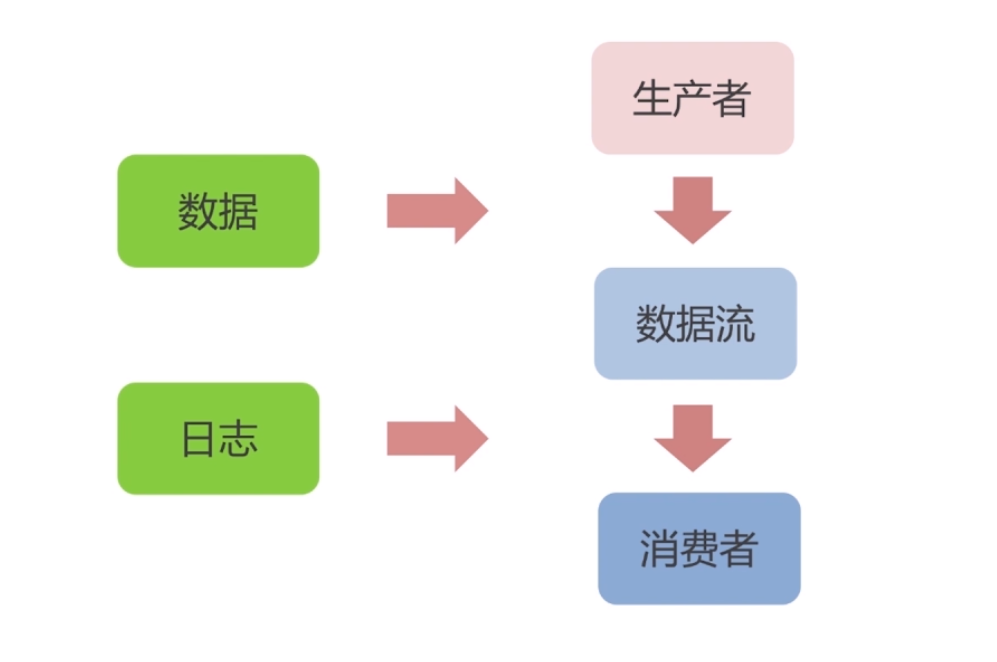
Kafka流处理平台

Kafka是目前主流的流处理平台，同时作为消息队列家族的一员，其高吞吐性作为很多场景下的主流选择。同时作为流处理平台，在大数据开发中，作为黏合剂串联各个系统。

<https://github.com/AnAngryMan/KafkaSimpleExample>









Kafka流处理平台特性：

1. 它是可以发布或订阅数据的流的系统，类似于消息队列
2. 数据流存储的平台，并且具备错误容忍
3. 当数据产生时就对数据进行处理

两类应用：

1. 构建实时数据流管道
2. 构建实时数据处理应用，转换或响应数据流

kafka是一个面向于数据流的生产、转换、存储、消费整体的流处理平台。

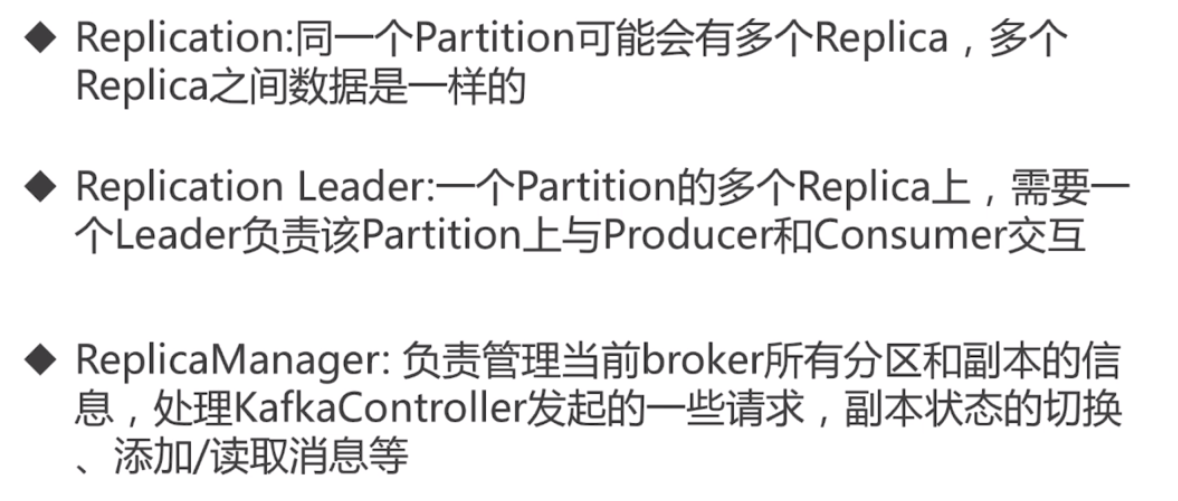
kafka是一个消息队列，但不仅仅是一个消息队列。

**Kafka基本概念**

* **Producer:消息和数据的生产者，向Kafka的一个topic发布消息的进程/代码/服务**
* **Consumer:消息和数据的消费者，订阅数据（Topic）并且处理其发布的消息的进程/代码/服务**
* **Consumer Group:逻辑概念，对于同一个topic，会广播给不同的group，一个group中，只有一个consumer可以消费该消息**
* Broker:物理概念，Kafka集群中的每个Kafka节点
* Topic:逻辑概念，Kafka消息的类别，对数据进行区分、隔离
* Partition：物理概念，Kafka下数据存储的基本单元。一个Topic数据，会被分散存储到多个Partition，每一个Partition是有序的
* Replication（副本、备份）:同一个Partition可能会有多个Replica，多个Replica之间数据是一样的
* Replication Leader:一个Partitionn的多个Replica上，需要一个Leader负责该Partition上与Producer和Consumer交互
* ReplicaManager:负责管理当前broker所有分区和副本的信息，处理KafkaController发起的一些请求，副本状态的切换、添加/读取消息、Leader的选举等







kafka组成：

broker 节点

topic 数据分类

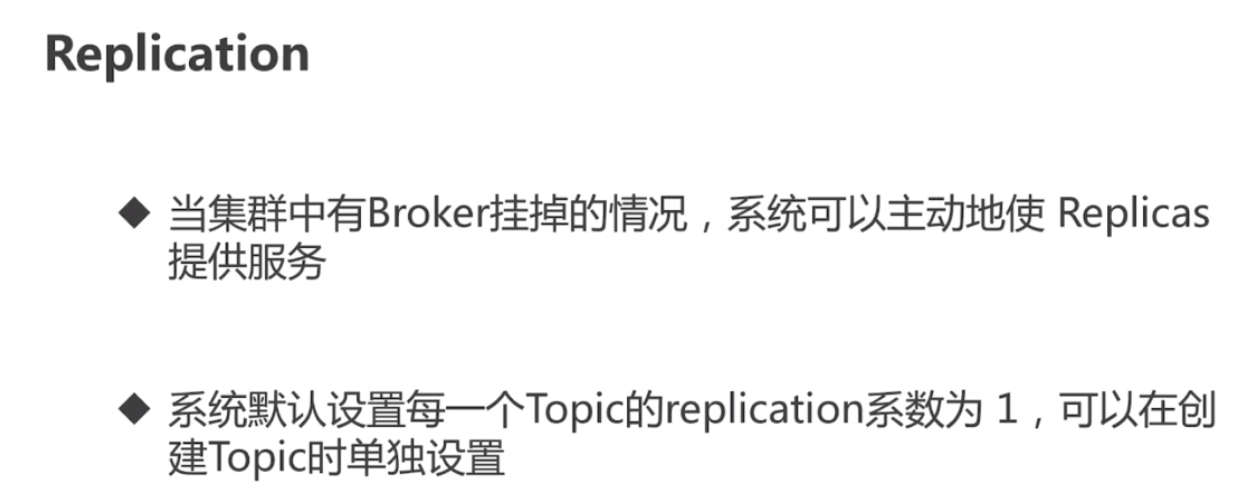
partition 分区

replication 副本



Partition

* 每一个Topic被切分为多个Partitions(Partition属于消费者存储的基本单位)
* 消费者数目小于或等于Partition的数目（多个消费者若消费同个Partition会出现数据错误，所有Kafka如此设计）
* Broker Group中的每一个Broker保存Topic的一个或多个Partitions(一个Broker只会保存一个Partition,若Partition太大则多个Broker保存同个Partition)
* Consumer Group中的仅有一个Consumer读取Topic的一个或多个Partitions，并且是唯一的Consumer(避免同一个Partition被多个Consumer消费)



