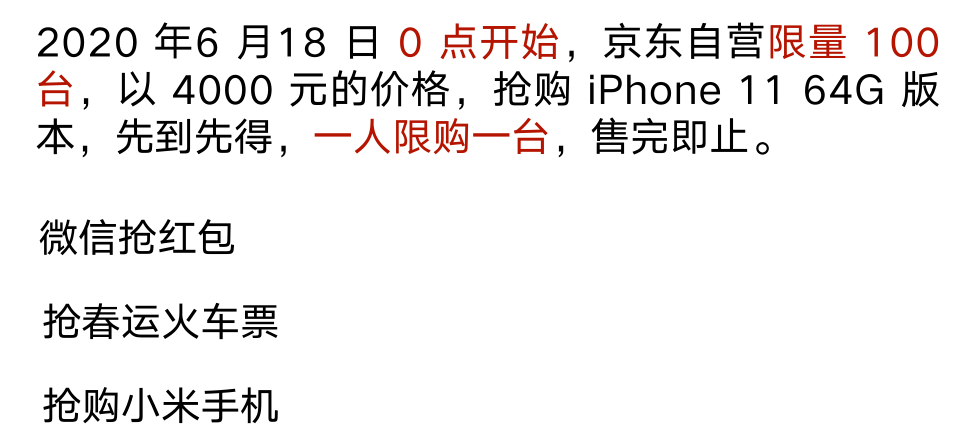
Scenario 场景：





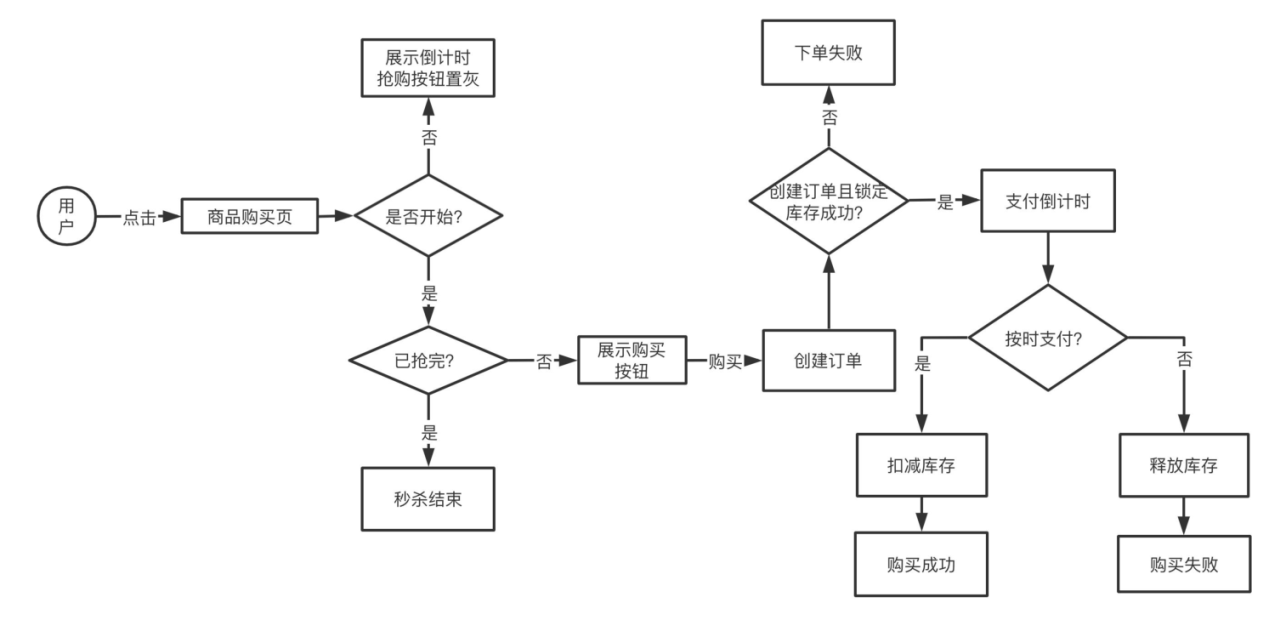
秒杀QPS分析：

平时每秒1000人访问该页面

秒杀时每秒几十万人访问该页面

QPS增加100倍以上。

秒杀系统的流程图：



秒杀系统需要解决的问题：

瞬时大流量高并发：服务器、数据库等能承载的QPS有限，如数据库一般是单机1000QPS，这就需要根据业务预估并发量。

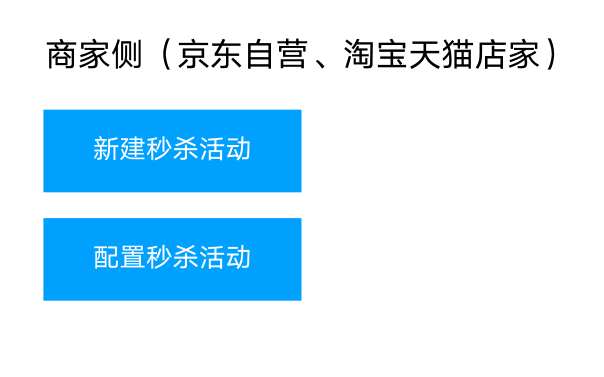
有限库存，不能超卖：库存是有限的，需要精准控制卖出数

黄牛恶意请求：使用脚本模拟用户购买，模拟出几十万请求，增加服务器压力

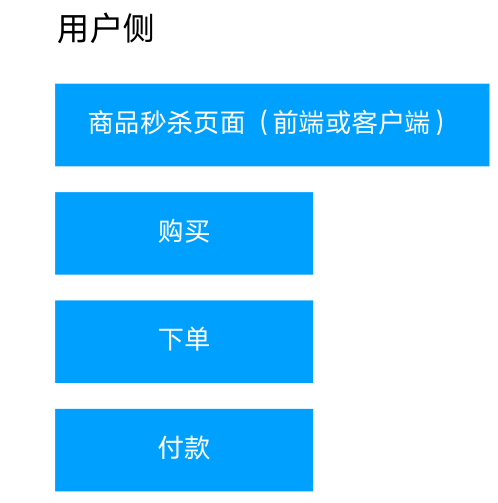
固定时间开启：

严格限购：一个用户，只能购买1个或M个。

需求拆解：商家侧（京东自营，天猫店家）



用户侧：



Service 服务：

老式的单体服务架构的缺点：



服务架构设计---微服务：



Storage 存储：

数据库表设计：

1. 商品信息表：



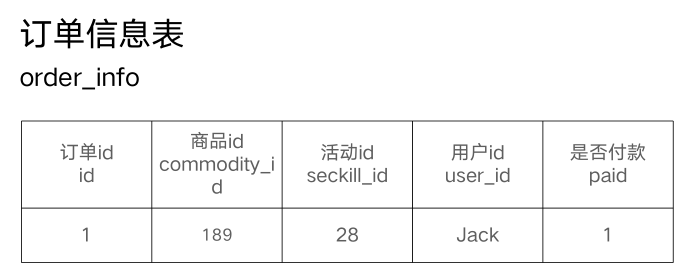
1. 秒杀活动表：



1. 库存信息表：



1. 订单信息表：



如何添加索引(index)?

秒杀操作--扣减库存

查询库存

Redis：一种主要将数据存储于内存中的非关系型的键值对数据库(Nosql)，但可以数据库持久化。

支持多种数据非关系型的数据结构：

1. 字符串/数字(STRING)
2. 哈希表(HASH)
3. 链表(LIST)
4. 集合(SET)
5. 有序集合(ZSET)