Year	System/ Paper	Scale to 1000s	Primary Index	Secondary Indexes	Transactions	Joins/ Analytics	Integrity Constraints	Views	Language/ Algebra	Data model	my label
1971	RDBMS	O	✔	✔	✔	✔	✔	✔	✔	tables	SQL-like
2003	memcached	✔	✔	O	O	O	O	O	O	key-val	lookup
2004	MapReduce	✔	O	O	O	✔	O	O	O	key-val	MR
2005	CouchDB	✔	✔	✔	record	MR	O	✔	O	document	filter/MR
2006	BigTable (Hbase)	✔	✔	✔	record	compat. w/MR	/	O	O	ext. record	filter/MR
2007	MongoDB	✔	✔	✔	EC, record	O	O	O	O	document	filter
2007	Dynamo	✔	✔	O	O	O	O	O	O	key-val	lookup
2008	Pig	✔	O	O	O	✔	/	O	✔	tables	RA-like
2008	HIVE	✔	O	O	O	✔	✔	O	✔	tables	SQL-like
2008	Cassandra	✔	✔	✔	EC, record	O	✔	✔	O	key-val	filter
2009	Voldemort	✔	✔	O	EC, record	O	O	O	O	key-val	lookup
2009	Riak	✔	✔	✔	EC, record	MR	O			key-val	filter
2010	Dremel	✔	O	O	O	/	✔	O	✔	tables	SQL-like
2011	Megastore	✔	✔	✔	entity groups	O	/	O	/	tables	filter
2011	Tenzing	✔	O	O	O	O	✔	✔	✔	tables	SQL-like
2011	Spark/Shark	✔	O	O	O	✔	✔	O	✔	tables	SQL-like
2012	Spanner	✔	✔	✔	✔	?	✔	✔	✔	tables	SQL-like
2012	Accumulo	✔	✔	✔	record	compat. w/MR	/	O	O	ext. record	filter
2013	Impala	✔	O	O	O	✔	✔	O	✔	tables	SQL-like

Google BigTable

- OSDI paper in 2006
  - Some overlap with the authors of the MapReduce paper
- Complementary to MapReduce
  - Recall: What is MapReduce *not* designed for?

Data model

- "a sparse, distributed, persistent multi-dimensional sorted map"

$$(\text{row:string}, \text{column:string}, \text{time:int64}) \rightarrow \text{string}$$

Rows

- Data is sorted lexicographically by row key
- Row key range broken into *tablets*
  - Recall: What was Teradata's model of distribution?
- A tablet is the unit of distribution and load balancing

Column families

- Column names of the form *family:qualifier*
- "family" is the basic unit of
  - access control
  - memory accounting
  - disk accounting
- Typically all columns in a family the same type

