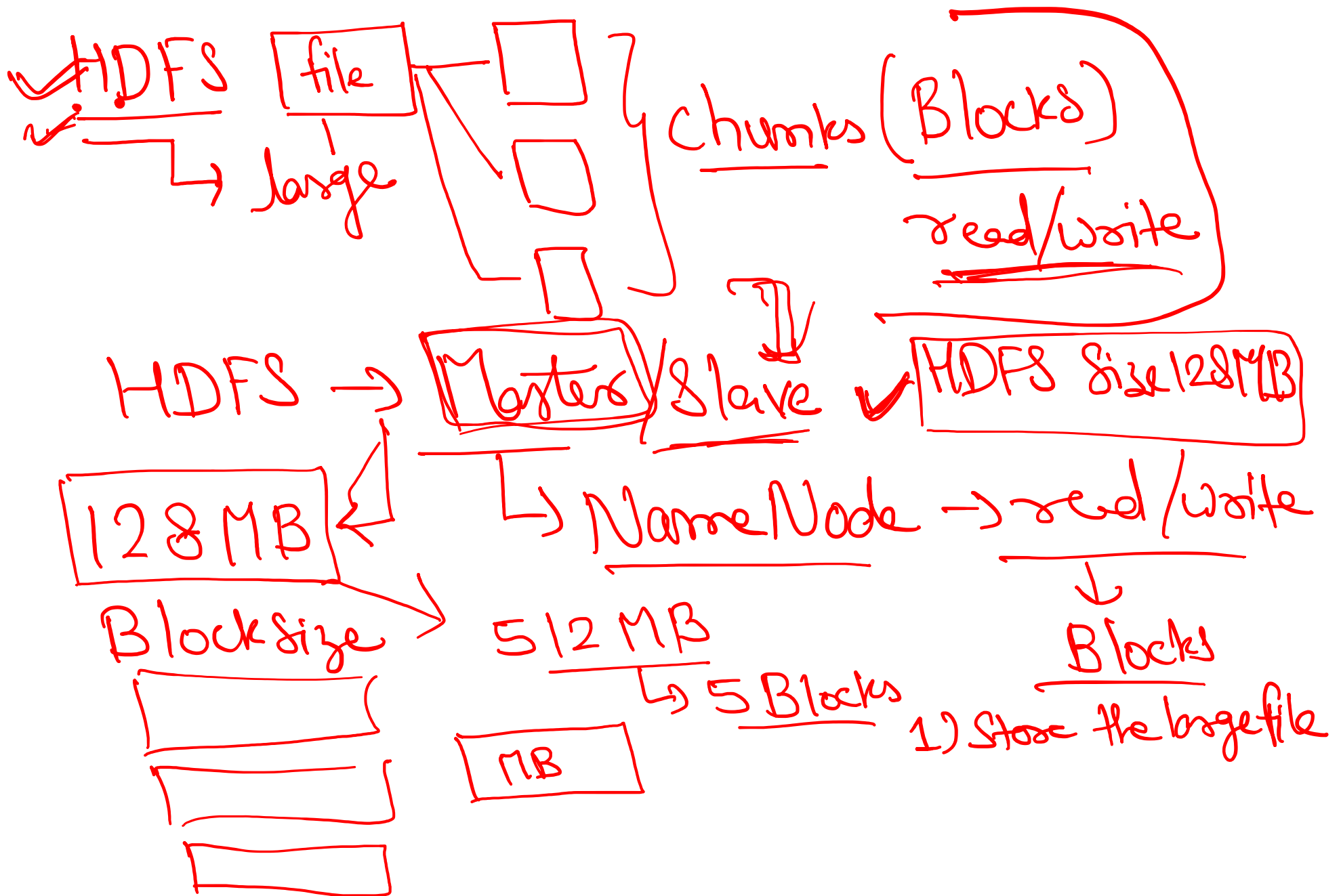- More data are being collected and stored
- Open source code
- Commodity hardware / Cloud