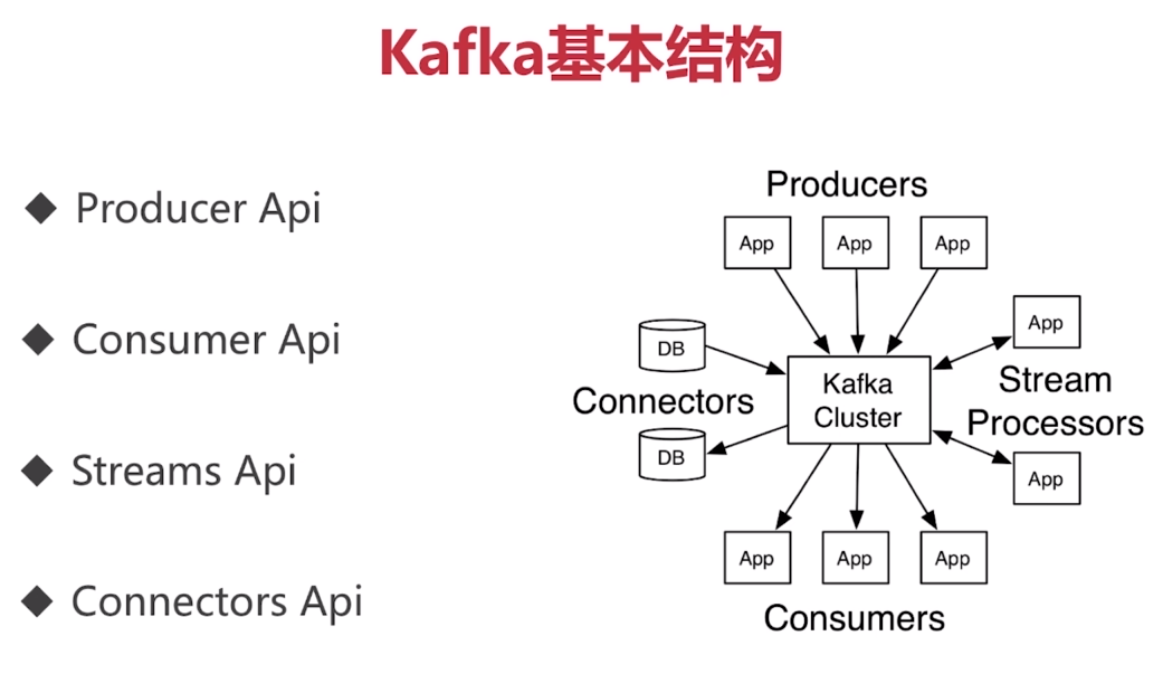
Replication

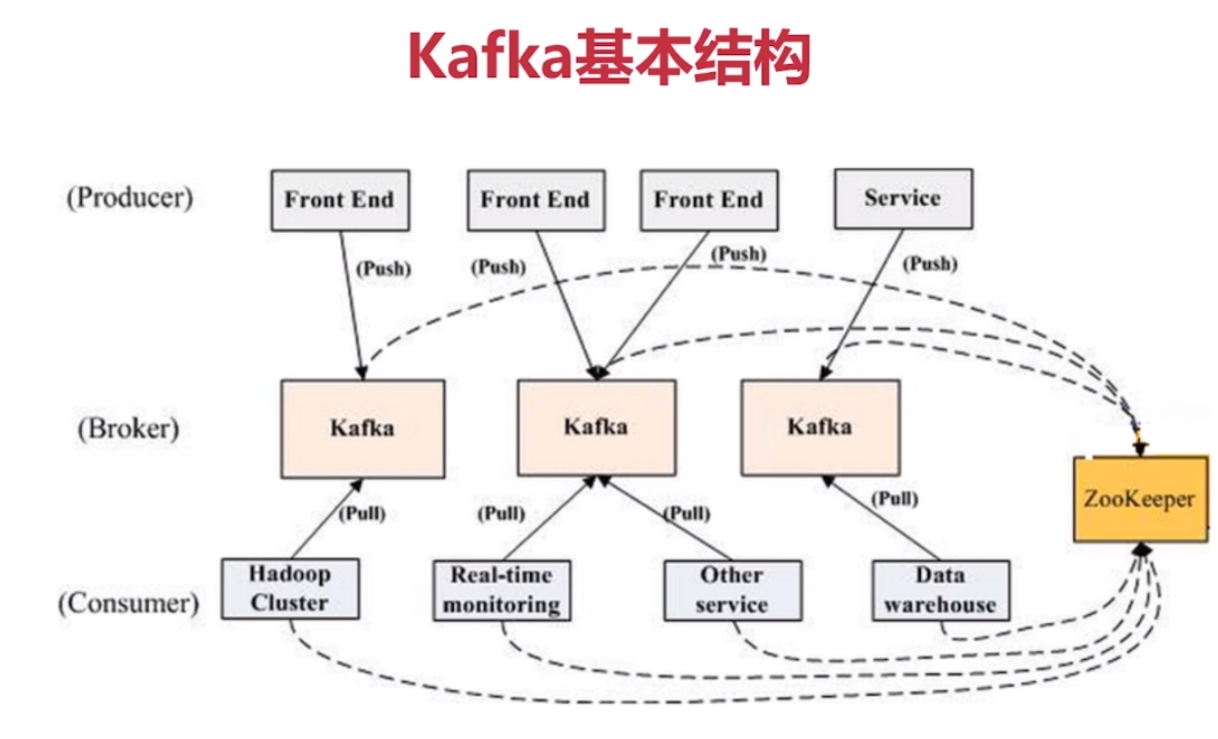
* 当集群中有Broker挂掉的情况，系统可以主动地使Replicas提供服务
* 系统默认设置每一个Topic的replication系数为1（即默认没有副本，节省资源），可以在创建Topic时单独设置

