Whirlwind tour of Spring, Hibernate and Maven

Jim Schmidt
jjs@javautil.org

November 12, 2010

Contents

Part I

I

Chapter 1

Introduction

This book is a whirlwind tour of technologies that are .....

- Java
- Relational Databases
- Spring (Dependency Injection)
- Hibernate
- Maven

You've just been assigned to a new project group that uses some of the most popular development libraries and methodologies.

This is a series of lessons designed to quickly give you an introduction to Maven, Spring and Hibernate.

During the course of this the project specifications will be continually revised.

1.1 Project Specs

1.1.1 Initial Request

Design a program to scan a disk drive and store the contents of an MP3 files into an open ended number of formats and persistence mechanisms.

1.1.2 Support Multiple drives and systems

1.2 Command Line

I love eclipse, but everything should be supportable from the command line.

1.3 Systems Analyst

A disk drive? A variety of data persistent stores, the internet, the local area network

1.4 Steps

Download and install Maven.

Create your directories.

create new directory in Eclipse

Domain model

create project

in eclipse install maven plugin

enable dependence management (show screen shots)

We want to use the features available in version 1.5 of java

A lot of tabs, describe them

plugins

click add

need to add

Now add the following property

¡junit.version¿4.4¡/junit.version¿ because the build tool defaulted to a different version.

Working from the command line so that we can isolate out eclipse configuration problems.

maven from the command line

mvn eclipse:clean mvn eclipse:eclipse

configure in junit 4.4

1.5 installing library

We have decided to use jidlib now we must install it into our repository

Need to

Need to change to use java 1.6 but unable to configure through gui in eclipse need to edit pom.

Create a bean for the MP3 metadata

1.6 Design

Need a bean to hold the information from an MP3 file. TODO need to preserve tabs

TODO want to test that it works correctly

test the class

1.7 Create the database objects

Talk about what comes first database objects or domain objects.

1.8 Creating an argument Beans

see javautil-commandline alter the pom.

Cut and paste this into the pom.

```
<dependency>
                    <groupId>org.javautil</groupId>
                    <artifactId>javautil-commandline</artifactId>
                    <version>${org.javautil.commandline.version}</version>
 </dependency>
```

Create the bean Create the properties file, we will show you how to use to process command line arguments later.

configure a log4j.xml in src/test/resources and make sure it gets deployed.

How much have we tested? Code coverage.

CSV's as test and expected results.

1.9 Now what

Now we have MP3MetaDataExtractorTest how can we configure this another way?

1.10 Inject over factories

public interface Mp3Persistence

How the persistence came into being is not a matter of concern of the application that wishes to store data.

Is it writing to a socket? A database? A file? Why do I care?

Every one of these implementations require different constuctors and or setter methods before than can do their work

1.11 write to flat file as

1.12 Implementations

Document Object Model Streaming XML CSV Excel Workbook HTML JASON Serialized Java Object JDBC Hibernate Service Call Dataset EBCDIC

1.13 The canonical representation, the domain model

Chapter 2

Miscellaneous

2.1 Logging

2.2 Private Methods

2.3 TODO

show cascading inherited beans.

2.4 JDBC

now we add a dependency on javautil-jdbc to the project configuration, rebuild the eclipse project.

2.5 Creating The Schema

In the real world large amounts of data are stored in relational databases.

Without support, based on my experience, scripts are used to create database objects in production databases. These scripts are reviewed by Database Administrators.

The data generally outlives the applications that created them. This is a fairly safe statement.

Generally use scripts, for the purpose of partitions etc. Table comments, column comments.

Script runner

2.6 HenPlus

2.7 H2

2.8 Oracle

Chapter 3

Reverse Engineering

Reverse Engineering database tables consists of four files

3.1 Ant

build.xml

3.2 hibernate.cfg.xml

3.3 hibernate.reveng.xml

http://docs.jboss.org/tools/2.1.0.Beta1/hibernatetools/html/reverseengineering.html#custom-reve

3.4 Strategy

http://docs.jboss.org/tools/2.1.0.Beta1/hibernatetools/html/reverseengineering.html#custom-reve

3.5 Dialect

org.hibernate.cfg.reveng.dialect.JDBCMetaDataDialect contribute for oracle TODO

TODO sequences for primary keys

edit the build.xml file.

3.6 Dependencies

Build schema Build Reverse engineer strategy build maps

TODO now it gets ugly how do we define the location of all of the files necessary to make this work? Input into lib? Why can't we reference our maven repository.

Now the ant file has a dependency on the reverse strategy that has not yet been built.

After a clean the test database needs to be created which requires

does database username in hibernate.cfg.xml have to be in upper case for revenge? Otherwise connect but don't find any objects. http://mojo.codehaus.org/maven-hibernate3/hibernate3-maven-plugin/usage.html

The code in the repositories is different than the compiled code in Eclipse.

TODO get rid of /etc entries

building with passwords for the database in the project Build the mapping files from the database mvn hibernate3:hbm2hbmxml

hibernate3:hbm2dao

. hibernate3:hbm2doc

hibernate3:hbm2cfgxml Generates hibernate.cfg.xml hibernate3:hbm2hbmxml

hibernate3:hbm2java

hibernate3:hbmtemplate

hibernate3:help Display help information on

The package name should be specified as package ¡componentProperties¿ ¡drop¿true¡/drop¿ ¡package¿org.javautil.mp3.hibernate¡/package¿ ¡configurationfile¿/src/main/resource ¡/componentProperties¿

Not TODO packagename as specified in the documentation.

failure to specify a package name will result in no package.

http://stackoverflow.com/questions/1900234/maven-java-source-code-generation-for-hibernate

3.7 Transaction

Track artists. Musicians by song. Found Name vs

Part II

Hibernate

http://mojo.codehaus.org/maven-hibernate3/hibernate3-maven-plugin/componentproperties.html

configurationfile	String	src/main/resources/hibernate.cfg.xml
propertyfile	String	src/main/resources/database.properties
entityresolver	String	org.xml.sax.EntityResolver
namingstrategy	String	org.hibernate.cfg.DefaultNamingStrategy
persistenceunit	String	
packagename	String	
revengfile	String	
reversestrategy	String	
detectmanytomany	boolean	true
detectoptmisticlock	boolean	true
export	boolean	true
update	boolean	false
drop	boolean	false
create	boolean	true
outputfilename	String	
delimiter	String	
format	boolean	false
jdk5	boolean	false
ejb3	boolean	false
filepattern	String	package-name/class-name.ftl
template	String	

basedir/ exporterclass String scan-classes boolean false

should one have a separate class that is an InitializingBean that extends a bean

InitializingBean afterPropertiesSet

eliminate Spring depenendencies

Part III

Maven