

Hibernate Sales Data

Jim Schmidt
`jjs@javautil.org`

September 29, 2010

Contents

1	Introduction	5
1.1	Populating the Data	5
1.2	Problems	5
1.3	Configuration Files	5
1.3.1	hibernate.cfg.xml	5

Chapter 1

Introduction

Guide you through

Configuring your hibernate file. Configuring the spring configuration file. Populating the Data

1.1 Populating the Data

DataPopulatorTests

The problem with creating a database schema and then populating.

The SchemaExport utility wants a Configuration which comes from a hibernate.cfg.xml file

SessionFactory such as org.springframework.orm.hibernate3.LocalSessionFactoryBean are configure through an Spring application context

When trying to use hibernate in memory database a Connection cannot be shared, when closed the database goes away

1.2 Problems

currently using etc/hibernate.cfg.xml and application-context.xml must be configured the same

1.3 Configuration Files

1.3.1 hibernate.cfg.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE hibernate-configuration PUBLIC
"-//Hibernate/Hibernate Configuration DTD 3.0//EN"
"http://hibernate.sourceforge.net/hibernate-configuration-3.0.dtd">
<hibernate-configuration>
  <session-factory>
    <property name="hibernate.connection.driver_class">oracle.jdbc.driver.OracleDriver</property>
    <!-- Oracle type 4 URLs are of the form jdbc:oracle:thin@HHHHH:PPPP:SSSS
        where HHHHH is the host name or IP
            PPPP is the port probably 1521
            SSSS is the service name or SID
    -->
    <!--
    <property name="hibernate.connection.url">jdbc:oracle:thin:@localhost:1521:DEV11F</property>
    <property name="hibernate.connection.username">sales</property>
    <property name="hibernate.connection.password">areup</property>
    <property name="hibernate.dialect">org.hibernate.dialect.Oracle10gDialect</property>
    -->
    <!-- needed for oracle or reverse engineering will try for all users -->
    <!--
    <property name="hibernate.default_schema">SALES</property>
    -->
    <property name="hibernate.connection.driver_class">org.h2.Driver</property>
```

```

<property name="hibernate.connection.url">jdbc:h2:target/database/sales</property>
<property name="hibernate.dialect">org.hibernate.dialect.H2Dialect</property>
<property name="hibernate.connection.username">sales</property>
<property name="hibernate.connection.password"></property>
  </session-factory>
</hibernate-configuration>

```

1.3.2 Spring configuration

```

<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:aop="http://www.springframework.org/schema/aop"
xmlns:tx="http://www.springframework.org/schema/tx"
xsi:schemaLocation="http://www.springframework.org/schema/beans
    http://www.springframework.org/schema/beans/spring-beans-2.5.xsd
    http://www.springframework.org/schema/tx http://www.springframework.org/schema/tx/spring-tx-2.5.xsd
    http://www.springframework.org/schema/aop http://www.springframework.org/schema/aop/spring-aop-2.5.xsd">

<bean id="extractor" class="org.springframework.jdbc.support.nativejdbc.CommonsDbcpNativeJdbcExtractor" />

<!--
<bean id="datasource" class="org.apache.commons.dbcp.BasicDataSource"
destroy-method="close">
<property name="driverClassName" value="oracle.jdbc.driver.OracleDriver" />
<property name="url" value="jdbc:oracle:thin:@localhost:1521:DEV11F" />
<property name="username" value="sales" />
<property name="password" value="areup" />
<property name="initialSize" value="1" />
<property name="maxActive" value="10" />
</bean>
-->
<bean id="datasource" class="org.apache.commons.dbcp.BasicDataSource"
destroy-method="close">
<property name="driverClassName" value="org.h2.Driver" />
<property name="url" value="jdbc:h2:target/database/sales" />
<property name="username" value="sales" />
<property name="password" value="" />
</bean>
<!-- runtime hibernate config properties set here -->
<bean id="sessionFactory"
class="org.springframework.orm.hibernate3.LocalSessionFactoryBean">
<property name="dataSource" ref="datasource" />
<!--
property name="mappingLocations" value="classpath*/hbm/*.hbm.xml" /
-->
<property name="mappingLocations" value="classpath*:**/*.hbm.xml" />

<property name="hibernateProperties">
<value>

hibernate.transaction.factory_class=org.hibernate.transaction.JDBCTransactionFactory
hibernate.current_session_context_class=thread

<!-- DB dialect -->
hibernate.dialect=org.hibernate.dialect.Oracle10gDialect

<!-- 2nd level cache -->
<!--
hibernate.cache.provider_class=org.hibernate.cache.jbc2.JBossCacheRegionFactory
hibernate.cache.provider_class=org.hibernate.cache.TreeCacheProvider
-->

```

```
hibernate.cache.provider_class=org.hibernate.cache.EhCacheProvider

<!-- repeatable read isolation -->
hibernate.connection.isolation=4

<!-- 2nd level cache debugging -->
hibernate.generate_statistics=true
hibernate.cache.use_structured_entries=true

<!-- sql debugging -->
hibernate.show_sql=false
hibernate.format_sql=true

<!--
following causes sql to fail w/ derby DB
-->
hibernate.use_sql_comments=true

</value>
</property>
</bean>

<bean id="daoFactory" class="org.javautil.persist.hibernate.Persistence">
<property name="sessionFactory">
<ref bean="sessionFactory" />
</property>
<property name="extractor">
<ref bean="extractor" />
</property>
</bean>

<bean id="transactionManager"
class="org.springframework.orm.hibernate3.HibernateTransactionManager">
<property name="sessionFactory" ref="sessionFactory" />
</bean>

</beans>
```