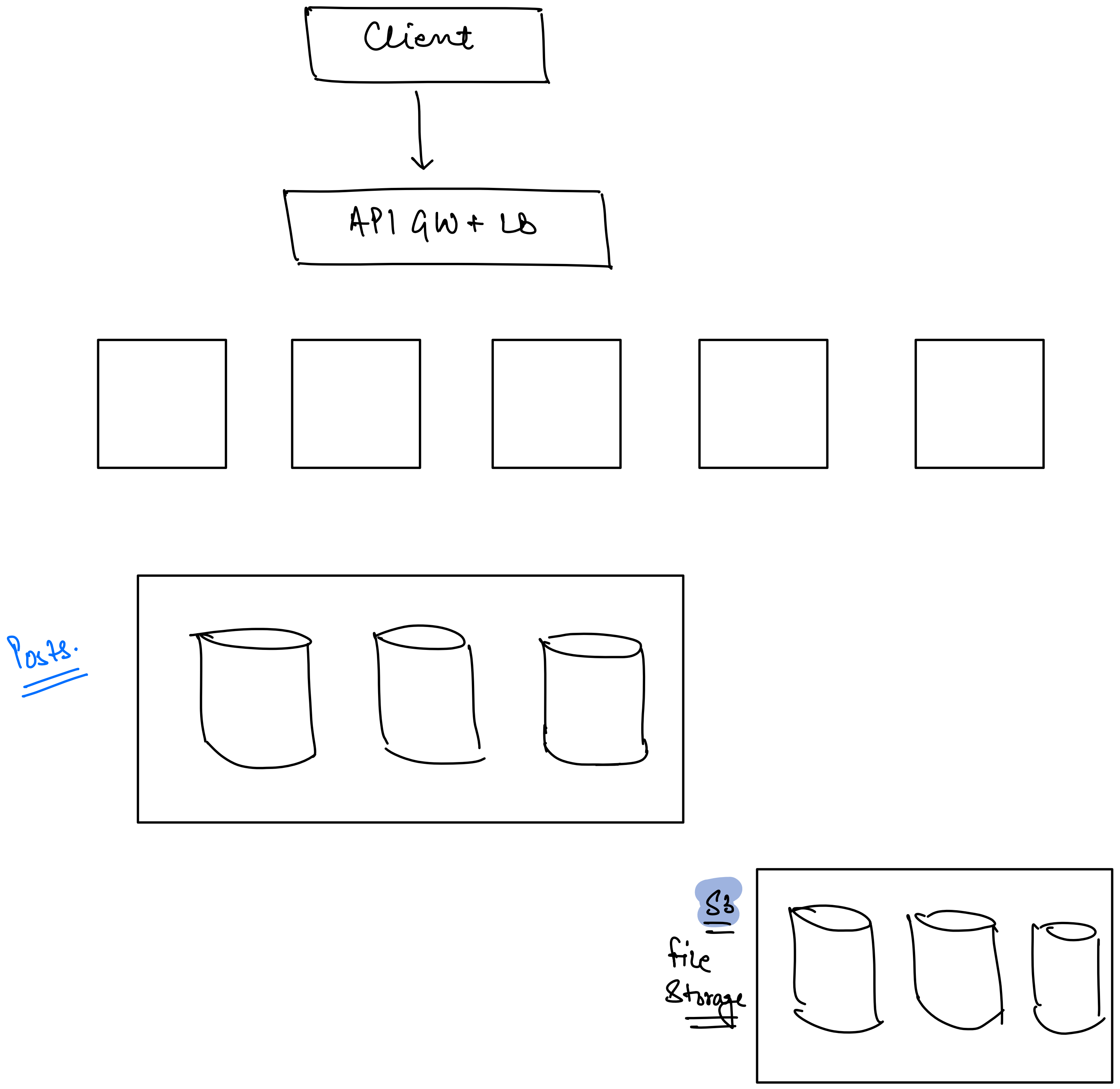
Agenda.

→ Design S3 → AWS.

↳ File Storage (for storing big files)

- → Blob storage
- → S3
- → HDFS.



⇒ File Storage.

- → Media Content
  (Image | Video | file etc)
- → In main DB (SQL or NOSQL), we just store the S3 URL to refer the content from S3.

→ File Storage System should be able to store big files.

