# Lab0: Installation d'un Cluster Hadoop avec docker

Pour rappel, **Apache Hadoop** est un framework open-source employer pour le stockage et le traitement de grandes quantités de données. Le framework de base est composé des modules:
**Hadoop Common** - contient les bibliothèques nécessaires aux autres modules Hadoop;
**Hadoop Distributed File System (HDFS)** - un système de fichiers distribué qui stocke les données sur des machines, fournissant une bande passante globale très élevée à travers le cluster;
**Hadoop MapReduce** : une implémentation du modèle de programmation MapReduce pour le traitement de données à grande échelle.
**Hadoop YARN** - une plate-forme chargée de gérer les ressources informatiques dans les clusters et de les utiliser pour planifier les applications des utilisateurs;

**L'objectif de ce TP est de :**

- Installer un cluster hadoop avec docker
- Se familiariser avec les commandes HDFS

**Docker** est un service de gestion de conteneurs. Il permet d'isoler les applications dans des conteneurs avec des instructions indiquant exactement ce dont elles ont besoin pour survivre et pouvant être facilement transférées d'une machine à l'autre. Dans ce TP, nous allons utiliser Docker pour démarrer notre cluster Hadoop à base d'une image.
Nous allons utiliser trois conteneurs représentant respectivement un nœud maître (**Namenode**) et deux nœuds esclaves (**Datanodes**).

# I. Installation Cluster Hadoop

## 1. Installation docker

- Mettre à jour le système

```
sudo apt-get update
```

- i**nstaller docker** via la commande suivante

```
sudo apt-get install docker.io
```

- ou bien suivre les instructions du ***Site officiel***
- Pour **vérifier** si docker a été correctement installé

```
docker version
```

## 2. Télécharger l'image hadoop-spark-cluster

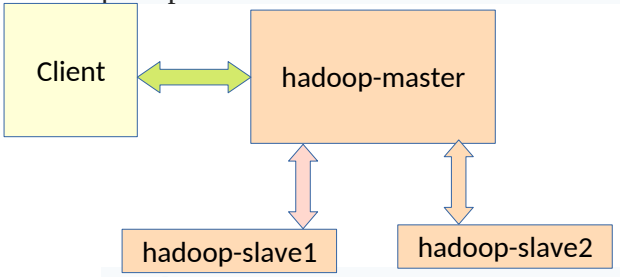
- Sur le terminal importer la dernière version de l'image hadoop-spark-cluster depuis dockerHub

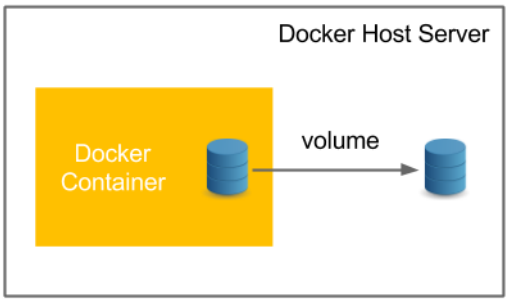
```
docker pull  yassern1/hadoop-spark-jupyter:1.0.3
```

- pour afficher la listes des images disponibles

```
docker images
```

- pour plus de détails sur les commandes docker ***documentation***




## 3. Création d'un volume de partage

*Créer un dossier **hadoop_project** pour l'échange de documents entre votre machine et le container (tout ce que vous aller y mettre sera visible dans le container)*

*sur l'explorateur ou bien sur la ligne de commande*

*SOUS LINUX:* ***~/Documents/hadoop_project***

*SOUS WINDOWS:* ***C:/USERS/.../Documents/hadoop_project***

## 4. Création du cluster (trois conteneurs)

*Créer les trois conteneurs à partir de l'image téléchargée en réseau. Pour se faire :*

- *Créer d'abord un réseau qui permettra de relier les trois conteneurs:*

```
docker network create --driver=bridge hadoop
```

- *Créer et lancer les trois conteneurs*

***Conteneur 1 : hadoop-master***

```
docker run -itd -v ~/Documents/hadoop_project/:/shared_volume
--net=hadoop -p 9870:9870 -p 8088:8088 -p 7077:7077 -p
19888:19888 -p 8080:8080 -p 9000:9000 --name hadoop-master --
hostname hadoop-master yassern1/hadoop-spark-jupyter:1.0.3
```

***Conteneur 2 hadoop-slave1***

```
docker run -itd -p 8040:8042 --net=hadoop --name hadoop-slave1
--hostname hadoop-slave1 yassern1/hadoop-spark-jupyter:1.0.3
```

