

Spark Core

Part 1



Justin Pihony

@JustinPihony | justin-pihony.blogspot.com

Spark Core Maintainers



Matei Zaharia



Reynold Xin



Patrick Wendell

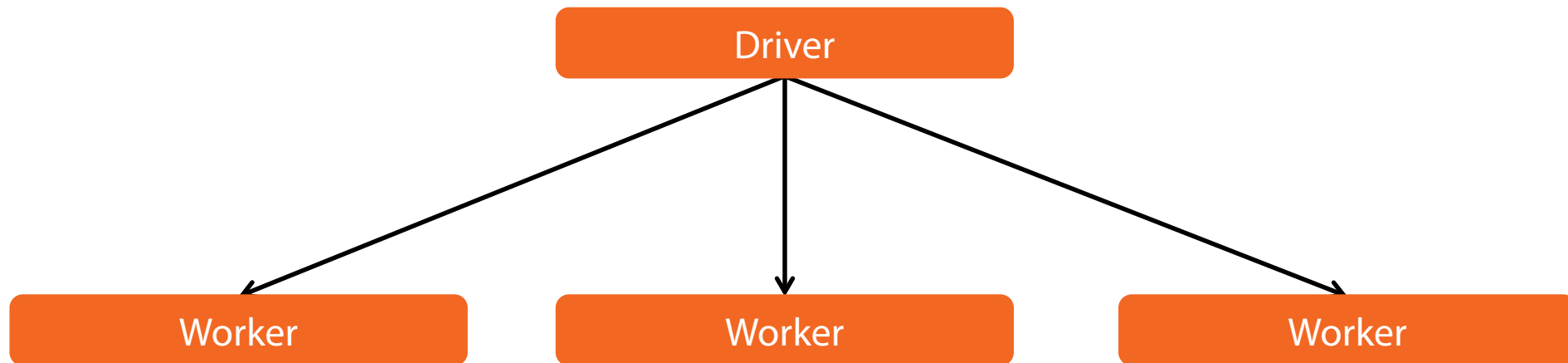


Josh Rosen

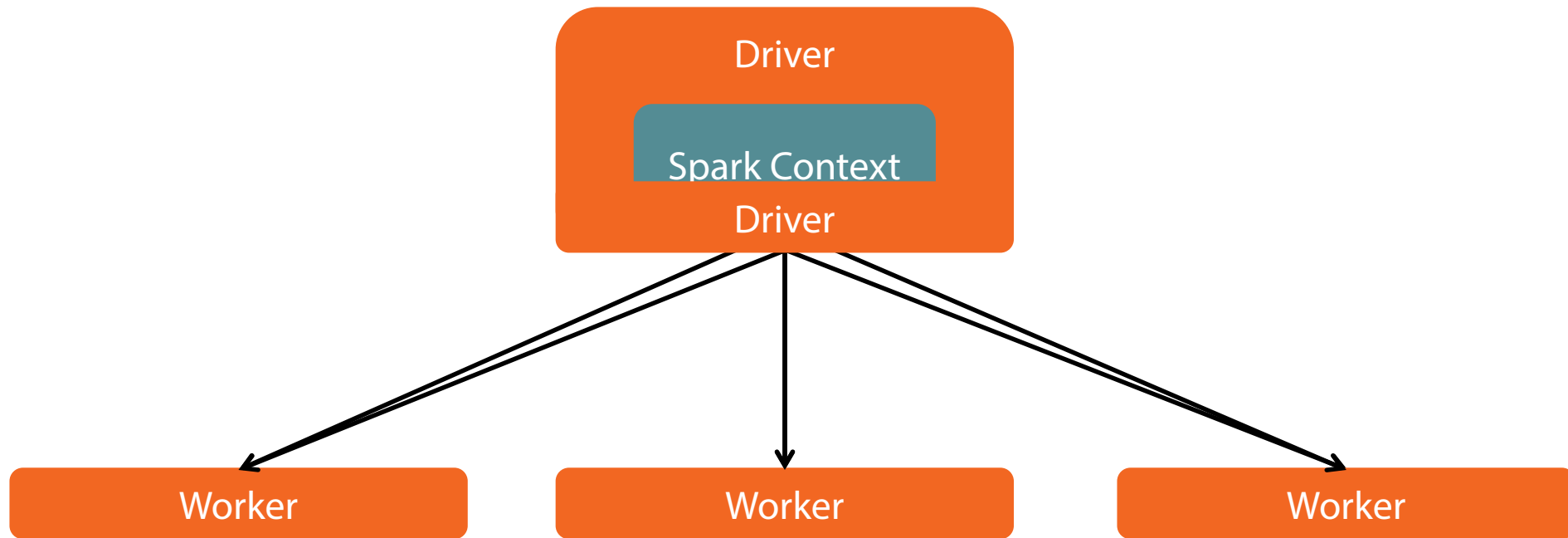
Course Overview

- Basics of Spark
- Core API
- Cluster Managers
- Spark Maintenance
- Libraries
 - SQL
 - Streaming
 - MLlib/GraphX
- Troubleshooting / Optimization
- Future of Spark

Spark Mechanics



Spark Mechanics



Spark Context



Task creator

Scheduler

Data locality

Fault tolerance

RDD

...collection of elements partitioned across the nodes of the cluster that can be operated on in parallel...

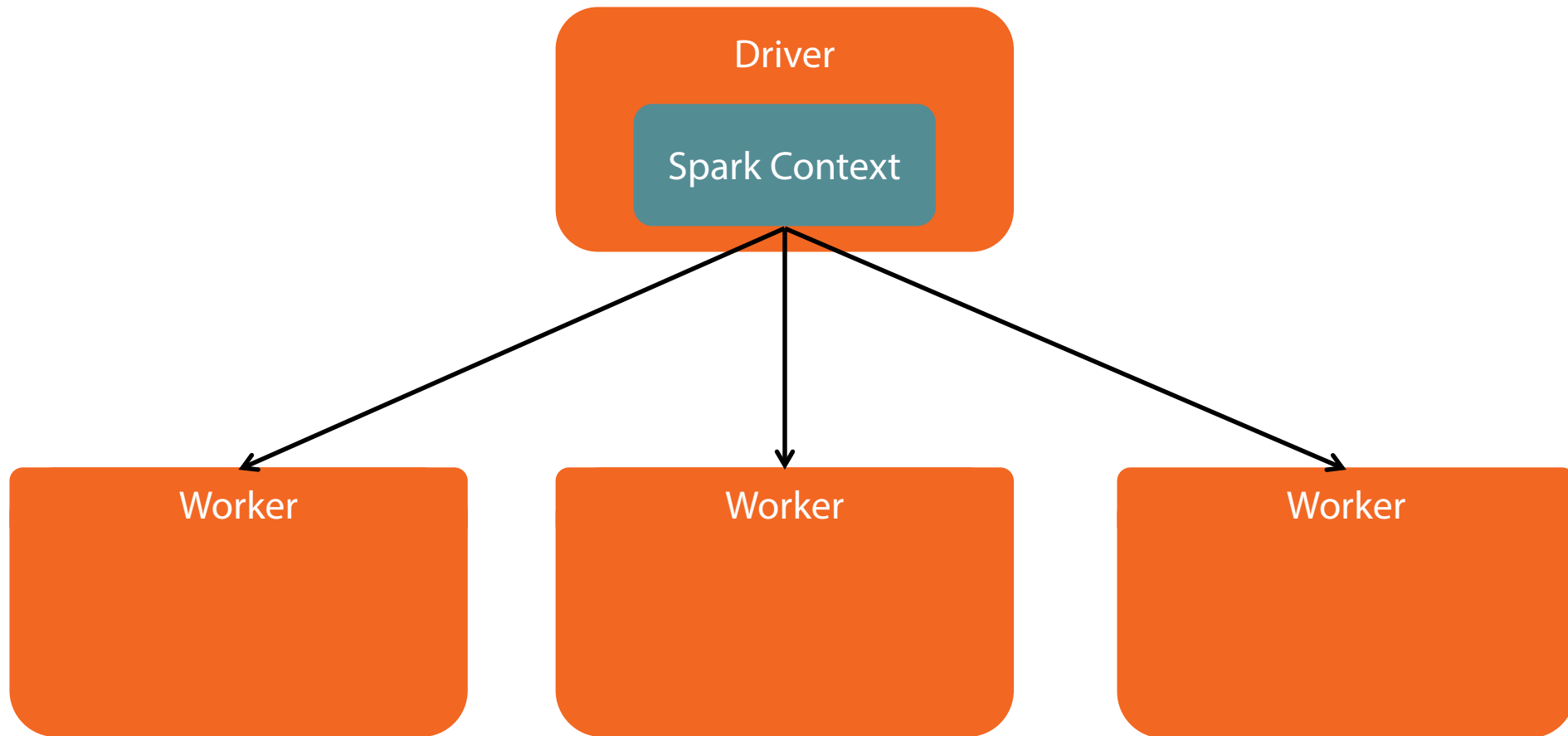
<https://spark.apache.org/docs/latest/programming-guide.html#overview>

