CIT650: Introduction to Big Data

Lab #2

Lab Goals

- YARN Architecture
- Running Hadoop Cluster using Docker Compose
- Job Verbose
- Hadoop Different Configurations
- Customize YARN Configurations
- Monitor YARN
- Application Monitoring
- Application Logs using Terminal
- Lab Task

YARN Architecture

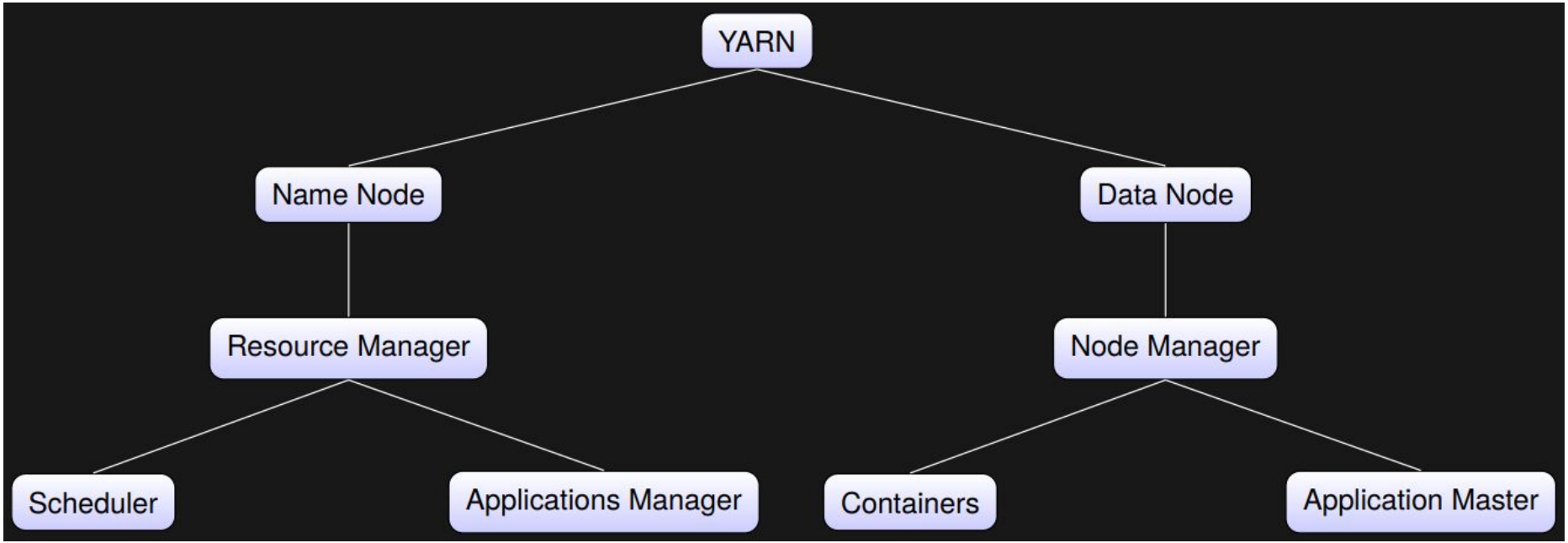
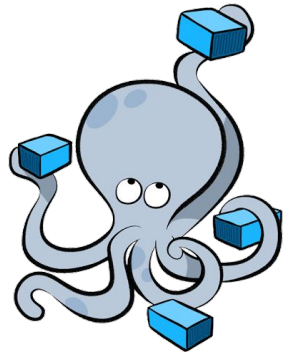
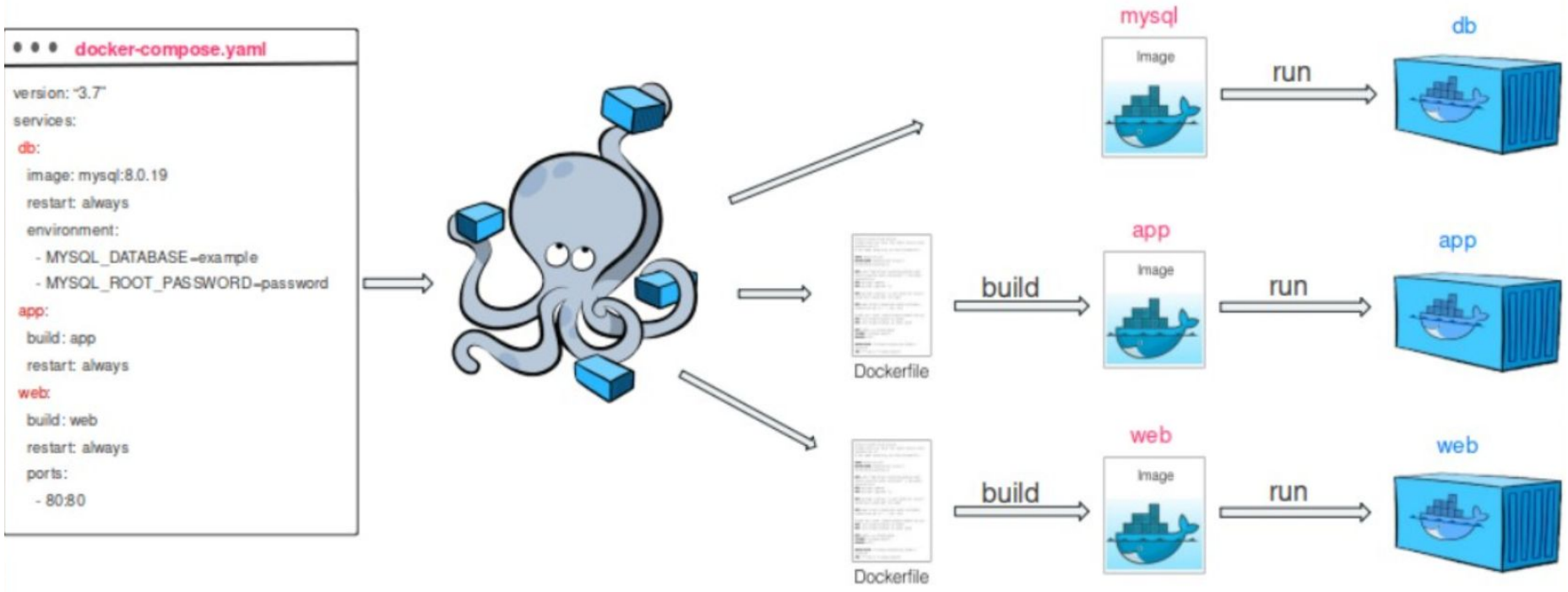


