



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

第三版：大数据 80 道

hdfs 运行原理

- 1、HDFS 自动保存多个副本，移动计算。缺点是小文件存取占用 namenode 内存，写入只支持追加，不能随机修改。
- 2、它存储的逻辑空间称为 block，文件的权限类似 linux。整体架构分三种节点，NN,SNN,DN
- 3、NN 负责读写操作保存 metadata(Ownership Permission blockinfo)
- 4、SNN 负责辅助 NN 合并 fsimage 和 edits，减少 nn 启动时间
- 5、DN 负责存数据，每个数据（文件）分割成若干 block，每个 block 默认 3 个副本。启动后像 NN 发送心跳保持联系
- 6、NN 保存的 metadata 在 hdfs 启动后加载到计算机内存，除 block 位置信息的 metadata 保存在 OS 文件系统中的 fsimage 文件中，对 metadata 的操作日志保存在 OS 文件系统中的 edits 文件中。block 位置信息是 hdfs 启动后由 DN 上报 NN 再加载到内存的。
- 7、HDFS 的安全模式：直到 NN 完全加载完 metadata 之前的这段时间。期间不能写入文件，DN 检查各个 block 完整性，并修复

MapReduce 运行原理

离线计算框架，过程分为 split map shuffle reduce 四个过程

关注公众号：磊哥聊编程，回复：面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

架构节点有：Jobtracker TaskTracker

Split 将文件分割，传输到 mapper，mapper 接收 KV 形式的数 据，经过处理，再传到 shuffle 过程。

Shuffle 先进行 HashPartition 或者自定义的 partition，会有数据倾斜和 reduce 的负载均衡问题；再进行排序，默认按字典排序；为减少 mapper 输出数据，再根据 key 进行合并，相同 key 的数据 value 会被合并；最后分组形成 (key,value{}) 形式的数据，输出到下一阶段

Reduce 输入的数据就变成了，key+迭代器形式的数据，再进行处理。

Hadoop 中的 MapReduce 是一个使用简易的软件框架，基于它写出来的应用程序能够运行在由上千个商用机器组成的大型集群上，并以一种可靠容错的式并行处理上 T 级别的数据集。

一个 MapReduce 作业 (job) 通常会把输入的数据集切分为若干独立的数据块，由 map 任务 (task) 以完全并行的方式处理它们。框架会对 map 的输出先进行排序，然后把结果输入给 reduce 任务。通常作业的输入和输出都会被存储在文件系统中。整个框架负责任务的调度和监控，以及重新执行已经失败的任务。

通常，MapReduce 框架和分布式文件系统是运行在一组相同的节点上的，也就是说，计算节点和存储节点通常在一起。这种配置允许框架在那些已经存好数据的节点上高效地调度任务，这可以使整个集群的网络带宽被非常高效地利用。

MapReduce 框架由一个单独的 master JobTracker 和每个集群节点一个 slave TaskTracker 共同组成。master 负责调度构成一个作业的所有任务，这些任务分布在不同的 slave 上，master 监控它们的执行，重新执行已经失败的任务。而 slave 仅负责执行由 master 指派的任务

关注公众号：磊哥聊编程，回复：面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

MapReduce 如何优化

1、Combiner 优化

在之前的 shuffle 过程中的 map 端生成最终溢写文件时，例如：aaa 1, aaa 1 这样的数据会被合并为{aaa [1,1]}，而我们可以给每个 map 显示的设定 combiner，来将这个节点的输出优化，即如前面的例如，如果他们是一节点的数据，他们会合并为{aaa, 2}，combiner 相当于小型的 reduce，不过他们这要求 reduce 的输入数据格式和输出数据格式一致，并且 value 的值是满足结合律的

2、压缩

通过对数据压缩，来减少网络 IO，减少磁盘占用，可以用在三个地方：输入数据压缩、map 输出压缩、reduce 输出压缩。当然压缩也会增加 CPU 计算

3、map 的数量

Map task 的数量使用 split 的数量决定的，split 的数据越少，每个 map task 的执行时间就越短，但是相应的 job 的执行时间也会被拉长，因为内部调度的时间更长了。

4、-reduce 的数量

可以在 client 设置，增加 reduce 的数量

5、大量小文件的优化

Mapreduce 不怕大文件，就怕小文件，默认情况下，TextInputFormat 对任务的切片机制是按照文件规划切片，不管有多少个小文件，都会是单独的切片，都会交给一个 maptask，这样，如果有大量的小文件，就会产生大量的 maptask，处

关注公众号：磊哥聊编程，回复³：面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

理效率极端底下

最好的方法：在数据处理的最前端（预处理、采集），就将小文件合并成大文件，在上传到 HDFS 做后续的分析，如果已经是大量的小文件在 HDFS 中了，可以使用另一种 inputformat 来做切片（CombineFileInputformat），它的切片逻辑跟 Fileinputformat 不同：它可以将多个小文件从逻辑上规划到一个切片中，这样，多个小文件就可以交给一个 maptask 了

