

Hbase

STARTSEV D.S., BIT, MIPT 2022,

Introduction

- NOSQL
- Distributed
- Wide-Column family
- Key-Value multi-version storage
- HDFS based

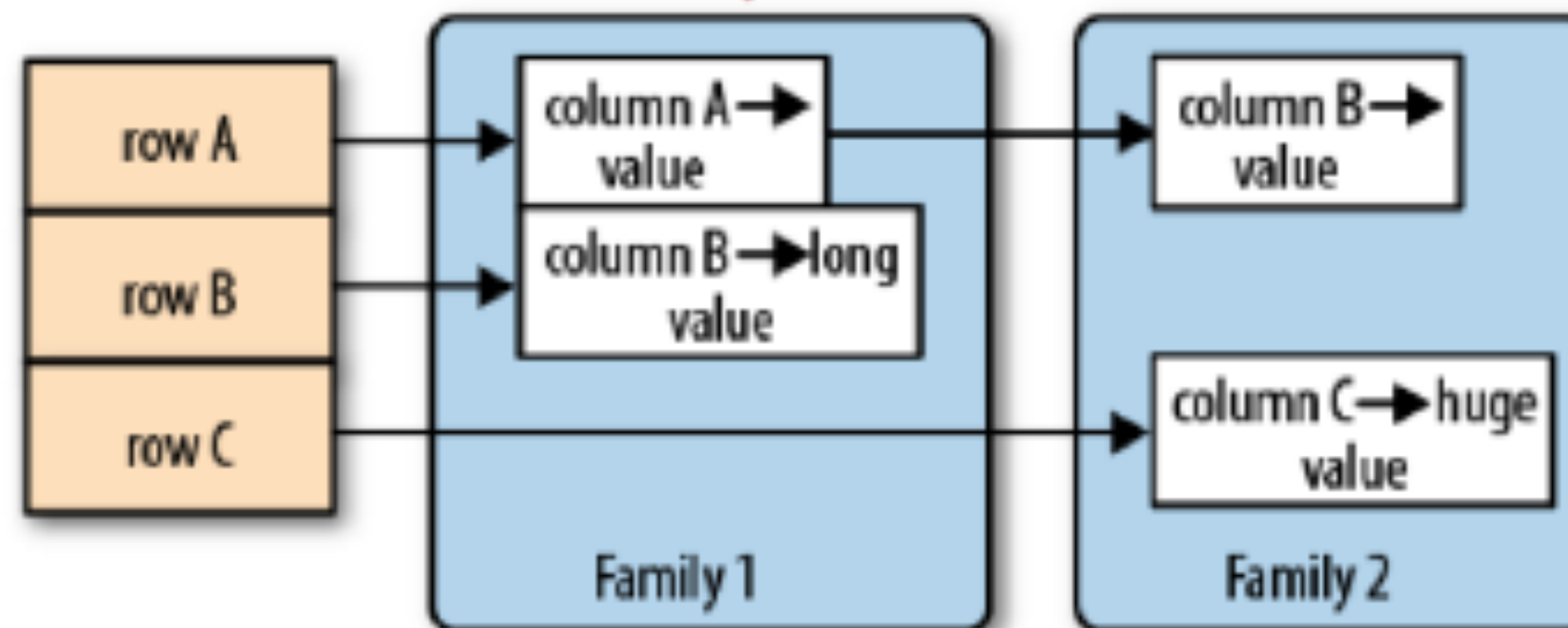


Data model

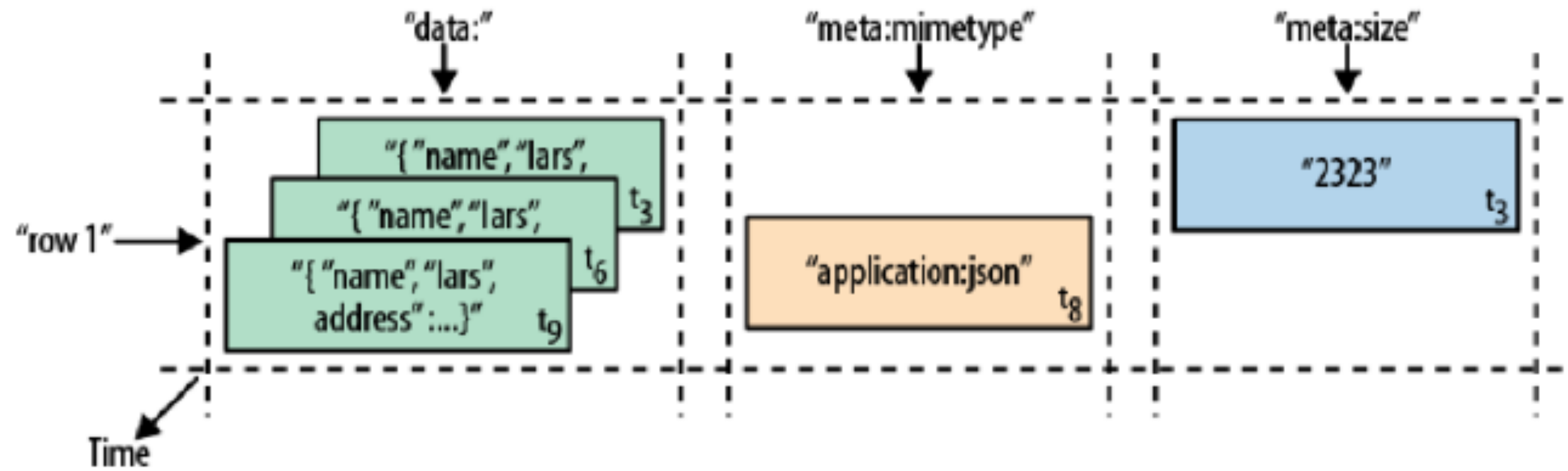
- $\langle \text{table, RowKey, Column Family, Column, timestamp} \rangle \rightarrow \text{Value}$
- TTL for each CF
- Timestamps for each attribute
- Sorted by RowKey
- Free NULL

Table -> Wide Column

	column A (int)	column B (varchar)	column C (boolean)	column D (date)
row A				
row B				
row C			NULL?	
row D				



