TP1

HDFS et mapreduce

Remarques préliminaires générales :

Lecture de ce document

Dans la suite de ce document les commandes et le code à utiliser seront présentés de la sorte :

Shell :

```
# un commentaire shell
maCommandeShell
```

Si la commande shell est trop longue le symbole « \ » sera présent pour indiquer que la commande est écrite sur plusieurs ligne. Le « \ » ne devra pas être copié.

```
# Exemple d'une longue commande shell
hadoop jar /usr/lib/hadoop-mapreduce/hadoop-streaming.jar \
-input matching/cv \
-output matching/cv_wc_stream
```

La Sandbox Hortonworks

Tous les exercices seront effectués sur une sandbox Hortonwork. Elle a pour nom **Hortonworks** dans le VMcatalog.

Attention, le premier lancement de la VM prend plusieurs minutes.

Les outils que nous allons utiliser pour manipuler la VM:

Connexion à la VM

Nous allons manipuler la VM et donc les outils Hadoop en nous connectant à la VM en ssh pour les interfaces en ligne de commande. Un certain nombre d'interfaces web sont aussi disponibles.

Une fois démarrée, la Sandbox (qui est elle même dans un container Docker) est accessible en ssh sur le port 2222.

Nous allons utiliser pour ce TP le user **raj_ops** pré-enregistré sur la Sandbox.

Pour ssh et ambari:

User: raj_ops

Password: raj_ops

Lancez un emulateur de terminal

```
ssh -p 2222 raj_ops@localhost
```

(Pour information, la liste de commandes utiles pour le Terminal :
http://cheatsheetworld.com/programming/unix-linux-cheat-sheet/

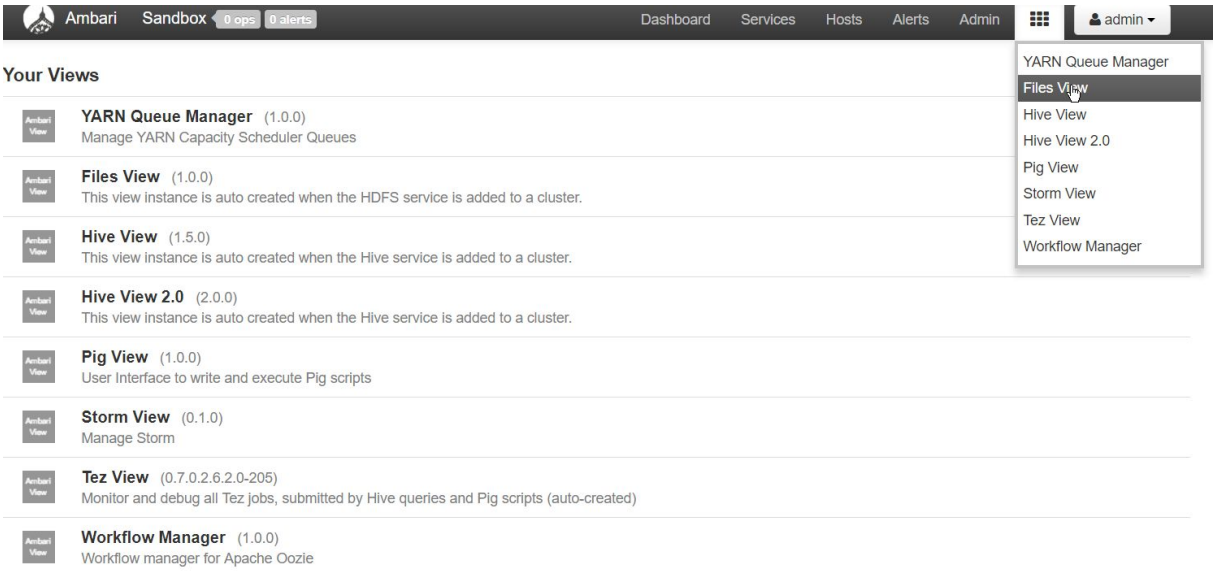
hadoop, hive, spark (pyspark / spark-shell)
Ces programmes sont directement accessibles et utilisables en ligne de commande dans le terminal. Pour sortir de beeline, Hive ou Spark faites CTRL+D (ou respectivement !q pour beeline , exit ; pour Hive et exit() pour spark)

Ambari

Ambari permet d'avoir une vision graphique des services disponibles sur un cluster Hadoop. Vous pouvez y accéder sur en allant à l'adresse http://localhost:8080 depuis votre navigateur.

