**Projet BigData**

**Processing and storage of time series**

USER COMMAND

**Database**

**Spark RDD** 🡪 the database is save on local hdd

Table1(id, value)

Table2(id, value)

…

Table3(id, value)

**User Request :**

* OPEN database\_name
* CLOSE
* SHOW
* DROP
* EXIST timeserie\_name
* GET timeserie\_name [ : variable\_name\_save]
* CREATE timeserie\_name Schema [ : variable\_name\_save]



**Timeseries(name, Schema, Path, fileSystem, mode)**

**User Request:**

* CREATE\_SCHEMA
* SHOW\_SCHEMA timeserie\_name
* GET\_SCHEMA timeserie\_name [ : variable\_name\_save ]
* SELECT\_RANGE timeserie\_name min\_timestamp max\_timestamp [ : variable\_name\_save ]
* MAX\_VALUE timeserie\_name
* MIN\_VALUE timeserie\_name
* MIN\_TIMESTAMP timeserie\_name
* MAX\_TIMESTAMP timeserie\_name

**Compression**

Indexing will work directly with storage 🡪 user will define

**User Request:**

* COMPRESSION parameter

**Indexing**

Indexing will work directly with storage 🡪 user will define

**User Request:**

* INDEXING parameter

**Transformation**

**User Request:**

* SQRT\_TRANSFORM timeserie\_name [ : variable\_name\_save]
* LOG\_TRANSFORM timeserie\_name [ : variable\_name\_save]
* MEAN timeserie\_name
* SCALE timeserie\_name [ : variable\_name\_save]
* STD\_DEVIATION timeserie\_name
* NORMALIZE timeserie\_name [ : variable\_name\_save]
* SEARCH pattern timeserie\_name
* MOVING\_AVERAGE timeserie\_name [ : variable\_name\_save]
* DTF timeserie\_name [ : variable\_name\_save]
* DTW timeserie\_name timeserie\_name

**Standard command**

**User Request:**

* QUIT
* HELP
* VAR
* DEL variable\_name
* RENAME variable\_name new\_name

**Forcasting (application)**

**User Request:**

* PERIODICITY timeserie\_name [ : variable\_name\_save]
* FORCAST timeserie\_name  [ : confidence\_intervals ] [ :: number\_of\_time\_units\_to\_forcast ]

**Clustering**

**User Request:**