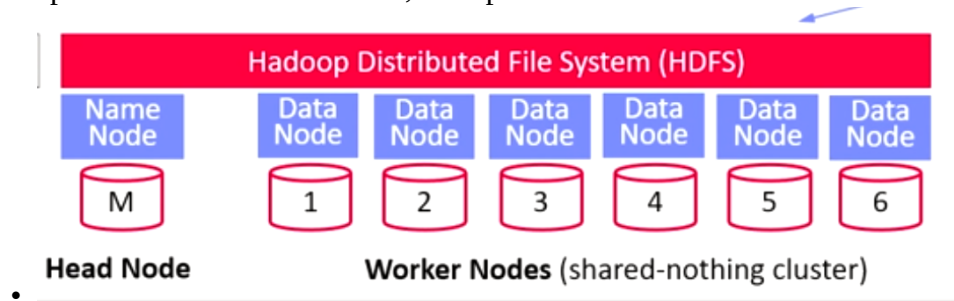


HDFS Overview

- distributed file system for large clusters and datasets
- splits files in 128 MB blocks, 3x replicated and distributed



HDFS NameNode

Master daemon that manages file system namespace and access by clients

Metadata for all files (e.g., replication, permissions, sizes, block ids, etc)

FSImage: checkpoint of FS namespace

EditLog: **write-ahead-log (WAL)** of file write operations (merged on startup)



HDFS DataNode

Worker daemon per cluster node that manages block storage (list of disks)

Block creation, deletion, replication as individual files in local FS

On startup: scan local blocks and send **block report** to name node

Serving block read and write requests

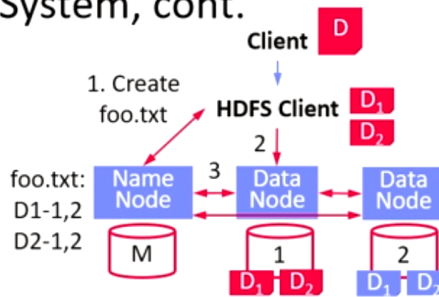
Send heartbeats to NameNode (capacity, current transfers) and receives replies (replication, removal of block replicas)

CRUD Operations

Hadoop Distributed File System, cont.

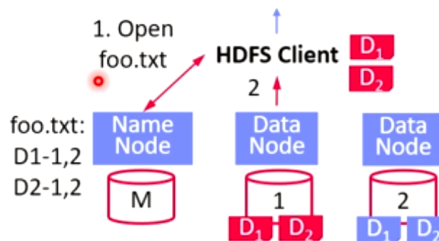
▪ HDFS Write

- #1 Client RPC to NameNode to create file → lease/replica DNs
- #2 Write blocks to DNs, pipelined replication to other DNs
- #3 DNs report to NN via heartbeat



▪ HDFS Read

- #1 Client RPC to NameNode to open file → DNs for blocks
- #2 Read blocks sequentially from closest DN w/ block
- InputFormats and RecordReaders as abstraction for multi-part files (incl. compression/encryption)

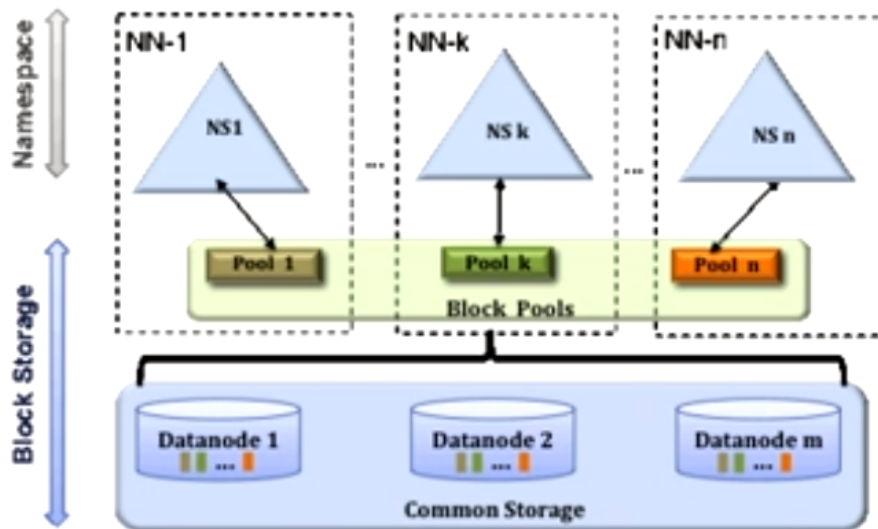


Data Locality

- HDFS is rack-aware
- schedule reads from closest DN
- replica placement 3x
 - local DN
 - other-rack DN
 - same-rack DN

HDFS Federation

- eliminates NN as namespace scalability bottleneck
- multiple independent NNs for name spaces
- each is responsible for subtrees of file system



Architecture

- Management (Ambari)
- Coordination / workflows (Zookeeper, Oozie)
- Storage (**HDFS**)
- Resources (**YARN**) [SoCC'13]
- Processing (**MapReduce**)

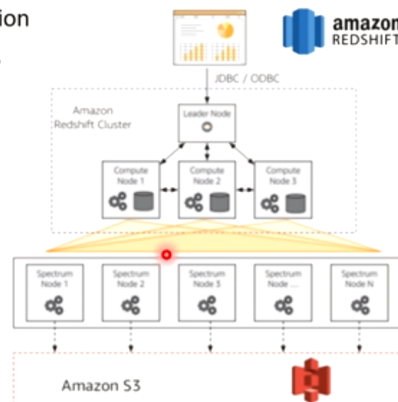


Excursus: Amazon Redshift

- Motivation** (release 02/2013)
 - Simplicity and cost-effectiveness** (fully-managed DWH at petabyte scale)
- System Architecture**
 - Data plane:** data storage and **SQL** execution
 - Control plane:** workflows for monitoring, and managing databases, AWS services
- Data Plane**
 - Leader node + sliced compute nodes in **EC2** with **local storage**
 - Replication across nodes + **S3 backup**
 - Query compilation** in C++ code
 - Support for **flat and nested files**

[Mengchu Cai et al.: Integrated Querying of SQL database data and S3 data in Amazon Redshift. *IEEE Data Eng. Bull.* 41(2) 2018]

[Nikos Armatzoglou et al.: Amazon Redshift Re-invented. *SIGMOD 2022*]



[[Distributed Data Storage]]