Apache Spark Workshop

WWC Meetup @XO Group | 1 June 2016

Welcome!

- Tonight's Agenda
  - Welcome and Networking
  - What is distributed computing?
  - What is Spark and why does it matter?
  - Hands-on Tutorial: "Hello, Spark!"

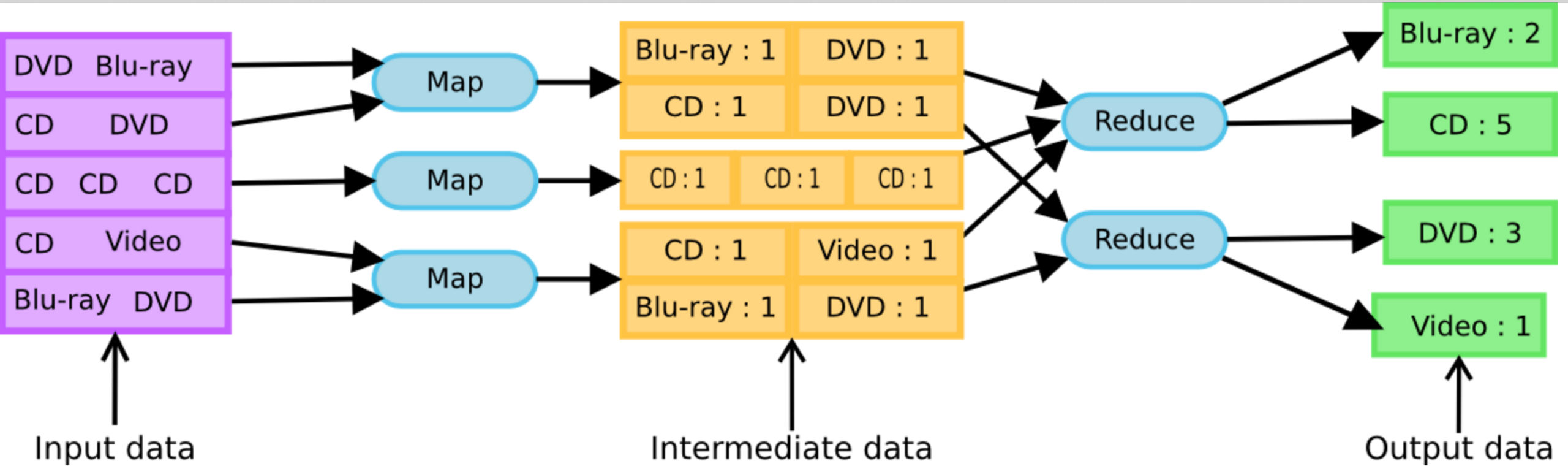
What is distributed computing?

- Computing across clusters
- Scalable, fault-tolerant
- Foundation of Hadoop

What is Spark and why is it important?

- Cluster computing framework for big data processing
- Spark APIs to code in Python (PySpark), R, Java, Scala
- SparkSQL for relational data querying
- MLlib for machine learning
- GraphX for graph processing

What is MapReduce?



Spark versus Hadoop MapReduce

- Spark can be 100x faster than MR due to in-memory procesing (contrast with MR storing to disk)
- Map-reduce concepts still exist in Spark!

Counting in MapReduce

```
def mapper(line):
    words = line.split()
    for word in words:
        yield word, 1

def reducer(word, counts):
    print word, sum(counts)
```

Counting in Spark

```
wordCounts = textDocument \
    .flatMap(lambda line: line.split()) \
    .map(lambda word: (word, 1)) \
    .reduceByKey(lambda x, y: x + y)
```

Some deeper Spark concepts…

- **Spark Context** (SC): must be created at the start of Spark session
- **Resilient Distributed Dataset** (RDD): data across cluster nodes that can be acted on in parallel
  - new RDDs are created lazily with each transformation, such as *map*, *reduceByKey*, etc
  - can be converted to/from Spark's relational **DataFrames**
- **SQL Context:** created from SC and provides RDMS operations

Hands-on Tutorial: ”Hello, Spark!”

- Take out your laptops (and/or share with a neighbor!)
- Head to: github.com/keiraqz/SparkIntro
- Questions? Let us know!