**[《Redis Mysql 双写一致性问题》](https://www.cnblogs.com/25-lH/p/11270255.html)**

**一：序**

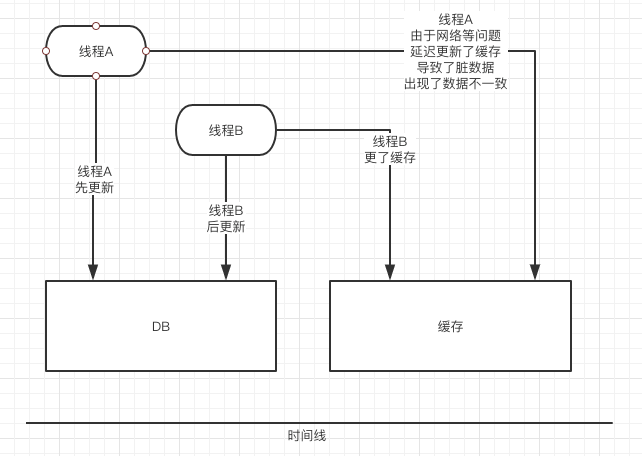
　　- 最近在对数据做缓存时候，会涉及到如何保证 数据库/Redis 一致性问题。

　　- 刚好今天来总结下 一致性问题 产生的问题，和可能存在的解决方案。

**二：（更新策略）-  先更新数据库，后更新缓存**

**- 产生的问题**

　　　　-



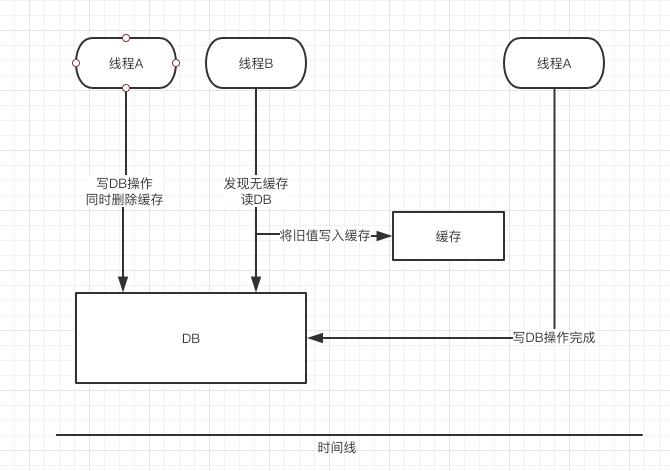
　　　　- 由上面流程图可知道，**请求A更新缓存应该比请求B更新缓存早才对，但是因为网络等原因，B却比A更早更新了缓存。**

　　　　- 这就导致了**脏数据**，因此不考虑 先更新数据库，后更新缓存 这个更新策略。

**三：（更新策略）-  先删除缓存，在更新数据库**

**- 产生的问题**

　　　　-



　　　　- 如果同时有一个请求A进行更新操作，另一个请求B进行查询操作。

　　　　- 就会导致不一致的情形出现。而且，如果不采用给缓存设置过期时间策略，该数据永远都是脏数据。

**四：（更新策略）-  先更新数据库，在删除缓存**

　　- FaceBook 也是采用这种方式。

　　- 当然，这种方式也会产生数据不一致问题。

　　　　- （1）缓存刚好失效

　　　　-（2）请求A查询数据库，得一个旧值

　　　　-（3）请求B将新值写入数据库

　　　　-（4）请求B删除缓存

　　　　-（5）请求A将查到的旧值写入缓存

　　- 前提是 **写操作耗时一定是低于 读操作的，在一般的条件下，这时不可能得。**

**五：小结**

　　- 这里只分析了平常可能想到的更新策略的分析。

　　- 其实，要**解决数据一致性的问题，还是要根据具体业务来具体判断。**

　　- 强一致性的，那么就需要悲观锁，使得一致。

　　- 同时还有**延时双写/延时双删 等策略。其实都是为了根据自身业务来进行的操作。**

　　- 知道了这些策略可能带来的问题，也就可以在合适的业务下选择合适的策略来满足我们的需求。