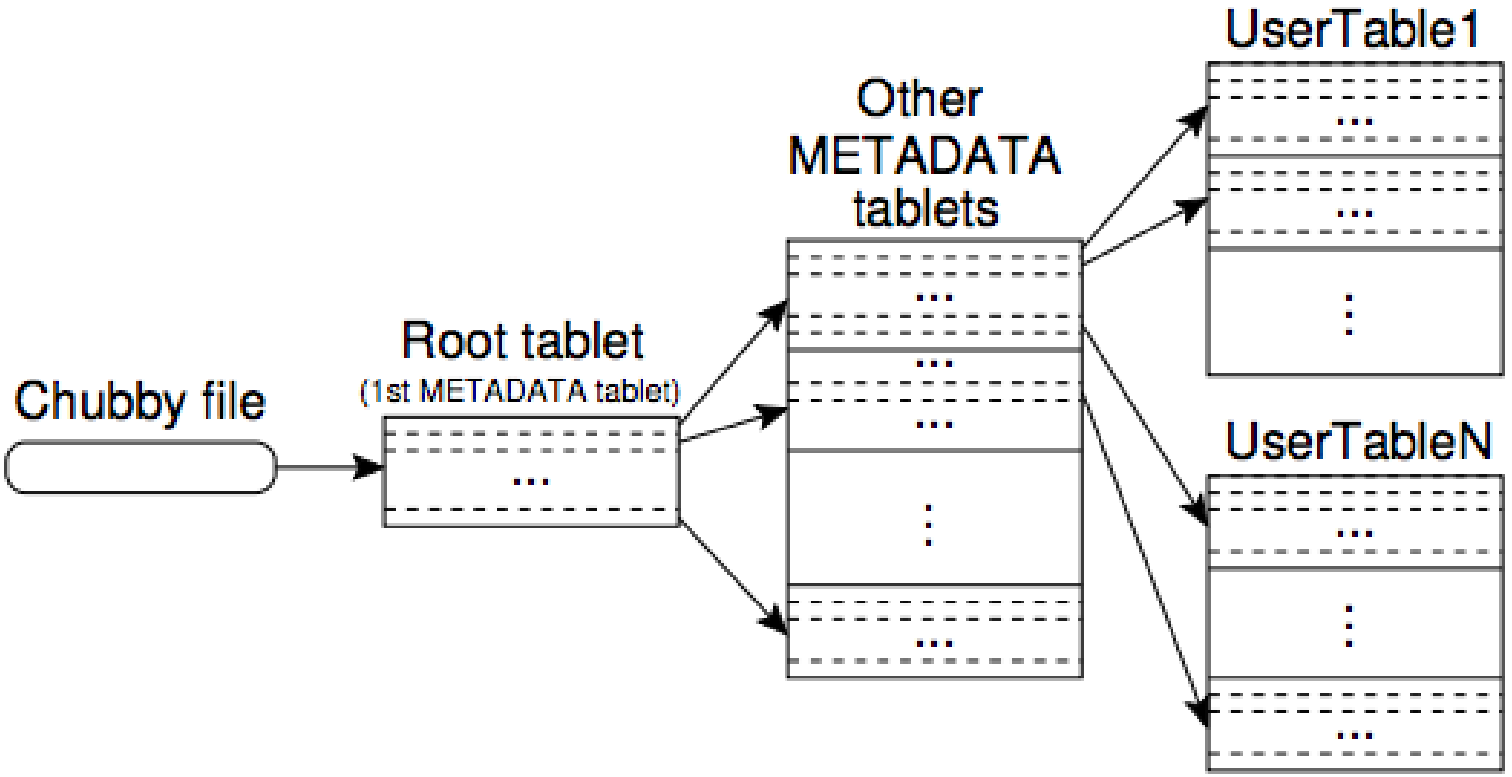
Timestamps

- Each cell can be *versioned*
- Each new version increments the timestamp
- Policies:
  - "keep only latest *n* versions"
  - "keep only versions since time *t*"

Tablet management

- Master assigns tablets to tablet servers
- Tablet server manages reads and writes from its tablets
- Clients communicate directly with tablet server
- Tablet server splits tablets that have grown too large.

Tablet location metadata

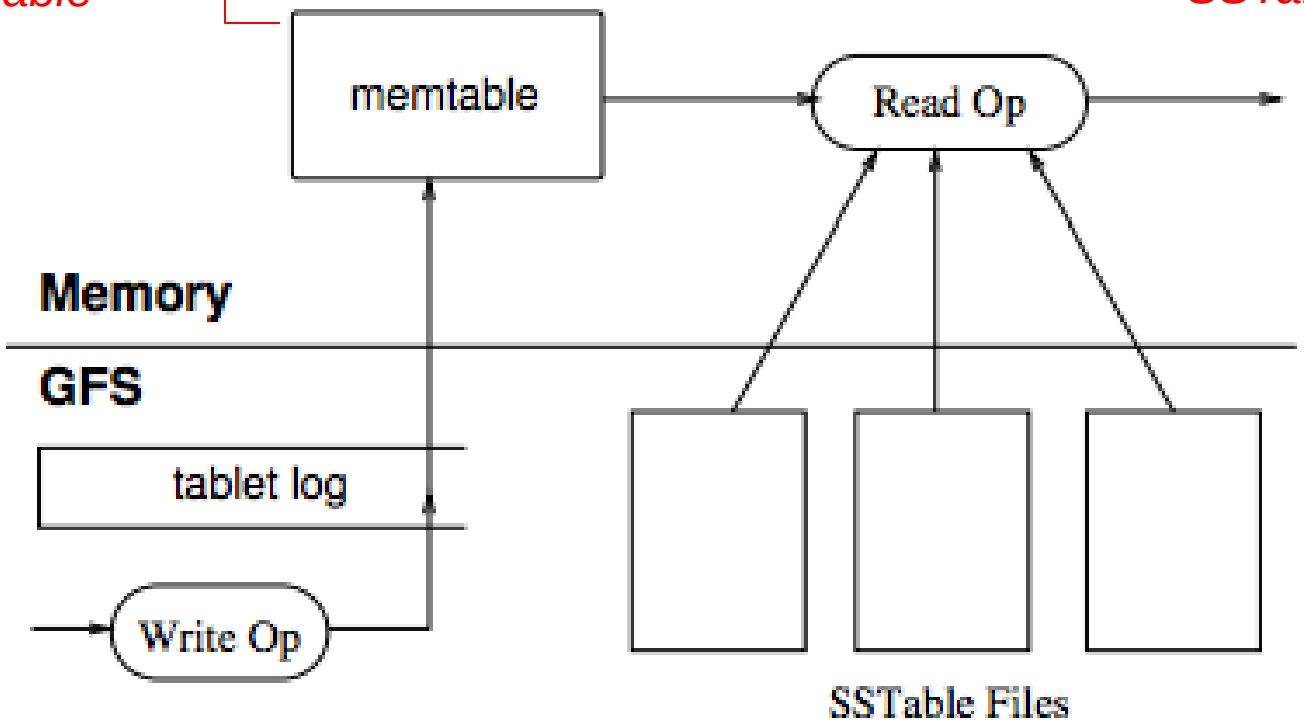


Chubby: distributed lock service

Write processing

recent sequence of updates in memtable

minor compaction: write memtable buffer to a new SSTable



major compaction: rewrite all SSTables into one SSTable; clean deletes

Other tricks

- Compression
  - specified by clients
- Bloom filters
  - fast set membership test for (row, column) pair
  - reduces disk accesses during reads
- Locality Groups
  - Groups of column families frequently accessed together
- Immutability
  - SSTables (disk chunks) are immutable
  - Only the memtable needs to support concurrent updates (via copy-on-write)

HBase

- Implementation of Google BigTable
- Compatible with Hadoop
  - TableInputFormat allows reading of BigTable data in the map phase
  - One mapper per tablet
  - Aside: Speculative Execution?

```
Table       (HBase table)
    Region       (Regions for the table)
         Store          (Store per ColumnFamily for each Region for the table)
              MemStore          (MemStore for each Store for each Region for the table)
              StoreFile         (StoreFiles for each Store for each Region for the table)
                   Block             (Blocks within a StoreFile within a Store for each Region for the table)
```