猪八戒网面试:

进去先填了表 个人信息 背景调查之类的

技术面试:

1.自我介绍

2.MapReduce的Shuffle过程

3.如果Reduce个数和分区数不一致时，会发生什么

4.Shffule过程中排序用的什么算法

5.使用过Hive解析JSON串么

6.Java多线程实现方法

7.start和run方法区别

8.线程的生命周期

9.线程之间是如何通信的

10.了解什么是JUC么

11.了解fail-fast机制么？

12.对JVM了解多少

13.公司集群规模，数据量，节点配置（内存大小，硬盘大小，CPU核心数）

14.CDH版本和对应Hadoop版本号

Spark的shuffle有几种方式啊

hadoop集群搭建

技术一直问次日到七日留存率用spark Hive Mr怎么求

问我业务逻辑问题

我第一个项目说的storm 一直问storm

Js埋点

反射和final从哪到哪还有rehash机制和redis的string怎么存储的

Redis的string类型底层是怎么存储的

有个问题，做数仓项目，数据需要先用MR清洗一下再传给ods层吗

然后面过这几家的就有问我为什么你们已经清洗过且和HIVE关联了，不用SQL直接做离线为什么要用Spark

他说的是，你们既然已经清洗过了，为什么还要用Spark core和Spark SQL，直接HIVE SQL不就可以实现了么

58同城

1.说一下最近或者最熟悉的项目？

2.中途根据我对项目的描述问了一些项目的细节？

2.你们一天的数据量多少？

2.项目中有没有遇到什么问题？怎么解决的？

3.说一下你在这个项目中的调优？

4.JVM在项目中是怎么调优的？和Executor的内存占比调优有什么关联？

5.项目完后，就针对简历上的技术点来提问。

6.你了解hive吗？说一下hive的架构？

7.你用hive做过调优吗？说一下做了哪些调优？中间根据回答来做深入的提问。

8.hbase了解多少?说说你知道的hbase的内容？

9.我看你们项目中使用了kafka，为什么离线数据也要存到kafka?而不直接存到hdfs上？

10.你还知道kafka哪一部分的内容？

二面：

1.说一下你们项目中的数据仓库是怎么搭建的，有哪些层？

2.有哪些外部数据？维度表。

3.hive和MR的区别？

4.为什么你们项目中有一部分指标要用MR来做？一部分指标用hive？

5.实时的那个项目中，有没有涉及到广告外链模块的内容？

6.现在有个需求是，实现一个实时显示在page页面上的各广告的点击次数，你怎么做？

7.使用离线的方式怎么做？

8.hive中有没有碰到什么问题？做了哪些调优？

9.对比一下hbase,mysql,redis的索引结构？

10.现在有个需求是要实现十几亿条数据的存储，你有什么解决方案？

11.说一说为什么hbase适合海量数据的查询？

12.如果现在我要统计某一天的确切的uv数,会涉及一个count(distinct)的操作？会导致什么问题？

13.你可以提供其他的方案吗？

58同城面试

1.Spark调优的几种方法？

2.Jvm调优了解哪些？

3.JVM内存中存储内存和执行内存保存哪些数据，如果有堆外内存，存储内存还有嘛？·

3.开发中遇到过哪些问题，如何解决的?

4.SparkStreaming如何获取数据的，两种方式有哪些区别？

5.SparkStreaming拉取数据具体调用了哪个api方法，多长时间拉取一次数据。

5.Driver如何与Executer进行联系的。

6.页面单跳转化率是如何实现的，如何进行优化？

7.处理超大型数据如何提高系统效率，如何调优？

8.Spark读取数据生成RDD分区默认多少？

9.快速排序和冒泡排序的时间复杂度是多少，哪个更快？

10.网络编程方面了解哪些，服务器线程方面了解哪些？

11.NIO是否了解?

12.hashMap如何保证线程安全？

13.hive是否了解，SQL语句写的如何？

14.生产者消费者的代码写一下（java方面的生产者消费者案例）

15.spark源码看过哪些？（答：任务提交流程）其它的源码呢？

1.介绍下你在公司的项目，你负责哪块？

2.你擅长哪些技术，项目中遇到过哪些问题？

3.spark-on-yarn任务启动，任务提交，资源分配划个流程图

4.iava代码、sql语句类问题

5.cdh用的什么版本，组件熟悉哪些，怎么升级cdh

6.spark从kafka拉取数据时，offset变量保存在哪里？

7.spark运行的三种模式是什么，你们公司在用那种，client和cluster有什么区别，你们公司用的哪个？

8.详细MR和spark有什么区别？

9.为什么说rdd是弹性分布式数据集？

10.hive数据仓库理解吗，简单说一下数据仓库的搭建

11.kafka拉取的超时数据如何处理的。

12.什么是shuffle，如何处理数据倾斜？

13.硬件资源一定的情况下，如何处理海量数据，保证执行速度?

14.如何判断一个循环链表？

15.你们公司大数据部门多少人，你们和那个部门对接，系统是谁在用

16.数据结构，算法方向了解嘛，简单的说一下。

点众科技

一面:

1 数据平台是什么, spark 版本 , hadoop版本 jdk?

Cdh 5.3.6

2 数据是什么格式,每天数据量,pv量 数据都有什么字段?

Json

Xml

Csv

普通文件

日志log

3 解释下: case class, appay方法?

手写 怎么分组 求topN,用下面的三种模式 ?

Spark hvie mr

4怎么解决hash碰撞?

5怎么求集合的top100 (考虑到内存的问题,这个集合的数据量特别巨大) ?

6 hdfs的读写流程?

7 saprk跟mr的区别?

8 hashmap的底层原理?

9 开窗函数 ?

10 解释下:linklist和arraylist区别?

11 解释下hashmap

12 手写排序算法?

二面:

1主要就是:就是每天数据量,集群规模,pv量

2项目的流程

3 指标怎么实现的?

4 mr怎么清洗的,多长时间清洗一次?

5 app数据量大还是网站数据量大

点众科技

kafka 数据一致性

kafka 保证时序性

hive的优化

hive的开窗函数

hdfs的读写流程

hdfs多磁盘挂载

sqoop架构

cdh的cloudera manager架构功能

项目 集群组成

写了两个项目 嫌我项目少

感觉面试官觉得项目眼熟 不好明说

嘟嘟一下（公司用sql比较多，出去面的话可以第一家去试水）：

1.人力面试：上家公司的情况，薪资，对公司的要求，了解我们公司吗？

2.技术：

1.自我介绍，介绍项目

2.项目中涉及到的维度有哪些？

3.如果实现按照每天每个地域的点击下单支付数量怎么实现？

4.按小时呢？

5.java的比较类 手写，如果想用做mr的输出类还差什么？

6.sql leftjioin和reightjoin的区别？

7.如果有两张表，A uid pv B pid count 用sql实现得到每个用户的购买商品的数量，要求不能有空

8.数据清洗步骤

9.数据从js埋点到服务器步骤

10.hive的优化

11.kafka多少台

12.每秒数据量多大

13.实时方向，sparkstreaming拉取kafka数据多长时间一次？

14.一次是全部拉取吗？

15.如果数据量过大的话，服务器不能承受怎么办？

16.代码优化。

17.spark调优

18.RDD常用算子。

19.map和mappartition的区别，reduby和groupby的区别

20.给一个例子分别说出每个RDD各计算了多少次。

会通教育（面试官不会大数据）

填表

笔试

技术：

1.简单介绍

2.项目流程

3.mysql中有多少张表，表中有什么字段，都存储了哪些值

4.一天的pv uv 数据量多大

5.你们的筛选条件有哪些？有多少个

6.项目周期多少？指标完成用多久

7.你是怎么优化项目的

8.常用的linux查看内存的命令有哪些？

9.你介绍一下kafka？你们公司为什么用这个？

10.介绍一下flume （又给他说了flume和kafka是怎么保证数据的安全性的，flume是如何采取数据的，kafka是怎么存储数据的，消费者是怎么消费数据的）

11.你们为什么用spark，如果我这里有一个文件里面有一行很大的整数，我们不用java也能实现，为什么还要用你说的spark？

12.如果我们服务器最多每次可以运行10个进程，现在有100个任务，你怎么设计让这台服务器快速有效的将任务运行完

13.说说多线程，死锁

14.了解spring框架吗？

15.说说jvm内存里面是怎样的，什么样的数据放在堆内存，什么样的放在栈内存

1.hbase与mysql的区别

2.spark实时作业宕掉，kafka指定的topic数据堆积怎么办？

3.java的设计模式知道么，项目中运用在哪？

4.开发中遇到的最困难的事，你是怎么解决的？

5.集群规模？集群配置？

6.spark作业有多少个？

7.项目中当前batch获取前面的batch的算子是哪个？

8.hive开窗函数写过没有？怎么写的？除了row\_number还知道哪些窗口函数？（就那四个排序窗口函数）他们有啥区别？

9.100个分片，我想聚合成两个分片，用哪个算子？-（当时忘了，但是记得是C开头的一个算子，我就说我不会读那个单词，是c开头的算子）--coalesce

10.spark的作业提交流程

11.公司原来大数据部门规模

12.flatmap和map的区别

13.spark对接hbase

14，dateframe和dataset区别

哗啦啦面试问题总结

1 stage 的划分

mr的流程。hdfs读写流程

spark submint 参数设直 级任务提交流程

produce向kafka中发送数据产生的offset 怎么算（给你传入几条大小的消息 求offset是多少）

项目中遇到的问题 及解决

项目的数据怎么跑的 ，测试过程出现问题怎么查看 （死扣项目）

yarn 的参数 mr参数 hive设置过什么参数

flatMap 算子怎么压平的

reduceBykey 能否替换掉 groupBykey (不能：那就问你 哪些情况不能”)

手写 wordCount scala 分 几个stage

旷视科技面试总结

1,项目的架构（几个项目都会简单问一问的，重点讲你的负责的模块，项目时间的衔接，中间你干嘛的）

flume +hdfs +kafka +hive +mysql

描述的要清晰，最好将为什么选型，为什么用，

项目中一些业务逻辑要说清，会临时穿插一些逻辑扩展，怎么解决？

项目后期迭代中，遇到的实际问题,比如添加某些逻辑指标（可以从原有的基础数据查询处理的），（原本不存在的字段，新逻辑呢？）添加一些字段呢怎么去高效的修改代码进行迭代？参考hadoop项目中的脚本sql语句

2,项目中负责的功能模块的实现,具体到用什么方法实现什么功能产生什么结果,你的离线数据以什么为单位划分的,如果任务宕掉的话,怎样能监控到,你们的任务一般都是什么时候进行运行的,如何解决数据延迟的问题

3,项目中遇到的问题:我当时说了数据倾斜,然后他问我什么情况下什么数据产生了数据倾斜,具体到字段信息，

4,根据需求场景写hive语句

5,wordcount的代码如何转换为sql,请书写一下

session =new SparkSession

read.textfile("dir")

lines.split(" ").withColumnRenamed("value","word").createTempView("W\_C)

session.sql(select word ,count(\*) counts from w\_c goroup by word order by counts desc )

6,MR的reduce和SQL中的group by的比较

7,stage,task和job的区别与划分方式

我们称为Mapreduce程序，一个Mapreduce程序就是一个Job，而一个Job里面可以有一个或多个Task，Task又可以区分为Map Task和Reduce Task.

一个Application和一个SparkContext相关联，每个Application中可以有一个或多个Job，可以并行或者串行运行Job。Spark中的一个Action可以触发一个Job的运行。在Job里面又包含了多个Stage，Stage是以Shuffle进行划分的。在Stage中又包含了多个Task，多个Task构成了Task Set。

Mapreduce中的每个Task分别在自己的进程中运行，当该Task运行完的时候，该进程也就结束了。和Mapreduce不一样的是，Spark中多个Task可以运行在一个进程里面，而且这个进程的生命周期和Application一样，即使没有Job在运行。

8,kafka,flume,hdfs,sqoop,storm,spark系列,MR的所有知识他都会问到,另外他不想听到理论性的知识,希望有实际经验，遇到的问题，踩过的坑，

9,因为他们的项目组尚在发展阶段,所以去了公司,公司项目的架构与spark项目的架构类似,但是需要对一系列组件都需要了解,另外项目实施环境的运行和细节方面的会很多,需要注意.

10，对哪方面比较了解，精通，说说，说说工作中你认为追比较有难度，有意思和挑战性的项目有哪些？

11，有什么要了解的? 你的优缺点，对公司有什么了解?，这个挺重要的，说不清立马会减分，最后的沟通很重要，

12,算法，红黑二叉树优先遍历，递归思想的应用，排序冒泡等，他们会炸你，说你写的不对。别怂就行了，他们在对你做测试。

13，关于数仓，数仓的建设，项目中你都用哪些表？哪些字段呢？

14，平台的数据量，项目组以及公司的人员配备。

先自我介绍一下

说一下项目架构 你具体负责的项目中的那一块

我说的是前期的数据拉取和简单的数据清洗 中期的项目模块开发 后期的一部分维护

然后直接问拉取的工具 flume从哪里拉取 数据什么格式 每天数据量 中间管道是用的哪种类型 最后放到哪里 用的什么格式 为什么用这些类型 不会出问题吗

sqoop熟悉吗(不熟悉) 那当时的命令是怎么写的

kafka这一块你们项目中具体是怎么做的 主要用来做什么 kafka热点问题是怎么解决的 kafka怎么存到Redis的 存的时候是什么类型 kafka你们是怎么配置的 比较关键的参数有哪些 有没有来调整优化

kafka怎么保证不重复消费 设置自动提交和down到Redis/MySQL中每次都来对比数据和offset

kafka你有没有踩过什么坑 或者是对你印象深刻的地方

然后hbase 你有没有参与表的设计 rowkey是怎么设计的 那你们的rowkey是怎么做的 业务的rowkey具体是什么样的 关于region分裂是怎么处理的 用hbase的时候有没有调优

MySQL和Redis了解到什么程度 用连接池吗 连接池怎么实现的 说一下主要步骤

项目开发的过程中你负责哪些模块 聊一下你最熟悉的

还有就是kafka,flume,Redis,hbase和项目对接的时候用的那些api

还有很多很细节的东西 反正就是负责哪一块他就往深处问往细处问

https://blog.csdn.net/zhi\_fu/article/details/77776961

1.空闲时间都在做什么?

