necesitar recuperar todos los datos consultados de una sola vez ya que el usuario final consume los datos durante un período.

Las búsquedas de datos de consulta pueden organizarse en distintas partes definidas por el desarrollador a través de dos propiedades de ODP.NET OracleCommand: FetchSize y RowSize. FetchSize informa a ODP.NET qué cantidad de datos recuperar por recorrido completo de la base de datos. RowSize indica la extensión de cada fila de datos. RowSize es una propiedad de solo lectura que se establece después de que se ejecuta una consulta. Si un desarrollador desea buscar diez filas de datos por recorrido de la base de datos, todo lo que se necesita es establecer un FetchSize igual a diez multiplicado por el RowSize. Lo maravilloso acerca de RowSize es que su valor es determinado en el tiempo de ejecución. Por lo tanto, si se produce un cambio de esquema o un cambio de consulta en el futuro, no hay necesidad de modificar el código para garantizar la búsqueda de diez filas de datos por cada recorrido.

### **Optimizar los Datos LOB**

Una característica similar para el ajuste del volumen de las búsquedas se aplica a los tipos de datos LOB. Estos tipos de datos se utilizan para almacenar imágenes y documentos, que algunas veces pueden medirse en términos de gigabytes. Para las aplicaciones LOB, el desempeño es a menudo un tema clave debido al posible tamaño de los datos LOB y debido a cómo se consumen los datos LOB. Enviar gigabytes de datos entre el servidor y el cliente puede atascar una red, a menos que la recuperación de datos se administre de manera inteligente.

Con ODP.NET, los desarrolladores pueden especificar cómo deben recuperarse los datos LOB. Al momento de ejecutar una consulta LOB, los desarrolladores pueden elegir buscar todos los datos LOB inmediatamente en una cantidad mínima de recorridos de la base de datos, o retrasar la búsqueda LOB hasta que el usuario intente leer los datos. Si los desarrolladores eligen retrasar la búsqueda LOB, entonces pueden especificar qué cantidad de datos recuperar para cada llamada de lectura LOB. Si el usuario final solo necesita leer 10KB de datos simultáneamente, los desarrolladores pueden recuperar solo 10KB de datos para cada lectura LOB.

Asimismo, los desarrolladores ODP.NET pueden recuperar cualquier parte de LOB por medio de un acceso aleatorio. Tal vez, ellos solo necesiten los últimos 100MB de datos de una LOB de 1GB. Los desarrolladores pueden ajustar la recuperación LOB para buscar solamente los últimos 100MB sin mostrar al cliente los datos de los primeros 900MB. Estas opciones de ajuste brindan a los desarrolladores .NET la flexibilidad para crear aplicaciones con un mejor desempeño.

Debido a que los datos LOB a menudo pueden ser extensos, por defecto, la búsqueda de datos LOB se demora después de ejecutarse una consulta. Cuando se recuperan muchos grupos de LOBs, este comportamiento es óptimo para evitar la sobrecarga de la red con los datos LOB entregados al cliente. No obstante, para las LOB pequeñas, este comportamiento puede ser lento y generar más recorridos de la base de datos que lo necesario.

Para permitir la búsqueda inmediata de una pequeña cantidad de datos LOB, ODP.NET presenta la propiedad `InitialLOBFetchSize` en las clases `OracleCommand` y `OracleDataReader`. Si `InitialLOBFetchSize` se establece con un valor mayor a cero, los datos LOB iniciales de todas las LOB consultadas se buscan en un solo recorrido para la cantidad de caracteres o bytes especificada en esta propiedad. Por ejemplo, si `InitialLOBFetchSize` se estableciera en 10 KB, los primeros 10 KB de todas las LOB seleccionadas serían recuperados para el cliente en un solo recorrido de la base de datos. Esto puede acelerar significativamente las aplicaciones que utilizan muchas LOB pequeñas.

### **Datos de Matriz**

Una de las características exclusivas de ODP.NET es la capacidad de trasladar matrices entre la base de datos y el Entorno .NET. Las matrices facilitan el hecho de compartir grandes grupos de datos del mismo tipo de datos entre la base de datos y el cliente. ODP.NET utiliza matrices PL/SQL asociativas en la base de datos para transmitir los datos dentro y fuera de las matrices .NET.

### **Entorno .NET de 64 bits**

Con la incorporación del Entorno .NET de 64 bits, los desarrolladores .NET tienen acceso a los sistemas de hardware más escalables y de alto desempeño. Tienen la opción de elegir entre los procesadores AMD64 e Intel EM64T para Windows x64 y los procesadores Itanium para Windows Itanium. Los sistemas de 64 bits tienen la capacidad de abarcar directamente una mayor cantidad de memoria que los sistemas de 32 bits y tienen componentes optimizados de hardware para informática de alto desempeño. A partir de la versión 10.2.0.3, ODP.NET soporta ambos entornos .NET de 64 bits con un *driver* de acceso a datos nativo de 64 bits para cada

plataforma. Los desarrolladores pueden ahora implementar su nivel medio ODP.NET como una aplicación de 64 bits para aprovechar un hardware más escalable.

#### **Desempeño – Nuevo en ODP.NET 11g**

Oracle Database 11g presenta nuevas optimizaciones de desempeño, muchas de las cuales pueden ser utilizadas por los desarrolladores de aplicaciones .NET sin realizar ningún cambio en sus códigos cliente. Estas nuevas características incluyen un caché de resultados cliente, recuperaciones LOB más rápidas y caché optimizado de sentencias.

#### **Caché de Resultados Cliente**

Con el cliente y servidor Oracle Database 11g, las aplicaciones ODP.NET pueden utilizar el caché de resultados cliente de Oracle para mejorar los tiempos de respuesta de las consultas ejecutadas varias veces. Esta característica activa el caché del lado del cliente respecto de los grupos de resultados de las consultas SQL en la memoria. El caché de resultados cliente es completamente transparente a las aplicaciones ODP.NET, y el caché de datos de los resultados se mantiene en concordancia automática con cualquier cambio del lado del servidor de la base de datos o con cualquier sesión que pudiera alterar los resultados.

**Las nuevas características de Oracle Database 11g están disponibles para mejorar el desempeño de ODP.NET. Estas características incluyen un caché de resultados cliente, búsquedas LOB más rápidas y desempeño más rápido con el caché de sentencias.**

Las aplicaciones .NET que invocan la misma consulta varias veces tienen un mejor desempeño ya que los resultados de las consultas se recuperan localmente. El procesamiento cliente local es más rápido que realizar varios recorridos hacia la base de datos para volver a realizar una consulta y buscar los resultados. Si las aplicaciones frecuentemente ejecutan las mismas consultas, experimentarán una mejora significativa en el desempeño cuando sus resultados se coloquen en el caché cliente, así como una reducción en la carga del servidor de base de datos.

