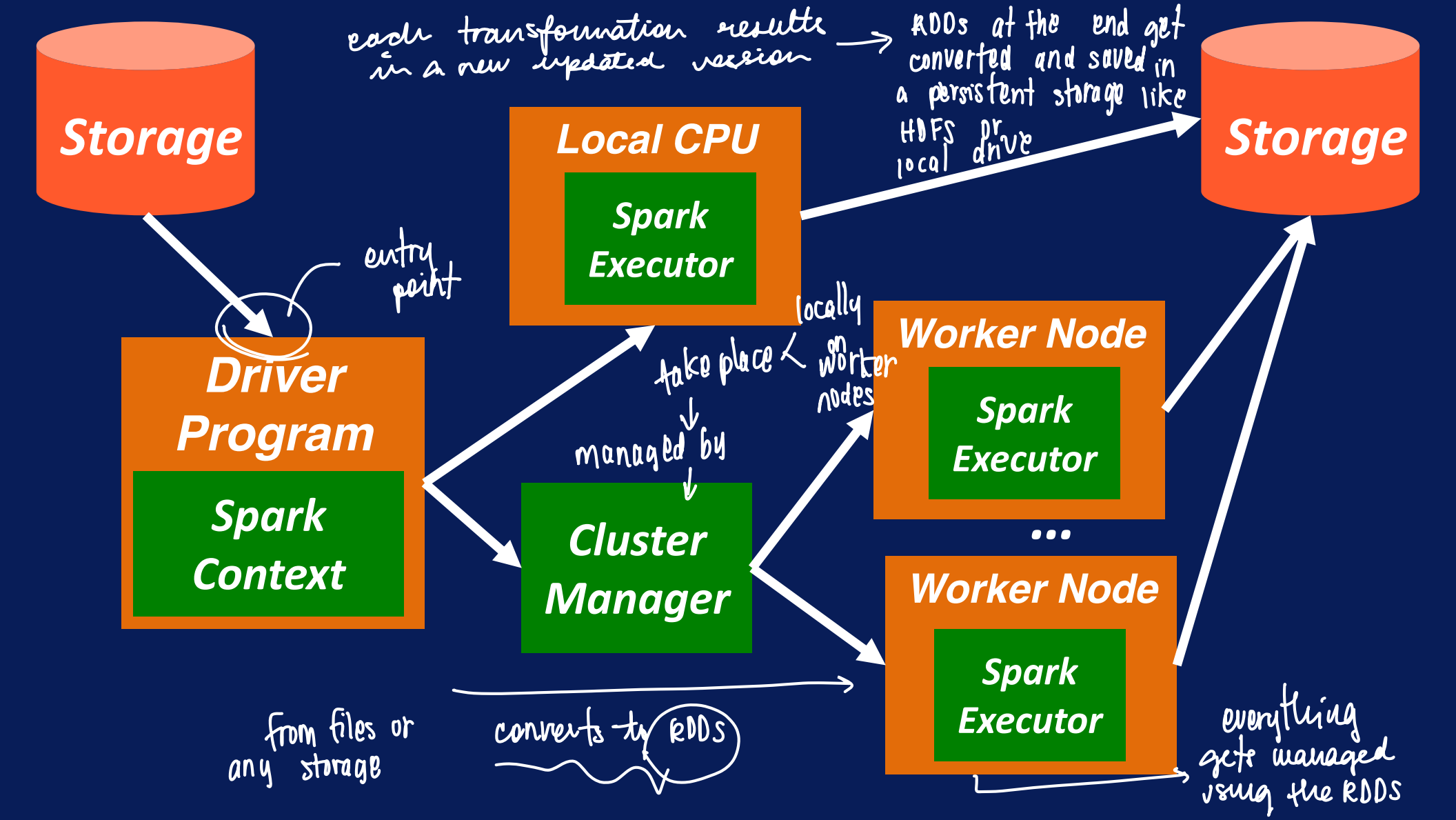
Spark Core:
Programming In Spark

After this video you will be able to..

