



慕课大数据学习群

群号：938632081



扫一扫二维码，入群聊。



慕课网大数据体系课程



扫码体验项目【电商+直播+中台】



扫码关注获取更多课程信息



扫码体验课程配套电子书

消息队列之Kafka

学习目标

- ◆ 了解什么是Kafka
- ◆ 掌握Kafka生产和消费数据
- ◆ 掌握Kafka的高级特性



快速了解Kafka

什么是消息队列

- ◆ 消息队列(Message Queue): 简称为MQ
- ◆ 顾名思义, 消息+队列, 保存消息的队列
- ◆ 主要提供生产、消费接口供外部调用做数据的存储和读取

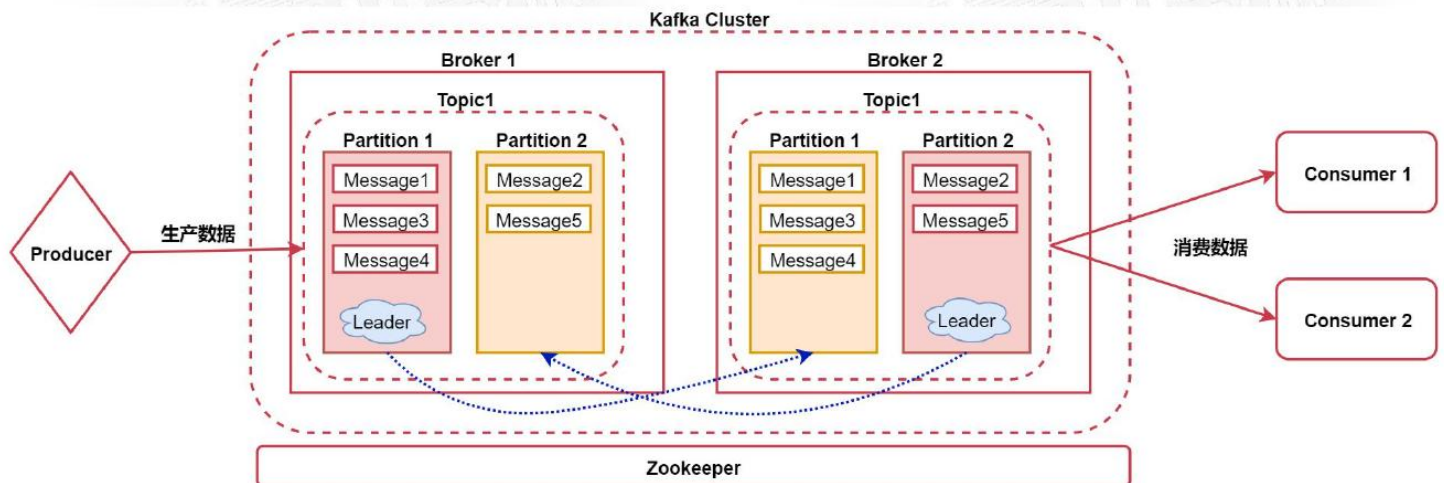
消息队列分类

- ◆ 消息队列大致可以分为两种: 点对点(P2P)、发布订阅(Pub/Sub)
- ◆ 点对点(P2P)模型包含: 消息队列(Queue)、发送者(Sender)、接收者(Receiver)
- ◆ 发布订阅(Pub/Sub)模型包含: 消息队列(Queue)、主题(Topic)、发布者(Publisher)、订阅者(Subscriber)

什么是Kafka

- ◆ Kafka 是一个 **高吞吐量** 的、**持久性** 的、**分布式** 发布订阅消息系统
- ◆ Kafka的数据是存储在磁盘中的，为什么可以满足每秒百万级别消息的生产和消费？
- ◆ Kafka主要应用在实时计算领域，可以和Flume、Spark、Flink等框架结合在一块使用

Kafka组件介绍



Kafka组件介绍

- ◆ Broker: 消息的代理, Kafka集群中的一个节点称为一个Broker
- ◆ Topic: 主题, Kafka处理的消息的不同分类(逻辑概念)
- ◆ Partition: Topic物理上的分组, 一个Topic在Broker中被分为1个或者多个Partition, 分区在创建Topic的时候指定, 每个Topic都是有分区的, 至少1个

