edureka!

Big Data & Hadoop

Agenda

- ✓ Revision
- ✓ Where Map-Reduce is used?
- ✓ Weather Data(Where MapReduce is Used
- ✓ Usecases in Healthcare & Architecture
- ✓ Traditional Way
- ✓ Map Reduce Process
- ✓ Anatomy of a MapReduce program
- ✓ MapReduce Way
- ✓ Revisit de-identification architecture
- ✓ Why MapReduce
- ✓ Input Splits
- ✓ Relation between Input Splits and HDFS blocks
- ✓ MapReduce Flow
- ✓ Overview of MapReduce
- ✓ Combiner & Partitioner
- ✓ Input Formats

Revision

Conf/ files

- ✓ Core-site.xml
- ✓ Mapred-site.xml
- ✓ Hdfs-site.xml
- ✓ Slaves
- ✓ Masters

Where MapReduce Is Used?

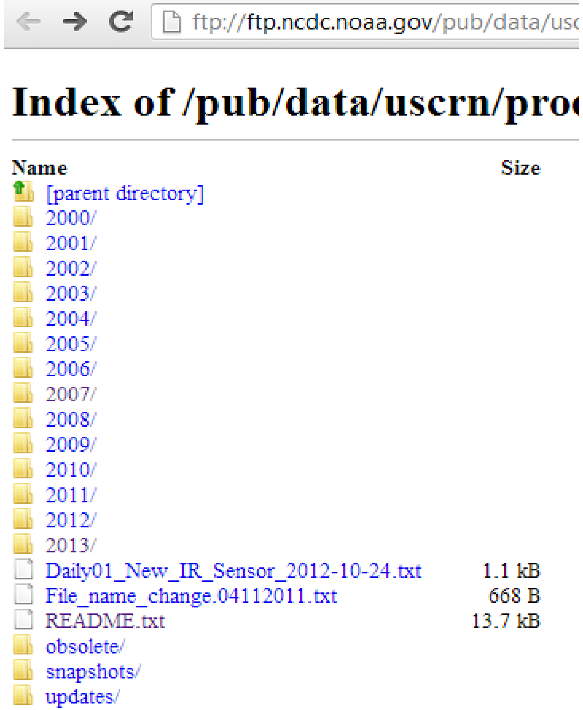
✓ Weather Forecasting

✓ Health Care

✓ Oil & Gas Industry

Weather Data

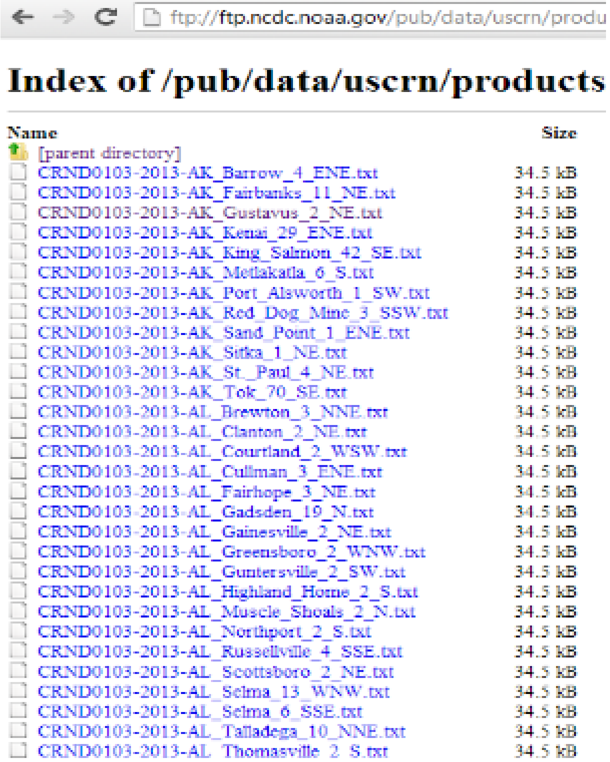
ftp://ftp.ncdc.noaa.gov/pub/data/uscrn/products/daily01/



ftp://ftp.ncdc.noaa.gov/pub/data/usc

Index of /pub/data/uscrn/pro

Name	Size
[parent directory]	
2000/	
2001/	
2002/	
2003/	
2004/	
2005/	
2006/	
2007/	
2008/	
2009/	
2010/	
2011/	
2012/	
2013/	
Daily01_New_IR_Sensor_2012-10-24.txt	1.1 kB
File_name_change.04112011.txt	668 B
README.txt	13.7 kB
obsolete/	
snapshots/	
updates/	



