HIBERNATE

Hibernate C3P0 Connection Pool Configuration Tutorial

SEPTEMBER 4, 2014 | LOKESH+ LEAVE A COMMENT

By default, Hibernate uses JDBC connections in order to interact with a database. Creating these connections is expensive—probably the most expensive single operation Hibernate will execute in a typical-use case. Since JDBC connection management is so expensive that possibly you will advise to use a pool of connections, which can open connections ahead of time (and close them only when needed, as opposed to "when they're no longer used").

Thankfully, Hibernate is designed to use a connection pool by default, an internal implementation. However, Hibernate's built-in connection pooling isn't designed for production use. In production, you would use an external connection pool by using either a database connection provided by JNDI or an external connection pool configured via parameters and classpath.

C3P0 is an example of an external connection pool. In this tutorial, we will learn to use it with hibernate.

Table of Contents

1) Maven dependencies

To configure c3p0 with hibernate, we need to add c3p0 and Hibernate's c3p0 connection

provider as dependencies in the pom.xml. Please note that the version of the hibernate-c3p0 dependency should match the Hibernate's compatible version.

Below configuration uses the latest versions of both i.e. hibernate as well as c3p0.

```
<dependency>
    <groupId>org.hibernate</groupId>
    <artifactId>hibernate-c3p0</artifactId>
    <version>4.3.6.Final</version>
</dependency>
<dependency>
    <groupId>c3p0</groupId>
    <artifactId>c3p0</artifactId>
    <version>0.9.1.2</version>
</dependency>
```

2) Configure C3P0 Connection Pool with Hibernate

The best part is that the configuration of C3P0 with hibernate is really very easy. **Just add any c3p0 property inside your hibernate.cfg.xml file** and that's it. e.g. add the following entry in hibernate.cfg.xml.

```
<property name="hibernate.c3p0.min_size">10</property>
```

Congratulations !! Connection pooling with C3P0 is now configured with your application's hibernate layer. You are good to start testing the things. Really easy, Isn't it?

A rather detailed configuration can be setup using following properties in hibernate.cfg.xml.

```
<property name="hibernate.c3p0.min_size">10</property>
<property name="hibernate.c3p0.max_size">20</property>
<property name="hibernate.c3p0.acquire_increment">1</property>
<property name="hibernate.c3p0.idle_test_period">3000</property>
<property name="hibernate.c3p0.max_statements">50</property>
<property name="hibernate.c3p0.timeout">1800</property>
```

You can find detailed information about about above configuration switches in **official documentation**.

3) Test connection pooling in runtime

Without C3P0 Connection Pool Configuration

Without C3P0 configured, if you see the debug logs of hibernate, you will see something like this:

```
DEBUG Configuration:1841 - Preparing to build session factory with filters :
```

```
WARN DriverManagerConnectionProviderImpl:93 - HHH000402: Using Hibernate bui
INFO DriverManagerConnectionProviderImpl:166 - HHH000401: using driver [org.
INFO DriverManagerConnectionProviderImpl:172 - HHH000046: Connection properi
INFO DriverManagerConnectionProviderImpl:180 - HHH000006: Autocommit mode: f
INFO DriverManagerConnectionProviderImpl:102 - HHH000115: Hibernate connecti
DEBUG DriverManagerConnectionProviderImpl:104 - Initializing Connection pool
...
...
...
EBUG JdbcTransaction:113 - committed JDBC Connection
DEBUG SessionFactoryImpl:1339 - HHH000031: Closing
DEBUG AbstractServiceRegistryImpl:406 - Implicitly destroying ServiceRegistr
INFO DriverManagerConnectionProviderImpl:281 - HHH000030: Cleaning up connec
```

Using C3P0 Connection Pool Configuration

After configuring C3P0 Connection Pool, you will be able to see in the logs that connections are now acquired from C3P0 Connection Pool itself.

```
DEBUG Configuration:1841 - Preparing to build session factory with filters :
 INFO C3P0ConnectionProvider:133 - HHH010002: C3P0 using driver: org.hsqldb.
 INFO C3P0ConnectionProvider:134 - HHH000046: Connection properties: {user=s
 INFO C3P0ConnectionProvider:137 - HHH000006: Autocommit mode: false
 INFO MLog:92 - MLog clients using log4j logging.
 INFO C3P0Registry:216 - Initializing c3p0-0.9.2.1 [built 20-March-2013 10:4
DEBUG DynamicPooledDataSourceManagerMBean:258 - MBean: com.mchange.v2.c3p0:t
DEBUG DynamicPooledDataSourceManagerMBean:253 - MBean: com.mchange.v2.c3p0:t
DEBUG DynamicPooledDataSourceManagerMBean:258 - MBean: com.mchange.v2.c3p0:t
 INFO AbstractPoolBackedDataSource:522 - Initializing c3p0 pool... com.mchan
 ...
 ...
 ...
 DEBUG ActiveManagementCoordinator:97 - C3P0Registry mbean unregistered.
DEBUG BasicResourcePool:1022 - Preparing to destroy resource: com.mchange.v2
DEBUG C3P0PooledConnectionPool:616 - Preparing to destroy PooledConnection:
DEBUG AbstractPoolBackedDataSource:477 - com.mchange.v2.c3p0.PoolBackedDataS
java.lang.Exception: DEBUG STACK TRACE for PoolBackedDataSource.close().
    at com.mchange.v2.c3p0.impl.AbstractPoolBackedDataSource.close(AbstractP
    at com.mchange.v2.c3p0.impl.AbstractPoolBackedDataSource.close(AbstractP
    at com.mchange.v2.c3p0.DataSources.destroy(DataSources.java:372)
    at com.mchange.v2.c3p0.DataSources.destroy(DataSources.java:348)
    at org.hibernate.c3p0.internal.C3P0ConnectionProvider.stop(C3P0Connectio
    at org.hibernate.service.internal.AbstractServiceRegistryImpl.stopServic
    at org.hibernate.service.internal.AbstractServiceRegistryImpl.destroy(Ab
    at org.hibernate.service.internal.AbstractServiceRegistryImpl.deRegister
    at org.hibernate.service.internal.AbstractServiceRegistryImpl.destroy(Ab
    at org.hibernate.internal.SessionFactoryImpl.close(SessionFactoryImpl.ja
    at com.howtodoinjava.demo.util.HibernateUtil.shutdown(HibernateUtil.java
    at com.howtodoinjava.test.TestHibernate.main(TestHibernate.java:22)
```

That's all for this easy but useful tutorial about configuring C3P0 Connection Pool with hibernate.

Happy Learning !!

Reference: http://www.mchange.com/projects/c3p0/#configuration