

1f3)

这样一个简陋的队列进程，其实就是所谓的**消息队列**。
而像 A 服务这样负责发数据到消息队列的角色，就是**生产者**，像 B 服务这样处理消息的角色，就是**消费者**。

![生产者和消费者](https://p3-juejin.byteimg.com/tos-cn-i-k3u1fbpfcp/0bc7737ebeae463481c9185b37549c2b~tplv-k3u1fbpfcp-jj-mark:3024:0:0:0:q75.awebp#?w=1920&h=1080&s=263167&e=jpg&b=f0f1f3)

但这个消息队列属实过于简陋，像什么高性能，高扩展性，高可用，它是一个都不沾。
我们来看下怎么优化它。

高性能

B 服务由于性能较差，消息队列里会不断堆积数据，为了提升性能，我们可以扩展更多的消费者，这样消费速度就上去了，相对的我们就可以增加更多生产者，提升消息队列的吞吐量。

![增加生产者和消费者](https://p3-juejin.byteimg.com/tos-cn-i-k3u1fbpfcp/29e2446e29364f2eb4bd5cdd24337544~tplv-k3u1fbpfcp-jj-mark:3024:0:0:0:q75.awebp#?w=1920&h=1080&s=347992&e=jpg&b=f0f1f3)

随着生产者和消费者都变多，我们会发现它们会同时争抢同一个消息队列，抢不到的一方就得等待，这不纯纯浪费时间吗！

有解决方案吗？有！

首先是对消息进行分类，每一类是一个 **topic**，然后根据 topic 新增队列的数量，生产者将数据按 topic 投递到不同的队列中，消费者则根据需要订阅不同的 topic。这就大大降低了 topic 队列的压力。

![多个topic](https://p3-juejin.byteimg.com/tos-cn-i-k3u1fbpfcp/8e77b424c487474ebe94fc0cd8cffd4c~tplv-k3u1fbpfcp-jj-mark:3024:0:0:0:q75.awebp#?w=1920&h=1080&s=271794&e=jpg&b=f0f1f3)

但单个 topic 的消息还是可能过多，我们可以将单个队列，拆成好几段，每段

就是一个 **partition 分区**，每个消费者负责一个 partition。
这就大大降低了争抢，提升了消息队列的性能。

![partition](https://p3-juejin.byteimg.com/tos-cn-i-k3u1fbpfcp/2092d101ee244186ada5dfdfc929f35d~tplv-k3u1fbpfcp-jj-mark:3024:0:0:0:q75.awebp#?w=1920&h=1080&s=44) ker 是不是跪了，某些消费组消费到哪了。

![加入ZooKeeper](https://p3-juejin.byteimg.com/tos-cn-i-k3u1fbpfcp/81bb50e12d3848cda13c5668fe6beaf7~tplv-k3u1fbpfcp-jj-mark:3024:0:0:0:q75.awebp#?w=1920&h=1080&s=487962&e=jpg&b=eff0f2)

Kafka 是什么

好了，到这里，当初那个简陋的消息队列，就成了一个高性能，高扩展性，高可用，支持持久化的超强消息队列，没错，它就是我们常说的消息队列 **Kafka**，上面涉及到各种概念，比如 partition 和 broker 什么的，都出自它。

![Kafka是什么](https://p3-juejin.byteimg.com/tos-cn-i-k3u1fbpfcp/f6ee8fbcf393433e818f6af59cb05646~tplv-k3u1fbpfcp-jj-mark:3024:0:0:0:q75.awebp#?w=1920&h=1080&s=560438&e=jpg&b=eff0f2)

kafka 的应用场景

消息队列是架构中最常见的中间件之一，使用场景之多，堪称万金油！
比如上游流量忽高忽低，想要 **削峰填谷**，提升 cpu/gpu 利用率，用它。
又比如系统过大，消息流向盘根错节，想要拆解组件，**降低系统耦合**，还是用它。
再比如秒杀活动，请求激增，想要 **保护服务** 的同时又尽量不影响用户，还得用它。
当然，凡事无绝对，方案还得根据实际情况来定，做架构做到最后，都是在做 **折中**。

![Kafka的应用场景](https://p3-juejin.byteimg.com/tos-cn-i-k3u1fbpfcp/0688e9a92016455a826f5a69adc133a7~tplv-k3u1fbpfcp-jj-

mark:3024:0:0:0:q75.awebp#?w=1920&h=1080&s=319519&e=jpg&b=eef0f3)

总结

--

* kafka 是消息队列，像消息队列投递消息的是生产者，消费消息的是消费者。增加生产者和消费者的实例个数可以提升系统吞吐。多个消费者可以组成一个消费者组，不同消费者组维护自己的消费进度，互不打搅。

* kafka 将消息分为多个 topic，每个 topic 内部拆分为多个 partition，每个 partition 又有自己的副本，不同的 partition 会分布在不同的 broker 上，提升性能的同时，还增加了系统可用性和可扩展性。

原文链接: <https://juejin.cn/post/7361326309995511862>