

## 大数据面试题（一）

### 一、.hdfs写文件的步骤

答案：

- (1)client向NameNode申请上传.../xxx.txt文件
- (2)NN向client响应可以上传文件
- (3)Client向NameNode申请DataNode
- (4)NN向Client返回DN1,DN2,DN3
- (5)Client向DN1,DN2,DN3申请建立文件传输通道
- (6)DN3,DN2,DN1依次响应连接
- (7)Client向DN1上传一个block，DN1向DN2,DN3冗余文件

### 二、hdfs读取文件步骤

答案：

- (1)client向NN请求下载.../xxx.txt文件
- (2)NN向client返回文件的元数据
- (3)Client向DN1请求访问读数据blk\_1
- (4)DN1向Client传输数据
- (5)Client向DN2请求访问读数据blk\_2
- (6)DN2向Client传输数据

### 三、hadoop的shuffle过程

#### 1.Map端的shuffle

Map端会处理输入数据并产生中间结果，这个中间结果会写到本地磁盘，而不是HDFS。每个Map的输出会先写到内存缓冲区中，当写入的数据达到设定的阈值时，系统将会启动一个线程将缓冲区的数据写到磁盘，这个过程叫做spill。

在spill写入之前，会先进行二次排序，首先根据数据所属的partition进行排序，然后每个partition中的数据再按key来排序。partition的目的是将记录划分到不同的Reducer上去，以期能够达到负载均衡，以后的Reducer就会根据partition来读取自己对应的数据。接着运行combiner(如果设置了的话)，combiner的本质也是一个Reducer，其目的是对将要写入到磁盘上的文件先进行一次处理，这样，写入到磁盘的数据量就会减少。最后将数据写到本地磁盘产生spill文件(spill文件保存在{mapred.local.dir}指定的目录中，Map任务结束后就会被删除)。

最后，每个Map任务可能产生多个spill文件，在每个Map任务完成前，会通过多路归并算法将这些spill文件归并成一个文件。至此，Map的shuffle过程就结束了。

#### 2.Reduce端的shuffle

Reduce端的shuffle主要包括三个阶段，copy、sort(merge)和reduce。

首先要将Map端产生的输出文件拷贝到Reduce端，但每个Reducer如何知道自己应该处理哪些数据呢？因为Map端进行partition的时候，实际上就相当于指定了每个Reducer要处理的数据(partition就对应了Reducer)，所以Reducer在拷贝数据的时候只需拷贝与自己对应的partition中的数据即可。每个Reducer会处理一个或者多个partition，但需要先将自己对应的partition中的数据从每个Map的输出结果中拷贝过来。

接下来就是sort阶段，也成为merge阶段，因为这个阶段的主要工作是执行了归并排序。从Map端拷贝到Reduce端的数据都是有序的，所以很适合归并排序。最终在Reduce端生成一个较大的文件作为Reduce的输入。

最后就是Reduce过程了，在这个过程中产生了最终的输出结果，并将其写到HDFS上。

#### 四、fsimage和edit的区别？

当NN,SN要进行数据同步时叫做checkpoint时就用到了fsimage与edit，fsimage是保存最新的元数据的信息，当fsimage数据到一定的大小时会去生成一个新的文件来保存元数据的信息，这个新的文件就是edit，edit会回滚最新的数据。

#### 五、简单说一下hadoop的map-reduce模型

首先map task会从本地文件系统读取数据，转换成key-value形式的键值对集合，使用的是hadoop内置的数据类型，如Text，LongWritable等。

将键值对集合输入mapper进行业务处理过程，将其转化成需要的key-value再输出。

之后会进行一个partition分区操作，默认使用的是hashpartitioner，可以通过重写hashpartitioner的getPartition方法来自定义分区规则。

之后会对key进行sort排序，grouping分组操作将相同key的value合并分组输出，在这里可以使用自定义的数据类型，重写WritableComparator的Comparator方法来自定义排序规则，重写RawComparator的compara方法来自定义分组规则。

