Apache Spark

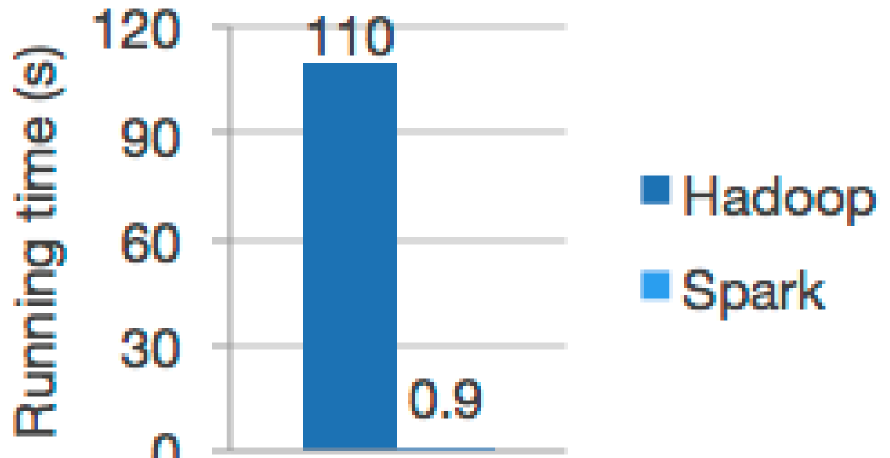
AmirReza Mohammadi

2019 winter

Apache Spark

- *Lightning-fast unified analytics engine* for large-scale data processing.
- Speed:

  Run workloads 100x faster.



Spark.apache.org

Apache Spark

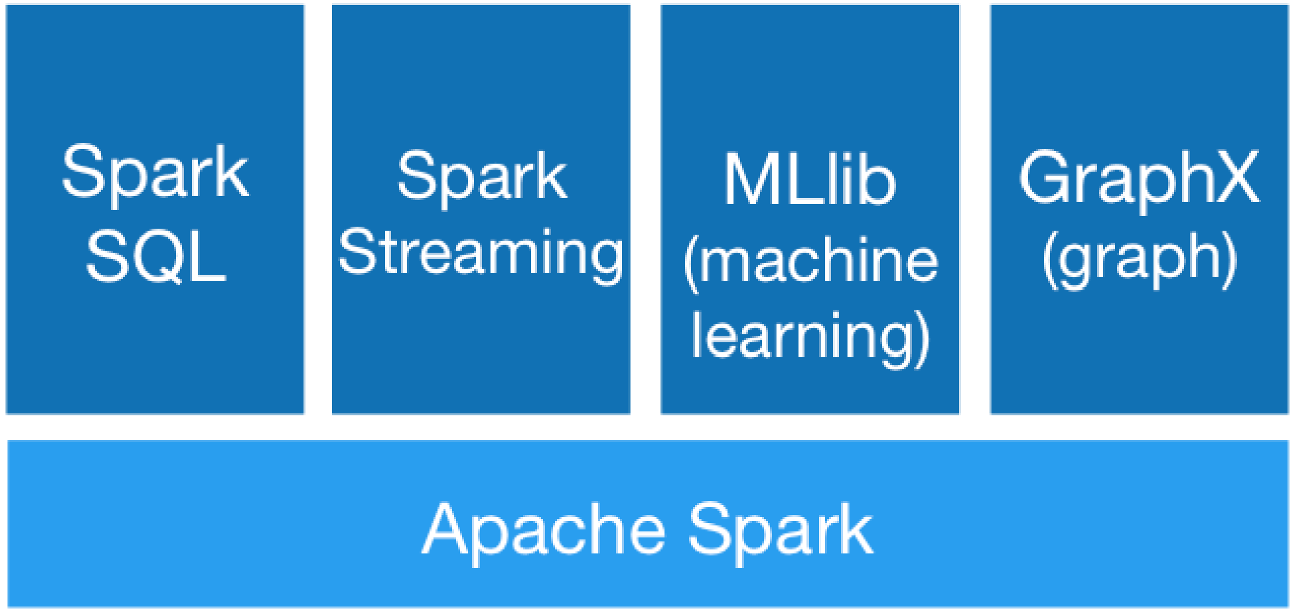
- *Lightning-fast unified analytics engine* for large-scale data processing.
- Speed:

  Run workloads 100x faster.
- Ease of Use

  Write applications quickly in Java, Scala, Python, R, and SQL.
- Generality

  Combine SQL, streaming, and complex analytics.

