ADVANCING DATABRICKS

Understanding Spark to build next-gen ETL systems

Microsoft Partner
Microsoft
Gold Data Analytics
Gold Data Platform
Silver Cloud Platform

databricks

Sponsors

Track

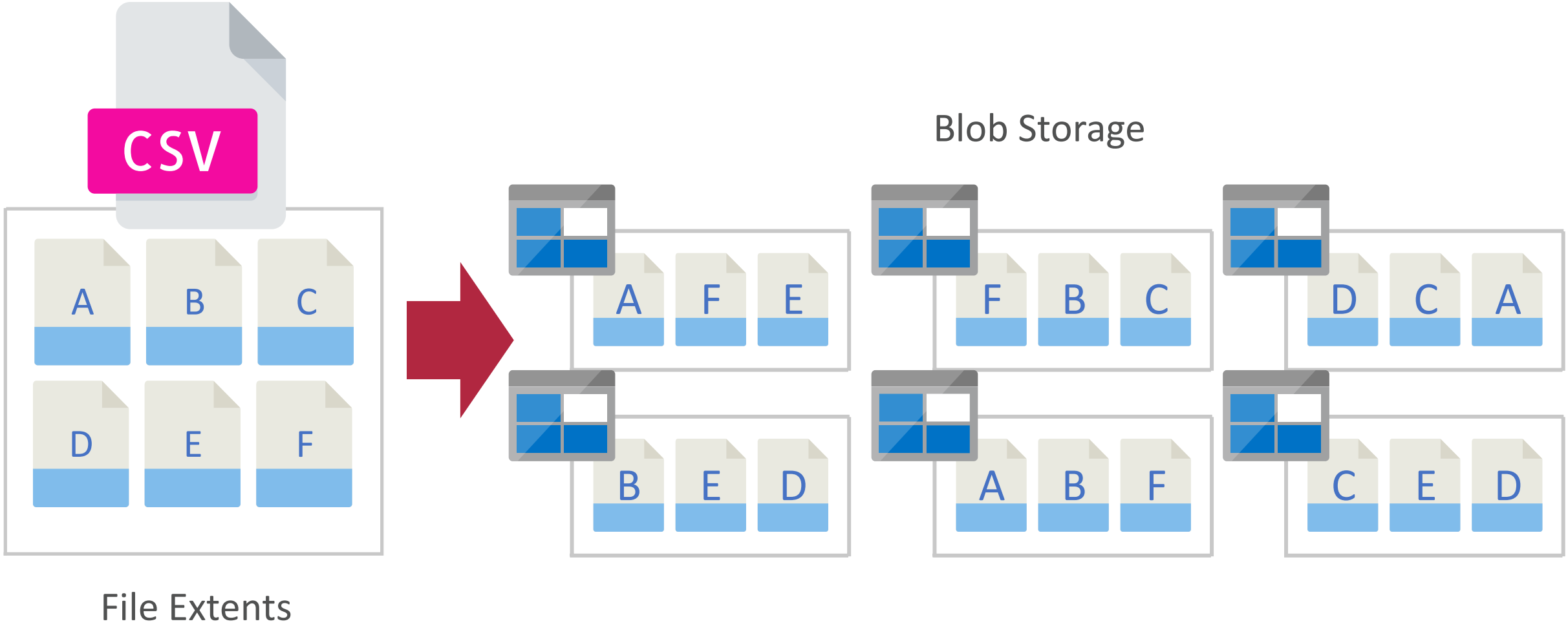
Event

HTTPS://GITHUB.COM/SIWHITELEY/DATABRICKSETL

Google File System Papers Released

2003

HDFS IN AZURE

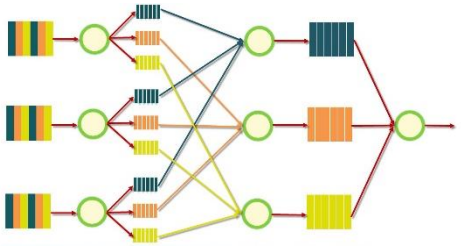


Google File System Papers Released

2003

Google MapReduce Papers

2004



ADVANCING ANALYTICS

Input

This is a large document

This might come from a log file

Or it might come from your user input

Split

This is a large document

This might come from a log file

Or it might come from your user input

Map

This, 1
is, 1
a, 1
large, 1
document, 1

This, 1
might, 1
come, 1
from, 1
a, 1
log, 1
file, 1

Or, 1
it, 1
might, 1
come, 1
from, 1
your, 1
user, 1
input, 1

Shuffle

This (1,1)

is (1)

a (1,1)

large (1)

document (1)

might (1,1)

come (1,1)

from (1,1)

log (1)

Or (1)

file (1)

it (1)

your (1)

user (1)

input (1)

Reduce

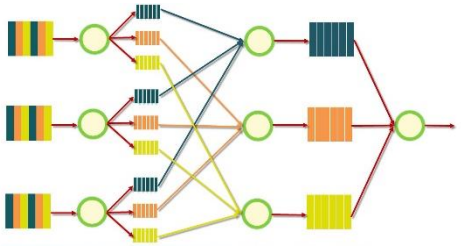
This, 2
might, 2
come, 2
from, 2
a, 2
is, 1
large, 1
document, 1
log, 1
Or, 1
file, 1
it, 1
your, 1
user, 1
input, 1

