缓存雪崩、缓存穿透、缓存预热、缓存更新、缓存降级

# 缓存雪崩

由于原有缓存失效,新缓存未到期间,(例如设置缓存时采用了相同的过期时间,在同一时刻出现大面积的缓存过期)原本应该访问缓存的请求都去查询数据库了,对数据库CPU和内存造成巨大的压力,严重的会造成数据库宕机,从而形成一系列的连锁反应,造成整个系统崩溃.

大多数系统设计者用加锁或者队列的方式保证不会有大量的线程对数据库进行一次性读写.从而避免失效时大量的并发请求落到底层的存储系统上.将缓存失效时间散开,在原有的失效时间上增加一个随机值,1到5分钟,每一个缓存的过期时间的重复率就会降低.

# 缓存穿透

用户在查询数据时,,数据库中没有,自然缓存中也没有.用户在查询时,缓存中找不到,去数据中再查找一遍,然后返回空,请求绕过缓存直接查询数据库,缓存命中率的问题.

采用布隆过滤器,将所有的可能存在的数据hash到一个足够大的bitmap中,一个一定不存在的数据就会被这个bitmap拦截掉,从而避免了对底层存储系统的查询压力.

简单粗暴的方法,将结果为空的也进行缓存,设置的过期时间很短,第二次查询就在缓存中获取.

# 缓存预热

系统上线后,将相关的缓存数据直接加载到缓存系统中,,避免了请求时先查询数据库,将数据缓存的问题,用户直接查询事先被预热的缓存.

# 缓存更新

除了缓存服务器自带的缓存失效策略外,

定时清理过期的缓存,

用户请求过来时,在判断这个请求所用到的缓存是否过期,过期的就去底层系统得到新数据缓存.

# 缓存降级

当访问量剧增,服务出现问题(响应时间慢或不响应)或者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,即使是受损服务.系统可以根据一些关键数据进行自动降级,也可以配置实现人工降级.

降级是保证核心服务可用,有些服务是无法降级的(加入购物车,结算).

梳理降级的级别:

参考日志:

一般:有些服务偶尔因为网络抖动或者服务正在上线而超时,可以自动降级

警告:一段时间内有波动大于95,可以自动降级,人工降级,并发送告警.

错误:比如可用率低于90,数据库连接池被打爆了,访问量突然猛增到系统能承受的最大阈值,,此时可以自动降级,或者人工降级.

严重错误:因为特殊原因数据错误了,此时需要