SDSC Summer Institute
Scalable Machine Learning

SDSC SAN DIEGO SUPERCOMPUTER CENTER

UC San Diego

Scalable Machine Learning Agenda

8:30 - 10:00 – R in HPC

10:00 - 10:15 – Break

10:15 - 10:45 – Machine Learning with Spark

10:45 - 11:15 – PySpark Hands-On

11:15 - 11:45 – SparkR Hands-On

11:45 - 12:00 – Wrap-Up

Machine Learning with Spark

Mai H. Nguyen, Ph.D.

Spark Topics

- Spark Overview
- Programming in Spark
- MLlib

Spark Overview

What is Spark?

- **General framework for distributed computing**
- **Provides built-in data parallelism and fault-tolerance for big data processing on a cluster**
- **Goals: speed, ease of use, generality**
  - Multiple analytics applications, data sources, platforms
- **Open-source**

Basics of Distributed Processing with Spark

Expressive programming environment

In-memory processing

Support for diverse workloads

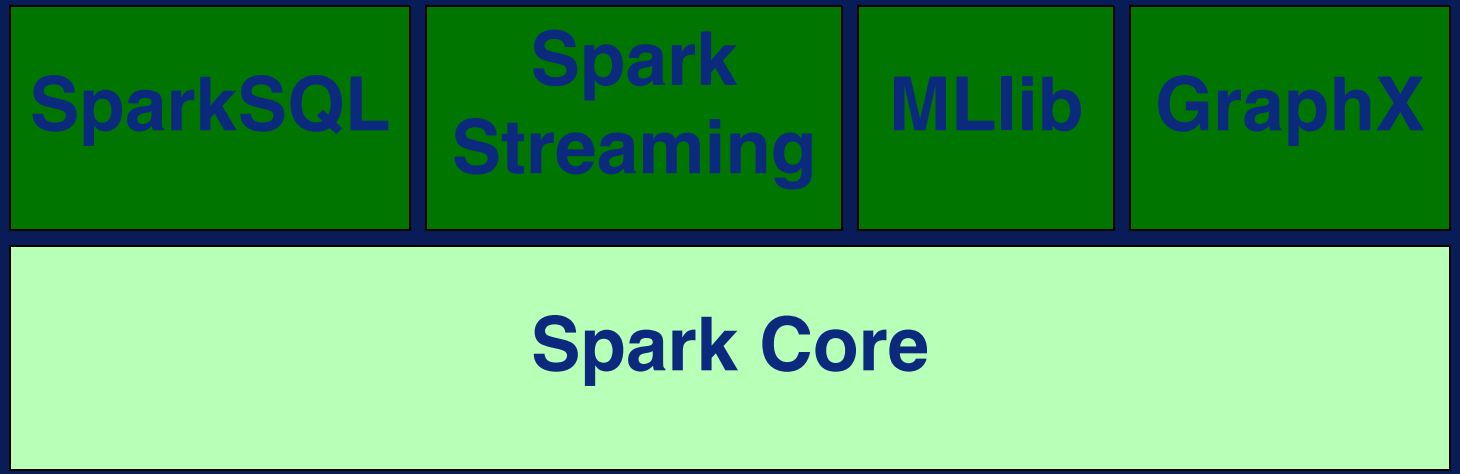
Interactive shell

The Spark Stack

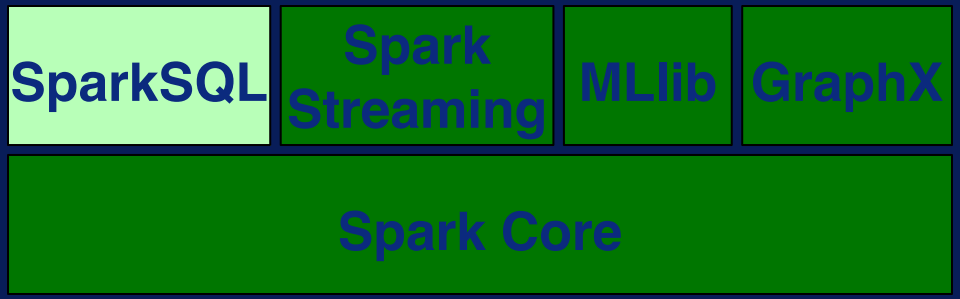
SparkSQL	Spark Streaming	MLlib	GraphX

Spark Core

The Spark Stack



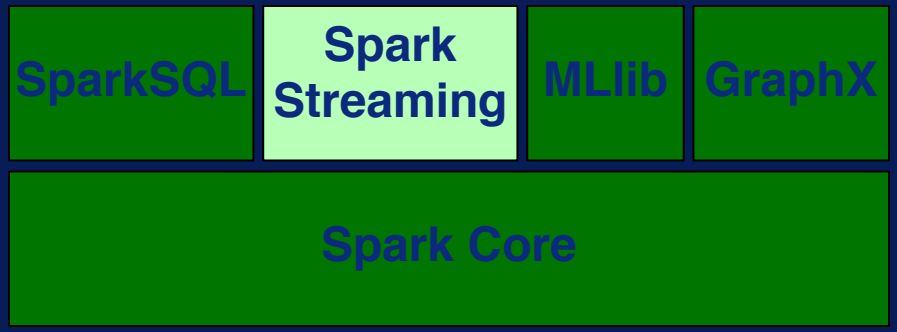
Distributed computing



Spark SQL

Structured Data Processing

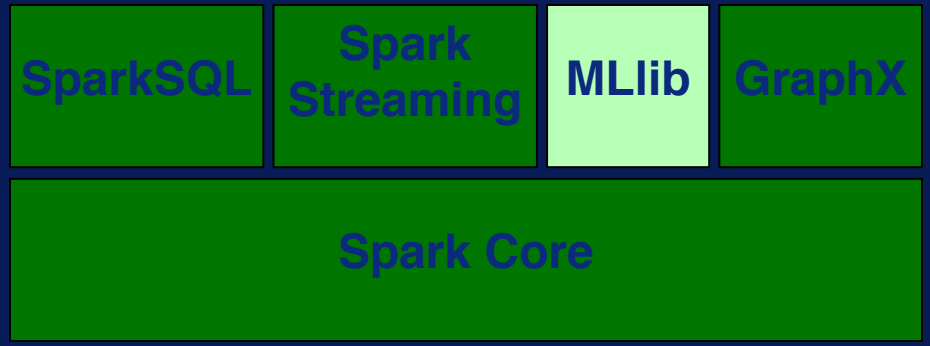
- Enables querying structured data through Spark
- Can use SQL and Hive Query Language
- Has APIs for Scala, Java, Python, and R
- Embed SQL queries in Spark programs