ftp://ftp.ncdc.noaa.gov/pub/data/uscrn/produ

Index of /pub/data/uscrn/products

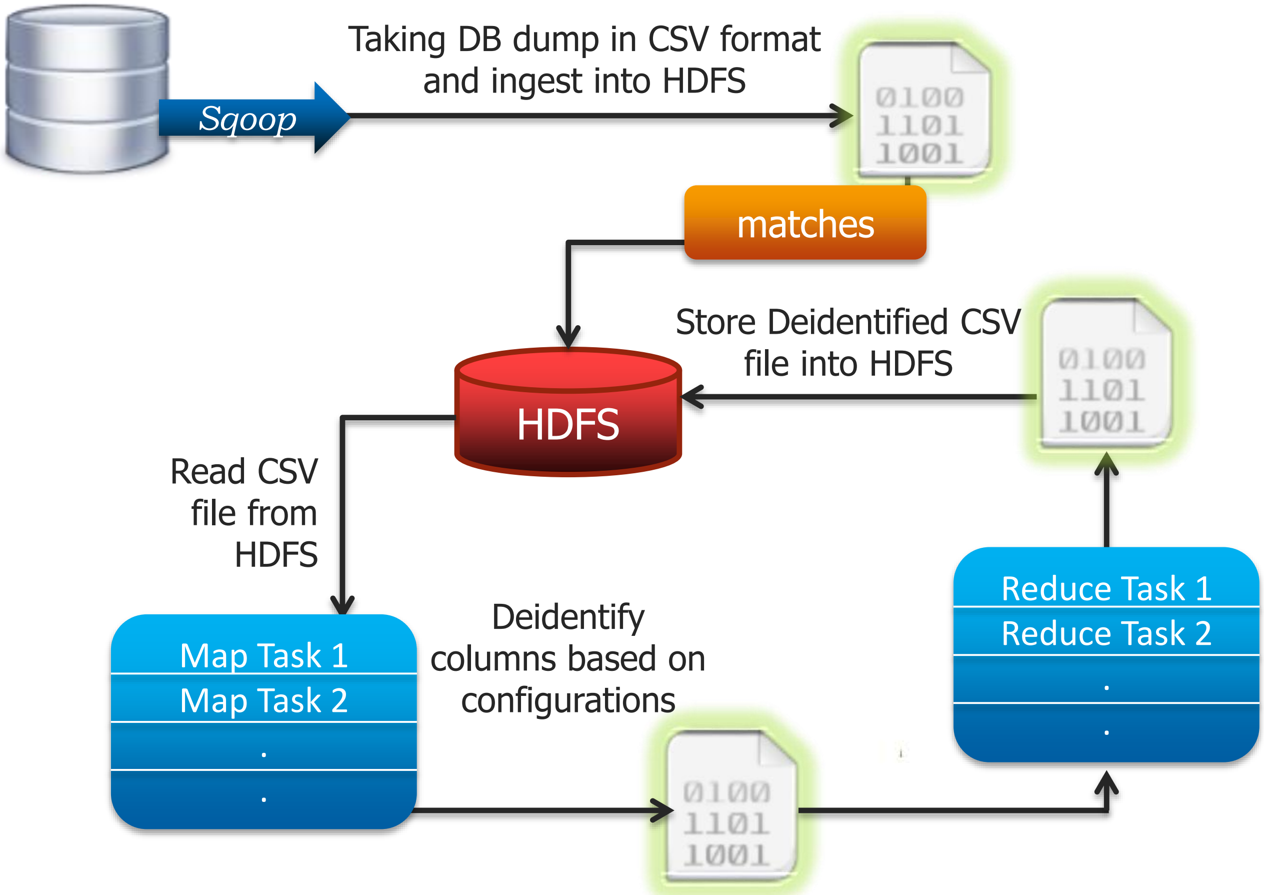
Name	Size
[parent directory]	
CRND0103-2013-AK_Barrow_4_ENE.txt	34.5 kB
CRND0103-2013-AK_Fairbanks_11_NE.txt	34.5 kB
CRND0103-2013-AK_Gustavus_2_NE.txt	34.5 kB
CRND0103-2013-AK_Kenai_29_ENE.txt	34.5 kB
CRND0103-2013-AK_King_Salmon_42_SE.txt	34.5 kB
CRND0103-2013-AK_Metlakatla_6_S.txt	34.5 kB
CRND0103-2013-AK_Port_Alsworth_1_SW.txt	34.5 kB
CRND0103-2013-AK_Red_Dog_Mine_3_SSW.txt	34.5 kB
CRND0103-2013-AK_Sand_Point_1_ENE.txt	34.5 kB
CRND0103-2013-AK_Sitka_1_NE.txt	34.5 kB
CRND0103-2013-AK_St._Paul_4_NE.txt	34.5 kB
CRND0103-2013-AK_Tok_70_SE.txt	34.5 kB
CRND0103-2013-AL_Brewton_3_NNE.txt	34.5 kB
CRND0103-2013-AL_Clanton_2_NE.txt	34.5 kB
CRND0103-2013-AL_Courtland_2_WSW.txt	34.5 kB
CRND0103-2013-AL_Cullman_3_ENE.txt	34.5 kB
CRND0103-2013-AL_Fairhope_3_NE.txt	34.5 kB
CRND0103-2013-AL_Gadsden_19_N.txt	34.5 kB
CRND0103-2013-AL_Gainesville_2_NE.txt	34.5 kB
CRND0103-2013-AL_Greensboro_2_WNW.txt	34.5 kB
CRND0103-2013-AL_Guntersville_2_SW.txt	34.5 kB
CRND0103-2013-AL_Highland_Home_2_S.txt	34.5 kB
CRND0103-2013-AL_Muscle_Shoals_2_N.txt	34.5 kB
CRND0103-2013-AL_Northport_2_S.txt	34.5 kB
CRND0103-2013-AL_Russellville_4_SSE.txt	34.5 kB
CRND0103-2013-AL_Scottsboro_2_NE.txt	34.5 kB
CRND0103-2013-AL_Selma_13_WNW.txt	34.5 kB
CRND0103-2013-AL_Selma_6_SSE.txt	34.5 kB
CRND0103-2013-AL_Talladega_10_NNE.txt	34.5 kB
CRND0103-2013-AL_Thomasville_2_S.txt	34.5 kB

