

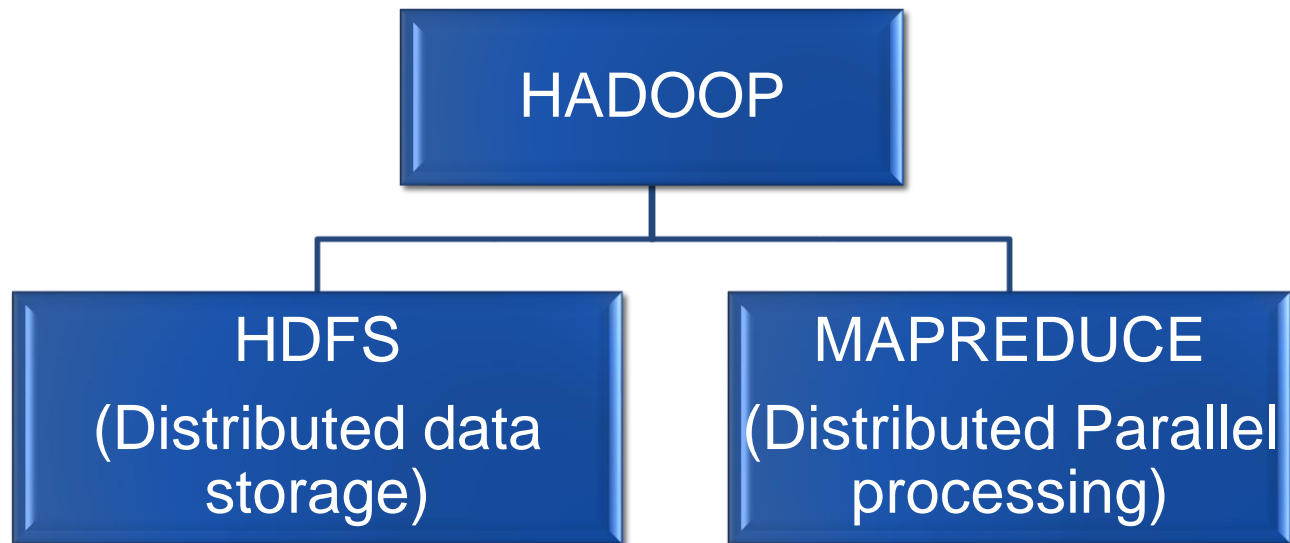
CLOUDERA

A Quick Overview

by Suchitra Jayaprakash
suchitra@cmi.ac.in

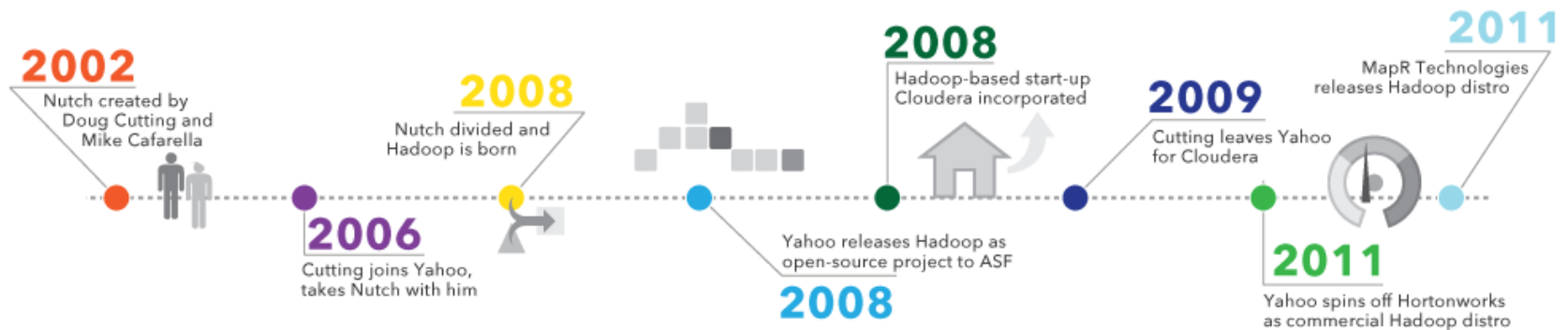
Apache Hadoop

- Hadoop is open source software framework used for processing data on distributed commodity computing environment.

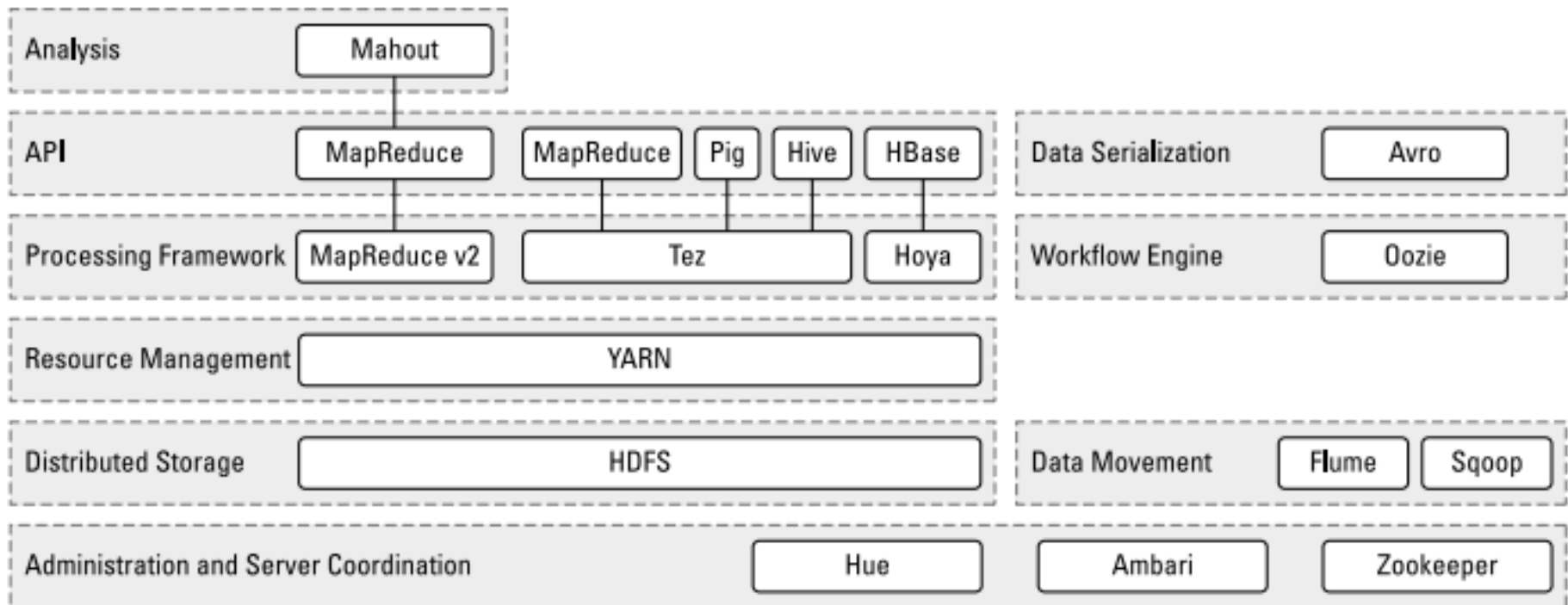


Apache Hadoop

- It is a java based software managed by Apache Software Foundation.
- Hadoop is designed to scale up from single server to thousands of machines.
- Doug Cutting & Mike Cafarella are co-founders of Hadoop. It is based on google's white paper on Google File System & mapreduce.