Spark Streaming

Streaming Data Processing

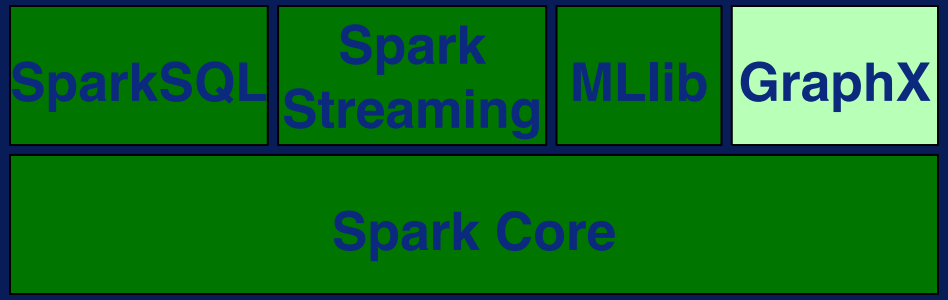
- Scalable processing for real-time analytics
- Data streams divided into micro-batches of data
- Has APIs for Scala, Java, and Python



Spark MLlib

Machine Learning

- Scalable machine learning library
- Provides distributed implementations of common machine learning algorithms and utilities
- Has APIs for Scala, Java, Python, and R



Spark GraphX

Graph Computation

Enables distributed graph processing.

