

“Big data is like teenage sex: everyone talks about it, nobody really knows how to do it, everyone thinks everyone else is doing it, so everyone claims they are doing it...”—[Dan Ariely](#)

MYSQL AND HBASE ECOSYSTEM FOR REAL-TIME BIG DATA OVERVIEWS

Lukas Putna, Tomas Komenda
Seznam.cz

Who are we?



- Web portal, search engine in the Czech Republic
- 40+ different web services (search, news, email, media, ...)



- PPC ads, AdWords competitor in CZ
- Lukáš Putna, Tomáš Komenda
 - Senior developers, database specialists
 - MySQL, HBASE, Hadoop, MongoDB
 - MySQL trainings, internal consultations



Sklik.cz admin web

- Advertising data + its **daily** statistics
- Should show real time analytics
- Takes hours to aggregate and serve

The screenshot shows a web-based administrative interface for Sklik.cz. At the top, there's a header bar with a back arrow, forward arrow, refresh button, and a search bar. Below the header are date selection fields for 'From date' and 'To Date', and a 'Filter' section with a dropdown menu set to 'Only active'. A large table below displays various keywords along with their status, CPC, impressions, CTR, and some summary rows at the bottom.

Keyword	▼▲	Status	▼▲	CPC	▼▲	Impressions ▼▲	...	CTR	▼▲
Notebooks		Active		1.00 €		123 456 223	...	1.56 %	
Web cams		Active		0.20 €		85 663 312	...	0.02 %	
Computers		Paused		0.20 €		20 231	...	1.02 %	
Harddisk		Active		0.20 €		250 232 666	...	0.56 %	
Printers		Deleted		0.20 €		56	...	0.01 %	
Headphones		Deleted		0.16 €		456 123	...	0.08 %	
Sum Context network				0.14 €		888 996 236	...	1.20 %	
Sum Search Network				0.10 €		145 568 986	...	2.50 %	
Total sum				0.13 €		1 034 565 222	...	0.88 %	

At the bottom, there are navigation links: « Previous, 1, 2, 3, 4, 5, Next ».

Account example

- 10M keywords
- 120 statistical values per keyword per day

With date filter of 1 year

- 42 billion of values
- 3 aggregated sum rows

Sklik.cz admin web

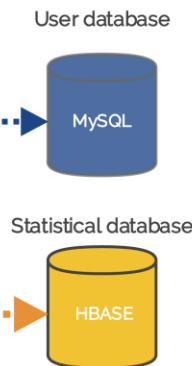
- Advertising data + its daily statistics
- Needs optimization
- Hours -> seconds

Sklik.cz - Keywords for campaign Conference

From date To Date Filter

Keyword	Status	CPC	Impressions	...	CTR
Notebooks	Active	1.00 €	123 456 223		1.56 %
Web cams	Active	0.20 €	85 663 312		0.02 %
Computers	Paused	0.20 €	20 231		1.02 %
Harddisk	Active	0.20 €	250 232 666		0.56 %
Printers	Deleted	0.20 €	56		0.01 %
Headphones	Deleted	0.16 €	456 123		0.08 %
Sum Context network		0.14 €	888 996 236		1.20 %
Sum Search Network		0.10 €	145 568 986		2.50 %
Total sum		0.13 €	1 034 565 222		0.88 %

« Previous 1 2 3 4 5 Next »