En el servidor de base de datos, el caché cliente reduce la carga del tráfico de red y del CPU servidor que se habría consumido para el procesamiento y la generación de los resultados de consulta, mejorando así la escalabilidad del servidor. Las sentencias ODP.NET de múltiples sesiones pueden concordar con el mismo grupo de resultados en caché de la memoria del proceso cliente si tienen parámetros de sesión, valores de enlace, texto SQL y esquemas similares. Caso contrario, la realización de consultas se produce en el servidor. Esto implica que múltiples usuarios ODP.NET tienen acceso al mismo caché de resultados, lo cual minimiza las redundancias de caché y ahorra memoria.

Debido a que el caché cliente mantiene una concordancia automática con los datos del servidor de base de datos, los desarrolladores no necesitan escribir el código para garantizar que el caché y el servidor están sincronizados. Si ocurre algún cambio en el servidor que pueda invalidar los datos en el caché cliente, el cliente Oracle automáticamente invalidará el caché y lo actualizará la próxima vez que se realice la consulta.

#### **Búsquedas LOB Más Rápidas**

ODP.NET 11g mejora el desempeño de recuperación de LOB pequeñas al reducir la cantidad de recorridos hacia la base de datos necesarios para buscar, con anterioridad, los metadatos LOB. Esta mejora está disponible a partir de Oracle Database 11g y es transparente para el desarrollador, quien no necesita ningún cambio de código para utilizar esta característica.

#### **Caché Optimizado de Sentencias**

ODP.NET 11g mejora la actual infraestructura del caché de sentencias para poner en caché los contextos de parámetros ODP.NET. Esta mejora funciona con cualquier versión del servidor de base de datos Oracle actualmente soportada. Los

desarrolladores .NET experimentarán una mejora de desempeño al ejecutar consultas de sentencias en caché. Esta mejora es transparente para los desarrolladores, y no requiere cambios de código.

#### **Notificación de Cambios en la Base de Datos**

La notificación de cambios en la base de datos permite a las aplicaciones cliente recibir notificaciones cuando se realizan cambios DML o DDL en un objeto de interés de la base de datos, incluso cuando un cliente ya no tiene una conexión con el servidor de base de datos. Los desarrolladores .NET ahora pueden colocar sus datos en caché en el nivel medio sin tener que preocuparse por los datos en caché que pierden sincronización con la base de datos. Si ocurre un cambio en uno de estos objetos de datos en caché o en las filas de datos, entonces ODP.NET recibirá una notificación de la base de datos. Esta característica puede utilizarse tanto en el Entorno .NET 1.x, 2.0 como en versiones posteriores.

Para utilizar la notificación de cambios, la aplicación cliente registra una consulta con la base de datos. Cuando una consulta tiene dependencias en los objetos de base de datos subyacentes y se produce un cambio en un objeto validado, la base de datos emite una notificación de cambio a la aplicación cliente. La notificación solo puede contener metadatos de los datos u objetos que cambiaron; no contiene los datos cambiados.

Los desarrolladores .NET pueden crear un administrador de eventos cliente para reemitir la consulta registrada con el fin de obtener los datos cambiados. Tradicionalmente, el caché de datos es efectivo para mejorar la escalabilidad de la aplicación a fin de permitir el rápido acceso a los datos sin realizar costosos recorridos a la base de datos. Pero esta escalabilidad se presenta con una compensación, ya que no es una garantía que los datos concuerden con el servidor de base de datos después de la consulta inicial. Así, los datos del caché cliente corren el riesgo de volverse obsoletos.

La notificación de cambios en la base de datos resuelve el problema de la pérdida de vigencia de los datos en caché. A pesar de que la notificación de cambios en la base de datos es similar a un *trigger* en cuanto a que responde a un evento particular, un *trigger* actúa inmediatamente, mientras que una notificación de base de datos es solo un alerta, no una acción. Determinar qué acción tomar, si existe alguna, y cuándo tomarla, depende de la aplicación. La aplicación puede actualizar inmediatamente los objetos antiguos, posponer la actualización o ignorar la notificación. Cada aplicación .NET puede responder de distintas maneras a un cambio particular de la base de datos. Asimismo, a medida que se agregan aplicaciones adicionales a la base de datos, es más fácil modificar el administrador de eventos de la aplicación que modificar un *trigger* de base de datos. Modificar el *trigger* puede requerir volver a probar cómo trabajan las aplicaciones existentes con el nuevo código *trigger* de base de datos, mientras que modificar solo la nueva aplicación .NET aísla mejor los límites de prueba.

Las aplicaciones web a menudo colocan en caché una variedad de datos y no todos deben ser actualizados en tiempo real. Por ejemplo, un pronóstico de tiempo solo puede actualizarse periódicamente. Los usuarios finales no necesitan consultar la base de datos cada vez que se visita la página web. Como muchas personas solicitarán los mismos datos, la escalabilidad y el desempeño de la aplicación se mejoran enormemente al colocar en caché los resultados y recuperar los datos del caché. En algún momento, el pronóstico de tiempo se actualiza, y el caché debe actualizarse también. Esto puede realizarse en el momento que cambia el pronóstico de tiempo en el servidor de base de datos.

Recibir notificaciones de cambio de base de datos requiere que el administrador de base de datos otorgue al usuario de aplicaciones el privilegio CAMBIAR NOTIFICACIÓN. Después de conectarse a la base de datos, los usuarios .NET pueden registrar sus consultas específicas para una notificación de cambio. El desarrollador crea un administrador de eventos del lado del cliente para indicar qué deberían hacer las aplicaciones al momento de recibir una notificación de cambio en la base de datos. Generalmente, el administrador de eventos volverá a consultar el servidor de base de datos y actualizará el caché.

Las siguientes clases ODP.NET se utilizan al crear aplicaciones para la notificación de cambios:

Un desafío constante para los caché del lado del cliente es mantener los datos en sincronización con los cambios en los datos del Servidor. Al utilizar la notificación de cambios de la base de datos, los clientes ODP.NET reciben un alerta cuando se modifican los datos en el servidor, incluso si no hay conexión activa de vuelta hacia la base de datos. Esto permite al cliente garantizar que su caché de datos permanezca en sincronización con la base de datos.

- OracleDependency – Crea una dependencia entre una aplicación y una base de datos Oracle. Permite a la aplicación recibir información de un cambio de datos (por ejemplo, una sentencia ACTUALIZAR), un cambio de esquema (por ejemplo, ALTERAR TABLA), o un evento global (por ejemplo, un corte de servicio de la base de datos). El administrador de eventos OnChange en esta clase ofrece al cliente la lógica para saber qué hacer después de recibir la notificación.
- OracleNotificationEventArgs – Brinda todos los detalles del evento cuando se produce una notificación de cambio.
- OracleNotificationRequest – Especifica las características de un pedido de notificación y su notificación, como el valor de intervalo del registro de notificación.