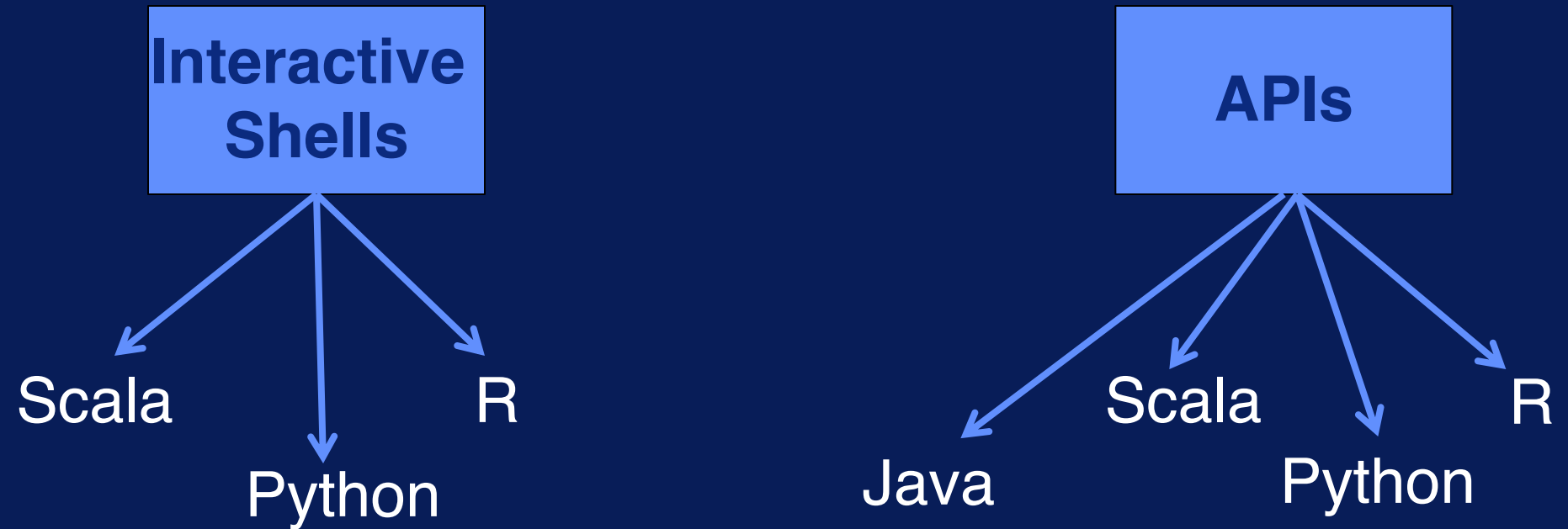
The Spark Stack

SparkSQL	Spark Streaming	MLlib	GraphX

Spark Core

Supports diverse analytics applications

Spark Interface

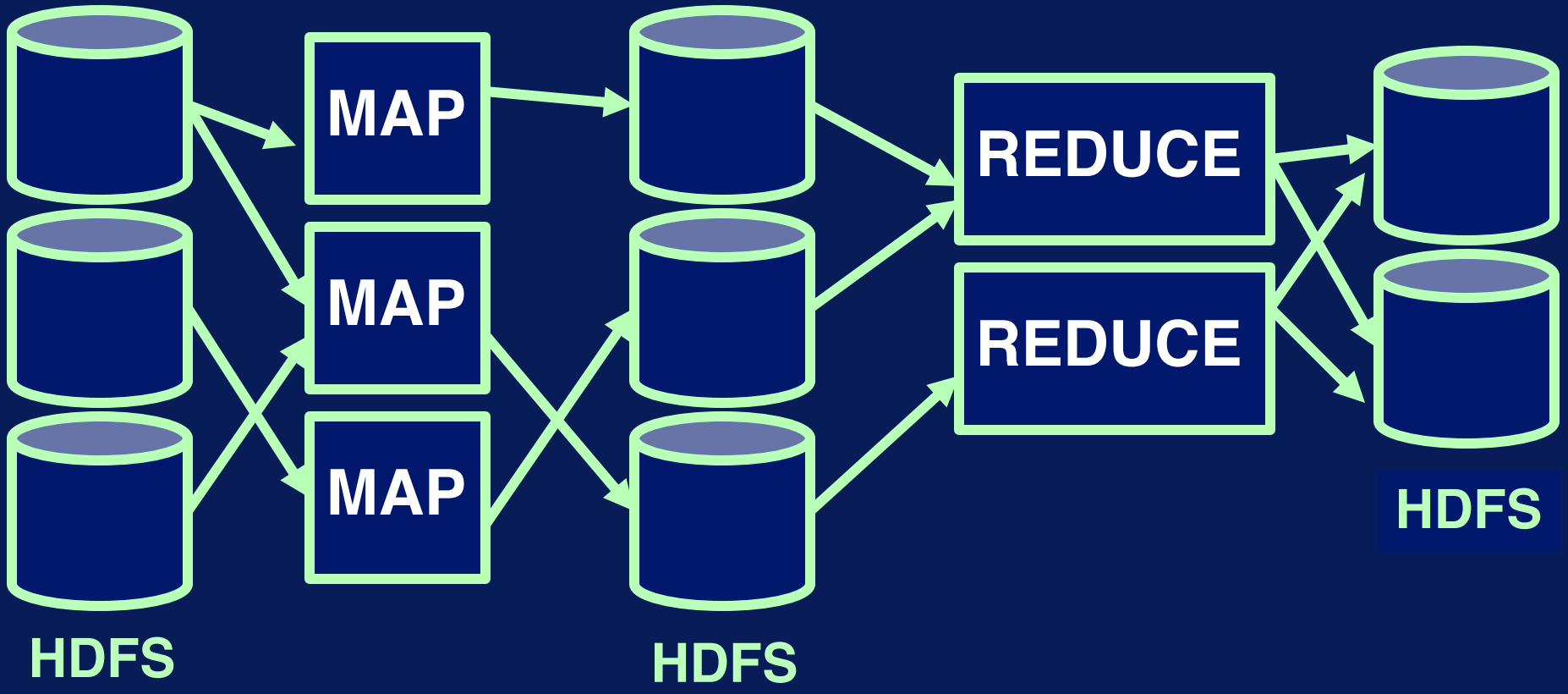


Provides ease of use

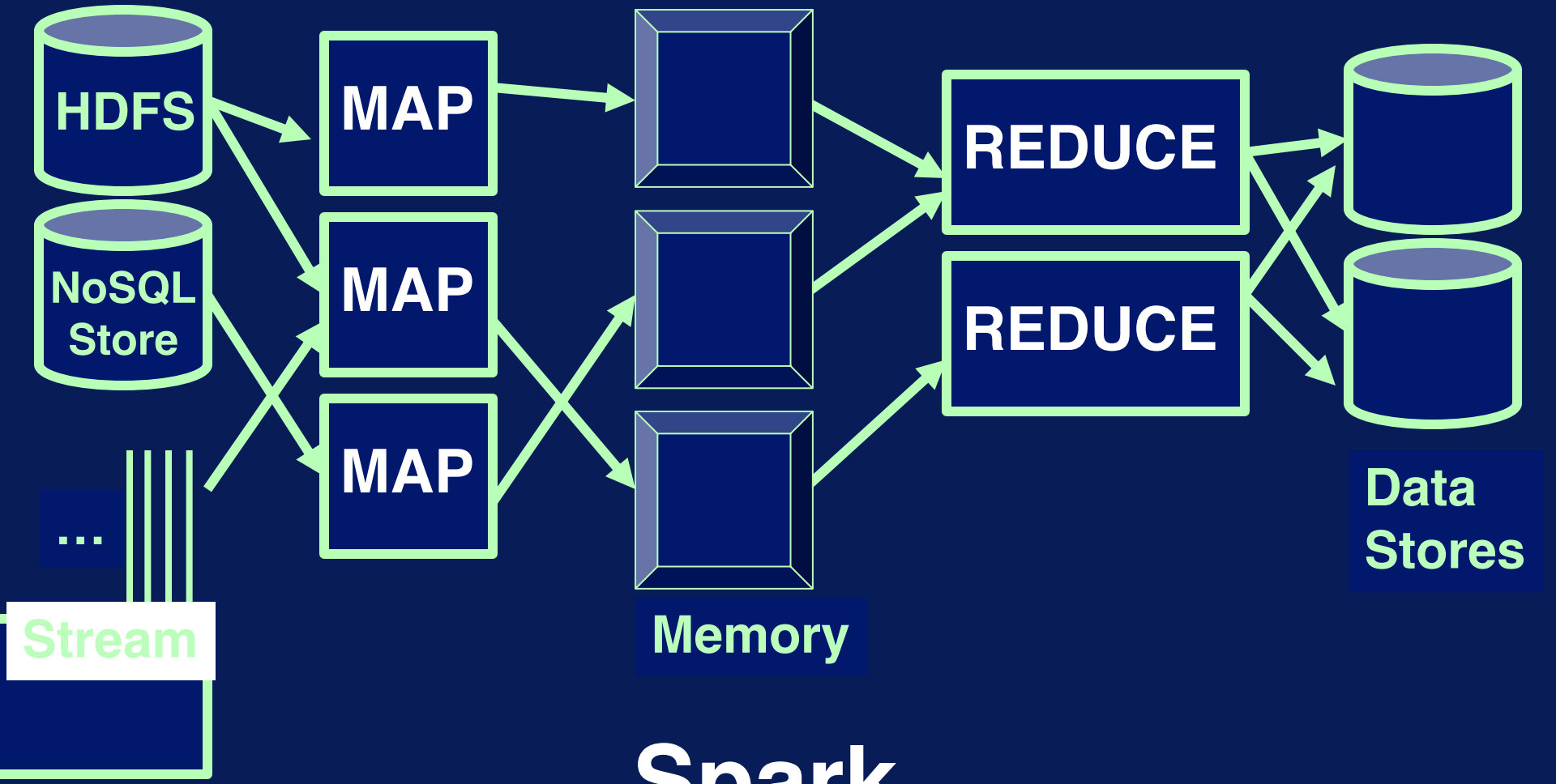
In Memory Processing

Provides speed

What does in memory processing mean?



