Bigtable: A Distributed Storage System for Structured Data

TOCS'08

201924169 Ju-Won Park

May 27, 2019

Problem statement

- Google has many applications which need a system that allows them to store/retrieve structured data. The main challenge is to have a system which can provide that service reliably and efficiently.
- to build this system so that it is suitable to all applications.

Solution approach

- The authors in this paper have described Bigtable, a distributed system for storing structured data at Google.
- It is a sparse and persistent multi-dimensional sorted map.
- The map is indexed by a row key, column key, and a timestamp; each value in the map is an uninterpreted array of bytes.

Strong points

- For me a big plus of Bigtable was to introduce an API which provides functions for creating and deleting tables and column families.
- Its also good to see that it provides functions for changing cluster, table, and column family metadata, such as access control rights.
- It is a good feature that Bigtable schema parameters let clients dynamically control whether to serve data out of memory or from disk.

Weak points / Limitations

- it does not support transactions over multiple rows.
- the users cannot simply use conventional SQL-like commands! It makes it difficult for users to adapt to using it.

Questions

- What if they could provide support for transactions over multiple rows? Does it need much modifications?

New ideas / Comments

- I am thinking maybe in the future there could be a wrapper or API bring convenience for the users to use conventional SQL-like commands.
- There is an increasing demand on keeping large data and it seems that NoSql systems are prefered to satisfy this demand. BigTable is a very good example of a NoSql system to deal with keeping large amounts of data and responding large number of requests efficiently.