- Use two methods to create RDDs in Spark
- Explain what immutable means
- Interpret a Spark program as a pipeline of transformations and actions
- List the steps to create a Spark program


Storage
each transformation results in a new updated version
RDDs at the end get converted and saved in a persistent storage like HDFS or local drive
Storage
Local CPU
Spark Executor
entry point
Driver Program
Spark Context
take place locally on worker nodes
managed by
Cluster Manager
Worker Node
Spark Executor
...
Worker Node
Spark Executor
from files or any storage
converts to RDDs
everything gets managed using the RDDs

Creating RDDs in Driver Program

you can just read files →

```
In [1]: lines = sc.textFile("hdfs:/user/cloudera/words.txt")
```

```
lines = sc.parallelize(["big", "data"])
```

provide existing collection (like list) to be turned into a distributed collection

```
numbers = sc.parallelize(range(10), 3)
```

integer RDD + provide number of partitions

creates 3 partitions

[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

Parallelize range output into 3 partitions

[0, 1, 2], [3, 4, 5], [6, 7, 8, 9]

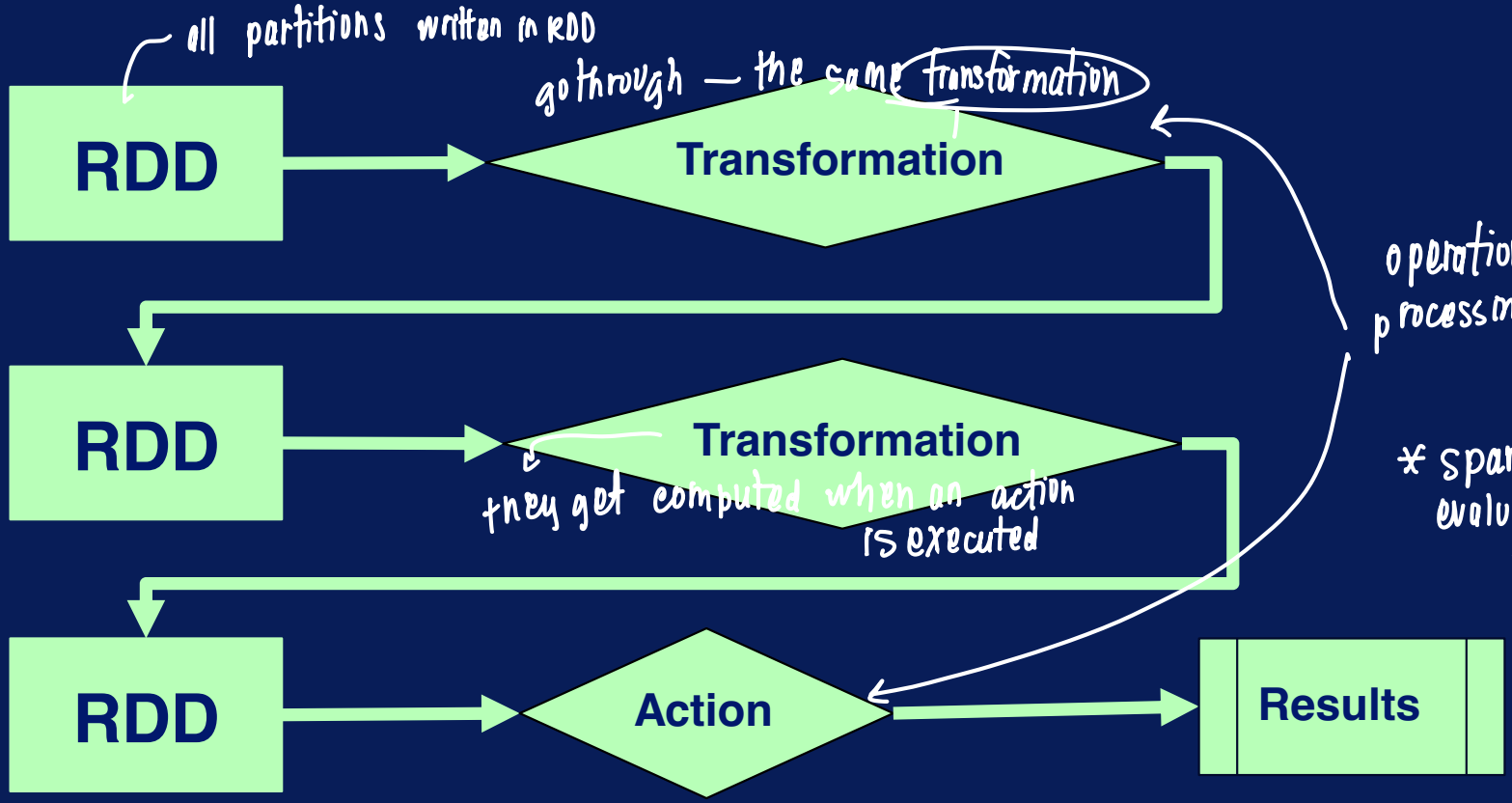
* spark decides how to assign partitions to our executors and worker nodes

