E516 - ECC Assignment 1 (Fall ‘22)

Submitted by : Aniket Kale (ankale@iu.edu)

Setup

1. Create Jetstream 2 instance (medium-allocation). Connect using SSH to Jetstream 2.
2. Make user "hadoopuser".
3. Give sudo access to "hadoop user".
4. SSH to local-host.
5. Download Hadoop latest version and configure using this tutorial.
6. Start using command "start-dfs.sh" and "start-yarn.sh"
7. Run command "jps" to check if all service are running

```
hadoopuser@assignment1-ankale-main:~$ jps
17091 SecondaryNameNode
29364 Jps
17415 ResourceManager
16648 NameNode
17582 NodeManager
16830 DataNode
hadoopuser@assignment1-ankale-main:~$
```

Test wordcount program:

```
                Peak Reduce Physical memory (bytes)=225320960
                Peak Reduce Virtual memory (bytes)=2741301248
        Shuffle Errors
                BAD_ID=0
                CONNECTION=0
                IO_ERROR=0
                WRONG_LENGTH=0
                WRONG_MAP=0
                WRONG_REDUCE=0
        File Input Format Counters
                Bytes Read=69
        File Output Format Counters
                Bytes Written=47
2022-11-03 01:13:03,334 INFO streaming.StreamJob: Output directory: /word_count_in_python/output
hadoopuser@assignment1-ankale-main:~$ hdfs dfs -cat /word_count_in_python/output/part-00000
GeeksforGeeks   1
Hello   2
I       2
Intern  1
am      2
an      1
hadoopuser@assignment1-ankale-main:~$
```

Part 1:

Get top 3 IPs for every hour from the log files.

1.1 Run map-reduce for top 3 on "sample.log":

a. Move "sample.log", "mapper-ip.py", "reducer-main.py" to Jetstream2

b. Copy “sample.log” to HDFS:

```
hadoopuser@assignment1-ankale-main:~/part1$ ls
mapper-ip.py  reducer-main.py  sample.log
hadoopuser@assignment1-ankale-main:~/part1$ cd ..
hadoopuser@assignment1-ankale-main:~$ cd part1/
hadoopuser@assignment1-ankale-main:~/part1$ chmod 777 mapper-ip.py reducer-main.py sample.log
hadoopuser@assignment1-ankale-main:~/part1$ cd ..
hadoopuser@assignment1-ankale-main:~$ hdfs dfs -mkdir smallLog
mkdir: `hdfs://127.0.0.1:9000/user/hadoopuser': No such file or directory
hadoopuser@assignment1-ankale-main:~$ hdfs dfs -mkdir /smallLog
hadoopuser@assignment1-ankale-main:~$ hdfs dfs -ls
ls: `.': No such file or directory
hadoopuser@assignment1-ankale-main:~$ hdfs dfs -ls /smallLog
hadoopuser@assignment1-ankale-main:~$ hdfs dfs -copyFromLocal /home/hadoopuser/part1/sample.log /smallLog
hadoopuser@assignment1-ankale-main:~$ hdfs dfs -ls /smallLog
Found 1 items
-rw-r--r--   1 hadoopuser supergroup     102399 2022-11-05 17:17 /smallLog/sample.log
hadoopuser@assignment1-ankale-main:~$
```

c. Run the following command:

```
  > hadoop jar
/home/hadoopuser/hadoop-3.3.4/share/hadoop/tools/lib/hadoop-streaming-3.3.
4.jar
-input /smallLog/sample.log
-output /smallLog/output -mapper "python3
/home/hadoopuser/part1/mapper-ip.py"
-reducer "python3 /home/hadoopuser/part1/reducer-main.py"
```

d. Run command:

```
hadoopuser@assignment1-ankale-main:~$ hadoop jar /home/hadoopuser/hadoop-3.3.4/share/hadoop/tools/lib/hadoop-streaming-3.3.4.jar
  -input /smallLog/sample.log -output /smallLog/output  -mapper "python3 /home/hadoopuser/part1/mapper-ip.py" -reducer "python3
/home/hadoopuser/part1/reducer-main.py"
packageJobJar: [/tmp/hadoop-unjar7987586106297474119/] [] /tmp/streamjob2315091795510642602.jar tmpDir=null
2022-11-05 17:29:30,371 INFO client.DefaultNoHARMFailoverProxyProvider: Connecting to ResourceManager at /127.0.0.1:8032
2022-11-05 17:29:30,510 INFO client.DefaultNoHARMFailoverProxyProvider: Connecting to ResourceManager at /127.0.0.1:8032
2022-11-05 17:29:30,690 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/hadoopus
er/.staging/job_1667431627519_0009
2022-11-05 17:29:30,894 INFO mapred.FileInputFormat: Total input files to process : 1
2022-11-05 17:29:31,343 INFO mapreduce.JobSubmitter: number of splits:2
2022-11-05 17:29:31,504 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1667431627519_0009
2022-11-05 17:29:31,504 INFO mapreduce.JobSubmitter: Executing with tokens: []
2022-11-05 17:29:31,642 INFO conf.Configuration: resource-types.xml not found
2022-11-05 17:29:31,643 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2022-11-05 17:29:31,706 INFO impl.YarnClientImpl: Submitted application application_1667431627519_0009
2022-11-05 17:29:31,737 INFO mapreduce.Job: The url to track the job: http://assignment1-ankale-main.js2local:8088/proxy/applica
tion_1667431627519_0009/
2022-11-05 17:29:31,739 INFO mapreduce.Job: Running job: job_1667431627519_0009
2022-11-05 17:29:36,794 INFO mapreduce.Job: Job job_1667431627519_0009 running in uber mode : false
2022-11-05 17:29:36,795 INFO mapreduce.Job:  map 0% reduce 0%
```

