

IBM[®] DB2[®] Universal Database



Desarrollo de aplicaciones Enterprise Java utilizando DB2 UDB, Versión 7.2

Antes de utilizar esta información y el producto al que da soporte, asegúrese de leer la información general incluida en el apartado "Avisos" en la página 21.

Esta publicación es la traducción del original inglés IBM® DB2® *Universal Database Developing Enterprise Java Applications Using DB2 UDB, Version 7.2.*

Este documento contiene información sobre productos patentados de IBM. Se proporciona de acuerdo con un contrato de licencia y está protegido por la ley de la propiedad intelectual. La presente publicación no incluye garantías del producto y las declaraciones que contiene no deben interpretarse como tales.

Puede solicitar publicaciones a través del representante de IBM o sucursal de IBM de su localidad, o bien llamando a los números de teléfono 1-800-879-2755, en los Estados Unidos, o 1-800-IBM-4YOU, en Canadá.

Cuando envía información a IBM, otorga a IBM un derecho no exclusivo para utilizar o distribuir dicha información en la forma en que IBM considere adecuada, sin contraer por ello ninguna obligación con el remitente.

© Copyright International Business Machines Corporation 2000, 2001. Reservados todos los derechos.

Contenido

Desarrollo de aplicaciones Enterprise Java utilizando DB2 UDB, Versión 7.2 1

Introducción	1
Habilitación de Java	1
Soporte de controladores JDBC	1
Soporte de SQLj	4
Procedimientos almacenados de Java y funciones definidas por el usuario (UDF).	4
Desarrollo de aplicaciones Java para servidores Web	6
VisualAge para Java, Professional Edition	6
VisualAge para Java, Enterprise Edition	7
WebSphere Studio	9
WebSphere Application Server	10
Extensiones para el desarrollo de aplicaciones	15

XML Extender	15
Habilitación de MQSeries	15
Net.Data	18
Net Search Extender	18
Spatial Extender	19
Guía de Aprendizaje de DB2 Integrated Web Services.	19
Diseño de aplicaciones utilizando UML (Unified Modeling Language)	20
Resumen	20
Información adicional.	20
Avisos	21
Marcas registradas	24

Desarrollo de aplicaciones Enterprise Java utilizando DB2 UDB, Versión 7.2

por: Grant Hutchison, DB2/IBM Integration Center

Introducción

DB2 Universal Database (UDB) da soporte a todos los estándares clave de Internet, lo que la convierte en una base de datos ideal para utilizarla en la Web. Su velocidad de memoria interna, que facilita las búsquedas en Internet y un sistema de emparejamiento de textos complejos, se combina con las características de escalabilidad y disponibilidad de una base de datos relacional. DB2 Universal Database da soporte a WebSphere, Java y XML Extender, lo que facilita al usuario el despliegue de aplicaciones de e-business. Este documento describe las características de DB2 UDB Universal Developer's Edition Versión 7.2 (UDE) e incluye herramientas de desarrollo tales como VisualAge para Java, Professional Edition y WebSphere Studio para utilizarlas en la creación de aplicaciones Web.

Habilitación de Java

DB2 Universal Database da soporte a muchos tipos de programas Java. Proporciona soporte a controladores para aplicaciones cliente y applets escritas en Java utilizando Java Database Connectivity (JDBC). También proporciona soporte para SQL incorporado para Java (SQLj), funciones Java definidas por el usuario (UDF) y procedimientos Java almacenados.

Soporte de controladores JDBC

El controlador de aplicaciones DB2 JDBC (Tipo 2) (Figura 1 en la página 2) permite que las aplicaciones Java efectúen llamadas a DB2 a través de JDBC. Las llamadas al controlador de aplicaciones JDBC se convierten a los métodos nativos de Java. Las aplicaciones Java que utilizan este controlador deben ejecutarse en un cliente DB2, a través del cual las peticiones de JDBC fluyen al servidor DB2.

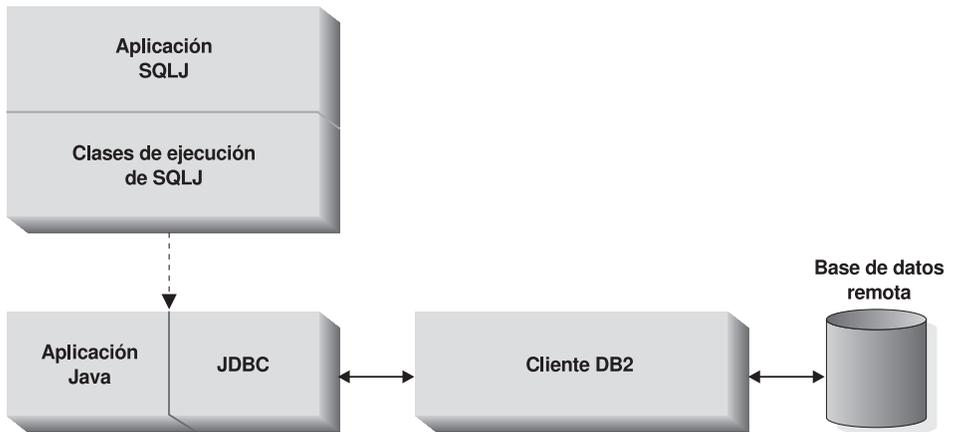


Figura 1. Implementación de una aplicación Java de DB2

El controlador de applets DB2 JDBC (Tipo 3) (Figura 2 en la página 3) se compone de un cliente JDBC y un servidor JDBC. El navegador Web carga el controlador de applets DB2 JDBC junto con la applet. Cuando la applet solicita una conexión a una base de datos DB2, el controlador de applets abre un socket TCP/IP al servidor de applets DB2 JDBC en la máquina en la que se ejecutan el servidor Web y el cliente DB2. Después de establecer una conexión, el controlador de applets envía cada una de las peticiones posteriores de acceso a la base de datos desde la applet al servidor JDBC a través de la conexión TCP/IP. A continuación, el servidor JDBC realiza las llamadas de DB2 correspondientes para realizar la tarea. Al terminar, el servidor JDBC devuelve los resultados al cliente JDBC a través de la conexión.

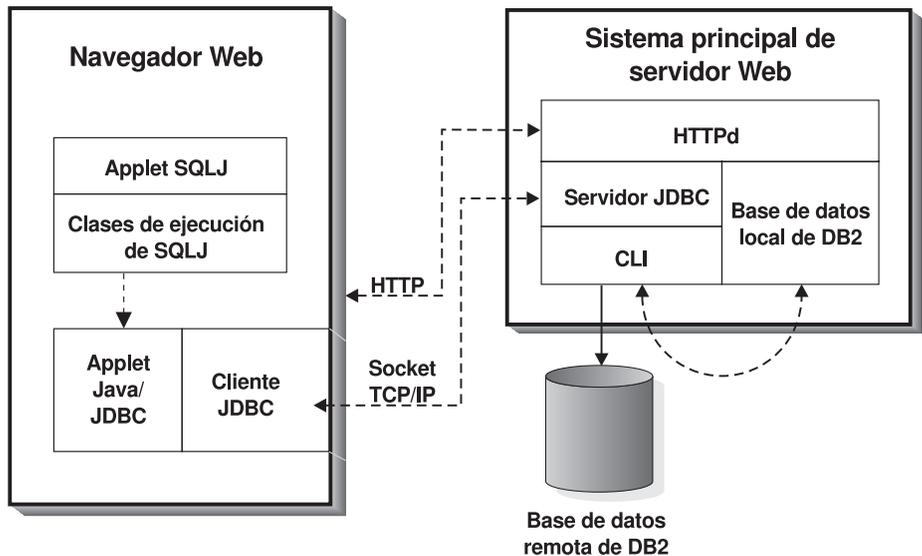


Figura 2. Implementación de applet Java de DB2

JDBC 1.0 define las API Java para acceder a bases de datos relacionales. Con la introducción de JDBC 2.0, las API se han dividido en dos partes:

API de imagen de memoria JDBC 2.0

contiene mejoras evolutivas, pero se ha mantenido pequeña y dedicada como la API JDBC 1.0 para promover la facilidad de su uso. El código escrito para la API 1.0 sigue funcionando en la API 2.0. Las clases de la API 2.0 se encuentran en el paquete `java.sql`.