Google File System Papers Released

2003

Google MapReduce Papers

2004



2006

Apache Hadoop project created

Matei Zaharia starts Spark project

2009

Project donated to Apache Foundation

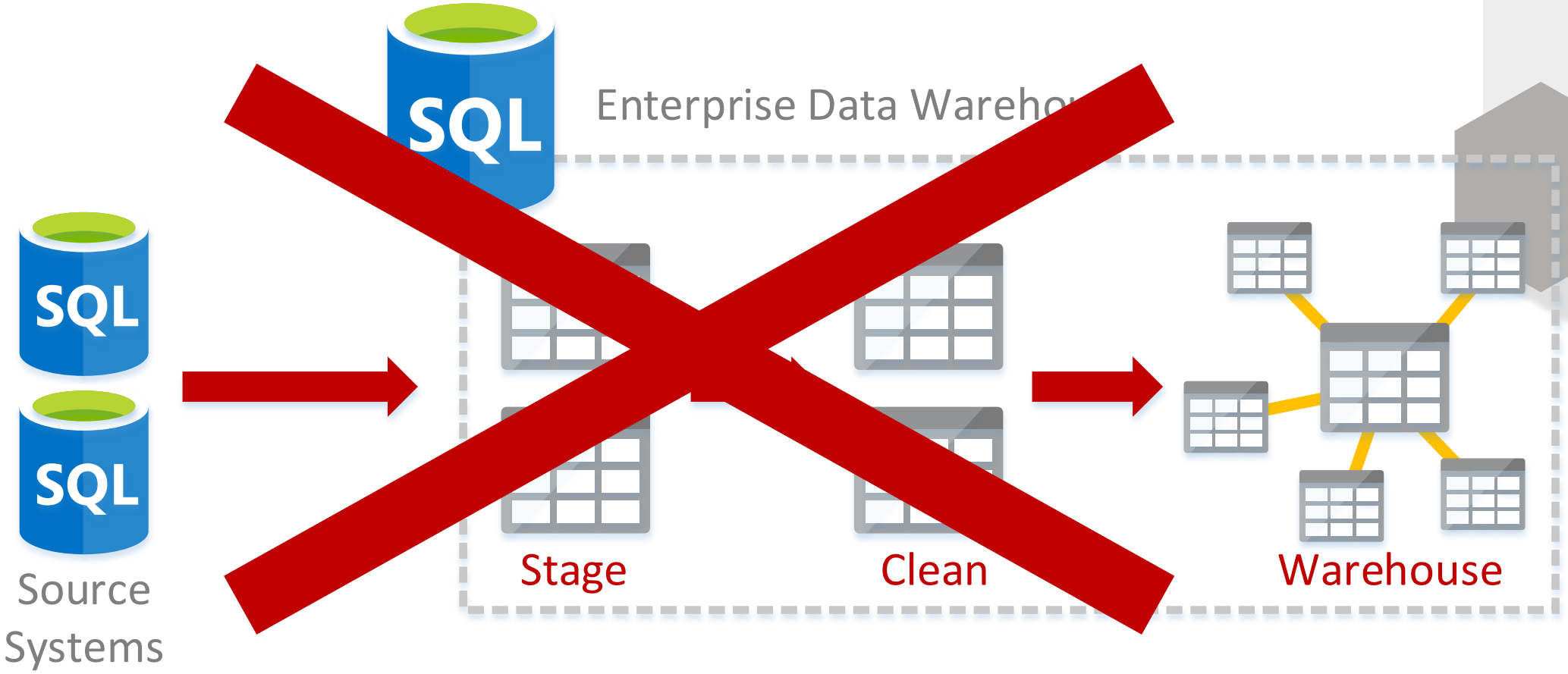
2013

Databricks founded by Matei & UC Berkley Colleagues

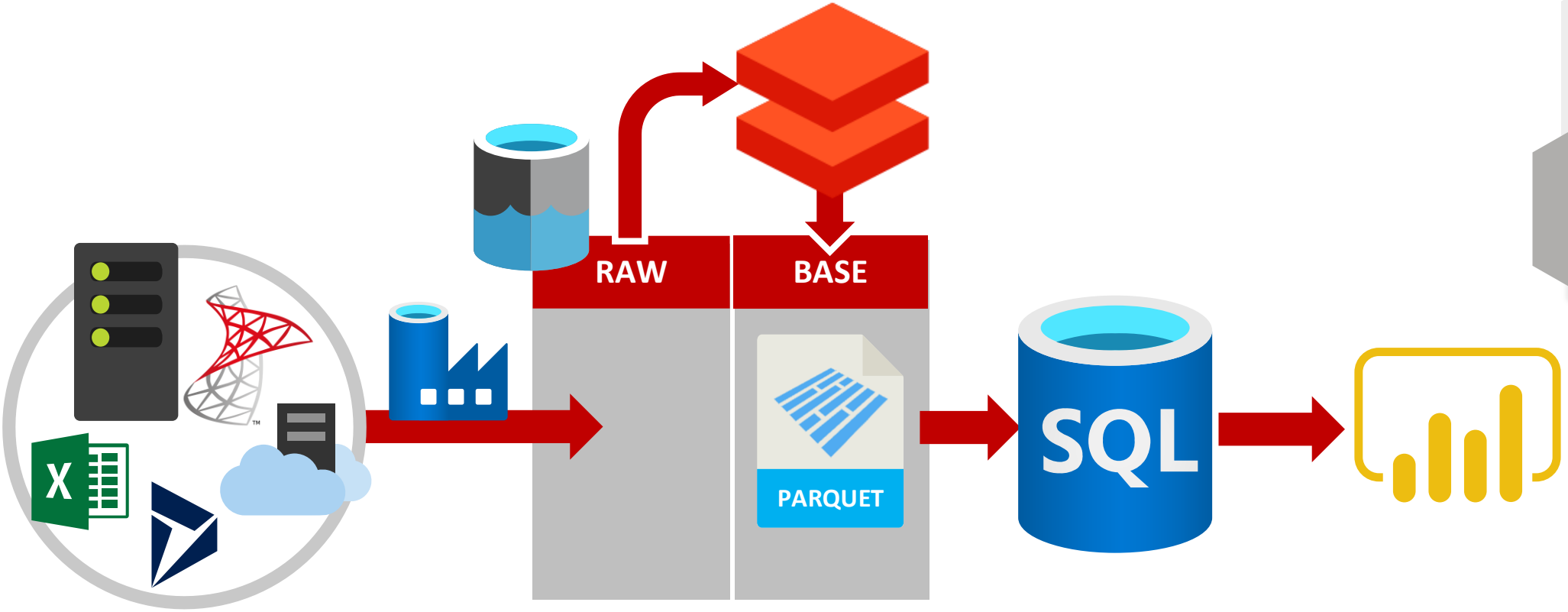
2013


JSON
jupyter
Language Flexibility
SQL
R
SQL
SQL
Integrations

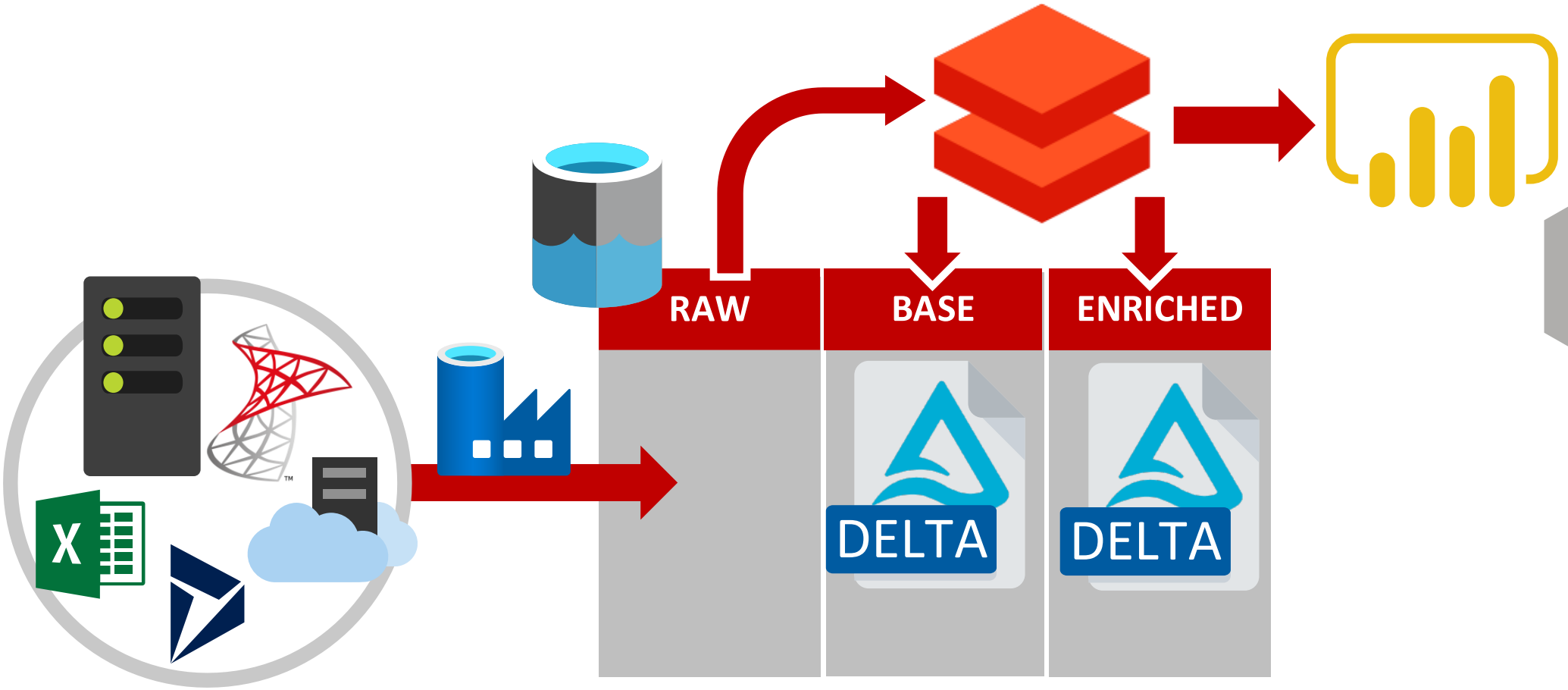
BUT MOST OF ALL…



MODERN DATA WAREHOUSE

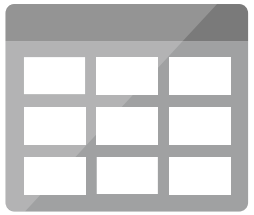


DATA LAKEHOUSE



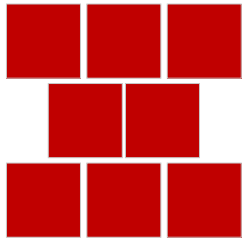
SPARK INTERNALS

UNDER THE HOOD



DataFrame API

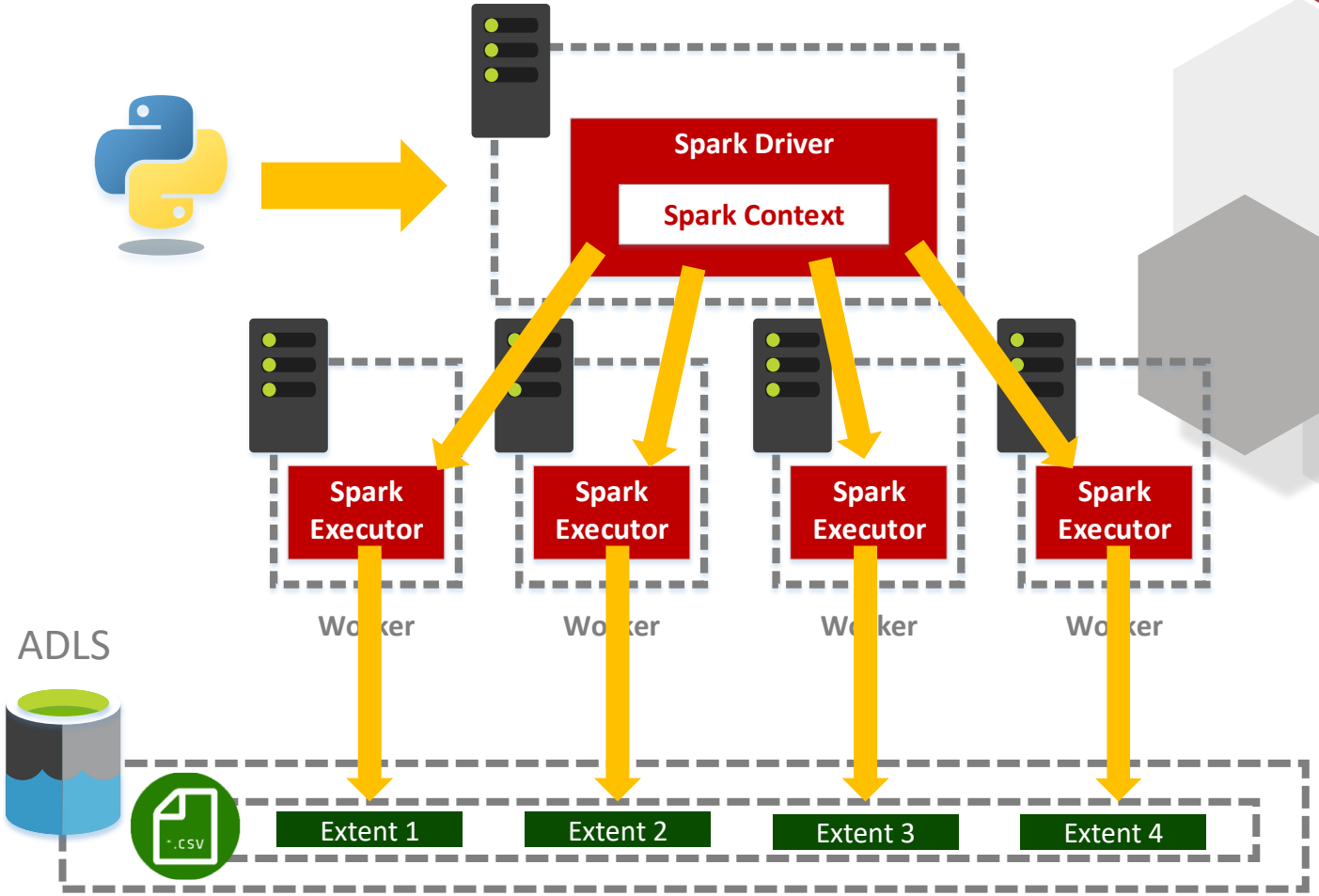
SQL API



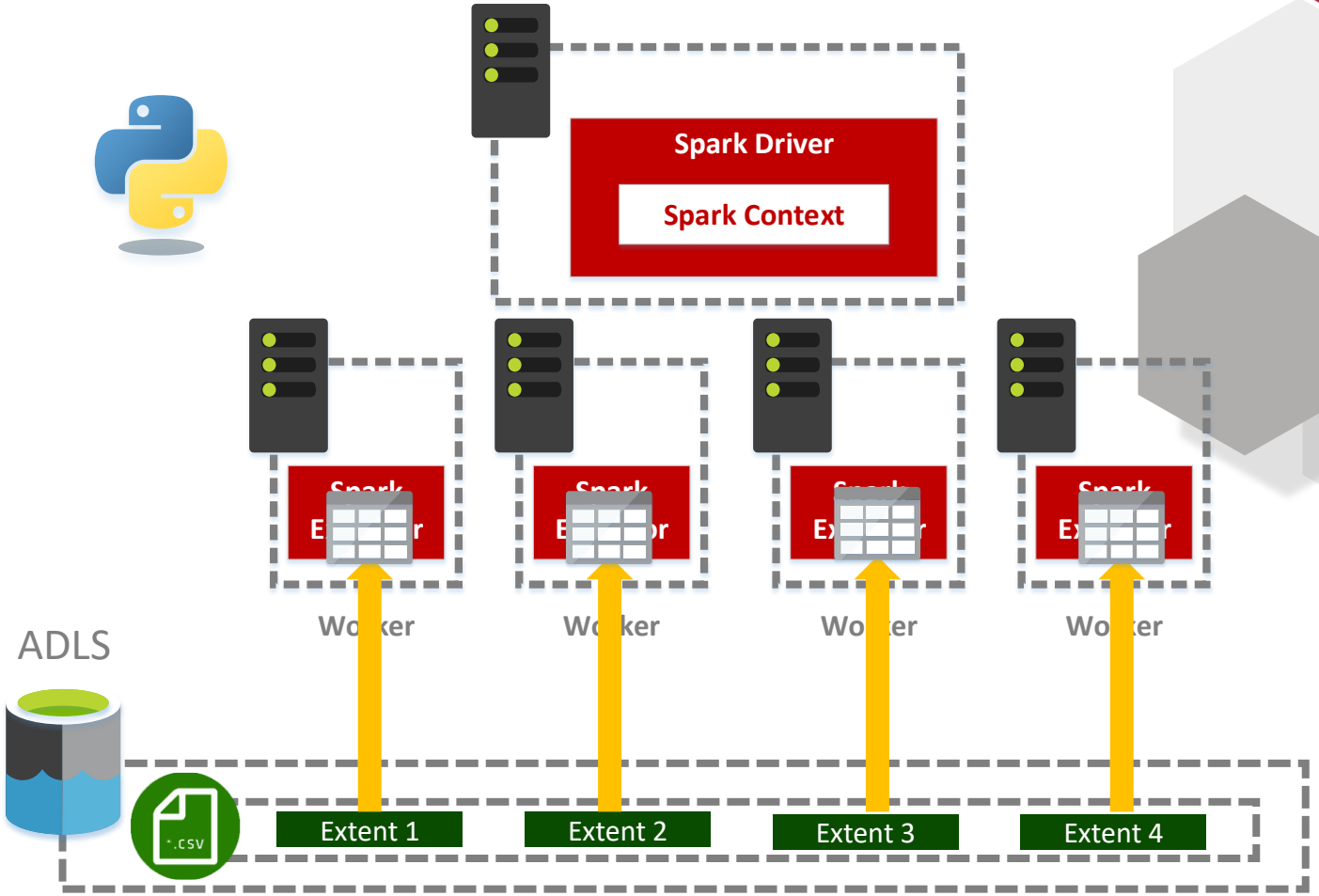
Resilient Distributed Datasets – In-Memory Data Blocks