特点：

1. Replication的基本单位是Topic的Partition;
2. 所有的读和写都从Leader进，Followers只是做为备份（只有Leader管理读写，其他的Replication只做备份）
3. Follower必须能够及时复制Leader的数据
4. 增加容错性与可拓展性









kafka消息结构：

Offset: 消息的偏移量

Length: 消息的长度

CRC32 : 消息校验字段，校验信息的完整性

Magic:  用于判断该消息是不是kafka消息

attributes: 可选字段，存放当前消息的属性

Timestamp:  消费时间戳

Key Length: key的长度

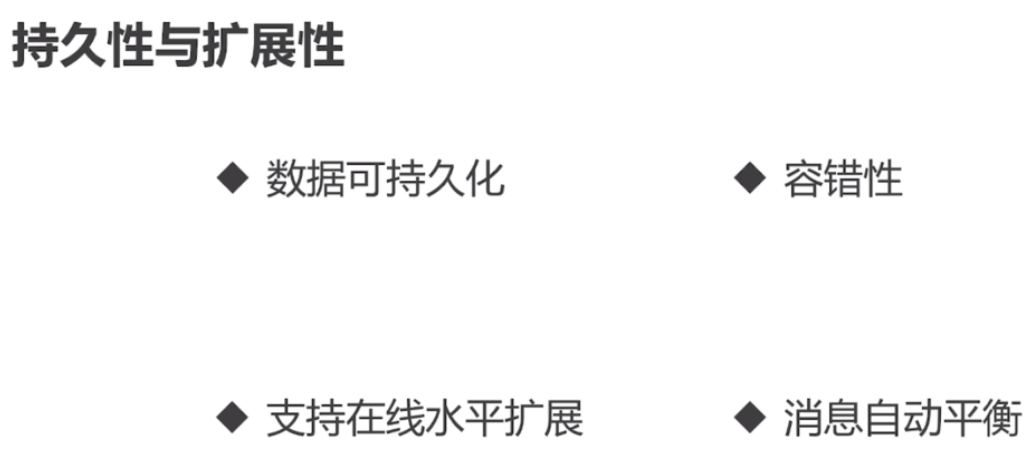
Key : key的值

Value Length: 值的长度

Value: 消息内容







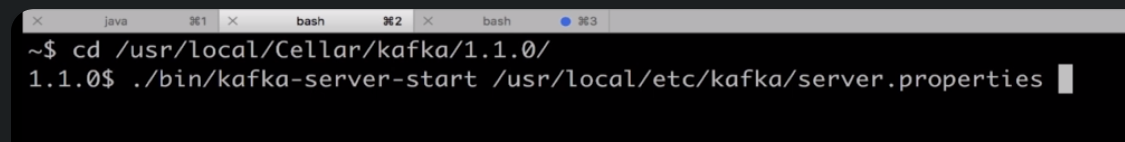
kafka应用场景

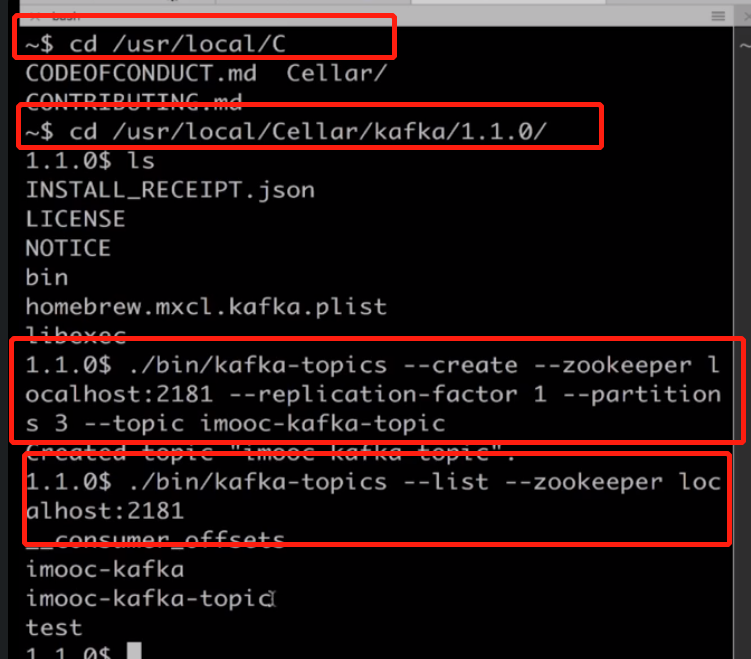
1. 消息队列
2. 行为跟踪
3. ​元信息监控
4. 日志收集
5. 流处理
6. 事件源
7. 持久性日志（commit log）

对故障节点的恢复提供一种重新投入的机制。日志压缩









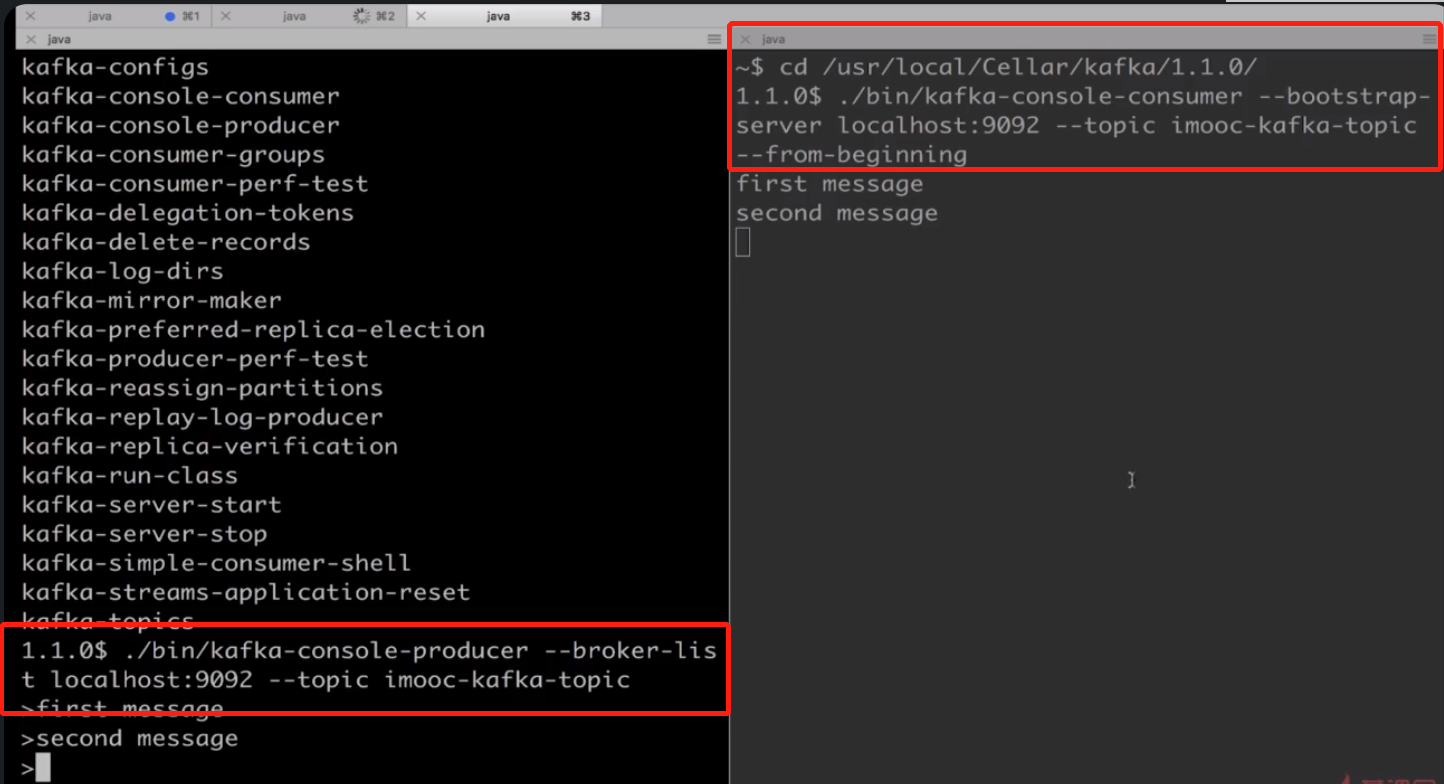
启动zookeeper命令: zookeeper-server-start /usr/local/etc/kafka/zookeeper.properties