1、 hadoop 中 combiner 的作用

2、 combiner 最基本是实现本地 key 的聚合，对 map 输出的 key 排序，value 进行迭代

3、 combiner 还具有类似本地的 reduce 功能。

4、 如果不用 combiner，那么，所有的结果都是 reduce 完成，效率会相对低下。使用 combiner，先完成的 map 会在本地聚合，提升速度。

对于 hadoop 自带的 wordcount 的例子，value 就是一个叠加的数字，所以 map 一结束就可以进行 reduce 的 value 叠加，而不必要等到所有的 map 结束再去进行 reduce 的 value 叠加。

combiner 使用的合适，可以在满足业务的情况下提升 job 的速度，如果不合适，则将导致输出的结果不正确。

hadoop 进程名

关注公众号：磊哥聊编程，回复：面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

Datanode

负责存储文件

a.DataNode 的需要完成的首要任务是 K-V 存储

b.完成和 namenode 通信，这个通过 IPC 心跳连接实现。

此外还有和客户端 其它 datanode 之前的信息交换

c.完成和客户端还有其它节点的大规模通信，这个需要直接通过 socket 协议实现。

SecondaryNamenode

合并快照

namenode

相当于一个领导者，负责调度

NodeManager

是 YARN 中每个节点上的代理，它管理 Hadoop 集群中单个计算节点包括与 ResourceManger 保持通信，监督 Container 的生命周期管理，监控每个 Container 的资源使用（内存、CPU 等）情况，追踪节点健康状况，管理日志和不同应用程序用到的附属服务（auxiliary service）

ResourceManager

在 YARN 中，ResourceManager 负责集群中所有资源的统一管理和分配，它接收来自各个节点（NodeManager）的资源汇报信息，并把这些信息按照一定的策略分配给各个应用程序（实际上是 ApplicationManager）

RM 与每个节点的 NodeManagers (NMs)和每个应用的 ApplicationMasters (AMs)一起工作。

a.NodeManagers 遵循来自 ResourceManager 的指令来管理单一节点上的可用资源。

b.ApplicationMasters 负责与 ResourceManager 协商资源与 NodeManagers 合作启动容器

关注公众号：磊哥聊编程，回复：面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

我们开发 Job 是否能去掉 reduce 阶段

可以去掉。设置 reduce 数为 0 即可

datanode 在什么情况下不会备份

强制关闭或非正常断电时

3 个 datanode 中有一个 datanode 出现错误会怎样?

这个 datanode 的数据会在其他的 datanode 上重新做备份。

Hive 生产环境中为什么建议使用外部表?

- 1、因为外部表不会加载数据到 hive，减少数据传输，数据还能共享。
- 2、hive 不会修改数据，所以无需担心数据的损坏
- 3、删除表时，只删除表结构、不删除数据。

MapReduce 的 map 数量和 reduce 数量怎么确定，怎么配置

map 的数量有数据块决定，reduce 数量由配置文件配置

combiner 出现在哪个过程

关注公众号：磊哥聊编程，回复：面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

出现在 map 阶段的 map 方法后，shuffle 过程

请列出你在工作中使用过的开发 mapreduce 的语言

java, hive, (python, c++) hadoop streaming

请简述 mapreduce 中的 combine 和 partition 的作用

答：combiner 是发生在 map 的最后一个阶段，其原理也是一个小型的 reducer，主要作用是减少输出到 reduce 的数据量，缓解网络传输瓶颈，提高 reducer 的执行效率。

