

你好，欢迎来到分布式消息队列模块的加餐环节，本课时我将和你一起梳理面试中消息队列的高频考点，做到温故知新。

## 面试中如何考察消息队列

消息队列作为日常开发中应用最高频的基础组件之一，相关的问题自然也是面试中的常客。

在面试中对消息队列的考察方式，主要包括两种形式，一种是针对消息队列的相关理论，比如消息队列重复消费、消费幂等性、消息队列的可靠传输等；另一种考察方式是针对某个具体的消息队列中间件，考察组件应用的原理，实现方案和应用细节，比如常见的 Kafka、RabbitMQ、RocketMQ 等消息队列组件。

下面我梳理了一些面试中的高频问题，你可以对照这些问题，检测自己是否掌握了问题考察的内容，针对自己薄弱的环节，进行针对性地提高。

## 消息队列理论高频问题

对消息队列应用相关理论和设计的考察，面试官可以提出下面一系列的问题：

- 如何保证消息队列的高可用？
- 如何保证消息不被重复消费？
- 如何保证消费的时候是幂等？
- 如何保证消息的可靠性传输？
- 传输过程出现消息丢失了怎么办？
- 如何保证消息的顺序性？
- 如何解决消息队列的延时问题？
- 如何解决消息队列的过期失效问题？
- 消息队列满了以后该怎么处理？
-

有几百万消息持续积压几小时，应该怎么解决？

- 如果让你写一个消息队列，该如何进行架构设计？

可以看到，这方面的问题非常重视考察候选人对实际问题处理的经验，不过没有固定的答案。我在专栏里多次强调，授人以鱼不如授人以渔，关于分布式的方法论是最重要的。如果让你从零到一设计一个消息队列，该如何展开呢？你可以从分布式的基础理论出发，从数据存储的一致性，集群扩展结合我在分布式消息队列模块所讲解的内容，同时融入自己对系统架构的理解，最后形成自己的观点。

## 消息队列应用高频问题

面试中对具体某一种消息组件的考察，一般是候选人有过该组件的应用经验，重点是考察候选人对基础组件掌握的深度，出现问题后的解决办法等。

以 Kafka 为例，可以提出以下的问题：

- 描述一下 Kafka 的设计架构？
- Kafka、ActiveMQ、RabbitMQ、RocketMQ 之间都有什么区别？
- Kafka 消费端是否可能出现重复消费问题？
- Kafka 为什么会分区？
- Kafka 如何保证数据一致性？
- Kafka 中 ISR、OSR、AR 是什么？
- Kafka 在什么情况下会出现消息丢失？
- Kafka 消息是采用 Pull 模式，还是 Push 模式？
- Kafka 如何和 ZooKeeper 进行交互？

- Kafka 是如何实现高吞吐率的？

如果是 RocketMQ，很多问题都是类似的，可以从以下的问题出发进行考察：

- RocketMQ 和 ActiveMQ 有哪些区别？
- 为什么 RocketMQ 不会丢失消息？
- RocketMQ 的事务消息都有哪些应用？
- RocketMQ 是怎么保证系统高可用的？

这些问题中一部分可以在专栏中找到思路，但大部分的问题还要靠你在平时多积累与思考，比如消息队列的高可用，你可以多机器部署，防止单点故障；主从结构复制，通过消息冗余防止消息丢失；消息持久化，磁盘写入的 ACK 等角度进行分析。

今天的内容就到这里了，也欢迎你留言分享自己的面试经验，和大家一起讨论。

---

## 精选评论