Spark Components



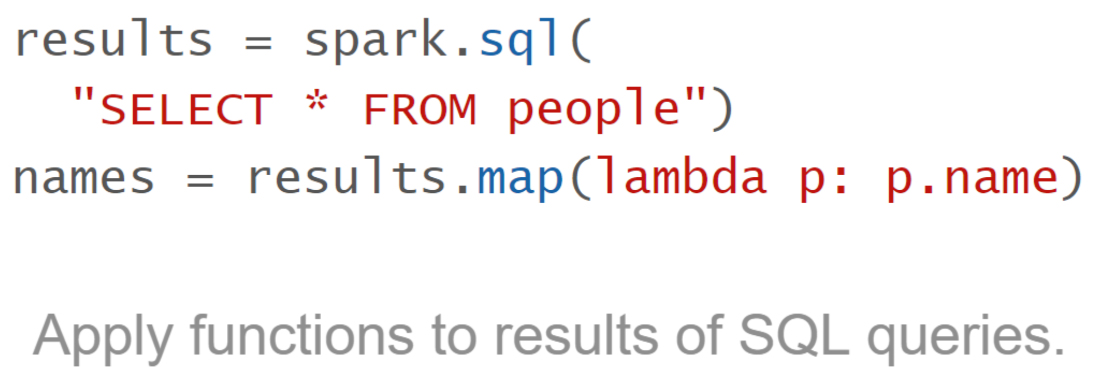
Spark.apache.org

Spark SQL

Apache Spark's module for working with structured data.

- Integrated

  Seamlessly mix SQL queries with Spark programs.



```
results = spark.sql(
  "SELECT * FROM people")
names = results.map(lambda p: p.name)
```

Apply functions to results of SQL queries.

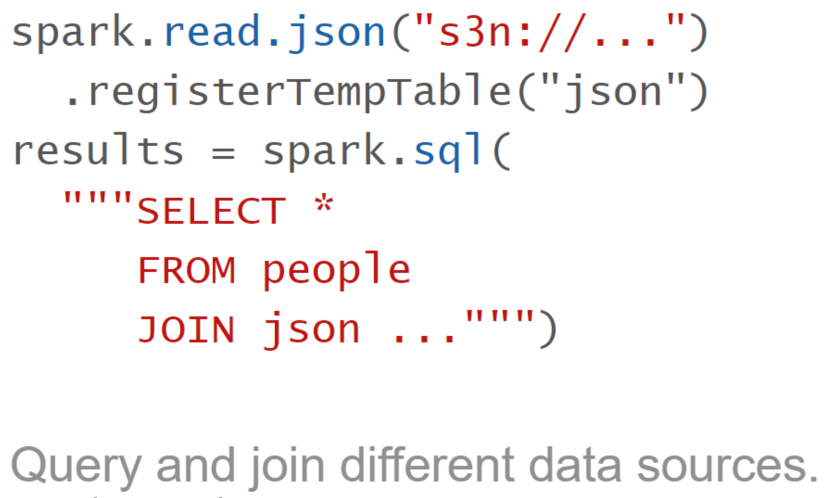
Spark.apache.org

Spark SQL

Apache Spark's module for working with structured data.

- Uniform Data Access

  Connect to any data source the same way.



```
spark.read.json("s3n://...")
  .registerTempTable("json")
results = spark.sql(
  """SELECT *
     FROM people
     JOIN json ...""")
```

Query and join different data sources.