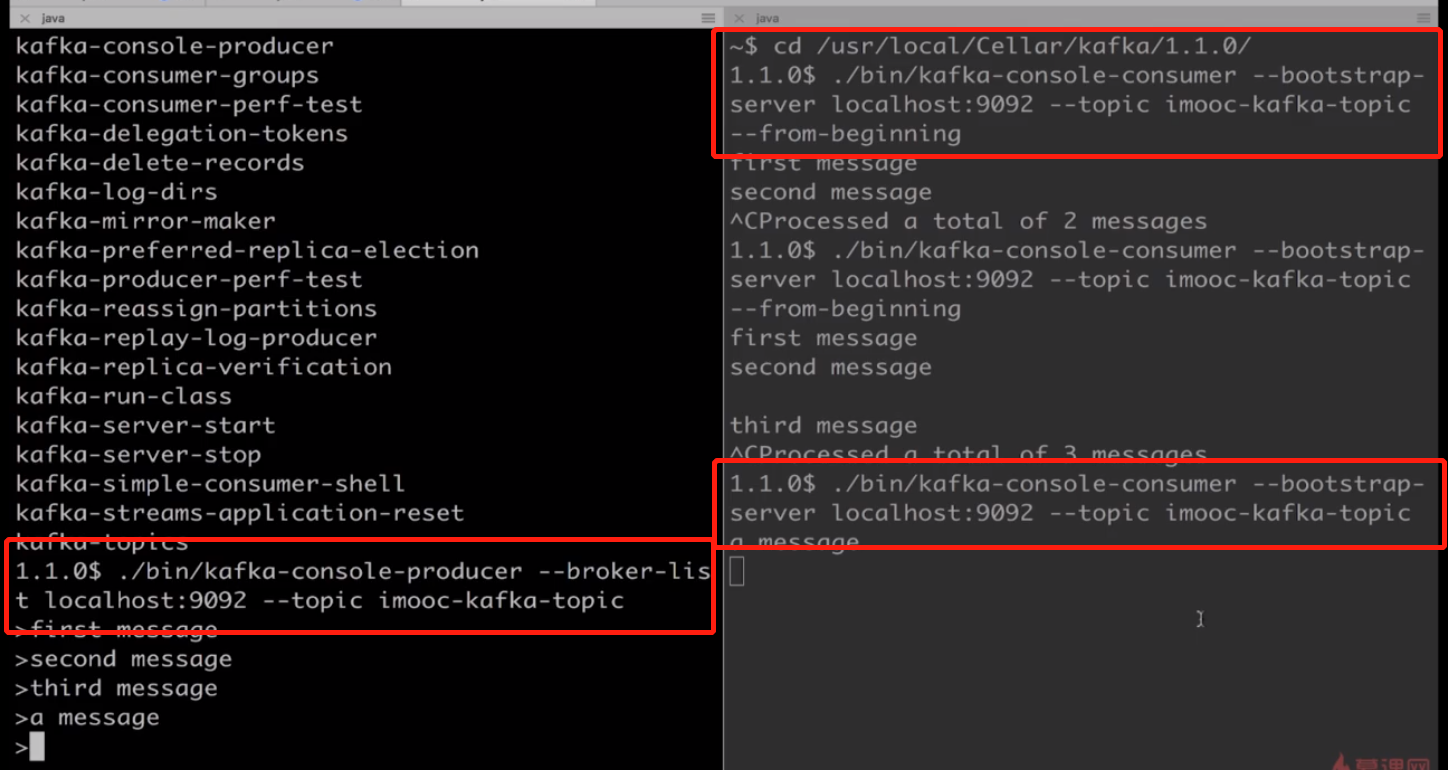
启动kafka命令： 进入kafka安装目录，cd /usr/local/Cellar/kafka/1.1.0/

启动命令：./bin/kafka-server-start /usr/local/etc/kafka/server.properties

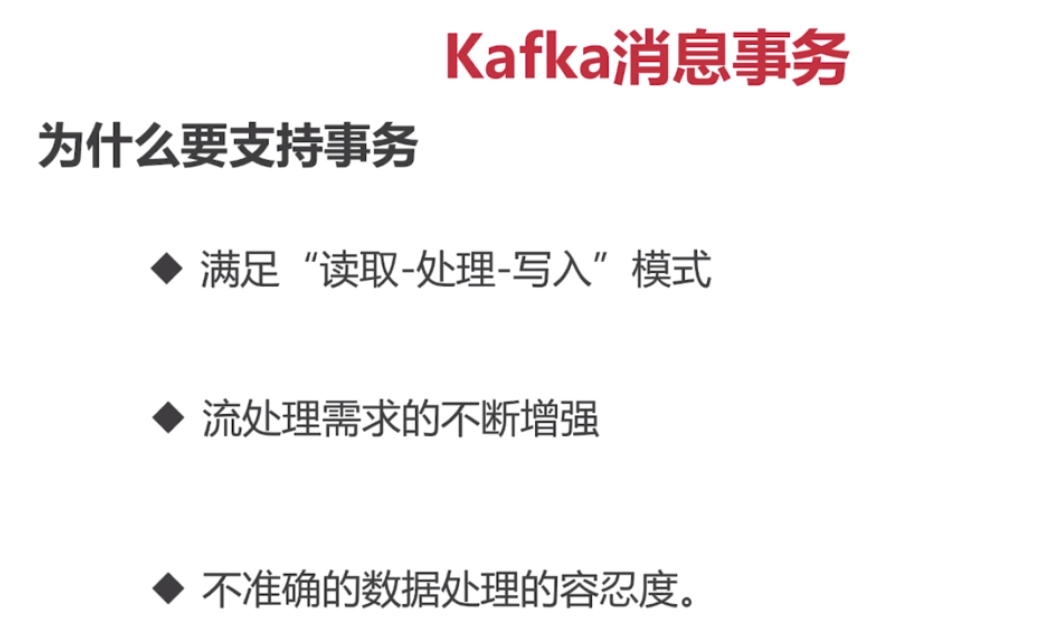
启动kafka生存者：./bin/kafka-console-producer --broker-list localhost:9092 --topic imooc-kafka-topic