Use Case In HealthCare

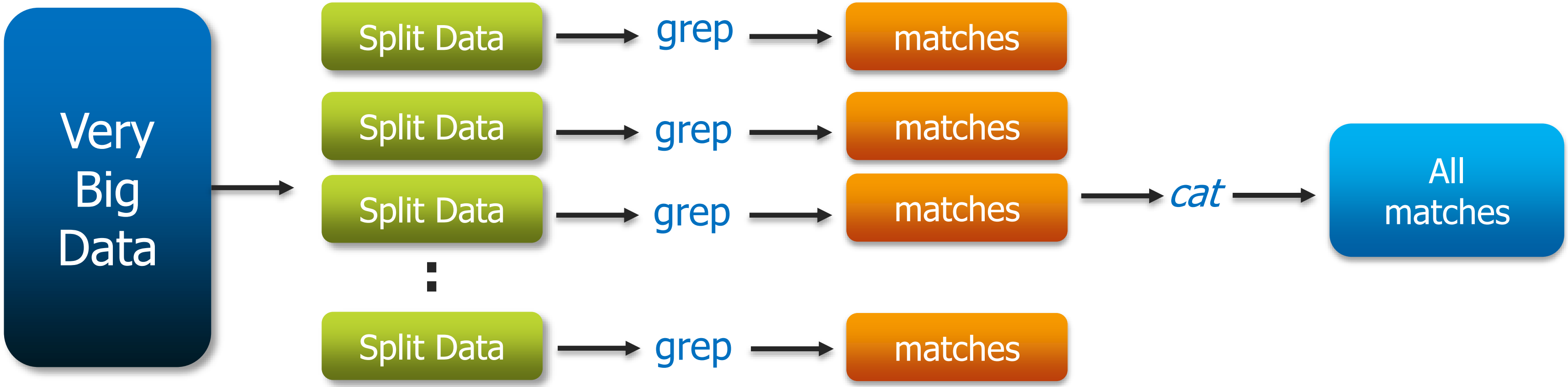
- ✓ Problem Statement:
  - ✓ De-identify personal health information.

- ✓ Challenges:
  - ✓ Huge amount of data flows into the systems daily and there are multiple data sources that we need to aggregate data from.
  - ✓ Crunching this huge data and deidentifying it in a traditional way had problems.

Architecture

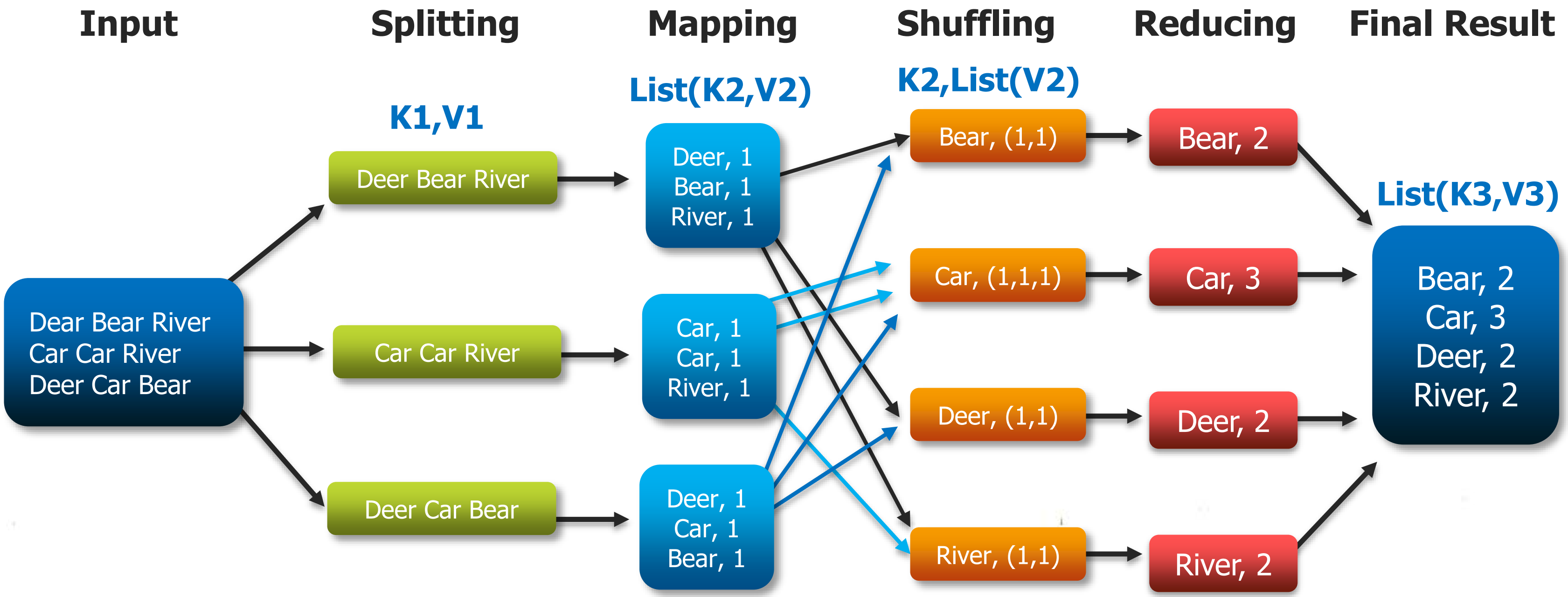

Sqoop
Taking DB dump in CSV format and ingest into HDFS
0100
1101
1001
matches
Store Deidentified CSV file into HDFS
0100
1101
1001
HDFS
Read CSV file from HDFS
Map Task 1
Map Task 2
.
.
Deidentify columns based on configurations
0100
1101
1001
Reduce Task 1
Reduce Task 2
.
.