Example

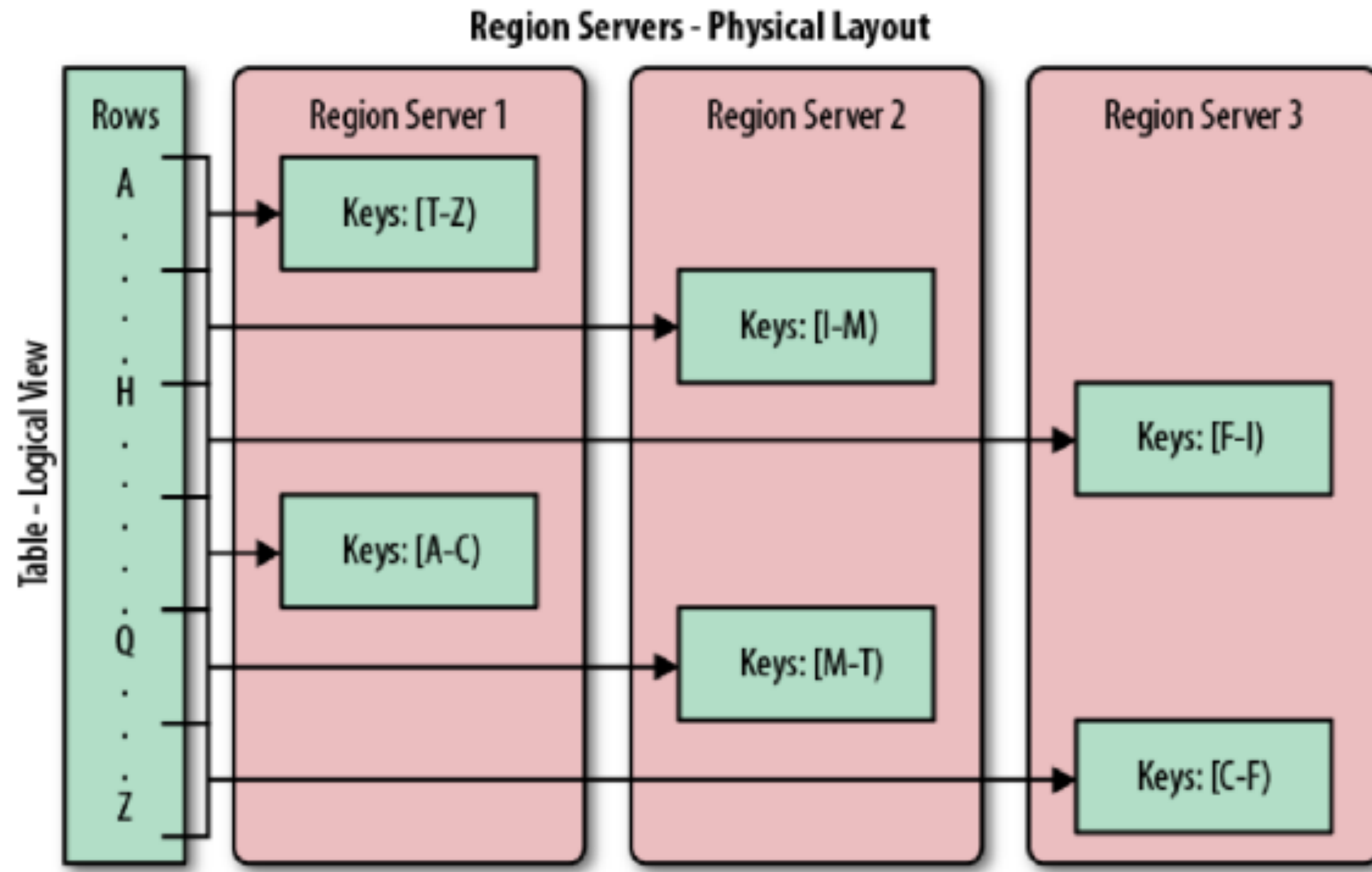


Wide Column -> Table

Row Key	Time Stamp	Column "data:"	Column "meta:" "mimetype" "size"		Column "counters:" "updates"
"row1"	t ₃	"{ "name": "lars", "address": ...}"		"2323"	"1"
	t ₆	"{ "name": "lars", "address": ...}"			"2"
	t ₈		"application/json"		
	t ₉	"{ "name": "lars", "address": ...}"			"3"

