## 架构与解决方案篇

#### 秒杀类（以下方案不止秒杀，只要大流量都可以借鉴）

**高性能**

读性能（一致性要求不高的，结算页的库存，商品等信息）

本地缓存

Redis的主从，从扩展读

写性能

**只能写主，但是热点信息可以单独隔离出一套主从分片体系，在redis集群上层再加一层专门处理热点数据的分片，这样热点的单点插入问题就得以解决（热点的识别见热点识别专门方案）**

**安全性**

防刷规则，验证码，语音识别等

每步骤埋cookie，直接刷会报错

限流，lua，nginx，redis的集群限流

限购，作出限购规则

**一致性**

Redis的一致性可以用lua脚本去保证原子性

Redis中的lua写的redis怎么做回滚。Lua是用伪客户端执行lua中的命令，若一个命令成功，下个命令失败，怎么回滚

**热点识别**

通过nginx的日志做flink的流式统计，这个有十多秒的误差，不实时

**更实时的是通过线程的自我更新调用某个商品自增，进行线程级的实时统计**

**热点处理**

高性能中已经提及

可以写入日志（db或者redis），后续操作

**批量进行提交，事务批量提交（alisql中就可以做到sql的hint，进行批量的提交）**

#### 库存方向

一致性

Redis + lua

数据库的行级锁，一条语句 update set stock -1 where stock > 0

哪些步骤会导致超卖

库存在减的过程中，erp进行维护库存的redis

#### 分布式事务方向

Rocketmq事务mq 能做 库存扣减这些事情吗，为什么

因为库存扣减需要预占资源，若mq发送的时候，消费者去执行扣减，发现库存已经没了，那到底返回成功还是失败，毕竟主流程已经结束了

Rocketmq的事务只有在通知类的消息中有效，业务操作和通知一致性保证

Tcc和soga模型在seata中的应用，什么时候事务悬挂，什么场景会事务空回滚。

事务二阶段异步化是怎么回事

#### 设计方向

设计一个朋友圈系统

设计一个秒杀系统

设计一个网关系统

设计一个开放平台

设计一个限流降级平台

设计一个应急预案平台

#### 数据库平滑切库的问题

1： db如果是加字段或者改索引，create的语句，不停机的情况下，怎么执行

2：db扩容了，怎么实现平滑迁移，

日志追平法

双写法

双倍的秒级扩容法

#### 中间件高可用方面

怎么做db主从的状态型数据的不丢失的

流水型数据的db高可用

Zk脑裂处理

#### DB

高可用和扩容方案见上面，常考

回表是什么，怎么解决回表问题

一张表如何判断他的容量确实快影响到性能了

B+树为何三四层能装下百万级数据

慢sql的排查

DB的缓存的lru具体的增删流程，怎么防止buffer pool 污染

DB中的text和blob 怎么解决buffer pool 污染的

DB七种锁的使用场景，甚至写下sql，问你锁的范围（innodb那本书有）

Mysql的insert 数据和 查询数据，整个流程 包括 redo缓冲，change缓存，double write

Mysql的高可用这边不同版本，binlog拉取和执行线程是什么样子的

Mycat中的 分库分表的查询，join查询，groupby 等这些查询的原理

#### Redis

各种数据类型的数据结构，他们的增删改的算法复杂度

Redis6.0的多线程，是指什么环节用到了多线程

redis aof，rdb，aof中命令和aof怎么保证一致的

redis各种数据类型的应用场景，稍微举些复杂点的，比方说限购中的排序

redis中的布隆过滤器，是什么样的结构，用在什么场景

分布式锁用redis做和zk，etcd来做的区别，为什么

#### MQ

Kafka与rocketmq 究竟什么区别

怎么保证发送并仅消费一次

Rocketmq的事务，在提交的时候失败了，返回去查订单状态是成功的，那我这边消费者发现库存已经扣除完了，这种怎么提示用户。这种流程对吗为什么

怎么处理mq积压问题

哪些操作会导致kafka丢数据，或者重复消费