SNS COLLEGE OF TECHNOLOGY

(An Autonomous Institution)

Re-accredited by NAAC with A+ grade, Accredited by NBA(CSE, IT, ECE, EEE & Mechanical)
Approvedy by AICTE, New Delhi, Recognized by UGC, Affiliated to Anna University, Chennai

Department of MCA

Topic: Components of Hadoop

Course

16CA817
Big Data Analytics

Unit II

Hadoop

Elective

V Semester /
III MCA

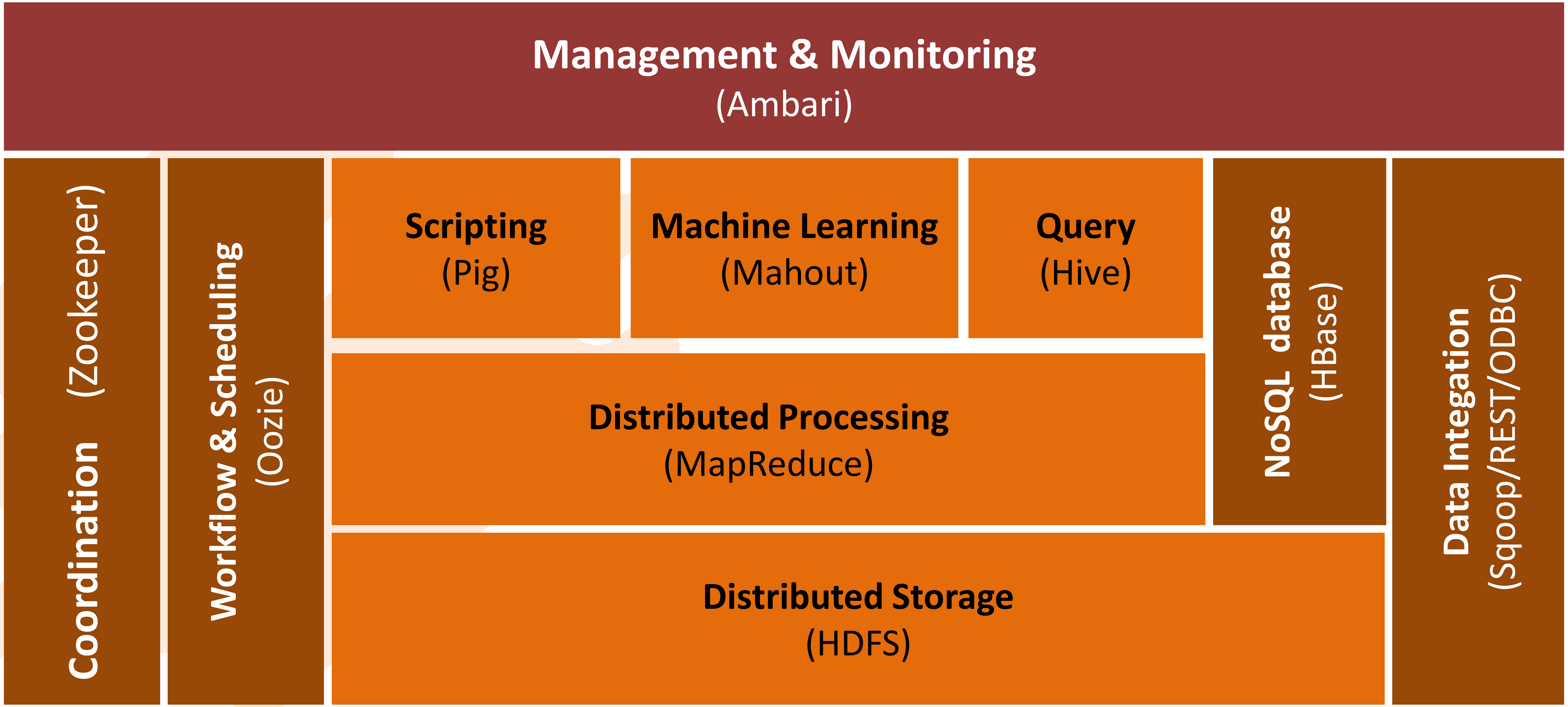
Learning Outcome

- ❑ Know the components that constitutes Hadoop Framework
- ❑ Identify the components used for Process logic
- ❑ Gain knowledge on backend supporting components