Architecture

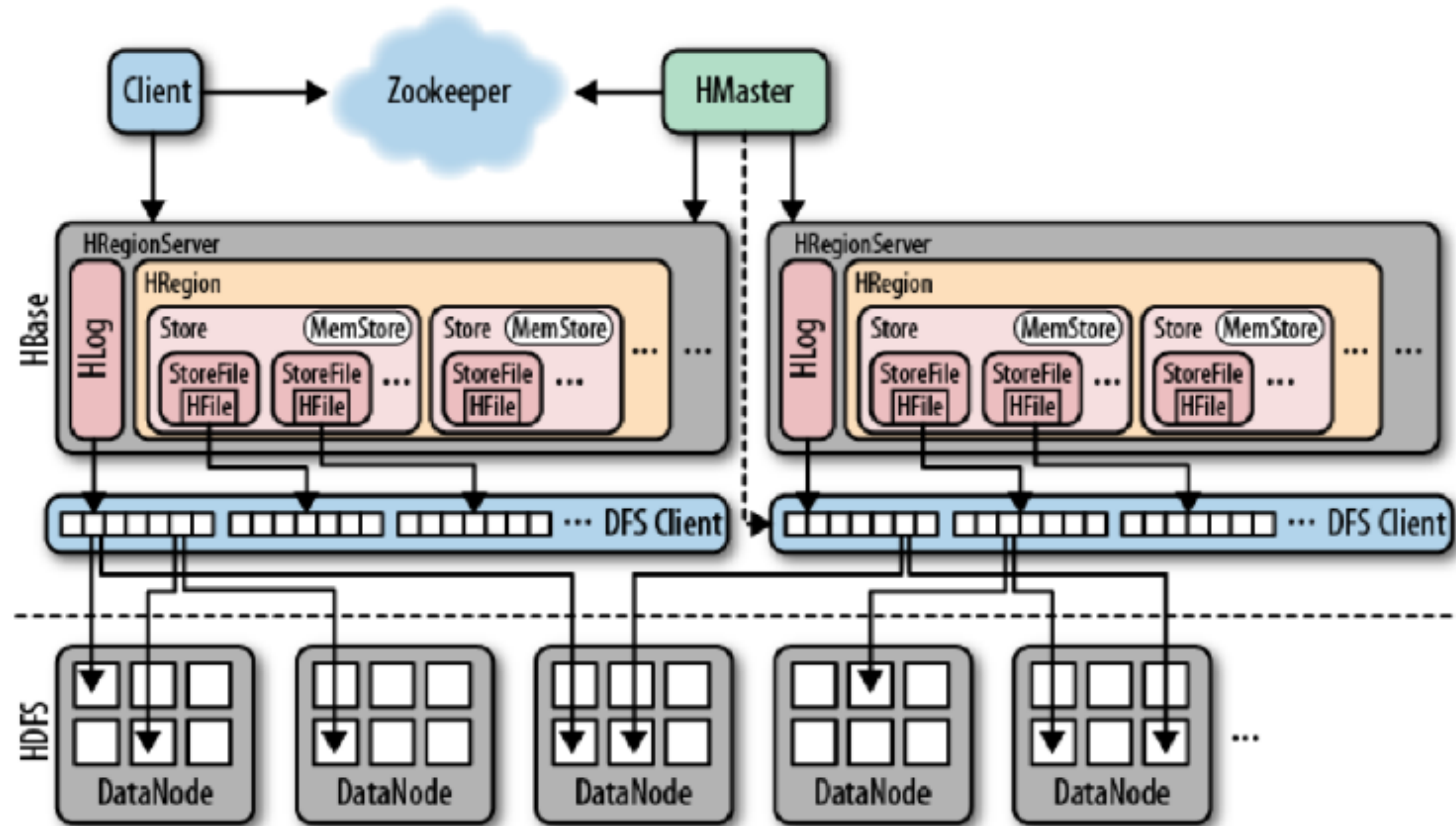
Regions



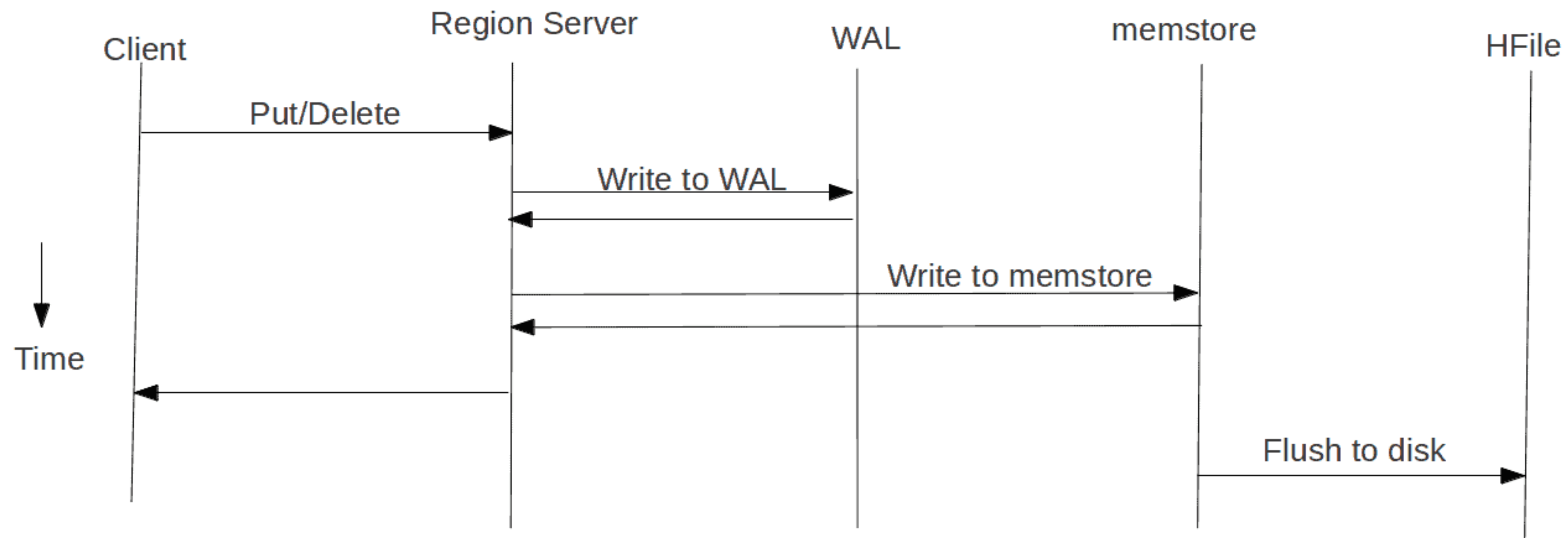
Regions

- Data storage
- MemStore
- BlockCache
- Write Ahead Log

Zookeeper, ♂ ♂ Master && Slaves ♂ ♂



Writing

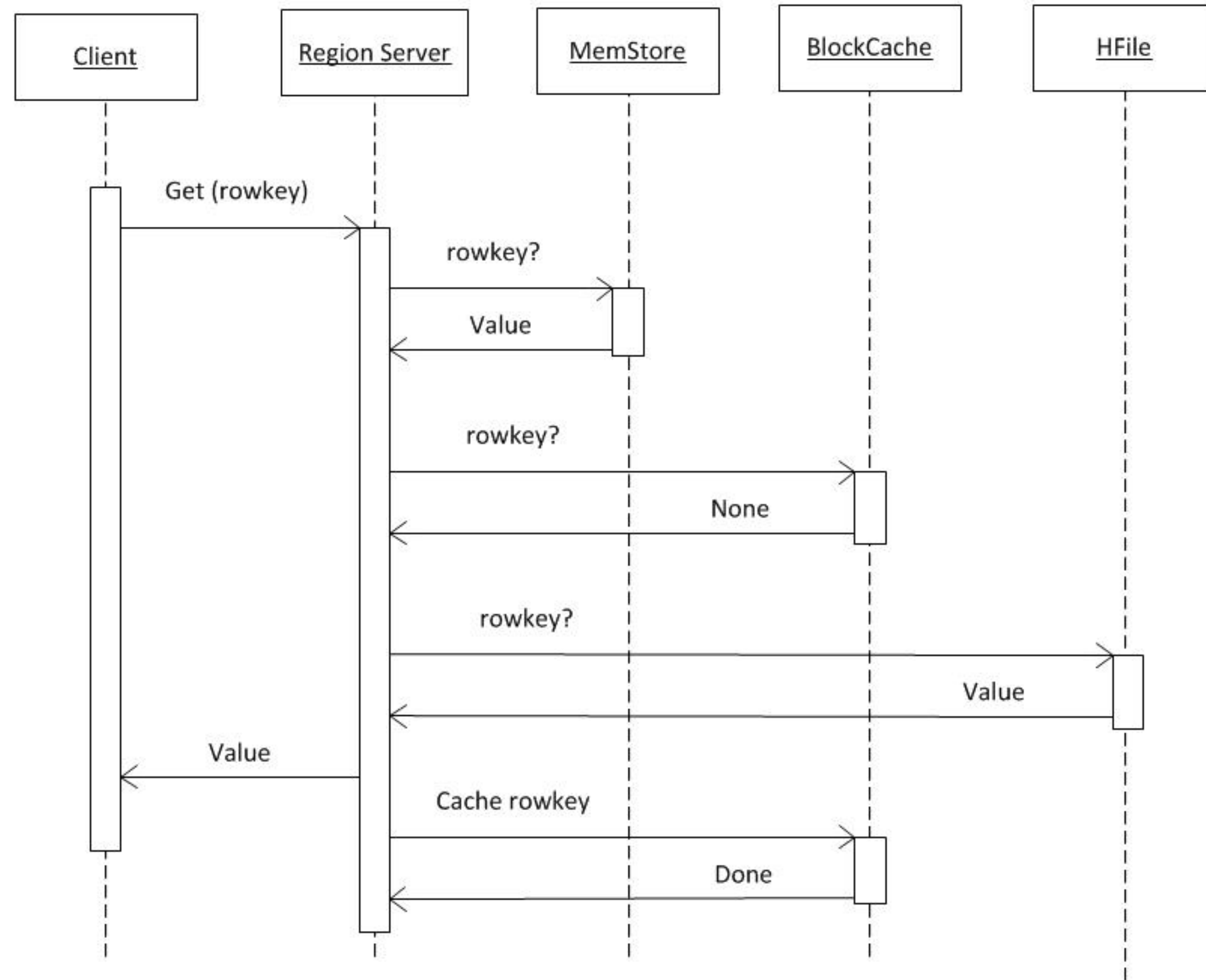


HBase Write Path

Compactions

- HFile merging
- Minor
- Major

Reading



CRUD

Instruments

- HBase Shell
- Native Java API
- Thrift API
- Rest API



WE USE SHELL!!!

Create

- Creates new table
- create '<table name>', '<column family>'
- create '<table name>', {NAME => 'column family1', VERSIONS => N}

Put

- Add new entry or update the current
