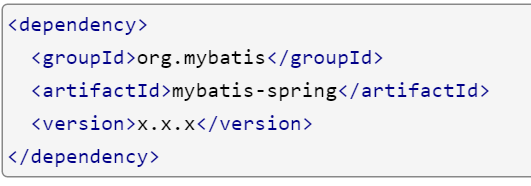
Spring + Mybatis

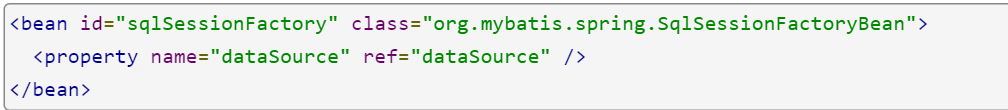
Quick start

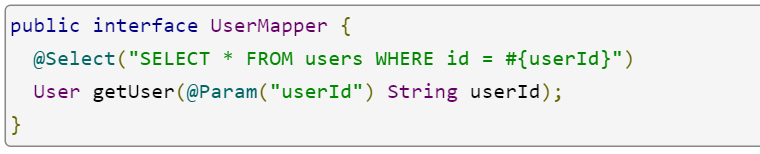
Maven dependency:



Two Mandatory stuff:

Two things must have when we start to use myBatis. SqlSessionFactory and at least one mapper interface.





After having these two things, we must integrate the mapper into spring as bean (there are 3 ways to do that). One basic way which is adopted by ahm is



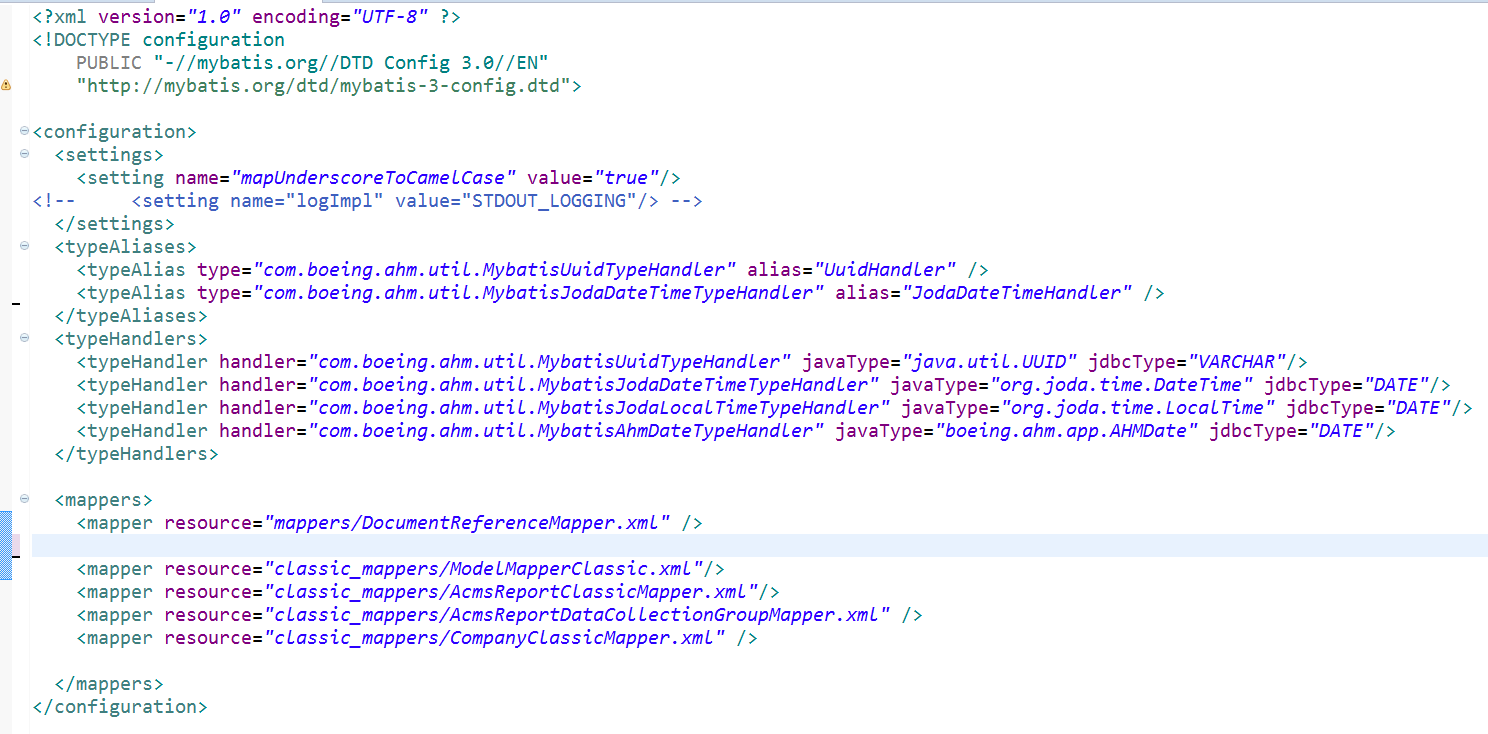
After that, we can inject the mapper directly to spring repository level of service where to define transaction.



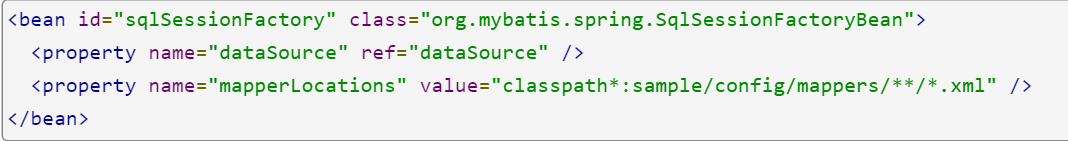
The above quick start gave the simplest case of integration of spring with mybatis. In realistic, we always need to define our own mybatis-configuration.xml file. These are the most common reasons.

1. Define the handlers of self-defined classes
2. Define the alias of classes with package name
3. Complicated mapper has both mapper interface and xml file

See the example below



Instead of declare the mapper one by one. We can define them in sqlSessionFactory.



The defined myBatis configuration can be loaded by sqlSessionFactory by property <configureLocation> and we can remove the mappers section from myBatis configuration file if we have mapperLocations in sqlSessionFactory.

When we have multiple db vendors, we can define sqlSessionFactory as below



As my understanding, myBatis mappers are the replacements of DAOs. They are defined as interface and also can go with xml file (for complicated queries). It is no need to create separated DAOs. The mappers can be injected into Manager or Services classes which have transaction declared. In AHM, I found we created DAOs which are the wrapper of mappers and DAOs don’t have interfaces defined. Therefore, they must be defined in the spring application xml file as <bean> since @Autowire can only be used by interfaces.

Reference:

MyBatis official website

<http://www.mybatis.org/spring/transactions.html>

May 16, 2017

Spring mybatis self learning

. MyBatis configuration in spring context

. declare domain class

. define mapper interface (simple queries defined in interface)

. define mapper.xml file for complicated query which defined in interface method

Good reference:

<http://www.thespringriver.com/simple-example-of-mybatis-java-maven-implementation-8-customized-type-handler/>

MyBatis mapper.xml syntax:

<http://www.mybatis.org/mybatis-3/sqlmap-xml.html#Parameters>