



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 76 道

hadoop 框架怎么来优化

可以从很多方面来进行：比如 hdfs 怎么优化，mapreduce 程序怎么优化，yarn 的 job 调度怎么优化，hbase 优化，hive 优化。

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

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

集群管理：zookeeper + hmaster（职责） + hregionserver（职责）

我们在开发分布式计算 job 的时候，是否可以去掉 reduce 阶段

关注公众号：磊哥聊编程，回复：面试题，获取最新版面试题¹



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

可以，例如我们的集群就是为了存储文件而设计的，不涉及到数据的计算，就可以将 mapReduce 都省掉。

比如，流量运营项目中的行为轨迹增强功能部分

怎么样才能实现去掉 reduce 阶段

去掉之后就不排序了，不进行 shuffle 操作了

hadoop 中常用的数据压缩算法

Lzo

Gzip

Default

Snappy

如果要对数据进行压缩，最好是将原始数据转为 SequenceFile 或者 Parquet File (spark)

mapreduce 的调度模式 (题意模糊，可以理解为 yarn 的调度模式，也可以理解为 mr 的内部工作流程)

appmaster 作为调度主管，管理 maptask 和 reducetask

关注公众号：磊哥聊编程，回复：面试题，获取最新版面试题²



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

Appmaster 负责启动、监控 maptask 和 reducetask

Maptask 处理完成之后，appmaster 会监控到，然后将其输出结果通知给 reducetask，然后 reducetask 从 map 端拉取文件，然后处理；

当 reduce 阶段全部完成之后，appmaster 还要向 resourcemanager 注销自己

hive 底层与数据库交互原理

Hive 的查询功能是由 hdfs + mapreduce 结合起来实现的

Hive 与 MySQL 的关系：只是借用 MySQL 来存储 hive 中的表的元数据信息，称为 metastore

hbase 过滤器实现原则

可以说一下过滤器的父类（比较过滤器，专用过滤器）

过滤器有什么用途：

增强 hbase 查询数据的功能

减少服务端返回给客户端的数据量

reduce 之后数据的输出量有多大（结合具体场景，比如 pi）

Sca 阶段的增强日志（1.5T—2T）

过滤性质的 mr 程序，输出比输入少

关注公众号：磊哥聊编程，回复³：面试题，获取最新版面试题³



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

解析性质的 mr 程序，输出比输入多（找共同朋友）

现场出问题测试 mapreduce 掌握情况和 hive 的 ql 语言掌握

情况

datanode 在什么情况下不会备份数据

在客户端上传文件时指定文件副本数量为 1

combine 出现在哪个过程

shuffle 过程中

具体来说，是在 maptask 输出的数据从内存溢出到磁盘，可能会调多次

Combiner 使用时候要特别谨慎，不能影响最后的逻辑结果

hdfs 的体系结构

集群架构：

namenode datanode secondarynamenode

(active namenode ,standby namenode)journalnode zkfc

内部工作机制：

关注公众号：磊哥聊编程，回复：面试题，获取最新版面试题⁴



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

数据是分布式存储的

对外提供一个统一的目录结构

对外提供一个具体的响应者 (namenode)

数据的 block 机制, 副本机制

Namenode 和 datanode 的工作职责和机制

读写数据流程

flush 的过程

flush 是在内存的基础上进行的, 首先写入文件的时候, 会先将文件写到内存中, 当内存写满的时候, 一次性的将文件全部都写到硬盘中去保存, 并清空缓存中的文件,

什么是队列

是一种调度策略, 机制是先进先出

List 与 set 的区别

List 和 Set 都是接口。他们各自有自己的实现类, 有无顺序的实现类, 也有有顺序的实现类。

最大的不同就是 List 是可以重复的。而 Set 是不能重复的。

关注公众号：磊哥聊编程，回复：[面试题](#)，获取最新版面试题⁵



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

List 适合经常追加数据，插入，删除数据。但随即取数效率比较低。

Set 适合经常地随即储存，插入，删除。但是在遍历时效率比较低。

数据的三范式

第一范式 (1NF) 无重复的列

第二范式 (2NF) 属性完全依赖于主键 [消除部分子函数依赖]

第三范式 (3NF) 属性不依赖于其它非主属性 [消除传递依赖]

三个 datanode 中当有一个 datanode 出现错误时会怎样?