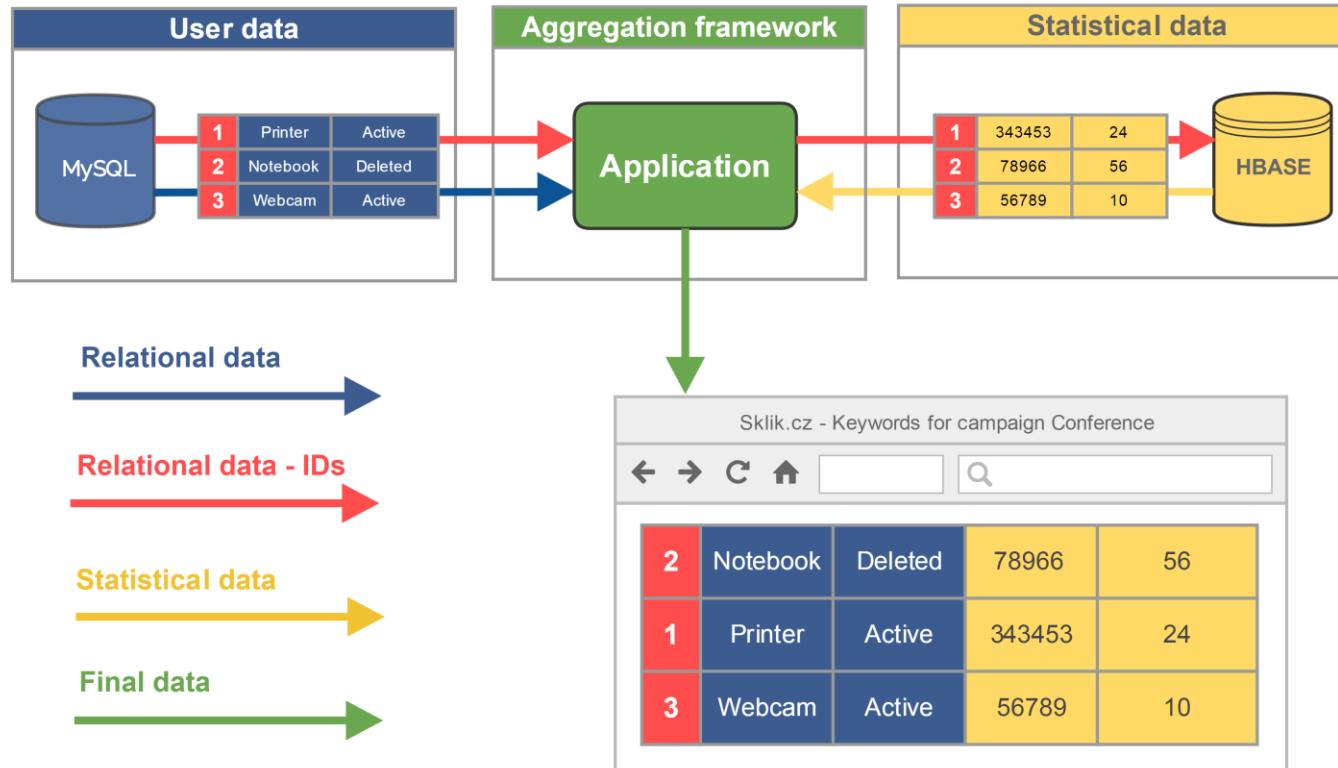


Optimization possibilities

- w/o optimization, unrealistic in real time
- Dedicated databases
 - MySQL for relational data
 - NoSQL (HBase) for non-relational stats
- **Basic idea** is distributed processing
- Major optimization - filtering and pagination
 - Early row elimination -> aggregating and sorting far less data
 - 1st page faster than 1000th
 - 2 DBs means more implementation details
- Other optimization - transferring a lot of IDs

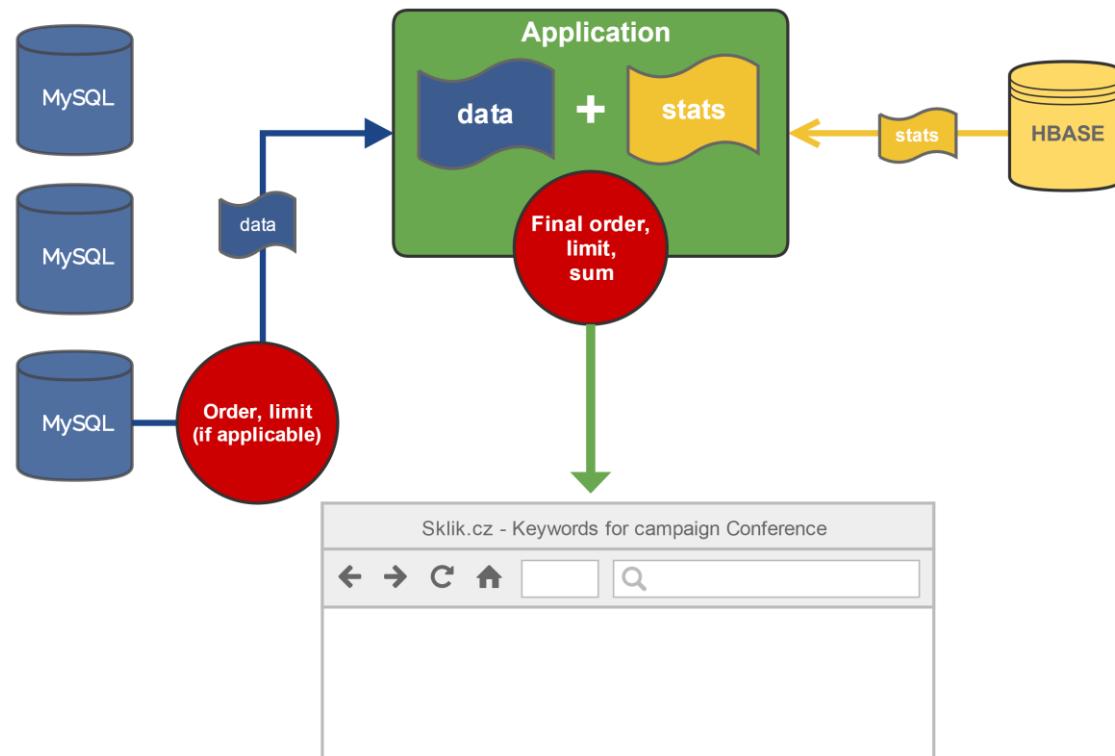
Data flow

- Data from two databases
 - Joined using entity ID



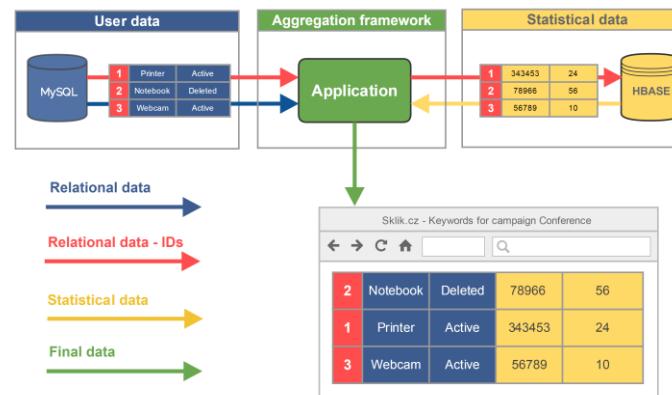
MySQL Optimization

- MySQL shard (could be cluster as well)
- Most of the dataset in memory
- Indexes for ordering, pagination (limit and offset)



“IDs” optimization

- Transferring a lot of relational IDS
 - **Text vs binary** protocol (prepared statements - PS)
 - Millions of $int \rightarrow string \rightarrow int$ conversions
 - 1M 4B ints: 600 ms \rightarrow 250 ms (more than 2x)
- **SuperiorMySql++**
 - open source library by Seznam.cz, C API Wrapper
 - advantage of PS in sending binary data
 - <https://github.com/seznam/SuperiorMySqlpp>



HBase



*"Finance here - we're not sure
about this Hadoop thing...
Could you just dump it all
into Excel for us?"*

TimoElliott.com

HBase - introduction

Database management system:

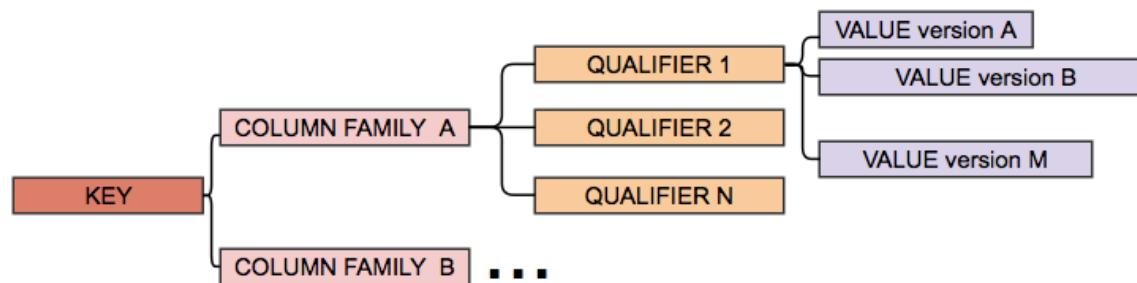
- distributed
- usually runs on top of HDFS
- data physically stored in key order
- MR1/Yarn integration
- self-managed data partitioning
- suited for range scans and high inserts volume (LSM trees)
- No-SQL
- Row-level write atomicity



HBase – data organization

Comprises set of tables, each table contains:

- rows and columns (but no fixed schema)
- primary key (data stored in PK order, not SK)
- divided into column-families
- more versions of value



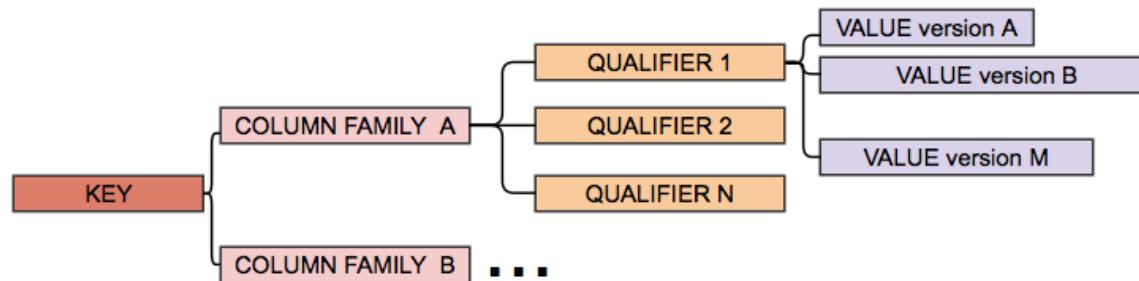
```
CREATE TABLE_A ();
```

Explanation via SQL pseudo-language

HBase – data organization

Comprises set of tables, each table contains:

- rows and columns (but no fixed schema)
- primary key (data stored in PK order, not SK)
- divided into column-families
- more versions of value



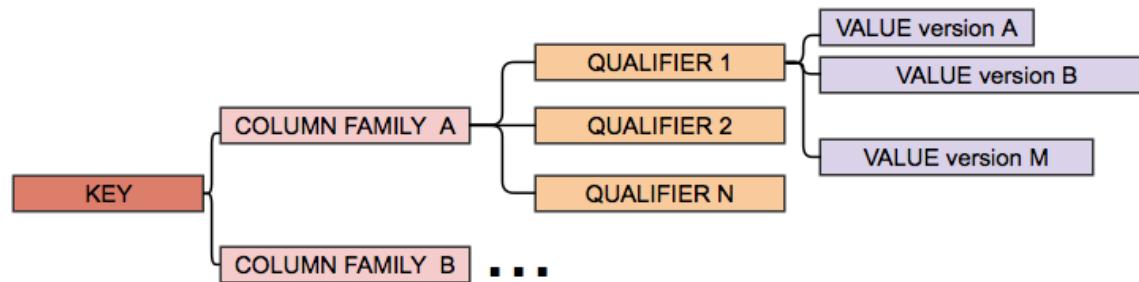
```
CREATE TABLE_A (
    key BINARY);
```

Explanation via SQL pseudo-language

HBase – data organization

Comprises set of tables, each table contains:

- rows and columns (but no fixed schema)
- primary key (data stored in PK order, not SK)
- divided into column-families
- more versions of value



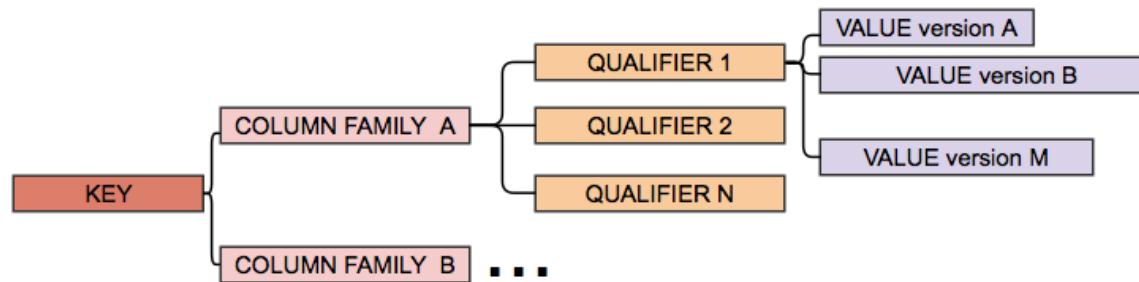
```
CREATE TABLE_A (
    key BINARY,
    column_family VARCHAR(...)) ;
```

Explanation via SQL pseudo-language

HBase – data organization

Comprises set of tables, each table contains:

- rows and columns (but no fixed schema)
- primary key (data stored in PK order, not SK)
- divided into column-families
- more versions of value



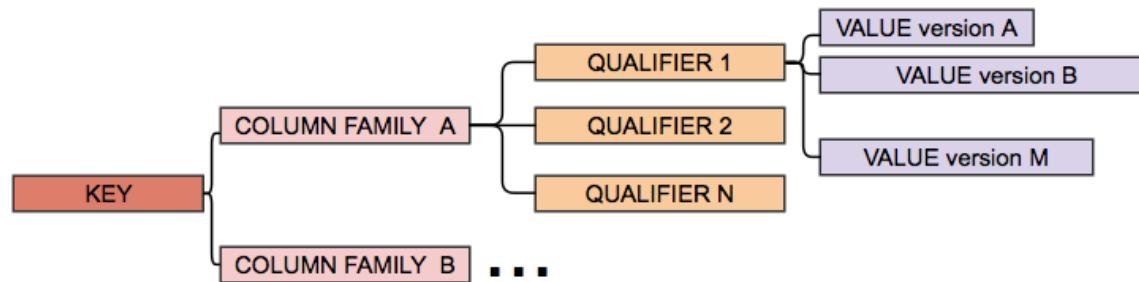
```
CREATE TABLE_A (
    key BINARY,
    column_family VARCHAR(...),
    qualifier BINARY);
```

