Informix Cloudscape Virtual Table Interface

In today's complex IS environment, it is rare for an application to stand alone. Instead, it must participate in a web of complex interactions with other applications, which may be based on a variety of database management systems, or provide data in a non-relational format. The Informix® Cloudscape™ DBMS is equipped with a virtual table interface that makes it easy to integrate data from other systems into a Cloudscape application.

Informix Cloudscape Background

The Informix Cloudscape product line provides a small-footprint, platform-independent, 100% Pure Java™ object-relational database management system (ORDBMS) that can be embedded in a Java™ application, or used in a classical client-server or Web-server mode. Cloudscape provides a standard SQL database that can be extended with Java objects. You can create a table in which a column structure is defined by a Java data type. The Java object is not stored as a binary large object (BLOB) as in other relational databases, but rather as a serialized object. The internal structure of the class is available to methods that run in the same standard Java Virtual Machine (JVM) as the DBMS. Cloudscape uses standard Java for application and database logic, which drastically reduces the learning curve for your engineers. Data access is via SQL-92E calls mediated by the standard JDBC protocol. With the addition of Cloudscape's industry leading advanced synchronization technology, a Cloudscape application can be deployed to a remote site and periodically resynchronized and upgraded via the Web.

Seamless Data Integration

The Informix Cloudscape Virtual Table Interface™ enables Java programmers to integrate data from a variety of sources as if the data arose from tables in a Cloudscape database. The Virtual Table Interface makes the output of any Java class that implements java.sql.ResultSet indistinguishable from a database table. You can run queries against the result set, join it to other tables, and even run a report writer against it. This allows external data, such as real-time data feeds, data from files, from other databases (even non-SQL databases), and any other tabular or non-tabular data sources to be seamlessly integrated. From the point of view of the front-end application, this entire medley of data looks like tables in a single database.

Cloudscape Virtual Table Interface

Using the Virtual Table Interface, any Java class that implements java.sql.ResultsSet is indistinguishable from a table: the rows and columns in the result set look exactly the same as those returned by the internal operations of the DBMS on a real table. A virtual table can appear in a SQL statement anywhere that a real table name can appear. The virtual table can be restricted using a SQL WHERE clause, it can be joined against another table, and it can act as the source of an INSERT statement for populating real tables.

This example shows how to join a Cloudscape table to a table from another database to get up-to-date stock prices for a locally held portfolio. In this example, `portfolio` is a real table in a Cloudscape database that holds information on the stocks that make up a particular portfolio. The Java method `vti.PriceQuery` makes a call to some external source of up-to-date stock pricing information in order to determine the current pricing of each of the stocks in the portfolio. The example shows a join of the real `portfolio` table against the output of the `vti.PriceQuery` call that updates the portfolio with the most recent prices.

```
UPDATE portfolio
SET closing_price =
    (SELECT close
    from NEW COM.cloudscape.vti.PriceQuery
    ('jdbc:odbc:stockdb',
    'SELECT close
    FROM stocks where symbol =
    portfolio.symbol')
```

Java Methods Provide Portability

The stock portfolio example shows some of the power of programming the Cloudscape DBMS in Java. Cloudscape stored procedures and triggers are also Java methods that access the database using standard Java Database Connectivity (JDBC) calls. Cloudscape supports the use of a Java method in a WHERE clause to qualify rows returned, or in a SELECT clause to manipulate the results of a search. This tight integration with Java makes it easy to migrate data-intensive methods from the application into the database for higher performance, even on a thin or disconnected client.

Summary

Cloudscape, with its class library footprint of about 2 MB, is agile enough to be readily embedded in a Java application and, because it is written in 100% Pure Java, none of the platform portability of the original Java application is lost. It runs on any Java VM that complies with at least the Java Development Kit (JDK) 1.1.6 specification and supports Java 2. Java applications embedding Cloudscape run on virtually any platform, regardless of hardware and operating system environment.

About Informix

Based in Menlo Park, California, Informix Corporation specializes in advanced information management technologies that help enterprises in the i.Economy get to market quickly, generate new revenue, build a unique strategic advantage, and solve their most complex business problems. Informix offers customers a complete software infrastructure for the Web that delivers highly scalable transaction processing, personalized content management, integrated business intelligence, full multimedia capabilities and complete e-commerce solutions. For more information, contact the sales office nearest you or visit our Web site at www.informix.com.