Las notificaciones de cambio de la base de datos Oracle tienen varias características que no están disponibles en el Servidor SQL. Oracle soporta todos los tipos de uniones, mientras que el Servidor SQL no soporta las consultas que contienen uniones externas ni automáticas. El Servidor SQL no soporta las notificaciones para las sentencias que utilizan visualizaciones, mientras que la notificación de cambio de la base de datos Oracle soporta las visualizaciones con la excepción de las visualizaciones fijas (por ejemplo, las tablas V\$) y las vistas materializadas. Las notificaciones del Servidor SQL también requieren referencias de columnas explícitas, mientras que las notificaciones de la Base de Datos Oracle soportan tanto las referencias de columnas explícitas como las de SELECCIONAR \*

Las notificaciones del Servidor SQL no son continuas. Cuando se publica una notificación del Servidor SQL, su administrador de notificaciones es eliminado de la base de datos. Si el administrador de notificaciones aún es necesario, la aplicación debe registrar un nuevo administrador de notificaciones. La notificación de cambios de la base de datos Oracle brinda la opción de continuar con el registro incluso después de cambios reiterativos. Esto es posible al configurar OracleNotificationRequest.IsNotifiedOnce como falso.

### Real Application Clusters (RAC)

RAC es una base de datos en cluster con una arquitectura compartida que supera las limitaciones de los enfoques tradicionales de bases de datos federadas y no compartidas. La base de datos RAC en cluster se aloja en múltiples nodos informáticos que permiten una mejor disponibilidad y escalabilidad que un simple servidor. Debido a que no se requieren cambios especiales de aplicaciones .NET ni de hardware, RAC puede crearse con hardware de commodity como *back end* para las aplicaciones existentes sin requerir ningún cambio de codificación.

ODP.NET siempre ha soportado el acceso a los datos para RAC de manera transparente. A fin de mejorar la administración de conexión ODP.NET basada en la información de carga de trabajo de la base de datos en tiempo real, Oracle introdujo dos propiedades del grupo de conexión para ODP.NET 10.2. La primera

característica, el balanceo de la carga de conexión en tiempo de ejecución, mejora la carga de trabajo en todas las instancias RAC, especialmente después de agregar o eliminar nodos de un cluster. La segunda característica, el failover de conexión rápida, elimina automáticamente las conexiones RAC afectadas del grupo de conexiones.

Con el balanceo de carga de conexión en tiempo de ejecución, el modo de asignación de las conexiones ODP.NET se basa en el objetivo de servicio y en el asesor de balanceo de carga de la base de datos durante el tiempo de ejecución. El balanceo de carga distribuye el trabajo en todas las instancias de base de datos RAC disponibles.

En general, las conexiones no se crean muy frecuentemente y tienen una larga duración. El trabajo ingresa al sistema con mucha frecuencia, utiliza estas conexiones del grupo y tiene una duración relativamente corta. El asesor del balanceo de carga indica qué instancias ODP.NET de RAC deberían asignarse a los trabajos entrantes a fin de obtener una calidad óptima de servicio. El balanceo de carga minimiza la necesidad de reasignar luego el trabajo a una instancia diferente para garantizar que los trabajos existentes se completen rápidamente. Las métricas por las cuales este trabajo se distribuye en todas las instancias están determinadas por el objetivo de servicio. El administrador de base de datos establece el objetivo de servicio por tiempo de servicio o por rendimiento.

La métrica del tiempo de servicio se basa en la rapidez con la cual la base de datos puede completar las tareas, esencialmente su tiempo de respuesta. Los datos del asesor de balanceo de carga se basan en el tiempo transcurrido para el trabajo realizado en el servicio, así como el ancho de banda disponible para el servicio. El tiempo de servicio es principalmente útil para las aplicaciones de base de datos donde las tareas se completan en distintos niveles, como en un sistema de compras por Internet.

**ODP.NET tiene dos características de grupo de conexiones para las bases de datos RAC: 1) Las conexiones ODP.NET equilibran automáticamente la carga en todos los nodos existentes de acuerdo con las métricas de cluster en tiempo real y las conexiones de base de datos son automáticamente eliminadas del grupo de conexiones.**

Asimismo, la métrica de rendimiento rastrea la rapidez con la que se completan las tareas de la base de datos, asignando más trabajo a los nodos donde las tareas se completan con la mayor rapidez. A diferencia del tiempo de servicio, la métrica de rendimiento intenta medir la eficacia de cada instancia. Mide la cantidad de recursos disponibles en el procesador para realizar tareas adicionales. Cuando se genera una nueva operación, ésta puede ser direccionada al nodo con la mayor cantidad de tiempo de procesador disponible. Esta métrica se aplica mejor a tareas relativamente uniformes, como ejecutar procesos *batch* homogéneos.

Al utilizar estos objetivos de servicio, el sistema genera un feedback. El trabajo es ruteado para brindar el mejor tiempo de servicio globalmente y responder positivamente a las cambiantes condiciones del sistema. Cuando un nodo RAC se incorpora o elimina del cluster, el balanceo de carga permite trabajar para una distribución más rápida y uniforme en todos los nodos a fin de justificar el cambio del sistema. Los usuarios finales luego experimentarán menos interrupciones o servicios lentos. En un estado estable, el sistema se acerca al equilibrio con un mejor rendimiento en todas las instancias RAC.

Para utilizar esta característica en ODP.NET, la aplicación debe activar el grupo de conexiones y establecer el parámetro del grupo de conexiones, el Balanceo de Carga, en verdadero.

La segunda característica configurable para el grupo de conexiones RAC, el failover de rápida conexión, permite a ODP.NET liberar automáticamente los recursos relacionados con las conexiones afectadas que fueron generadas por un nodo, miembro del servicio o servicio RAC inactivo. Sin esta característica, si un nodo RAC falla, el grupo de conexiones retendrá los recursos de conexión que ya no son válidos. Los usuarios finales pueden intentar utilizar estas conexiones afectadas del

grupo. Sin una manera de identificar estas conexiones, los administradores tendrían que volver a establecer todas las conexiones ODP.NET en el grupo cada vez que falla alguna parte del cluster RAC.

Estas conexiones ODP.NET afectadas ahora se eliminan automáticamente sin ninguna intervención del administrador durante el tiempo de ejecución. Para permitir esta característica ODP.NET, utilice el grupo de conexiones y establezca el parámetro del grupo de conexión, HA Events, en verdadero.

#### **Características XML**

Debido a que XML es un lenguaje conocido para la integración de datos y servicios web, muchos programadores .NET lo incorporan como parte integral de sus aplicaciones. XML es parte clave tanto de la base de datos Oracle como del Entorno .NET. ODP.NET permite a los desarrolladores explotar estas dos tecnologías conjuntamente: Oracle XML DB y .NET System.XML.

