CS644 1J1: INTRODUCTION TO BIGDATA, Fall-2020

PROJECT: Flight Data Analysis

Develop cloud-based Big Data workflows to process and analyze a large volume of flight data.

Team:

Anandan Dhanaraj

Jonathan Vidal

Parvathy Neelakanta Sarma

Introduction

In this project, we have analyzed 3 years of flight data. We have configured Hadoop in fully distributed mode.
Furthermore, we have developed oozie workflow for that to solve following 3 problems:

a. The 3 airlines with the highest and lowest probability, respectively, for being on schedule

b. The 3 airports with the longest and shortest average taxi time per flight (both in and out), respectively.

c. The most common reason for flight cancellations.

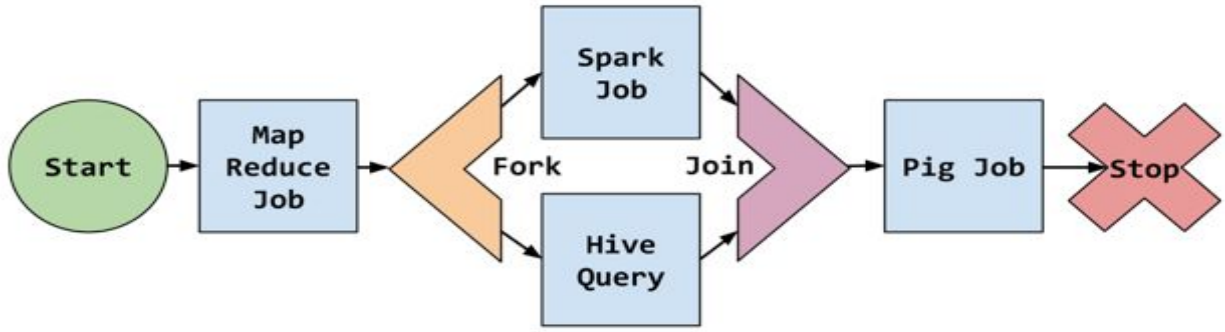
Hadoop MapReduce *Overview*

- Hadoop MapReduce is a software framework for easily writing applications which process vast amounts of data (multi-terabyte data-sets) in-parallel on large clusters (thousands of nodes) of commodity hardware in a reliable, fault-tolerant manner.
- A *MapReduce job* usually splits the input data-set into independent chunks which are processed by the *map tasks* in a completely parallel manner.
- The framework sorts the outputs of the maps, which are then input to the *reduce tasks*. Typically, both the input and the output of the job are stored in a file-system.
- The framework takes care of scheduling tasks, monitoring them and re-executes the failed tasks.
- Typically, the compute nodes and the storage nodes are the same, that is, the MapReduce framework and the Hadoop Distributed File System are running on the same set of nodes.
- This configuration allows the framework to effectively schedule tasks on the nodes where data is already present, resulting in very high aggregate bandwidth across the cluster.
- The MapReduce framework consists of a single master JobTracker and one slave TaskTracker per cluster-node.
- The master is responsible for scheduling the jobs' component tasks on the slaves, monitoring them, and re-executing the failed tasks.
- The slaves execute the tasks as directed by the master.
- Minimally, applications specify the input/output locations and supply *map* and *reduce* functions via implementations of appropriate interfaces and/or abstract-classes.
- These, and other job parameters, comprise the *job configuration*. The Hadoop *job client* then submits the job (jar/executable etc.) and configuration to the JobTracker which then assumes the responsibility of distributing the software/configuration to the slaves, scheduling tasks and monitoring them, providing status and diagnostic information to the job-client.

