ASSIGNMENT-1
AMOGH HUILGOL
CS6240- SECTION 02

For each of the versions of your sequential and multithreaded program detailed in B and C, report the minimum, average, and maximum running time observed over the 10 runs. (5 points)

SEQUENCE WITHOUT FIB()

```
Maximum execution time for 10.0 rounds of program execution is 5.918 seconds
Minimum execution time for 10.0 rounds of program execution is 2.724 seconds
Average execution time for 10.0 rounds of program execution is 3.4140000000000006 seconds
```

SEQUENCE WITH FIB()

```
Maximum execution time for 10.0 rounds of program execution is 16.042 seconds
Minimum execution time for 10.0 rounds of program execution is 13.772 seconds
Average execution time for 10.0 rounds of program execution is 14.065800000000001 seconds
```

No LOCK WITHOUT FIB()

```
Maximum execution time for 10 rounds of program execution is 2.863 seconds
Minimum execution time for 10 rounds of program execution is 1.419 seconds
Average execution time for 10 rounds of program execution is 1.6292000000000002 seconds
```

NO_LOCK WITH FIB()

```
Maximum execution time for 10 rounds of program execution is 11.809 seconds
Minimum execution time for 10 rounds of program execution is 11.021 seconds
Average execution time for 10 rounds of program execution is 11.3602 seconds
```

Coarse Lock Without FIB()

```
Maximum execution time for 10 rounds of program execution is 2.698 seconds
Minimum execution time for 10 rounds of program execution is 1.468 seconds
Average execution time for 10 rounds of program execution is 1.6396000000000002 seconds
```

Coarse Lock With FIB()

```
Maximum execution time for 10 rounds of program execution is 28.951 seconds
Minimum execution time for 10 rounds of program execution is 24.61 seconds
Average execution time for 10 rounds of program execution is 26.6085 seconds
```

Fine Lock Without FIB()

```
Maximum execution time for 10.0 rounds of program execution is 2.792 seconds
Minimum execution time for 10.0 rounds of program execution is 1.58 seconds
Average execution time for 10.0 rounds of program execution is 1.8167000000000002 seconds
```

FINE LOCK WITH FIB()

```
Maximum execution time for 10.0 rounds of program execution is 13.062 seconds
Minimum execution time for 10.0 rounds of program execution is 10.848 seconds
Average execution time for 10.0 rounds of program execution is 12.046600000000002 seconds
```

No Shared WITHOUT FIB()

```
Maximum execution time for 10 rounds of program execution is 4.385 seconds
Minimum execution time for 10 rounds of program execution is 1.431 seconds
Average execution time for 10 rounds of program execution is 1.7981000000000003 seconds
```

NO SHARED WITH FIB()

```
Maximum execution time for 10 rounds of program execution is 8.07 seconds
Minimum execution time for 10 rounds of program execution is 5.99 seconds
Average execution time for 10 rounds of program execution is 6.775499999999999 seconds
```

1) **Report the number of worker threads used and the speedup of the multithreaded versions based on the corresponding average running times. (5 points)**

The number of worker threads spawned are 4. the speedup obtained is as follows

NO_LOCK :
PROGRAM B : 2.09
PROGRAM C : 1.23

COARSE_LOCK :
PROGRAM B : 2.08
PROGRAM C : 0.528

FINE_LOCK :
PROGRAM B : 1.879
PROGRAM C : 1.16

NO_SHARED :
PROGRAM B : 1.898
PROGRAM C : 2.07

3) Answer the following questions in a brief and concise manner: (4 points each)

i) Which program version (SEQ, NO-LOCK, COARSE-LOCK, FINE-LOCK, NO-SHARING) would you normally expect to finish fastest and why? Do the experiments confirm your expectation? If not, try to explain the reasons.

Answer) NO_LOCKING will run the fastest as compared to the other versions because

a) Since there are no locks, we can utilize maximum parallelism and no worker will have to wait for resource
b) All workers will update the resource at the same time. hence there is a chance for values to be inconsistent
c) The average values confirm with the expectation

ii) Which program version (SEQ, NO-LOCK, COARSE-LOCK, FINE-LOCK, NO-SHARING) would you normally expect to finish slowest and why? Do the experiments confirm your expectation? If not, try to explain the reasons.

