

Exercise – The Google File System

Ute Wappler

Systems Engineering Group
Dresden University of Technology

Systems Engineering 2 Exercises
Summer Semester 2006

Today's Exercise

We will discuss

The Google File System [GGL03]

Objective of the Google File System

1. What is the Google File System?
For which specific environment is it optimized?

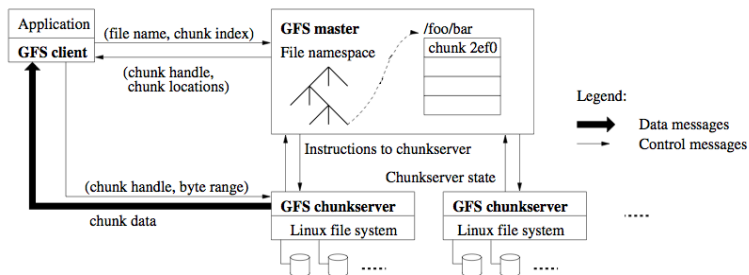
-
2. How is GFS adapted to expected applications and environment?

Interface

3. Which operations are supported by GFS?
Which of these operations distinguish GFS from other file systems?
What are these operations doing and why were they added?

Architecture

4. Use the following picture to explain the GFS architecture.
Where are files saved and how often?
Where is required metadata stored?
Describe a read operation.



Master

5. Name further tasks controlled by the master.
Why is there only one master? Isn't that a bottleneck?

Chunk Size

6. Name pros and cons for large vs. small chunks.

Replica Location

7. How does the master choose the location for replicas of chunks.

What are the objectives of re-replication and rebalancing and when are these operations executed?

Garbage Collection

8. Which types of garbage collection are done?
When are files deleted which were explicitly deleted by a client?
What are the advantages of this approach?

Metadata

9. Which metadata does the master store and where?
Where are chunk locations stored?
What is the operations log?
What is done if the operations log gets too big?

Metadata-Example

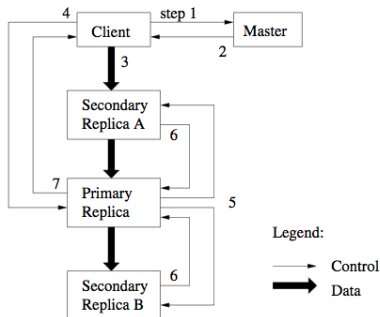
10. Assume you have 1000 files with the following file size distribution
- ▶ 50% 1 GB
 - ▶ 25% 512 MB
 - ▶ 25% 256 MB

and a chunksize of 64MB.

Give an upper bound for the memory the master has to use for storing namespace and file-to-chunck mapping.

Write Operation

11. Describe the application flow of a write operation.
How is consistency between the replicas ensured?
Why are data and control flow separated?



Consistency Model

12. Describe the consistency model implemented by GFS.
What is the notion of *consistent*?
What is the notion of *defined*?
Which type of results do successful `write` and `record append` operation in case of serial or concurrent execution generate? Defined, undefined, consistent, inconsistent?

Implications for Applications

13. How have applications to adapt to this relaxed consistency model?

Chunkserver Failure

14. What happens if a chunkserver fails?
Awhile? A long time?

Corrupted Data

15. How is corrupted data recognized and what is done to repair it?

Stale Replicas

16. What are stale replicas?

How does a replica become stale?

How are stale replicas recognized?

Is it possible that a client reads a stale replica?

Master Failure

17. What is done to make the master reliable?



Sanjay Ghemawat, Howard Gobioff, and Shun-Tak Leung.

The google file system.

In *SOSP '03: Proceedings of the nineteenth ACM symposium on Operating systems principles*, pages 29–43, New York, NY, USA, 2003. ACM Press.