Apache Oozie Workflow Scheduler for Hadoop *Overview*

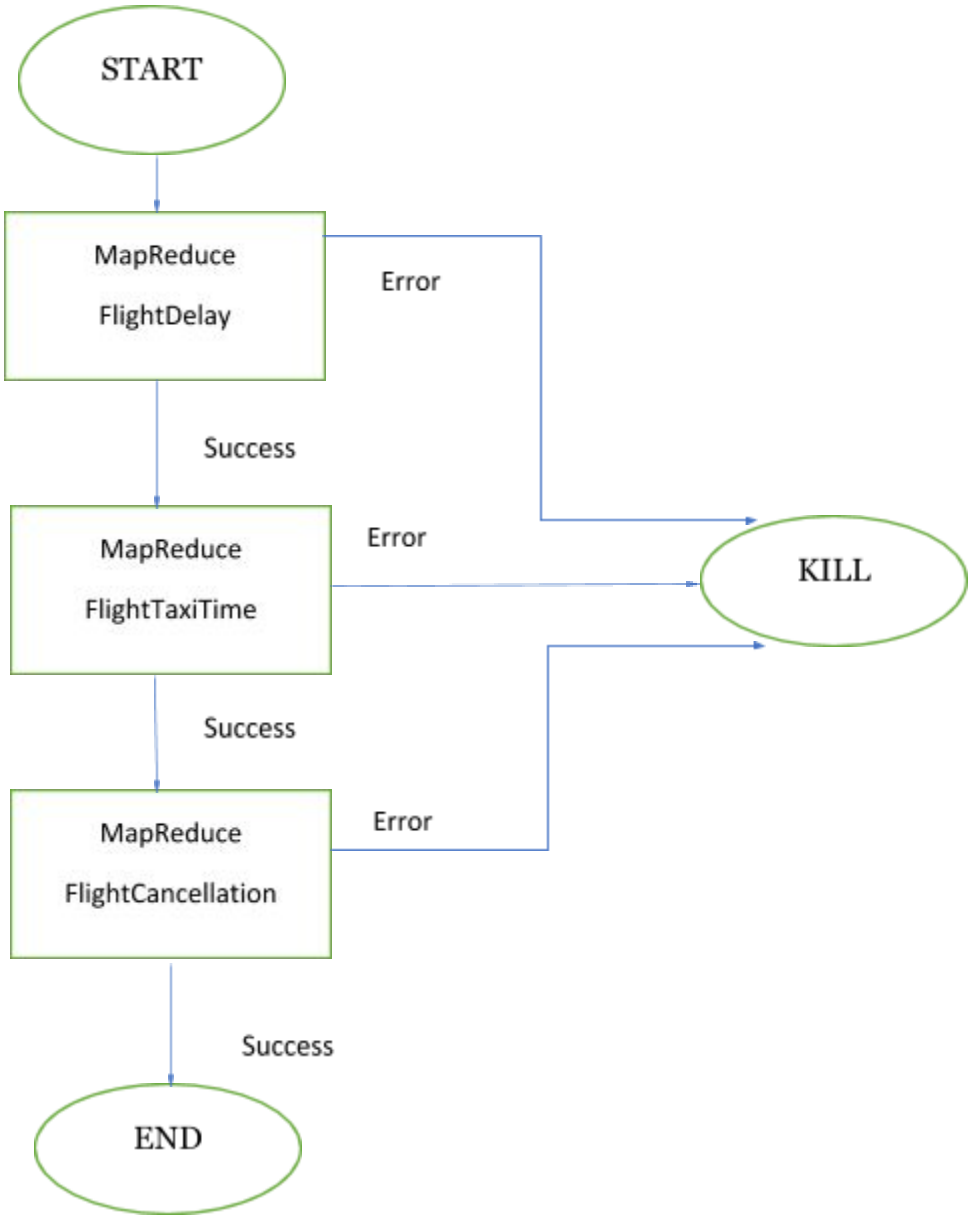
- Oozie is a workflow scheduler system to manage Apache Hadoop jobs.
- Oozie Workflow jobs are Directed Acyclical Graphs (DAGs) of actions.
- Oozie Coordinator jobs are recurrent Oozie Workflow jobs triggered by time (frequency) and data availability.
- Oozie is integrated with the rest of the Hadoop stack supporting several types of Hadoop jobs out of the box (such as Java map-reduce, Streaming map-reduce, Pig, Hive, Sqoop and Distcp) as well as system specific jobs (such as Java programs and shell scripts).
- Oozie is a scalable, reliable, and extensible system.

Oozie Standard Workflow:



5. a)

Structure of Oozie Workflow for Flight data analysis:



5. b)
Algorithm designed to solve each of the problems

The 3 airlines with the highest and lowest probability, respectively, for being on schedule

Mapper Phase:

1. Read input files line by line.
2. Since it is a comma separated file(csv) we split it based on comma and store all the fields in an array.
3. Fetch the values for fields corresponding to airlines unique carrier code and arrival delay.
4. For each data row, it generates a key as Carrier Code with prefix appended "a-". And, it also generate key as Carrier Code with prefix appended "b-" if that data row has ArrDelay > 10 mins and, it generates 1 as values for each of them
   Example: For data row with Carrier Code "A", and Air Delay 11
   mapper will generate --> **key - a-A; value 1**
   --> **key - b-A; value 1**
   For data row with Carrier Code "B", and Air Delay 4
   mapper will generate --> **key - a-B; value 1**
5. Write to context.

Reducer Phase:

1.Read context.
2.It calculates probabilities for each carrier(flight) being late.
3.It finds the top 3 most reliable and less reliable flights from all flights.
4.So, it will produce more reliable flight codes as keys and their probabilities of being late as values.
   Example : **key - X; value 0.10**
   key - Y; value 0.15
   key - Z; value 0.20
5.So, it will produce least reliable flight codes as keys and their probabilities of being late as values .
   Example : **key -A; value 0.40**
   key - B; value 0.35
   key - C; value 0.30
6. Write the output to context.

Calculation of 3 airports with longest and shortest average Taxi time (In and Out)

Mapper Phase:

1. Read input files line by line.
2. Since it is a comma separated file(csv) we split it based on comma and store all the fields in an array.

3. Fetch the values for fields corresponding to origin, taxiIn, destination and taxiOut.
4. Adding origin and taxiIn time in the context
5. Adding destination and taxiOut time in the context

AirportTaxiInTimeMapper
It calculates average TaxiIn time for each airport, generates a key as Airport Code.

And, it generates airport Taxi In time as value.
So, mapper will generate --> **key - A; value 2**
--> **key - A; value 4**
--> **key - B; value 8**
--> **key - B; value 6**

AirportTaxiTimeAvgReducer
It calculates the average Taxi In time for all the airports. So it will produce average Taxi In time for all airports.

Example : **A 3**
B 7

Airport TaxiIn time Max
It finds the top 3 average TaxiIn time.

FindingTop3Mapper
It generates a key as average TaxiIn time of the airport and generates value as average airport code.
Means it reveres key and value to perform sorting based on TaxiIn time.
So, A 3 will be converted to --> **key - 3; value - A**

Sorting - It sorts key-pairs in decreasing order.

FindingTop3Combiner
Each combiner produces the first 3 key-pairs by decreasing order of avg. taxi time from all sorted key-pairs to make sure that reducers will have less records to process.

FindingTop3Reducer
It produces the first 3 key-pairs by decreasing order of avg. taxi time, which is our output.
Example: **B 7** (here B is airport code and 7 is its average Taxi In time)
A 3 (here A is airport code and 3 is its average Taxi In time)
C 2 (here C is airport code and 2 is its average Taxi In time)

Airport TaxiIn time Min
It finds the lowest 3 average TaxiIn times.

FindingTop3Mapper

It generates a key as average TaxiIn time of the airport and It generates value as average airport code.
Means it reveres key and value to perform sorting based on Taxi In time.
So, A 3 will be converted to --> **key - 3; value - A**

Sorting - It sorts key-pairs in increasing order.

