# Xestión de HDFS

### Abraham Trashorras Rivas

## Primeira parte

Comezamos a practica con 1 nodename e 5 datanodes (numerados do 1 ao 6 excluíndo o 4) nunha estrutura topográfica de 3 racks asignados por orde numérica onde o rack 2 solo ten un datanode.

```
Rack: /rack1
172.19.0.4:9866 (datanode2.hadoop-cluster) In Service
172.19.0.3:9866 (datanode1.hadoop-cluster) In Service

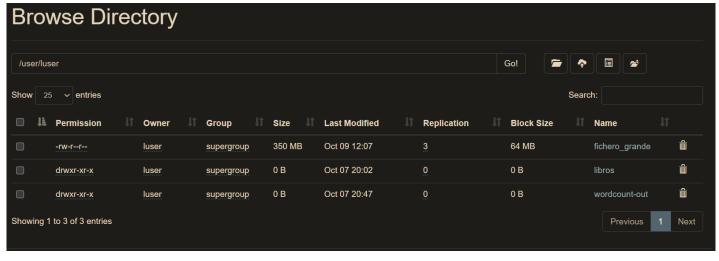
Rack: /rack2
172.19.0.5:9866 (datanode3.hadoop-cluster) In Service

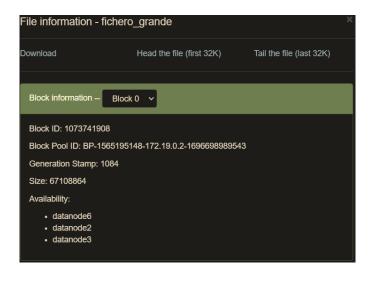
Rack: /rack3
172.19.0.8:9866 (datanode6.hadoop-cluster) In Service
172.19.0.7:9866 (datanode5.hadoop-cluster) In Service
```

Agora, como usuario luser, creamos un ficheiro grande e movémolo ao HDFS:

```
dd if=/dev/urandom of=fichero_grande bs=1M count=350
hdfs dfs -moveFromLocal fichero_grande /user/luser
```

Unha vez cargado fichero\_grande podemos consultalo na interface web dispoñible en <a href="http://localhost:9870">http://localhost:9870</a>:





Podemos ver na primeira imaxe da interface a carpeta /user/luser que contén ficheiro\_grande e na segunda imaxe as características deste por bloque, neste caso a do bloque 0. Este documento tiña asignados 6 bloques gardados da seguinte forma:

- Bloque 0: [datanode6, datanode2, datanode3]
- Bloque 1: [datanode6, datanode1, datanode3]
- Bloque 2: [datanode6, datanode2, datanode3]
- Bloque 3: [datanode6, datanode2, datanode3]
- Bloque 4: [datanode6, datanode2, datanode3]
- Bloque 5: [datanode6, datanode1, datanode2]

