

# Rapport Projet Données Réparties

## Test de Performance

Axel GRUNIG  
Thomas GUILLAUD  
Chloé LAPLAGNE  
Nathan RAZAFIMANANTSOA

Département Sciences du Numérique - ASR  
2020-2021

# 1 Performances HDFS

- Tests fonctionnels avec medium.txt (4.1MB) puis GO (2.9GB) sur 4 machines serveurs situées à l’N7 et une machine client connectée par le VPN (débit autour de 5.5Mo/s) :

```
hadoop> hdfs -w medium.txt --chunks-size=1MB
Splitting file in 5 chunks...
# 100 %
medium.txt successfully saved.
— time (ms) : 1796
hadoop> hdfs -r medium.txt
Reading file ...
# 100 %
medium.txt successfully read to r_medium.txt.
— time (ms) : 1264
hadoop> hdfs -d medium.txt
Deleting file ...
# 100 %
medium.txt successfully deleted.
— time (ms) : 103
hadoop> hdfs -w GO --chunks-size=64MB
Splitting file in 46 chunks...
# 100 %
GO successfully saved.
— time (ms) : 534723
```

- Tests intégralement sur le réseau de l’N7 : pour un fichier de 6.5 Go, le temps d’écriture du fichier dans HDFS est resté relativement stable [ 68.2s ; 72.4s ] pour un nombre de chunks allant de 4 à 384. Le temps de déléition (directement lié au nombre de chunks à supprimer) était compris dans l’intervalle [ 67ms ; 678ms ].

## 2 Performance distribué/séquentiel

Les lignes commençant par "hadoop>" sont les commandes exécutées pour faire ce test de performance.  
Taille des fichiers utilisés : big.txt → 120Mo    *medium.txt* → 4Mo

```
hadoop> hdfs -w medium.txt --chunks-size = 1000000
```

```
Splitting file in 5 chunks...
```

```
100 %
```

```
medium.txt successfully saved.
```

```
-- time (ms) : 1167
```

```
hadoop> mmr -ip 172.22.232.226 medium.txt
```

```
Launching workers...
```

```
0 signaux en attente
```

```
Fin Attente
```

```
Reading file...
```

```
100 %
```

```
medium.txt-res successfully read to medium.txt-res.
```

```
Launching reduce task...
```

```
Reduce done : medium.txt-tot
```

```
-- time (ms) : 1306
```

```
Deleting file...
```

```
100 %
```

```
medium.txt-res successfully deleted.
```

```
hadoop> cmp_ref medium.txt
```

```
-- time (ms) : 375
```

```
hadoop> hdfs -w big.txt --chunks-size=10000000
```

```
Splitting file in 13 chunks...
```

```
100 %
```

```
big.txt successfully saved.
```

```
-- time (ms) : 25292
```

```
hadoop> mmr -ip 172.22.232.226 big.txt
```

```
Launching workers...
```

```
0 signaux en attente
```

```
Fin Attente
```

```
Reading file...
```

```
100 %
```

```
big.txt-res successfully read to big.txt-res.
```

```
Launching reduce task...
```

```
Reduce done : big.txt-tot
```

```
-- time (ms) : 2533
```

```
Deleting file...
```

```
100 %
```

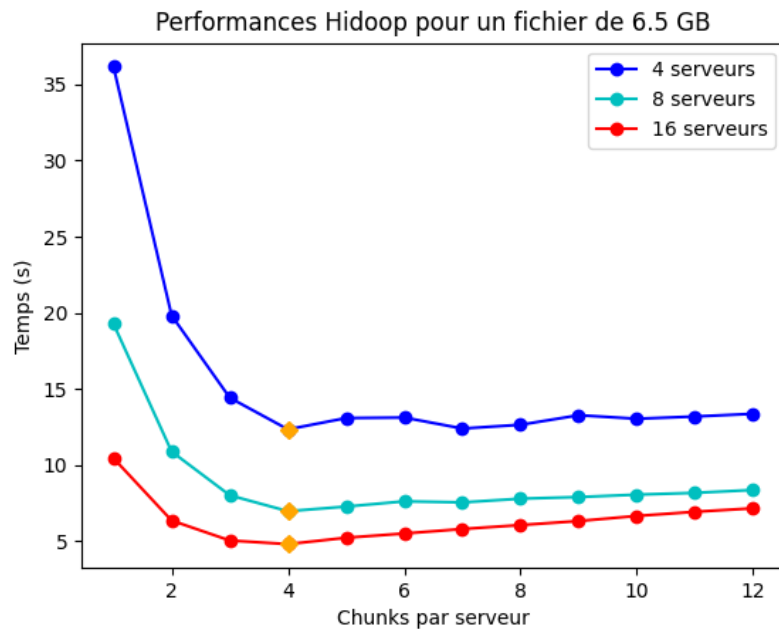
```
big.txt-res successfully deleted.
```

```
hadoop> cmp_ref big.txt
```

```
-- time (ms) : 3037
```

On constate donc qu'il devient intéressant d'utiliser la version MapReduce de l'application WordCount plutôt que sa version séquentielle à partir d'un fichier d'environ 150 Mo.

