

hadoop之hdfs简介

1、产生背景及定义

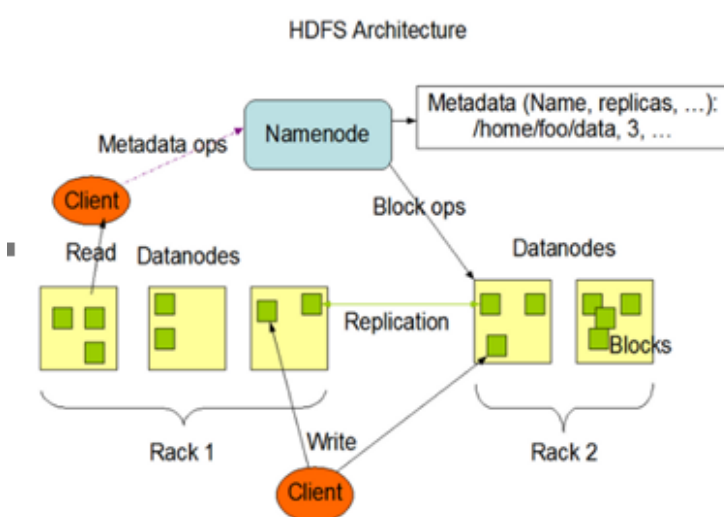
1.1、产生背景：随着数据量得越来越大，在一个操作系统存不下所有得数据，那么就需要分配到更多得操作系统管理得磁盘中，但是 不方便管理和维护，迫切需要一种操作系统来管理多台机器上得文件，这就是分布式文件管理系统。hdfs只是分布式 文件管理系统中得一种。 1.2、hdfs定义： hdfs (Hadoop Distributed File System)，他是一个文件系统，用于存储文件，通过目录树来定位文件;其次，他是 分布式的，由很多服务器联合起来实现其功能，集群中的服务器有各自的角色。 hdfs使用场景：适合一次写入，多次读出的场景，且不支持文件修改哦，适合用来做数据分析，并不适合用来做网盘应 用。

2、hdfs优缺点

2.1、优点： 2.1.1、高容错性 数据自动保存多个副本，通过增加副本的形式提高容错性 某个副本丢失，它可以自动恢复 2.1.2、适合处理大数据 数据规模：能够处理数据规模达到gb、tb甚至pb级别的数据 文件规模：能够处理百万规模以上的文件数量，数量相当之大 2.1.3、可构建在廉价机器上，通过多副本机制，提高可靠性 2.2、缺点： 2.2.1、不适合低延迟数据访问，比如毫秒级的存储数据，是做不到的。 2.2.2、无法高效的对大量小文件进行存储 存储大量小文件的话，他会占用namenode大量内存来存储文件目录和块信息，这样是不可取的，由于namenode内存是有限的。 小文件存储的寻址时间会超出读取时间，违反了hdfs的设计目标 2.2.3、不支持并发写入、文件随机修改 一个文件只能有一个线程写，不允许多个线程同时写 仅支持数据append（追加），不支持文件随机修改

3、hdfs组成架构

3.1、组成架构图



1) NameNode (nn)：就是Master，它是一个主管、管理者。

- (1) 管理HDFS的名称空间;
- (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。

3.2、client即客户端 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 与 NameNode交互，获取文件位置信息 与DataNode交互，读取或写入数据 Client提供一些命令来管理hdfs，比如namenode格式化 Client通过一些命令来访问hdfs，比如对hdfs的增删改查操作 3.3、Secondary NameNode:并非NameNode的热备，当NameNode挂掉的时候并不能马上替换namenode并提供服务 辅助namenode，分担其工作量比如定期融合Fsimage和Edite，并推送给NameNode 在紧急的情况下，可以辅助恢复NameNode

4、hdfs块大小

4.1、hdfs中的文件在物理上是分块存储（Block），块的大小可以通过配置参数（dfsblocksize）来规定，默认大小在2.x中是128M，老版本中是64M

