“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?

Seznam.cz

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

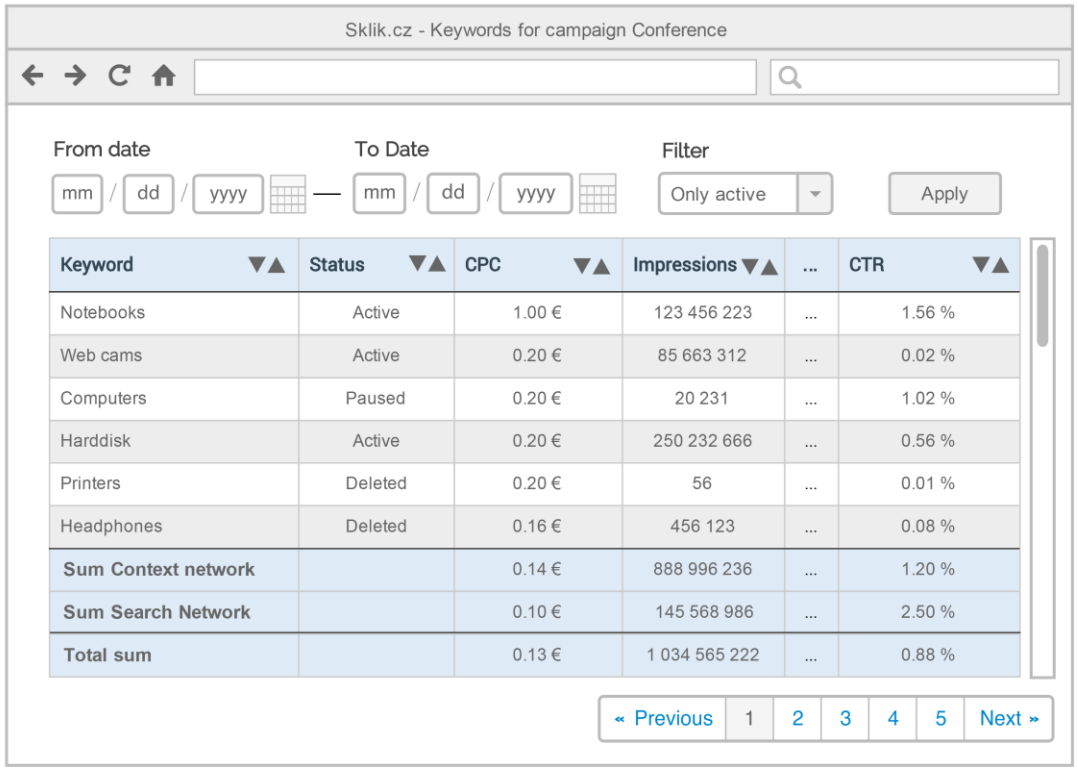
Sklik Seznam.cz

- 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



Account example

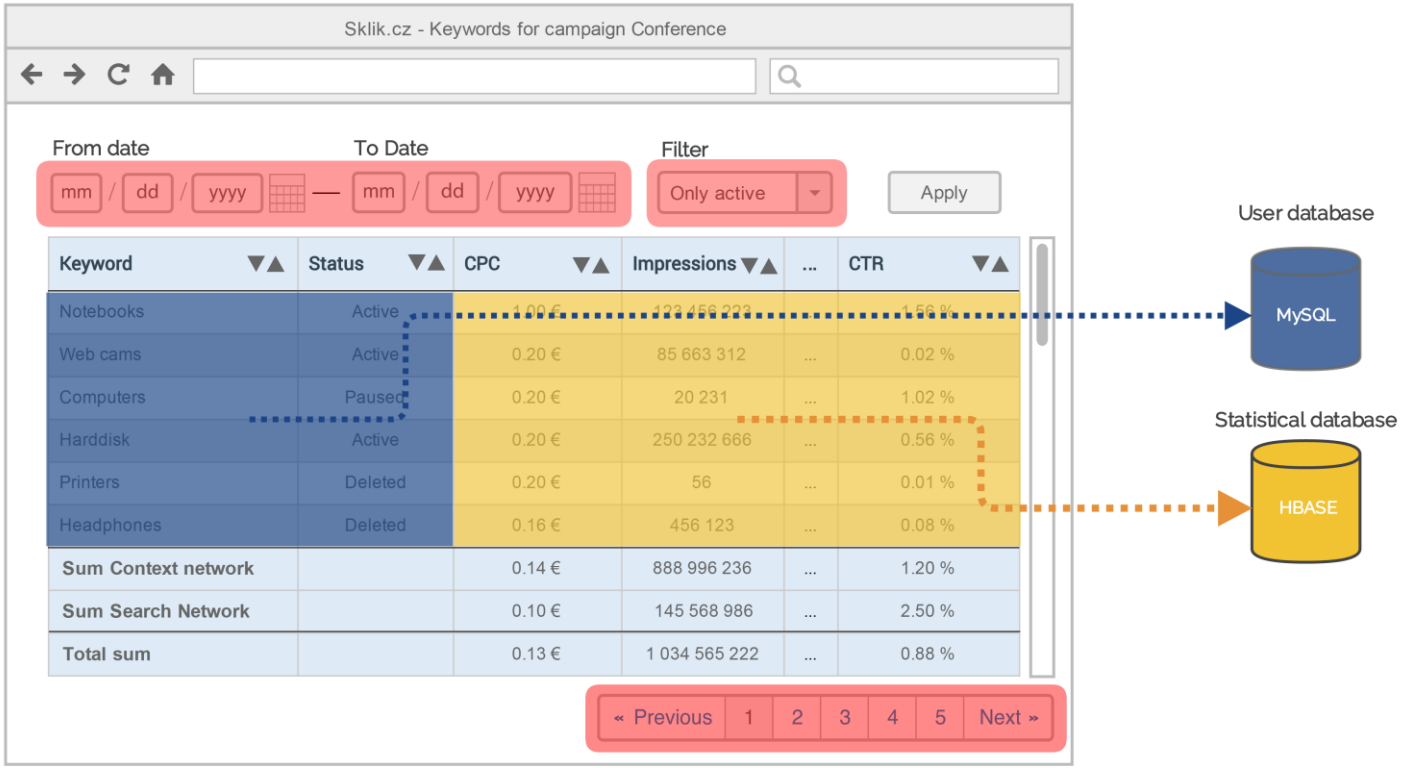
- 10M keywords
- 120 statistical values per keyword per day