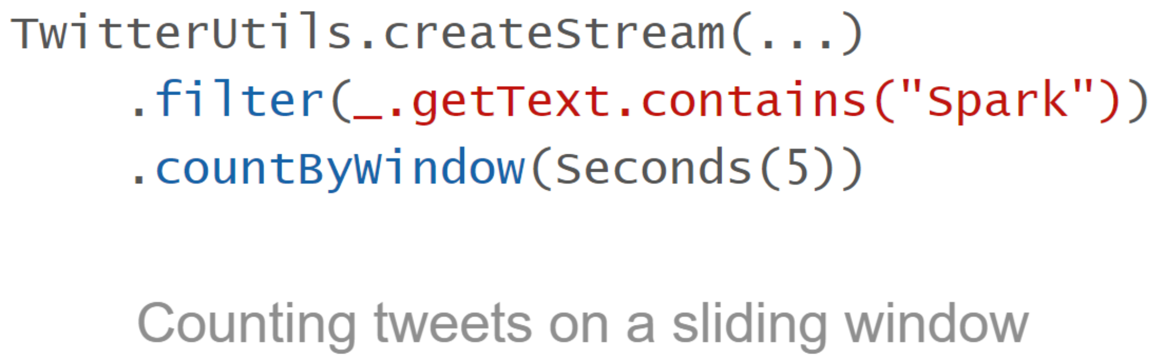
Spark.apache.org

Spark Streaming

Spark Streaming makes it easy to build scalable fault-tolerant streaming applications.

- Ease of Use

  Build applications through high-level operators.



```
TwitterUtils.createStream(...)
    .filter(_.getText.contains("Spark"))
    .countByWindow(Seconds(5))
```

Counting tweets on a sliding window

Spark.apache.org

Spark Streaming

Spark Streaming makes it easy to build scalable fault-tolerant streaming applications.

- Spark Integration

    Combine streaming with batch and interactive queries.

```
stream.join(historicCounts).filter {
  case (word, (curCount, oldCount)) =>
    curCount > oldCount
}
```

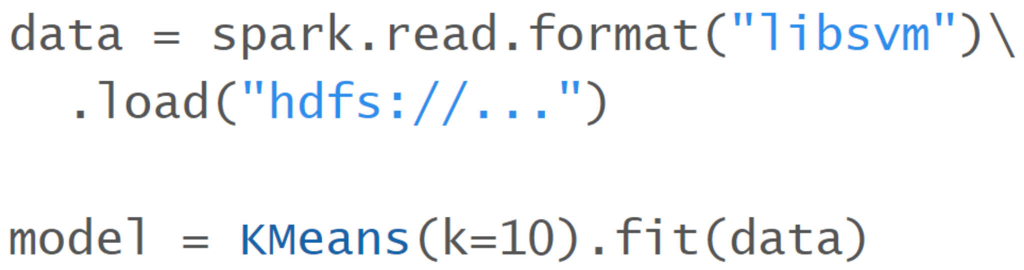
Find words with higher frequency than historic data

Spark.apache.org

Spark MLlib

- **MLlib** is Apache Spark's scalable machine learning library.
- Ease of Use

  Usable in Java, Scala, Python, and R.