RDD

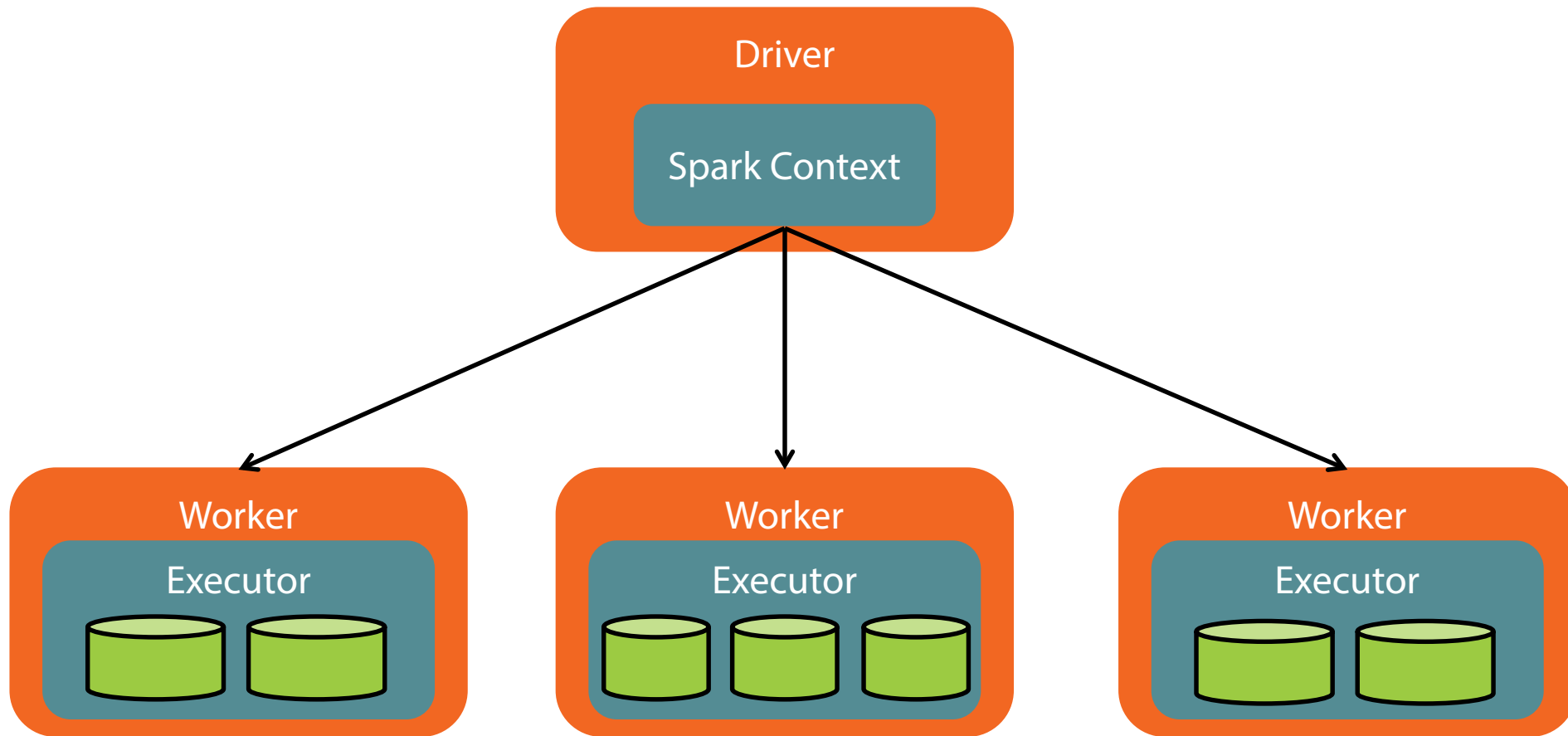
- Resilient Distributed Dataset
- DAG
- Transformations
 - map
 - filter
 - ...
- Actions
 - collect
 - count
 - reduce
 - ...



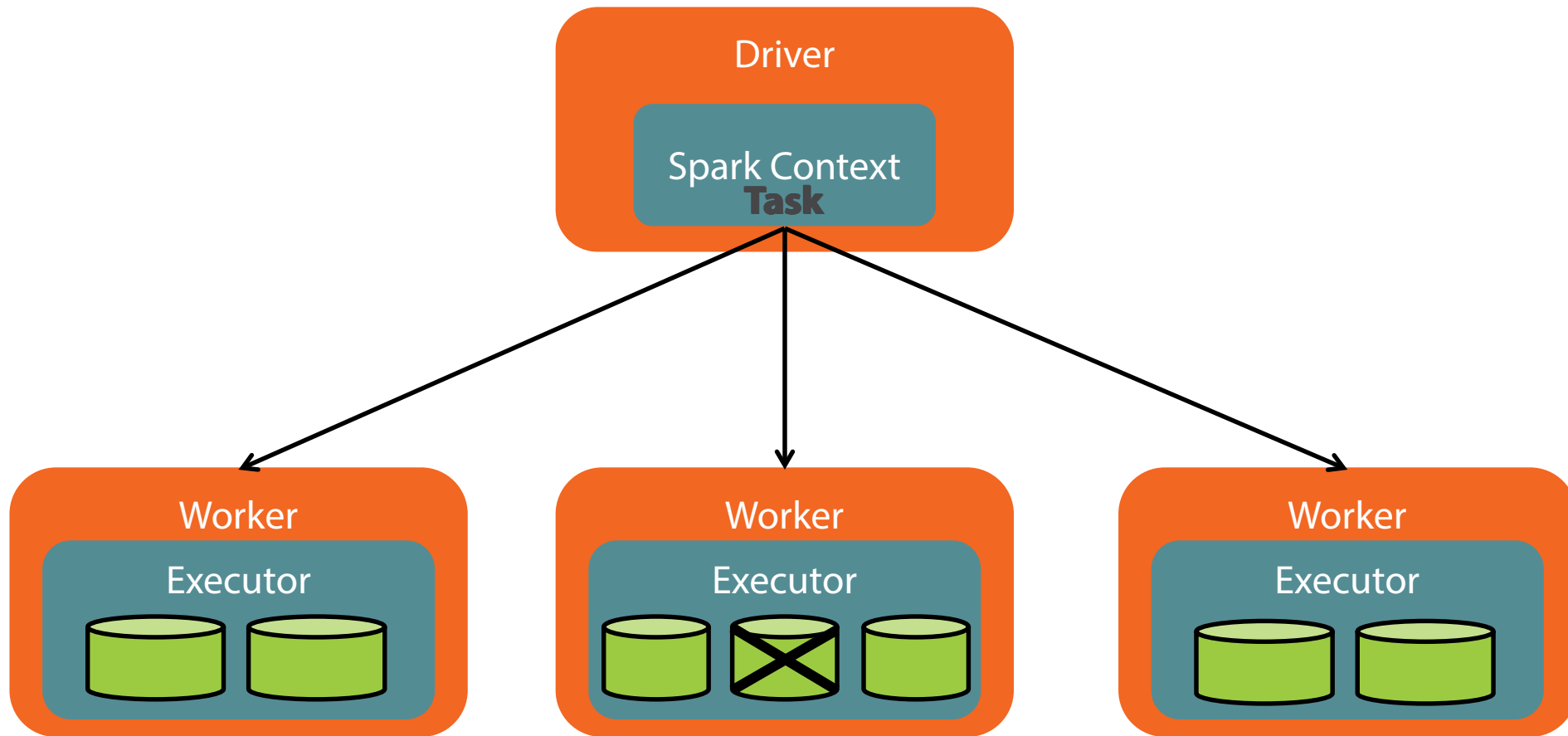
Spark Mechanics



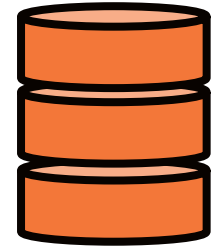
Spark Mechanics



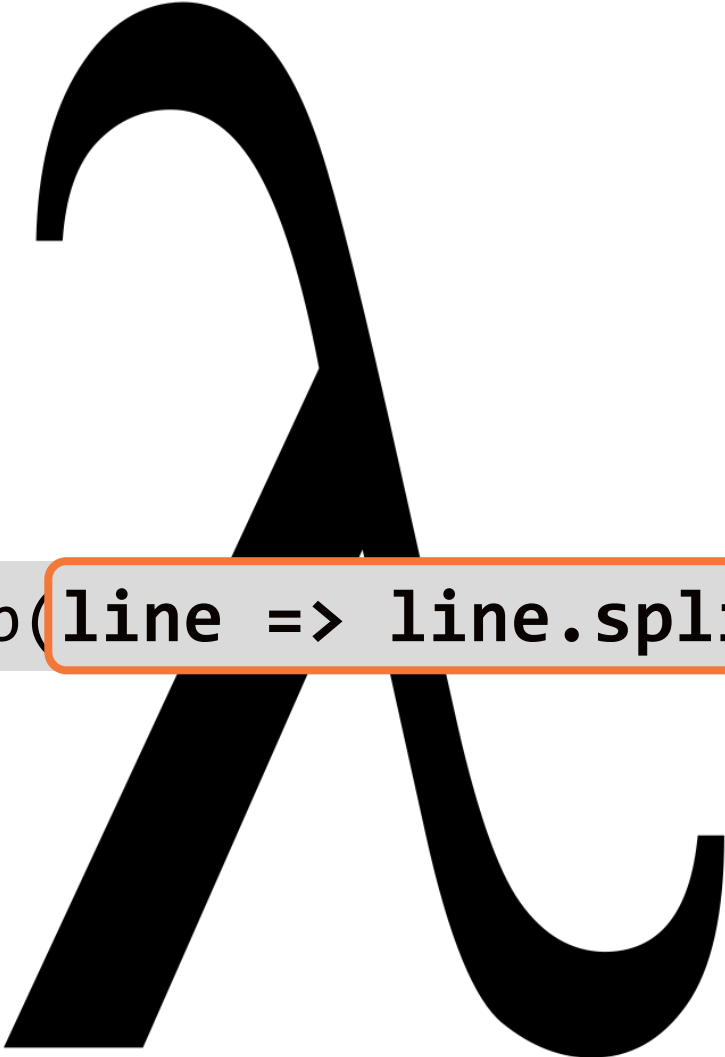
Spark Mechanics



Input



S3



```
rdd.flatMap(line => line.split(" "))
```

Named Method

```
def addOne(item: Int) = {  
    item + 1  
}
```

```
val intList = List(1,2)  
  
for(item <- intList) yield {  
    addOne(item)  
} // List(2,3)
```

