

IBM DB2 Universal Database



# Entwickeln von Enterprise-Java- Anwendungen mit DB2 UDB Version 7.2

Vor Verwendung dieser Informationen und des darin beschriebenen Produkts sollten die allgemeinen Informationen unter „Bemerkungen“ auf Seite 25 gelesen werden.

- Die IBM Homepage finden Sie im Internet unter: **ibm.com**
- IBM und das IBM Logo sind eingetragene Marken der International Business Machines Corporation.
- Das e-business Symbol ist eine Marke der International Business Machines Corporation
- Infoprint ist eine eingetragene Marke der IBM.
- ActionMedia, LANDesk, MMX, Pentium und ProShare sind Marken der Intel Corporation in den USA und/oder anderen Ländern.
- C-bus ist eine Marke der Corollary, Inc. in den USA und/oder anderen Ländern.
- Java und alle Java-basierenden Marken und Logos sind Marken der Sun Microsystems, Inc. in den USA und/oder anderen Ländern.
- Microsoft Windows, Windows NT und das Windows-Logo sind Marken der Microsoft Corporation in den USA und/oder anderen Ländern.
- PC Direct ist eine Marke der Ziff Communications Company in den USA und/oder anderen Ländern.
- SET und das SET-Logo sind Marken der SET Secure Electronic Transaction LLC.
- UNIX ist eine eingetragene Marke der Open Group in den USA und/oder anderen Ländern.
- Marken anderer Unternehmen/Hersteller werden anerkannt.

#### **Änderungen in der IBM Terminologie**

Die ständige Weiterentwicklung der deutschen Sprache nimmt auch Einfluß auf die IBM Terminologie. Durch die daraus resultierende Umstellung der IBM Terminologie, kann es u. U. vorkommen, dass in diesem Handbuch sowohl alte als auch neue Termini gleichbedeutend verwendet werden. Dies ist der Fall, wenn auf ältere existierende Handbuchausschnitte und/oder Programmteile zurückgegriffen wird.

Diese Veröffentlichung ist eine Übersetzung des Handbuchs  
*IBM DB2 Universal Database Developing Enterprise Java Applications Using DB2 UDB, Version 7.2,*

herausgegeben von International Business Machines Corporation, USA

© Copyright International Business Machines Corporation 2001

© Copyright IBM Deutschland GmbH 2001

Informationen, die nur für bestimmte Länder Gültigkeit haben und für Deutschland, Österreich und die Schweiz nicht zutreffen, wurden in dieser Veröffentlichung im Originaltext übernommen.

Möglicherweise sind nicht alle in dieser Übersetzung aufgeführten Produkte in Deutschland angekündigt und verfügbar; vor Entscheidungen empfiehlt sich der Kontakt mit der zuständigen IBM Geschäftsstelle.

Änderung des Textes bleibt vorbehalten.

Herausgegeben von:  
SW TSC Germany  
Kst. 2877  
Juni 2001

---

# Inhaltsverzeichnis

<b>Entwickeln von Enterprise-Java- Anwendungen mit DB2 UDB Version 7.2 . . . 1</b>	XML Extender . . . . . 17
Einführung. . . . . 1	MQSeries-Aktivierung . . . . . 17
Java-Aktivierung . . . . . 1	Net.Data . . . . . 21
JDBC-Treiberunterstützung . . . . . 1	Net Search Extender . . . . . 21
SQLj-Unterstützung . . . . . 4	Spatial Extender . . . . . 21
Gespeicherte Java-Prozeduren und benutzerdefinierte Java-Funktionen. . . . 5	DB2-Lernprogramm für integrierte Webser- vices . . . . . 22
Java-Anwendungsentwicklung für Webserver 6	Entwerfen von Anwendungen mit Unified
VisualAge für Java, Professional Edition . . 6	Modeling Language (UML) . . . . . 23
VisualAge für Java, Enterprise Edition . . 8	Zusammenfassung . . . . . 23
WebSphere Studio . . . . . 10	Zusätzliche Informationen . . . . . 23
WebSphere Application Server . . . . . 12	
Erweiterungen für die Anwendungs- entwicklung . . . . . 17	<b>Bemerkungen . . . . . 25</b>
	Marken . . . . . 28



---

# Entwickeln von Enterprise-Java-Anwendungen mit DB2 UDB Version 7.2

von: Grant Hutchison, DB2/IBM Integration Center

---

## Einführung

DB2 Universal Database (UDB) unterstützt alle wichtigen Internet-Standards, was es zu einer idealen Datenbank für den Einsatz im Web macht. Es weist die nötige Speicherübertragungsgeschwindigkeit auf, um Internet-Suchen und komplexe Textvergleiche in Verbindung mit den Skalierbarkeits- und Verfügbarkeitskenndaten einer relationalen Datenbank zu ermöglichen. DB2 Universal Database unterstützt WebSphere, Java und XML Extender, was es Ihnen erleichtert, e-business-Anwendungen einzusetzen. Im vorliegenden Dokument werden die Funktionen von DB2 UDB Universal Developer's Edition Version 7.2 (UDE) und der darin enthaltenen Entwicklungs-Tools wie VisualAge für Java, Professional Edition, und WebSphere Studio für die Entwicklung von Webanwendungen erörtert.

---

## Java-Aktivierung

DB2 Universal Database unterstützt viele Arten von Java-Programmen. Es bietet Treiberunterstützung für Client-Anwendungen und Applets, die mit Java Database Connectivity (JDBC) in Java geschrieben wurden. Es bietet auch Unterstützung für eingebettetes SQL für Java (SQLj), benutzerdefinierte Java-Funktionen und gespeicherte Java-Prozeduren.

### JDBC-Treiberunterstützung

Der DB2-JDBC-Anwendungstreiber (Typ 2) (Abb. 1 auf Seite 2) ermöglicht Java-Anwendungen, DB2-Aufrufe über JDBC durchzuführen. Aufrufe an den JDBC-Anwendungstreiber werden in systemeigene Java-Methoden übersetzt. Die Java-Anwendungen, die diesen Treiber verwenden, müssen auf einem DB2-Client ausgeführt werden, über den JDBC den Datenfluss zum DB2-Server anfordert.

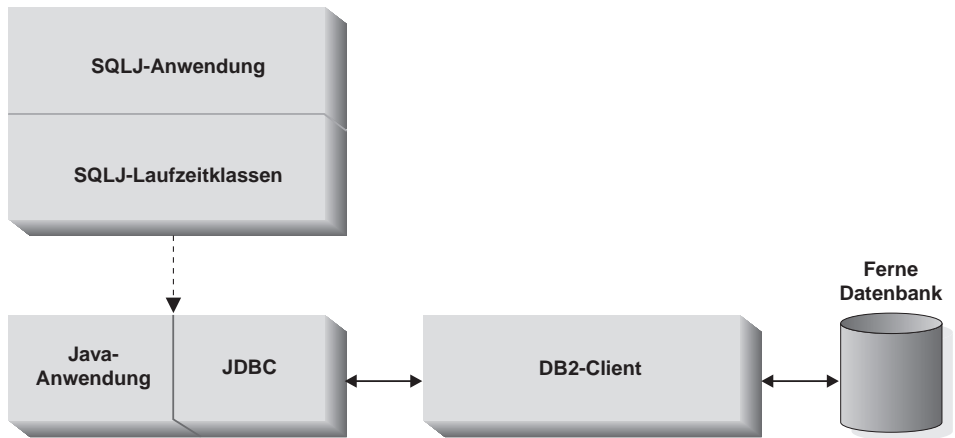


Abbildung 1. DB2-Java-Anwendungsimplementierung

Der DB2-JDBC-Applet-Treiber (Typ 3) (Abb. 2 auf Seite 3) umfasst einen JDBC-Client und einen JDBC-Server. Der DB2-JDBC-Applet-Treiber wird vom Web-Browser zusammen mit dem Applet geladen. Wenn das Applet eine Verbindung zur DB2-Datenbank anfordert, öffnet der Applet-Treiber einen TCP/IP-Socket auf der Maschine, auf der der Webserver und der DB2-Client ausgeführt werden. Nachdem eine Verbindung aufgebaut wurde, sendet der Applet-Treiber jede der nachfolgenden Anforderungen eines Datenbankzugriffs über die TCP/IP-Verbindung vom Applet an den JDBC-Server. Der JDBC-Server führt dann die entsprechenden DB2-Aufrufe für die Task durch. Bei Beendigung sendet der JDBC-Server die Ergebnisse über die Verbindung an den JDBC-Client zurück.