Namenode 会通过心跳机制感知到 datanode 下线

会将这个 datanode 上的 block 块在集群中重新复制一份，恢复文件的副本数量

会引发运维团队快速响应，派出同事对下线 datanode 进行检测和修复，然后重新上线

sqoop 在导入数据到 MySQL 中，如何不重复导入数据，如果存在数据问题，sqoop 如何处理?

FAILED java.util.NoSuchElementException

此错误的原因因为 sqoop 解析文件的字段与 MySQL 数据库的表的字段对应不上造成的。因此需要在执行的时候给 sqoop 增加参数，告诉 sqoop 文件的分隔符，使它能够在正确的解析文件字段。

关注公众号：磊哥聊编程，回复：⁶面试题，获取最新版面试题⁶



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 默认的分隔符为' \001'

描述一下 hadoop 中，有哪些地方使用到了缓存机制，作用分别是什么？

Shuffle 中

Hbase----客户端/regionserver

MapReduce 优化经验

- 1、 设置合理的 map 和 reduce 的个数。合理设置 blocksize
- 2、 避免出现数据倾斜
- 3、 combine 函数
- 4、 对数据进行压缩
- 5、 小文件处理优化：事先合并成大文件，combineTextInputFormat，在 hdfs 上用 mapreduce 将小文件合并成 SequenceFile 大文件（key:文件名，value:文件内容）
- 6、 参数优化

关注公众号：磊哥聊编程，回复：面试题，获取最新版面试题⁷



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

请列举出曾经修改过的/etc/下面的文件，并说明修改要解决什么问题？

/etc/profile 这个文件，主要是用来配置环境变量。让 hadoop 命令可以在任意目录下执行。

/etc/sudoers

/etc/hosts

/etc/sysconfig/network

/etc/inittab

请描述一下开发过程中如何对上面的程序进行性能分析，对性能分析进行优化的过程。

现有 1 亿个整数均匀分布，如果要得到前 1K 个最大的数，求最优的算法。

参见《海量数据算法面试大全》

95.mapreduce 的大致流程

主要分为八个步骤

1/对文件进行切片规划

2/启动相应数量的 maptask 进程

关注公众号：磊哥聊编程，回复：面试题，获取最新版面试题⁸



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

3/调用 FileInputFormat 中的 RecordReader，读一行数据并封装为 k1v1

4/调用自定义的 map 函数，并将 k1v1 传给 map

5/收集 map 的输出，进行分区和排序

6/reduce task 任务启动，并从 map 端拉取数据

7/reduce task 调用自定义的 reduce 函数进行处理

8/调用 outputformat 的 recordwriter 将结果数据输出

用 mapreduce 实现 sql 语 select count (x) from a group by b;

搭建 hadoop 集群，master 和 slaves 都运行哪些服务

master 主要是运行我们的主节点，slaves 主要是运行我们的从节点。

pig , latin , hive 语法有什么不同

描述 Hbase, ZooKeeper 搭建过程

hadoop 运行原理

hadoop 的主要核心是由两部分组成，HDFS 和 mapreduce，首先 HDFS 的原理就是分布式的文件存储系统，将一个大的文件，分割成多个小的文件，进行存储在多台服务器上。

关注公众号：磊哥聊编程，回复：面试题，获取最新版面试题⁹



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 的原理就是使用 JobTracker 和 TaskTracker 来进行作业的执行。Map 就是将任务展开，reduce 是汇总处理后的结果。

mapreduce 的原理

mapreduce 的原理就是将一个 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 maste 指派的任务。

HDFS 存储机制

HDFS 主要是一个分布式的文件存储系统，由 namenode 来接收用户的操作请求，然后根据文件大小，以及定义的 block 块的大小，将大的文件切分成多个 block 块来进行保存

举一个例子说明 mapreduce 是怎么运行的。

Wordcount

如何确认 hadoop 集群的健康状况

有完善的集群监控体系 (ganglia, nagios)

```
Hdfs dfsadmin -report
```

```
Hdfs haadmin -getServiceState nn1
```