Kafka组件介绍

- ◆ Message: 消息, 是通信的基本单位, 每个消息都属于一个Partition
- ◆ Producer: 消息和数据的生产者, 向Kafka的Topic生产数据
- ◆ Consumer: 消息和数据的消费者, 从Kafka的Topic消费数据

使用Kafka生产消费数据

Kafka安装部署

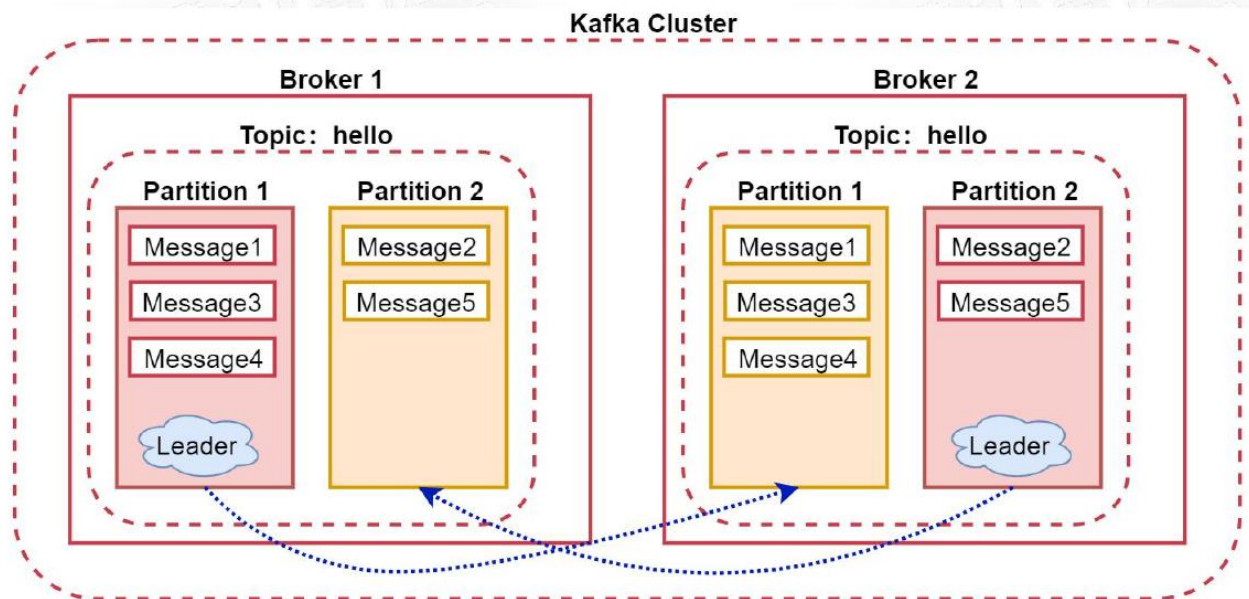
- ◆ Kafka单机安装和集群安装
- ◆ 注意：Kafka需要依赖于Zookeeper
- ◆ Zookeeper的单机安装和集群安装



Kafka中Topic的操作

- ◆ 新增Topic: 指定2个分区, 2个副本, 副本数不能大于集群中Broker的数量
- ◆ 查询Topic: 查询Kafka中的所有Topic列表以及查看指定Topic的详细信息

Kafka中Topic的操作



Kafka中Topic的操作

- ◆ 修改Topic: 修改Topic的Partition数量, 只能增加
- ◆ 删除Topic: 删除Kafka中的指定Topic, 删除操作是不可逆的
- ◆ 注意: Kafka从1.0.0开始默认开启了删除操作, 之前的版本只会把Topic标记为删除状态, 需要设置delete.topic.enable为true才可以真正删除

Kafka中的生产者和消费者

- ◆ 创建生产者: bin/kafka-console-producer.sh
- ◆ 创建消费者: bin/kafka-console-consumer.sh

案例：QQ群聊天

- ◆ 通过生产者和消费者模拟QQ群聊天的场景



Kafka的高级特性

Broker扩展

- ◆ 配置文件: server.properties
- ◆ Log Flush Policy: 设置数据flush到磁盘的时机
- ◆ Log Retention Policy: 设置数据保存周期, 默认7天