完成了工作之后,我就会去看一技术的东西,不断提高和完善自己的技术

2.项目的spark的版本?

1.5.1

3.如何实现分组求topn?

说一步

简单讲一下你项目中写的SparkSQL语句?

17.你项目中最后的Spark调优是怎么做的？

18.你项目中发生数据倾斜了吗？怎么处理的？

19.你的第一步是找到数据倾斜的key，那你这个找是怎么找的？

你们一天数据的吞吐量你有了解么？

10.你们公司业务一般是线上的还是局域网的？

在线上

11.一天数据量有多大？存储在哪？

12 flume在项目中需要配置什么?

13你项目中flume用的源是什么? 就是source的模式是什么?

14 Linux提交命令的话，怎么后台提交？

15 简单介绍一下你的项目流程?

16 这个具体是要实现一个什么样的业务?

17 实时业务要搞清楚

18 像kafka是一个高吞吐的一个存储中间件嘛，像你们业务中每次可以同时处理多少条数据（这个我真不知道，一阵冷场）

19 最后数据存哪里?数据量？

5、公司有其他的大数据小组吗？不同开发小组技术架构如何协调？

6、项目周期，集群规模（spark集群架构图）

实时项目大概是一个什么样的流程?必须中kafka做中间件吗?

你的项目中kafka是一个队列,esb也是一个队列,为什么不用esb做消息队列而要用kafka呢

集群规模有多大?

项目中kafka的吞吐量?

300m/s

==========================kakfa模块=========================

kafka技术点:

11.那每一个topic里的数据，最多能有多少个consumer进行并发消费？（懵逼，10个？）摇头（不记得了）

1 在kafka中zookeeper的作用?

无论是kafka集群，还是producer和consumer都依赖于zookeeper集群保存一些meta信息，来保证系统可用性.

2 项目中为什么要用到kafka?(从解耦,异步,并行这几个角度去说)

3 kakfa是什么?

4 kakfa的核心组件?(就是栋哥那个kafka核心组件图)

5 consumer之间消费的数据各不相同,同一group中数据不能重复消费

解决办法:就是更改group的组名,这个是kafka的一个机制

6 kafka的文件存储机制?

https://www.cnblogs.com/jun1019/p/6256514.html

同一个topic下有多个不同的partion,每个partition为一个目录,partion命名的规则是topic的名称加上一个序列号,序号从0开始

每个parttion目录下的文件被平均分割为大小相等(默认一个文件是500m)的数据文件,每一个数据文件称为一段(segment file),但是每段消息的

数量一定相等,这种特性能够使得老大segement可以快速被清楚,默认识保留七天的数据,

另外每一个partition支持顺序读写,partition中的每一个segment端的生命周期由我们配置文件指定的一个参数决定的,默认情况下

每满500兆就会创建一个新的segment端,默认保存七天

首先00000000000000000000.log文件是最早产生的文件,该文件达到1G后又产生了新的000000000000000665453.log文件,新的文件会

向这个文件里面写,这个文件袋达到1G之后,数据就会向下一个文件中里面写,也就是说它只会往文件末尾追加数据,这就是顺序写的过程

生产者只会对每一个partition做数据的追加操作.

如何保证消息消费的有序性呢?比如说生产者生产了0-100个商品,那么消费者如何按照这个0-100个商品顺序消费呢?

那么kafka如何保证这种有序性呢?难度就在于生产者在生产出1-100条数据会后,通过分组策略,brokeer的partition中的时候

比如0-10条消息在这个partition,10-20在那个partition,这样的话消息在分组的时候,就被我们到的分组策略搞到无序了.那么能够做到

全局有序性呢,? 遇到这种情况我们大多数情况系来说我们做不到全局有序性,只有在某种情况下我们才可以做到全局有序

结论:一个partion中的数是有序的吗?回答:间隔有序,不连续

在一个tipic里面的数据,只能做到partition内部有序,不能做到全局有序,做到全局要有序,只能就是有一个partion

segment file是什么?

partition在磁盘上就是一个目录,该目录名就是topic的名称加上一个序号,在每个partiition下面,有两类文件,一类文件是以log为后缀

一类文件是以index为后缀的文件,每个一个log和index文件是相对应的,这一对文件就是一个segment file 也就是一段

其中log文件就是数据文件,里面存放的是消息,而index文件就是索引文件,记录的了元数据信息

segement 文件命名规则:partition全局的第一个segment从0开始,后序的每个segement文件名就是上一个文件segement文件最后一条消息

的offset值,查找起来快

这样我们就可以用二分法定位文件快所在的位置,具体位置我们我们需要去这个这个数据对应的index目中去查找,在index文件中并不时为每条数据

都建立一个索引,没事采用稀疏方式,每个一定字节的数据建立一条索引,这样就避免索引文件占用了太多的内存空间,从而可以将索引文件

保存到内存中,没有建立索引的message并不能一次就定位到其数据文件的位置,从而只要做一次顺序扫描,就可以知道了,但是这次顺扫描的

范围就很小了

7 kafka的原理?

kafka的一致性和完整性?

afaka消息同步机制?

https://www.cnblogs.com/jun1019/p/6256514.html

7 kafka的sendfile技术,pagecase理解?

怎么解决kafka数据重复消费?

kafka为什么要进行分区?

kafaka怎么保证数据不丢失?

kafka direct是什么? 为什么要用这个,有什么优点? 和receiver.

kafka为什么依赖zookeeper,zookeeper中存储了哪些kafka元数据?

那每一个topic里的数据，最多能有多少个consumer进行并发消费？

========================hadoop模块===================

深入理解yarn,yarn是个什么东西

yarn的提交流程

什么是comtainer?

comtainer 是 yarn中的抽象资源,它封装了某个节点多维度资源,如内存,cup,磁盘,网络等.当am想rm申请资源的时候

RM为AM返回的资源便是用comtainer表示的.yarn会每个任务分配一个container,且该任务只能使用该comtainer中描述的资源,

需要注意点是continner不同于MRv1中的slot.他是一个动态资源划分单位,是根据应用程序的需求动态生成的.目前为止,

yarn仅支持cpu和内存两种资源

一 怎么解决hdfs上的碎片问题?

碎片文件带来的问题:

小文件是指文件size小于HDFS上block大小的文件。这样的文件会给hadoop的扩展性和性能带来严重问题。

首先，在HDFS中，任何block，文件或者目录在内存中均以对象的形式存储，每个对象约占150byte，如果有1000 0000个小文件，

每个文件占用一个block，则namenode大约需要2G空间。如果存储1亿个文件，则namenode需要20G空间（见参考资料[1][4][5]）。

这样namenode内存容量严重制约了集群的扩展。 其次，访问大量小文件速度远远大于访问几个大文件。

HDFS最初是为流式访问大文件开发的，如果访问大量小文件，需要不断的从一个datanode跳到另一个datanode，

严重影响性能。最后，处理大量小文件速度远远小于处理同等大小的大文件的速度。每一个小文件要占用一个slot，

而task启动将耗费大量时间甚至大部分时间都耗费在启动task和释放task上。

Hadoop本身也提供了几个解决方案，分别为：Hadoop Archive，Sequence file和CombineFileInputFormat。

hadoop archive:通过归档的方式来处理,是一个高效地将小文件放入HDFS块中的文件存档工具，

它能够将多个小文件打包成一个HAR文件，这样在减少namenode内存使用的同时，仍然允许对文件进行透明的访问。

对小文件进行存档后，原文件并不会自动被删除，需要用户自己删除,一旦创建archive不可以改变,要增加或者删除文件必须重新归档

sequence file是由一系列的key/value组成,如果为key小文件名，value为文件内容，则可以将大批小文件合并成一个大文件。

二 Mapreduce怎么去重

就是给他说,shuffle阶段会相同的key进行合并,map端读取偏移量对应的数据,value为空,这样我们reducer端的时候我们只要把key输出就行

三 如果给你一百万条数据怎么去重?

四 Mapreducer和spark有什么区别?

五 hdfs上的端口是多少?

namenode web端管理端口:50070 hdfs端口9000

secondary NameNode web管理端口 50090

h

六 yarn上怎么查看spark的log日志

七 持久化是怎么?

八 hdfs上各个组件的作用

namenode:负责维护元数据和客户端的读写数据请求

secondaryname:负责辅助namenode,用于合并镜像文件和日志,较少namendoe这一块的资源消耗

datenode:负责存储数据,每个数据被分割为若干个块,每个块默认有三个副本,启动后定时给namenode发送心跳,和块的报告信息

安全模式:从集群启动到加载

HDFS的安全模式：从集群启动直到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。

九 什么是2pc?具体应用有哪些场景

十 简答介绍下redis,了

十一:linux提交命令的话怎么后台提交?

在控制台提交命令的时候,在后面加上&符号就可以了

十三 yarn上怎么查看spark的日志?

通过spark ui查看,或者用yarn logs-applicationid + spark进程id | 查看状态:yarn application -status + 进程id

十四 怎么把日志追加到一个文件中？ >> 和 > 有什么区别

>> 追加 > 覆盖

十五 写过mr吗? mr如何实现join

分为reducer端算法实现和map端算法实现

reducer端算法的实现:

通过关联的条件作为map输出的key,将两个满足join条件的数据并携带数据来源的文件信息,发往同一个reducetask,在reduce中进行数据的串联

缺点:这种方式,join的操作实在reduce端完成的,reduce端处理压力会太大,map节点运算负载会很低,资源利用不高,且在reducer阶段非常容易

产生数据倾斜

map端join算法实现:

适用于关联表中有小表的情形;

可以将小表发送到所有的map节点,这样,map节点就可以在本地对自己读到大表数据进行join并输出最终结果,可以大大提高join操作的

并发读,加快处理速度

十六 hadoop生态系统系统的构成及每个组件的作用?

1 HDFS是一个分布式文件系统,是hadoop体系中数据管理的基础,可以看做是一个超大的硬盘

2 yarn 分布式资源管理系统,用于统一管理集群中的资源

3 MapReduce分布式计算框架,用map,reduce方法实现分布式程序设计

4 zookeeper:分布式协调服务,用于维护集群配置的一致性,集群管理,状态同步,统一命名,作用,分布式系协调

5 hbase 分布式数据库

6 hvie 数据仓库的管理工具,开发人员可以向使用sql一样使用mr

7 sooq主要用于数据的迁移

8 flume数据采集工具

十七 hadoop中hdfs的读写流程

写入流程:

1 向namenode请求上传文件,namenode检查目标文件是否已经存在

2 namenode给客户端返回是否可以上传

3 client会先对文件进行切分,然后想namenode请求第一个块该上传到哪些datenode服务器上

4 namenode给client返回哪些datanode上传文件

5 client 请求一台datanode上传数据(本质是rpc调用,建立pipline),第一个datenode收到请求后会继续调用第二个datenode

然后第三个datendoe,将整个pipline建立完成,然后逐级返回给客户端

6 client 开始向A上传一个block(先从磁盘读取数据放到一个本地内存缓存),写入的时候datenode会对数据进行校验,然后上传到

第一datendoe,然后第一个收到之后会传给第二台,然后第二台传给第三台

7 当一个block传输完成之后,client再次请求namenode上传第二个blok的服务器.

读取详细步骤:

1 跟namenode通数据,(就是文件block块,所在的节点)

2 挑选一台datenode(就近原i则) 服务器,请求建立socket连接

3 datenode开始发送数据(从磁盘中读取数据放入流中)

4 客户端接收数据,现在本地缓存,然后写入目标文件,后面的block追加在前面的block块,最终合并成为需要的文件

十八 HA脑裂?

========================hive=================================

一 hive的udf怎么写?

首先什么UDF,UDF全称是user-defined-function 用户自定义函数,就是hvie的内置函数无法完成我们的业务逻辑的时候,这个时候我们

就要自定义函数,在hvie中自定义函数必须用java写,hive本身就是用java写的

udf函数作用于单个数据行,并且产生一个数据行作为输出,大多数函数就属于这一类

udf函数必须满足,下面有个条件 ①必须继承udf这个类 ② 这个类中必须实现至少一个evaluate

udaf 用户自定义聚合函数 接收多个数据输入行,并且产生一个数据输入行,就像count,max这样的函数

user/hive/warehouse

二 hvie的元数据库存储了hive的哪些元数据信息

在把元数据存储配合在mysql中的时候,一共有57张表

1 存贮hive版本的元数据表version

2 DBS存储了hvie中所有数据库的基本信息

3 datebase parms 存储了数据库的相关参数

4 hvie表和视图相关的元数据 就是表的字段,视图的权限,视图的属性

5 hvie分区相关的元数据信息

6 hive表相关字段的元数据信息

hive内部表和外部表的区别?

