Cloud - 5

ESIR

Djob Mvondo

Proliferation of data

- Due to Cloud advent, companies can store several chunks of data
    - Airbus generates up to 40TB of data per flight test
    - Facebook generates 4 PB every day
    - Twitter generates approximately 500 million tweets per day
    - …

- With Cloud resources, we have enough processing power right?

Proliferation of data

- With Cloud resources, we have enough processing power right?

It depends on how they are used

Example

- Write a program that counts the number of occurrences of each word in a text file.

- Measure the performance of your program for different input sizes : https://archive.ics.uci.edu/ml/machine-learning-databases/bag-of-words/

- Does your program performance scale ? Now working in groups of 4/3, to parallelize the work between the different servers.

Example

- Does your program performance scale ? Now working in groups of 8/10, to parallelize the work between the different servers.

- What are the different pitfalls you faced ?

To summarize

We need new programming abstractions to process big chunks of data :

- (1) very fast, such that it can
- (2) scale across different servers, while efficiently using
- (3) available resources while achieving
- (4) fault tolerance.

To summarize

- Fast processing is essential to meet stringent demands
  - Finance
  - Marketing
  - Recommender systems
  - Face recognition systems
  - ….

- Scaling is essential to efficiently use available resources and meet workload bursts

- Fault tolerance is necessary to reduce unecessary work performed and detect processing errors that can cost alot

Two programming abstractions

Batch processing (MapReduce)

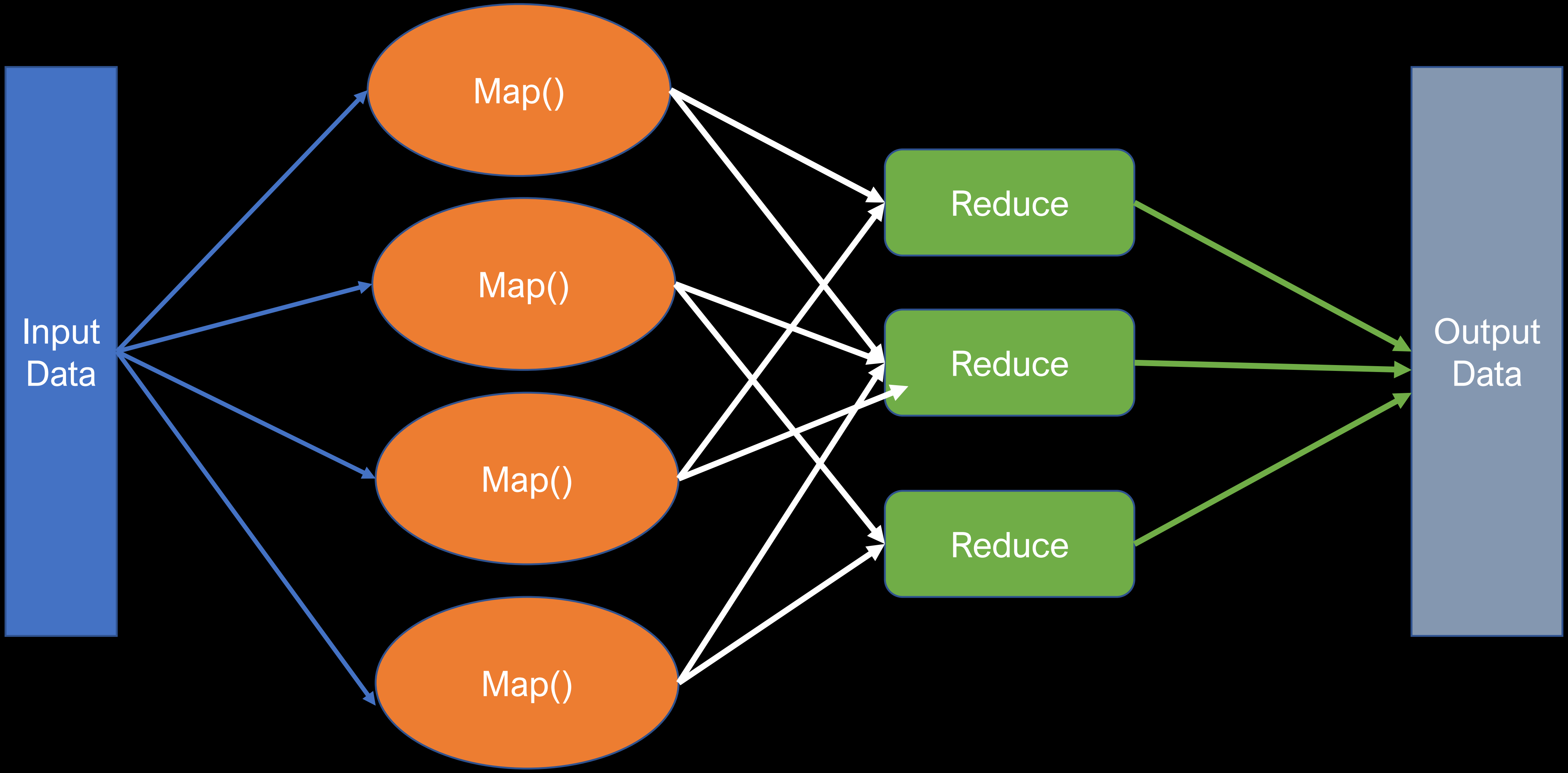
Stream processing

Batch processing (MapReduce)

