技术要点:

1.消息队列

2.消息队列的技术选型

3.引入消息队列后该如何保证其高可用性

4.为什么消息队列里消费到了重复的数据

5.为什么发到消息队列里面的数据不见了

6.如何保证从消息队列里拿到的数据按顺序执行

7.几百万消息在消息队列里积压了几个小时

8.如何设计消息队列中间件

9.分布式搜索引擎

10.他的架构师怎么设计的

11，分布式搜索引擎写入和查询的工作流程是什么样的

12.如何在几十亿数据场景下优化查询性能

13.怎么部署

14.分布式缓存的问题

15.redis的线程模型，单线程还能有很高的效率

16.redais数据类型，分别在那些场景下使用比较合适

17.redis的过期策略，手写一个LRU

18.怎么保证redis是高并发以及高可用的

19.怎么保证redis挂掉之后再重启数据可以惊醒恢复

20.redis cluster集群模式的原理

21应对缓存雪崩以及穿透问题

22. 保证缓存与数据可双写时的数据一致性

23.redis的并发竞争问题该如何解决

24.redis集群的部署架构是什么样的

25.为什么要把系统拆分成分布式的

26.为什么要用dubbo

27.dubbo的工作原理是啥

28.注册中心挂了可以继续通信吗

29.dubbo都支持那些通信协议以及序列化协议

30.dubbo支持那些负载均衡，高科应以及动态代理的策略

31.spi是啥思想

32.dubbo的spi机制是怎么玩的

33.基于dubbo如何做服务治理，服务降级以及重试

34.接口的幂等性该如何保证，如何不能重复扣款

35.分布式系统中的接口调用如何保证顺序性

36.如何设计一个类似dubbo的rpc框架，架构上该如何考虑

37.说说zookeeper一般有哪些使用场景

38.分布式锁是啥，对比下redais和zk两种分布式锁的优劣

39.说说你们的分布式session方案是啥，怎么做的

40.分布式事务方案，你们都咋做的，有啥坑

41.一般如何世界一个高并发的系统架构

42.怎么玩分库分表

43.如何系统不停机迁移到分库分表的

44如何设计可以动态扩容的分库分表方案

45.分库分表之后全局id咋生成

46.MYSQI读写分离的原理，主从同步延时咋解决的

47.如何设计高可用系统架构，限流，熔断，降级