之后进行一个combiner归约操作，就是一个本地的reduce预处理，以减小shuffle，reducer的工作量。

Reduce task会用网络将各个数据收集进行reduce处理，最后将数据保存或者显示，结束整个job。

#### 六、运行hadoop集群需要哪些守护进程？

DataNode,NameNode,TaskTracker和JobTracker都是运行Hadoop集群需要的守护进程。

#### 七、hadoop的TextInputFormat作用是什么，如何自定义实现？

InputFormat会在map操作之前对数据进行两方面的预处理。

1.是getSplits，返回的是InputSplit数组，对数据进行Split分片，每片交给map操作一次。

2.是getRecordReader，返回的是RecordReader对象，对每个Split分片进行转换为key-value键值对格式传递给map常用的InputFormat是TextInputFormat，使用的是LineRecordReader对每个分片进行键值对的转换，以行偏移量作为键，行内容作为值。

自定义类继承InputFormat接口，重写createRecordReader和isSplittable方法在createRecordReader中可以自定义分隔符。

## 八、hadoop和spark都是并行计算，那么他们有什么相同和区别？

两者都使用mr模型来进行并行计算，hadoop的一个作业称为job，job里面分为map task和reduce task，每个task都是在自己的进程中运行的，当task结束时，进程也会结束。

Spark用户提交的任务称为application，一个application对应一个SparkContext，app中存在多个job，没触发一个action操作就会产生一个job。

这些job可以并行或者串行执行，每个job有多个stage，stage是shuffle过程中DAGScheduler通过RDD之间的依赖关系划分job而来的，每个stage里面有多个task，组成taskset有TaskScheduler分发到各个executor中执行，executor的生命周期是和application一样的，即使没有job运行也是存在的，所以task可以快速启动读取内存进行计算的。

Hadoop的job只有map和reduce操作，表达能力比较欠缺而且在mr过程中会重复的读写hdfs，造成大量的io操作，多个job需要自己管理关系。

Spark的迭代计算都是在内存中进行的，API中提供了大量的RDD操作join，groupby等，而且通过DAG图可以实现良好的容错。

## 九、为什么要用flume导入hdfs，hdfs的架构是怎样的？

Flume可以实时的导入数据到hdfs中，当hdfs上的文件达到一个指定大小的时候会形成一个文件，或者超时所指定时间的话也形成一个文件。

文件都是存储在datanode上的，namenode存储着datanode的元数据信息，而namenode的元数据信息是存在内存中的，所以当文件切片很小或者很多的时候会卡死。

## 十、MR程序运行的时候会有什么比较常见的问题？

比如说作业中大部分都完成了，但是总有几个reduce一直在运行。

这是因为这几个reduce中的处理的数据要远远大于其他的reduce，可能是对键值对任务划分的不均匀造成的数据倾斜。

解决的方法可以在分区的时候重新定义分区规则对于value数据很多的key可以进行拆分、均匀打散等处理，或者是在map端的combiner中进行数据预处理的操作。

## 十一、简单说一下hadoop和spark的shuffle过程

Hadoop：map端保存分片数据，通过网络收集到reduce端。

Spark：spark的shuffle实在DAGScheduler划分Stage的时候产生的，TaskScheduler要分发Stage到各个worker的executor。减少shuffle可以提高性能。