FindingTop3Combiner

Each combiner produces the first 3 key-pairs by increasing order of avg. taxi time from all shorted key-pairs to make sure that reducers will have less records to process.

FindingTop3Reducer

It produces the first 3 key-pairs by increasing order of avg. taxi time , which is our output.
Example: **C 2 (here C is airport code and 2 is its average Taxi In time)**
A 3 (here A is airport code and 3 is its average Taxi In time)
B 7 (here B is airport code and 7 is its average Taxi In time)

Airport Average TaxiOut time

It calculates the average Taxi Out time for each airport.

AirportTaxiOutTimeMapper

It generates a key as airport Code and it generates airport Taxi Out time as value.it will generate --> **key - A; value 2**
--> **key - A; value 4**
--> **key - B; value 8**
--> **key - B; value 6**

AirportTaxiTimeAvgReducer

It calculates average Taxi Out time for all the airports.So, it will produce average Taxi Out time for all airports.
Example : **A 3**
B 7

Airport Taxi Out time Max

It finds the top 3 average Taxi Out time.

FindingTop3Mapper

It generates a key as the average Taxi Out time of the airport. and It generates value as an average airport code.
Means it reveres key and value to perform sorting based on Taxi Out time.
So, A 3 will be converted to --> **key - 3; value - A**

It sorts key-pairs in decreasing order.

FindingTop3Combiner

Each combiner produces the first 3 key-pairs by decreasing order of avg. taxi time from all sorted key-pairs to make sure that reducers will have less records to process.

FindingTop3Reducer

It produces the first 3 key-pairs by decreasing order of average taxi time , which is our output.
Example: **B 7**
(here B is airport code and 7 is its average Taxi Out time)
A 3
(here A is airport code and 3 is its average Taxi Out time)
C 2
(here C is airport code and 2 is its average Taxi Out time)

Airport Taxi Out time Min

It finds the lowest 3 average TaxiOut time.

FindingTop3Mapper

It generates key as average TaxiOut time of airport and generates value as average airport code.
Means it reveres key and value to perform sorting based on TaxiOut time.
So, A 3 will be converted to --> **key - 3; value - A**

Sorting - It sorts key-pairs in ascending order.

FindingTop3Combiner

Each combiner produces the first 3 key-pairs by ascending order of average taxi time from all sorted key-pairs to make sure that reducers will have less records to process.

FindingTop3Reducer

It produces the first 3 key-pairs by ascending order of avg. taxi time , which is our output.
Example: **C 2**
(here C is airport code and 2 is its average Taxi Out time)
A 3
(here A is airport code and 3 is its average Taxi Out time)
B 7
(here B is airport code and 7 is its average Taxi Out time)

Find most common reason for cancellation of flight:

Mapper Phase:

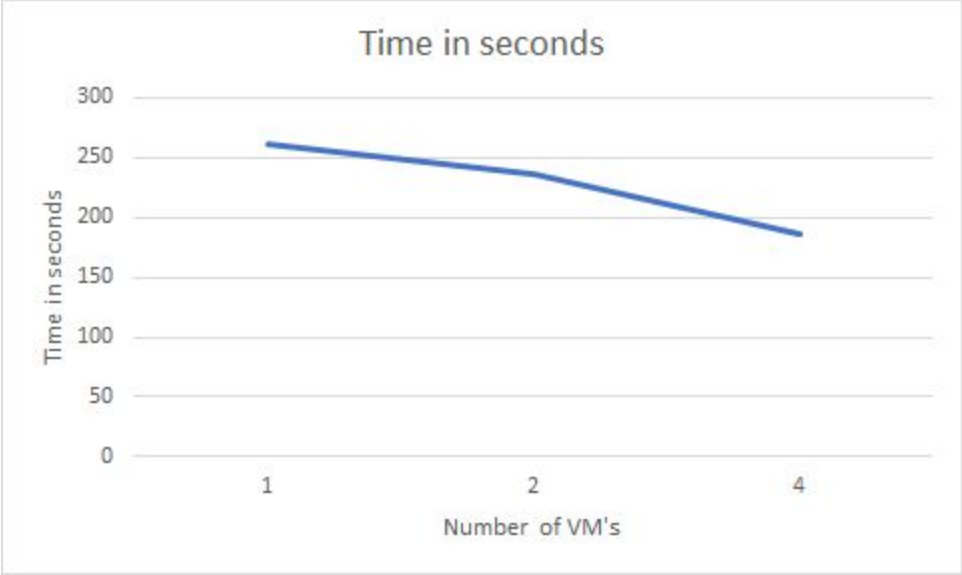
1. Read input files line by line.
2. Since it is a comma separated file(csv) we split it based on comma and store all the fields in an array.
3. Fetch the values for fields corresponding to the CancellationCode column.
4. Generate key as Flight Cancelled Code to count the occurrence of it and generate 1 as value for each of them.
   Example:**key:A;value:1**
   key:B;value:1
5. Write to context.

Reducer Phase:

1. Read context.
2. Iterate over the context values and Calculate sum of all the values.
3. It calculates occurrence of all Flight Cancelled Code, also finds which is maximum.
4. So it produces max Flight Cancelled code and its count.
   Example: **A 54**
5. Write the output to context.