Tamén podemos acceder a esta información mediante comandos:

```
hdfs fsck /user/luser/fichero_grande -files -blocks -locations
                            0. BP-1565195148-172.19.0.2-1696698989543:blk_1073741908_1084 len=67108864 Live_repl=3
[DatanodeInfoWithStorage[172.19.0.4:9866,DS-c9eeb7ce-b4b8-4b45-acf1-3aa529704383,DISK],
DatanodeInfoWithStorage[172.19.0.5:9866,DS-79b74aa9-c175-449d-8cdb-d3b4a721fbb8,DISK],
DatanodeInfoWithStorage[172.19.0.8:9866,DS-b4dc9d7a-55a4-4ffc-8d15-006545c35f50,DISK]]
                           1. BP-1565195148-172.19.0.2-1696698989543:blk_1073741909_1085 len=67108864 Live_repl=3
[DatanodeInfoWithStorage[172.19.0.5:9866,DS-79b74aa9-c175-449d-8cdb-d3b4a721fbb8,DISK],
DatanodeInfoWithStorage[172.19.0.8:9866,DS-b4dc9d7a-55a4-4ffc-8d15-006545c35f50,DISK],
DatanodeInfoWithStorage[172.19.0.3:9866,DS-fdec86e1-d9f1-470d-8a9b-569ffe7bfc1b,DISK]]
                            2. BP-1565195148-172.19.0.2-1696698989543:blk_1073741910_1086 len=67108864 Live_repl=3
[Data node Info With Storage [172.19.0.4:9866, DS-c9eeb7ce-b4b8-4b45-acf1-3aa529704383, DISK], and the storage of the storag
Data node Info With Storage [172.19.0.5:9866, DS-79b74aa9-c175-449d-8cdb-d3b4a721fbb8, DISK], A storage [172.19.0.5:9866, DS-79b74ab4, DISK], A storage [172.19.0.5:9866, DS-79b74ab4, DS-79b74ab4, DS-79b74ab4, DS-79b74ab4
DatanodeInfoWithStorage[172.19.0.8:9866,DS-b4dc9d7a-55a4-4ffc-8d15-006545c35f50,DISK]]
                            3. BP-1565195148-172.19.0.2-1696698989543:blk_1073741911_1087 len=67108864 Live_repl=3
[DatanodeInfoWithStorage[172.19.0.4:9866,DS-c9eeb7ce-b4b8-4b45-acf1-3aa529704383,DISK],
DatanodeInfoWithStorage[172.19.0.5:9866,DS-79b74aa9-c175-449d-8cdb-d3b4a721fbb8,DISK],
DatanodeInfoWithStorage[172.19.0.8:9866,DS-b4dc9d7a-55a4-4ffc-8d15-006545c35f50,DISK]]
                            4. BP-1565195148-172.19.0.2-1696698989543:blk_1073741912_1088 len=67108864 Live_repl=3
[DatanodeInfoWithStorage[172.19.0.4:9866,DS-c9eeb7ce-b4b8-4b45-acf1-3aa529704383,DISK],
DatanodeInfoWithStorage[172.19.0.5:9866,DS-79b74aa9-c175-449d-8cdb-d3b4a721fbb8,DISK],
DatanodeInfoWithStorage[172.19.0.8:9866,DS-b4dc9d7a-55a4-4ffc-8d15-006545c35f50,DISK]]
                            5. BP-1565195148-172.19.0.2-1696698989543:blk_1073741913_1089 len=31457280 Live_repl=3
[Data node Info with Storage [172.19.0.4:9866, DS-c9eeb7ce-b4b8-4b45-acf1-3aa529704383, DISK], and the storage of the storag
DatanodeInfoWithStorage[172.19.0.3:9866,DS-fdec86e1-d9f1-470d-8a9b-569ffe7bfc1b,DISK],
DatanodeInfoWithStorage[172.19.0.8:9866,DS-b4dc9d7a-55a4-4ffc-8d15-006545c35f50,DISK]]
```

Que unha vez extraídas as IP para cada bloque nos daría:

- [172.19.0.4, 172.19.0.5, 172.19.0.8] -> [datanode2, datanode3, datanode6]
- [172.19.0.3, 172.19.0.5, 172.19.0.8] -> [datanode1, datanode3, datanode6]
- [172.19.0.4, 172.19.0.5, 172.19.0.8] -> [datanode2, datanode3, datanode6]
- [172.19.0.4, 172.19.0.5, 172.19.0.8] -> [datanode2, datanode3, datanode6]
- [172.19.0.4, 172.19.0.5, 172.19.0.8] -> [datanode2, datanode3, datanode6]
- [172.19.0.3, 172.19.0.4, 172.19.0.8] -> [datanode1, datanode2, datanode6]

## Segunda parte

Como usuario hdadmin, creamos unha nova carpeta de HDFS quota4 na ruta /user/hdadmin/quota4 e establecemos un limite de 4 nombres asignados no directorio:

```
su - hdadmin
hdfs dfs -mkdir /user/hdadmin/quota4
hdfs dfsadmin -setQuota 4 /user/hdadmin/quota4
```