Labeled Methods

```
def addOne(item: Int) = {  
    item + 1  
}
```

```
val intList = List(1,2)
```

```
intList.map(x => {  
    addOne(x)  
})//List(2,3)
```

Lambda Functions

```
val intList = List(1,2)  
intList.map(item => item + 1)//List(2,3)
```


Lambda Functions

```
val intList = List(1,2)  
intList.map(item => item + 1)//List(2,3)
```

Lambda Functions

```
val intList = List(1,2)
```

```
intList.map(item => item + 1) // List(2,3)
```

```
def addOne(item: Int) = {  
    item + 1  
}
```

```
val intList = List(1,2)
```

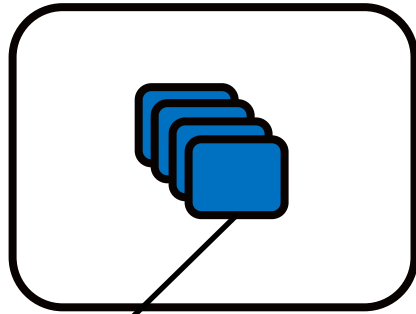
```
for(item <- intList) yield {  
    addOne(item)  
} // List(2,3)
```

Transformations

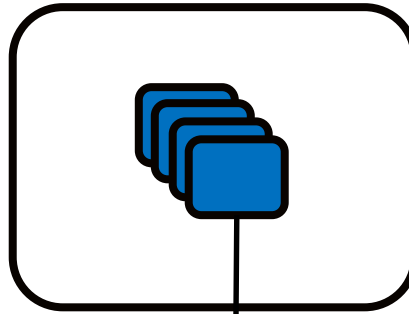


map

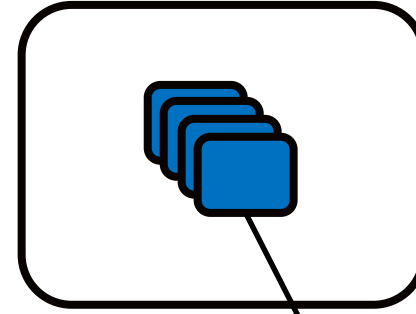
Node1



Node2



NodeN



```
for(item <-items) {
```

```
  yield mapFunction item
```

```
}
```

```
for(item <-items) {
```

```
  yield mapFunction
```

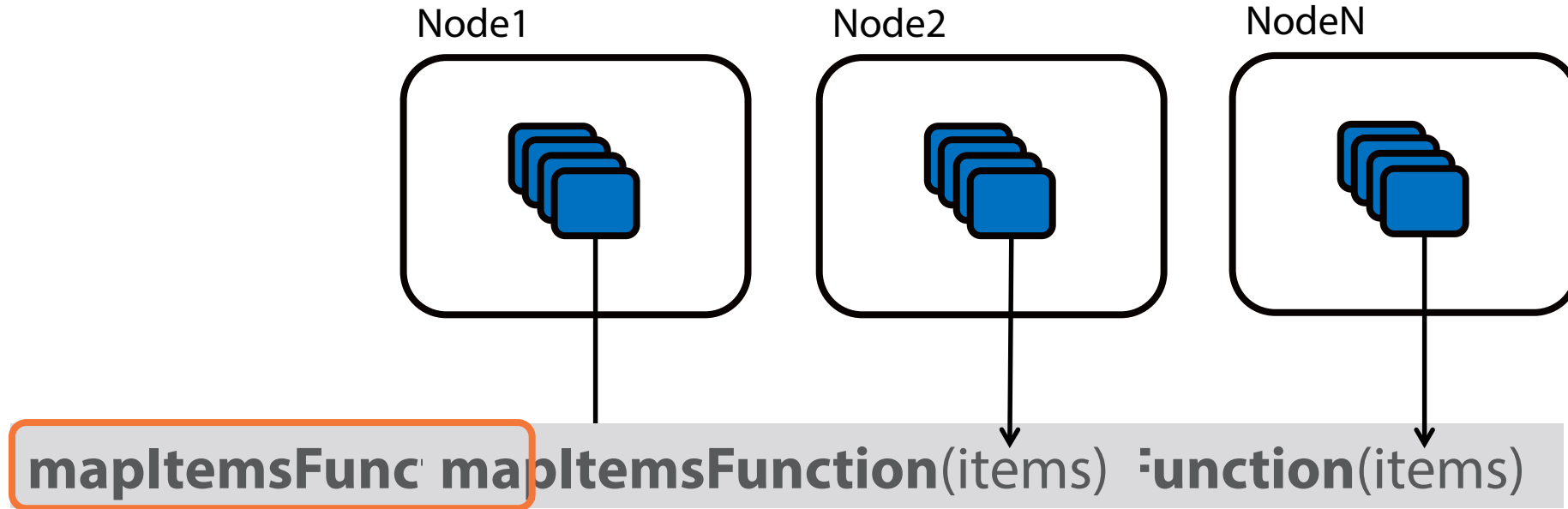
```
}
```

```
for(item <-items) {
```

```
  yield mapFunction(item)
```

```
}
```

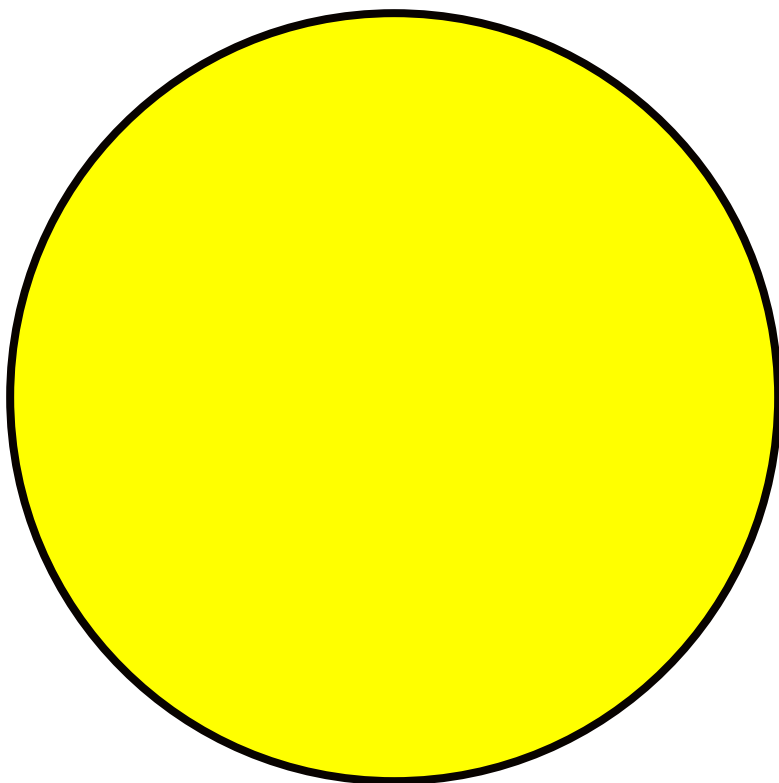
mapPartitions (glom)



