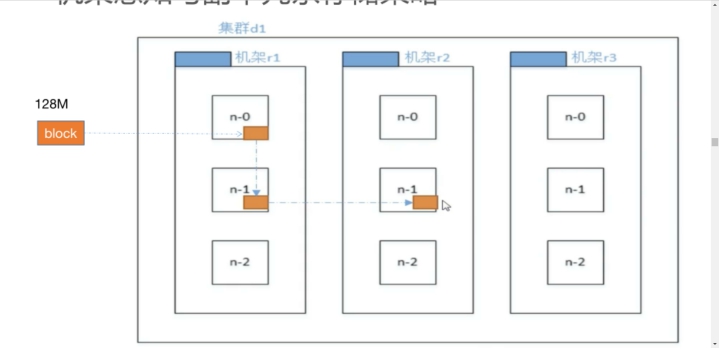
## 选择题

机架感知理论



HDFS Federation 的优缺点

优点：

负载在客户端，对nn完全没有影响

实现简单

缺点：

维护成本高，每次修改挂载表，都必须手动同步配置文件

人工维护负载均衡，不能随业务变化

NameNode HA 的作用和特点

HDFS 的特点

Reducer 的作用

安全模式的作用特点

YARN 调度器

YARN 的架构和特点

大数据的基本特征

数据体量大、数据类型多、处理速度快、价值密度低

HDFS 的架构：例如 NamdeNode、DataNode 的职责等

Hadoop 各个服务的端口号

HDFS 回收站的操作和特点

、

## 填空题

YARN 的架构组成

MapReduce 的思想，任务组成

大数据的核心问题

大数据的核心：整理，分析，预测，控制

HDFS 的 edits 文件和 fsimage 文件的区别

edits（记录操作日志）、fsimage文件（HDFS的元信息）

HDFS 的架构组成和作用

1)NameNode:就是Master,它是一个主管,管理者

1 管理HDFS的目录结构;

2 配置副本策略

3 管理数据块(Block)映射信息

4 处理客户端读写请求

2)DataNode:就是Slave,NameNode下达命令,DataNode执行实际的操作

1 存储实际的数据块

2 执行数据块的读/写操作.

3)Client:就是客户端

1 文件切分.文件上传HDFS的时候,Client将文件切分成一个一个的Block,然后进行上传.

2 与NameNode交互,获取文件的位置信息.

3 与DataNode交互,读取或者写入数据.

4 Client提供一些命令来管理HDFS,比如NameNode格式化

5 Client可以通过一些命令来访问HDFS,比如对HDFS增删操作.

4)Secondary NameNode(已弃用): Hadoop1.X的组件,并非NameNode的热备.当NameNode挂掉的时候它并不能马上替换NameNode并提供服务

1 辅助NameNode,分担其工作量,比如定期合并Fsimage和Edits,并推送给NameNode;

2 在紧急情况下,可辅助恢复NameNode

Hadoop 生态圈

hdfs，mapreduce，yarn

Hadoop 安装好后的进程

1）NameNode它是hadoop中的主服务器，管理文件系统名称空间和对集群中存储的文件的访问，保存有metadate。

2）SecondaryNameNode它不是namenode的冗余守护进程，而是提供周期检查点和清理任务。帮助NN合并editslog，减少NN启动时间。

3）DataNode它负责管理连接到节点的存储（一个集群中可以有多个节点）。每个存储数据的节点运行一个datanode守护进程。

4）ResourceManager（JobTracker）JobTracker负责调度DataNode上的工作。每个DataNode有一个TaskTracker，它们执行实际工作。

5）NodeManager（TaskTracker）执行任务。

6）DFSZKFailoverController高可用时它负责监控NN的状态，并及时的把状态信息写入ZK。它通过一个独立线程周期性的调用NN上的一个特定接口来获取NN的健康状态。FC也有选择谁作为Active NN的权利，因为最多只有两个节点，目前选择策略还比较简单（先到先得，轮换）。

7）JournalNode 高可用情况下存放namenode的editlog文件。

## 简答题（注意：只有图形作为答案是会扣分的！）

Checkpoint 的工作流程。

SecondaryName会定期地创建命名空间的检查点（CheckPoint）操作：把edits中最新的状态信息合并到fsimage文件中，防止edits过大。也可以做冷备，对一定范围内数据做快照性备份。