Figure Source

Running Hadoop Cluster using Docker Compose

- Docker Compose is a tool for defining and running multi-container Docker applications.
- It allows you to define an entire application stack, including services, networks, and volumes, in a single file called docker-compose.yml.
- This file provides a concise and human-readable way to specify how different Docker containers should interact with each other.




docker-compose.yaml
version: "3.7"
services:
db:
image: mysql:8.0.19
restart: always
environment:
- MYSQL_DATABASE=example
- MYSQL_ROOT_PASSWORD=password
app:
build: app
restart: always
web:
build: web
restart: always
ports:
- 80:80
mysql
Image
run
db
Dockerfile
build
app
Image
run
app
Dockerfile
build
web
Image
run
web

Running Hadoop Cluster using Docker Compose (CONT.)

- **Step #1**: Make sure you've installed the docker-compose plugin
- **Step #2**: Run the following command `git clone https://github.com/big-data-europe/docker-hadoop.git`
- **Step #3**: Move to the cloned repo. `cd docker-hadoop`
- **Step #4**: Deploy the cluster by running `docker-compose up -d`
- **Step #5**: Check the running containers for Hadoop using `docker container ls`
- **Step #6**: Get the IP of each service using `docker inspect <container-name>`

Service	NameNode	DataNode	ResourceManager	NodeManager	HistoryServer
Port	9870	9864	8088	8042	8188

Running Hadoop Cluster using Docker Compose (CONT.)

