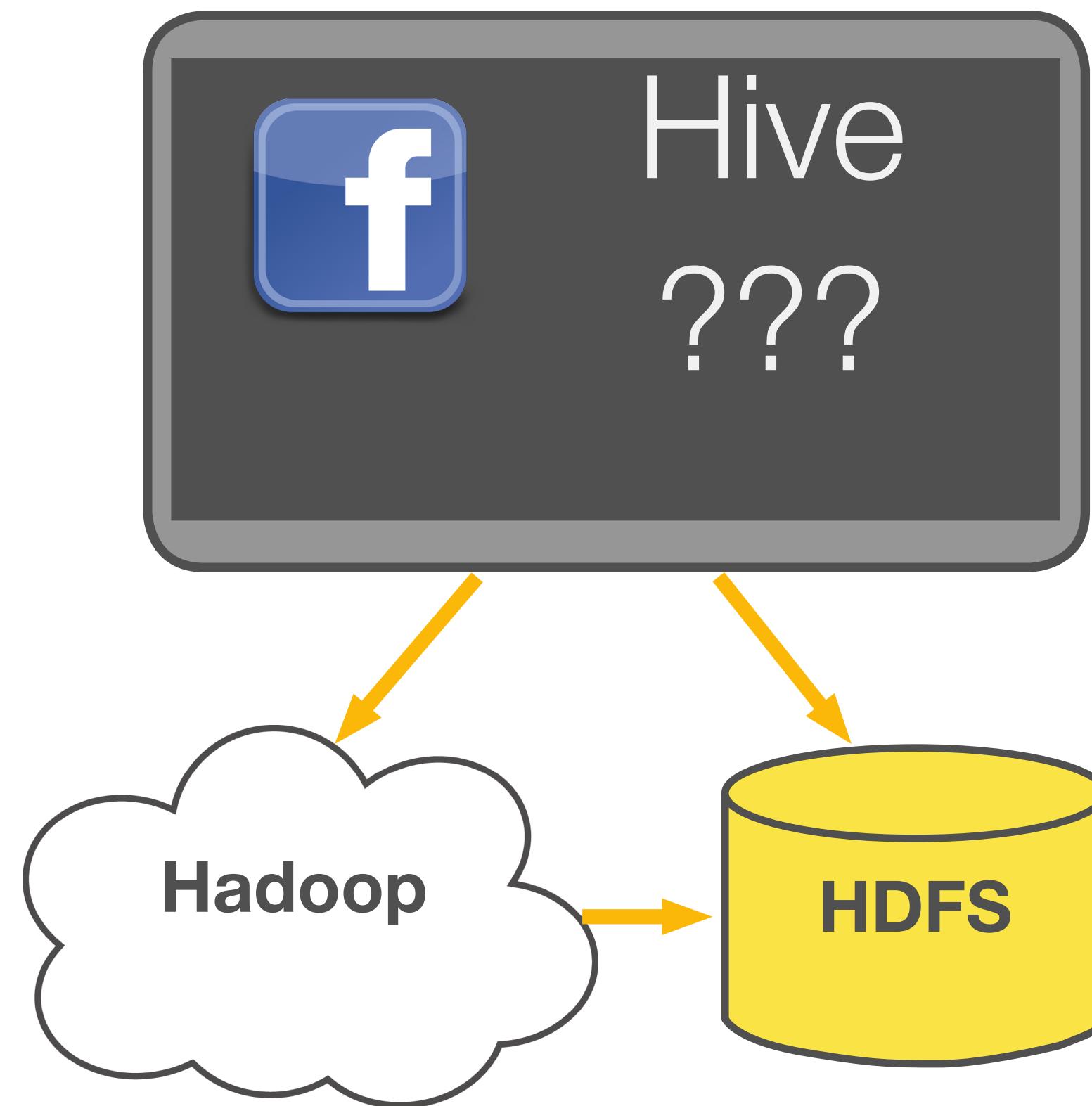
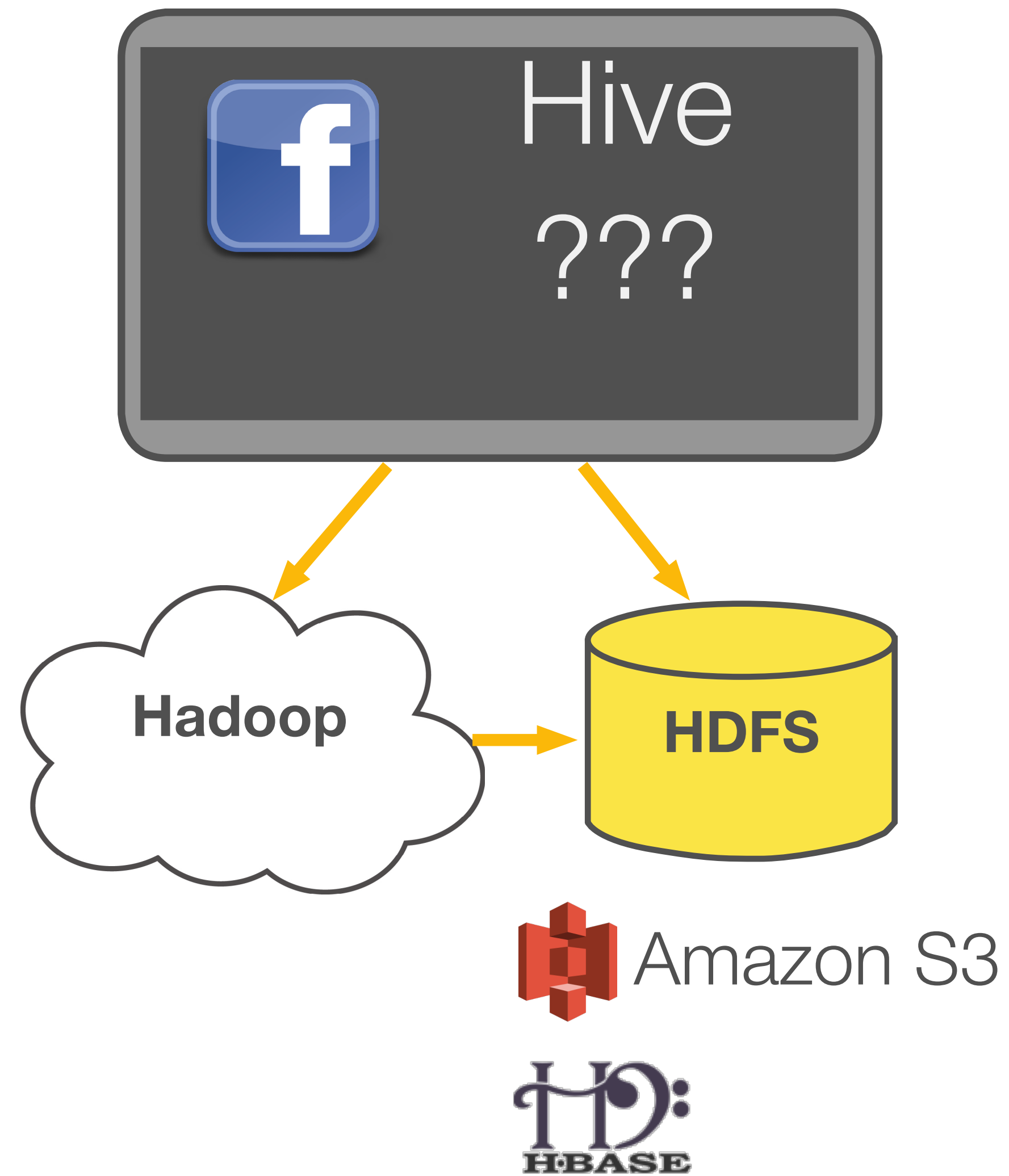
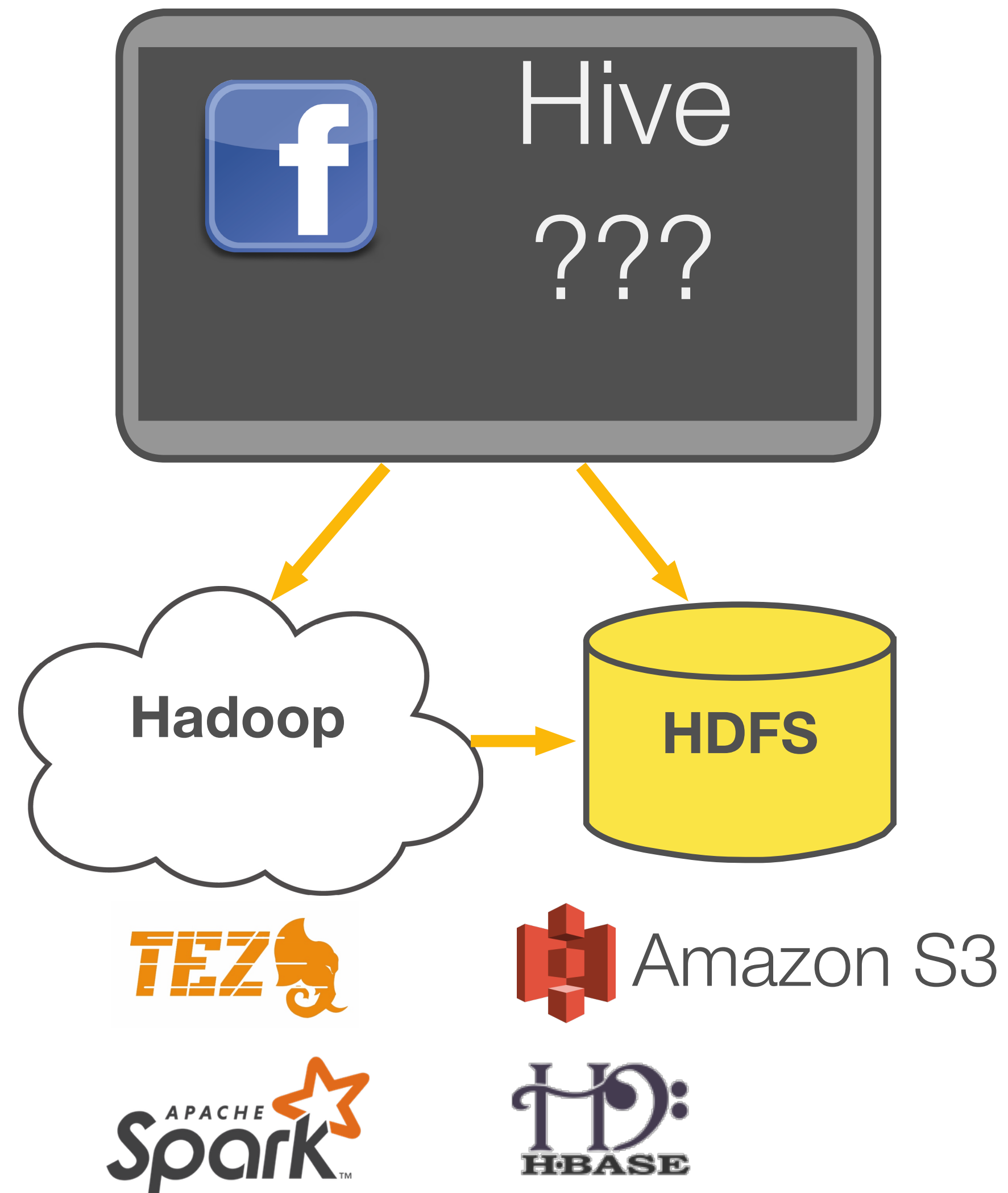