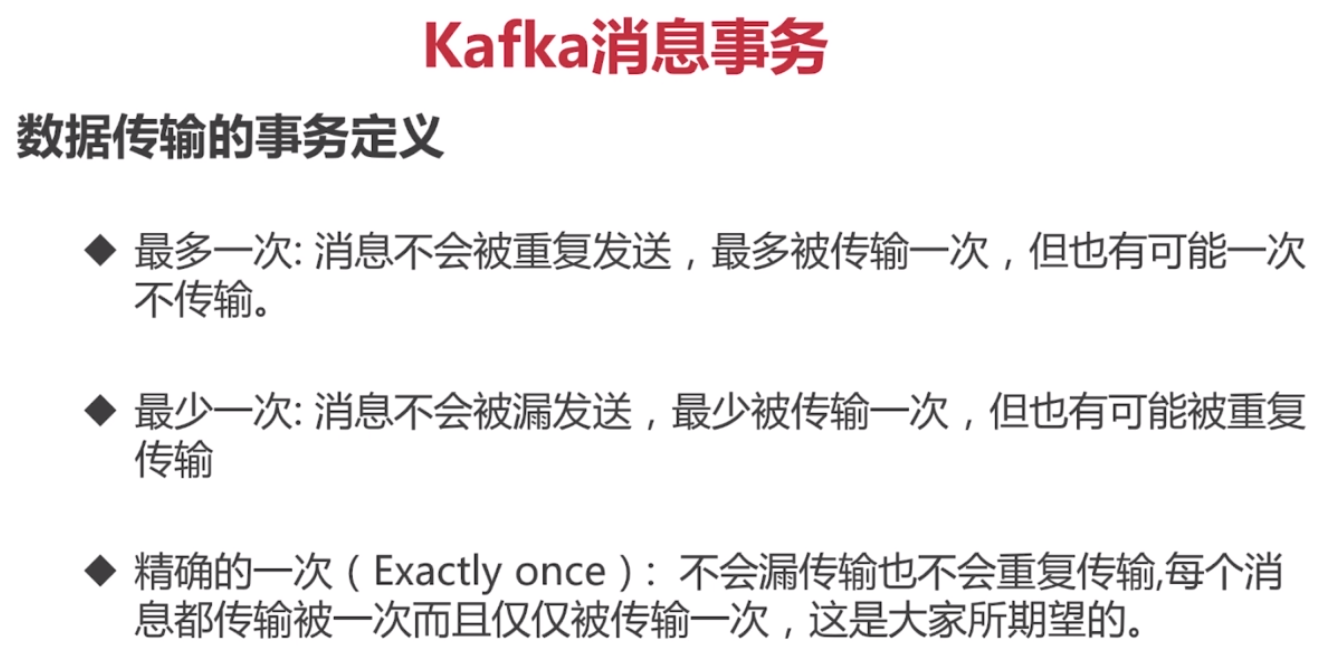
启动kafka消费者：./bin/kafka-console-consumer --bootstrap-server localhost:9092 --topic imooc-kafka-topic --from-beginning