Unha vez creado o directorio creamos 4 documentos e tratamos de introducilos en HDFS na dirección /user/hdadmin/quota4. No cuarto elemento obtemos o seguinte resultado:

```
hdfs dfs -copyFromLocal archivo4.txt /user/hdadmin/quota4/copyFromLocal: The NameSpace quota (directories and files) of directory /user/hdadmin/quota4 is exceeded: quota=4 file count=5
```

Isto ocorre porque o límite conta tanto os nomes dos documentos dentro do ficheiro como o nome do propio directorio, polo que o límite efectivo sería de 3 documentos.

#### Terceira parte

Vamos a comprobar o estado do HDFS (omito Erasure Coded Block Groups porque non contén información):

```
hdfs fsck /
Status: HEALTHY
Number of data-nodes: 5
Number of racks:
                              3
Total dirs:
                              19
Total symlinks:
Replicated Blocks:
Total size: 374395293 B
Total files: 32
                             36 (avg. block size 10399869 B)
Total blocks (validated):
Minimally replicated blocks: 36 (100.0 %)
Over-replicated blocks: 0 (0.0 %) 0 (0.0 %)
Under-replicated blocks:
Mis-replicated blocks:
                             0 (0.0 %)
Default replication factor:
                            3
Average block replication:
                              3.0
Missing blocks:
                              0
Corrupt blocks:
                              0
Missing replicas:
                               0 (0.0 %)
Blocks queued for replication: 0
(...)
The filesystem under path '/' is HEALTHY
```

O sistema está san e libre de erros. Agora matamos 3 datanodes (os números 1, 3 e 5) e esperamos 10 minutos a que o sistema detecte que están apagados. Pasado este tempo volvemos a comprobar o estado:

```
hdfs dfsadmin -report
(...)
Live datanodes (2):
(...)
Hostname: datanode2
Rack: /rack1
Num of Blocks: 36
(...)
Hostname: datanode6
Rack: /rack3
Num of Blocks: 36
(...)
Dead datanodes (3):
(...)
Hostname: datanode1
/rack1
(...)
Hostname: datanode3
Rack: /rack2
(...)
Hostname: datanode5
Rack: /rack3
```

Podemos ver como quedan dous datanodes vivos (os número 2 e 6) onde cada un garda 36 bloques.

Tamén podemos volver a revisar o estado do HDFS e comprobar se existen bloques de documento corruptos:

```
hdfs fsck / -list-corruptfileblocks
Connecting to namenode via http://namenode:9870/fsck?ugi=hdadmin&listcorruptfileblocks=1&path=%2F
The filesystem under path '/' has 0 CORRUPT files
hdfs fsck / -files -blocks -locations -racks
Status: HEALTHY
Number of data-nodes: 2
Number of racks:
                                 2
Total dirs:
                                 19
Total symlinks:
Replicated Blocks:
Total size:
                374395293 B
Total files:
               32
Total blocks (validated): 36 (avg. block size 10399869 B)
Minimally replicated blocks: 36 (100.0 %)
Over-replicated blocks: 0 (0.0 %)
Under-replicated blocks: 36 (100.0 %)
Mis-replicated blocks: 0 (0.0 %)
Default replication factor: 3
Average block replication:
                                2.0
Missing blocks:
                                 0
Corrupt blocks:
                                 0
Missing replicas:
                                 36 (33.333332 %)
Blocks queued for replication: 0
```

Como podemos ver en "Average block replication" baixamos dos 3 que estableceramos a 2 actualmente e que faltan 36 réplicas de bloques, así como temos a mesma cantidade "underreplicated".

