

MyBatis

Calvin edited this page on 14 Nov 2013 · 8 revisions

Overview

Mybatis比之于Spring JDBC的三大优势:

- 一可以联合查询多个表然后把结果转换为复杂对象，如Order->OrderItem->Product。見 [官方中文文档result map](#)
- 二是复杂SQL可以定义于XML文件，可自由换行缩进，且支持一些语法，简化SQL的拼装，見 [官方中文文档dynamic sql](#)
- 三還是在XML裡，可以根據不同的數據庫，定義不同語法的SQL，見 [官方中文文档dynamic sql](#)的末尾

因为都有了中文的手册了，Mybatis基本知识所以不再详细讲述。

Showcase里UserMapper.xml演示了很多小技巧，TeamMapper.xml做了一个算是非常复杂的对象转换演示:Team里有一个Scrum Master和一群开发。

MyBatis + Spring

值得留意是Showcase里Mybatis+Spring的结合，最新的官方默认做法已经很潮了，不再需要Config.xml，也不用在编写Dao实现(只需要DAO接口)。

而Showcase里对默认做法还做了两个合理化的改进，第一是DAO们不再叫Mapper而是直接叫回DAO，第二是Mapper.xml们不放在和Entity一样深的目录，放回一个浅的目录。

整个方案是这样的:

1. 在applicationContext.xml中，SqlSessionFactoryBean先指明type Aliases的扫描目录，将所有entity目录配进去，就省掉了用Config.xml的alias配置。
2. 在applicationContext.xml中，SqlSessionFactoryBean再指明mapper xml文件的目录，这样就不用将mapper文件放到与entity一样深的目录了。
3. 在applicationContext.xml中，MapperScannerConfigurer指明需要扫描的DAO接口的目录，动态创建其实现类。
如果DAO接口在多个目录中，简单的做法是创建一个MyBatisDAO的annotation，所有DAO都用这个annotation标识，然后用annotations属性指定它，那么package配到项目顶层package就可以了。
4. DAO接口，方法名必须与Mapper.xml中sql的名称一样。
5. Mapper.xml文件，nameSpace属性，必须是DAO接口的完整Class名。