e. Show output:

```
                Peak Reduce Physical memory (bytes)=226676736
                Peak Reduce Virtual memory (bytes)=2743332864
        Shuffle Errors
                BAD_ID=0
                CONNECTION=0
                IO_ERROR=0
                WRONG_LENGTH=0
                WRONG_MAP=0
                WRONG_REDUCE=0
        File Input Format Counters
                Bytes Read=106495
        File Output Format Counters
                Bytes Written=58
2022-11-05 17:29:45,928 INFO streaming.StreamJob: Output directory: /smallLog/output
hadoopuser@assignment1-ankale-main:~$ hdfs dfs -ls /smallLog/output
Found 2 items
-rw-r--r--   1 hadoopuser supergroup          0 2022-11-05 17:29 /smallLog/output/_SUCCESS
-rw-r--r--   1 hadoopuser supergroup         58 2022-11-05 17:29 /smallLog/output/part-00000
hadoopuser@assignment1-ankale-main:~$ hdfs dfs -cat /smallLog/output/part-00000
03      66.111.54.249   38
03      5.211.97.39     36
03      66.249.66.194   31
hadoopuser@assignment1-ankale-main:~$
```

1.2 Run map-reduce for top 3 on "access.log" (3.5 GB file):

a. Move “access.log” to Jetstream2 and copy to HDFS

```
hadoopuser@assignment1-ankale-main:~$ hdfs dfs -copyFromLocal /home/hadoopuser/part1/access.log /accessLog/access.log
hadoopuser@assignment1-ankale-main:~$ hdfs dfs -ls /accessLog/
Found 1 items
-rw-r--r--   1 hadoopuser supergroup 3502440823 2022-11-06 15:56 /accessLog/access.log
hadoopuser@assignment1-ankale-main:~$
```

b. Run MapReduce

```
hadoopuser@assignment1-ankale-main:~$ hadoop jar /home/hadoopuser/hadoop-3.3.4/share/hadoop/tools/lib/hadoop-streaming-3.3.4.jar  -input /accessLog/access.log -output /accessLog/output  -mapper "python3 /home/hadoopuser/part1/mapper-ip.py"  -reducer "python3  /home/hadoopuser/part1/reducer-main.py"
packageJarJar: [/tmp/hadoop-unjar11338771494255523375/] [] /tmp/streamjob2095926519263414969.jar tmpDir=null
2022-11-06 16:01:33,945 INFO client.DefaultNoHARMFailoverProxyProvider: Connecting to ResourceManager at /127.0.0.1:8032
2022-11-06 16:01:34,068 INFO client.DefaultNoHARMFailoverProxyProvider: Connecting to ResourceManager at /127.0.0.1:8032
2022-11-06 16:01:34,233 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/hadoopuser/.staging/job_1667431627519_0010
2022-11-06 16:01:34,405 INFO mapred.FileInputFormat: Total input files to process : 1
2022-11-06 16:01:34,423 INFO net.NetworkTopology: Adding a new node: /default-rack/127.0.0.1:9866
2022-11-06 16:01:34,457 INFO mapreduce.JobSubmitter: number of splits:26
2022-11-06 16:01:34,600 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1667431627519_0010
2022-11-06 16:01:34,600 INFO mapreduce.JobSubmitter: Executing with tokens: []
2022-11-06 16:01:34,712 INFO conf.Configuration: resource-types.xml not found
2022-11-06 16:01:34,713 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2022-11-06 16:01:34,763 INFO impl.YarnClientImpl: Submitted application application_1667431627519_0010
2022-11-06 16:01:34,788 INFO mapreduce.Job: The url to track the job: http://assignment1-ankale-main.js2local:8088/proxy/application_1667431627519_0010/
2022-11-06 16:01:34,789 INFO mapreduce.Job: Running job: job_1667431627519_0010
2022-11-06 16:01:39,843 INFO mapreduce.Job: Job job_1667431627519_0010 running in uber mode : false
2022-11-06 16:01:39,843 INFO mapreduce.Job:  map 0% reduce 0%
2022-11-06 16:01:49,926 INFO mapreduce.Job:  map 8% reduce 0%
2022-11-06 16:01:57,971 INFO mapreduce.Job:  map 14% reduce 0%
2022-11-06 16:02:03,998 INFO mapreduce.Job:  map 17% reduce 0%
2022-11-06 16:02:05,005 INFO mapreduce.Job:  map 19% reduce 0%
2022-11-06 16:02:06,013 INFO mapreduce.Job:  map 21% reduce 0%
2022-11-06 16:02:11,046 INFO mapreduce.Job:  map 22% reduce 0%
2022-11-06 16:02:12,050 INFO mapreduce.Job:  map 26% reduce 0%
2022-11-06 16:02:13,054 INFO mapreduce.Job:  map 30% reduce 0%
2022-11-06 16:02:16,069 INFO mapreduce.Job:  map 32% reduce 0%
2022-11-06 16:02:18,081 INFO mapreduce.Job:  map 37% reduce 0%
2022-11-06 16:02:20,089 INFO mapreduce.Job:  map 41% reduce 0%
2022-11-06 16:02:21,095 INFO mapreduce.Job:  map 42% reduce 0%
2022-11-06 16:02:22,103 INFO mapreduce.Job:  map 44% reduce 0%
2022-11-06 16:02:25,141 INFO mapreduce.Job:  map 48% reduce 0%
2022-11-06 16:02:26,147 INFO mapreduce.Job:  map 52% reduce 0%
2022-11-06 16:02:28,157 INFO mapreduce.Job:  map 56% reduce 0%
2022-11-06 16:02:30,167 INFO mapreduce.Job:  map 60% reduce 0%
2022-11-06 16:02:31,171 INFO mapreduce.Job:  map 60% reduce 19%
2022-11-06 16:02:32,174 INFO mapreduce.Job:  map 64% reduce 19%
2022-11-06 16:02:33,179 INFO mapreduce.Job:  map 67% reduce 19%
2022-11-06 16:02:34,183 INFO mapreduce.Job:  map 68% reduce 19%
2022-11-06 16:02:36,204 INFO mapreduce.Job:  map 75% reduce 19%
2022-11-06 16:02:37,211 INFO mapreduce.Job:  map 75% reduce 23%
2022-11-06 16:02:38,215 INFO mapreduce.Job:  map 79% reduce 23%
2022-11-06 16:02:43,239 INFO mapreduce.Job:  map 79% reduce 26%
2022-11-06 16:02:48,273 INFO mapreduce.Job:  map 81% reduce 26%
2022-11-06 16:02:51,283 INFO mapreduce.Job:  map 83% reduce 26%
2022-11-06 16:02:53,295 INFO mapreduce.Job:  map 84% reduce 26%
2022-11-06 16:02:54,300 INFO mapreduce.Job:  map 87% reduce 26%
2022-11-06 16:02:57,311 INFO mapreduce.Job:  map 88% reduce 26%
```