Explanation via SQL pseudo-language

HBase – data organization

Comprises set of tables, each table contains:

- rows and columns (but no fixed schema)
- primary key (data stored in PK order, not SK)
- divided into column-families
- more versions of value



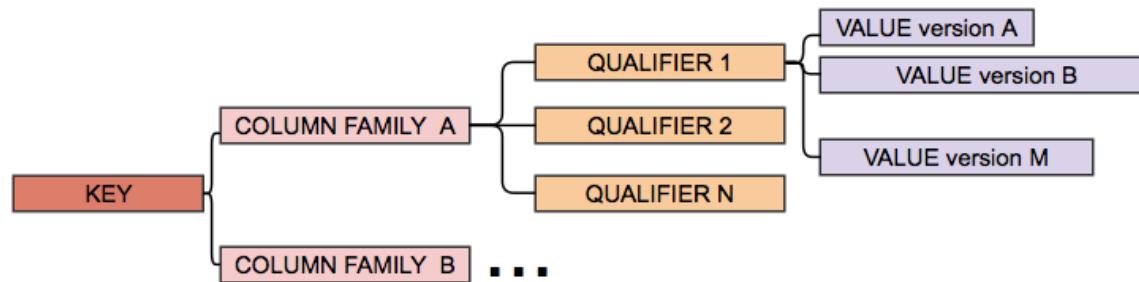
```
CREATE TABLE_A (
    key BINARY,
    column_family VARCHAR(...),
    qualifier BINARY,
    version TIMESTAMP,
    value BINARY);
```

Explanation via SQL pseudo-language

HBase – data organization

Comprises set of tables, each table contains:

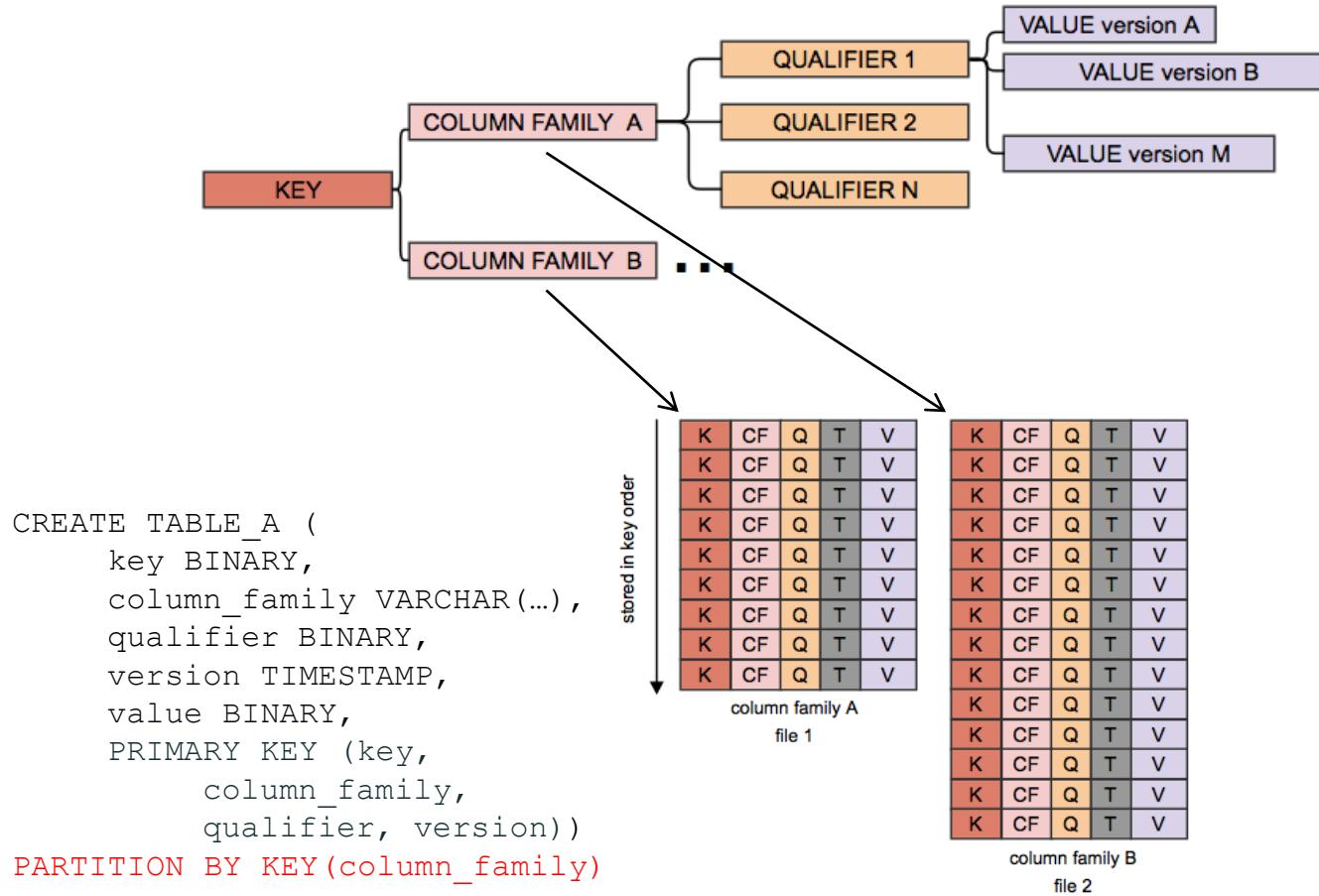
- rows and columns (but no fixed schema)
- primary key (data stored in PK order, not SK)
- divided into column-families
- more versions of value



```
CREATE TABLE_A (
    key BINARY,
    column_family VARCHAR(...),
    qualifier BINARY,
    version TIMESTAMP,
    value BINARY,
    PRIMARY KEY (key, column_family, qualifier, version));
```