- **Step #7:** Copy the JAR file used to run a MapReduce job into the namenode using the following command `docker cp <jar-file-path-on-host> namenode:<path-in-the-container>` (This jar performs a word count)
- **Step #8:** Move a simple data file to into the container by running the following command `docker cp <data-file-path-on-host> namenode:<path-in-the-container>`
- **Step #9:** Create a new directory inside the HDFS to put the data file in by running the following command in the namenode container `hdfs dfs -mkdir <dir-name>`
- **Step #10:** Move the data file into the HDFS using `hdfs dfs -put <input-dir-name >`
- **Step #11:** Run the job by `hadoop jar <jar-path-on-namenode> org.apache.hadoop.examples.WordCount <input-dir-hdfs> <output-dir-hdfs>`

Job Verbose

```
root@b9d303c6a919:/home# hadoop jar mrExample.jar org.apache.hadoop.examples.WordCount /tI /tO
2023-11-25 17:43:50,257 INFO client.RMProxy: Connecting to ResourceManager at resourcemanager/172.20.0.4:8032
2023-11-25 17:43:50,398 INFO client.AHSProxy: Connecting to Application History server at historyserver/172.20.0.3:10200
2023-11-25 17:43:50,573 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/root/.staging/job_1700931379311_0001
2023-11-25 17:43:50,658 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted = false, remoteHostTrusted = false
2023-11-25 17:43:50,759 INFO input.FileInputFormat: Total input files to process : 1
2023-11-25 17:43:50,786 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted = false, remoteHostTrusted = false
2023-11-25 17:43:51,213 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted = false, remoteHostTrusted = false
2023-11-25 17:43:51,219 INFO mapreduce.JobSubmitter: number of splits:1
2023-11-25 17:43:51,340 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted = false, remoteHostTrusted = false
2023-11-25 17:43:51,362 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1700931379311_0001
2023-11-25 17:43:51,363 INFO mapreduce.JobSubmitter: Executing with tokens: []
2023-11-25 17:43:51,498 INFO conf.Configuration: resource-types.xml not found
2023-11-25 17:43:51,499 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2023-11-25 17:43:51,774 INFO impl.YarnClientImpl: Submitted application application_1700931379311_0001
2023-11-25 17:43:51,809 INFO mapreduce.Job: The url to track the job: http://resourcemanager:8088/proxy/application_1700931379311_0001/
2023-11-25 17:43:51,810 INFO mapreduce.Job: Running job: job_1700931379311_0001
```

Job Verbose

```
2023-11-25 17:43:58,939 INFO mapreduce.Job:  map 0% reduce 0%
2023-11-25 17:44:03,984 INFO mapreduce.Job:  map 100% reduce 0%
2023-11-25 17:44:09,009 INFO mapreduce.Job:  map 100% reduce 100%
2023-11-25 17:44:09,015 INFO mapreduce.Job: Job job_1700931379311_0001 completed successfully
2023-11-25 17:44:09,096 INFO mapreduce.Job: Counters: 54
```

Job Verbose

```
File System Counters
        FILE: Number of bytes read=49
        FILE: Number of bytes written=458589
        FILE: Number of read operations=0
        FILE: Number of large read operations=0
        FILE: Number of write operations=0
        HDFS: Number of bytes read=123
        HDFS: Number of bytes written=22
        HDFS: Number of read operations=8
        HDFS: Number of large read operations=0
        HDFS: Number of write operations=2
        HDFS: Number of bytes read erasure-coded=0
```

Job Verbose

```
Job Counters
        Launched map tasks=1
        Launched reduce tasks=1
        Rack-local map tasks=1
        Total time spent by all maps in occupied slots (ms)=8176
        Total time spent by all reduces in occupied slots (ms)=15376
        Total time spent by all map tasks (ms)=2044
        Total time spent by all reduce tasks (ms)=1922
        Total vcore-milliseconds taken by all map tasks=2044
        Total vcore-milliseconds taken by all reduce tasks=1922
        Total megabyte-milliseconds taken by all map tasks=8372224
        Total megabyte-milliseconds taken by all reduce tasks=15745024
```