1 创建方式不一样,在创建外部表的时候需要加上external关键字

2 创建内部表的时候会将数据移动到数据仓库指向的路径;但是在创建外部表的时候仅仅记录数据所在的路径,不对数据位置最更改

3 在删除表的时候,内部表会将元数据和数据一块删除;而外部表知识删除元数据,不删除数据,这样外部表更加安全一些,数据组织也更灵活

hvie是如何加载一个动态分区的

1 设置好两个参数,开启动动态分区,模式设置没nostrict

2 创建好分区表

3 查询结果集,集合中有分区字段,动态插入到分区表中,同时要指定分区字段

sql非常重要,每天都要写sql

hvie调优?

https://blog.csdn.net/u013412535/article/details/45439929

https://www.cnblogs.com/muchen/p/5310732.html

数据仓库方面的知识

行转列?

你做过数据仓库吗?

做过

解释下是数据仓库?

数据仓库的定义:数据仓库是一个面向主题的,集成的,相对稳定的,反映历史变化的数据的集合,用户支持企业或组织用于决策处理.

数据仓库为了便于多维分析和多角度展示而将数据按照特定的模式进行存贮而建立起来的关系型数据库,首先是用于决策支持,面向

分析型数据处理,他不同于企业现有操作的型数据库.其次实多个异构数据源进行有效集成,集成后按照主题进行重组,并包含历史数据,

而且数据仓库里面的数据一般不做修改

怎么构建企业数据仓库?

第一步:需求分析

第二步:确定主题

第三步:确定量度

第四步:确定数据粒度

第五步:确定纬度

第六步:创建事实表

数据仓库的基本架构是什么?

数据源 ETl 数据仓库存储与管理, OLAP bi工具

你认为数据集市起到什么作用?

数据集市,数据集市可以理解为是一种"小型的数据仓库",如果说全公司的数据,那么数据集市可以理解为某个部门的数据,是从数据仓库中独立出来的

一部分数据,也可以成为部门数据或者主题数据.

数据集市可以分为两种,一种是独立的数据集市,这类数据集市有自己的数据源和ETL架构;另外一种非独立的数据集市,这种数据集市没有自己的源系统

他的数据来源数据仓库.

数据仓库中是怎么分层的?

ods层:原始数据 切分完成之后就放在了ods层, 然后根据各个业务抽取dw层, 然后抽取单个主题dm层

什么是维度表?

表示对分析主题所属类型的描述.比如"昨天早上张三在京东上花了200元购买了一个皮包".那么购买的为主题进行分析,可以从这段信息中提取三个维度:

时间纬度(昨天早上),地点纬度:(京东),商品纬度(皮包).通常来说维度表信息比较固定,且数据量小

什么是事实表?

表示对分析主题的度量,比如上面那个例子中,200元就是事实信息.事实表包含了与各维度表相关联的外码.并通过join与纬度表关联

事实表的度量通常是数值类型,且记录数会不断增加,表规模迅速增长.

维度建模的三种模式?

1 星形模式:星形式最常用的纬度建模方式,星形模式的纬度建模由一个事实表和一组维度表组成,且有一下特点:

维表和事实表关联,维表之间没有关联;

每个纬度表的主吗为单列,且主吗放置在事实表中,作为两边连接的外码;

以事实表为核心,维表围绕核心呈星形分布

2 雪花模式:雪花模式是对星形模式的扩展,每个纬度表可以继续向外连接多个子维表.星形模式中纬度表相对于雪花模式来说要大,而且不满足规范化设计

雪花模式相当于星形的大维度表拆成小维表,满足了规范化设计,然而这种模式在项目中很少见,因为这种设计会导致开发难度的增大,数据冗余

3 星座模式:也是星形模式的扩展.基于雪花模式才有了星座模式.前面两种纬度建模方法都是多维表对应单事实表,但是很多时候纬度空间的事实表不止一个

且一个维度表可能被多个事实表用到.在业务发展的后期,决大部分纬度建模方式都是星座模式

什么是OLAP?

在线联机分析,在我们数据仓库创建好了,我们就要写sql语句对里面的数据进行分析,但是每次查询都要写一次sql语句,而且对某些纬度数据进行分析处理的sql代码

比较固定,于是便有了olap工具主要用具建模数据的多角度,多维度,多层次分析.

OLAP的三种架构:MOLAP ,ROLAP ,HOLAP

数据立方体:可以理解为纬度扩展后的二维表

数据库和数据仓库有什么区别？

数据组成差别:

1 操作型数据库,只会存放90天以内的数据,而分析型数据库里面存放的是数年内的数据.

2 操作性数据库中存放的是主要是细节数据,而分析型数据库中虽然有细节数据,又有汇总数据,但是对于用户来说,最重要的就是汇总数据

技术别差:

1 操作性数据库允许用户进行增删改查,但是分析型数据库只允许查询

2 操作数据库,操作频率高,数据量少,分析数据库数据量大频率少

ETL:抽取,转换,加载

ETL工作的实质就是从各个数据源抽取数据,对数据进行转换,并最终加载填充数据到数据仓库纬度建模后的表中.只有这些纬度/事实表的被填充好了

.ETL工作才能算完成.

=========================hbase===============================

hbase的组件?

hbase的热点问题?

=============flume/ekl技术栈===============================

https://www.cnblogs.com/zhangyinhua/p/7803486.html#\_lab2\_2\_3

1 flume监控一个目录,动态生产文件时监控文件个数,选择哪个source源

2 flume监控一个目录,动态生成文件拉取文件内容,选择哪个source源?

3 logstash如何应用,如何拉取数据?

4 logstash和flume的区别

logstash 和 flume比较：

组件比较：

logstash： input filter output ====>可以同时读取kafka中多个topic中的数据(input)

flume： source channel sink

特性：

logstash：

安装包体积小，安装简单，配置简单

有filter组件，该组件具有切分字段和数据过滤的功能，可以实现简单的数据清洗(重要优势)

可以与ES无缝结合(也很重要)

可以断点续传(重要)

主要是用于日志数据的采集(中啊哟)

flume：

在高可用方面优于logstash

flume比较注重数据传输的安全性，在数据传输的过程中是由事务控制的

可以用于多类型数据的采集

========================spark模块=============================

spark的原理具体是什么样的?

先介绍spark是干什么的,然后各个组件,组件中的进程,然后就是spark任务生成和任务的提交

什么是rdd,rdd有什么特性?(用自己的话描述出来)?

1.弹性分布式数据集,是spark中最基本的数据抽象,代表不可变,可分区,里面的元素可并行计算数据的集合.

特性:

1.就是rdd有多个分区

2.rdd的每个分区使用同一个函数计算()

3.rdd之间时候依赖关系的,就是rdd的lineage,就是在计算的时候,rdd有分区的数据丢失的时候,可以通过lineage记录元数据和操作记录,

可以重新计算丢失分区的数据,而不是rdd所有分区的数据都从新计算

4.对于key-value类型的rdd可以使用partionner分区器进行分区,默认有两种分区器(hashpartionenr和rangepartioner),也可以自定义分区器.

5.就是在计算的时候按照"移动数据不如移动计算的理念",就是在计算的时候把计算的任务放到对应数据块的地方,这样计算的时候就可以直接从

本地拿取数据,这样就大大减少了网络io流

RDD来源：这使得Spark SQL得以洞察更多的结构信息，从而对藏于DataFrame背后的数据源以及作用于DataFrame之上的变换进行了针对性的优化，

使用程序中的集合来创建RDD

使用本地文件系统创建RDD

从HDFS创建RDD

从数据库（DB ==包括nosql数据库==）创建RDD

从数据流创建RDD

从kafka,flume创建RDD

怎么理解dateframe?

在Spark中，DataFrame是一种以RDD为基础的分布式数据集，类似于传统数据库中的二维表格。

DataFrame与RDD的主要区别在于，前者带有schema元信息，即DataFrame所表示的二维表数据集的每一列都带有名称和类型。

最终达到大幅提升运行时效率的目标。反观RDD，由于无从得知所存数据元素的具体内部结构，

Spark Core只能在stage层面进行简单、通用的流水线优化

(有schema信息,就是记录了数据的结构,清楚的知道数据集包含了哪些列,每列名称和类型,相当于二维表格,

所以可以用sql语句来查询)

dateframe和dateset?(要解释和rdd区别)

都是分布式不可变,弹性数据集,然后里面他们俩的算子类型都是一样的,

dateset是在spark 1.6.0 版本才出现的. dateset里面是强类型,可以再case clsss 中指定类型, 而dateframe是row类型,

(每列数据类型是不定的) dateframe=dateset[row]

他们俩之间是可以互相转换的.

在spark2.0 以后 就合并了,就是没有dateframe,之后dateset.

的dateset.

怎么理解dstream?

一段时间段的数据流,经过tranform处理后的数据流,再抽象一下可以说是一系列的rdd

groupbykey和reducebykey区别?

返回值类型不同：reduceByKey返回的是RDD[(K, V)]，而groupByKey返回的是RDD[(K, Iterable[V])]，

举例来说这两者的区别。比如含有一下数据的rdd应用上面两个方法做求和：（a,1）,(a,2),(a,3),(b,1),(b,2),(c,1);

reduceByKey产生的中间结果(a,6),(b,3),(c,1);

而groupByKey产生的中间结果结果为（(a,1)(a,2)(a,3)）,((b,1)(b,2)),(c,1),（以上结果为一个分区中的中间结果）

可见groupByKey的结果更加消耗资源

作用不同，reduceByKey作用是聚合，异或等，groupByKey作用主要是分组，也可以做聚合（分组之后）

map端中间结果对键对应的值得聚合方式不同

(自己的话描述:首先这个两个算子,都会发生shuffer过程,groupbykey仅仅把key相同的数据放在了一起生成的rdd格式是:RDD[(K, Iterable[V])]

它的作用是分组;reducebykey返回的数据格式是RDD[(k ,v)]主要作用是聚合

reducebykey发生shuffer之前把相同的key进行合并,所以发生shffer时网络io量更少,但是groupbykey算子在shuffle不会合,所以reducebykey的

效率更高,groupbykey更加消耗资源)

map和flatmap区别?

map()可以对里面的RDD循环,操作每个元素,然后返回值构成一个新的rdd

flatmap 对里面的rdd循环,然后压平,返回的是迭代器,然后迭代器里面的所有rdd构成要给新的rdd

val rdd = sc.parallelize(List("coffee panda","happy panda","happiest panda party"))

输入

rdd.map(x=>x).collect

结果

res9: Array[String] = Array(coffee panda, happy panda, happiest panda party)

输入

rdd.flatMap(x=>x.split(" ")).collect

结果:

res8: Array[String] = Array(coffee, panda, happy, panda, happiest, panda, part

spark里UDAF、UDF有啥区别？分别继承哪个类？

UDF:implement UDF3,UDF2 implements UDF3<Long, String, String, String>,主要是看后面参数是个数.

UDAF:继承UserDefinedAggregateFunction

他们都是自定义函数:

udf:函数的输入时一条具体的数据记录,作用的是一条数据.

udaf:是用户自定义的聚合函数,函数本身作用数据集合,能够在聚合操作的基础上进行自定义的操作,一般就是和sql中group by (分组操作连用)

spark中日志查看方式?

spark日志确切的存放路径和部署模式相关

(1) 如果是spark standalone模式,我们可以直接在master ui 界面查看应用程序的日志,在默认情况下这些日志是存储在worker节点的work目录下，

这个目录可以通过SPARK\_WORKER\_DIR参数进行配置. 8085端口

(2) 如果是mosos模式,我们同样可以通过mesos的master的 master ui 界面来查看日志,这些日志是储存在mesos slave的work目录下.

(3)如果是在yarn模式下,最简单是就是使用yarn的 日志收集工具(yarn logs application),这个工具可以收集应用程序相关的运行日志

但是跟这个工具有限制的,应用程序必须运行完,因为yarn必须首先聚合这些日志,而且还必须开始日志聚合功能(yarn log -agregation enable)

如果你运行在YARN模式，你可以在ResourceManager节点的WEB UI页面选择相关的应用程序，在页面点击表格中Tracking UI列的ApplicationMaster，

这时候你可以进入到Spark作业监控的WEB UI界面，

spark和Mapreduce快？ 为什么快呢？ 快在哪里呢？

1.MapReduce每次shuffer的时候结果都要写到磁盘中,而spark的shffer的中间结果不用写到磁盘中,而是可以缓存到内存中.

如果任务很复杂,要经过多次shuffer的才能完成,那么mapreduce在读写磁盘中文件的时候,io流上会消耗很多资源,运行时间大大增加

2.MapReduce在发生shuffer的时候,就是一个完整的MapReduce程序,就需要提交一次job,但是spark中shuffer结果可以放在内存中,

这样就不用发生shuffle就提交job,这个过程中spark也会节省大量的时间.

3.hadoop的操作是基于进程的操作,每次启动一个task变会启动一次jvm,而spak中每次操作是基于线程的,只是在启动executor的时候

启动一次jvm.在spark每执行一次task,都是复用executor中的线程.但是每次启动jvm的时候都会花费很长的时间,如果task数量非常多,

那么基于hadoop,就会在启动jvm的时候浪费大量的时间.

Spark Sql为什么比Hive快呢？

Spark Sql最终内部调用的是Spark RDD，而Hive则是调用的MapReduce

spark工作流程? (就是spark的启动和任务提交流程)

从源码角度来分析,原码要非常熟悉

spark on yarn 的两种模式? clinet模式和cluster模式 (区别)

区别一:driver的程序运行的位置不一样,cluster模式的driver在yarn中运行,而client模式的driver的client端运行.

区别二:cluster模式运行结果不在客户端展示,一般都保存在外部存户介质中,客户端显示的仅仅是job的运行状况,

clinet模式的运行结果会在客户端显示

区别三:cluster模式主要用于生产模式,而client主要用于交互和测试.

简要描述spark分布式HA的搭建步骤?