Traditional Way

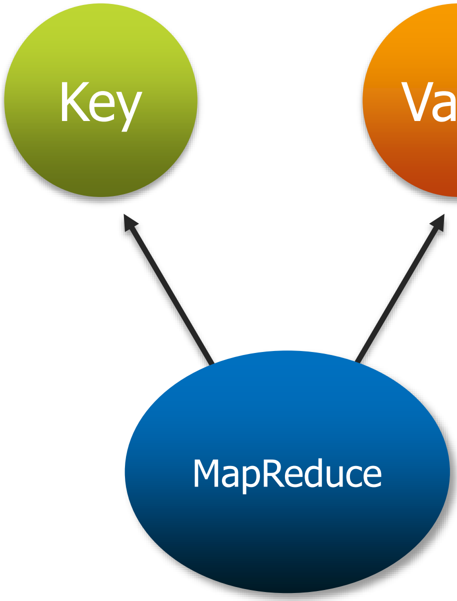


Map Reduce Process

The Overall MapReduce Word Count Process



Anatomy of a MapReduce Program



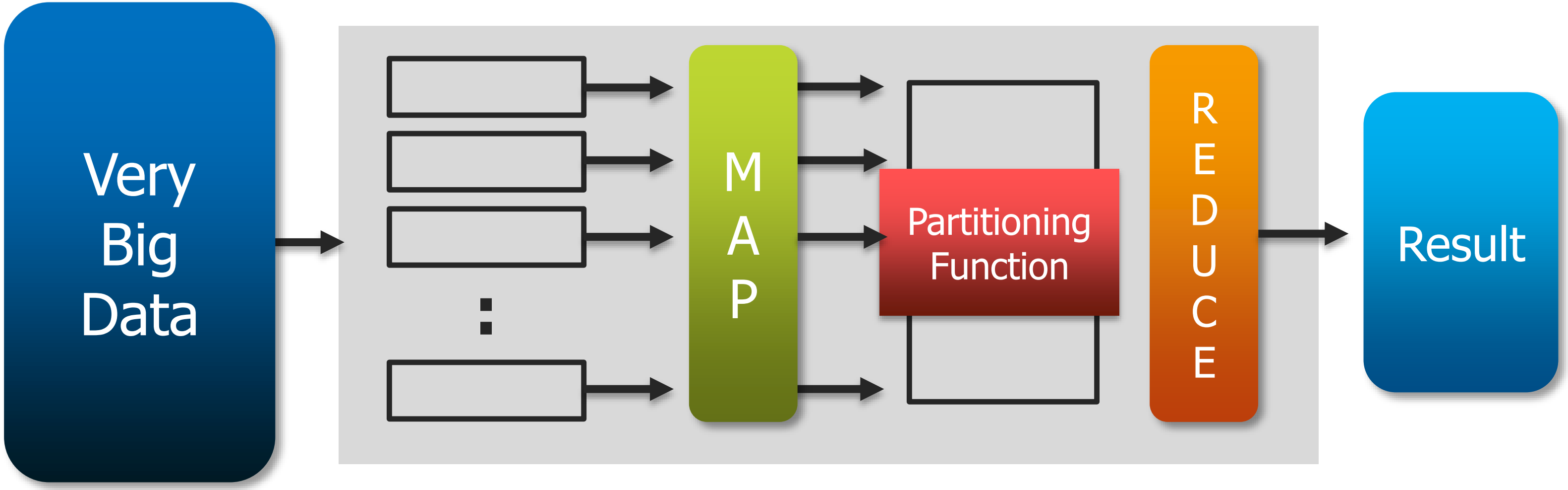
Map:

(K1, V1) List (K2, V2)