Answer) a) Sequential is generally expected to finish the slowest since each step is executed one after the other . In parallel approach , program makes use of multiple cores available in the system. Hence parallel programs usually finish faster.
b) Yes the experiments confirm with expectation

iii) Compare the temperature averages returned by each program version. Report if any of them is incorrect or if any of the programs crashed because of concurrent accesses.

Answer) The effects of parallelism are better felt when input size is large . However we can see from our experiments that the average values for NO_LOCK mechanism is wrong or inconsistent . This could occur due to

a) Program crashing because there is lack of synchronization between threads which leads to one thread trying to read a value in Hashmap which is not present . this results in code throwing null pointer exception
b) Since there is no lock involved , multiple threads may update a data structure based on old record leading to overwriting of one record over other . this leads to inconsistency or wrong results.

iv) Compare the running times of SEQ and COARSE-LOCK. Try to explain why one is slower than the other. (Make sure to consider the results of both B and C—this might support or refute a possible hypothesis.)

Answer) Sequential is generally slower than coarse lock mechanism . However we have seen that when delay fib(17) is introduced , sequential program performs faster . the possible reasons could be :

a) When inputs are small , the tasks are performed quickly with less transfer of locks between threads . Hence parallelism is utilized to speed up the computation

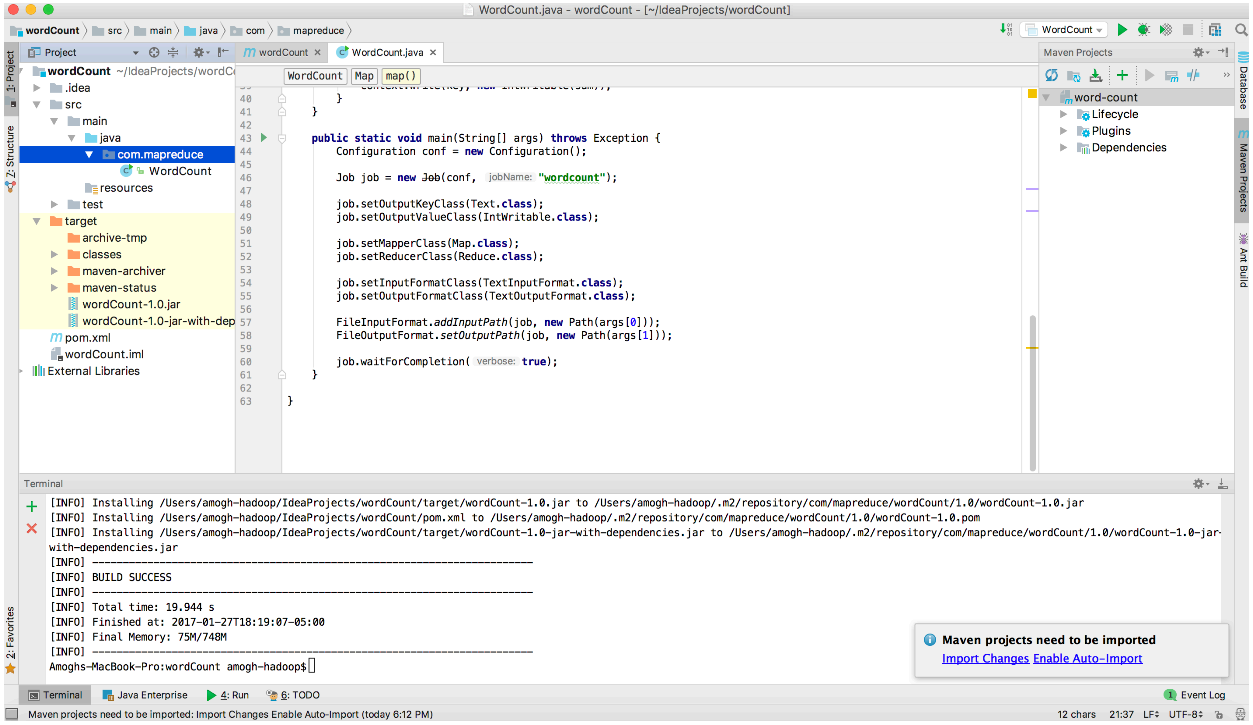
b) However as the inputs increase , the wait time for threads increases as they have to wait more often for other thread to release locks . This defeats the improvement in performance we obtain by running a program parallel

v) How does the higher computation cost in part C (additional Fibonacci computation) affect the difference between COARSE-LOCK and FINE-LOCK? Try to explain the reason.