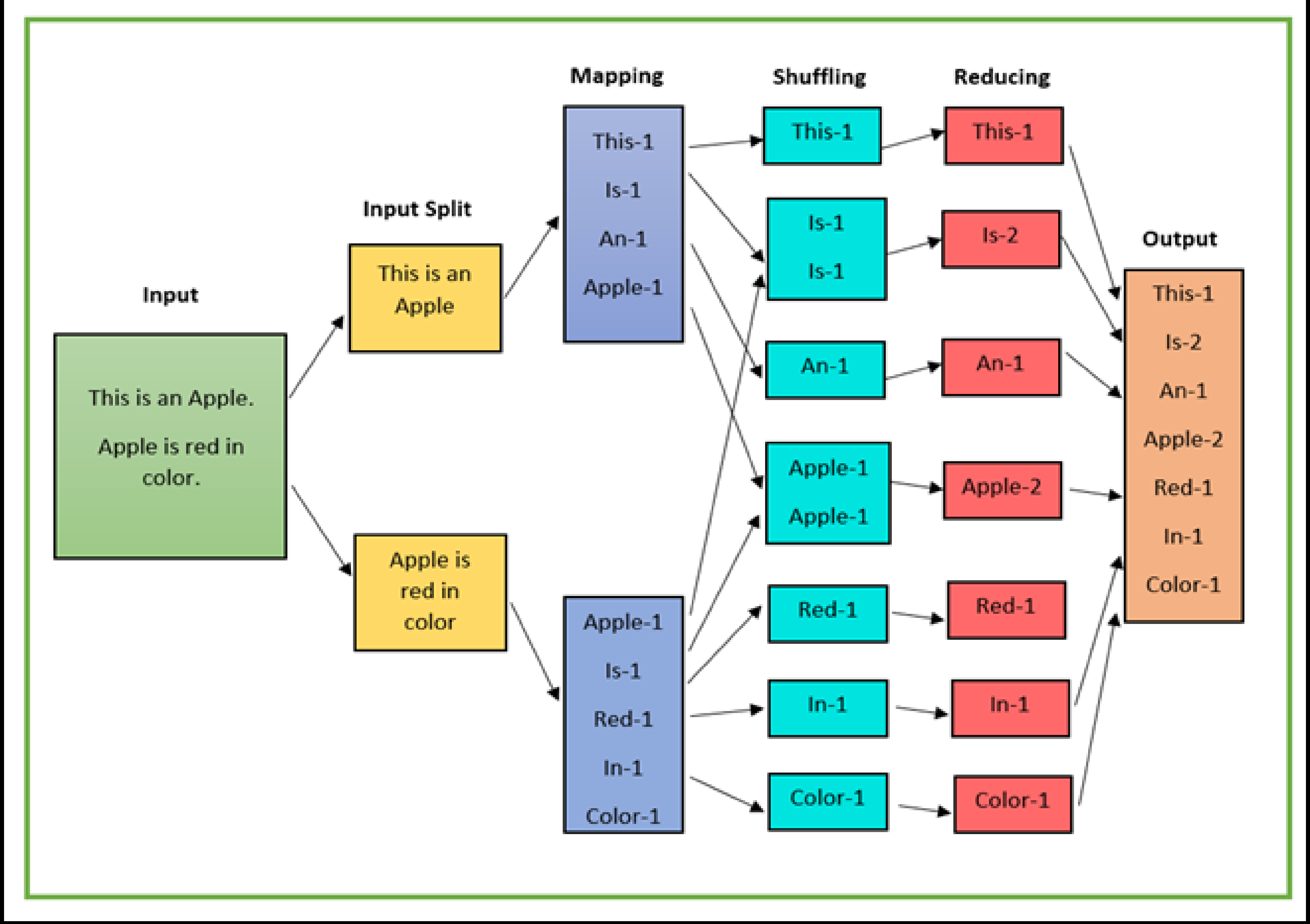
- Perform processing on big chunks of data (usually distributed) introduced by Dean and Sanjay from Google[1].
- The core idea is to divide and conquer
- A set of jobs divides the data to be processed by several entities, then the data chunks are sorted (map) and then aggregated to get the final result (reduce).
  - Sort-Map: Which data interest me ?
  - Aggregated-Reduce: How should I combine the results ?

