Week 5: HBase

Big Data Exercise Session

Plan for today

1. Recap

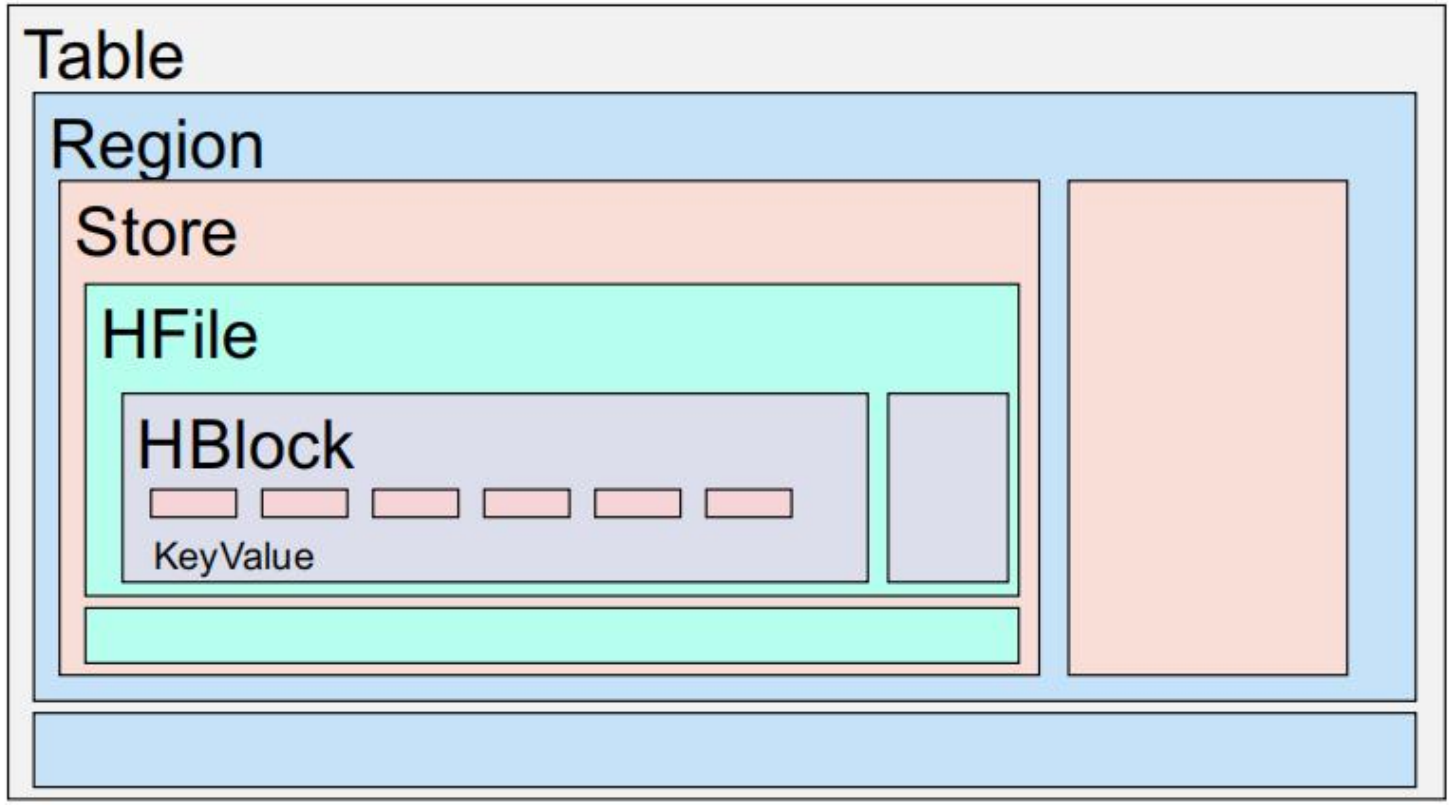
2. Exercise05 and Quiz05

Design principles

- Joins are expensive
- Store together what is accessed together -> batch processing
- Random read and write

Logical model: a big sparse table stored as key-value pairs.
Key = row ID + column family + column qualifier + version

Physical architecture:



Logical model

Row ID	A	B	1	2	I
000					
002					
0A1					
1E0					
22A					
4A2					

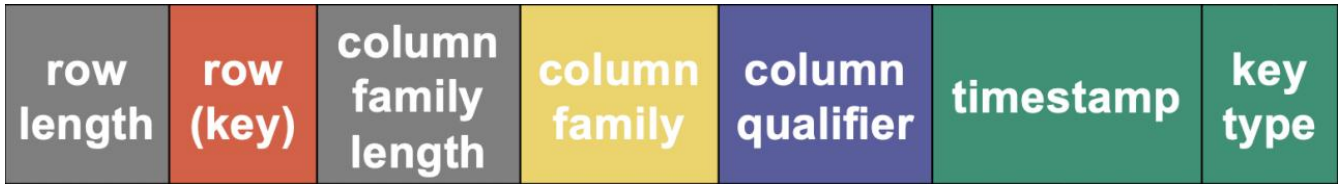
- Row IDs are sortable.
- Column families must be pre-defined; columns can be arbitrarily added.