Answer) In COARSE_LOCK , the lock is held over the entire data structures and this ensures that no other thread is trying to update the hashmap. thus it can be guaranteed that no two Fibonacci computations will happen in parallel. Thus all delays will happen in sequential way. However for FINE_LOCK there is a chance that some Fibonacci computations may happen in parallel since each thread holds lock over the variable which it updates (not the whole hashmap) . Hence with higher computation cost , the fine lock performs better than coarse lock

WORD COUNT LOCAL EXECUTION

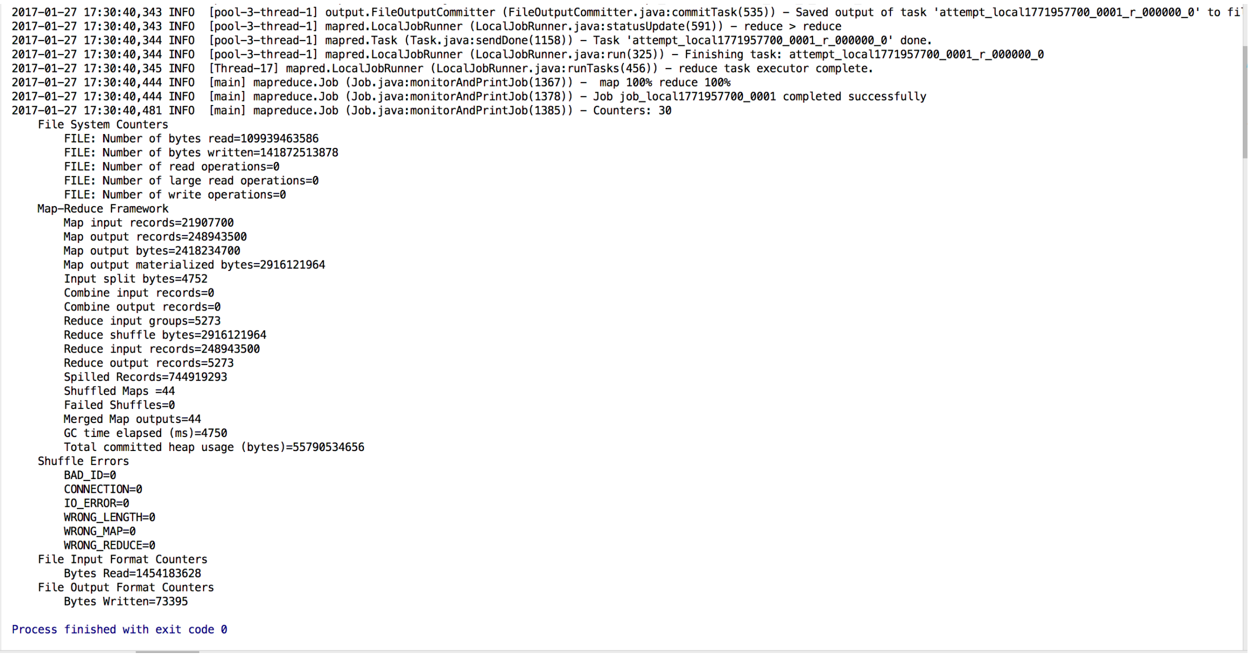
Screenshot showing the directory structure



```
        }
    }

    public static void main(String[] args) throws Exception {
        Configuration conf = new Configuration();

        Job job = new ~~Job~~(conf,  jobName: "wordcount");

        job.setOutputKeyClass(Text.class);
        job.setOutputValueClass(IntWritable.class);

        job.setMapperClass(Map.class);
        job.setReducerClass(Reduce.class);

        job.setInputFormatClass(TextInputFormat.class);
        job.setOutputFormatClass(TextOutputFormat.class);

        FileInputFormat.addInputPath(job, new Path(args[0]));
        FileOutputFormat.setOutputPath(job, new Path(args[1]));

        job.waitForCompletion( verbose: true);
    }

}
```

```
[INFO] Installing /Users/amogh-hadoop/IdeaProjects/wordCount/target/wordCount-1.0.jar to /Users/amogh-hadoop/.m2/repository/com/mapreduce/wordCount/1.0/wordCount-1.0.jar
[INFO] Installing /Users/amogh-hadoop/IdeaProjects/wordCount/pom.xml to /Users/amogh-hadoop/.m2/repository/com/mapreduce/wordCount/1.0/wordCount-1.0.pom
[INFO] Installing /Users/amogh-hadoop/IdeaProjects/wordCount/target/wordCount-1.0-jar-with-dependencies.jar to /Users/amogh-hadoop/.m2/repository/com/mapreduce/wordCount/1.0/wordCount-1.0-jar-
with-dependencies.jar
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time: 19.944 s
[INFO] Finished at: 2017-01-27T18:19:07-05:00
[INFO] Final Memory: 75M/748M
[INFO] ------------------------------------------------------------------------
Amoghs-MacBook-Pro:wordCount amogh-hadoop$
```

