现象：每次对数据进行事务性的操作如插入、更新、删除后，同一查询时偶然会出现数据不一致的问题。

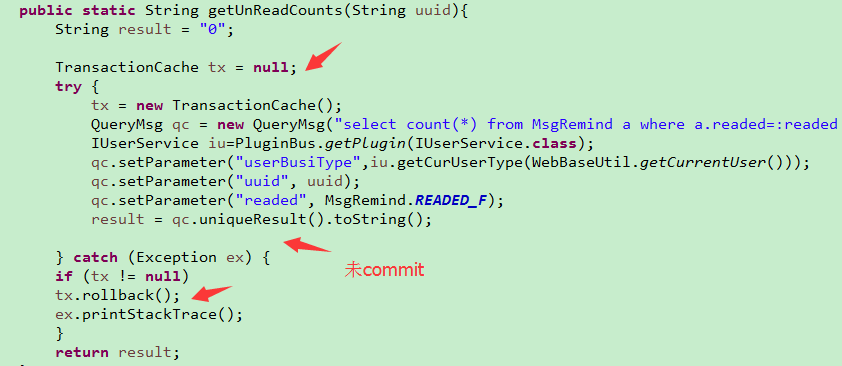
解决过程：

1. 怀疑是Memcached缓存导致：通过研究代码，发现在不使用Memcached缓存进行查询的时候也会导致，出现该问题的查询也并未使用到缓存查询。故排除。
2. 怀疑是Tomcat容器配置问题：通过部署到不同的Tomcat容器，如Tomcat7、Tomcat8、Tomcat9依然出现。故排除。
3. 怀疑是数据源的问题：EBS采用的是Alibaba Druid数据源，通过jndi的方式进行配置，将Druid1.2升级到1.3，1.4,1.5或者把数据源切换回JDBC的方式后问题依然出现。故排除。
4. 怀疑是MySQL自身BUG：通过资料查询到MySQL同时使用order by 和 limit进行查询时，在某些特定场景下会出现查询时数据不一致的问题，该BUG记录在Mysql BUG列表中<https://bugs.mysql.com/bug.php?id=32933>。但由于出现该bug的版本为5.0.52, 5.1.22-rc-community，EBS采用的是最新版本MySQL，应不会有此BUG。而且通过情景模拟在当前版本无法重现此BUG，故排除。
5. 通过仔细观测出现此问题时的Tomcat日志，发现线程池中当某一线程读取了hibernate\_msg.cfg.xml后（会员登录后首页上获取消息提醒时会发起一次http请求），在后续的查询中该线程就会出现查询不到数据库中最新数据的情况，发生数据不一致的问题。在此更为本质的现象上通过DEBUG日志观测到在Session的处理上似乎有问题！

