

Apache Kafka

发布 & 订阅

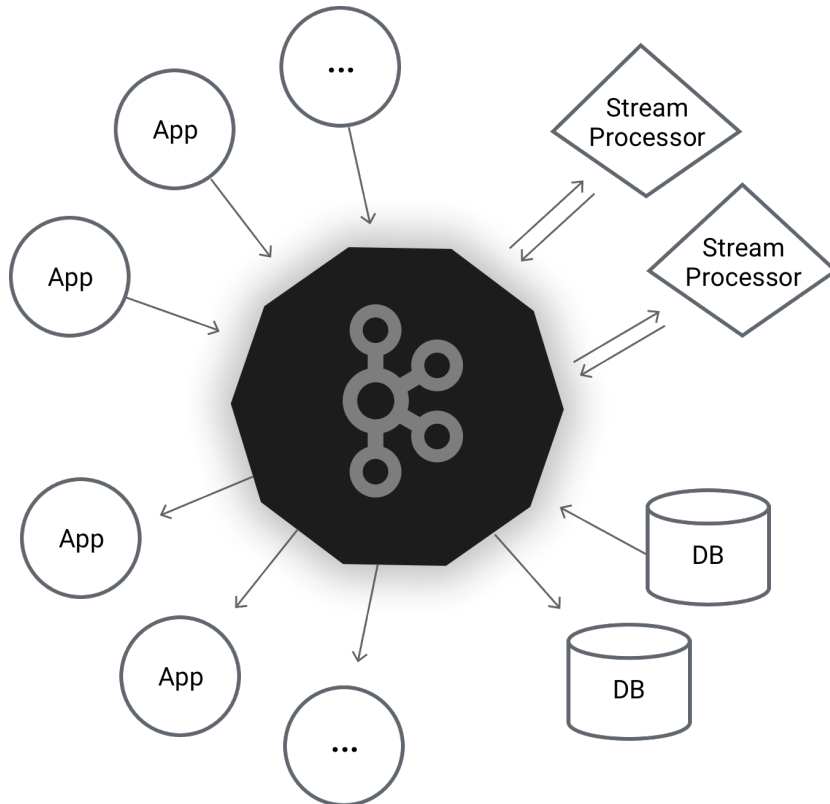
数据流，如消息传递系统

处理

高效并实时

存储

数据流安全地在分布式集群中复制存储



kafka是用于构建实时数据管道和流应用程序。具有横向扩展，容错，wicked fast（变态快）等优点，并已在成千上万家运行。

简单说明什么是kafka

Apache kafka是消息中间件的一种，我发现很多人不知道消息中间件是什么，在开始学习之前，我这边就先简单的解释一下什么是消息中间件，只是粗略的讲解，目前kafka已经可以做更多的事情。

举个例子，生产者消费者，生产者生产鸡蛋，消费者消费鸡蛋，生产者生产一个鸡蛋，消费者就消费一个鸡蛋，假设消费者消费鸡蛋的时候噎住了（系统宕机了），生产者还在生产鸡蛋，那新生产的鸡蛋就丢失了。再比如生产者很强劲（大交易量的情况），生产者1秒钟生产100个鸡蛋，消费者1秒钟只能吃50个鸡蛋，那要不一会，消费者就吃不消了（消息堵塞，最终导致系统超时），消费者拒绝再吃了，“鸡蛋”又丢失了，这个时候我们放个篮子在它们中间，生产出来的鸡蛋都放到篮子里，消费者去篮子里拿鸡蛋，这样鸡蛋就不会丢失了，都在篮子里，而这个篮子就是“kafka”。

鸡蛋其实就是“数据流”，系统之间的交互都是通过“数据流”来传输的（就是tcp、http什么的），也称为报文，也叫“消息”。

消息队列满了，其实就是篮子满了，“鸡蛋”放不下了，那赶紧多放几个篮子，其实就是kafka的扩容。

各位现在知道kafka是干什么的了，它就是那个“篮子”。

kafka名词解释

后面大家会看到一些关于kafka的名词，比如topic、producer、consumer、broker，我这边来简单说明一下。

producer：生产者，就是它来生产“鸡蛋”的。

consumer：消费者，生出的“鸡蛋”它来消费。

topic：你把它理解为标签，生产者每生产出来一个鸡蛋就贴上一个标签（topic），消费者可不是谁生产的“鸡蛋”都吃的，这样不同的生产者生产出来的“鸡蛋”，消费者就可以选择性的“吃”了。

broker：就是篮子了。

大家一定要学会抽象的去思考，上面只是属于业务的角度，如果从技术角度，topic标签实际就是队列，生产者把所有“鸡蛋（消息）”都放到对应的队列里了，消费者到指定的队列里取。

怎么样才算真正的学会kafka

最近面试发现，很多人用过kafka，但是没人了解原理，我们可是很注重原理的(PS:要不然怎么知道你真的会呢)。

- kafka节点之间如何复制备份的？
- kafka消息是否会丢失？为什么？
- kafka最合理的配置是什么？
- kafka的leader选举机制是什么？
- kafka对硬件的配置有什么要求？
- kafka的消息保证有几种方式？

这些问题在下面的文章中都可以找到答案，kafka之所以有这么火热，建议各位一定要看一下。

如何学习kafka

还是那句话，学习任何技术，跟学骑自行车一样，不要一开始只关注它的具体细节是什么。先学着怎么骑，骑着骑着就了解大致的原理，这个时候在去看它的原理，会很轻松。

如果你在学习的过程中遇到什么问题，直接评论或者在问题专区中提问，我们在线人员会回复你的问题。

空间最近更新

空间贡献者

- [HoneyMoose](#) (664 days ago)

[HoneyMoose](#)

[Apache Kafka](#) updated 12/31/2018 [view change](#)

[Documentation](#) created 11/06/2018

[Use cases](#) updated 11/06/2018 [view change](#)

[Quickstart](#) updated 11/06/2018 [view change](#)

[Introduction](#) updated 11/06/2018 [view change](#)