



## 什么是缓存雪崩、缓存穿透, 怎么解决?

- 缓存雪崩我们可以简单的理解为:由于原有缓存失效, 新缓存未到期间所有原本应该访问缓存的请求都去查询数据库了, 而对数据库 CPU 和内存造成巨大压力, 严重的会造成数据库宕机。从而形成一系列连锁反应, 造成整个系统崩溃。一般有三种解决办法:

\1. 一般并发量不是特别多的时候, 使用最多的解决方案是加锁排队。 \2. 给每一个缓存数据增加相应的缓存标记, 记录缓存的是否失效, 如果缓存标记失效, 则更新数据缓存。 \3. 为 key 设置不同的缓存失效时间。

- 缓存穿透是什么?

- 缓存穿透是指用户查询数据, 在数据库没有, 自然在缓存中也不会有。这样就导致用户查询的时候, 在缓存中找不到, 每次都要去数据库再查询一遍, 然后返回空(相当于进行了两次无用的查询)。这样请求就绕过缓存直接查数据库, 这也是经常提的缓存命中率问题。

- 怎么解决缓存穿透?

- 有很多种方法可以有效地解决缓存穿透问题, 最常见的则是采用布隆过滤器, 将所有可能存在的数据哈希到一个足够大的 bitmap 中, 一个一定不存在的数据会被这个 bitmap 拦截掉, 从而避免了对底层存储系统的查询压力。另外也有一个更为简单粗暴的方法, 如果一个查询返回的数据为空(不管是数据不存在, 还是系统故障), 我们仍然把这个空结果进行缓存, 但它的过期时间会很短, 最长不超过五分钟。通过这个直接设置的默认值存放到缓存, 这样第二次到缓存中获取就有值了, 而不会继续访问数据库。

- 缓存预热

- 缓存预热就是系统上线后, 将相关的缓存数据直接加载到缓存系统。这样就可以避免在用户请求的时候, 先查询数据库, 然后再将数据缓存的问题!用户直接查询事先被预热的缓存数据!

- 缓存更新

- 缓存更新除了缓存服务器自带的缓存失效策略之外(Redis 默认的有 6 中策略可供选择), 我们还可以

根据具体的业务需求进行自定义的缓存淘汰, 常见的策略有两种: (1)定时去清理过期的缓存;

(2)当有用户请求过来时, 再判断这个请求所用到的缓存是否过期, 过期的话就去底层系统得到新数据并更新缓存。

- 缓存降级

- 当访问量剧增、服务出现问题(如响应时间慢或不响应)或非核心服务影响到核心流程的性能时, 仍然需要保证服务还是可用的, 即使是有损服务。系统可以根据一些关键数据进行自动降级, 也可以配置开关实现人工降级。降级的最终目的是保证核心服务可用, 即使是有损的。而且有些服务是无法降级的(如加入购物车、结算)。