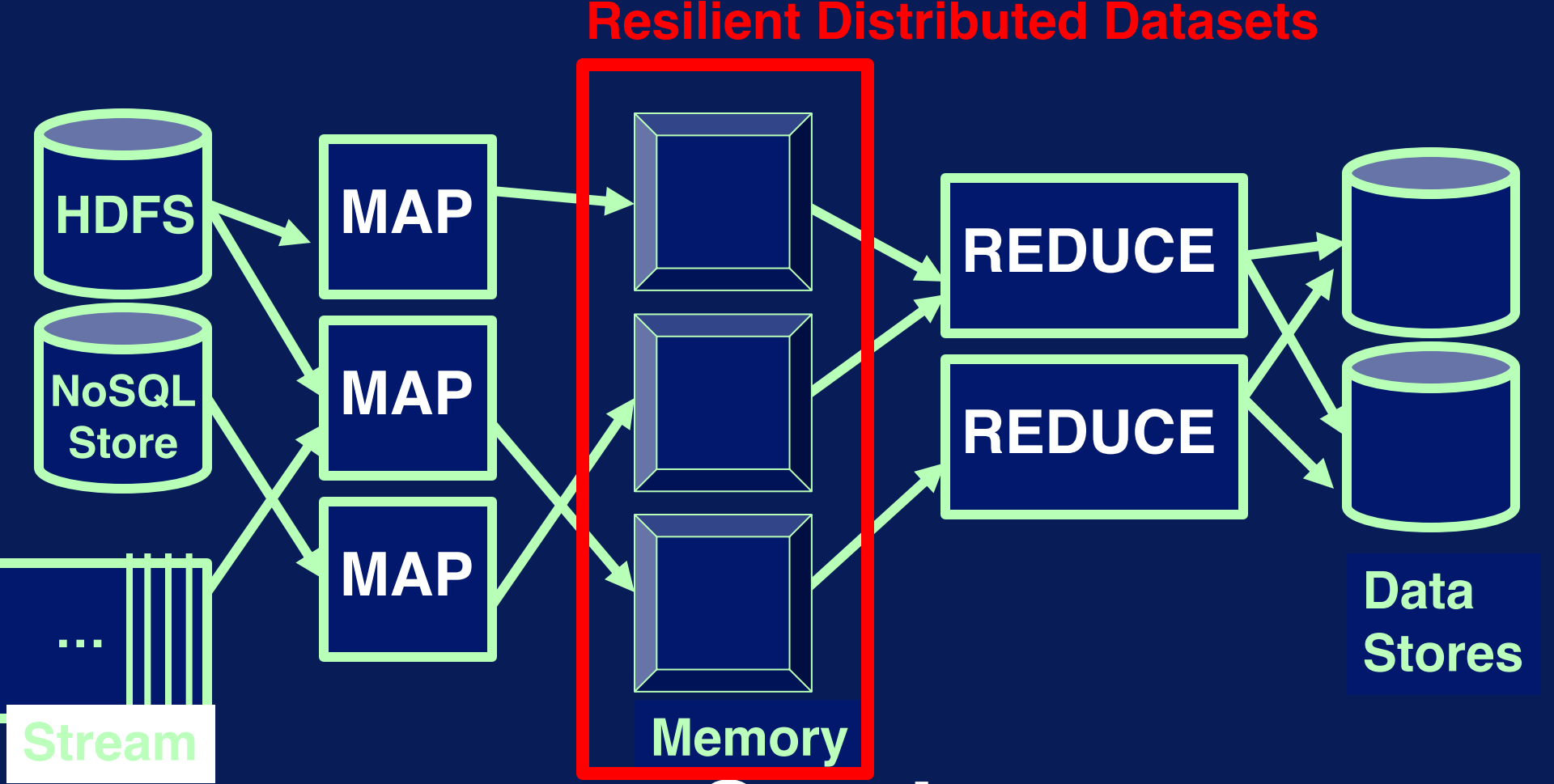
MapReduce



Spark

SDSC SAN DIEGO SUPERCOMPUTER CENTER

UC San Diego



SDSC SAN DIEGO SUPERCOMPUTER CENTER

Spark

UC San Diego

Resilient Distributed Datasets

Dataset

Data storage created from: HDFS, S3, HBase, JSON, text, Local hierarchy of folders

Or created transforming another RDD

Resilient Distributed Datasets

Distributed

Distributed across the cluster of machines

Divided in partitions, atomic chunks of data

Resilient Distributed Datasets

Resilient

Recover from errors, e.g. node failure, slow processes

Track history of each partition, re-run

DataFrames & DataSets

DataFrame

DataSet

- **Extensions to RDDs**
- **Provide higher-level abstractions, improved performance, better scalability**

Programming in Spark

Start Spark Session

Driver Program

```
from pyspark.sql import SparkSession

spark = SparkSession \
    .builder \
    .appName ("PySpark Example") \
    .config("config.option","config.value") \
    .getOrCreate()
```

Read in Data

```
df = spark.read.csv("data.csv"),\
   inferSchema=True,header=True)
```

Read in Data

```
df = spark.read.csv("data.csv"),\
   inferSchema=True,header=True)
```

```
df = spark.read.jdbc \
 (url=my_url, \
  dbtable=table_name, \
  user=username,password=pwd)
```