Physical architecture

- HMaster + RegionServer: processes on nodes
- Convention: HMaster on namenode; RegionServer on datanode
- HMaser: assign regions to RegionServers; reassign regions when necessary
- Regionserver: read&write of regions; split too big regions

- Regions: a range of row IDs
- Store = region ∩ column family
- Store ⊃ HFlie ⊃ HBlock (64KB by default)
- HBlock: ordered key-values
- KeyValue pair:

- Key:



How do we read and write?

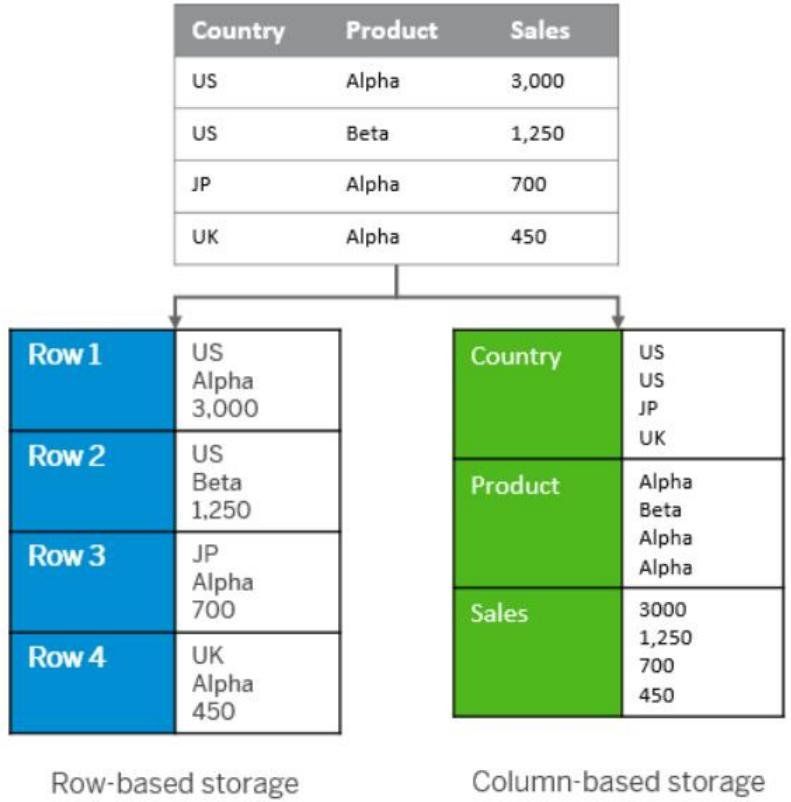
- Memstore (one per store): in memory
- Write: write cells to Memstore (sorted)
- Memstore reaches maximum size -> flush into HFlie
- Too many HFiles -> compaction (log-structured tree)
- Write ahead log (one per RegionServer): a file stored in HDFS

- Read: from HFlies and Memstores

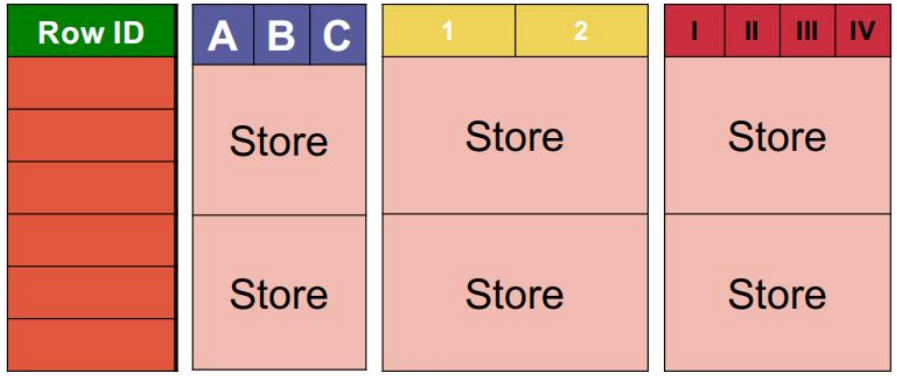
Other designs

- A meta table for lookup (also a HBase table)
- Caching frequently asscessed cells
- Bloom filters (one per HFile): tell with certainty that a key does **not** belong to an HFile
- Short-circuiting: flush HFile to the local disk to the machine running the RegionServer -> direct read from local disk

- Row store vs column store



- Wide column store



Bloom filters

- Build a bloom filter for a HFlie:
  an array *a* with length n, *a[0]= ... =a[n-1]=0* at the beginning
  hash functions $h_1, ..., h_k$: *{keys} -> {0,1,...,n-1}*
  for key in HFlie:
    for *i* in *1, ..., k*
      $a[h_i(key)]=1$

- Lookup a key
  If $a[h_1(key)]=...=a[h_k(key)]=1$, the bloom filter considers the key in the HFlie.