1.为什么使用kafka

首先kafka作为一个消息系统，有着消息队列解耦应用的好处。并且其分布式特性给了其易于扩展的能力，为在当前分布式系统以及大数据里崭露头角的机会，在分布式系统中容错和负载均衡也是以难题，其副本和分区记忆消息持久化的技术为其提供额支持。跟其他消息队列不同的是，kafka并没有采用发送-消费-删除设计模式，，保证了消息的可靠性。

2.kafka设计架构

Zookeeper

Consumer

Consumer

Consumer

Consumer

Broker

Broker

Broker

Producer

Producer

Producer

Producer

Producer：生产者采用push当时发送消息到指定分区

Zookeeper：主要元数据的存储

Broker：kafka服务节点

Consumer：以pull形式主动拉取的方式消费

吞吐量设计：数据磁盘持久化，消息不在内存cache，直接写入磁盘，充分利用磁盘顺序读写性能，底层其实是利用直接使用linux 文件系统的cache，来高效缓存数据，以及采用linux Zero-Copy提高发送性能。传统的数据发送需要发送4次上下文切换，采用sendfile系统调用之后，数据直接在内核态交换，系统上下文切换减少为2次。根据测试结果，可以提高60%的数据发送性能。那什么是zero-copy呢？

负载均衡设计：主要是生产者-分区-消费者具有一定的映射关系，不会存在始终的唯一。生产者不指定key的时候可以系统采用轮询的方式选择分区放置消息，同一个消费组的系统相同topic下不会共享一个分区

拉去系统：主动拉取，按需索取，避免一下吃的太多。

可扩展性：在zookeeper的帮助下，可以对新加入的broker进行感知也就是watcher，及时做些调整，leader replica的选举，迁移，回复等等。

Kafka还有一些特性：日志监控搜集，流处理（跟storm有点像），事物支持