Hadoop Ecosystem



(source: Hadoop for Dummies)

HADOOP DISTRIBUTION

- Customisation for industry needs resulted in emergence of commercial distribution.
- Base version Apache Hadoop + features (UI , Security , Monitoring , logging, Support).
- Top Vendors offering Big Data Hadoop solution :

- Cloudera

CLOUDERA

- Hortonworks



- MapR



- Amazon Web Services Elastic MapReduce Hadoop Distribution



- Microsoft Azure's HDInsight -Cloud based Hadoop Distribution



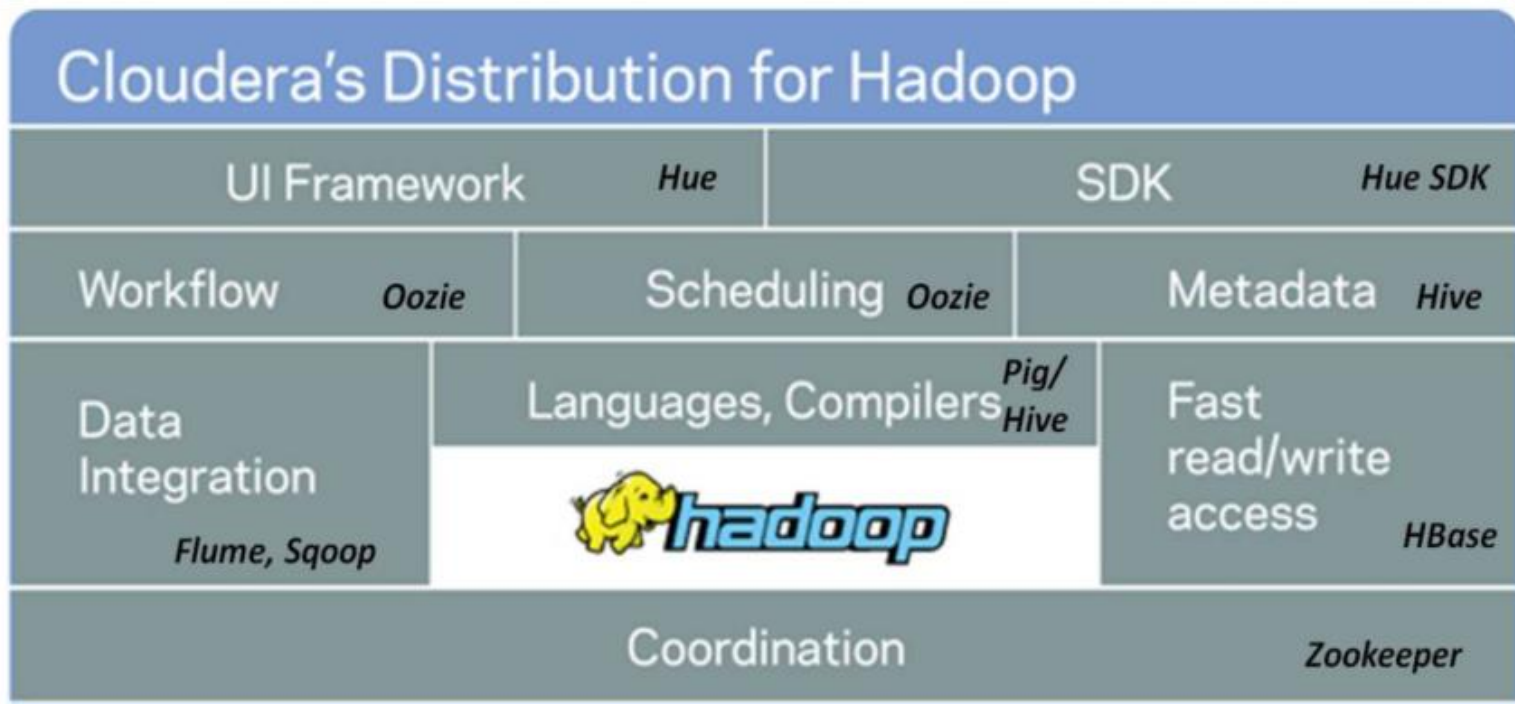
- IBM InfoSphere Insights



CLOUDERA

- Founded in 2008 by three engineers from Google, Yahoo! and Facebook (Christophe Bisciglia, Amr Awadallah and Jeff Hammerbacher).
- Major code contributor of Apache Hadoop ecosystem.
- First company to develop and distribute Apache Hadoop based software in March 2009.
- Additional feature includes user interface, security, interface for third party application integration.
- Offers customer support for installing , configuring , optimising Cloudera distribution through its enterprise subscription service.
- Provides a proprietary Cloudera Manager for easy installation , monitoring & trouble shooting.
- In 2016, Cloudera was ranked #5 on the Forbes Cloud 100 list
(source: Cloudera wiki)

CLOUDERA DISTRIBUTION



An illustration of Cloudera's open-source Hadoop distribution (source: cloudera website).

CLOUDERA QUICKSTART

- Cloudera QuickStart VM is a sandbox environment of CDH.
- It gives a hands-on experience with CDH for demo and self-learning purposes.
- CDH deployed via Docker containers or VMs, are not intended for production use. Latest version is QuickStarts for CDH 5.13.
- System Requirement: Cloudera's 64-bit VMs require a 64-bit host OS and a virtualization product that can support a 64-bit guest.
- The amount of RAM required by the VM (separate from system RAM) varies by the run-time option you choose:

CDH and Cloudera Manager Version	RAM Required by VM
CDH 5 (default)	4+ GiB*
Cloudera Express	8+ GiB*
Cloudera Enterprise (trial)	12+ GiB*

*Minimum recommended memory.

(source: Cloudera website)

Quiz

Q) Which of the following is false?

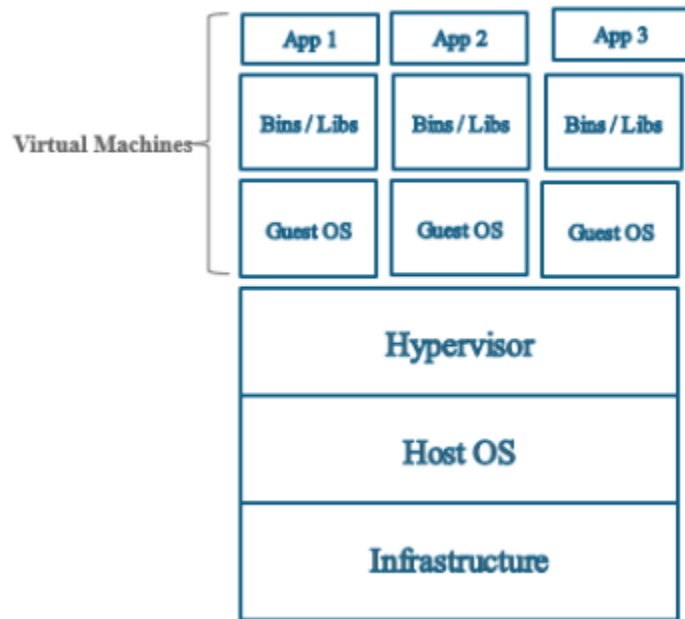
- A. Cloudera products and solutions enable you to deploy and manage Apache Hadoop and related projects.
- B. Cloudera QuickStart VM is a sandbox environment of CDH.
- C. CDH contains all the products and frameworks belonging to the hadoop ecosystem.
- D. Hadoop is open source software framework used for processing data on distributed commodity hardware.

