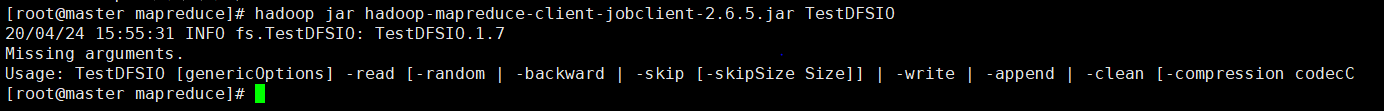
1. TestDFSIO write的性能测试

进入该路径 cd /root/hadoop-2.6.5/share/hadoop/mapreduce/



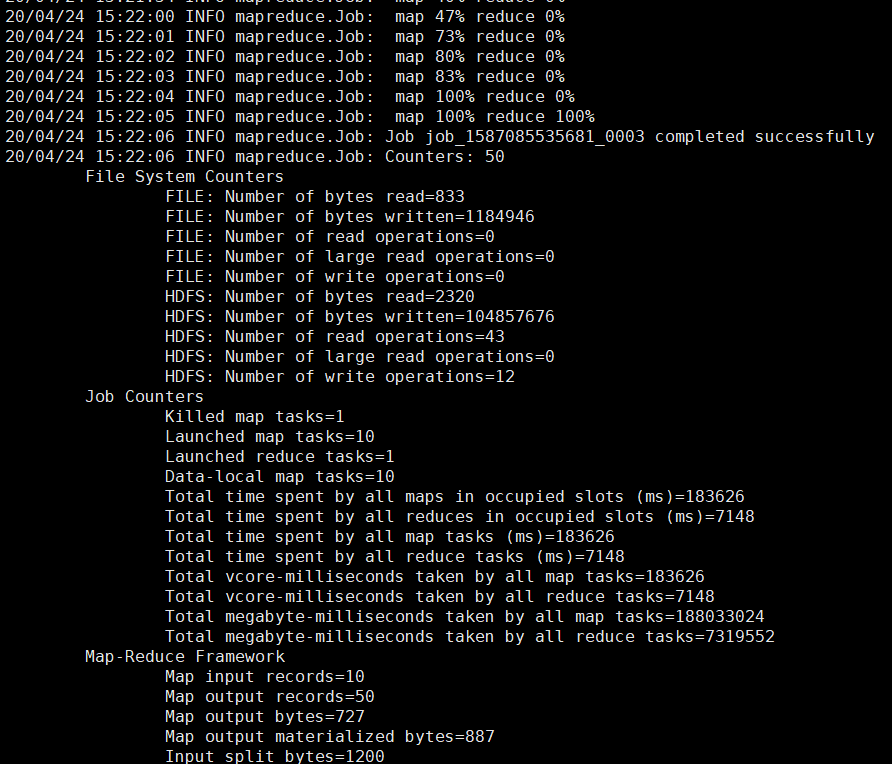
测试Hadoop写的速度：

例子：将往HDFS中写入10个10MB的文件：

1.1 运行 hadoop jar hadoop-mapreduce-client-jobclient-2.6.5.jar TestDFSIO -write -nrFiles 10 -size 10MB