Job Verbose

```
Map-Reduce Framework
        Map input records=2
        Map output records=4
        Map output bytes=38
        Map output materialized bytes=41
        Input split bytes=101
        Combine input records=4
        Combine output records=3
        Reduce input groups=3
        Reduce shuffle bytes=41
        Reduce input records=3
        Reduce output records=3
        Spilled Records=6
        Shuffled Maps =1
        Failed Shuffles=0
        Merged Map outputs=1
        GC time elapsed (ms)=110
        CPU time spent (ms)=890
        Physical memory (bytes) snapshot=488210432
        Virtual memory (bytes) snapshot=13523062784
        Total committed heap usage (bytes)=524288000
        Peak Map Physical memory (bytes)=301412352
        Peak Map Virtual memory (bytes)=5090160640
        Peak Reduce Physical memory (bytes)=186798080
        Peak Reduce Virtual memory (bytes)=8432902144
```

Hadoop Different Configurations

- **Core Hadoop Configurations (core-site.xml):**
  - **io.compression.codecs:** Configures the compression codecs used in Hadoop. In this case, it includes the Snappy codec. Specifies the compression algorithms to be used for Hadoop data, and here, Snappy is included.
- **HDFS Configurations (hdfs-site.xml):**
  - **dfs.permissions.enabled:** Enables or disables HDFS permissions. All users have unrestricted access to HDFS, useful for certain setups but may compromise security.
- **MapReduce Configurations (mapred-site.xml):**
  - **mapreduce.map.memory.mb=4096:** Sets the amount of memory allocated for each map task.
  - **mapreduce.reduce.memory.mb=8192:** Sets the amount of memory allocated for each reduce task.

Customize YARN Configurations

- We can modify YARN configurations to adjust settings such as Memory, vCores.
- The environment variables used by Hadoop in our setup exist in ***hadoop.env*** file.
- So, we can change the required parameter according to our use case, then we need to rebuild the container, because the environment variables are only read during the initialization of the container.
- **YARN Configurations (yarn-site.xml):**
    - **yarn.resourcemanager.scheduler-class:** Specifies the class for the ResourceManager scheduler.
    - Determines the algorithm for scheduling resources among competing applications.
    - **yarn.nodemanager.resource.memory-mb=16384:** Specifies the maximum amount of memory a NodeManager can use for running containers.
    - **yarn.nodemanager.resource.cpu-vcores=4:** Specifies the maximum number of virtual CPU cores a NodeManager can use for running containers.

Monitor YARN

- You can access the YARN ResourceManager web UI at **http://<resource-manager-container-ip>:8088** in your browser.
- This UI provides information about the cluster, running applications, and resource usage. **Let's take a look at the applications.**

Cluster Metrics

Memory Total
16 GB

Scheduler Metrics

Scheduler Type
Capacity Scheduler

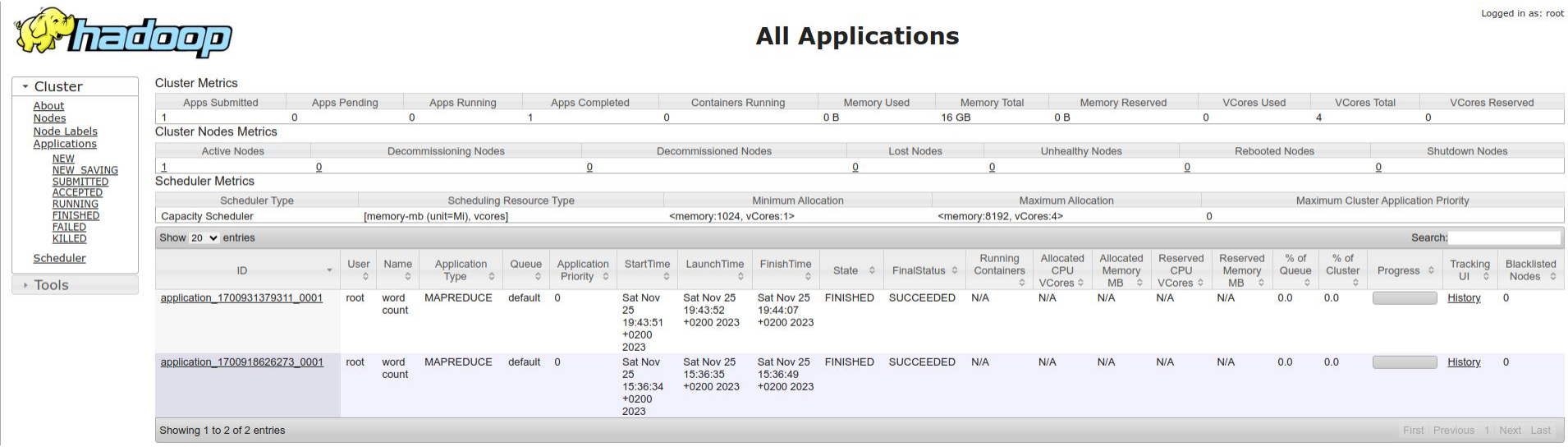
Maximum Allocation
<memory:8192, vCores:4>