→
- High-Volume
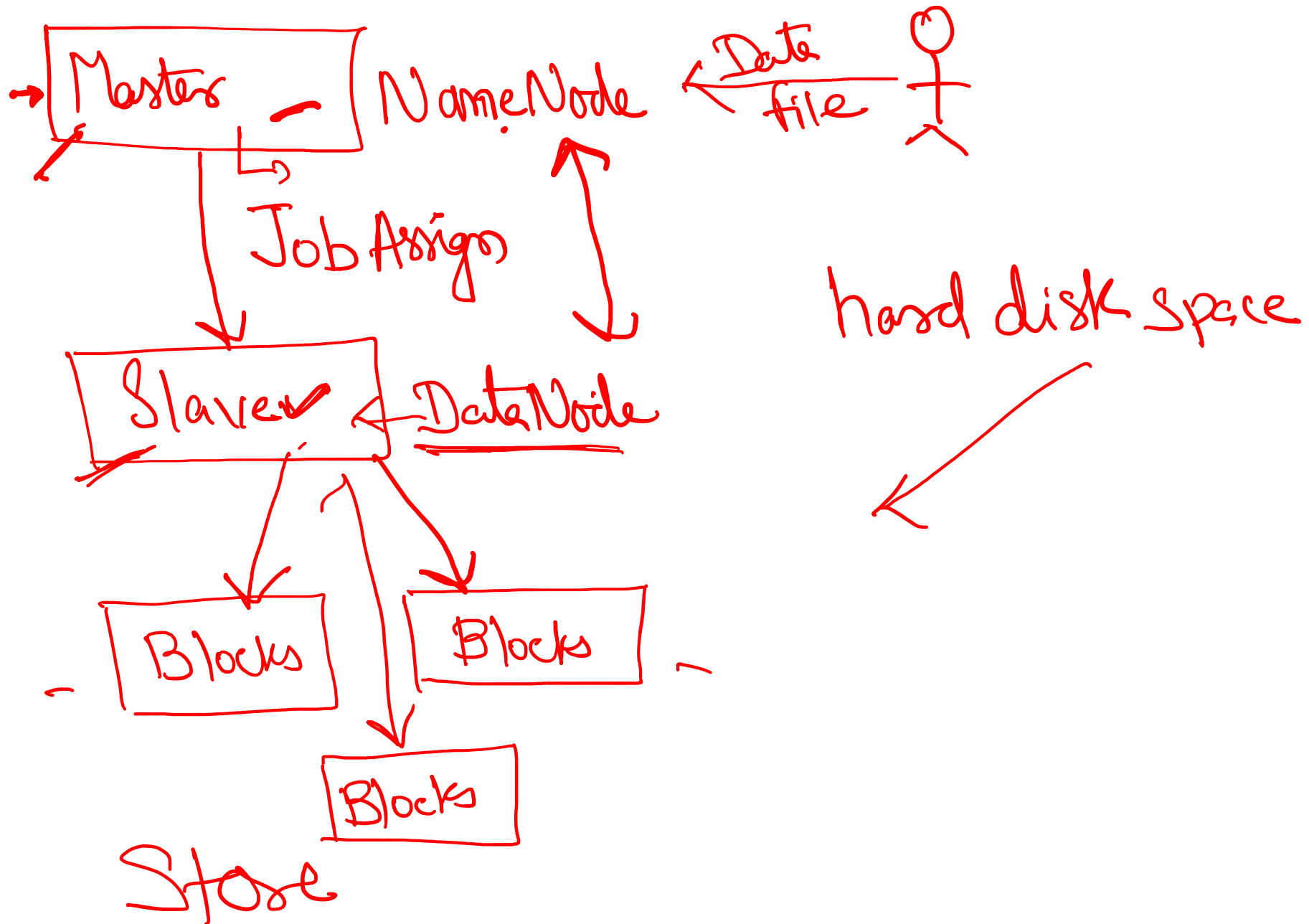- High-Velocity
- High-Variety

7 V's

Value is bigdata

→ Artificial Intelligence

1

— Data Blocks ✓

✓ HDFS [file] □ } Chunks (Blocks)
└→ large   □
           □   read/write

HDFS → [Masters] Slave ✓ [HDFS Size 128MB]

