1.介绍

一句话(官方):分布式存储系统HDFS(Hadoop Distributed File System)。其实就是一个文件系统，类似于linux的文件系统。有目录，目录下可以存 储文件。但它又是一个分布式的文件系统。

基本原理

1. 将文件切分成等大的数据块，分别存储到多台机器上。
2. 每个数据块存在多个备份。 将数据切分、容错、负载均衡等功能透明化。

3. 可将HDFS看成是一个巨大、具有容错性的磁盘。

缺点

1. 不适合存储大量小文件。
2. 不适合低延迟数据访问 。
3. 不支持多用户写入及任意修改文件 。

2.HDFS实现原理

数据块

1. 每个磁盘都有默认的数据块大小，这是磁盘进行数据读/写的最小单位。构建于单个磁盘之 上的文件系统通过磁盘块来管理该文件系统中的块，该文件系统块的大小可以是磁盘块的整数倍。
2. HDFS 同样也有块 (block) 的概念，但是大得多，默认为 128 MB 。与单一磁盘上的文件系统相似， HDFS 上的文件也被划分为块大小的多个分块 (chunk) ，作为独立的存储单元。但与其他 文件系统不同的是， HDFS 中小于一个块大小的文件不会占据整个块的空间。(好像老版本的会占用整个空间)

2.1 NameNode

NameNode是HDFS架构中的主节点。 当NameNode启动时会从fsimage中读取数据,缩短启动时间,并会写入editlog, Namenode只有 在启动时候合并fsimage和edits log.

功能

1. 管理各个从节点的状态(DataNode)。
2. 记录存储在HDFS上的所有数据的元数据信息。例如:block存储的位置，文件大小，文件权限，文件层级等等。这些信息以两个文件形式永久保存在本地磁盘上。

    a. 命名空间镜像文件(FsImage): fsimage是HDFS文件系统存于硬盘中的元数据检查点，里面记录 了自最后一次检查点之前HDFS文件系统中所有目录和文件的序列化信息

    b. 编辑日志(edit-logs)文件:保存了自最后一次检查点之后所有针对HDFS文件系统的操作，比如: 增加文件、重命名文

件、删除目录等等。

3. 记录了存储在HDFS上文件的所有变化，例如文件被删除，namenode会记录到editlog中。
4. 接受DataNode的心跳和各个datanode上的block报告信息，确保DataNode是否存活。
5. 负责处理所有块的复制因子。
6. 如果DataNode节点宕机，NameNode会选择另外一个DataNode均衡复制因子，并做负载均衡。

2.2 Sencondary NameNode

S NameNode是NameNode的助手，不要将其理解成是NameNode的备份。Secondary NameNode 的整个目的在HDFS中提供一个Checkpoint Node，所以也被叫做checkpoint node。

功能

1. 定时的从NameNode获取EditLogs,并更新到FsImage上。
2. 一旦它有新的fsimage文件，它将其拷贝回NameNode上，NameNode在下次重启时会使用这个新的fsimage文件，从而减少重启的时间。
3. 关于NameNode是什么时候将改动 写到edit logs中的?

这个操作实际上是由 DataNode的写操作触发的，当我们往 DataNode写文件时，DataNode会跟 NameNode通信，告诉NameNode什么文 件的第几个block放在它那里，NameNode 这个时候会将这些元数据信息写到edit logs文件中。

2.3 DataNode

DataNode是HDFS架构的从节点，管理各自节点的Block信息。

功能

1. 多个数据实际是存储到DataNode上面。
2. DataNode分别运行在独立的节点上。
3. DataNode执行客户端级别的读写请求。
4. 向NameNode发送心跳(默认是3s)，报告各自节点的健康状况。

3. HDFS读写流程

3.1 写数据

1. client将数据分片

2. client请求namenode，要将多个块写入到HDFS。例如这里的Block A和Block B。

3. NameNode会给client赋予写权限，并为client提供可以写入数据的DataNode的IP地址。Namenode在选

择可写入数据的dataNode的规则是结合了DN的健康状态、复制因子、机架感知等因素随机选择的DN。

假如复制因子是3(默认值)，那么会为每个block返回三个IP地址。例如NN为client提供了以下的IP地

址列表。

4. 在写入数据之前，client首先 要确认namenode提供的ip列表 是否准备好了接收数据。Client通过连接各个块的ip列表来为 每个块创建流水线。（传递的方式）