XML DB representa la tecnología Oracle para la recuperación y el almacenamiento XML nativo de alto desempeño disponible dentro del servidor de base de datos. Ofrece la capacidad exclusiva de administrar y almacenar tanto los datos estructurados como no estructurados conforme a un modelo de datos W3C XML estándar. XML DB brinda transparencia completa y capacidad de intercambio entre las metáforas XML y SQL. ODP.NET ofrece a los clientes .NET la funcionalidad XML DB, permitiendo a los desarrolladores compartir y realizar cambios a XML entre la base de datos y .NET. Este soporte se extiende a XML basado en esquemas y no basado en esquemas a fin de brindar flexibilidad para distintos requisitos de aplicaciones. Asimismo, ODP.NET contiene dos tipos de datos XML nativos, OracleXMLType y OracleXMLStream para una mejor administración de datos XML cliente. XML puede ser fácilmente compartido entre los servicios XML DB y Microsoft System.XML.

System.XML constituye un grupo de interfaces para manipular los grupos de datos XML de los proveedores de datos .NET. ODP.NET interopera con las interfaces de programación de System.XML, alimentando los datos por medio de las interfaces ODP.NET DataAdapter. Una de las diferencias clave entre XML DB y System.XML es que el primero brinda servicios XML donde residen los datos, en el servidor de base de datos; el último manipula XML del lado del cliente. Como tal, ODP.NET ofrece una opción más amplia para que los programadores tomen la tecnología XML que mejor se ajuste a los requerimientos de su proyecto. Con ODP.NET, los datos relacionales y de objeto relacional pueden ser accedidos como XML desde una instancia de base de datos Oracle hasta un entorno Microsoft .NET. Los cambios en XML pueden realizarse y guardarse como XML o datos relacionales en el servidor. ODP.NET incluye el soporte de OracleXMLType basado en esquemas, así como XML no basado en esquemas.

ODP.NET brinda soporte para Oracle XML DB e interopera con System.XML. ODP.NET puede trabajar con XML basado en esquemas y no basado en esquemas con soporte para tipos de datos Oracle XML nativos utilizando OracleXMLType y OracleXMLStream.

#### **ADO.NET 2.0**

A partir de la versión 10.2.0.2, ODP.NET brinda soporte para ADO.NET 2.0, e introduce un nuevo nivel de abstracción para el Nivel de Acceso a Datos (DAL). En lugar de utilizar clases de proveedores específicos que implementen una interface genérica, ADO.NET 2.0 ofrece las clases DbCommon, que son heredadas de los espacios de nombre System.Data.Common y permiten a los desarrolladores utilizar *factory classes*. *Factory classes* del proveedor de base de datos es una manera sencilla de crear un solo grupo de código genérico de acceso a datos en cualquier base de datos.

La clase OracleClientFactory soporta la creación de todas las clases en System.Data.Common. Debido a que estas clases concretas se heredan de las clases de base abstracta DbCommon, usted puede escribir códigos DAL genéricos utilizando nombres de objetos de clase base DbCommon. Existen algunas áreas de

DAL que siguen siendo específicas de fuentes de datos, con inclusión de la cadena de conexión, SQL, y las llamadas a procedimientos almacenados.

Una incorporación clave en ADO.NET 2.0 es una mejor administración de la cadena de conexión. La clase `DbConnectionStringBuilder` tiene un diccionario que mapea los nombres de parámetros genéricos hasta los nombres de los parámetros específicos del proveedor. La clase `OracleConnectionStringBuilder` se hereda de la clase genérica `DbConnectionStringBuilder`, y la extiende, para exponer las propiedades de la cadena de conexión específica de Oracle. Los programadores pueden utilizar `OracleConnectionStringBuilder` dinámicamente para establecer los parámetros de la cadena de conexión en tiempo de ejecución y/o obtener los parámetros de la cadena de conexión del archivo de configuración de aplicaciones.

Las API de reconocimiento para el esquema ADO.NET 2.0 ofrecen una manera genérica de recuperar metadatos desde cualquier fuente de datos. Los desarrolladores pueden obtener los metadatos de Oracle al utilizar una invocación del método `OracleConnection.GetSchema`. Existen cinco tipos de metadatos comunes que pueden exponerse: `MetaDataCollections`, `Restrictions`, `DataSourceInformation`, `DataTypes` y `ReservedWords`. Asimismo, existe información adicional de fuentes de datos específicas de ODP.NET que puede ser recuperada. Cuando se trabaja con grandes `DataSets`, los desarrolladores querrán minimizar la cantidad de recorridos completos a la base de datos necesarios para realizar las actualizaciones. Al utilizar la propiedad de `OracleDataAdapter.UpdateBatchSize`, los programadores pueden especificar la cantidad de filas que deben actualizar por recorrido. Para actualizar todas las filas modificadas en un solo recorrido, `UpdateBatchSize` puede establecerse en cero.

**ODP.NET soporta las características de ADO.NET 2.0, con inclusión de las fábricas del proveedor, el generador de cadenas de conexión, las API de reconocimiento de esquema, y las actualizaciones *batch* de `DataAdapter`.**

#### Tipos Nativos de Oracle

Microsoft ha incorporado un grupo de tipos de datos unificados entre los diferentes lenguajes de programación .NET. Con ODP.NET, los programadores .NET tienen acceso a los tipos de datos .NET así como los tipos de datos de Oracle en .NET. Los tipos de datos de Oracle pueden manipularse totalmente dentro de una aplicación .NET e interoperar con los tipos de datos .NET. Los tipos nativos de Oracle ofrecen funcionalidad avanzada para almacenar y manipular las estructuras de datos de la base de datos, como XML, los grupos de resultados, imágenes, o documentos de Microsoft Office. Incluso con *scalar types*, como `OracleDecimal`, el equivalente al tipo decimal .NET, los tipos de Oracle brindan funcionalidad adicional. En el ejemplo de `OracleDecimal`, este tipo de datos ofrece un mayor nivel de precisión de 38 dígitos que el decimal .NET con una precisión de 28 dígitos.

ODP.NET soporta una variedad de tipos Oracle avanzados dentro del entorno .NET, con inclusión de los Cursores REF, `XMLType`, LOBs (CLOBs, BLOBs, NCLOBs), BFILEs, LONGs, RAWs, LONG RAWs y tipos de datos N. Una de las limitaciones para utilizar otros proveedores de datos .NET es que los usuarios pueden verse limitados en cuanto a la funcionalidad del tipo de datos Oracle. Por ejemplo, en ODP.NET, múltiples grupos de resultados obtenidos de un procedimiento almacenado como parámetros de salida del Cursor REF pueden accederse de manera arbitraria. Se pueden leer los resultados del segundo Cursor REF sin recuperar los resultados del primero. Con otros proveedores .NET, se puede acceder a los datos en forma lineal, donde el primer grupo de resultados debe ser recuperado antes de acceder al segundo. Esto podría afectar negativamente el desempeño.

A partir de ADO.NET 2.0, los tipos de datos Oracle pueden almacenarse nativamente dentro de .NET DataSet. Anteriormente, solo los tipos de datos .NET podían utilizarse en DataSet. Al establecer `OracleDataAdapter.ReturnProviderSpecificTypes` como verdadero, DataSet se



completará con los tipos de datos específicos de ODP.NET cuando se convoca OracleDataAdapter.Fill.