- → 1 GB | 10GB | 1TB | 10TB . . .

→ Durable

→ Performance of uploading (or) downloading.

- → What happens if our connection drops in b/w ?

Option 1: File stored as 1 unit in 1 machines.

Pros: 1) No need to maintain the chunk info.
(Cost of entries)

2) We need not to collate the chunks.

Cons: 1) File size is limited by machine size.

2) Parallelism NOT possible.

Option 2: Divide file into smaller chunks and store each chunk into a different m/c.

Pros:

Cons:

Note: We are going to divide a file into multiple chunks so that the file size isn't limited by the m/c size & parallelism would be possible.

But we shouldn't be dividing file into very small chunks otherwise managing these many chunks would become an overhead.

⇒ **HDFS.**

↳ Hadoop Distributed File **System.**

Repliation

1) **Data Nodes** : Machines where we store file **Chunks.**

2) **Name Nodes** : Store metadata about the **files.**

→ Maintains the mapping of file chunks and in which m/c these chunks are **present.**

→ Repliation

$F_1 \rightarrow$ 200 Chunks

$F_1C_1 \rightarrow D_7$

$F_1C_2 \rightarrow D_{10}$

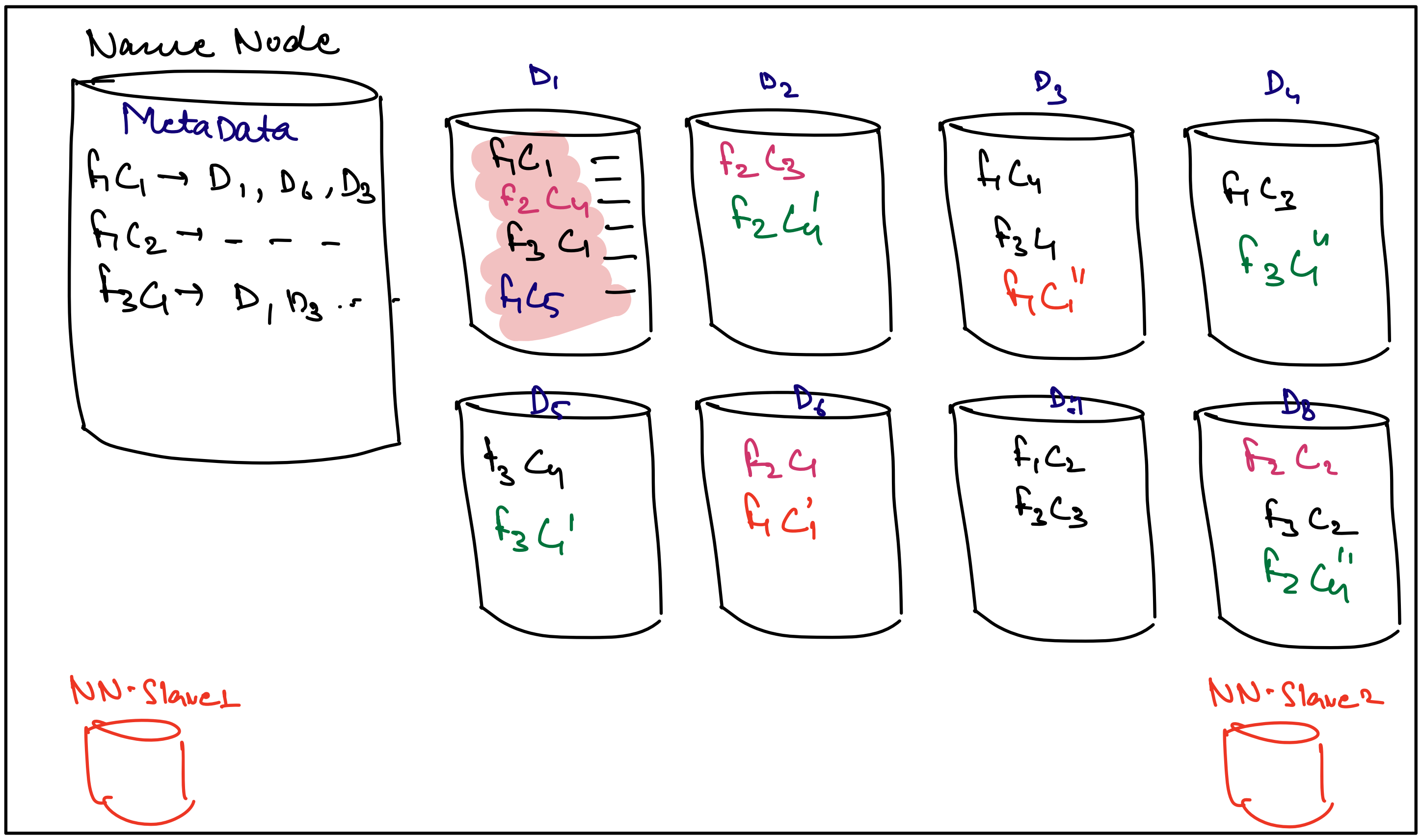
$F_1C_3 \rightarrow D_4$

$F_1C_4 \rightarrow D_8$

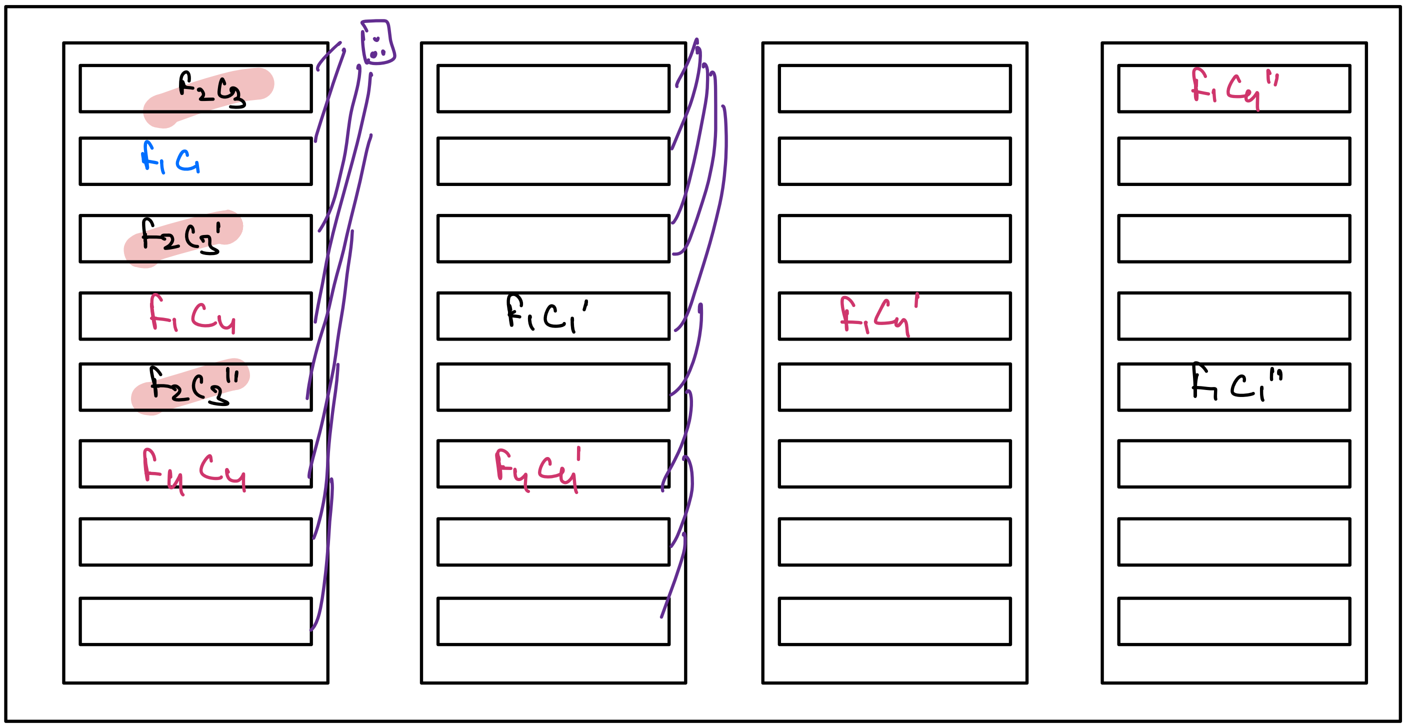
⋮ ⋮

⇒ In one HDFS cluster, we don't shard a Name Node server but replication would be **there.**

⇒ HDFS Cluster.