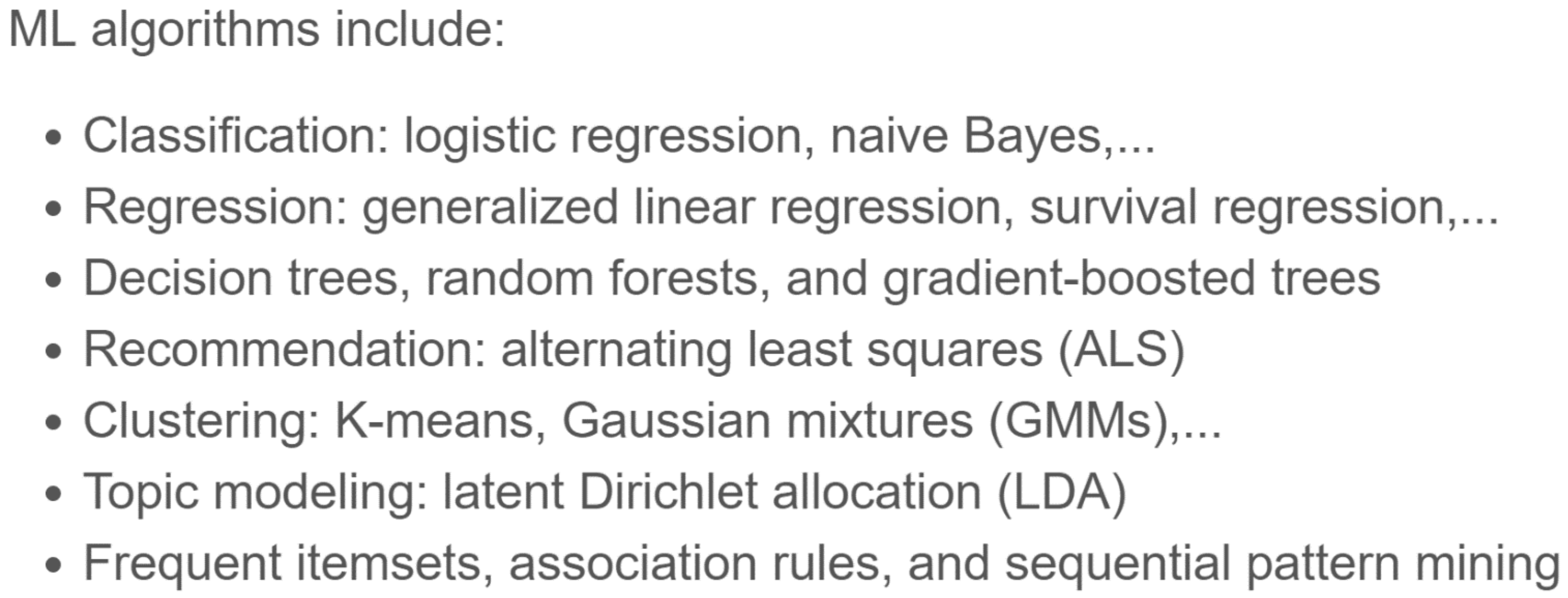
```
data = spark.read.format("libsvm")\
  .load("hdfs://...")

model = KMeans(k=10).fit(data)
```

Calling MLlib in Python

Spark.apache.org

Spark MLlib



ML algorithms include:

- Classification: logistic regression, naive Bayes,...
- Regression: generalized linear regression, survival regression,...
- Decision trees, random forests, and gradient-boosted trees
- Recommendation: alternating least squares (ALS)
- Clustering: K-means, Gaussian mixtures (GMMs),...
- Topic modeling: latent Dirichlet allocation (LDA)
- Frequent itemsets, association rules, and sequential pattern mining

Spark MLlib

ML workflow utilities include:

- Feature transformations: standardization, normalization, hashing,...
- ML Pipeline construction
- Model evaluation and hyper-parameter tuning
- ML persistence: saving and loading models and Pipelines

Other utilities include:

- Distributed linear algebra: SVD, PCA,...
- Statistics: summary statistics, hypothesis testing,...

Spark GraphX

GraphX is Apache Spark's API for graphs and graph-parallel computation.

- Flexibility

  Seamlessly work with both graphs and collections.

- Algorithms

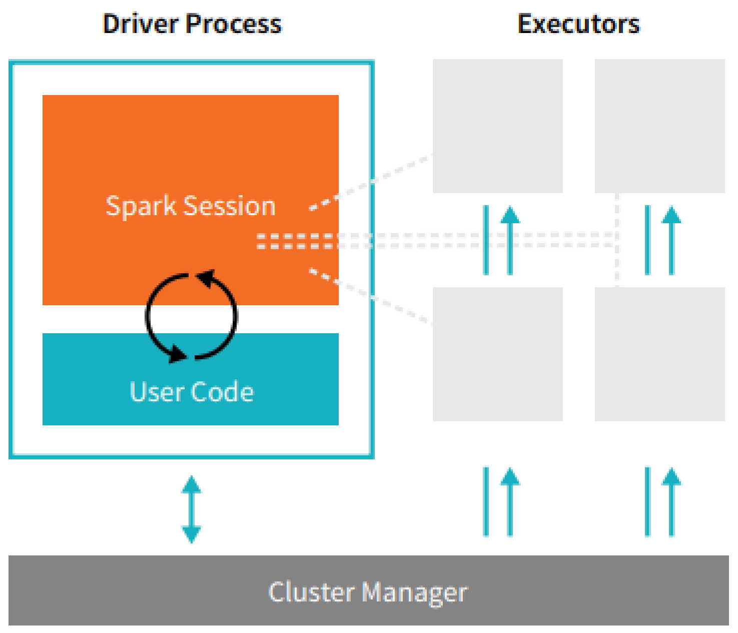
  Choose from a growing library of graph algorithms.