Cluster Nodes Metrics

Active Nodes
1

Application Monitoring

- Visit the ResourceManager page through its ip and port.



Logged in as: root

All Applications

Cluster
About
Nodes
Node Labels
Applications
NEW
NEW_SAVING
SUBMITTED
ACCEPTED
RUNNING
FINISHED
FAILED
KILLED
Scheduler
Tools

Cluster Metrics

Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Memory Used	Memory Total	Memory Reserved	VCores Used	VCores Total	VCores Reserved
1	0	0	1	0	0 B	16 GB	0 B	0	4	0

Cluster Nodes Metrics

Active Nodes	Decommissioning Nodes	Decommissioned Nodes	Lost Nodes	Unhealthy Nodes	Rebooted Nodes	Shutdown Nodes
1	0	0	0	0	0	0

Scheduler Metrics

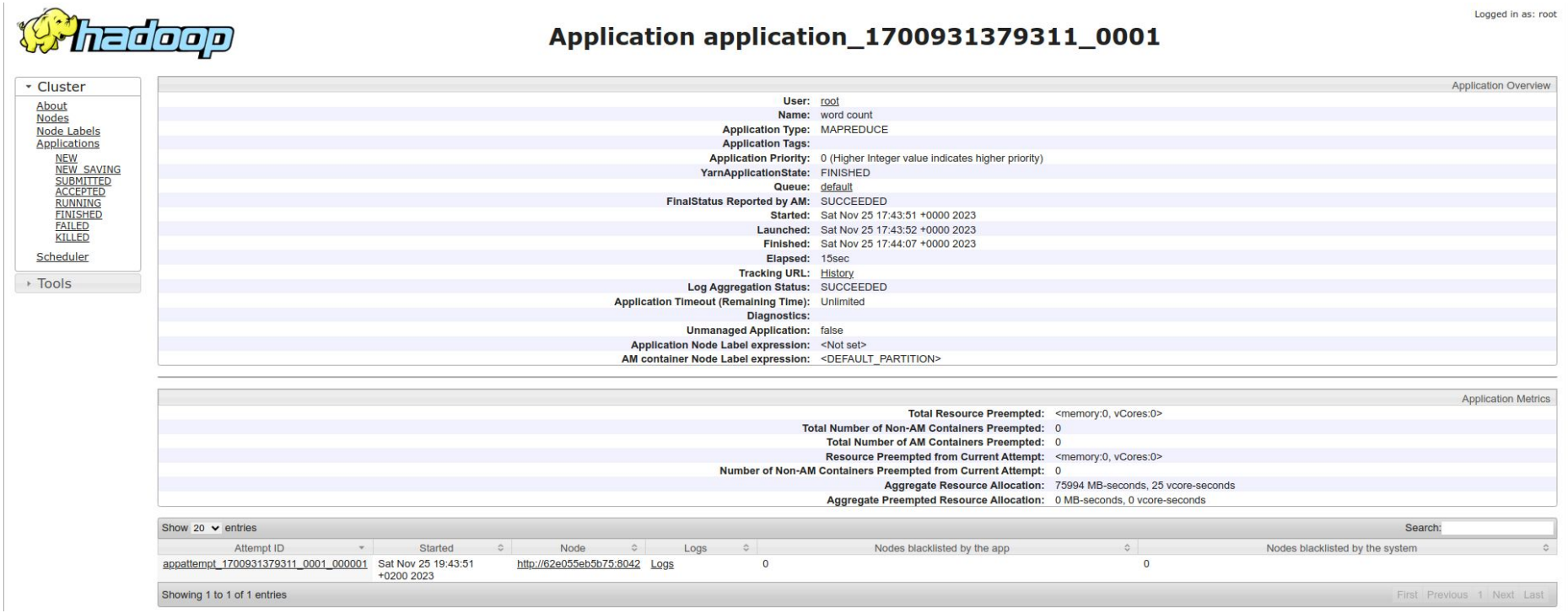
Scheduler Type	Scheduling Resource Type	Minimum Allocation	Maximum Allocation	Maximum Cluster Application Priority
Capacity Scheduler	[memory-mb (unit=Mi), vcores]	<memory:1024, vCores:1>	<memory:8192, vCores:4>	0