API de paquete opcional JDBC 2.0

define clases específicas de funcionalidad adicional cuando los proveedores están listos para proporcionar la funcionalidad y los desarrolladores están preparados para utilizarla. IBM ha implementado estas nuevas clases e interfaces en paquetes nuevos. Puede utilizar las nuevas clases e interfaces en sus aplicaciones utilizando las siguientes sentencias de importación:

- `import javax.sql.*;`
- `import COM.ibm.db2.jdbc.*;`

El controlador DB2 JDBC 2.0 da soporte a las siguientes características de la API de paquete opcional JDBC 2.0:

- Soporte de `DataSource`
- Java Naming and Directory Interface (JNDI) para la denominación de bases de datos

- API de transacciones Java (JTA) — sólo disponible con el controlador de aplicación DB2 JDBC
- Agrupación de conexiones de bases de datos

Las interfaces de la agrupación de conexiones y JNDI están definidas en el controlador JDBC y están implementadas por WebSphere Application Server, Versión 3.5.

La agrupación de conexiones está definida como parte de la API de paquete opcional JDBC 2.0. Los objetos JNDI y DataSource se proporcionan como alternativa al uso de objetos DriverManager para acceder a servidores de datos relacionales.

El soporte de JDBC 2.0 está disponible en los sistemas operativos de 32 bits AIX, Solaris, HP-UX, Linux, Linux para OS/390 y Windows. Para aprovechar las nuevas características de JDBC 2.0, debe instalar el controlador JDBC 2.0 y el Kit de Desarrollo en Java (JDK) 1.2, ya que el controlador JDBC 1.2 es el controlador por omisión. Para instalar el controlador de JDBC 2.0, entre el mandato `usejdbc2` desde el directorio `sql11ib/java12`. Para volver al controlador JDBC 1.2, ejecute el mandato `usejdbc1` desde el directorio `sql11ib/java12`.

Para más detalles, consulte:

- *DB2 Application Development Guide, Version 7*, capítulo 21: "Programming in Java"
- <http://www.software.ibm.com/data/db2/java>

Soporte de SQLj

El soporte de DB2 SQLj permite crear y ejecutar applets y aplicaciones SQLj. Estos programas Java contienen sentencias de SQL incorporado que se precompilan y se enlazan a una base de datos DB2 UDB. Las aplicaciones SQLj utilizan el soporte de JDBC y necesitan que las clases de ejecución de SQLj autentiquen y ejecuten cualquier paquete SQL que esté enlazado a la base de datos en la fase de precompilación y enlace.

El estándar SQLj tiene tres componentes: SQLj incorporado, un conversor y un entorno de ejecución. El conversor convierte archivos SQLj para producir archivos y perfiles de fuente Java. El entorno de ejecución realiza las operaciones de SQL en JDBC y utiliza el perfil para obtener detalles acerca de las conexiones de base de datos.

Procedimientos almacenados de Java y funciones definidas por el usuario (UDF)

Puede crear y utilizar los procedimientos almacenados en Java como lo haría en otros lenguajes, con tan sólo algunas ligeras diferencias. Después de codificar el procedimiento almacenado, regístrelo en la base de datos mediante

la sentencia `CREATE PROCEDURE`. A continuación, puede llamar al procedimiento almacenado desde su aplicación. El procedimiento almacenado puede ser `FENCED` o `NOT FENCED`.

También puede crear y utilizar las UDF de Java como lo haría en otros lenguajes, sólo con algunas ligeras diferencias. Después de codificar la UDF, regístrela en la base de datos mediante la sentencia `CREATE FUNCTION`. A continuación, puede hacer referencia a la UDF en el SQL de su aplicación. La UDF puede ser `FENCED` o `NOT FENCED`.

Puede utilizar el Creador de procedimientos almacenados (SPB) de DB2 para desarrollar procedimientos almacenados de Java. Como ayuda para crear procedimientos almacenados, DB2 SPB proporciona asistentes de diseño que le guiarán para elaborar patrones básicos de diseño, le ayudarán a crear consultas SQL y calcularán qué impacto tendrá en el rendimiento si se invoca un procedimiento almacenado.

DB2 SPB se implementa con Java y todas las conexiones de base de datos se gestionan mediante JDBC. Utilizando un controlador JDBC, puede conectar con cualquier alias local de DB2 o cualquier otra base de datos DB2 UDB para los que pueda especificar un sistema principal, un puerto y un nombre de base de datos. Hay varios controladores JDBC instalados con DB2 SPB para utilizarlos en distintos sistemas operativos.

Para ejecutar las UDF y los procedimientos almacenados en el servidor, DB2 invoca la Máquina virtual Java (JVM). Antes de iniciar la base de datos, el administrador de bases de datos debe instalar y configurar el JDK adecuado en el servidor DB2. Puede optar por utilizar archivos de clases Java individuales para los procedimientos almacenados y UDF, o recoger los archivos de clases en un archivo de Archivado Java (JAR) e instalar el archivo JAR en la base de datos.

La JVM se carga en la primera llamada a una UDF o procedimiento almacenado de Java. En todos los casos, la JVM sigue cargada hasta que finaliza el proceso de incorporación. Para las UDF y los procedimientos almacenados que son `NOT FENCED`, se carga una JVM por instancia de base de datos y se ejecuta dentro del espacio de direcciones del motor de base de datos para obtener un mejor rendimiento. Para las UDF que son `FENCED`, se carga una JVM diferenciada dentro del proceso `db2udf`; de manera similar, los procedimientos almacenados que son `FENCED` cargan una JVM diferenciada dentro del proceso `db2dari`.

Desarrollo de aplicaciones Java para servidores Web

Hay varias herramientas que se proporcionan con UDE Versión 7.2 que proporcionan soporte de habilitación para la Web. VisualAge para Java, Versión 3.5, es un entorno de desarrollo integrado (IDE) que le permite crear, probar y desplegar aplicaciones Java en WebSphere Application Server y en DB2 Universal Database. WebSphere Studio es un conjunto de herramientas que reúne todos los aspectos del desarrollo de sitios Web en una interfaz común. WebSphere Application Server Standard Edition versión 3.5 proporciona un entorno potente de desarrollo para las aplicaciones e-business. Sus componentes le permiten crear y desplegar contenido Web personalizado y dinámico de manera rápida y fácil.