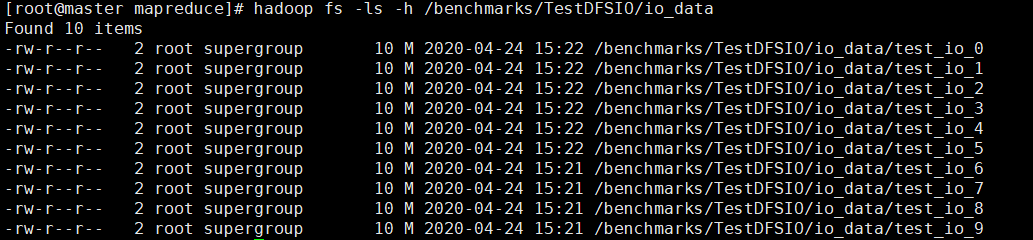
1.2 运行过程：



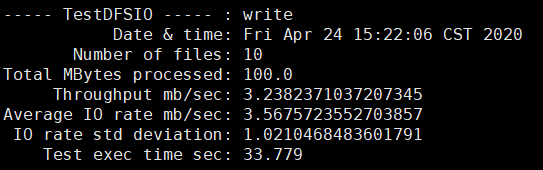
1.3 查看写入数据：

hadoop fs -ls -h  /benchmarks/TestDFSIO/io\_data

下图为写入HDFS的10个10MB的数据包



1.4查看执行结果



执行写操作

文件数量为10

总写入的数据量 100M

写速率：

总共需要写入的数据量/每个map任务实际写入数据的执行时间之和：3.238

每个map需要写入的数据量/每个map任务实际写入数据的执行时间之和：3.35675

上一个值的标准差：1.0210468

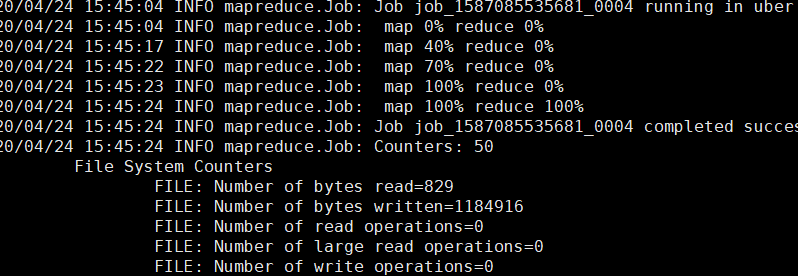
整个job执行时间：33.779

1. TestDFSIO read的性能测试

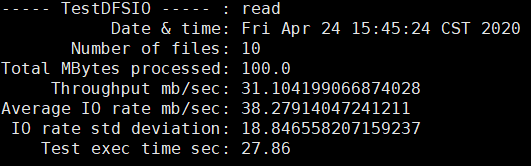
测试Hadoop读文件的速度

2.1 运行 hadoop jar hadoop-mapreduce-client-jobclient-2.6.5.jar TestDFSIO -read -nrFiles 10 -size 10MB ，读取刚才写入的文件

2.2 运行过程



2.3 查看运行的情况



文件数量为10

总读取的数据量 100M

读速率：

总共需要读取的数据量/每个map任务实际读取数据的执行时间之和：31.104199

每个map需要读取的数据量/每个map任务实际读取数据的执行时间之和：38.27914

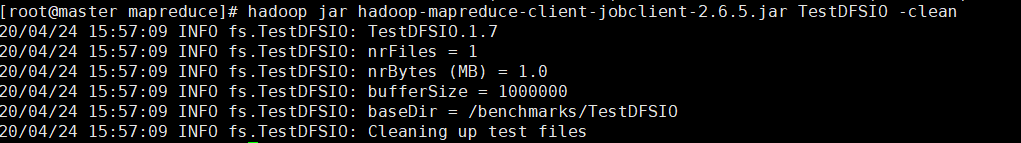
上一个值的标准差：18.846558

整个job执行时间：27.86

结果各项意思与write相同，但其读速率比写速率快很多，而总执行时间非常接近。真正测试时，应该用较大的数据量来执行，才可体现出二者的差异。

* 1. 清空测试数据

hadoop jar hadoop-mapreduce-client-jobclient-2.6.5.jar TestDFSIO -clean



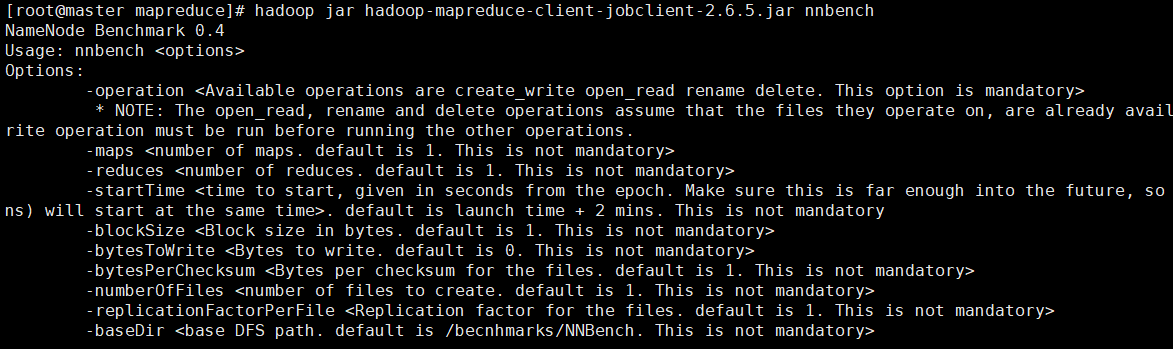
清空数据完毕

1. Nnbench测试（NameNode测试）

Nnbench用于测试NameNode的测试，它会生成很多与HDFS相关的请求，给NameNode施加较大的压力。这个测试能够在HDFS上创建、读取、重命名和删除文件操作。