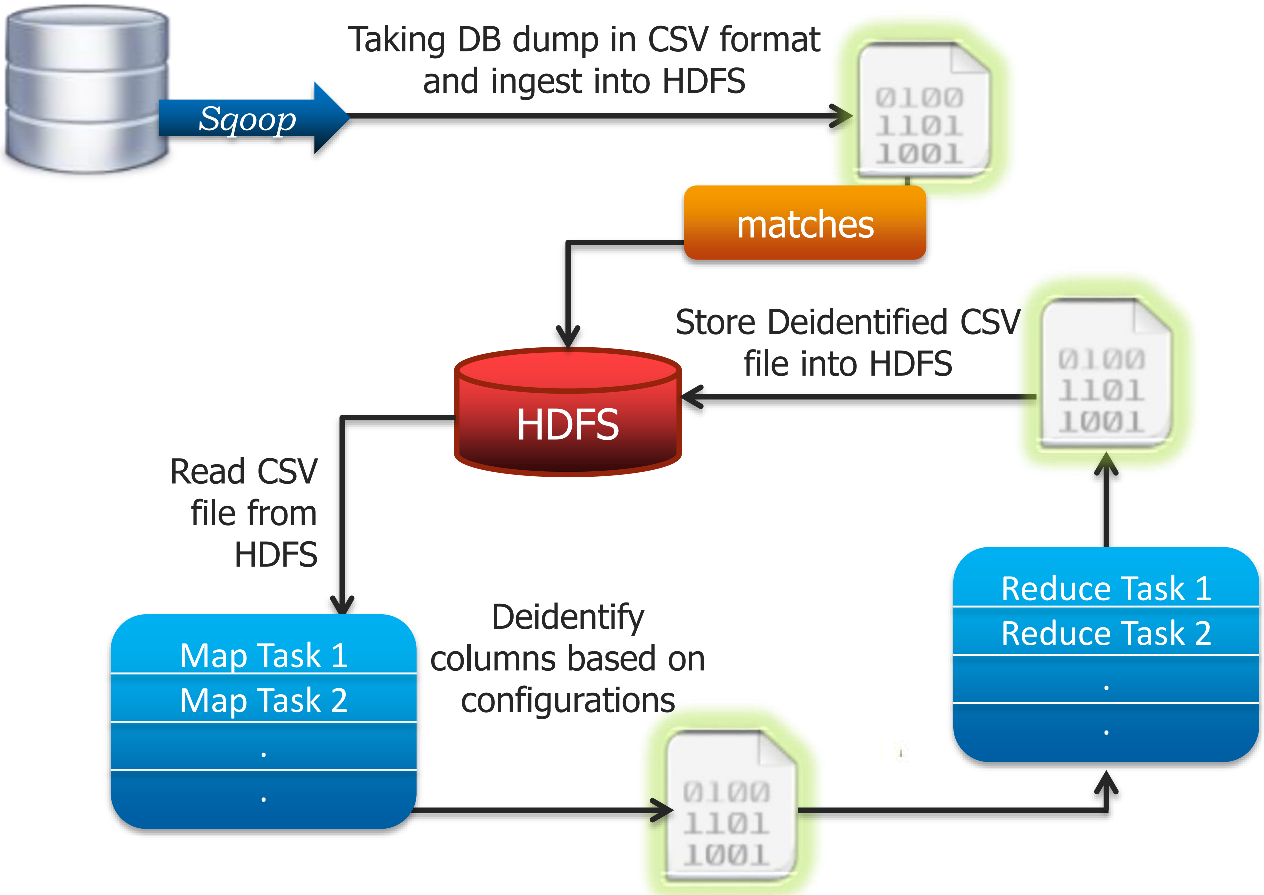
Reduce:

(K2, list (V2)) List (K3, V3)

MapReduce Way

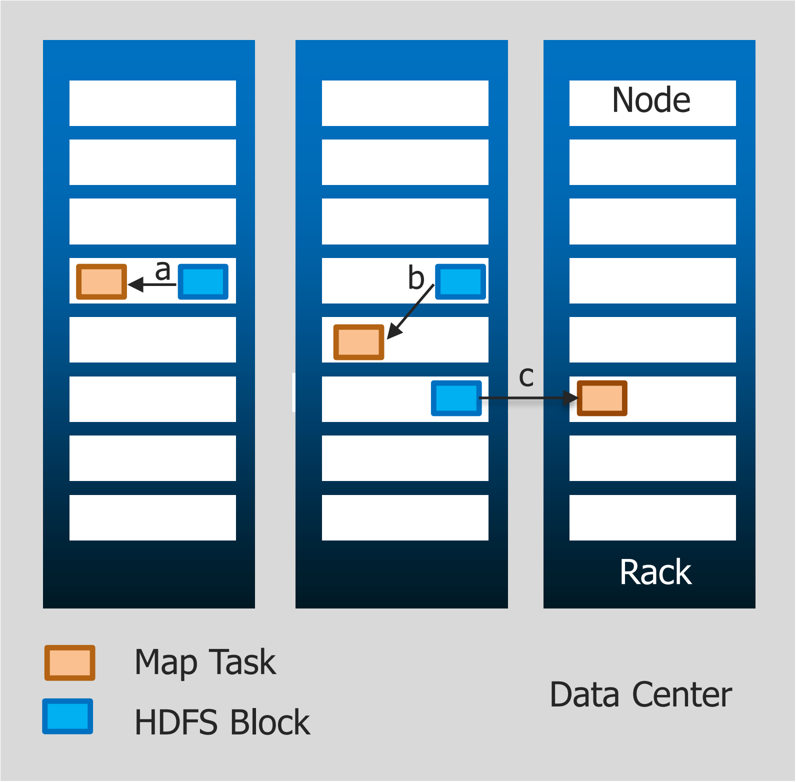


Revisit de-identification Architecture

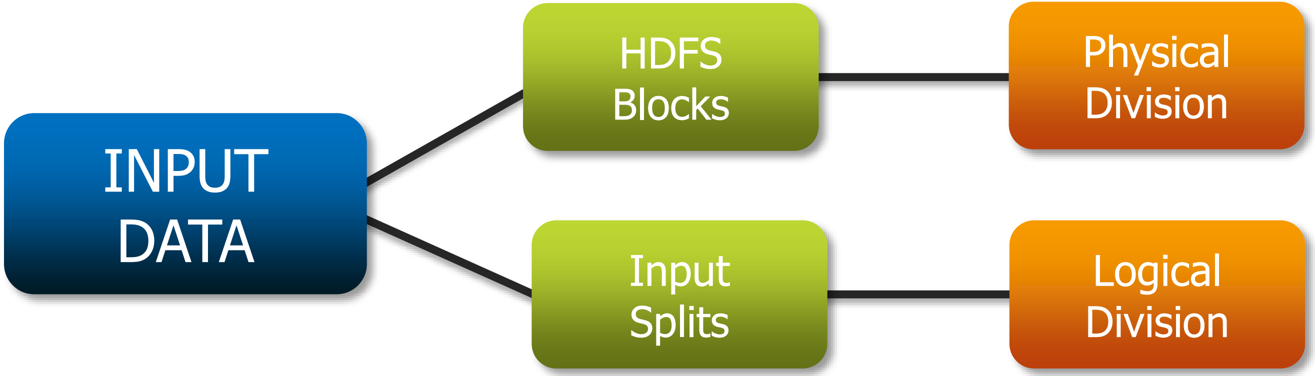

Sqoop
Taking DB dump in CSV format and ingest into HDFS
0100
1101
1001
matches
Store Deidentified CSV file into HDFS
HDFS
Read CSV file from HDFS
Map Task 1
Map Task 2
.
.
Deidentify columns based on configurations
Reduce Task 1
Reduce Task 2
.
.