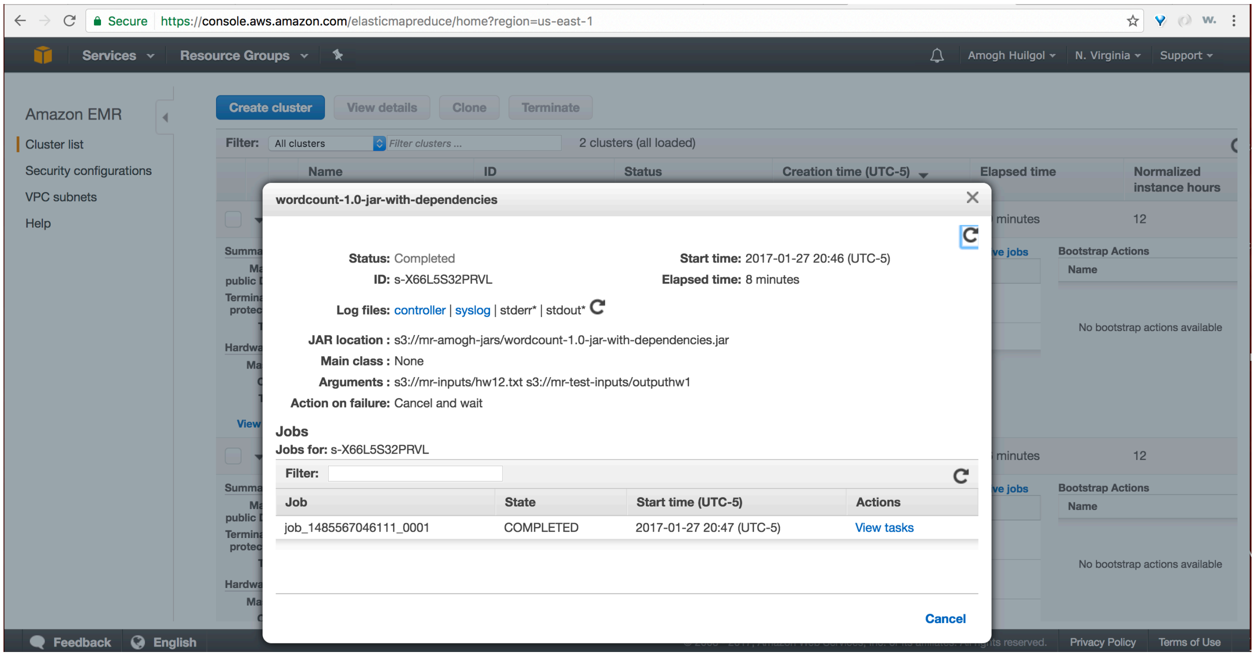
Screen shot showing last 20 lines of Hadoop execution in IDE



```
2017-01-27 17:30:40,343 INFO  [pool-3-thread-1] output.FileOutputCommitter (FileOutputCommitter.java:commitTask(535)) - Saved output of task 'attempt_local1771957700_0001_r_000000_0' to fi
2017-01-27 17:30:40,343 INFO  [pool-3-thread-1] mapred.LocalJobRunner (LocalJobRunner.java:statusUpdate(591)) - reduce > reduce
2017-01-27 17:30:40,344 INFO  [pool-3-thread-1] mapred.Task (Task.java:sendDone(1158)) - Task 'attempt_local1771957700_0001_r_000000_0' done.
2017-01-27 17:30:40,344 INFO  [pool-3-thread-1] mapred.LocalJobRunner (LocalJobRunner.java:run(325)) - Finishing task: attempt_local1771957700_0001_r_000000_0
2017-01-27 17:30:40,345 INFO  [Thread-17] mapred.LocalJobRunner (LocalJobRunner.java:runTasks(456)) - reduce task executor complete.
2017-01-27 17:30:40,444 INFO  [main] mapreduce.Job (Job.java:monitorAndPrintJob(1367)) -  map 100% reduce 100%
2017-01-27 17:30:40,444 INFO  [main] mapreduce.Job (Job.java:monitorAndPrintJob(1378)) - Job job_local1771957700_0001 completed successfully
2017-01-27 17:30:40,481 INFO  [main] mapreduce.Job (Job.java:monitorAndPrintJob(1385)) - Counters: 30
	File System Counters
		FILE: Number of bytes read=109939463586
		FILE: Number of bytes written=141872513878
		FILE: Number of read operations=0
		FILE: Number of large read operations=0
		FILE: Number of write operations=0
	Map-Reduce Framework
		Map input records=21907700
		Map output records=248943500
		Map output bytes=2418234700
		Map output materialized bytes=2916121964
		Input split bytes=4752
		Combine input records=0
		Combine output records=0
		Reduce input groups=5273
		Reduce shuffle bytes=2916121964
		Reduce input records=248943500
		Reduce output records=5273
		Spilled Records=744919293
		Shuffled Maps =44
		Failed Shuffles=0
		Merged Map outputs=44
		GC time elapsed (ms)=4750
		Total committed heap usage (bytes)=55790534656
	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=1454183628
	File Output Format Counters 
		Bytes Written=73395

Process finished with exit code 0
```