c. Show output

```
hadoopuser@assignment1-ankale-main:~$ hdfs dfs -cat /access
00      66.249.66.194   14298
00      66.249.66.91    12232
00      66.249.66.92    4291
01      66.249.66.91    13874
01      66.249.66.194   12485
01      66.249.66.92    2924
02      66.249.66.91    11697
02      66.249.66.194   10345
02      17.58.102.43    898
03      66.249.66.194   8644
03      66.249.66.91    7914
03      17.58.102.43    958
04      66.249.66.194   10805
04      66.249.66.91    7571
04      23.101.169.3    1081
05      66.249.66.194   10534
05      66.249.66.91    7035
05      23.101.169.3    1352
06      66.249.66.194   10283
06      66.249.66.91    7968
06      23.101.169.3    1674
07      66.249.66.194   12267
07      66.249.66.91    9116
07      23.101.169.3    1449
08      66.249.66.194   12964
08      66.249.66.91    10237
08      151.239.241.163 6256
09      66.249.66.194   14833
09      66.249.66.91    11450
09      151.239.241.163 9169
10      66.249.66.194   17292
10      66.249.66.91    13213
10      151.239.241.163 9824
11      66.249.66.194   15572
11      66.249.66.91    13631
11      151.239.241.163 8642
12      66.249.66.194   16966
12      66.249.66.91    12656
12      151.239.241.163 8564
13      66.249.66.194   18372
13      66.249.66.91    16166
13      151.239.241.163 7881
14      66.249.66.194   19249
14      66.249.66.91    17893
14      151.239.241.163 8786
15      66.249.66.194   18273
15      66.249.66.91    16662
15      151.239.241.163 6558
16      66.249.66.91    17849
16      66.249.66.194   17512
16      151.239.241.163 7187
17      66.249.66.194   18954
17      66.249.66.91    17107
17      151.239.241.163 8571
18      66.249.66.194   17531
18      66.249.66.91    16727
18      104.222.32.91   7159
19      66.249.66.91    18911
19      66.249.66.194   18569
19      104.222.32.91   9876
20      66.249.66.91    15834
20      66.249.66.194   15729
20      66.249.66.92    5589
21      66.249.66.194   14075
21      66.249.66.91    13783
21      66.249.66.92    4552
22      66.249.66.91    14094
22      66.249.66.194   13576
22      66.249.66.92    4901
23      66.249.66.194   14355
23      66.249.66.91    10902
23      66.249.66.92    4259
hadoopuser@assignment1-ankale-main:~$
```