5.c)

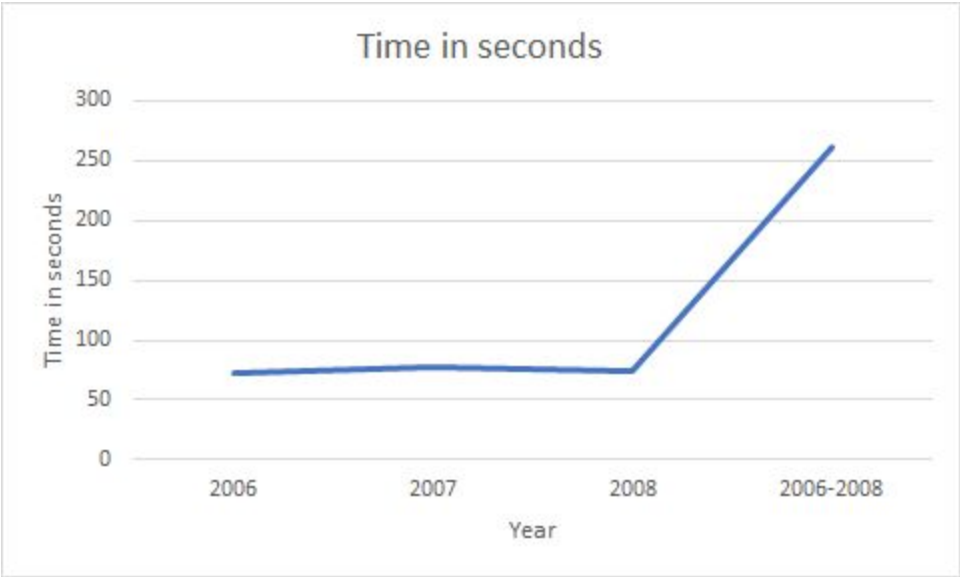
A performance measurement plot that compares the workflow execution time in response to an increasing number of VMs used for processing the entire data set (3 years) and an in-depth discussion on the observed performance comparison results.



- Here we are doing an experiment on performance of workflow having MapReduce jobs by varying the number of resources used.
- We are keeping data constant for all the runs i.e. flight data for 3 years (2006 - 2008). We are starting by using Hadoop on 1,2 and 4 Virtual Machines.
- The total execution time taken is 262 seconds.
- Now we increase 1 VM and then 2 VM's.
- We notice that as we increase the number of VMs there is a significant drop-in time taken for execution.

5.d)

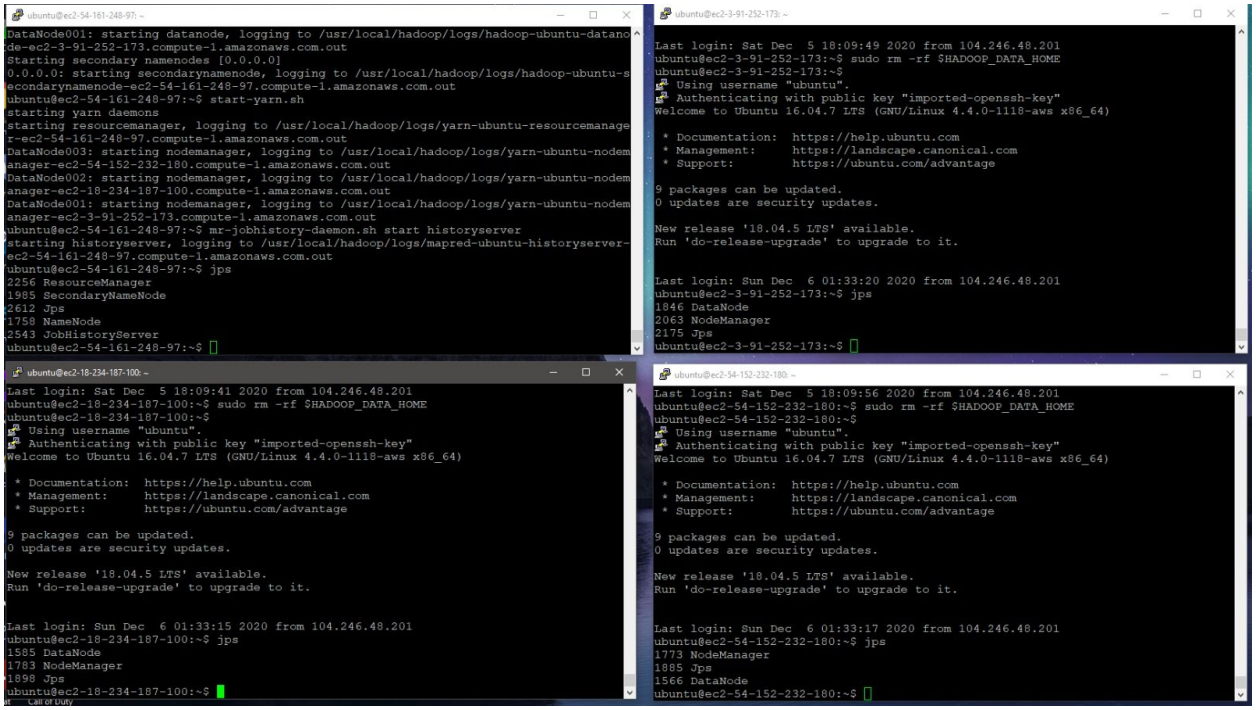
A performance measurement plot that compares the workflow execution time in response to an increasing data size (from 1 year to 3 years) and an in-depth discussion on the observed performance comparison results.