VisualAge para Java, Professional Edition

VisualAge para Java, Professional Edition Versión 3.5 (proporcionado con UDE Versión 7.2) contiene características y mejoras en el rendimiento que facilitan más que nunca la creación de aplicaciones e-business escalables y resistentes. Su fuerte integración con IBM WebSphere Application Server, WebSphere Studio y DB2 Universal Database reduce el tiempo de desarrollo y mejora la productividad, al tiempo que proporciona un acceso seguro y más sencillo a los datos de las empresas.

IBM Data Access JavaBeans, una característica instalable opcionalmente de VisualAge para Java, permite que los desarrolladores de aplicaciones puedan acceder fácilmente a bases de datos relacionales habilitadas por JDBC. IBM Data Access JavaBeans, que se encuentra en el paquete *com.ibm.db*, incluye clases para simplificar el acceso a las bases de datos relacionales y proporcionar las siguientes características mejoradas:

Colocación en antememoria de los resultados de las consultas

Los resultados de una consulta SQL pueden recuperarse todos a la vez y colocarse en una antememoria. La aplicación o servlet puede avanzar o retroceder por la antememoria o saltar directamente a cualquier fila de resultados de la antememoria. Compare esto con los recursos del paquete *java.sql*, en el que las filas se recuperan de la base de datos una a una, sólo en la dirección de avance, y una fila recién recuperada solapa la última fila recuperada a menos que se escriba código adicional para ampliar la funcionalidad. Para los conjuntos de resultados grandes, IBM Data Access JavaBeans proporcionan maneras de recuperar y gestionar paquetes, que son subconjuntos del conjunto entero de resultados.

Actualizaciones a través de la antememoria de resultados

La servlet puede utilizar métodos Java estándares para cambiar, añadir o suprimir filas de la antememoria de resultados. Los cambios en la antememoria pueden propagarse a la tabla relacional subyacente.

Soporte de parámetros de consulta

La consulta SQL básica se define como una serie Java, con unos parámetros que sustituyen a algunos de los valores reales. Cuando se ejecuta la consulta, IBM Data Access JavaBeans proporciona una manera de sustituir los parámetros por valores que están disponibles durante la ejecución. Por ejemplo, un usuario podría someter los valores de los parámetros en un formulario HTML.

Soporte de metadatos

Un objeto StatementMetaData contiene la consulta SQL básica. Es posible añadir niveles superiores de metadatos al objeto para ayudarle a pasar parámetros a la consulta y trabajar con los resultados devueltos. Cuando se ejecuta la consulta, los parámetros se convierten automáticamente entre tipos de datos de Java y los tipos de datos de SQL correspondientes.

VisualAge para Java, Enterprise Edition

VisualAge para Java, Enterprise Edition Versión 3.5 (no se proporciona con UDE Versión 7.2) le permite crear, probar y desplegar aplicaciones Java en un único entorno. Además de sus características de programación visual, VisualAge para Java proporciona unos asistentes que le conducirán rápidamente por numerosas tareas, incluida la creación de applets, servlets y aplicaciones. También le permite importar el código existente y exportar el código, como sea necesario, desde el sistema de archivos subyacente.

VisualAge para Java proporciona una Herramienta de SQLj que implementa el estándar SQLj, lo que le permite simplificar el acceso a las bases de datos. El componente conversor está integrado en el IDE, lo que le permite importar, convertir y editar archivos SQLj. El entorno de ejecución es una característica instalable que se añade al espacio de trabajo. Los archivos fuente de SQLj originales se mantienen en el directorio de recursos de proyectos, al igual que los perfiles. La Herramienta de SQLj de VisualAge para Java crea perfiles en el directorio de recursos de proyectos.

Gestión de permanencia

El Creador de permanencia es una característica de VisualAge para Java que genera una capa de código que implementa todas las llamadas de acceso a JDBC que son necesarias para insertar, actualizar o recuperar los datos correspondientes a un objeto de la base de datos. El Creador de permanencia es una herramienta de productividad que simplifica muchos de los retos existentes al desarrollar aplicaciones de bases de datos. Si la aplicación se ha desarrollado utilizando el modelado de objetos, puede utilizarse el Examinador de esquemas del Creador de permanencia para generar las tablas necesarias para que persistan los objetos, lo que se conoce como *correlación de arriba a abajo*. Si existe la base de datos, puede utilizarse el Examinador de esquemas para realizar la ingeniería inversa de las tablas y convertirlas en objetos, lo que se conoce como *correlación de abajo a arriba*. La tercera técnica

se denomina *de fuera a adentro* o “correlación media”, ya que se mantiene cada modelo y se realizan cambios adaptativos en el modelo de objeto o en el modelo relacional. La utilización del Creador de permanencia de VisualAge para Java puede mejorar la eficacia de los programadores y separar el modelo de datos y el modelo de objeto. Los servidores WebSphere Application Server pueden proporcionar una agrupación de conexiones para las conexiones de bases de datos DB2 UDB, mientras que el código de creador de permanencia puede generarse para utilizar la agrupación de conexiones en lugar de conectarse directamente a la base de datos. El Creador de permanencia se utiliza principalmente en el desarrollo de los EJB y también puede proporcionar enlaces entre asociaciones de objetos y transacciones mediante las limitaciones y transacciones equivalentes en DB2.

Depuración

VisualAge para Java proporciona diversas herramientas para desarrollar Páginas JavaServer (JSP). El Ejecutor de servlets le permite iniciar un servidor Web, abrir el navegador Web y ejecutar una servlet. El Supervisor de ejecución de JSP le permite supervisar la ejecución del fuente JSP, las servlets generadas y el fuente HTML a medida que se genera. VisualAge para Java también le permite definir puntos de interrupción en el código de la servlet, actualizar dinámicamente la servlet en los puntos de interrupción, y seguir ejecutando la servlet con los cambios incorporados.

IBM WebSphere Test Environment es una característica instalable opcional de VisualAge para Java. Esta característica le permite ejecutar servlets y Enterprise JavaBeans (EJB) en el entorno de VisualAge para Java antes de desplegar los servidores WebSphere Application Servers. El entorno proporciona un motor de servlets y los servicios necesarios para aplicaciones EJB. Dado que este entorno de prueba se proporciona en el IDE, pueden definirse puntos de interrupción en servlets o JSP y puede invocarse una aplicación cliente o un navegador para probar todos los aspectos de los programas Java correspondientes al servidor. Puede utilizarse el depurador remoto para definir puntos de interrupción en los EJB que se ejecuten en un Servidor EJB como WebSphere Application Server.

Mediante el Creador de procedimientos almacenados (SPB) de DB2 y el depurador remoto de VisualAge, puede depurar remotamente un procedimiento almacenado instalado en un servidor DB2. Para depurar un procedimiento almacenado, ejecútelo en la modalidad de depuración. No es necesario que depure los procedimientos almacenados desde el interior de un programa de aplicación. Puede probar el procedimiento almacenado de forma separada del programa de aplicación que llama.

Mediante el cuaderno Propiedades de depuración en SPB, puede cambiar, añadir o eliminar registros de depuración en la tabla de depuración de procedimientos almacenados. Si es un administrador de bases de datos y ha

creado el procedimiento almacenado seleccionado, puede otorgar autorización sobre bases de datos a otros usuarios para que depuren el procedimiento almacenado.