Show 20 entries Search:

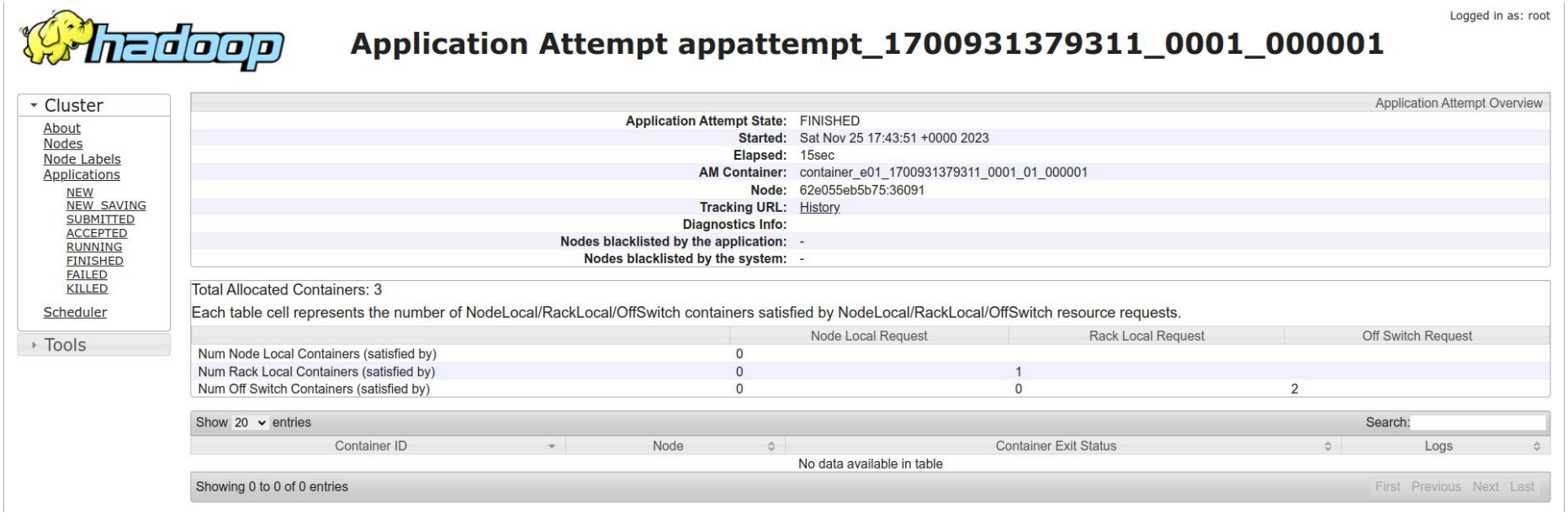
ID	User	Name	Application Type	Queue	Application Priority	StartTime	LaunchTime	FinishTime	State	FinalStatus	Running Containers	Allocated CPU VCores	Allocated Memory MB	Reserved CPU VCores	Reserved Memory MB	% of Queue	% of Cluster	Progress	Tracking UI	Blacklisted Nodes
application_1700931379311_0001	root	word count	MAPREDUCE	default	0	Sat Nov 25 19:43:51 +0200 2023	Sat Nov 25 19:43:52 +0200 2023	Sat Nov 25 19:44:07 +0200 2023	FINISHED	SUCCEEDED	N/A	N/A	N/A	N/A	N/A	0.0	0.0		History	0
application_1700918626273_0001	root	word count	MAPREDUCE	default	0	Sat Nov 25 15:36:34 +0200 2023	Sat Nov 25 15:36:35 +0200 2023	Sat Nov 25 15:36:49 +0200 2023	FINISHED	SUCCEEDED	N/A	N/A	N/A	N/A	N/A	0.0	0.0		History	0

