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)
```

[Baker, Bond, Corbett, et al. 2011]

Megastore

- Argues that loose consistency models complicate application programming
- Synchronous replication
- Full transactions within a partition

Spanner

Even though many projects happily use Bigtable [9], we have also consistently received complaints from users that Bigtable can be difficult to use for some kinds of applications: **those that have complex, evolving schemas**, **or** **those that want strong consistency in the presence of wide-area replication**.

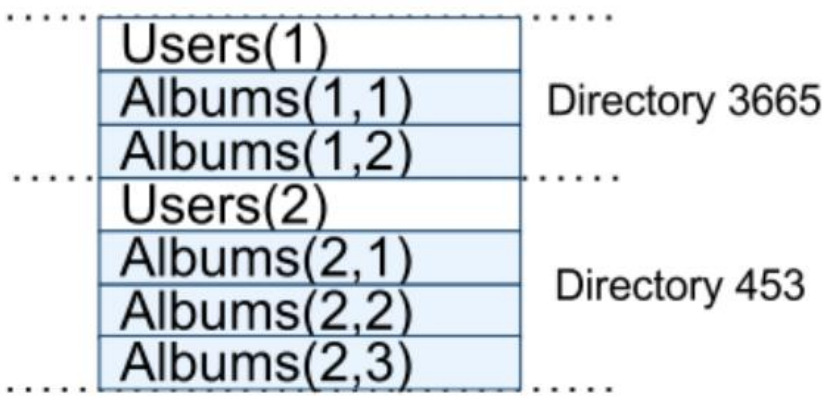
"We believe it is better to have application programmers deal with performance problems due to overuse of transactions as bottlenecks arise, rather than always coding around the lack of transactions."

"Although Spanner is scalable in the number of nodes, the node-local data structures have relatively poor performance on complex SQL queries, because they were designed for simple key-value accesses. Algorithms and data structures from DB literature could improve singlenode performance a great deal."

Spanner Data Model

- Directory: set of contiguous keys with a shared prefix

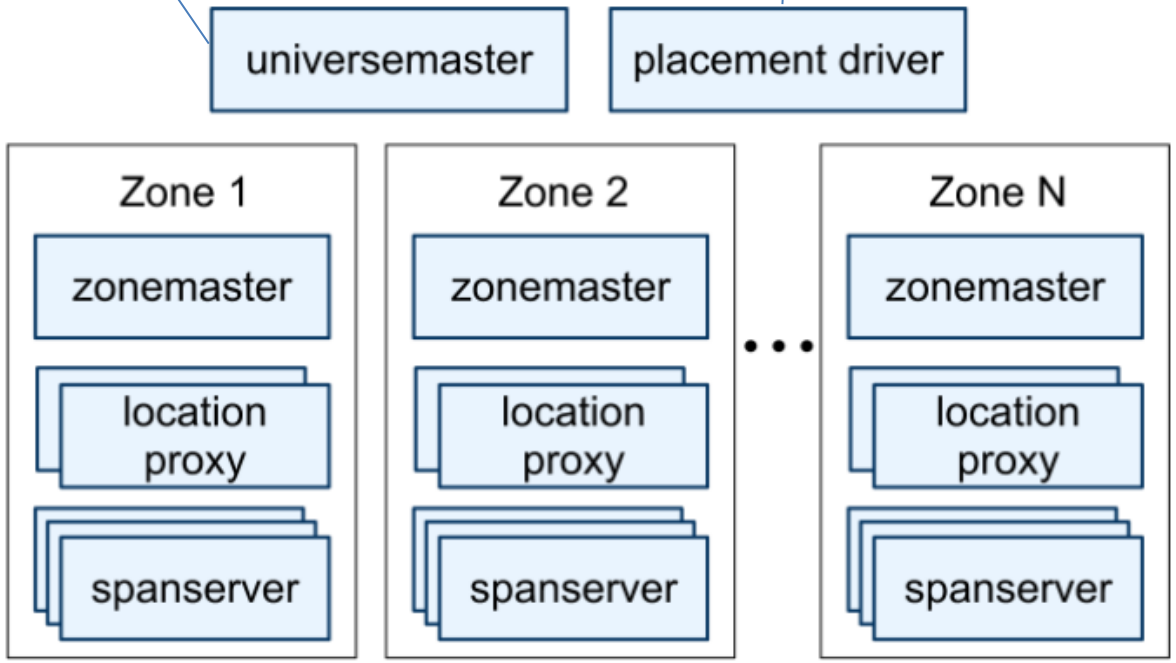
```
CREATE TABLE Users {
uid INT64 NOT NULL,
email STRING
} PRIMARY KEY (uid), DIRECTORY;