- Speed

  Comparable performance to the fastest specialized graph processing systems.

Spark Application

driver process and a set of ***executor*** processes.



Databricks.com

Spark Context and environment

sparkContext

sqlContext

Spark 2.X we have sparkSession

The Data Interfaces

DataFrame:

collection of distributed Row types.

RDD (Resilient Distributed Dataset):

an interface to a sequence of data objects that consist of one or more types that are located across a variety of machines in a cluster.

DataSet:

combination of DataFrames and RDDs.

Spark Applications

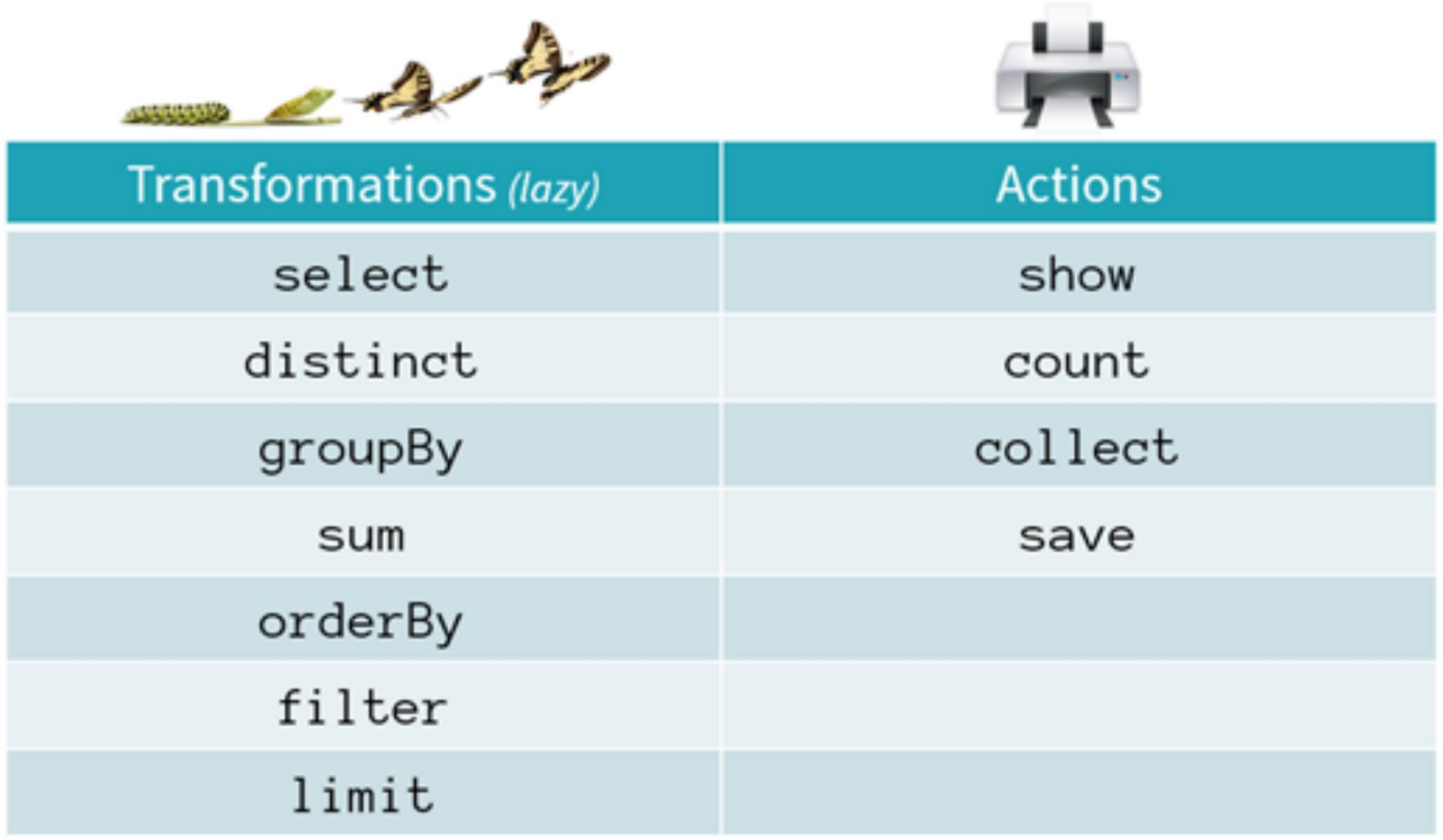
Transformations

Transformations are **operations** that will **not** be completed at the time you write and execute the code in a cell - they will only get executed once you have called a **action**.

Actions

Actions are commands that are computed by Spark right at the time of their execution.