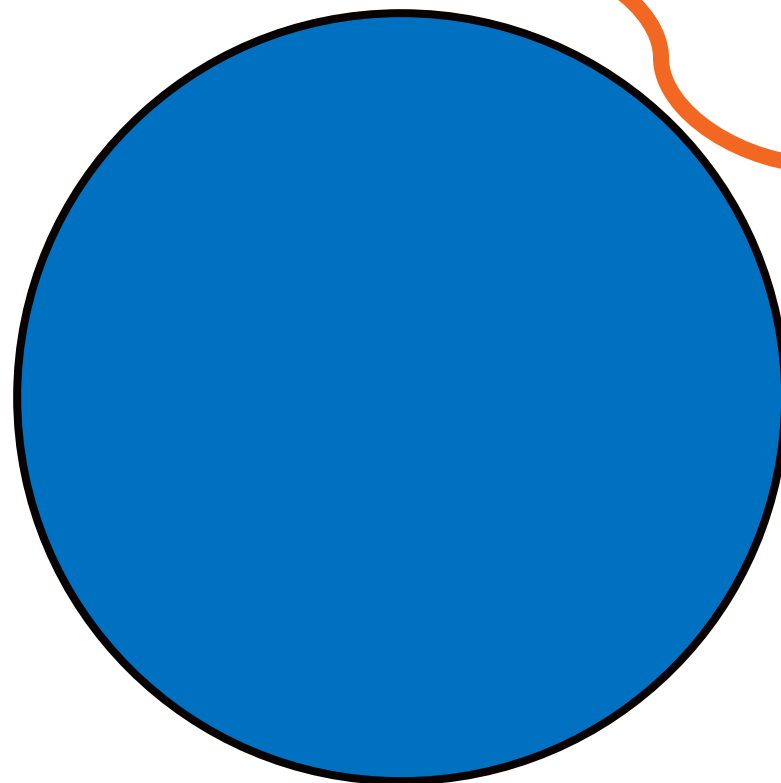
 mongoDB

RDD Combiners

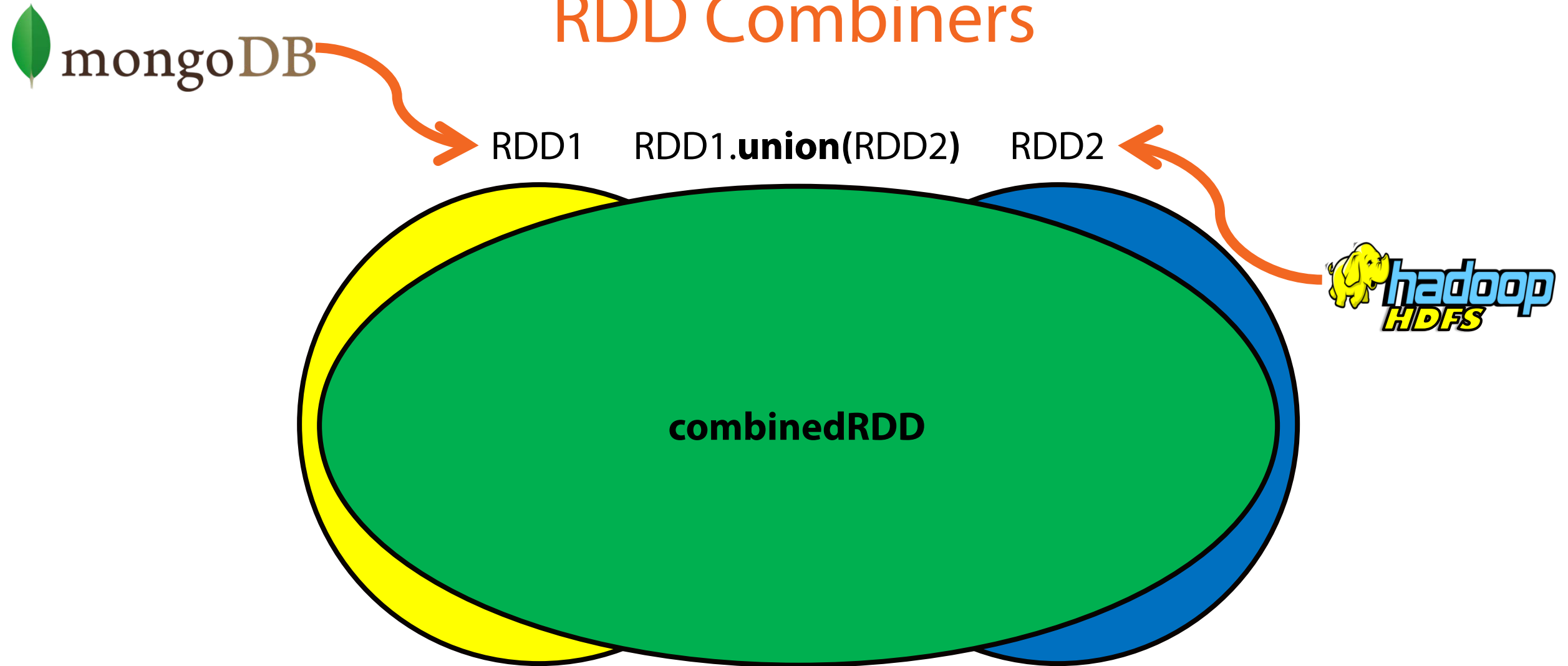
RDD1



RDD2

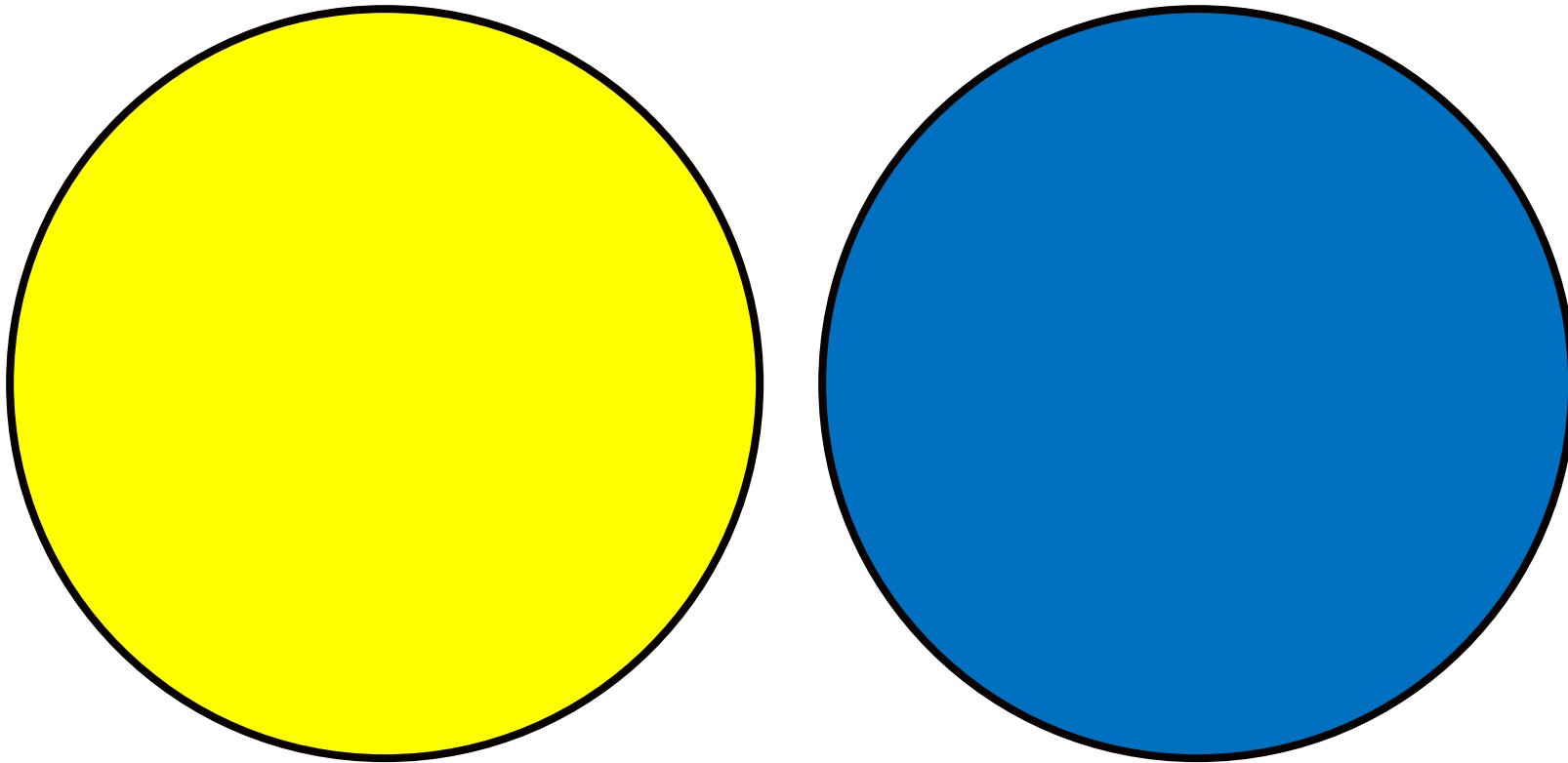


RDD Combiners



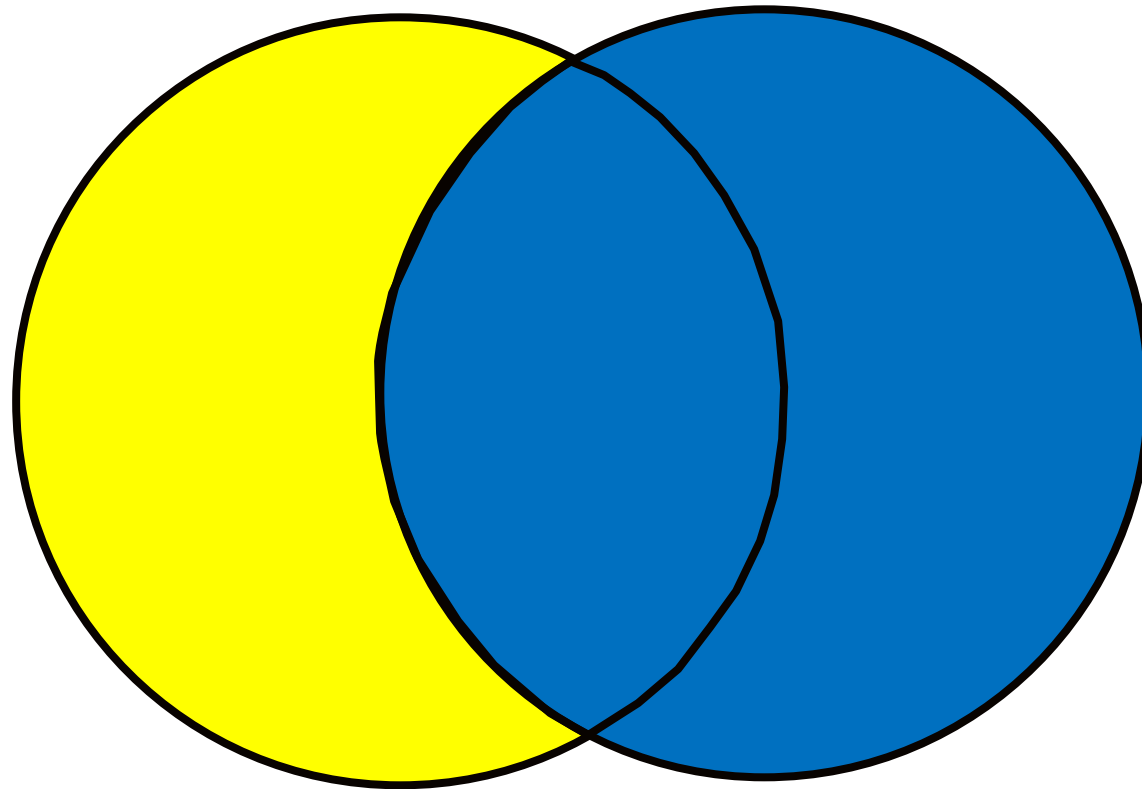
RDD Combiners

RDD1 RDD1.**intersection**(RDD2) RDD2



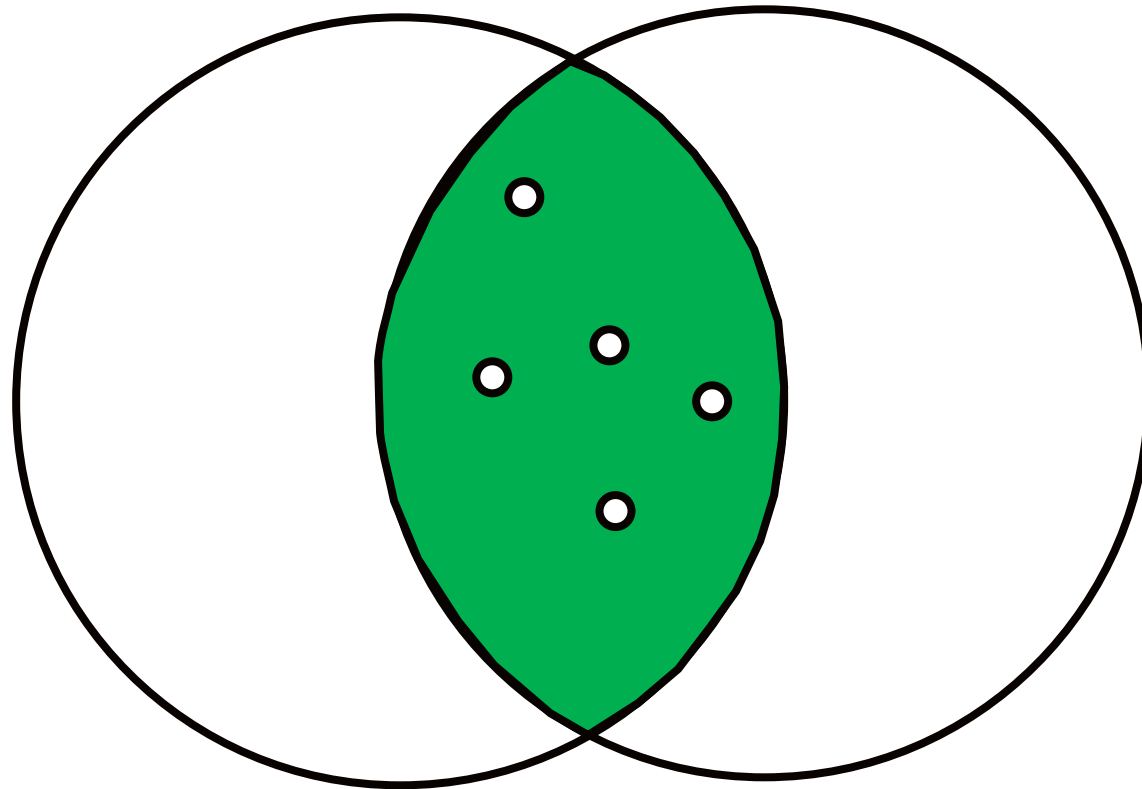
