Google code

Google App Engine

Using JDO with App Engine

Java Data Objects (JDO) is a standard interface for storing objects containing data into a database. The standard defines interfaces for annotating Java objects, retrieving objects with queries, and interacting with a database using transactions. An application that uses the JDO interface can work with different kinds of databases without using any database-specific code, including relational databases, hierarchical databases, and object databases. As with other interface standards, JDO simplifies porting your application between different storage solutions.

The App Engine Java SDK includes an implementation of JDO 2.3 for the App Engine datastore. The implementation is based on DataNucleus Access Platform, the open source reference implementation for JDO 2.3.

See the Access Platform 1.1 documentation for more information about JDO. In particular, see "JDO Mapping" and "JDO API".

Setting Up JDO

To use JDO to access the datastore, an App Engine app needs the following:

- The JDO and DataNucleus App Engine plugin JARs must be in the app's `war/WEB-INF/lib/` directory.
- A configuration file named `jdoconfig.xml` must be in the app's `war/WEB-INF/classes/META-INF/` directory, with configuration that tells JDO to use the App Engine datastore.
- The project's build process must perform a post-compilation "enhancement" step on the compiled data classes to associate them with the JDO implementation.

If you are using the Google Plugin for Eclipse, these three things are taken care of for you. The new project wizard puts the JDO and DataNucleus App Engine plugin JARs in the correct location, and creates the `jdoconfig.xml` file. The build process performs the "enhancement" step automatically.

If you are using Apache Ant to build your project, you can use an Ant task included with the SDK to perform the enhancement step. You must copy the JARs and create the configuration file when you set up your project. See Using Apache Ant for more information about the Ant task.

Copying the JARs

The JDO and datastore JARs are included with the App Engine Java SDK. You can find them in the `appengine-java-sdk/lib/user/orm/` directory.

Copy the JARs to your application's `war/WEB-INF/lib/` directory.

Make sure the `appengine-api.jar` is also in the `war/WEB-INF/lib/` directory. (You may have already copied this when creating your project.) The App Engine DataNucleus plugin uses this JAR to access the datastore.

Creating the jdoconfig.xml File

The JDO interface needs a configuration file named `jdoconfig.xml` in the application's `war/WEB-INF/classes/META-INF/` directory. You can create this file in this location directly, or have your build process copy this file from a source directory.

Create the file with the following contents:

```
<?xml version="1.0" encoding="utf-8"?>
<jdoconfig xmlns="http://java.sun.com/xml/ns/jdo/jdoconfig"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="http://java.sun.com/xml/ns/jdo/jdoconfig">

    <persistence-manager-factory name="transactions-optional">
        <property name="javax.jdo.PersistenceManagerFactoryClass"

value="org.datanucleus.store.appengine.jdo.DatastoreJDOPersistenceManagerFactory"/>
```

Google Code offered in: English - Español - 日本語 - 한국어 - Português - Русский - 中文(简体) - 中文(繁體)