[128 MB] └→ NameNode → read/write
                               ↓
Block Size → 512 MB   Blocks
                 └→ 5 Blocks
        [MB]   1) Store the large file

fstorage - Stores the snapshot of fs

Namenode - holds metadata for HDFS in blocks
info for only

```
┌─────────────┐                          ┌──────────────────────┐
│  Namenode   │ ───────────────────────> │ Secondary Namenode   │
└─────────────┘                          └──────────────────────┘
      │                                              │
      │ Update the                                   │
      │ editlogs                                     │
      ▼                                              ▼
┌─────────────┐     Replication          ┌──────────────────────┐
│  fsimage    │ ──────────────────────>  │      fsimage         │
│             │        Process           │                      │
└─────────────┘                          └──────────────────────┘
```
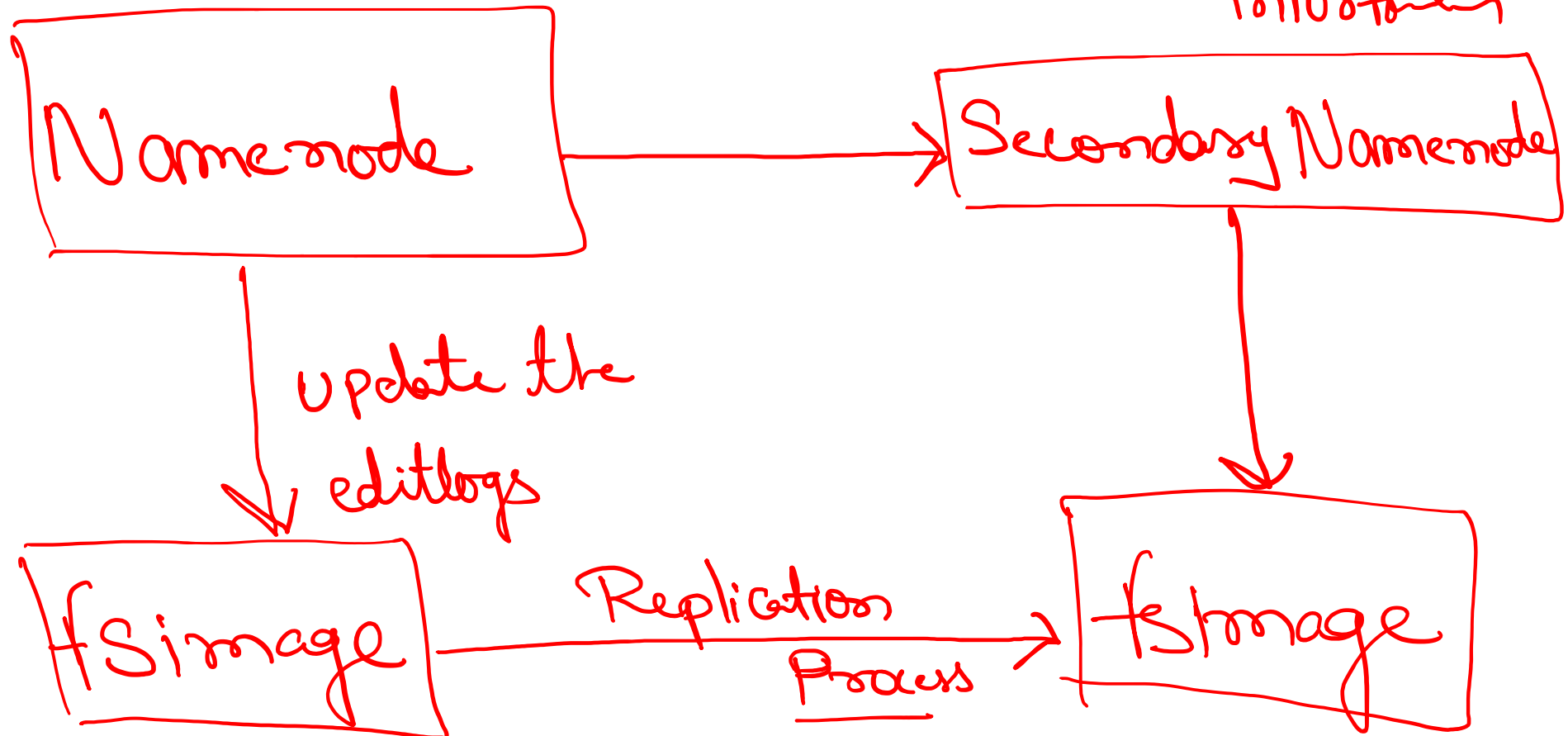
Editlogs → keeps tracks of every change of HDFS.