### 3 Performance suivant la taille des fichiers, et le nombre de chunks

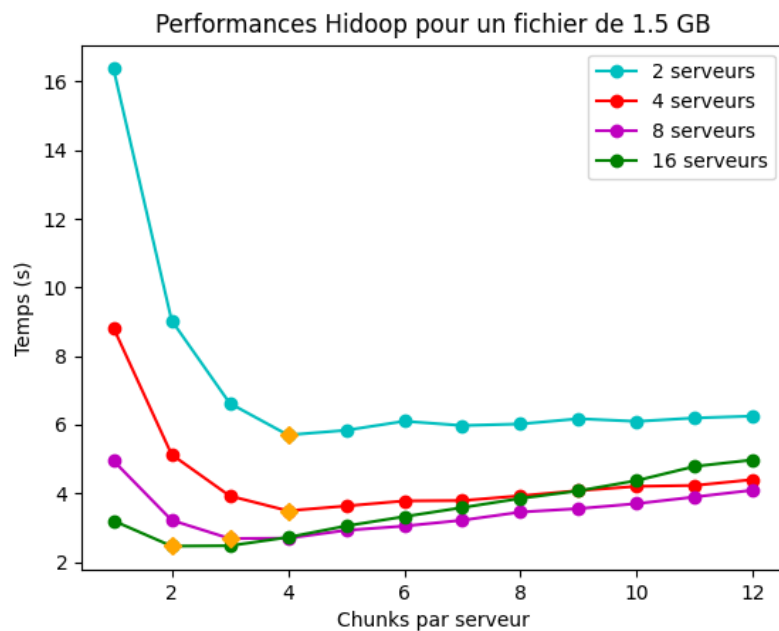


Min : 4 chunks, de 405.02 MB. Temps (s) : 12.3433

Min : 4 chunks, de 202.51 MB. Temps (s) : 6.969

Min : 4 chunks, de 101.26 MB. Temps (s) : 4.8086

Sur processeurs à 4 coeurs. Temps séquentiel : 136.2 s



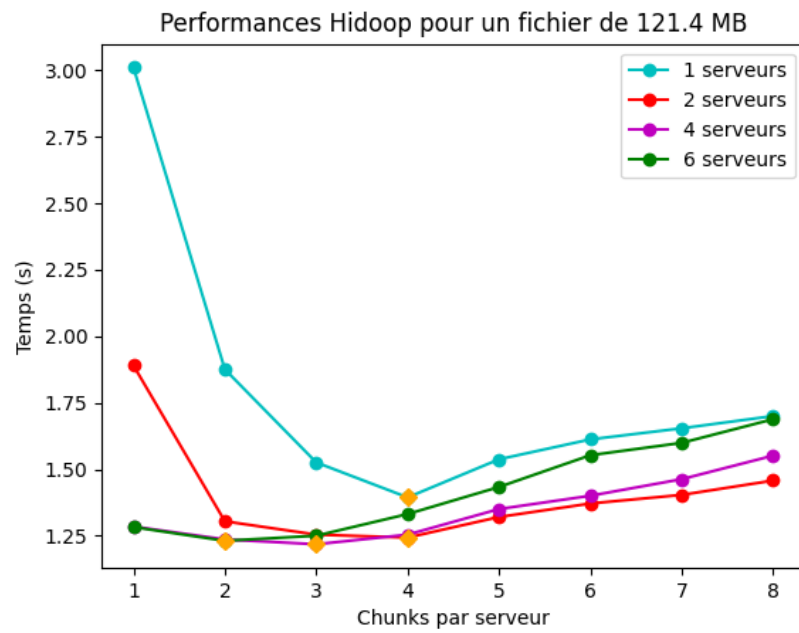
Min : 4 chunks, de 182.12 MB. Temps (s) : 5.6995

Min : 4 chunks, de 91.06 MB. Temps (s) : 3.4906

Min : 3 chunks, de 60.71 MB. Temps (s) : 2.6813

Min : 2 chunks, de 45.53 MB. Temps (s) : 2.4611

Sur processeurs à 4 coeurs. Temps séquentiel : 30.8 s



Min : 4 chunks, de 30.35 MB. Temps (s) : 1.3939

Min : 4 chunks, de 15.18 MB. Temps (s) : 1.2429

Min : 3 chunks, de 10.12 MB. Temps (s) : 1.2178

Min : 2 chunks, de 10.12 MB. Temps (s) : 1.2308

Temps séquentiel : 2.8 s