#### Otras Características Importantes

ODP.NET se expone a muchas otras características de la base de datos Oracle, con inclusión del soporte de Unicode, transacciones y PL/SQL. Los usuarios de ODP.NET pueden ejecutar todas las funciones y los procedimientos PL/SQL almacenados en la base de datos. PL/SQL puede estar empaquetado o no empaquetado, o incluso existir como PL/SQL anónimo dentro de .NET. El PL/SQL anónimo generalmente se implementa para realizar un proceso *batch* de un grupo de sentencias SQL y ejecutar las sentencias en el recorrido de una base de datos. Las sentencias *batch* son una técnica útil para la optimización de desempeño.

ODP.NET puede participar en aplicaciones transaccionales con la base de datos Oracle como administrador de recursos. ODP.NET implementa Microsoft Distributed Transaction Coordinator (DTC) para supervisar una transacción en un entorno Windows. Oracle Services para Microsoft Transaction Server (OraMTS) actúa como proxy entre ODP.NET, DTC y la base de datos Oracle para coordinar estas transacciones. OraMTS brinda una sólida arquitectura para que los programadores ODP.NET logren que sus aplicaciones transaccionales mantengan una alta disponibilidad y escalabilidad. En .NET Framework 2.0 y versiones posteriores, ODP.NET soporta tanto las transacciones locales como las distribuidas por medio del espacio de nombre System.Transactions.

ODP.NET tiene soporte total de Unicode, de modo que los usuarios .NET pueden globalizar sus aplicaciones fácilmente en múltiples idiomas escritos. Este soporte de globalización permite a los desarrolladores crear un solo grupo de códigos para múltiples configuraciones de cultura/idiomas. La globalización de ODP.NET extrae la configuración del idioma nacional informático del cliente para desplegar la información en el formato local específico. Por ejemplo, un browser establecido en japonés tendrá su moneda expresada en yen. Sin una codificación adicional, la misma aplicación puede implementarse en Alemania para mostrar la moneda en euros. Esto hace que el desarrollo de aplicaciones para múltiples escenarios sea más fácil y más rápido, ya que no se necesita una codificación adicional.

#### Oracle Database Extensions para .NET

Oracle Database Extensions para .NET es una característica de Oracle Database 11g en Windows que facilita el desarrollo, la implementación y ejecuta funciones y procedimientos almacenados escritos en un lenguaje .NET administrado como C# o VB.NET.

Las funciones y procedimientos almacenados .NET fueron desarrollados utilizando Visual Studio y luego implementados utilizando el Wizard de Implementación .NET fuertemente integrado, incluido con Oracle Developer Tools para Visual Studio .NET. El wizard de desarrollo carga la compilación .NET recientemente creada en Oracle y registra el procedimiento o la función con la base de datos.

Los procedimientos almacenados que requieren acceso a los datos utilizan Oracle Data Provider para .NET. Tanto ADO.NET 2.0 como .NET Framework 2.0 son respaldados en los procedimientos almacenados .NET. El código utilizado para el procedimiento o la función es casi exactamente el mismo que el escrito para una aplicación del lado del cliente, a excepción de una “conexión de contexto” opcional en la cual el procedimiento utiliza la misma conexión que quien la invoca. El código también puede ser escrito de modo que pueda ejecutarse tanto del lado del cliente como en un procedimiento almacenado sin requerir cambios.

Después de la implementación, un procedimiento almacenado .NET puede ser invocado desde el código .NET, desde un *trigger* o sentencia SQL, desde otro

ODP.NET soporta tipos de base de datos Oracle nativos, como REF Cursors, dentro del entorno .NET. Otras importantes características incluyen la capacidad de poder utilizar cualquier tipo PL/SQL; transacciones locales y distribuidas; y aplicaciones internacionalizadas con Unicode.

procedimiento almacenado .NET, PL/SQL o Java, o desde cualquier lugar donde se permita invocar una función o procedimiento almacenado.

## IMPLEMENTACIÓN DE APLICACIONES

Implementar aplicaciones en Windows requiere la estrecha integración con los servicios Windows nativos y los servidores de nivel medio a fin de garantizar la adecuada interoperabilidad de las aplicaciones. Oracle Database 11g ofrece esta sólida integración, permitiendo a las empresas aprovechar la principal funcionalidad del servidor y del sistema operativo con el grupo de características avanzadas de Oracle. La integración de Oracle garantiza que la interoperabilidad transparente ofrecida sea escalable, segura y se encuentre a disposición. De esta manera, las empresas pueden focalizar sus recursos hacia la creación de una lógica comercial de sus aplicaciones, en lugar de que se produzcan ineficiencias o incompatibilidades de producto.

Oracle Database Extensions para .NET es una característica de la base de datos Oracle en Windows que facilita el desarrollo, la implementación y ejecuta funciones y procedimientos almacenados escritos en un lenguaje .NET administrado como C# o VB.NET. Oracle ofrece muchos productos diferentes que integran las aplicaciones Oracle en entornos de Windows, con inclusión de Oracle Fail Safe, la integración de seguridad de Windows y la integración con Active Directory.

Oracle Database 11g incluye muchos grupos de herramientas para garantizar un entorno de implementación de aplicaciones Windows optimizado. Al implementar servidores Oracle críticos en clusters de Windows, Oracle Fail Safe garantiza que esos servidores sigan a disposición y proporcionen sólido soporte de failover a través de Microsoft Cluster Service. Oracle se integra con las herramientas de directorio y seguridad nativas de Windows que mejoran la seguridad de la base de datos y facilitan la administración de los usuarios.

Oracle Database 11g en Windows introduce soporte adicional para Active Directory, mejor soporte Kerberos, y nueva integración con el Servicio de Copias de Respaldo (*Volume Shadow Copy Service*).

## Directorio

Los servidores de directorio a menudo facilitan la administración de recursos y usuarios al centralizar su administración. Incluso con esta centralización, un servidor de directorio debe poder integrarse con otras aplicaciones, organizaciones departamentales o socios comerciales, cada uno de los cuales posee su propio servidor de directorio o repositorio de usuarios separado. Como tal, las soluciones de directorio deben ser escalables para manejar grandes grupos de usuarios e interoperar con otros repositorios externos.

Oracle Internet Directory (OID) permite a las empresas centralizar sus servidores de directorio y aprovechar sus servidores de directorio existentes, como Active Directory (AD). Oracle Identity Management consta de un grupo de interfaces y servicios incorporados que facilitan el control de acceso, la conexión única, la administración de perfiles de usuarios, la sincronización y el suministro entre Oracle y otros repositorios. Estos repositorios incluyen otros repositorios, como AD; los repositorios de usuarios de aplicaciones; o las tablas de base de datos que contienen información de recursos humanos.