partition 的主要作用将 map 阶段产生的所有 kv 对分配给不同的 reducer task 处理，可以将 reduce 阶段的处理负载进行分摊

hive 内部表和外部表的区别

Hive 向内部表导入数据时，会将数据移动到数据仓库指向的路径；若是外部表，数据的具体存放目录由用户建表时指定

在删除表的时候，内部表的元数据和数据会被一起删除，

而外部表只删除元数据，不删除数据。

这样外部表相对来说更加安全些，数据组织也更加灵活，方便共享源数据。

Hbase 的 rowKey 怎么创建比较好？列簇怎么创建比较好？

关注公众号：磊哥聊编程，回复：面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

rowKey 最好要创建有规则的 rowKey，即最好是有序的。

经常需要批量读取的数据应该让他们的 rowkey 连续；

将经常需要作为条件查询的关键词组织到 rowkey 中；

列族的创建：

按照业务特点，把数据归类，不同类别的放在不同列族

用 mapreduce 怎么处理数据倾斜问题

本质：让各分区的数据分布均匀

可以根据业务特点，设置合适的 partition 策略

如果事先根本不知道数据的分布规律，利用随机抽样器抽样后生成 partition 策略再处理

hbase 内部机制是什么

hbase 是一个能适应联机业务的数据库系统

物理存储：hbase 的持久化数据是存放在 hdfs 上

存储管理：一个表是划分为很多 region 的，这些 region 分布式地存放在很多 regionserver 上

关注公众号：磊哥聊编程，回复：面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

Region 内部还可以划分为 store，store 内部有 memstore 和 storefile

版本管理：hbase 中的数据更新本质上是不断追加新的版本，通过 compact 操作来做版本间的文件合并

Region 的 split

hadoop 中，有哪些地方使用到了缓存机制，作用分别是什么？

答：缓存机制就是 DistributedCache，就是在 job 任务执行前，将需要 Task 机器上进行缓存，提高 mapreduce 的执行效率

如何确定 hadoop 集群的健康状态

UI 监控 shell 监控

shuffle 阶段你怎么理解

shuffle 过程包括在 Map 和 Reduce 两端中。

在 Map 端的 shuffle 过程是对 Map 的结果进行分区（partition）、排序（sort）和分割（spill），然后将属于同一个划分的输出合并在一起

（merge）并写在硬盘上，同时按照不同的划分将结果发送给对应的 Reduce（Map 输出的划分与 Reduce 的对应关系由 JobTracker 确定）。

Reduce 端又会将各个 Map 送来的属于同一个划分的输出进行合并（merge），然后对 merge 的结果进行排序，最后交给 Reduce 处理。通俗的讲，就是对 Map 输出结果先进行分区（partition），如“aaa”经过 Partitioner 后返回 0，也就是这对值应当交由第一个 reducer 来处理。接下来，

需要将数据写入内存缓冲区中，缓冲区的作用是批量收集 map 结果，减少磁盘 IO

关注公众号：磊哥聊编程，回复：面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

的影响。我们的 key/value 对以及 Partition 的结果都会被写入缓冲区。当然写入之前，key 与 value 值都会被序列化成字节数组。这个内存缓冲区是有大小限制的，默认是 100MB。当 map task 的输出结果很多时，需要在一定条件下将缓冲区中的数据临时写入磁盘，然后重新利用这块缓冲区。这个从内存往磁盘写数据的过程被称为 Spill。Spill 可以认为是一个包括 Sort 和 Combiner（Combiner 是可选的，用户如果定义就有）的过程。先进行 sort 可以把缓冲区中一段范围 key 的数据排在一起，（如果数据多的时候，多次刷新往内存缓冲区中写入的数据可能会有属于相同范围的 key，也就是说，多个 spill 文件中可能会有统一范围的 key，这就是需要下面 Map 端 merge 的原因），这里有点绕，具体的介绍可以看下面的详细过程，执行过 sort 之后，如果用户定义了 combiner 就会执行 combine，然后执行 merge 操作，接着就是 Reduce 端。

请列出正常工作的 hadoop 集群中 hadoop 都需要启动哪些进程，他们的作用分别是什么？

NameNode: HDFS 的守护进程，负责记录文件是如何分割成数据块，以及这些数据块分别被存储到那些数据节点上，它的主要功能是对内存及 IO 进行集中管理

Secondary NameNode: 辅助后台程序，与 NameNode 进行通信，以便定期保存 HDFS 元数据的快照。

DataNode: 负责把 HDFS 数据块读写到本地的文件系统。

JobTracker: 负责分配 task，并监控所有运行的 task。

TaskTracker: 负责执行具体的 task，并与 JobTracker 进行交互。

关注公众号：磊哥聊编程，回复：¹⁰面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

请列出你所知道的 hadoop 调度器，并简要说明其工作方法？

比较流行的三种调度器有：默认调度器 FIFO，计算能力调度器 CapacityScheduler，公平调度器 Fair Scheduler

默认调度器 FIFO

hadoop 中默认的调度器，采用先进先出的原则

计算能力调度器 CapacityScheduler

选择占用资源小，优先级高的先执行

公平调度器 FairScheduler

同一队列中的作业公平共享队列中所有资源

简答说一下 hadoop 的 map-reduce 编程模型

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，比如 longwritable、text 等

将键值对集合输入 mapper 进行业务处理过程，将其转换成需要的 key-value 在输出

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

关注公众号：磊哥聊编程，回复：¹¹面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

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，其实就是一个本地段的 reduce 预处理，以减小后面 shuffle 和 reducer 的工作量

reduce task 会通过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

为什么要用 flume 导入 hdfs, hdfs 的构架是怎样的

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，或者超过指定时间的话也形成一个文件

文件都是存储在 datanode 上面的，namenode 记录着 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

map-reduce 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，可能是因为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