Why MapReduce?

- ✓ Two biggest Advantages:
  - ✓ Taking processing to the data
  - ✓ Processing data in parallel

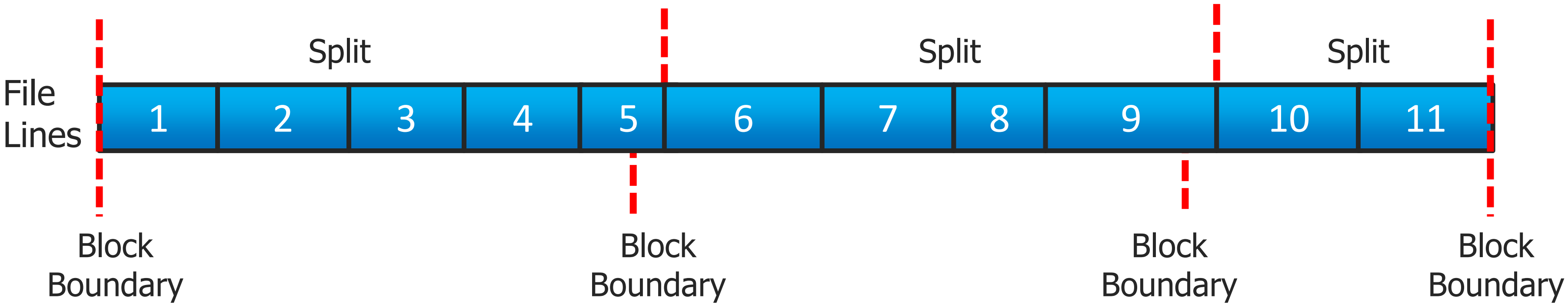


Input Splits



Relation between Input Splits and HDFS Blocks

edureka!



- ✓ Logical records do not fit neatly into the HDFS blocks.
- ✓ Logical records are lines that cross the boundary of the blocks.
- ✓ First split contains line 5 although it spans across blocks.

MapReduce Flow

Input data is distributed to nodes

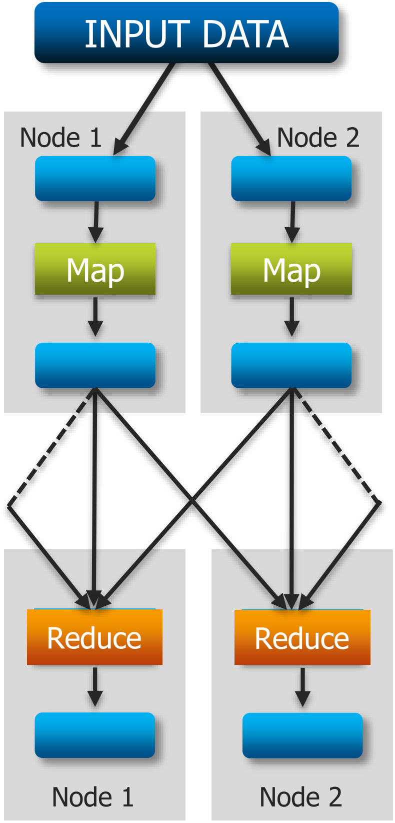
Each map task works on a “split” of data

Mapper outputs intermediate data

Data exchange between nodes in a “shuffle” process

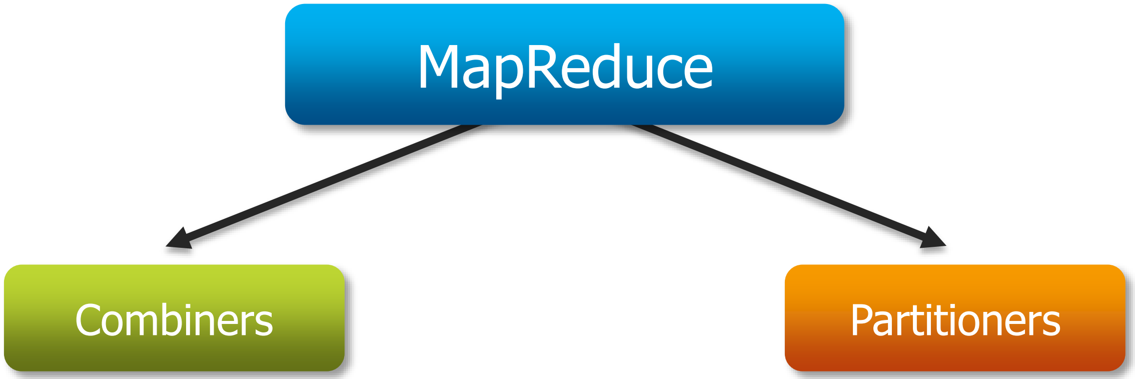
Intermediate data of the same key goes to the same reducer