Core Spark Engine – 80% Scala Code Libraries

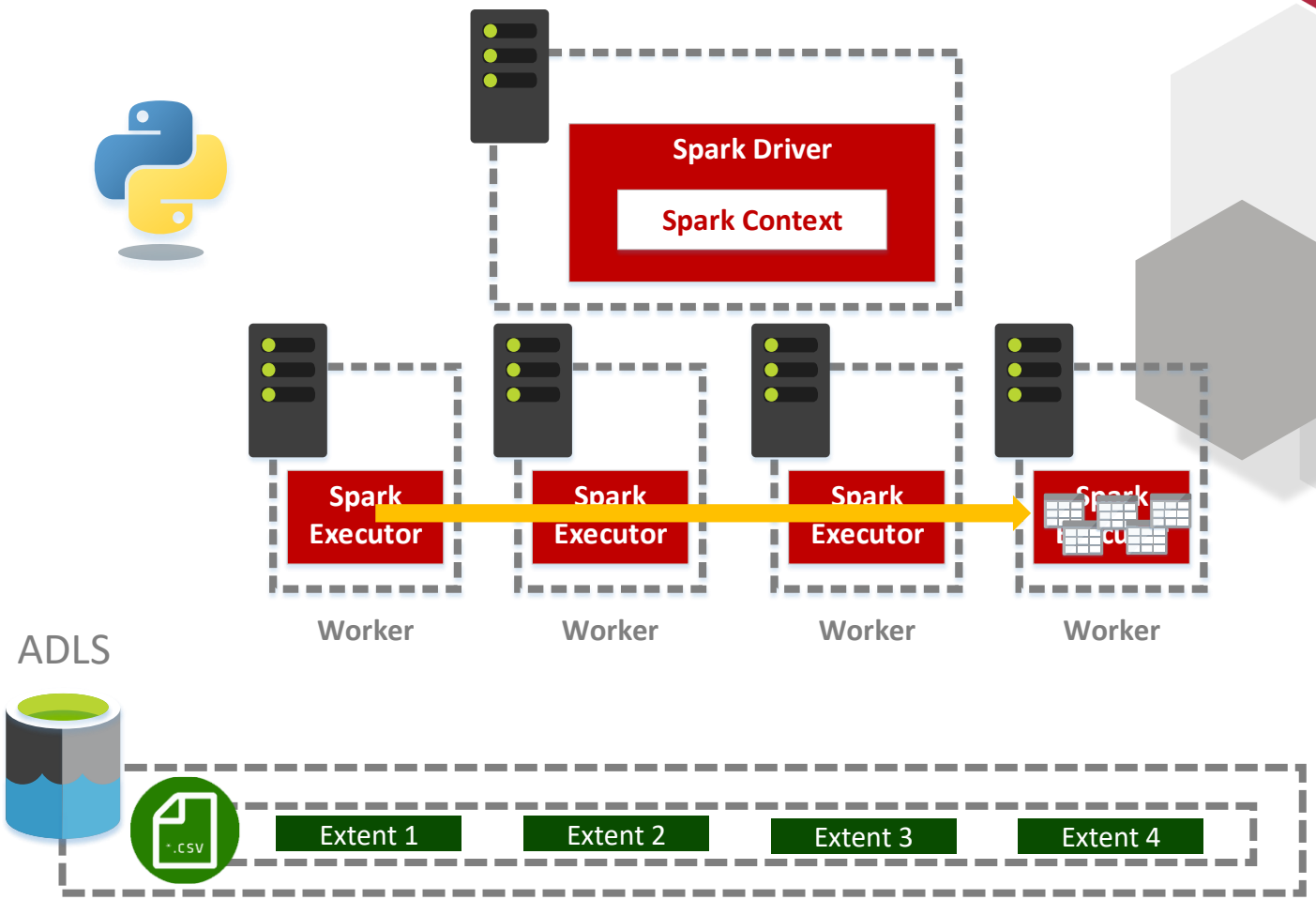
DISTRIBUTED COMPUTE



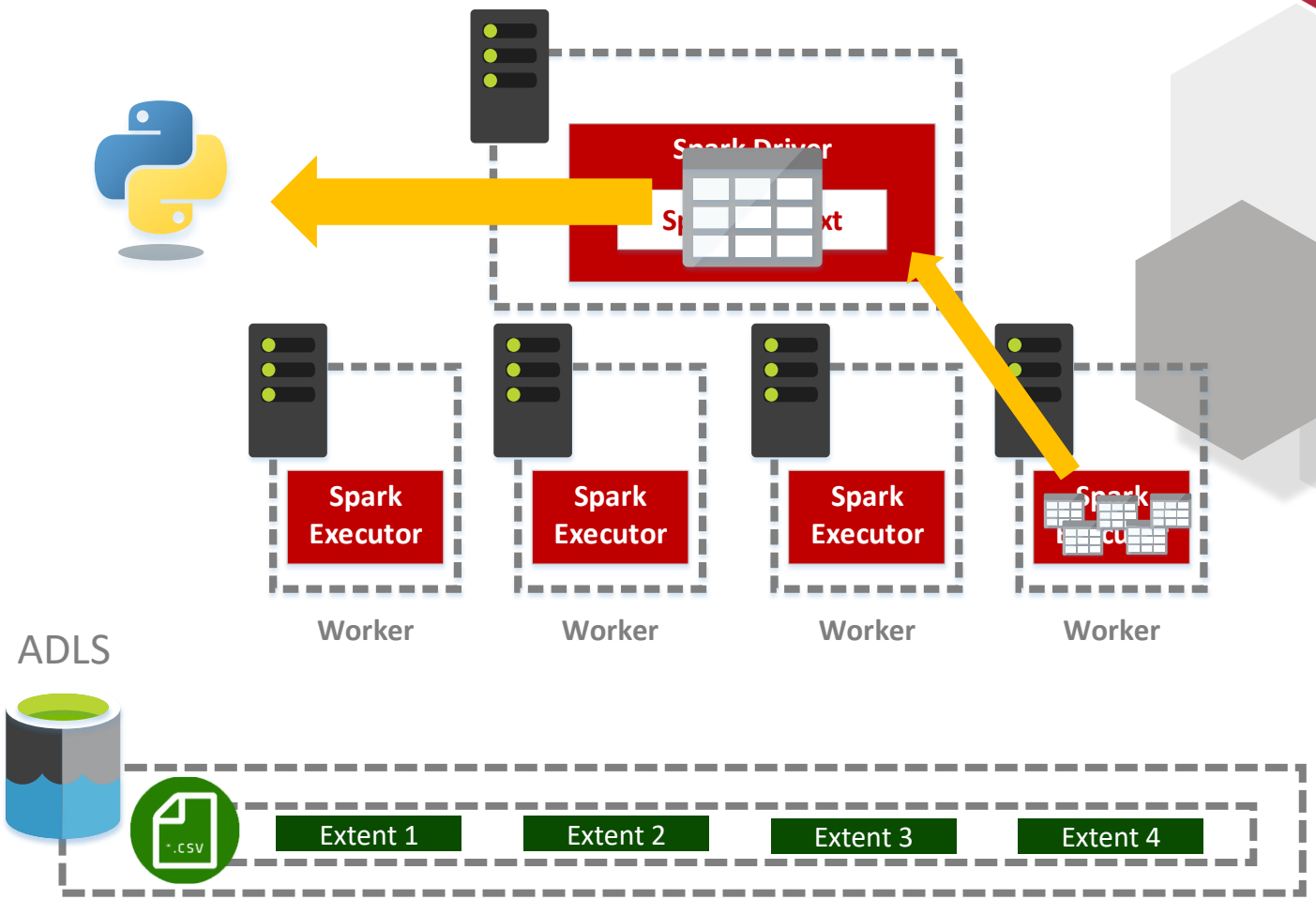
DISTRIBUTED COMPUTE



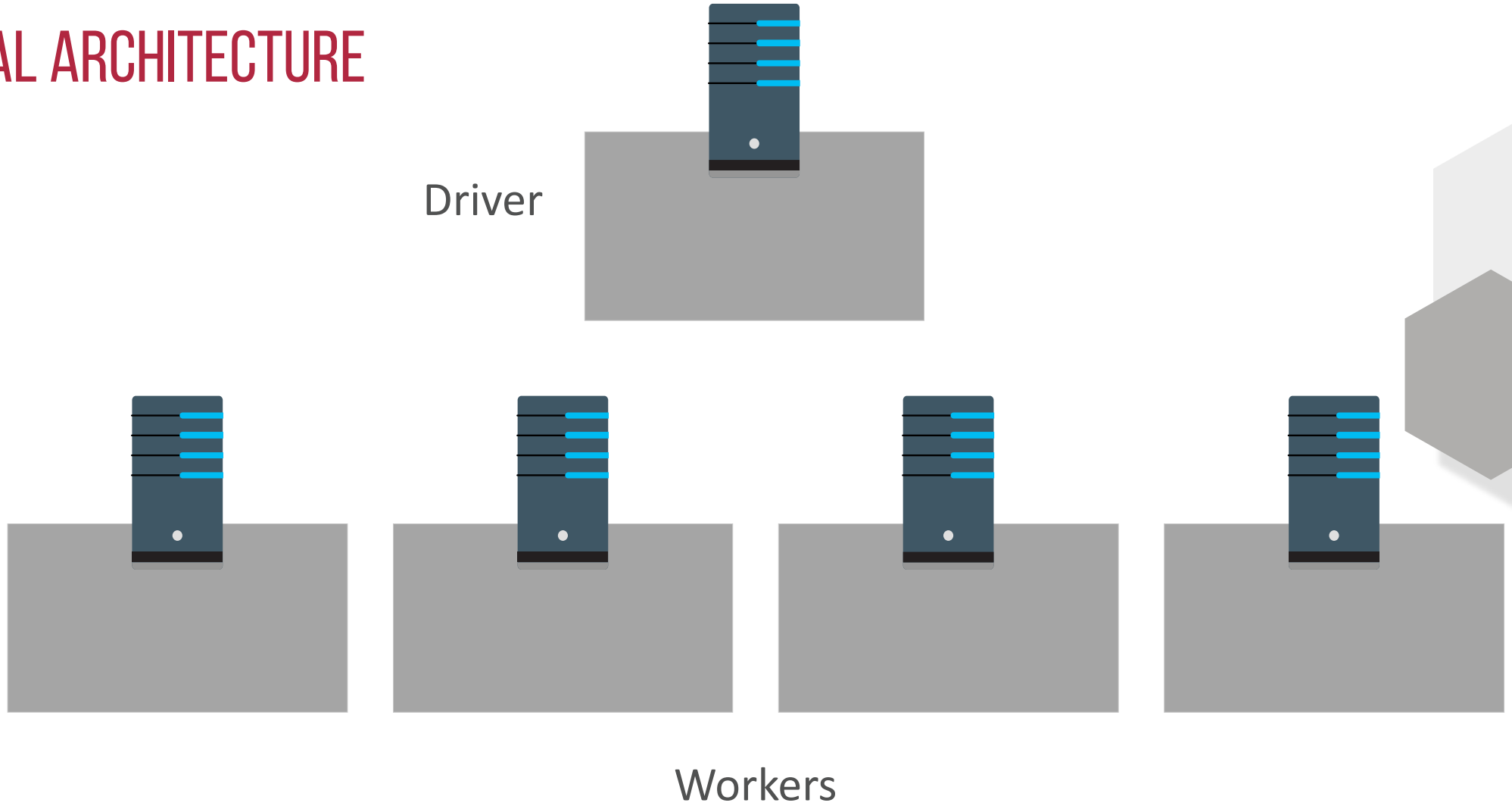
DISTRIBUTED COMPUTE



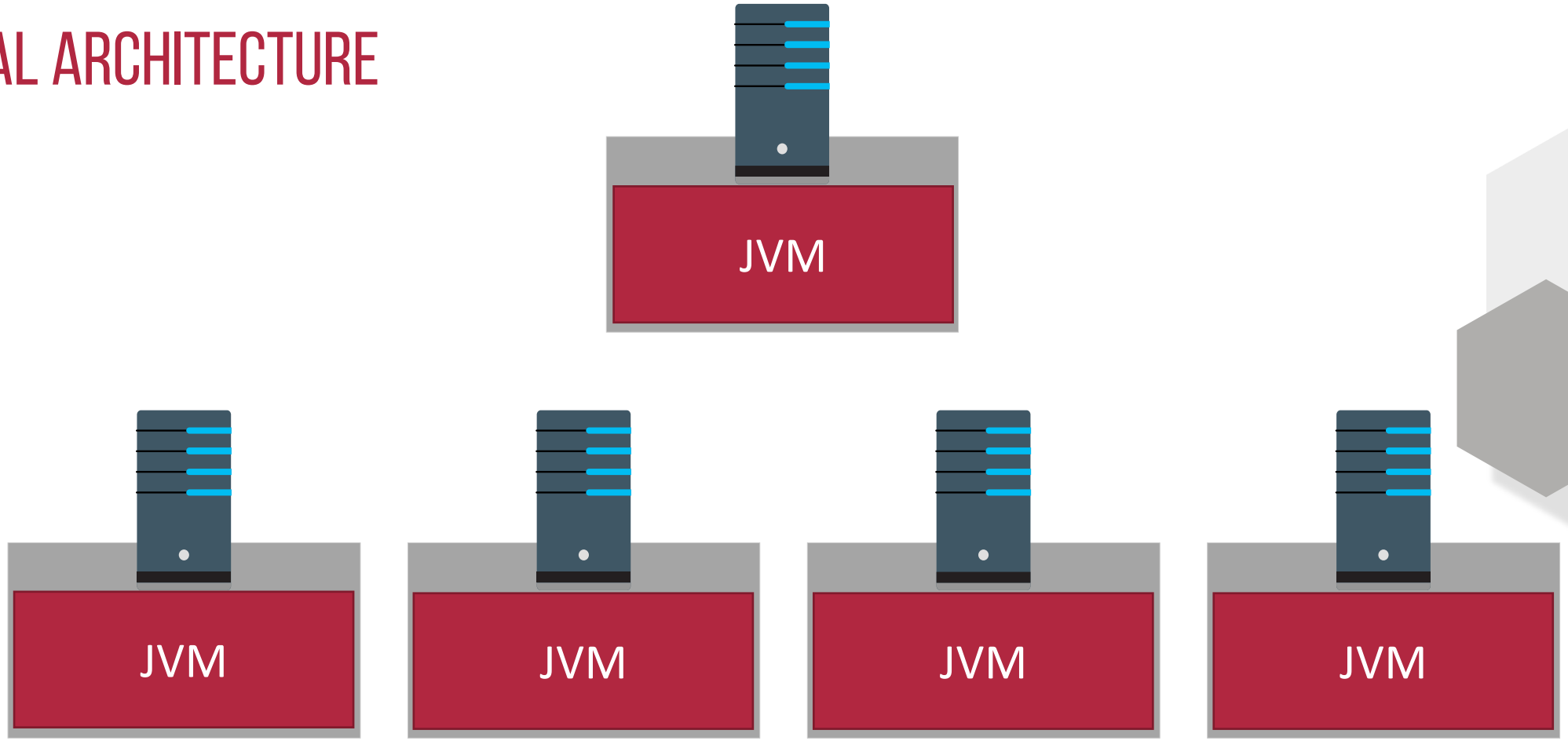