关注公众号：磊哥聊编程，回复：¹²面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

Hive 中存放是什么？

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。

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 CURD 操作。

是一次写入多次读取的操作，可以看成是 ETL 工具。

Sqoop 工作原理是什么？

hadoop 生态圈上的数据传输工具。

可以将关系型数据库的数据导入非结构化的 hdfs、hive 或者 bbase 中，也可以将 hdfs 中的数据导出到关系型数据库或者文本文件中。

使用的是 mr 程序来执行任务，使用 jdbc 和关系型数据库进行交互。

import 原理：通过指定的分隔符进行数据切分，将分片传入各个 map 中，在 map 任务中在每行数据进行写入处理没有 reduce。

export 原理：根据要操作的表名生成一个 java 类，并读取其元数据信息和分隔符对非结构化的数据进行匹配，多个 map 作业同时执行写入关系型数据库

Hadoop 性能调优？

调优可以通过系统配置、程序编写和作业调度算法来进行。

hdfs 的 block.size 可以调到 128/256（网络很好的情况下，默认为 64）

调优的大头：mapred.map.tasks、mapred.reduce.tasks 设置 mr 任务数（默认

关注公众号：磊哥聊编程，回复：¹³面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

都是 1)

mapred.tasktracker.map.tasks.maximum 每台机器上的最大 map 任务数

mapred.tasktracker.reduce.tasks.maximum 每台机器上的最大 reduce 任务数

mapred.reduce.slowstart.completed.maps 配置 reduce 任务在 map 任务完成到百分之几的时候开始进入

这个几个参数要看实际节点的情况进行配置，reduce 任务是在 33%的时候完成 copy，要在这之前完成 map 任务，（map 可以提前完成）

mapred.compress.map.output,mapred.output.compress 配置压缩项，消耗 cpu 提升网络和磁盘 io

合理利用 combiner

注意重用 writable 对象

hadoop 数据倾斜及解决办法

1、 增加 jvm 内存,这适用于第一种情况(唯一值非常少，极少数值有非常多的记录值(唯一值少于几千)),这种情况下,往往只能通过硬件的手段来进行调优,增加 jvm 内存可以显著的提高运行效率。

2、 增加 reduce 的个数,这适用于第二种情况(唯一值比较多，这个字段的某些值有远远多于其他值的记录数,但是它的占比也小于百分之一或千分之一),我们知道,这种情况下,最容易造成的结果就是大量相同 key 被 partition 到一个分区,从而一个 reduce 执行了大量的工作,而如果我们增加了 reduce 的个数,这种情况相对来说会减轻很多,毕竟计算的节点多了,就算工作量还是不均匀的,那也要小很多。

3、 自定义分区,这需要用户自己继承 partition 类,指定分区策略,这种方式效果比较显著。

4、 重新设计 key,有一种方案是在 map 阶段时给 key 加上一个随机数,有了随机数的 key 就不会被大量的分配到同一节点(小几率),待到 reduce 后再把随机数去掉即可。

关注公众号：磊哥聊编程，回复：¹⁴面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

5、使用 combiner 合并,combiner 是在 map 阶段,reduce 之前的一个中间阶段,在这个阶段可以选择性的把大量的相同 key 数据先进行一个合并,可以看做是 local reduce,然后再交给 reduce 来处理,这样做的好处很多,即减轻了 map 端向 reduce 端发送的数据量(减轻了网络带宽),也减轻了 map 端和 reduce 端中间的 shuffle 阶段的数据拉取数量(本地化磁盘 IO 速率),推荐使用这种方法。