Vous pouvez alors vous connectez avec raj_ops/raj_ops.

Vous avez accès à différentes vues :



La files view vous permet de naviguer sur HDFS.

Organisation du répertoire « TP1» sur la sandbox :

Pour réaliser le TP, vous devez uploader des fichiers sur la sandbox, puis sur hdfs.

Une fois logués en ssh sur la machine, créez un dossier TP1 :

```
mkdir TP1
cd TP1
```

Commencez par télécharger wc.jar et gulliver.txt

```
wget https://s3.eu-west-3.amazonaws.com/hadoop-telecom/TP1/gulliver.txt
wget https://s3.eu-west-3.amazonaws.com/hadoop-telecom/TP1/wc.jar
```

Vous pouvez visualiser le fichier gulliver.txt:

```
cat gulliver.txt
```

Exercice 1 : Manipulation HDFS

Objectif de l'exercice :

Se familiariser avec les outils d'interaction avec HDFS la ligne de commande (« hadoop fs -cmd » et l'interface web Ambari)

Rappel :

Pour la suite du TP, voila les commandes essentielles de la CLI HDFS

```
# visualiser le contenu d'un répertoire sur HDFS
hadoop fs –ls

# créer le répertoire « mon_repertoire » sur HDFS
hadoop fs –mkdir nom_repertoire

# placer le fichier local nom_fichier, sur hdfs dans le
répertoire
# HDFS rep_cible
hadoop fs –put nom_ficher rep_cible

# récupère le fichier nom_fichier d'HDFS et le place en local
hadoop fs –get nom_fichier

# visualise les 5 premières lignes du fichier fic :
hadoop fs –cat fic | head -5
```

Documentation Apache :

https://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html

Tâches à réaliser

Preliminaire :

Les commandes hadoop sont lancées par défaut sur un repertoire de travail HDFS par defaut s'appuyant sur le nom du user unix qui lance la commande: /user/<mon_user> (ce sera raj_ops pour nous). /user/raj_ops sera donc notre répertoire de travail HDFS pour le reste du TP.

Créez sur HDFS un repertoire /user/raj_ops/TP1/input

```
# creation du repertoire sur HDFS
hadoop fs -mkdir -p TP1/input
```

Uploadez le fichier gulliver.txt vers ce dossier

```
hadoop fs -put gulliver.txt TP1/input
```

notez que dans la commande ci-dessus, gulliver.txt est dans le répertoire local (sur la VM linux) alors que TP1/input fait référence à un dossier sur HDFS.

Affichez avec la commande «hadoop fs -cat » et piper avec « tail » (voir cheatsheet) les 5 dernières lignes du fichier gulliver.txt **situé sur HDFS**.

Hadoop fs -cat TP1/input/gulliver.txt | tail -n 5

Manipulation d'Ambari

Lancer Ambari (voir chapitre 0) et connectez-vous avec le user existant raj_ops

Rendez-vous dans le file browser HDFS (depuis cette icone: )

Déplacez-vous dans le répertoire /user/raj_ops

Visualisez le répertoire que vous avez créés

Visualisez le contenu du fichier que vous avez chargé

Dans le dossier TP1/input, uploadez gulliver2.txt

Recupérez le en local:

```
https://s3.eu-west-3.amazonaws.com/hadoop-telecom/TP1/gulliver2.txt
```

Téléchargez le avec firefox ou autre (enregistrez sous si nécessaire)

Utilisez l'interface ambari file view pour uploadez le fichier dans TP1/input

Vérifiez le contenu de ce fichier depuis Ambari

Fin de l'exercice 1

Exercice 2 : Lancement d'un job MR

Objectif de l'exercice :

Exécuter un job mapreduce

Emplacement de l'exercice :

Rendez-vous via ssh dans le répertoire TP1, ou vous avez normalement downloadé le job mapreduce wc.jar .

