Introduction to Big Data

ECAP456

Dr. Rajni Bhalla

Associate Professor

Learning Outcomes

After this lecture, you will be able to

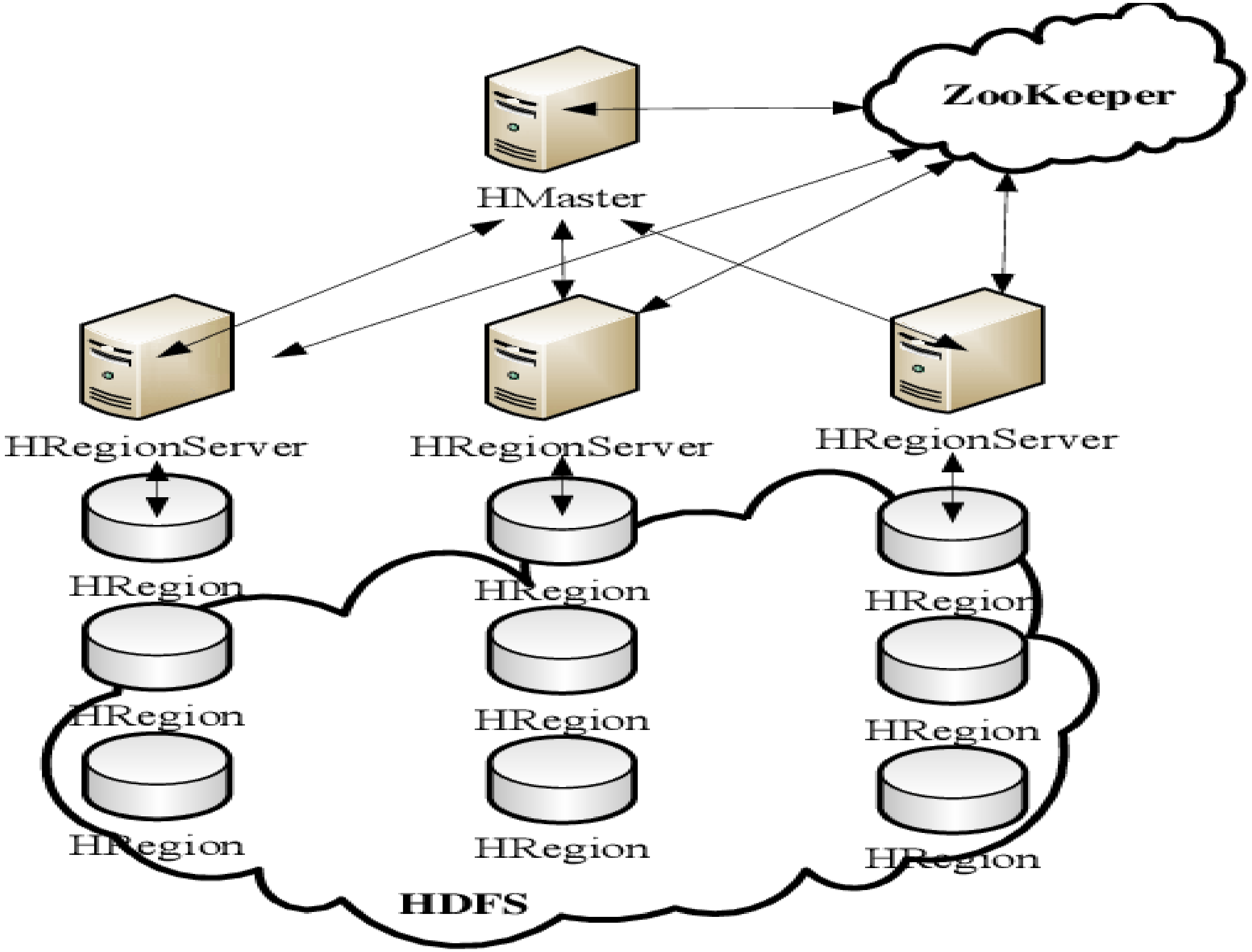
- learn HDFS
- HDFS Architecture
- goals of HDFS