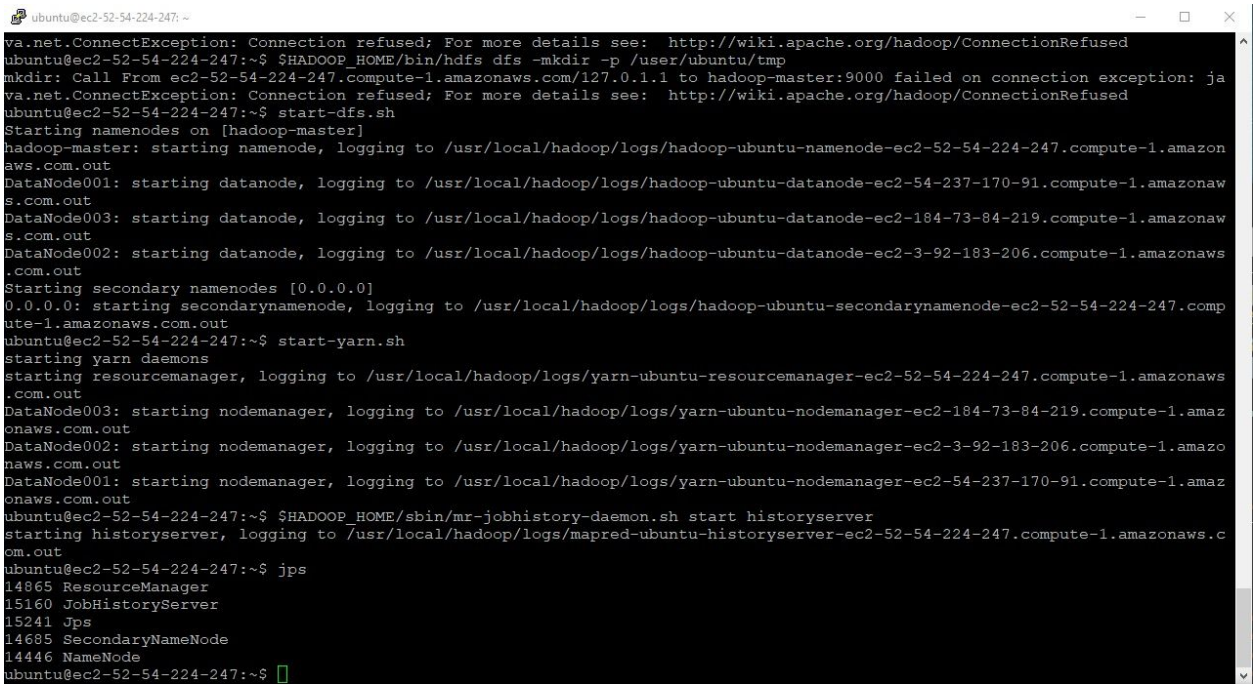
- In this experiment, we want to find out performance with respect to varying input data. We are using 1 Virtual machine throughout this experiment.
- First, we execute workflow on only one data file (2006.csv).
- We see that execution completes in 73 seconds. Now, we run for 2007.csv and 2008.csv. And 3 years of data together record the execution time of 262 seconds.
- We observe that the execution time gradually increases as the input data increases.

Some of the PrintScreens:

Hadoop Multi Clusters



NameNode



```
20/12/06 00:36:27 INFO mapred.Task: Task 'attempt_local819654000_0008_m_000020_0' done.
20/12/06 00:36:27 INFO mapred.LocalJobRunner: Finishing task: attempt_local819654000_0008_m_000020_0
20/12/06 00:36:27 INFO mapred.LocalJobRunner: Map task executor complete.
20/12/06 00:36:27 INFO mapred.Task:  Using ResourceCalculatorPlugin : null
20/12/06 00:36:27 INFO mapred.LocalJobRunner:
20/12/06 00:36:27 INFO mapred.Merger: Merging 21 sorted segments
20/12/06 00:36:27 INFO mapred.Merger: Merging 3 intermediate segments out of a total of 21
20/12/06 00:36:27 INFO mapred.Merger: Merging 10 intermediate segments out of a total of 19
20/12/06 00:36:28 INFO mapred.JobClient:  map 100% reduce 0%
20/12/06 00:36:29 INFO mapred.Merger: Down to the last merge-pass, with 10 segments left of total size: 129860765 bytes
20/12/06 00:36:29 INFO mapred.LocalJobRunner:
a-9E
a-AA
a-AQ
a-AS
a-B6
a-CO
a-DL
a-EV
a-F9
a-FL
a-HA
a-MQ
a-NW
a-OH
a-OO
a-UA
a-US
a-WN
20/12/06 00:36:33 INFO mapred.LocalJobRunner: reduce > reduce
a-XE
a-YV
b-9E
20/12/06 00:36:34 INFO mapred.JobClient:  map 100% reduce 87%
b-AA
b-AQ
b-AS
b-B6
b-CO
b-DL
b-EV
b-F9
b-FL
b-HA
b-MQ
```

```
b-FL
b-HA
b-MQ
b-NW
b-OH
b-OO
b-UA
b-US
b-WN
b-XE
b-YV
20/12/06 00:36:35 INFO mapred.Task: Task:attempt_local819654000_0008_r_000000_0 is done. And is in the process of commiting
20/12/06 00:36:35 INFO mapred.LocalJobRunner: reduce > reduce
20/12/06 00:36:35 INFO mapred.Task: Task attempt_local819654000_0008_r_000000_0 is allowed to commit now
20/12/06 00:36:35 INFO output.FileOutputCommitter: Saved output of task 'attempt_local819654000_0008_r_000000_0' to output_AirlinesBeingOnSchedule
20/12/06 00:36:35 INFO mapred.LocalJobRunner: reduce > reduce
20/12/06 00:36:35 INFO mapred.Task: Task 'attempt_local819654000_0008_r_000000_0' done.
20/12/06 00:36:35 INFO mapred.JobClient:  map 100% reduce 100%
20/12/06 00:36:35 INFO mapred.JobClient: Job complete: job_local819654000_0008
20/12/06 00:36:35 INFO mapred.JobClient: Counters: 17
20/12/06 00:36:35 INFO mapred.JobClient:   Map-Reduce Framework
20/12/06 00:36:35 INFO mapred.JobClient:     Spilled Records=30889995
20/12/06 00:36:35 INFO mapred.JobClient:     Map output materialized bytes=129860871
20/12/06 00:36:35 INFO mapred.JobClient:     Reduce input records=8657383
20/12/06 00:36:35 INFO mapred.JobClient:     Map input records=7009729
20/12/06 00:36:35 INFO mapred.JobClient:     SPLIT_RAW_BYTES=2205
20/12/06 00:36:35 INFO mapred.JobClient:     Map output bytes=112545979
20/12/06 00:36:35 INFO mapred.JobClient:     Reduce shuffle bytes=0
20/12/06 00:36:35 INFO mapred.JobClient:     Reduce input groups=40
20/12/06 00:36:35 INFO mapred.JobClient:     Combine output records=0
20/12/06 00:36:35 INFO mapred.JobClient:     Reduce output records=8
20/12/06 00:36:35 INFO mapred.JobClient:     Map output records=8657383
20/12/06 00:36:35 INFO mapred.JobClient:     Combine input records=0
20/12/06 00:36:35 INFO mapred.JobClient:     Total committed heap usage (bytes)=47551873024
20/12/06 00:36:35 INFO mapred.JobClient:   File Input Format Counters
20/12/06 00:36:35 INFO mapred.JobClient:     Bytes Read=689495264
20/12/06 00:36:35 INFO mapred.JobClient:   FileSystemCounters
20/12/06 00:36:35 INFO mapred.JobClient:     FILE_BYTES_WRITTEN=38138685135
20/12/06 00:36:35 INFO mapred.JobClient:     FILE_BYTES_READ=90407821764
20/12/06 00:36:35 INFO mapred.JobClient:   File Output Format Counters
20/12/06 00:36:35 INFO mapred.JobClient:     Bytes Written=228
```