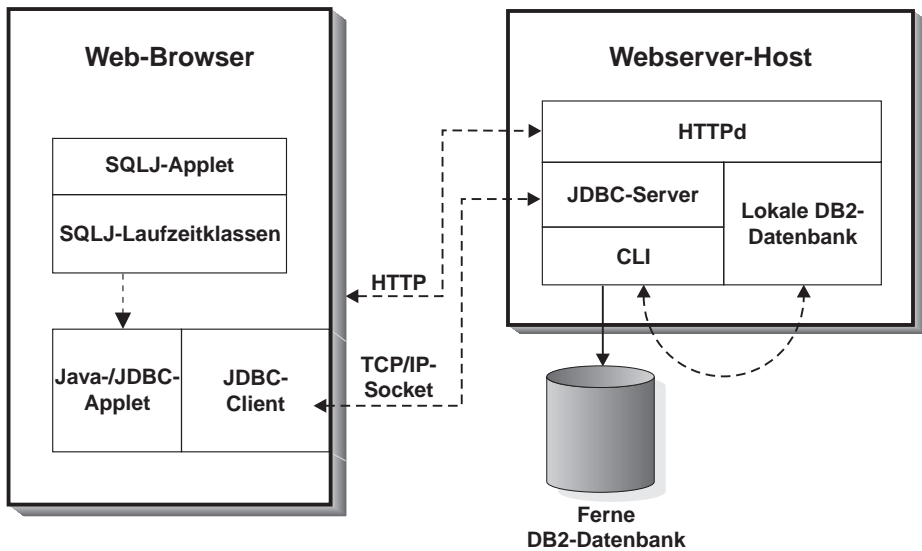


Abbildung 2. DB2-Java-Applet-Implementierung

JDBC 1.0 definiert die Java-APIs für den Zugriff auf relationale Datenbanken. Mit der Einführung von JDBC 2.0 wurden die APIs auf zwei Teile aufgeteilt:

#### JDBC 2.0 Core API

enthält im Laufe der Zeit vorgenommene Verbesserungen, wurde jedoch wie die JDBC 1.0-API klein und zweckorientiert gehalten, um die Benutzerfreundlichkeit zu wahren. Code, der für die 1.0-API geschrieben wurde, funktioniert auch mit der 2.0-API. Die 2.0 API-Klassen sind im Paket `java.sql` enthalten.

#### JDBC 2.0 Optional Package-API

definiert bestimmte Arten von zusätzlicher Funktionalität, wenn Lieferanten bereit sind, die Funktionalität zu liefern, und Anwendungsentwickler bereit sind, die Funktionalität zu verwenden. IBM hat diese neuen Klassen und Schnittstellen in neuen Paketen implementiert. Sie können die neuen Klassen und Schnittstellen mit Hilfe der folgenden Importanweisungen in Ihren Anwendungen verwenden:

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

Der DB2 JDBC 2.0-Treiber unterstützt die folgenden Funktionen der JDBC 2.0 Optional Package-API:

- DataSource-Unterstützung
- Java Naming and Directory Interface (JNDI) für die Benennung von Datenbanken
- Java Transaction API (JTA) — nur beim DB2-JDBC-Anwendungstreiber verfügbar
- Zusammenschluss von Datenbankverbindungen

Die JNDI- und Verbindungszusammenschluss-Schnittstellen sind im JDBC-Treiber definiert, und ihre Implementierung wird von WebSphere Application Server Version 3.5 bereitgestellt. Der Verbindungszusammenschluss ist als Teil der JDBC 2.0 Optional Package-API definiert. JNDI- und DataSource- Objekte werden als Alternative zur Verwendung von DriverManager- Objekten für den Zugriff auf relationale Datenserver bereitgestellt.

JDBC 2.0-Unterstützung ist unter den 32-Bit-Betriebssystemen AIX, Solaris, HP-UX, Linux, Linux für OS/390 und Windows verfügbar. Damit Sie die neuen Funktionen von JDBC 2.0 nutzen können, müssen Sie sowohl den JDBC 2.0-Treiber als auch das Java Development Kit (JDK) 1.2 installieren, da der JDBC 1.2-Treiber der Standardtreiber ist. Geben Sie zum Installieren des JDBC 2.0-Treibers den Befehl **usejdbc2** aus dem Verzeichnis `sql1lib/java12` ein. Wenn Sie wieder den JDBC 1.2-Treiber verwenden möchten, führen Sie den Befehl **usejdbc1** aus dem Verzeichnis `sql1lib/java12` aus.

Weitere Informationen finden Sie in folgenden Quellen:

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

## SQLj-Unterstützung

Die DB2-SQLj-Unterstützung ermöglicht Ihnen die Erstellung und Ausführung von SQLj-Applets und -Anwendungen. Diese Java-Programme enthalten eingebettete SQL-Anweisungen, die vorkompiliert und mit einer DB2 UDB-Datenbank verbunden sind. SQLj-Anwendungen nutzen die JDBC-Unterstützung und benötigen die SQLj-Laufzeitklassen, um SQL-Pakete zu authentifizieren und auszuführen, die bei der Vorkompilierung und Bindung mit der Datenbank verbunden wurden.

Der SQLj-Standard umfasst drei Komponenten: eingebettetes SQLj, ein Umsetzungsprogramm und eine Laufzeitumgebung. Das Umsetzungsprogramm übersetzt die SQLj-Dateien, um Java-Quellendateien und -Profile zu erzeugen. Die Laufzeitumgebung führt SQL-Operationen in JDBC aus und verwendet das Profil, um Einzelangaben zu Datenbankverbindungen zu erhalten.



## Gespeicherte Java-Prozeduren und benutzerdefinierte Java-Funktionen

Sie können gespeicherte Prozeduren in Java in gleicher Weise wie in anderen Sprachen erstellen und verwenden, wobei es nur wenige kleine Unterschiede gibt. Nach der Codierung der gespeicherten Prozedur muss sie mit der Anweisung `CREATE PROCEDURE` bei der Datenbank registriert werden. Anschließend können Sie die gespeicherte Prozedur in Ihrer Anwendung aufrufen. Die gespeicherte Prozedur kann abgesichert (`FENCED`) oder nicht abgesichert (`NOT FENCED`) sein.

Sie können auch benutzerdefinierte Funktionen in Java in gleicher Weise wie in anderen Sprachen erstellen und verwenden, wobei es auch hier nur wenige kleine Unterschiede gibt. Nach der Codierung der benutzerdefinierten Funktion muss sie mit der Anweisung `CREATE FUNCTION` bei der Datenbank registriert werden. Die benutzerdefinierte Funktion kann dann in SQL-Anweisungen in Ihrer Anwendung verwendet werden. Die benutzerdefinierte Funktion kann abgesichert (`FENCED`) oder nicht abgesichert (`NOT FENCED`) sein.

Sie können DB2 Stored Procedure Builder (SPB) verwenden, um gespeicherte Java-Prozeduren zu entwickeln. Als Hilfe zum Erstellen von gespeicherten Prozeduren bietet DB2 SPB Entwicklungsassistenten, die Sie durch die Grundmuster der Entwicklung führen und Ihnen beim Erstellen von SQL-Abfragen sowie beim Ermitteln des Leistungsaufwands für den Aufruf einer gespeicherten Prozedur helfen.

DB2 SPB ist in Java implementiert, und alle Datenbankverbindungen werden mit Hilfe von JDBC verwaltet. Mit Hilfe eines JDBC-Treibers können Sie eine Verbindung zu einem beliebigen DB2-Aliasnamen oder einer anderen DB2 UDB-Datenbank herstellen, für die Sie Host, Anschluss und Datenbanknamen angeben können. Mit DB2 SPB werden mehrere JDBC-Treiber installiert, die unter verschiedenen Betriebssystemen verwendet werden können.

Zum Ausführen Ihrer benutzerdefinierten Funktionen und gespeicherten Prozeduren auf dem Server ruft DB2 JVM (Java Virtual Machine) auf. Vor dem Initialisieren der Datenbank muss Ihr Datenbankadministrator das entsprechende JDK auf Ihre DB2-Server installieren und konfigurieren. Sie haben die Möglichkeit, einzelne Java-Klassendateien für Ihre gespeicherten Prozeduren und benutzerdefinierten Funktionen zu verwenden oder aber die Klassendateien in einer Java-Archivdatei (JAR) zu sammeln und die JAR-Datei in der Datenbank zu installieren.

JVM wird beim ersten Aufruf einer benutzerdefinierten Java-Funktion oder einer gespeicherten Java-Prozedur geladen. In jedem Fall bleibt JVM solange geladen, bis der Einbettungsprozess endet. Für nicht abgeschirmte benutzerdefinierte Funktionen und gespeicherte Prozeduren wird eine JVM pro Datenbanke Exemplar geladen und im Adressraum der Datenbanksteuerkomponente ausgeführt, um die bestmögliche Leistung zu erzielen. Für abgeschirmte benutzerdefinierte Funktionen wird eine separate JVM innerhalb des db2udf-Prozesses geladen. Auch für abgeschirmte gespeicherte Prozeduren wird eine separate JVM innerhalb des db2dari-Prozesses geladen.