2 如果寻址时间约为10ms，即直找到目标block的时间为10ms。

3 寻址时间为传输时间的1%时，则为最佳状态。
因此，传输时间
 $= 10\text{ms} / 0.01 = 1000\text{ms} = 1\text{s}$

4 而目前磁盘的传输速率普遍为100MB/s。

5 block大小
 $= 1\text{s} * 100\text{MB/s} = 100\text{MB}$

1 集群中的block

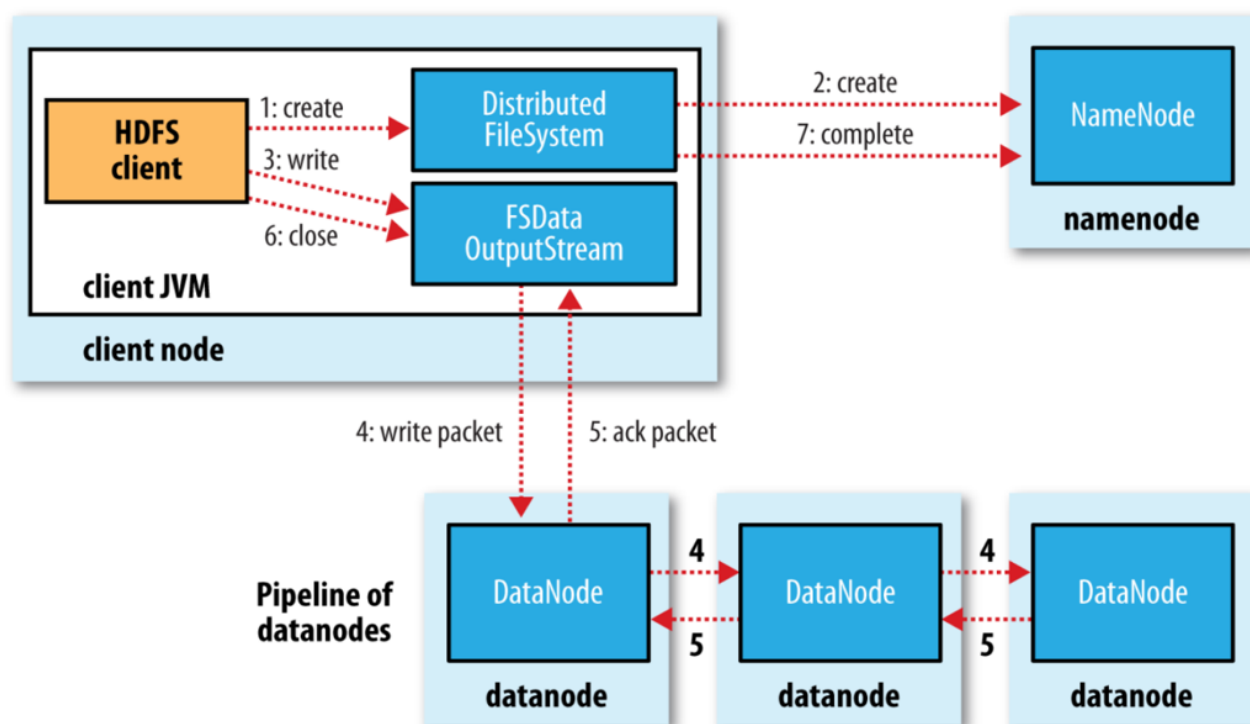
block1

block2

blockn

4.2、为什么块大小不能设置太小？块大小设置太小，会增加寻址时间，程序一直在找块的开始位置 块大小设置太大，从磁盘传输数据的时间会明显大于定位这个块开始位置所需时间，导致程序在处理这块数据时，非常慢
hdfs块大小设置主要取决于磁盘传输速率

5、hdfs数据写入流程



6、hdfs数据读取流程