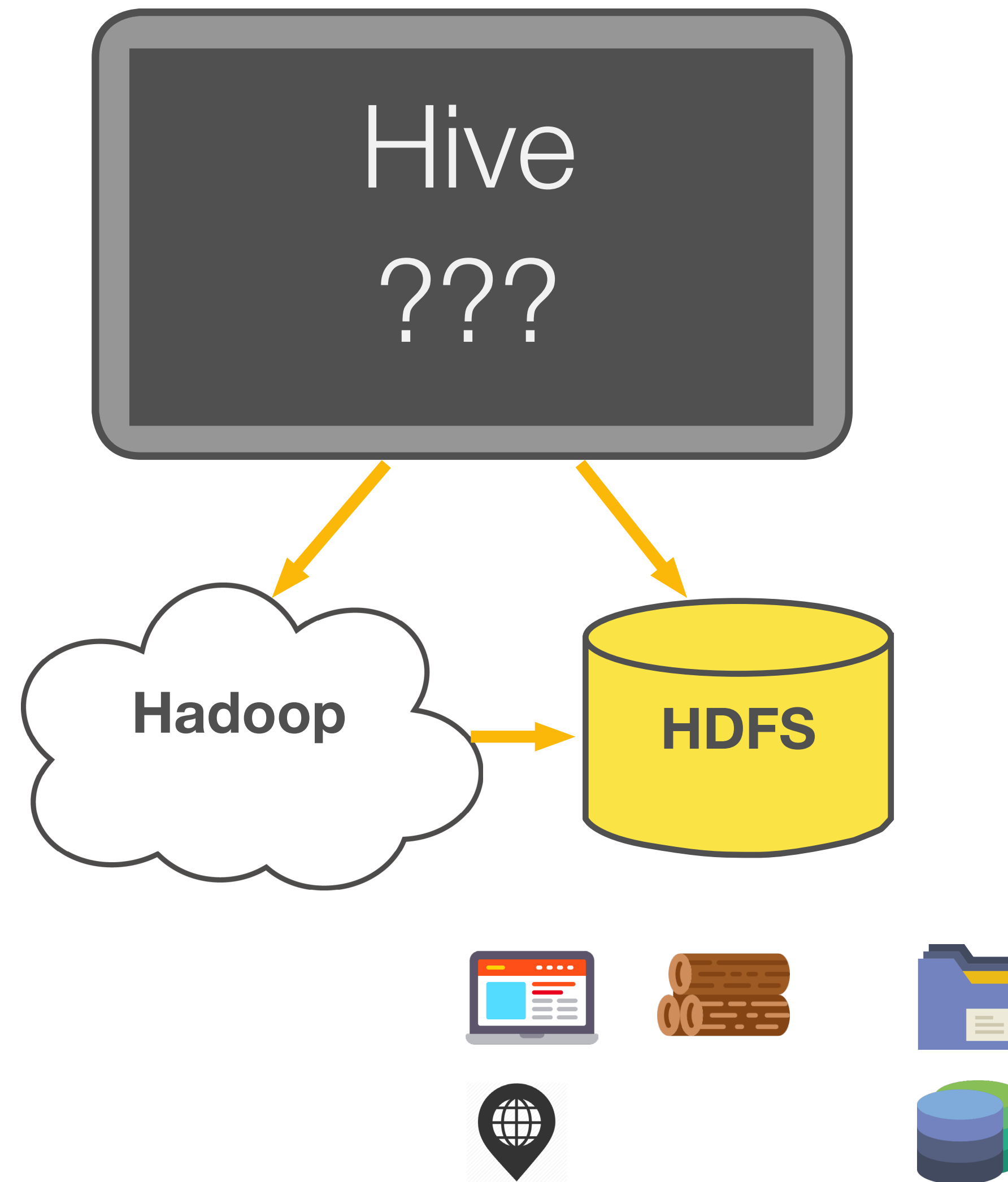
# SQL over BigData

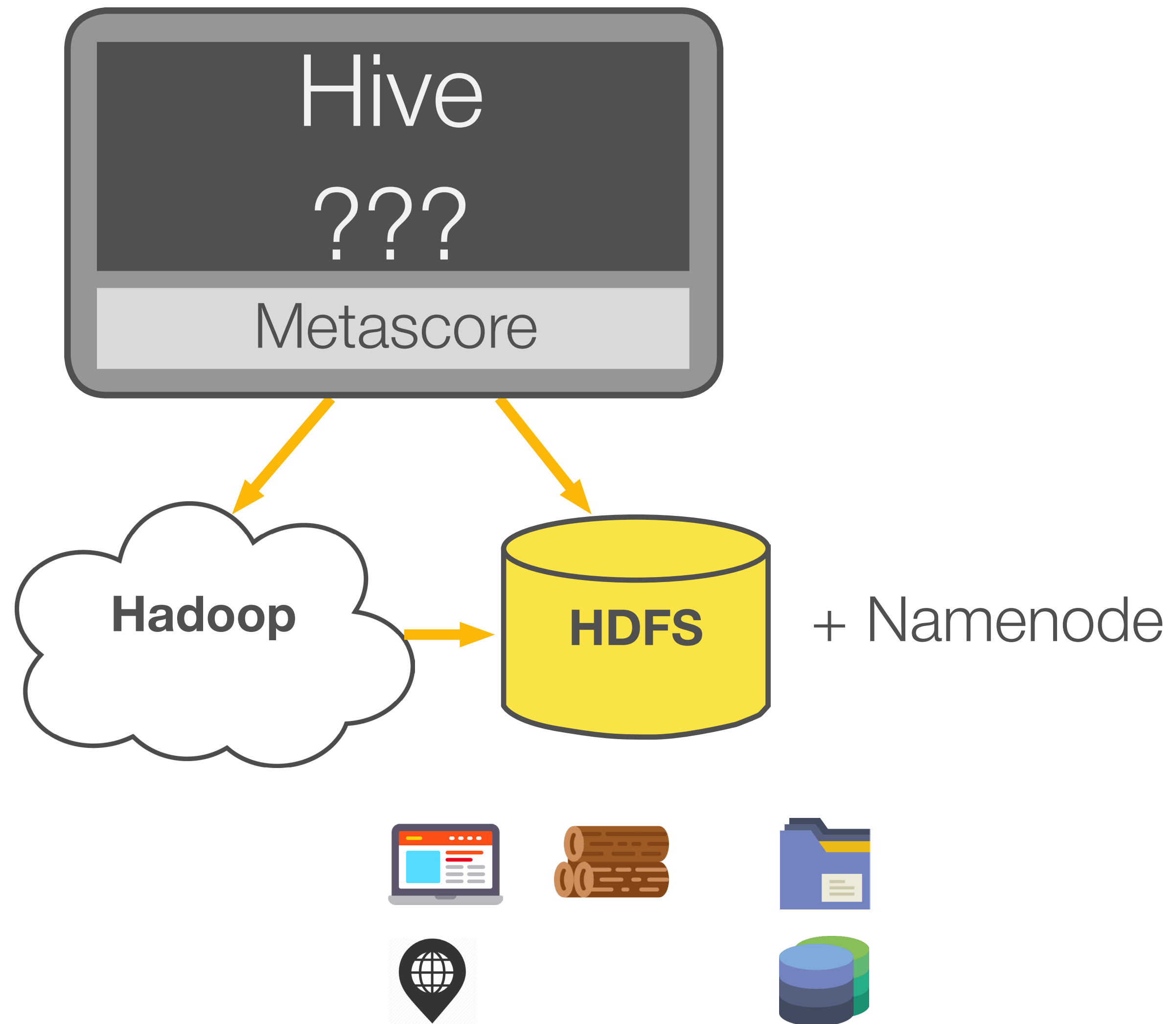
## Hive Data Definition Language (DDL)