---

## Java-Anwendungsentwicklung für Webserver

Mit UDE Version 7.2 werden mehrere Tools geliefert, die Unterstützung für die Webaktivierung bieten. VisualAge für Java Version 3.5 ist eine integrierte Entwicklungsumgebung, die es Ihnen ermöglicht, Java-Anwendungen zu erstellen, testen und unter WebSphere Application Server und DB2 Universal Database einzusetzen. WebSphere Studio ist ein Paket mit Tools, die alle Aspekte der Website-Entwicklung in einer gemeinsamen Schnittstelle vereinen. WebSphere Application Server Standard Edition Version 3.5 bietet eine stabile Entwicklungsumgebung für e-business-Anwendungen. Die zugehörigen Komponenten ermöglichen Ihnen, angepasste, dynamische Webinhalte rasch und einfach zu erstellen und einzusetzen.

### VisualAge für Java, Professional Edition

VisualAge für Java, Professional Edition Version 3.5 (im Lieferumfang von UDE Version 7.2 enthalten) enthält Funktionen und Leistungsverbesserungen, die es leichter denn je machen, skalierbare, leistungsfähige e-business-Anwendungen zu erstellen. Die enge Integration mit IBM WebSphere Application Server, WebSphere Studio und DB2 Universal Database verkürzt die Entwicklungszeit und verbessert die Produktivität, während sie gleichzeitig einfacheren, sicheren Zugriff auf Unternehmensdaten bietet.

IBM Data Access JavaBeans, eine optional installierbare Funktion von VisualAge für Java, ermöglicht Anwendungsentwicklern, auf einfache Weise auf JDBC-aktivierte relationale Datenbanken zuzugreifen.

IBM Data Access JavaBeans, im Paket *com.ibm.db* enthalten, beinhaltet Klassen zur Vereinfachung des Zugriffs auf relationale Datenbanken und bietet die folgenden erweiterten Funktionen:

### **Caching von Abfrageergebnissen**

SQL-Abfrageergebnisse können alle gleichzeitig abgerufen und in einen Cache gestellt werden. Die Anwendung oder das Servlet können den Cache vor- und rückwärts durchgehen oder direkt ein beliebiges Ergebnis im Cache ansteuern.

Im Gegensatz dazu stehen die Funktionen im Paket *java.sql*, in dem Zeilen einzeln und nur vorwärts aus der Datenbank abgerufen werden, und eine neu abgerufene Zeile die letzte abgerufene Zeile überlagert, sofern nicht zusätzlicher Code geschrieben wird, um die Funktionalität zu erweitern. Für große Ergebnismengen bietet IBM Data Access JavaBeans Möglichkeiten, Pakete abzurufen und zu verwalten, die Untermengen der vollständigen Ergebnismenge sind.

### **Aktualisierungen über den Ergebnis-Cache**

Das Servlet kann Standard-Java-Methoden verwenden, um Zeilen im Ergebnis-Cache zu ändern, hinzuzufügen oder zu löschen. Änderungen am Cache können an die zugrundeliegende relationale Tabelle weitergegeben werden.

### **Abfrageparameterunterstützung**

Die Basis-SQL-Abfrage ist als eine Java-Zeichenfolge definiert, wobei Parameter einige der eigentlichen Werte ersetzen. Wenn die Abfrage ausgeführt wird, bietet IBM Data Access JavaBeans eine Möglichkeit, die Parameter durch Werte zu ersetzen, die zur Laufzeit zur Verfügung gestellt werden. Zum Beispiel könnten die Parameterwerte von einem Benutzer in einem HTML-Formular übergeben werden.

### **Metadatenunterstützung**

Ein *StatementMetaData*-Objekt enthält die Basis-SQL-Abfrage. Höhere Ebenen von Metadaten können dem Objekt hinzugefügt werden, um die Übergabe von Parametern an die Abfrage und das Arbeiten mit den zurückgegebenen Ergebnissen zu vereinfachen. Wenn die Abfrage ausgeführt wird, werden die Parameter automatisch zwischen Java-Datentypen und den entsprechenden SQL-Datentypen konvertiert.

## VisualAge für Java, Enterprise Edition

VisualAge für Java, Enterprise Edition Version 3.5 (nicht im Lieferumfang von UDE Version 7.2 enthalten) ermöglicht es Ihnen, Java-Anwendungen zu erstellen, zu testen und in einer einzelnen Umgebung einzusetzen. Neben seinen visuellen Programmierungsfunktionen bietet VisualAge für Java Assistenten, die Sie rasch durch viele Tasks führen, einschließlich der Erstellung von Applets, Servlets und Anwendungen. Es ermöglicht Ihnen auch, nach Bedarf aus dem zugrundeliegenden Dateisystem vorhandenen Code zu importieren und Code zu exportieren. VisualAge für Java bietet ein SQLj-Tool, das den SQLj-Standard implementiert, was es Ihnen ermöglicht, den Datenbankzugriff zu vereinfachen. Die Umsetzungsprogrammkomponente ist in die Entwicklungsumgebung integriert, so dass Sie SQLj-Dateien importieren, übersetzen und editieren können. Die Laufzeitumgebung ist eine installierbare Funktion, die Ihrem Arbeitsbereich hinzugefügt wird. Die ursprünglichen SQLj-Quelldateien werden wie auch die Profile im Projektressourcenverzeichnis verwaltet. Das SQLj-Tool von VisualAge für Java erstellt Profile im Projektressourcenverzeichnis.

### Permanenzverwaltung

Die Funktion Persistence Builder von VisualAge für Java generiert eine Code-Ebene, die alle JDBC-Zugriffsaufrufe implementiert, die zum Einfügen von Daten für ein Objekt aus der Datenbank sowie zum Aktualisieren oder Abrufen dieser Daten erforderlich sind. Persistence Builder ist ein Produktivitätstool, das viele der Problemstellungen bei der Entwicklung von Datenbankanwendungen vereinfacht. Wenn die Anwendung mit Hilfe der Objektmodellierung entwickelt wurde, kann der Schema-Browser von Persistence Builder verwendet werden, um die nötigen Tabellen zu generieren, damit die Objekte bestehen bleiben (*Top-Down-Zuordnung*). Wenn die Datenbank vorhanden ist, kann der Schema-Browser verwendet werden, um die Tabellen in Objekte zurückzuwandeln (*Bottom-Up-Zuordnung*). Das dritte Verfahren wird als *Outside-In-Verfahren* bezeichnet und stellt eine Kompromisslösung dar, bei der beide Modelle beibehalten und entsprechende Anpassungen am Objektmodell bzw. am relationalen Modell vorgenommen werden. Mit Persistence Builder von VisualAge für Java kann die Effizienz der Programmierer verbessert werden. Darüber hinaus können damit Datenmodell und Objektmodell getrennt werden. WebSphere Application Server-Systeme können den Verbindungszusammenschluss für die DB2 UDB-Datenbankverbindungen bereitstellen, während der Persistence Builder-Code generiert werden kann, um den Verbindungszusammenschluss zu nutzen, statt direkt eine Verbindung zur Datenbank herzustellen. Persistence Builder wird in erster Linie bei der Entwicklung von EJBs verwendet, kann aber auch Links zwischen Objektzuordnungen und Transaktionen bereitstellen. Hierfür werden die entsprechenden Integritätsbedingungen und Transaktionen in DB2 verwendet.

## **Debug-Funktionen**

VisualAge für Java bietet mehrere Tools zur Entwicklung von JSPs (JavaServer Pages). Servlet Launcher ermöglicht Ihnen, einen Webserver zu starten, Ihren Web-Browser zu öffnen und ein Servlet aufzurufen. JSP Execution Monitor ermöglicht Ihnen, die Ausführung der JSP-Quelle, der generierten Servlets und der HTML-Quelle bei ihrer Generierung zu überwachen. VisualAge für Java ermöglicht Ihnen auch, Unterbrechungspunkte innerhalb des Servlet-Codes zu setzen, das Servlet an Unterbrechungspunkten dynamisch zu aktualisieren und die Ausführung des Servlets mit den vorgenommenen Änderungen fortzusetzen.

IBM WebSphere Test Environment ist eine optional installierbare Funktion von VisualAge für Java. Diese Funktion ermöglicht Ihnen, Servlets und Enterprise JavaBeans (EJBs) innerhalb der VisualAge für Java-Umgebung auszuführen, bevor sie auf WebSphere Application Server-Systemen eingesetzt werden.