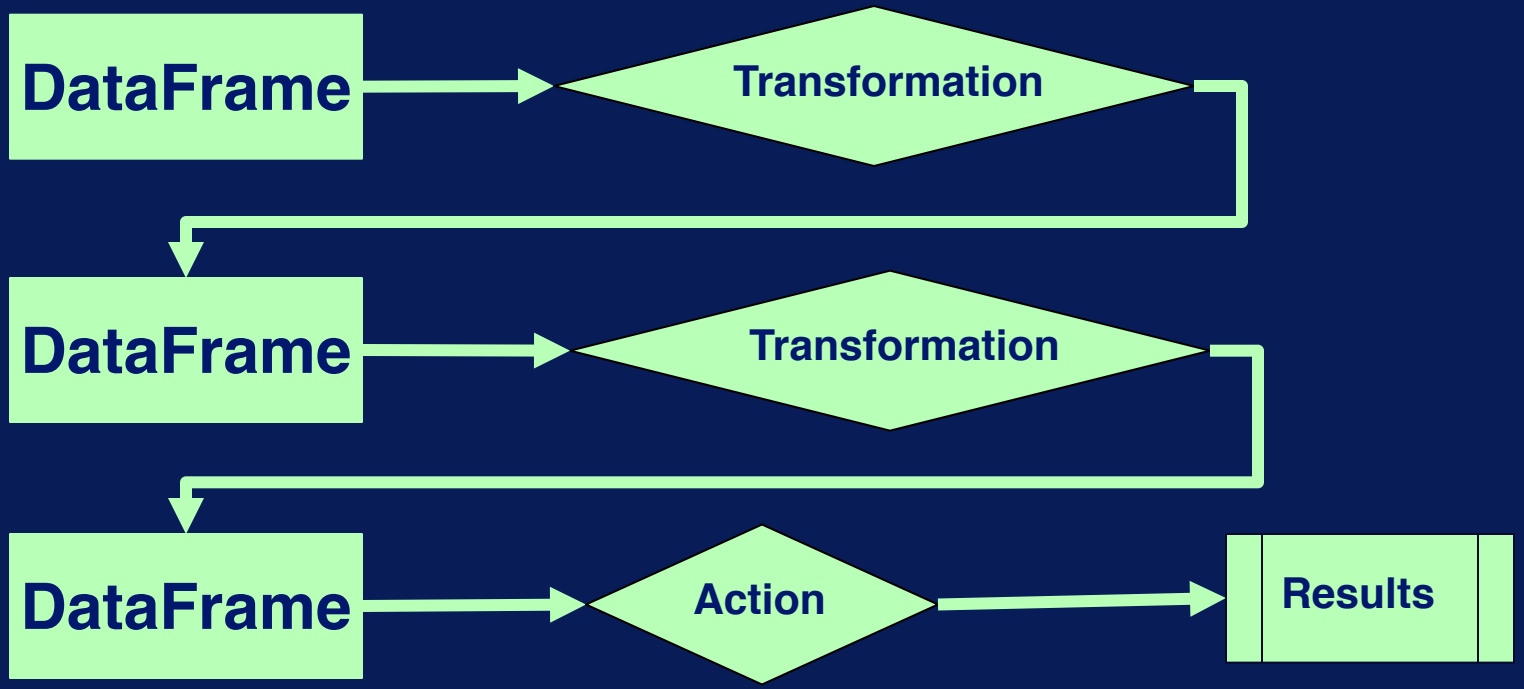
Read in Data

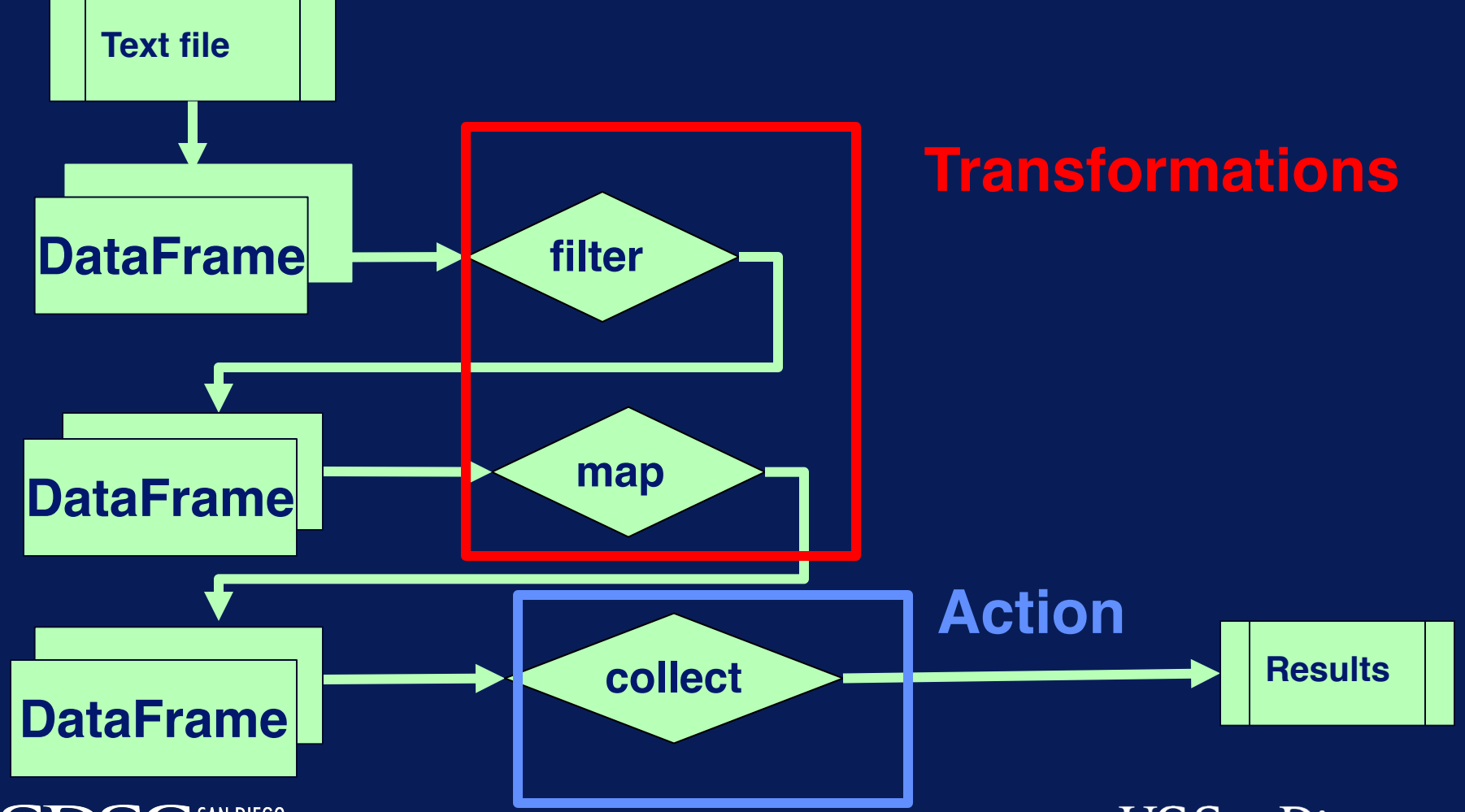
```
df = spark.read.csv("data.csv"),\
   inferSchema=True,header=True)
```