EXECUTION ON AWS

Screenshot showing successful execution of word count in AWS-EMR



wordcount-1.0-jar-with-dependencies

Status: Completed
ID: s-X66L5S32PRVL
Start time: 2017-01-27 20:46 (UTC-5)
Elapsed time: 8 minutes

Log files: controller | syslog | stderr* | stdout*

JAR location : s3://mr-amogh-jars/wordcount-1.0-jar-with-dependencies.jar
Main class : None
Arguments : s3://mr-inputs/hw12.txt s3://mr-test-inputs/outputhw1
Action on failure: Cancel and wait

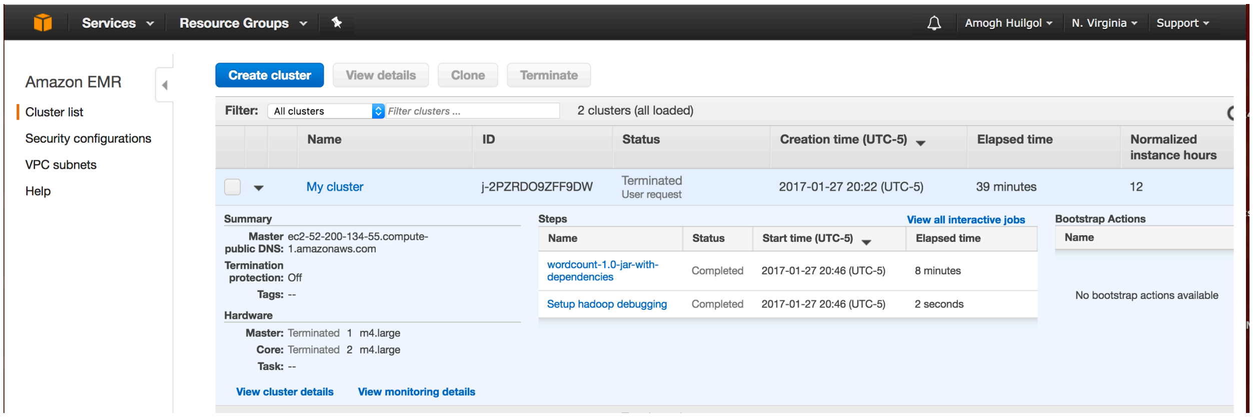
Jobs
Jobs for: s-X66L5S32PRVL

Filter:

Job	State	Start time (UTC-5)	Actions
job_1485567046111_0001	COMPLETED	2017-01-27 20:47 (UTC-5)	View tasks

Cancel

Screenshot showing cluster details



Filter: All clusters | Filter clusters ... 2 clusters (all loaded)

Name	ID	Status	Creation time (UTC-5)	Elapsed time	Normalized instance hours
My cluster	j-2PZRDO9ZFF9DW	Terminated User request	2017-01-27 20:22 (UTC-5)	39 minutes	12