- put '<table name>','row1','<colfamily:colname>','<value>'

Get

- Get entry by KeyRow
- get '<table name>','rowid'
- get '<table name>','rowid', {COLUMN \Rightarrow 'column family:column name',
VERSIONS \Rightarrow N}

Scan

- Return entries one by one
- scan '<table name>'
- scan '<table name>', {COLUMNS => ['c1', 'c2'], LIMIT => N, STARTROW => 'rowkey1'}
- scan '<table name>', {TIMERANGE => [T1, T2]}
- scan '<table name>', {ROWPREFIXFILTER => 'prefix', FILTER => (QualifierFilter (>=, 'binary:xyz')) AND (TimestampsFilter (T1, T2))}

Delete

- Tombstone mark for entry
- delete '<table name>', '<row>', '<column name >', '<timestamp>'

Indices

Indices

- RowKey is primary index
- No support for other types
- Table transposition may help

Transactions

Transactions

- No transactions
- Atomicity of operations at a row-level
- Scan guarantees

Back up

Back up

- Replication mechanism
- Snapshots

Security

Security

- RBAC
- VL
- Data encryption in some states

Conclusion

Key points

- Java and HDFS based
- Distributed
- Open Source
- Horizontal sharding
- No Indices & Transaction
- No «execution plan»
- BigData & Data Mining

Useful links

- [Official site](#)
- [Documentation](#)
- [Git](#)
- [Graceful lecture about DFS](#)
- [Docker && Hbase](#)
- [More about Zookeeper](#)

Thanks for attention!