WebSphere Studio

WebSphere Studio Versión 3.5 es un conjunto de herramientas que reúne todos los aspectos del desarrollo de sitios Web en una interfaz común. WebSphere Studio facilita más que nunca el crear, ensamblar, publicar y mantener aplicaciones dinámicas de Web en equipo. Los autores de contenido, artistas gráficos, programadores y Webmasters pueden trabajar en los mismos proyectos mientras cada uno de ellos tiene acceso a los archivos que necesita. Studio se compone del Banco de pruebas, el Diseñador de páginas y el Depurador remoto, además de los asistentes, y se suministra con copias de prueba de productos complementarios de desarrollo en Web, como Macromedia Flash, Fireworks, Freehand y Director. WebSphere Studio le permite realizar todo lo que necesita para crear sitios Web interactivos que den soporte a las funciones avanzadas de negocios, incluidas las siguientes:

- Utilizar los asistentes de Studio para crear beans Java, consultas de bases de datos y servlets Java.
- Agrupar los archivos de los sitios Web en proyectos y carpetas. Los filtros y las posibilidades de búsqueda global le permiten encontrar los archivos que necesita. Los proyectos pueden exportarse a VisualAge para Java y se despliegan en WebSphere Application Server.
- Mantener los archivos de forma individual o en un sistema de control de versiones compartido.
- Editar y actualizar los archivos con sus herramientas preferidas. Al abrir un archivo Studio, puede ejecutar rápidamente su selección por omisión o elegir una de las herramientas alternativas.
- Evaluar rápidamente las relaciones de archivos y buscar enlaces rotos.
- Publicar el sitio Web durante cualquier fase del desarrollo en cualquiera de los servidores WebSphere Application Servers. Vaya directamente desde el desarrollo de sitios a la publicación de sitios dentro del Banco de pruebas de Studio.

El Banco de pruebas de Studio le permite gestionar y mantener los archivos y aplicaciones de sitios Web y proporciona las siguientes posibilidades y características:

- Una visualización gráfica de las relaciones de enlaces entre los archivos de un proyecto.
- La actualización automática de los enlaces siempre que se modifiquen o muevan los archivos.
- La posibilidad de establecer fases en el ciclo de producción de sitios Web y publicar diversas fases en distintos (y diversos) servidores.

- Un asistente de importación que simplifica la transferencia del contenido existente de los sitios a un proyecto de Studio.
- Una manera rápida de archivar sitios o subsitios Web en un solo archivo.
- La posibilidad de integrar fácilmente herramientas de otras compañías en el entorno del Banco de pruebas.
- Un entorno de equipo mejorado con una vista común del trabajo en marcha, mediante la integración de software muy utilizado de gestión de control del fuente como IBM VisualAge TeamConnection.

El Diseñador de páginas de Studio proporciona un entorno de diseño visual que le permite crear JSP, servlets Java y otras herramientas Web basadas en Java. Por ejemplo, puede utilizar el entorno visual para arrastrar y soltar JavaBeans en aplicaciones JSP. El Diseñador de páginas de Studio también puede utilizarse para crear páginas DHTML (Lenguaje de marcación de hipertexto dinámico) y HTML e incluye la posibilidad de editar y conmutar fácilmente entre el fuente HTML o DHTML y la vista de navegador. El Depurador remoto de Studio proporciona una depuración a nivel de fuente de los archivos JSP y servlets Java en el entorno de Studio.

WebSphere Application Server

WebSphere Application Server combina la portabilidad de las aplicaciones de negocios de servidor con el rendimiento y la capacidad de gestión de las tecnologías Java para ofrecer una plataforma amplia para diseñar aplicaciones Web basadas en Java. Permite realizar potentes interacciones con bases de datos y sistemas de transacciones de las empresas.

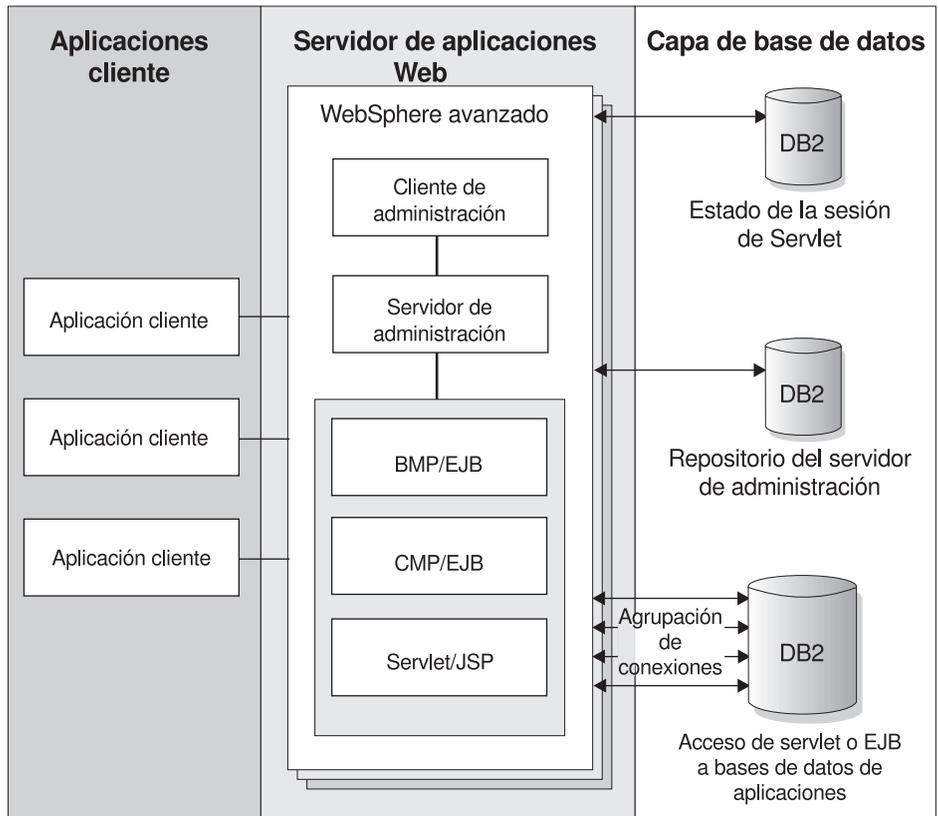


Figura 3. WebSphere Application Server

Los servidores de aplicaciones amplían las posibilidades de un servidor Web para gestionar peticiones de aplicaciones. Dada una aplicación compuesta de páginas HTML, servlets y beans de empresa (enterprise beans), el servidor de aplicaciones hace posible el siguiente intercambio:

1. Un usuario que utiliza un navegador Web en la Internet pública visita el sitio Web de una compañía. El usuario solicita el uso de una aplicación que proporciona acceso a datos en una base de datos.
2. La petición de usuario fluye al servidor Web.
3. El servidor Web determina que la petición implica una aplicación que contiene servlets y beans de empresa. Reenvía la petición a IBM WebSphere Application Server.
4. El producto IBM WebSphere Application Server reenvía la petición a uno de sus servidores de aplicaciones en los que se ejecuta la aplicación.
5. La aplicación procesa la petición del usuario.
6. El servidor de aplicaciones colabora con el servidor Web para devolver los resultados al navegador Web del usuario.

WebSphere Application Server proporciona un motor de servlets que implementa la API de Java Servlet 2.1. Incluye sus propios paquetes que amplían y añaden a la API de Java Servlet y JDK de nivelación 1.2 a través de todos los sistemas operativos soportados. Las extensiones y las adiciones facilitan gestionar el estado de la sesión, crear páginas Web personalizadas, generar mejores informes de errores de servlets y acceder a las bases de datos.