Summary

Master public DNS: ec2-52-200-134-55.compute-1.amazonaws.com

Termination protection: Off

Tags: --

Hardware

Master: Terminated 1 m4.large

Core: Terminated 2 m4.large

Task: --

View cluster details   View monitoring details

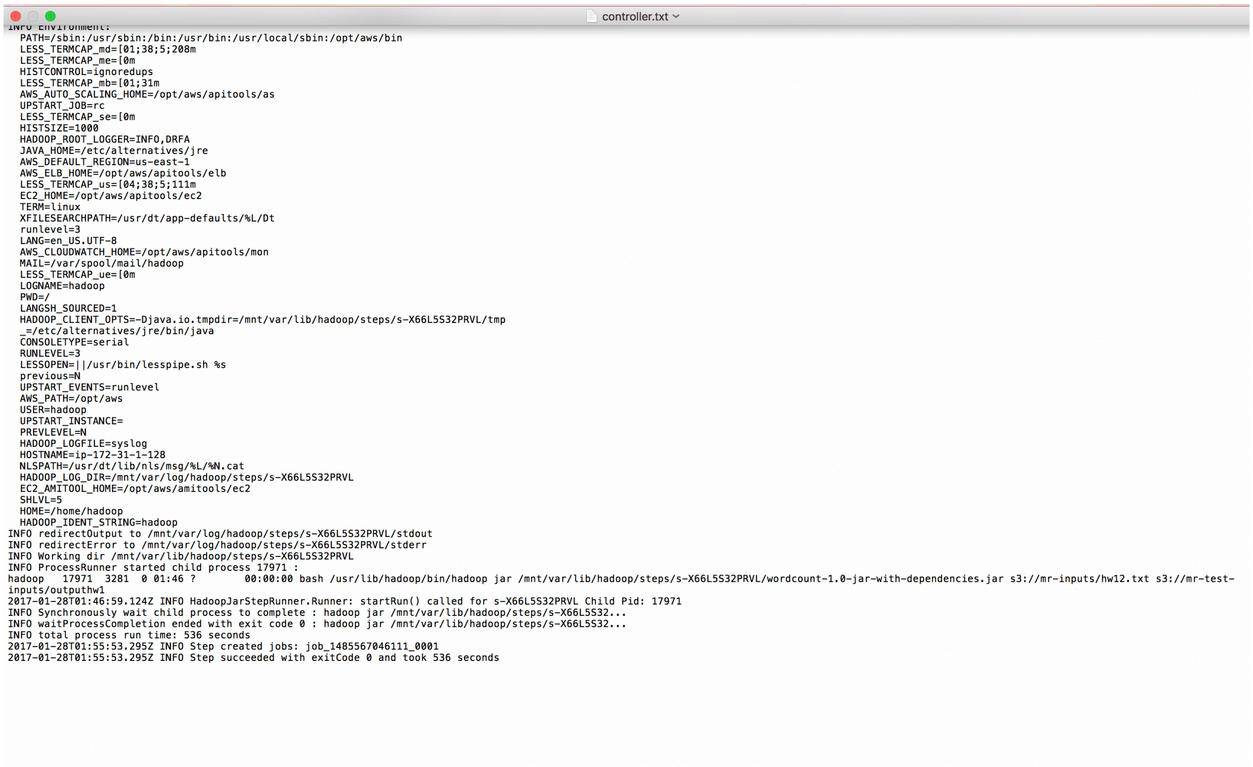
Steps   View all interactive jobs

Name	Status	Start time (UTC-5)	Elapsed time
wordcount-1.0-jar-with-dependencies	Completed	2017-01-27 20:46 (UTC-5)	8 minutes
Setup hadoop debugging	Completed	2017-01-27 20:46 (UTC-5)	2 seconds