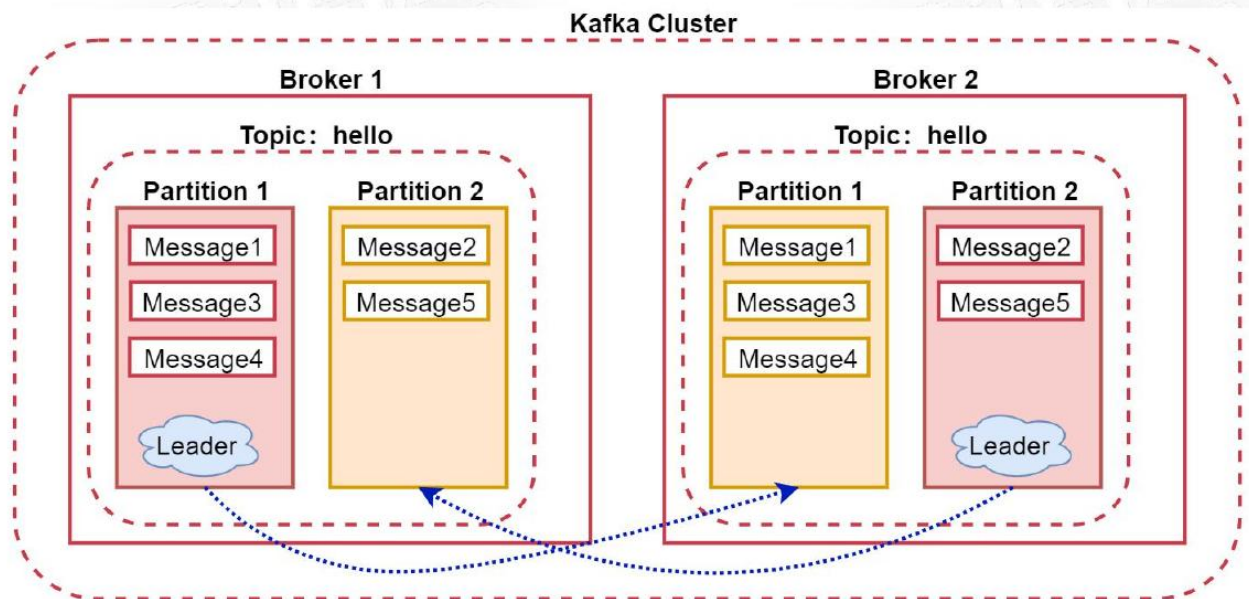
Producer扩展

- ◆ partitioner: 根据用户设置的算法来计算发送到哪个分区, 默认是随机发送到不同分区
- ◆ 数据通讯方式: 同步发送和异步发送

Kafka如何保证数据不丢

- ◆ acks: 默认为1, 表示需要Leader节点回复收到消息
- ◆ acks: all, 表示需要所有Leader+副本节点回复收到消息 (acks=-1)
- ◆ acks: 0, 表示不需要任何节点回复