Con Oracle Identity Management, las empresas pueden administrar todo el ciclo de vida de las identidades de los usuarios a través de todos los recursos empresariales tanto dentro como fuera del firewall. Los administradores pueden implementar aplicaciones de manera más rápida, aplicar la protección más minuciosa en los recursos empresariales, eliminar automáticamente los privilegios de acceso latente, y mucho más. Oracle Identity Management es miembro de la familia de productos de Oracle Fusion Middleware, lo cual proporciona mayor agilidad, mejor toma de decisiones y costo reducido para los diversos entornos de IT de hoy.

Sin OID ni Oracle Identity Management, Oracle ofrece servicios para la integración directa de la base de datos con AD. La base de datos Oracle incluye dos características de integración nativa que aprovechan AD:

- Conexión única de Oracle a través de una autenticación AD nativa
- Administración de Oracle Net Naming con AD

Estas características no requieren OID, solo AD y la base de datos Oracle. No obstante, los administradores tienen la opción de utilizar ambas características con OID, de ser necesario.

#### **Autenticación Nativa y Active Directory**

Los administradores de Oracle a menudo requieren seguridad de escala empresarial y administración de esquemas. Esto es particularmente cierto para las empresas con grandes poblaciones de usuarios. OID está diseñado para manejar estos entornos empresariales e interoperar con los servidores de directorios departamentales, como Active Directory. La administración y la seguridad de Oracle se integran con ambos servidores de directorio para ofrecer administración y acceso sin dificultades.

Oracle permite la conexión única a través de los mecanismos de autenticación de Microsoft, permitiendo a Active Directory realizar la identificación de los usuarios para las bases de datos Oracle. Con la autenticación nativa activada, los usuarios de base de datos pueden aprovechar la conexión única para acceder a Oracle ingresando en Active Directory. Esta integración simplifica la administración de los usuarios finales y elimina las credenciales redundantes de seguridad.

El mapeo de usuarios empresariales de Oracle permite que múltiples usuarios del sistema operativo accedan a la base de datos como usuario único de base de datos global. En entornos exclusivos de Windows, estos mapeos de usuarios empresariales pueden almacenarse en el Directorio Activo. Por ejemplo, toda una unidad organizacional (OU) LDAP en el Directorio Activo puede mapearse a un solo usuario de base de datos.

Oracle almacena el mapeo de roles empresariales en LDAP. En Windows, esta característica de almacenamiento LDAP está certificada en OID, así como en el Directorio Activo. Con los roles empresariales, los privilegios para múltiples bases de datos pueden administrarse en el nivel de dominio a través de los directorios. Esto se logra al asignar grupos y usuarios de Windows a los roles empresariales de Oracle registrados en el almacén LDAP.

#### **Oracle Net Naming con Active Directory**

Oracle aprovecha la tecnología LDAP a través de OID y Active Directory para mejorar la administración de la información de conectividad de la base de datos. Tradicionalmente, los usuarios finales se refieren a los servidores de base de datos con nombres de clase Oracle Net determinados a través del archivo de configuración TNSNAMES.ORA. Este archivo debe administrarse en cada máquina cliente. Para facilitar la capacidad de administración, los nombres de Oracle Net pueden almacenarse a través de OID o el Directorio Activo. Centralizar dicha información en un directorio elimina los gastos administrativos y libera a los usuarios de la obligación de configurar sus máquinas cliente individuales.

Asimismo, las herramientas de Windows, como Windows Explorer y Active Directory Users and Computers, pueden conectarse a la base de datos y probar la conectividad de la base de datos. Las herramientas de Oracle también pueden participar. Oracle Database Configuration Assistant automáticamente registra los objetos de la base de datos en un directorio LDAP. Oracle Net Manager, mientras tanto, registra los objetos de servicio neto con el directorio. Estas herramientas simplifican aún más la administración de la base de datos y el directorio.

#### **Características de Active Directory – Nuevo en Oracle Database 11g**

En Oracle Database 11g, los administradores pueden decidir no permitir el acceso anónimo a la información de servicios de la base de datos de un directorio. Los clientes deberán autenticarse antes de realizar consultas de nombres sobre la base del

La plataforma de administración de identidad de Oracle permite a Oracle Internet Directory (OID) centralizar la sincronización y el abastecimiento con otros directorios, con inclusión de Active Directory. Oracle hace que la conexión única de Windows sea posible y facilita la administración de los usuarios al permitir que los mapeos de usuarios se almacenen en Active Directory (Directorio Activo) o en OID. Asimismo, Net naming (tnsnames.ora) puede almacenarse en un Directorio Activo o en OID para lograr una administración centralizada. Las nuevas características del Directorio Activo en Oracle Database 11g mejoran la seguridad al impedir el acceso anónimo.

directorio LDAP. Con las consultas de nombres basadas en Microsoft Active Directory, Oracle Database utiliza la autenticación nativa basada en Windows. Con las consultas de nombre basadas en Oracle Internet Directory (OID), Oracle Database realiza la autenticación utilizando *wallets*.

## Seguridad

Oracle Database 11g brinda una estrecha integración con el modelo de seguridad nativo de Windows para proteger mejor la seguridad de las aplicaciones. En un entorno cliente/servidor, existen tres opciones para los sitios que desean aprovechar al máximo las credenciales Windows o los servicios de autenticación para ingresar a la base de datos Oracle:

- La base de datos Oracle incluye Windows Native Authentication Adapter, que se instala automáticamente con Oracle Net en el servidor cliente. Este adaptador permite a los usuarios de base de datos tener capacidades de conexión única al utilizar las credenciales de los usuarios Windows para la autenticación de la base de datos. Para utilizar esta característica, los usuarios de Windows deben estar definidos como usuarios de base de datos externos. Estos usuarios pueden tener roles externos asignados en Microsoft Active Directory Services, los cuales serán respetados por la base de datos.
- Oracle Advanced Security, una opción de base de datos, soporta la autenticación de la base de datos utilizando *tickets* Kerberos emitidos por Microsoft Key Distribution Center (MSKDC). Esta capacidad permite a los usuarios a quienes se les haya emitido un *ticket* Kerberos válido en el entorno de Windows ingresar a sus cuentas de base de datos sin tener que proveer el nombre de usuario/contraseña.
- La opción Oracle Advanced Security también soporta la autenticación de base de datos sobre SSL utilizando un certificado X.509v3. Microsoft Certificate Store (MCS) puede emitir este certificado. Para utilizar esta característica, el certificado debe estar contenido en un Oracle Wallet configurado en el cliente. Oracle Wallet puede almacenarse tanto en el área del perfil de usuario de Windows Registry como en la ubicación de un archivo arbitrario del cliente.

Para los entornos web, la integración con MCS es posible a través de Oracle Fusion Middleware (FMW). Además de respaldar los mecanismos de autenticación basados en contraseñas, Oracle FMW respalda la autenticación basada en certificados. Esta certificación puede ser generada por MCS. Para utilizar la autenticación basada en certificados, un certificado de usuario debe completarse en el *wallet* browser y en el ingreso de usuario correspondiente en Oracle Internet Directory.