DEPLOYMENT MODES - DOCKER

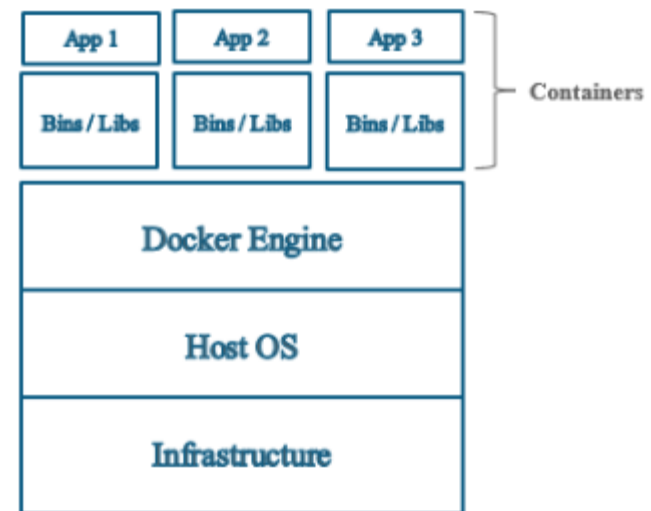


- Docker is an open source tool that uses containers to create, deploy, and manage distributed applications.
- Developers use containers to create packages for applications that include all libraries that are needed to run the application in isolation.

DEPLOYMENT MODES : VM vs DOCKER



Virtual Machine / Virtual Box



Docker Container

- Virtual machine has its guest operating system above the host operating system.
- Docker containers share the host operating system.

Virtual Machine vs Docker Container



```
MINGW64/c:/Program Files/Docker Toolbox
Setting OOOZIE_HTTP_HOSTNAME: quickstart.cloudera
Setting OOOZIE_HTTP_PORT: 11000
Setting OOOZIE_ADMIN_PORT: 11001
Using OOOZIE_HTTPS_PORT: 11443
Setting OOOZIE_BASE_URL: http://quickstart.cloudera:11000/oozie
Using CATALINA_BASE: /var/lib/oozie/tomcat-deployment
Setting OOOZIE_HTTPS_KEYSTORE_FILE: /var/lib/oozie/.keystore
Using OOOZIE_HTTPS_KEYSTORE_PASS: password
Setting OOOZIE_INSTANCE_ID: quickstart.cloudera
Setting CATALINA_OUT: /var/log/oozie/catalina.out
Using CATALINA_PID: /var/run/oozie/oozie.pid
Using CATALINA_OPTS: -Doozie.https.port=11443 -Doozie.https.keystore.pass=password -Xmx1024m -Dderby.stream.error.file=/var/log/oozie/derby.log
Adding to CATALINA_OPTS: -Doozie.home.dir=/usr/lib/oozie -Doozie.config.dir=/etc/oozie/conf -Doozie.log.dir=/var/log/oozie -Doozie.data.dir=/var/lib/oozie -Doozie.instance.id=quickstart.cloudera -Doozie.config.file=oozie-site.xml -Doozie.log4j.file=oozie-log4j.properties -Doozie.log4j.reload=10 -Doozie.http.hostname=quickstart.cloudera -Doozie.admin.port=11001 -Doozie.http.port=11000 -Doozie.https.port=11443 -Doozie.base.url=http://quickstart.cloudera:11000/oozie -Doozie.https.keystore.file=/var/lib/oozie/.keystore -Doozie.https.keystore.pass=password -Djava.library.path=/usr/lib/hadoop/lib/native:/usr/lib/hadoop/lib/native
Using CATALINA_BASE: /var/lib/oozie/tomcat-deployment
Using CATALINA_HOME: /usr/lib/bigtop-tomcat
Using CATALINA_TMPDIR: /var/lib/oozie
Using JRE_HOME: /usr/java/jdk1.7.0_67-cloudera
Using CLASSPATH: /usr/lib/bigtop-tomcat/bin/bootstrap.jar
Using CATALINA_PID: /var/run/oozie/oozie.pid
Starting Solr server daemon: [ OK ]
Using CATALINA_BASE: /var/lib/solr/tomcat-deployment
Using CATALINA_HOME: /usr/lib/solr/..bigtop-tomcat
Using CATALINA_TMPDIR: /var/lib/solr/
Using JRE_HOME: /usr/java/jdk1.7.0_67-cloudera
Using CLASSPATH: /usr/lib/solr/..bigtop-tomcat/bin/bootstrap.jar
Using CATALINA_PID: /var/run/solr/solr.pid
Started Impala Catalog Server (catalogd): [ OK ]
Started Impala Server (impalad): [ OK ]
[root@quickstart ~]#
```

QUICKSTART : DOCKER INSTALL

- The Cloudera Docker image is a single-host deployment of the Cloudera open-source distribution.
- Single Node Hadoop Cluster has only a single machine
 - DataNode, NameNode run on the same machine
- Multi-Node Hadoop Cluster will have more than one machine
 - DataNode, NameNode run on different machines.

