GT2003Christian

Talk by Christian Haul at the 2003 GetTogether: Integrating Databases with Cocoon

Raw notes taken collaboratively during the talk.

Christian Haul discusses connection relational databases to Cocoon

He's mostly involved with ESQL and database actions in Cocoon

Ways to access databases in Cocoon

- Esql
- Transformer
- Actions
- Database Actions
- Object Relational Bridge

You can write JDBC code directly from your own components, or use Avalon connection Pools, (copy your DB driver to WEB-INF/lib, Set up a datasource in your cocoon.xconf) for access to Datasources for J2EE, Oracle, Informix, etc.

Incompatible API changes in JDBC between V2 and V3 - causes problems if one mixes versions. 2.1 reduces connection problems

Checking connections

- See Christian's slides later on, basically explains the possible ways to debug connection problems.
- Look at the beginning of the log, because the connection would fail at startup

Example tables

Examples will be based on users, users_groups and groups tables with simple relations. Similar to the samples included with the Cocoon distribution.

Using JDBC directly

Explains Java scenario to access JDBC data sources in Cocoon - gives you complete control but you do not leverage the framerwork.

ESQL

- XSP logicsheet, thin layer over JDBC
- You need to know about JDBC to make effective use of ESQL
- (Christian, showing some samples from Cocoon Dist)
- Warns people about SQL injection security problems (a simple Google search will help)
- Most everything (connection, query, etc) can be dynamic in ESQL.
- Paged display supported.
- Conditional tags (only do this if there were no results etc.)
- Good for fast Prototyping
- You can use ESQL for developing custom actions (XSPActions)

Good summary + advices slides on ESQL, keepers!

Chris recommends using ESQL for reading data only, don't do updates.

Caveat about ESQL page size, compile limit of 64 K Java class

SQL Transformer

- No compilation involved, so a bit slower
- Query can come from different sources (XInclude)
- Can be fed values from the Sitemap eg. using input modules
- Results are in XML (obviously) will be processed by a subsequent transformer
- Not cacheable, but pipeline could be made cachable up to this point
- Better SoC (than ESQL)
- No computations in the query
- difficult to use for modifying data

Database Actions

- separate file for metadata
- 'if' behaviour for the sitemap
- New 'modular' actions use input modules

- several Actions use SQL access for you behind the scenes, where you don't have to write SQL Queries
- results can be used via output modules
- update, delete, select
- auto increments
- multiple rows, tables
- easy prototyping
- can make the sitemap more complex

Object Relational Bridges

- Actions on steroids??
- OJB (example in CVS), Hibernate, Castor etc.
- Separate connection pool outside cocoon
- You generally get a Bean from one of these tools, these can be accessed via Castor, JXTemplate etc.
- How to use it
    - get a connection
    - make a query
    - get a Bean
    - display/modify the Bean
- Advantages
    - Integrate well with Woody + Flow
    - no need to write SQL queries
    - useable outside of Cocoon
    - meta data available through xDoclet
    - meta data obtainable from DBMS through eg. Druid
- Using from Flow
    - call any java code
    - access a persistence layer
    - Database.js in petstore sample
    - Legacy access to Actions

Make sure to have a look at Chris's summary pages on the various technologies (summary + advice), they're real good!

We should talk Christian into writing a summary of his presentation for our docs. A good summary of the various DB access technologies would be great (or is there one already).

Audience Questions

- How can caching be applied to databases?
    - Caching is not really possible at the database level, although individual pipelines can be configured to be cachable by the browser.
- Does paging in ESQL retrieve only necessary rows or all
    - This depends on the abilities of the JDBC driver. ESQL will try to retrieve the minimum number of rows. At most the row that should be skipped are retrieved in addition to the displayed rows.

Attachment: cocoon_and_databases_2.pdf