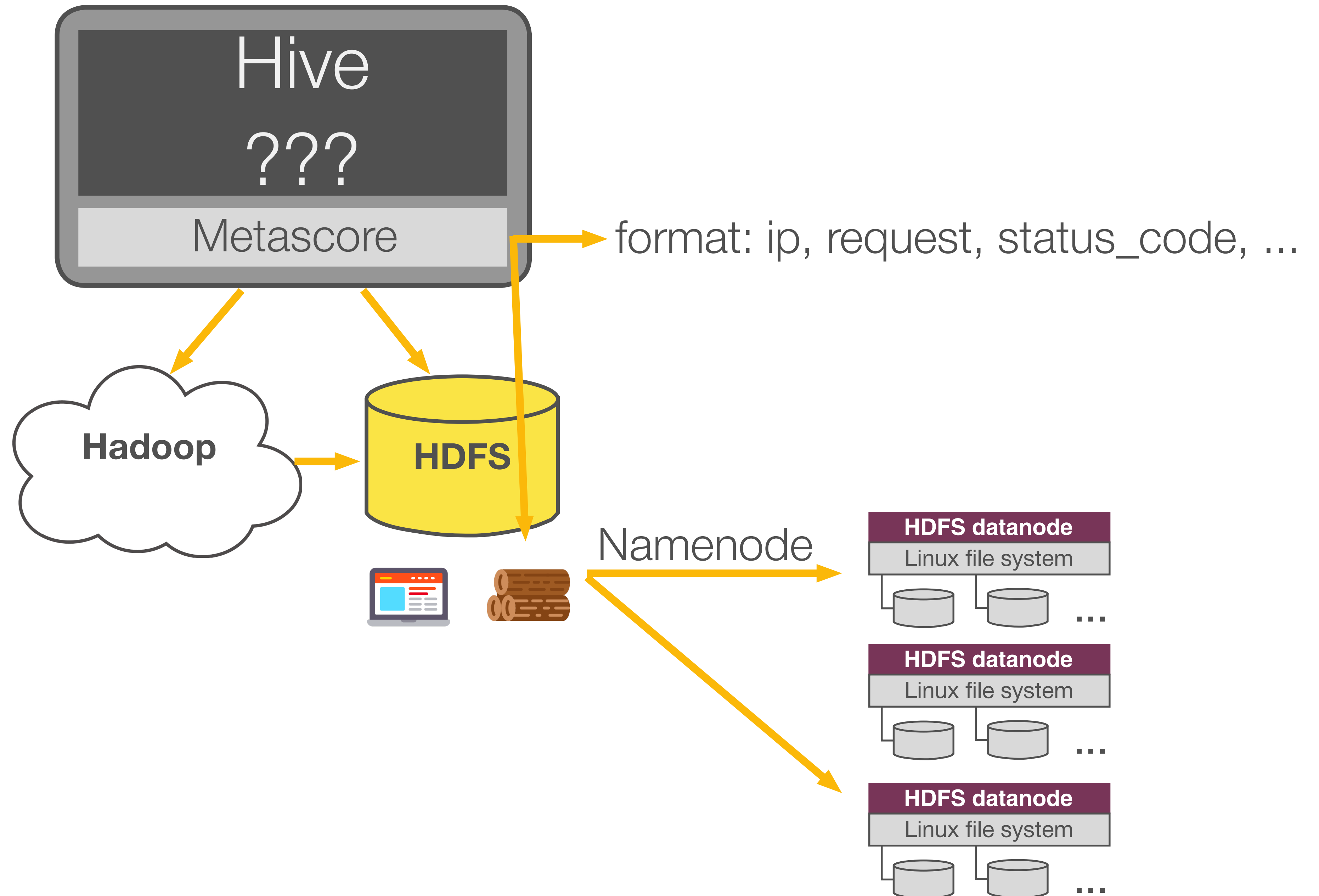


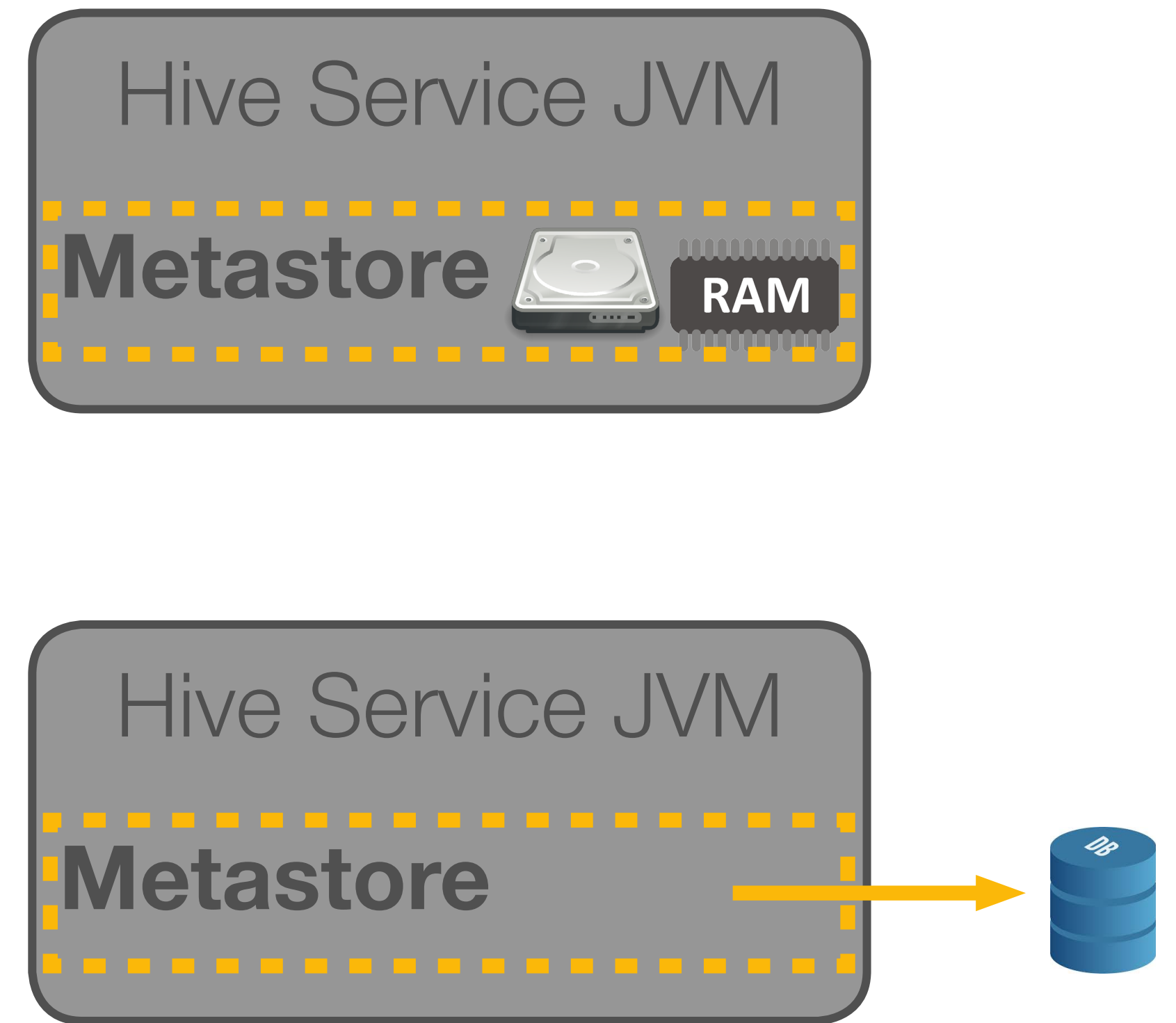
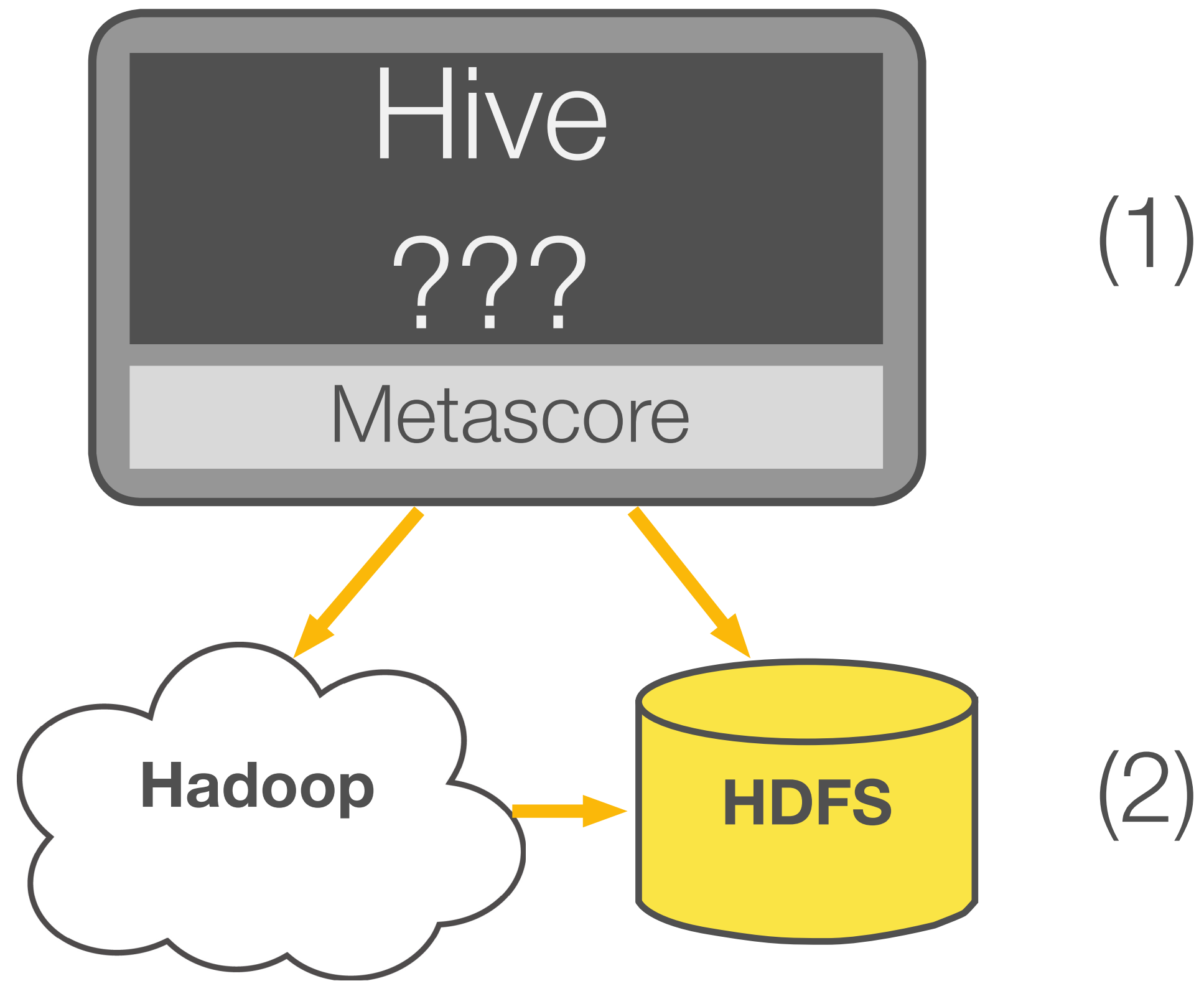














```
$ ls
```

```
a.txt
```

```
b.txt
```

```
...
```

```
$ hive
```

```
hive >
```

```
$ ls  
a.txt  
b.txt  
...  
$ hive  
hive >
```

```
CREATE TABLE my_table_name (  
    dummy_column STRING,  
    another_column STRING  
);
```

```
$ ls  
a.txt  
b.txt  
...  
$ hive  
hive >
```

```
CREATE TABLE [database_name.]my_table_name (  
    dummy_column STRING,  
    another_column STRING  
);
```