QUICKSTART : DOCKER INSTALL

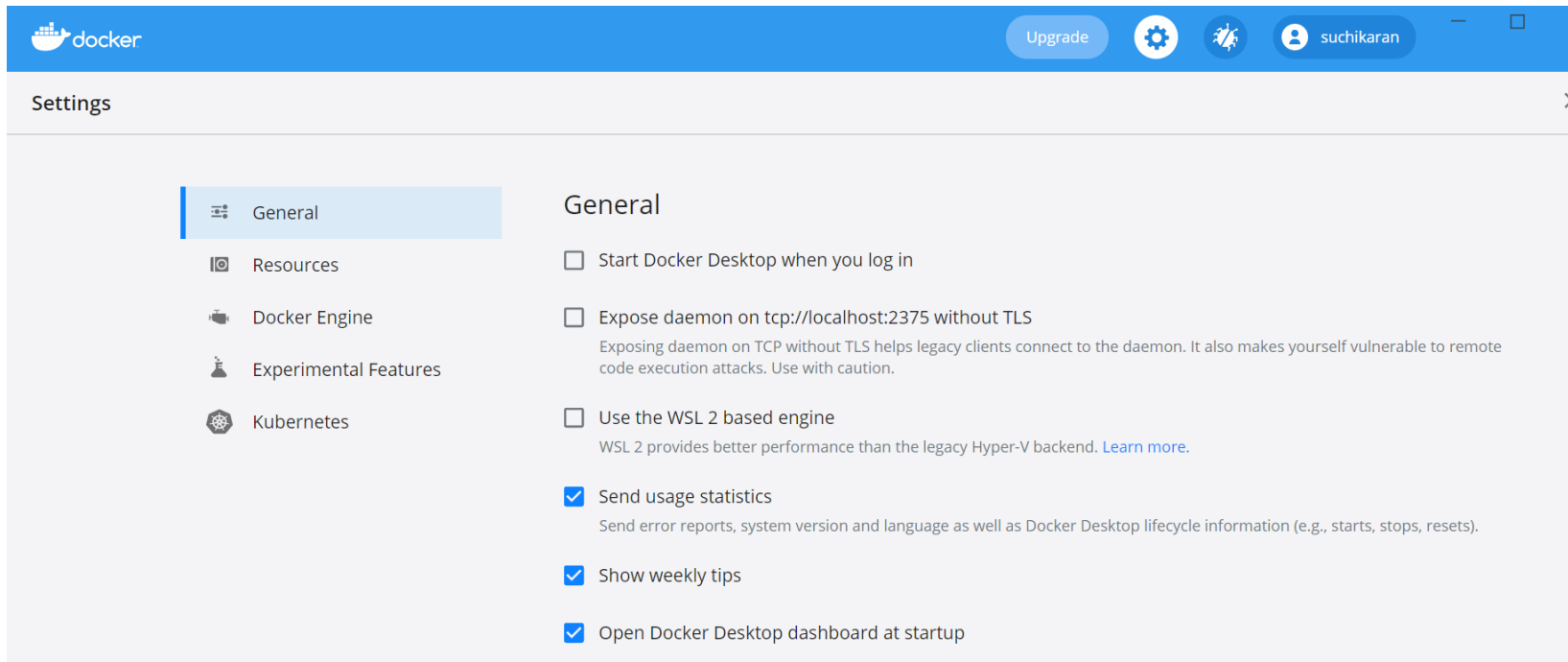
- **Installation Steps for Windows :**

- 1. Install Docker :**

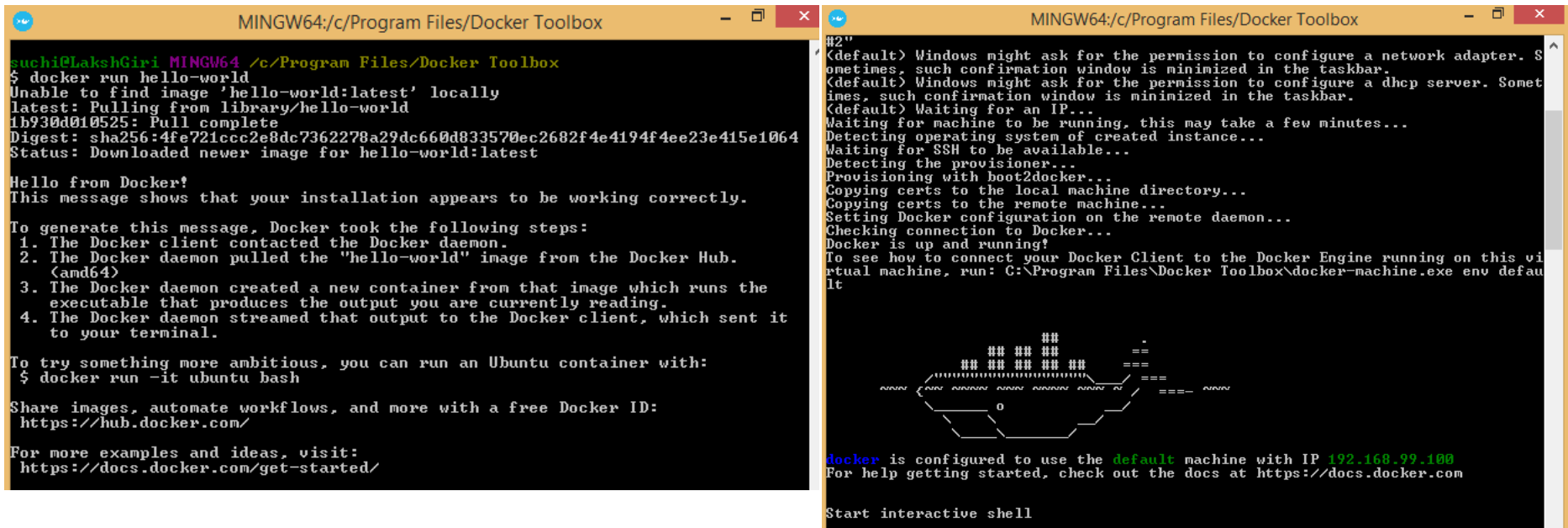
- Sign up to <https://docs.docker.com/>
- Follow instructions at <https://docs.docker.com/docker-for-windows/install/>
- For Windows 10 64-bit Home , Pro, Enterprise, or Education (Build 15063 or later) : Install Docker Desktop.
- For Other Windows OS :
Install Docker Toolbox (refer below link for instructions.
https://docs.docker.com/toolbox/toolbox_install_windows/)

QUICKSTART : DOCKER INSTALL

- Don't select WSL2 while installing docker. Cloudera Quick start VM is not compatible.



QUICKSTART : DOCKER INSTALL



The image shows two screenshots of a Windows terminal window titled "MINGW64:/c/Program Files/Docker Toolbox".

The left screenshot shows the output of the command `docker run hello-world`. It displays the process of pulling the `hello-world:latest` image from Docker Hub, showing the image ID `1b930d010525` and the digest `sha256:4fe721ccc2e8dc7362278a29dc660d833570ec2682f4e4194f4ee23e415e1064`. The output then shows "Hello from Docker!" and a message indicating that the installation appears to be working correctly. It lists the steps taken: 1. The Docker client contacted the Docker daemon. 2. The Docker daemon pulled the "hello-world" image from the Docker Hub. 3. The Docker daemon created a new container from that image which runs the executable that produces the output you are currently reading. 4. The Docker daemon streamed that output to the Docker client, which sent it to your terminal. It also suggests trying something more ambitious by running an Ubuntu container with `docker run -it ubuntu bash`. At the bottom, it provides a link to share images and automate workflows with a free Docker ID: <https://hub.docker.com/>, and a link for more examples and ideas: <https://docs.docker.com/get-started/>.

