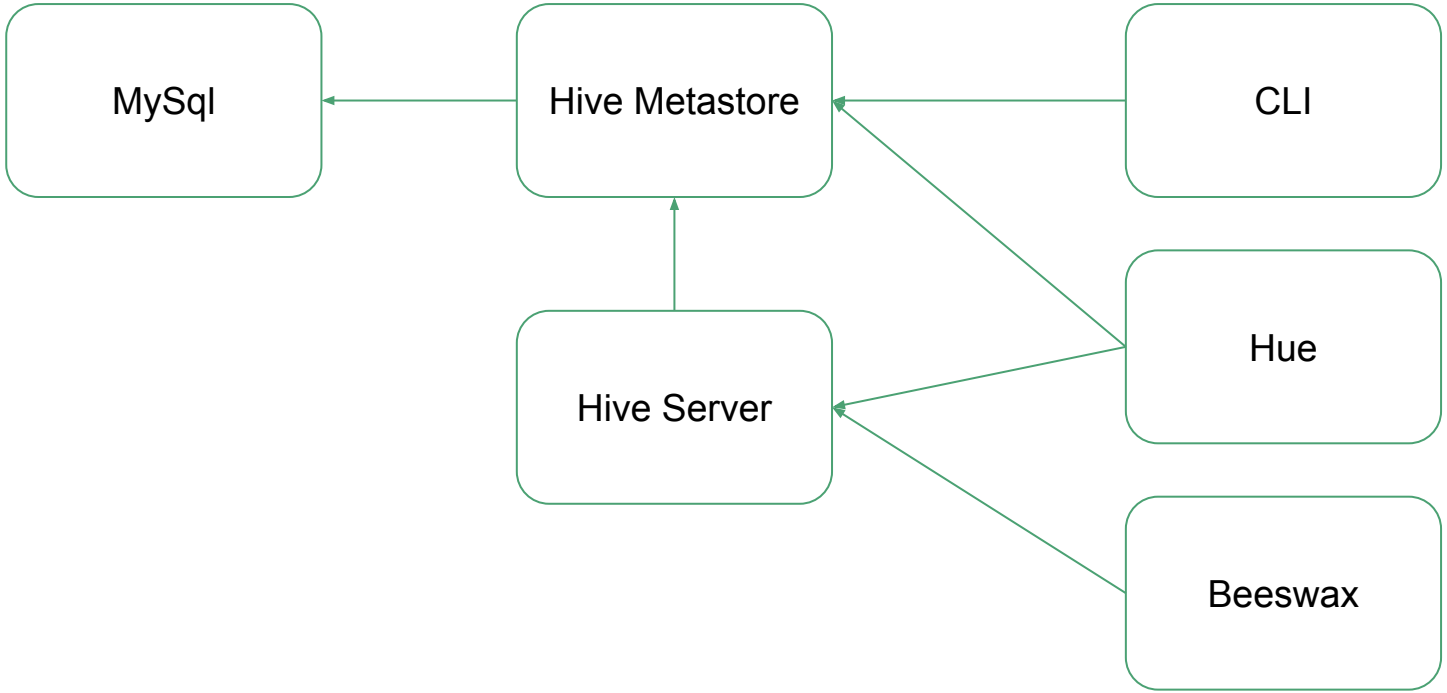
# Metody przetwarzania danych

Hive

# Agenda

1. Czym jest Hive?
2. Zarządzanie strukturami danych
3. HiveQL
4. Metody dostępu do danych (Część warsztatowa)
    a. Hue
    b. CLI
    c. JDBC

# Hive - Architektura



# Zarządzanie strukturami danych

```
CREATE TABLE <>;
```

```
CREATE external TABLE <>;
```

# Zarządzanie strukturami danych

```
CREATE TABLE example
(COLUMNS)
PARTITIONED BY v_date VARCHAR
    [ ROW FORMAT format || STORED AS <file_format> ]
LOCATION 'hdfs:///mydatalocation'
TBLPROPERTIES ('sensitive_data'='true');

CREATE TABLE x AS SELECT ... ;
```

# Zarządzanie strukturami danych

```
SET hive.execution.engine=tez;
SET hive.execution.engine=mr;
SET mapred.job.queue.name=my_queue;
SET tez.job.queue.name=my_queue;
SET hive.exec.dynamic.partition = TRUE;
SET hive.exec.dynamic.partition.mode = nonstrict;
SET mapred.output.compress=TRUE;
```

# Zarządzanie strukturami danych

```
CREATE TABLE example
(COLUMNS)
PARTITIONED BY v_date VARCHAR
LOCATION 'hdfs:///mydatalocation'
TBLPROPERTIES ('sensitive_data'='true');
```

```
ALTER TABLE example ADD PARTITION (v_date='2018-09-24') LOCATION
'hdfs:///mypartitiondata';
```

# Zarządzanie strukturami danych

```
DESCRIBE [extended] <table>;
SHOW DATABASES;
SHOW TABLES;
SHOW CREATE TABLE <table>;
MSCK [REPAIR] TABLE <table_name>; (swiat)
LOAD DATA ...;
CREATE TABLE <table>
[PARAMETERS]
AS
SELECT ... ;
```

# Hive QL

```
SELECT [ALL | DISTINCT] select_expr, select_expr, ...
  FROM table_reference
  [WHERE where_condition]
  [GROUP BY col_list]
  [ORDER BY col_list]
  [SORT BY col_list]
  [LIMIT [offset,] rows]
```

# Hive QL - typy danych

- TINYINT (1-byte signed integer, from -128 to 127)
- SMALLINT (2-byte signed integer, from -32,768 to 32,767)
- INT/INTEGER (4-byte signed integer, from -2,147,483,648 to 2,147,483,647)
- BIGINT (8-byte signed integer, from -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807)
- FLOAT (4-byte single precision floating point number)
- DOUBLE (8-byte double precision floating point number)
- DECIMAL
- TIMESTAMP
- DATE
- INTERVAL
- STRING
- arrays: ARRAY<data_type>
- maps: MAP<primitive_type, data_type>
- structs: STRUCT<col_name : data_type [COMMENT col_comment], ...>
- union: UNIONTYPE<data_type, data_type, ...>