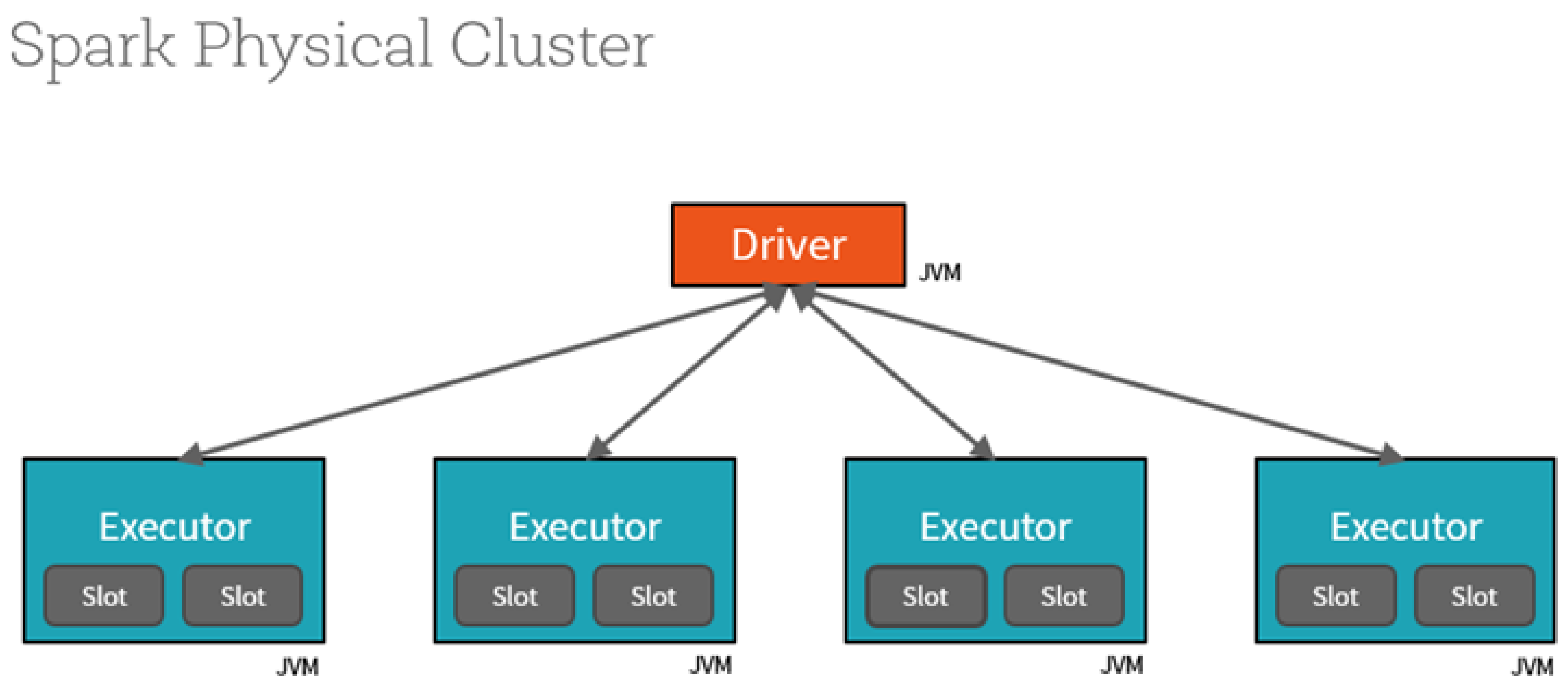
Spark Applications



Transformations *(lazy)*	Actions
select	show
distinct	count
groupBy	collect
sum	save
orderBy	
filter	
limit	

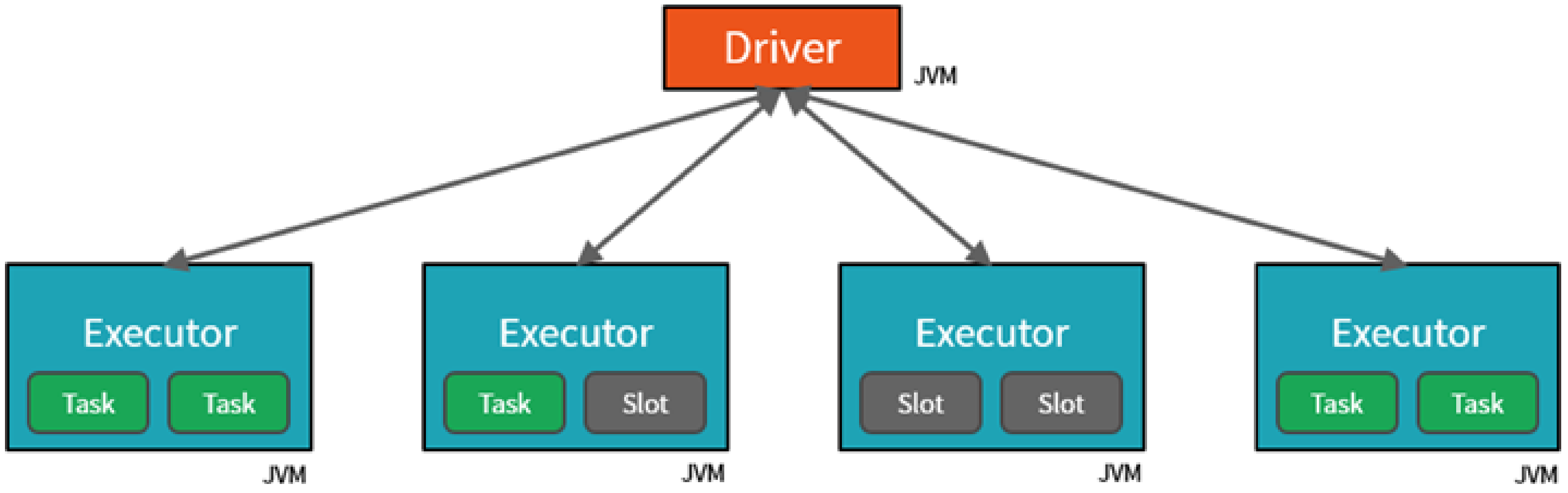
Databricks.com

Apache Spark Architecture



Databricks.com

Apache Spark Architecture



Databricks.com

The End

Any Question?

“Sometimes when you innovate, you make mistakes. It is best to admit them quickly, and get on with improving your other innovations.”

Steve Jobs

Transformations and actions

```
# An example of a transformation
# select the ID column values and multiply them by 2
secondDataFrame = firstDataFrame.selectExpr("(id * 2) as value")
```

```
# an example of an action
# take the first 5 values that we have in our firstDataFrame
print firstDataFrame.take(5)
```