RDD Combiners

`RDD1.intersection(RDD2)`



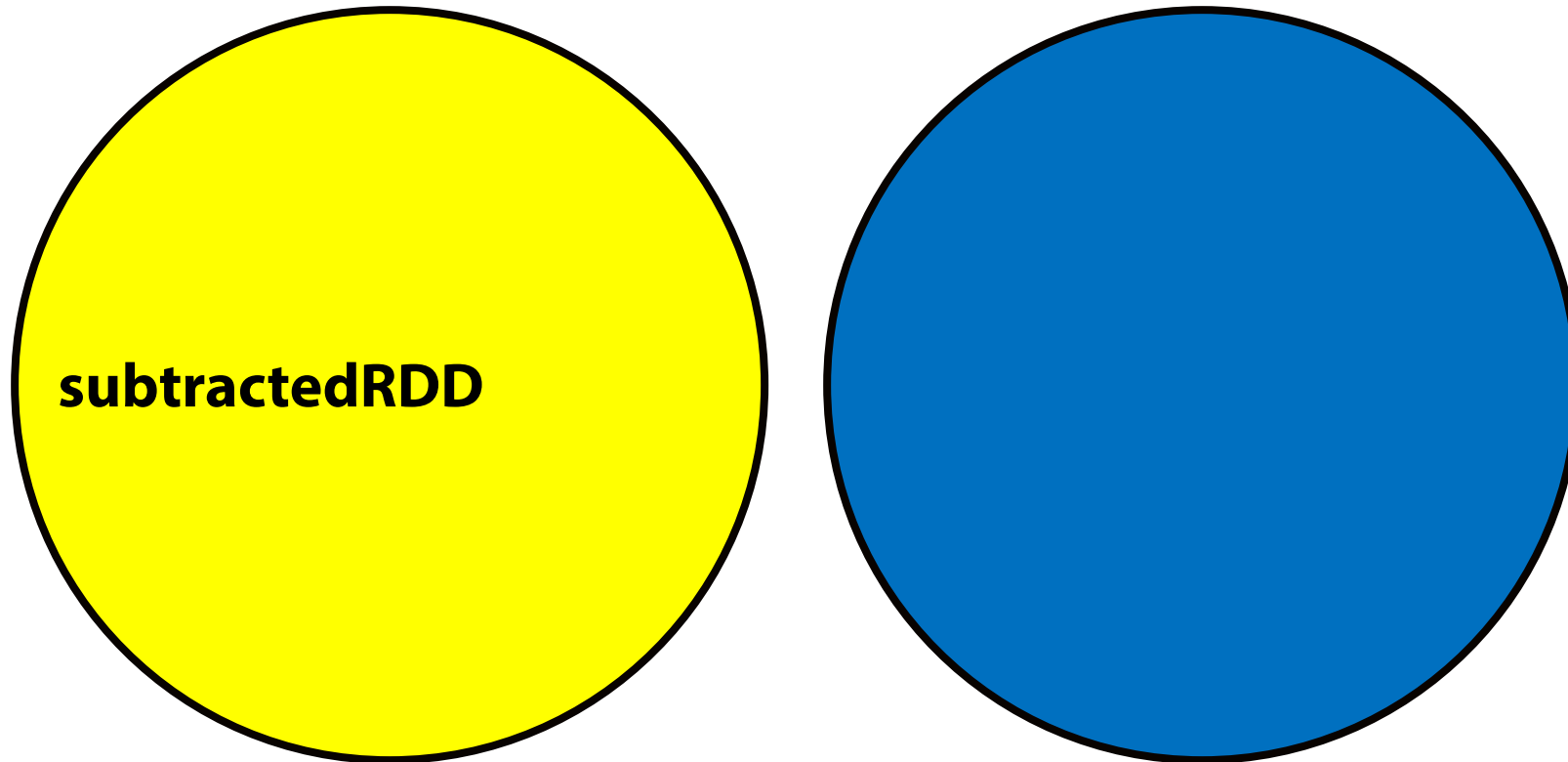
RDD Combiners

`RDD1.intersection(RDD2)`



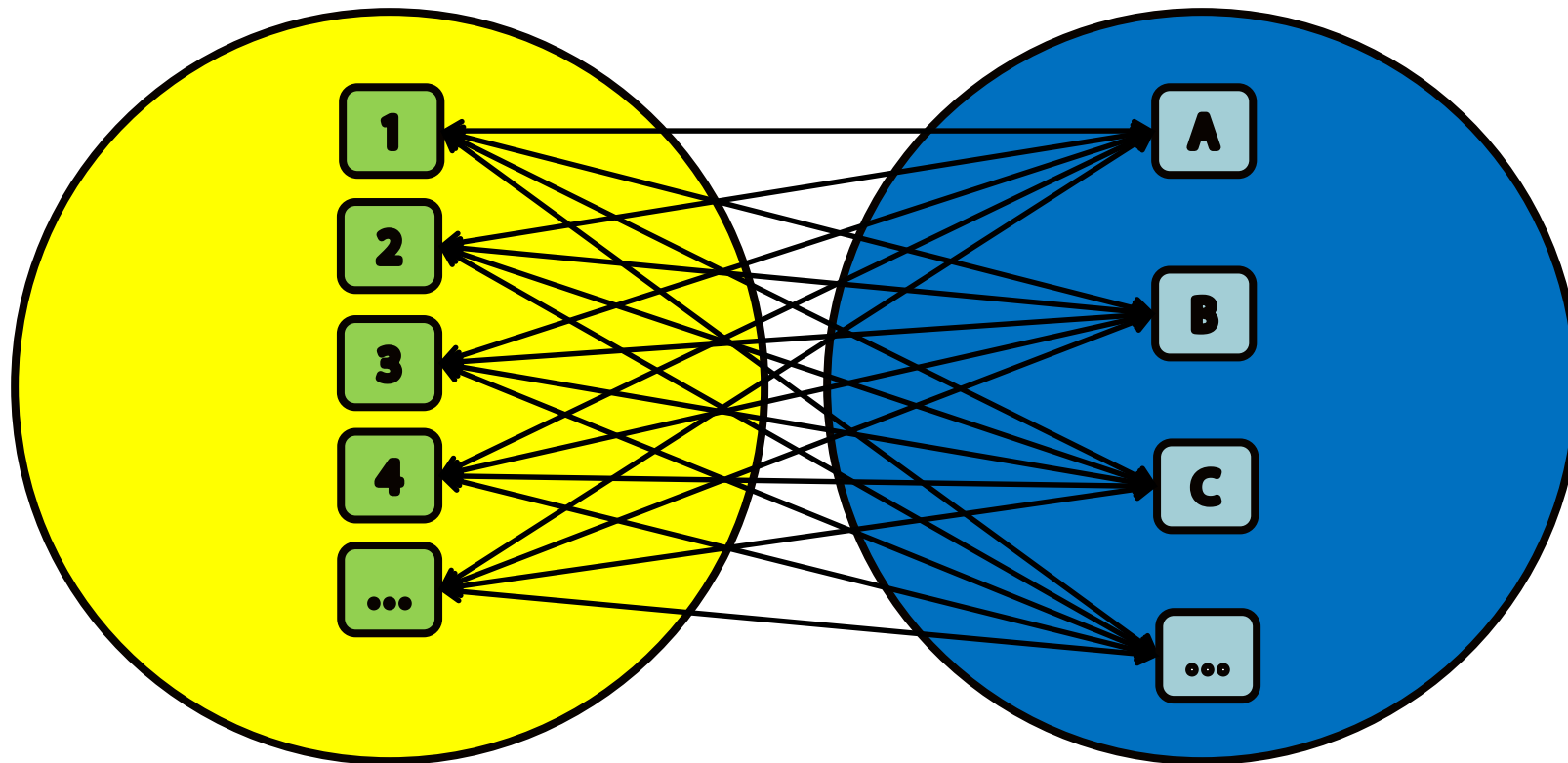
RDD Combiners

RDD1 `RDD1.subtract(RDD2)` RDD2



RDD Combiners

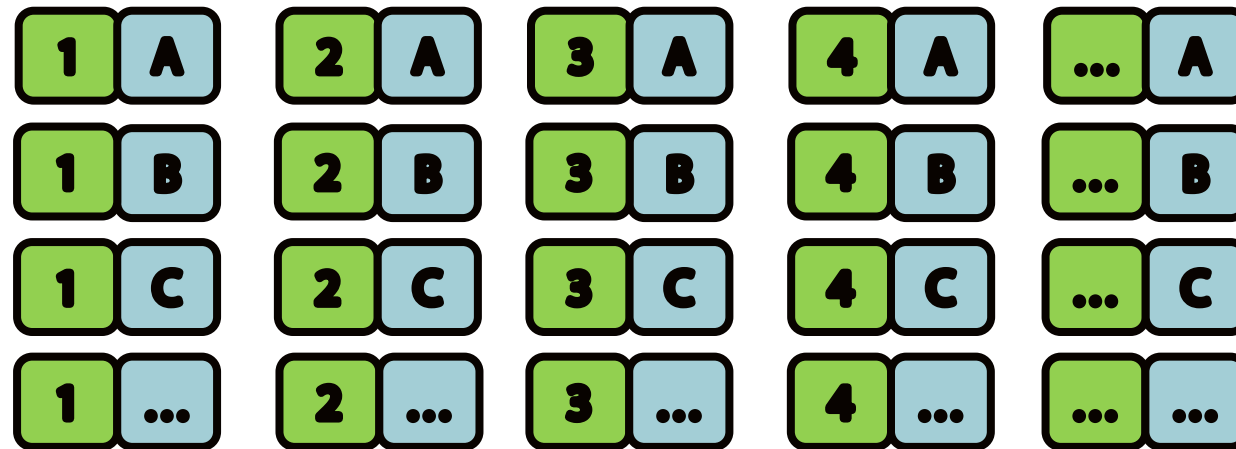
RDD1 `RDD1.cartesian(RDD2)` RDD2



