|  |  |  |
| --- | --- | --- |
| name | value | Description |
| dfs.default.chunk.view.size | 32768 | namenode的http访问页面中针对每个文件的内容显示大小，通常无需设置。 |
| dfs.datanode.du.reserved | 1073741824 | 每块磁盘所保留的空间大小，需要设置一些，主要是给非hdfs文件使用，默认是不保留，0字节 |
| dfs.name.dir | /opt/data1/hdfs/name, /opt/data2/hdfs/name, /nfs/data/hdfs/name | NN所使用的元数据保存，一般建议在nfs上保留一份，作为1.0的HA方案使用，也可以在一台服务器的多块硬盘上使用 |
| dfs.web.ugi | nobody,nobody | NN,JT等所使用的web tracker页面服务器所使用的用户和组 |
| dfs.permissions | true | false | dfs权限是否打开，我一般设置false，通过开发工具培训别人界面操作避免误操作，设置为true有时候会遇到数据因为权限访问不了。 |
| dfs.permissions.supergroup | supergroup | 设置hdfs超级权限的组，默认是supergroup，启动hadoop所使用的用户通常是superuser。 |
| dfs.data.dir | /opt/data1/hdfs/data, /opt/data2/hdfs/data, /opt/data3/hdfs/data, ... | 真正的datanode数据保存路径，可以写多块硬盘，逗号分隔 |
| dfs.datanode.data.dir.perm | 755 | datanode所使用的本地文件夹的路径权限，默认755 |
| dfs.replication | 3 | hdfs数据块的复制份数，默认3，理论上份数越多跑数速度越快，但是需要的存储空间也更多。有钱人可以调5或者6 |
| dfs.replication.max | 512 | 有时dn临时故障恢复后会导致数据超过默认备份数。复制份数的最多数，通常没什么用，不用写配置文件里。 |
| dfs.replication.min | 1 | 最小份数，作用同上。 |
| dfs.block.size | 134217728 | 每个文件块的大小，我们用128M，默认是64M。这个计算需要128\*1024^2，我碰上过有人直接写128000000，十分浪漫。 |
| dfs.df.interval | 60000 | 磁盘用量统计自动刷新时间，单位是毫秒。 |
| dfs.client.block.write.retries | 3 | 数据块写入的最多重试次数，在此次数之前不会捕获失败。 |
| dfs.heartbeat.interval | 3 | DN的心跳检测时间间隔。秒 |
| dfs.namenode.handler.count | 10 | NN启动后展开的线程数。 |
| dfs.balance.bandwidthPerSec | 1048576 | 做balance时所使用的每秒最大带宽，使用字节作为单位，而不是bit |
| dfs.hosts | /opt/hadoop/conf/hosts.allow | 一个主机名列表文件，这里的主机是允许连接NN的，必须写绝对路径，文件内容为空则认为全都可以。 |
| dfs.hosts.exclude | /opt/hadoop/conf/hosts.deny | 基本原理同上，只不过这里放的是禁止访问NN的主机名称列表。这在从集群中摘除DN会比较有用。 |
| dfs.max.objects | 0 | dfs最大并发对象数，HDFS中的文件，目录块都会被认为是一个对象。0表示不限制 |
| dfs.replication.interval | 3 | NN计算复制块的内部间隔时间，通常不需写入配置文件。默认就好 |
| dfs.support.append | true | false | 新的hadoop支持了文件的APPEND操作，这个就是控制是否允许文件APPEND的，但是默认是false，理由是追加还有bug。 |
| dfs.datanode.failed.volumes.tolerated | 0 | 能够导致DN挂掉的坏硬盘最大数，默认0就是只要有1个硬盘坏了，DN就会shutdown。 |
| dfs.secondary.http.address | 0.0.0.0:50090 | SNN的tracker页面监听地址和端口 |
| dfs.datanode.address | 0.0.0.0:50010 | DN的服务监听端口，端口为0的话会随机监听端口，通过心跳通知NN |
| dfs.datanode.http.address | 0.0.0.0:50075 | DN的tracker页面监听地址和端口 |
| dfs.datanode.ipc.address | 0.0.0.0:50020 | DN的IPC监听端口，写0的话监听在随机端口通过心跳传输给NN |
| dfs.datanode.handler.count | 3 | DN启动的服务线程数 |
| dfs.http.address | 0.0.0.0:50070 | NN的tracker页面监听地址和端口 |
| dfs.https.enable | true | false | NN的tracker是否监听在HTTPS协议，默认false |
| dfs.datanode.https.address | 0.0.0.0:50475 | DN的HTTPS的tracker页面监听地址和端口 |
| dfs.https.address | 0.0.0.0:50470 | NN的HTTPS的tracker页面监听地址和端口 |