关注公众号：磊哥聊编程，回复：¹⁰面试题，获取最新版面试题¹⁰



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 作业，不让 reduce 输出，用什么代替 reduce 的功能。

hive 如何调优

hive 最终都会转化为 mapreduce 的 job 来运行，要想 hive 调优，实际上就是 mapreduce 调优，可以有下面几个方面的调优。解决收据倾斜问题，减少 job 数量，设置合理的 map 和 reduce 个数，对小文件进行合并，优化时把握整体，单个 task 最优不如整体最优。按照一定规则分区。

HBase 写数据的原理是什么？

hive 能像关系型数据库那样建多个库吗？

当然能了。

106.### 宕机分为 HMaster 宕机和 HRegioner 宕机，如果是 HRegioner 宕机，HMaster 会将其所管理的 region 重新分布到其他活动的 RegionServer 上，由于数据和日志都持久在 HDFS 中，该操作不会导致数据丢失。所以数据的一致性和安全性是有保障的。

如果是 HMaster 宕机，HMaster 没有单点问题，HBase 中可以启动多个 HMaster，通过 Zookeeper 的 Master Election 机制保证总有一个 Master 运行。即 ZooKeeper 会保证总会有一个 HMaster 在对外提供服务。

假设公司要建一个数据中心，你会如何处理？

关注公众号：磊哥聊编程，回复：¹¹面试题，获取最新版面试题 ¹¹



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

先进行需求调查分析

设计功能划分

架构设计

吞吐量的估算

采用的技术类型

软硬件选型

成本效益的分析

项目管理

扩展性

安全性，稳定性

hadoop 的 namenode 宕机怎么解决

宕机后 client 无法方文，内存中的元数据丢失，硬盘中的元数据还在，如果节点坏了，重启即可，如果机器挂了，重启机器看节点是否能重启，如果不能修复找到原因修复。最好布局成 hadoop ha 高可用，防止出现此问题

为什么会出现 hadoop



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

采用多 nn 做成联邦，nn 是独立的，nn 之间不需要相互调用。NN 是联合的，同属于一个联邦，作为 DN 的元数据公共存储

收集日志的模型

分析业务，确定数据源，埋点收集日志，收集 nginx 日志，存储大 hdfs，清洗数据，放入建模好的 hive 中，查询分析，结果导出，数据可视化

storm

storm 特点

编程模型简单，可扩展，高可用性，高容错性，支持多种编程语言，支持本地模式，高效，亚秒级

吞吐量低

nimbus: 主控节点，用于提交任务，分配集群任务并架空集群运行状态等，可分配多个 nimbus，做高可用

Zookeeper: 协调集群，公共数据的存放（如心跳数据，集群的状态和配置信息），nimbus 将分配给 supervisor 的任务写入到 Zookeeper

supervisor: 负责接收 nimbus 分配的任务，管理属于自己的 worker 进程

worker: 运行具体处理组件逻辑的进程，worker 中每一个 spout/bolt 的线程为一个 task

spout: 是接收外部数据的组件，将外部数据源转化为 Storm 内部数据结构，以 Tuple 为基本的传输单元发给 bolt.

tuple: 是 storm 内部数据传输的基本单元，里面封装了一个 List 对象，用来保存数据



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

storm 进程参数

worker：一个进程

executor：worker 启动的线程

task：实际执行数据处理的最小工作单元

并行度主要取决于 Kafka 中 topic 的数据量，分析 topic 每个 partition 的每秒数据量， $\text{partition 数据量} = \text{spouttask 数据量} \times \text{spouttask 数量} = \text{partition 数量} \times \text{worker 数量}$

storm 怎么保障消息不丢失

nimbus 告诉 Zookeeper 调度接收数据，spout 接收数据 ack 响应，其他节点进程在 spout 消费拉去数据，每个 tuple 发送个 bolt 进行处理，如果成功处理则发送 ack 消息给 zookeeper，发送消息 tuple 消费失败则标记为 fail，Zookeeper 根据偏移量重新发送数据直到消费为止

Kafka

什么是 Kafka?