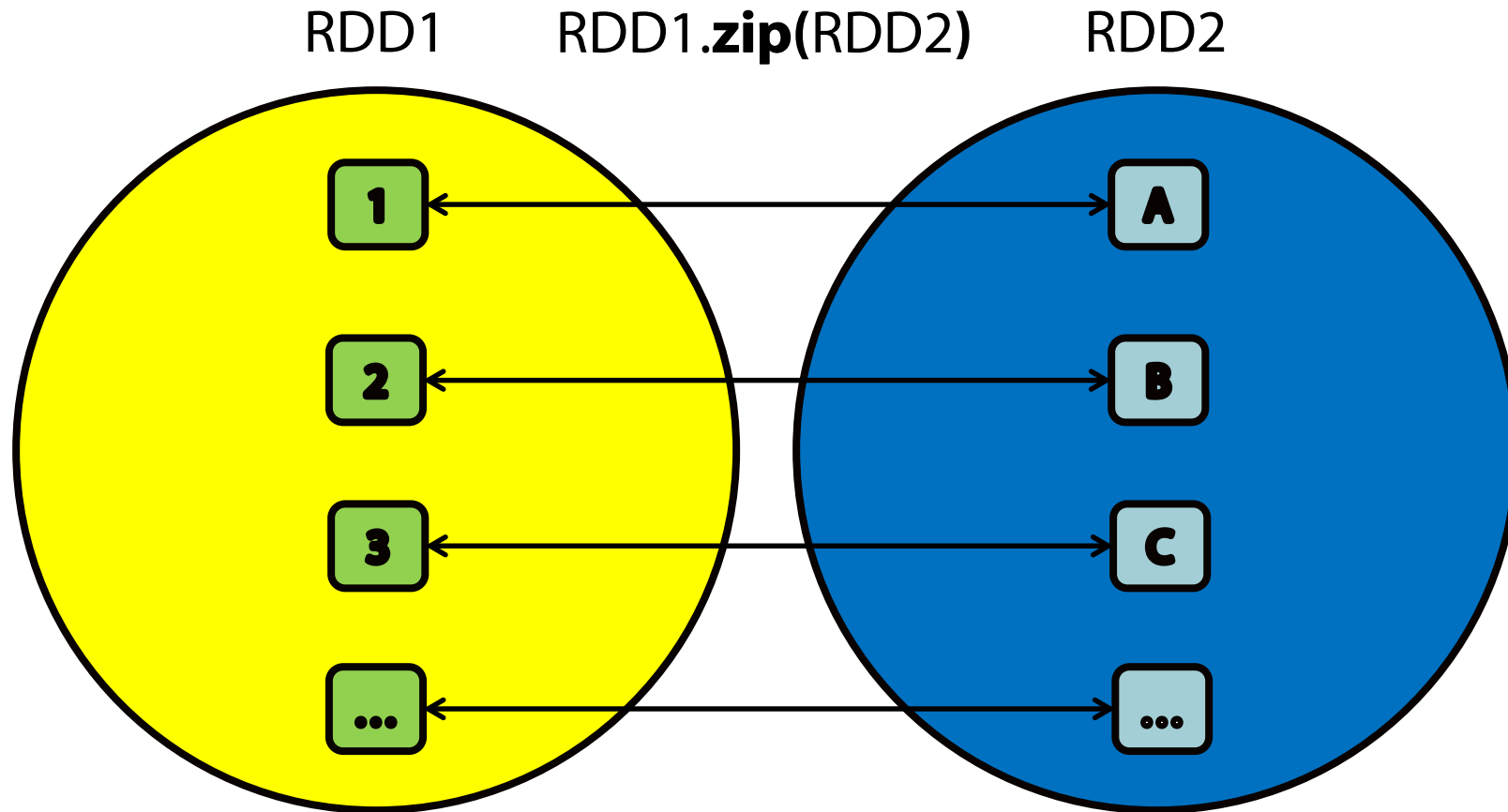
RDD Combiners

`RDD1.cartesian(RDD2)`

NxMRDD

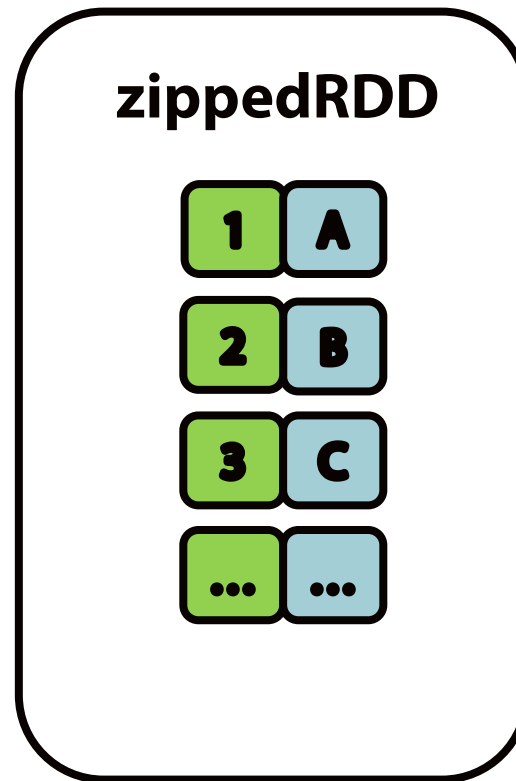


RDD Combiners

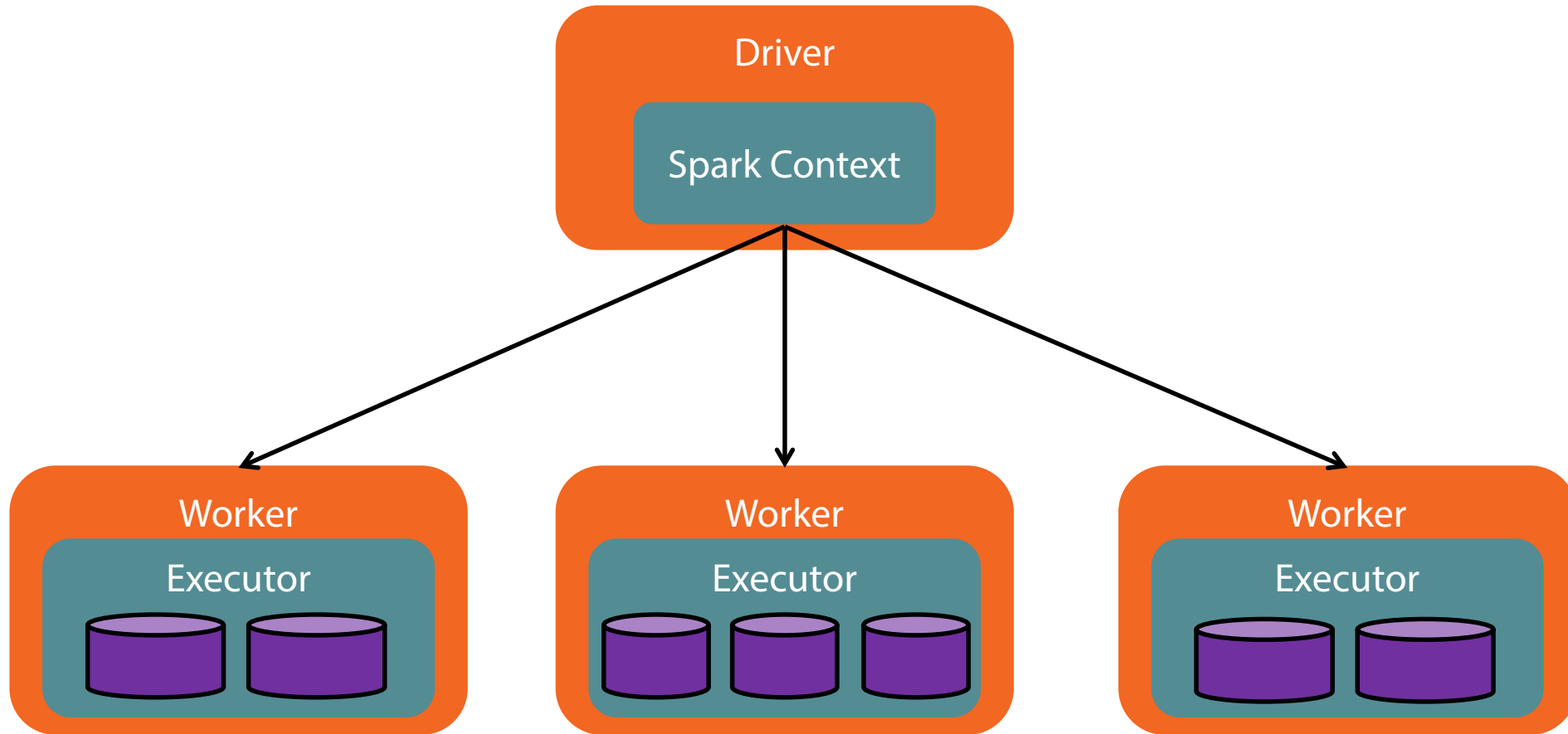


RDD Combiners

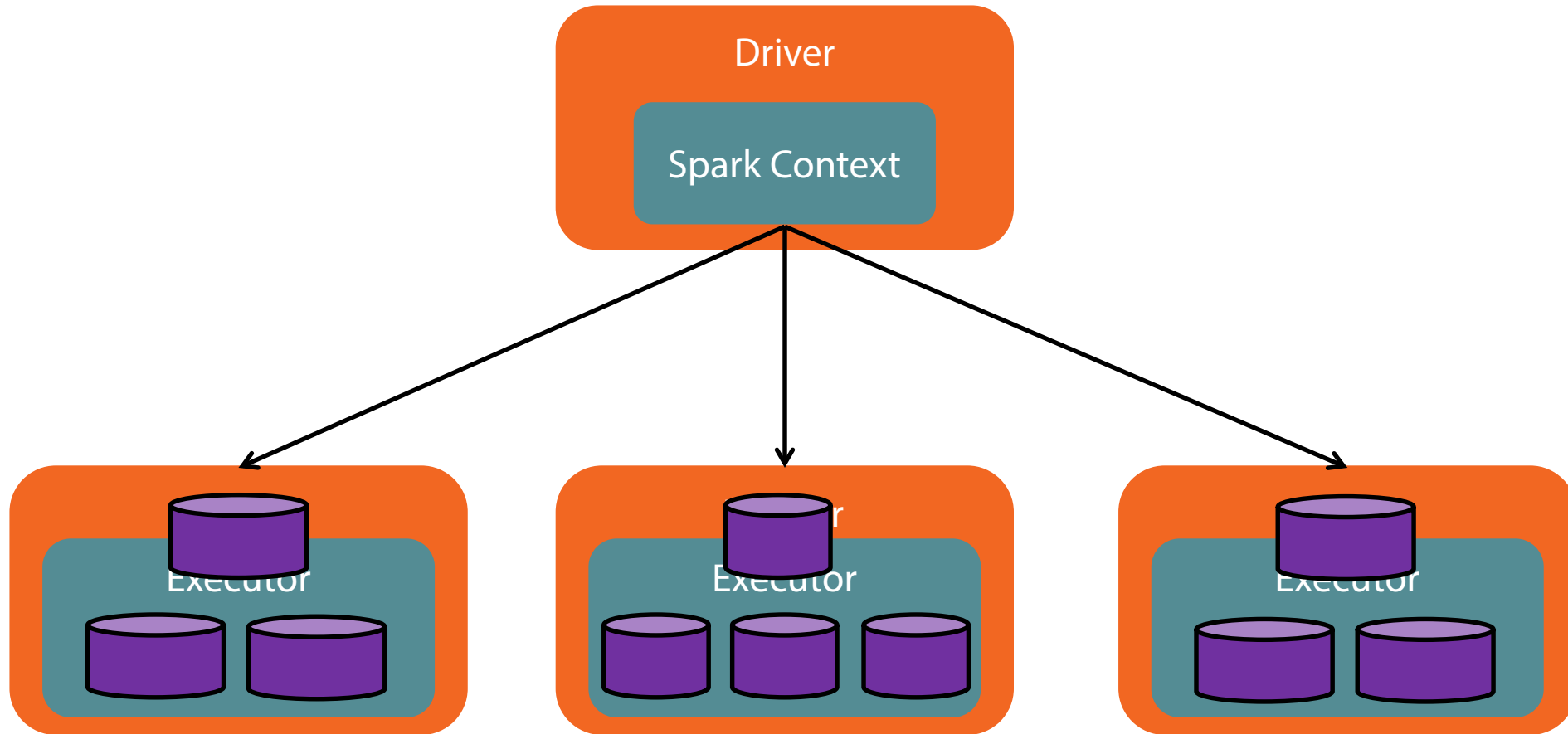
RDD1.**zip**(RDD2)



Actions



Actions



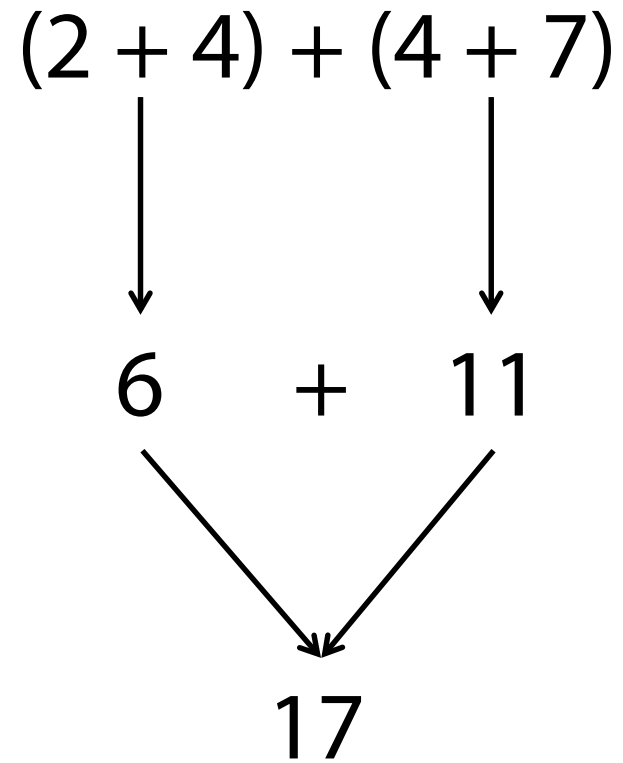
Associative Property