Comp.

Job

Client

Job Tracker → NameNode

Monitor

mapReduce
→ Programming

Task Tracker

Slave

Block

Block

Block

Block Block

Job Tracker is down in HDFS

mapReduce → halt

Masters ⟶ Job Tracker

Task Trac

Task Tracker

Task Trac.

<u>Monitor</u>

Slave

| Blck | Blck | Blck | Blck |
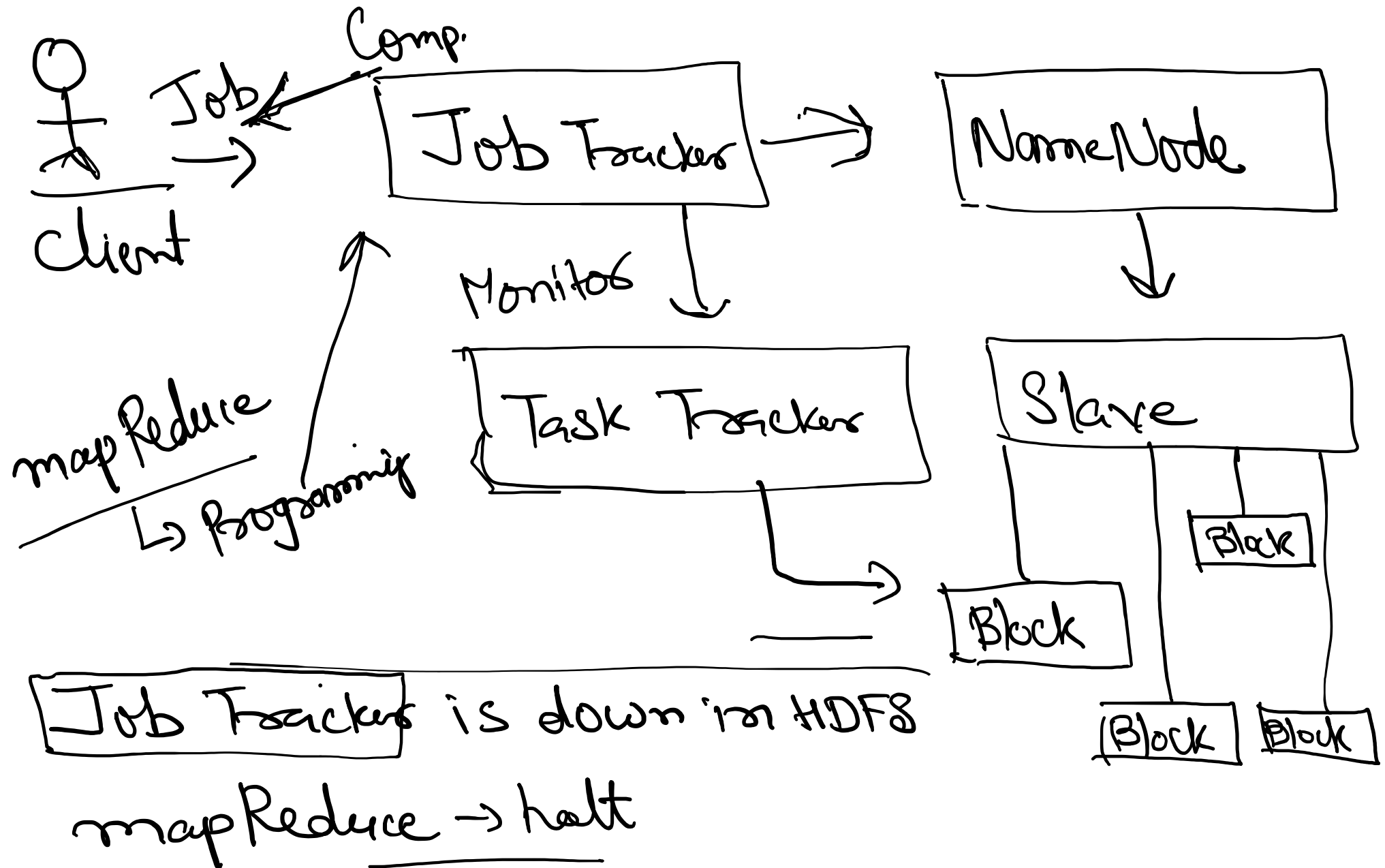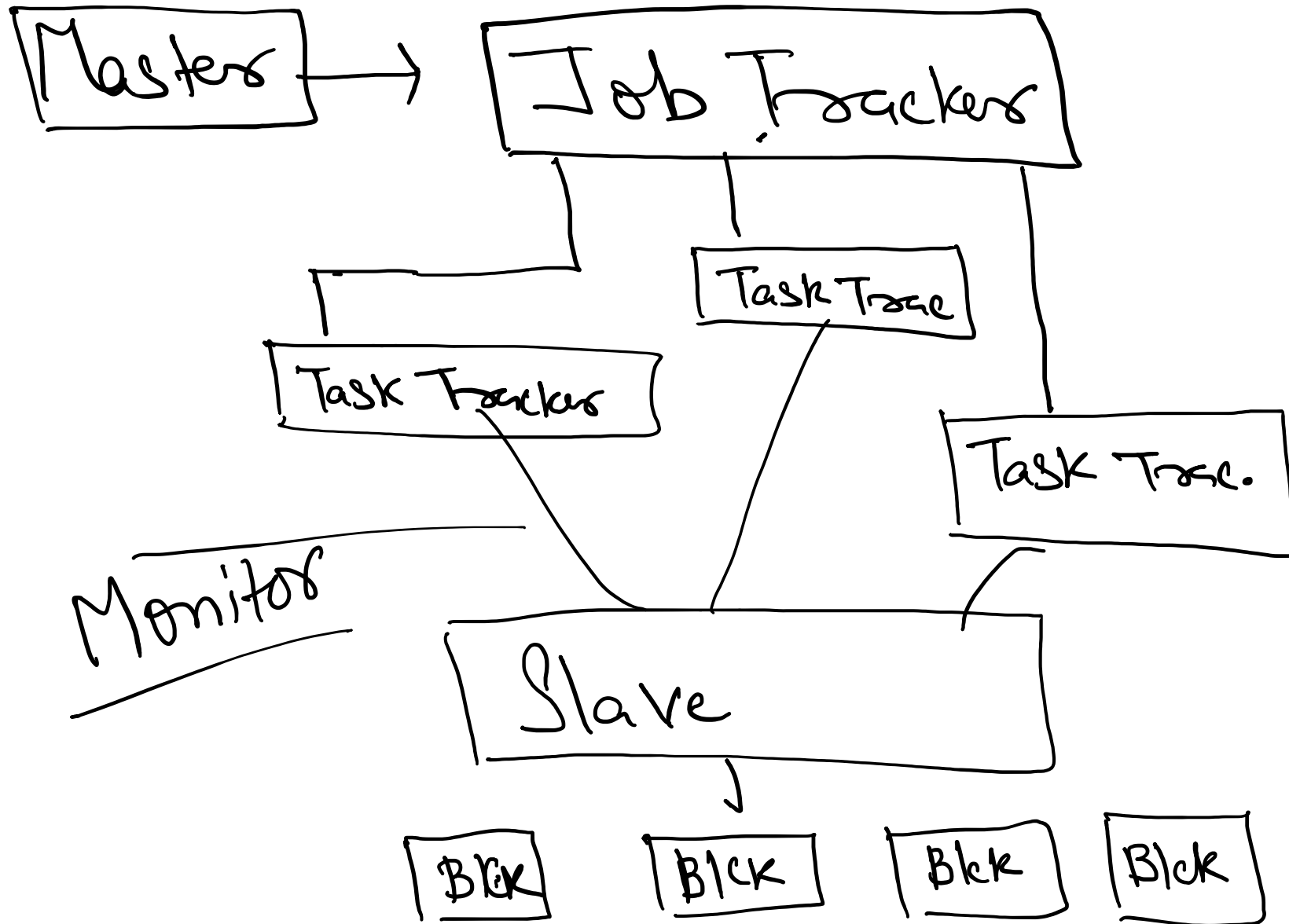