开源消息系统，由 scala 写成

Kafka 是一个分布式消息队列：生产者、消费者的功能，他提供类似于 JMS 的特性，但是设计上完全不同，此外它并不是 JMS 规范的实现

Kafka 对消息保存 根据 Topic 进行归类，Producer 生产者发送消息，接收消息为 Consumer 消费者，Kafka 集群有多个 Kafka 实例组成，每个实例为 broker Kafka 所有组件都依赖于 Zookeeper 集群保存一些元数据信息，来保证系统可用性



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

Kafka 与传统消息队列的区别

RabbitMQ 吞吐量稍差 Kafka，支持对消息可靠的传递，支持事务，不支持批量的操作，存储于内存或者磁盘

Kafka 遵从一般的 MQ 结构，producer，broker，consumer，以 consumer 为中心，消费的消费信息保存在客户端 consumer 上，consumer 根据消费的点，从 broker 批量 pull 数据；无消息确认机制

Kafka 具有搞得吞吐量，内部采用消息的批量处理，数据的存储和获取是本地磁盘顺序批量操作，消息处理的效率高

Kafka 的 broker 支持主备模式

Kafka 负载均衡 Zookeeper 方向

Kafka 采用 Zookeeper 进行管理，可以注册 topic 到 Zookeeper 上，通过 zoo 的协调机制，生产者保存对应 topic 的 broker 消息，可以随机或者轮询发送到 broker 上；并且生产者可以基于予以定义指定分片，消息发送到 broker 的某分片上

Kafka 各组件介绍

producer：生产者，push 数据到 Kafka 集群

topic：一类消息的高级抽象，一类消息的集合，每个 topic 被分为多个 partition

partition：分区的概念，存放在多个不同的服务器上，实现数据的横向扩展

broker：Kafka 服务器，一个 broker 就代表一个服务器的节点

repliaction：副本，所有的分区可以指定存放几个副本，做到数据冗余，保证数据安全

segment：每个分区由多个 segment 组成，segment 由 .log 文件，一个 .index 文件。



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、s 存储多条信息，消息 id 由其逻辑位置决定，即通过 id 可以直接查询到消息的存储位置，避免 id 的额外映射

2、当 segment 消息条数到达配置值或消息发布时机超过阈值，segment 会被 flush 到磁盘

3、默认值 1G，超过 1G 将节点将重新建立 segment

.log:存放我们的日志文件，所有的数据，最后都以日志文件的形式存放到 Kafka 集群中

.index:索引文件，存放所有。log 的索引，便于我们查询

如何知道消费者消费到哪一条消息了？

Kafka 本地文件

Zookeeper 当中的节点数据记录

offset 偏移量，就是记录我们消费到那一条记录

offset

offset 是每天消息的偏移量

每个日志文件都有一个 offset 来唯一标记一条信息，由 8 个自己数字表示，表示此消息在分区中所处的起始位置

每个分区再物理存储层面，由多个 logfile 组成 (segment)

最小的 offset 表示 segment 中起始消息的 offset

Kafka 怎么保证消息不丢失机制



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

producer: 分为同步模式与异步模式，同步模式效率低，异步模式效率高

Kafka 的 ack 机制，在 Kafka 发送数据的时候，每次发送消息都会有一个确认反馈机制，确保消息正常能够被收到

同步模式：ack 机制能够保数据的不丢失，不建议设置为 0

```
producer.type=sync
```

```
request.required.acks=1
```

异步模式：通过 buffer 来进行控制数据的发送，时间阈值与消息数量阈值，如果 buffer 满了数据未发送，如果设置立即清理模式，风险很大，一定设置为阻塞模式

```
producer.type=sync
```

```
request.required.acks=1
```

```
queue.buffering.max.ms=5000
```

```
queue.buffering.max.messages=10000
```

```
queue.enqueue.timeout.ms=-1
```

```
batch.num.messages=200
```

怎么保证 Kafka 集群的负载均衡?