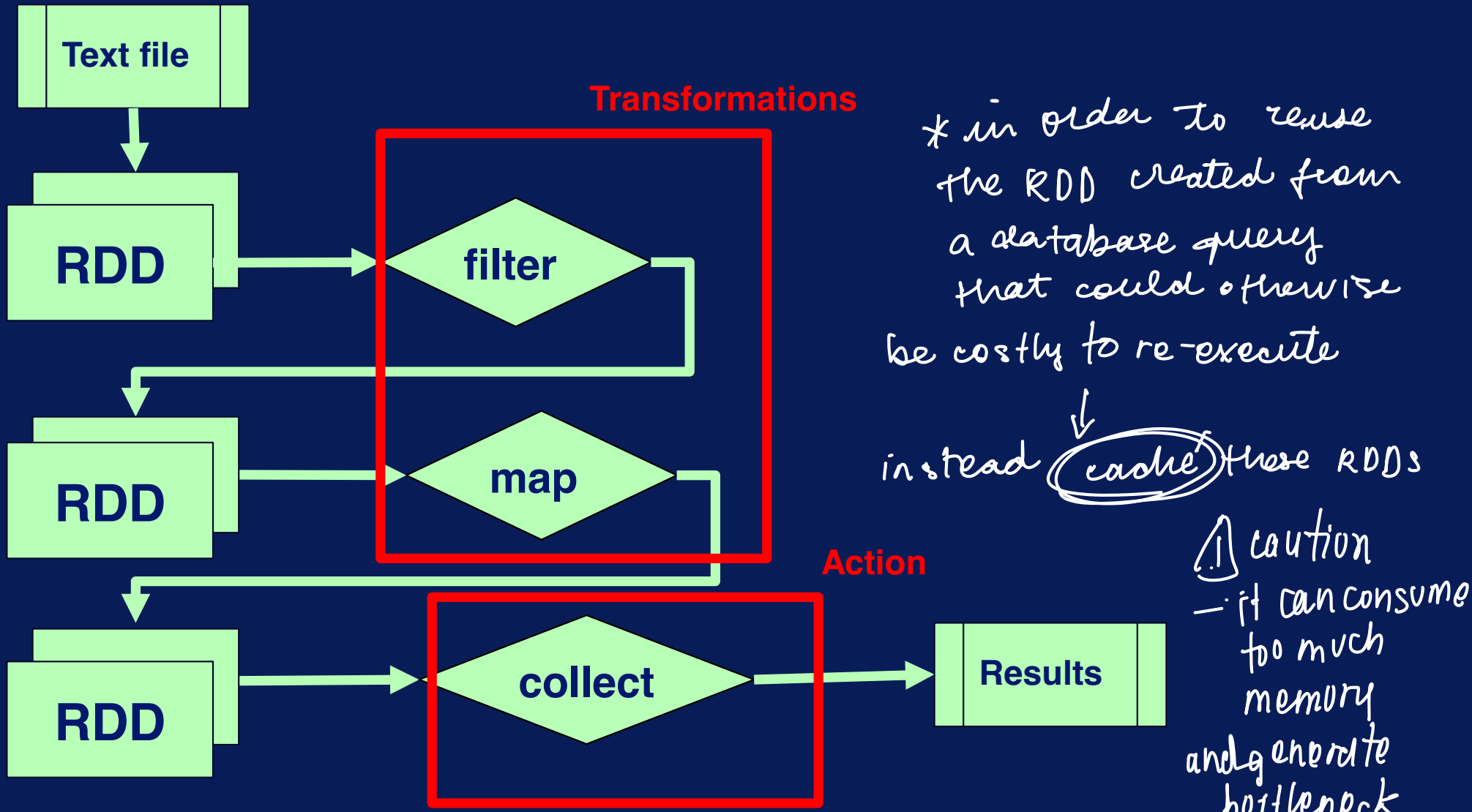
```
numbers.collect()
```

[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

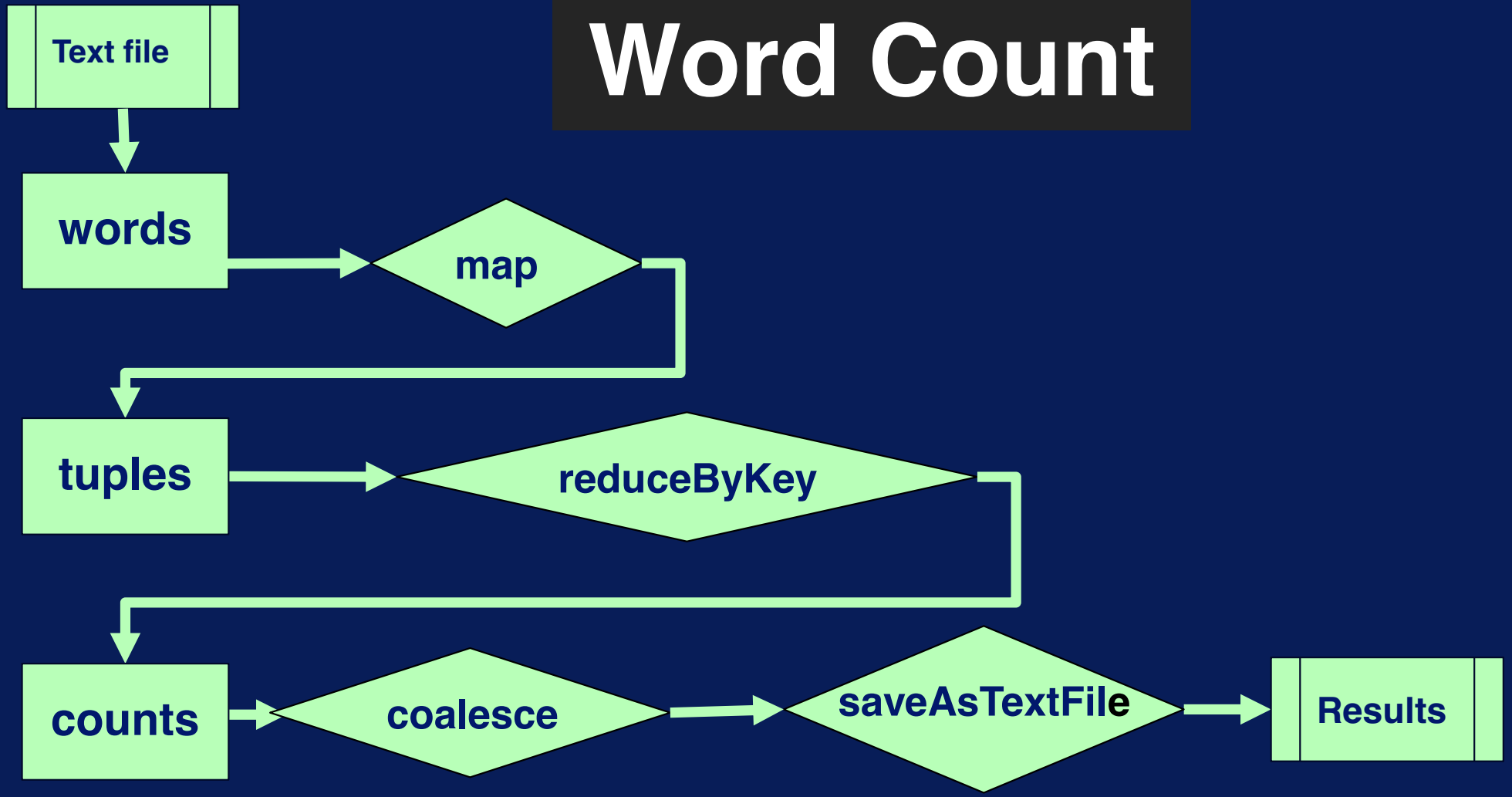
can then be gathered into a single partition on the driver using the collect transformat

Processing RDDs




Text file
RDD
RDD
RDD
Transformations
filter
map
Action
collect
Results
* in order to reuse the RDD created from a database query that could otherwise be costly to re-execute
instead cache these RDDs
caution
- it can consume too much memory and generate bottleneck

Word Count


Text file
words
map
tuples
reduceByKey
counts
coalesce
saveAsTextFile
Results

Programming in Spark