hive 是什么

hive 是基于 hadoop 的一个数据仓库工具,可以将结构化数据文件映射为一张表,提供 sql 查询功能。

hive 利用 hdfs 存储,利用 mr 查询,将 sql 转为 mr 程序执行,比 mr 开发效率高

hive 的原数据存储

hive 中的原数据存放在关系型数据库中 MySQL、derby 中,元数据包括 表的名字,表的列和分区以及属性,表的属性是否是外部表,表的数据所在目录

hive 与传统 DB 的区别

传统数据库: OLTP 面向事务操作型处理,主要支持业务,面向业务

HIVE: OLAP 面向分析型处理,就是数据仓库,面对的是历史数据,一部分数据来自数据库开展分析

hive 数仓开发的基本流程

关注公众号: 磊哥聊编程, 回复: ¹⁵面试题, 获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

从业务系统获取数据

sqoop 从关系型数据库中 导入数据到 hdfs 中,也可以将 hdfs 数据倒进到关系型数据库中

flume:采集数据存放到 hdfs 中

ftp: 从文件服务器上下载分析所需的数据

数据存储

分为 ods 层 , dw 层, da 层

源数据层: 没有格式化的数据, 不利于分析

数据仓库层: 来自 ODS 层要经过 ETL 的过程 格式统一 数据规则

数据应用层: 使用 DW 层, 数据使用者

写 sql

配置调度系统

导出数据展示

hive sql 知识点

DML 数据操纵语言

DDL 数据定义语言, 用语定义和管理数据库中的对象

hive 的内表和外表



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

内部表：未被 external 修饰，由 hive 自身管理，删除后会直接删除元数据及存储数据，对表的修改会将修改直接同步给元数据的内部表

外表：被 ex 修饰，由 hdfs 管理，表存储有自己决定，删表仅会删除元数据，hdfs 文件并不会被删除

为什么 hive 的分区

为了避免 select 扫描全表，hive 提出了分区表 partitionedby 的概念，给文件归类打上表示

静态分区：

单分区建表

```
create table par_tab(name string,nation string) partitioned by (sex string)
row format delimited fields terminated by ' ' ;
```

```
加载：load data local inpath '/hdfs/...' into table par_tab partition(sex=
'man' )
```

在创建分区表的时候，系统会在 hive 数据仓库默认路径/user/hive/warehouse/ 创建目录，在创建 sex=man 的目录，最后在分区名下存放实际的数据文件

多分区建表

```
create table par_tab(name string,nation string) partitioned by (sex
string,dt string) row format delimited fields terminated by ';;
```

```
load data local inpath '/hdfs/...' into table par_tab
partition(sex='man',dt="2019-08-08")
```

当我们查询所有的 man 时候，man 下的所有日期下的数据都会被查出来；如果只查询日期分区，那么 hive 会对路径进行修剪，从而只扫描日期分区，性别分区不做过滤

关注公众号：磊哥聊编程，回复：¹⁷面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

动态分区

动态分区与静态分区区别就是不指定分区目录，有系统自己选择
开启动态分区 `set hive.exec.dynamic.partition=true`

hive 为何分区

避免数据倾斜，查询效率提升

什么是 udf

当 hive 提供的函数无法满足业务需求，可以选择 udf 函数，用户自定义函数；需要继承 udf 类

hive 如何优化

排序优化：sort by 高于 order by

小文件合并大文件

打开 map 端的 combine 合并

使用静态分区，建立好分区一个分区对应 hdfs 上的一个目录，减少 job 和 task 数量；使用表连接操作

解决 groupby 数据倾斜问题：设置 `hive.groupby.skewindata=true`，那么 hive 会自动均衡负载，小文件合并为大文件：表连接操作，使用 udf 或 udaf 函数

hive 跟 hbase 的区别

共同点都是用 hadoop 作为底层存储

关注公众号：磊哥聊编程，回复：¹⁸面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

区别：hive 是为了减少 mrjobs 编写工作的批处理系统，处理速度慢。hive 本身不存储数据和计算数据，依赖于 hadoop，纯逻辑表

hbase 是为了 hadoop 对实时操作的缺陷的项目，处理速度快，是物理表，提供一个超大的内存 hash 表，方便查询操作

如果全表扫描用 hive+hadoop

如果用索引查询与 hbase+hadoop

是处理数据库文件还是读取文本文件

先读取文本文件进行清洗，然后放入 hdfs，进行处理

或者直接读取 MySQL 中格式化数据

hive 底层与数据库交互原理

Hive 是用 SQL 替代写 MR 的编程框架，做 Hadoop 上会把用户提交的 SQL 语句做语法分析，执行计划等一堆乱七八糟的事后变成 MR job 提交去跑，返回结果给用户。不然每次都写 MR 很麻烦的，有这个写个 SQL 就可以拿到等效的效果

hive 有哪些保存元数据的方式，各有什么特点

- 1、内存数据库 derby，安装小不稳定
- 2、MySQL 数据库，持久化好，查看方便

生产环境为什么建议使用外部表

- 1、因为表数据不会加载到 hive，减少数据传输，数据能共享

关注公众号：磊哥聊编程，回复：¹⁰面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

2、hive 不会修改数据，所以无需担心数据损坏

3、删除表示，只删除表结构，不删除数据

数据导入 hive 的方式

1、本地数据导入 hive

2、hdfs 中导入 hive

3、hbase 导入 hive

4 查询方式导入 hive

你们数据库如何导入 hive 的出现什么错误

通过 sqoop 导入，如果数据库中有 bold 和 text 字段会报错，导致导入的时候出现缓存不够的情况，查看 sqoop 文档，加上 limit 属性解决

MySQL 权限 问题

grant all privileges on , to 'root' '@' '%' identified by 'root' with grant option;