一个 partition 只能被一个消费者消费，

一个消费者可以消费多个 partition，如果设置分区数量小于消费者数量，则会导致消费者空闲，所以设置参数 partition 数量一定要大于 consumer 数量

partition 数量一定要大于 broker 数量（节点数量），这样 leader 的分区就会均匀分布在各个 broker 上，实现负载的均衡



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

Kafka 命令使用

创建 topic:

```
./Kafka-topic.sh --create --partition 3 --replication-factor 2 --topic test  
--zookeeper node01:2181,node2:2181,node3:2181
```

查看所有的 topic

```
./Kafka-topic.sh --list --zookeeper node01 xx 省略
```

Kafka 的消息发送

```
./Kafka-console-producer.sh --broker-list  
node01:2181,node2:2181,node3:2181
```

Kafka 的消费

```
./Kafka-console-consumer.sh --bootstrap-server  
node01:2181,node2:2181,node3:2181 --from-beginning --topic test
```

使用 zk 来连接集群

```
./Kafka-console-consumer.sh --Zookeeper  
node01:2181,node2:2181,node3:2181 --from-beginning --topic test
```



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

KafkaUtils.createDstream 和

KafkaUtils.createDirectstream 区别

使用一个 receiver 接收器接收数据，接收到的数据将会保存到 executor 中，然后通过 sparkStreaming 启动 job 来处理数据，默认不会丢失，可启动 WAL 日志，保存到 hdfs 上

spark.streaming.receiver.writeAheadLog.enable=true 同时开启
StorageLevel.MeMORY_AND_DISK_SER_2

KafkaUtils.createDirectstream 方式，他定期从 Kafka 的分区中查询偏移量，再根据偏移量范围在每个 batch 里面处理数据

优点：简化并行 高效 恰好一次被消费

hbase

数据库 OLAP OLTP 的介绍和比较

OLTP 联机事务处理，传统关系型数据的主要应用，用于高可用的在线系统，强调数据库内存效率，强调并发操作

OLAP 联机分析处理，数据仓库的主要应用，支持复杂的分析操作。强调数据分析，强调 sql 执行市场，强调磁盘 IO，强调分区
在 OLAP 系统中，常使用分区技术，并行技术

介绍一下 hbase



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

hbase 典型的 key/value 系统，建立在 hdfs 之上，提供高可靠性，高性能，列存储，可伸缩，实时读写 nosql 的数据库系统。

主要用于海量结构化和半结构化数据存储

hbase 查询数据功能很简单，不支持复杂操作，不支持复杂的事务

hbase 主要依靠横向扩展

hbase 中表的特点

大：上亿行，上百万列

无模式：每行都有一个可排序的主键和任意多的列，列可以根据需要动态增加，

同一张表中不同的行可以有截然不同的列

面向列的存储和权限控制，列族独立索引

对于为 null 的列，并不占用存储空间，因此表可以设置的非常稀疏

数据多版本：可以有多个版本，系统自动分配，时间戳为版本号

数据类型单一：只有字节数组 byte[]

129.hbase 表逻辑结构

表有 行 和 列组成，列划分为若干个列族 cloumn

表明 test

rowkey 列族 1: base_info 列族 2: xxx_info

0001 name:zhangsan age:20 address:bj

0002 name:lishi age:50 sex:male address:shanghai hobbies:sing

rowkey:hbase 的行索引，按照 rowkey 字典顺序排序



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

cell 可以锁定一个值：Rowkey+列族+列族下的列的名字+值+时间戳

如何访问 hbase 中的行

- 1、通过单个 rowkey 访问
- 2、通过 rowkey 的范围
- 3、全表扫描

存储特点

存储时，按照 rowkey 的字典序排列存储，设计 key 时，要充分排序存储这个特性，将经常一起读取的行存储放到一起

hbase 和 hive 的区别

- 1、hive 是为了简化 mr 程序而生的，

hive 是基于 hadoop 的一个数据仓库工具，可以将结构化工具映射成一张表数据库，并提供 sql 查询功能，可以将 sql 语句转化为 mr 程序执行运行

hive 适用于离线查询，因为查询时间较长

Hbase 是为查询而生的

hbase 是 hadoop 的数据库，是一个非关系型数据库，非常适合用来进行大数据的实时查询



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

谈谈数据倾斜，如何发生的，并给出相应的解决办法

mr 程序执行时，reduce 阶段大部分执行完毕，但是有一个或几个 reduce 任务运行很慢，导致整个程序的处理时间很长；有的 reduce 数据多，有的数据特别少

数据预处理

combiner 聚合

大量值 key 相同，需要将这部分数据打散 key+随机数，平均分配数据到 reduce

解释下 hbase 实时查询原理

实时查询，可疑人物是从内存中查询，一般响应为 1 秒内。

Hbase 机制是数据先写入到内存中，当数据达到一定量，再写入磁盘中，在内存中不进行数据的更新和合并操作，值增加数据，使得用户的写操作值进入内存中可以立即返回，保证了 Hbase 的高性能