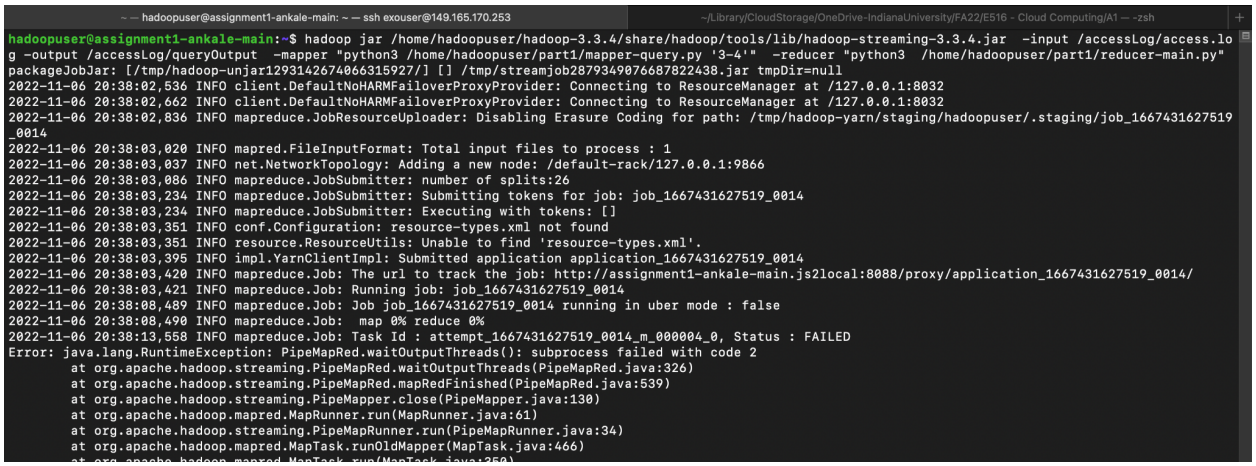
Part 2

2.1 MapReduce like a query

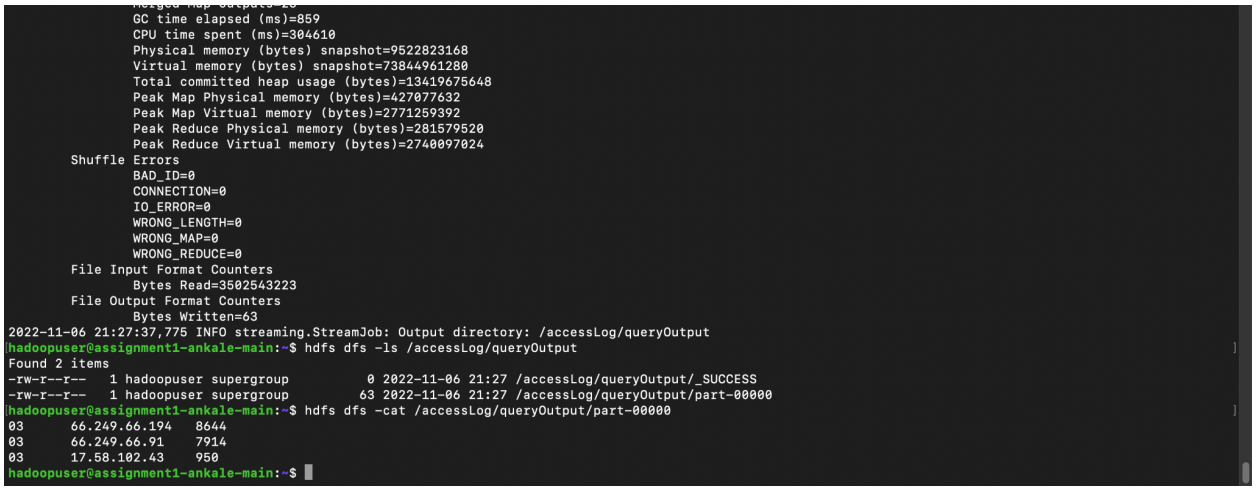
Query the MapReduce run using command line arguments. User input ‘x-y’ will output top 3 IPs between x and y hours (24 hr format).
Only change from Part1 is the mapper which accepts command line inputs.

Example 1: arg = ’3-4’

a. Run command

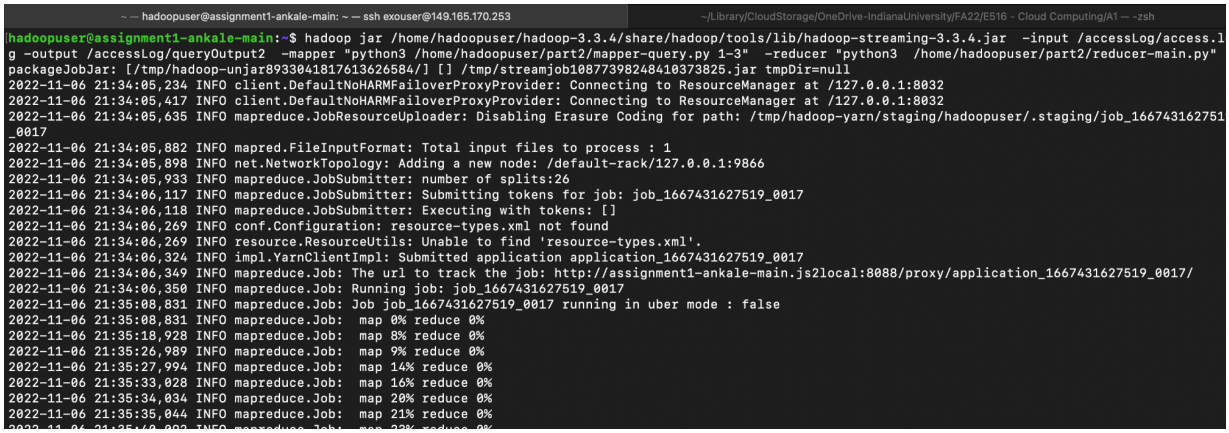


b. Show output:

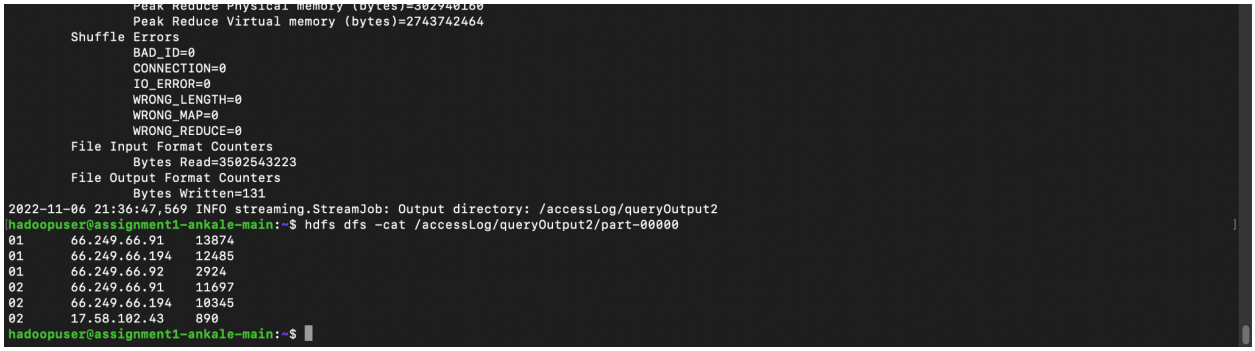


Example 2: arg = ’1-3’

a. Run command

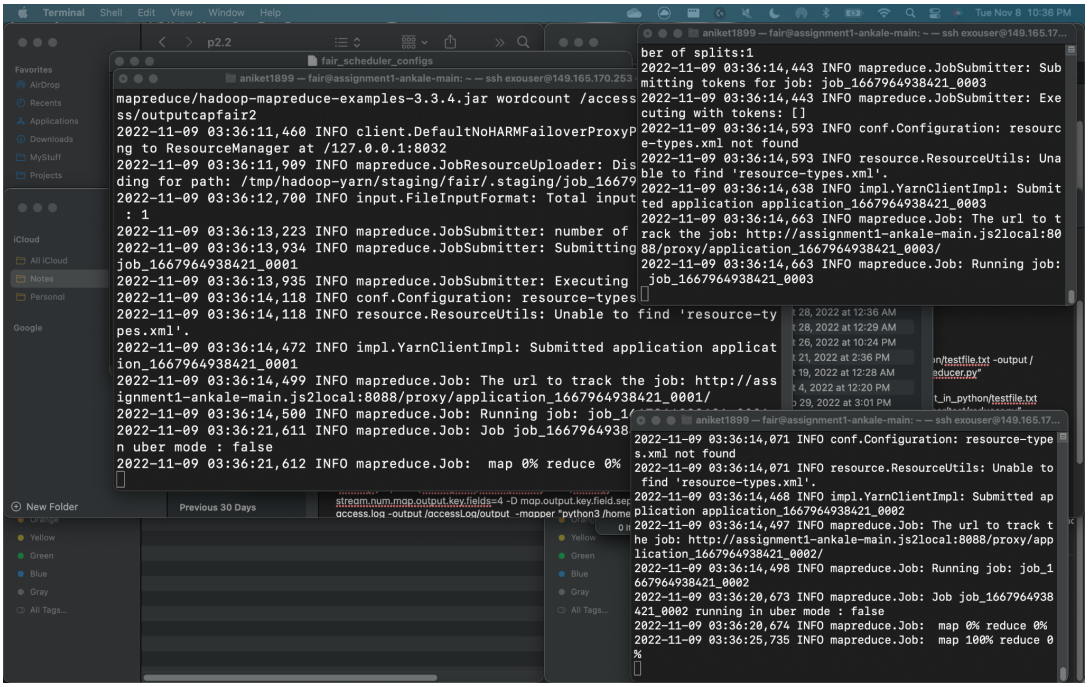


b. Show output:

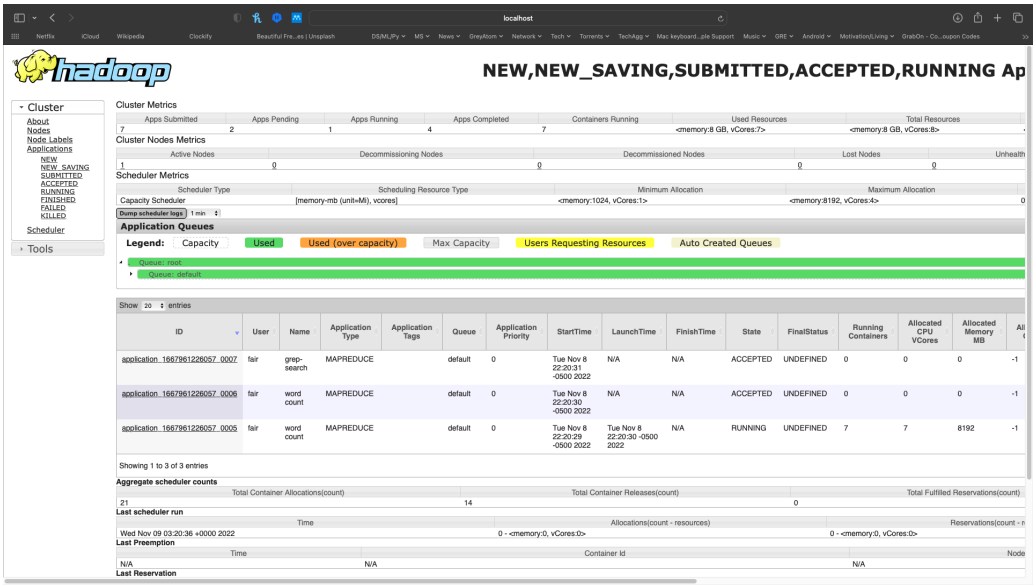


2.2 Fair and capacity scheduler

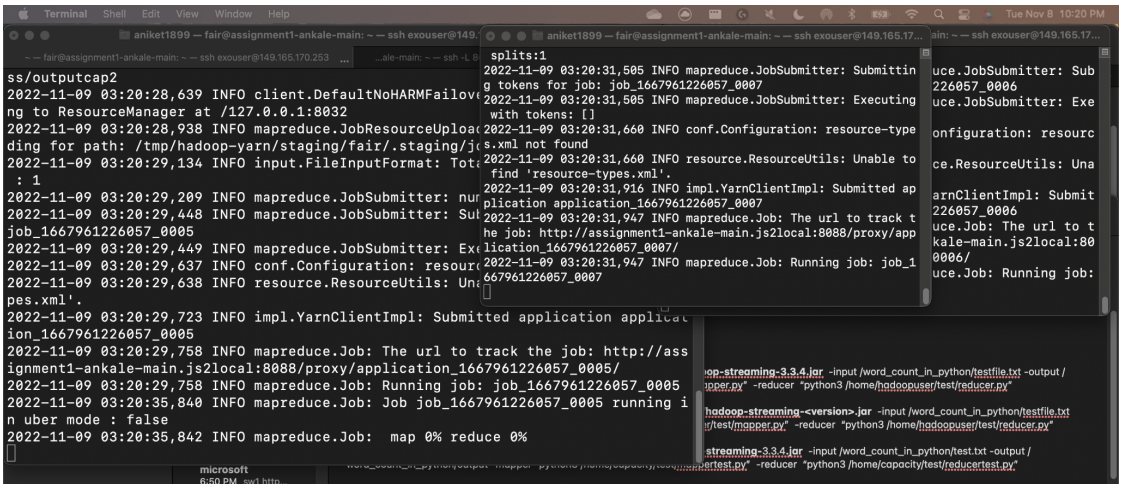
a. Capacity scheduler
   i. Change configs for queues over capacity scheduler
   ii. Run stop-all.sh and start-all.sh
   iii. Refresh queues using
   > yarn rmadmin -refreshQueues
   iv. Run 3 jobs (wordcount on access.log, wordcount on sample.log and grep on sample.log) simultaneously:



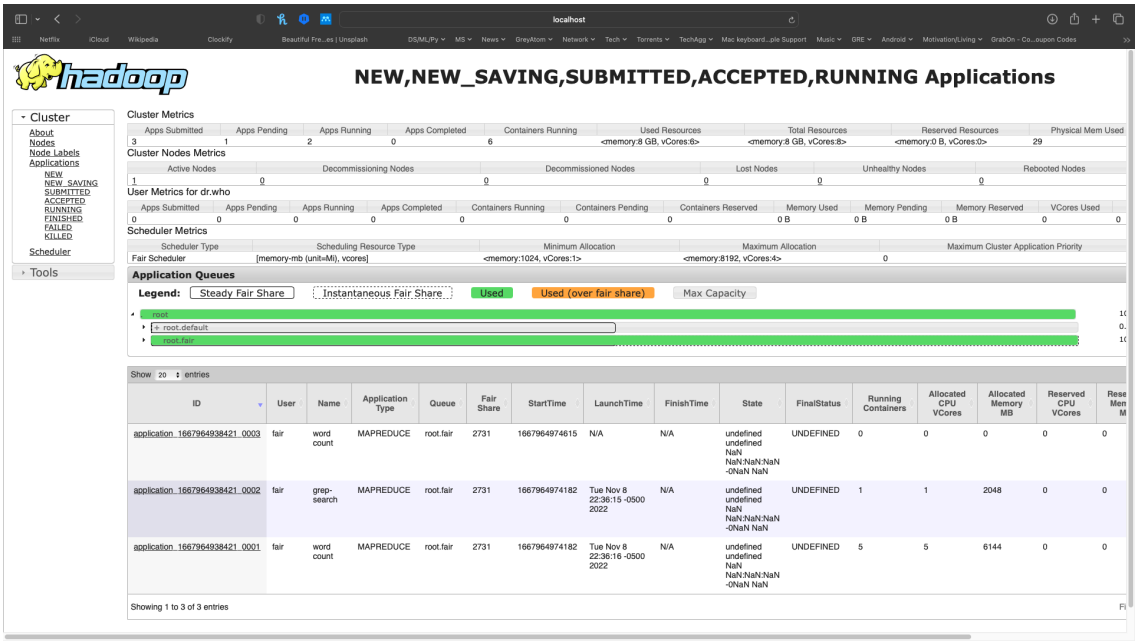
   v. Schedules:



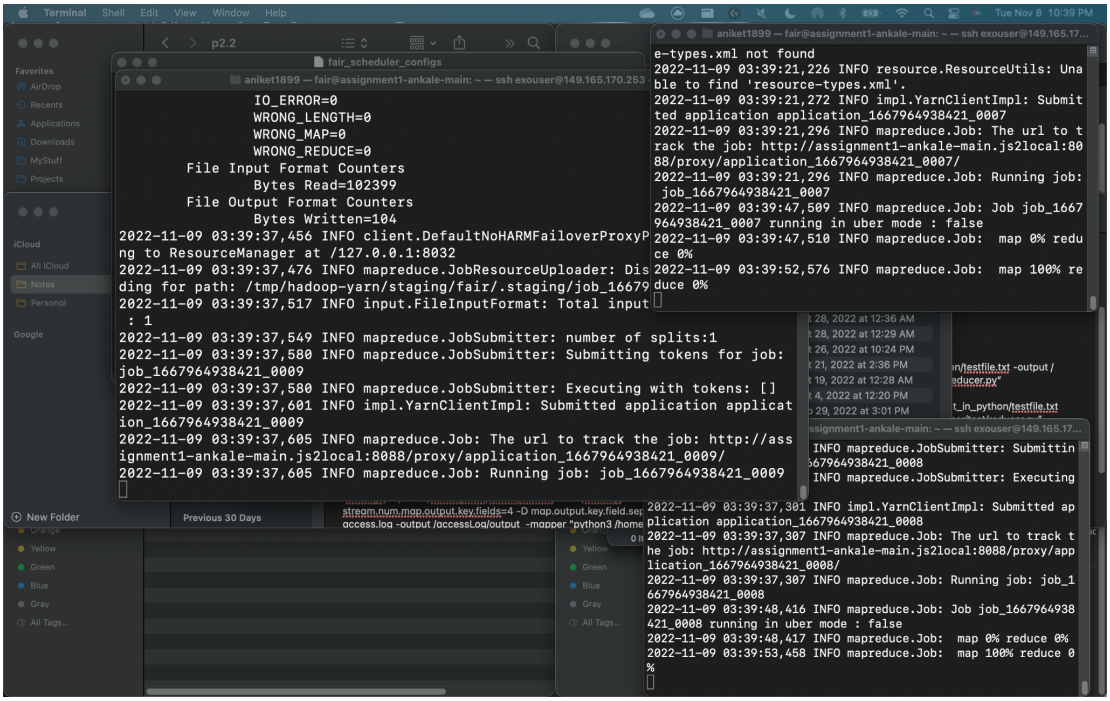
b. Fair share scheduler:
   i. Change yarn-site.xml config
   ii. Run stop-all.sh and start-all.sh and check for “fair share” in localhost:8088 port
   iii. Run 3 commands (wordcount on access.log, wordcount on sample.log and grep on sample.log) simultaneously:



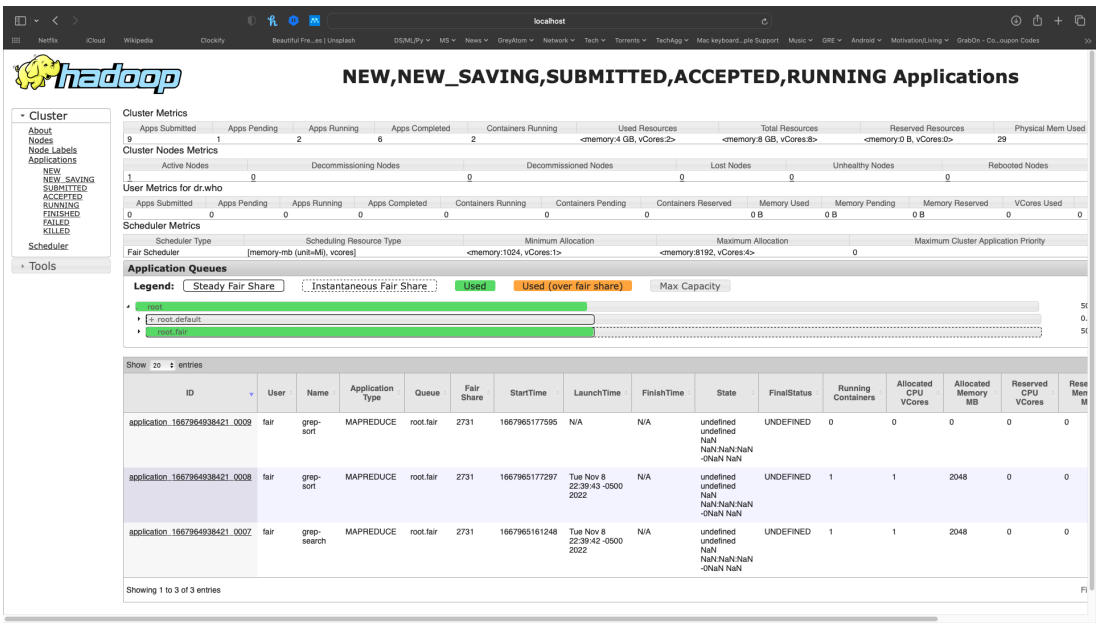
   iv. Schedules:

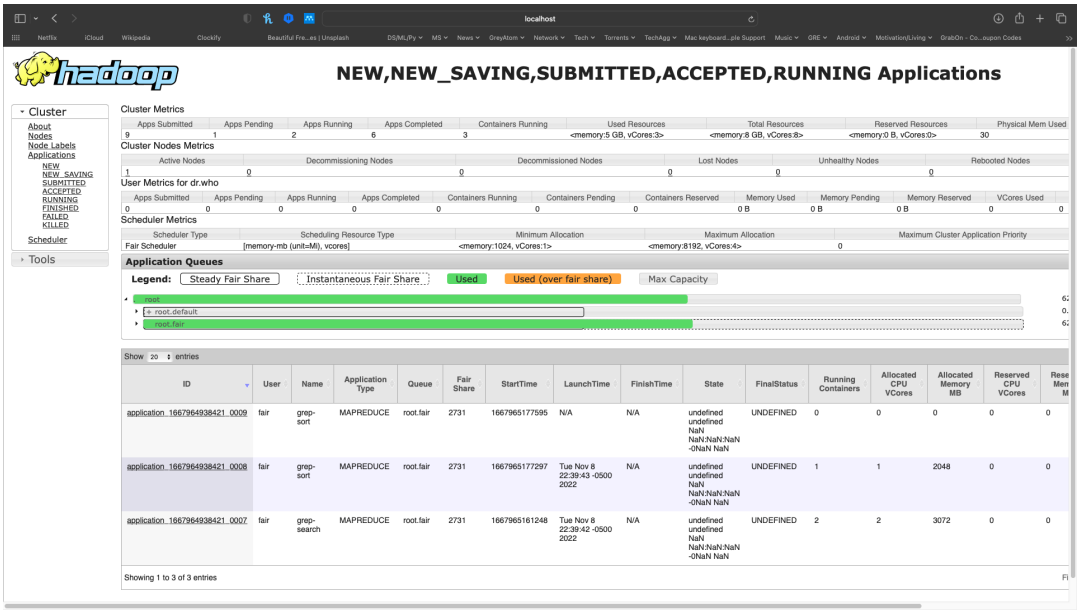


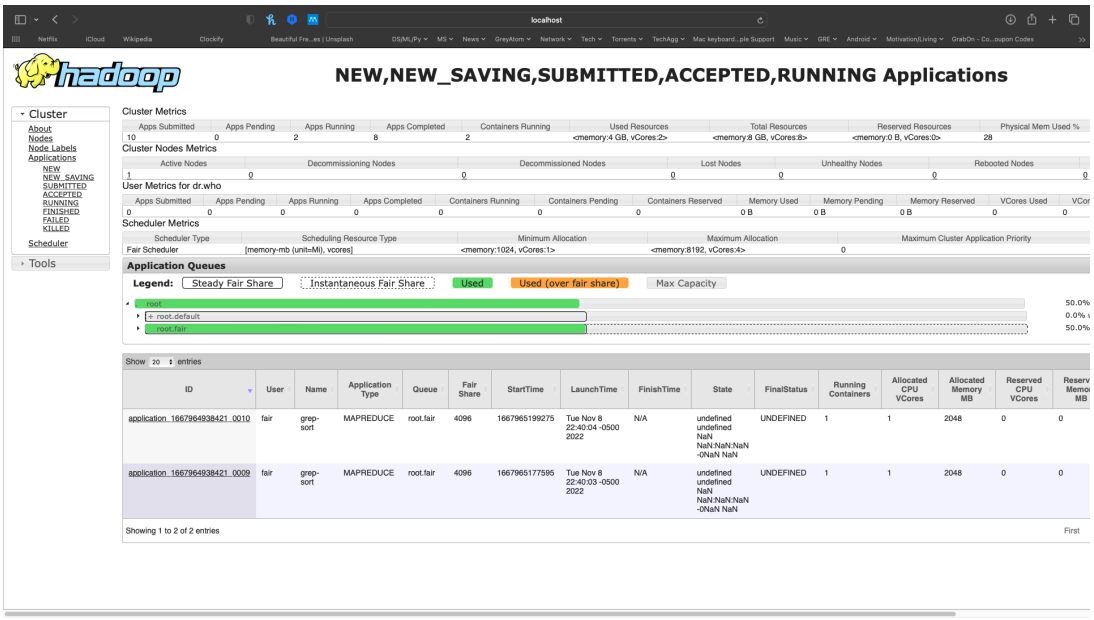
v. Another experiment: Run 3 jobs (grep on sample.log x3) simultaneously



vi. Schedules







Comparison of Capacity v/s Fair scheduler:

The Capacity scheduler schedules jobs in queues. But the fair scheduler gives priority to the first job then allocates resources to the second and then third job according to the requirements. In the fair share example 1, the highest priority was given to the most resource intensive job, followed by the other. This can also be seen in example 2 for fair share, where resources were almost equally shared for scheduled jobs of the same type (i.e. grep).

The Jobs run:

Wordcount 1 on sample.log
hadoop jar /home/fair/hadoop-3.3.4/share/hadoop/mapreduce/hadoop-mapreduce-examples-3.3.4.jar wordcount /sample/sample.log /sample/output

Wordcount 1 on access.log
hadoop jar /home/fair/hadoop-3.3.4/share/hadoop/mapreduce/hadoop-mapreduce-examples-3.3.4.jar wordcount /access/access.log /access/output

Grep on sample.log
hadoop jar /home/fair/hadoop-3.3.4/share/hadoop/mapreduce/hadoop-mapreduce-examples-3.3.4.jar grep /sample/sample.log /sample/outputGrepfair33 '[:*]'*

Bonus:

I have run all my codes in the Jetstream instance and the access.log files takes less than 2 minutes to complete. Refer screenshots above.