### Oracle Wallets en Windows Registry

Oracle Database wallets pueden almacenarse en Windows Registry, brindando una mayor seguridad para los clientes Windows. Sin esta característica, los wallets de Oracle a menudo se almacenan en el sistema de archivos Windows. Si los permisos de archivos del sistema operativo no son seguros, los wallets de Oracle tampoco son seguros. Los permisos de archivos no seguros pueden provenir de una administración indebida o de la falta de seguridad en los archivos del sistema operativo. En los sistemas Windows, por consiguiente, la seguridad de la infraestructura de claves públicas (PKI) se mejora al almacenar Oracle wallets en el área del perfil de usuarios de Windows Registry. Esta área de registro es accesible solamente para el usuario debidamente registrado. Se pueden utilizar y almacenar múltiples Oracle wallets en el área de los perfiles.

La conexión única para las aplicaciones PKI de Oracle se configura a través de las herramientas de Oracle Wallet Manager y Oracle Enterprise Login Assistant. Wallet Manager crea Oracle wallets encriptados, mientras que Enterprise Login Assistant crea wallets descriptados. Los wallets descriptados son luego utilizados por las

aplicaciones PKI de Oracle para la autenticación SSL. Ambas herramientas han sido mejoradas para soportar el almacenamiento y la ubicación de Oracle wallets en el registro, así como el sistema de archivos por defecto, en caso de ser necesario.

La base de datos Oracle respalda una cantidad de mecanismos de seguridad de Windows, con inclusión de la conexión única, Windows Kerberos, y Windows SSL. La capacidad de almacenar Oracle wallets en Windows Registry elimina el almacenamiento de los archivos inseguros del sistema operativo. Oracle se integra con Microsoft Certificate Store, permitiendo a las aplicaciones Oracle PKI y Windows PKI operar juntas sin dificultades. Las nuevas características de Kerberos en Oracle Database 11g mejoran las opciones de seguridad y la interoperabilidad de Windows.

#### **Integración con Microsoft Certificate Store**

La integración de la base de datos Oracle con Microsoft Certificate Store permite a las aplicaciones PKI de Oracle interoperar con los productos que utilizan Windows PKI. Cuando esta característica está activada, la seguridad de Oracle PKI utiliza Microsoft CryptoAPIs para acceder a Microsoft Certificate Store. Los CryptoAPI se utilizan para operaciones, como el ingreso, la encriptación, la desencriptación y la validación. Wallet Resource Locator (WRL) determina el tipo PKI y brinda toda la información necesaria para localizar el wallet.

Con esta característica, las aplicaciones Oracle, y las que no son de Oracle, pueden aprovechar el mismo grupo de credenciales PKI (por ejemplo, certificados, claves, lista de revocaciones y puntos de confianza) para los servicios de seguridad de claves públicas, como la autenticación y la encriptación.

#### **Características de Seguridad de Windows – Nuevo en Oracle Database 11g**

La implementación de Kerberos en Oracle Database 11g ahora utiliza los algoritmos de encriptación segura como 3DES y AES que están disponibles en DES. Esto hace que utilizar Kerberos sea más seguro. El mecanismo de autenticación de Kerberos en la base de datos Oracle ahora respalda los siguientes tipos de encriptación:

- DES3-CBC-SHA (algoritmo DES3 en modo CBC con HMAC-SHA1 como *checksum*)
- RC4-HMAC (algoritmo RC4 con HMAC-MD5 como *checksum*)
- AES128-CTS (algoritmo AES con clave de 128 bits en modo CTS con HMACSHA1 como *checksum*)
- AES256-CTS (algoritmo AES con clave de 256 bits en modo CTS con HMACSHA1 como *checksum*)

La implementación de Kerberos ha sido mejorada para interoperar sin dificultades con Microsoft Key Distribution Center. Asimismo, el nombre principal de Kerberos ahora puede contener más de 30 caracteres. La cantidad de caracteres permitidos en el nombre del usuario de base de datos ya no está restringida. Estas mejoras permiten asegurar mejor las aplicaciones con las bases de datos Oracle y Windows.

#### **Backup y recuperación con el Servicio de Copias de Respaldo – Nuevo en Oracle Database 11g**

El Servicio de Copias de Respaldo (VSS) en las plataformas del servidor Windows permite realizar backups de volumen mientras las aplicaciones, como las de Oracle, continúan realizando escrituras en los volúmenes. Una copia “de respaldo” es una *snapshot* consistente de los datos de un volumen o componente en un momento determinado. Un grupo de copias de “respaldo” es una recopilación de copias tomadas en el mismo momento. Oracle ahora se integra con VSS para permitir el backup y la recuperación de los archivos de la base de datos Oracle.

VSS incluye una interfase específica de Windows que permite la coordinación entre las personas que solicitaron el backup de datos, las personas que actualizan los datos en el disco y los proveedores que administran el almacenamiento. Dentro de las novedades de Oracle Database 11g, se encuentran las funciones de la base de datos Oracle, como el Escritor VSS que se integra con las aplicaciones activadas por VSS. Como escritor, puede realizar un backup de la base de datos Oracle y almacenarlo con los sistemas de almacenamiento y el software activado por VSS. Un beneficio clave es la capacidad de utilizar una aplicación activada por VSS para realizar un backup online de toda la base de datos.

El escritor VSS de Oracle es un servicio de Windows que coordina una instancia de base de datos Oracle y otros componentes VSS. El servicio del escritor, el cual se inicia conforme a una cuenta de usuario con privilegios SYSDBA, se ejecuta de modo separado desde la instancia de base de datos. El escritor Oracle VSS soporta las copias de respaldo basadas en componentes, que son un grupo de archivos de la base de datos. Durante el backup, el escritor VSS de Oracle guarda el *redo* generado durante la creación de una snapshot en un documento de metadatos. Durante una operación de restauración, el escritor automáticamente extrae el *redo* de un documento de metadatos y lo aplica a los archivos restaurados de una *snapshot*.

El escritor VSS de Oracle también soporta copias de respaldo basadas en volúmenes, las cuales son una *snapshot* de volúmenes o unidades completas. La base de datos Oracle colocará sus archivos en el estado adecuado para crear copias de respaldo. Por ejemplo, los archivos de datos se colocan en modo de backup activo y se crea un nuevo archivo de control de *snapshots* para la base de datos en modo ARCHIVELOG. El escritor VSS de Oracle excluye archivos como el archivo de control actual y los archivos *redo* online de las copias de respaldo. El escritor emite un error si no se pudo sacar la *snapshot*. Por ejemplo, si una base de datos NOARCHIVELOG se abre en modo de lectura/escritura, luego el escritor emite un error indicando que la snapshot no ha sido posible.