El servidor de aplicaciones da soporte a las JSP, que son un enfoque potente del contenido de páginas Web dinámicas. Una de las ventajas de las JSP es que le permiten separar de forma eficaz la codificación HTML de la lógica de negocios en las páginas Web. Las extensiones de IBM a la especificación de JSP incluyen códigos JSP semejantes a HTML que facilitan a los autores de HTML añadir la potencia de Java a sus páginas Web.

Agrupación de conexiones

Con WebSphere Application Server se proporciona soporte de agrupación de conexiones. La agrupación de conexiones le permite controlar y reducir los recursos utilizados por las aplicaciones basadas en la Web. Las aplicaciones basadas en la Web que acceden a los servidores de datos tienen una actividad general más elevada y menos previsible que las aplicaciones que no son de la Web, porque los usuarios de la Web se conectan y desconectan con frecuencia. A menudo, se gastan más recursos al conectarse y desconectarse que durante las interacciones propiamente dichas.

La agrupación de conexiones extiende la actividad general de conexión entre varias peticiones de usuario, al establecer una agrupación de conexiones que las servlets pueden utilizar. Después de haber gastado los recursos iniciales para producir las conexiones de la agrupación, la actividad general adicional de conexión y desconexión es insignificante porque las conexiones existentes se vuelven a utilizar repetidamente.

La agrupación de conexiones, tal como WebSphere Application Server la proporciona, especifica cómo una servlet puede acceder a los servidores de datos relacionales mediante las nuevas API de extensión estándar de JDBC 2.0. Este es el nuevo modelo. Las servlets que se ejecutan en WebSphere Application Server Versión 3.x pueden codificarse para utilizar de forma eficaz los recursos de conexión a los servidores de datos.

Ventajas de la agrupación de conexiones: Cada vez que un recurso intenta acceder a una base de datos, debe conectarse a ella. Una conexión a base de datos tiene una actividad general – necesita unos recursos para crear la conexión, mantenerla y liberarla cuando ya no es necesaria. La actividad general es especialmente elevada en el caso de las aplicaciones basadas en la Web, debido a la frecuencia de las conexiones y desconexiones. Las interacciones de los usuarios suelen ser cortas, debido a la naturaleza transitoria de la navegación por la Web. Dado que las peticiones de Internet

pueden llegar prácticamente de cualquier parte, los volúmenes de utilización pueden ser grandes y difíciles de prever. Para resolver este problema, WebSphere Application Server establece una agrupación de conexiones a bases de datos compartida por las aplicaciones en los servidores de aplicaciones.

La agrupación de conexiones permite que el administrador controle y reduzca los recursos utilizados por las aplicaciones basadas en la Web, extendiendo la actividad general de conexión entre varias peticiones de usuarios. Además, la agrupación de conexiones puede mejorar los tiempos de respuesta de las aplicaciones basadas en la Web.

Cuando un usuario realiza una petición a un recurso a través de la Web, el recurso accede a una fuente de datos. Dado que la fuente de datos localiza y utiliza una conexión existente de la agrupación, la petición de usuario no produce la actividad general de crear una nueva conexión.

Cada conexión va asociada a una petición de usuario determinada. Cuando se satisface la petición y se devuelve la respuesta al usuario, la fuente de datos devuelve la conexión a la agrupación de conexiones para que pueda volver a utilizarse. Una vez más, se evita la actividad general de una desconexión.

Cada petición de usuario utiliza una fracción del coste de una conexión o desconexión. Después de haber utilizado los recursos iniciales para producir las conexiones de la agrupación, la actividad general adicional es insignificante porque las conexiones existentes se vuelven a utilizar.

Cómo gestiona WebSphere Application Server las agrupaciones de conexiones: WebSphere Application Server establece y mantiene agrupaciones de conexiones según como especifique el administrador. Una vez establecidas las conexiones, se distribuyen en respuesta a las peticiones de los usuarios y, a continuación, realizan operaciones de mantenimiento para conservar un equilibrio entre las conexiones disponibles y la demanda de conexiones. Esto asegura que haya una conexión existente disponible cuando una servlet o un servidor de aplicaciones necesite una.

Por ejemplo, la agrupación de conexiones identifica de forma periódica las conexiones desocupadas o abandonadas. Termina las conexiones desocupadas y devuelve las conexiones abandonadas a la agrupación de conexiones. Esto quiere decir que hay menos conexiones disponibles (y se utilizan menos recursos) cuando la demanda de conexiones es baja. Una conexión desocupada es aquella que no se ha utilizado durante el período de tiempo especificado en la propiedad Tiempo de espera de desocupado de la fuente de datos. Una conexión está abandonada cuando la servlet o el servidor de aplicaciones que es propietario de la misma ha terminado o no responde.

WebSphere Standard Edition

WebSphere Application Server, Standard Edition Versión 3.5 (que se proporciona con UDE Versión 7.2) es un componente de WebSphere Studio. InstantDB es la base de datos por omisión utilizada para el repositorio de WebSphere Standard Edition. No obstante, puede configurarse DB2 Universal Database para que sea el repositorio de servidor administrativo de WebSphere. Puede ejecutar el servidor DB2 en la misma máquina que WebSphere Application Server o en un servidor Web distinto.

WebSphere Advanced Edition

WebSphere Application Server Advanced Edition (que no se proporciona con UDE Versión 7.2) se construye sobre el Standard Application Server, ya que necesita una base de datos para su funcionamiento. DB2 Universal Database se proporciona con WebSphere Application Server Advanced Edition Versión 3.5 para que se utilice como repositorio de servidor administrativo. Introduce posibilidades de servidor para las aplicaciones creadas según la Especificación EJB de Sun Microsystems, que proporciona soporte para la integración de aplicaciones Web en sistemas de negocios que no sean de la Web.

El servidor EJB es el servidor de aplicaciones de la arquitectura en tres niveles de WebSphere Application Server. Conecta el nivel de cliente de servlets, applets, aplicaciones y JSP Java con el nivel de gestión de recursos, la fuente de datos.

Hay dos tipos de beans de empresa: los beans de *sesión* y los beans de *entidad*. Los beans de sesión contienen los datos temporales asociados a un cliente determinado. Los beans de entidad contienen los datos permanentes que están almacenados en una fuente de datos. El servicio de permanencia asegura que los datos asociados a los beans de entidad estén correctamente sincronizados con sus datos correspondientes en la fuente de datos. Para realizar esta tarea, el servicio de permanencia colabora con el servicio de transacciones para insertar, actualizar, extraer y eliminar datos de la fuente de datos en los momentos adecuados.

Hay dos tipos de beans de entidad: los de permanencia gestionados por el contenedor (CMP) y los de permanencia gestionados por el bean (BMP). En los beans de entidad con CMP, el servicio de permanencia maneja casi todas las tareas necesarias para gestionar los datos permanentes. Los beans de entidad CMP pueden implementarse mediante DB2 UDB (incluido DB2 para OS/390) como almacén de datos permanente. En los beans de entidad con BMP, el propio bean maneja casi todas las tareas necesarias para gestionar los datos permanentes.