哪些算子设计到shuffer?

stage怎么划分的?

spark怎么读取hbase吗?

spark on hhase

JavaSparkContext sc = newJavaSparkContext(master, "hbaseTest",

System.getenv("SPARK\_HOME"), System.getenv("JARS"));

Configuration conf = HBaseConfiguration.create()

Scan scan = newScan()

scan.addFamily(Bytes.toBytes("cf"));

scan.addColumn(Bytes.toBytes("cf"), Bytes.toBytes("airName"));

try{

String tableName = "flight\_wap\_order\_log";

conf.set(TableInputFormat.INPUT\_TABLE, tableName);

ClientProtos.Scan proto = ProtobufUtil.toScan(scan);

String ScanToString = Base64.encodeBytes(proto.toByteArray());

conf.set(TableInputFormat.SCAN, ScanToString);

JavaPairRDD<ImmutableBytesWritable, Result> myRDD =

sc.newAPIHadoopRDD(conf, TableInputFormat.class,

ImmutableBytesWritable.class, Result.class)

catch(Exception e) {

e.printStackTrace();

}

这样本段代码段是从Hbase表名为flight\_wap\_order\_log的数据库中读取cf列簇上的airName一列的数据，

这样我们就可以对myRDD进行相应的操作：

同时在pom.xml的文件中加入依赖

spark怎么整合hvie?

Spark On Hive hviecomtext

def main(args: Array[String]): Unit = {

val conf = new SparkConf().setAppName("test").setMaster("local[2]")

val sc = new SparkContext(conf)

val sqlContext = new HiveContext(sc)

sqlContext.table("test.person") // 库名.表名 的格式

.registerTempTable("person") // 注册成临时表

sqlContext.sql(

"""

| select \*

| from person

| limit 10

""".stripMargin).show()

sc.stop()

}

看过spark原码吗?最熟悉那一部分

master work sparksubmit sparkcontext

spark sql怎么处理结构化数据和非结构数据?(就是转换成为dateframe)

结构化数据:就是读取json文件,转化为dateframe,然后注册成为临时表,用sql开始操作.

DataFrame df = sqlContext.read().json("hdfs://192.168.226.129:9000/txt/sparkshell/people.json")

//输出表结构

df.printSchema();

//显示DataFrame的内容。

df.show();

非结构化数据:两种方式

①:通过反射的方式推断schema,转为为dateframe,然后注册成为临时表 case class 导入隐式区域 toDF

②:由structype的类型,来指定schema, sqlContext.createDataFrame(rowRDD, schema)

spark中宽窄依赖的依据什么啊?

宽依赖:就是多个子RDD的分区会依赖同一个父类RDD的分区

窄依赖:就是每个父RDD的分区,最对被子RDD的一个分区使用

怎么理解fearchrdd?

怎么理解Spark是基于内存的?

我是这样理解的有两点:

第一点:Spark支持数据集persist或者cache到内存中,如果将频繁使用的数据集cache到内存中,这样可以大大较少网络io,网络传输

和重新计算的大家,我们都知道shuffer过程中非常耗费资源,如果把shuffer之后数据集进行缓存可以显著的提高运行效率

第二点:基于pipeline的计算策略,可以再内存中实现中间数据重用的最大化,大大减少了中间结果的io操作,在hahoop的MapReduce中在

计算流程中,涉及到大量中间结果存储问题,在计算过程中这些中间结果都是存入到磁盘中的,然后再出入到内存中造成了大量的网络io

而spark是基于内存的就很好的避免了这个问题

像spark处理数据量过大的时候可能会宕机，如何处理？

这个就是spark的oom问题

spark on yarn提交任务的运行流程？

1 客户端提交一个Application,会在客户端启动一个driver进程.

2 driver进程会向RS(resoucemanger),发送请求,启动ApplicationMaster资源

3 re收到请求后,根据nodemanager上划分一个comtainer,启动AM,这里的nodemanager相当于standalone中的work节点

3 A启动后会向RS申请一批omtainer资源,用于启动executor

4 RS调度好一批NM给AM,用于启动executor

5 AM向NM发出命令启动executor

6 executor启动后,会向driver反向注册,driver发送task到executor上执行,然后executor给driver返回执行结果

driver如何发送task到executor？指定位置还是随机配发？

从源码的角度来说,这个应该是随机的

对spark2.0的了解?

1 dateframe和dateset统一,dateframe知识dateset[row]的别名

2 sparksession:统一了SQLcontext和hiveContext,新的上下文入口

3 accumulate功能曾强

4 sql功能增强,支持子查询

同时加入了一些新特性,支持csv文件

同时也移除了一些功能,比如dateframe

spark2.0发布了测试版的struct Streaming

spark的shuffler?

https://blog.csdn.net/zongzhiyuan/article/details/77676662

在spark的原码中,负责shuffer过程的执行,计算和处理的组件主要是shuffermanger,也即shuffer管理器,而随着spark的版本的发展,shufer

shufflemanager也在不断迭代,变的越来越先进了.

在spark1.2以前,默认的shuffle计算引擎是Hashshufflemananger.改shufflemanager有一个非常严重的弊端,就是会产生大量的中间磁盘文件

进而有大量的磁盘io操作影响了性能

因此在spark1.2以后的版本中,默认的shuffleMannager改成了sortshufflemanager,sortshufflemanager相比较hashshufflemanager来说

有了一定的改进,主要在于,每个task在进行shuffle操作时,虽然也会产生较多的临时磁盘文件,但是最后所有的临时文件合并(merge)成一个磁盘

文件,因此每个task就只有一个磁盘文件.在下一个stage的shuffle read task拉去自己的数据时,只有根据索引读取磁盘文件中的部分数据即可

用自己的话说一遍

spark 运行原理,工作机制?

先说出几个组件,每个组件的作用,然后就是spark的任务生成和提交过程

spark中oom问题?

spark中数据倾斜技术点总结?

spark调优?(就是spark性能优化主要有哪些手段)

第一:性能调优

① 在提交spark作业的时候,通过参数的调整,分配能提供分配的更多资源,就是在能够提供资源的前提下,把资源调到最大

② 提高作业的并行度,避免cpu core的浪费

③ 对于要多次计算的和使用的公共的RDD,一定要持久化,cache或者persist

④ 在executor端有用到的数据我们可以通过广播变量直接发送到executor的内存中,要用的时候直接去本地的内存中拿去,减少网络io

⑤ 把map list set 中的数据放入到fastutil中.因为fastutil能够提供更少是内存在占用,同时提供更快到读取速度

第二:JVM调优

① 降低cache操作的内存占比

spark中,堆内内存被划分为三部分,一块是给RDD cache,persist进行数据缓存的 storage.另外一块是给spark算子函数的运行使用的

存放函数总自己创建的对象 execution.默认情况下,有0.6的内存都分给了cache了,但是实际情况中cache内存不是那么紧张,但是task算子

创建的对象过多,然后执行内存又不是很大,这样就会导致频繁的minor gc ,甚至是full gc 导致spark频繁停止工作.性能影响会很大.

调整cache的内存,让task执行计算函数时候,有更多的内存可以使用. 就会减少gc 频率,减少spark停止工作的时间.从而使性能大大提升

② 调节executor堆外内存和连接等待时长

调节executor的堆外内存: 在spark处理的数据量特别大,几亿数据量,然后spark运行,是不是就会报错, task lost ,out of memoury

如果就是spark的堆外内存不够用的时候,就会导致executor在运行的时候内存溢出.如果碰到这样的情况就可以去调节executor的堆外

内存,一般是默认识300M ,但是在通常的项目中,真正处理大数据的时候,都会出现问题,导致spark作业反复崩溃,我们可以调节这个参数

调节连接的等待时长: executor会优先从自己本地关联的BlockManager中获取某份数.如果本地没有,就会远程连接到其余的exector的blockmanager

去获取. 此时如果task中创建的对象特别大,特别多,频繁的的让jvm堆内存满溢,正在进行垃圾回收,而处于垃圾回收过程中,所有的工作线程就会

全部停止,相当于只要一旦进行垃圾回收,spark/executor停止通过,无法提供响应.此时没有响应,无法建立网络连接,会卡主.spark默认提供的连接时长

是60s,如果卡住60秒无法建立连接,那么就会宣告失败. 如果几次都拉去不到数据的话,可能就会导致spark作业的崩溃.所以这个时候我们可以

考虑调节连接的响应时长,这个时间调长之后,就可以避免偶尔连接的时候,拉取文件失败.从而提高spark走也性能.

第三:shuffle调优

spark shuffle过程中是非常耗费资源的,著要的就是磁盘io对性能和spark作业的执行速度,基本上spark作业的性能,都消耗在shuffle中了

主要是从两个方面来调整:

①:开启shuffle map端的文件合并机制 new sparkconf.set("spark.shuffle.consolidateFile","true") 默认情况下是不开启的,就会

发生大量map端输出文件严重影响性能. map task写入磁盘的io较少,同时第二stage去上游拉去文件的时候文件数目也会较少,此时网络

传输性能消耗也会大大减少. 在实际生产过程中,这种调优机制对于性能的提升是相当可观的

②:调节map端内存缓冲与reduce端内存占比.map task内存缓冲变大了,就减少了spill到磁盘文件的次数,而且也减少了后面聚合读取磁盘

文件的数量.reduce内存聚合变大了reduce内存聚合变大了,减少了spill到磁盘文件的次数.

第四:算子调优

①:当数据量不是很大的时候,我们可以用mappartions提升map类型的操作.我们的都知道在普通的map中,如果你的function要执行一万次

那么你的function就要执行一万.在使用mappartion是的时候,一个task仅仅会执行一次function.function一次接受所有partion的数据

只要执行一次就可以了,性能比较高.但是如果数据量特别大的时候,mappartion可能会出现oom问题,但是map是不会的.

②filter之后使用coalesce减少分区数量; 就是在用filter算子之后,我们有的分区的数据就会特别少,如果数据量非常少的分区还是用

一个task来处理,就有点浪费task的资源了. 同时task在处理的时候有的分区数据少有的分区数量多,就会造成task处理的数量不一样,

有的task在处理的时候有些处理快有些处理慢,就会造成数据倾斜. 我们使用colesece来减少分区数,使分区数尽量紧凑一些. 这样在一定程度

上也是提升性能.

③使用foreachpartion 来优化写入数据库性能. 如果在写入数据库的时候用foreach的一条数据就拿一个连接,非常非常消耗性能.用foreachpartion

一个分区拿一个连接,这样我们就可以大大优化这个性能了.

④通过reparation提高作业的并行度.

⑤reducebykey 就是在做类似wordcout程序的时候,用reducebykey对性能提高有帮助.因为会现在本地聚合,这样map端的数据就会变少,减少磁盘

io,下一个stage在拉去数据的时候也就变少了,减少网络的

用scala手写wordcout程序?

========================主要的算法===========================

https://blog.csdn.net/marcosyw/article/details/72794216

https://blog.csdn.net/marcosyw/article/details/72794216

你发 各种排序方式的动态展示

都要手写出来

1.冒泡排序

搞定

2.快速排序

https://blog.csdn.net/morewindows/article/details/6684558

快速排序主要思路是:"挖坑填数+分治法",首先令i=L;j=R;将a[i]挖出形成第一个坑,称a[i]为基准数.然后j--由后向前找比基准数小的数

,然后挖出此数,填入前一个坑a[i]中,再i++由前向后找比基准大的数,找到后也挖出次数填到前一个a[j]中,这样i之前的数据都比基准数小

i之后的数都比基准数大,因此将数组分成二部分再分别重复上述操作步骤完成了排序

public class Unit1\_1<T> {

public static void main(String[] args) {

Integer[] nums = {10,58,72,5,9,7,45,15};//需要排序的数组

nums = sort(nums,0,nums.length-1);

System.out.println(Arrays.toString(nums));

}

public static Integer[] sort(Integer[] nums,Integer left,Integer right){

int i,j,t,temp;

if(left>right){

return nums;

}else{

//获取基准数

temp = nums[left];

i = left;

j = right;

//从右边往左遍历,获取第一个小于基准数的下标

while(nums[j]>=temp && i<j){

j--;

}

//从左往右遍历，获取第一个大于基准数的下标

while(nums[i]<=temp && i<j){

i++;

}

//然后交换两个数在数组中的位置

if(i<j){//判断两数下标有没有相遇

//若没有相遇

//交换两数的位置

t = nums[i];

nums[i] = nums[j];

nums[j] = t;

return sort(nums,left,right);

}else{

nums[i] = temp;

sort(nums, left, i-1);

sort(nums, j+1, right);

return nums;

}

}

}

}

搞定

3.二分查找

搞定

4.二叉树的遍历'

是数据结构中很重要的一部分,存储方式有顺序存储,但是这种存方式会浪费一些内存,有延伸出了链式存储,然后再这个基础上就演变了

二叉树,深度优先遍历.........和广度优先遍历......

https://blog.csdn.net/yimingsilence/article/details/54783208

public void levelTraverse(TreeNode root) {

if (root == null) {

return;

}

LinkedList<TreeNode> queue = new LinkedList<>();

queue.offer(root);

while (!queue.isEmpty()) {

TreeNode node = queue.poll();

System.out.print(node.val+" ");

if (node.left != null) {

queue.offer(node.left);

}

if (node.right != null) {

queue.offer(node.right);

}

}

}

二叉树是一种非常重要的数据结构,很多其他数据结构都是基于二叉树的基础演变而来的,对于二叉树,有深度遍历和广度遍历,深度表里

有前序遍历,中序遍历,和后序遍历,三种遍历方式,广度遍历及是我们平航所说的的层次遍历,因为数的定义本身就是递归定义,因此采用

递归的方法去实现数的三种遍历不仅容易理解而且代码很见解,而对于广度遍历来说,需要其他数据结构的支撑,比如堆了,所以对于一段

代码来说,可读性有时候要比代码本身的 效率重要的多

前度遍历:根节点 左节点 右节点

中序遍历:左子树 根节点 右子树

后序遍历: 左子树 右子树 根节点

层次遍历: 只需按层次遍历即可

5 插入排序

https://blog.csdn.net/hellozhxy/article/details/79911867

=======================java===================================

1.线程池?

https://www.cnblogs.com/aspirant/p/6920418.html

什么是线程池?