DISTRIBUTED COMPUTE



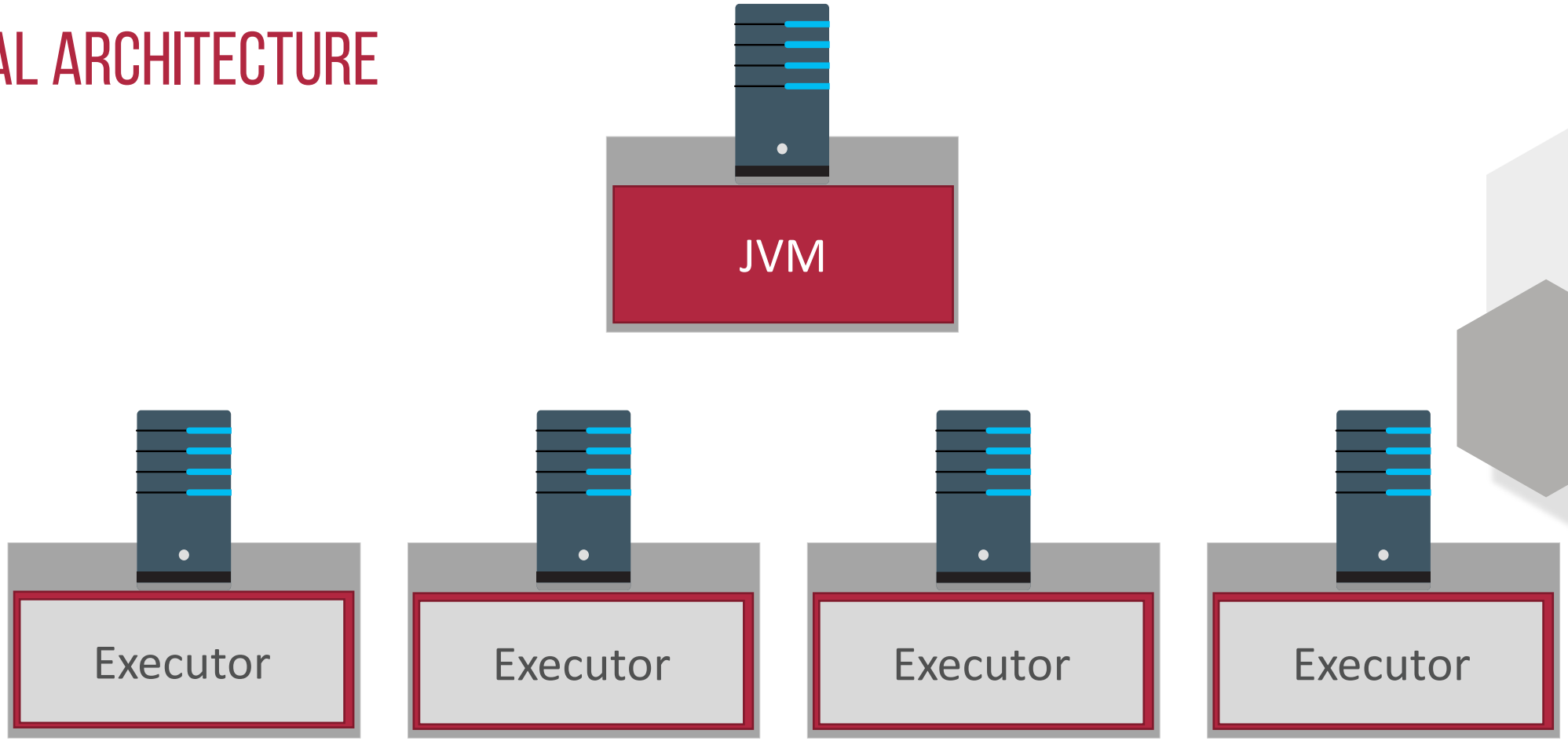
PHYSICAL ARCHITECTURE



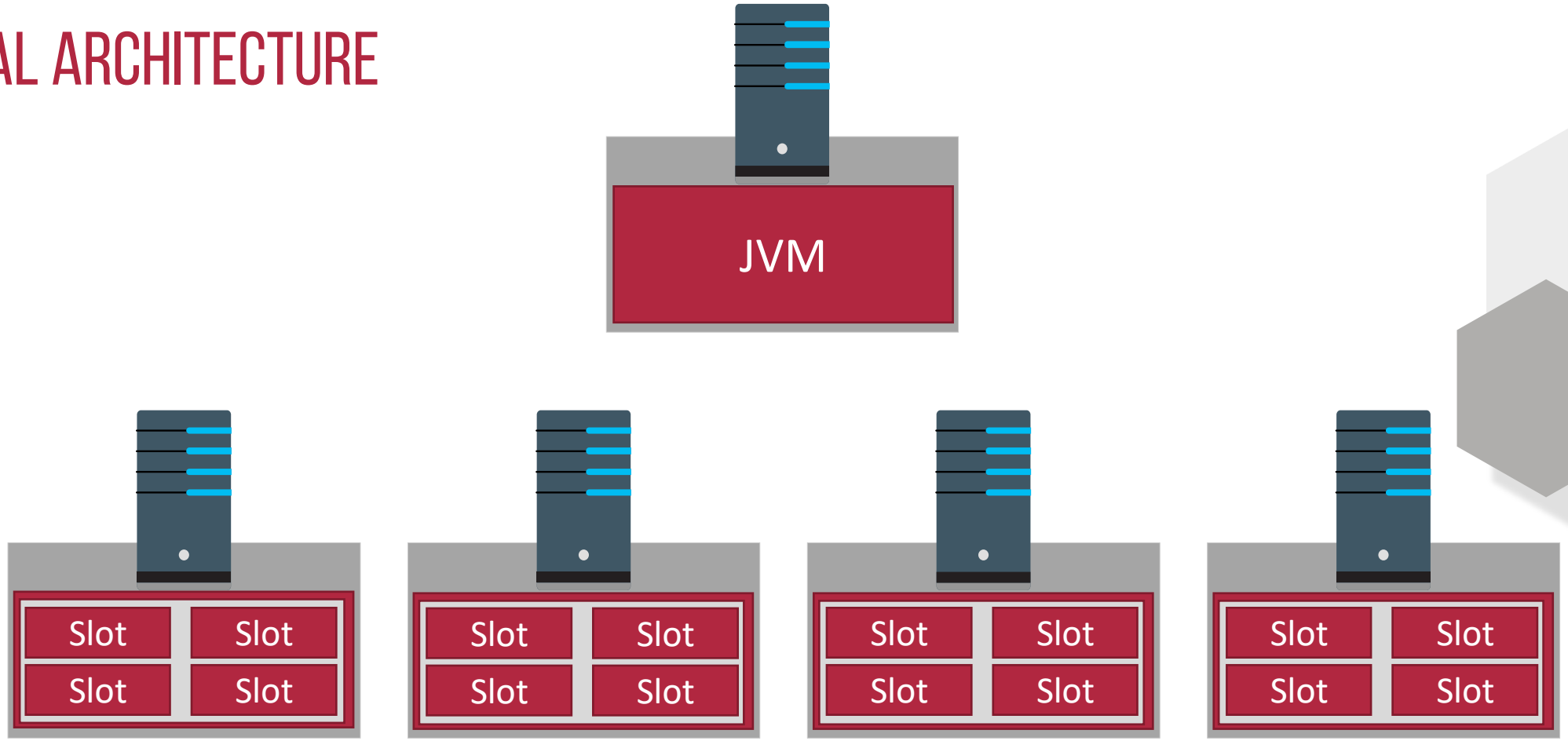
PHYSICAL ARCHITECTURE



PHYSICAL ARCHITECTURE



PHYSICAL ARCHITECTURE



EXECUTION HIERARCHY

```
df = spark.read…
     .filter(…)
     .join(…)
     .write…
```