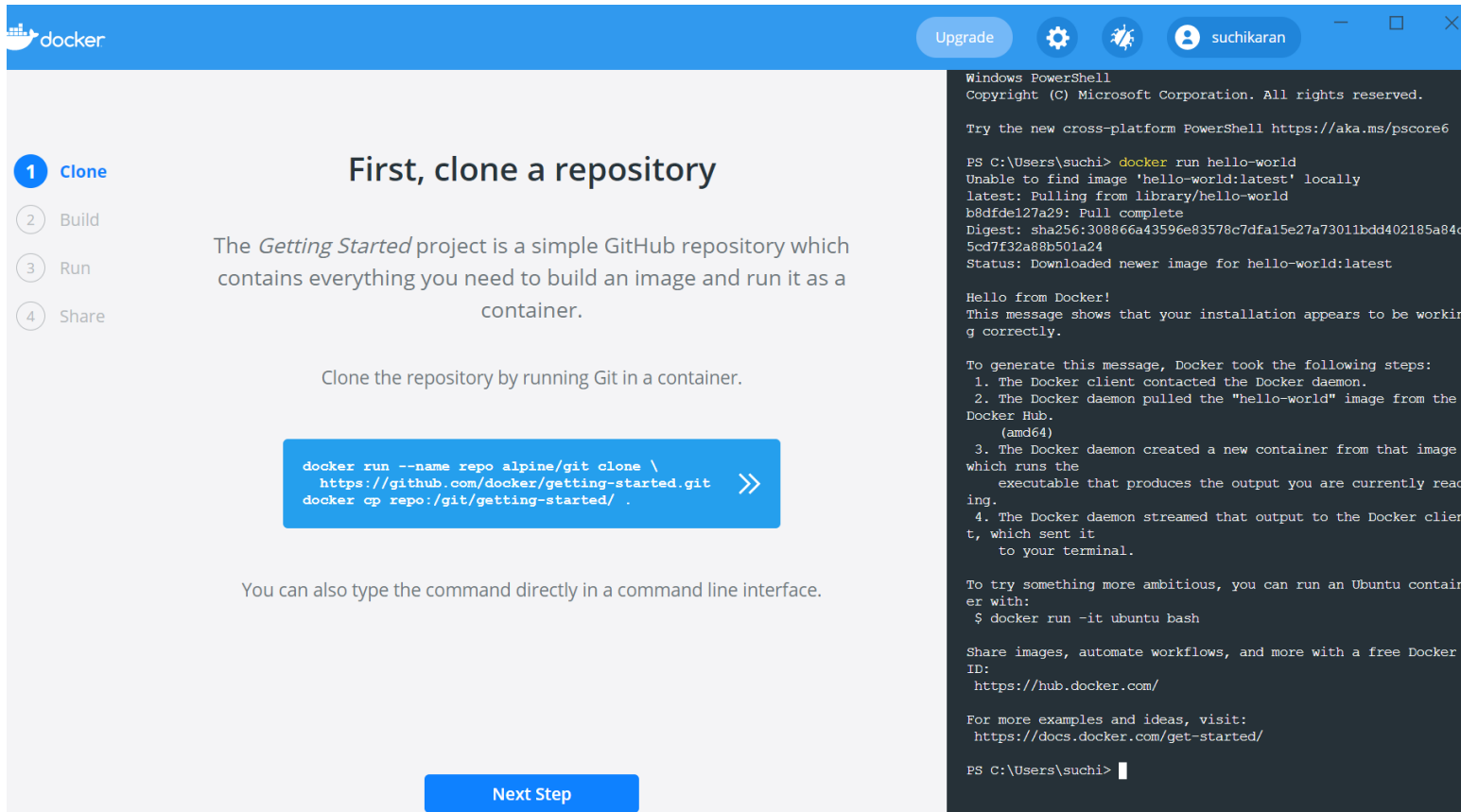
The right screenshot shows the output of the command `docker-machine env default`. It displays the process of creating a new Docker machine named `default`. The output includes the following steps: (default) Windows might ask for the permission to configure a network adapter. Sometimes, such confirmation window is minimized in the taskbar. (default) Windows might ask for the permission to configure a dhcp server. Sometimes, such confirmation window is minimized in the taskbar. (default) Waiting for an IP... Waiting for machine to be running, this may take a few minutes... Detecting operating system of created instance... Waiting for SSH to be available... Detecting the provisioner... Provisioning with boot2docker... Copying certs to the local machine directory... Copying certs to the remote machine... Setting Docker configuration on the remote daemon... Checking connection to Docker... Docker is up and running! To see how to connect your Docker Client to the Docker Engine running on this virtual machine, run: C:\Program Files\Docker Toolbox\docker-machine.exe env default. Below this, there is a diagram of a Docker machine architecture. At the bottom, it states: `docker` is configured to use the `default` machine with IP `192.168.99.100`. For help getting started, check out the docs at <https://docs.docker.com>. It also shows the command `Start interactive shell`.

- To check docker installation is proper , type below command in docker terminal.

docker run hello-world

- If you get above output in the terminal then docker installation is fine.

QUICKSTART : DOCKER INSTALL



1 Clone

First, clone a repository

The *Getting Started* project is a simple GitHub repository which contains everything you need to build an image and run it as a container.

Clone the repository by running Git in a container.

```
docker run --name repo alpine/git clone \
https://github.com/docker/getting-started.git
docker cp repo:/git/getting-started/ .
```

You can also type the command directly in a command line interface.

Next Step

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\suchi> docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
b8dfde127a29: Pull complete
Digest: sha256:308866a43596e83578c7dfa15e27a73011bdd402185a84c5cd7f32a88b501a24
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/

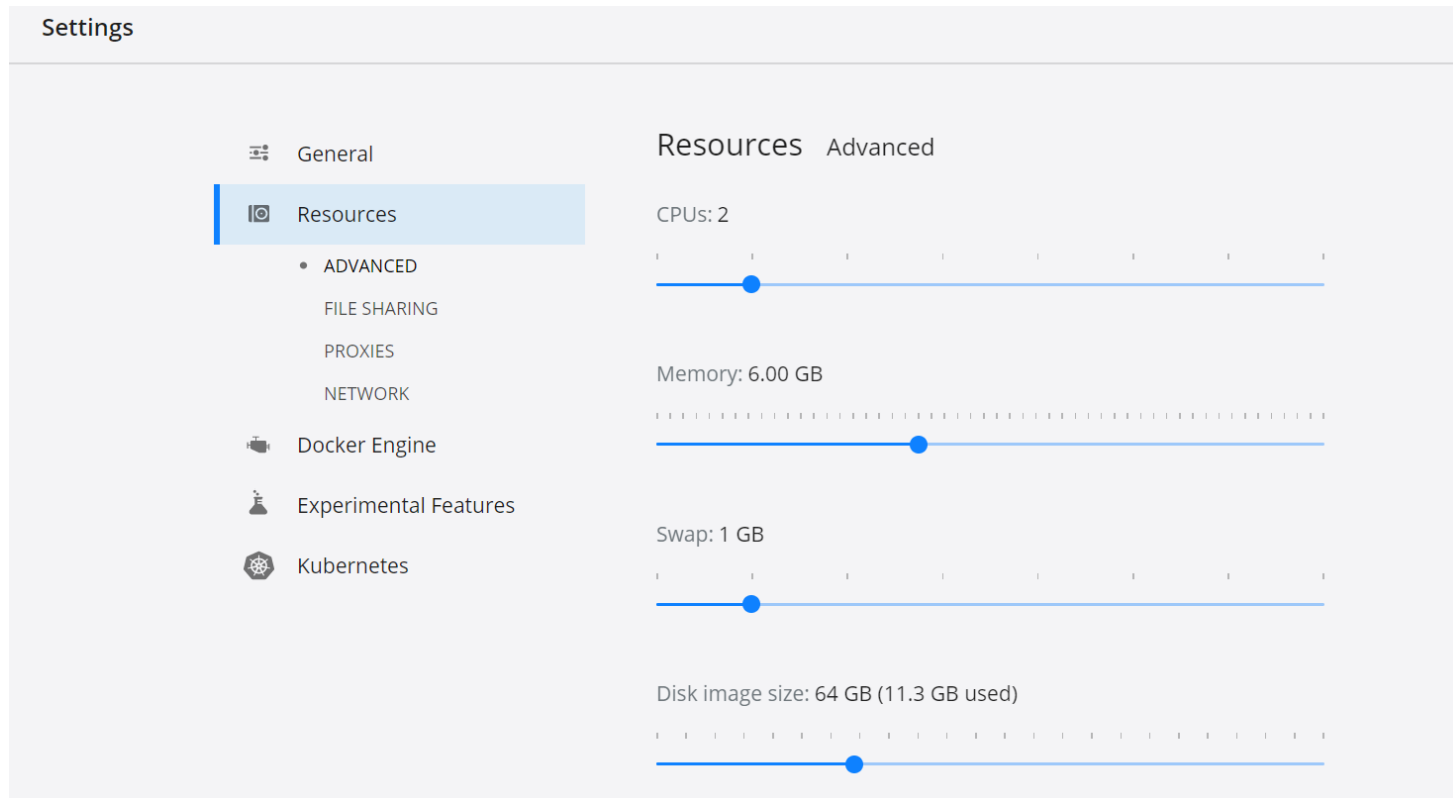
PS C:\Users\suchi>
```

- Docker for Desktop output
- For windows 10 : Run docker command in powershell or command prompt

QUICKSTART : DOCKER INSTALL

2. Docker Desktop: Update Docker memory

Under setting select Resources and update CPU & Memory as mentioned below:



QUICKSTART : DOCKER INSTALL

2. Docker Toolbox : Update Docker memory (optional)

2. Create a new VM with 1 CPUs and 4GB of memory (recommended).

3. Run the following command in docker terminal:

- Remove the default vm.

docker-machine rm default

- Re-create the default vm.

docker-machine create -d virtualbox --virtualbox-cpu-count=1 --virtualbox-memory=4096 --virtualbox-disk-size=50000 default

options	Description
--virtualbox-cpu-count	number of cpus
--virtualbox-memory	amount of RAM
-virtualbox-disk-size	amount of disk space

QUICKSTART : DOCKER INSTALL

3. Install Cloudera Quickstart:

Type following command in the docker terminal to import Cloudera Quickstart image from Docker Hub:

docker pull cloudera/quickstart:latest

(refer link <https://hub.docker.com/r/cloudera/quickstart>)