DataNode #1

```
2020-12-07 00:53:44,045 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: STARTUP_MSG:
/************************************************************
STARTUP_MSG: Starting DataNode
STARTUP_MSG:   host = ec2-54-237-170-91.compute-1.amazonaws.com/127.0.1.1
STARTUP_MSG:   args = []
STARTUP_MSG:   version = 2.9.2
STARTUP_MSG:   classpath = /usr/local/hadoop/etc/hadoop:/usr/local/hadoop/share/hadoop/common/lib/commons-codec-1.4.jar:/usr/local/hadoop/share/hadoop/common/lib/
io-2.4.jar:/usr/local/hadoop/share/hadoop/mapreduce/lib/guice-3.0.jar:/usr/local/hadoop/share/hadoop/mapreduce/lib/log4j-1.2.17.jar:/usr/local/hadoop/share/hadoop
STARTUP_MSG:   build = https://git-wip-us.apache.org/repos/asf/hadoop.git -r 826afbeae31ca687bc2f8471dc841b66ed2c6704; compiled by 'ajisaka' on 2018-11-13T12:42Z
STARTUP_MSG:   java = 1.8.0_275
************************************************************/
2020-12-07 00:53:44,070 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: registered UNIX signal handlers for [TERM, HUP, INT]
2020-12-07 00:53:45,122 INFO org.apache.hadoop.hdfs.server.datanode.checker.ThrottledAsyncChecker: Scheduling a check for [DISK]file:/home/ubuntu/hadoop_data/hdfs
2020-12-07 00:53:45,270 INFO org.apache.hadoop.metrics2.impl.MetricsConfig: loaded properties from hadoop-metrics2.properties
2020-12-07 00:53:45,539 INFO org.apache.hadoop.metrics2.impl.MetricsSystemImpl: Scheduled Metric snapshot period at 10 second(s).
2020-12-07 00:53:45,539 INFO org.apache.hadoop.metrics2.impl.MetricsSystemImpl: DataNode metrics system started
2020-12-07 00:53:45,547 INFO org.apache.hadoop.hdfs.server.common.Util: dfs.datanode.fileio.profiling.sampling.percentage set to 0. Disabling file IO profiling
2020-12-07 00:53:45,551 INFO org.apache.hadoop.hdfs.server.datanode.BlockScanner: Initialized block scanner with targetBytesPerSec 1048576
2020-12-07 00:53:45,560 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: Configured hostname is ec2-54-237-170-91.compute-1.amazonaws.com
2020-12-07 00:53:45,560 INFO org.apache.hadoop.hdfs.server.common.Util: dfs.datanode.fileio.profiling.sampling.percentage set to 0. Disabling file IO profiling
2020-12-07 00:53:45,560 WARN org.apache.hadoop.conf.Configuration: No unit for dfs.datanode.outliers.report.interval(1800000) assuming MILLISECONDS
2020-12-07 00:53:45,566 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: Starting DataNode with maxLockedMemory = 0
2020-12-07 00:53:45,614 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: Opened streaming server at /0.0.0.0:50010
2020-12-07 00:53:45,623 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: Balancing bandwidth is 10485760 bytes/s
2020-12-07 00:53:45,623 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: Number threads for balancing is 50
2020-12-07 00:53:45,819 INFO org.mortbay.log: Logging to org.slf4j.impl.Log4jLoggerAdapter(org.mortbay.log) via org.mortbay.log.Slf4jLog
2020-12-07 00:53:45,838 INFO org.apache.hadoop.security.authentication.server.AuthenticationFilter: Unable to initialize FileSignerSecretProvider, falling back to
2020-12-07 00:53:45,884 INFO org.apache.hadoop.http.HttpRequestLog: Http request log for http.requests.datanode is not defined
2020-12-07 00:53:45,891 INFO org.apache.hadoop.http.HttpServer2: Added global filter 'safety' (class=org.apache.hadoop.http.HttpServer2$QuotingInputFilter)
2020-12-07 00:53:45,892 INFO org.apache.hadoop.http.HttpServer2: Added filter static_user_filter (class=org.apache.hadoop.http.lib.StaticUserWebFilter$StaticUserF
2020-12-07 00:53:45,893 INFO org.apache.hadoop.http.HttpServer2: Added filter static_user_filter (class=org.apache.hadoop.http.lib.StaticUserWebFilter$StaticUserF
2020-12-07 00:53:45,893 INFO org.apache.hadoop.http.HttpServer2: Added filter static_user_filter (class=org.apache.hadoop.http.lib.StaticUserWebFilter$StaticUserF
2020-12-07 00:53:45,921 INFO org.apache.hadoop.http.HttpServer2: Jetty bound to port 43574
```

