



Lotus Expeditor 6.1 Education

IBM® Lotus® Expeditor 6.1 Client for Desktop

Database lifecycle management

Lotus software



@business on demand software

© 2006 IBM Corporation

This presentation explains the Database Lifecycle Management in IBM Lotus Expeditor 6.1 Client for Desktop.

Goals

- Understand the database lifecycle management support provided in IBM Lotus Expeditor Client for Desktop

The goal of this presentation is to understand the database lifecycle management support provided in IBM Lotus Expeditor 6.1 Client for Desktop.

Agenda

- Database lifecycle management
- Extension points
- Summary

The agenda of this presentation is to explain the database lifecycle management capabilities which the client platform provides to you and details about the extension points supported by IBM Lotus Expeditor 6.1 Client for Desktop.

Section

Database lifecycle management

So, let's describe more details about database lifecycle management

Database lifecycle management

DB lifecycle

- The Database Lifecycle Management framework offers Lotus Expeditor applications the ability to interoperate with a relational database in a uniform and transparent manner (using the notion of managed data sources).
- It provides the platform administrator/developer:
 - ▶ A declarative way of specifying the required database structure
 - ▶ The ability to initially populate databases prior to client access
 - ▶ Supports migration from one database structure to another
 - ▶ The database population and migration is handled either by scripts or specialized database programming tasks.

The Database Lifecycle Management framework offers Lotus Expeditor applications the ability to interoperate with a relational database in a uniform and transparent manner (using the notion of managed data sources).

It provides the platform administrator or developer:

- A declarative way of specifying the required database structure
- The ability to initially populate databases prior to client access
- Supports migration from one database structure to another

The database population and migration is handled either by scripts or specialized database programming tasks that have the power to manipulate a database's contents.

Default managed datasource - Access

DB lifecycle

- By default, the platform includes a default datasource based on the Apache Derby database.
- Provides a reusable platform datasource for applications to store encrypted data on the Expeditor platform. Applications can provide their own schemas through virtual data sources in order to interact with the platform default datasource.
- To access it, perform a JNDI lookup using its defined JNDI name (jdbc/DerbyDS):

```
InitialContext context = new InitialContext();  
DataSource source = (DataSource) context.lookup("jdbc/DerbyDS");  
Connection connection = source.getConnection();
```

Included in the framework is a defined platform default datasource which is based on the Apache Derby embedded database. The reason this datasource exists is because many applications have a need to store encrypted data on the Lotus Expeditor platform so having a reusable platform datasource can allow applications to do so. Applications can provide their own schemas through virtual data sources in order to interact with the platform default datasource.

To access it, perform a JNDI lookup using its defined JNDI name (jdbc/DerbyDS) as shown in the slide.

Create data source - Extension point

DB lifecycle

- A Datasource is the primary definition of a database, and typically defines database access properties and locations
- Extension Point:
 - ▶ `com.ibm.rcp.database.core.datasource`
- Declares a managed datasource to the database infrastructure
- A datasource allows one to connect a database provider with associated schemas to manage lifecycle of the managed datasource.
- The datasource object can be retrieved from JNDI, via a `context.lookup()`
- Virtual datasources allow for the reuse of already defined datasources

The create data source extension point is used to declare a managed datasource to the database infrastructure. A datasource allows one to connect a database provider with associated schemas to manage lifecycle of the managed datasource. If declared, the data source will be created and associated with a `ManagedDatasourceObjectFactory`. From this, an instantiated version of the datasource object can be retrieved from JNDI, via a `context.lookup()`

Virtual datasources allow for the reuse of already defined datasources

Database provider – Extension point

DB lifecycle

- For each database that is being provided there is a corresponding database provider. Database providers manage the lifecycle of the database they represent. In particular they manage:
 - ▶ Creating a datasource instance
 - ▶ Creating a database
 - ▶ Checking for the existence of a database
 - ▶ Shutting down the database
- This extension point allows for the definition of new database providers.
- To use another database besides the platform's default Derby database, you must implement this extension.
- Once the extension is implemented, users of the platform may access the provided database via the database provider id.
- Extension Point:
 - ▶ `com.ibm.rcp.database.core.provider`

Another feature of the framework is the ability to define other database providers. By default, the platform includes a default datasource that is based on the Apache Derby database. If a platform administrator requires another database type, they have the power to define their own provider to be accessed by datasources platform-wide. A good example of when a new database provider would be defined is in the embedded space where Apache Derby is not the best database to use due to memory constraints.

