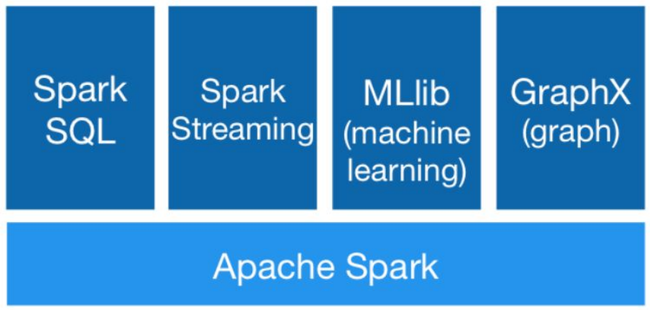
PySpark (Basic) Cheat Sheet

V2020.11.5
(Dr Yan Xu)

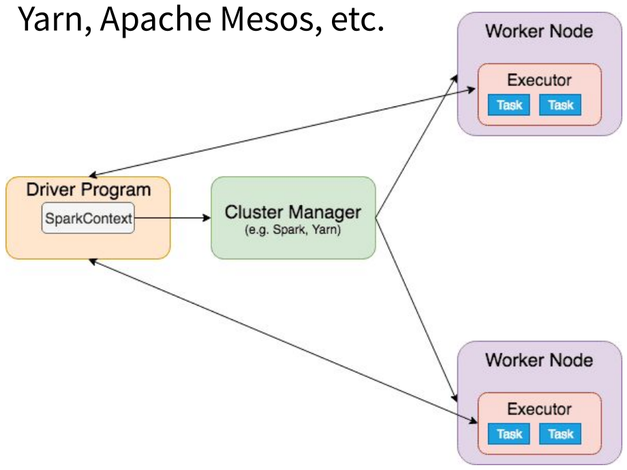
Spark Features (key: **Spark SQL**)

Apache Spark is a unified analytics engine for large-scale data processing.



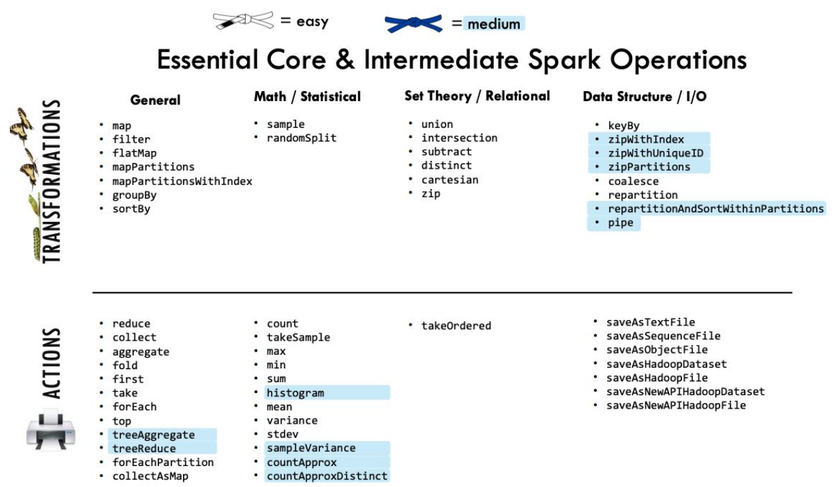
Spark Computing Framework

- Spark is developed by Scala, so nodes have JVM installed and will run Java code. PySpark is a Python API for Spark.
- SparkContext is the coordinator (both hardware and software)
- The cluster is not necessarily managed by Spark Cluster Manager, it could be Hadoop Yarn, Apache Mesos, etc.



Resilient (or immutable) Distributed Dataset

- RDD will be divided into logical partitions, which may be computed on different nodes of the cluster.
- RDD Construction:
  - sc.parallelize()
  - sc.textFile() //HDFS, S3, local, URI
- Key Value Convention
- Transformations
- Actions
  - Actions may return huge data to the driver memory, so be careful
  - *Lazy Evaluation*: nothing happens until an action is called. So, it could be hard to debug code.
  - Every parent transformation is re-computed for each action operation, RDD persistence (different storage level) can be used.



(source: https://training.databricks.com/visualapi.pdf)

- Standard Code:

```
from pyspark import SparkConf, SparkContext

conf = SparkConf().setAppName("FakeCode")
sc = SparkContext(conf = conf)

lines_rdd = sc.textFile("s3a:///bucketname/filename.txt")
data_rdd = lines_rdd.map(line_to_key_paid_func)
results = data_rdd.reduceByKey(lambda x, y: x + y).collect()

print(results)
```

- Run code: spark-submit XXX.py

Spark SQL & DataFrame

- Designed for structured data, e.g. csv
- The trend: more DataFrame, less RDD
- DataFrame Construction:
  - spark.read.parquet()
  - spark.read.json()
  - spark.read.csv()
  - spark.read.load()
  - spark.read.text() #header as "value"
- DataFrame key Functions
  - cache / persist()
  - collect()
  - createTempView()
  - describe()
  - dropna / fillna()
  - filter()
  - groupBy() + agg()
    - df.groupBy('name').agg({'age': 'mean'}).collect()
  - join()
  - orderBy()
  - select()
  - toPandas() # to pandas DataFrame
  - withColumn() # weird name, to add col
- Built-in DataFrame functions
- Standard Code:

```
from pyspark.sql import SparkSession
from pyspark.sql import functions as func
from pyspark.sql.types import StructType, StructField, IntegerType, StringType

spark = SparkSession.builder.appName("FakeCode").getOrCreate()

schema = StructType([ StructField("id", IntegerType(), True), StructField("name", StringType(), True)])
df1 = spark.read.schema(schema).csv("s3a:///bucketname/filename.txt")
results = df1.sort(func.col("id").desc()).first()

print(results)
```

MLLib

- Do not use (algorithm quality is low)

PySpark Platform

- AWS EMR
- Databricks Cluster