Google File System

Paul McCusker

11/22/2013

Paper by : Sanjay Ghemawat, Howard Gobioff, and Shun-Tak Leung

Main Idea

The GFS is a distributed file system - A method of storing and accessing files based in a client/server architecture.

Hundreds or even thousands of storage machines

Necessities of a file system of this magnitude (error detection, fault tolerance, recovery)

How this file system is ultimately different from others of the same type, how it shares many qualities but has been tailored for “key observations”

Design and Implementation of the GFS

Implementation

Implemented on “commodity hardware”

Design tailored for future application, and current workloads.

The file system consists of hundreds or even thousands of storage machines

Storing all different size files (multi GB files are common)

Appending data rather than then overwriting existing (performance optimization)

Personal Analysis

Because the physical component of these file systems is so “fragile” the implementation of these large scale operations is quite interesting.

My personal opinion on the GFS is that I am quite intrigued by the extravagance of it. They have implemented a operation where more than one client can append files at the same time without any extra syncronization between the clients. This is huge for a system of this size to have compatibility and be able to work at such a rate.

Throughout the paper its stated about key points or factors that must be filled: Physical failure, Size, How data is edited, Flexibility ect.
My opinion is that if all the key points are implemented successfully then the system should be successful, which clearly it is.

Pros/Cons

Pros:
Flexibility
Hardware Requirements
Single Master
Chunk size 64 MB
Flow Control
Server availability
Fault Tolerance
Cons:
Different API from standard
Garbage Collection (timing for space)
Server reliability

Use Cases

Because only google uses the GFS real world uses are not prevalent.

Implementations of files systems are used in all areas of computing research and development.

In the GFS paper we are given two clusters one for research and development and the other for production data processing.

one that has long run time with large size data generated, and the other that runs several hours and is initiated by human users.

In both cases the task consists of many processes on many machines reading and writing many files.