The database provider extension point allows for the definition of new database providers. If one wishes to use another database besides the platform's default Derby database, one has to implement this extension. Once the extension is implemented, users of the platform may access the provided database via the database provider id.

For each database that is being provided there is a corresponding database provider. Database providers manage the lifecycle of the database they represent. In particular they manage:

- Creating a datasource instance
- Creating a database
- Checking for the existence of a database
- Shutting down the database

Database listener – Extension point

DB lifecycle

- Supports Listeners to hook into data sources
- Listeners are useful when you want to know the status of operations being applied to their datasource
- For example, you may want to inform the user of some progress if extensive schema processing on a datasource is about to occur.
- Extension Point:
 - ▶ `com.ibm.rcp.database.core.listener`

The database listener extension point allows one to define listeners to hook into data sources. Listeners are useful when one wants to know the status of operations being applied to their datasource. For example, one may want to inform the user of some progress if extensive schema processing on a datasource is about to occur.

Create schema – Extension point

DB lifecycle

- This extension point allows for the definition of schemas
- A schema is considered to be a set of create, populate and migrate tasks
- Can be valid SQL script or db task
- Schema lifecycle
 - ▶ Schemas are processed when a JNDI context lookup is performed on a datasource.
 - ▶ The managed datasource initiates the processing of the schemas when loading a datasource.
 - ▶ The schema manager gets called to decide whether or not process schemas based on what schemas are available.
- Extension Point:
 - ▶ `com.ibm.rcp.database.core.schema`

The create schema extension point allows for the definition of schemas. A schema is considered to be a set of create, populate and migrate tasks. Schema definitions do not necessarily correspond to a specific database vendor schema, but form the logical unit of a database that undergoes lifecycle events. Each schema is versioned independently, and may contain its own set of tasks to maintain its lifecycle.

Schemas are processed when a JNDI context lookup is performed on a datasource. The managed datasource initiates the processing of the schemas when loading a datasource. The schema manager gets called to decide whether or not to process schemas based on what schemas are available. To describe the lifecycle in detail:

- A schema is defined
- A JNDI lookup gets performed
- Schema processing is performed
 - The managed datasource calls the schema manager to process schemas for a given datasource
 - The schema manager checks if there are any later versions of the schema out there in order to do processing
 - The database catalog gets notified that a schema has been processed on a database

DB lifecycle management - Summary

DB lifecycle

- Interoperate with a relational database in uniform and transparent manner
 - ▶ Uses managed datasource, which simply refers to a database definition and its associated schemas
 - ▶ Provides the platform administrator / developer
 - A declarative way of specifying the required database structure
 - Ability to initially populate databases prior to client access
 - Migration from one database structure to another
- The platform includes a default datasource that is based on the Derby database.
- Other providers can be defined to be accessed by datasources platform-wide.
- API
 - ▶ com.ibm.rcp.database.core - Provides API's for managed datasource interaction
- Extension points
 - ▶ com.ibm.rcp.database.core.datasource – Declare a managed data source
 - ▶ com.ibm.rcp.database.core.listener - Define listeners to get status of operations being applied to a datasource
 - ▶ com.ibm.rcp.database.core.provider - Define new database providers
 - ▶ com.ibm.rcp.database.core.schema – Define schemas
- Target feature: Database Lifecycle Management
- Reference: Developer's Guide : Database Lifecycle Management

Database Lifecycle Management provides interoperability with a relational database in a uniform and transparent manner, while providing the platform administrators and developers with a declarative way of specifying the required database structure and the ability to initially populate that database prior to client access, as well as supporting migration from one database structure to another.

You can declare, populate, and migrate databases that your organization's Lotus Expeditor applications access using the Database Life Cycle Management component. The Database Life Cycle Management component allows Lotus Expeditor applications to interoperate with a relational database. As administrator, you can specify the required database structure and populate the database before client access. You can also migrate the database. With the Lotus Expeditor, a database can be a specific instance of any type of SQL database, tabular data source, spreadsheet, or flat file that can provide access to its content using a JDBC-compliant driver.

Trademarks, copyrights, and disclaimers

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM Lotus

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements or changes in the products or programs described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

Information is provided "AS IS" without warranty of any kind. THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (for example, IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products.

IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

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

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

© Copyright International Business Machines Corporation 2006. All rights reserved.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.



This concludes the presentation