Apache ⟹ big data analytics, process tasks to be broken down into smaller tasks that can be performed into parallel by using any algo like mapReduce.
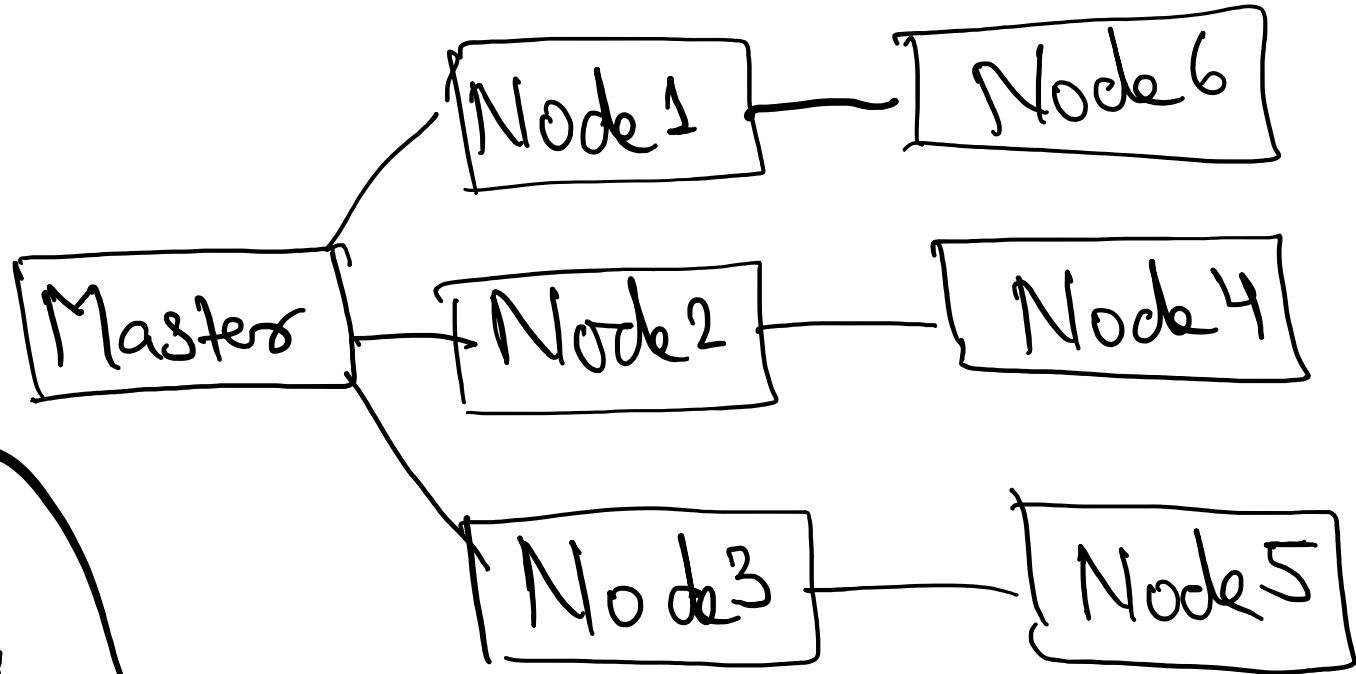
Master ← Server