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



Optimization possibilities

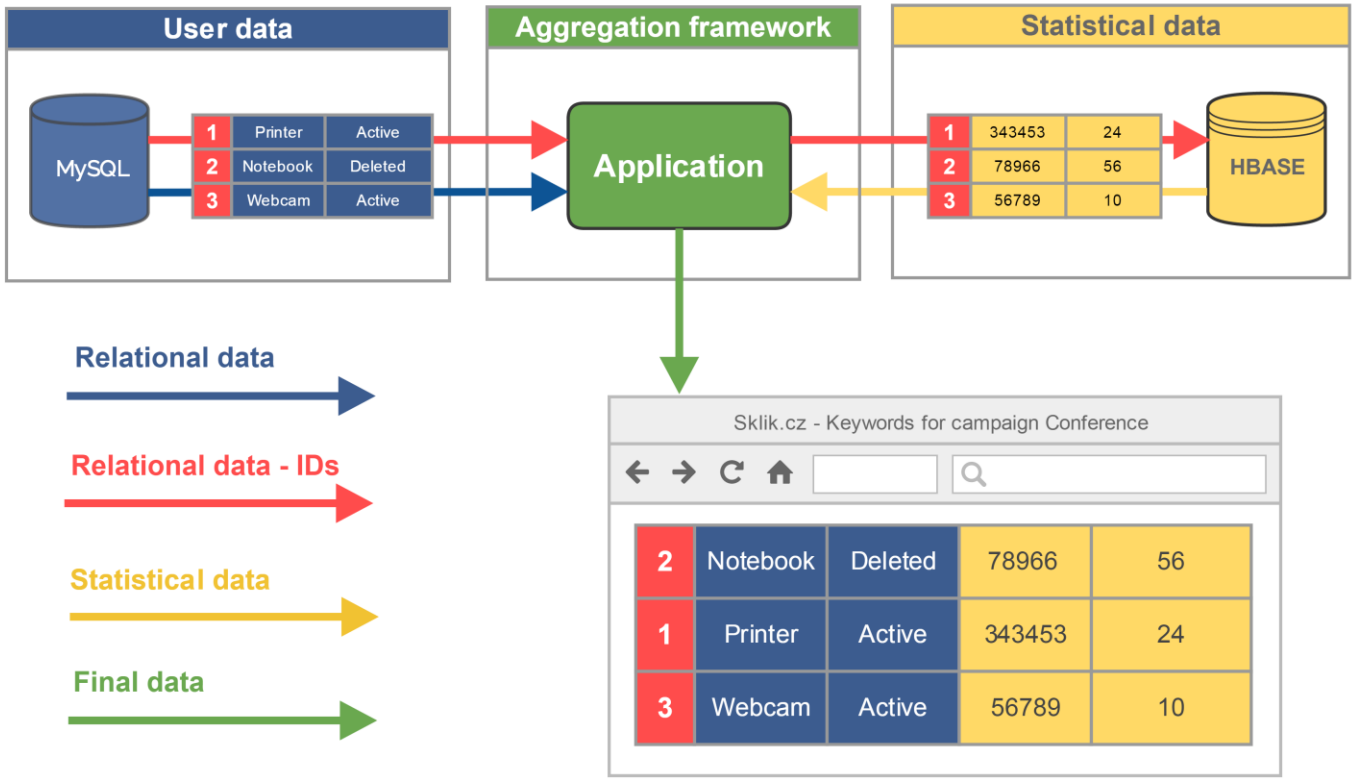
- w/o optimization, unrealistic in real time
- Dedicated databases
  - MySQL for relational data
  - NoSQL (HBase) for non-relational stats