## 十二、hive中存放的是什么？

表。

存的是和hdfs的映射关系，hive是逻辑上的数据仓库，实际操作的都是hdfs上的文件，HQL就是用SQL语法来写的MR程序。

## 十三、Hive与关系型数据库的关系？

没有关系，hive是数据仓库，不能和数据库一样进行实时的CRUD操作。

是一次写入多次读取的操作，可以看成是ETL的工具。

#### 十四、Flume的工作机制是什么？

核心概念是agent，里面包括source，channel和sink三个组件。

Source运行在日志收集节点进行日志采集，之后临时存储在channel中，sink负责将channel中的数据发送到目的地。

只有发送成功channel中的数据才会被删除。

首先书写flume配置文件，定义agent、source、channel和sink然后将其组装，执行flume-ng命令。

#### 十五、Hbase行键列族的概念，物理模型，表的设计原则？

行键：是hbase表自带的，每个行键对应一条数据。

列族：是创建表时指定的，为列的集合，每个列族作为一个文件单独存储，存储的数据都是字节数组，其中数据可以有很多，通过时间戳来区分。

物理模型：整个hbase表会拆分成多个region，每个region记录着行键的起始点保存在不同的节点上，查询时就是对各个节点的并行查询，当region很大时使用.META表存储各个region的起始点，-ROOT又可以存储.META的起始点。

Rowkey的设计原则：各个列族数据平衡，长度原则、相邻原则，创建表的时候设置表放入regionserver缓存中，避免自动增长和时间，使用字节数组代替string，最大长度64kb，最好16字节以内，按天分表，两个字节散列，四个字节存储时分毫秒。

列族的设计原则：尽可能少(按照列族进行存储，按照region进行读取，不必要的io操作)，经常和不经常使用的两类数据放入不同列族中，列族名字尽可能短。

#### 十六、请列出正常的hadoop集群中hadoop都分别需要启动 哪些进程，他们的作用分别都是什么，请尽量列的详细一些。

namenode：负责管理hdfs中文件块的元数据，响应客户端请求，管理datanode上文件block的均衡，维持副本数量

Secondname:主要负责做checkpoint操作；也可以做冷备，对一定范围内数据做快照性备份。

Datanode:存储数据块，负责客户端对数据块的io请求

Jobtracker :管理任务，并将任务分配给 tasktracker。

Tasktracker: 执行JobTracker分配的任务。

Resource manager、Node manager、Journal node、Zookeeper、Zkfc

#### 十七、请说明hive中Sort By、Order By、Cluster By、Distribute By各代表什么意思？

order by：会对输入做全局排序，因此只有一个reducer（多个reducer无法保证全局有序）。只有一个reducer，会导致当输入规模较大时，需要较长的计算时间。

sort by：不是全局排序，其在数据进入reducer前完成排序。

distribute by：按照指定的字段对数据进行划分输出到不同的reduce中。

cluster by：除了具有 distribute by 的功能外还兼具 sort by 的功能。

## 十八、HBase简单读写流程？

读：

找到要读数据的region所在的RegionServer，然后按照以下顺序进行读取：先去BlockCache读取，若BlockCache没有，则到Memstore读取，若Memstore中没有，则到HFile中去读。

写：

找到要写数据的region所在的RegionServer，然后先将数据写到WAL(Write-Ahead Logging，预写日志系统)中，然后再将数据写到Memstore等待刷新，回复客户端写入完成。

## 十九、您对“大数据”一词有何了解？

答：大数据是与复杂和大型数据集相关的术语。关系数据库无法处理大数据，这就是使用特殊工具和方法对大量数据执行操作的原因。大数据使公司能够更好地了解其业务，并帮助他们从定期收集的非结构化和原始数据中获取有意义的信息。大数据还允许公司采取数据支持的更好的业务决策。

## 二十、大数据的五个V是什么？

答：大数据的五个V如下：

- Volume -Volume表示体积大，即以高速率增长的数据量，即以PB为单位的数据量
- Velocity -Velocity是数据增长的速度。社交媒体在数据增长速度方面发挥着重要作用。
- Variety -Variety是指不同的数据类型，即各种数据格式，如文本，音频，视频等。
- Veracity -Veracity是指可用数据的不确定性。由于大量数据带来不完整性和不一致性，因此产生了准确性。
- Value -价值是指将数据转化为价值。通过将访问的大数据转换为价值，企业可以创造收入。