就是一个保存了一定数量线程的空间,已经提前申请好了空间和资源,在被销毁之前里面的线程一直等待被调用,里面线程可以反复调动.

减少了创建和销毁线程所消耗的资源.

引入线程池的原因:

由于线程的生命周期中包括,创建-就绪-运行-阻塞-挂机-结束 阶段,当我们处理的任务数目比较小的时候,我们可以自己创建几个线程来处理

处理相应的任务,但是有大量的任务的时候,由于创建和销毁线程都需要很大的开销,运行线程池就可以大大缓解这些内存开销很大问题,

因为减少了创建和销毁线程的次数,每个工作线程都可以被重复利用,可执行多个任务..

为什么要用线程池?

①:减少了创建和销毁线程的次数,每个工作线程都可以被重复利用,可执行多个任务.

②:可以根据系统的承受能力,调整线程池中工作现成的数目,防止因为消耗过多的内存,而把服务器累趴下(每个线程需要大约1mb内存

,线程开的越多,消耗的内存也约大,最后死机 )

常用的线程池有哪些?

单线程线程池: newsingleThreadExecutor

这个线程只有一个线程在工作,也就是相当于单线程串行执行所有任务.如果这个唯一的线程因为异常结束,那么会有一个

新的线程来替代它.此线程池保证所有的任务的执行顺序按照任务的提交顺序执行.

固定大小的线程池:newFixedThreadPool

创建固定大小的线程池,每次提交一个任务就创建一个线程,知道线程达到线程池的最大大小.线程池的大小一旦达到了最大值就会

保持不变.如果这个线程因为执行异常而结束,那么线程池会补充一个新线程.

可缓存的线程池:newacachedThreadpool

创建一个可缓存的线程池,如果线程的大小超过了处理任务所需要的线程,那么就会回收部分空闲(60s不执行任务)的线程,当任务

增加的时候,此线程池又可以智能的添加新线程来处理任务,此线程池不会对线程的池的大小做限制,线程池的大小完全依赖jvm能够提供最大

线程大小

创建一个无线大小的线程数量:

线程池能按时间计划来执行任务,允许用户设定计划任务的时间.参数corepoolsize设定线程池中线程的最小数目.当任务较多的时候

线程池可能会创建更多的工作线程来执行任务.

线程池处理流程:

①首先会判断核心线程池里是否有线程可执行,有空闲线程则创建一个线程来执行任务

②当核心线程池里已经没有线程可执行的时候,此时将任务丢到任务队列去.

③如果任务队列已经满了的话,但运行线程的数量小于最大线程池的数量的时候,此时将会创建一个线程用于执行任务,但如果运行线程

数已经达到了最大线程池的数量的时候,此时将无法创建线程来执行任务.这个时候就会按照策略来处理任务.

进程和线程的区别:

线程是进程的子集,一个进程有多个线程,每条线程并行执行不同的任务.不同的进程使用不同的空间,而一个应用程序中的线程共享一个空间

2.arraylist和linklist区别?

Arraylist和linklsit都是list接口的实现类,Aarraylist底层实现是动态数组,linklist是基于链表结构

对于随机访问的get和set方法,arraylist要优于linklist,因为linklist要移动指针.

对于新增和删除操作add和remove,Linklist比较占优势,因为arraylsit要移动数据.

对于他们俩的使用,我们要看情况而定

3.手写一个单利懒汉模式,饿汉模式?

单件模式用途：

单件模式属于工厂模式的特例，只是它不需要输入参数并且始终返回同一对象的引用。

单件模式能够保证某一类型对象在系统中的唯一性，即某类在系统中只有一个实例。它的用途十分广泛，打个比方，

我们开发了一个简单的留言板，用户的每一次留言都要将留言信息写入到数据库中，

最直观的方法是没次写入都建立一个数据库的链接。这是个简单的方法，在不考虑并发的时候这也是个不错的选择。

但实际上，一个网站是并发的，并且有可能是存在大量并发操作的。如果我们对每次写入都创建一个数据库连接，

那么很容易的系统会出现瓶颈，系统的精力将会很多的放在维护链接上而非直接查询操作上。这显然是不可取的。

如果我们能够保证系统中自始至终只有唯一一个数据库连接对象，显然我们会节省很多内存开销和cpu利用率。

这就是单件模式的用途。

单利模式有两种: 单利模式的意思就是只有一个实例,就是确保类只有一个实例.

一种是饿汉模式,就是在类初始化的时候,创建对象,所以类加载比较慢,但是获取对象的速度快,这种方式线程是安全的,

在程序运行期间就这一个对象.

public class Singleton{

//类加载的时候初始化

private static final Singleton instance = new Singleton;

private Singleton(){}

public static Singleton getInstane(){

return instance;

}

}

另外一种是懒汉模式,懒汉模式是在第一次使用的时候才创建对象,但是如果在多线程环境中要考虑线程安全问题.类加载的时候

不创立实例,因此类加载比较快,单运行时获取对象的速度慢

public class Singleton{

private static Singleton instance = null;

private Singleton(){}

public static Singleton getInstane(){

if(instance == null){

synchronized (Singleton.class){

if (instance == null){

instance = new Singleton;

}

}

}

return instance;

}

}

别忘了,私有构造方法, 别忘了synchronized(Singleton.class){ ......} 别忘了后面的return

4.什么是工厂?

在个项目工程中,需要创建很多对象,工厂就是为了管理对象的创建,减少管理成本,一般工厂会和单利一起使用,为了防止通过其他方式

创建对象,创建对象只能通过一种途径就是工厂

6 字符串具体的值存在哪里?方法区中的常量区

7 collection中有哪些分类?

java集合框架都位于java.util包中

collection接口下面我们常用的主要有list接口和set接口 list实现类有:arrarylsit和linklist set的实现类有:hashset和treeset

比如在LinkList.get(10),它具体是怎么找到元素的。LinkList的主节点下标示多少?

9 代码中实现jdbc的步骤是什么样的? 最终的结果ResultSet是怎么遍历的?

乱序数组找出最大值,写出代码

数据结构熟悉吗？说一下二叉树？

18、双向链表插入一个新的数据，写出代码

简述一下对集合的理解?

主要是解决复杂数据存储问题

可以相对于数组来说,数据是固定大小的,且同一个数据组只能存放一种数据类型.

java集合存储的大小不固定

所有的java集合都位于java util中,java集合中只能存放应用类型的数据,不能存放基本数据.

java集合主要分为三种类型 set list map collection接口 set和list都是继承collection,map没有

set(集合)

set是一个接口主要有两个实现类:

hashset:hashset类按照哈希算法来存储集合中的对象,存储速度比较快,里面没有重复元素,不排序,所以一般在向这个集合中存储对象的

时候我们都要重写hashcode和equils方法,来防止里面有相同的元素.

treeset:treeset类实现了sortset接口,能够对集合中的对象进行排序.

list(列表)

list的特征是其元素以线性的方式存储,集合中可以存放重复的对象.

下面有Arraylist:底层实现原理是可扩容数组,查找和更新比较快,删除和插入的时候比较慢,需要移动元素.

linklist底层实现是采用链表结构插入和删除元素比较快

map没有继承collection接口,里面主要存贮的就是key value键值对

hashset和hashmap有什么区别?

Hashset:

Hashset实现了set接口,它不允许集合中出先重复的元素,第一件事就是在将对象存储在Hashtset集合之前,要确保重写hashcode()方法

和equal()方法.这样才能比较对象的值是否相等,确保集合有没有相同的对象,如果不使用这个两个方法,将使用下面两个方法实现

public boolean add(Object obj)方法在set添加元素是,如果元素值重复时返回"fale",如果添加成功则返回"true"

HashMap:

hashmap实现了map接口,map接口对键值对进行映射.Map中不允许出现重复的键key,map接口有两个基本实现Treemap和hashmap.

treemap保存了对象的排列次序,而hashmap不行.hashmap有空的键值对(key(null)-value(null))hashmap是非线程安全的(非synchronize)

要想实现线程安全,那么需要调用collections类的静态方法synchronizemap()实现

hashmap相对hashset较快,因为他是使用唯一的键获取对象

hashmap调用put向map中添加元素,hashset调用set方法向set中添加元素.

hashmap的原理?

当系统开始初始化HashMap的时候，系统会创建一个长度为capacity的Entry数组。这个数组存储的元素是一个系列元素的索引，

也称为“桶”，当一个元素要增加的时候，会计算他的hashcode，然后再数组中寻找他的位置，比如，他的位置有元素占据了，

那么会在该元素上，扩展出一条索引链，将数据插入到这个索引链上

手写代码 二叉树层次遍历的逆向层次遍历,注意是逆向层次

=========================redis================================

项目中怎么操作redis?

======================面试题===============================

http://www.aboutyun.com/thread-23889-1-1.html 面试题总汇

看你推荐系统中用到了Redis,能简单介绍redis么?

手写代码。用scala写一个wordCount程序?

搜狐一面二面的面试题都要搞懂

https://blog.csdn.net/u013412535/article/details/45439929

三、CEO

1 自我介绍

5、一般你在项目过程中遇到问题怎么解决的？

6、你对自己的行业发展有什么计划？

入职之后先立马熟悉公司的项目,我会自己多思考,多请教.努力提高自己的,让自己很好的完成给我安排的工作.最短的时间,努力工作成为公司的

7、你平时对自己有一些技术上的自我提升（学习）么？

8、你觉得自己有什么优点，有什么缺点

学习能力强,做事认真专注,有责任心.

9、你还有什么想要说的么？还有什么想了解的？

服务器有多台 多少条数据 每条 数据多大 多少个字段?

哪个资源调度管理系统? Flume怎么采集到kafka

hdfs读取

亮点知识:分布式中有几个保持数据一致性的算法,?然后果然接着问我有哪些?然后我回答了 2pc,3pc然后问我2pc是什么

hbase有什么优点? 跟mysql有什么区别?

1 列可以动态增加,并且列为空就不存储数据,节省存储空间

2 hbase自动切分数据,使得数据存储具有水平扩展

3 hbase可以支持高并发读写操作

区别: 1 两个文件基于不同的文件系统,mysql是基于操作系统的,hbase是基于大数据平的hdfs文件系统

2 存储数据量:因为 两者是基于不同的文件系统,mysql是基于操作系统,所以mysql是存放的数据量注定不可能太大的.但是hbase是可以

存放海量的数据数十亿条,甚至是更大

3 mysql的列在创建好了,就固定了,在插入数据时,只能往固定的列中插入数据,而hbase的列插入数据时,可以动态增加

namenode是怎么管理元数据的?

为什么要kafka消息中间件?

为什么要用kafka导数据?

集合的知识

wait sleep?

线程池怎么创建?什么是死锁?线程为什么是不安全的?

kafka spark steaming 从 kafka拿去数据的两种方式? 再搞的仔细点

酒仙网地址

科创十一街 经海五路 亦庄科技园区 住在南五环那边 秀水花园 地铁亦庄线经海路站

工作总结

1、团队协作是项目成功的重要因素。分工合作，协调完成工作内容就像分布式集群工作原理一样，用有限的资源高效的完成任务

2、不断地学习才能够不断的提高自己。技术不断地在更新，而固步自封地停留在原来的位置注定会被淘汰。

3、能够把大数据只是和生活结合起来，会让我们的工作不会那么的乏味。我每次下班回家都避免不了坐电梯和扶梯，

一次我发现这正对应大数据中的批处理和流式处理。生活中这样的细节还有很多，每次的发现，都能给我带来惊喜。而随着慢慢的积累，就越能发现代码之美，也越来热爱自己的工作

https://blog.csdn.net/mrlevo520/article/details/76339075

hvie的优化?

在处理hive优化的时候我们必须把hvie sql当做mapreducer来读,从mr的运行角度来考虑优化性能,从更底层来思考如何优化运算性能,而不应该

仅仅只限于逻辑代码层面

要就是从以下几个方面入手:

1. 解决数据倾斜

2. 减少job数

3. 合理设置map 和 reduce的task数,能有效提升性能

4. 对小文件进行合并从行之有效的提高效率

5. 优化时要把握整体,单个作业优化不如把握整体优化

配置角度优化:

1 Map阶段的优化,主要就是确定合适的map数.通过调整数据的分割单元的大小 mapred.max.split.size, 减小max就可以增加map数

增大max就可以减少map数

2 reduce阶段的优化:

reduce阶段的优化主要就是选择合适的reduce task的数量, 可以从mapred.reduce.tasks参数的设置从而来指定reduce的个数

需要注意的是:reduce的个数对于整个作业的性能有很大影响,如果reduce设置的过大,那么将会产生很多小文件,对namenode会产生一定的

的影响,而整个作业的运行时间也未必会减少;如果reduce设置的过小,那么单个reduce处理的数据将会加大,很可能引起oom问题

3 列剪裁:

就是在hvie读取数据的时候,可以只读取查询中需要用到列,而忽略其他列,这样做可以节省读取开销,中间表存储开销和数据整合开销

裁剪所对应的参数项为:hvie.optimize.cp=true

4 分区剪裁:

可以在查询过程中减少读取不必要的分区

分区裁剪对应的参数为:hive.optimize.pruner=true

join操作

在写join操作查询语句时有一条原则:应该讲条数少的表/子查询放在join操作符的左边,原因是在join操作的reduce阶段,位于join

操作左边表的内容会被加载如内存中,将数目少的表放在左边,可以有效减少发生oom出错的几率. 就是小表放在左边

map join操作

如果有一张表非常非常小,而另外一张关联的表非常大的时候,我们就可以使用map端join, 就是join在map端发生,不需要reduce,也就是

不需要shuffle的过程,从而能在一定程度上节省资源,提高join效率.

gruop by 操作

有数据倾斜的时进行负载均衡

此处需要设定:hive.groupby.skewindata,当设定为tue时,生成的查询计划有两个Mapreduce任务.