.代表所有数据库的所有表。identified by 'root' root 是密码。

hive 中定义的表字段属性与 MySQL 中定义的字段的属性不一致，导致 MySQL 数据库中的个别字段存不下

hadoop 环境整合了 hbase 和 hive，是否有必要给 hdfs 和 hbase 都分配压缩策略，请给出压缩策略的建议

hdfs 在存储的时候不会将数据进行压缩，如果想进行压缩，我们可以在向 hdfs

关注公众号：磊哥聊编程，回复：²⁰面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

传输时候进行压缩

采用压缩流 宝典 66 页

采用序列化方法

hbase 为列存储数据库，本身存在压缩机制，所以无需设计

简述 hive 中的虚拟列的作用？使用它注意事项

三个虚拟列

INPUT_FILE_NAME: mapper 任务的输出文件名

BLOCK_OFFSET_INSIDE_FILE: 当前全局文件偏移量，对于块压缩文件，就是当前快的文件偏移量，及当前块的第一个字节在文件中的偏移量

ROW_OFFSET_INSIDE_BLOCK, 默认不开启，设置 hive.exec.rowoffset 为 true 可用，可以用来排查有问题的输入数据

Redis, 传统数据库, hbase, hive 每个之间的区别

Redis 是缓存

hbase 是列式数据库，存在 hdfs 上，写入速度快，数据量大，查询速度快

hive 是数据仓库，是用来分析数据，不是增删改查数据的

hive 使用版本

使用外部表，hive1.0 hive1.3

insert into 和 override write 区别

关注公众号：磊哥聊编程，回复：²¹面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

insert into：将某一个表的数据写到另一个表

override write：覆盖之前的表

假如一个分区的数据逐步错误怎么通过 hivesql 删除

alter table ptable drop partition(daytime= '20140921',city= 'bj');全部删除，文件夹还在

简述 hadoop spark storm hive 的特点及使用场景

hadoop 是一种分布式系统基础架构当处理海量数据对的程序，开始要求高可靠，高扩展，高效，低容错，低成本场景

MapReduce 是一种编程模型，用于大规模数据集的并行计算，目前日志分析居多 spark 拥有 mr 的所具有的优点；但不同于 mr 的是 job 中间输出的结果可以保存到内存中，从而不需要读写 hdfs，由此 spark 能更好的适用于数据挖掘与机器学习等需要迭代式计算，极大的提高效率的场景

storm：一个分布式实时计算系统 storm 是一个任务并行连续计算引擎，storm 并不在 hadoop 集群运行，他是用 Zookeeper 的和自己的主从工作进程，协调拓扑和工作状态

hive 数据仓库

hbase：数据量大，传统数据库无法胜任，联机业务功能开发，离线数据分析

hive 底层与数据库交互原理

hive 有一套自己的 sql 解析引擎成为 metastore，存储在 MySQL 中可以将 sql 转化为 mrjob 任务执行

关注公众号：磊哥聊编程，回复：²²面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

sqoop 在导入到 MySQL 中，如果不重复导入数据，如果数据存在问题，sqoop 如何处理？

Sqoop 从 Hadoop 或 Hive 或 HBase 将数据导出到 MySQL，是导出，你可以查看一下 Sqoop 的增量导出。

如果 Sqoop 的导出你没有设置增量导出，则如果数据表没有设置主键，那么数据重复的时候不会报错，但是如果数据表设置了主键，当数据重复时，将导出不成功

hive 如何控制权限

持简单的权限管理，默认情况下是不开启，这样所有的用户都具有相同的权限，同时也是超级管理员，也就对 hive 中的所有表都有查看和改动的权利，这样是不符合一般数据仓库的安全原则的。Hive 可以是基于元数据的权限管理，也可以基于文件存储级别的权限管理，此次以介绍 MetaData 权限管理为主。通过以下配置开启 Hive 身份认证功能进行权限检查：

你认为用 java ， streaming ， pipe 方式开发 map/reduce ， 各有哪些优点

Java 写 mapreduce 可以实现复杂的逻辑，如果需求简单，则显得繁琐。

HiveQL 基本都是针对 hive 中的表数据进行编写，但对复杂的逻辑（杂）很难进行实现。写起来简单。

关注公众号：磊哥聊编程，回复：²³面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

hive 能像关系型数据库那样创建多个库吗

当然能

简要描述如何安装配置 apache 的一个开源 hadoop，只描述即可，无需列出具体的步骤，列出具体的步骤更好。

- 1、使用 root 账户登录
- 2、修改 IP
- 3、修改 host 主机名
- 4、配置 SSH 免密码登录
- 5、关闭防火墙
- 6、安装 JDK
- 6、解压 hadoop 安装包
- 7、配置 hadoop 的核心文件 `hadoop-env.sh`，`core-site.xml`，`mapred-site.xml`，`hdfs-site.xml`
- 8、配置 hadoop 环境变量
- 9、格式化 `hadoop namenode-format`