Transformations

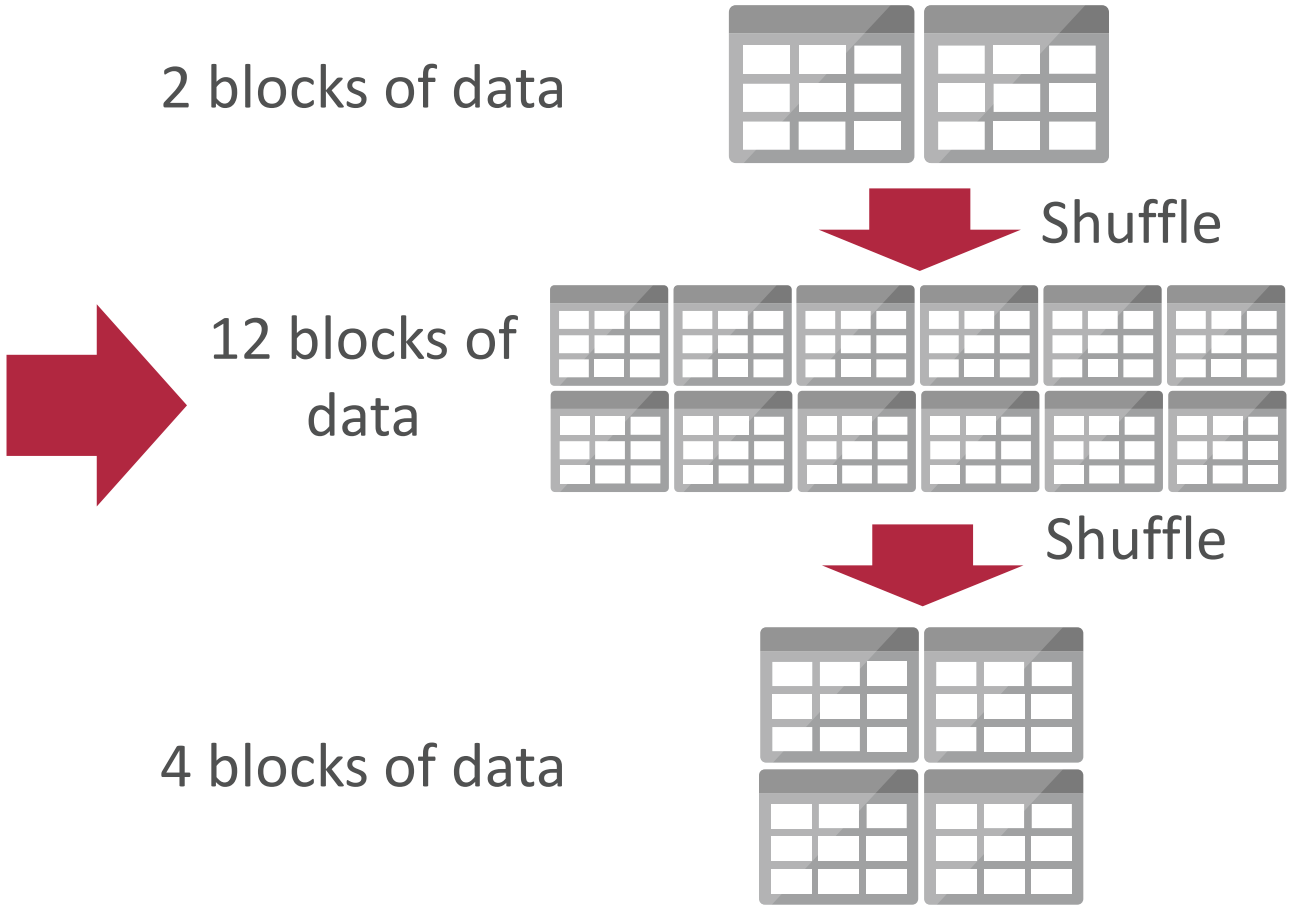
Narrow

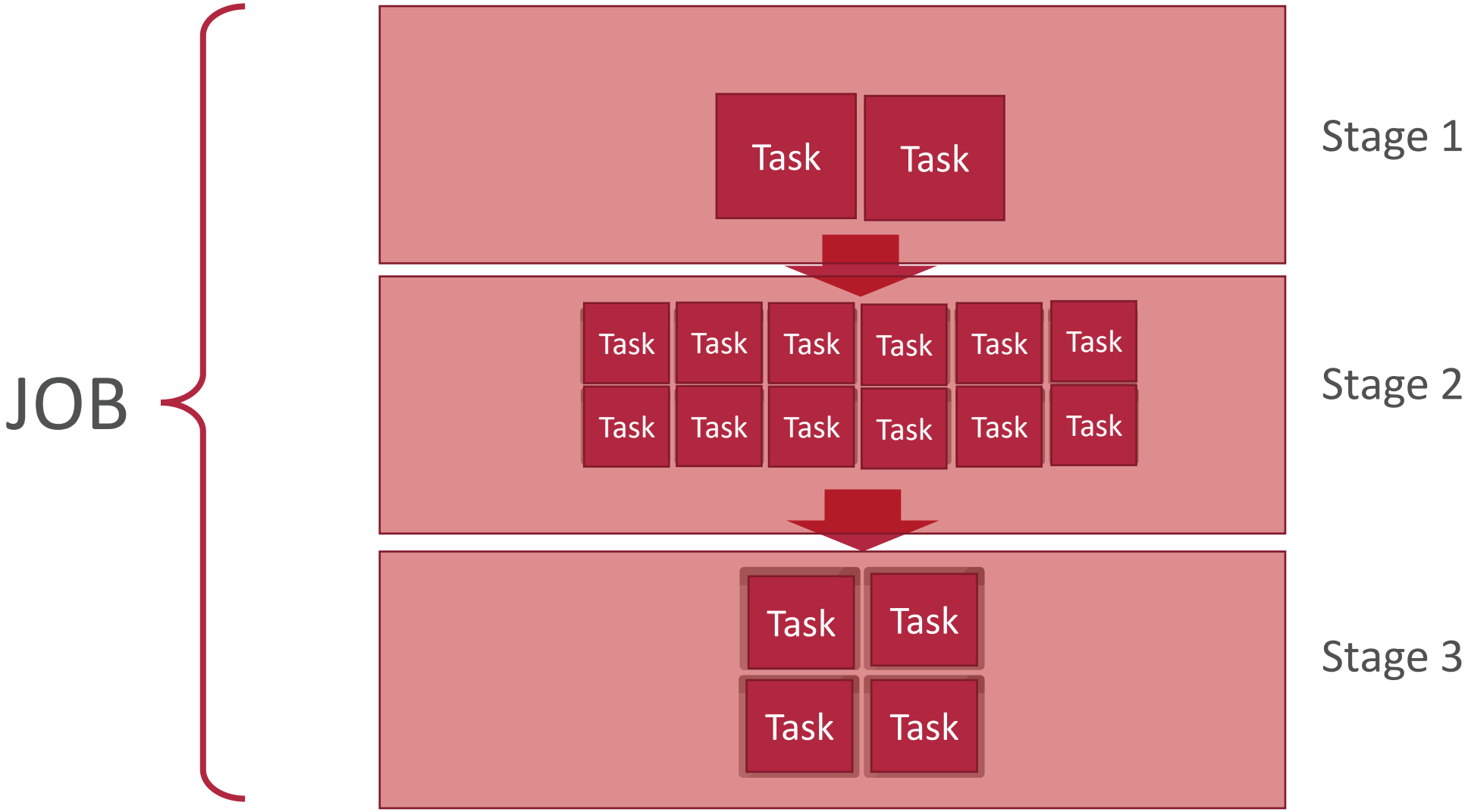
Wide

Actions

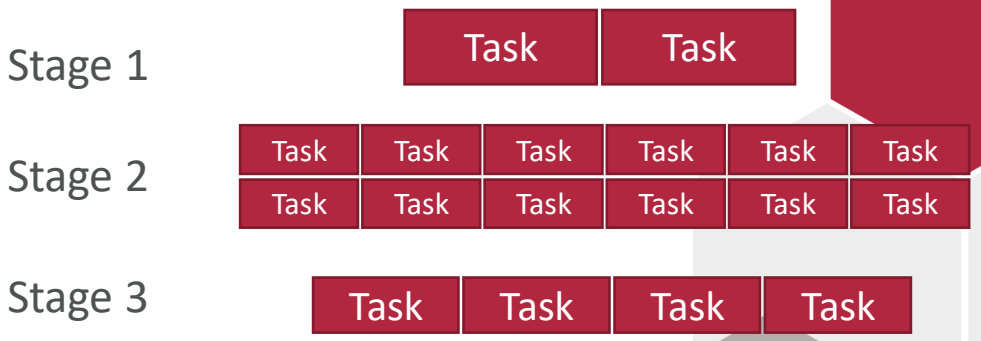
EXECUTION HIERARCHY

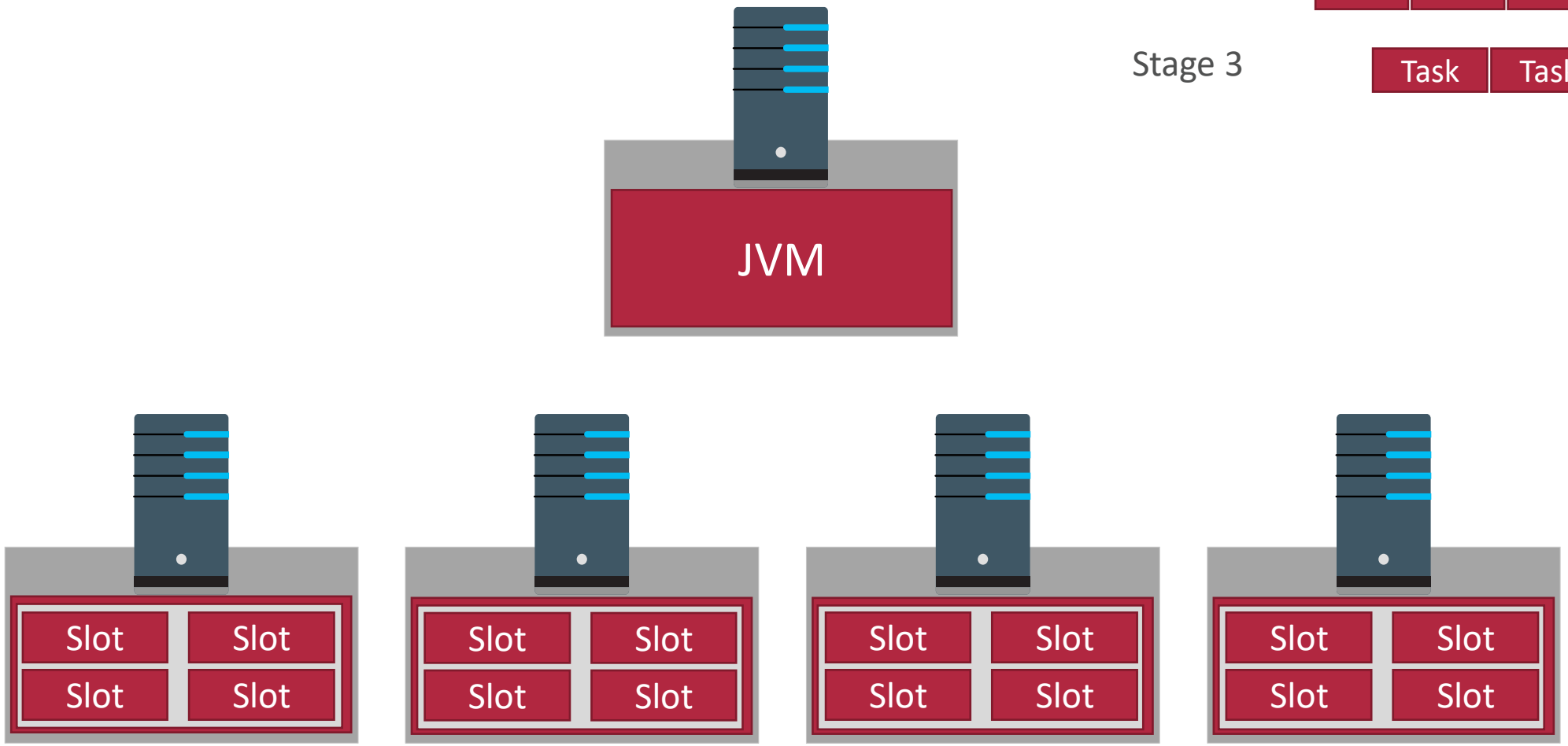
```
df = spark.read…
.filter(…)
.join(…)
.write…
```



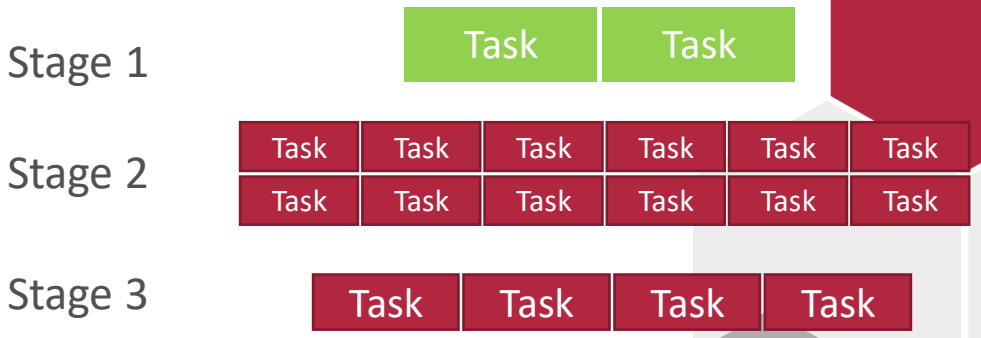

JOB
Task
Task
Stage 1
Task
Task
Task
Task
Task
Task
Task
Task
Task
Task
Task
Task
Stage 2
Task
Task
Task
Task
Stage 3