$$2 + 4 + 4 + 7$$

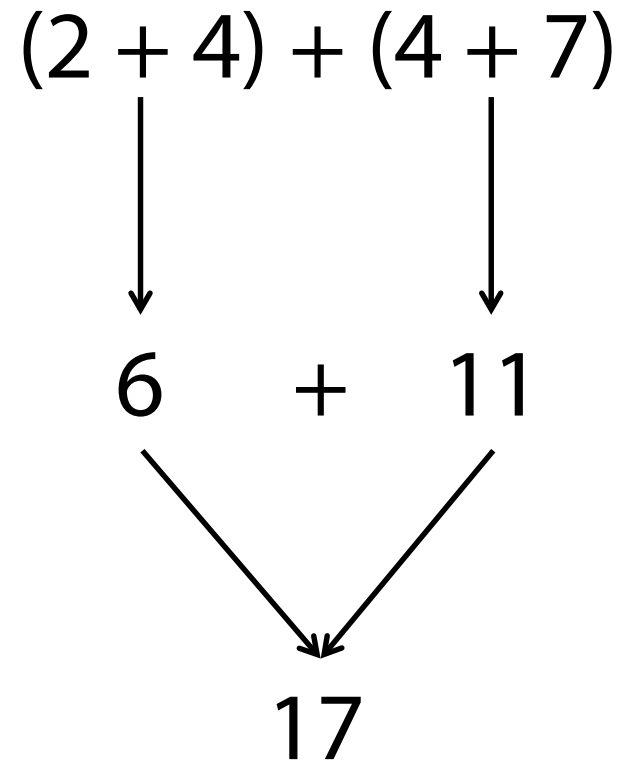
Associative Property

$$(2 + 4) + (4 + 7)$$

Associative Property



Associative Property



Associative Property

$$\begin{array}{c} (2 + 4) + (4 + 7) \\ \downarrow \qquad \downarrow \\ 6 \quad + \quad 11 \\ \swarrow \quad \searrow \\ 17 \end{array}$$