- **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

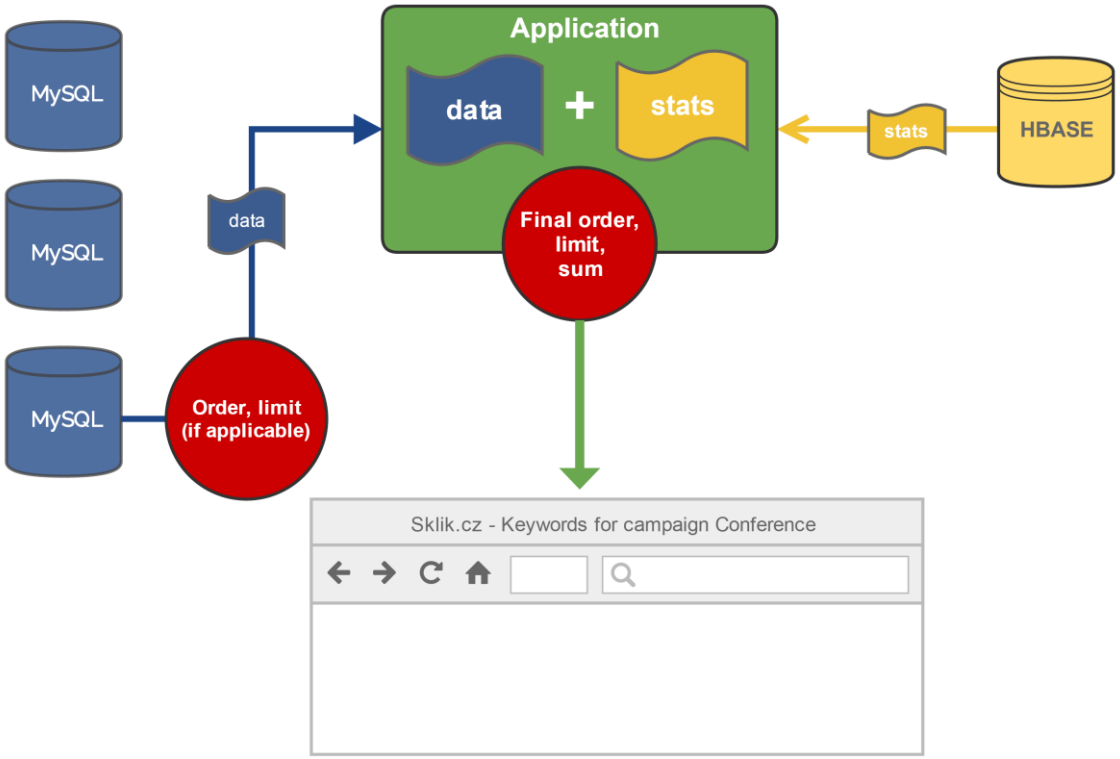
Data flow

- Data from two databases
  - Joined using entity ID



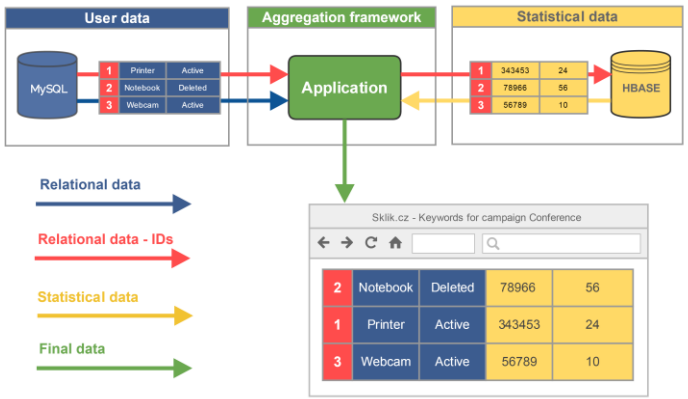
MySQL Optimization

- 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 -> string -> int* conversions
  - 1M 4B ints: 600 ms -> 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

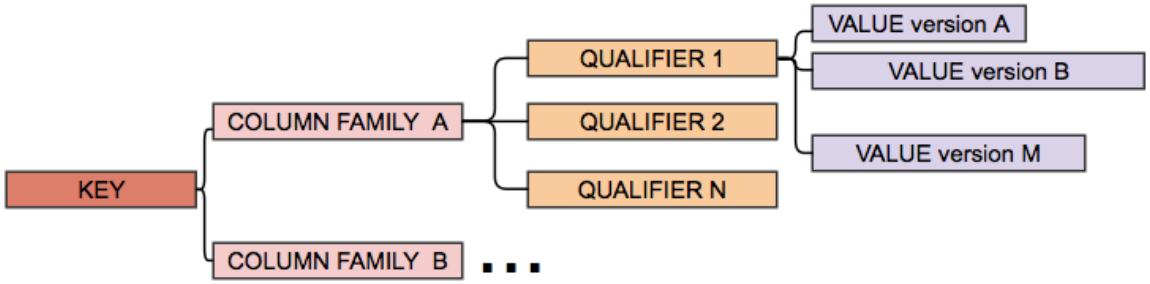
Database management system:

- distributed
- usually runs on top of HDFS
- data physical 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 scheme)
- primary key (data stored in PK order, not SK)
- divided into column-families
- more versions of value



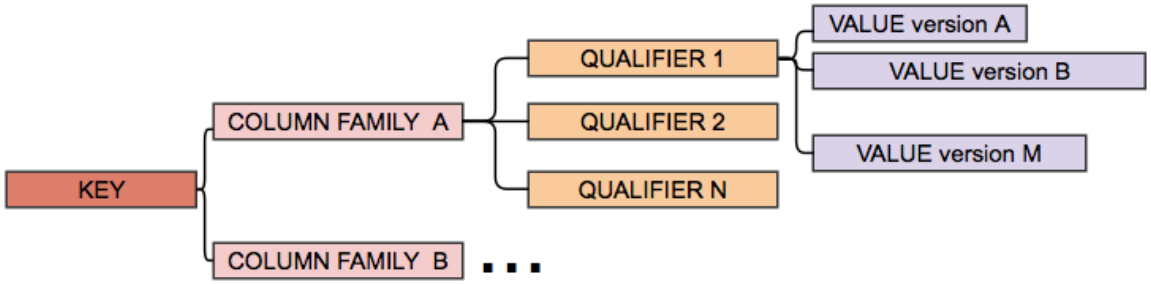

```
CREATE TABLE_A ();
```

Explanation via SQL pseudo-language

HBase – data organization

Comprises set of tables, each table contains:

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



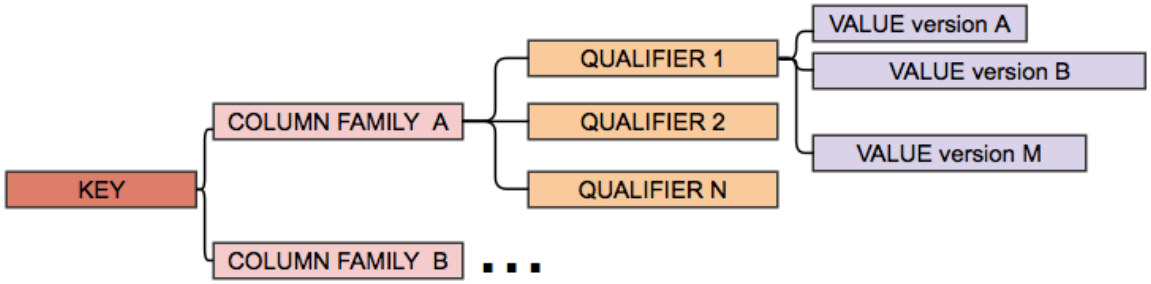

```
CREATE TABLE_A (
     key BINARY);
```

Explanation via SQL pseudo-language

HBase – data organization

Comprises set of tables, each table contains:

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



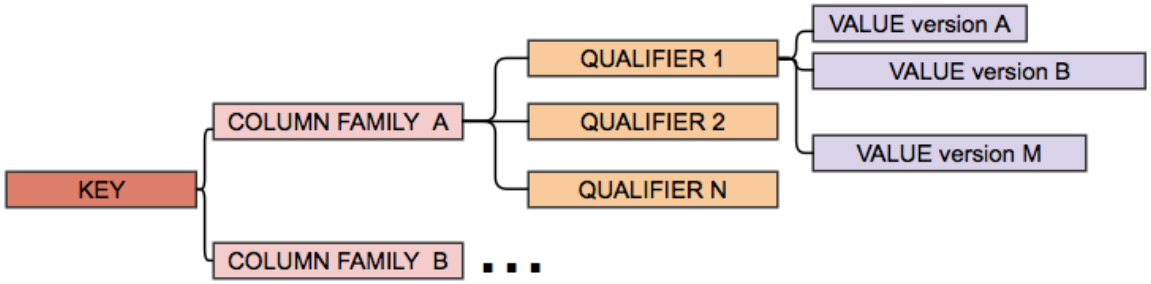

```
CREATE TABLE_A (
     key BINARY,
     column_family VARCHAR(…));
```

