CLOUD COMPUTING CONCEPTS

with Indranil Gupta (Indy)

KEY-VALUE STORES NoSQL

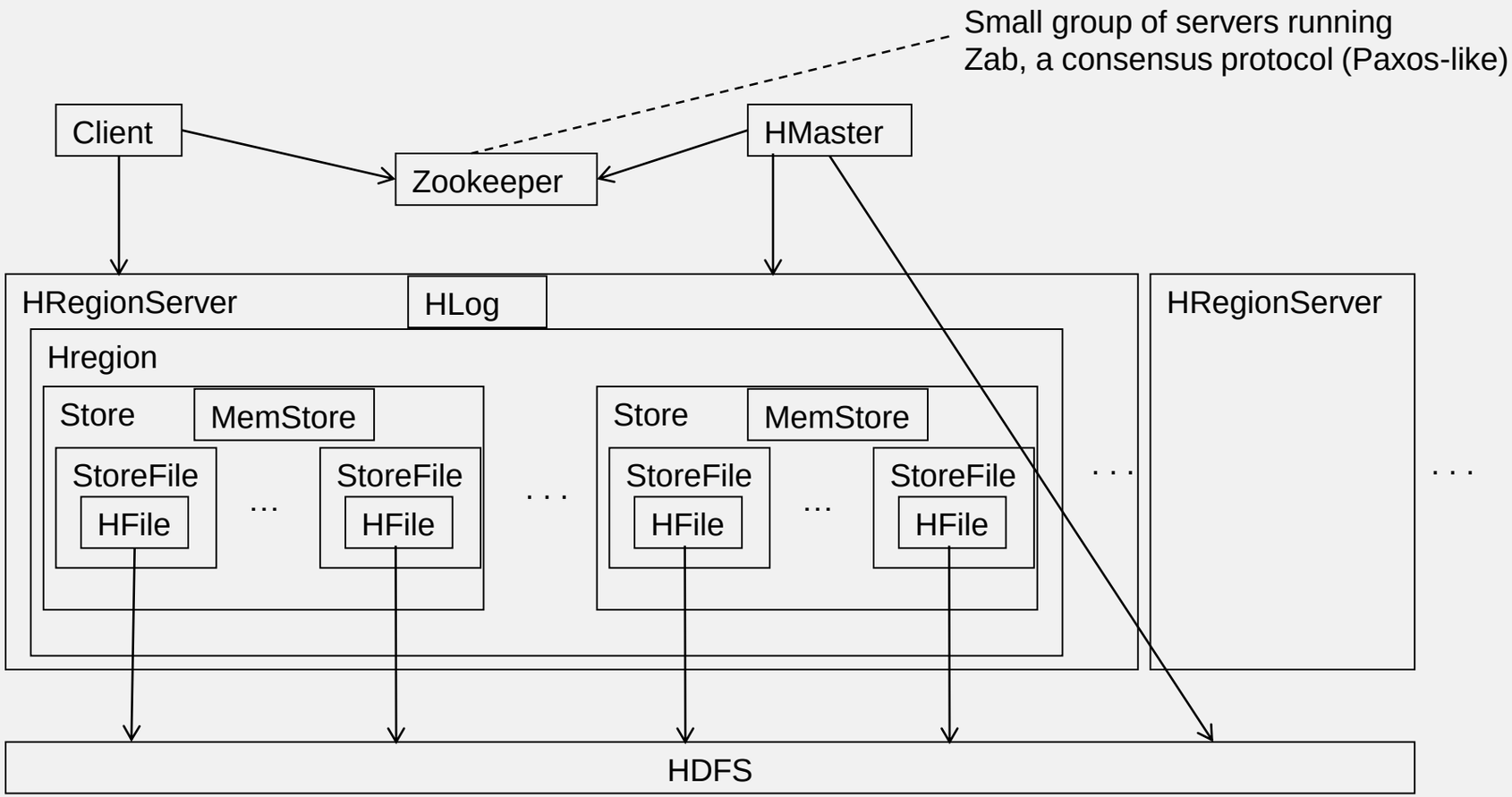
Lecture E

HBASE

HBase

- Google's BigTable was first "blob-based" storage system
- Yahoo! Open-sourced it → HBase
- Major Apache project today
- Facebook uses HBase internally
- API functions
  - Get/Put(row)
  - Scan(row range, filter) – range queries
  - MultiPut
- Unlike Cassandra, HBase prefers consistency (over availability)

