

图文 89 为什么要给RocketMQ增加消息限流功能保证其高可用性?

392 人次阅读 2020-02-05 07:49:15

详情 评论



狸猫技术

进店逛

相关频道



从 0 开  
间件实站  
已更新1



继《从零开始带你成为JVM实战高手》后，救火队长携新作再度出山，重磅推荐：

(点击下方蓝字试听)

[《从零开始带你成为MySQL实战优化高手》](#)

今天是我们这个专栏老版本大纲的最后两篇文章，我们依然讲点轻松点的，内容也不会太多，让大家有一个缓冲，明天开始我们就要进入专栏全新增加的30讲的RocketMQ源码分析的部分，到时候全称都是源码分析的硬干货，比较烧脑，大家做好准备。

今天先给大家讲第一个内容，就是RocketMQ的限流，这个功能我们文章里不会带着大家去做出来，但是会给大家讲一下为什么要给RocketMQ增加限流功能保证其高可用性呢？其实本质上来说，限流功能就是对系统的一个保护功能。

大家可以思考一个场景，假如有一个新来的工程师，因为没搞明白RocketMQ到底应该怎么使用，结果代码里写出了一个大bug，导致他负责的系统拼命的往MQ里不停的写入消息，可能一下子流量激增会导致MQ出现故障。

下面的代码就是曾经真实我们见过的一个工程师写出来的代码，大家可以看看：

```
try {  
    // 业务代码  
    producer.send(message);  
} catch (Exception e) {  
    while (true) {  
        producer.send(message);  
    }  
}
```

上面是简化以后的代码，实际上当时那段代码里混杂了很多的业务逻辑，但是当时出现的一个问题，就是业务代码报错了然后进入了catch代码块，结果那个工程师居然在catch代码块里写了一个while死循环，不停的发送消息。

而且上述系统当时是部署在10多台机器上的一个系统，所以相当于10多台机器都频繁的开足CPU的马力，拼命的往MQ里写消息，瞬间就导致MQ集群的TPS飙升，里面混入了大量的重复消息，而且MQ集群都快挂了。

所以针对这种场景，其实站在MQ的角度而言，你是没有办法去避免各种系统用上述的白痴方法来使用你的，毕竟公司大了，什么样的人都有，什么样的情况都可能出现，所以对MQ而言，你就必须去改造一下开源MQ的内核源码。

在接收消息这块，必须引入一个限流机制，也就是说要限制好，你这台机器每秒钟最多就只能处理比如3万条消息，根据你的MQ集群的压测结果来，你可以通过压测看看你的MQ最多可以抗多少QPS，然后就做好限流。

一般来说，限流算法可以采取**令牌桶算法**，也就是说你每秒钟就发放多少个令牌，然后只能允许多少个请求通过。关于限流算法的实现，不在我们的讨论范围内，大家可以自己查阅一下资料，也并不是很难。

我们这里主要是给大家讲一下，很多互联网大厂其实都会改造开源MQ的内核源码，引入限流机制，然后只能允许指定范围内的消息被在一秒内被处理，避免因为一些异常的情况，导致MQ集群挂掉。

End

专栏版权归公众号**狸猫技术窝**所有

未经许可不得传播，如有侵权将追究法律责任

---

狸猫技术窝精品专栏及课程推荐：

[《从零开始带你成为JVM实战高手》](#)  
[《21天互联网Java进阶面试训练营》（分布式篇）](#)  
[《互联网Java工程师面试突击》（第1季）](#)  
[《互联网Java工程师面试突击》（第3季）](#)

---

#### 重要说明：

如何提问：每篇文章都有评论区，大家可以尽情留言提问，我会逐一答疑

如何加群：购买狸猫技术窝专栏的小伙伴都可以加入狸猫技术交流群，一个非常纯粹的技术交流的地方

具体加群方式，请参见目录菜单下的文档：《付费用户如何加群》（[购买后可见](#)）

