Bigtable: A Distributed Storage System for Structured Data

Jackelyn Trivino

April 30,2014

Fay Chang, Jeffrey Dean, Sanjay Ghemawat, Wilson C. Hsieh, Deborah A. Wallach

Mike Burrows, Tushar Chandra, Andrew Fikes, Robert E. Gruber

Main Idea

- The author talks about how they wanted to designed and implemented a scalable data storage system that was high performance in order to handle the ever growing demand for their processing needs.
- The new data storage system needs to be flexible in terms of data type and data size because they handled and managed thousands of servers and petabytes of data.
- They need dynamic control over how to serve their data and the flexibility of how clients use and organize their data in their data system.

Implemented

- Data is stored in a persistent multidimensional sorted map that indexed by row key, timestamp, and column key
- The master server controlled and was responsible for all tablet servers that hold the row ranges of data. Data is maintained in lexicographic order by row key.
- Each table in the cluster is a SSTable file format to provide an ordered map from keys to values.

Analysis of Idea and Implementation

- Bigtable accomplishes its goal when it comes to their implementation of their distributed storage system. It can handle all kinds of data at impressive speeds because of its capability to have lookups be on one single disk.
- Entirely relies on Google products which is an advantage as it increases performance furthermore.
- Chubby is a great mechanism that shows traffic in servers, track down clusters and see any inactive server tables.

Comparisons in Implementation

Differences:

- Bigtable supports a schema defining model while MapReduces doesn't.
- MapReduce has a better use for small applications rather than bigger ones.

Similarities:

- Bigtable is very similar to the parallel DBMS and has MapReduce features.
- Has the ability to minimize data transfers through parallel query optimizer due to indexes and query planning and reliability.

Comparison with both Papers

Advantages:

- They have performance advantages because they index their data using B-trees to speed-up access as well as their query planning.
- They also increase read performances by using 2 levels of caching for the tablet servers.
- Bigtable requires much less code to perform the same amount of tasks that it take for MapReduce.

Disadvantages:

- Response time to errors are slower than MapReduce
- Because Bigtable doesn't has no proper SQL standard, it increases failures which would be much too complicated to create different protocols and complications in the long term.