Introduction

Introduction

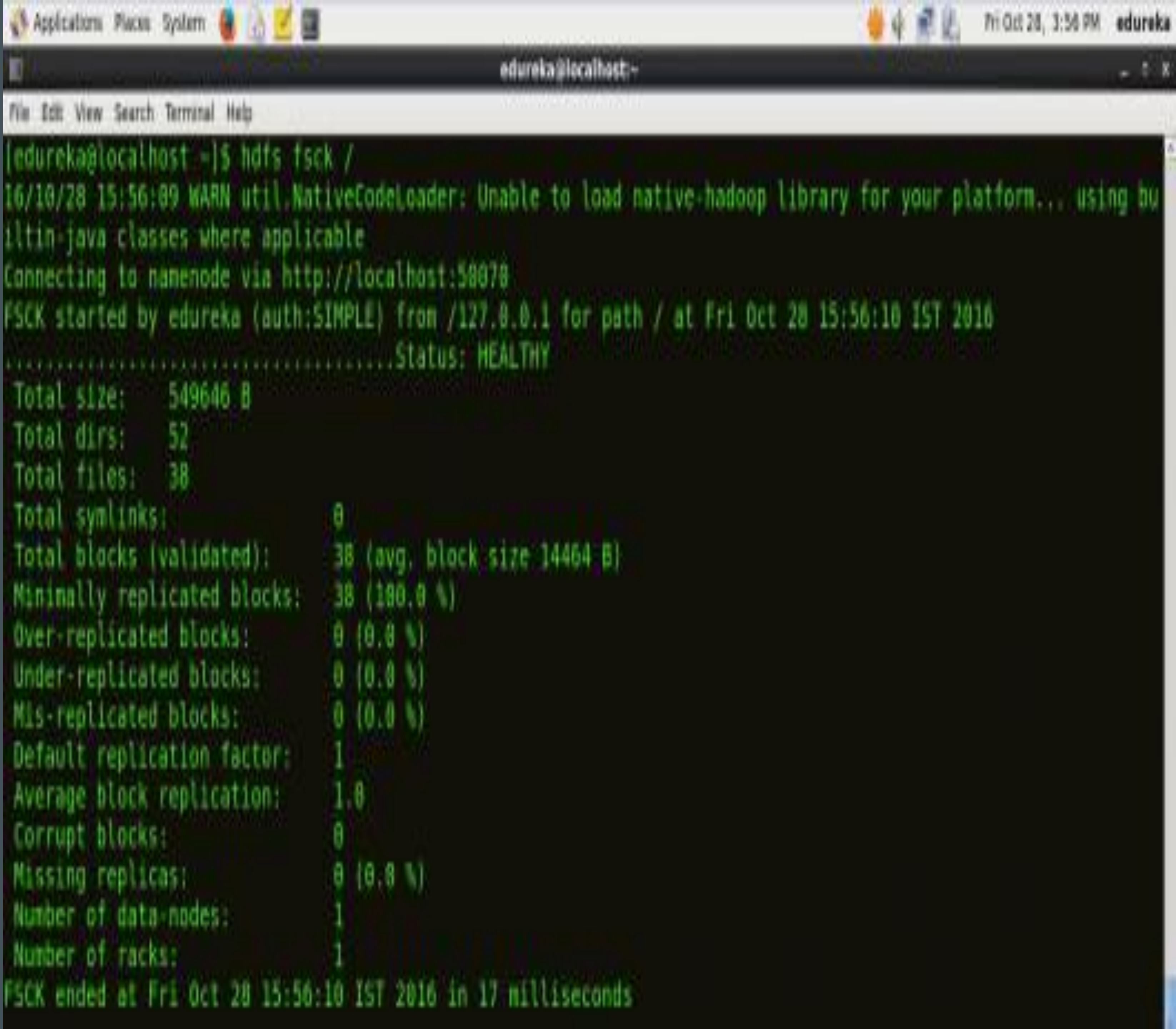
- Large amount of data.
- Easier access.
- To store such huge data, the files are stored across multiple machines.
- Stored in redundant fashion to rescue.
- Makes applications available to parallel processing.