Die Umgebung bietet eine Servlet-Steuerkomponente und die erforderlichen Services für EJB-Anwendungen. Da diese Testumgebung innerhalb der Entwicklungsumgebung bereitgestellt wird, können Unterbrechungspunkte in Servlets oder JSPs gesetzt werden, und es kann eine Client-Anwendung oder ein Browser aufgerufen werden, um alle Aspekte der serverseitigen Java-Programme zu testen. Der ferne Debugger kann verwendet werden, um Unterbrechungspunkte in EJBs zu setzen, die auf einem EJB-Server wie WebSphere Application Server ausgeführt werden.

Mit DB2 Stored Procedure Builder (SPB) und dem fernen Debugger von VisualAge können Sie eine gespeicherte Prozedur, die auf einem DB2-Server installiert ist, fern testen. Zum Testen einer gespeicherten Prozedur müssen Sie die gespeicherte Prozedur im Debug-Modus ausführen. Es ist nicht nötig, die gespeicherten Prozeduren aus einem Anwendungsprogramm heraus zu testen. Sie können eine gespeicherte Prozedur unabhängig vom aufrufenden Anwendungsprogramm testen.

Mit dem Notizbuch für die Debug-Merkmale in SPB können Sie Debug-Datensätze in der Debug-Tabelle für gespeicherte Prozeduren ändern, hinzufügen oder entfernen. Wenn Sie ein Datenbankadministrator sind und die ausgewählte gespeicherte Prozedur erstellt haben, können Sie anderen Benutzern die Datenbankberechtigung zum Testen der gespeicherten Prozedur erteilen.

## WebSphere Studio

WebSphere Studio Version 3.5 ist ein Paket mit Tools, die alle Aspekte der Website-Entwicklung in einer gemeinsamen Schnittstelle vereinen. WebSphere Studio macht es leichter denn je, gemeinsam dynamische Webanwendungen zu erstellen, zu assemblieren, zu veröffentlichen und zu pflegen. Autoren, Grafiker, Programmierer und Webmaster können an denselben Projekten arbeiten und jederzeit auf die benötigten Dateien zugreifen. Studio umfasst die Komponenten Workbench, Page Designer, Remote Debugger und Assistenten. Es wird mit Testversionen von Webentwicklungsprodukten wie Macromedia Flash, Fireworks, Freehand und Director geliefert. WebSphere Studio ermöglicht Ihnen die Durchführung aller zur Erstellung interaktiver Websites, die Ihre erweiterten Geschäftsfunktionen unterstützen, nötigen Schritte.

Dazu zählt Folgendes:

- Verwenden von Studio-Assistenten zum Erstellen von Java-Beans, Datenbankabfragen und Java-Servlets.
- Gruppieren der Website-Dateien in Projekten und Ordern. Mit Hilfe von Filtern und globalen Suchfunktionen finden Sie rasch die Dateien, die Sie benötigen. Projekte können nach VisualAge für Java exportiert und unter WebSphere Application Server eingesetzt werden.
- Pflegen der Dateien einzeln oder in einem gemeinsamen Versionssteuerungssystem.
- Editieren und Aktualisieren der Dateien mit den von Ihnen bevorzugten Tools. Beim Öffnen einer Studio-Datei können Sie rasch die Standardauswahl aufrufen oder ein alternatives Tool auswählen.
- Rasches Bewerten von Dateibeziehungen und Ermitteln fehlerhafter Links.
- Veröffentlichen Ihrer Website in jedem Entwicklungsstadium auf einem der WebSphere Application Server-Systeme. Innerhalb von Studio Workbench gelangen Sie direkt von der Entwicklung der Site zu deren Veröffentlichung.

Studio Workbench hilft Ihnen dabei, die Anwendungen und Dateien der Website zu verwalten und pflegen, und bietet die folgenden Funktionen:

- Eine grafische Darstellung der Link-Beziehungen zwischen den Dateien in einem Projekt.
- Die automatische Aktualisierung von Links, wenn sich Dateien ändern oder versetzt werden.
- Die Möglichkeit, den Website-Produktionszyklus in Schritte zu untergliedern und verschiedene Stadien auf verschiedenen (und auf mehreren) Servern zu veröffentlichen.
- Einen Importassistenten, der die Übertragung vorhandener Site-Inhalte direkt in ein Studio-Projekt vereinfacht.
- Eine rasche Möglichkeit, Websites oder untergeordnete Sites in einer einzelnen Datei zu archivieren.
- Die Möglichkeit, Tools anderer Anbieter rasch in die Workbench-Umgebung zu integrieren.
- Eine erweiterte Teamumgebung mit einer gemeinsamen Sicht der laufenden Arbeiten durch Integration gängiger Software für die Quellensteuerungsverwaltung wie IBM VisualAge TeamConnection.

Studio Page Designer bietet eine visuelle Entwicklungsumgebung, die es Ihnen ermöglicht, JSP, Java-Servlets und andere Java-basierte Webtools zu erstellen. Zum Beispiel können Sie die visuelle Umgebung verwenden, um JavaBeans in JSP-Anwendungen zu ziehen und dort zu übergeben. Studio Page Designer kann auch verwendet werden, um DHTML- (Dynamic Hyper Text Markup Language) und HTML-Seiten zu erstellen. Diese Komponente bietet weiterhin die Möglichkeit, auf einfache Weise HTML- und DHTML-Quellcode zu editieren und zwischen Quellcode- und Browser-Sicht umzuschalten. Studio Remote Debugger ermöglicht das Testen von JSP-Dateien und Java-Servlets auf Quellcodeebene innerhalb der Studio-Umgebung.

## WebSphere Application Server

WebSphere Application Server verbindet die Portierbarkeit serverseitiger Geschäftsanwendungen mit der Leistung und Handhabung von Java-Technologien und bietet damit eine umfassende Plattform für die Entwicklung Java-basierter Webanwendungen. Diese Komponente ermöglicht leistungsfähige Interaktionen zwischen Unternehmensdatenbanken und Transaktionssystemen.

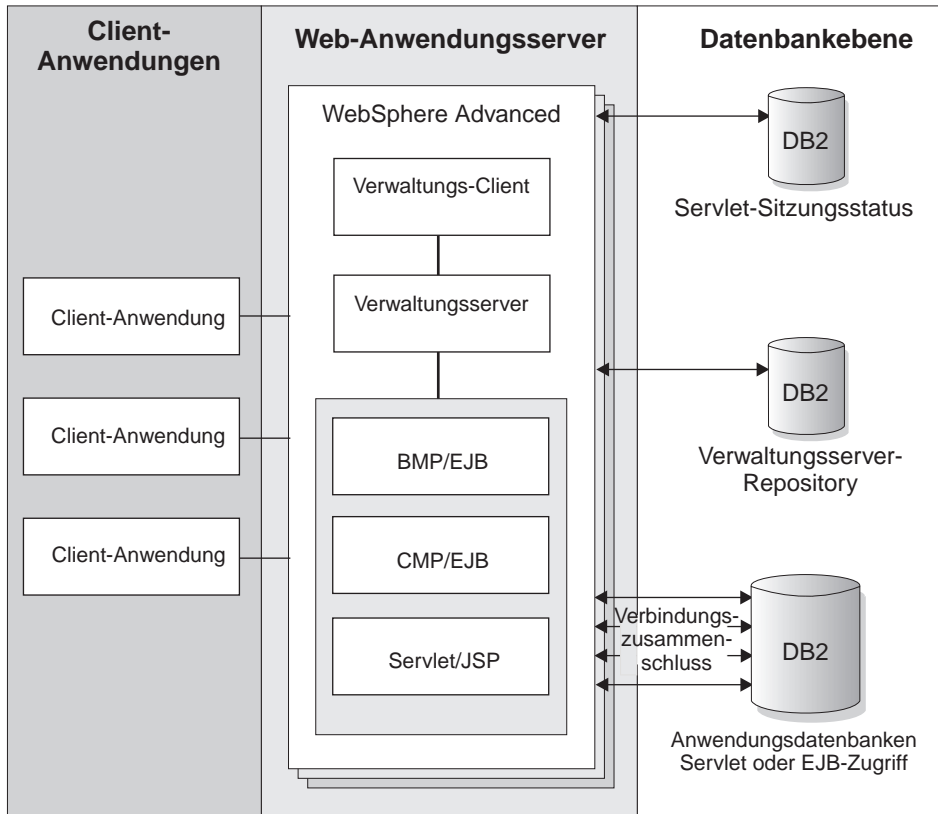


Abbildung 3. WebSphere Application Server



Anwendungsserver erweitern das Leistungsspektrum eines Webservers um die Verarbeitung von Anwendungsanforderungen. Angenommen, eine Anwendung umfasst HTML-Seiten, Servlets und Enterprise-Beans.