Slave ← VM's → Hadoop clusters

Client

↑ (person) → Client

Size
{ Number of Nodes
{ Config of each type Node
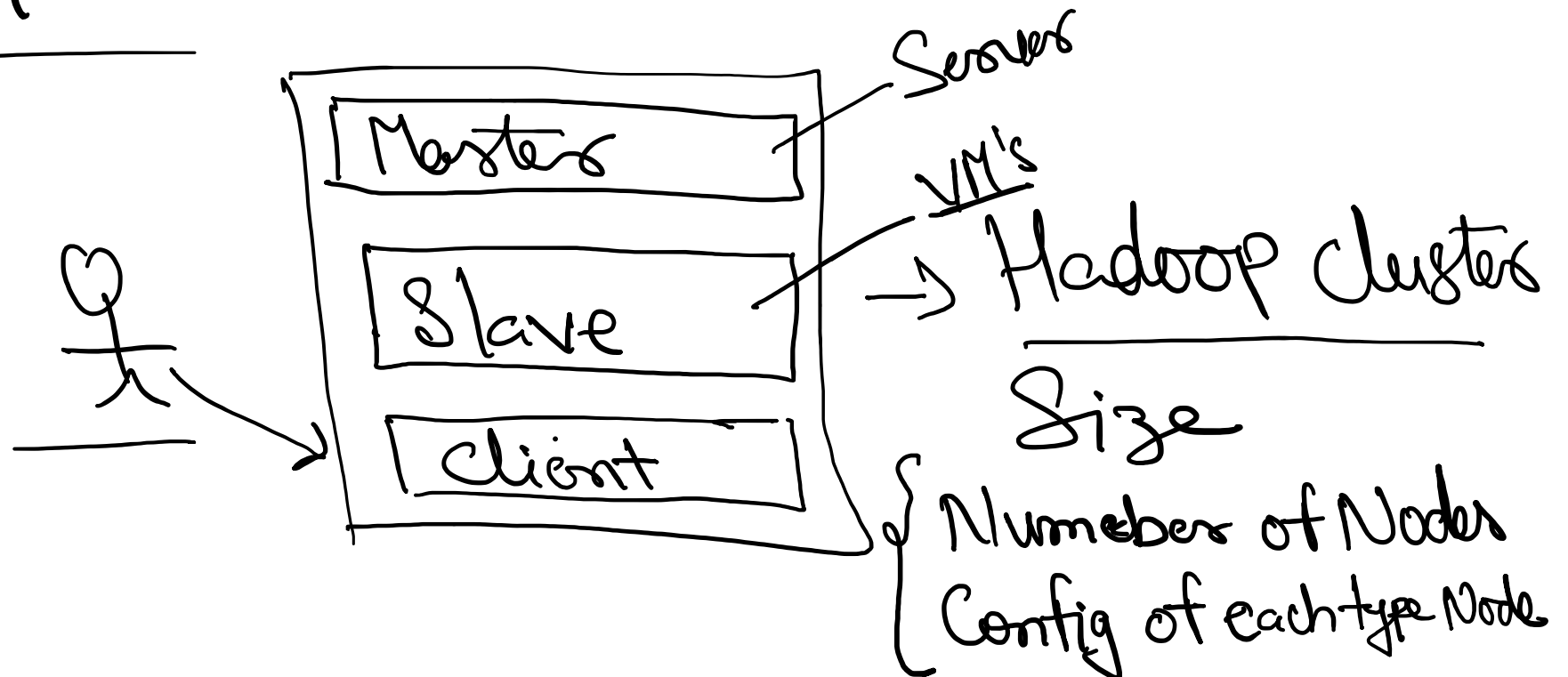
Data Store — Struct, Semi-Struct & Unstruct Data

Schema =>

Master

Node1 —— Node6

Node2 —— Node4

Node3 —— Node5

Slaves

Properties
- Scalability
- flexibility
- Speed
- No data loss — Economial