Université Sultan Moulay Slimane
EST – Beni Mellal
Département : ITG
Pr. Siham BAKKOURI

Filières : DUT – IDIA & SIR – S3
Module : Big Data & BD NoSQL (M236)
Année universitaire : 2025/2026

Atelier N° 02 : Manipulation avancée de HDFS et MapReduce sous Docker

→ Objectifs

- Manipuler les fichiers et répertoires dans HDFS (création, suppression, lecture, copie).
- Comprendre la logique de réplication et tolérance aux pannes.
- Lancer et analyser des jobs MapReduce.
- Visualiser et interpréter les résultats des traitements dans HDFS.
- Suivre et diagnostiquer les jobs dans les interfaces Web Hadoop.

Cet atelier est à rendre via ce lien

Partie 1 : Vérification et préparation de l'environnement

→ Étape 1 : Démarrage du cluster Hadoop sous Docker

```
docker compose up -d
```

→ Étape 2 : Accéder au conteneur principal (NameNode)

```
docker exec -it namenode bash
```

→ Étape 3 : Vérifier les services Hadoop

```
jps
```

→ Étape 4 : Accès aux interfaces Web

- **NameNode UI :** http://localhost:9870
- **YARN UI :** http://localhost:8088

Partie 2 — Manipulation du système de fichiers HDFS

→ Étape 1 : Créer une arborescence de travail

```
hdfs dfs -mkdir -p /user/etudiant/hdfs_test
hdfs dfs -mkdir -p /data/textes
hdfs dfs -ls /
```

→ **Étape 2 : Ajouter des fichiers dans HDFS**

```
≥ echo "Hadoop est un système distribué" > texte1.txt
≥ echo "Le Big Data repose sur HDFS" > texte2.txt
≥ hdfs dfs -put -f texte1.txt texte2.txt /data/textes
```

→ **Étape 3 : Explorer le contenu**

```
≥ hdfs dfs -ls /data/textes
≥ hdfs dfs -cat /data/textes/texte1.txt
```

→ **Étape 4 : Vérifier la réplication**

```
≥ hdfs fsck /data/textes/texte1.txt -files -blocks -locations
```

➢ **But** : voir comment Hadoop découpe les fichiers en blocs et les réplique sur les DataNodes.

Partie 3 — Commandes HDFS essentielles

→ **Copie et déplacement**

```
≥ hdfs dfs -cp /data/textes/texte1.txt /user/etudiant/hdfs_test/
≥ hdfs dfs -mv /user/etudiant/hdfs_test/texte1.txt /data/
```

→ **Suppression**

```
≥ hdfs dfs -rm /data/textes/texte2.txt
```

→ **Informations sur l'espace**

```
≥ hdfs dfs -du -h /
≥ hdfs dfsadmin -report
```

→ **Changer les permissions**

```
≥ hdfs dfs -chmod -R 755 /data
≥ hdfs dfs -chown -R root:root /data
```

Partie 4 — Lancer et analyser des jobs MapReduce

→ **Étape 1 : Job WordCount (compte de mots)**

```
≥ hdfs dfs -mkdir -p /wordcount_input
≥ echo "Hadoop simplifie le traitement des grandes données" > wc1.txt
≥ echo "MapReduce permet d'analyser de gros volumes" > wc2.txt
≥ hdfs dfs -put -f wc1.txt wc2.txt /wordcount_input
```

Supprimer la sortie précédente :

```
≥ hdfs dfs -rm -r /wordcount_output
```

Exécuter le job :

```
≥ hadoop jar /opt/hadoop-3.2.1/share/hadoop/mapreduce/hadoop-mapreduce-examples-3.2.1.jar \
≥ wordcount /wordcount_input /wordcount_output
```

Afficher le résultat :

```
≥ hdfs dfs -cat /wordcount_output/part-r-00000
```

→ Étape 2 : Job "grep" (recherche de mot clé)

```
≥ hdfs dfs -rm -r /grep_output
≥ hadoop jar /opt/hadoop-3.2.1/share/hadoop/mapreduce/hadoop-mapreduce-examples-3.2.1.jar \
≥ grep /wordcount_input /grep_output "Hadoop"
≥ hdfs dfs -cat /grep_output/part-r-00000
```

Partie 5 — Exercices pratiques

Exercice 1

- Créer un dossier ***/etudiant/donnees*** dans HDFS.
- Ajouter trois fichiers texte.
- Afficher la taille totale avec ***hdfs dfs -du -h /etudiant/donnees***.
- Supprimer l'un d'eux et vérifier avec `ls`.

Exercice 2

- Dans **`hdfs-site.xml`**, modifier :
- `<property>`
- `  <name>dfs.replication</name>`
- `  <value>2</value>`
- `</property>`
- Redémarrer Hadoop.
- Vérifier la nouvelle réplication :
- `hdfs fsck /data -files -blocks -locations`

Exercice 3

- Créer un dossier **`/livres`** dans HDFS.
- Ajouter deux fichiers texte (extraits d'un texte long).
- Lancer ***WordCount*** pour compter le nombre de mots.
- Modifier un fichier et relancer le job → comparer les résultats.

Exercice 4

- Utiliser le job "grep" pour extraire les lignes contenant le mot "**données**".
- Vérifier le résultat dans **`/grep_output`**.