Tribhuvan University

Institute of Engineering

Pulchowk Campus

Distributed System

Assignment1

Submitted by:

Bishal Katuwal

075BCT028

Submitted on:

16th June, 2022

1 Assignment 1

Discuss and compare AFS, NFS, HDFS and GFS.

AFS

AFS stands for Andrew File System. It is used in Andrew Workstation developed by Carnegie Mellon University since 1983. Using a set of trusted servers, collectively called Vice, the Andrew file system presents a homogeneous, location transparent file name space to workstations. The file name space on an Andrew workstation is partitioned into a shared and a local name space. The shared namespace has important files whereas local namespace only has temp or init files. Thus a user can always find the same image of the file across all workstation. To limit server-client interaction, AFS caches a copy of file under read/write. This means all changes are done to local cache and updated after completion. This reduces network traffic. This cache is managed by a system called Venus. When a file is opened, Venus checks for a copy in its cache. If found, local file is opened. If not, a copy is cached from the original. The cache consistency is managed by callback. This means whenever a file is copied from the original, a promise is made to inform the server regarding any changes in the copy in cache. This use of callback further reduce client-server interaction. AFS focuses in minimizing client-server interaction as scalability is the dominant feature of AFS.

NFS

NFS stands for Network File System. It was developed by Sun Microsystems in 1985 and it is the most widely used file system since. Originally NFS was motivated by wanting to extend a Unix file system to a distributed environment, however it has been extended to support other OS too. NFS systems are stateless. This means that NFS doesn't store the state of client accesses to the files. This possesses some pros and cons. The main advantage being immediate backup in case of system failure. However the cons outweigh the pros. Clients can have conflicting copies of same file. Update process is slow. Cannot manage atomicity of transactions. Basically, an NFS is a file system that allows users to mount file systems over a network and interact with them as if they car locally mounted. NFS is a centralized file system pretending to be decentralized. Currently NFS have three versions

- NFSv2
- NFSv3
- NFSv4

HDFS

HDFS stands for Hadoop Distributed File System. It is a distributed file system designed to run on low-cost hardware. It has a master slave architecture. Every HDFS has a master(Namenode) and multiple slaves(Datanode)). The Namenode manages file systems, namespaces and access controls. It also deals with opening, closing and renaming files. The Datanode deals with read and write operations. Similarly, the namenode divides the system into files such that user data can be written into these files. It maps these blocks to datanodes. The datanodes then

deal with creation, deletion and replication of these blocks. A deployment of HDFS has a dedicated device for Namenode and each device is cluster has a datanode. It is highly fault tolerant and has high throughput and is mainly used for systems with large dataset.

GFS

GFS stands fpr Google File System. The GFS consists of large number of storage devices accessed by equally large number of clients. In GFS, files are divided into fixed sized chunks. Thus a cluster consists of a master and multiple chunkservers.

The chunks have immutable and globally unique handle assigned by master during creation. The chunkservers then store chunks on local disks and read/write chunkdata as specified by chunkhandler. The chunkservers can be supported in the same device as the client leading to flexibility. To increase reliability, these chunks are replicated into multiple chunkservers. The use of chunkservers also help eliminate the need of local file cacheing.

The master deals with metadata like chunk creation, access control, chunk mapping, file mapping to chunk, etc. And the chunkservers deal with all data-related operations. When a client enters GFS, it issues the requests to the master however the data-related and actual file operations are handled by chunkservers.

Comparison

SN	Basis	NFS	AFS	GFS	HDFS
1	Scalability	Limited, more overhead	Scalable	Scalable	Scalable
2	Reliability	Not reliable	Reliable	Reliable	Reliable
3	Flexibility	Not flexible	Limited flexibility	Flexible	Flexible
4	Transparency	Not transparent	Limited transparency	Transparent	Transparent
5	Fault Tolerance	Limited Tolerance	Limited Tolerance	Tolerant	Tolerant
6	Security	Simple Authentication	Token Authentication	Authentication, Chunk Server side 32-bit check summing	Authentication, Client side 32-bit check summing
7	Architecture	Centralized	Decentralized, Parallel	Decentralized, Parallel	Decentralized, Parallel
8	Process	Stateless	Stateless	Stateful	Stateful
9	Naming	Central File Server	Central Metadata Server	Central Metadata Server	Central Metadata Server