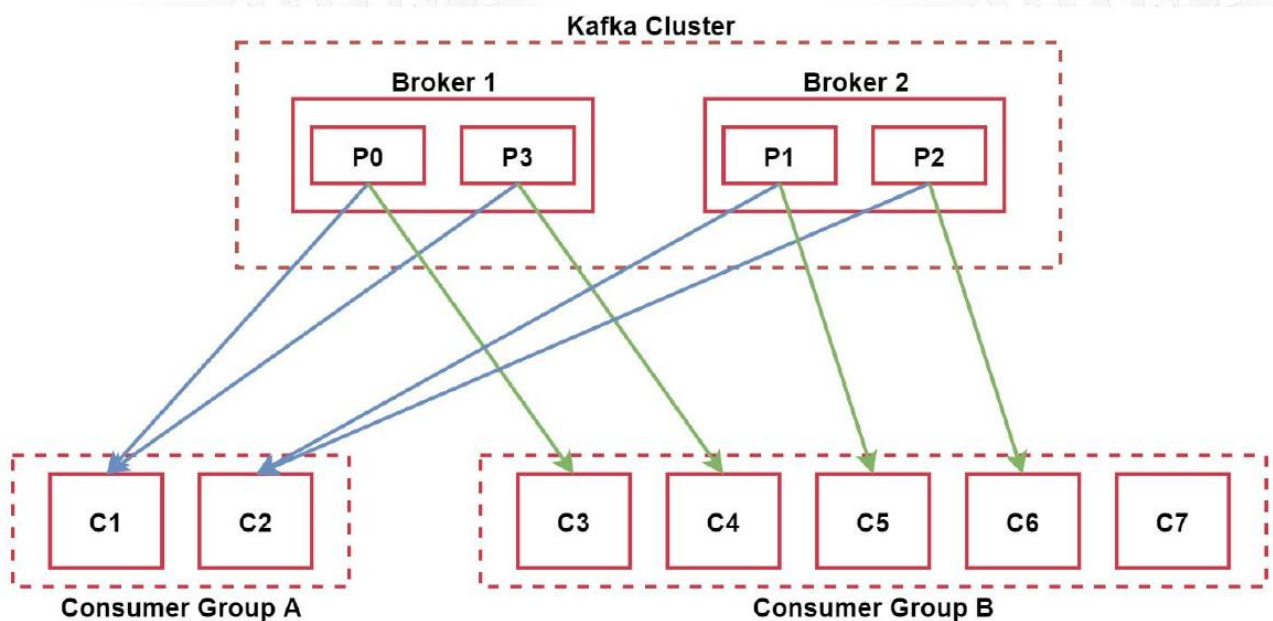
Kafka如何保证数据不丢



Consumer扩展

- ◆ 每个Consumer属于一个消费者组，通过group.id指定消费者组
- ◆ 组内消费和组间消费的区别？

组内消费和组间消费



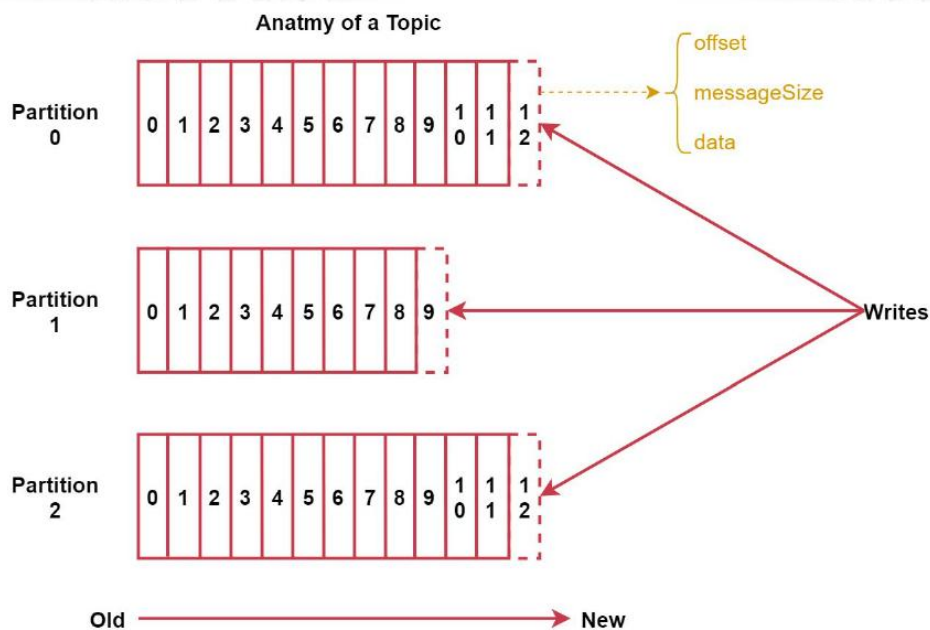
Topic、Partition扩展

- ◆ 每个Partition在存储层面是Append Log文件。新消息都会被直接追加到log文件的尾部，每条消息在log文件中的位置称为Offset(偏移量)
- ◆ 越多Partition可以容纳更多的Consumer，有效提升并发消费的能力
- ◆ 业务类型增加需要增加Topic、数据量大需要增加Partition

Message扩展

- ◆ 1: offset, 类型为long, 表示此消息在一个Partition中的起始位置, 可以认为Offset是Partition中Message的id, 自增
- ◆ 2: MessageSize, 类型为int32, 表示此消息的字节大小
- ◆ 3: data, 类型为bytes, 表示message的具体内容