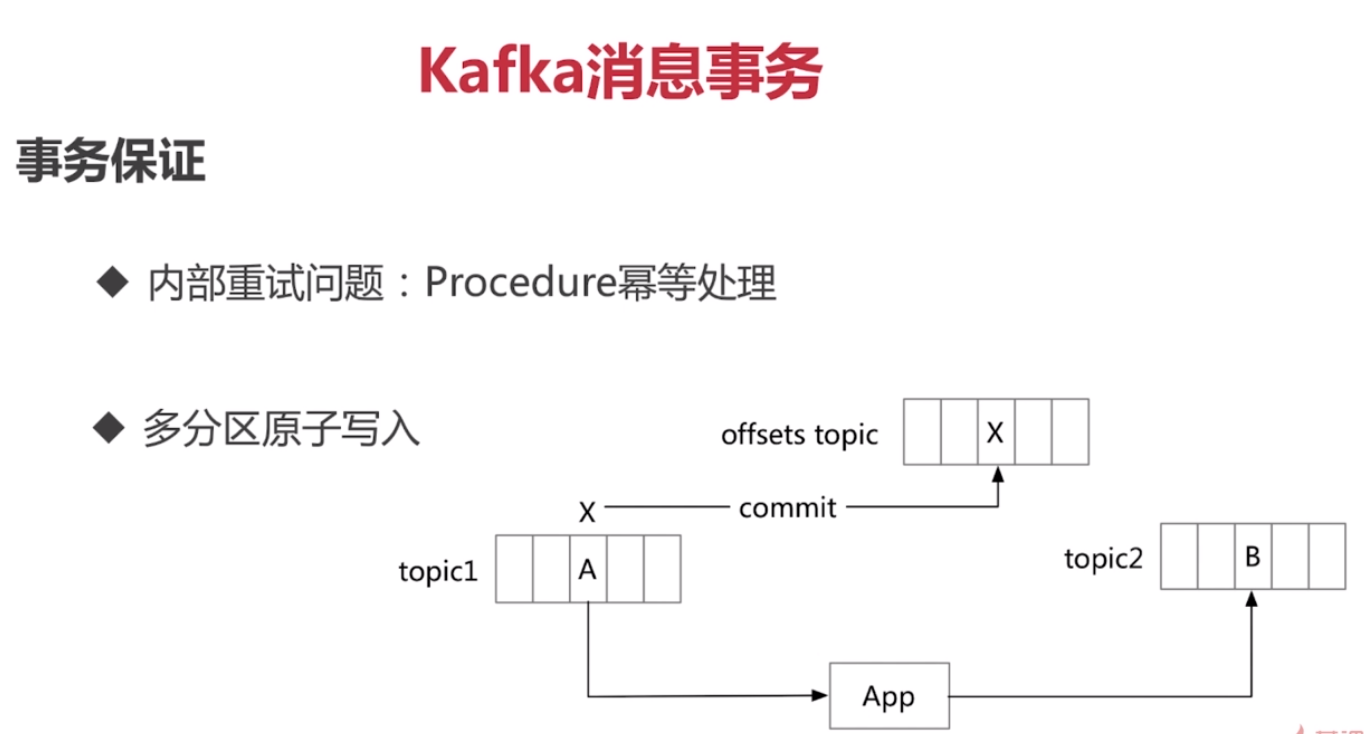




--from-beginning 是否从头消费

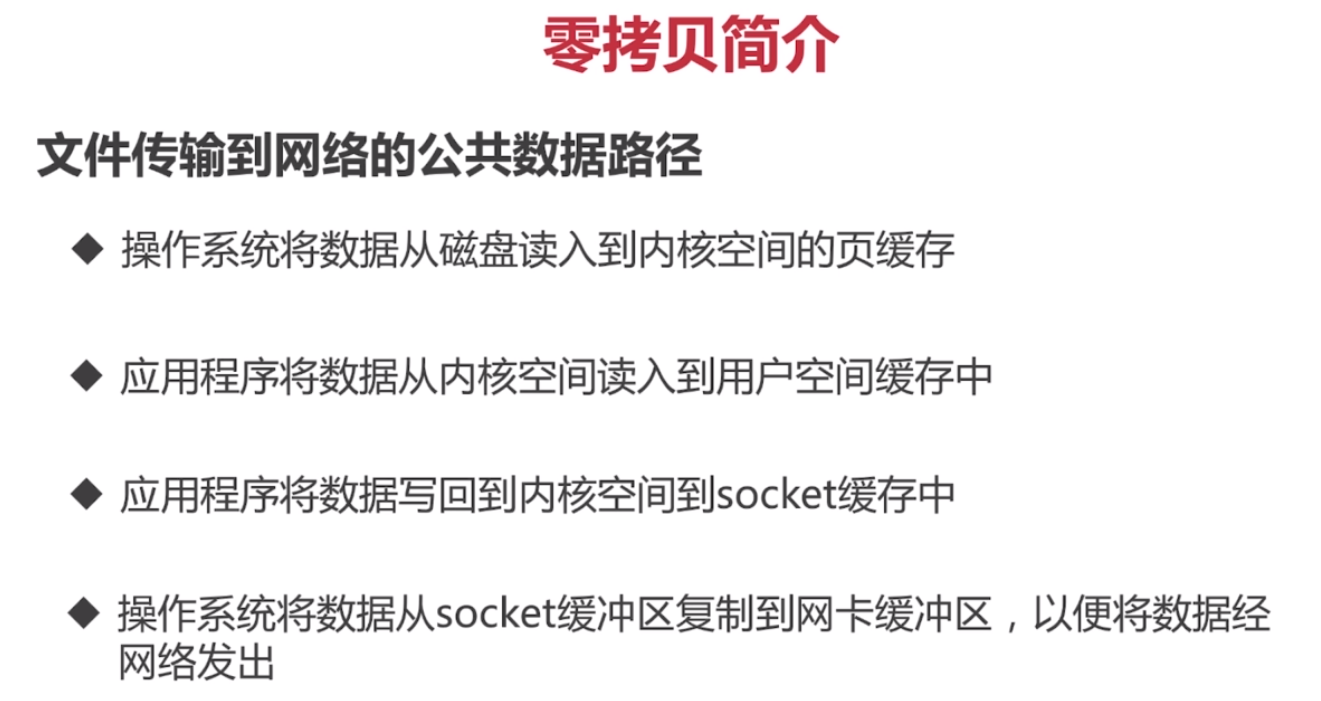


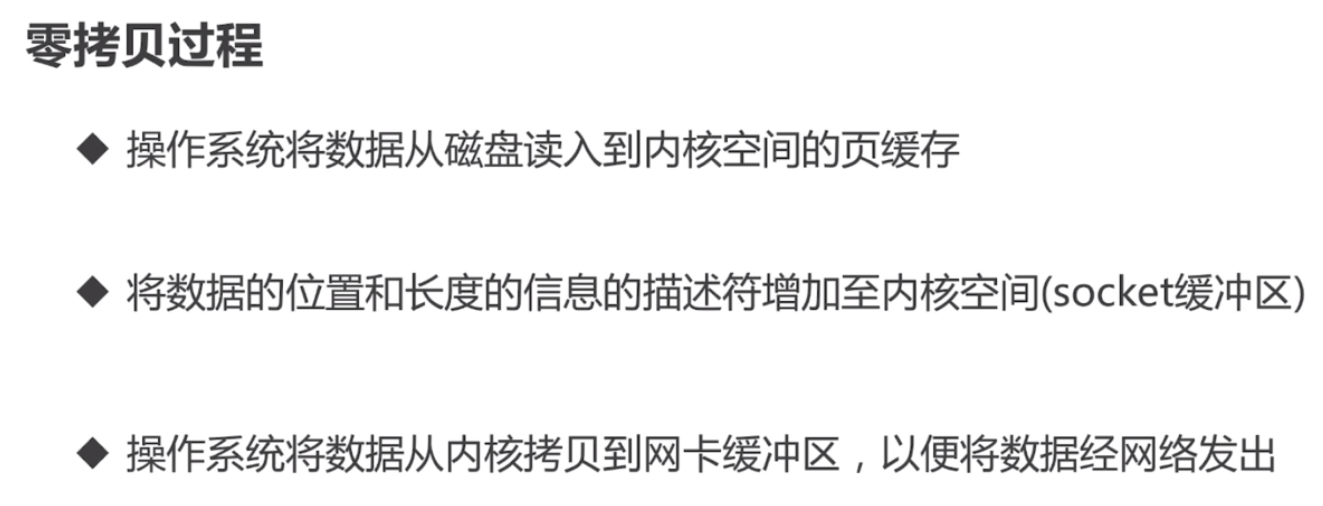


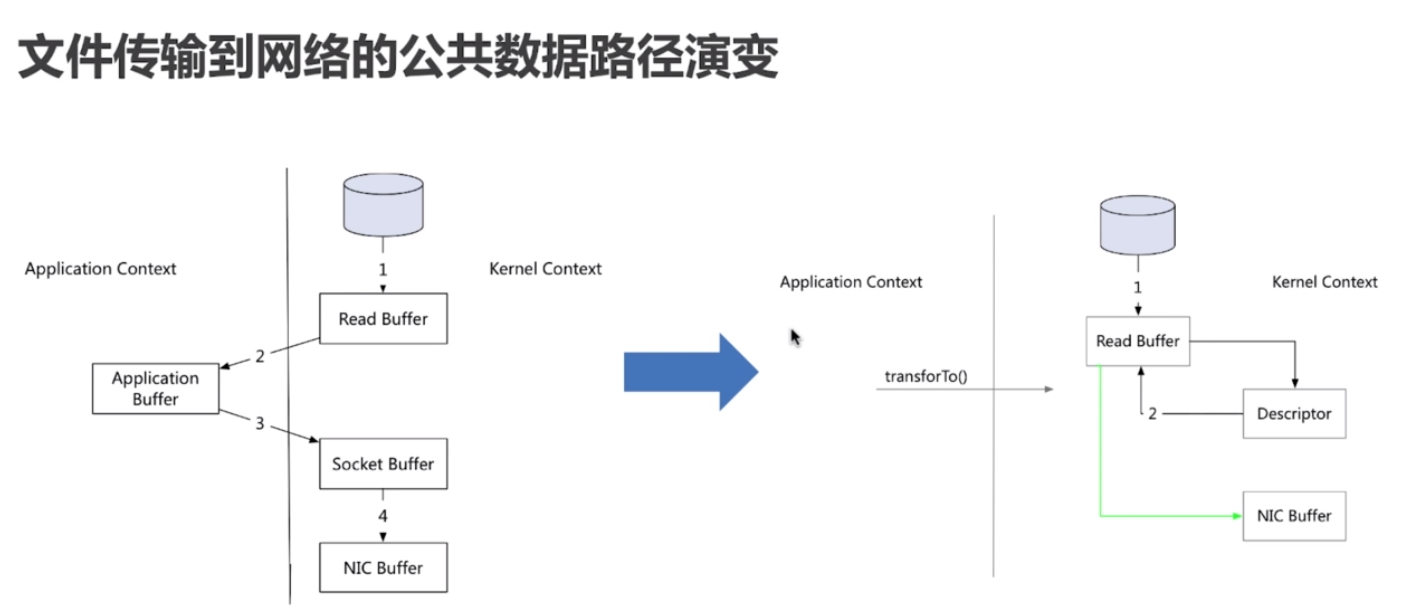


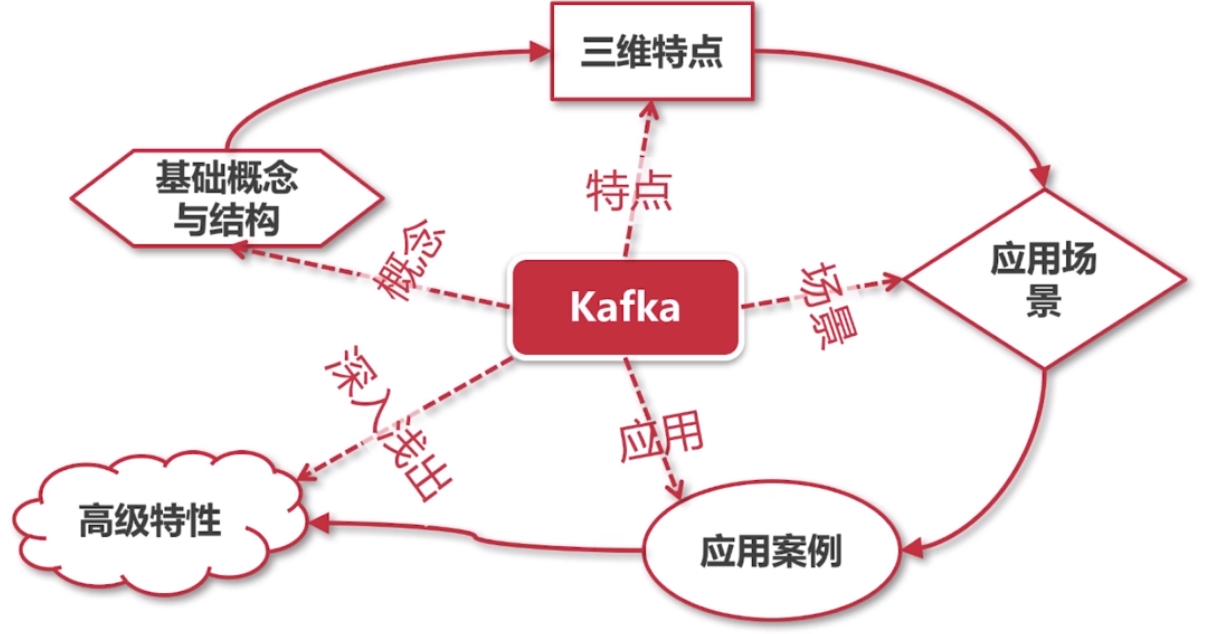










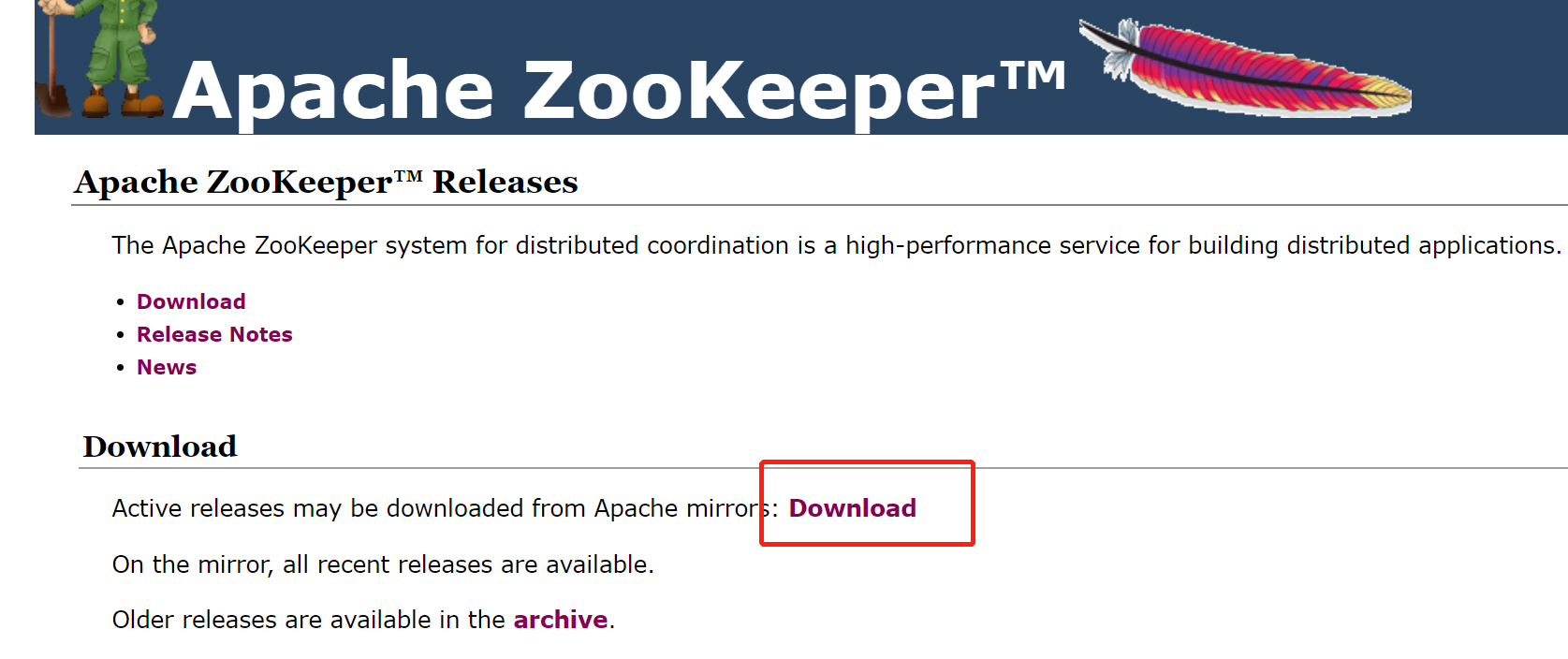


# 在Windows环境下安装运行kafka

## 安装ZooKeeper

1、 下载安装包