1 在第一个MapReduce中,map的输出结果集合会随机分布到reduce中,而每个reduce都会做局部聚合,并输出结果,这个样处理的

结果是,相同groupby key 有可能有分发到不同的reduce中,从而达到负载均衡的目的;

2 第二个 mapreduce 任务在根据预处理的数据按照groupbykey分配到reduce中(这个过程可以保证相同的gruop by key 分布)

到同一个reduce中,最后完成最终的聚合操作

合并小文件:

我们都知道文件数目小,容易在文件存储端造成瓶颈,给hdfs带来压力,影响处理效率,对此,可以合并map和recue的结果

文件来消除这样的影响.

设定参数属性:合并map端文件,合并Reduce端输出文件

无效id在关联时数据倾斜问题:

问题:日志中常会出现信息丢失,比如每日约为20亿的全网日志,日中useid为主键,在日志收集过程中会出现丢失,出现主键为null

的情况,如果其中userid 和 bmw\_user关联,就会碰到数据倾斜问题,原因是,主键为null值的项会被当做相同的key同一reduce中

不同类型关联产生的数据倾斜问题:就是不容类型id关联产生的数据倾斜问题

把数据类型转换为字符串类型

select

\*

from log a

left outer join

r\_id r

on a.id=case(b.id as string)

sqoop架构

sqoop架构是非常简单的,它主要有三部分组成,sqoop client hdfs/hbase hvie database 组成

用户向sqoop 发起一个命令之后,这个命令会转化一个基于map task 的MapReduce作业,maptask 会访问数据的元数据信息, 通过buffer进行数据的

并行的map task 将数据库中的数据读取出来, 然后倒入到 hadoop中. 当然可以将hadooop中的数据导入到关系型数据库中.

他的核心思想就是通过基于maptask 的MapReduce作业,实现数据的并发拷贝和传输,这样可以大大提高效率

https://www.cnblogs.com/yutingliuyl/p/7294935.html

hdfs的读写流程

1 首先调用FileSystem对象的open方法,事实是一个DistributedFileSystem的实例

2 distributedSytem通过rpc获得文件的第一批block的locations,同一个block会返回多个locations,这些locations依照hadoop拓扑

结构排序,距离client近的排在前面.

3 前两步回返回一个FSDataInputStream对象,该对象会被封装DFSInputStrean对象,DFSInputSteam能够方便管理datanode和namenode

数据流,client会调用read方法,DFSInputSteam最会找出离client近的datanode并连接

4 数据从datanode源源不断的流向client

5 假设第一块读取完了,就会关闭指向第一块的datanode连接,接着读取下一块,这些操作对client来说是透明的,client的角度看来

就是仅仅是读取一个持续不断的流

6 假设第一批block都读取完了,dfsInputStream就会去namenode拿下一批block的locations.然后继续读,假设全部读取完,这时就会

关闭掉全部的流.

7 假设在读取数据的时候,dfsinputSteam和datandoe的通讯发生异常,就会尝试正在读的排序第二近的datanode,而且会记录哪个datanode

错误发生,剩余blocks读的时候就会直接跳过该datanode,DFSinputsteam也会检查block数据校验和,假设发现一个坏的block,就会先报告

到namenode节点,然后dfsinputstream在其他的datanode上该block的镜像文件

这种设计模式就是client直接连接到datanode来检索数据而且namenode来负责为每个block提供最优的datanode,namenode仅仅处理block

locations的请求,这些信息都记录到了namenode的内存中,hdfs通过datanode集群能承受大量client的并发访问

58面试

sql 采样

什么是有向无环图DAG 的理解

map的4中遍历方法

介绍一下kafka kafka的API写过吗？

手写mr 的join mr中小表join大表 怎么优化

sql 语句 怎样查询非关联字段

项目中 你们的任务是定时跑吗？ 那如果你的数据需要A表的数据 A表今天没有跑完 你怎么解决 （ps 不能退给测试）

foreach 和map d 区别 foreachRDD 用过吗？

你们数据仓库是怎么建的

介绍项目 项目有多少人 怎么分工

linux

马蜂窝

线程池 了解多少 线程池的参数 多线程了解多少

flume 的channel 用到是那种 （）

hashmap的底层实现 链表 hash冲突

项目中按时间抽取数据的意义 以及实现

flume中向kafka拉取 数据 数据丢失怎么办

我设定consume 中从头开始消费kafka的数据 如果consumer 关掉 在启动时 数据怎么消费

数据一致性怎么保证

1.kafka的consumer消费能力很低的情况下的处理方案

2.kafka的producer具体如何写入一个topic中（可能表大的不对面试官大概问的就是这个单个topic的kafka写入流程ba)

3.mysql优化具体方案

4.DAG原理，为什么sparkRDD用DAG？（感觉应该是让说DAG优点，）

5.RDD算子为什么可以保证血缘关系（应该是跟源码有关系的回答）

面试总结

1. 现场根据需求编程

例如：我想要对用户行为进行分析 统计出下单用户的行为路径 并对统计结果取top5

1. 资源分配

比如你提交了一个需要多个core执行的大任务和多个需要一个core执行的小任务

任务提交的时候先会执行哪个任务

如果你正在执行一个大任务的话会不会把执行完的空闲的core分配给小任务去执行

如果小任务一直占用空间 又有一个大任务需要立即执行怎么办

1. 数据压缩

你的数据有没有进行压缩？使用了什么方法压缩？压缩成什么格式？压缩比例是多少？在哪一步进行的？

1. 数据库问题

在你将数据持久化到数据库以后有没有进行过更新操作？如何进行的更新？如果写脚本的话如何触发更新操作？

1. kafka问题

我看你的项目中使用了kafka，kafka是如何保证高可靠性的？如果发生了数据丢失怎么办？

kafka的副本存在哪里？又是怎么读取你存储的副本的？

在你用Direct模式时offset偏移量你说是手动存在zookeeper集群上的能不能大致写一下代码？

1. 宕机问题

在job执行的过程中你的节点宕机了会发生什么，如何解决？

我有大量的节点，如果没有cdh等管理工具，我的节点发生了宕机如何知道是哪一个节点？

1. executor问题

executor和executor之间是什么关系，你的jar包从解包开始到传到分配给executor的流程说一下 你的jar怎么分配到executor上的？executor如何知道要执行哪一个task的？

1. 数据倾斜问题

发生数据倾斜有神解决方法？

你说的加随机数 怎么加的？在你的task分配完以后又是怎么去掉的？写一下代码

面试总结:

1. 主要考Java，包括集合（底层原理）、线程、拦截器、协议
2. Javaweb:servlet、五大框架、jsp、jQuery、HTML等
3. 对本专业的理解
4. 数据结构，算法，以及你用过哪些算法。以及时间复杂度。
5. 输入两个数n，m，求m个n的开方和。
6. 求一个数组中任意两个数相加=n的所有组合，要求时间复杂度为1。
7. 项目多想想怎么实现的，把项目说细，各个表和字段都要介绍一下尽量把自己会的都说出来。
8. 要随机应变。了解面试人的想法。
9. 要把握好自己的方向。
10. 写MongoDB语句。

1.Spark-Streaming获取kafka数据的两种方式,并简要介绍他们的优缺点?

2.简述MR的shuffle和Spark的shuffle过程?

3.说一说spark-On-Yarn的提交任务的流程?

4.谈一下Hive,HBase的框架...

Hive你了解多少?怎么用的?

HBase rowkey的设计?

5.如何保证从Kafka获取数据不丢失?

6.SQL最好不熟练的最好多看看,每次都会当场出题.

7.画一下你们的项目的框架图.你们的项目为什么这么设计?(尽量提一些自己的想法)

谈一谈你的项目中用到的技术.

1. 你们的项目都做了那些优化?

你用flume怎么保证数据不丢失比如突然挂掉。

怎么处理flume重新启动数据的重复

说说spark

Rdd是怎么保持数据的不丢失的，（血统）

什么是血统

项目中遇到什么算子啦（reduceBykey，Groupby，filter，count）

那你说下reduceBykey是怎么读取数据的

你这个项目中怎么实现实时消费数据呢（窗口）

一分钟的数据量有多大

你怎么保证俩批次数据能对接上呢（就是这次数据能接上次数据）

如果产生小文件你是怎么处理的

说说你这个top10，

你这个逻辑用sql也就是几分钟的事，

哦，你们数据是存到mysql里，你们是存在一张表吗（不是）

那一张表最多存多少数据

你实时数据都存在mysql中，能放下吗

说下HBase框架

你们数据仓库在哪建的

调优会吗（executor core cpu 并行度）

说说为什么这样挑

Zookeeper是怎么回事

你说下他的读写流程吧

你还用过mr是吧，那你说说shuffle吧怎么实现的

Mr中一共几次排序啊

Reduce段做了什么

Mr遇到小文件问题怎么处理的

你说说yarn的流程，

Nodemanager 和appmaster什么关系，打他datanode呢

你们做展示吗，会可视化工具吗？

你们不做自己不做展示吗？直接保存到库里吗？（是的）

你从hbase取出一条数据需要多长时间

我们这需要做一个数据清洗，这个部门就你一个人，任务有点挑战，你能胜任吗

来，你写下这俩个sql语句，十分钟够吗

嗯，你用spark实现下我这个top10（有效代码，能运行）

你给我求下这个树高，我给你把头写好,最简单的（二叉树）

奥维云网：

主要问项目，问spark处理的pu uv ，集群规模，然后是spark streaming和kafka的细节，比如采用什么模式获取数据，

kafka的消费者高阶和低阶API有什么区别，还有就是spark使用了哪些算子，foreach怎么使用的，连接数据库是用foreach里面连接的？

当时忘记了是在项目中写了个连接池，在foreach中获取连接的，然后就是spark的实时处理的批处理间隔怎样设置的，有什么影响，

然后问了下期望薪资后就over了。

爱空间：

主要招一个能搭建集群的，而且懂运维的数据分析师，招人比较急，自我介绍后问了下项目，然后问数据仓库建模流程

以及搭建集群环境能行么，后面问了他只打算招两个人做大数据，7台服务器集群，简单聊了下期望薪资和到岗日期就离开了。

优目科技：

主要是基于spark开发的，两个人同时面试的，主要聊项目的集群规模，以及spark调优，spark处理了哪些需求，

结果有哪些字段，采集数据的有哪些字段，pv uv是多少，七日留存率怎么算的，留存率是多少，spark开发中遇到了哪些问题，

jvm怎么调优，说说多线程，加锁有几种方法（只回答了sysc那个关键字，记不起来还有lock），kafka的原理，kafka对接spark，

flume的负载均衡和高可用以及数据重复丢失问题（两个人轮流问了一小时，有点懵逼了，高可用忘记了），然后问了下期望薪资就over了。

百度总部--新成立的硬件生态渠道事业部：

首先自我介绍，然后介绍项目，就项目的一个指标怎么实现的，topN怎样实现。然后介绍下hadoop和spark，说说spark的RDD，说说spark shuffle

和hadoop的shuffle，spark的常用算子（主要说了foreach，reducebykey groupbykey，map等）在项目中遇到过什么问题（回答oom和数据倾斜），

为什么会oom？介绍下jvm，熟悉java么？说说hashmap底层结构吧，hashmap如果达到扩容阈值了，有两个同时访问hashmap会怎样

（这个我回答如果是多线程会用currenthashmap，然后他绕回去了，就问如果用hashmap扩容会怎样）？然后介绍kafka，

说说数据从客户端到kafka的流程（当时没想出来，就简单回答了数据通过socket套接字传递的，然后问了我是什么协议，回答tcp协议）

最后给一个数组，里面12个数字，写代码求出这个数组的递增序列最大个数集， 3 2 5 7 8 4 10 9 6这种，2 5 7 8这四个是递增个数最大的，

这里本来想着说说思路，就是比较每个数字，左边的比右边的数字小，然后存到一个集合里面，最后比较集合个数最大的，其实这题考了几个方面，

一是代码书写，二是对集合的运用，三是算法的熟悉度，你用什么算法来查找排序。思路说完了然后问我有什么想问的，我就问了下项目的集群，

他说还没开始搭建，目前是招人，然后问了下主要招聘什么人员，他说主要招聘数据分析师，也就是数据仓库方向，这个坑从他让你自己介绍项目

的时候就埋下了，你自己介绍的数据仓库内容多，证明你有这方面经验，他不会具体问你会什么，而是让你介绍一些某个方面，有没有经验就判断出来了。

然后我问了他一个问题，觉得我还有哪些地方需要提升的，他就说我代码能力还需要提升，因为我只说了思路，没有写完整代码。。。。。。

百卓网络：

主要是spark开发，项目介绍一下，然后问spark的算子使用了哪些，spark streaming和kafka的数据对接，spark shuffle过程，介绍下DAG，

什么情况下发生shuffle，spark调优，jvm怎样调优（可以使用jdk自带的jstat或者jmap工具，在linux中使用他的命令可以显示出堆内存信息），

数据倾斜的问题。说说nginx的作用(负载均衡和反向代理)，说说flume，flume的极限是多大（就是event的数量，，当时懵逼了，就说了一般设置几十万左右，没测试过）