Bootstrap Actions

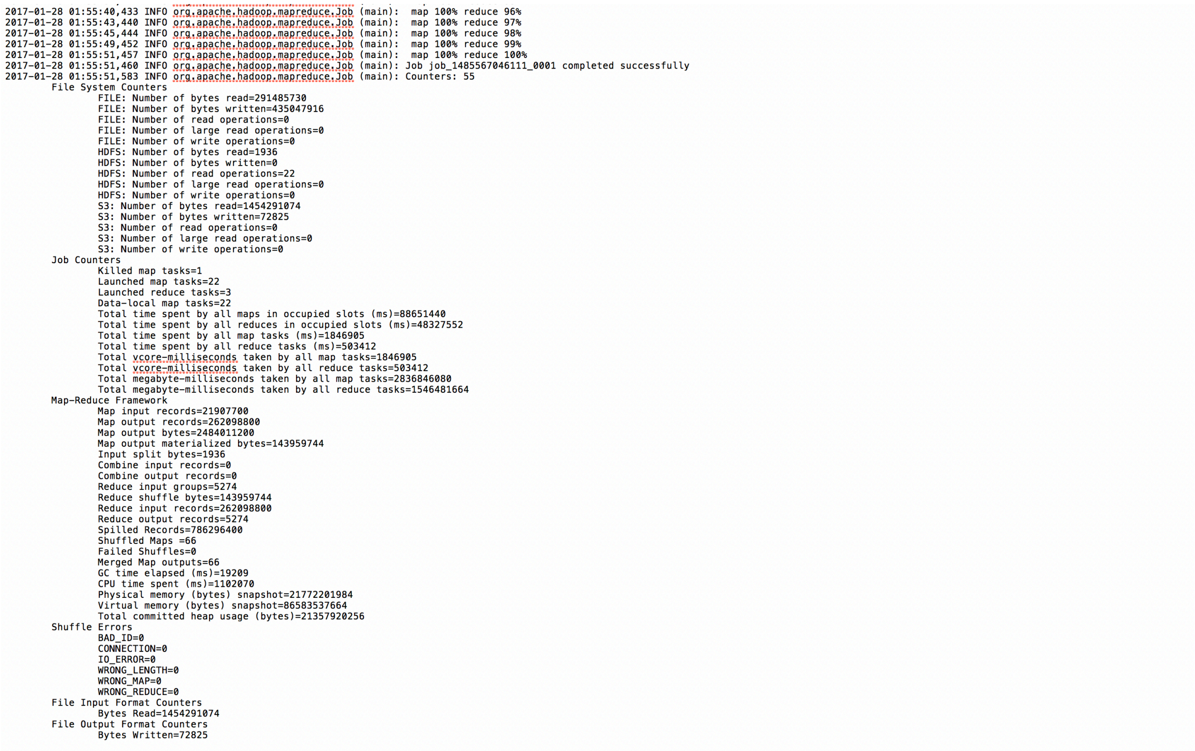
Name
No bootstrap actions available

CONTROLLER.txt



```
INFO Environment:
  PATH=/sbin:/usr/sbin:/bin:/usr/bin:/usr/local/sbin:/opt/aws/bin
  LESS_TERMCAP_md=[01;38;5;208m
  LESS_TERMCAP_me=[0m
  HISTCONTROL=ignoredups
  LESS_TERMCAP_mb=[01;31m
  AWS_AUTO_SCALING_HOME=/opt/aws/apitools/as
  UPSTART_JOB=rc
  LESS_TERMCAP_se=[0m
  HISTSIZE=1000
  HADOOP_ROOT_LOGGER=INFO,DRFA
  JAVA_HOME=/etc/alternatives/jre
  AWS_DEFAULT_REGION=us-east-1
  AWS_ELB_HOME=/opt/aws/apitools/elb
  LESS_TERMCAP_us=[04;38;5;111m
  EC2_HOME=/opt/aws/apitools/ec2
  TERM=linux
  XFILESEARCHPATH=/usr/dt/app-defaults/%L/Dt
  runlevel=3
  LANG=en_US.UTF-8
  AWS_CLOUDWATCH_HOME=/opt/aws/apitools/mon
  MAIL=/var/spool/mail/hadoop
  LESS_TERMCAP_ue=[0m
  LOGNAME=hadoop
  PWD=/
  LANGSH_SOURCED=1
  HADOOP_CLIENT_OPTS=-Djava.io.tmpdir=/mnt/var/lib/hadoop/steps/s-X66L5S32PRVL/tmp
  _=/etc/alternatives/jre/bin/java
  CONSOLETYPE=serial
  RUNLEVEL=3
  LESSOPEN=||/usr/bin/lesspipe.sh %s
  previous=N
  UPSTART_EVENTS=runlevel
  AWS_PATH=/opt/aws
  USER=hadoop
  UPSTART_INSTANCE=
  PREVLEVEL=N
  HADOOP_LOGFILE=syslog
  HOSTNAME=ip-172-31-1-128
  NLSPATH=/usr/dt/lib/nls/msg/%L/%N.cat
  HADOOP_LOG_DIR=/mnt/var/log/hadoop/steps/s-X66L5S32PRVL
  EC2_AMITOOL_HOME=/opt/aws/amitools/ec2
  SHLVL=5
  HOME=/home/hadoop
  HADOOP_IDENT_STRING=hadoop
INFO redirectOutput to /mnt/var/log/hadoop/steps/s-X66L5S32PRVL/stdout
INFO redirectError to /mnt/var/log/hadoop/steps/s-X66L5S32PRVL/stderr
INFO Working dir /mnt/var/lib/hadoop/steps/s-X66L5S32PRVL
INFO ProcessRunner started child process 17971 :
hadoop   17971  3281  0 01:46 ?        00:00:00 bash /usr/lib/hadoop/bin/hadoop jar /mnt/var/lib/hadoop/steps/s-X66L5S32PRVL/wordcount-1.0-jar-with-dependencies.jar s3://mr-inputs/hw12.txt s3://mr-test-
inputs/outputhw1
2017-01-28T01:46:59.124Z INFO HadoopJarStepRunner.Runner: startRun() called for s-X66L5S32PRVL Child Pid: 17971
INFO Synchronously wait child process to complete : hadoop jar /mnt/var/lib/hadoop/steps/s-X66L5S32...
INFO waitProcessCompletion ended with exit code 0 : hadoop jar /mnt/var/lib/hadoop/steps/s-X66L5S32...
INFO total process run time: 536 seconds
2017-01-28T01:55:53.295Z INFO Step created jobs: job_1485567046111_0001
2017-01-28T01:55:53.295Z INFO Step succeeded with exitCode 0 and took 536 seconds
```