Datanode #2

```
2020-12-07 00:53:44,602 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: STARTUP_MSG:
/************************************************************
STARTUP_MSG: Starting DataNode
STARTUP_MSG:   host = ec2-3-92-183-206.compute-1.amazonaws.com/127.0.1.1
STARTUP_MSG:   args = []
STARTUP_MSG:   version = 2.9.2
STARTUP_MSG:   classpath = /usr/local/hadoop/etc/hadoop:/usr/local/hadoop/share/hadoop/common/lib/commons-codec-1.4.jar:/usr/local/hadoop/share/hadoop/common/lib/
io-2.4.jar:/usr/local/hadoop/share/hadoop/mapreduce/lib/guice-3.0.jar:/usr/local/hadoop/share/hadoop/mapreduce/lib/log4j-1.2.17.jar:/usr/local/hadoop/share/hadoop
STARTUP_MSG:   build = https://git-wip-us.apache.org/repos/asf/hadoop.git -r 826afbeae31ca687bc2f8471dc841b66ed2c6704; compiled by 'ajisaka' on 2018-11-13T12:42Z
STARTUP_MSG:   java = 1.8.0_275
************************************************************/
2020-12-07 00:53:44,625 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: registered UNIX signal handlers for [TERM, HUP, INT]
2020-12-07 00:53:45,914 INFO org.apache.hadoop.hdfs.server.datanode.checker.ThrottledAsyncChecker: Scheduling a check for [DISK]file:/home/ubuntu/hadoop_data/hdfs
2020-12-07 00:53:46,084 INFO org.apache.hadoop.metrics2.impl.MetricsConfig: loaded properties from hadoop-metrics2.properties
2020-12-07 00:53:46,497 INFO org.apache.hadoop.metrics2.impl.MetricsSystemImpl: Scheduled Metric snapshot period at 10 second(s).
2020-12-07 00:53:46,497 INFO org.apache.hadoop.metrics2.impl.MetricsSystemImpl: DataNode metrics system started
2020-12-07 00:53:46,507 INFO org.apache.hadoop.hdfs.server.common.Util: dfs.datanode.fileio.profiling.sampling.percentage set to 0. Disabling file IO profiling
2020-12-07 00:53:46,509 INFO org.apache.hadoop.hdfs.server.datanode.BlockScanner: Initialized block scanner with targetBytesPerSec 1048576
2020-12-07 00:53:46,519 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: Configured hostname is ec2-3-92-183-206.compute-1.amazonaws.com
2020-12-07 00:53:46,520 INFO org.apache.hadoop.hdfs.server.common.Util: dfs.datanode.fileio.profiling.sampling.percentage set to 0. Disabling file IO profiling
2020-12-07 00:53:46,520 WARN org.apache.hadoop.conf.Configuration: No unit for dfs.datanode.outliers.report.interval(1800000) assuming MILLISECONDS
2020-12-07 00:53:46,547 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: Starting DataNode with maxLockedMemory = 0
2020-12-07 00:53:46,634 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: Opened streaming server at /0.0.0.0:50010
2020-12-07 00:53:46,647 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: Balancing bandwidth is 10485760 bytes/s
2020-12-07 00:53:46,647 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: Number threads for balancing is 50
2020-12-07 00:53:46,835 INFO org.mortbay.log: Logging to org.slf4j.impl.Log4jLoggerAdapter(org.mortbay.log) via org.mortbay.log.Slf4jLog
2020-12-07 00:53:46,850 INFO org.apache.hadoop.security.authentication.server.AuthenticationFilter: Unable to initialize FileSignerSecretProvider, falling back to
2020-12-07 00:53:46,896 INFO org.apache.hadoop.http.HttpRequestLog: Http request log for http.requests.datanode is not defined
2020-12-07 00:53:46,908 INFO org.apache.hadoop.http.HttpServer2: Added global filter 'safety' (class=org.apache.hadoop.http.HttpServer2$QuotingInputFilter)
2020-12-07 00:53:46,909 INFO org.apache.hadoop.http.HttpServer2: Added filter static_user_filter (class=org.apache.hadoop.http.lib.StaticUserWebFilter$StaticUserF
2020-12-07 00:53:46,910 INFO org.apache.hadoop.http.HttpServer2: Added filter static_user_filter (class=org.apache.hadoop.http.lib.StaticUserWebFilter$StaticUserF
2020-12-07 00:53:46,910 INFO org.apache.hadoop.http.HttpServer2: Added filter static_user_filter (class=org.apache.hadoop.http.lib.StaticUserWebFilter$StaticUserF
2020-12-07 00:53:46,977 INFO org.apache.hadoop.http.HttpServer2: Jetty bound to port 42979
```

