**kafka**

# 概述

kafka是一个分布式流平台。

1. 允许发布和订阅流式数据。在这一点上类似于MQ系统
2. 能够以容错式的方式存储流式数据
3. 能够实时处理流式数据。

使用场景：

构建实时流数据管道可以在系统和应用件可靠获取数据。

构建实时流数据处理应用来处理和响应实时流数据。

* apache kafak是一个开源消息系统，由scala写成。是由Apache软件基金会开发的一个开源消息系统项目。
* kafka最初是由LinkedIn开发，并于2011年初开源。2012年10月从Apache incubator毕业。该项目的目标是为处理实时数据提供一个统一、高通量、低等待的平台。
* kafka是一个分布式消息队列(生产者、消费者的功能)。它提供了类似于JMS的特性、但是在设计实现上完全不同，此外它并不是JMS规范的实现。
* kafka对消息保存时根据Topic进行归类，发送消息者称为Producer，消息接收者称为Consumer，此外kafka集群有多个kafka实例组成，每个实例(server)成为broker。
* 无论是kafka集群，还是producer和consumer都依赖于zookeeper集群保存一些meta信息，来保证系统可用性。

# kafka和其它MQ的区别

1. kafka没有消息确认机制
2. kafka 只有发布订阅模式
3. 吞吐量：kafka具有高吞吐量，内部采用消息的批处理，zero-copy机制，数据的存储和获取是本地磁盘顺序批量操作，具有O(1)的复杂度，消息处理的效率很高。
4. 可靠性：MQ支持对消息的可靠的传递，支持事物，不支持批量的操作；基于存储的可靠性的要求存储可以采用内存或者硬盘。
5. 抗压性：kafka在大数据量下依然保持O(1)的方式提供消息持久化能力，即使对TB级以上数据也能保证常数时间复杂度的访问性能。而MQ则会出现性能下降。
6. 是否支持离线数据处理和实时数据处理：

kafka同时支持离线数据处理和实时数据处理。

1. 扩展性比较：kafka支持在线水平扩展。
2. 消息安全性：kafka可以保证在绝大多数情况下消息的安全性。当消息算作commit的时候才会暴露给consumer，并保证at-least-once的投递原则。
3. 可用性，稳定容错性：kafka天然支持HA，leader挂了，follower自动选举出新的leader
4. 适用场景：kafka是分布式消息系统。mq是基础的消息系统。kafka适用于流式计算。如果需要消息可靠性传输，使用MQ，如果处理海量日志数据等，使用kafka。如果需要同时处理在线应用(消息)和离线应用(数据文件，日志)