GFS Paper Review

By: Andrew Langford

November 25th 2013

Main Idea of GFS (Google File System)

The main Idea of this paper is to explain the design and implementation of the Google File System (GFS). In a nutshell, the idea behind GFS is a distributed file system, which is a system for storing and accessing files via servers. In the case of GFS, the idea was to design it to support large volumes of data and allow for many clients to create,delete, write, read, and append to many files concurrently, while maintaining data integrity, consistency, and atomicity. GFS was intended to run on inexpensive hardware, and because of the quantity of data stored on this hardware, the system was designed with fault and error in mind. Basically, the concept was to deliver very high performance and sustained bandwidth to a large number of client machines while minimalizing any sort of error.

How GFS was Implemented

Each Google File System cluster has a single master and multiple chunk servers, with each chunk server being accessed by many clients. Any file stored is segmented into chunks (64 bits each), which is uniquely tagged by the master when it is created so that it can be identified. A chunk server stores these chunks on a local disk. Each chunk is reproduced multiple times (the default is three) on multiple chunk servers. This is done to maintain the systems reliability. The master maintains the file systems metadata, controls system activity, and communicates with each chunk server. GFS client code is linked into each application and is implemented so that clients can interact with the chunk servers to create, read, write, etc.

Analysis

-The Google File System is very efficient at storing and maintaining large amounts of data which can be accessed by multiple clients simultaneously.

-Data is pushed to client very quickly because it is sent across a pipeline of optimally selected chunk servers in a linear fashion. This optimizes the networks bandwidth.

-The system is reliable since files are broken down into small pieces and replicated across multiple chunk servers, it is nearly impossible to lose a file due to any circumstance. The operation log and master state are replicated in the same way so that the most recent state will be available at startup. No matter how a master or chunk server is terminated, the system will restore it's state. Therefore, recovery is very quick and efficient.

-Checksumming is implemented to detect data corruption, and a non corrupted replica is created to replace a corrupted one, therefore data integrity is upheld.

-Also, since data is almost always appended to a file rather than overwriting existing data, the atomicity of a file and its additions is generally upheld.

Advantages and Disadvantages of GFS

-**Advantages:**

- The system is very reliable.
- The system boasts a high performance rate.
- Specific data in the system is very available.
- Large chunk sizes reduce client/master interaction, reduce network overhead, and reduce metadata size on the master.
- Garbage collection is dependable and consistent.

-**Disadvantages:**

- Chunk servers can become flooded with activity because of large chunk sizes on small files.
- GFS is not designed for small files and small reads/writes. Performance is lacking in these areas.
- The systems capacity is limited by how much memory the master has.
- The system favors high bandwidth over low latency.

Real World Applications

- Software Development firms where multiple developers are working on the same project from different locations, and where many projects are being developed at the same time.
- Version control manager sites such as BitBucket or GitHub where multiple versions of files are stored as well as their histories.
- Applications such as Google Drive, DropBox, SkyDrive would find this useful. Pretty much any application that uses the cloud to store massive amounts of data.
- Filesharing sites such as UTorrent

Bibliography

Ghemawat, Sanjay; Gobioff, Howard; Leung, Shun-Tak (October 2003), "The Google File System", 19th Symposium on Operating Systems Principles (conference), Lake George, NY: The Association for Computing Machinery