BIGTABLE

A DISTRIBUTED STORAGE SYSTEM FOR STRUCTURED DATA

David Gunter

25 November 2013

Chang, Fay; Dean, Jeffrey; Ghemawat, Sanjay; Hsieh, Wilson C; Wallach, Deborah A; Burrows, Michael 'Mike'; Chandra, Tushar; Fikes, Andrew; Gruber, Robert E (2006), "Bigtable: A Distributed Storage System for Structured Data" (PDF), Research, Google.

EXECUTIVE SUMMARY

- A versatile distributed database/storage system that accommodates massive amounts of data (petabytes and beyond) through distribution amongst many servers.
- Aims for “wide applicability, scalability, high performance, and high availability”
- Dynamic, bare-bones data formats
  - Not a true relational database; it’s up to the user to configure data relations

IMPLEMENTATION

- Maps
  - Row key, column key, timestamp
    - Rows: read/writes are atomic
      - Items of the same “domain” are grouped together to speed up searching
      - Row ranges = *tablets* which are distributed across machines
    - Column keys: sets of *column families*
      - Families comprised of many columns, but tables have small number of families (families grouped bye same data type)
    - Timestamps: used for garbage collecting & versioning

IMPLEMENTATION (CONT.)

- Distribution
  - BigTable runs on machines shared with other processes
  - 3 Components: library, master server, tablet servers
    - Master responsible for assigning tablets to table servers
      - Rarely communicated with. Master servers = lightly loaded
      - Responsible for ensuring server is serving tablets and reassigning tablets that are not being served
    - Tablet servers handle r/w requests & manages tablet size
      - Perform the bulk of operations
      - No tablet <200MB in size

ANALYSIS

- Wildly versatile
  - The DB storage system is bare-bones, providing only what is necessary for organization. Giving users control of how data is received and transformed is very powerful
  - Being able to store literally any kind of data enables wide applicability to a multitude of industries
    - i.e. storing thousands of images or even entire webpages in cells
- Highly scalable, fault-tolerant
  - The master & tablet servers do an excellent job of both balancing and resolving errors
    - The tablet server's ability to reduce and remap locations is highly intelligent and helps reduce data loss
    - The master server and its redistribution upon error notification is also incredible in its data quality capabilities

ADVANTAGES & DISADVANTAGES

- **Advantages**
  - Highly flexible with data types and organization
  - Self-managing
    - Tablets automatically broken down once they reach a size threshold
  - Does not require dedicated resources
    - BigTable runs on computers that share processes with other applications
- **Disadvantages**
  - Reliant upon Google File System (GFS)
    - Many points of failure (hardware, maintenance, GFS quotas)
  - Some performance hiccups in random reads
    - Requiring “transfer of 64KB block for every 1000-byte read” slows down reads

HOW CAN I USE THIS?

- Google Earth
  - One table to hold all images for geo. segments (70TB+ of data!)
  - Rows organized so that “adjacent geographic segments are stored near each other”
    - Reduces search time!
- Personalized Search
  - Grouping user actions together
    - Decrease latency, r/w time
  - Highly replicated
    - Increased availability, latency reduced by replicating to servers near the user