SYSLOG.txt



```
2017-01-28 01:55:40,433 INFO org.apache.hadoop.mapreduce.Job (main):  map 100% reduce 96%
2017-01-28 01:55:43,440 INFO org.apache.hadoop.mapreduce.Job (main):  map 100% reduce 97%
2017-01-28 01:55:45,444 INFO org.apache.hadoop.mapreduce.Job (main):  map 100% reduce 98%
2017-01-28 01:55:49,452 INFO org.apache.hadoop.mapreduce.Job (main):  map 100% reduce 99%
2017-01-28 01:55:51,457 INFO org.apache.hadoop.mapreduce.Job (main):  map 100% reduce 100%
2017-01-28 01:55:51,460 INFO org.apache.hadoop.mapreduce.Job (main): Job job_1485567046111_0001 completed successfully
2017-01-28 01:55:51,583 INFO org.apache.hadoop.mapreduce.Job (main): Counters: 55
        File System Counters
                FILE: Number of bytes read=291485730
                FILE: Number of bytes written=435047916
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=1936
                HDFS: Number of bytes written=0
                HDFS: Number of read operations=22
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=0
                S3: Number of bytes read=1454291074
                S3: Number of bytes written=72825
                S3: Number of read operations=0
                S3: Number of large read operations=0
                S3: Number of write operations=0
        Job Counters
                Killed map tasks=1
                Launched map tasks=22
                Launched reduce tasks=3
                Data-local map tasks=22
                Total time spent by all maps in occupied slots (ms)=88651440
                Total time spent by all reduces in occupied slots (ms)=48327552
                Total time spent by all map tasks (ms)=1846905
                Total time spent by all reduce tasks (ms)=503412
                Total vcore-milliseconds taken by all map tasks=1846905
                Total vcore-milliseconds taken by all reduce tasks=503412
                Total megabyte-milliseconds taken by all map tasks=2836846080
                Total megabyte-milliseconds taken by all reduce tasks=1546481664
        Map-Reduce Framework
                Map input records=21907700
                Map output records=262098800
                Map output bytes=2484011200
                Map output materialized bytes=143959744
                Input split bytes=1936
                Combine input records=0
                Combine output records=0
                Reduce input groups=5274
                Reduce shuffle bytes=143959744
                Reduce input records=262098800
                Reduce output records=5274
                Spilled Records=786296400
                Shuffled Maps =66
                Failed Shuffles=0
                Merged Map outputs=66
                GC time elapsed (ms)=19209
                CPU time spent (ms)=1102070
                Physical memory (bytes) snapshot=21772201984
                Virtual memory (bytes) snapshot=86583537664
                Total committed heap usage (bytes)=21357920256
        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=1454291074
        File Output Format Counters
                Bytes Written=72825
```