Extensiones para el desarrollo de aplicaciones

XML Extender

El Lenguaje de marcación extensible (XML) es la técnica estándar aceptada para el intercambio de datos entre las aplicaciones. Un documento XML es un documento con códigos que se compone de datos de tipo carácter y códigos de marcación. El autor del documento puede definir los códigos de marcación. Se utiliza una Definición de tipo de documento (DTD) para declarar las definiciones y limitaciones de la marcación. DB2 XML Extender proporciona un mecanismo para que los programas manipulen los datos XML mediante extensiones de SQL.

Normalmente, los documentos XML se almacenan como archivos individuales en un sistema de archivos. DB2 XML Extender proporciona un almacenamiento y un entorno de manipulación alternativos. Los documentos XML pueden almacenarse en una sola columna o como una colección, utilizando un conjunto de columnas.

DB2 XML Extender introduce tres tipos de datos nuevos: XMLVARCHAR, XMLCLOB y XMLFILE. Extender proporciona unas UDF para almacenar, extraer y actualizar los documentos XML almacenados en una sola columna. Pueden realizarse búsquedas en todo el documento XML o según componentes estructurales utilizando la vía de acceso de ubicación, que utiliza un subconjunto de la sintaxis abreviada definida por el Lenguaje de vía de acceso de XML (XPath). Pueden utilizarse tablas complementarias para mejorar el rendimiento de las búsquedas para los elementos o atributos que se consultan con frecuencia.

Para facilitar el almacenamiento de documentos XML como un conjunto de columnas, DB2 XML Extender proporciona una herramienta de administración para ayudar al diseñador a realizar la correlación de XML con bases de datos relacionales. Se utiliza la Definición de acceso a documentos (DAD) para mantener los datos estructurales y de correlación correspondientes a los documentos XML. La DAD se define y se almacena como un documento XML, lo que simplifica su manipulación y comprensión. Hay procedimientos almacenados nuevos disponibles para componer o descomponer el documento.

Habilitación de MQSeries

Se proporciona un conjunto de funciones de MQSeries con DB2 Universal Database Versión 7.2 para permitir que las aplicaciones DB2 UDB interactúen con operaciones de mensajería asíncronas. Esto quiere decir que el soporte de MQSeries está disponible para las aplicaciones escritas en cualquier lenguaje de programación soportado por DB2 UDB. Por razones de simplicidad, todos los ejemplos que se muestran en esta sección son sentencias SQL. Una aplicación WebSphere puede utilizar estas sentencias SQL de MQSeries.

Estilos de mensajería

MQSeries no exige que los mensajes que transporta se adhieran a una estructura determinada. Los mensajes XML suelen tener una estructura de mensaje autodescriptiva. Los mensajes también pueden estar desestructurados, lo que requiere que el código del usuario analice o construya el contenido del mensaje. Los mensajes de este tipo suelen estar semiestructurados, es decir, utilizan posiciones de bytes o delimitadores fijos para separar los campos dentro de un mensaje.

MQSeries da soporte a tres modelos de mensajes: datagramas, publicación/suscripción (p/s) y petición/respuesta (p/r). Los mensajes enviados como datagramas se envían a un único destino sin esperar respuesta. En el modelo p/s, uno o más remitentes envían un mensaje a un servicio de publicación que distribuye el mensaje a uno o más suscriptores interesados. El modelo de petición/respuesta es similar al datagrama, pero el remitente espera recibir una respuesta. Las funciones de DB2 MQSeries suministradas dan soporte a los tres modelos de mensajes.

MQSeries se utiliza de muchas maneras distintas. Los datagramas más sencillos se intercambian para coordinar varias aplicaciones, intercambiar información, solicitar servicios y notificar sucesos interesantes. El estilo de publicación/suscripción se utiliza con mayor frecuencia para distribuir información en tiempo real y a tiempo. El estilo de petición/respuesta se utiliza por lo general como una forma sencilla de llamada a procedimiento remoto (RPC) pseudosíncrona. Pueden elaborarse modelos más complejos combinando estos estilos básicos.

DB2 UDB Versión 7.2 proporciona un nuevo asistente de MQSeries Assist. Este asistente crea una función de tabla que lee de una cola de MQSeries utilizando las UDF de MQSeries, que también son nuevas en DB2 UDB Versión 7.2. El asistente puede tratar cada mensaje de MQSeries como una serie delimitada o una serie de columna de longitud fija. La función de tabla creada analiza la serie según las especificaciones del usuario y devuelve cada mensaje de MQSeries como una fila de la función de tabla. El asistente también le permite crear una vista encima de la función de tabla y tener una vista previa de un mensaje de MQSeries y del resultado de la función de tabla.

Infraestructura de DB2 / MQ

En una configuración básica, como se muestra en la Figura 4 en la página 17, un servidor MQSeries está ubicado en la máquina servidor de bases de datos junto con DB2 Universal Database. Las funciones de MQSeries están disponibles desde un servidor DB2 y proporcionan acceso a otras aplicaciones de MQSeries. Varios clientes DB2 pueden acceder simultáneamente a las funciones de MQSeries a través de la base de datos. Las operaciones de MQSeries permiten que las aplicaciones DB2 se comuniquen de forma

asíncrona con otras aplicaciones de MQSeries. Por ejemplo, las nuevas funciones proporcionan una manera sencilla de que una aplicación DB2 publique sucesos de base de datos a aplicaciones remotas de MQSeries, inicie un flujo de trabajo a través del producto opcional MQSeries Workflow, o comunicarse con un paquete de aplicaciones existente mediante el producto opcional MQSeries Integrator.

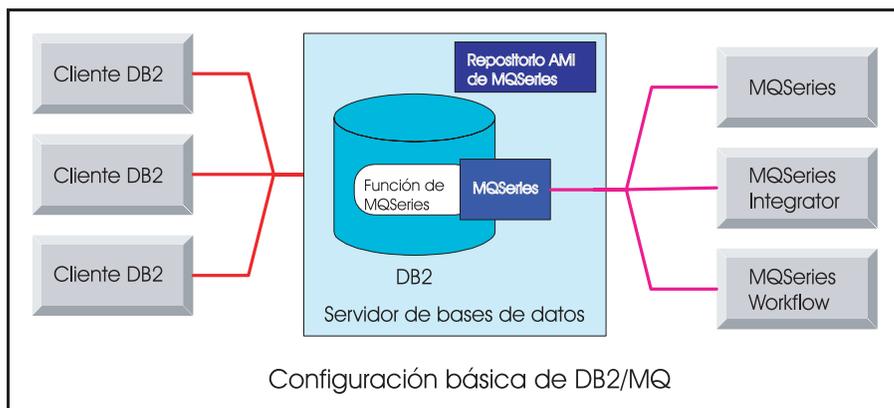


Figura 4. Configuración básica de DB2/MQ

Cuando se instala el soporte de MQSeries como parte de DB2 Universal Database, un script de configuración establece automáticamente una configuración por omisión que las aplicaciones cliente pueden utilizar sin realizar ninguna acción administrativa adicional. La configuración por omisión permite que los programadores de aplicaciones comiencen a trabajar con rapidez y, además, proporciona una interfaz de desarrollo más sencilla.

Ejemplos de mensajes

La siguiente sentencia SQL envía un mensaje básico utilizando la configuración por omisión:

```
values MQSEND('mensaje sencillo')
```

Esto enviará el mensaje mensaje sencillo al gestor de colas de MQSeries y a la cola especificados por la configuración por omisión.

