Druid的了解与使用

连接池介绍

连接池是系统在初始化时，将数据库连接作为对象存储在内存中，当用户要访问数据库时，并非重建新连接，而是从池中取出一个已建立的空闲连接对象。使用完后，连接被返还到连接池中，以供后续请求访问使用。连接的建立、断开都由连接池自身来管理。而且，还可通过设置连接池的参数来控制池中连接的初始创建数、一次创建数、上下限数以及每个连接的最大使用次数、最大空闲时间等等，也可通过自身的管理机制来监视数据库当前连接数、使用情况等。

本设计利用Spring配置Druid多数据源，按照实际应用项目内部对应为水平分库，项目间对应为垂直分库，在service层面利用注解标注该class所有服务对应的数据库。相关配置详情示例：见附件九。

附件九：

Spring增加配置：

*<!--载入druid数据源-->*<**import resource="db\_config/druid\_db1.xml"**/>  
<**import resource="db\_config/druid\_db2.xml"**/>  
*<!--多数据源管理-->*<**bean id="multipleDataSource" class="assistplat.db\_police.DynamicDataSource"**>  
 *<!--<property name="defaultTargetDataSource" ref="datasource"/>-->* <**property name="targetDataSources"**>  
 <**map**>  
 <**entry key="datasource" value-ref="dataSource\_druid1"**/>  
 <**entry key="datasource1" value-ref="dataSource\_druid2"**/>  
 </**map**>  
 </**property**>  
</**bean**>

某一个数据库的druid连接配置：

*<!-- 加载properties文件-->*<**bean id="propertyConfigurer" class="org.springframework.beans.factory.config.PreferencesPlaceholderConfigurer"**>  
 <**property name="locations" value="classpath:properties/db\_properties/druid\_db1.properties"** />  
 <**property name="fileEncoding" value="UTF-8"**/>  
</**bean**>  
*<!-- druid 数据库连接池 -->*<**bean id="dataSource\_druid1" class="com.alibaba.druid.pool.DruidDataSource" init-method="init" destroy-method="close"**>  
 *<!-- 数据库基本信息配置 -->* <**property name="url" value="${url}"**/>  
 <**property name="username" value="${username}"**/>  
 <**property name="password" value="${password}"**/>  
 <**property name="driverClassName" value="${driver}"**/>  
 <**property name="filters" value="${filters}"**/>  
 *<!-- 最大并发连接数 -->* <**property name="maxActive" value="${maxActive}"**/>  
 *<!-- 初始化连接数量 -->* <**property name="initialSize" value="${initialSize}"**/>  
 *<!-- 配置获取连接等待超时的时间 -->* <**property name="maxWait" value="${maxWait}"**/>  
 *<!-- 最小空闲连接数 -->* <**property name="minIdle" value="${minIdle}"**/>  
 *<!-- 配置间隔多久才进行一次检测，检测需要关闭的空闲连接，单位是毫秒 -->* <**property name="timeBetweenEvictionRunsMillis" value="${timeBetweenEvictionRunsMillis}"**/>  
 *<!-- 配置一个连接在池中最小生存的时间，单位是毫秒 -->* <**property name="minEvictableIdleTimeMillis" value="${minEvictableIdleTimeMillis}"**/>  
 <**property name="validationQuery" value="${validationQuery}"**/>  
 <**property name="testWhileIdle" value="${testWhileIdle}"**/>  
 <**property name="testOnBorrow" value="${testOnBorrow}"**/>  
 <**property name="testOnReturn" value="${testOnReturn}"**/>  
 <**property name="maxOpenPreparedStatements" value="${maxOpenPreparedStatements}"**/>  
 *<!-- 打开 removeAbandoned 功能 -->* <**property name="removeAbandoned" value="${removeAbandoned}"**/>  
 *<!-- 1800 秒，也就是 30 分钟 -->* <**property name="removeAbandonedTimeout" value="${removeAbandonedTimeout}"**/>  
 *<!-- 关闭 abanded 连接时输出错误日志 -->* <**property name="logAbandoned" value="${logAbandoned}"**/>  
 *<!-- 监控数据库 -->  
 <!-- <property name="proxyFilters">  
 <list>  
 <ref bean="stat-filter" />  
 </list>  
 </property>-->  
 <!--启用公共监控数据库-->* <**property name="useGlobalDataSourceStat" value="true"**/>  
</**bean**>

### 参考：

druid多数据源配置与分布式事务

<http://blog.csdn.net/zheng0518/article/details/46459047>

Druid 配置多数据源

<http://blog.csdn.net/u010310183/article/details/52884909>

Spring 多个数据源配置，并实现数据源的动态切换

<http://blog.csdn.net/gaofuqi/article/details/46417281>

开启druid监控

<https://github.com/alibaba/druid/wiki/%E9%85%8D%E7%BD%AE_StatFilter>

<http://blog.csdn.net/szzt_lingpeng/article/details/50612258>

druid-monitor设计

<https://github.com/alibaba/druid/wiki/druid-monitor%E8%AE%BE%E8%AE%A1>

[Druid 介绍及配置](http://www.cnblogs.com/niejunlei/p/5977895.html) <https://www.cnblogs.com/niejunlei/p/5977895.html>

三大连接池对比

<http://blog.csdn.net/qq_34359363/article/details/72763491>