CREATE TABLE Albums {
uid INT64 NOT NULL,
aid INT64 NOT NULL,
name STRING
} PRIMARY KEY (uid, aid),
INTERLEAVE IN PARENT Users
```



Tables are interleaved to create a hierarchy

mostly status info about zones

Moves directories for load balancing every few minutes



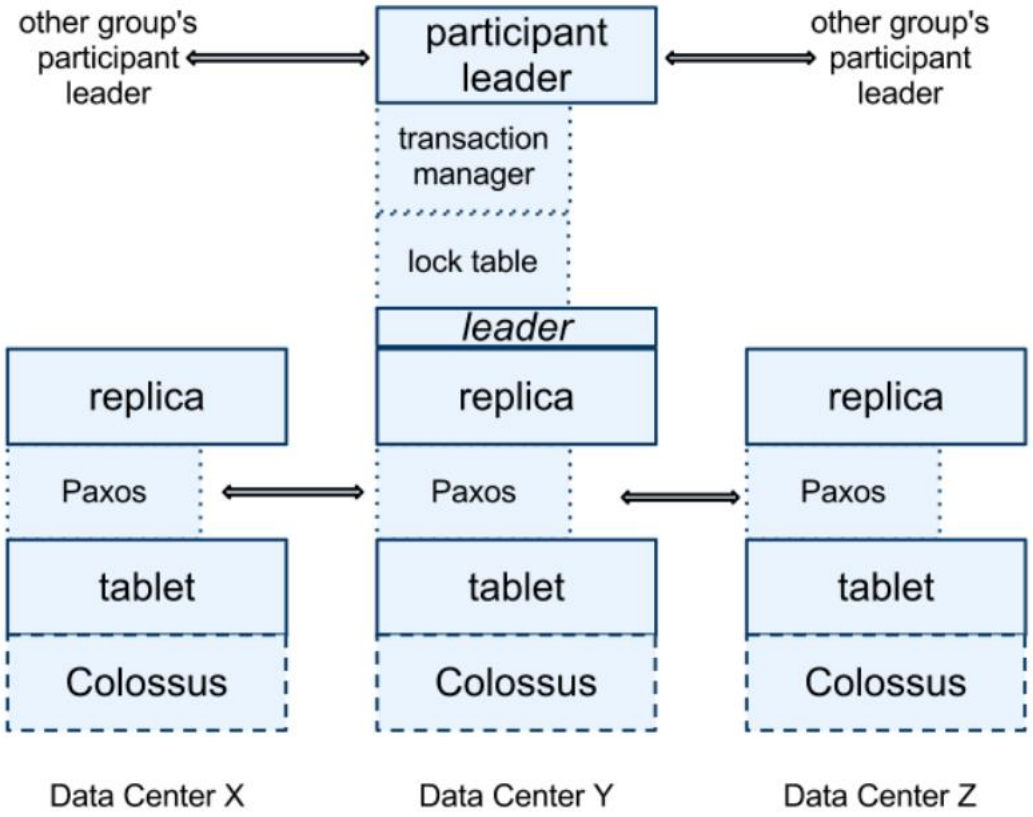
Zone ~ BigTable deployment

Assigns data to spanservers

Routes requests from clients

Serves data

Spanserver



2-phase commit across groups (when needed)

Paxos across replicas

Colossus successor to GFS

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	nosql
2004	MapReduce	✔	O	O	O	✔	O	O	O	key-val	batch
2005	CouchDB	✔	✔	✔	record	MR	O	✔	O	document	nosql
2006	BigTable (Hbase)	✔	✔	✔	record	compat. w/MR	/	O	O	ext. record	nosql
2007	MongoDB	✔	✔	✔	EC, record	O	O	O	O	document	nosql
2007	Dynamo	✔	✔	O	O	O	O	O	O	ext. record	nosql
2008	Pig	✔	O	O	O	✔	/	O	✔	tables	sql-like
2008	HIVE	✔	O	O	O	✔	✔	O	✔	tables	sql-like
2008	Cassandra	✔	✔	✔	EC, record	O	✔	✔	O	key-val	nosql
2009	Voldemort	✔	✔	O	EC, record	O	O	O	O	key-val	nosql
2009	Riak	✔	✔	✔	EC, record	MR	O			key-val	nosql
2010	Dremel	✔	O	O	O	/	✔	O	✔	tables	sql-like
2011	Megastore	✔	✔	✔	entity groups	O	/	O	/	tables	nosql
2011	Tenzing	✔	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	nosql
2013	Impala	✔	O	O	O	✔	✔	O	✔	tables	sql-like

Google's Systems

Google Big Data Systems