```
$ docker pull cloudera/quickstart:latest
latest: Pulling from cloudera/quickstart
Image docker.io/cloudera/quickstart:latest uses outdated schema1 manifest format
Please upgrade to a schema2 image for better future compatibility. More information at https://docs.docker.com/registry/spec/deprecated-schema-v1/
d00652ce734: Downloading 39.28MB/4.444GB
```

Cloudera quickstart download will take a while to complete. After download is complete , type following in terminal :

docker images

```
suchi@LakshGiri MINGW64 /c/Program Files/Docker Toolbox
$ docker images
REPOSITORY          TAG          IMAGE ID          CREATED
SIZE
cloudera/quickstart latest       4239cd2958c6      3 years ago
6.34GB
```

QUICKSTART : DOCKER INSTALL

4. Run Cloudera Quickstart container

- Click on “Docker Quickstart Terminal” Icon and Type below command in docker terminal to start Cloudera Quickstart

```
docker run --hostname=quickstart.cloudera --privileged=true -t -i -p 8888:8888 -p 8080:8080 -p 8088:8088 -p 7180:7180 -p 50070:50070 cloudera/quickstart /usr/bin/docker-quickstart
```

Options	Required	Description
--hostname=quickstart.cloudera	Yes	Pseudo-distributed configuration assumes this as hostname.
--privileged=true	Yes	For HBase, MySQL-backed Hive metastore, Hue, Oozie, Sentry, and Cloudera Manager.
-t	Yes	Allocate a pseudoterminal. Once services are started, a Bash shell takes over. This switch starts a terminal emulator to run the services.
-i	Yes	Enable interactive terminal i.e. If you want to use the terminal, either immediately or connect to the terminal later.
--publish-all=true	No	opens up all the host ports to the docker ports
-p 8888	Yes - Recommended	Map the Hue port in the guest to port on the host.
-p [PORT]	No	Map any other ports in the guest to port on the host.
cloudera/quickstart	Yes	Name of image which run as new container
/usr/bin/docker-quickstart	Yes	Start all CDH services, and then run a Bash shell.

QUICKSTART : DOCKER INSTALL

List of common ports used in Cloudera :

Port	Purpose
8888	Hue web interface
50070	Name node web interface
8088	job tracker :- yarn
7180	Cloudera manager
80	Cloudera examples

5. Host – Guest port mapping

- Open new docker terminal & type below command.

docker ps

```
$ docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED
STATUS            PORTS              NAMES
b636a46d51d0      cloudera/quickstart "/usr/bin/docker-qui 4 minutes ago
Up 4 minutes      0.0.0.0:7180->7180/tcp, 0.0.0.0:8088->8088/tcp, 0.0.0.
0:8888->8888/tcp, 0.0.0.0:50070->50070/tcp, 0.0.0.0:8080->80/tcp crazy_proskur
iakova
```

- Copy the docker container ID.
- Type below to check memory allocation

docker stats [CONTAINER ID]

```
CONTAINER ID        NAME               CPU %               MEM USAGE / LIMIT
MEM %               NET I/O            BLOCK I/O           PIDS
cde59eb01eeb       goofy_williamson   9.39%              3.362GiB / 3.856GiB
87.19%             2.39kB / 4.33kB    1.48GB / 59.4MB    1328
```

QUICKSTART : DOCKER INSTALL

- Type below command and get see which Host port Hue and YARN are working.

docker inspect [CONTAINER ID]

- YARN is working on port
8088 inside the docker machine
8088 outside on host machine

Note : in case of docker tool box, host machine is mapped to ip address 192.168.99.100. Use url

<http://192.168.99.100:50070/>

For other docker install use localhost

<http://localhost:50070/>

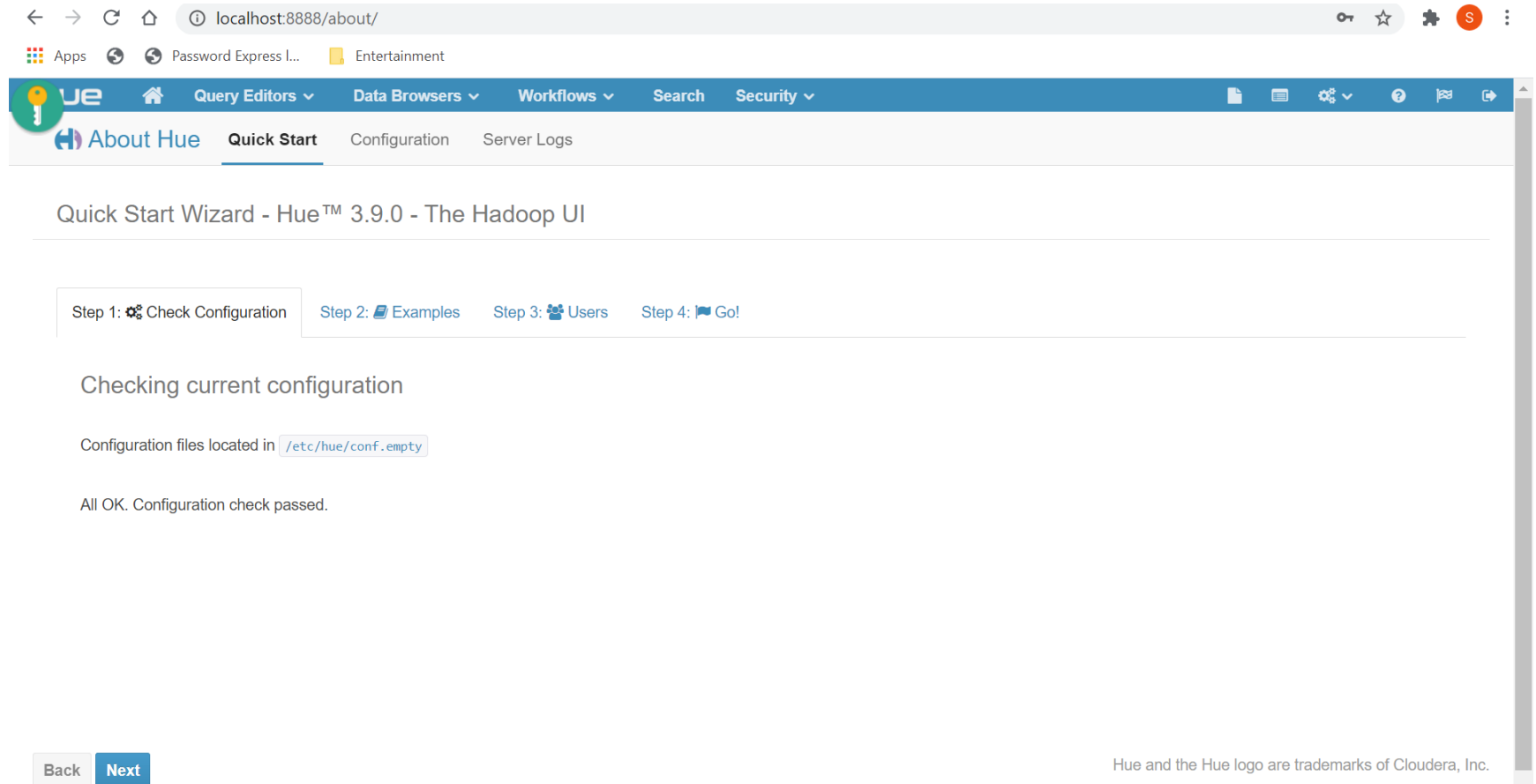
- Installation Steps for Ubuntu** : <https://medium.com/@dataakkadian/how-to-install-and-running-cloudera-docker-container-on-ubuntu-b7c77f147e03>