大数据的5V

**注意：** 这是大数据访谈中提出的基本和重要问题之一。如果您看到面试官有兴趣了解更多信息，您可以选择详细解释五个V。但是，如果您被问及“大数据”这一术语，甚至可以提及这些名称。

## 二十一、告诉我们大数据和Hadoop如何相互关联。

答：大数据和Hadoop几乎是同义词。随着大数据的兴起，专门从事大数据操作的Hadoop框架也开始流行起来。专业人员可以使用该框架来分析大数据并帮助企业做出决策。

**注意：** 这个问题通常在大数据访谈中提出。可以进一步去回答这个问题，并试图解释的Hadoop的主要组成部分。

## 二十二、大数据分析如何有助于增加业务收入？

答：大数据分析对企业来说非常重要。它可以帮助企业将自己与众不同并增加收入。通过预测分析，大数据分析为企业提供定制的建议和建议。此外，大数据分析使企业能够根据客户需求和偏好推出新产品。这些因素使企业获得更多收入，因此公司正在使用大数据分析。通过实施大数据分析，公司可能会收入大幅增加5-20%的收入。一些使用大数据分析来增加收入的受欢迎公司是 - 沃尔玛，LinkedIn，Facebook，Twitter，美国银行等。

## 二十三、解释部署大数据解决方案时应遵循的步骤。

答：以下是部署大数据解决方案所遵循的三个步骤

## I、数据摄取

部署大数据解决方案的第一步是数据提取，即从各种来源提取数据。数据源可以是像Salesforce这样的CRM，像SAP这样的企业资源规划系统，像MySQL这样的RDBMS或任何其他日志文件，文档，社交媒体源等。数据可以通过批处理作业或实时流来提取。然后将提取的数据存储在HDFS中。

## II、数据存储

在数据摄取之后，下一步是存储提取的数据。数据存储在HDFS或NoSQL数据库（即HBase）中。HDFS存储适用于顺序访问，而HBase适用于随机读/写访问。

## III、数据处理

部署大数据解决方案的最后一步是数据处理。数据通过Spark，MapReduce，Pig等处理框架之一进行处理。

## 二十四、定义HDFS和YARN的相应组件

答：HDFS的两个主要组成部分：

- NameNode - 这是用于处理HDFS内数据块的元数据信息的主节点
- DataNode / Slave节点 - 这是作为从节点存储数据的节点，供NameNode处理和使用

除了提供客户端请求之外，NameNode还执行以下两个角色之一：

- CheckpointNode - 它在与NameNode不同的主机上运行
- BackupNode-它是一个只读的NameNode，它包含不包括块位置的文件系统元数据信息

YARN的两个主要组成部分：

- ResourceManager-该组件接收处理请求，并根据处理需要相应地分配给各个NodeManager。
- NodeManager-它在每个单个数据节点上执行任务

## 二十五、为什么Hadoop可用于大数据分析？

答：由于数据分析已成为业务的关键参数之一，因此，企业正在处理大量结构化，非结构化和半结构化数据。在Hadoop主要支持其功能的情况下，分析非结构化数据非常困难

- 存储
- 处理
- 数据采集

此外，Hadoop是开源的，可在商用硬件上运行。因此，它是企业的成本效益解决方案。

## 二十六、什么是fsck？

答：fsck代表文件系统检查。它是HDFS使用的命令。此命令用于检查不一致性以及文件中是否存在任何问题。例如，如果文件有任何丢失的块，则通过此命令通知HDFS。

## 二十七、NAS（网络附加存储）和HDFS之间的主要区别是什么？

答：NAS（网络附加存储）和HDFS之间的主要区别 -

- HDFS在一组计算机上运行，而NAS在单个计算机上运行。因此，数据冗余是HDFS中的常见问题。相反，复制协议在NAS的情况下是不同的。因此，数据冗余的可能性要小得多。
- 在HDFS的情况下，数据作为数据块存储在本地驱动器中。在NAS的情况下，它存储在专用硬件中。

## 二十八、格式化NameNode的命令是什么？

答：\$ hdfs namenode -format。