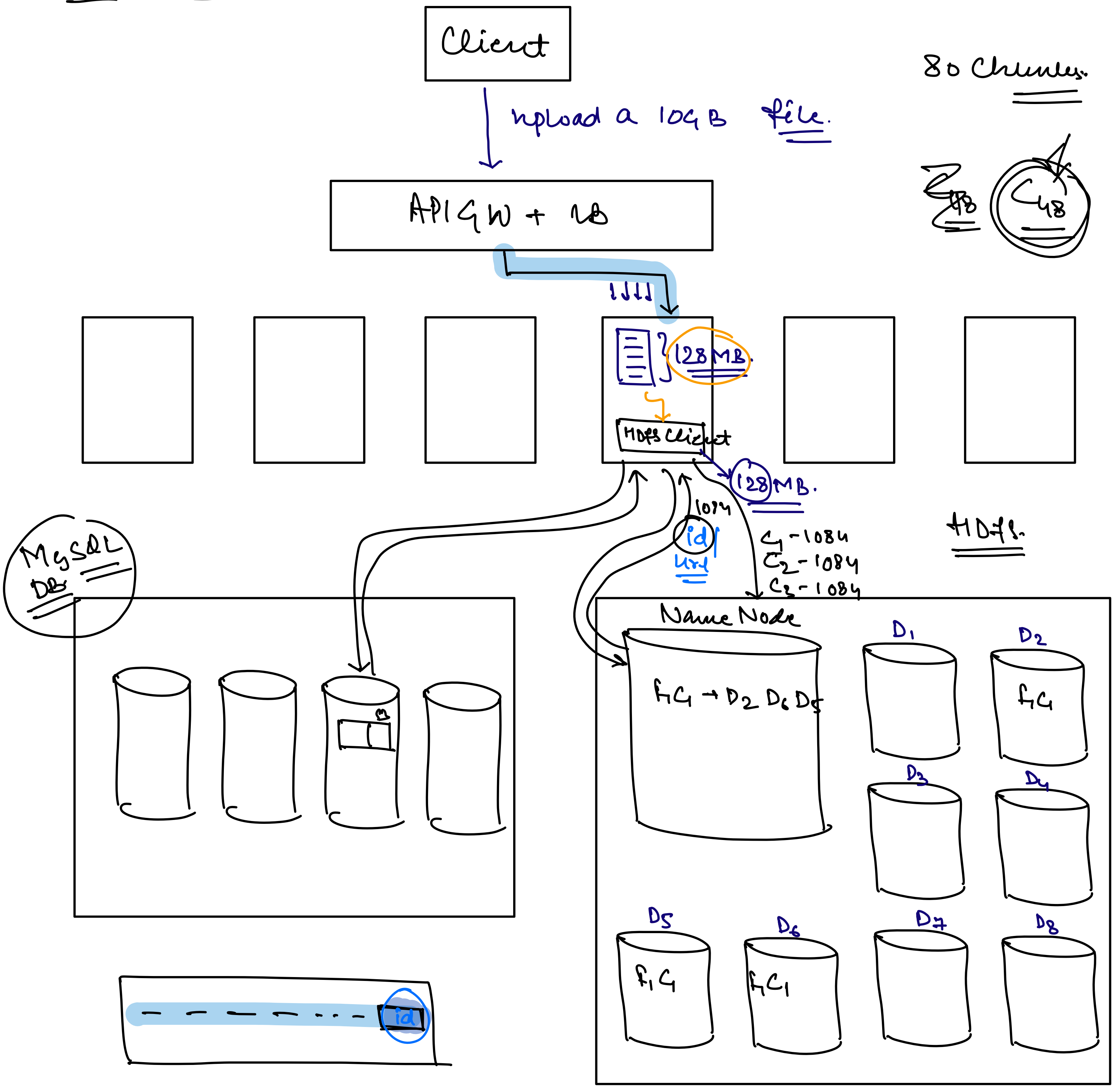
Rack Aware Algo

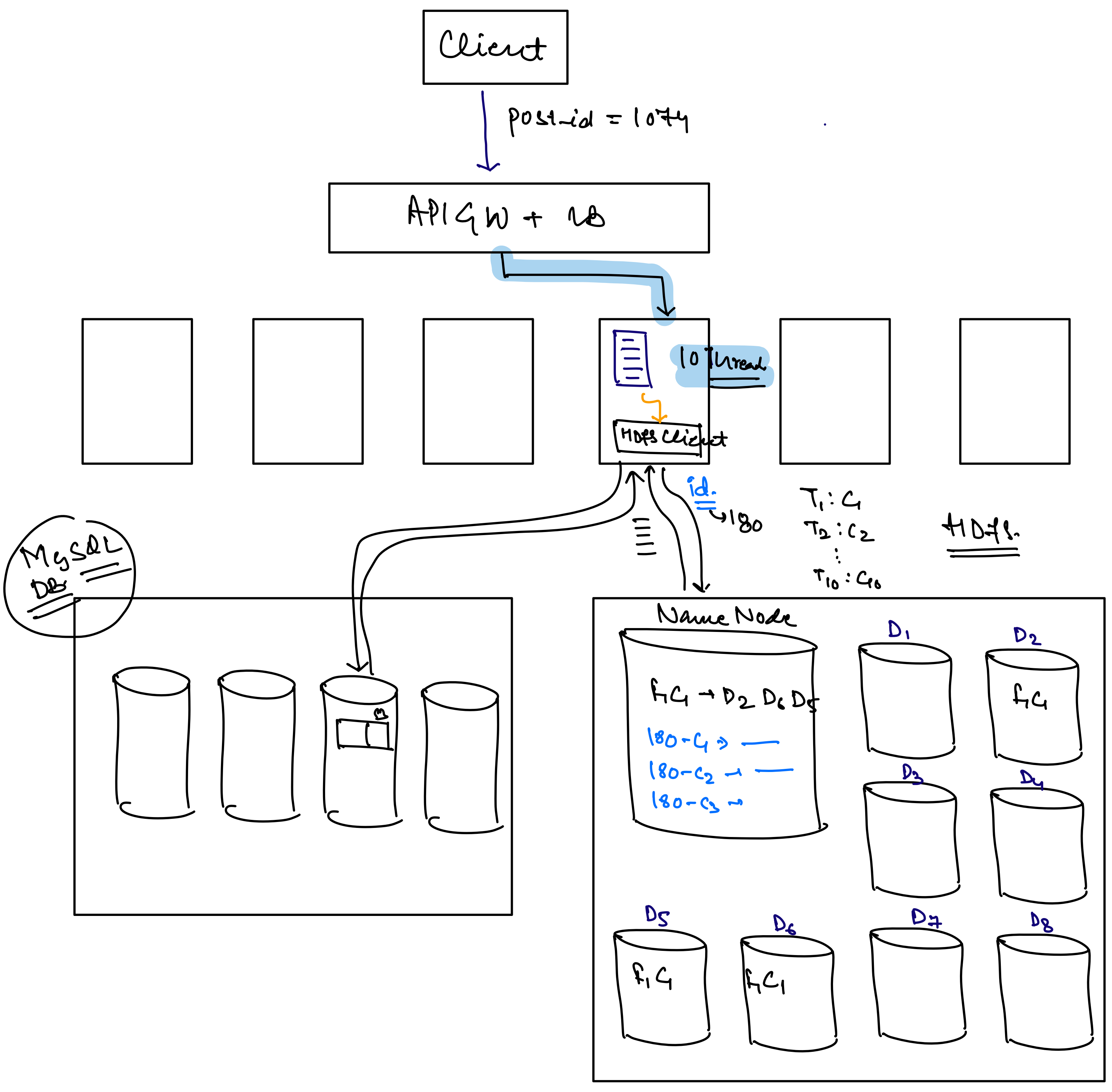


For more reliability, HDFS keeps data on different racks so that we do not loose our data even if a rack goes down.

→ Avoid replicating same chunks within the same rack.

Upload flow.




Client
post_id = 1074
APIGW + LB
10 thread
HDFS Client
id.
180
T1 : C1
T2 : C2
T10 : C10
HDFS.
MySQL DB
Name Node
f1C1 → D2 D6 D5
180-C1 →
180-C2 →
180-C3 →
D1
D2
f1C1
D3
D4
D5
f1C1
D6
f1C1
D7
D8