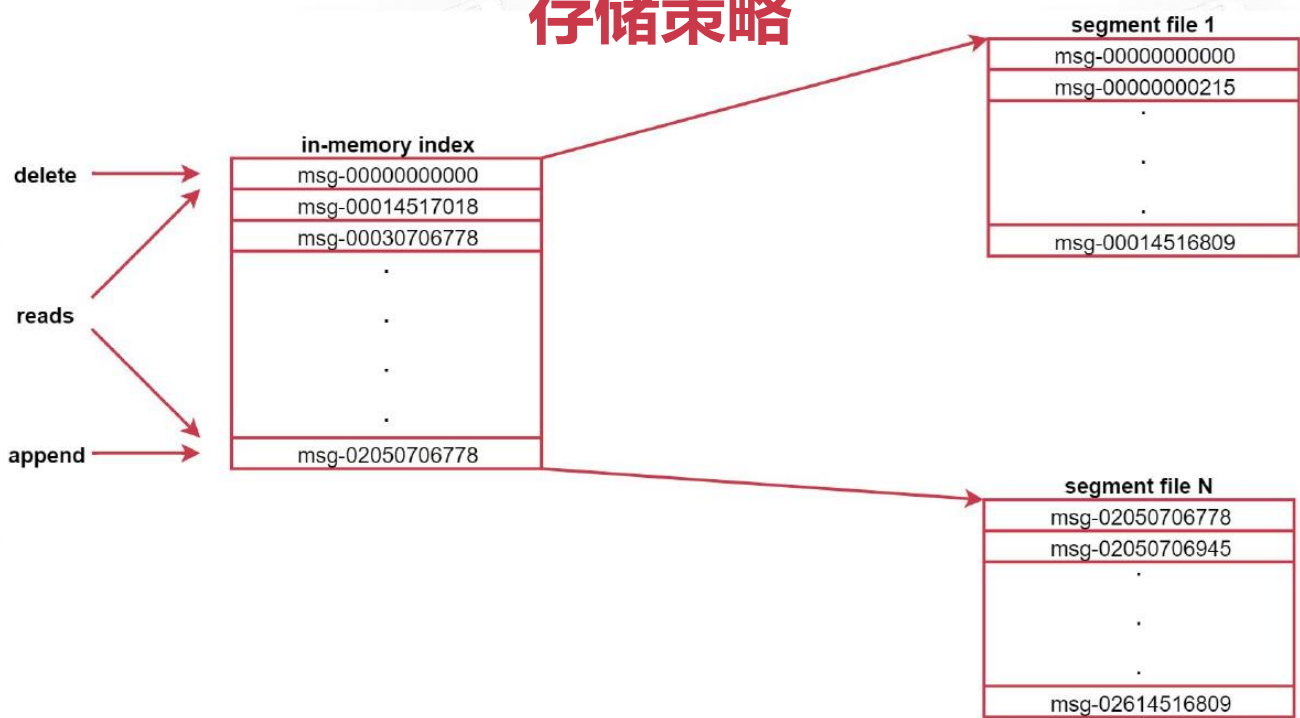
Topic、Partition、Message



存储策略

- ◆ 在Kafka中每个Topic包含1到多个Partition，每个Partition存储一部分Message。每条Message包含三个属性，其中有一个是Offset。
- ◆ Offset相当于Partition中这个Message的唯一id，那么如何通过id高效的找到Message？
- ◆ 两大法宝：分段+索引

存储策略



容错机制

- ◆ 当Kafka集群中的一个Broker节点宕机，会出现什么现象？
- ◆ 当Kafka集群中新增一个Broker节点，会出现什么现象？
- ◆ 新启动的节点不会是所有分区的Leader，怎么重新均匀分配？

Kafka Java代码编程

Kafka Java代码编程

◆ Java代码实现生产者代码

◆ Java代码实现消费者代码

◆ 消费者代码扩展



Consumer消费offset查询

- ◆ kafka0.9版本以前，消费者的offset信息保存在zookeeper中
- ◆ 从kafka0.9开始，使用了新的消费API，消费者的信息会保存在kafka里面的 `__consumer_offsets` 这个topic中
- ◆ 如何查询保存在kafka中的Consumer的offset信息呢？

