**Kafka知识点总结1 digest-time：2021-10-03**

**1. 什么是Kafka**

        kafka是一个分布式，**分区的（partition）**，**多副本（replica）的**，多订阅者的消息发布订阅系统。

**2. Kafka的使用场景**

        应用解耦（or 消息驱动）、异步处理、限流削峰的系统。

### 3. Kafka优缺点

**优点**：

        可靠性强（分布式-分区-副本）、扩展性强（可伸缩）、性能高（数据读写、吞吐量大）、具备数据持久化能力、时效性强。

**缺点**：

* 由于是批量发送，数据并非真正的实时（准实时级别）。
* 仅支持统一分区内消息有序，无法实现全局消息有序（同个partition有序）；
* 有可能消息重复消费（需要业务层进行控制）；
* 依赖zookeeper进行元数据管理。

### 4. Kafka架构（流程）

        生产者、kafka集群（brokers）、消费者、zookeeper

### 5. Kafka架构（API）

        生产者、消费者、StreamAPI、ConnectAPI

### 6.Topic内部有哪些组成？

        每个Topic包含一个或者多个Partition，一个partition当中存在多个segment文件段，每个segment分为两部分，.log文件和.index文件。

### 7.分区和消费组内的消费者之间的关系有哪些情况？

        Partition = 消费任务的并发度=刚刚好，每个任务读取一个partition数据

        Partition > 消费任务的并发度=有部分消费任务读取多个分区的数据

 Partition < 消费任务的并发度=有部分消费任务空闲(可以创建多于分区的消费者数量)

### 8. 分区数、消费者与读取效率之间的关系

        分区数越多，同一时间可以有越多的消费者来进行消费，消费数据的速度就会越快，提高消费的性能

### 9. 副本数与broker之间的关系

        数据副本（包含本身）数一般情况下小于等于broker的个数

### 10. 什么是主/从副本

        被复制的分区叫做主副本（Leader）,复制出来的叫做从副本（Follower）

### 11. 主/从复本的作用是什么

        主副本负责数据的读写。  
        从副本只做数据备份，不做数据读写。

### 12. Isr是什么

        ISR是一组与leaders**完全同步的消息副本**（包括leaders本身）。

### 13. 生产者生产数据到kafka集群，数据到分区的方式

        a)没有指定分区编号，没有指定key时采用**轮询方式**存储数据

        b)没有指定分区编号，指定key时，数据分发策略为**对key求取hash值，这个值与分区数量取余**，余数就是分区编号。

        c)指定分区编号，所有数据输入到指定的分区内

        d)自定义分区

### 14. Consumer消费数据的流程

        1、首先Consumer连接指定的Topic **partition**所在leader **broker**，使用折半/**二分查找**，先确定数据所在的**segment**。

        2、确定在哪个segment后，使用确定的segment内的**index**文件找到数据具体的位置采用**pull**方式从**kafkalogs**中获取消息。

### 15. Kafka中的数据删除机制是什么？

        1、时间：默认存储168小时（一周）

        2、数据的大小：默认 -1 （不删除），可以自行设置。

### 16. Kafka如何保证数据不丢失

        从大体上来看的话，Kafka中主要的角色有数据生产者(**Producer**)，Kafka集群中负责存储数据的**Broker**，数据消费者（**Consumer**），因此这个问题需要从三个角度来进行回答!

        1、生产者如何保证数据不丢失？？ 通过**ack 机制**确保数据不丢失。

        2、kafka集群如何保证数据不丢失？？ 通过**数据副本**保证数据不丢失。

        3、消费者如何保证数据不丢失？？ 通过维护数据的**offset** 保证数据不丢失。

### 17. Kafka高性能的原因有哪里

        顺序读写、分区、批量发送、数据压缩

### 18. Kafka高效查询数据的有哪些原因

        1、Kafka把topic中**一个parition**大文件**分成多个小文件段**，通过多个小文件段，就容易定期清除或删除已经消费完文件，减少磁盘占用。

        2、通过索引信息可以快速定位message和确定response的最大大小。

        3、通过index元数据全部映射到memory，可以避免segment file的IO磁盘操作（**TODO**）。

        4、通过索引文件稀疏存储，可以大幅降低index文件元数据占用空间大小（**TODO**）。

### 19. 如何从Kafka得到准确的信息(不是重读数据)?

        在数据生产过程中避免重复。

        在数据消耗期间避免重复。

### 20. Kafka的设计是什么样的呢？

        Kafka将消息以topic为单位进行归纳。将向Kafka topic发布消息的程序成为producers。将预订topics并消费消息的程序成为consumer。 Kafka以集群的方式运行，可以由一个或多个服务组成，每个服务叫做一个broker。producers通过网络将消息发送到Kafka集群，集群向消费者提供消息。