Explanation via SQL pseudo-language

HBase – data organization

Comprises set of tables, each table contains:

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



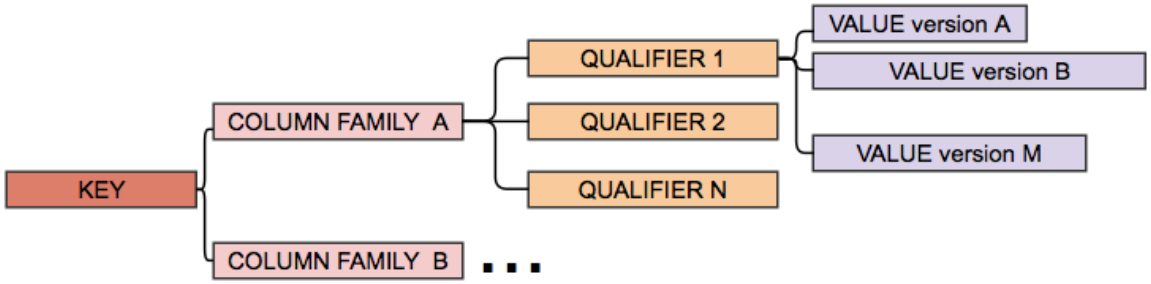

```
CREATE TABLE_A (
     key BINARY,
     column_family VARCHAR(…),
     qualifier BINARY);
```

Explanation via SQL pseudo-language

HBase – data organization

Comprises set of tables, each table contains:

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



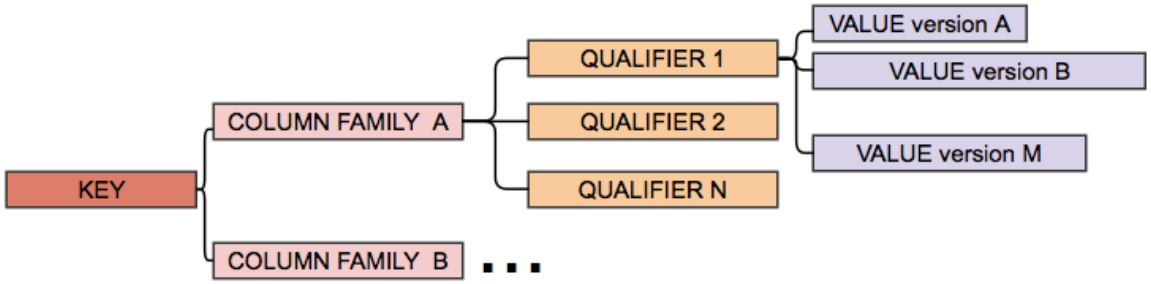

```
CREATE TABLE_A (
     key BINARY,
     column_family VARCHAR(…),
     qualifier BINARY,
     version TIMESTAMP,
     value BINARY);
```

Explanation via SQL pseudo-language

HBase – data organization

Comprises set of tables, each table contains:

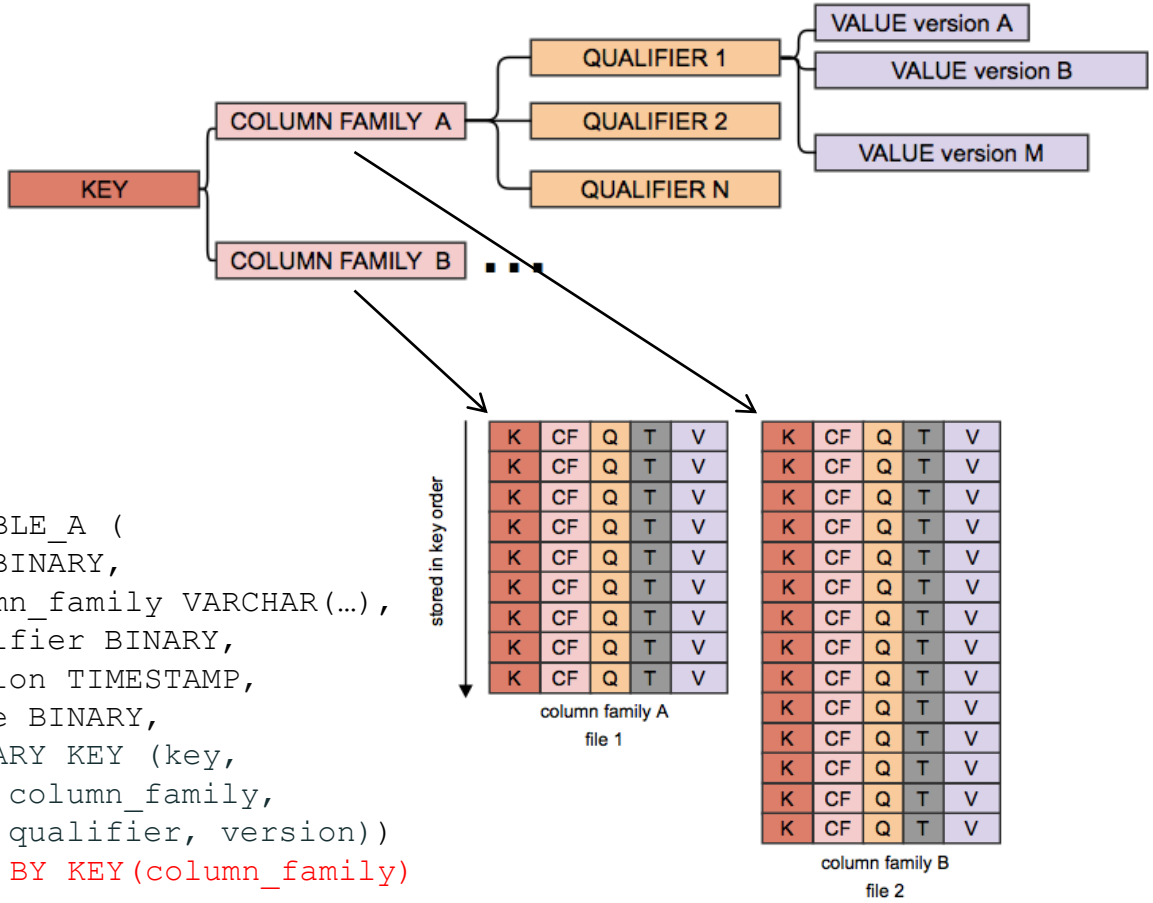
- rows and columns (but no fixed scheme)
- primary key (data stored in PK order, not SK)
- divided into column-families
- more versions of value




```
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

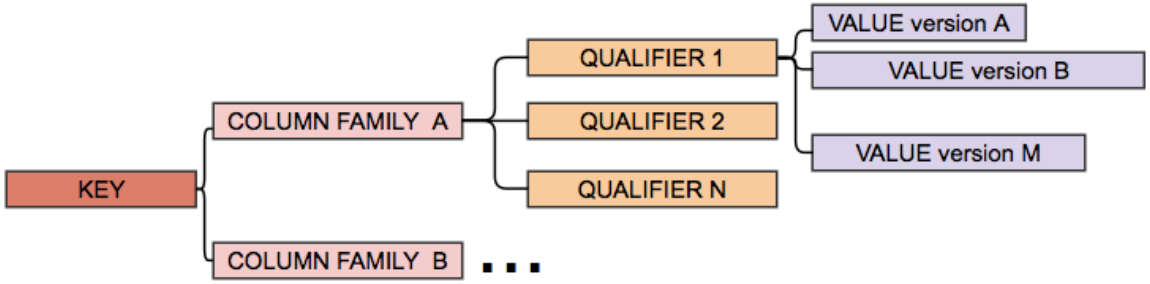
HBase – physical organization



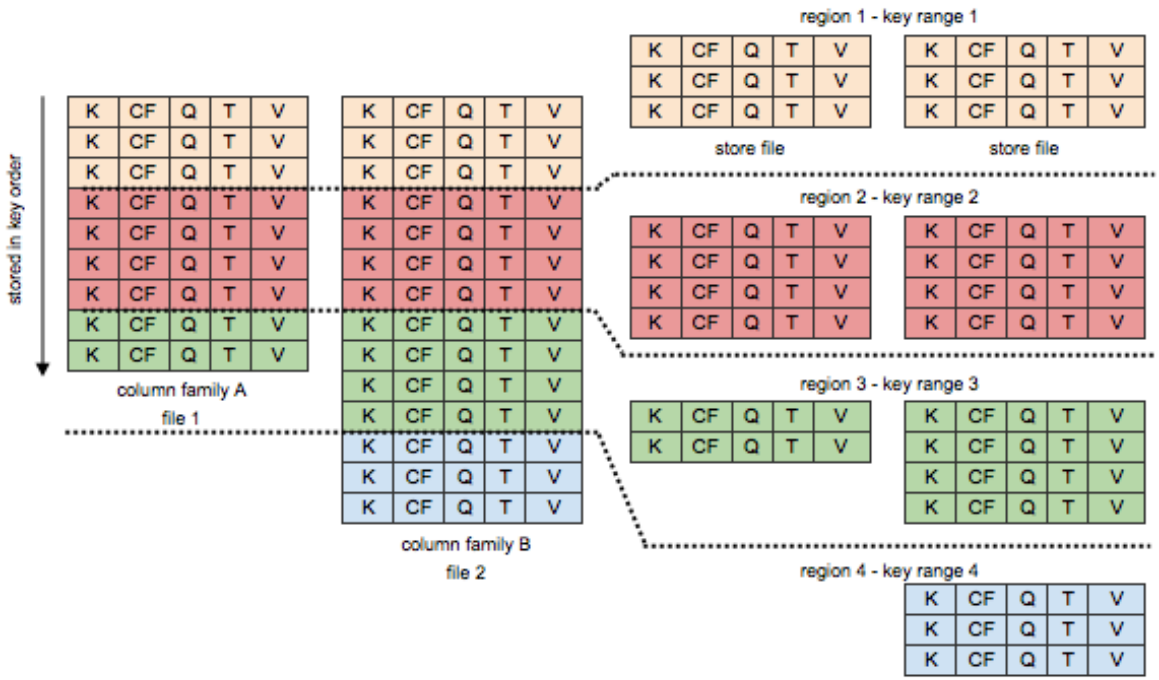