Con Oracle, las aplicaciones activadas por VSS pueden trabajar fácilmente con la base de datos Oracle para realizar el backup y recuperar las *snapshots*.

### Alta Disponibilidad con Oracle Fail Safe

Oracle Fail Safe es una característica clave de Oracle Database 11g que ofrece gran disponibilidad de las soluciones críticas implementadas en los clusters de Windows. Un cluster elimina los sistemas host individuales como puntos de falla. Oracle Fail Safe trabaja con Microsoft Cluster Service (MSCS) para garantizar que si se produce una falla en un nodo de cluster, entonces, la base de datos y las aplicaciones de Oracle que se ejecuten en ese nodo experimentarán un *fail over* (se trasladarán) automática y rápidamente al nodo que queda. A diferencia de RAC, Oracle Fail Safe es un cluster activo-pasivo, por medio del cual la máquina *standby* no reemplaza la carga de trabajo hasta que el nodo activo falla.

Oracle Fail Safe ha sido optimizado para los clientes de Windows con cargas de trabajo de aplicaciones y bases de datos que pueden manejarse por un solo sistema. Las soluciones de Oracle Fail Safe pueden implementarse en todos los clusters de commodity de Windows. Soporta clusters de hasta ocho nodos en Windows Server 2003 Datacenter, la configuración máxima de nodos actualmente disponible en Windows. Los productos soportados incluyen:

- Bases de datos Oracle (Oracle 11g, Oracle10g, Oracle9i)
- Oracle Applications versión 11i
- Oracle iAS, a excepción de los productos en caché, que incluyen
  - Oracle Forms Server
  - Oracle Reports Server
  - Oracle HTTP Server
- Oracle Services para Microsoft Transaction Server
- Oracle Intelligent Agent
- Aplicaciones instaladas como servicios de Windows

SAP, Baan, PeopleSoft, Lawson, J.D. Edwards, y otros proveedores de aplicaciones también han validado sus soluciones de software con Oracle Fail Safe.

A través de la Integración con Microsoft Cluster Service, Oracle Fail Safe garantiza que el software de Oracle continúa siendo altamente disponible en Windows. Fail Safe soporta la base de datos Oracle, el servidor de aplicaciones, y E-Business Suite

Oracle Fail Safe incluye dos componentes principales, el servidor y el administrador. El componente servidor, Oracle Services para MSCS, trabaja con el software cluster para garantizar un *fail over* rápido y automático durante los cortes de servicio planificados y no planificados. El administrador, Oracle Fail Safe Manager, es una interfase gráfica fácil de usar que trabaja con Oracle Fail Safe Server en uno o más clusters para realizar la configuración, administración, verificación y el balanceo de carga estático. Juntos, estos componentes brindan un valioso conjunto de características y herramientas integradas para la solución de problemas que permiten la rápida implementación de aplicaciones y bases de datos altamente disponibles— completas soluciones e-business.

#### **Capacidad de Administración Standby Lógica/Física y de Múltiples Clusters**

Con la integración de la base de datos Standby Lógica/Física, la base de datos Oracle Fail Safe puede ser ejecutada en un lugar primario y en cualquier lugar standby remoto. Fail Safe monitoreará la base de datos, la pondrá en línea y la sacará de línea. Esto permite a los clientes tener una única instancia de *fail over* dentro de un cluster con tiempo de baja mínimo, junto con beneficios incorporados para mantener los sitios standby. Oracle Fail Safe se utiliza comúnmente para los *failovers* planificados, como las actualizaciones de software y hardware comunes para los entornos de Windows. Con la base de datos Standby Lógica/Física, Fail Safe ahora brinda una verdadera protección ante desastres, ofrecida por la base de datos standby. Juntos, los dos productos se complementan entre sí para proporcionar una mayor disponibilidad y mayor protección antes desastres con cualquiera de estos productos.

**Oracle Fail Safe respalda la protección ante desastres al permitir el soporte de bases de datos Standby Lógicas/Físicas. Para una mejor capacidad de administración, se pueden administrar múltiples clusters desde la misma consola.**

Para los clientes que implementan múltiples cluster, Oracle Fail Safe Manager ofrece administración de más de un cluster, permitiendo al usuario administrar todos los clusters desde la misma ventana, en vez de hacerlo desde una ventana separada para cada cluster. Los administradores, por ejemplo, podrán modificar las credenciales de las cuentas de base de datos para las bases de datos en todos sus clusters Fail Safe de una sola vez utilizando un solo wizard. Los administradores podrán evitar la tediosa tarea de comparar políticas o modificar las preferencias de alerta o despliegue de clusters individualmente.

## **CONCLUSIÓN**

A medida que avanza la tecnología de implementación y desarrollo de las aplicaciones Windows, Oracle también realiza avances. Oracle Database 11g ofrece flexibilidad sin precedentes para que los clientes elijan las herramientas de aplicaciones .NET y la arquitectura que mejor se adapta a sus requisitos sin sacrificar el desempeño, la escalabilidad, la facilidad de uso y la seguridad. El compromiso de Oracle con los desarrolladores de Windows incluye la integración con ADO.NET y Visual Studio. Para los administradores Windows, Oracle se ha integrado estrechamente con muchos servicios operativos, como Active Directory, los Servicios de Información de Internet (IIS), la seguridad de Windows, y los Servicios Cluster de Microsoft. Al crear Oracle Database 11g teniendo en cuenta Windows, los clientes de aplicaciones de software pueden lograr interoperabilidad entre el sistema operativo y la base de datos.

Para obtener más información de Oracle sobre Windows y .NET, visite

el Centro de Tecnología OTN de Windows: <http://otn.oracle.com/windows>

el Centro para Desarrolladores OTN .NET: <http://otn.oracle.com/dotnet>

Oracle en Windows: <http://www.oracle.com/windows>



Oracle Database 11g en Windows: Desarrollo e Implementación  
Septiembre de 2007  
Autor: Alex Keh  
Coautor: Christian Shay

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

Consultas:  
Teléfono: +1.650.506.7000  
Fax: +1.650.506.7200  
[www.oracle.com](http://www.oracle.com)

Copyright © 2007, Oracle. Todos los derechos reservados.

El presente documento tiene solo fines informativos y su contenido está sujeto a cambios sin que medie notificación alguna. El presente documento puede contener errores y no está sujeto a ninguna otra garantía ni condición, ya sea oral o que se encuentre implícita en la ley, con inclusión de garantías y condiciones implícitas de comerciabilidad o aptitud para un fin específico. En especial, negamos cualquier responsabilidad con respecto al presente documento, el cual no crea obligación contractual alguna, sea en forma directa o indirecta. El presente documento no podrá ser reproducido ni transmitido de ninguna forma ni por ningún medio, sea electrónico o mecánico, con ningún fin, sin que hayamos otorgado previamente nuestro consentimiento por escrito. Oracle es una marca registrada de Oracle Corporation y/o sus afiliadas. Otros nombres pueden ser marcas comerciales de sus respectivos propietarios.