**"default"**

# DESCRIBE DATABASE default;

```
hive> describe database default;
OK
db_name comment location owner_name owner_type parameters
default Default Hive database hdfs://virtual-master.atp-fivt.org:8020/user/hive/warehouse public ROLE
Time taken: 0.018 seconds, Fetched: 1 row(s)
```

# DESCRIBE TABLE [EXTENDED|FORMATTED] table\_name;


```
hive> describe formatted tmp_table;
OK
col_name      data_type      comment
# col_name      data_type      comment

some_field      string

# Detailed Table Information
Database:      adral
Owner:         adral
CreateTime:    Fri Apr 28 16:52:55 CEST 2017
LastAccessTime: UNKNOWN
Protect Mode:  None
Retention:     0
Location:      hdfs://virtual-master.atp-fivt.org:8020/user/adral/warehouse/tmp_table
Table Type:    MANAGED_TABLE
Table Parameters:
```

## DESCRIBE DATABASE default;

```
hive> describe database default;
OK
db_name comment location owner_name owner_type parameters
default Default Hive database hdfs://virtual-master.atp-fivt.org:8020/user/hive/warehouse public ROLE
Time taken: 0.018 seconds, Fetched: 1 row(s)
```




## DESCRIBE TABLE [EXTENDED|FORMATTED] table\_name;

```
hive> describe formatted tmp_table;
OK
col_name      data_type      comment
# col_name      data_type      comment

some_field      string

# Detailed Table Information
Database:      adral
Owner:         adral
CreateTime:    Fri Apr 28 16:52:55 CEST 2017
LastAccessTime: UNKNOWN
Protect Mode:  None
Retention:     0
Location:      hdfs://virtual-master.atp-fivt.org:8020/user/adral/warehouse/tmp_table
Table Type:    MANAGED_TABLE
Table Parameters:
```



```
USE <database_name>;
```

```
CREATE TABLE <table_name> (  
    <column_name> <column_type>, ...  
)  
LOCATION "path/to/hdfs/location";
```

```
$ hdfs dfs -cat /user/adral/course2/week1/tab_dataset/*
```

```
first      line      1
```

```
second     line      3
```

```
last       line      5
```

```
USE adral;
```

```
CREATE TABLE tab_dataset (  
    first_column STRING,  
    second_column STRING,  
    value INT  
)
```

```
LOCATION '/user/adral/course2/week1/tab_dataset/';
```

```
$ hdfs dfs -cat /user/adral/course2/week1/tab_dataset/*
```

```
first    line    1
```

```
second   line    3
```

```
last     line    5
```

```
USE adral;
```

```
CREATE TABLE tab_dataset (  
    first_column STRING,  
    second_column STRING,  
    value INT  
)
```

```
LOCATION '/user/adral/course2/week1/tab_dataset/';
```

```
hive> select * from tab_dataset;
```

```
first    line    1          NULL    NULL
```

```
second   line    3          NULL    NULL
```

```
last     line    5          NULL    NULL
```



# Default Delimiters



<tab> or “\t”



???