In diesem Fall macht der Anwendungsserver folgenden Austausch möglich:

1. Ein Benutzer eines Web-Browsers im öffentlichen Internet besucht eine Firmen-Website. Der Benutzer fordert die Nutzung einer Anwendung an, die den Zugriff auf Daten in einer Datenbank ermöglicht.
2. Die Benutzeranforderung wird dem Webserver übermittelt.
3. Der Webserver stellt fest, dass die Anforderung eine Anwendung erfordert, die Servlets und Enterprise-Beans beinhaltet. Er leitet die Anforderung an IBM WebSphere Application Server weiter.
4. Das Produkt IBM WebSphere Application Server leitet die Anforderung an einen der zugehörigen Anwendungsserver weiter, auf dem die Anwendung ausgeführt wird.
5. Die Anwendung verarbeitet die Benutzeranforderung.
6. Der Anwendungsserver arbeitet mit dem Webserver zusammen, um die Ergebnisse dem Web-Browser des Benutzers zu übermitteln.

WebSphere Application Server stellt eine Servlet-Steuerkomponente bereit, die die Java-Servlet-API 2.1 implementiert. Die Komponente beinhaltet eigene Pakete, die die Java-Servlet-API erweitern und ergänzen und JDK 1.2 unter allen unterstützten Betriebssystemen einsatzfähig machen. Der Erweiterungen und Ergänzungen machen es einfacher, den Sitzungsstatus zu verwalten, angepasste Webseiten zu erstellen, bessere Servlet-Fehlerberichte zu generieren und auf Datenbanken zuzugreifen.

Application Server unterstützt JSPs, einen leistungsfähigen Ansatz für dynamische Webseiteninhalte. Einer der Vorteile von JSPs besteht darin, dass sie es Ihnen ermöglichen, die HTML-Codierung wirkungsvoll von der Geschäftslogik in Ihren Webseiten zu trennen. Die IBM Erweiterungen der JSP-Spezifikation umfassen HTML-ähnliche JSP-Befehle, die es HTML-Autoren erleichtern, ihren Webseiten die Leistungsfähigkeit von Java hinzuzufügen.

## **Verbindungszusammenschluss**

Die Unterstützung von Verbindungszusammenschlüssen wird von WebSphere Application Server bereitgestellt. Der Verbindungszusammenschluss ermöglicht Ihnen das Steuern und Reduzieren der Ressourcen, die von webbasierten Anwendungen verwendet werden. Webbasierte Anwendungen, die auf Datenserver zugreifen, führen zu einem höheren und schlechter vorhersehbaren Systemaufwand als andere Webanwendungen, weil Webbenutzer häufig Verbindungen herstellen und trennen. Oft werden mehr Ressourcen für das Herstellen und Trennen von Verbindungen als für die Interaktionen selbst aufgewendet.

Beim Verbindungszusammenschluss wird der Systemaufwand für Verbindungen auf mehrere Benutzeranforderungen verteilt, indem ein Pool von Verbindungen eingerichtet wird, auf den Servlets zurückgreifen können. Nachdem die anfänglichen Ressourcen aufgewendet wurden, um die Verbindungen im Pool herzustellen, ist der zusätzliche Systemaufwand für das Herstellen bzw. Trennen von Verbindungen unerheblich, weil die vorhandenen Verbindungen laufend wiederverwendet werden.

Der Verbindungszusammenschluss, wie ihn WebSphere Application Server bietet, definiert, wie ein Servlet über die Erweiterungs-APIs des neuen JDBC 2.0-Standards auf relationale Datenserver zugreift. Es handelt sich dabei um ein neues Modell. Servlets, die unter WebSphere Application Server Version 3.x ausgeführt werden, können so codiert werden, dass sie Verbindungsressourcen von Datenservern effizient nutzen.

**Vorteile des Verbindungszusammenschlusses:** Jedes Mal, wenn ein Resource versucht, auf eine Datenbank zuzugreifen, muss sie eine Verbindung zur betreffenden Datenbank herstellen. Eine Datenbankverbindung erfordert einen bestimmten Systemaufwand. Sie erfordert Ressourcen, um die Verbindung herzustellen, aufrechtzuerhalten und schließlich zu trennen, wenn sie nicht mehr benötigt wird. Der Systemaufwand ist aufgrund der Häufigkeit der Herstellung und Trennung von Verbindungen für webbasierte Anwendungen besonders hoch. Benutzerinteraktionen sind aufgrund des „Umherschweifens“ beim Surfen im Web meist kurz. Da Internet-Anforderungen von praktisch überall eingehen können, können die Nutzungsvolumina groß und schwer vorhersehbar sein. WebSphere Application Server richtet aus diesem Grund einen Pool von Datenbankverbindungen ein, der von Anwendungen auf Anwendungsservern gemeinsam genutzt wird.

Der Verbindungszusammenschluss ermöglicht dem Administrator, die Ressourcen zu steuern und zu reduzieren, die von webbasierten Anwendungen verwendet werden, indem der Systemaufwand für Verbindungen auf mehrere Benutzeranforderungen verteilt wird. Der Verbindungszusammenschluss kann auch die Antwortzeiten von webbasierten Anwendungen verbessern.

Wenn ein Benutzer über das Web eine Ressource anfordert, greift die Ressource auf eine Datenquelle zu. Weil die Datenquelle eine vorhandene Verbindung aus dem Pool sucht und verwendet, erfordert die Benutzeranforderung keinen Systemaufwand zum Erstellen einer neuen Verbindung.

Jede Verbindung wird einer bestimmten Benutzeranforderung zugeordnet. Wenn die Anforderung erfüllt ist und der Benutzer die Antwort erhält, gibt die Datenquelle die Verbindung zur erneuten Benutzung im Pool frei. Es wird wiederum der Systemaufwand zum Trennen der Verbindung vermieden.

Jede Benutzeranforderung verursacht nur einen Bruchteil des Aufwands für das Herstellen und Trennen von Verbindungen. Nachdem die anfänglichen Ressourcen verwendet wurden, um die Verbindungen im Pool herzustellen, ist der zusätzliche Systemaufwand unerheblich, weil die vorhandenen Verbindungen laufend wiederverwendet werden.

**Verwaltung von Verbindungszusammenschlüssen durch WebSphere Application Server:** WebSphere Application Server richtet entsprechend den Angaben des Administrators Verbindungspools ein und verwaltet diese. Sobald die Verbindungen hergestellt sind, werden sie den Benutzeranforderungen entsprechend verteilt. Zwischendurch werden Bereinigungsoperationen ausgeführt, um das Gleichgewicht zwischen verfügbaren Verbindungen und dem Bedarf an Verbindungen zu wahren. Dadurch wird sichergestellt, dass eine bestehende Verbindung verfügbar ist, wenn ein Servlet oder ein Anwendungsserver eine solche benötigt.

Zum Beispiel ermittelt der Verbindungspool in regelmäßigen Zeitabständen Verbindungen, die sich im Leerlauf befinden oder nicht mehr benötigt werden. Im Leerlauf befindliche Verbindungen werden beendet, und nicht mehr benötigte Verbindungen werden wieder dem Verbindungspool zur Verfügung gestellt. Das bedeutet, dass weniger Verbindungen verfügbar sind (und weniger Ressourcen verwendet werden), wenn der Bedarf an Verbindungen gering ist. Eine Verbindung im Leerlauf ist eine Verbindung, bei der das für die Datenquelle festgelegte Leerlaufzeitlimit überschritten wurde. Eine Verbindung wird nicht mehr benötigt, wenn das Eigner-Servlet oder der betreffende Anwendungsserver beendet wurde oder nicht mehr reagiert.

### **WebSphere Standard Edition**

WebSphere Application Server, Standard Edition Version 3.5 (im Lieferumfang von UDE Version 7.2 enthalten) ist eine Komponente von WebSphere Studio. InstantDB ist die Standarddatenbank, die für das WebSphere Standard Edition-Repository verwendet wird. Allerdings kann DB2 Universal Database als Verwaltungsserver-Repository für WebSphere konfiguriert werden. Sie können den DB2-Server auf der gleichen Maschine wie WebSphere Application Server oder auf einem anderen Webserver ausführen.

### **WebSphere Advanced Edition**

WebSphere Application Server Advanced Edition (nicht im Lieferumfang von UDE Version 7.2 enthalten) basiert auf dem Standardanwendungsserver, da eine Datenbank für den Betrieb benötigt wird. DB2 Universal Database ist im Lieferumfang von WebSphere Application Server Advanced Edition Version 3.5 enthalten und kann als Verwaltungsserver-Repository verwendet werden. Es führt Serverfunktionen für Anwendungen ein, die entsprechend der EJB-Spezifikation von Sun Microsystems erstellt wurden, die Unterstützung für die Integration von Webanwendungen in webferne Geschäftssysteme bietet.