5. 流水线建立好以后，client将 会向流水线中写入数据（block 是并行写入的）。Client只会将block A向 DN1复制。其他节点复制是在DN 之间完成的。

6. 当数据复制到所有的DN完成之 后，按照ip地址列表相反的方向，依次反馈写入成功的信息。

7. DN1将确认信息反馈给client，client再将确认信息反馈给NN， NN更新元数据信息，client关 闭pipline。

3.2 文件读取

1. Client请求namenode要读取exaple. txt文件。

2. NN根据自己的元数据信息，反馈给 client一个DataNode的列表(存储Block A和B)。

3. Client连接DN，读取BlockA,Block B的数据。

4. Client合并block A和Block B的数 据。

4. HDFS balancer

集群磁盘数据不均衡导致的原因有很多情况。

1. 添加新的DataNode节点。

2. 人为干预，修改block副本数。

3. 各个机器磁盘大小不一致。

4. 长时间运行大量的delete操作等。

5. HDFS快照

Hdfs的快照（snapshot）是在某一时间点对指定文件系统拷贝，快照采用只读模式，可以对重要数据进行恢复、防止用户错误性的操作。

快照分两种：

一种是：建立文件系统的索引，每次更新文件不会真正的改变文件，而是新开辟一个空间用来保存更改的文件，(hdfs属于该种)

一种是：拷贝所有的文件系统

> 和VM做快照差不多,相当于做了个备份,可以做回滚

1. 快照创建是瞬间的:成本是0(1)排除查找信息节点的时间 。
2. 额外的内存使用仅仅当对快照进行修改时产生：内存使用时0(M),M是修改文件/目录的数量。
3. 是一个只读的基于时间点文件系统拷贝。快照可以是整个文件系统的也可以是一部分。常用来作为数据备 份，防止用户错误和容灾。
4. 在datanode 上面的blocks 不会复制，做Snapshot 的文件是纪录了block的列表和文件的大小，但是没有数据的复制 Snapshot 并不会影响HDFS 的正常操作:
5. 修改会按照时间的反序记录，这样可以直接读取到最新的数据。
6. 快照数据是 当前数据减去修改的部分计算出来的。
7. 快照会存储在snapshottable的目录下。snapshottable下存储的snapshots 最多为65535个。没有限制snapshottable目录 的数量。管理员可以设置任何的目录成为snapshottable。如果snapshottable里面存着快照，那么文件夹不能删除或者 改名。

6. Hadoop 配额设置

6.1 什么是配额

Hadoop 分布式文件系统(HDFS)允许管理员为每个目录设置配额,限制该目录下占用空间大小,和目录/文件数量.新建立的目录没有配额。最 大的配额是Long.Max_Value。配额为 1 可以强制目录保持为空。

目录配额是对目录树上该目录下的名字数量做硬性限制。如果创建文件或目录时超过了配额， 该操作会失败。重命名不会改变该目录的配额;如果重命名操作会导致违反配额限制，该操作 将会失败。如果尝试设置一个配额而现有文件数量已经超出了这个新配额，则设置失败。

所有超出配额的操作都会失败

配额和 fsimage 保持一致。当启动时，如果 fsimage 违反了某个配额限制(也许 fsimage 被偷 偷改变了)，则启动失败并生成错误报告。设置或删除一个配额会创建相应的日志记录。

6.2 配额种类

1. Name Quotas:设置某一个目录下文件总数

> 从文件数目上限制

2. Space Quotas:设置某一个目录下可使用空间大小

> 从空间上限制

注意：这里需要特别注意的是“Space Quota”的设置所看的不是 Hdfs 的文件大小，而是写入 Hdfs 所有 block 块的大小。包含备份的部分，而且不足一个块也要按照一个块计算

> hdfs 的配额管理是跟着目录走，如果目录被重命名，配额依然有效。 麻烦的是，在设置完配额以后，如果超过限制，虽然文件不会写入到 hdfs，但是文件名依然会存在，只是文件 size 为 0。当加大配额设置后，还需要将之前的空文件删除才能进一步写入。如果新设置的 quota 值，小于该目录现有的 Name Quotas 及 Space Quotas，系统并不会给 出错误提示，但是该目录的配置会变成最新设置的 quota。

