

Druid连接池配置优化

参考配置：

<https://github.com/alibaba/druid/wiki/DruidDataSource%E9%85%8D%E7%BD%AE%E5%B1%9E%E6%80%A7%E5%88%97%E8%>

从整体系统的角度，我们要考虑几个点，数据库连接数配置多少合适，针对空闲连接，网络异常的超时时间，如何高效复用连接，druid版本选择这几个方面来介绍。

1、如何设置连接池大小

一个请求的耗时为5ms，每个连接1秒能处理的请求数量是200个，如果需要在支持2000并发，那么理论上的连接数为10个。

minIdle：最小连接数

统计 QPS 和 请求响应时间的平均值，以此计算 minIdle，并设置 initialSize = minIdle。

推荐配置：5

maxActive：最大连接池数量，允许的最大同时使用中的连接数。

统计峰值时的 QPS 和此时的请求响应时间，以此计算 maxActive。实际需要比计算值略大一些。

配置 maxActive 千万不要好大喜多，虽然配置大了看起来业务流量飙升后还能处理更多的请求，但切换到DB视角会发现其实连接数的增多在很多场景下反而会减低吞吐量，一个非常典型的例子就秒杀，在更新热点数据时DB 需要加锁操作，这个时候再让更多的连接操作DB就有点像假日往高速上涌入的车辆，只会给DB添堵。

推荐配置：20，多数场景下 20 已完全够用，当然这个参数跟使用场景相关性很大，一般配置成正常连接数的3~5倍。

maxWait：从连接池获取连接的超时等待时间，单位毫秒。

需要注意这个参数只管理获取连接的超时。获取连接等待的直接原因是池子里没有可用连接，具体包括：连接池未初始化，连接长久未使用已被释放，连接使用中需要新建连接，或连接池已耗尽需等待连接用完后归还。这里有一个很关键的点是 maxWait 未配置或者配置为 0 时，表示不设等待超时时间。如果不配置maxWait，当突发大流量涌入时，造成连接池耗尽，所有新增的DB请求处于等待获取连接的状态中，在请求速度大于处理速度的情况下等待队列会越排越长，最终业务上的表现就是业务接口大量超时，流量越大造成实际吞吐量反而越低。

推荐配置：内网(网络状况好)800；网络状况不是特别好的情况下推荐大于等于 1200。

推荐配置表：

initialSize	5	初始化配置
minIdle	5	最小连接数
maxActive	20	最大连接数
maxWait	3000	获取连接超时时间（单位：ms）
testOnBorrow	false	获取连接检测
testOnReturn	false	归还连接检测
testWhileIdle	true	在获取连接后，确定是否要进行连接空闲时间的检查

不需要配置validationQuery，如果不配置的情况下会走ping命令，性能更高

JVM配置模板

如果内存不大，比如4核8G的机器，可以用默认的Parallel垃圾收集器，如果对停顿时间有一定要求，Jdk 1.8版本可以使用ParNew+CMS垃圾收集器组合，比如下面配置(参考JVM调优课程)：

```
1 -Xms3072M -Xmx3072M -Xmn2048M -Xss1M -XX:PermSize=256M -XX:MaxPermSize=256M -XX:SurvivorRatio=8
2 -XX:MaxTenuringThreshold=5 -XX:PretenureSizeThreshold=1M -XX:+UseParNewGC -XX:+UseConcMarkSweepGC
3 -XX:CMSInitiatingOccupancyFraction=92 -XX:+UseCMSCompactAtFullCollection -XX:CMSFullGCsBeforeCompaction=0
```

文档：05-电商项目JVM&Mysql线上整体优化及面试串讲

链接：[http://note.youdao.com/noteshare?](http://note.youdao.com/noteshare?id=a841dcd95385f9809227b68e5a5f1dc4&sub=CF5E3F0938FF4257A1EF9DDA2BEED63D)

id=a841dcd95385f9809227b68e5a5f1dc4&sub=CF5E3F0938FF4257A1EF9DDA2BEED63D