<http://zookeeper.apache.org/releases.html#download>





2、 解压并进入ZooKeeper目录，寡人的目录：

D:\softPackage\zookeeper\zookeeper-3.4.13\conf

3、 将“zoo\_sample.cfg”重命名为“zoo.cfg”

4、 打开“zoo.cfg”找到并编辑：

dataDir=D:\\softPackage\\zookeeper\\zookeeper-3.4.13\\tmp

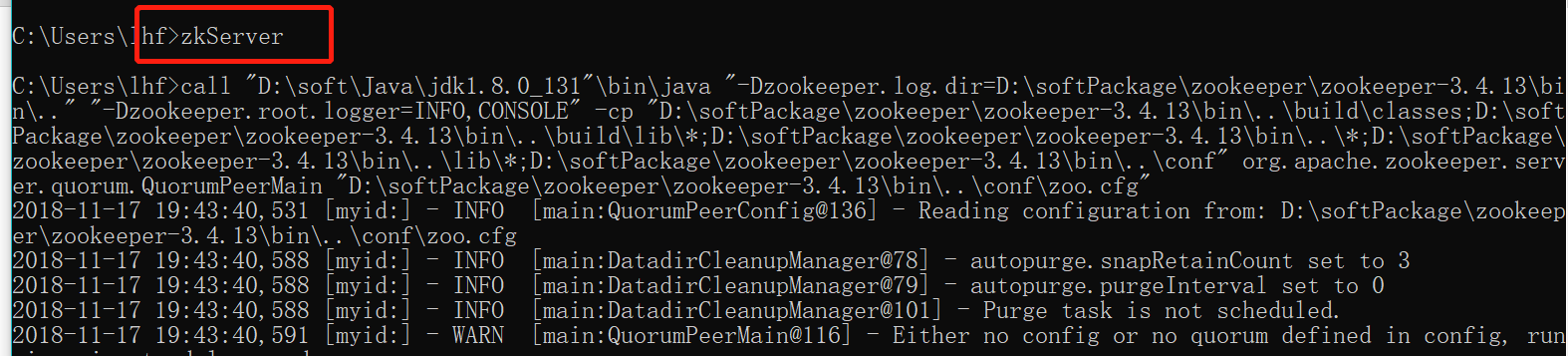
5、 添加系统变量：ZOOKEEPER\_HOME=D:\softPackage\zookeeper\zookeeper-3.4.13