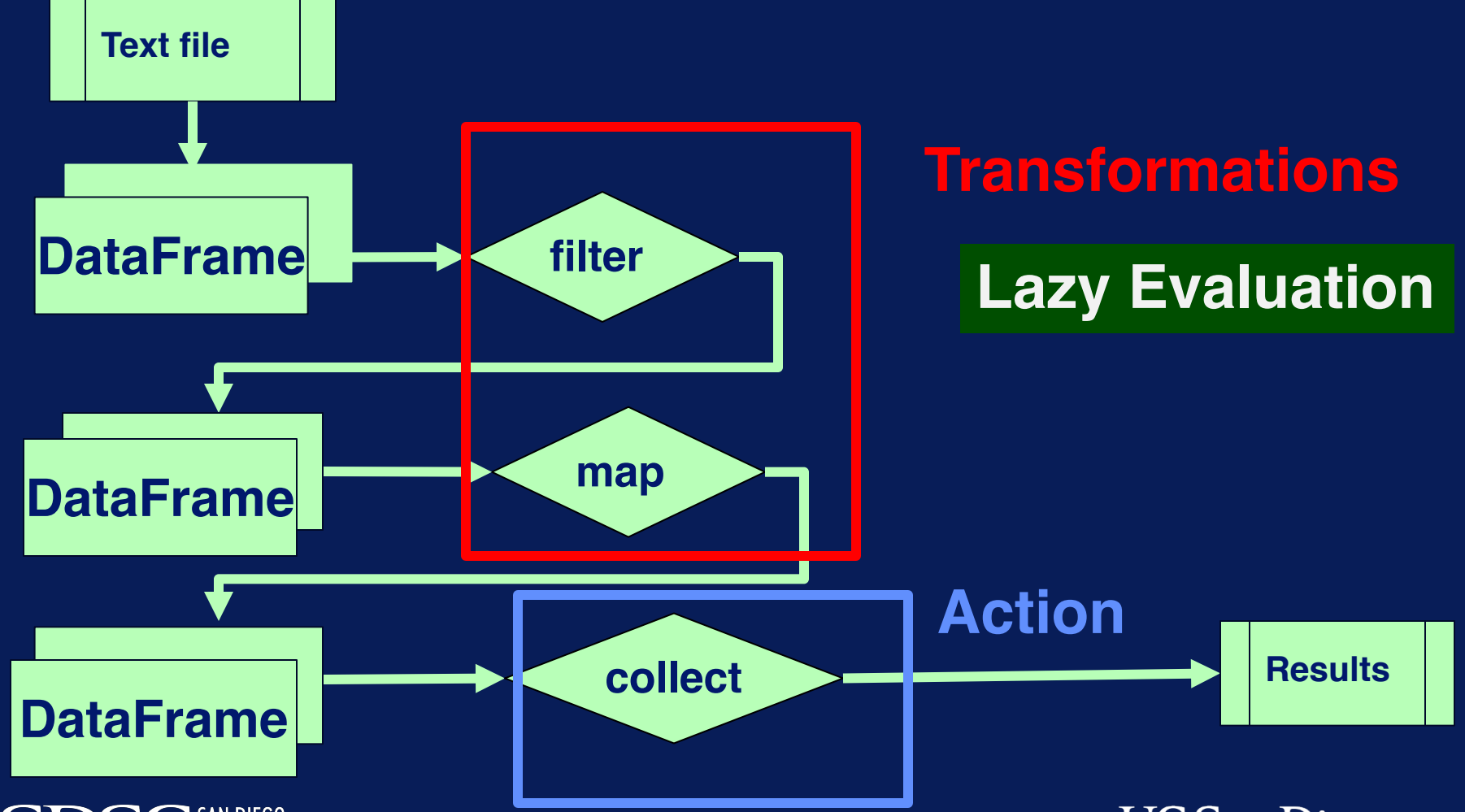
```
df = spark.read.jdbc \
 (url=my_url, \
  dbtable=table_name, \
  user=username,password=pwd)
```

```
empl_0 = Row(id='123',name='John')
empl_1 = Row(id='456',name='Mary')
employees = [empl_0, empl_1]
df = spark.createDataFrame(employees)
```

Processing Data




Text file
DataFrame
filter
DataFrame
map
DataFrame
collect
Results
Transformations
Action


Text file
DataFrame
filter
DataFrame
map
DataFrame
collect
Results
Transformations
Lazy Evaluation
Action

Lazy Evaluation

- Transformations not immediately processed
- Plan of transformations is built
- Transformations executed when action is performed.
- Allows for efficient physical plan to be generated