还问了ELK（熟悉ES和logstash）。感觉挺扯淡的，中间问了调优，我说oom，他问我O什么，我说是内存溢出。。。，接着就说了说jvm内存模型。然后说说多线程，

聊了下生产者消费者。然后问了下期望薪资，又去找他老大，回来说老大不在，今天就到这里吧（哈哈 都是这样的套路）

第一家 随支付

为什么离职

遇到了什么瓶颈

具体是那些方面的瓶颈

hbase的优化

spark的核心原理

跟我客气的问问

第二次

你知道反射吗，他是从那个地方到哪个地方还有反射的原理和应用

还有fianl是从哪到哪，我就讲了final+变量+方法+类 她不满意这个回答，他还是问从哪到哪

你画一画架构图，你擅长哪一块（结果不问这一块，我引上来也被打断了）

问了map，我说map是个接口，介绍了hashmap的原理和hash碰撞

问我hashmap默认是存多少个，你了解rehash机制吗

还有redis，他问我几种数据类型

问redis的string的底层原理

然后问kafka，自由回答，尽可能把会的都说了

问了个flume怎么写

问js埋点具体细节

人事问你薪资能谈吗，千万不要回答我觉得我能为公司创造更大的价值

第三家 斗米

进去人事先来了句你们公司离职的有点多啊，我见到好几个了

一共三面，

一面基本是我在说，问了spark，介绍源码

二面，先问了一个进程和线程

问了linux的命令，java进程除了jps还有那些命令，还有vi编辑器怎么返回，其他的我不会，没听懂记不住了，linux不是很熟悉的就别写了

开始次日到7日用hive，mr，spark怎么写，又问你这个又更简单的

三面，这个感觉很瞧不起人

上来就给了一个业务，就是微博互相关组，5亿条数据，求互相关组和关注C的用户关注的其他人，说我的方法太麻烦，而且效率不高

我问他如果是您的话怎么做，鸟都不鸟我，瞪我一眼接着玩手机

快乐茄

1.源数据格式是什么样？有多少个字段?有做压缩吗？一天的数据量清洗前是多大？清洗后是多大？hdfs上的源数据文件一个是多大？

2.一个Executor给分配多少内存？

3.mr中map端task的数量是多少？

4.spark任务的提交流程

5.spark的shuffle过程

6.hive中有没有遇到数据倾斜？是怎么解决的？

7.spark开发中遇到过哪些问题？

金山云

1.给一个单项链表 判断它是不是“回文”（A->B->C->B->A 这种正反都一样的叫回文）要求时间复杂度是O（n） 空间复杂度是O（1）

2.while(true){

Thread.sleep(1);

}

单核的cpu执行这个程序，不考虑其他干涉的理想情况下 CPU占用率是多少

3.了解图吗？

4.说一下红黑二叉树；写一个普通的二叉树；二叉树的先序、中序、后序遍历

5.知道B树和B+树吗

6.知道哪些设计模式

7.写一个死锁

8.说一下broadcast的实现方式

hdfs 的读写操作具体原理流程

oozie的组件关系 workflow coordinator bundle

hive表的字段名

如何调试oom

oom在哪个文件查看，具体的报错信息是什么

内存溢出 内存泄漏

spark on yarn的提交流程

一般都用yarn的8088端口看哪些信息

提交参数

hdfs如何查看分区分块信息

spark的shuffle原理流程

hive hbase的数据迁移

task partition block数量跟什么有关

linux常用指令

杉数科技：（笔试+面试两小时）

笔试题：

1、数据量未知的文件，如何实现对其进行排序。

2、两个文件，50亿URL，每条50K左右(记不清了)，内存4G，找出两个文件中共有的点击次数前五的URL。

3、给出文件格式如下：

2018,12,23,10

2018,12,24,-10

2012,12,1,30

2012,12,15,20

要求输出文件格式为(需要实现二次排序)

2012-12 30,20

2018-12 10,-10

面试题：

1、简单介绍一下项目

2、你们项目有哪些模块

3、你们spark项目是使用spark-on-yarn还是standalone方式

4、你们提交运行的脚本如何编写的（我答有人写shell脚本，不建议之后这样答）

5、如果提交的任务出现bug，而提交的人不是你，你们怎么进行沟通协调

6、简单说明一下MR的shuffle过程

7、给定两张表，手写MR进行join操作

表一：NAME，SEX 表二：NAME，AGE

8、还是上面的表，手写Spark进行join操作，要求不使用join算子，spark的shuffle和MR的shuffle有什么区别

9、上面你在进行spark的join操作时用到了map算子，我可不可以用foreach算子替代

10、action类算子和tranform类算子哪一种更容易引起OOM，Spark算子调优问题

11、手写链表的增删改查

12、单例设计模式简述，并说明适用情景

13、你在开发Spark的时候调用的tuple能解释下么，或者说我们为什么用tuple不用map，这两者有什么区别

14、python了解多少

15、有没有集群的搭建经验（实机的），有没有租用云平台搭建的经验

16、你对我们公司有多少了解，为何选择我们公司，有什么未来的规划

1、介绍一下你的项目

2、你简历上写你有参与项目架构设计，你能说一下架构情况么（钊哥钥匙）

3、你这边采集的数据怎么分别去发送到实时和离线的？

4、实时这边具体想要达到一个什么样的预期？

5、离线这个数据要做怎样的一个分析？

6、如何分析的能讲一下具体的过程么？

7、你这边离线数据存hive，mysql，hbase，这个hbase的一个rowkey设计是怎么考虑的？

8、有用过kudu么？（自然是没有）

9、我有一个文件的用户数据，里面有sessionid，时间，session信息，而且非常大的量级，内存很小，你要怎么做去取出他其中某个时间段内的sessionid，做一个这个时间段内的点击topN？

10、如果用多线程的方法，你这种方法会更加效率么？

11、你对你希望就职的公司有一个什么样的个人意见？

12、你为什么觉得公司的发展目标对于你个人的发展是一个航标的性质？

13、你对我们公司有什么想要了解的？

1、作一下自我介绍吧

2、你这个项目用到了哪些技术，可以介绍一下么？

3、你们这个有做实时的部分么？

4、你们实时是用的什么？

5、kafka到spark streaming怎么保证数据完整性，怎么保证数据不重复消费？

6、除了做记录点还有别的方法么？

7、你对kudu有了解么？

8、hive表你们用来存什么数据？

9、这些数据具体是存在hive的什么结构里？

10、hbase用的多么？

11、hbase里的热点问题遇到过么？怎么解决的？

12、spark sql用的多吗？什么业务场景用的？

13、用的什么api操作？（api是啥，我不记得了）

14、emm，就是你的api怎么去调用他hive的数据库数据的？

15、这样问吧，你使用map和flatmap的时候有什么区别？

16、lmpala有用过么？

17、你感觉你项目里最满意的是哪一方面？

18、最不满意的方面呢？

19、你感觉你的优势在哪？

20、你有什么想要了解么？

周一：

vivo：

1.hbase建表的原则

2.kafka怎么保证数据消费一次且仅消费一次

3.hive优化（结合hdfs存储）

4.历史数据怎么处理

5.redis数据类型

6.flume有用过吗

7.hbase里的hlog的作用、

8.kafka保证数据一致性和完整性

9.项目中遇到的技术难题

可能大公司都会问一些底层原理，然后项目中遇到的难题

其他问题记不住，两个人面我，他问一个他问一个的

面试之前还有点击紧张，看到两个面试官不紧张了，虽然

技术还可以，但是没气场，就和他们聊。很多原理性东西

回答的不好，所以技术面之后就结束了

超盟金服：

小公司，基本没问什么技术性问题，主要就是聊项目，互相聊看法

1.mr的shuffle

2.sparkRDD的算子

聊完就等人力过来，但是人力过来的有点慢，我们俩又开始聊想法，聊公司大数据相关的

然后人力聊（主要背景和薪资，我说的23k）

没坐上地铁就给我说，给我23k能否接受，我说可以，她问我

确定能来吗？答曰：很看好贵公司，那个总监面试官我也很喜欢他的管理方式，但是现在还

不能100%确定，然后晚上就给邮件offer

周二：

跨越速运：

1.hive抽取第2-10个数据（开窗函数）

2.hive断点续传

3.怎么把一条数据拆分为两条

4.Hbase热点问题

5.Hbase二级排序

公司在城市的物流中心，很偏，吗，满满的乡土气息

又是两个没气场，技术一般的小伙子来面试我

我们聊完一些技术，就开就某些问题进行讨论（因为聊了一会，感觉我应该往一些方向去引导了，就根据他们的

一些问题进行一个深入探讨和争论，因为感受到他们技术一般）

然后人力，聊要了25k，说等通知，两天内给回复

随手科技：

技术大牛来面试，问的很深，很会问，然后就很多不会的，聊完就结束了，公司还不错

一般大数据部门怎么样，就看面试你的技术人员，技术怎么样就行了

1.Hbase组件及其作用

1.Client

包含访问HBase的接口，并维护cache来加快对HBase的访问，比如region的位置信息

2.Master

为Region server分配region

负责Region server的负载均衡

发现失效的Region server并重新分配其上的region

管理用户对table的增删改查操作

3.Region Server

Regionserver维护region，处理对这些region的IO请求

Regionserver负责切分在运行过程中变得过大的region

4.Zookeeper作用

1.通过选举，保证任何时候，集群中只有一个master，Master与RegionServers 启动时会向ZooKeeper注册

2.存贮所有Region的寻址入口

3.实时监控Region server的上线和下线信息。并实时通知给Master

4.存储HBase的schema和table元数据

5.默认情况下，HBase 管理ZooKeeper 实例，比如， 启动或者停止ZooKeeper

6.Zookeeper的引入使得Master不再是单点故障

2.yarn资源调度流程

1.client发出请求到resourceManager

2.resourceManager启动Appication master为任务分配相关的资源，分配完成后返回信息到resourceManager

3.再由resourceManager将任务分发到不同的NodeManager。

4.NodeManager启动Container执行需要的MapReduce工作

3.kafka存储机制

1）kafka以topic来进行消息管理，每个topic包含多个partition，每个partition对应一个逻辑log，有多个segment组成。

2）每个segment中存储多条消息（见下图），消息id由其逻辑位置决定，即从消息id可直接定位到消息的存储位置，避

免id到位置的额外映射。

3）每个part在内存中对应一个index，记录每个segment中的第一条消息偏移。

4）发布者发到某个topic的消息会被均匀的分布到多个partition上（或根据用户指定的路由规则进行分布），broker收到

发布消息往对应partition的最后一个segment上添加该消息，当某个segment上的消息条数达到配置值或消息发布时间超

过阈值时，segment上的消息会被flush到磁盘，只有flush到磁盘上的消息订阅者才能订阅到，segment达到一定的大小后

将不会再往该segment写数据，broker会创建新的segment。

4.zookeeper什么地方用到他

5.linux常用操作10个，（简单的就算了）（一些日志的查看，会用到的）

6.mapPartition和map执行流程上有什么区别

1）map的输入变换函数是应用于RDD中每个元素，而mapPartitions的输入函数是应用于每个分区。

假设一个rdd有10个元素，分成3个分区。如果使用map方法，map中的输入函数会被调用10次；而使用mapPartitions

方法的话，其输入函数会只会被调用3次，每个分区调用1次。

2）从输入函数（myfuncPerElement、myfuncPerPartition）层面来看，map是推模式，数据被推到myfuncPerElement

中；mapPartitons是拉模式，myfuncPerPartition通过迭代子从分区中拉数据

3）大数据集情况下的资源初始化开销和批处理处理，如果在myfuncPerPartition和myfuncPerElement中都要初始化一个

耗时的资源，然后使用，比如数据库连接。在上面的例子中，myfuncPerPartition只需初始化3个资源（3个分区每个1次），

而myfuncPerElement要初始化10次（10个元素每个1次），显然在大数据集情况下（数据集中元素个数远大于分区数），

mapPartitons的开销要小很多，也便于进行批处理操作。

7.foreach和foreachPartition区别

1）foreach是直接在每个partition中直接对iterator执行foreach操作,而传入的function只是在foreach内部使用,

2）foreachPartition是在每个partition中把iterator给传入的function,让function自己对iterator进行处理（可以避免内存溢出）