```
CREATE TABLE_A (
     key BINARY,
     column_family VARCHAR(…),
     qualifier BINARY,
     version TIMESTAMP,
     value BINARY,
     PRIMARY KEY (key,
          column_family,
          qualifier, version))
PARTITION BY KEY(column_family)
```

Explanation via SQL pseudo-language

HBase – physical organization

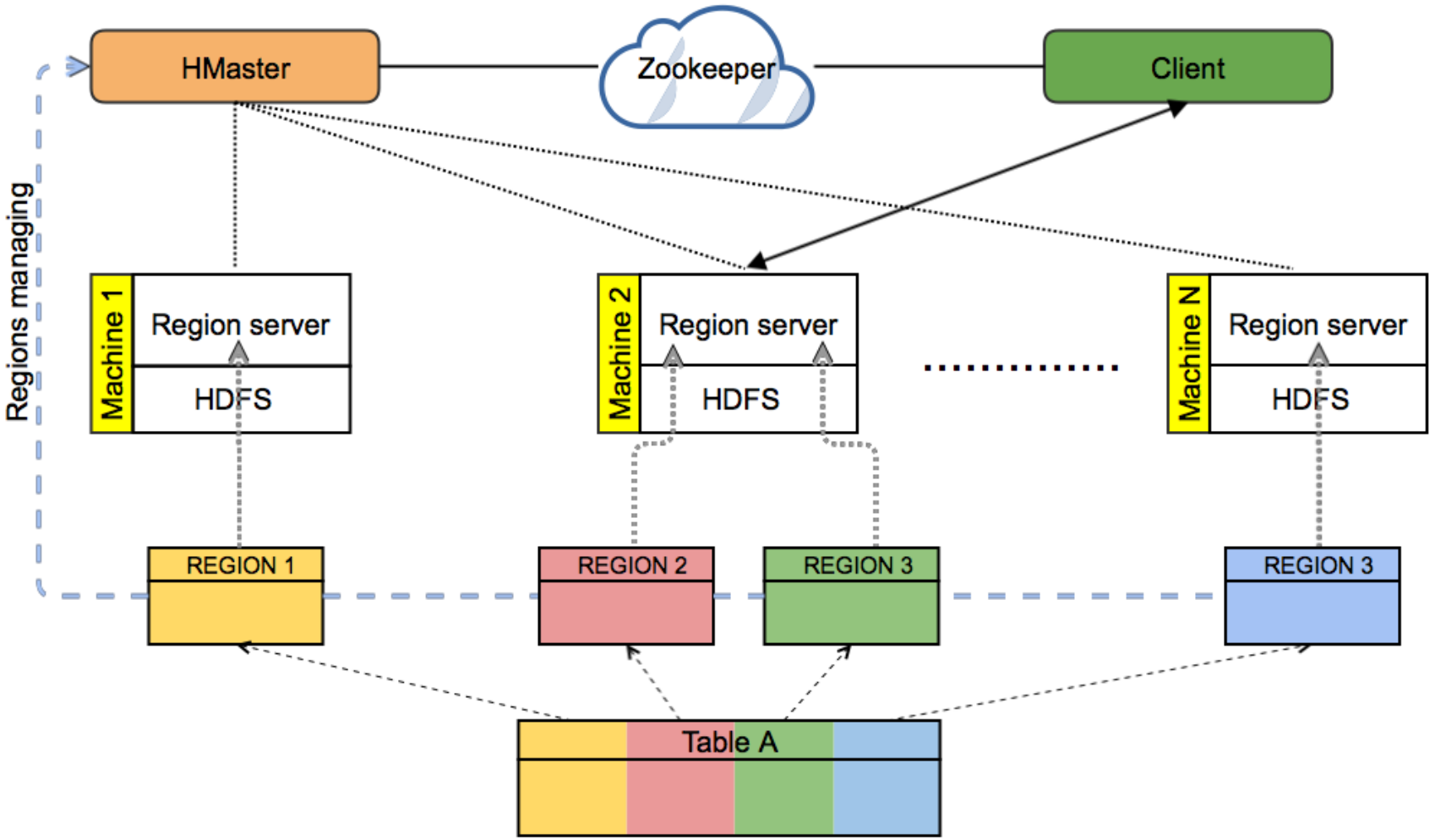