设置

启用设定:hadoop dfsadmin -setQuota 10000 /user/seamon
清除設定: hadoop dfsadmin -clrQuota /user/seamon
可以使用m,g,t 代表 MB,GB,TB
启用设定: hadoop dfsadmin -setSpaceQuota 1g /user/seamon/
清除設定: hadoop dfsadmin -clrSpaceQuota /user/seamon

7. 复制因子

HDFS为我们提供了可靠的存储，就是因为这个复制因子。默认复制因子是3。

复制因子是每个block备份的数目，nameNode保证每个block在集群中有复制因子数目的备份，不多也不少。每个block的备份分在不同的dataNode中。

通常情况下，当复制因数是3时，HDFS的放置策略是一个数据块放置到本地机架的一个节点，另一份数据块放置到本地机架的另一个节点，最后一份数据块放置到不同机架的一个节点。这个策略减少了机架之间的通信而极大提高了写性能。机架发生故障的可能性远小于节点，这个策略并不影响数据的可靠性和可用性，但是，在读数据时，它确实降低了整体带宽，因为数据块仅放置在两个而不是三个机架上，这种策略下，一个文件的复制块没有在机架之间均匀分布，三分之一的复制块在一个节点上；三分之二的复制块在一个机架上，另三分之一的复制块均匀分布在另外的机架上。这个策略提高了写性能，没有降低数据可靠性或读性能。

为减少全局的带宽和读延迟，HDFS尝试从离要求读的客户端最近的地方读取复制块，如果读者节点与复制块节点在同一机架上，则这个复制块优先用来满足读请求，如果HDFS集群有多个数据中心，则优先使用本地数据中心中的复制块，而不是远方的数据块。

> 就近原则

> https://www.aboutyun.com/forum.php?mod=viewthread&tid=17306

8. 安全模式

安全模式是hadoop的一种保护机制，用于保证集群中的数据块的安全性。如果HDFS处于安全模式，则表示HDFS是只读状态。

当集群启动的时候，会首先进入安全模式。当系统处于安全模式时会检查数据块的完整性。假设我们设置的副本数（即参数dfs.replication）是5，那么在datanode上就应该有5个副本存在，假设只存在3个副本，那么比例就是3/5=0.6。在配置文件hdfs-default.xml中定义了一个最小的副本的副本率0.999

dfs.namenode.safemode.threshold-pct

我们的副本率0.6明显小于0.99，因此系统会自动的复制副本到其他的dataNode,使得副本率不小于0.999.如果系统中有8个副本，超过我们设定的5个副本，那么系统也会删除多余的3个副本(估计是心跳时由NameNode抉择,指定DataNode去删除)。当副本率达标后,过30秒之后,NameNode自动退出安全模式

- 查看安全模式状态：`hdfs dfsadmin -safemode get`
- 进入安全模式状态：`hdfs dfsadmin -safemode enter`
- 离开安全模式状态：`hdfs dfsadmin -safemode leave`

> https://www.cnblogs.com/TiePiHeTao/p/5105682165be5a124a01c2cd06a89c64.html

9.空间回收

如果一个文件被用户或应用程序删除了，它并不立刻从HDFS中删除，相反，HDFS首先改变它的名字，将其放到/trash目录中，只要它保留在/trash中，这个文件可以迅速恢复。

当/trash中的文件到达其生命期后，NameNode从HDFS命名空间中删除这个文件。删除这个文

件将释放与之关联的数据块的空间。需要注意的是，用户删除一个文件的时刻与HDFS增加剩余空间的时刻之间相比，有一个适当的滞后。

只要这个文件还在/trash目录，用户可以恢复一个删除的文件。如果用户想恢复，则他/她可以转到/trash目录，恢复这个文件。/trash目录仅包含最新的被删除文件，/trash目录与其他目录相似，除了一点特征之外：HDFS施加特别的自动删除策略给这个目录下的文件。目前，这个缺省的策略是删除超过6个小时的文件

> https://www.aboutyun.com/forum.php?mod=viewthread&tid=17306

> https://blog.csdn.net/weixin_42297075/article/details/104365795
> https://www.aboutyun.com/thread-18075-1-3.html
> https://www.cnblogs.com/nucdy/p/5684196.html
> https://www.aboutyun.com/thread-17301-1-3.html
> https://www.aboutyun.com/forum.php?mod=viewthread&tid=17304