```
"Ports": {  
  "50070/tcp": [  
    {  
      "HostIp": "0.0.0.0",  
      "HostPort": "50070"  
    }  
  ],  
  "7180/tcp": [  
    {  
      "HostIp": "0.0.0.0",  
      "HostPort": "7180"  
    }  
  ],  
  "80/tcp": [  
    {  
      "HostIp": "0.0.0.0",  
      "HostPort": "8080"  
    }  
  ],  
  "8088/tcp": [  
    {  
      "HostIp": "0.0.0.0",  
      "HostPort": "8088"  
    }  
  ],  
  "8888/tcp": [  
    {  
      "HostIp": "0.0.0.0",  
      "HostPort": "8888"  
    }  
  ]  
}
```

QUICKSTART : DOCKER INSTALL

HUE- <http://localhost:8888/>

Default username / password : cloudera / cloudera



The screenshot shows a web browser window with the address bar displaying `localhost:8888/about/`. The browser's address bar includes navigation buttons (back, forward, refresh, home) and icons for key, star, puzzle, and a red circle with an 'S'. Below the address bar, there are search and navigation icons, and a list of bookmarks including 'Apps', 'Password Express I...', and 'Entertainment'.

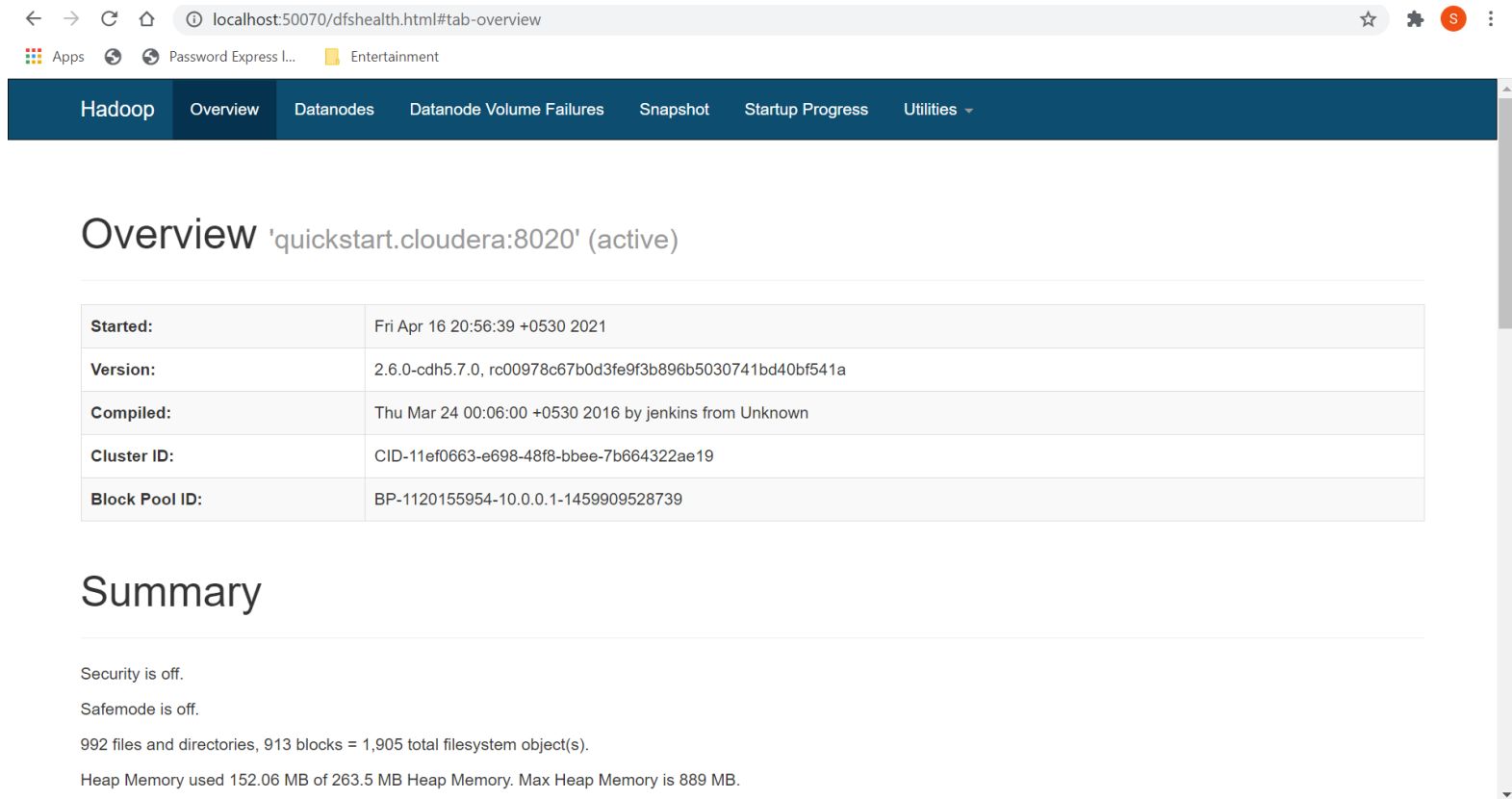
The Hue application interface is visible, featuring a blue header bar with the Hue logo and navigation links: 'Query Editors', 'Data Browsers', 'Workflows', 'Search', and 'Security'. Below the header, there is a sub-header with 'About Hue', 'Quick Start' (highlighted), 'Configuration', and 'Server Logs'.

The main content area displays the 'Quick Start Wizard - Hue™ 3.9.0 - The Hadoop UI'. It includes a progress bar with four steps: 'Step 1: Check Configuration' (active), 'Step 2: Examples', 'Step 3: Users', and 'Step 4: Go!'. Below the progress bar, the text 'Checking current configuration' is shown, followed by 'Configuration files located in `/etc/hue/conf.empty`'. A message at the bottom states 'All OK. Configuration check passed.'

At the bottom of the page, there are 'Back' and 'Next' buttons. In the bottom right corner, a small text note reads: 'Hue and the Hue logo are trademarks of Cloudera, Inc.'

QUICKSTART : DOCKER INSTALL

Name Node - <http://localhost:50070/>