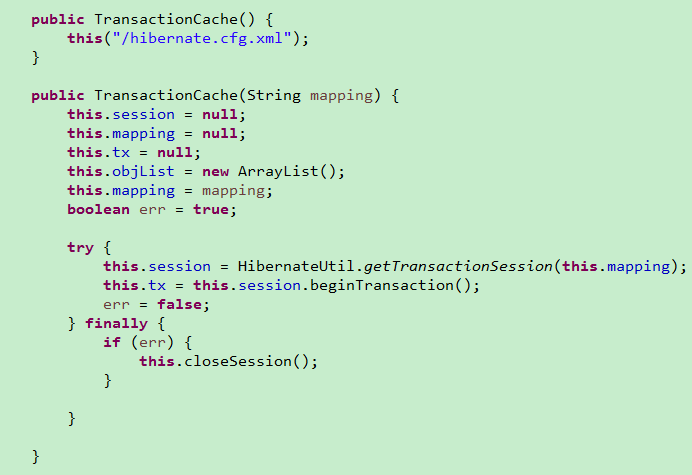
**摘要：**



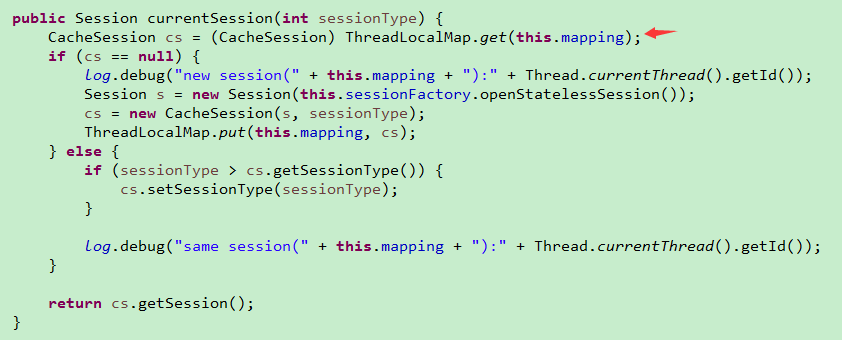
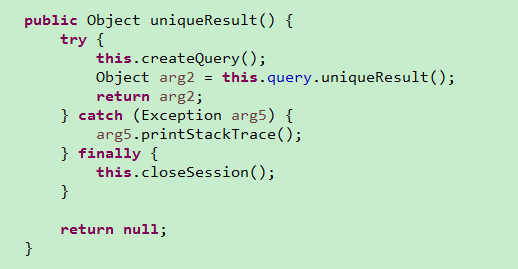
通过阅读别人写的代码发现：



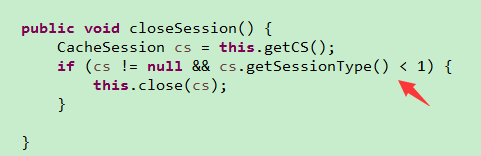
在数据库的查询中使用到了事务，通过分析源码，在new TransactionCache()时开启了一个Session，此Session是一个TransactionSession（中软的源码将Session分为三类：FunctionSession、TransactionSession、ThreadSession）



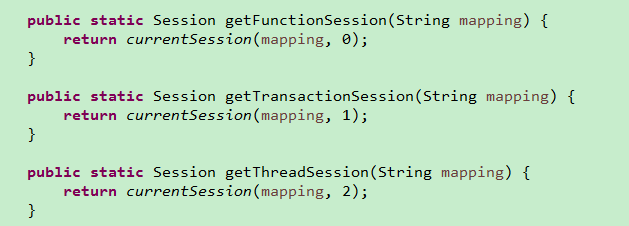
而后面使用new QueryMsg(“hql”).uniqueResult()进行查询时，首先获取当前Session，也就是new TransactionCache()时创建的这个TransactionSession，源码如下：



注意到在uniqueResult()方法最后的 finally块中有进行closeSession()，进一步查看其源码：

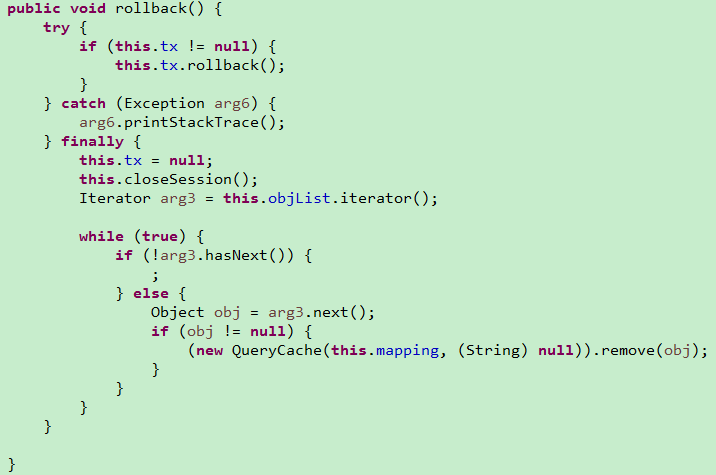
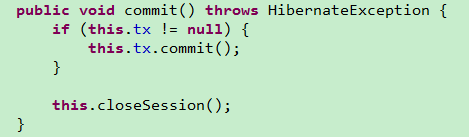


注意此时Session关闭的条件，再次查看源码：



所以最初创建的TransactionSession这里是没有关闭的。

那么正确的关闭时机应该在什么地方呢？答案是在事务提交或者回滚的时候。查看源码：



其实源码已经封装了一层，只需要开发人员在手动commit或者rollback的时候，就会自动关闭，不需要关注此时的session是getCurrentSession还是openStatelessSession。

所以归根到底，这是由于开发人员使用不当，对Hibernate Session机制理解不深入导致的一个隐蔽性很高的Bug.