Der EJB-Server ist der Anwendungsserver in der dreischichtigen Architektur von WebSphere Application Server. Er verbindet die Client-Schicht von Java-Servlets, Applets, Anwendungen und JSPs mit der Ressourcenverwaltungsschicht, der Datenquelle.

Es gibt zwei Arten von Enterprise-Beans: *Sitzungs-Beans* und *Entitäts-Beans*. Sitzungs-Beans kapseln temporäre Daten, die einem bestimmten Client zugeordnet sind. Entitäts-Beans kapseln permanente Daten, die in einer Datenquelle gespeichert sind. Der Permanenzservice stellt sicher, dass die Daten, die den Entitäts-Beans zugeordnet sind, korrekt mit den entsprechenden Daten in der Datenquelle synchronisiert werden. Der Permanenzservice arbeitet zur Durchführung dieser Task mit dem Transaktionsservice zusammen, um Daten zur richtigen Zeit in die Datenquelle einzufügen, darin zu aktualisieren, daraus zu extrahieren und zu entfernen.

Es gibt zwei Arten von Entitäts-Beans: solche mit Container-verwalteter Permanenz (CMP - Container-Managed Persistence) und solche mit Bean-verwalteter Permanenz (BMP - Bean-Managed Persistence). Bei Entitäts-Beans mit CMP verarbeitet der Permanenzservice beinahe alle Tasks, die zur Verwaltung permanenter Daten nötig sind. CMP-Entitäts-Beans können mit DB2 UDB (einschließlich DB2 für OS/390) als permanenter Datenspeicher implementiert werden. Bei Entitäts-Beans mit BMP verarbeitet die Bean selbst die meisten Tasks, die zur Verwaltung permanenter Daten nötig sind.

### XML Extender

XML (Extensible Markup Language) ist ein anerkanntes Standardverfahren für den Datenaustausch zwischen Anwendungen. Ein XML-Dokument ist ein Dokument, das Zeichendaten und Formatierungsbefehle enthält. Die Formatierungsbefehle können vom Autor des Dokuments definiert werden. Eine Dokumenttypdefinition (DTD - Document Type Definition) wird verwendet, um die Formatierungsdefinitionen und -einschränkungen zu definieren. DB2 XML Extender bietet einen Mechanismus, der es Programmen ermöglicht, XML-Daten durch Nutzung von SQL-Erweiterungen zu bearbeiten. In der Regel sind XML-Dokumente in Form von Einzeldateien in einem Dateisystem gespeichert. DB2 XML Extender bietet eine alternative Speicher- und Bearbeitungsumgebung. XML-Dokumente können als einzelne Spalte oder als Spaltengruppe gespeichert werden.

DB2 XML Extender führt drei neue Datentypen ein: XMLVARCHAR, XMLCLOB und XMLFILE. Der Extender bietet benutzerdefinierte Funktionen zum Speichern, Extrahieren und Aktualisieren von XML-Dokumenten, die in einer einzelnen Spalte gespeichert sind. Suchen können für das gesamte XML-Dokument oder auf Basis von strukturellen Komponenten unter Verwendung des Standortpfads durchgeführt werden, der eine Teilmenge der abgekürzten Syntax verwendet, die durch XML Path Language (XPath) definiert ist. Nebentabellen können verwendet werden, um die Suchleistung für Elemente oder Attribute zu verbessern, die häufig abgefragt werden.

Zum einfacheren Speichern von XML-Dokumenten als Spaltengruppe bietet DB2 XML Extender ein Verwaltungs-Tool, das dem Entwickler bei der Zuordnung zwischen XML und relationaler Datenbank hilft. Eine Dokumentzugriffsdefinition (DAD - Document Access Definition) wird zur Verwaltung der strukturellen und der Zuordnungsdaten für die XML-Dokumente verwendet. Die Dokumentzugriffsdefinition wird als XML-Dokument definiert und gespeichert, wodurch sie einfach bearbeitet und nachvollzogen werden kann. Neue gespeicherte Prozeduren sind verfügbar, um das Dokument zusammenzufügen bzw. aufzuspalten.

### MQSeries-Aktivierung

DB2 Universal Database Version 7.2 enthält eine Reihe von MQSeries-Funktionen, die DB2 UDB-Anwendungen ermöglichen, mit asynchronen Nachrichtenübertragungsoperationen zu interagieren. Das bedeutet, dass MQSeries-Unterstützung für Anwendungen verfügbar ist, die in einer beliebigen, von DB2 UDB unterstützten Programmiersprache geschrieben wurden. Aus Gründen der Einfachheit handelt es sich bei allen in diesem Abschnitt aufgeführten Beispielen um SQL-Anweisungen. Eine WebSphere-Anwendung kann diese MQSeries-SQL-Anweisungen nutzen.

## Nachrichtendarstellungen

MQSeries erfordert nicht, dass die Nachrichten, die es transportiert, eine bestimmte Struktur aufweisen. XML-Nachrichten haben meistens eine selbsterklärende Nachrichtenstruktur. Nachrichten können auch unstrukturiert sein, so dass Benutzercode erforderlich ist, um den Nachrichteninhalt syntaktisch zu analysieren bzw. zu erstellen. Solche Nachrichten sind häufig halbstrukturiert, d. h. sie verwenden entweder Bytepositionen oder feste Begrenzer, um die Felder innerhalb einer Nachricht zu trennen.

MQSeries unterstützt drei Nachrichtenübertragungsmodelle: Datagramme, Veröffentlichen/Teilnehmen (Publish/Subscribe - p/s) und Anfordern/-Antworten (Request/Reply - r/r). Nachrichten, die als Datagramme gesendet werden, werden an ein einzelnes Ziel gesendet, ohne dass eine Antwort erwartet wird. Beim P/S-Modell sendet mindestens ein Herausgeber eine Nachricht an einen Veröffentlichungsservice, der die Nachricht an mindestens einen interessierten Subskribenten verteilt. Das R/R-Modell ist dem Datagramm-Modell ähnlich, der Sender erwartet jedoch, eine Antwort zu erhalten. Die bereitgestellten DB2-MQSeries-Funktionen unterstützen alle drei Nachrichtenmodelle.

MQSeries wird auf vielerlei Arten verwendet. Einfache Datagramme werden ausgetauscht, um mehrere Anwendungen zu koordinieren, Daten auszutauschen, Services anzufordern und über interessante Ereignisse zu benachrichtigen. Die P/S-Darstellung wird häufig verwendet, um Echtzeitdaten rechtzeitig zu verbreiten. Die R/R-Darstellung wird im Allgemeinen als einfache Form von pseudosynchronen Fernprozeduraufrufen verwendet. Durch Kombination der drei Grunddarstellungen können komplexe Modelle erstellt werden.

DB2 UDB Version 7.2 bietet einen neuen MQSeries Assist-Assistenten. Dieser Assistent erstellt eine Tabellenfunktion, die aus einer MQSeries-Warteschlange liest. Zu diesem Zweck werden die benutzerdefinierten MQSeries-Funktionen verwendet, die in DB2 UDB Version 7.2 ebenfalls neu sind. Der Assistent kann jede MQSeries-Nachricht als Zeichenfolge mit begrenzter Länge oder als Spaltenzeichenfolge mit fester Länge behandeln. Die erstellte Tabellenfunktion wertet die Zeichenfolge Ihren Angaben entsprechend aus und gibt jede MQSeries-Zeichenfolge als eine Zeile der Tabellenfunktion zurück. Der Assistent ermöglicht Ihnen auch, eine Sicht auf die Tabellenfunktion zu erstellen und eine MQSeries-Nachricht und das Ergebnis der Tabellenfunktion in einer Voranzeige darzustellen.

## DB2-/MQ-Infrastruktur

In einer Grundkonfiguration, wie in Abb. 4 gezeigt, befindet sich ein MQSeries-Server zusammen mit DB2 Universal Database auf der Datenbankservermaschine. Die MQSeries-Funktionen werden von einem DB2-Server zur Verfügung gestellt. Sie ermöglichen den Zugriff auf andere MQSeries-Anwendungen. Mehrere DB2-Clients können über die Datenbank gleichzeitig auf die MQSeries-Funktionen zugreifen. Die MQSeries-Operationen ermöglichen DB2-Anwendungen, synchron mit anderen MQSeries-Anwendungen Daten auszutauschen. Zum Beispiel bieten die neuen Funktionen eine einfache Methode, mit der eine DB2-Anwendung Datenbankereignisse fernen MQSeries-Anwendungen mitteilen, einen Arbeitsablauf über das optionale Produkt MQSeries Workflow einleiten oder mit dem optionalen Produkt MQSeries Integrator mit einem vorhandenen Anwendungspaket Daten austauschen kann.