```
CREATE TABLE_A (
      key BINARY,
      column_family VARCHAR(…),
      qualifier BINARY,
      version TIMESTAMP,
      value BINARY,
      PRIMARY KEY (key,
            column_family,
            qualifier, version))
PARTITION BY KEY(column_family)
SUBPARTITION BY RANGE(key);
```

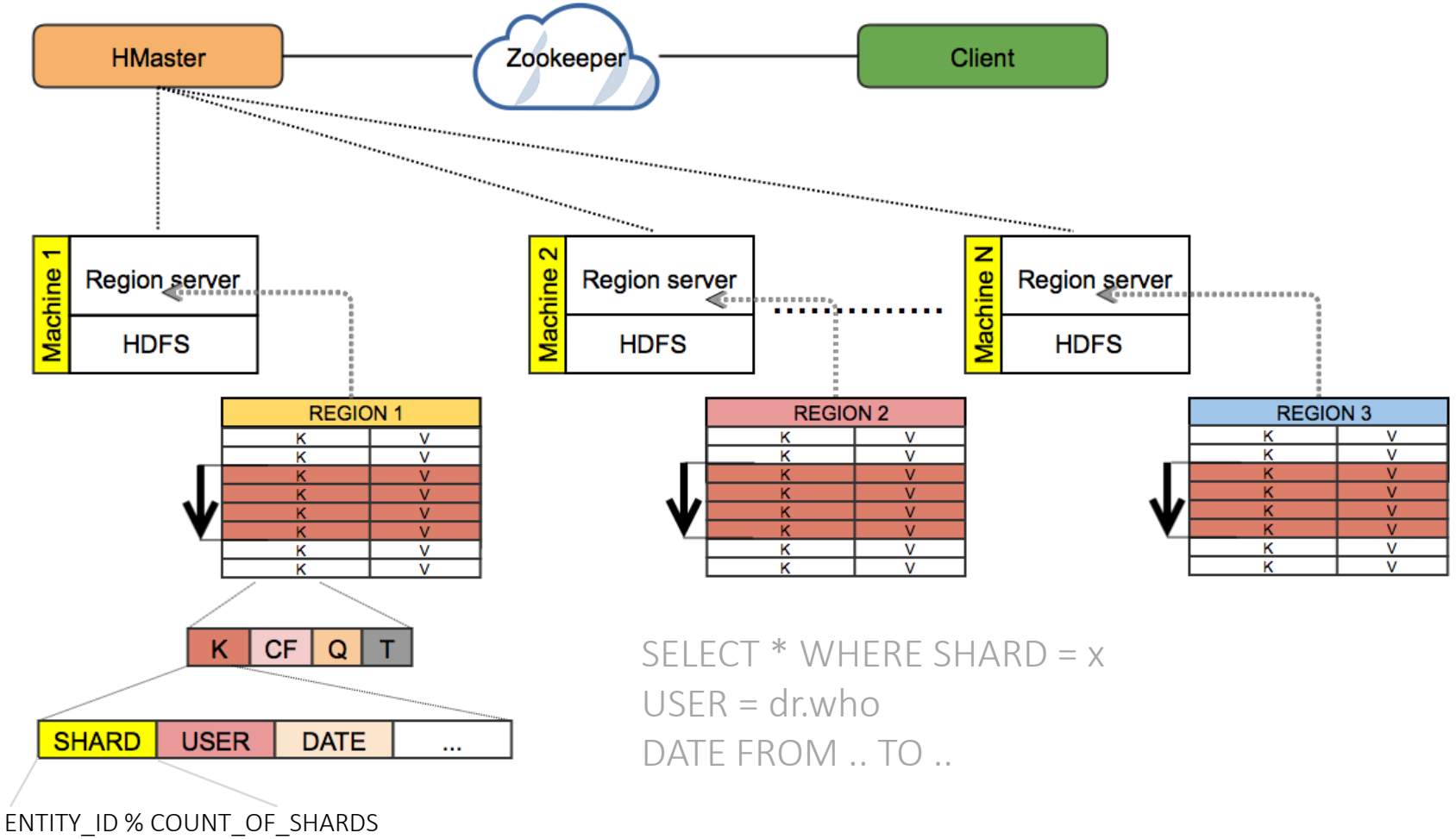


Explanation via SQL pseudo-language

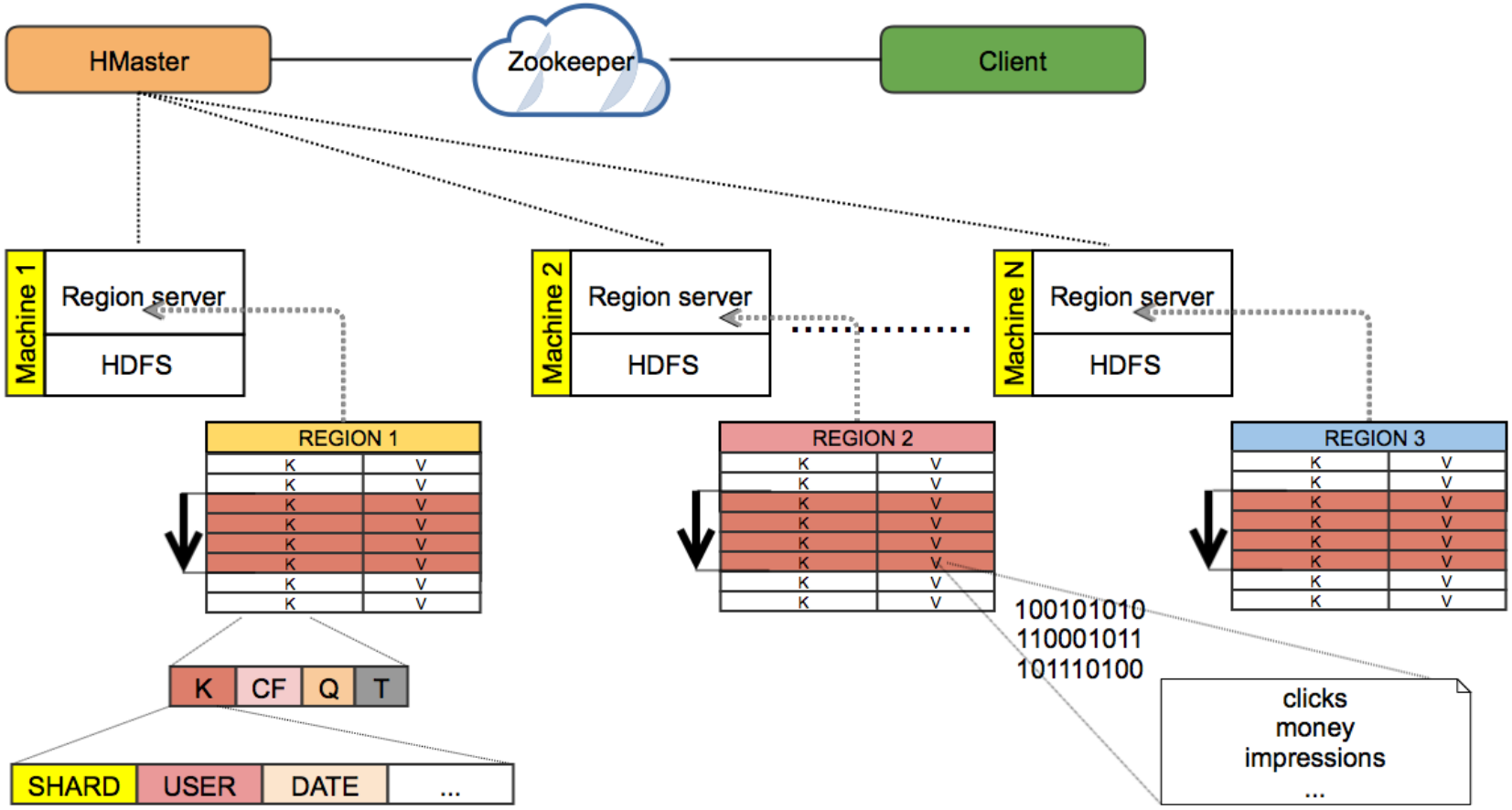