Showing 1 to 2 of 2 entries First Previous 1 Next Last

Application Monitoring



Application Monitoring



Application Logs using Terminal

```
2023-11-25 13:36:47,799 INFO [RMCommunicator Allocator] org.apache.hadoop.mapreduce.v2.app.rm.RMContainerAllocator: Got allocated containers 1
2023-11-25 13:36:47,799 INFO [RMCommunicator Allocator] org.apache.hadoop.mapreduce.v2.app.rm.RMContainerAllocator: Assigned to reduce
2023-11-25 13:36:47,800 INFO [RMCommunicator Allocator] org.apache.hadoop.mapreduce.v2.app.rm.RMContainerAllocator: Assigned container container_1700918626273_0001_01_000003 to attempt_1700918626273_0001_r_000000_0
2023-11-25 13:36:47,800 INFO [RMCommunicator Allocator] org.apache.hadoop.mapreduce.v2.app.rm.RMContainerAllocator: After Scheduling: PendingReds:0 ScheduledMaps:0 ScheduledReds:0 AssignedMaps:0 AssignedReds:1 CompletedMaps:1 CompletedReds:0 ContAlloc:2 ContRel:0 HostLocal:0 RackLocal:1
```

```
2023-11-25 13:36:49,657 INFO [AsyncDispatcher event handler] org.apache.hadoop.mapreduce.v2.app.job.impl.JobImpl: Num completed Tasks: 2
2023-11-25 13:36:49,657 INFO [AsyncDispatcher event handler] org.apache.hadoop.mapreduce.v2.app.job.impl.JobImpl: job_1700918626273_0001Job Transitioned from RUNNING to COMMITTING
2023-11-25 13:36:49,658 INFO [CommitterEvent Processor #1] org.apache.hadoop.mapreduce.v2.app.commit.CommitterEventHandler: Processing the event EventType: JOB_COMMIT
2023-11-25 13:36:49,682 INFO [AsyncDispatcher event handler] org.apache.hadoop.mapreduce.v2.app.job.impl.JobImpl: Calling handler for JobFinishedEvent
```

```
2023-11-25 13:36:50,829 INFO [Thread-73] org.apache.hadoop.mapreduce.v2.app.MRAppMaster: Deleting staging directory hdfs://namenode:9000 /tmp/hadoop-yarn/staging/root/.staging/job_1700918626273_0001
2023-11-25 13:36:50,838 INFO [Thread-73] org.apache.hadoop.ipc.Server: Stopping server on 40151
2023-11-25 13:36:50,838 INFO [IPC Server listener on 0] org.apache.hadoop.ipc.Server: Stopping IPC Server listener on 0
2023-11-25 13:36:50,839 INFO [TaskHeartbeatHandler PingChecker] org.apache.hadoop.mapreduce.v2.app.TaskHeartbeatHandler: TaskHeartbeatHandler thread interrupted
```

Lab Task

- Implement the Inverted Index using MapReduce.
- Use the Hadoop Cluster using Docker Compose
- Deliverables:
  - ONE ZIP file contains:
    - The Java class contains Driver, Mapper, and Reducer.
    - The output file from HDFS.
    - Screenshot for the Application Logs from the ResourceManager Page.
- Grading:
  - 2 Marks for Driver.
  - 4 Marks for Mapper.
  - 4 Marks for Reducer.
- Deadline: April, 1st 2024 at 11:45 PM.

Thanks