# Default Delimiters



<tab> or “\t”



Ctrl-A or “^A” (ASCII)

```
USE adral;
```

```
DROP TABLE if exists tab_dataset;
```

```
CREATE TABLE tab_dataset (  
    first_column STRING,  
    second_column STRING,  
    value INT  
)
```

```
ROW FORMAT delimited
```

```
fields terminated by '\t'
```

```
LOCATION '/user/adral/course2/week1/tab_dataset/';
```

```
DROP TABLE if exists tab_dataset;
```

```
CREATE TABLE tab_dataset (  
    first_column STRING,  
    second_column STRING,  
    value INT  
)
```

```
ROW FORMAT delimited
```

```
fields terminated by '\t'
```

```
LOCATION '/user/adral/course2/week1/tab_dataset/';
```

```
hive> select * from tab_dataset;  
OK  
Time taken: 0.218 seconds
```

```
DROP TABLE if exists tab_dataset;
```

```
CREATE TABLE tab_dataset (  
    first_column STRING,  
    second_column STRING,  
    value INT  
)
```

```
ROW FORMAT delimited
```

```
    fields terminated by '\t'
```

```
LOCATION '/user/adral/course2/week1/tab_dataset/';
```

```
hive> select * from tab_dataset;
```

```
OK
```

```
Time taken: 0.218 seconds
```

```
$ hdfs dfs -ls /user/adral/course2/week1/tab_dataset/
```

```
<empty>
```

```
CREATE TABLE tab_dataset ...;  
DROP TABLE tab_dataset;
```

```
hive> describe formatted tmp_table;  
OK  
col_name      data_type      comment  
# col_name      data_type      comment  
  
some_field      string  
  
# Detailed Table Information  
Database:      adral  
Owner:         adral  
CreateTime:    Fri Apr 28 16:52:55 CEST 2017  
LastAccessTime: UNKNOWN  
Protect Mode:  None  
Retention:     0  
Location:      hdfs://virtual-master.atp-fivt.org:8020/user/adral/warehouse/tmp_table  
Table Type:    MANAGED_TABLE ←  
Table Parameters:
```

```
CREATE EXTERNAL TABLE tab_dataset ...;  
DROP TABLE tab_dataset;
```

```
hive> describe formatted tab_dataset;
```

```
OK
```

| # col_name    | data_type | comment |
|---------------|-----------|---------|
| first_column  | string    |         |
| second_column | string    |         |
| value         | int       |         |

```
# Detailed Table Information
```

|                 |                                                                              |
|-----------------|------------------------------------------------------------------------------|
| Database:       | adral                                                                        |
| Owner:          | adral                                                                        |
| CreateTime:     | Sat Apr 29 13:17:21 CEST 2017                                                |
| LastAccessTime: | UNKNOWN                                                                      |
| Protect Mode:   | None                                                                         |
| Retention:      | 0                                                                            |
| Location:       | hdfs://virtual-master.atp-fivt.org:8020/user/adral/course2/week1/tab_dataset |
| Table Type:     | EXTERNAL_TABLE ←                                                             |

```
CREATE EXTERNAL TABLE tab_dataset ...;  
DROP TABLE tab_dataset;
```

```
hive> describe formatted tab_dataset;  
OK  
# col_name          data_type          comment  
  
first_column        string  
second_column        string  
value               int  
  
# Detailed Table Information  
Database:           adral  
Owner:              adral  
CreateTime:         Sat Apr 29 13:17:21 CEST 2017  
LastAccessTime:     UNKNOWN  
Protect Mode:       None  
Retention:          0  
Location:           hdfs://virtual-master.atp-fivt.org:8020/user/adral/course2/week1/tab_dataset  
Table Type:         EXTERNAL_TABLE ←
```

```
CREATE TEMPORARY [EXTERNAL] TABLE tab_dataset ...;  
DROP TABLE tab_dataset;
```



```
USE adral;
```

```
DROP TABLE if exists tab_dataset;
```

```
CREATE EXTERNAL TABLE tab_dataset (  
    first_column STRING,  
    second_column STRING,  
    value INT  
)
```

```
ROW FORMAT delimited
```

```
    fields terminated by '\t'
```

```
LOCATION '/user/adral/course2/week1/tab_dataset/';
```

```
hive> select * from tab_dataset;
```

```
OK
```

```
first      line      1
```

```
second     line      3
```

```
last       line      5
```

```
Time taken: 0.157 seconds, Fetched: 3 row(s)
```

```
CREATE EXTERNAL TABLE tab_dataset (  
    first_column    STRING,  
    second_column   STRING,  
    value           INT  
)  
ROW FORMAT DELIMITED  
    FIELDS TERMINATED BY '\001'  
    COLLECTION ITEMS TERMINATED BY '\002'  
    MAP KEYS TERMINATED BY '\003'  
    LINES TERMINATED BY '\n'  
LOCATION '/user/adra1/course2/week1/tab_dataset/';
```

```
CREATE EXTERNAL TABLE tab_dataset (  
    first_column    STRING,  
    second_column   STRING,  
    value           INT  
)  
ROW FORMAT DELIMITED  
    FIELDS TERMINATED BY '\001'  
    COLLECTION ITEMS TERMINATED BY '\002' ← '^B'  
    MAP KEYS TERMINATED BY '\003' ← '^C'  
    LINES TERMINATED BY '\n'  
LOCATION '/user/adra1/course2/week1/tab_dataset/';
```

```
CREATE EXTERNAL TABLE tab_dataset (  
    first_column    STRING,  
    second_column   STRING,  
    value           INT  
)  
ROW FORMAT DELIMITED  
    FIELDS TERMINATED BY '\001'  
    COLLECTION ITEMS TERMINATED BY '\002' ← '^B'  
    MAP KEYS TERMINATED BY '\003' ← '^C'  
    LINES TERMINATED BY '\n'  
LOCATION '/user/adra1/course2/week1/tab_dataset/';
```

```
CREATE EXTERNAL TABLE tab_dataset (  
  first_column    STRING,  
  second_column  STRING,  
  value           INT  
)
```