Transformations & Actions

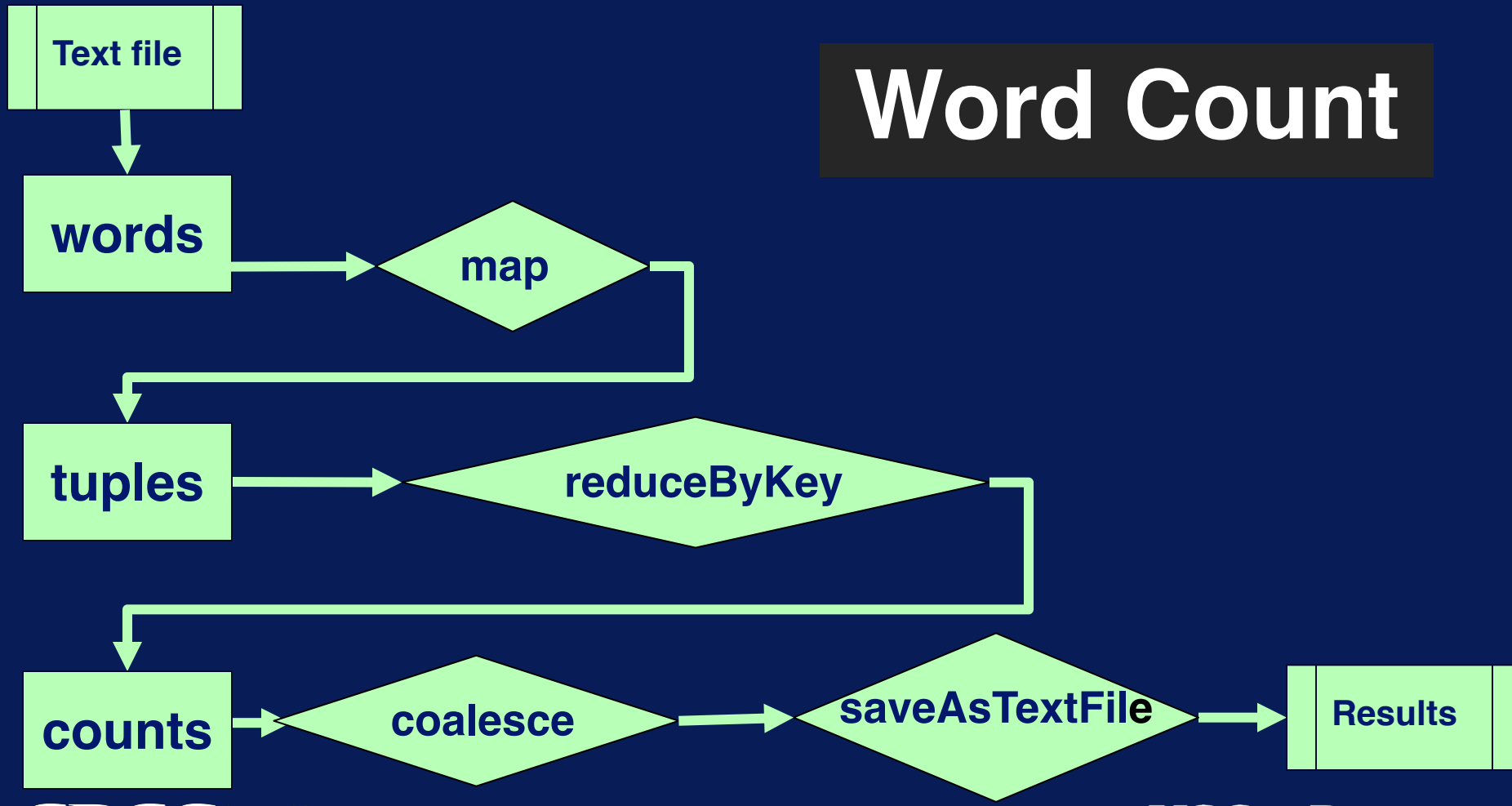
Transformations

- map
- filter
- coalesce
- reduceByKey

Actions

- take
- collect
- reduce
- saveAsText

Word Count

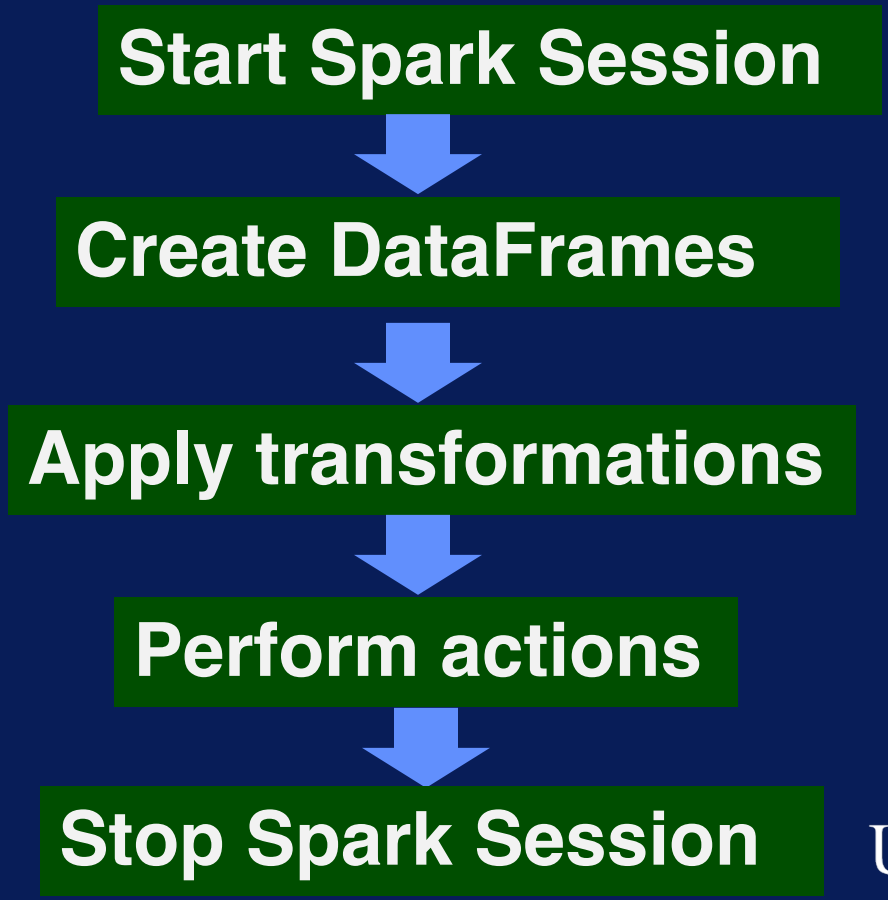

Text file
words
map
tuples
reduceByKey
counts
coalesce
saveAsTextFile
Results

Stop Spark Session

Driver Program

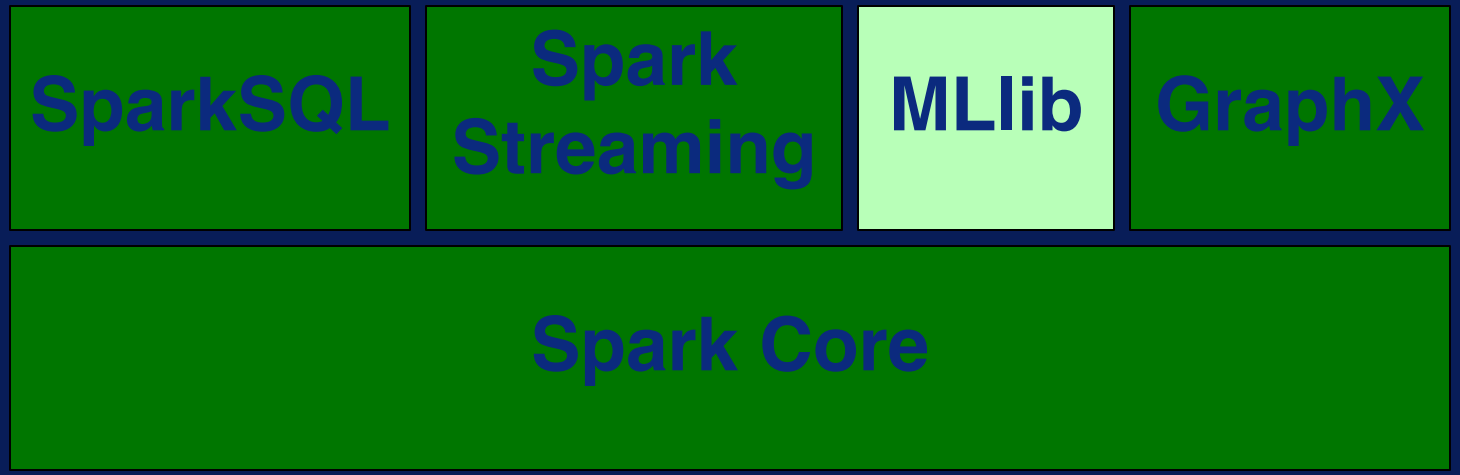
```
spark = stop()
```

Programming in Spark



Spark MLlib: Machine Learning

Spark MLlib



Machine learning

MLlib Algorithms & Techniques

- Machine Learning
  - Classification, regression, clustering, etc.
  - Evaluation metrics
- Statistics
  - Summary statistics, sampling, etc.
- Utilities
  - Dimensionality reduction, transformation, etc.

MILib Example –Statistics

```
from pyspark.sql.functions import rand

# Generate random numbers
df = sqlContext.range(0,10)
                .withColumn('rand1', rand(seed=10))
                .withColumn('rand2', rand(seed=27))


# Show summary statistics
df.describe().show()


# Compute correlation
df.stat.corr('rand1','rand2')