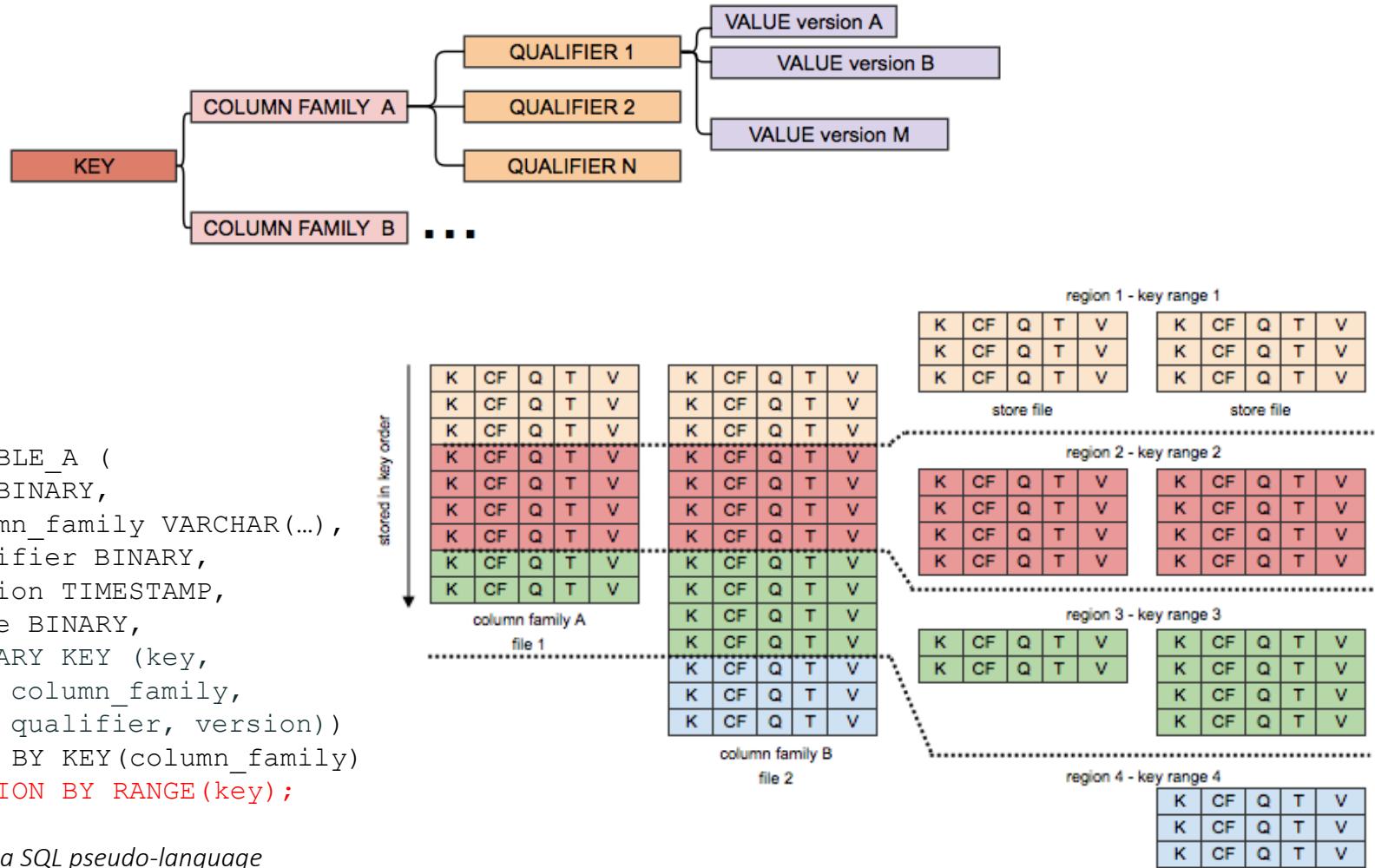
Explanation via SQL pseudo-language

HBase – physical organization

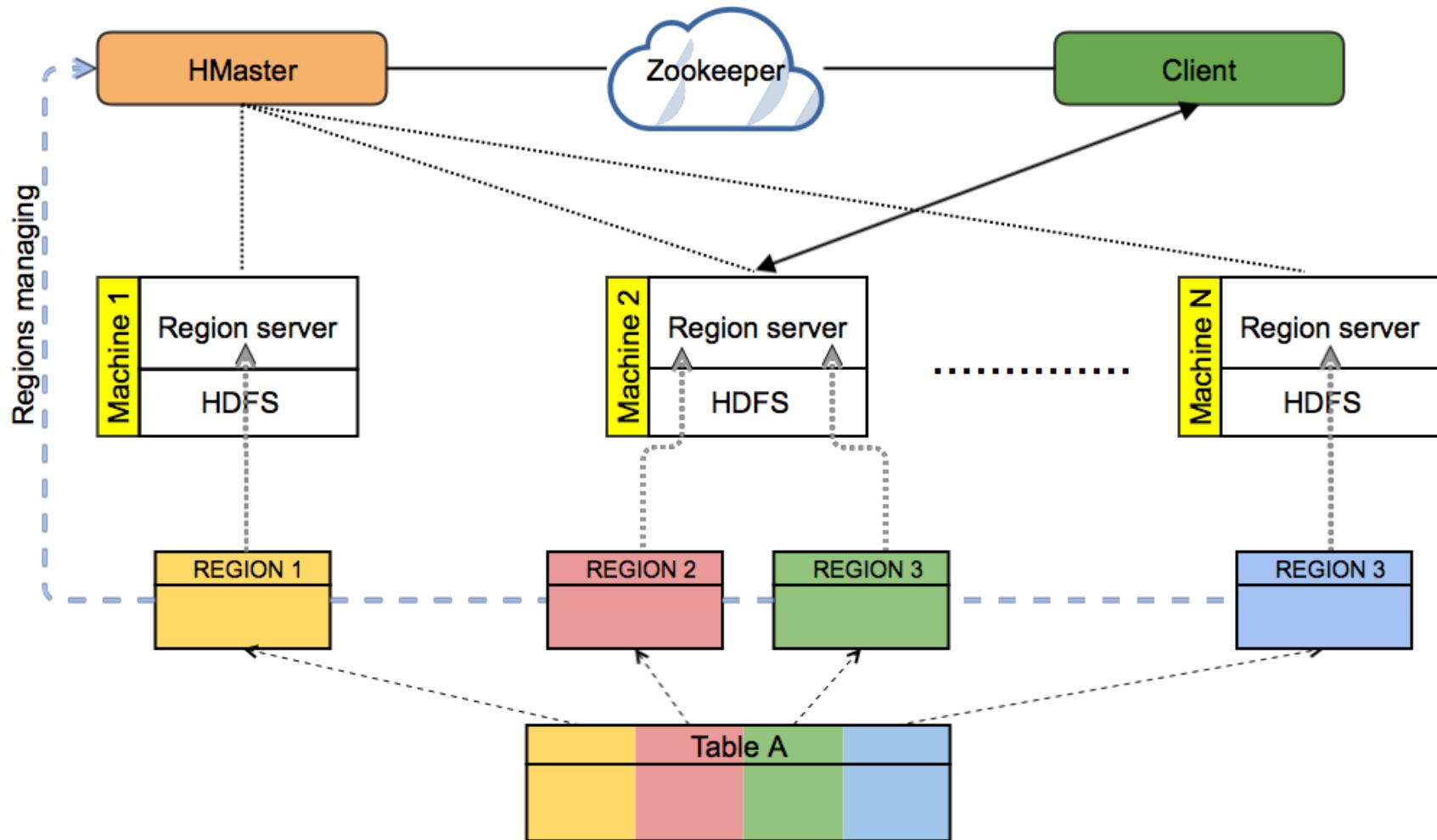