Las funciones de MQSeries proporcionadas con DB2 Universal Database se basan en la Interfaz de mensajería de aplicaciones (AMI). AMI da soporte al uso de un archivo de configuración externo, denominado Repositorio de AMI, para almacenar la información de configuración. La configuración por omisión incluye un Repositorio AMI de MQSeries configurado para su utilización con DB2 UDB. Para obtener más información, consulte: <http://www-3.ibm.com/software/ts/mqseries/api/>.

Los *puntos de servicio* y las *políticas* son dos conceptos clave en la AMI de MQSeries, transferidos a las funciones MQSeries de DB2 UDB. Un punto de servicio es un punto final lógico desde el que se puede enviar o recibir un mensaje. En el repositorio de AMI, cada punto de servicio se define con un nombre de cola y un gestor de colas de MQSeries. Las políticas definen las opciones de calidad de servicio que deben utilizarse para una operación de mensajería determinada. Las cualidades de servicio claves son la prioridad y la permanencia del mensaje. Se proporcionan unos puntos de servicio y unas definiciones de políticas por omisión que los desarrolladores pueden utilizar para simplificar aún más sus aplicaciones. El ejemplo anterior puede reescribirse de la siguiente manera para especificar de manera explícita el punto de servicio y el nombre de política por omisión:

```
values MQSEND('DB2.DEFAULT.SERVICE',  
'DB2.DEFAULT.POLICY', 'mensaje sencillo')
```

Una o más aplicaciones pueden dar servicio a las colas desde el servidor donde residen. En muchas configuraciones, se definirán varias colas para dar soporte a distintas aplicaciones y finalidades. Por esta razón, suele ser importante definir puntos de servicio diferentes al crear peticiones MQSeries. Esto se demuestra en el ejemplo siguiente.

```
values MQSEND('ODS_Input', 'mensaje sencillo')
```

Observe que, en este ejemplo, no se especifica la política, por lo que se utilizará la política por omisión.

Para más información:

<http://www.software.ibm.com/data/integration/MQSeries>

Net.Data

Net.Data es una aplicación que conecta aplicaciones Web a DB2 Universal Database. Ahora proporciona soporte de XML, lo que permite generar códigos XML como salida de la macro Net.Data del usuario, en vez de tener que entrar los códigos manualmente. También puede especificar una hoja de estilo XML (XSL) para que se utilice para formatear y visualizar la salida generada.

Net Search Extender

Net Search Extender (que no se suministra con UDE Versión 7.2) utiliza una técnica de indexación, conocida como índice n-gram, para proporcionar un nuevo extensor de búsqueda de texto a alta velocidad. Este extensor tiene numerosos usos en el área de las aplicaciones Web, ya que suelen ser los usuarios finales los que consultan los campos de texto. El encontrar documentos relevantes basados en índices de campos de texto puede mejorar la satisfacción de los usuarios de la Web. Es posible indexar cualquier columna basada en CHAR, VARCHAR o LONG VARCHAR utilizando un índice n-gram. Una vez creado y activado el índice, es posible realizar

búsquedas utilizando un nuevo procedimiento almacenado. Los índices activos se almacenan en la memoria compartida para optimizar el rendimiento de las búsquedas.

Spatial Extender

Spatial Extender (que no se suministra con UDE Versión 7.2) permite que los usuarios integren datos espaciales en sus consultas. Da soporte a los tipos espaciales para modelar entidades del mundo real, tales como la ubicación de los clientes, los límites de los parques y las rutas de las líneas de cables.

Guía de Aprendizaje de DB2 Integrated Web Services

Una nueva Guía de Aprendizaje de DB2, denominada eVideoCentral, está disponible para bajarla de <http://www.ibm.com/software/data/developer/samples/evideo>. La Guía de Aprendizaje DB2 eVideoCentral presenta un ejemplo de solución para una compañía que proporciona servicios empresariales a otra compañía a través de la Web. Este concepto se conoce como aplicación de negocio a negocio (Business-to-Business). La Guía de Aprendizaje eVideoCentral es una demostración de la integración de DB2, WebSphere y MQSeries. Se han utilizado muchas tecnologías para diseñar y crear eVideoCentral, incluyendo las siguientes: IBM WebSphere Application Server, VisualAge for Java y la herramienta de interacción DB2 XML Extender. Se puede acceder a un conjunto de servicios centralizados para vídeo-clubs minoristas individuales a través del Protocolo de acceso simple a objetos (SOAP) (consulte <http://www.w3.org/2000/xp/>).

SOAP es un protocolo ligero para intercambiar información en un entorno distribuido y descentralizado. Es un protocolo basado en XML que se compone de tres partes:

- Un sobre que define un marco para describir qué hay en un mensaje y cómo procesarlo
- Un conjunto de reglas de codificación para expresar instancias de tipos de datos definidos por una aplicación
- Un convenio para representar llamadas a procedimientos remotos y sus respuestas

Los servicios SOAP pueden proporcionar un mecanismo para que una compañía acceda a los servicios DB2 desde otra compañía.

La Guía de Aprendizaje de eVideoCentral es una demostración del diseño, desarrollo e implementación del repositorio de datos sencillos (inserción/modificación) y servicios de consulta. Los servicios de consulta utilizan el DB2 XML Extender existente. La primera versión de eVideoCentral se proporciona a través de la Web e incluye: servlets Java, JSP (Páginas de servidor Java), un esquema para los objetos de base de datos DB2 y la documentación complementaria de la Guía de Aprendizaje.

Diseño de aplicaciones utilizando UML (Unified Modeling Language)

Ahora es posible acceder a los modelos de datos DB2 UDB desde el interior de un conocido producto de modelado UML, Rational Rose de Rational Software. Este programa puede extraer esquemas existentes a un modelo lógico, o generar esquemas directamente a partir del modelo. Hay muchas ventajas en utilizar un lenguaje de diseño común (como UML) para modelos de objeto y modelos de datos durante el proceso de desarrollo de aplicaciones.

Referencia: *Database Design for Smarties: Using UML for Data Modeling*; Robert J. Muller; ISBN 1558605150

Resumen

DB2 UDB Universal Developer's Edition proporciona todas las herramientas necesarias para crear y desplegar aplicaciones rápidamente. El paquete incluye un entorno de desarrollo integrado totalmente funcional, un servidor de aplicaciones Web escalable y características de DB2 UDB, tales como XML Extender. DB2 Universal Database es una base de datos escalable de capacidad industrial que será la base para la gestión de datos en su e-business.

Información adicional

Para obtener información adicional, consulte los siguientes sitios Web:

Recursos de DB2 Universal Database:

- <http://www.software.ibm.com/data/developer>
- <http://www.software.ibm.com/data/db2/java>

Dominio de desarrolladores de Visual Age:

- <http://www.software.ibm.com/vadd>

Dominio de desarrolladores de WebSphere:

- <http://www.software.ibm.com/webservers/appserv>
- <http://www.ibm.com/websphere/developer>

Acceso a procedimientos almacenados de DB2 desde EJB:

- <http://www.software.ibm.com/developer/library/j-spejb/?dwzone=java>

Avisos

Es posible que IBM no comercialice en todos los países algunos productos, servicios o características descritos en este manual. Consulte al representante local de IBM para obtener información sobre los productos y servicios que actualmente pueden adquirirse en su zona geográfica. Cualquier referencia a un producto, programa o servicio de IBM no pretende afirmar ni implicar que sólo se puede utilizar dicho producto, programa o servicio de IBM. En su lugar se puede utilizar cualquier producto, programa o servicio funcionalmente equivalente que no infrinja ninguno de los derechos de propiedad intelectual de IBM. Sin embargo, es responsabilidad del usuario evaluar y verificar el funcionamiento de cualquier producto, programa o servicio que no sea de IBM.