Features of HDFS



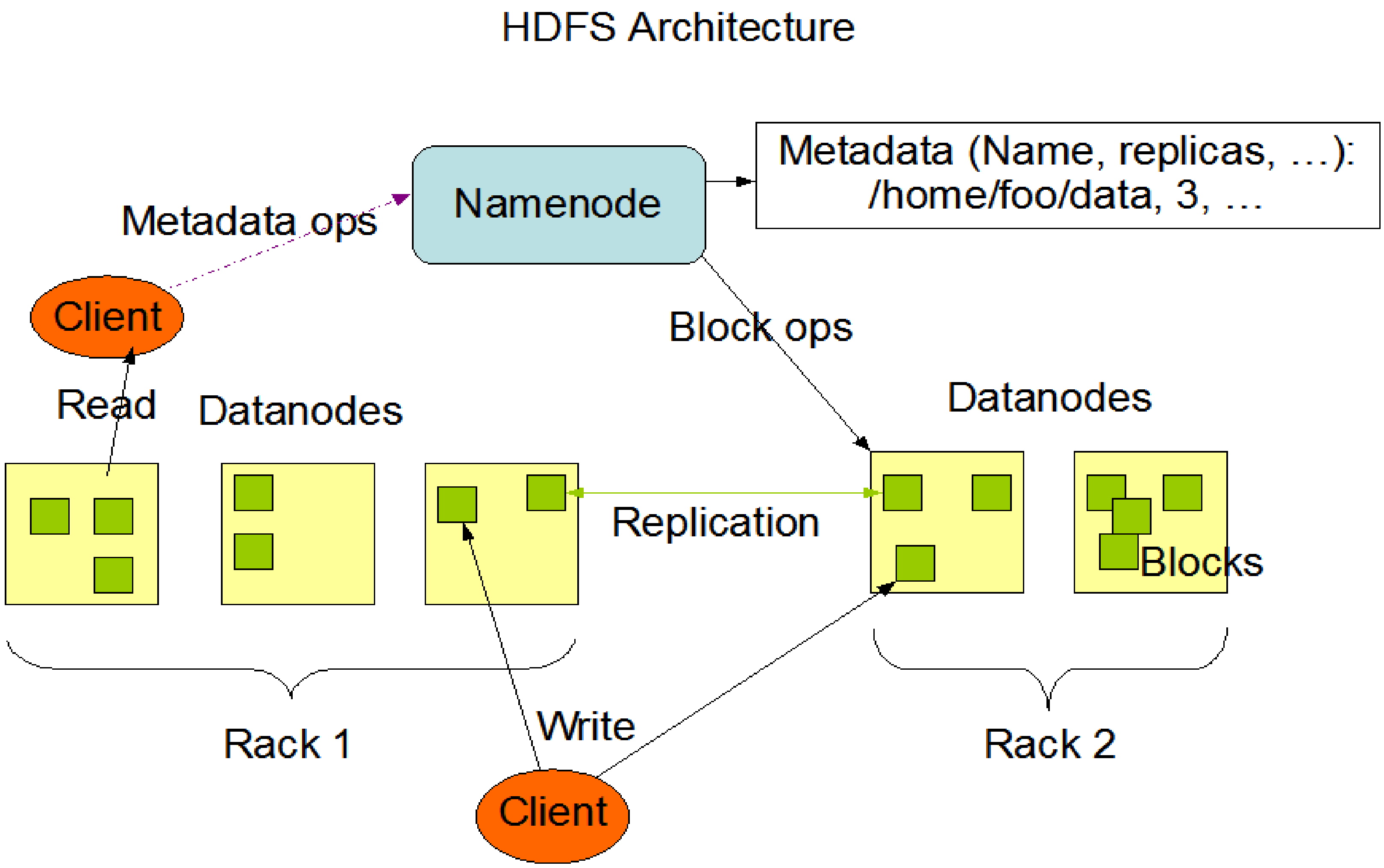
Distributed Storage and Processing.