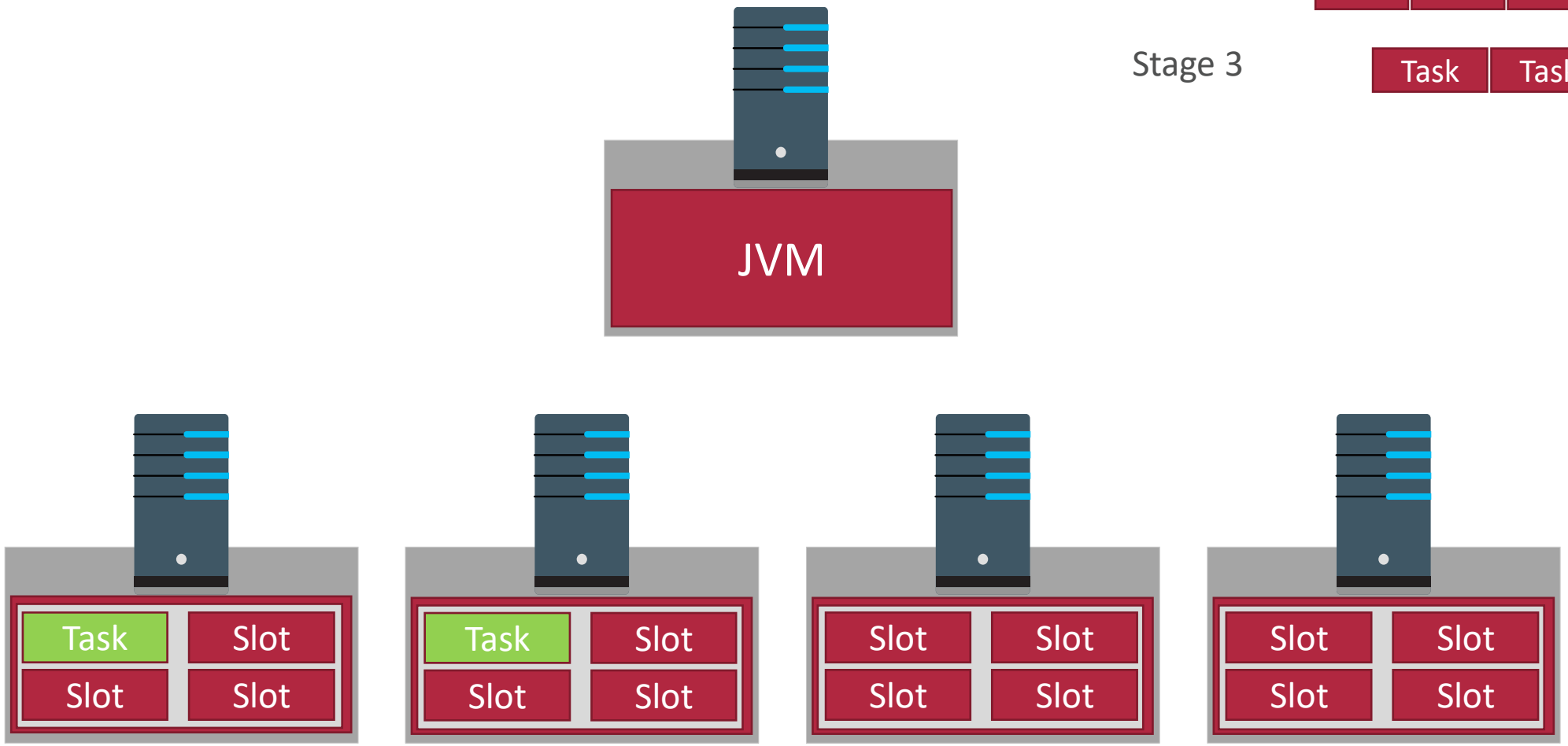
PHYSICAL ARCHITECTURE



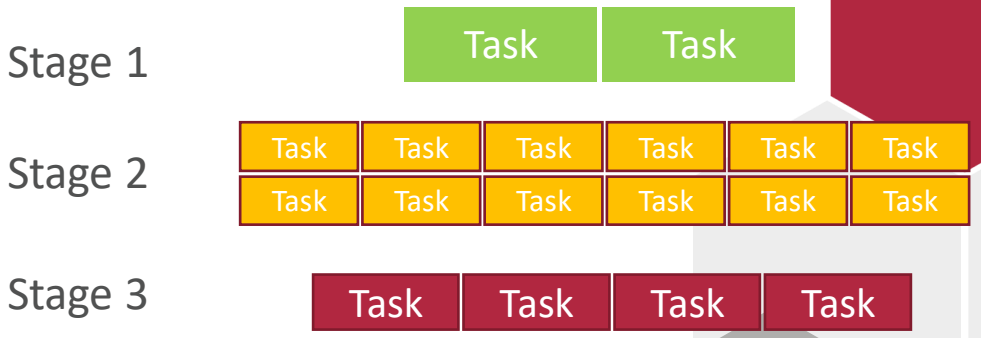


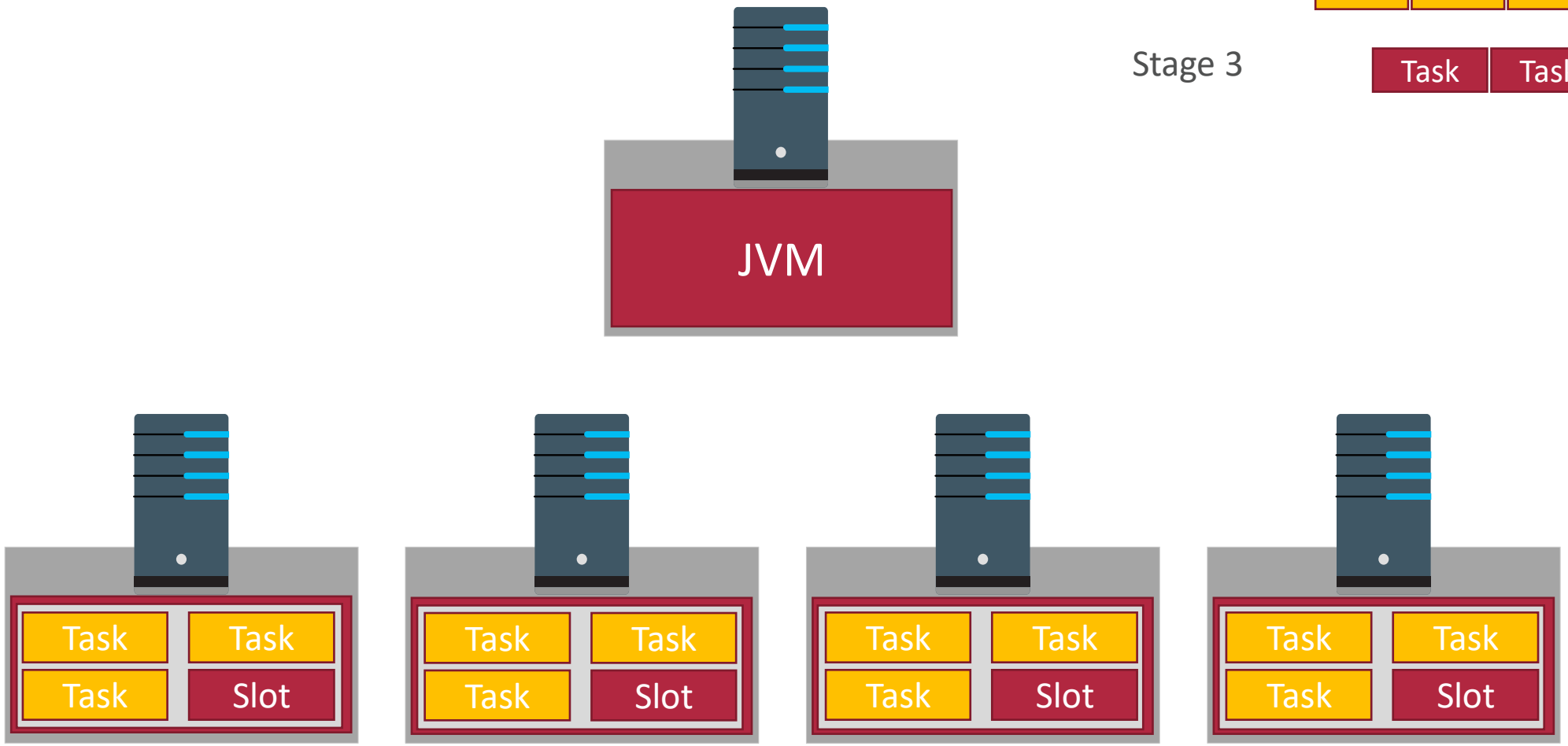
PHYSICAL ARCHITECTURE



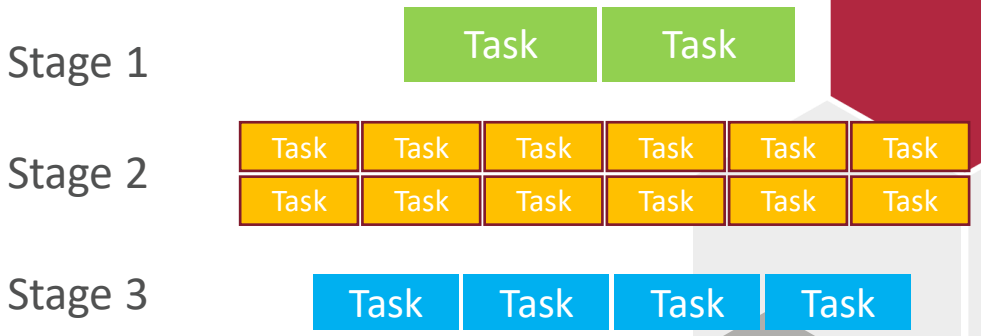


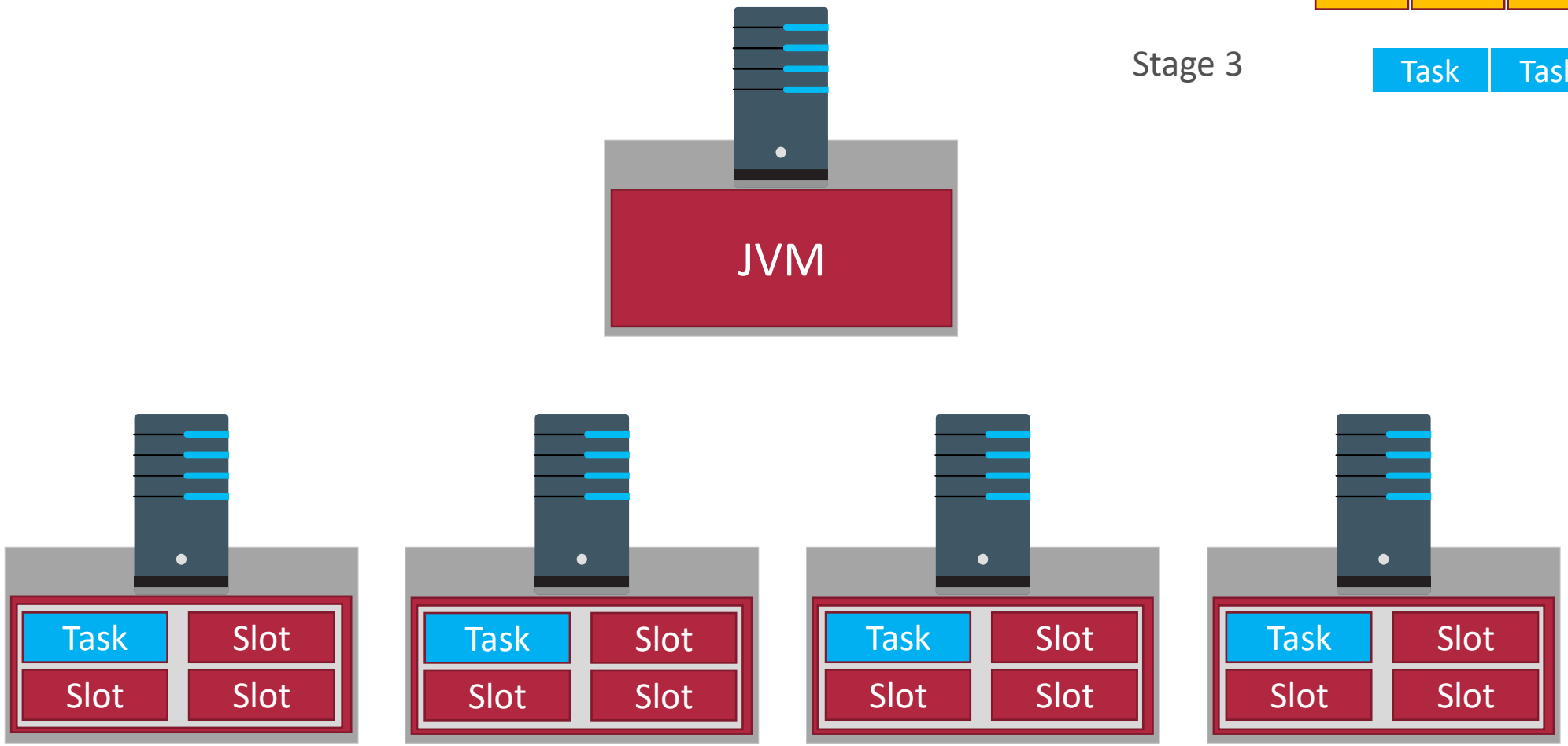
PHYSICAL ARCHITECTURE





PHYSICAL ARCHITECTURE





ETL IN DATABRICKS

SCHEMA ON READ – INFER SCHEMA

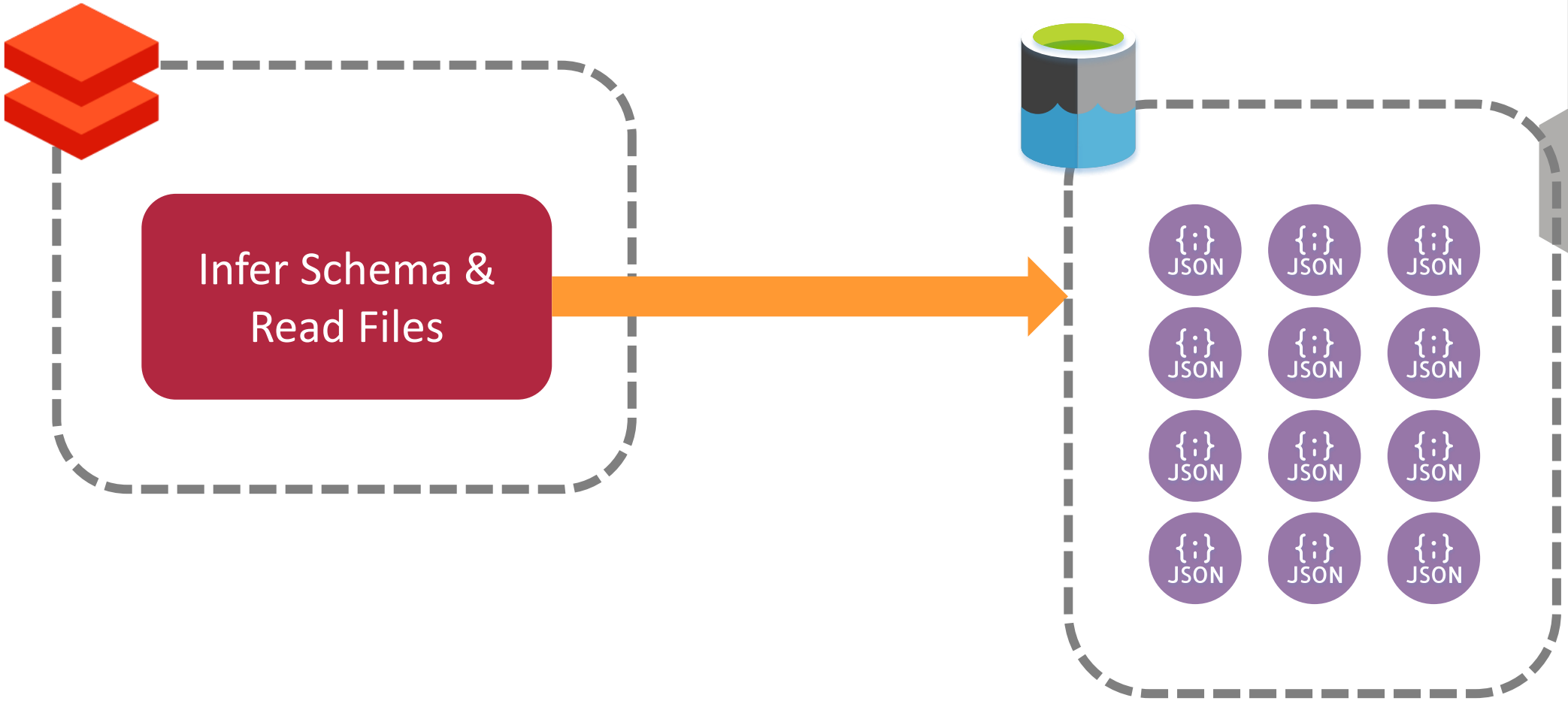
Cmd 3

