

60_技术挑战：引入 **Spring Cache** 后，如何解决列表缓存数据过多的问题？

儒猿架构官网上线，内有石杉老师架构课最新大纲，儒猿云平台详细介绍，敬请浏览

官网：www.ruyuan2020.com（建议 PC 端访问）

1、开篇

上节课带大家把 **Spring Cache** 缓存机制的代码测试了一下，当第一次访问教师列表以后，会将教师列表信息缓存起来，当再次访问的时候会从缓存中取出返回给消费者。这节课会思考在引入 **Spring Cache** 后，如何解决列表缓存数据过多的问题。今天课程的内容包括以下几个部分：

- 缓存过多会造成什么问题？
- 如何限制缓存记录的数量？

2、缓存过多会造成什么问题？

前面几节课针对 **Spring Cache** 缓存教师列表的问题展开了讨论，并且通过代码的方式将缓存的构想付诸实现。回到“首页”中的教师列表信息，之前是将所有教师列表的信息缓存下来，并且对搜索的教师信息进行缓存。如图 1 所示，实际上我们每次能够看到的教师信息是有限的，一个手机屏幕差不多可以看到 5 个教师的信息。

13:37

首页

...


🎯


请输入教师名称搜索

销量↓

评分↓

价格↓





钱老师


化学

销量73

评分4.98

价格 ¥ 205

专注于6-10岁儿童教育



赵六


物理

销量66

评分4.62

价格 ¥ 160

专注于6-10岁儿童教育



王五


英语

销量55

评分4.41

价格 ¥ 190

专注于6-10岁儿童教育



李四

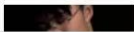
数学

销量44

评分4.88

价格 ¥ 200

专注于6-10岁儿童教育



孙老师

生物

🏠

我的优惠券

👤

首页

我的优惠券

我的

图 1 教师列表

如果说缓存所有的教师信息是不是太多了一点，多余的缓存信息有可能并不是消费者所需要的。那么怎么样的缓存方案是合理的呢？顺着这个思路往下想，能否根据每一屏的信息进行缓存，我们在设计 **Teacher** 实体类的时候就有 **page** 也就是页的概念，加入每页就显示 5 条信息，当消费者需要更多信息的时候再加载其余的信息。对于新增的信息再进行缓存是否效率会提高，假设数据量很大的情况，系统不用缓存所有的信息只有缓存前面几页的教师信息就足够了，如果还需要加载更多的教师信息就可以从数据库中加载。因此对于低频的信息就从数据库中获取，高频靠前的缓存信息是可以从缓存中获取的。

3、如何限制缓存记录的数量？

上面提到的缓存思路很简单：由于系统的能力有限、用户看到的数据也有限，因此不需要缓存所有的教师列表信息，只需要缓存其中一部分就可以了。基于这个思路我们将之前的教师列表信息的流程稍作调整。在调整之前对教师列表页面做这样的假设，在教师列表中按照 1 页 5 条记录，下滑列表的时候进行翻页操作，前面 2 页的数据对其进行缓存。从第 3 页的记录开始就不缓存，而是直接从数据库中读取。

如图 2 所示，在“从数据库获取教师列表”需要通过条件来判断是否将其加入到缓存中，这里需要满足两个条件：“缓存页数 ≤ 2 ”同时“教师名字为空”（没有进行教师姓名搜索）。

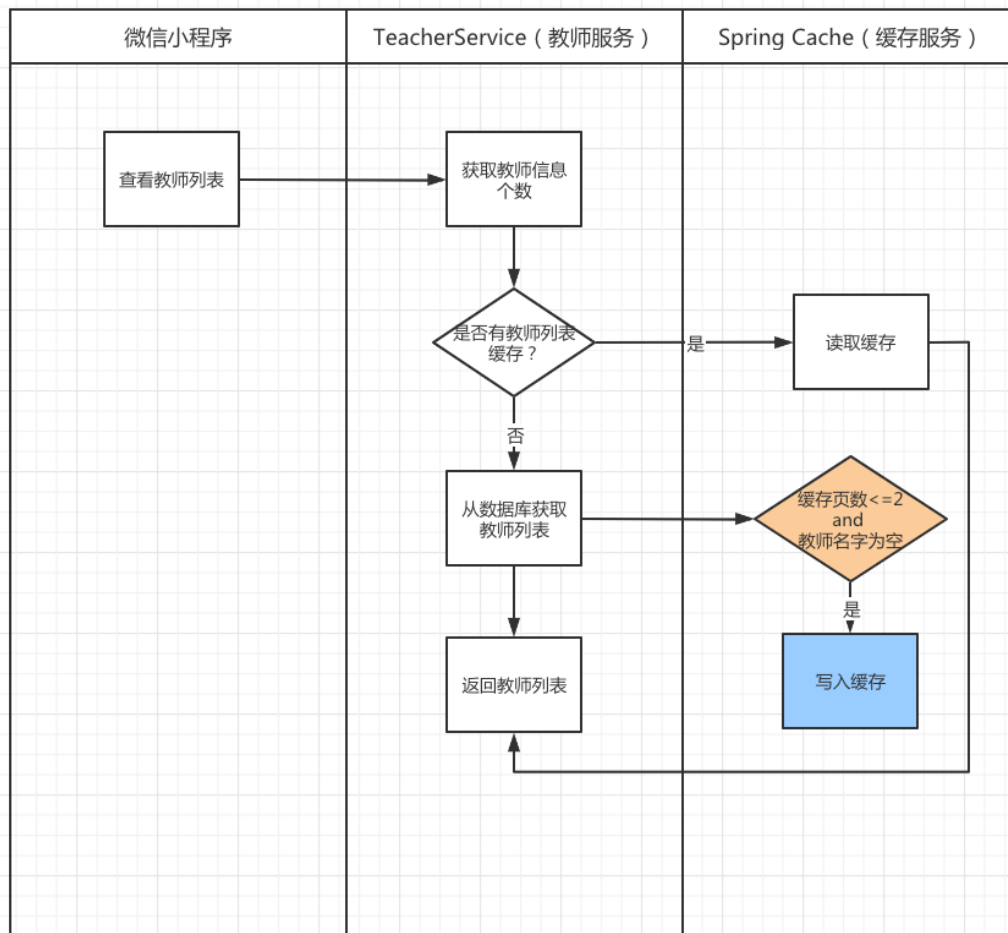


图 2 根据条件保存缓存信息

在调整流程以后将加入流程的条件进行了修改，值缓存前面 2 个教师列表页面，同时只对教师列表进行缓存针对教师搜索的列表就不进行缓存了。

4、总结

本节课思考当缓存数据过多情况下系统负荷会增大，然后通过分析教师列表的应用场景，发现不用缓存所有的教师信息，只需要按照一定规则缓存部分信息就可以满足消费者的需求。这个规则是值缓存前面 2 页的教师信息，只缓存教师列表不缓存教师搜索的信息。下节课我们会带大家修改缓存的代码。下期见，拜拜。