描述 Hbase 的 rowkey 的设计原则

- 1、rowkey 长度原则：设计为定长，如果 rowkey 过长，内存使用率降低，会降低检索效率
- 2、rowkey 散列原则：将 rowkey 的高位作为散列字段，有程序随机生成，这样将提高数据均衡分布在每个 RegionServer 上，以实现负载均衡。

如果没有散列字段，所有数据就会集中在一个 RegionServer 上，数据检索时负载会集中在个别 RegionServer 上，造成热点问题，降低效率



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

3、 rowkey 唯一原则：必须在设计上保证唯一性，rowkey 是按照字段顺序排序存储的，设计 rowkey 时，充分利用排序这个特点，将经常读取的数据存在一块，可能被访问的数据放在一起

列族怎么创建比较好 <=2

有序规则创建 rowkey，Hbase 一张表最好只创建 1-2 个列族比较好，hbase 不能很好的处理多个列族

Hbase 宕机如何处理

宕机分为 Hmaster 宕机和 HregionServer 宕机，如果 HR 宕机，HM 会将其所管理的 region 重新分布到其他活动的 regionserver 上。因为数据和日志都存在 hdfs 中，该操作不会导致数据丢失。

如果 HM 宕机，HM 没有单点问题，HM 可以启动多个 HM，通过 Zook 的管理机制保证只有一个 Master 在运行。

hbase 如何调优

- 1、 垃圾回收调优
- 2、 优化 region 拆分合并以及拆分 region
- 3、 客户端入库调优
- 4、 Hbase 配置文件



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

谈谈 Zookeeper 理解

Zookeeper 作为一个分布式的服务框架，主要用来解决分布式集群中应用系统的一致性问题(解决单点故障问题)。

Zookeeper 并不是用来专门存储数据的，它的作用主要是用来维护和监控你存储的数据的状态变化，通过监控这些数据状态的变化，从而达到基于数据的集群管理

总结: Zookeeper=文件系统+通知机制

spark

什么是 spark

基于内存计算发数据分析引擎，提高在大数据环境下数处理的实时性，spark 仅涉及数据计算

RDD 是什么

弹性分布式数据集，是 spark 中最基本的数据抽象，可以存于内存中或者磁盘中，分布式存储可用于分布式计算

一个不可变，可分区，里面的数据可并行计算的集合

为什么会产生 RDD



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

什么是窄依赖，宽依赖

窄依赖指的是每一个父 RDD 的 partition 最多被子 RDD 的一个 Partition 使用
一对一

宽依赖指的是多个子 RDD 的 partition 会依赖于同一个父 RDD 的 partition
多对一

说一下 RDD 的 Lineage 血统

RDD 的 lineage 会记录 RDD 的元数据信息和转换行为，当该 RDD 的部分分区数据丢失时，他可以根据这些信息来重新运算和恢复丢失数据

RDD 缓存

```
rdd.cache  
rdd.persist
```

什么是 DAG

叫有向无环图，原始 RDD 通过转换形成 DAG，RDD 之间的依赖关系的不同划分为不同的 Stage 调度阶段

那些 RDD 需要 cache

会被重复使用的 RDD，但是不能太大



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

怎么设置 RDD cache

rdd.cache

checkpoint

由于 spark 生产环境面临的 RDD 非常多，计算特别耗费时间

为了防止 RDD 丢失

为了节省更多计算时间

spark 调优

- 1、避免创建重复 RDD
- 2、尽可能复用同一个 RDD
- 3、对多次使用的 RDD 进行持久化
- 4、避免使用 shuffle 算子
- 5、使用 map-side 预聚合 shuffle 操作
- 6、使用高性能的算子
- 7、广播大变量
- 8、使用 Kryo 序列化
- 9、优化数据结构



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

如何解决 spark 数据倾斜

- 1、使用 ETL 预处理
- 2、过滤少数导致倾斜的 key
- 3、提高 shuffle 操作并行度
- 4、两阶段聚合 阶段聚合 和 全局聚合