```
df = sqlContext.read.format("csv") \
  .option("header", "true") \
  .option("inferSchema", "true") \
  .load("abfss://root@dblake.dfs.core.windows.net/RAW/Public/Taxi/v1/SmallSlice.csv")
```

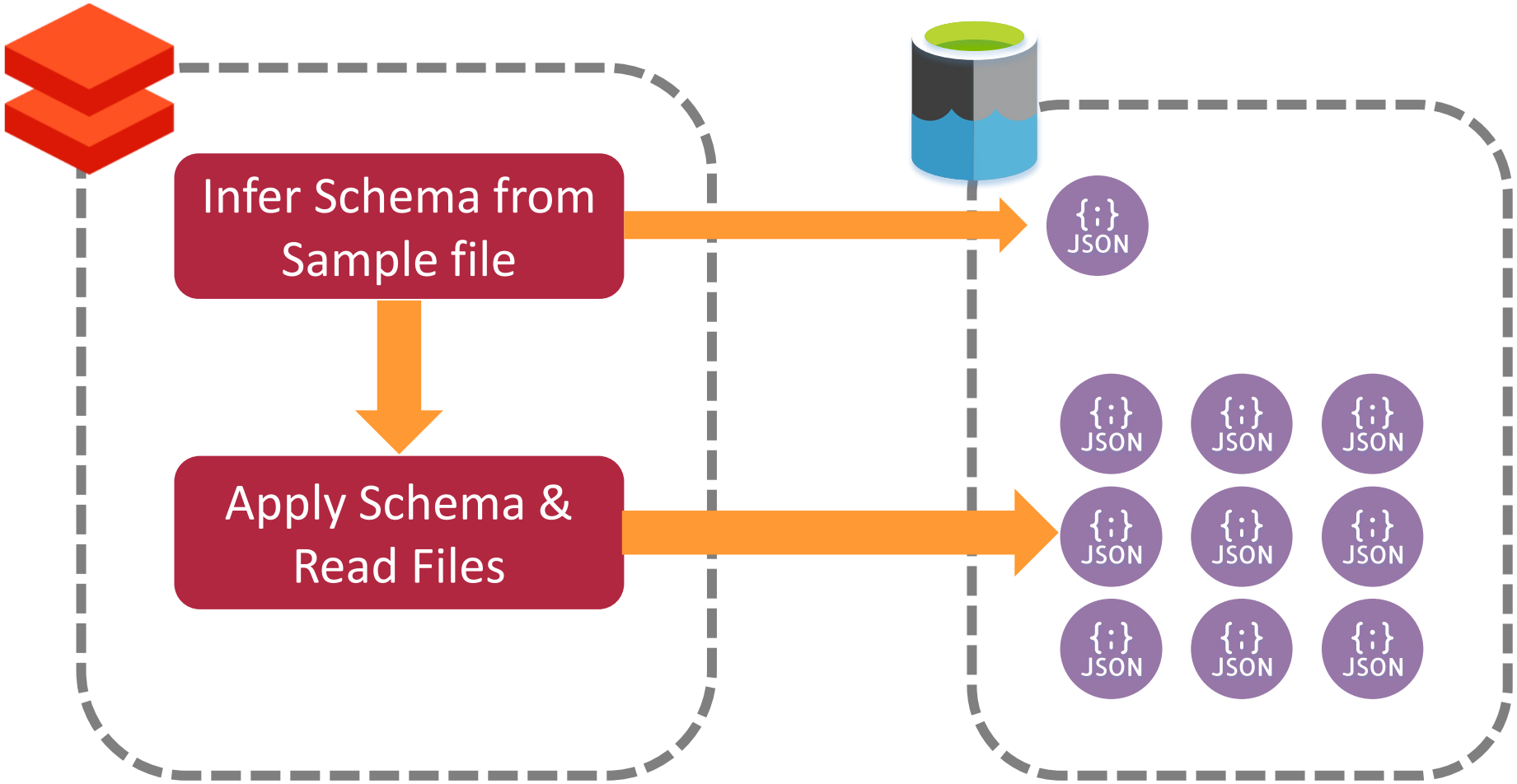
df: pyspark.sql.dataframe.DataFrame

```
Dispatching_base_num: string
Pickup_DateTime: timestamp
DropOff_datetime: string
PUlocationID: integer
DOlocationID: string
```