Podemos estudar o estado dun ficheiro en concreto, neste caso ficheiro\_grande, e ver como lle afectaron estes cambios:

```
hdfs fsck /user/luser/fichero_grande
Status: HEALTHY
Number of data-nodes: 2
Number of racks:
Total dirs:
                              0
Total symlinks:
Replicated Blocks:
Total size: 367001600 B
Total files: 1
Total blocks (validated):
                             6 (avg. block size 61166933 B)
Minimally replicated blocks: 6 (100.0 %)
Over-replicated blocks: 0 (0.0 %)
Under-replicated blocks:
                            6 (100.0 %)
Mis-replicated blocks:
                             0 (0.0 %)
                           2.0
Default replication factor:
Average block replication:
Missing blocks:
Corrupt blocks:
                              0
Missing replicas:
                              6 (33.333332 %)
Blocks queued for replication: 0
```

Para ficheiro grande faltan 6 réplicas, o 33%, pero aínda así é recuperable.

A continuación, engadimos un novo datanode7 incluíndoo nos ficheiros dfs.include e yarn.include e comprobamos o efecto que isto ten na replicación de bloques:

```
hdfs fsck / -files -blocks -locations -racksStatus: HEALTHY
Status: HEALTHY
Number of data-nodes: 3
Number of racks:
                               19
Total dirs:
Total symlinks:
Replicated Blocks:
 Total size:
              374395293 B
Total files:
              32
Total blocks (validated):
                              36 (avg. block size 10399869 B)
Minimally replicated blocks: 36 (100.0 %)
Over-replicated blocks:
                              0 (0.0 %)
Under-replicated blocks:
                              0 (0.0 %)
Mis-replicated blocks:
                              0 (0.0 %)
Default replication factor:
                              3
Average block replication:
                              3.0
Missing blocks:
                              0
Corrupt blocks:
                               0
                               0 (0.0 %)
Missing replicas:
```

Como podemos ver, o número de datanodes ascendeu a 3, o numero de bloques "underreplicated" reduciuse a 0 e o factor de replicación medio volve ser igual ao factor de replicación por defecto, 3.

Por último, podemos volver a consultar o estado dos datanodes para comprobar que efectivamente o novo datanode7 asumiu a carga dos bloques "under-replicated" quedandose con 36:

```
hdfs dfsadmin -report
Name: 172.19.0.3:9866 (datanode7.hadoop-cluster)
Hostname: datanode7
Rack: /rack1
Decommission Status : Normal
Configured Capacity: 1081101176832 (1006.85 GB)
DFS Used: 377345137 (359.86 MB)
Non DFS Used: 10016684943 (9.33 GB)
DFS Remaining: 1015714791424 (945.96 GB)
DFS Used%: 0.03%
DFS Remaining%: 93.95%
Configured Cache Capacity: 0 (0 B)
Cache Used: 0 (0 B)
Cache Remaining: 0 (0 B)
Cache Used%: 100.00%
Cache Remaining%: 0.00%
Xceivers: 0
Last contact: Mon Oct 09 16:45:18 CEST 2023
Last Block Report: Mon Oct 09 16:40:39 CEST 2023
Num of Blocks: 36 Blocks queued for replication: 0
```

#### Cuarta parte

Despois de levantar de novo os datanodes 1, 3 e 5 e comprobar que o sistema volve a incluílos sen erros vamos a comprobar as políticas de EC que temos dispoñibles:

```
hdfs ec -listPolicies

Erasure Coding Policies:

ErasureCodingPolicy=[Name=RS-10-4-1024k, Schema=[ECSchema=[Codec=rs, numDataUnits=10, numParityUnits=4]], CellSize=1048576, Id=5], State=DISABLED

ErasureCodingPolicy=[Name=RS-3-2-1024k, Schema=[ECSchema=[Codec=rs, numDataUnits=3, numParityUnits=2]], CellSize=1048576, Id=2], State=DISABLED

ErasureCodingPolicy=[Name=RS-6-3-1024k, Schema=[ECSchema=[Codec=rs, numDataUnits=6, numParityUnits=3]], CellSize=1048576, Id=1], State=ENABLED

ErasureCodingPolicy=[Name=RS-LEGACY-6-3-1024k, Schema=[ECSchema=[Codec=rs-legacy, numDataUnits=6, numParityUnits=3]], CellSize=1048576, Id=3], State=DISABLED

ErasureCodingPolicy=[Name=XOR-2-1-1024k, Schema=[ECSchema=[Codec=xor, numDataUnits=2, numParityUnits=1]], CellSize=1048576, Id=4], State=DISABLED
```