IBM puede tener patentes o solicitudes de patentes en tramitación que afecten al tema tratado en este documento. La posesión de este documento no confiere ninguna licencia sobre dichas patentes. Puede realizar consultas sobre licencias escribiendo a:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
E.E.U.U.

En el caso de consultas sobre licencias referentes a información de doble byte (DBCS), consulte al Departamento de Propiedad Intelectual de IBM en su país o envíe consultas por escrito a:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japón

El párrafo siguiente no es aplicable al Reino Unido ni a ningún país en el que tales disposiciones sean incompatibles con la legislación local:
INTERNATIONAL BUSINESS MACHINES CORPORATION PROPORCIONA ESTA PUBLICACIÓN "TAL CUAL", SIN GARANTÍA DE NINGUNA CLASE, NI EXPLÍCITA NI IMPLÍCITA, INCLUIDAS, PERO SIN LIMITARSE A ELLAS, LAS GARANTÍAS IMPLÍCITAS DE NO VULNERACIÓN DE DERECHOS, COMERCIALIZABILIDAD O IDONEIDAD PARA UN FIN DETERMINADO. Algunos estados no permiten la exclusión de garantías expresas o implícitas en determinadas transacciones, por lo que es posible que esta declaración no sea aplicable en su caso.

Esta publicación puede contener inexactitudes técnicas o errores tipográficos. Periódicamente se efectúan cambios en la información aquí contenida; dichos cambios se incorporarán a las nuevas ediciones de la publicación. IBM puede efectuar, en cualquier momento y sin previo aviso, mejoras y/o cambios en los productos y/o programas descritos en esta publicación.

Las referencias hechas en esta publicación a sitios Web que no son de IBM se proporcionan sólo para la comodidad del usuario y no constituyen un aval de esos sitios Web. La información contenida en esos sitios Web no forma parte de la información del presente producto IBM y el usuario es responsable de la utilización de esos sitios Web.

IBM puede utilizar o distribuir cualquier información que se le facilite de la manera que considere adecuada, sin contraer por ello ninguna obligación con el remitente.

Los licenciatarios de este programa que deseen obtener información sobre él con el fin de habilitar: (i) el intercambio de información entre programas creados de forma independiente y otros programas (incluido este) y (ii) el uso mutuo de la información intercambiada, deben ponerse en contacto con:

IBM Canada Limited
Office of the Lab Director
1150 Eglinton Ave. East
North York, Ontario
M3C 1H7
CANADÁ

Dicha información puede estar disponible, sujeta a los términos y condiciones apropiados, incluido en algunos casos, el pago de una tarifa.

El programa bajo licencia descrito en este manual y todo el material bajo licencia asociado a él, los proporciona IBM según los términos del Convenio del Cliente IBM, el Convenio Internacional de Licencia de Programas de IBM o cualquier convenio equivalente entre el usuario e IBM.

Los datos de rendimiento contenidos en este documento se obtuvieron en un entorno controlado. Por tanto, los resultados obtenidos en otros entornos operativos pueden variar significativamente. Algunas mediciones pueden haberse hecho en sistemas experimentales y no es seguro que estas mediciones sean las mismas en los sistemas disponibles comercialmente. Además, algunas mediciones pueden haberse calculado mediante extrapolación. Los resultados reales pueden variar. Los usuarios del presente manual deben verificar los datos aplicables para su entorno específico.

La información referente a productos que no son de IBM se ha obtenido de los proveedores de esos productos, de sus anuncios publicados o de otras

fuentes disponibles públicamente. IBM no ha probado esos productos y no puede confirmar la exactitud del rendimiento, la compatibilidad ni cualquier otra afirmación referente a productos que no son de IBM. Las preguntas sobre las prestaciones de productos que no son de IBM deben dirigirse a los proveedores de esos productos.

Todas las declaraciones de intenciones de IBM están sujetas a cambio o cancelación sin previo aviso, y sólo representan objetivos.

Esta publicación puede contener ejemplos de datos e informes que se utilizan en operaciones comerciales diarias. Para ilustrarlos de la forma más completa posible, los ejemplos incluyen nombres de personas, empresas, marcas y productos. Todos estos nombres son ficticios y cualquier similitud con nombres y direcciones utilizados por una empresa real es totalmente fortuita.

LICENCIA DE COPYRIGHT:

Este manual puede contener programas de aplicaciones de ejemplo escritos en lenguaje fuente, que muestran técnicas de programación en diversas plataformas operativas. Puede copiar, modificar y distribuir estos programas de ejemplo de la forma que desee, sin pago alguno a IBM, con la intención de desarrollar, utilizar, comercializar o distribuir programas de aplicaciones de acuerdo con la interfaz de programación de aplicaciones correspondiente a la plataforma operativa para la que están escritos los programas de ejemplo. Estos ejemplos no se han probado exhaustivamente bajo todas las condiciones. Por tanto, IBM no puede asegurar ni implicar la fiabilidad, utilidad o función de estos programas.

Cada copia o porción de estos programas de ejemplo o cualquier trabajo derivado debe incluir una nota de copyright como la siguiente:

© (nombre de la empresa) (año). Partes de este código derivan de programas de ejemplo de IBM Corp. © Copyright IBM Corp. _especifique el año o años_. Reservados todos los derechos.

Marcas registradas

Los términos siguientes, que pueden estar indicados por un asterisco (*), son marcas registradas de International Business Machines Corporation en los Estados Unidos y/o en otros países.

ACF/VTAM	IBM
AISPO	IMS
AIX	IMS/ESA
AIX/6000	LAN DistanceMVS
AIXwindows	MVS/ESA
AnyNet	MVS/XA
APPN	Net.Data
AS/400	OS/2
BookManager	OS/390
CICS	OS/400
C Set++	PowerPC
C/370	QBIC
DATABASE 2	QMF
DataHub	RACF
DataJoiner	RISC System/6000
DataPropagator	RS/6000
DataRefresher	S/370
DB2	SP
DB2 Connect	SQL/DS
DB2 Extenders	SQL/400
DB2 OLAP Server	System/370
DB2 Universal Database	System/390
Distributed Relational Database Architecture	SystemView
DRDA	VisualAge
eNetwork	VM/ESA
Extended Services	VSE/ESA
FFST	VTAM
First Failure Support Technology	WebExplorer
	WIN-OS/2

Los términos siguientes son marcas registradas de otras empresas:

Microsoft, Windows y Windows NT son marcas registradas de Microsoft Corporation.

Java, y las marcas registradas y logotipos basados en Java y Solaris, son marcas registradas de Sun Microsystems, Inc. en los Estados Unidos y/o en otros países.

Tivoli y NetView son marcas registradas de Tivoli Systems Inc. en los Estados Unidos y/o en otros países.

UNIX es una marca registrada en los Estados Unidos y/o en otros países bajo licencia exclusiva de X/Open Company Limited.

Otros nombres de empresas, productos o servicios, que pueden estar indicados por un doble asterisco (**), pueden ser marcas registradas o marcas de servicio de otras empresas.