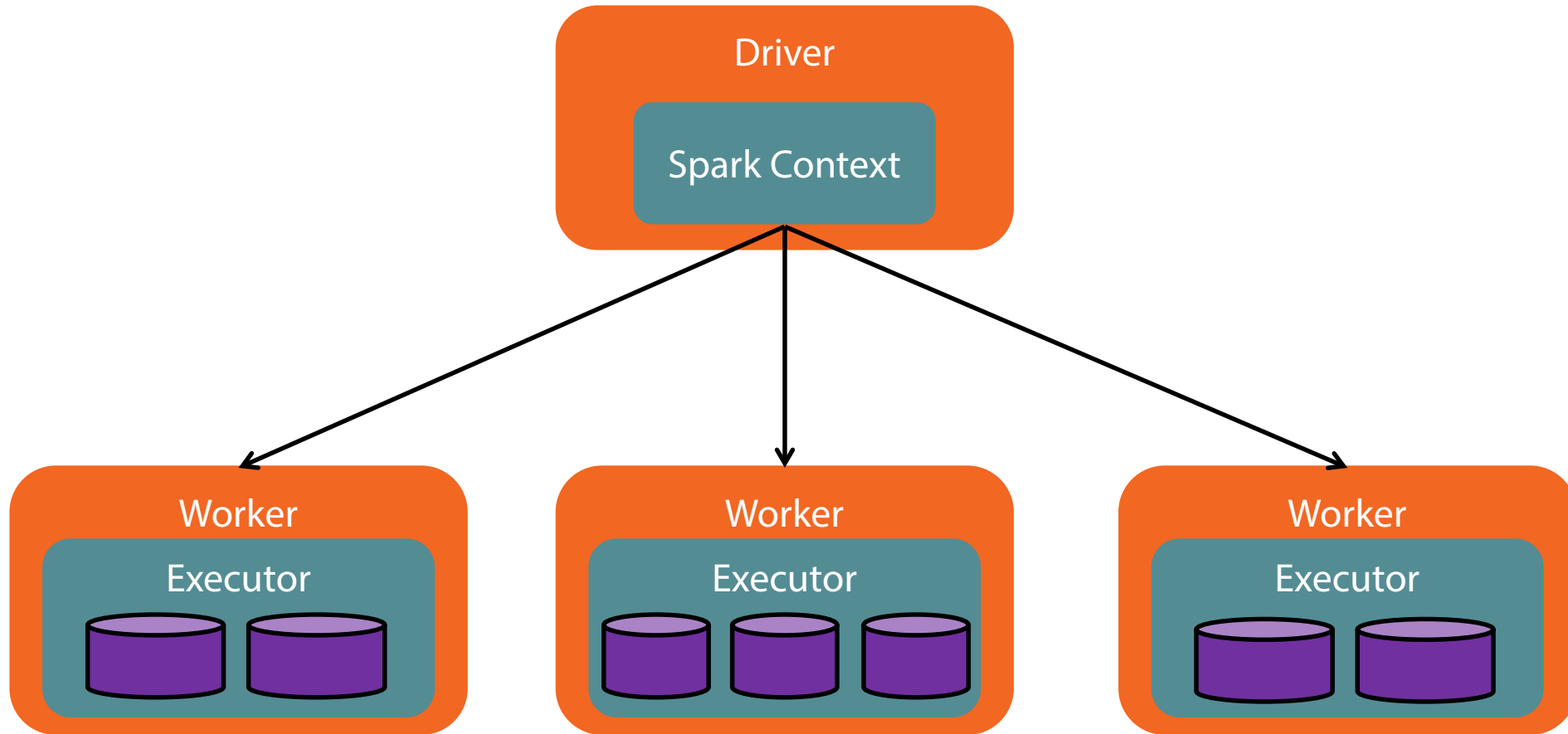
$$\begin{array}{c} (2 + 4 + 4) + 7 \\ \downarrow \qquad \downarrow \\ 10 \quad + \quad 7 \\ \swarrow \quad \searrow \\ 17 \end{array}$$

Associative Property

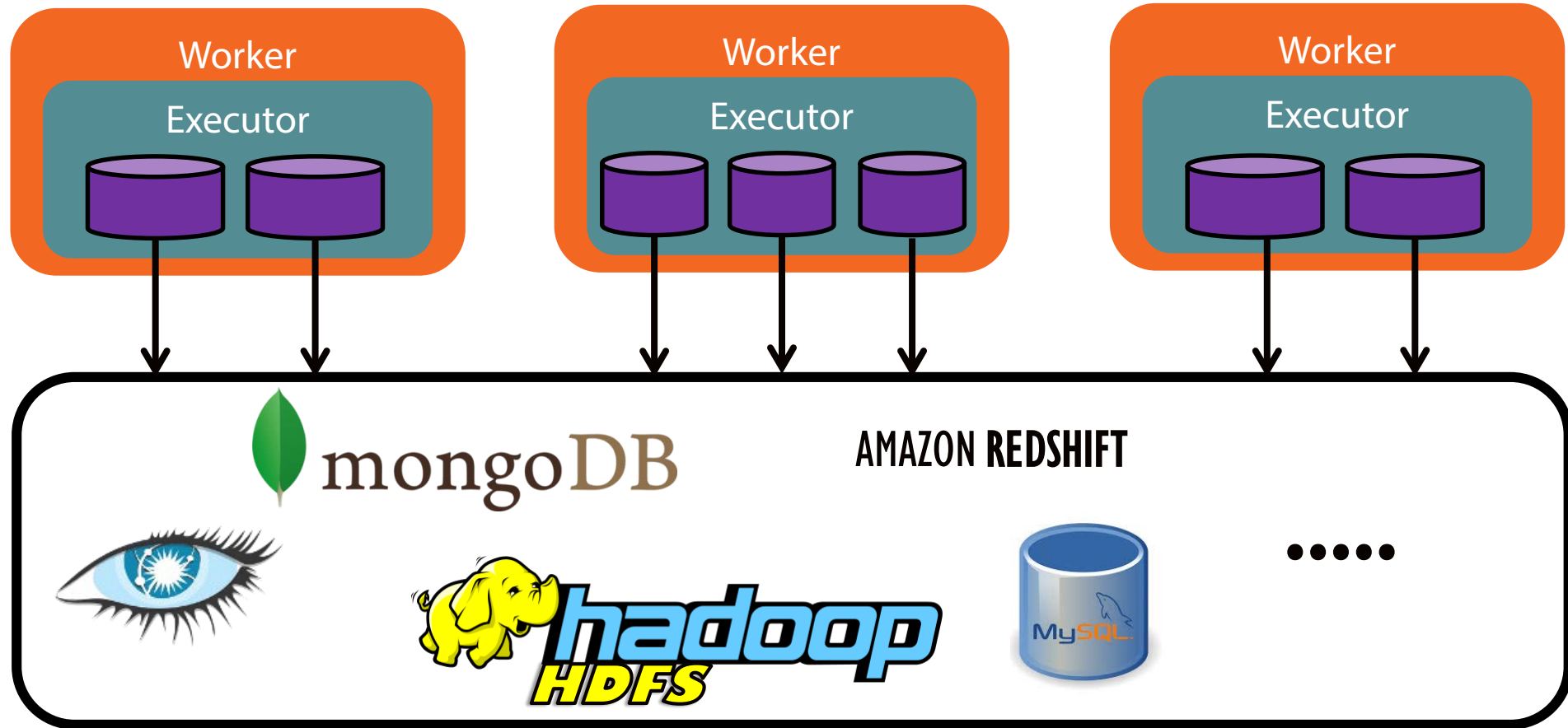
$$\begin{array}{c} (2 + 4) + (4 + 7) \\ \downarrow \qquad \downarrow \\ 6 \quad + \quad 11 \\ \swarrow \quad \searrow \\ 17 \end{array}$$

$$\begin{array}{c} (2 + 4 + 4) + 7 \\ \downarrow \qquad \downarrow \\ 10 \quad + \quad 7 \\ \swarrow \quad \searrow \\ 17 \end{array}$$

Actions



Actions



Actions

- `saveAsObjectFile(path)`
- `saveAsTextFile(path)`
- External connector
- `foreach(T => Unit)`
 - `foreachPartition(Iterator[T] => Unit)`

Resources

- RDD Research Paper
 - http://www.cs.berkeley.edu/~matei/papers/2012/nsdi_spark.pdf
- Lambdas
 - What's New in Java 8: Jose Paumard
 - Functional Programming With Java: Jessica Kerr
- Add ALL the Things: Avi Bryant
 - <http://www.infoq.com/presentations/abstract-algebra-analytics>

Summary

- Spark Context
- RDD
- Transformations
- Actions