Introducing Spark SQL

Advantages of Spark SQL

The coolest way of big data processing

Why we need to learn it

Bank transaction

sender id

recipient id

transaction date

amount

Bank transaction	Access log
sender id	visitor id
recipient id	visited page
transaction date	visit date
amount	

Bank transaction

sender id

recipient id

transaction date

amount

Access log

visitor id

visited page

visit date

SQL

SQL


HIVE


HIVE


hadoop

10 Gb/s

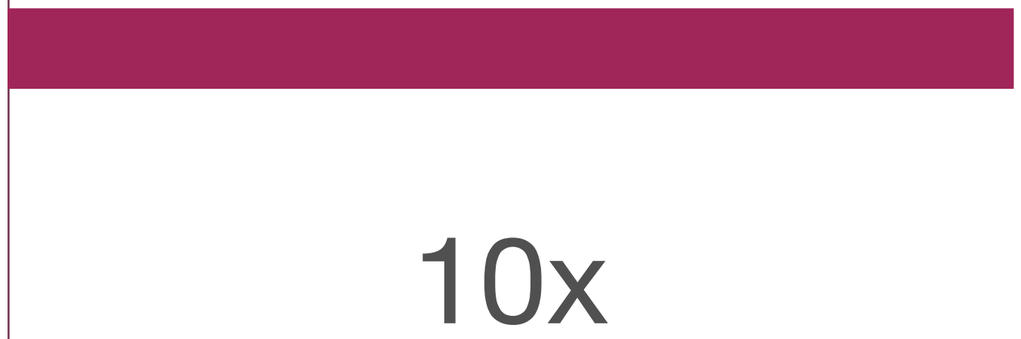
100 Mb/s


HIVE


APACHE
Spark
TM


Spark


hadoop


10x

Spark RDD API

```
rdd
rdd1 = rdd.map(lambda x: x.split("\t"))
rdd2 = rdd1.map(lambda x: (x[0], x[2]))
```

VS

SQL on Spark

```
select user_id, url from access_log
```

Spark RDD API

```
rdd
rdd1 = rdd.map(lambda x: x.split("\t"))
rdd2 = rdd1.map(lambda x: (x[0], x[2]))
```

VS

SQL on Spark

```
select user_id, url from access_log
```

- No data parsing

Spark RDD API

```
rdd
rdd1 = rdd.map(lambda x: x.split("\t"))
rdd2 = rdd1.map(lambda x: (x[0], x[2]))
```

VS

SQL on Spark

```
select user_id, url from access_log
```

- No data parsing
- Code optimization

Spark RDD API

```
rdd
rdd1 = rdd.map(lambda x: x.split("\t"))
rdd2 = rdd1.map(lambda x: (x[0], x[2]))
```

VS

SQL on Spark

```
select user_id, url from access_log
```

- No data parsing
- Code optimization
- Syntax is easier

Spark RDD API

```
rdd
rdd1 = rdd.map(lambda x: x.split("\t"))
rdd2 = rdd1.map(lambda x: (x[0], x[2]))
```

VS

SQL on Spark

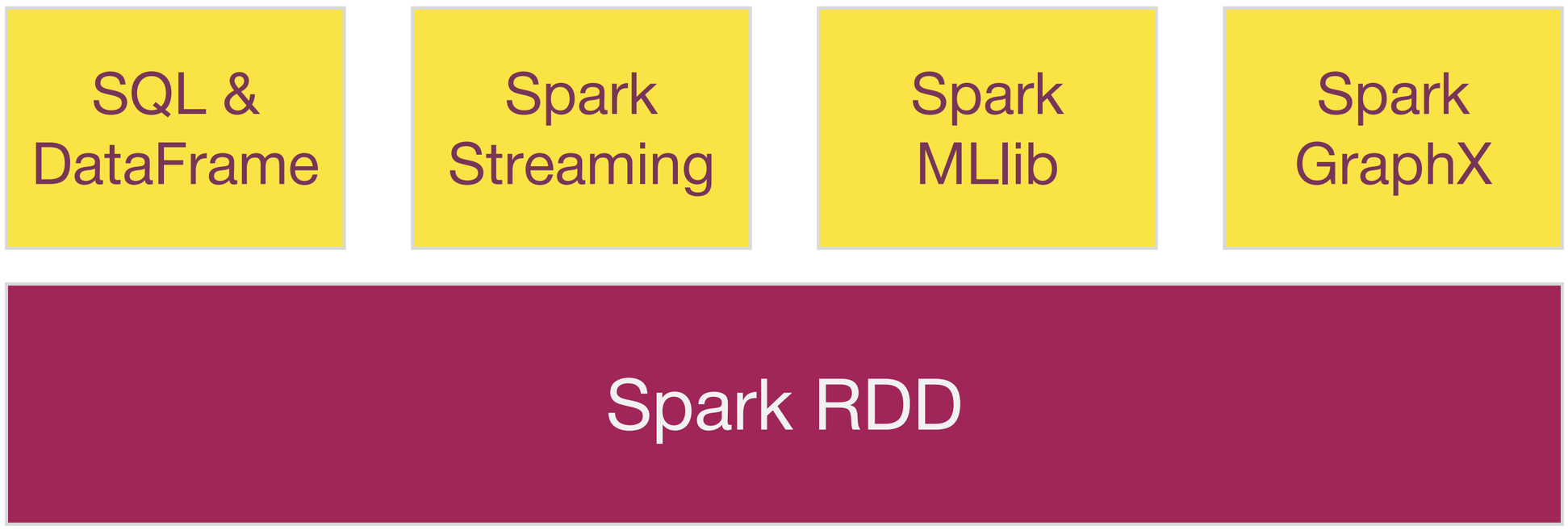
```
select user_id, url from access_log
```