8.map和foreach有什么异同

相同点：

1）都是循环遍历数组中的每一项

2）forEach和map方法里每次执行匿名函数都支持3个参数，参数分别是item（当前每一项）、index（索引值）、arr（原数组）

3）匿名函数中的this都是指向window

4）只能遍历数组

区别：

1）map方法返回一个新的数组，数组中的元素为原始数组调用函数处理后的值。

我的理解就是：原数组进行处理之后对应的一个新的数组。

map()方法按照原始数组元素顺序依次处理元素。

注意：map()方法不会对空数组进行检测。

map()方法不会改变原始数组。

2）forEach()方法用于调用数组的每个元素，将元素传给回调函数。

注意：forEach对于空数组是不会调用回调函数的。

9.leader和follower是干嘛呢

（1）只有1个leader线程，可以有若干的follower线程；

（2）线程有3种状态：leading/processing/following；

（3）有一把锁，抢到的就是leading；

（4）事件来到时，leading线程会对其进行处理，从而转化为processing状态；

（5）处理完成后，尝试抢锁，抢到则又变为leading，否则变为followering；

（6）followering不干事，就是抢锁，力图成为leading；

10.kafka的组件

晚上：（电话面试）

网心科技：

1.kafka为什么要进行分区，怎么分区

2.Spark on yarn有分为两种模式yarn-cluster和yarn-client主要区别

3.简单介绍一下sparkRDD

4.HIVE分区和分桶

5.hive为什么不支持很多文件夹

6.小文件夹怎么操作（他说是在运行的时候，map端应该怎么设置之类）

7.小文件夹怎么合并

8.cores给30个只用了10个，该怎么解决

9.yarn的资源调度

10.发现数据处理比较慢，该怎么解决

11.数据清洗，同一批数据，形成7，8个报表，怎么分解任务

12.flume熟悉吗？

13.存入HBASE或者mysql的依据是什么

14.几种排序算法，时间复杂度

周三：

公司：岩心科技

1.上来一套笔试题：没拍照，因为面试官做我对面（临走把试卷给收走了）

Hbase Rowkey设计原则

kafka的exactly-once

hashmap与hashtable与？？（没记住）的区别

jvm相关的

其他记不住了

因为我说java不熟，就把前三个做了一下

2.yarn的资源调度原理

3.spark shuffle和MR shuffle区别

4.聊项目

5.最后和人力聊

6.要了25k，说两天后给回复

公司：

网心科技：

突然联系我，说昨天技术面过了，什么时候来当面聊聊，我说只能下周，这周包括晚上已经全部排满，

结果发现我就在这个公司附近，就面试完上一家就不行去了，从11：30开始面试，面试到12：20

主要聊一些项目

1.项目中遇到的难题

2.昨天电话面试的没回答出来的问题

然后聊着聊着都感觉比较好，中间我会主动查一些话，比如公司spark用到的多还是MR

由于人力去吃饭了，我们俩又聊到1：00，就聊公司的一些情况，和我在前公司的情况，

然后一些想法的交流，人力聊完就1：40左右了，当场暗示说会给offer

公司：

星聚网络：（今天比较幸运，三家公司在两公里以内）

小公司，上来问我java，我说我只有清洗才会用java，所以java很多原理性的不懂

然后他不懂大数据，就象征性的问了几个没深度的问题，然后我就把话接过来了（此人可能是java开发的，不太会聊天）

说了我做了哪些，项目流程，

然后他就给我说，我们公司有一个大数据离职了，给我说公司大数据项目流程，问我愿不愿意接受挑战，

我就问了一下，公司每条数据多大，本来我想提示一下，结果他抢答说几百kb（我就。。），我问公司数据量多大，他说

几个G，然后我说，我曾经做过实验，就是几个G的数据可以用txt编辑器打开（这个一定要注意方式）

然后他就说如果你愿意接受挑战，我就叫我们领导过来，我就说了一下我职业规划，就婉拒了

然后结束之后，网心科技人力打电话过来，聊薪资，聊是否愿意来公司，然后六点的时候又给了一个电话，就把待遇说

了一下，问能否接受，然后就给邮件offer了

亦庄国际

1、为什么从上海来北京（这有两层含义，1为什么跳槽？2.为什么来北京？针对1当时说的是想学更多的技术，面完发现，这样说显得我的工作范围太局限，建议大家别这么说）

想换个方向发展，数仓或者AI

2、你们公司数据量多少？

100G左右，节假日翻倍

3、平时主用什么开发语言？Python有接触过吗？

主用Java，没用过Python，不过需要的话我可以学得很快，当时刚接触Spark的时候，为了看懂源码，我学了一周的Scala，感觉还不行

4、HDFS的备份有几个？

默认3个，一般不会去改变它

5、数据是定时跑的吗？多久跑一次？一次跑多久？

6、平时工作都做些什么？

做数据清洗，需求实现比较多，集群搭建我也可以，平时用来测试数据的虚拟机集群都是自己搭的，测试的数据通常都是模拟的，或者是小批量的

7、清洗过程中都遇到了哪些问题？

8、MR发生热点问题怎么办？为什么会发生热点问题？

9、每条数据都有哪些字段？

10、说说Storm是怎么工作的？

11、学过基因工程吗？（估计是以为我学过这个才找了个老大来二面。。。。）

12、Mahout里都用到了哪些jar包，哪些代码库？（AI做的，老大摇头）

二分查找的时间复杂度是多少？

logn

13、会搭集群吗？（会，参与过你们公司集群搭建吗？没有，摇头）

14、js埋点是怎么采集数据的？（前端做的，那就是没做过？摇头）

15、数据采集是你做的吗？（不是，但会，那还是没做过呀？摇头）

16、我看你的简历跟培训班出来的有点像（没有没有）

总结：路上来回4小时，面谈半小时，大概就记得这些了，感觉他们公司很注重实践，理论的东西不问,比如我想跟他说推荐系统原理，他直接问我用到了哪些jar包，

再有就是还在大数据初期，想找一个从采集到数据入库都实际做过 的人。第一次面试，两个词，心累，身累

信产集团（有利网）

1、公司数据量？数据大小？内存？多少节点？（每家必问）

2、为什么来北京（还是两层？针对为什么跳槽，这次我说的是想换个方向发展，数仓，或者AI，其实我知道他们公司是想招数仓）

3、说说最近的项目？

讲了推荐系统

4、用UDF做了什么？

做了维度的解析

5、Spark与Hadoop的区别？

6、MR的优化？

7、MR过程中遇到过哪些问题？

8、说说Redis？为什么要用到Redis?

9、给个场景，一组乱序数求top5,不用冒泡（其实就是想让你用快排再排序取前5）

10、冒泡的时间复杂度

n^2

说说项目中是怎么求TOPN的（说了开窗函数，然后给了个具体需求，求某一天的热门商品TOPN）

11、100G数据运行，一次要跑多久？（因为我的确不知道要跑多久，所以就说了个大概，说都是凌晨以后跑，一次几个小时吧，他说你没关注过它跑多久？你都不关注它跑多久，有没有出现什么问题，后期对它进行维护吗？？？。。。然后我很懵逼）

12、做数仓项目的流程（后知后觉他是想问数仓分层，当时说流程的时候提到了HBase，会说到HBase是我把Hadoop项目跟数仓的流程记混了。。。然后他就问Hbase去了）

13、说说HBase?

讲一下HBase的框架，读写流程，热点问题，RowKey的设计

14、什么时候用到了HBase?

15、为什么不直接在HDFS上进行查询，而是要用到HBase?为什么在HBase上查询效率要高？（这是最后一个问题，我回答的不太好）

最后他说他就问到这儿了，问我有没有想问的（我就问了下公司的数据量，如果入职之后的工作职责，他说他们现在主做数仓，也不太清楚每天有多大数据量，如果入职的话主要做数仓的数据分析，数据源主要是MySQL里的数据，基本不用清洗，只建模就行，然后我说没有了，他就说今天先到这儿，后期有再有面试或别的会有HR通知我，然后就凉凉了）

暂时记得这么多了。

赢销通：

写SQL

易企秀：

1、自我介绍

2、没毕业就可以进1号店吗

3、讲述一个完整的BI项目流程

4、为什么要清洗数据，不清洗行吗？

5、数据量多大，多久运行一次，怎么运行的，了解你们公司的其他运行数据方法吗？

6、运行数据时用什么监控数据？

7、清洗后的数据一直存在HDFS吗？

8、存储一年数据量是多大？

9、解释一下map和flatmap

10、参与过一个完整的BI项目吗？

11、平时工作中负责哪一块？

12、你跟谁对接工作？怎么交接？处理完数据给谁？怎么给？你的上级是谁？

13、每条日志的数据格式不都应该是一样的，只需要按照一套固定的方式去处理就可以啦，为什么还要每次都自己去清洗解析？？

14、你们的数据压缩了吗，源数据多大，压缩完多大？

酷开：

1、自我介绍

2、说说最近的项目，讲了几句打断我，问我hive是怎么调优的（因为我第一个写的是数仓项目）

3、问了我一个实际场景，写了一个表，有很多字段，非让我用RDD算子找出一个字段对应的最大时间，没做出来（他们想招一个数仓的往Spark转）

4、问我我认为最复杂的RDD算子有哪些（鄙人不知从何答起）

5、问了广播变量和累加器

6、问了我Spark的运行流程。

7、回到项目，让我讲了数仓的各种模型，以及架构

8、问我是怎么把数据从HDFS放到HBase里的？（他给了一个坑，问我是用Sqoop吗）

9、数据维度发生了变化怎么办

10、数据是怎么给前端的

1. 说一下MapReduce的shuffle过程?（）

2. Spark的通信机制？

3. Spark的任务提交过程？

4. RDD的属性？

5. Spark SQL的原理？

6. 能否说一下MR的join？

7. 你自定义的UDAF函数都实现了哪些功能？

8. scala的高阶函数？

9. 为什么说scala是函数式编程？

10. 说一下scala的下划线有哪些作用？

11. Scala的隐式转换了解么？

12. 说一下RPC框架？

13. Spark是怎么优化的？

14. Hive的开窗函数会么？（这里是出一道题，大概就是利用开窗函数取topN）

15. 说一下Combiner有什么作用？

16. sortByKey这个算子是全局排序吗？

17. 如何用Spark的算子实现取topN操作?

1 spark中项目离线可以用hive ，而且前面有etl清洗 为什么选用spark 进行分析，技你们术选型根据什么定的。

2 你们这个项目中用什么进行调度 ，怎么保证数据不出错

3 离线任务是定时跑吗？时间力度怎样？如果数据没有及时过来 怎么跑？

4 如果你前面的数据出现问题，然后数据为空，你需要跑历史数据，遇到过吗 ？怎么解决

5一些脏数据怎么处理 ，怎么脏数据过滤、出去，用工具还是？

6 实时和离线数据都放入MySQL中 ，能行？

7 你们数据怎么跑 ，你测试数据和生产数据差很多，怎么保证你的代码不出问题

8 spark 项目中 是行为数据多还是业务数据多，在服务器集群上拉去的数据大概多少张

9 spark 项目 你拉去的这张表是张大表是吗? 涉及别的表吗？

10 前端展示那你们怎么展示 ，用工具还是?

11你参数怎么设置的，你提交jar包时，不提前和测试的商量分配多少资源吗（spark）

12 hive 项目，分几个主题 （spark项目中这个问题也有问到过）

13 sparkstreaming 拉去实时数据怎么保证数据不丢失 ，你们代码中是怎么设置的，api的问题

14 怎么将数据存入到MySQL库中

15 看了技术框架后，问你没想从技术选型上进行优化过吗（spark）？

16 项目中涉及到的字段，两个项目都有问道

小度互娱

1.为啥最近面试的都是session项目

2.设计一个ETL

3.各个阶段需要注意的问题

4.flume如何采集数据，Flume和kafka怎么连接，Flume和kafka是不是部署在一个集群

5.flume怎么对数据进行监控，所在节点挂了怎么处理

6.处理后的数据怎么保证正确性

7.两个文件求交集代码

8.写sql语句

9.裂变代码（群里之前有，注意用递归）

10.关于ETL问的很细，记不清了

11.数据倾斜，Spark调优啥的

12.shuffle解决

13.MR对数据怎么清洗

小度互娱

1.将以下hive表

name course score

aa English 75

bb math 85

aa math 90

使用sql输出以下结果

name English math

aa 75 90

bb 0 85

2.hadoop生态体系的构成以及每个组件的作用

3.hadoop中hdfs读写过程

4.spark工作流程,提交任务流程

5.数据仓库与hive的联系

6.java书写用户裂变代码

要求:一个师父有多个徒弟,多个徒弟又有多个徒弟,以此类推,已知徒弟和师父的id,要求给一个师父的ID得出徒弟的所有信息(包括输出所有徒弟个数,徒弟有几层,每一层有几个徒弟)

7.RANK、DENSE\_RANK以及ROW\_NUMBER的区别

8.多表join如何处理?

9.hive以及mysql表数据的查询

10.map与reduce阶段的流程

11.各个调优实现的环境,方法,以及为什么要以这种方法调优

12.不借助任何平台将两个文件合并为一个新文件

13.两个单词文件求交集MR代码

14.分组求topn用mysql实现的sql代码

15.项目中页面转化率的spark代码

16.二分查找代码

Kafka读取一条数据的最大阀值是多少

项目上线了吗？都配置过哪些参数

topn、页面单跳转化率都是简单业务，很快就能做完，还做过什么复杂一点的业务

写过哪些udf，继承什么类，用到哪些函数

自己设计架构什么思路

Hbase底层是怎么实现的

spark什么情况下不会发生shuffle？

分区一致

怎么一致？

groupbykey一下

还有吗？

Kafka消费过的数据存到什么地方

项目上线之后都出过哪些问题，怎么解决的

rowkey设计用散列，前面随机数的规则是什么？？？？

(1)hadoop、hbase集群安装

(2)spark项目数据落地到哪，spark做过什么优化么

(3)Kafka节点和partition的关系

(4)除了Kafka还了解其他哪些消息队列（他们公司用的好像是ActiveMQ）

(5)熟悉哪些数据库

(6)hive了解过么，做过什么优化

(7)数据清洗怎么做的，为什么要做

1.map与mapPartitions的区别

2.foreach和foreachPartition的区别

3.groupByKey,reduceByKey和combineByKey的区别

4.各组件的版本

5.flume获取数据是用什么方式

6.spark sql中缓存方式有哪几种

registerTempTable是action类型的么,发生不发生缓存

7.sql怎么一条数据某个字段有多个值拆分成多条字段各有一个值,

udf,udaf,udtf区别,

除了udtf,用自带的

8.spark on yarn的两种模式的区别

9.用sparkcore,不用sparksql怎么写入数据库

10.主要是你写的项目中的问题问的比较细

数据量,集群数,分配的资源数

hive中存储为什么格式,有几张hive表,有什么字段

压没压缩,spark session的accumulator为什么要用字符串

task信息怎么从前台传到spark的

点击执行怎么能知道查询哪张hive表

kafka用的什么模式,区别

........

项目中的问的特别细..

11.shell问题