

# Bigtable: A Distributed Storage System for Structured Data

Chris Piccirillo

November 25, 2013

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

Mike Burrows, Tushar Chandra, Andrew Fikes, Robert E. Gruber  
{fay,jeff,sanjay,wilsonh,kerr,m3b,tushar,fikes,gruber}@google.com

Google, Inc

# Main Idea of Paper

- The purpose of Bigtable is to provide an adjustable working solution for Google Earth, Google Finance web indexing and other Google services.
- BigTable is a proprietary data storage system built on Google File System, Chubby Lock Service, SSTable and a few other Google technologies
- Deploys a distributed storage system for managing structured data at Google.
- Allows for storing data across various servers with the ability to access the data from any location.
- Designed to reliably scale data to sizes as large as petabytes on thousands of machines.
- To be used by a wide variety of Google products (60+)

# Main Idea Implementation

## Three major components

- A library that is linked into every client
- One master server
- Many tablet servers

## More implementation features

- Each and every tablet server is responsible for managing a different set of tables (10-1000 tablets per server typically)
- The tablet servers can be added or removed from a cluster dynamically
- Clients communicate directly with tablet servers for reads and writes
- A Bigtable cluster holds a number of tables, each table consists of a set of tablets, and each tablet contains all data that is associated with the row range

# Advantages and Disadvantages

## Advantages

- Clients are able to control the locality of their data by making their schema choices carefully
- Simple Data model
- Clients have dynamic control data layout and data format.
- Allows clients to reason about the locality properties of the data represented in the underlying storage
- The schema parameters in Bigtable let clients dynamically control whether to serve data out of memory or from disk.
- The row range for a table is dynamically partitioned. This makes reads of short row ranges incredibly efficient and only generally require communication with a small number of machines.

## Disadvantages

- Bigtable does not support a full relational data model
- Data duplication can occur
- A secondary index is not a supported feature
- Data loss is a possibility

# Main Idea and Implementation Analysis

- Overall good idea, provides a reliable way to manage large amounts of data.
- Makes the life of those utilizing this much easier to use than other alternatives (All data is accessible from one location)
- Minimal flaws, as opposed to the heavy weight of the benefits
- Will be usable in the future with old versions of certain data as a resource
- One commit log per tablet as opposed to one log per tablet server. (Good Change)
- Reducing the # of disk accesses could be beneficial
- Implementation is broken down into three simple parts. Makes the system easy to understand and makes the actual implementation process easy.

# Real-World use cases

## Google Earth

- Provides users with a collection of services with access to high-res satellite imagery of surfaces all over the earth
- Using two tables (about 70TB for imagery and 500GB for cache) (increasing)

## Google Analytics

- Provides users with aggregate statistics, for example the number of unique visitors per day and page views by URL
- Using two tables (Raw User Clicks: about 200TB) (Summary: about 20TB)

## More Examples

Google Finance, Google Web-Indexing, Orkut