Consumer消费顺序

- ◆ 当一个消费者消费一个partition时候，消费的数据顺序和此partition数据的生产顺序是一致的
- ◆ 当一个消费者消费多个partition时候，消费者按照partition的顺序，首先消费一个partition，当消费完一个partition最新的数据后再消费其它partition中的数据
- ◆ 总之：如果一个消费者消费多个partiton，只能保证消费的数据顺序在一个partition内是有序的

Kafka的三种语义

- ◆ 至少一次: at-least-once
- ◆ 至多一次: at-most-once
- ◆ 仅一次: exactly-once

Kafka集群参数调优

Kafka集群参数调优

◆ JVM参数调优

◆ Replication参数调优

◆ Log 参数调优



JVM参数调优

◆ 修改kafka-server-start.sh中的 KAFKA_HEAP_OPTS

- ◆ export KAFKA_HEAP_OPTS="-Xmx10g -Xms10g -XX:MetaspaceSize=96m -XX:+UseG1GC -XX:MaxGCPauseMillis=20 -XX:InitiatingHeapOccupancyPercent=35 -XX:G1HeapRegionSize=16M -XX:MinMetaspaceFreeRatio=50 -XX:MaxMetaspaceFreeRatio=80"

Replication参数调优

◆ `replica.socket.timeout.ms=60000`

◆ `replica.lag.time.max.ms=50000`

Log参数调优

◆ `log.retention.hours=24`



Kafka Topic命名小技巧

- ◆ Topic名称: action_r2p10
- ◆ r2: 表示Partition的副本因子是2
- ◆ p10: 表示这个Topic的分区数是10

Kafka集群监控管理工具

CMAK (KafkaManager)

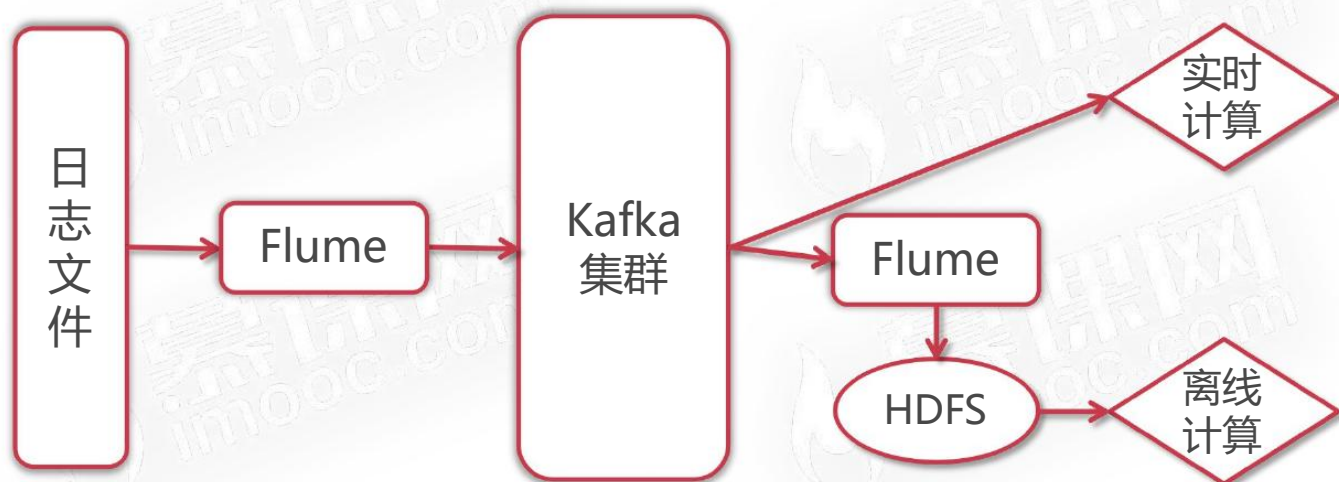
- ◆ CMAK是目前最受欢迎的Kafka集群管理工具，最早由雅虎开源，用户可以在Web界面上操作Kafka集群
- ◆ 可以轻松检查集群状态(Topic、Consumer、Offset、Brokers、Replica、Partition)

实战：Flume集成Kafka

- ◆ 需求：Flume采集日志文件实时新增数据，写入Kafka中，并且对Kafka中的数据进行落盘存储到HDFS上



实战：Flume集成Kafka



实战：Kafka集群平滑升级





本周寄语



- 内存并不是万能的
- Kafka是一个万金油
- 小技巧解决大麻烦