Explanation via SQL pseudo-language

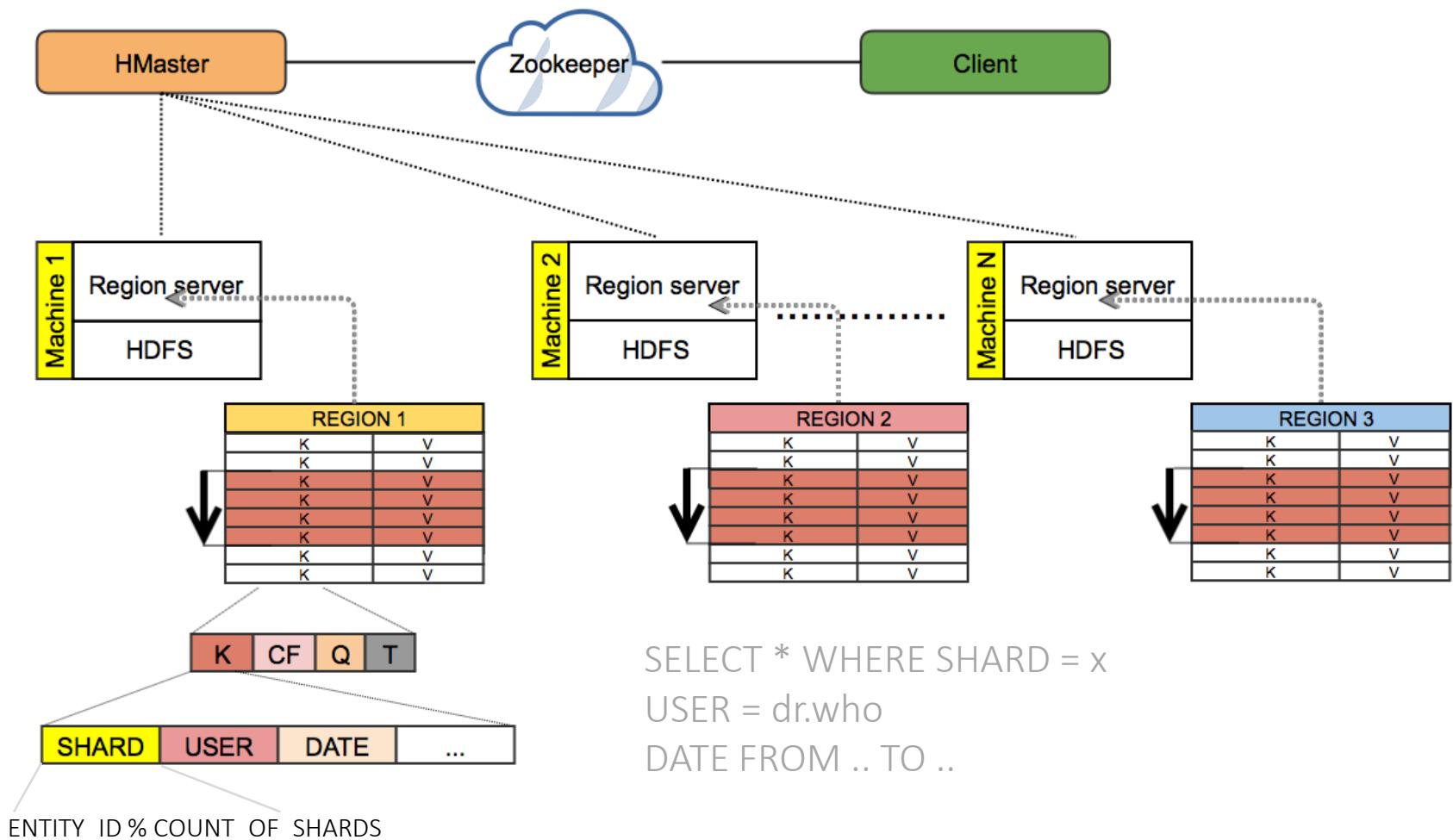
HBase – physical organization



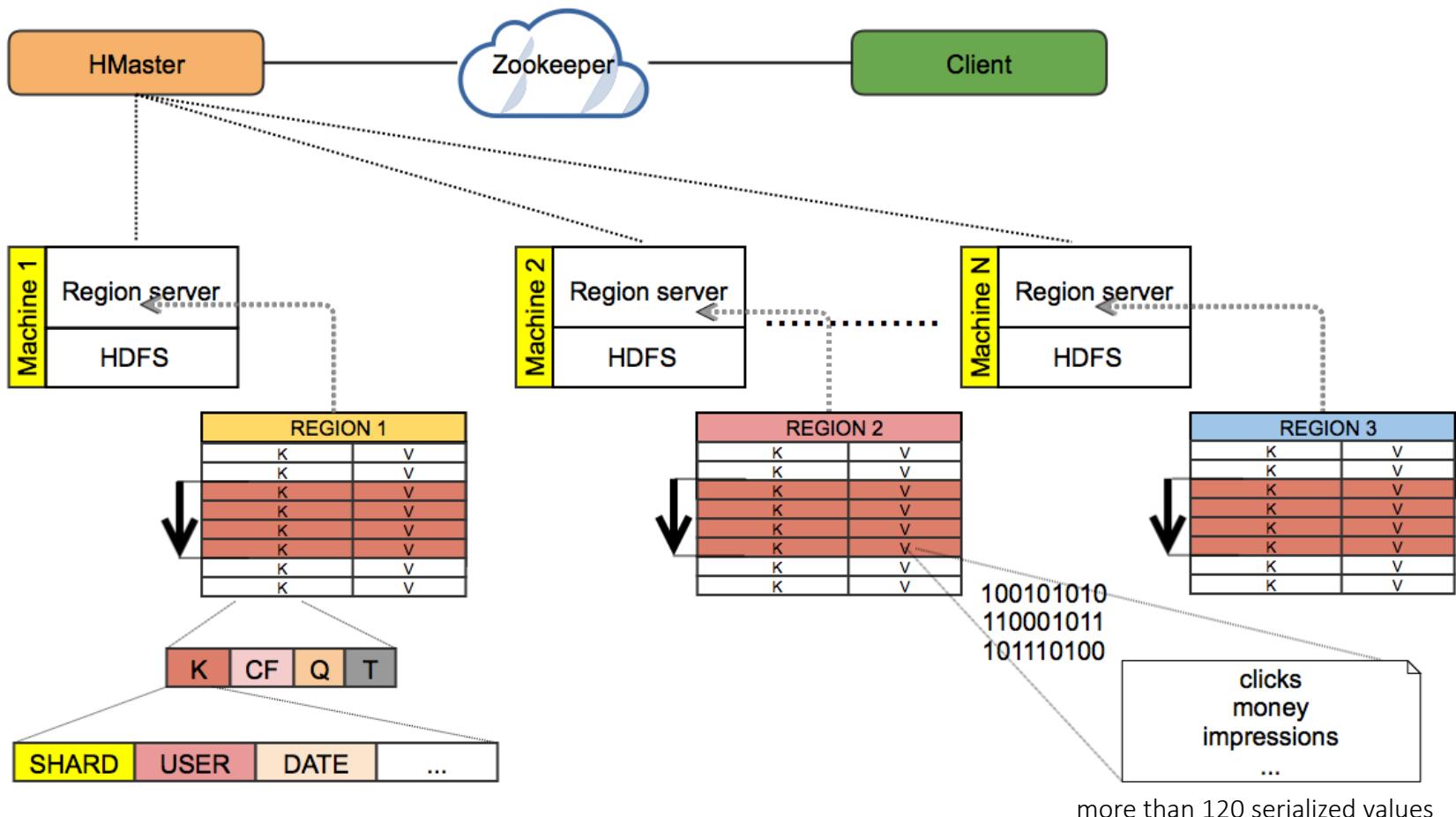
HBase cluster - architecture