DataNode #3

```
2020-12-07 00:54:57,419 INFO org.apache.hadoop.yarn.server.nodemanager.NodeManager: STARTUP_MSG:
/************************************************************
STARTUP_MSG: Starting NodeManager
STARTUP_MSG:   host = ec2-184-73-84-219.compute-1.amazonaws.com/127.0.1.1
STARTUP_MSG:   args = []
STARTUP_MSG:   version = 2.9.2
STARTUP_MSG:   classpath = /usr/local/hadoop/etc/hadoop:/usr/local/hadoop/etc/hadoop:/usr/local/hadoop/etc/hadoop:/usr/local/hadoop/share/hadoop/common/lib/common
io-2.4.jar:/usr/local/hadoop/share/hadoop/mapreduce/lib/guice-3.0.jar:/usr/local/hadoop/share/hadoop/mapreduce/lib/log4j-1.2.17.jar:/usr/local/hadoop/share/hadoop
STARTUP_MSG:   build = https://git-wip-us.apache.org/repos/asf/hadoop.git -r 826afbeae31ca687bc2f8471dc841b66ed2c6704; compiled by 'ajisaka' on 2018-11-13T12:42Z
STARTUP_MSG:   java = 1.8.0_275
************************************************************/
2020-12-07 00:54:57,432 INFO org.apache.hadoop.yarn.server.nodemanager.NodeManager: registered UNIX signal handlers for [TERM, HUP, INT]
2020-12-07 00:54:58,371 INFO org.apache.hadoop.yarn.server.nodemanager.NodeManager: Node Manager health check script is not available or doesn't have execute perm
2020-12-07 00:54:58,523 INFO org.apache.hadoop.yarn.event.AsyncDispatcher: Registering class org.apache.hadoop.yarn.server.nodemanager.containermanager.container.
2020-12-07 00:54:58,524 INFO org.apache.hadoop.yarn.event.AsyncDispatcher: Registering class org.apache.hadoop.yarn.server.nodemanager.containermanager.applicatio
2020-12-07 00:54:58,530 INFO org.apache.hadoop.yarn.event.AsyncDispatcher: Registering class org.apache.hadoop.yarn.server.nodemanager.containermanager.localizer.
2020-12-07 00:54:58,531 INFO org.apache.hadoop.yarn.event.AsyncDispatcher: Registering class org.apache.hadoop.yarn.server.nodemanager.containermanager.AuxService
2020-12-07 00:54:58,532 INFO org.apache.hadoop.yarn.event.AsyncDispatcher: Registering class org.apache.hadoop.yarn.server.nodemanager.containermanager.monitor.Co
2020-12-07 00:54:58,533 INFO org.apache.hadoop.yarn.event.AsyncDispatcher: Registering class org.apache.hadoop.yarn.server.nodemanager.containermanager.launcher.C
2020-12-07 00:54:58,533 INFO org.apache.hadoop.yarn.event.AsyncDispatcher: Registering class org.apache.hadoop.yarn.server.nodemanager.containermanager.scheduler.
2020-12-07 00:54:58,577 INFO org.apache.hadoop.yarn.event.AsyncDispatcher: Registering class org.apache.hadoop.yarn.server.nodemanager.ContainerManagerEventType f
2020-12-07 00:54:58,577 INFO org.apache.hadoop.yarn.event.AsyncDispatcher: Registering class org.apache.hadoop.yarn.server.nodemanager.NodeManagerEventType for cl
2020-12-07 00:54:58,641 INFO org.apache.hadoop.metrics2.impl.MetricsConfig: loaded properties from hadoop-metrics2.properties
2020-12-07 00:54:58,833 INFO org.apache.hadoop.metrics2.impl.MetricsSystemImpl: Scheduled Metric snapshot period at 10 second(s).
2020-12-07 00:54:58,833 INFO org.apache.hadoop.metrics2.impl.MetricsSystemImpl: NodeManager metrics system started
2020-12-07 00:54:58,864 INFO org.apache.hadoop.yarn.server.nodemanager.DirectoryCollection: Disk Validator: yarn.nodemanager.disk-validator is loaded.
```

Oozie:

```
[INFO] Apache Oozie Main ................................. SUCCESS [  0.950 s]
[INFO] Apache Oozie Hadoop Utils hadoop-2-4.3.1 ........... SUCCESS [  1.529 s]
[INFO] Apache Oozie Hadoop Distcp hadoop-2-4.3.1 .......... SUCCESS [  0.077 s]
[INFO] Apache Oozie Hadoop Auth hadoop-2-4.3.1 Test ....... SUCCESS [  0.153 s]
[INFO] Apache Oozie Hadoop Libs ........................... SUCCESS [  0.012 s]
[INFO] Apache Oozie Client ................................ SUCCESS [  8.263 s]
[INFO] Apache Oozie Share Lib Oozie ....................... SUCCESS [  2.486 s]
[INFO] Apache Oozie Share Lib HCatalog .................... SUCCESS [  1.860 s]
[INFO] Apache Oozie Share Lib Distcp ...................... SUCCESS [  0.534 s]
[INFO] Apache Oozie Core .................................. SUCCESS [ 33.010 s]
[INFO] Apache Oozie Share Lib Streaming ................... SUCCESS [  3.190 s]
[INFO] Apache Oozie Share Lib Pig ......................... SUCCESS [  3.032 s]
[INFO] Apache Oozie Share Lib Hive ........................ SUCCESS [  3.619 s]
[INFO] Apache Oozie Share Lib Hive 2 ...................... SUCCESS [  3.179 s]
[INFO] Apache Oozie Share Lib Sqoop ....................... SUCCESS [  1.458 s]
[INFO] Apache Oozie Examples .............................. SUCCESS [  2.607 s]
[INFO] Apache Oozie Share Lib Spark ....................... SUCCESS [  5.701 s]
[INFO] Apache Oozie Share Lib ............................. SUCCESS [ 13.171 s]
[INFO] Apache Oozie Docs .................................. SUCCESS [  6.401 s]
[INFO] Apache Oozie WebApp ................................ SUCCESS [ 24.754 s]
[INFO] Apache Oozie Tools ................................. SUCCESS [  2.356 s]
[INFO] Apache Oozie MiniOozie ............................. SUCCESS [  0.946 s]
[INFO] Apache Oozie Distro ................................ SUCCESS [ 19.807 s]
[INFO] Apache Oozie ZooKeeper Security Tests .............. SUCCESS [  3.116 s]
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time: 02:23 min
[INFO] Finished at: 2020-12-06T20:04:09+00:00
[INFO] Final Memory: 595M/3302M
[INFO] ------------------------------------------------------------------------

Oozie distro created, DATE[2020.12.06-20:01:45GMT] VC-REV[unavailable], available at [/home/ubuntu/oozie-4.3.1/distro/target]
```