# Primitive

| Category  | Type      | Description                                                                                   | Literal examples                              |
|-----------|-----------|-----------------------------------------------------------------------------------------------|-----------------------------------------------|
| Primitive | BOOLEAN   | True/false value                                                                              | TRUE                                          |
|           | TINYINT   | 1-byte (8-bit signed integer, from -128 to 127.                                               | 1Y                                            |
|           | SMALLINT  | 2-byte (16-bit) signed integer, from -32,768 to 32,767.                                       | 1S                                            |
|           | INT       | 4-byte (32-bit) signed integer, from -2,147,483,648 to 2,147,483,647.                         | 1                                             |
|           | BIGINT    | 8-byte (64-bit) signed integer, from -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807. | 1L                                            |
|           | FLOAT     | 4-byte (32-bit) single-precision floating-point number.                                       | 1.0                                           |
|           | DOUBLE    | 8-byte (64-bit) double-precision floating-point nubur                                         | 1.0                                           |
|           | DECIMAL   | Arbitrary-precision signed decimal number.                                                    | 1.0                                           |
|           | STRING    | Unbounded variable-length character string.                                                   | 'a', "a"                                      |
|           | VARCHAR   | Variable-length character string.                                                             | 'a', "a"                                      |
|           | CHAR      | Fixed-length character string.                                                                | 'a', "a"                                      |
|           | BINARY    | Byte array.                                                                                   | Not supported                                 |
|           | TIMESTAMP | Timestamp with nanosecond precision                                                           | 1325502245000, '2012-01-02 03:04:05.123456789 |
|           | DATE      | Date.                                                                                         | '2012-01-02'                                  |

```
CREATE EXTERNAL TABLE tab_dataset (  
  first_column    STRING,  
  second_column  STRING,  
  value           INT  
)
```

## Primitive

| Category  | Type      | Description                                                                                   | Literal examples                              |
|-----------|-----------|-----------------------------------------------------------------------------------------------|-----------------------------------------------|
| Primitive | BOOLEAN   | True/false value                                                                              | TRUE                                          |
|           | TINYINT   | 1-byte (8-bit signed integer, from -128 to 127.                                               | 1Y                                            |
|           | SMALLINT  | 2-byte (16-bit) signed integer, from -32,768 to 32,767.                                       | 1S                                            |
|           | INT       | 4-byte (32-bit) signed integer, from -2,147,483,648 to 2,147,483,647.                         | 1                                             |
|           | BIGINT    | 8-byte (64-bit) signed integer, from -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807. | 1L                                            |
|           | FLOAT     | 4-byte (32-bit) single-precision floating-point number.                                       | 1.0                                           |
|           | DOUBLE    | 8-byte (64-bit) double-precision floating-point nubur                                         | 1.0                                           |
|           | DECIMAL   | Arbitrary-precision signed decimal number.                                                    | 1.0                                           |
|           | STRING    | Unbounded variable-length character string.                                                   | 'a', "a"                                      |
|           | VARCHAR   | Variable-length character string.                                                             | 'a', "a"                                      |
|           | CHAR      | Fixed-length character string.                                                                | 'a', "a"                                      |
|           | BINARY    | Byte array.                                                                                   | Not supported                                 |
|           | TIMESTAMP | Timestamp with nanosecond precision                                                           | 1325502245000, '2012-01-02 03:04:05.123456789 |
|           | DATE      | Date.                                                                                         | '2012-01-02'                                  |

## Complex

| Category | Type   | Description                                                                                                                                                                              | Literal examples                                                       |
|----------|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------|
| Complex  | ARRAY  | An ordered collection of fieldss. The fields must all be of the same type.                                                                                                               | array(1, 2)                                                            |
|          | MAP    | An unordered collection of key-value pairs. Keys must be primitives; values may be any type. For a particular map, the keys must be the same type, and the values must be the same type. | map('a', 1, 'b', 2)                                                    |
|          | STRUCT | A collection of named fields. The fields may be of different types.                                                                                                                      | struct('a', 1, 1.0), named_struct('col1', 'a', 'col2', 1, 'col3', 1.0) |
|          | UNION  | A value that may be one of a number of defined data types. The alue is tagged with an integer (zero-indexed) representing its data type in the union.                                    | create_union(1, 'a', 63)                                               |

```
CREATE TABLE employees (  
    name          STRING,  
    salary        FLOAT,  
    subordinates  ARRAY<STRING>,  
    deductions    MAP<STRING, FLOAT>,  
    address       STRUCT<street:STRING, city:STRING, state:STRING, zip:INT>);
```

```
John Doe^A100000.0^AMary Smith^BTodd Jones^AFederal Taxes^C.2^BState  
Taxes^C.05^BInsurance^C.1^A1 Michigan Ave.^BChicago^BIL^B60600
```

```
Mary Smith^A80000.0^ABill King^AFederal Taxes^C.2^BState  
Taxes^C.05^BInsurance^C.1^A100 Ontario St.^BChicago^BIL^B60601
```

```
CREATE TABLE employees (
  name      STRING,
  salary    FLOAT,
  subordinates ARRAY<STRING>,
  deductions MAP<STRING, FLOAT>,
  address   STRUCT<street:STRING, city:STRING, state:STRING, zip:INT>);
```

```
John Doe^A100000.0^AMary Smith^BTodd Jones^AFederal Taxes^C.2^BState
Taxes^C.05^BInsurance^C.1^A1 Michigan Ave.^BChicago^BIL^B60600
```

```
Mary Smith^A80000.0^ABill King^AFederal Taxes^C.2^BState
Taxes^C.05^BInsurance^C.1^A100 Ontario St.^BChicago^BIL^B60601
```



```
CREATE TABLE employees (
  name      STRING,
  salary    FLOAT,
  subordinates ARRAY<STRING>,
  deductions MAP<STRING, FLOAT>,
  address   STRUCT<street:STRING, city:STRING, state:STRING, zip:INT>);
```

```
John Doe^A100000.0^AMary Smith^BTodd Jones^AFederal Taxes^C.2^BState
Taxes^C.05^BInsurance^C.1^A1 Michigan Ave.^BChicago^BIL^B60600
```

```
Mary Smith^A80000.0^ABill King^AFederal Taxes^C.2^BState
Taxes^C.05^BInsurance^C.1^A100 Ontario St.^BChicago^BIL^B60601
```

```
CREATE TABLE employees (
    name          STRING,
    salary        FLOAT,
    subordinates  ARRAY<STRING>,
    deductions    MAP<STRING, FLOAT>,
    address       STRUCT<street:STRING, city:STRING, state:STRING, zip:INT>);
```

```
John Doe^A100000.0^AMary Smith^BTodd Jones^AFederal Taxes^C.2^BState
Taxes^C.05^BInsurance^C.1^A1 Michigan Ave.^BChicago^BIL^B60600
```

```
Mary Smith^A80000.0^ABill King^AFederal Taxes^C.2^BState
Taxes^C.05^BInsurance^C.1^A100 Ontario St.^BChicago^BIL^B60601
```

```
CREATE TABLE employees (
    name          STRING,
    salary        FLOAT,
    subordinates  ARRAY<STRING>,
    deductions    MAP<STRING, FLOAT>,
    address       STRUCT<street:STRING, city:STRING, state:STRING, zip:INT>);
```

```
John Doe^A100000.0^AMary Smith^BTodd Jones^AFederal Taxes^C.2^BState
Taxes^C.05^BInsurance^C.1^A1 Michigan Ave.^BChicago^BIL^B60600
```

```
Mary Smith^A80000.0^ABill King^AFederal Taxes^C.2^BState
Taxes^C.05^BInsurance^C.1^A100 Ontario St.^BChicago^BIL^B60601
```

```
CREATE TABLE employees (
  name          STRING,
  salary        FLOAT,
  subordinates  ARRAY<STRING>,
  deductions    MAP<STRING, FLOAT>,
  address       STRUCT<street:STRING, city:STRING, state:STRING, zip:INT>);
```