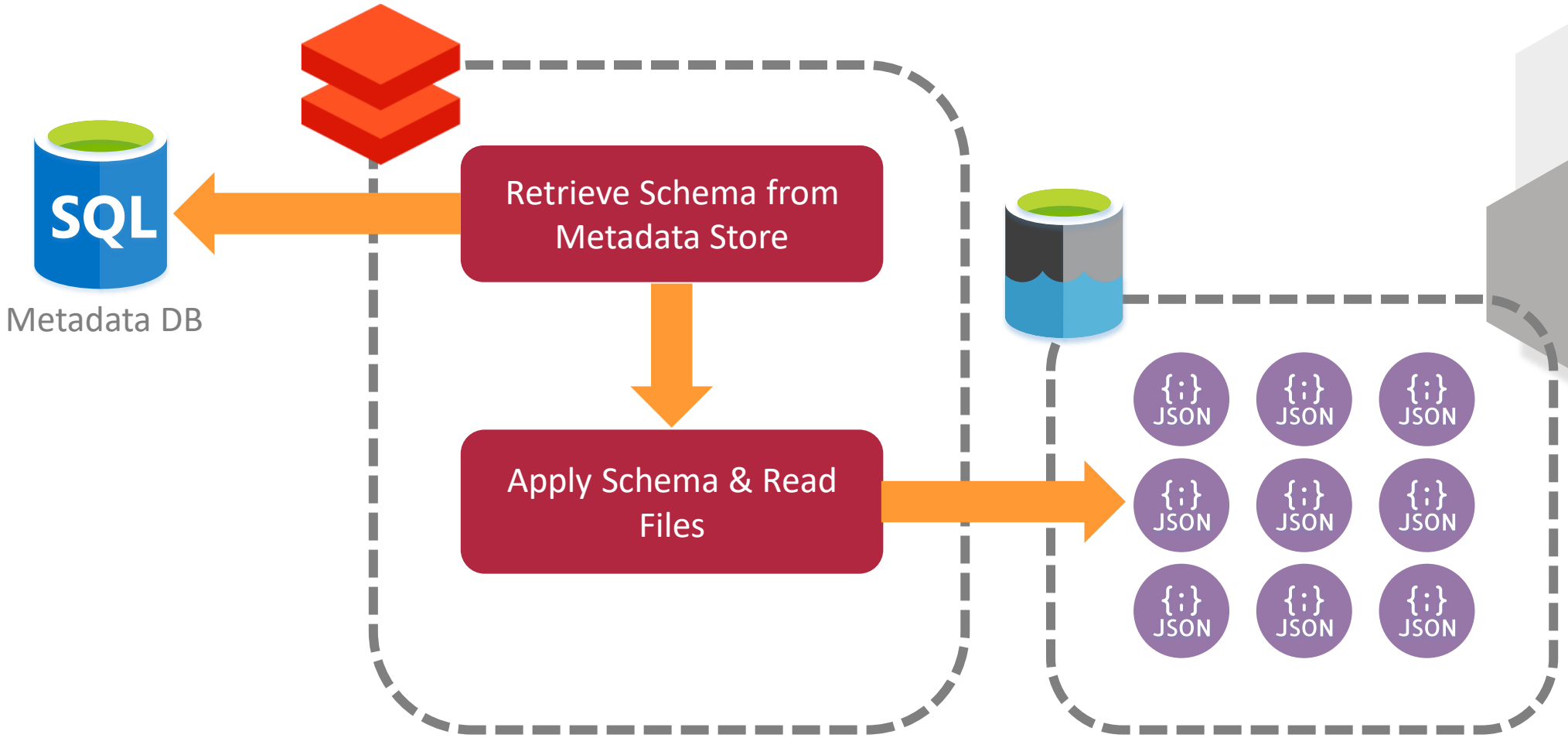
SCHEMA ON READ – INFER SCHEMA



SCHEMA ON READ – SAMPLE FILES

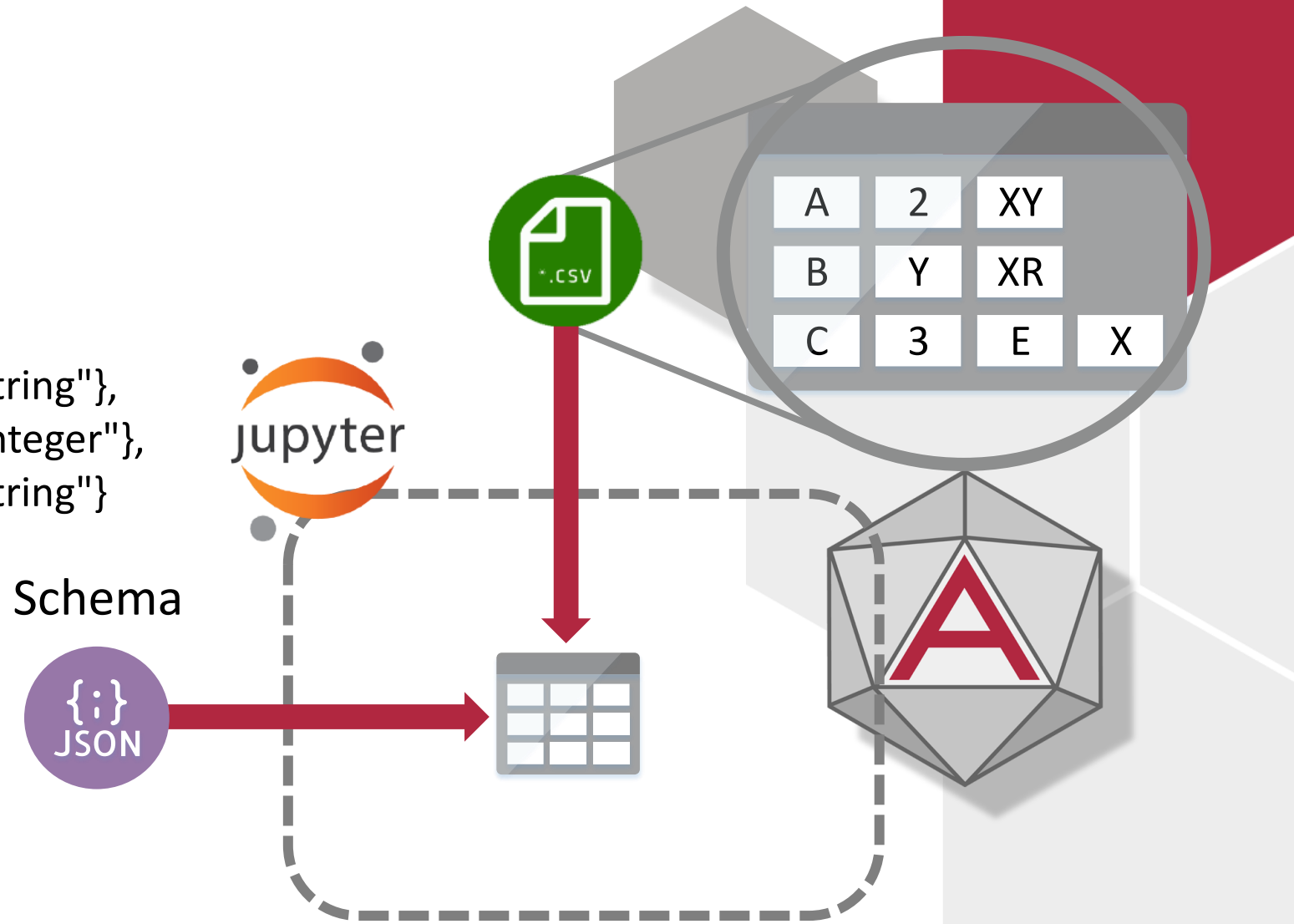


SCHEMA ON READ – METADATA STORE

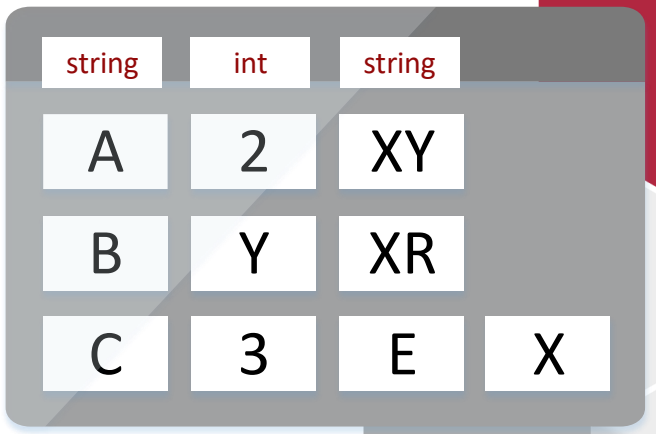


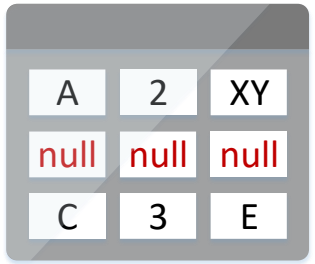
DATA VALIDATION

```
{"fields":[
{"name":“Col1","nullable":true,"type":"string"},
{"name":“Col2","nullable":true,"type":“integer"},
{"name":“Col3","nullable":true,"type":"string"}
]}
```



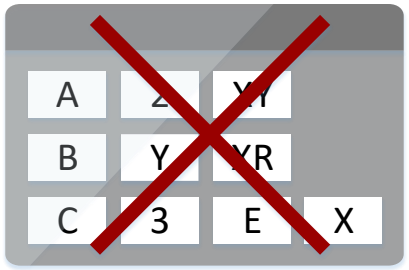
DATA VALIDATION





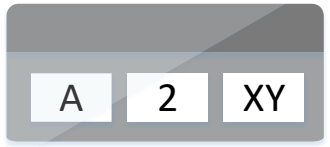
1 **PERMISSIVE**

NULL DataType Failures
Drop Extra Columns



2 **FAILFAST**

Fail on any error



3 **DROPMALFORMED**

Drop rows with DataType Failures
Drop rows with Extra Columns

DATA VALIDATION

```
#Add a system attribute to our structure
newSchema.add("_corrupt_record", StringType(), True)

#Load the data so we can inspect it
df = (spark
        .read
        .schema(newSchema)
        .option("mode", "PERMISSIVE")
        .csv(dataLocation)
      )
```


```
#Separate bad rows automatically
baddf = (spark
        .read
        .option("badRecordsPath", "/mnt/data/RAW/TaxiZones/_reject/")
        .schema(mySchema)
        .csv(location)
      )
```

DATA PROCESSING WITH SPARK

- Working with Flat Files
- Working with Dodgy Data
- Meta-Driven Systems

ORCHESTRATION

DATABRICKS WIDGETS

Cmd 3

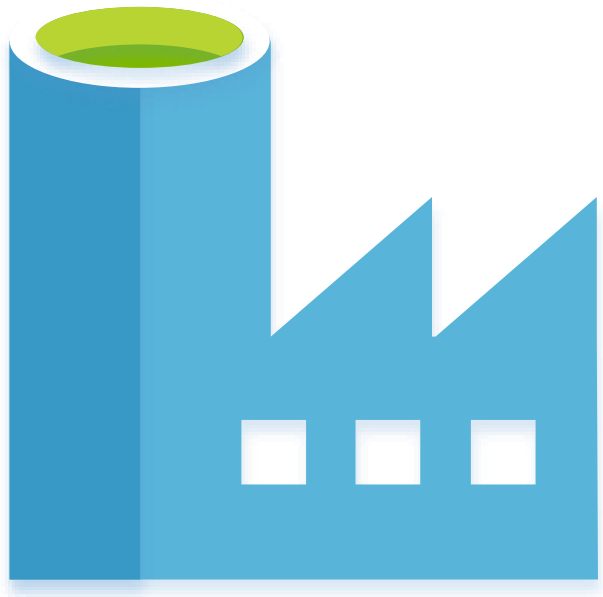
```
#dbutils.widgets.removeAll()
dbutils.widgets.text("fileName", "Product","AdventureWorks Table")
dbutils.widgets.dropdown("entity_name", "Taxi",["Taxi","TaxiZones"] ,"Entity Name")
```

AdventureWorks Table : ProductCategory

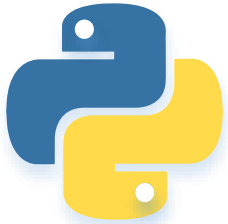
Entity Name : Taxi

Cmd 8

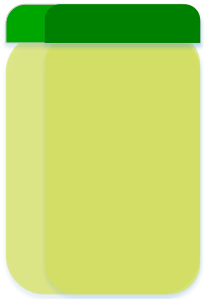
```
fileName = dbutils.widgets.get("fileName")
```



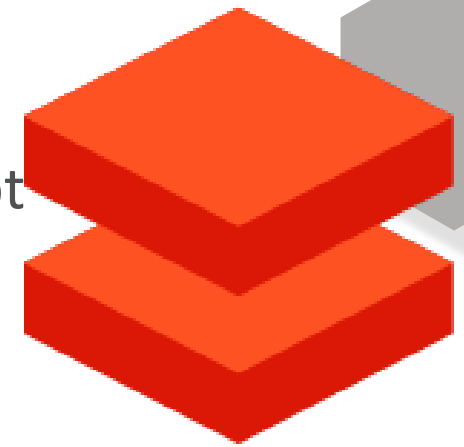
Notebook



Python Script

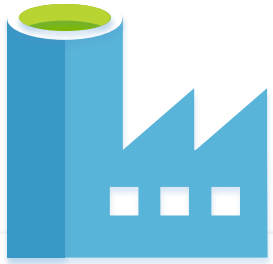


Jar File



ADVANCING ANALYTICS

AZURE DATA FACTORY



Notebook path * /Demonstrations/Dynamic Transformation/D Browse

Base Parameters

+ New | Delete

NAME	VALUE
entity_name	Taxi

Entity Name : Taxi

Cmd 1

Configure Widgets

```
#dbutils.widgets.removeAll()
dbutils.widgets.dropdown("entity_name", "Taxi",["Taxi","TaxiZones"] ,"Entity Name")
```

```
"runOutput": {
    "TransformationsApplied": 2,
    "Status": "Succeeded",
    "ProcessedRows": 66141344
},
```

Cmd 10

Return Results to Caller

```
import json
dbutils.notebook.exit(json.dumps({"processedRows":processedRows, "status":"Succeeded",
```

```
Notebook exited: {"Status": "Succeeded", "ProcessedRows": 66141344, "TransformationsApplie
```

ADVANCING ANALYTICS

DATABRICKS ORCHESTRATION

- Adding input & output parameters
- Orchestrating Databricks with ADF

THANKS!

youtube.com/c/AdvancingAnalytics

@MrSiWhiteley

www.advancinganalytics.co.uk

TEAMS MEETINGS

Azure BI - Join Microsoft Teams Meeting

Azure DB - Join Microsoft Teams Meeting

Cloud Development - Join Microsoft Teams Meeting

Data Platform - Join Microsoft Teams Meeting

Power BI - Join Microsoft Teams Meeting