Reducer output is stored



Overview of MapReduce

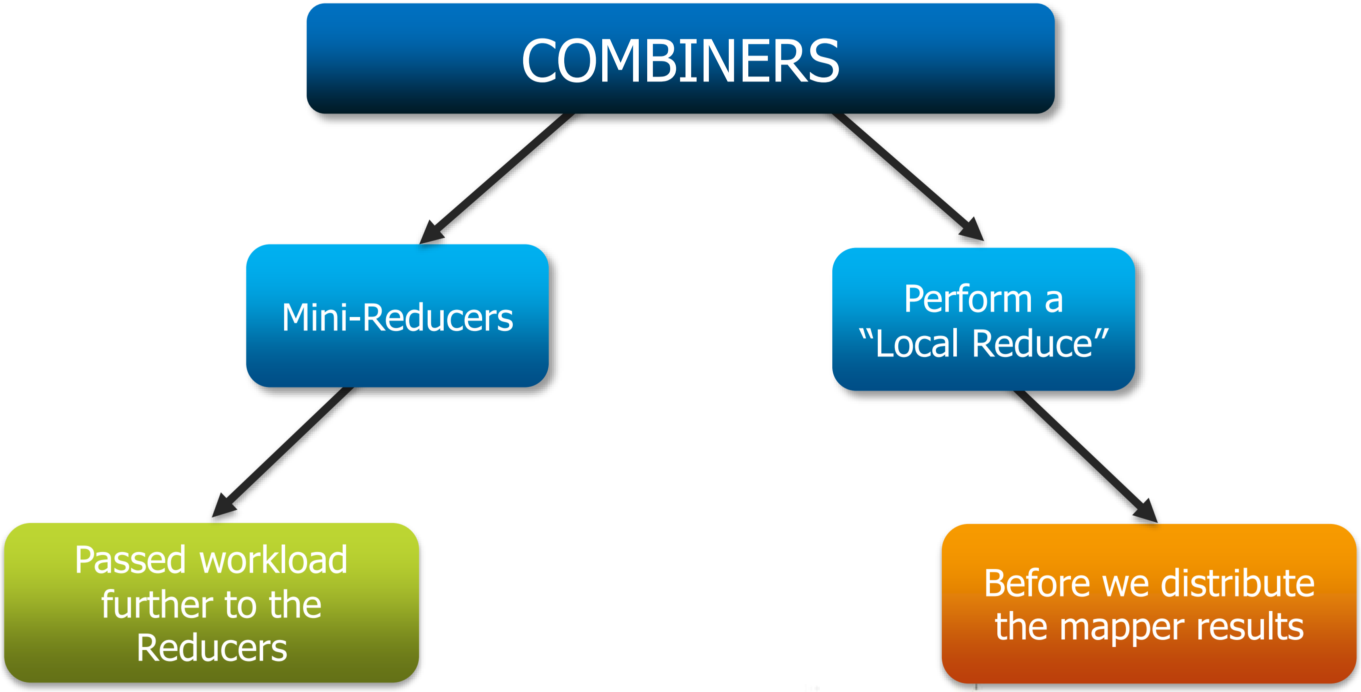
Complete view of MapReduce, illustrating combiners and partitioners in addition to Mappers and Reducers



Combiners can be viewed as ‘mini-reducers’ in the Map phase.

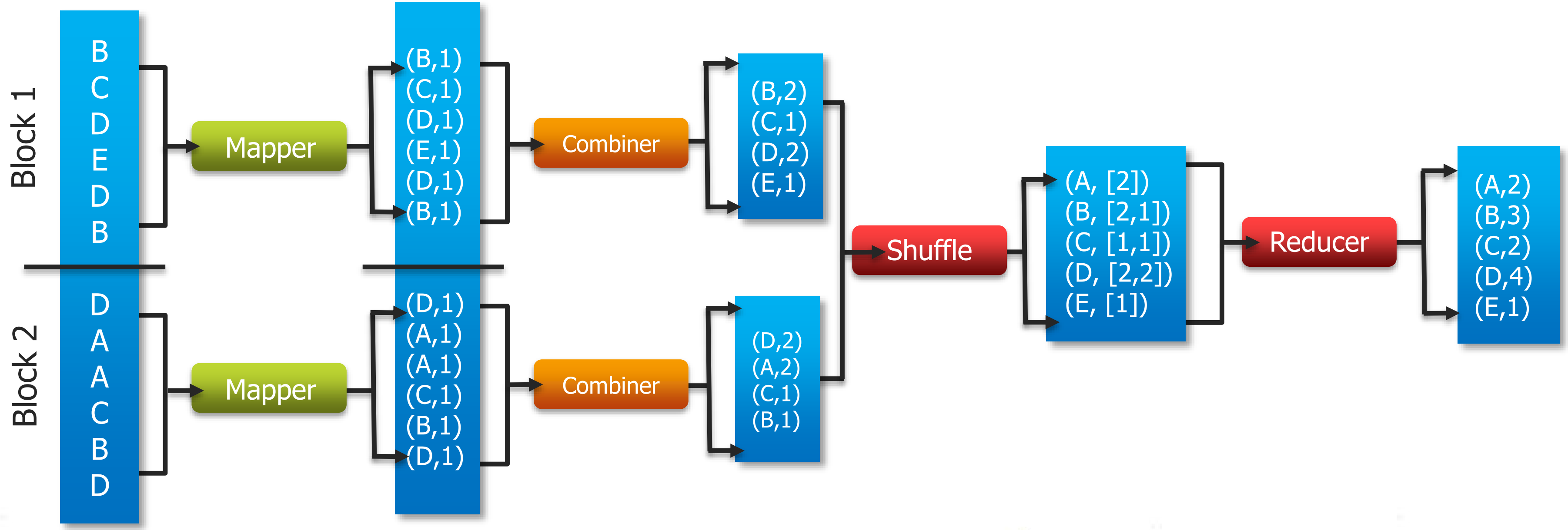
Partitioners determine which reducer is responsible for a particular key.