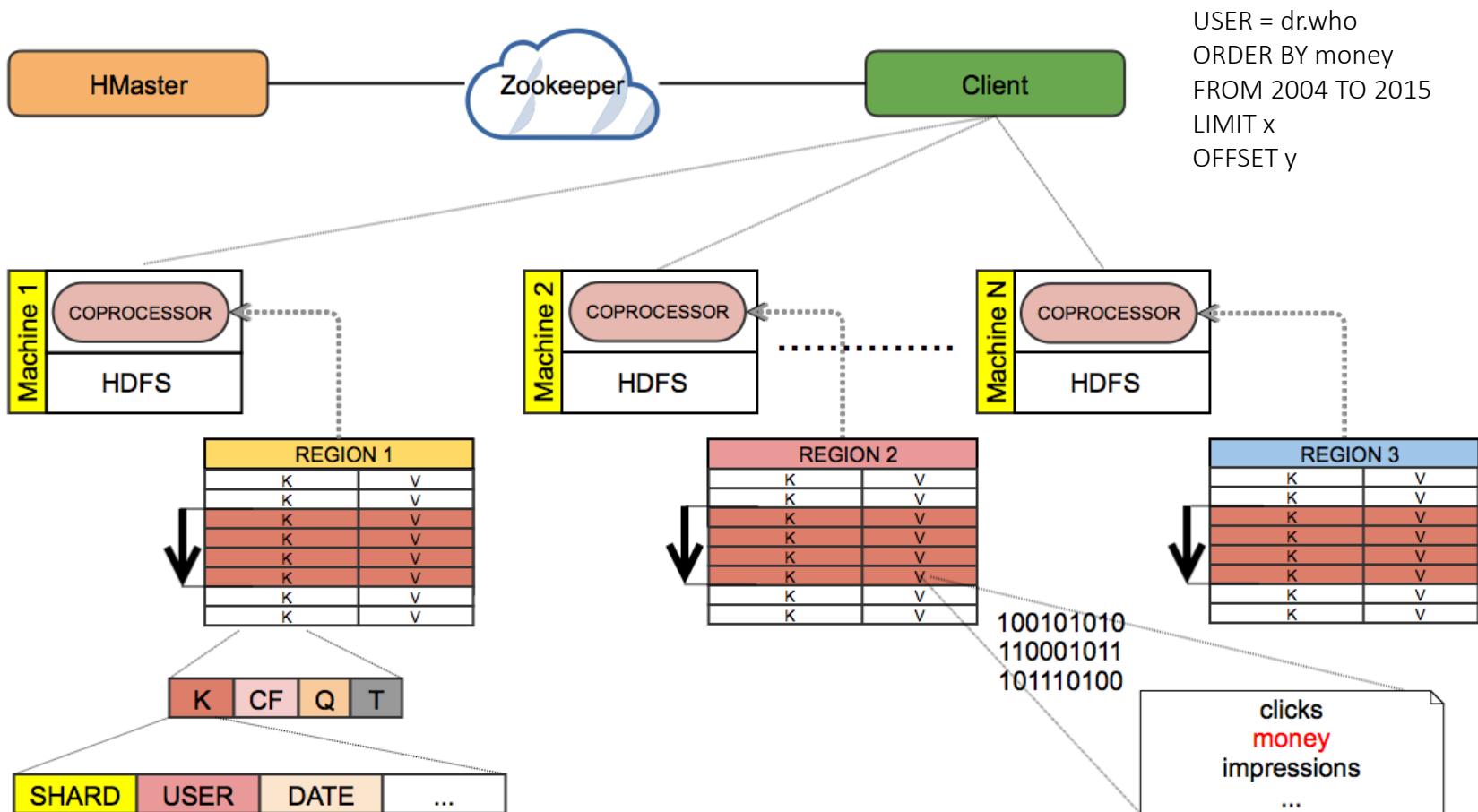
HBase – Sklik use case



HBase – Sklik use case

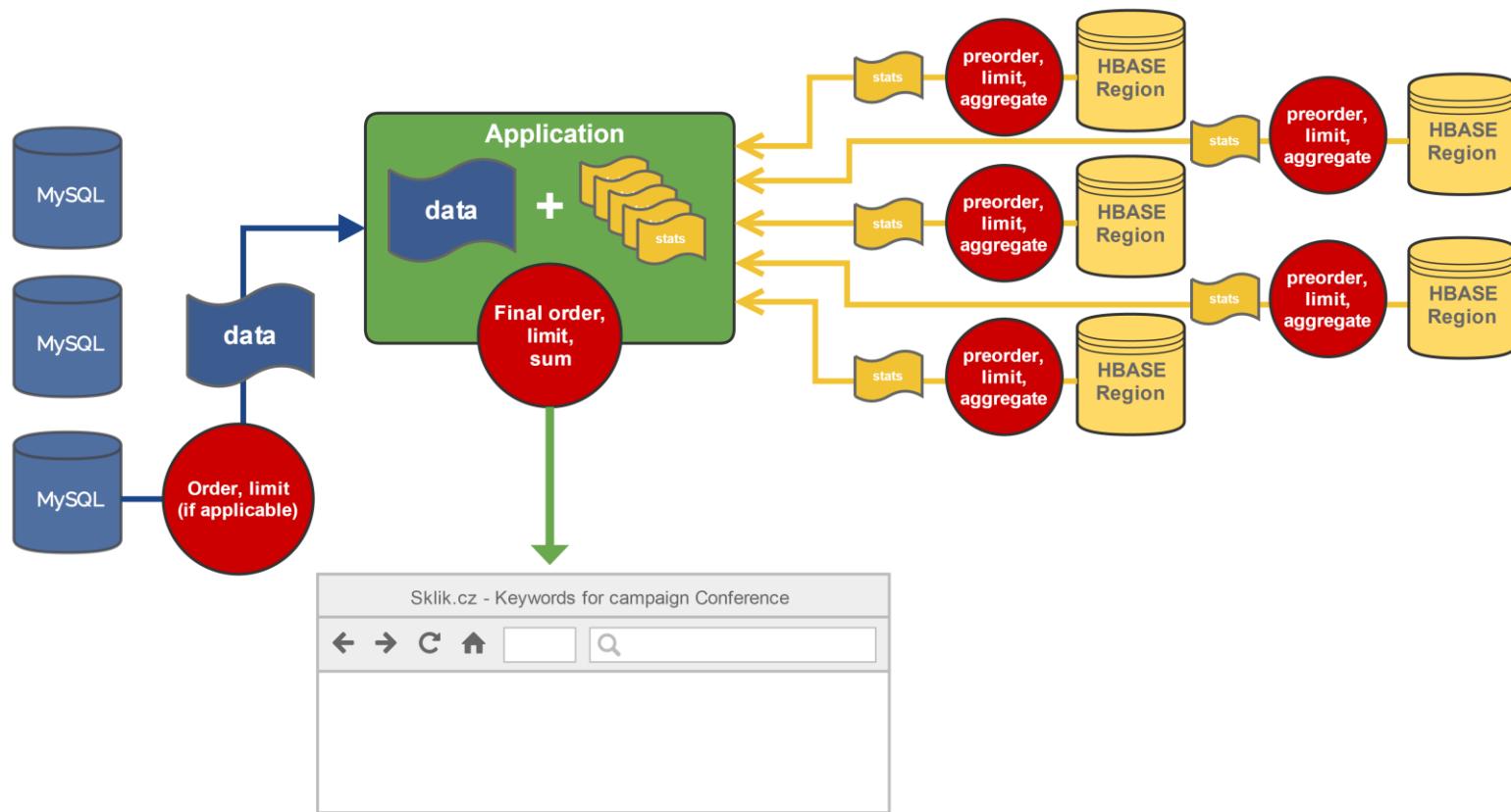


HBase – Sklik use case



Conclusion

- Processing is distributed over the whole ecosystem



Questions?

Thank you for listening

lukas.putna@firmaseznam.cz

tomas.komenda@firmaseznam.cz