### DB2-/MQ-Grundkonfiguration

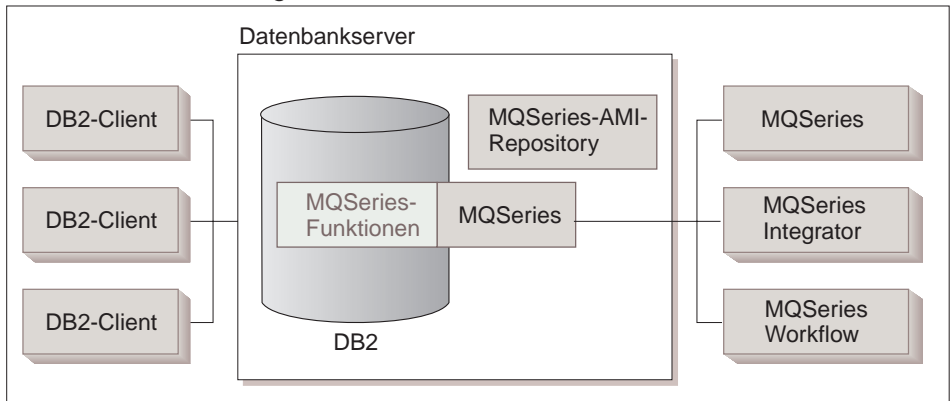


Abbildung 4. DB2-/MQ-Grundkonfiguration

Wenn die MQSeries-Unterstützung als Teil von DB2 Universal Database installiert wird, richtet ein Konfigurationsskript automatisch eine Standardkonfiguration ein, die Clientanwendungen ohne weitere Administratoreingriffe verwenden können. Die Standardkonfiguration ermöglicht Anwendungsprogrammierern, rasch anzufangen, und sie bietet eine einfache Entwicklungsschnittstelle.

## Beispiele für die Nachrichtenübertragung

Die folgende SQL-Anweisung sendet eine einfache Nachricht unter Verwendung der Standardkonfiguration:

```
values MQSEND('einfache Nachricht')
```

Damit wird die Nachricht einfache Nachricht an den MQSeries-Warteschlangenmanager und die MQSeries-Warteschlange geschickt, die in der Standardkonfiguration festgelegt sind. Die mit DB2 Universal Database bereitgestellten MQSeries-Funktionen basieren auf AMI (Application Messaging Interface). AMI unterstützt die Verwendung einer externen Konfigurationsdatei, als AMI-Repository bezeichnet, in der Konfigurationsdaten gespeichert werden. Die Standardkonfiguration enthält ein MQSeries-AMI-Repository, das für die Verwendung mit DB2 UDB konfiguriert ist. Weitere Informationen finden Sie im Internet unter <http://www-3.ibm.com/software/ts/mqseries/api/>.

*Servicepunkt* und *Richtlinien* sind zwei Schlüsselkonzepte beim MQSeries-AMI, die auch für die DB2 UDB-MQSeries-Funktionen gelten. Ein Servicepunkt ist ein logischer Endpunkt, an dem eine Nachricht gesendet oder empfangen werden kann. Im AMI-Repository ist jeder Servicepunkt mit einem MQSeries-Warteschlangennamen und -Warteschlangenmanager definiert. Richtlinien definieren die Servicequalitätsoptionen, die für eine gegebene Nachrichtenübertragungsoperation verwendet werden sollen. Schlüsselqualitäten des Services sind Nachrichtenpriorität und Permanenz. Es werden Standardservicepunkte und Richtliniendefinitionen bereitgestellt, die von Entwicklern verwendet werden können, um ihre Anwendungen zu vereinfachen. Das vorherige Beispiel kann wie folgt geändert werden, um den Servicepunkt und Richtliniennamen explizit anzugeben:

```
values MQSEND('DB2.DEFAULT.SERVICE',  
'DB2.DEFAULT.POLICY', 'einfache Nachricht')
```

Warteschlangen können von einer oder mehr Anwendungen auf dem Server, auf dem sie sich befinden, verwendet werden. In vielen Konfigurationen werden mehrere Warteschlangen definiert, um verschiedene Anwendungen und Zwecke zu unterstützen. Aus diesem Grund ist es oft wichtig, verschiedene Servicepunkte zu definieren, wenn MQSeries-Anforderungen ausgeführt werden. Dies wird im folgenden Beispiel verdeutlicht.

```
values MQSEND('ODS_Input', 'einfache Nachricht')
```

Beachten Sie, dass in diesem Beispiel die Richtlinie nicht angegeben wird, so dass die Standardrichtlinie verwendet wird.

### Weitere Informationen finden Sie unter:

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



## **Net.Data**

Net.Data ist eine Anwendung, die Webanwendungen mit DB2 Universal Database verbindet. Es bietet jetzt XML-Unterstützung, was es Ihnen ermöglicht, XML-Befehle als Ausgabe von Net.Data-Makros zu generieren, statt die Befehle manuell einzugeben. Sie können auch eine XML-Formatvorlage (XSL) angeben, die verwendet werden soll, um die generierte Ausgabe zu formatieren und anzuzeigen.

## **Net Search Extender**

Net Search Extender (nicht im Lieferumfang von UDE Version 7.2 enthalten) verwendet ein Indexierungsverfahren, als n-gram-Index bezeichnet, das einen neuen Textsuch-Extender für hohe Suchgeschwindigkeit bietet. Für diese Extender gibt es im Bereich der Webanwendungen viele Anwendungsmöglichkeiten, da Textfelder häufig von Endbenutzern abgefragt werden.

Das Auffinden der fraglichen Dokumente auf Basis von Textfeldindizes kann die Zufriedenheit von Webbenutzern steigern. Beliebige Spalten, die auf den Datentypen CHAR, VARCHAR oder LONG VARCHAR basieren, können anhand eines n-gram-Indexes indiziert werden. Wenn der Index erstellt und aktiviert wurde, können Suchen mit Hilfe einer neuen gespeicherten Prozedur durchgeführt werden. Aktive Indizes werden im gemeinsam benutzten Speicher gespeichert, um die Suchleistung zu optimieren.

## **Spatial Extender**

Spatial Extender (nicht im Lieferumfang von UDE Version 7.2 enthalten) ermöglicht Benutzern, räumliche Daten in ihre Abfragen zu integrieren. Diese Komponente unterstützt räumliche Typen zur Modellierung von realen Entitäten wie dem Standort von Kunden, Begrenzungen von Parks und dem Verlauf von Kabelleitungen.

## DB2-Lernprogramm für integrierte Webservices

Ein neues DB2-Lernprogramm mit dem Namen *eVideoCentral* steht unter <http://www.ibm.com/software/data/developer/samples/evideo> zum Download zur Verfügung. Das DB2-Lernprogramm *eVideoCentral* stellt eine Beispielösung für ein Unternehmen vor, das einem anderen Unternehmen über das Web Geschäftsservices zur Verfügung stellt. Dieses Konzept wird als Business-to-Business-Anwendung bezeichnet. Das Lernprogramm *eVideoCentral* veranschaulicht die Integration von DB2, WebSphere und MQSeries. Viele Technologien werden verwendet, um *eVideoCentral* zu entwickeln und zu erstellen. Dazu zählen IBM WebSphere Application Server, VisualAge für Java und das Interaktions-Tool DB2 XML Extender. Über das SOAP-Protokoll (Simple Object Access Protocol, siehe <http://www.w3.org/2000/xml/> ) wird auf eine Reihe von zentralen Services für eigenständige Videoverkaufsgeschäfte zugegriffen.

SOAP ist ein einfaches Protokoll zum Austausch von Daten in einer dezentralen, verteilten Umgebung. Es handelt sich dabei um ein XML-basiertes Protokoll, das drei Komponenten beinhaltet:

- Eine äußere Schicht, die ein Gerüst definiert, das beschreibt, was eine Nachricht enthält und wie sie zu verarbeiten ist
- Eine Reihe von Codierungsregeln, um Exemplare von anwendungsdefinierten Datentypen auszudrücken
- Eine Konvention zur Darstellung von Fernprozeduraufrufen und -antworten

SOAP-Services können einem Unternehmen einen Mechanismus an die Hand geben, um auf DB2-Services zuzugreifen, die von einem anderen Unternehmen bereitgestellt werden.

Das Lernprogramm *eVideoCentral* veranschaulicht die Konzeption, Entwicklung und Implementierung von einfachen Daten-Repository- (Einfügung/-Änderung) und Abfrageservices. Die Abfrageservices verwenden den vorhandenen DB2 XML Extender. Die erste Version von *eVideoCentral* wird über das Web zur Verfügung gestellt und umfasst Java-Servlets, JSPs (Java Server Pages), ein Schema für die DB2-Datenbankobjekte sowie die zugehörige Lernprogrammdokumentation.