***Conteneur 3 hadoop-slave2***

```
docker run -itd -p 8041:8042 --net=hadoop --name hadoop-slave2
--hostname hadoop-slave2 yassern1/hadoop-spark-jupyter:1.0.3
```

***N.B :** vous pouvez utiliser **docker compose** pour gérer l'ensemble des conteneurs en une seule commande. pour se faire :*

- *Écrire un fichier* `docker-compose.yml` *pour configurer les services, réseaux et volumes requis. Utiliser ensuite les commandes suivantes :*
  - ***docker compose up -d** pour lancer l'ensemble des conteneurs*

- ***docker compose stop/start*** *pour arrêter et démarrer l'ensemble des conteneurs*
- ***docker compose down -v*** *pour supprimer les conteneurs*

## 5. Accéder au master

*Entrer dans le conteneur master pour commencer à l'utiliser*

```
docker exec -it hadoop-master bash
```

## 6. Démarrer hadoop et yarn

*lancer hadoop et yarn en utilisant un script fourni appelé start-hadoop.sh. (consulter le script via la commande shell cat pour voir son contenu)*

```
./start-hadoop.sh
```

- A la fin du démarrage, vérifier si hadoop et yarn ont démarré correctement. Pour ce faire :
  - NameNode web UI: ***localhost:9870***
  - Ressource Manager UI: ***localhost:8088***
  - MapReduce JobHistory Server: ***localhost:19888***

## 7. Manipulations sur HDFS

- Créer un dossier input à l'aide de la commande

```
hdfs dfs -mkdir input
```

N.B : Si le bash vous retourne une erreur avec le message `: `.': No such file or directory`, Créer l'arborescence de l'utilisateur principal (root), comme suit:

```
hadoop fs -mkdir -p /user/root
```

- utiliser la commande suivante pour afficher le contenu de la racine

```
hdfs dfs -ls
```

- afficher les fichiers des sous-dossiers, avec une taille arrondie en Ko, Mo ou Go

```
hdfs dfs -ls -R -h ./
```

- Copier le fichier **purchases.txt** dans le **dossier de partage**

```
~/Documents/hadoop_project
```

- Copiez ce fichier sur HDFS par

```
hdfs dfs -put /shared_volume/purchases.txt .
```

- Pour vérifier Utiliser

```
hdfs dfs -ls -R
```

- Affiche le contenu du fichier à l'aide de la commande

```
hdfs dfs -cat purchases.txt
```

1. afficher la fin du fichier

```
hdfs dfs -tail purchases.txt
```

3. Supprimer ce fichier de HDFS

```
hdfs dfs -rm purchases.txt
```

- Remettre à nouveau ce fichier par

```
hdfs dfs -copyFromLocal /shared_volume/purchases.txt ./input
```

vérifier le contenu avec

```
hdfs dfs -ls
```

- Vérifiez son propriétaire, son groupe et ses droits

```
hdfs dfs -chmod 777 ./input/purchases.txt
```

```
hdfs dfs -chmod ugo-x ./input/purchases.txt (vérifiez les droits)
```

- Déplacer le fichier

```
hdfs dfs -mv /input/purchases.txt
```

- vérifier avec

```
hdfs dfs -ls -R
```

- transférer le fichier de HDFS vers votre machine en changeant le nom

```
hdfs dfs -get ./input/purchases.txt /shared_volume/achat.txt
```

- Copier le fichier dans une autre emplacement (la racine hdfs par exemple), ensuite vérifier avec -ls

```
hdfs dfs -cp ./input/purchases.txt ./purchases.txt
```

La documentation sur les commandes hadoop est disponible sur **le site officiel**

## 8. télécharger un fichier sur hdfs

- Créer un dossier input dans le HDFS

```
hdfs dfs -mkdir web_input
```

- Télécharger le livre alice se trouvant sur l'url ci-dessous dans votre machine locale

```
wget http://www.textfiles.com/etext/FICTION/alice.txt
```

- Copier le fichier vers le dossier partager hadoop_project
- Copier le fichier depuis la machine local vers le dossier input du hdfs

```
hdfs dfs -put /shared_volume/alice.txt web_input
```

- Vérifier l'opération

```
hdfs dfs -ls web_input
```

- Sortir de bash de hadoop-master **exit**
- Arrêter les trois conteneurs

```
docker stop hadoop-master hadoop-slave1 hadoop-slave2
```