6、 编辑path系统变量，添加路径：%ZOOKEEPER\_HOME%\bin

7、 在zoo.cfg文件中修改默认的Zookeeper端口（默认端口2181）

8、 打开新的cmd，输入“zkServer“，运行Zookeeper

9、 命令行提示如下：说明本地Zookeeper启动成功

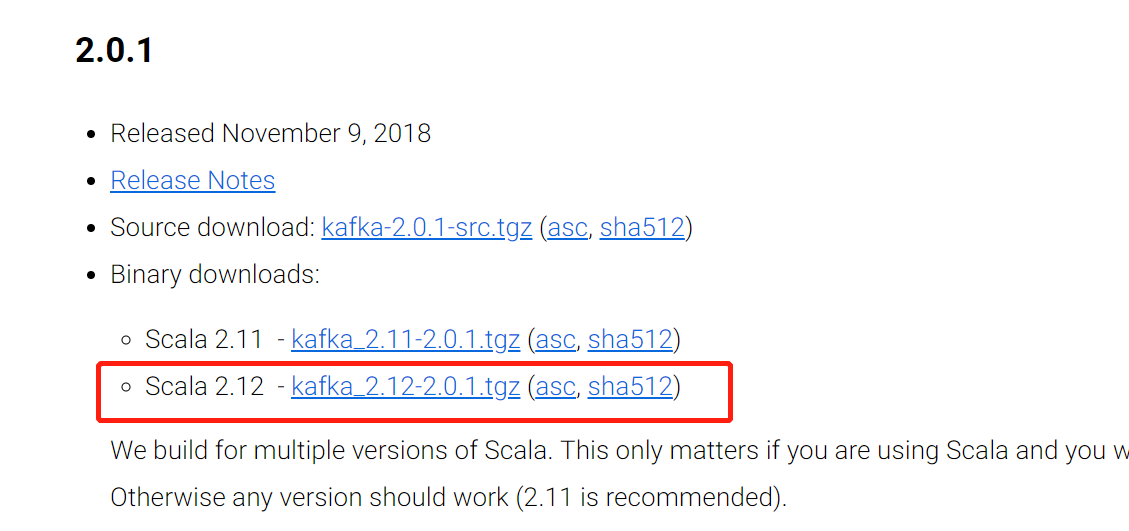


## 安装Kafka

1、 下载安装包

<http://kafka.apache.org/downloads>

注意要下载二进制版本



2、 解压并进入Kafka目录，寡人的目录：D:\softPackage\kafka\kafka\_2.12-2.0.1

3、 进入config目录找到文件server.properties并打开

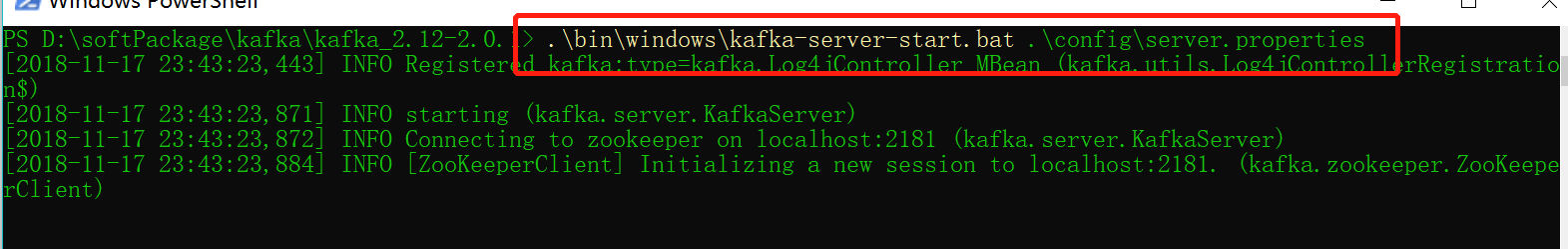
4、 找到并编辑log.dirs=D:\softPackage\kafka\kafka\_2.12-2.0.1\kafka-logs

5、 找到并编辑zookeeper.connect=localhost:2181

6、 Kafka会按照默认，在9092端口上运行，并连接zookeeper的默认端口：2181

7、 进入Kafka安装目录D:\softPackage\kafka\kafka\_2.12-2.0.1，按下Shift+右键，选择“打开命令窗口”选项，打开命令行，输入如下命令，启动kafka服务

**.\bin\windows\kafka-server-start.bat .\config\server.properties**

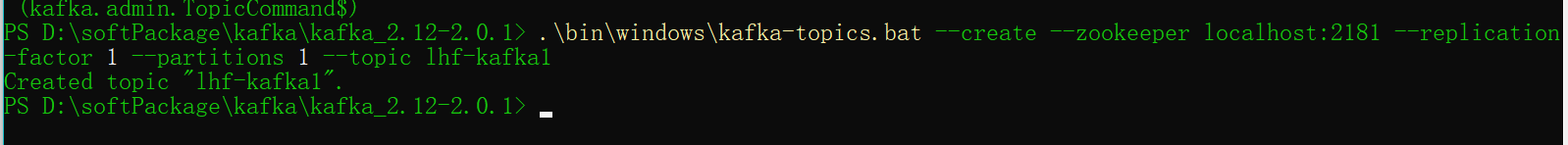


注意：不要关了这个窗口，启用Kafka前请确保ZooKeeper实例已经准备好并开始运行

## 实战应用

1、 创建主题，进入Kafka安装目录D:\softPackage\kafka\kafka\_2.12-2.0.1，按下Shift+右键，选择“打开命令窗口”选项，打开命令行，输入：

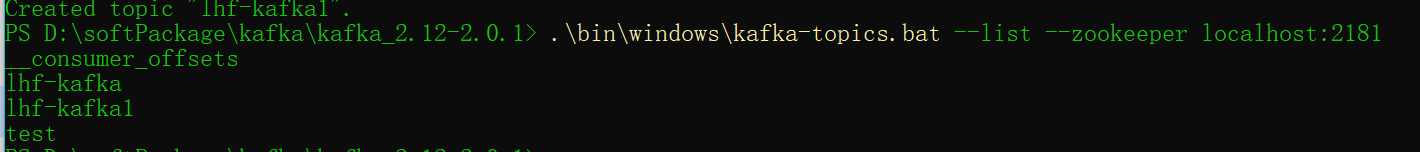
**.\bin\windows\kafka-topics.bat --create --zookeeper localhost:2181 --replication-factor 1 --partitions 1 --topic lhf-kafka**



注意：不要关了这个窗口

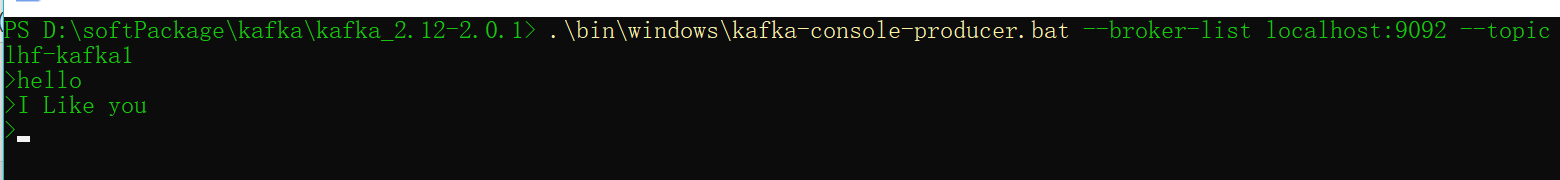
查看主题输入：

**.\bin\windows\kafka-topics.bat --list --zookeeper localhost:2181**



2、 创建生产者，进入Kafka安装目录D:\softPackage\kafka\kafka\_2.12-2.0.1，按下Shift+右键，选择“打开命令窗口”选项，打开命令行，输入：

**.\bin\windows\kafka-console-producer.bat --broker-list localhost:9092 --topic lhf-kafka**



注意：不要关了这个窗口

3、 创建消费者，进入Kafka安装目录D:\softPackage\kafka\kafka\_2.12-2.0.1，按下Shift+右键，选择“打开命令窗口”选项，打开命令行，输入：

**.\bin\windows\kafka-console-consumer.bat --bootstrap-server localhost:9092 --topic lhf-kafka --from-beginning**