Jeffrey Dean and Sanjay Ghemawat: MapReduce: Simplified Data Processing on Large Clusters, OSDI 2004

Batch processing (MapReduce)



Batch processing (MapReduce) - Wordcount



Can you implement it?
What are the main difficult aspect of implementing this architecture?

https://www.analyticsvidhya.com/blog/2022/05/an-introduction-to-mapreduce-with-a-word-count-example/

Batch processing (MapReduce)

- Used by several mainstream products e.g., MongoDB, Hadoop/HDFS, etc…
- Requires coordination, task initialization, coordination, scheduling, and monitoring
- Can achieve up to 100x faster processing times than standard naive abstractions.
- Several existing interfaces in several existing programming languages.

Batch processing (MapReduce)

- Used by several mainstream products e.g., MongoDB, Hadoop/HDFS, etc…
- Requires coordination, task initialization, coordination, scheduling, and monitoring
- Can achieve up to 100x faster processing times than standard naive abstractions.
- Several existing interfaces in several existing programming languages.

Stream processing

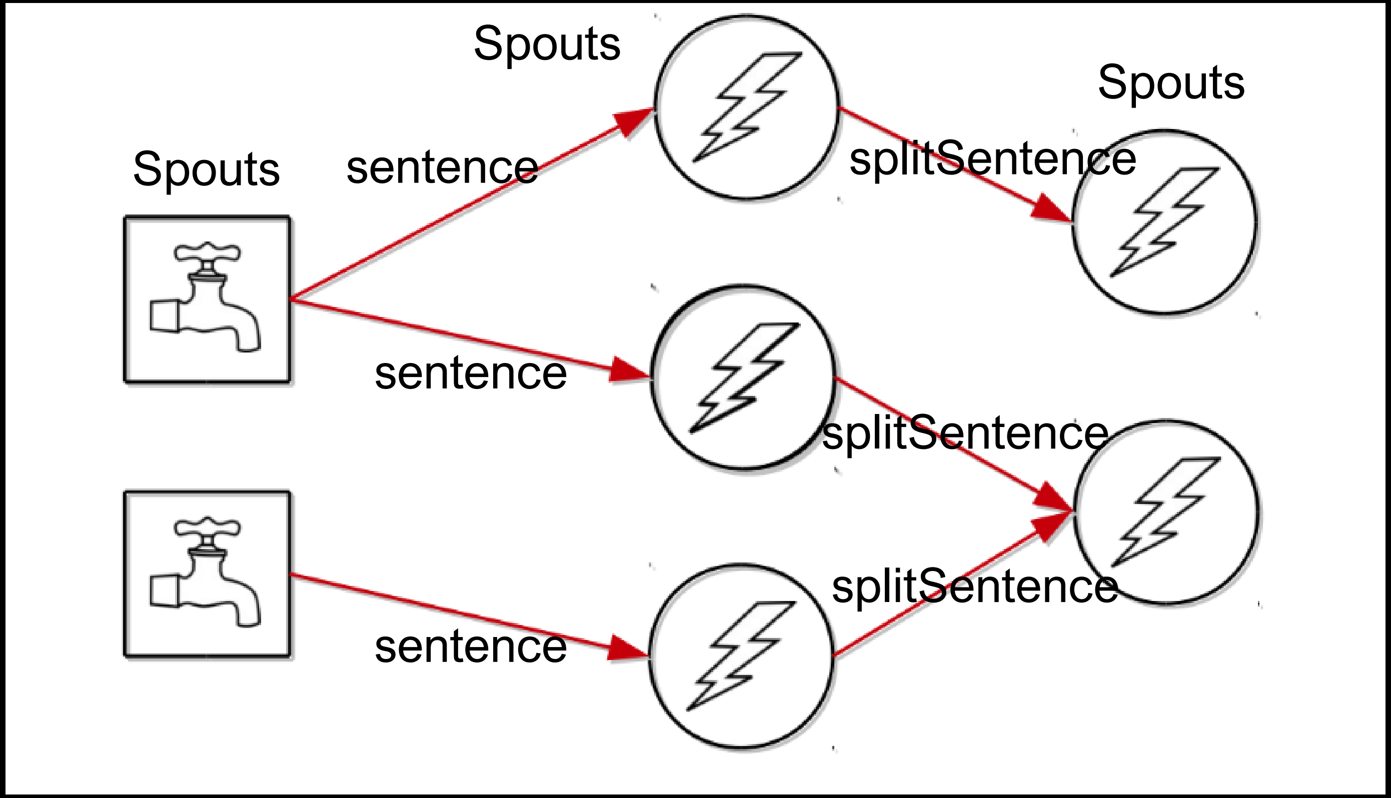
- Introduced by Apache Storm in 2011 mainly by Twitter Engineers to handle real-time rendering of tweets feed
- Meant for continuous execution where there are several data sources compared to batch processing where data is already registered/saved somewhere.

Stream processing

- Introduces the concept of spouts and bolts
- Spouts generate data (data sources)
- Bolts perform an operation and send the data to one or more other bolts
- A combination of spouts and bolts form a topology

Stream processing

- An example of a stream processing technology


Spouts
Spouts
Spouts
sentence
sentence
sentence
splitSentence
splitSentence
splitSentence