Combiner - Local Reduce

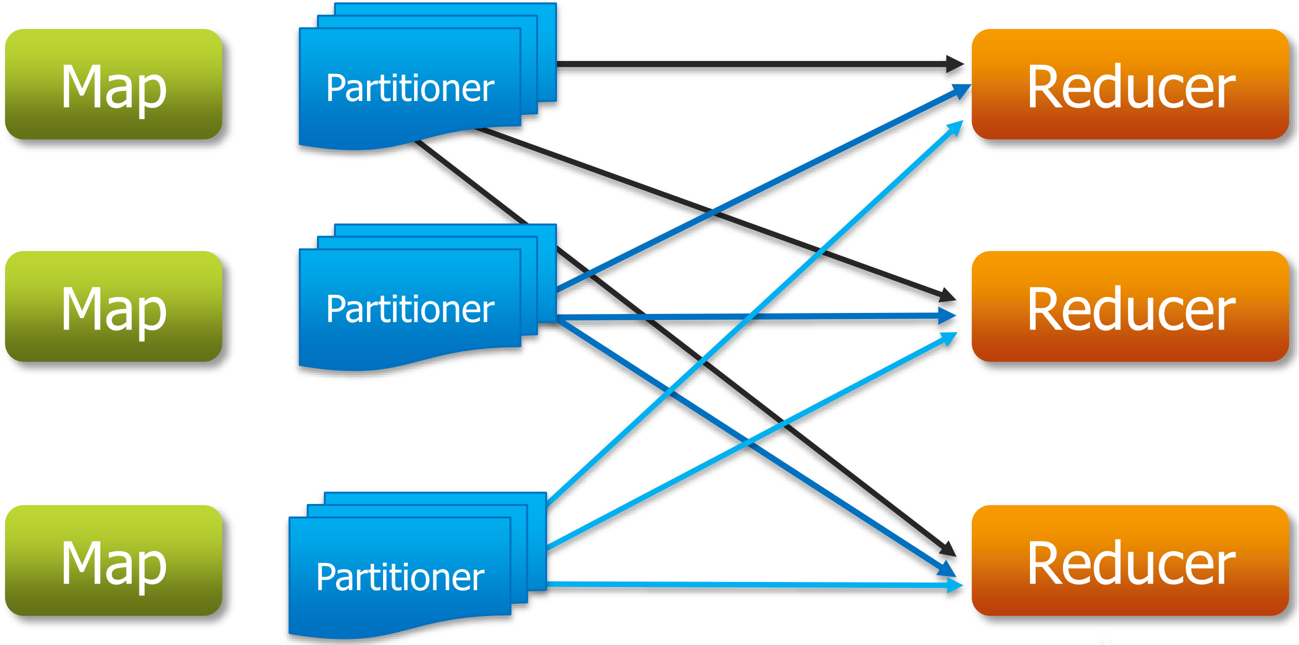

COMBINERS
Mini-Reducers
Perform a "Local Reduce"
Passed workload further to the Reducers
Before we distribute the mapper results

Combiner

edureka!

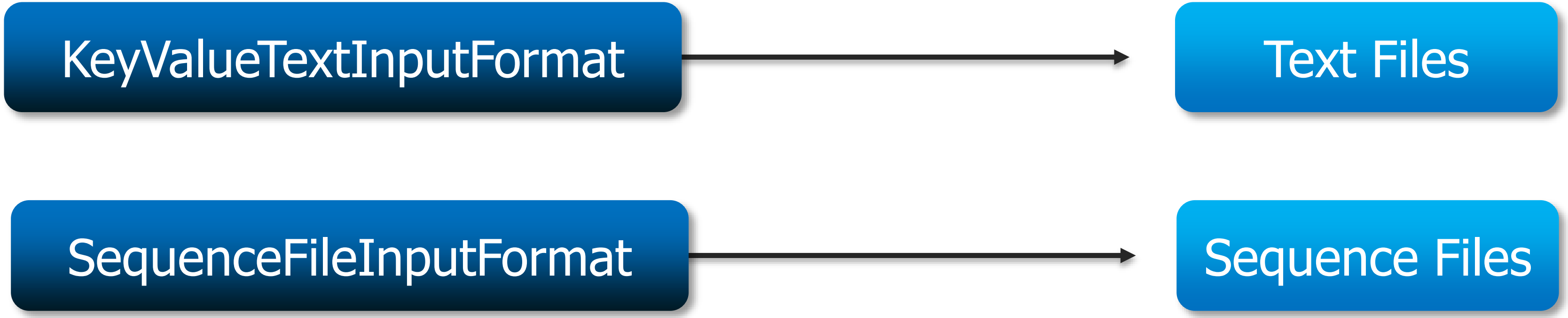


Partitioner— Redirecting Output from Mapper



Input Formats

Each line in the text files is a **record.**



KeyValueTextInputFormat → Text Files

SequenceFileInputFormat → Sequence Files

Assignment

Attempt the following assignment using the items present in the LMS under the tab Week 3:

- ✓ Watch video “Running MapReduce Program”.
- ✓ Execute Partitioner Code.
- ✓ Attempt Assignment Week 3 – Wordcount , Patents, Temperature ,Temperature2 & Alphabets

edureka!

Thank You

See You in Class Next Week