Google File System

BY RICHARD LIAO

NOVEMBER 23, 2013

Ghemawat, Sanjay, Howard Gobioff, and Shun-Tak Leung. "The Google file system." *ACM SIGOPS Operating Systems Review*. Vol. 37. No. 5. ACM, 2003.

Main Idea

The Google File System(GFS) is a solution to meet the large amount of storage that Google requires.

The GFS uses cheap hardware with large amounts of redundancy, maintaining reliability while keeping costs down

The GFS is scalable

Intelligent design allows multiple users to access and alter files, sometimes the same file, without conflict.

Designed for large files, but allows smaller ones too

Implementation

The Master: A single server that contains the location to all the chunks, but stores no actual data. Only Metadata and is in charge of a single cluster.

- When a request comes in for data, the master will send back which chunk server has the file.
- The Master decides where the files are stored, but the chunk servers have the final say in what is stored.
- After an operation is performed, the master will cache that location for a time to speed up any subsequent requests.

Chunk Servers: Servers that store actual data.

- Chunk servers break down files into 64 MB chunks.
- Each chunk is stored a minimum of 3 times, but can be increased on request
- Chunk servers will periodically report its state to the master
- Chunk servers use cheap linux based hardware and are expected to fail. Every chunk stored on 3 different chunk servers guarantees that a chunk is never inaccessible

Operation Logs: Operation Logs is a log of changes in metadata

- The operation log will keep a record of metadata and make a logical time line for concurrent alterations
- Files and Chunks, and their versions are stored by the logical timeline the operations log creates.
- If the master is corrupted, the operations log is critical in recovery. By creating checkpoints, the master can be quickly restored

Analysis

The GFS has a very intelligent design philosophy.

- The use of cheap hardware and emphasize redundancy by using master and chunk servers. It allows Google to maintain large amounts of storage without spending large amounts of money on maintenance.

  This is a smart choice. Instead of having large amounts of down time and taking a hit to productivity, they accept that hardware will fail and ensure that they have a backup

The GFS is fast

- Since the Master only contains pointers to file locations, the master doesn't bottleneck operations
- Files are pushed linearly through chunk servers to maximize the bandwidth used by each server.
- Checkpoints are created in a way that the master's resources aren't consumed.
- The master caches locations so subsequent requests for the same file is sped up

  The choices that ensure resources are always available are brilliant. While handling a large amount of operations, the system can still ensure redundancy and consistency throughout the system.

The GFS is reliable

- Everything in the GFS has redundancy.
- Chunks are stored on multiple servers to ensure that every file is always accessible
- The master can restore quickly to any checkpoint minimizing any potential down time.
- Multiple alterations to the files are available at all time to ensure the file is available

  Since the system is designed with hardware failure in mind, the small choices that ensure reliability on every end of the system.

Advantages and Disadvantages

Advantages

Speed
- The GFS is designed to use the maximum amount of bandwidth each Chunk server uses and the master is kept out of reads and writes so it doesn't bottleneck the system

Reliability
- Chunks are stored a minimum of 3 times
- The Operations log keeps track of all writes incase something fails
- Chunk servers and the master communicate changes in files to check the correctness of changes

Design philosophy
- Hardware is bound to fail. Plan for this instead of trying to prevent it.
- Buying cheap linux based hardware keeps costs down

Disadvantages

- Design philosophy
  - Buying cheap hardware creates more work; Someone has to fix the hardware failures.
- File optimization
  - The GFS is optimized for large files. While it allows small files, those operations are not the main concern

Real world Applications

Cluster A

- Used for research and development by hundreds of engineers.
- A task is initiated can run for hours and reads terabytes of data before writing the results back to the chunk.

Cluster B

- Used for production data processing.
- Tasks are longer and generate and process terabytes of data.
- Mostly automatic

Both scenarios are reading and writing on many machines.