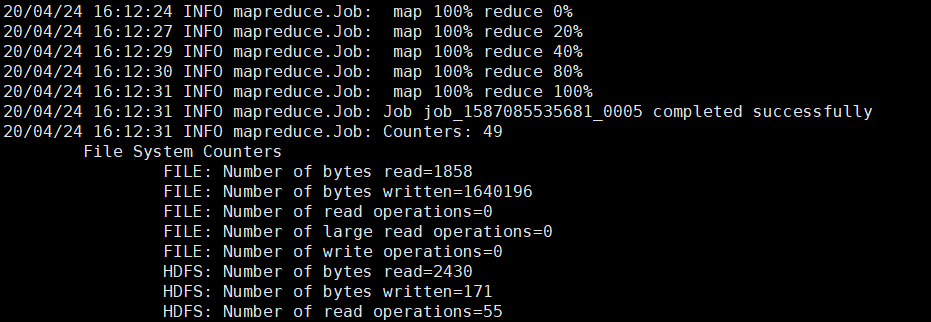
3.1 查看nnbench选项

hadoop jar hadoop-mapreduce-client-jobclient-2.6.5.jar nnbench

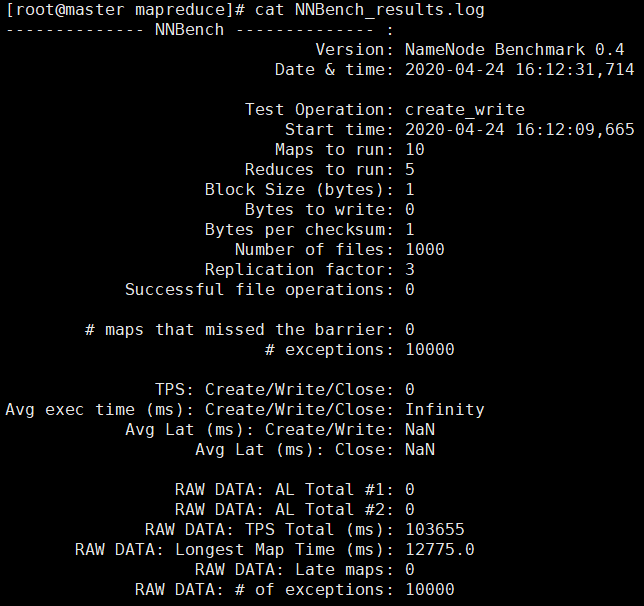


3.2 以下例子使用10个mapper和5个reducer来创建1000个文件

hadoop jar hadoop-mapreduce-client-jobclient-2.6.5.jar nnbench -operation create\_write -maps 10 -reduces 5 -numberOfFiles 1000 -replicationFactorPerFile 3 – readFileAfterOpen true



3.3 查看结果

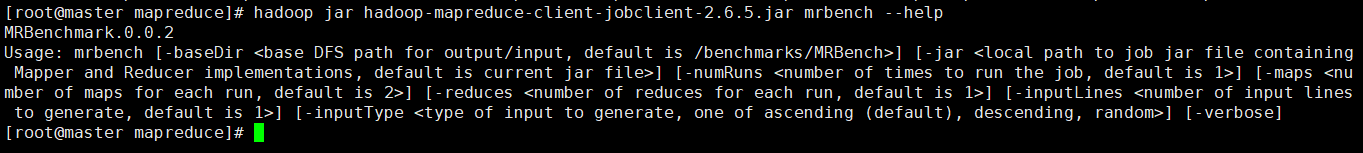


1. Mrbench测试

Mrbench会重复执行一个小作业，用于检查在机群上小作业的运行十分可重复以及运行是否高效。

4.1 查看帮助

hadoop jar hadoop-mapreduce-client-jobclient-2.6.5.jar mrbench --help



4.2 运行一个小作业2次

hadoop jar hadoop-mapreduce-client-jobclient-2.6.5.jar mrbench -numRuns 2

运行结果：