Tâches à réaliser

Lancer le job sur les fichiers que vous avez mis dans TP1/input sur HDFS

Visualisez le résultat

Pour lancer le job :

```
hadoop jar wc.jar WordCount TP1/input TP1/output
```

Le job MapReduce va s'exécuter sur YARN.

Vous pouvez observer son avancement sur l'interface web du ressource manager YARN

:

http://localhost:8088

Une fois le traitement terminé (c'est normal que ce soit lent), rendez vous sur le file viewer d'Ambari pour voir le résultat dans TP1/output.

Vous devriez y retrouver un fichier part-r-00000

Ouvrez le.

Observez que les résultats sont sous forme <texte><tabulation><nombre_d_occurence>

Retournez en ssh, exécutez une commande permettant d'afficher les 20 premiers mots triés par ordre d'apparition.

Indice : la commande permettant de trier des données sous unix est sort. Dans notre cas, on veut trier selon la 2 ème colonne d'un csv dont le séparateur est tabulation. Cette colonne est un entier. Ça donne :

sort -t$'\t' -k2 -rn

Utilisez le pipe | , et la commande head

Pour avoir des infos sur les différentes commandes, vous pouvez faire man *cmd*, par exemple man head.

Pour plus de détails sur le WordCount utilisé :

https://hadoop.apache.org/docs/r2.8.0/hadoop-mapreduce-client/hadoop-mapreduce-client-core/MapReduceTutorial.html#Example:_WordCount_v1.0

Pour aller plus loin:

Modifiez le job MapReduce pour qu'il ne compte pas les mots de 3 caractères ou moins et qu'il ne soit pas sensible à la casse.
https://docs.oracle.com/javase/8/docs/api/java/lang/String.html
Le code :
https://s3.eu-west-3.amazonaws.com/hadoop-telecom/TP1/src/WordCount.java

Pour créer le jar:

Il faut mettre en place ces variables d'environnement:

export JAVA_HOME=/usr/java/default
export PATH=${JAVA_HOME}/bin:${PATH}
export HADOOP_CLASSPATH=${JAVA_HOME}/lib/tools.jar

Puis:

hadoop com.sun.tools.javac.Main WordCount.java
jar cf wc.jar WordCount*.class

Attention vous aurez une erreur si vous utilisez le même dossier output sur HDFS. Supprimez le ou choisissez un autre dossier.

Je vous conseille d'utiliser vi ou emacs pour modifier le fichier WordCount.java. Vous pouvez faire des yum install si nécessaire.

Si vous avez besoin d'un éditeur graphique, travaillez en local avec Kate (applications → accessoires → Kate) et uploadez le code sur la Sandbox avec FileZilla (protocole sftp , host localhost, port 2222, user/mdp raj_ops)

Une fois ce nouveau MapReduce executé, verifiez que vous avez bien la liste des mots de plus de 3 lettres et leur fréquence.

Vérifiez sur l'interface web du Ressource Manager de YARN que le job a bien généré 2 mappers (un par fichier) et 1 reducer.

http://localhost:8088

Si vous avez une erreur page introuvable, remplacez sandbox-hdp.hortonworks.com par localhost dans l'URL. Il se peut que vous deviez le faire 2 fois de suite.

Pour avoir la Sandbox chez vous:

Vous pouvez la télécharger ici:
https://fr.hortonworks.com/downloads/#sandbox

Il vont vous demander un nom, prenom, profession etc… vous pouvez mettre n'importe quoi.
Vous aurez besoin de VirtualBox, VMWare ou Docker. Je vous conseille VirtualBox

C'est exactement la même image que celle installée dans les salles de TP.
Si vous êtes sur Windows, vous pouvez utiliser Putty ou MobaXterm comme client SSH.