Features of HDFS (Command Interface)


Applications Places System
Fri Oct 28, 3:56 PM edureka
edureka@localhost:~
File Edit View Search Terminal Help

```
[edureka@localhost ~]$ hdfs fsck /
16/10/28 15:56:09 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using bu
iltin-java classes where applicable
Connecting to namenode via http://localhost:50070
FSCK started by edureka (auth:SIMPLE) from /127.0.0.1 for path / at Fri Oct 28 15:56:10 IST 2016
......................................Status: HEALTHY
 Total size:    549646 B
 Total dirs:    52
 Total files:   38
 Total symlinks:                0
 Total blocks (validated):      38 (avg. block size 14464 B)
 Minimally replicated blocks:   38 (100.0 %)
 Over-replicated blocks:        0 (0.0 %)
 Under-replicated blocks:       0 (0.0 %)
 Mis-replicated blocks:         0 (0.0 %)
 Default replication factor:    1
 Average block replication:     1.0
 Corrupt blocks:                0
 Missing replicas:              0 (0.0 %)
 Number of data-nodes:          1
 Number of racks:               1
FSCK ended at Fri Oct 28 15:56:10 IST 2016 in 17 milliseconds
```

Features of HDFS

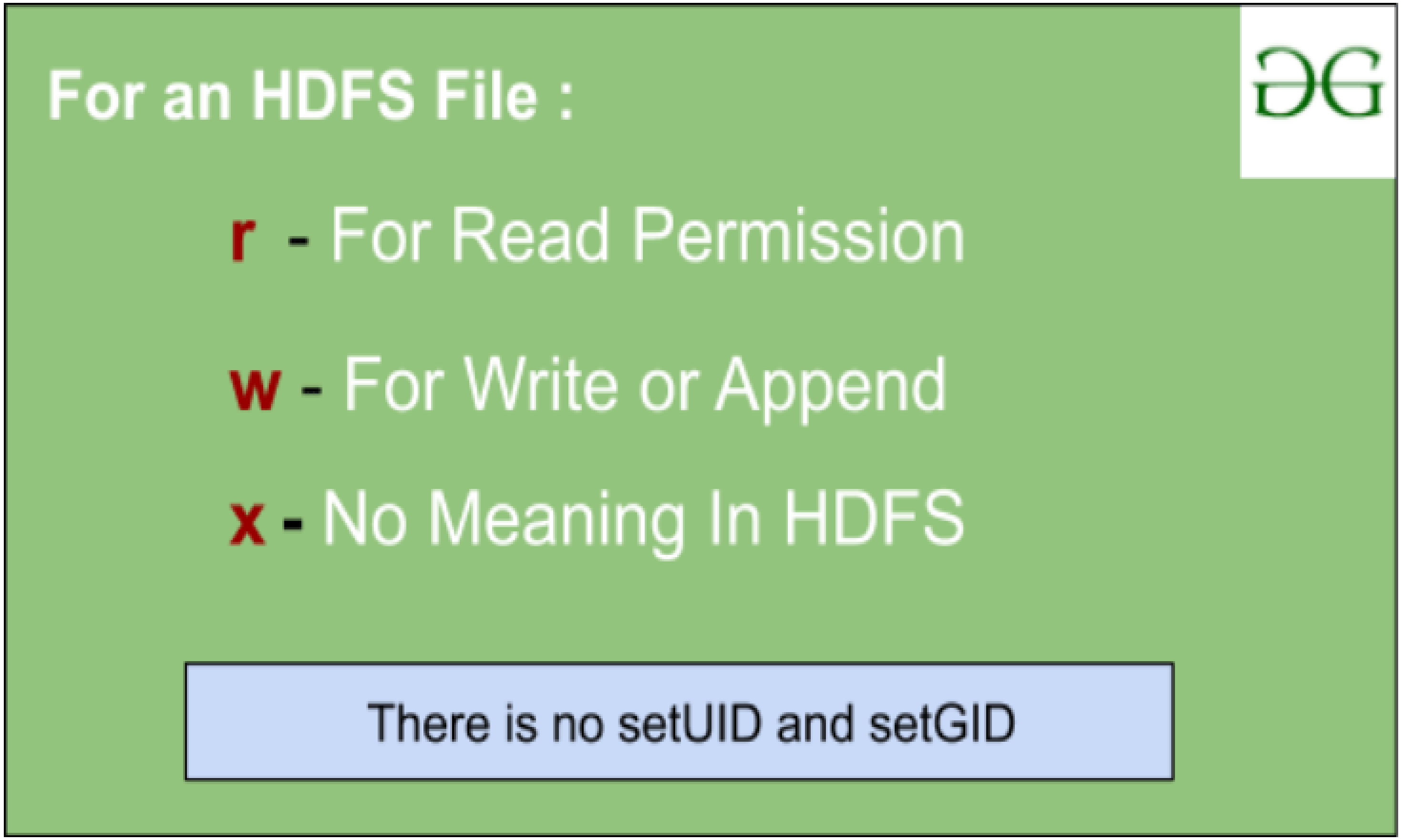


Built-in Servers of Namenode and Datanode

Features of HDFS

Streaming access to file system data.