关注公众号：磊哥聊编程，回复：²⁴面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

10、启动节点 start-all.sh

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。

答：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。

Resourcemanager

Nodemanager

Journalnode

Zookeeper

Zkfc

请写出以下的 shell 命令

关注公众号：磊哥聊编程，回复：²⁵面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

- 1、 杀死一个 job
- 2、 删除 hdfs 上的 /tmp/aaa 目录
- 3、 加入一个新的存储节点和删除一个节点需要执行的命令
- 4、 `hadoop job -list` 得到 job 的 id, 然后执行 `hadoop job -kill jobId` 就可以杀死一个指定 jobId 的 job 工作了。
- 5、 `hadoopfs -rmr /tmp/aaa`
- 6、 增加一个新的节点在新的几点上执行

```
Hadoop daemon.sh start datanode
```

```
Hadoop daemon.sh start tasktracker/nodemanager
```

下线时, 要在 conf 目录下的 excludes 文件中列出要下线的 datanode 机器主机名

然后在主节点中执行 `hadoop dfsadmin -refreshnodes` 下线一个 datanode

删除一个节点的时候, 只需要在主节点执行

```
hadoop mradmin -refreshnodes 一下线一个 tasktracker/nodemanager
```

请列出你所知道的 hadoop 调度器, 并简要说明其工作方法



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

答：Fifo scheduler :默认，先进先出的原则

Capacity scheduler :计算能力调度器，选择占用最小、优先级高的先执行，依此类推。

Fair scheduler:公平调度，所有的 job 具有相同的资源。

请列出你在工作中使用过的开发 mapreduce 的语言

答：java, hive, (python, c++) hadoop streaming

当前日志采样格式为

a, b, c, d

b, b, f, e

a, a, c, f

请你用最熟悉的语言编写 mapreduce, 计算第四列每个元素出现的个数

答：

```
public class WordCount1 {

public static final String INPUT_PATH = "hdfs://hadoop0:9000/in";
```

关注公众号：磊哥聊编程，回复：²⁷面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

```
public static final String OUT_PATH = "hdfs://hadoop0:9000/out";

public static void main(String[] args) throws Exception {

    Configuration conf = new Configuration();

    FileSystem fileSystem = FileSystem.get(conf);

    if(fileSystem.exists(newPath(OUT_PATH))){}

    fileSystem.delete(newPath(OUT_PATH),true);

    Job job = new Job(conf,WordCount1.class.getSimpleName());

    //1.0 读取文件，解析成 key,value 对

    FileInputFormat.setInputPaths(job,newPath(INPUT_PATH));

    //2.0 写上自己的逻辑，对输入的可以，value 进行处理，转换成新的
    key,value 对进行输出

    job.setMapperClass(MyMapper.class);

    job.setMapOutputKeyClass(Text.class);

    job.setMapOutputValueClass(LongWritable.class);

    //3.0 对输出后的数据进行分区
```

关注公众号：磊哥聊编程，回复：²⁸面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

//4.0 对分区后的数据进行排序，分组，相同 key 的 value 放到一个集合中

//5.0 对分组后的数据进行规约

//6.0 对通过网络将 map 输出的数据拷贝到 reduce 节点

//7.0 写上自己的 reduce 函数逻辑，对 map 输出的数据进行处理

```
job.setReducerClass(MyReducer.class);
```

```
job.setOutputKeyClass(Text.class);
```

```
job.setOutputValueClass(LongWritable.class);
```

```
FileOutputFormat.setOutputPath(job,new Path(OUT_PATH));
```

```
job.waitForCompletion(true);
```

```
}
```

```
static class MyMapper extends Mapper<LongWritable, Text, Text,  
LongWritable>{
```

```
    @Override
```

```
    protected void map(LongWritable k1, Text v1,
```

```
org.apache.hadoop.mapreduce.Mapper.Context context)
```

关注公众号：磊哥聊编程，回复：²⁹面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

```
        throws IOException,InterruptedException {

        String[] split =v1.toString().split("\t");

        for(String words :split){

                context.write(split[3],1);

        }

    }

}

static class MyReducer extends Reducer<Text,LongWritable, Text,
LongWritable>{

        protected void reduce(Text k2,Iterable<LongWritable> v2,

org.apache.hadoop.mapreduce.Reducer.Contextcontext)

                throws IOException,InterruptedException {

        Long count = 0L;

        for(LongWritable time :v2){

                count += time.get();

        }

    }

}
```

关注公众号：磊哥聊编程，回复：³⁰面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

```
    }  
  
    context.write(v2, newLongWritable(count));  
  
    }  
}  
}
```