Temos 5 políticas dispoñibles e solo unha activa:

- RS-10-4-1024k
- RS-3-2-1024k
- RS-6-3-1024k (ACTIVADA)
- RS-LEGACY-6-3-1024k
- XOR-2-1-1024k

Vamos a activar unha das políticas, RS-3-2-1024k, crear unha nova carpeta HDFS e asignarlle a nova política de EC activada:

```
hdfs ec -enablePolicy -policy RS-3-2-1024k

hdfs dfs -mkdir /user/grandes

hdfs ec -setPolicy -path /user/grandes -policy RS-3-2-1024k
```

Agora vamos a tomar os datos do HDFS co documento fichero\_grande na carpeta sen a política EC /user/luser:

```
dd if=/dev/urandom of=fichero_grande bs=1M count=350

hdfs dfs -put fichero_grande /user/luser/

dfs dfsadmin -report
(Capacidade ocupada)
DFS Used: 113276992 (1.05 GB)
(bloques en uso por datanode)
datanode1 -> 0
datanode2 -> 33
datanode3 -> 31
datanode5 -> 6
datanode6 -> 5
datanode7 -> 33
total -> 108
```

Tras isto, borramos o documento, valeiramos a papeleira e introducimos unha copia nova na carpeta /user/grandes. Fágoo así xa que se solo o movo entre carpetas non se aplica a nova política EC:

```
hdfs dfs -rm /user/luser/fichero_grande

hdfs dfs -expunge

hdfs dfs -put fichero_grande /user/grandes

dfs dfsadmin -report
(Capacidade ocupada)
DFS Used: 1132756992 (1.05 GB)
(bloques en uso por datanode)
datanode1 -> 2
datanode2 -> 32
datanode3 -> 30
datanode5 -> 2
datanode6 -> 3
datanode7 -> 31
total -> 100
```

Podemos ver unha diminución na capacidade usada e no número de bloques, polo que a política EC esta a funcionar. Por último vamos a consultar o estado da carpeta para ver os efectos desta:

```
hdfs fsck /user/grandes
Connecting to namenode via http://namenode:9870/fsck?ugi=hdadmin&path=%2Fuser%2Fgrandes
FSCK started by hdadmin (auth:SIMPLE) from /172.19.0.2 for path /user/grandes at Tue Oct 10 17:49:41
Status: HEALTHY
Number of data-nodes: 6
Number of racks:
Total dirs:
Total symlinks:
Replicated Blocks:
Total size: 0 B
Total files: 0
Total blocks (validated): 0
Minimally replicated blocks: 0
Over-replicated blocks: 0
Under-replicated blocks:
                              0
Mis-replicated blocks:
                               0
Default replication factor:
                               3
Average block replication:
                              0.0
Missing blocks:
                               0
Corrupt blocks:
                              0
Missing replicas:
Blocks queued for replication: 0
Erasure Coded Block Groups:
Total size: 367001600 B
Total files: 1
Total block groups (validated): 2 (avg. block group size 183500800 B)
Minimally erasure-coded block groups: 2 (100.0 %)
Over-erasure-coded block groups: 0 (0.0 %)
Under-erasure-coded block groups: 0 (0.0 %)
Unsatisfactory placement block groups: 0 (0.0 %)
Average block group size: 5.0
Missing block groups:
                              0
Corrupt block groups:
                              0
Missing internal blocks: 0 (0.0 %)
Blocks queued for replication: 0
FSCK ended at Tue Oct 10 17:49:41 CEST 2023 in 2 milliseconds
```

Podemos ver como existen dous bloques asignados á política de EC que corresponden ao documento fichero\_grande co seu peso orixinal de 367001600 B.