---

## Entwerfen von Anwendungen mit Unified Modeling Language (UML)

Auf DB2 UDB-Datenmodelle kann nun aus dem beliebten UML-Modellierungsprodukt Rational Rose von Rational Software zugegriffen werden. Dieses Programm kann vorhandene Schemata in ein logisches Modell extrahieren oder Schemata direkt aus dem Modell generieren. Die Verwendung einer gemeinsamen Sprache für das Entwerfen von Anwendungen (wie UML) bietet beim Anwendungsentwicklungsprozess viele Vorteile, sowohl für Objektmodelle als auch für Datenmodelle.

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

---

## Zusammenfassung

DB2 UDB Universal Developer's Edition umfasst sämtliche Tools, die Sie benötigen, um Anwendungen rasch zu erstellen und einzusetzen. Das Paket enthält eine komplette integrierte Entwicklungsumgebung, einen skalierbaren Server für Webanwendungen sowie DB2 UDB-Funktionen wie XML Extender. DB2 Universal Database ist eine skalierbare, auf die harten Anforderungen in Unternehmen abgestimmte Datenbank, die die Grundlage für die Datenverwaltung in Ihrem e-business bildet.

---

## Zusätzliche Informationen

Zusätzliche Informationen finden Sie auf den folgenden Websites:

### **DB2 Universal Database-Ressourcen:**

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

### **VisualAge-Anwendungsentwicklerbereich:**

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

### **WebSphere-Anwendungsentwicklerbereich:**

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

### **Zugriff auf gespeicherte DB2-Prozeduren aus EJBs:**

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



---

## Bemerkungen

Möglicherweise bietet IBM die in dieser Dokumentation beschriebenen Produkte, Services oder Funktionen in anderen Ländern nicht an. Informationen über die gegenwärtig im jeweiligen Land verfügbaren Produkte und Services sind beim IBM Ansprechpartner erhältlich. Hinweise auf IBM Lizenzprogramme oder andere IBM Produkte bedeuten nicht, dass nur Programme, Produkte oder Dienstleistungen von IBM verwendet werden können. An Stelle der IBM Produkte, Programme oder Dienstleistungen können auch andere ihnen äquivalente Produkte, Programme oder Dienstleistungen verwendet werden, solange diese keine gewerblichen oder anderen Schutzrechte der IBM verletzen. Die Verantwortung für den Betrieb der Produkte, Programme oder Dienstleistungen in Verbindung mit Fremdprodukten und Fremddienstleistungen liegt beim Kunden, soweit nicht ausdrücklich solche Verbindungen erwähnt sind.

Für in diesem Handbuch beschriebene Erzeugnisse und Verfahren kann es IBM Patente oder Patentanmeldungen geben. Mit der Auslieferung dieses Handbuchs ist keine Lizenzierung dieser Patente verbunden. Lizenzanfragen sind schriftlich an IBM Europe, Director of Licensing, 92066 Paris La Defense Cedex, France, zu richten. Anfragen an obige Adresse müssen auf englisch formuliert werden.

Trotz sorgfältiger Bearbeitung können technische Ungenauigkeiten oder Druckfehler in dieser Veröffentlichung nicht ausgeschlossen werden. Die Angaben in diesem Handbuch werden in regelmäßigen Zeitabständen aktualisiert. Die Änderungen werden in Überarbeitungen bekanntgegeben. IBM kann jederzeit Verbesserungen und/oder Änderungen an den in dieser Veröffentlichung beschriebenen Produkten und/oder Programmen vornehmen.

Verweise in diesen Informationen auf Web-Sites anderer Anbieter dienen lediglich als Benutzerinformationen und stellen keinerlei Billigung des Inhalts dieser Web-Sites dar. Das über diese Web-Sites verfügbare Material ist nicht Bestandteil des Materials für dieses IBM Produkt. Die Verwendung dieser Web-Sites geschieht auf eigene Verantwortung.

Werden an IBM Informationen eingesandt, können diese beliebig verwendet werden, ohne dass eine Verpflichtung gegenüber dem Einsender entsteht.

Lizenznehmer des Programms, die Informationen zu diesem Produkt wünschen mit der Zielsetzung: (i) den Austausch von Informationen zwischen unabhängigen, erstellten Programmen und anderen Programmen (einschließlich des vorliegenden Programms) sowie (ii) die gemeinsame Nutzung der ausgetauschten Informationen zu ermöglichen, wenden sich an folgende Adresse:

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

Die Bereitstellung dieser Informationen kann unter Umständen von bestimmten Bedingungen - in einigen Fällen auch von der Zahlung einer Gebühr - abhängig sein.

Die Lieferung des im Handbuch aufgeführten Lizenzprogramms sowie des zugehörigen Lizenzmaterials erfolgt im Rahmen der Allgemeinen Geschäftsbedingungen der IBM, der Internationalen Nutzungsbedingungen der IBM für Programmpakete oder einer äquivalenten Vereinbarung.

Alle in diesem Dokument enthaltenen Leistungsdaten stammen aus einer gesteuerten Umgebung. Die Ergebnisse, die in anderen Betriebsumgebungen erzielt werden, können daher erheblich von den hier erzielten Ergebnissen abweichen. Einige Daten stammen möglicherweise von Systemen, deren Entwicklung noch nicht abgeschlossen ist. Eine Garantie, dass diese Daten auch in allgemein verfügbaren Systemen erzielt werden, kann nicht gegeben werden. Darüber hinaus wurden einige Daten unter Umständen durch Extrapolation berechnet. Die tatsächlichen Ergebnisse können abweichen. Benutzer dieses Dokuments sollten die entsprechenden Daten in ihrer spezifischen Umgebung prüfen.

Informationen über Produkte anderer Hersteller als IBM wurden von den Herstellern dieser Produkte zur Verfügung gestellt, bzw. aus von ihnen veröffentlichten Ankündigungen oder anderen öffentlich zugänglichen Quellen entnommen. IBM hat diese Produkte nicht getestet und übernimmt im Hinblick auf Produkte anderer Hersteller keine Verantwortung für einwandfreie Funktion, Kompatibilität oder andere Ansprüche. Fragen zu den Leistungsmerkmalen von Produkten anderer Anbieter sind an den jeweiligen Anbieter zu richten.

Aussagen über Pläne und Absichten der IBM unterliegen Änderungen oder können zurückgenommen werden und repräsentieren nur die Ziele der IBM.

Diese Veröffentlichung enthält Beispiele für Daten und Berichte des alltäglichen Geschäftsablaufes. Sie sollen nur die Funktionen des Lizenzprogrammes illustrieren; sie können Namen von Personen, Firmen, Marken oder Produkten enthalten. Alle diese Namen sind frei erfunden, Ähnlichkeiten mit tatsächlichen Namen und Adressen sind rein zufällig.

#### COPYRIGHT-LIZENZ:

Diese Veröffentlichung enthält Beispielanwendungsprogramme, die in Quellsprache geschrieben sind. Sie dürfen diese Beispielprogramme kostenlos kopieren, ändern und verteilen, wenn dies zu dem Zweck geschieht, Anwendungsprogramme zu entwickeln, verwenden, vermarkten oder zu verteilen, die mit der Anwendungsprogrammierschnittstelle konform sind, für die diese Beispielprogramme geschrieben werden. Die in diesem Handbuch aufgeführten Beispiele sollen lediglich der Veranschaulichung und zu keinem anderen Zweck dienen. Diese Beispiele wurden nicht unter allen denkbaren Bedingungen getestet.

Kopien oder Teile der Beispielprogramme bzw. daraus abgeleiteter Code müssen folgenden Copyrightvermerk beinhalten:

© (Name Ihrer Firma) (Jahr). Teile des vorliegenden Codes wurden aus Beispielprogrammen der IBM Corp. abgeleitet. © Copyright IBM Corp. \_Jahr/Jahre angeben\_. Alle Rechte vorbehalten.

---

## Marken

Folgende Namen sind in gewissen Ländern Marken der International Business Machines Corporation.

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



Folgende Namen sind in gewissen Ländern Marken oder eingetragene Marken anderer Unternehmen:

Microsoft, Windows und Windows NT sind Marken oder eingetragene Marken von Microsoft Corporation.

Java und alle auf Java basierenden Marken und Logos sowie Solaris sind in gewissen Ländern Marken von Sun Microsystems, Inc.

Tivoli und NetView sind in gewissen Ländern Marken von Tivoli Systems Inc.

UNIX ist eine eingetragene Marke und wird ausschließlich von der X/Open Company Limited lizenziert.

Andere Namen von Unternehmen, Produkten oder Dienstleistungen können Marken anderer Unternehmen sein.