SecondaryNameNode的CheckPoint流程∶

（1）NameNode生成一个名叫edits.new的文件，用于记录合并过程中产生的日志信息；

（2）SecondaryNameNode将edits文件，与fsimag文件从NameNode上下载到SecondNamenode上；

（3）SecondaryNameNode将edits文件与fsimage进行合并操作，合并成一个fsimage.ckpt文件；

（4）SecondaryNameNode将生成的合并后的文件fsimage.ckpt文件上传到NameNode上；

（5）在NameNode上，将fsimage.ckpt变成fsimage文件替换NameNode上原有的fsinage文件。将edits.new文件上变成edits文件替换NameNode上原有的edits文件。

HDFS的体系结构。

HDFS采用了主从（Master/Slave）结构模型，一个HDFS集群是由一个NameNode和若干个DataNode组成的。其中NameNode 作为主服务器，管理文件系统的命名空间和客户端对文件的访问操作；

集群中的DataNode管理存储的数据。HDFS允许用户以文件的形式存储数据。从内部来看，文件被分成若干个数据块，而且这若干个数据块存放在一组DataNode上。

NameNode执行文件系统的命名空间操作，比如打开、关闭、重命名文件或目录等，它也负责数据块到具体DataNode的映射。DataNode负责处理文件系统客户端的文件读写请求，并在NaneNode的统一调度下进行数据块的创建、删除和复制工作。NameNode和DataNode都被设计成可以在普通商用计算机上运行。这些计算机通常运行的是GNU/Linux操作系统。HDFS采用Java语言开发，因此任何支持Java的机器都可以部署NameNode和DataNode。一个典型的部署场景是集群中的一台机器运行一个NaneNode实例，其他机器分别运行一个 DataNode实例。当然，并不排除一台机器运行多个DataNode实例的情况。集群中单一的 NameNode的设计则大大简化了系统的架构。NameNode是所有HDFS元数据的管理者，用户数据永远不会经过NameNode。

理解并掌握整个 MapReduce 工作的流程。

1．分片、格式化数据源

输入Map阶段的数据源，必须经过分片和格式化操作。其中：

分片操作：指的是将源文件划分为大小相等的小数据块（Hadoop2.x中默认128M），也就是分片(split)，Hadoop会为每一个分片构建一个Map任务，并由该任务运行自定义的map()函数，从而处理分片里的每一条记录；格式化操作：将划分好的分片（split）格式化为键值对<key，value>形式的数据，其中，key代表偏移量，value代表每一行内容。

2．执行MapTask

每个Map任务都有一个内存缓冲区（缓冲区大小100M），输入的分片（split）数据经过Map任务处理后的中间结果，会写入内存缓冲区中。如果写入的数据达到内存缓冲的阀值（80M），会启动一个线程将内存中的溢出数据写入磁盘，同时不影响map中间结果继续写入缓冲区。在溢写过程中，MapReduce框架会对Key进行排序，如果中间结果比较大，会形成多个溢写文件，最后的缓冲区数据也会全部溢写入磁盘形成一个溢写文件，如果是多个溢写文件，则最后合并所有的溢写文件为一个文件。

3．执行Shuffle过程

MapReduce工作过程中，map阶段处理的数据如何传递给Reduce阶段，这是MapReduce框架中关键的一个过程，这个过程叫做Shuffle。Shuffle会将MapTask输出的处理结果数据，分发给ReduceTask，并在分发的过程中，对数据按key进行分区和排序。

4．执行ReduceTask

输入ReduceTask的数据流是<key，{value list}>形式，用户可以自定义reduce()方法进行逻辑处理，最终以<key，value>的形式输出。

5．写入文件

MapReduce框架会自动把ReduceTask生成的<key,value>传入OutputFormat的write方法，实现文件的写入操作。

MapReduce 的编程模型，理解各个 k 和 v 的值及其数据类型。

MapReduce 1 与 MapReduce 2 的区别

NameNode 的元信息具体包括哪些内容？

YARN 三种调度器，并简要说明其工作方法

## 编程题：

理解并默写 WordCount 全部代码。

傻逼吧高小婷

HDFS Shell 基本操作命令 ，例如：创建、上传、下载、追加、统计、查看等等

（1）在HDFS上创建名为model的目录

（2）将本地的文件file上传至model目录下

（3）将本地的文件file1内容追加至model目录下的file中

（4）查看model的目录信息

（5）将model下的file文件下载至本地

正确答案∶

hdfs dfs -mkdir model

hdfs dfs -put file model

hdfs dfs -appendToFile file1 model/file

hdfs dfs -ls model

hdfs dfs -get file model