Features of HDFS



For an HDFS File :

r - For Read Permission

w - For Write or Append

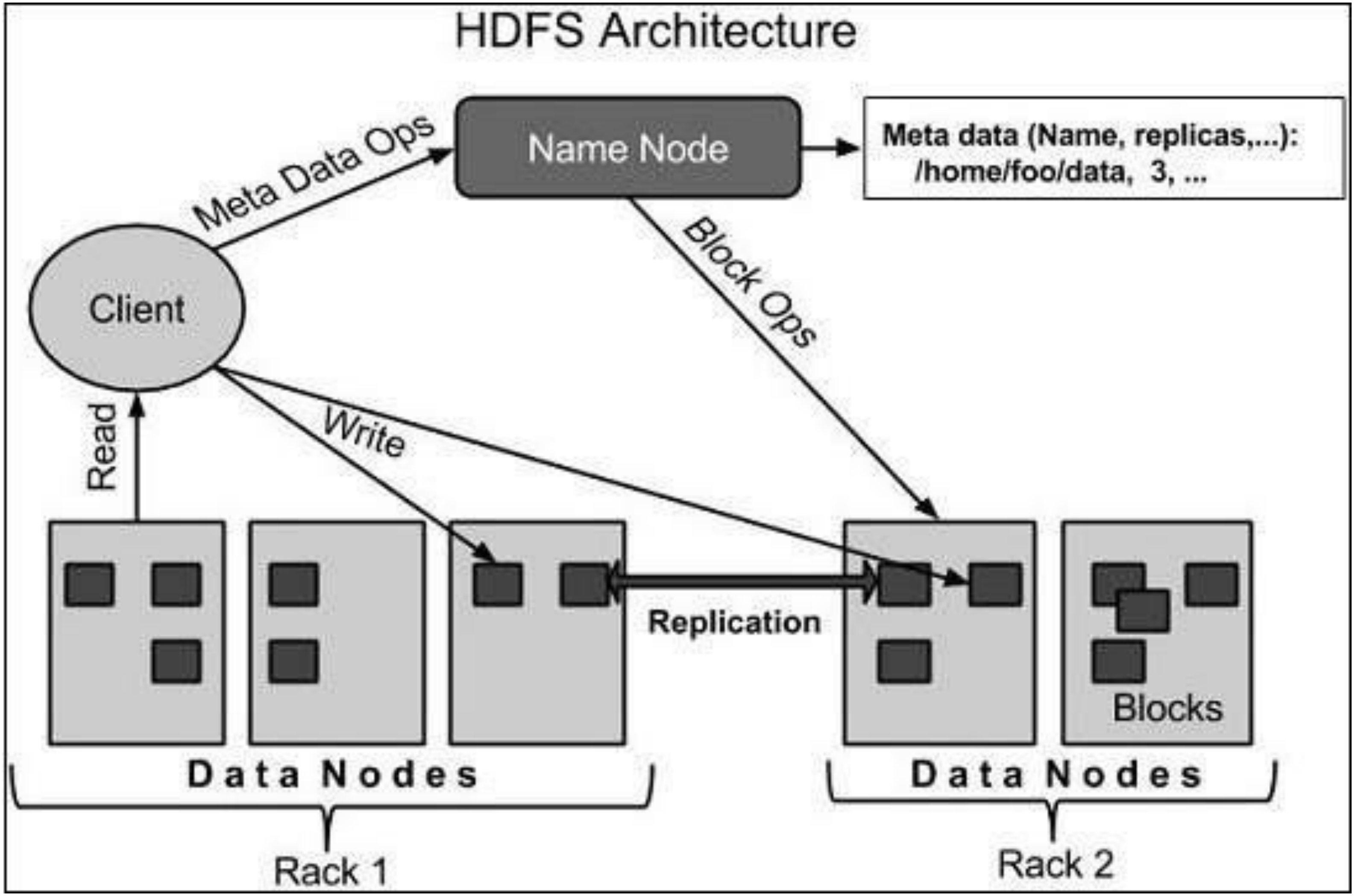
x - No Meaning In HDFS

There is no setUID and setGID

File Permissions and Authentication.