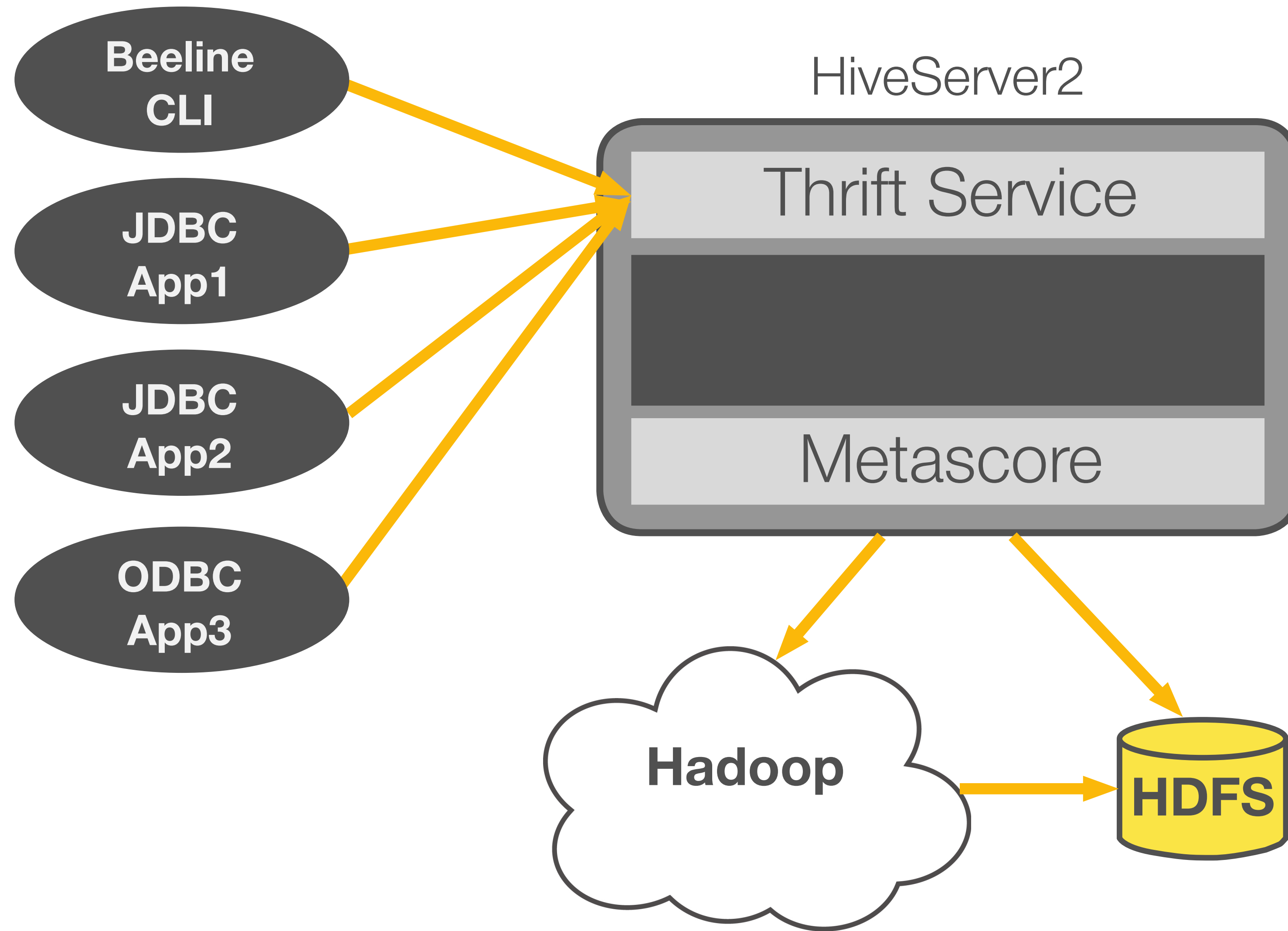
```
John Doe^A100000.0^AMary Smith^BTodd Jones^AFederal Taxes^C.2^BState
Taxes^C.05^BInsurance^C.1^A1 Michigan Ave.^BChicago^BIL^B60600
```

```
Mary Smith^A80000.0^ABill King^AFederal Taxes^C.2^BState
Taxes^C.05^BInsurance^C.1^A100 Ontario St.^BChicago^BIL^B60601
```

```
CREATE EXTERNAL TABLE tab_dataset (  
    first_column    STRING,  
    second_column  STRING,  
    value           INT  
)  
ROW FORMAT DELIMITED  
    FIELDS TERMINATED BY '\001'  
    COLLECTION ITEMS TERMINATED BY '\002'  
    MAP KEYS TERMINATED BY '\003'  
    LINES TERMINATED BY '\n'  
STORED AS file_format  
LOCATION '/user/adral/course2/week1/tab_dataset/';
```

```
CREATE EXTERNAL TABLE tab_dataset (  
    first_column    STRING,  
    second_column  STRING,  
    value           INT  
)  
ROW FORMAT DELIMITED  
    FIELDS TERMINATED BY '\001'  
    COLLECTION ITEMS TERMINATED BY '\002'  
    MAP KEYS TERMINATED BY '\003'  
    LINES TERMINATED BY '\n'  
STORED AS file_format  
LOCATION '/user/adra1/course2/week1/tab_dataset/';
```

default: **TEXTFILE**



# Summary



# Summary

- you can **configure** Hive metadata for data in HDFS

# Summary

- you can **configure** Hive metadata for data in HDFS
- you can **list** Hive service APIs

# Summary

- you can **configure** Hive metadata for data in HDFS
- you can **list** Hive service APIs
- you can **use** “explain” to get information about Hive tables

# Summary

- you can **configure** Hive metadata for data in HDFS
- you can **list** Hive service APIs
- you can **use** “explain” to get information about Hive tables
- you can **explain** how Hive serializes and deserializes row format fields (see: Hive primitive and complex data types)

# Summary

- you can **configure** Hive metadata for data in HDFS
- you can **list** Hive service APIs
- you can **use** “explain” to get information about Hive tables
- you can **explain** how Hive serializes and deserializes row format fields (see: Hive primitive and complex data types)

see: <https://cwiki.apache.org/confluence/display/Hive/LanguageManual+DDL>