### 21. 数据传输的事物定义有哪三种？

        1、最多一次:消息不会被重复发送，最多被传输一次，但也有可能一次不传输，可能会造成数据丢失。

        2、最少一次: 消息不会被漏发送，最少被传输一次，但也有可能被重复传输。可能会造成数据的重复消费。

        3、精确的一次（Exactly once）:不会漏传输也不会重复传输,每个消息都传输被一次而且仅仅被传输一次，这是大家所期望的。

### 22. Kafka判断一个节点是否还活着有哪些条件？

        1、节点必须可以维护和ZooKeeper的连接，Zookeeper通过**心跳机制**检查每个节点的连接

        2、如果节点是个follower,他必须能及时的同步leader的写操作，延时不能太久

### 23. Kafka 与传统消息系统之间有哪些区别？

        1、Kafka 持久化日志：这些日志可以被重复读取和无限期保留

        2、Kafka 是一个分布式系统：它以集群的方式运行，可以灵活伸缩，在内部通过复制数据提升容错能力和高可用性

        3、Kafka 支持实时的流式处理

### 24. Kafka创建Topic时将分区放置到不同的Broker的策略是什么？

        前提：副本因子不能大于 Broker 的个数；

        第一个分区（编号为0）的第一个副本放置位置是随机从 brokerList 选择的；

        其他分区的第一个副本放置位置相对于第0个分区依次往后移。

        例如：有5个 Broker，5个分区，假设第一个分区放在第四个 Broker 上，那么第二个分区将会放在第五个 Broker 上；第三个分区将会放在第一个 Broker 上；第四个分区将会放在第二个 Broker 上，第五个分区将会放在第三个 Broker 上；

### 25. Kafka新建的分区会在哪个目录下创建

        若 log.dirs 参数只配置了一个目录，那么分配到各个 Broker 上的分区将在这个目录下创建文件夹用于存放数据。

        若 log.dirs 参数配置了多个目录，那么 Kafka 会在含有分区目录总数最少的文件夹中创建新的分区目录，分区目录名为 Topic名+分区ID。注意（不是磁盘使用量最少的目录）

### 26. partition的数据如何保存到硬盘

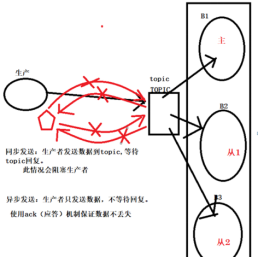
        topic中的多个partition以文件夹的形式保存到broker，每个分区序号从0递增， 且消息有序 Partition文件下有多个segment（xxx.index，xxx.log） segment 文件里的 大小和配置文件大小一致可以根据要求修改 默认为1g 如果大小大于1g时，会滚动一个新的segment并且以上一个segment最后一条消息的偏移量命名。

### 27. kafka的ack机制

        request.required.acks有三个值 0 1 -1

        0:生产者不会等待broker的ack，这个延迟最低但是存储的保证最弱当server挂掉的时候就会丢数据

        1：服务端会等待ack值 leader副本确认接收到消息后发送ack但是如果leader挂掉后它不确保是否复制完成新leader也会导致数据丢失

        -1：同样在1的基础上 服务端会等所有的follower的副本受到数据后才会受到leader发出的ack，这样数据不会丢失  


### 28. Kafka的消费者如何消费数据

        消费者每次消费数据的时候，消费者都会记录消费的物理偏移量（offset）的位置 等到下次消费时，他会接着上次位置继续消费。同时也可以按照指定的offset进行重新消费。

### 29. 如何使kafka集群内的数据是有序的？

        只创建一个分区。（但是实际这样会存在性能问题，具体业务具体分析后确认。）

### 30. Zookeeper汇总保留了哪些数据？

        1、消费者提交的偏移量。

 2、leader检测、分布式同步、配置管理、识别新节点何时离开或连接、集群、节点实时状态  
        3、分区和消费者的所有者关系  
        4、broker id

### 31. kafka consumer 什么情况会触发再平衡reblance?

        1、一旦消费者加入或退出消费组，导致消费组成员列表发生变化，消费组中的所有消费者都要执行再平衡。  
        2、订阅主题分区发生变化，所有消费者也都要再平衡。

### 32、描述下kafka consumer 再平衡步骤?

        1、关闭数据拉取线程，清空队列和消息流，提交偏移量；  
        2、释放分区所有权，删除zk中分区和消费者的所有者关系；

 3、将所有分区重新分配给每个消费者，每个消费者都会分到不同分区；  
        4、将分区对应的消费者所有关系写入ZK，记录分区的所有权信息；  
        5、重启消费者拉取线程管理器，管理每个分区的拉取线程。

### 33. 手动提交offset有什么好处？？

        使更新offset更及时，避免因offset更新不及时导致重复消费数据的问题。

### 34. 为什么kafka中的数据需要定期删除或合并？

        Kafka只用于做数据的临时存储、临时缓冲，不是永久存储（永久性存储使用HDFS）。