```

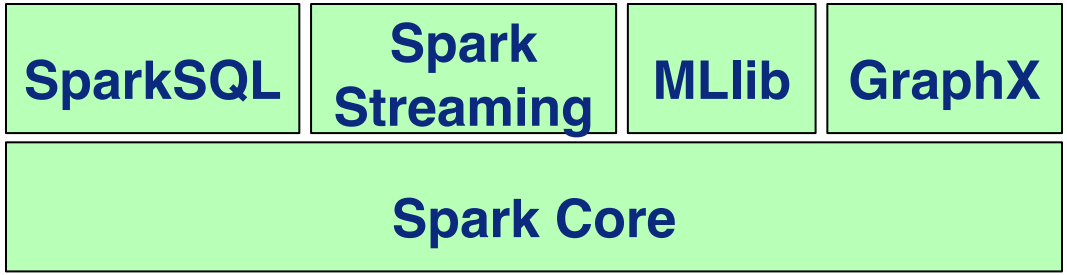
MLlib Example – Clustering

```
from pyspark.ml.clustering import KMeans

# Read and parse data
data = spark.read.csv("data.csv",inferSchema="true",
                      header="true")


# k-means model for clustering
kmeans = Kmeans().setK(3).setSeed(123)
model = kmeans.fit (data)
for center in model.clusterCenters()
   print (center)
```

Spark MLlib

- MLlib is Spark's machine learning library.
  - Distributed implementations
- Main categories of algorithms and techniques:
  - Machine learning
  - Statistics
  - Utilities for data preparation

Scalable Machine Learning Summary



- Spark core provides distributed computing
- Libraries support multiple analytics applications and workloads
- RDD/DF/DS provide data parallelism & fault-tolerance
- MLlib provides scalable machine learning

Spark Resources

- Spark
  - https://spark.apache.org/
- MLlib
  - https://spark.apache.org/mllib/
- Mastering Apache Spark
  - https://jaceklaskowski.gitbooks.io/mastering-apache-spark/content/

Questions?