HBase cluster - architecture



HBase – Sklik use case

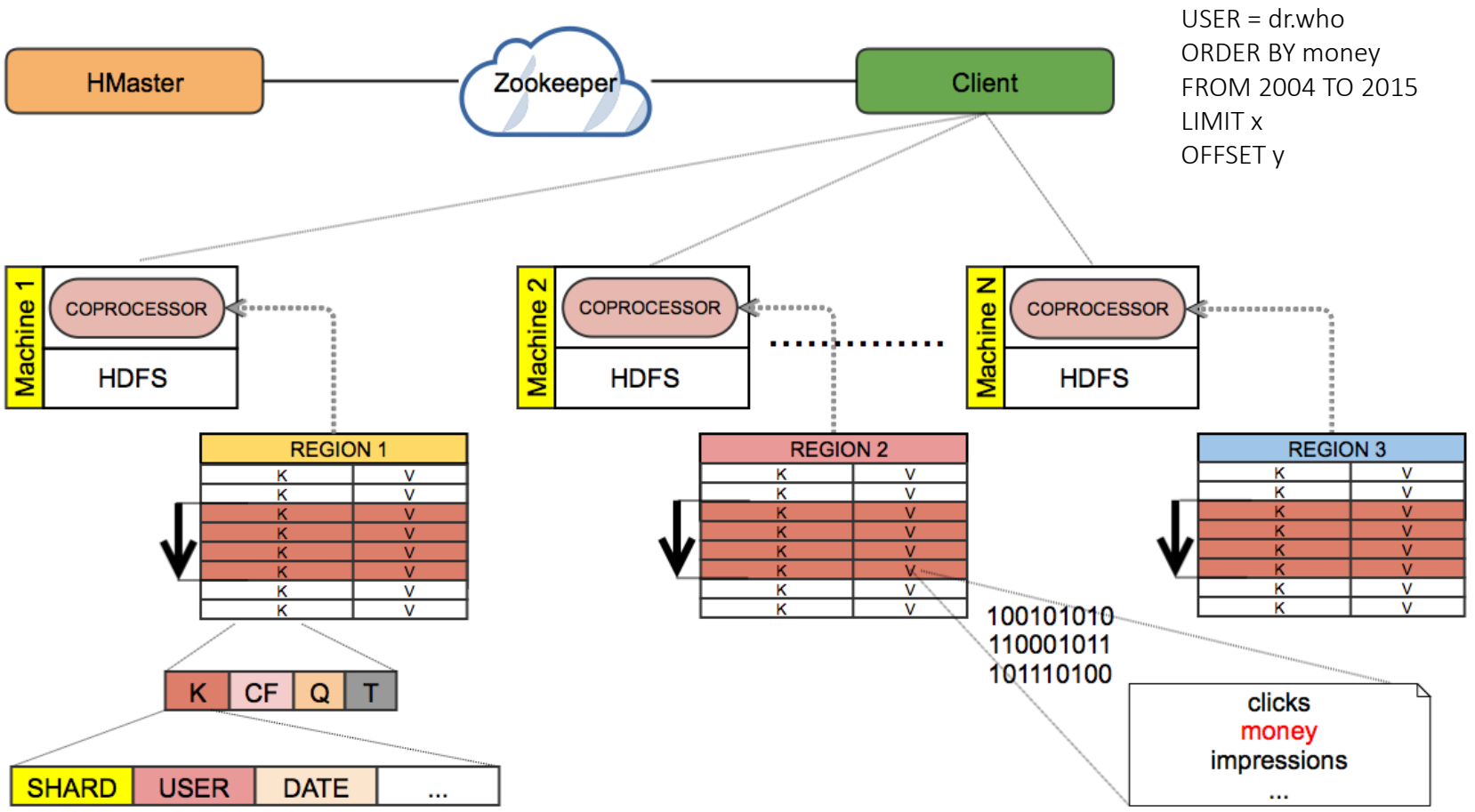


HBase – Sklik use case



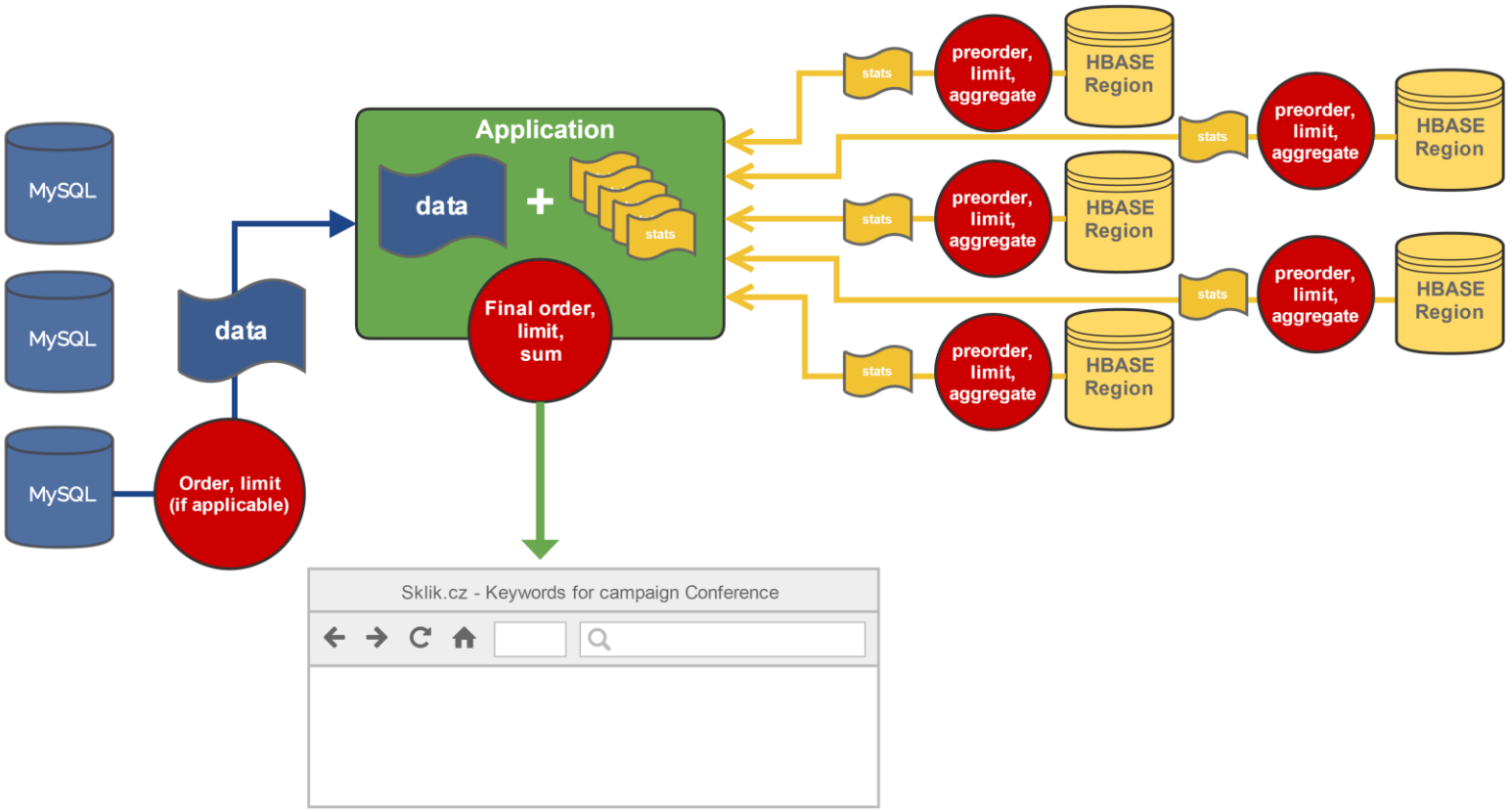
more than 120 serialized values

HBase – Sklik use case



Conclusion

- Processing is distributed over the whole ecosystem



Questions?

Thank you for listening

lukas.putna@firma.seznam.cz
tomas.komenda@firma.seznam.cz