- No data parsing
- Code optimization
- Syntax is easier
- No overhead

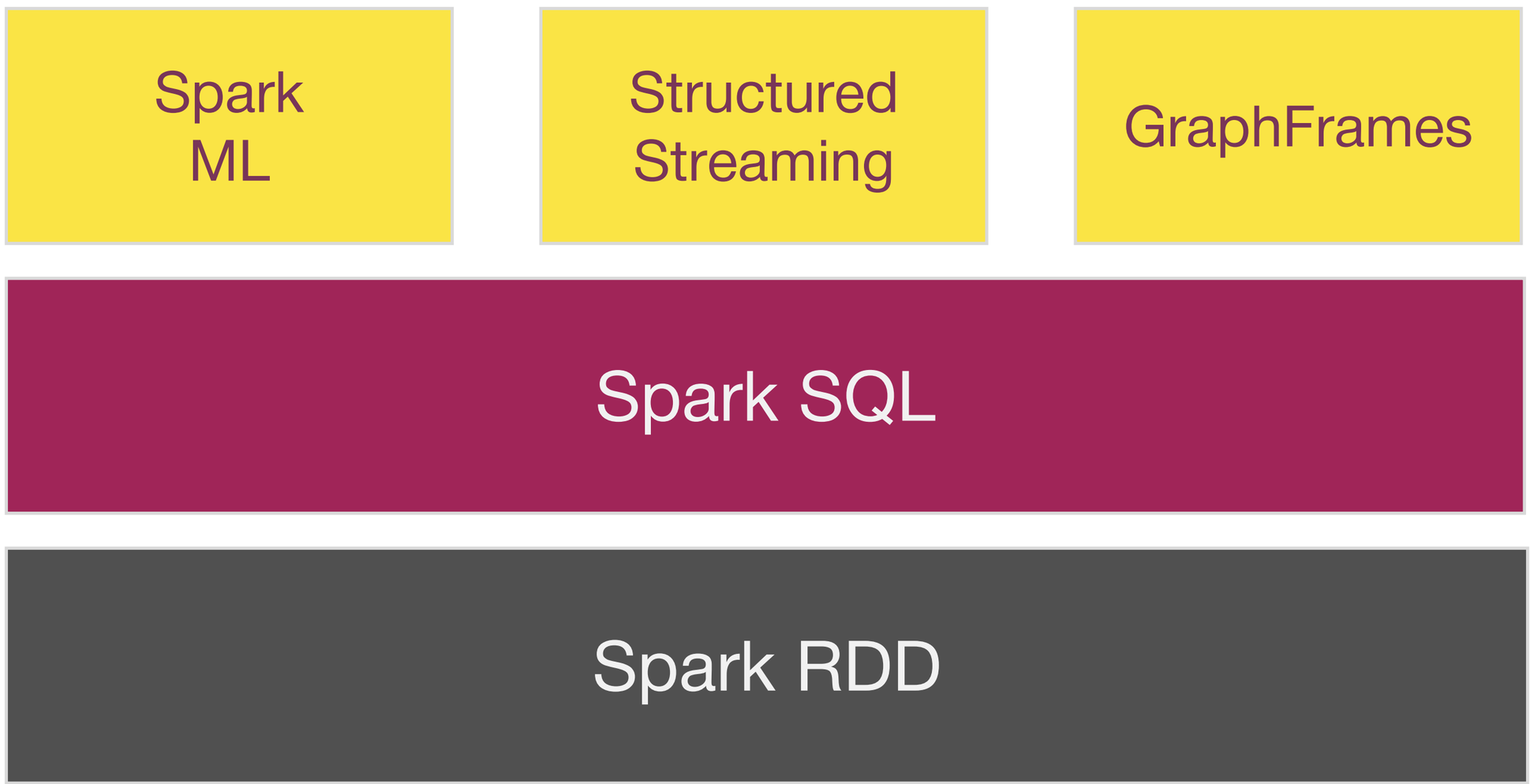

Hive
Structured files
External Databases
APACHE
Spark
TM
Hive
Structured files
External Databases

SQL and DataFrame

Spark 1.0


SQL &
DataFrame
Spark
Streaming
Spark
MLlib
Spark
GraphX
Spark RDD

Spark 2.0


Spark
ML
Structured
Streaming
GraphFrames
Spark SQL
Spark RDD

Spark SQL is...

- ... like hive but faster
- ... faster and more easy than spark RDD
- ... allows to read/write data from any sources
- ... a new core API in spark 2.0