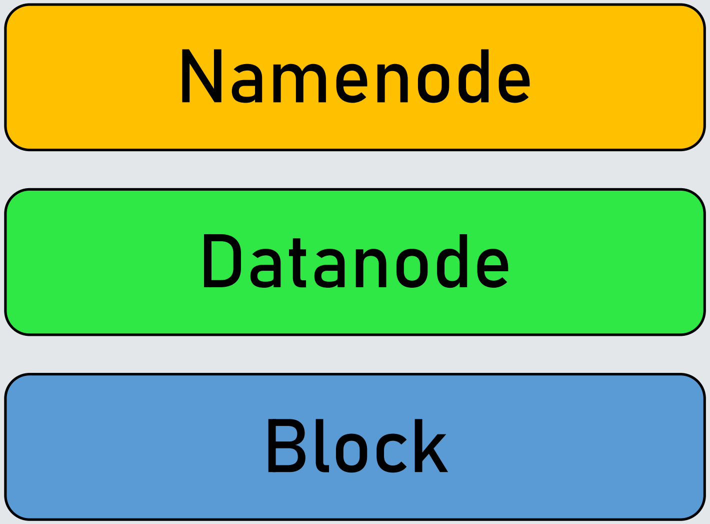
HDFS Architecture

HDFS Architecture



HDFS Architecture

HDFS follows the master-slave architecture



Namenode

Datanode

Block

HDFS Architecture (Namenode)

- Centerpiece of an HDFS file system.
- Directory tree of all files in the file system.
- Tracks where across the cluster the file data is kept.
- It does not store the data of these files itself.
- Client applications talk to the NameNode.
- NameNode responds the successful requests.

HDFS Architecture

NAMENODE

- Single Point of Failure
- When the Name Node goes down, _______________
- Optional SecondaryNameNode
- Creates checkpoints of the namespace
- BackupNameNode

HDFS Architecture

Name Node works as Master in Hadoop cluster. Below listed are the main function performed by Name Node:

1. Stores metadata of actual data.
2. Manages File system namespace.
3. Regulates client access request for actual file data file.

HDFS Architecture

Name Node works as Master in Hadoop cluster. Below listed are the main function performed by Name Node:

4. Assign work to Slaves (DataNode).
5. Executes file system name space operation like opening/closing files, renaming files and directories.

HDFS Architecture

Name Node works as Master in Hadoop cluster. Below listed are the main function performed by Name Node:

6. As Name node keep metadata in memory for fast retrieval, the huge amount of memory is required for its operation. This should be hosted on reliable hardware.

HDFS Architecture

Data Node works as Slave in Hadoop cluster. Below listed are the main function performed by Data Node:

1. Actually stores Business data.
2. his is actual worker node were Read/Write/Data processing is handled.
3. Upon instruction from Master, it performs creation/replication/deletion of data blocks.

HDFS Architecture

Data Node works as Slave in Hadoop cluster. Below listed are the main function performed by Data Node:

4. As all the Business data is stored on Data Node, the huge amount of storage is required for its operation. Commodity hardware can be used for hosting Data Node.

HDFS Architecture

- Storing the actual data in HDFS.
- DataNode is also known as the Slave
- NameNode and DataNode are in constant communication.
- When a DataNode starts up
- When a DataNode is down

HDFS Architecture

- DataNode is usually configured with a lot of hard disk space
- DataNode periodically send HEARTBEATS to NameNode

HDFS Architecture

Block

- Stored in the files of HDFS.
- Divided into one or more segments.
- HDFS can read or write.
- Block size is 64MB.
- Increased as per the need to change in HDFS configuration.

Goals of HDFS

Fault Detection and Recovery

Huge Datasets

Hardware at Data

That’s all for now…