HBase Architecture

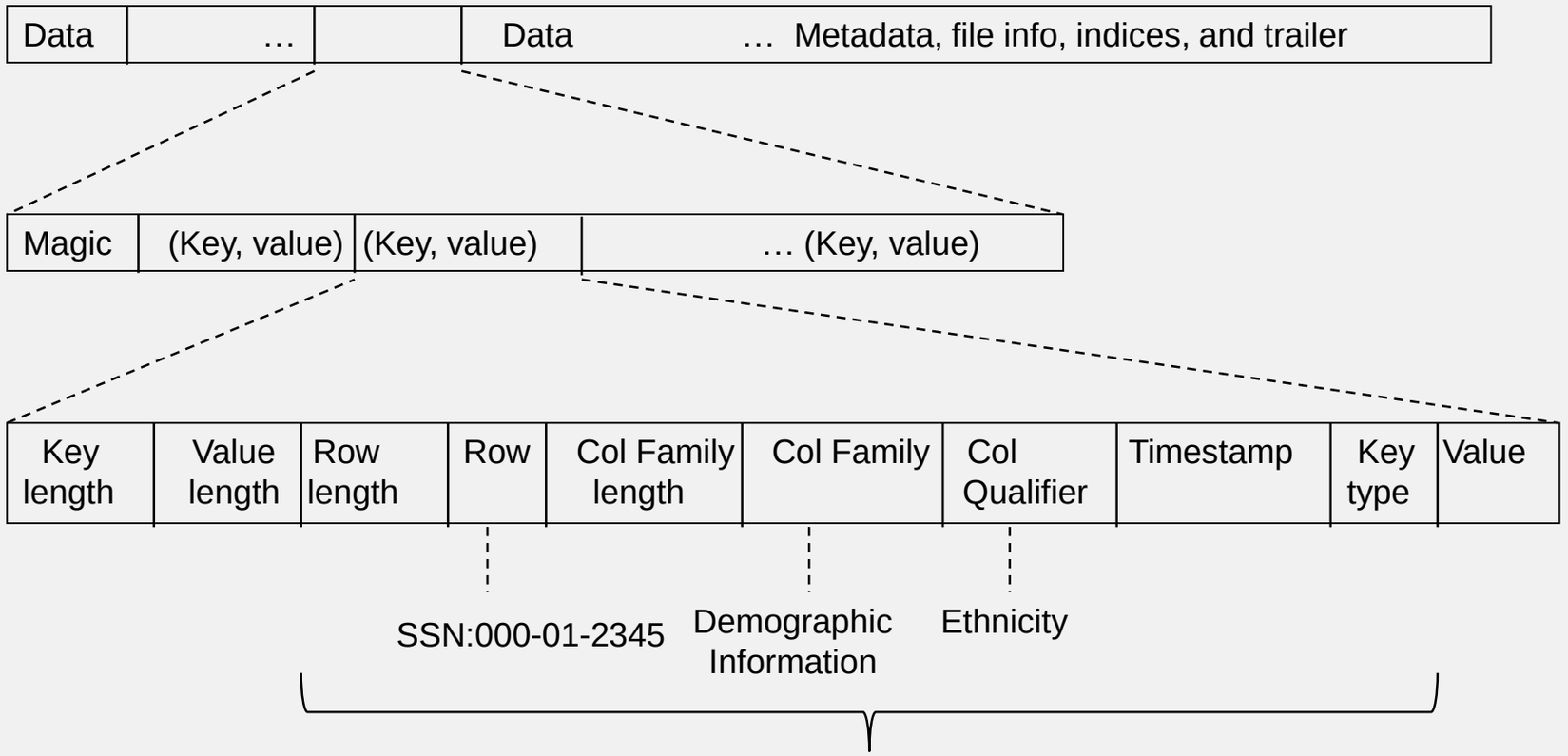


HBase Storage hierarchy

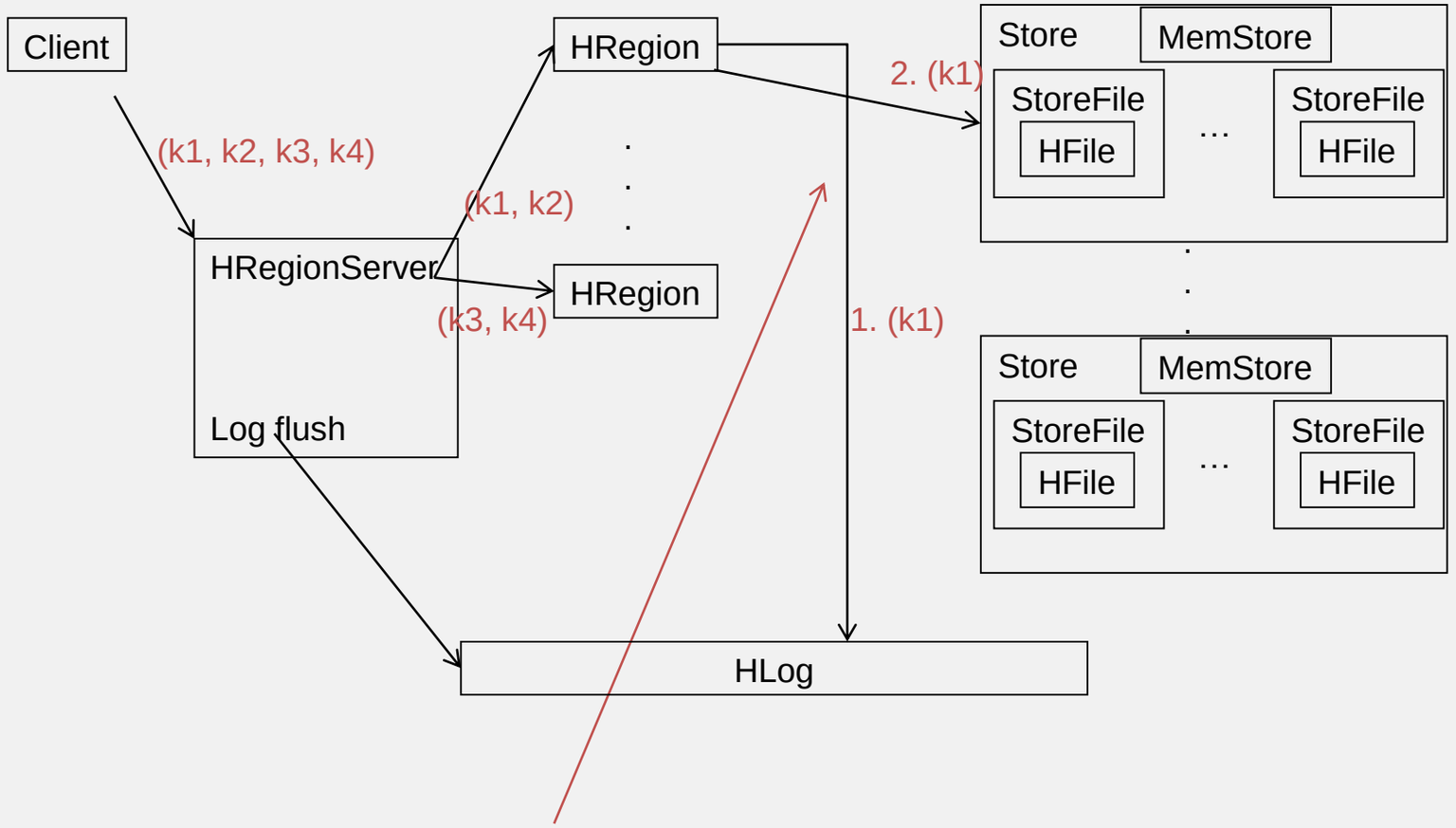
- HBase Table
  - Split it into multiple regions: replicated across servers
    - ColumnFamily = subset of columns with similar query patterns
    - One Store per combination of ColumnFamily + region
      - Memstore for each store: in-memory updates to store; flushed to disk when full
        - StoreFiles for each store for each region: where the data lives
          - HFile

- HFile
  - SSTable from Google's BigTable

HFile


Data
…
Data
… Metadata, file info, indices, and trailer
Magic
(Key, value)
(Key, value)
… (Key, value)
Key length
Value length
Row length
Row
Col Family length
Col Family
Col Qualifier
Timestamp
Key type
Value
SSN:000-01-2345
Demographic Information
Ethnicity
HBase Key

Strong Consistency: HBase Write-Ahead Log



Write to HLog before writing to MemStore
Helps recover from failure by replaying Hlog.

Log Replay

- After recovery from failure, or upon bootup (HRegionServer/HMaster)
  - Replay any stale logs (use timestamps to find out where the database is w.r.t. the logs)
  - Replay: add edits to the MemStore

Cross-Datacenter Replication

- Single “Master” cluster
- Other “Slave” clusters replicate the same tables
- Master cluster synchronously sends HLogs over to slave clusters
- Coordination among clusters is via Zookeeper
- Zookeeper can be used like a file system to store control information

1. */hbase/replication/state*
2. */hbase/replication/peers/<peer cluster number>*
3. */hbase/replication/rs/<hlog>*

Summary

- Traditional databases (RDBMSs) work with strong consistency and offer ACID
- Modern workloads don’t need such strong guarantees but do need fast response times (availability)
- Unfortunately, CAP theorem
- Key-value/NoSQL systems offer BASE
  - Eventual consistency, and a variety of other consistency models striving towards strong consistency
- We discussed design of
  - Cassandra
  - HBase