The screenshot shows a web browser window displaying the Hadoop NameNode Overview page. The browser's address bar shows the URL `localhost:50070/dfshealth.html#tab-overview`. The page has a dark blue navigation bar with tabs: **Hadoop**, **Overview** (selected), **Datanodes**, **Datanode Volume Failures**, **Snapshot**, **Startup Progress**, and **Utilities**. The main content area is titled "Overview 'quickstart.cloudera:8020' (active)". Below the title is a table with the following information:

Started:	Fri Apr 16 20:56:39 +0530 2021
Version:	2.6.0-cdh5.7.0, rc00978c67b0d3fe9f3b896b5030741bd40bf541a
Compiled:	Thu Mar 24 00:06:00 +0530 2016 by jenkins from Unknown
Cluster ID:	CID-11ef0663-e698-48f8-bbee-7b664322ae19
Block Pool ID:	BP-1120155954-10.0.0.1-1459909528739

Below the table is a section titled "Summary". The summary text includes:


- Security is off.
- Safemode is off.
- 992 files and directories, 913 blocks = 1,905 total filesystem object(s).
- Heap Memory used 152.06 MB of 263.5 MB Heap Memory. Max Heap Memory is 889 MB.

QUICKSTART : DOCKER INSTALL

Yarn page - <http://192.168.99.100:8088/>

← → ↺ 🏠 ⓘ Not secure | 192.168.99.100:8088/cluster

🌈 Apps 🔄



All Applications

▼ Cluster

[About](#)
[Nodes](#)
[Applications](#)
 [NEW](#)
 [NEW_SAVING](#)
 [SUBMITTED](#)
 [ACCEPTED](#)
 [RUNNING](#)
 [FINISHED](#)
 [FAILED](#)
 [KILLED](#)
[Scheduler](#)

Cluster Metrics

Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Memory Used	Memory Total	Memory Reserved	VCores Used	VCores Total	VCores Reserved	Active Nodes	Decommission Nodes
0	0	0	0	0	0 B	8 GB	0 B	0	8	0	1	0

User Metrics for dr.who

Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Containers Pending	Containers Reserved	Memory Used	Memory Pending	Memory Reserved	VCores Used
0	0	0	0	0	0	0	0 B	0 B	0 B	0

Show 20 ▼ entries

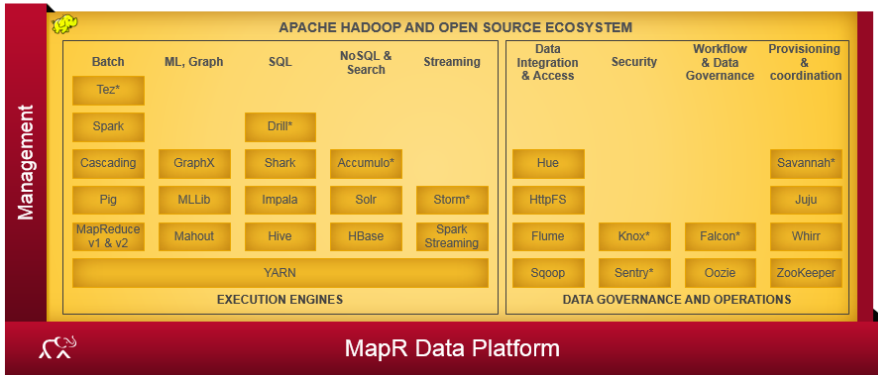
ID ▼	User ↕	Name ↕	Application Type ↕	Queue ↕	StartTime ↕	FinishTime ↕	State ↕	FinalStatus ↕	Running Containers ↕	Allocated CPU VCoers ↕	Allocated Memory MB ↕
No data available in table											

Showing 0 to 0 of 0 entries

Yarn is resource management layer of Apache Hadoop ecosystem.

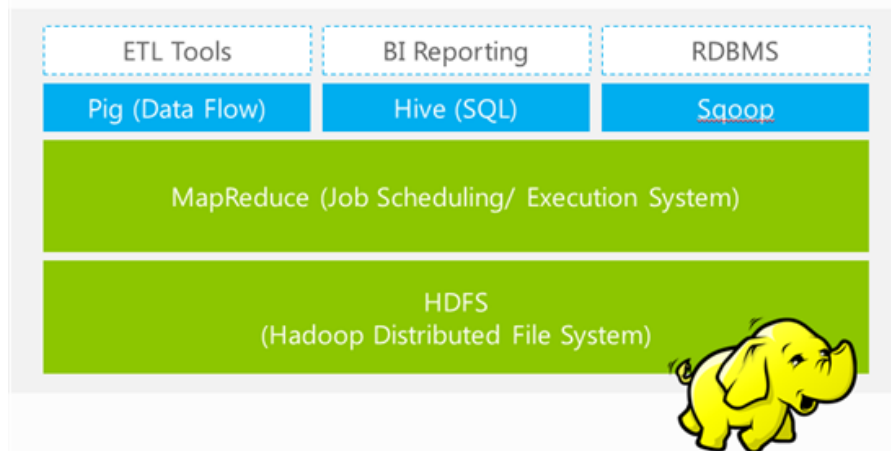
Other Vendors

MapR Distribution for Hadoop

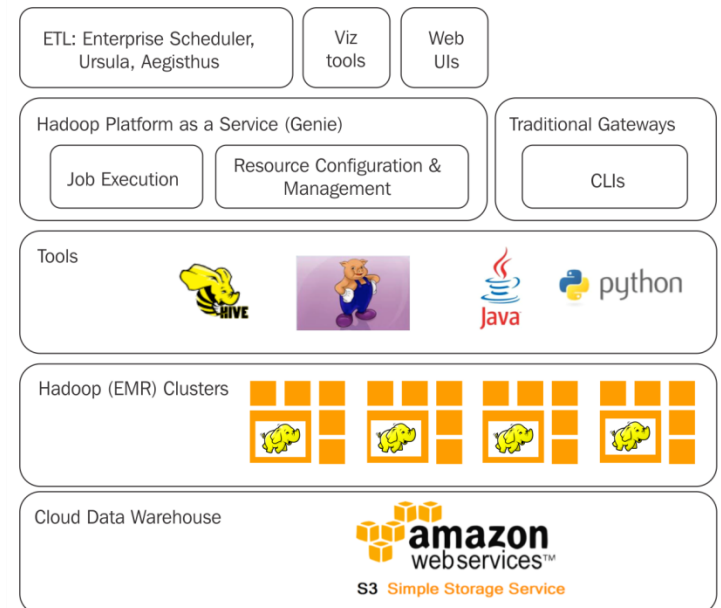


Windows Azure HDInsight

The Hadoop Ecosystem



AWS EMR



THANK YOU