Google File System

Nitesh Mor

Why Google File System?

Specific requirements of

- Use of commodity hardware (and still be reliable)
- File sizes (mostly large files)
- Specific access pattern
  - More reads
  - Large sequential writes (appends) and few random writes
  - Producer-consumer style access
- Throughput more important than latency

Architecture:

Single master, Multiple chunk-servers

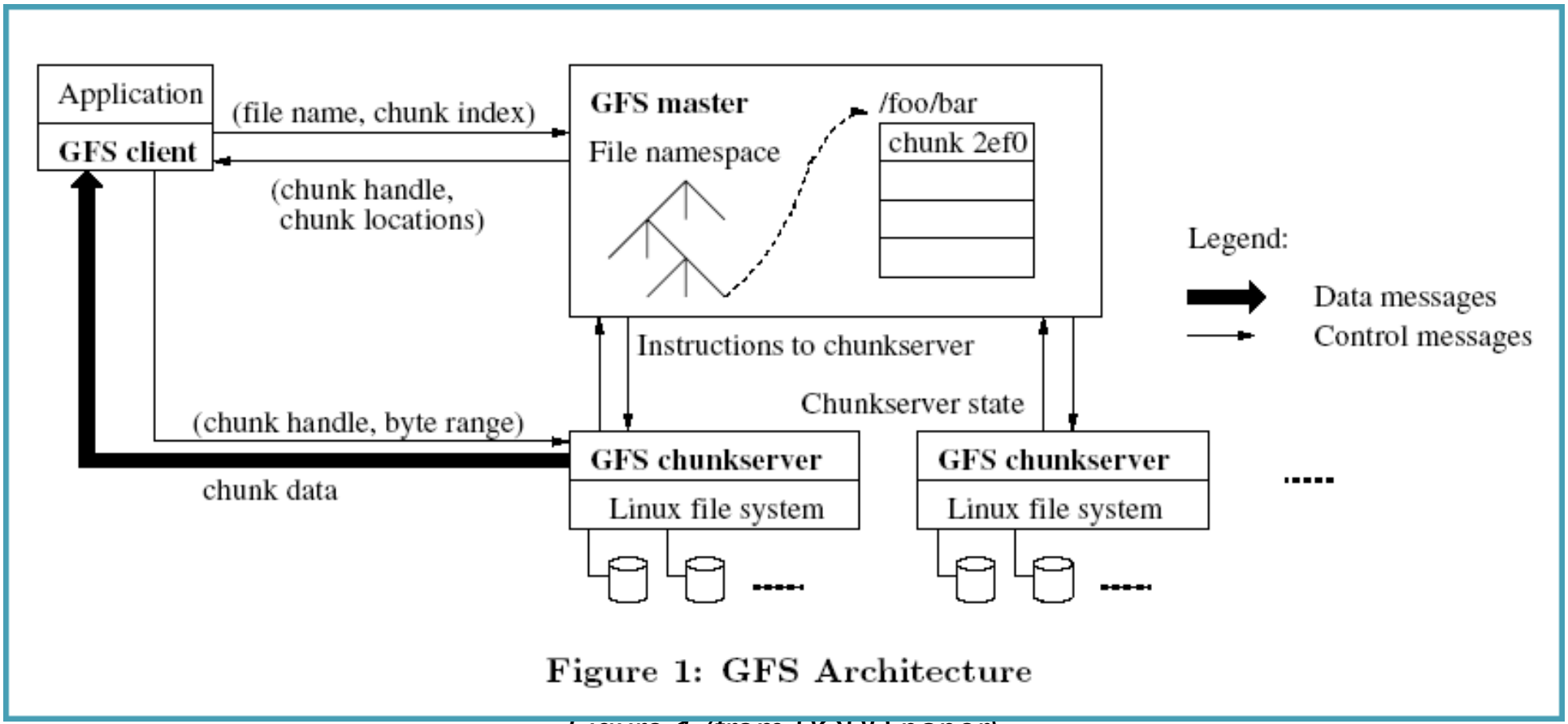

Figure 1 (from SOSP paper)

Key points:

- No POSIX interface
- Single Master for metadata
- Large chunk size

- Metadata, chunk locations, and operation log

- Data Flow and leases

Consistency

Consistent vs defined regions.

All replicas need not be exactly same.

Relaxed consistency model.

Master Operations

- Namespace Management
- Replica placement
- Chunk creation/rebalancing

Garbage Collection and stale replica deletion