你认为用 java , streaming , pipe 方式开发 map/reduce , 各有哪些优点

Java 写 mapreduce 可以实现复杂的逻辑，如果需求简单，则显得繁琐。

HiveQL 基本都是针对 hive 中的表数据进行编写，但对复杂的逻辑（杂）很难进行实现。写起来简单。

hive 有哪些方式保存元数据，各有哪些优点

三种：自带内嵌数据库 derby，挺小，不常用，只能用于单节点

请简述 hadoop 怎样实现二级排序（就是对 key 和 value 双排序）

关注公众号：磊哥聊编程，回复：³¹面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

第一种方法是，Reducer 将给定 key 的所有值都缓存起来，然后对它们再做一个 Reducer 内排序。但是，由于 Reducer 需要保存给定 key 的所有值，可能会导致出现内存耗尽的错误。

第二种方法是，将值的一部分或整个值加入原始 key，生成一个组合 key。这两种方法各有优势，第一种方法编写简单，但并发度小，数据量大的情况下速度慢(有内存耗尽的危险)。

第二种方法则是将排序的任务交给 MapReduce 框架 shuffle，更符合 Hadoop/Reduce 的设计思想。这篇文章里选择的是第二种。我们将编写一个 Partitioner，确保拥有相同 key(原始 key，不包括添加的部分)的所有数据被发往同一个 Reducer，还将编写一个 Comparator，以便数据到达 Reducer 后即按原始 key 分组。

68.简述 hadoop 实现 join 的几种方法

Map side join---大小表 join 的场景，可以借助 distributed cache

Reduce side join

请用 java 实现非递归二分查询

```
public class BinarySearchClass
{
    public static int binary_search(int[] array, int value)
    {
```

关注公众号：磊哥聊编程，回复：³²面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

```
int beginIndex = 0;// 低位下标

int endIndex = array.length - 1;// 高位下标

int midIndex = -1;

while (beginIndex <= endIndex) {

    midIndex = beginIndex + (endIndex - beginIndex) / 2;//防止溢出

    if (value == array[midIndex]) {

        return midIndex;

    } else if (value < array[midIndex]) {

        endIndex = midIndex - 1;

    } else {

        beginIndex = midIndex + 1;

    }

}

return -1;

//找到了，返回找到的数值的下标，没找到，返回-1

}
```

关注公众号：磊哥聊编程，回复³³：面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

//start 提示：自动阅卷起始唯一标识，请勿删除或增加。

```
public static void main(String[] args)
{
    System.out.println("Start...");

    int[] myArray = new int[] { 1, 2, 3, 5, 6, 7, 8, 9 };

    System.out.println("查找数字 8 的下标: ");

    System.out.println(binary_search(myArray, 8));
}

//end //提示：自动阅卷结束唯一标识，请勿删除或增加。
}
```

请简述 mapreduce 中的 combine 和 partition 的作用

答：combiner 是发生在 map 的最后一个阶段，其原理也是一个小型的 reducer，主要作用是减少输出到 reduce 的数据量，缓解网络传输瓶颈，提高 reducer 的执行效率。

partition 的主要作用将 map 阶段产生的所有 kv 对分配给不同的 reducer task 处理，可以将 reduce 阶段的处理负载进行分摊

关注公众号：磊哥聊编程，回复：³⁴面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

hive 内部表和外部表的区别

Hive 向内部表导入数据时，会将数据移动到数据仓库指向的路径；若是外部表，数据的具体存放目录由用户建表时指定

在删除表的时候，内部表的元数据和数据会被一起删除，

而外部表只删除元数据，不删除数据。

这样外部表相对来说更加安全些，数据组织也更加灵活，方便共享源数据。

Hbase 的 rowKey 怎么创建比较好？列簇怎么创建比较好？

答：

rowKey 最好要创建有规则的 rowKey，即最好是有序的。

经常需要批量读取的数据应该让他们的 rowkey 连续；

将经常需要作为条件查询的关键词组织到 rowkey 中；

列簇的创建：

按照业务特点，把数据归类，不同类别的放在不同列簇

用 mapreduce 怎么处理数据倾斜问题

本质：让各分区的数据分布均匀

可以根据业务特点，设置合适的 partition 策略

关注公众号：磊哥聊编程，回复³⁵：面试题，获取最新版面试题



微信搜一搜

磊哥聊编程

扫码关注



回复：面试题 获取最新版面试题

如果事先根本不知道数据的分布规律，利用随机抽样器抽样后生成 partition 策略再处理

磊哥聊编程
关注公众号，磊哥聊编程：得最新版，面试题
关注公众号，磊哥聊编程：得最新版，面试题
关注公众号，磊哥聊编程：得最新版，面试题
关注公众号，磊哥聊编程：得最新版，面试题
关注公众号，磊哥聊编程：得最新版，面试题
关注公众号，磊哥聊编程：得最新版，面试题
磊哥聊编程

关注公众号：磊哥聊编程，回复：³⁶面试题，获取最新版面试题