Components of Hadoop

- ❑ Hadoop is designed for parallel processing into a distributed environment
- ❑ Core Components: HDFS (storage) and MapReduce (processing)
  - HDFS – Distributed file system
  - MapReduce – framework on file system to process data

Hadoop Eco system



MapReduce

Programming model/framework to perform distributed and parallel processing on large data sets in a distributed environment

Intelligently distribute the computation over the cluster of nodes

Hadoop uses MapReduce to execute user jobs on files in HDFS

Hadoop uses MapReduce to execute user jobs on files in HDFS

MapReduce

Map function

- ❑ input data is in the form of a (key, value) pair
- ❑ The output is called intermediate (key, value) pairs
- ❑ Here, the goal is to process all input (key, value) pairs to the Map function in parallel

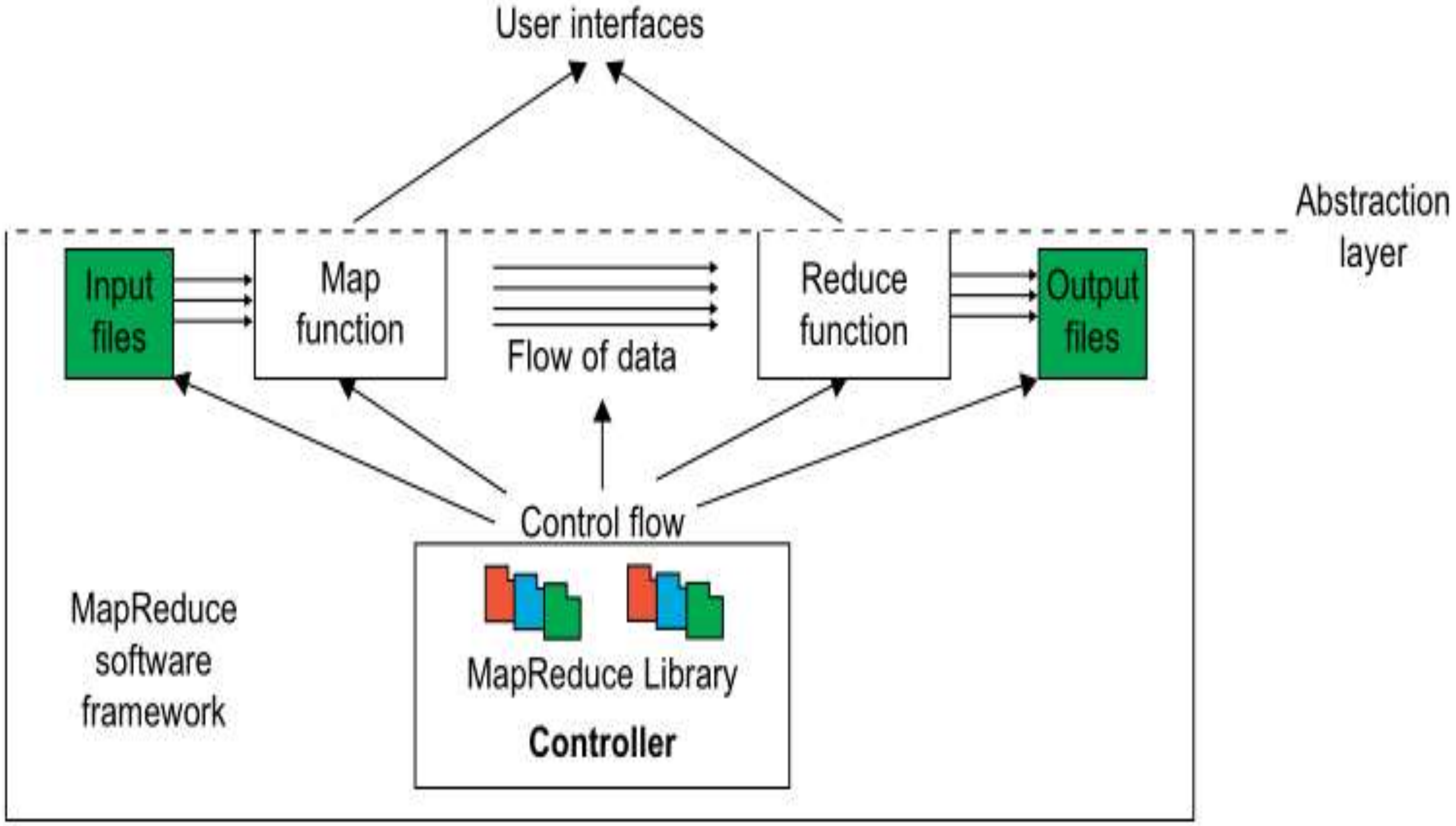
Reduce function

- ❑ Reduce function receives the intermediate (key, value) pairs in the form of a group of intermediate values associated with one intermediate key, (key, [set of values])

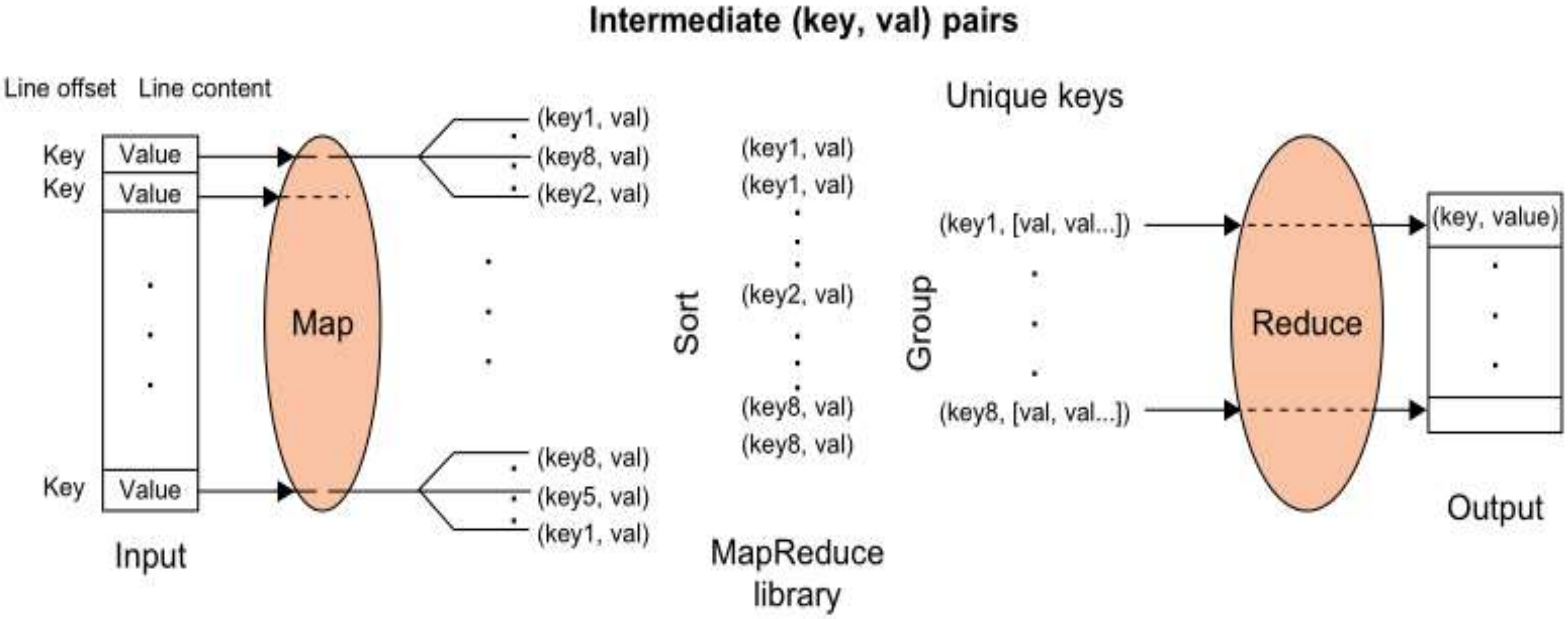
MapReduce Framework

- ❑ Groups by first sorting the intermediate (key, value) pairs and then grouping values with the same key
- ❑ Reduce function processes each (key, [set of values]) group and produces a set of (key, value) pairs as output

MapReduce – Logical flow



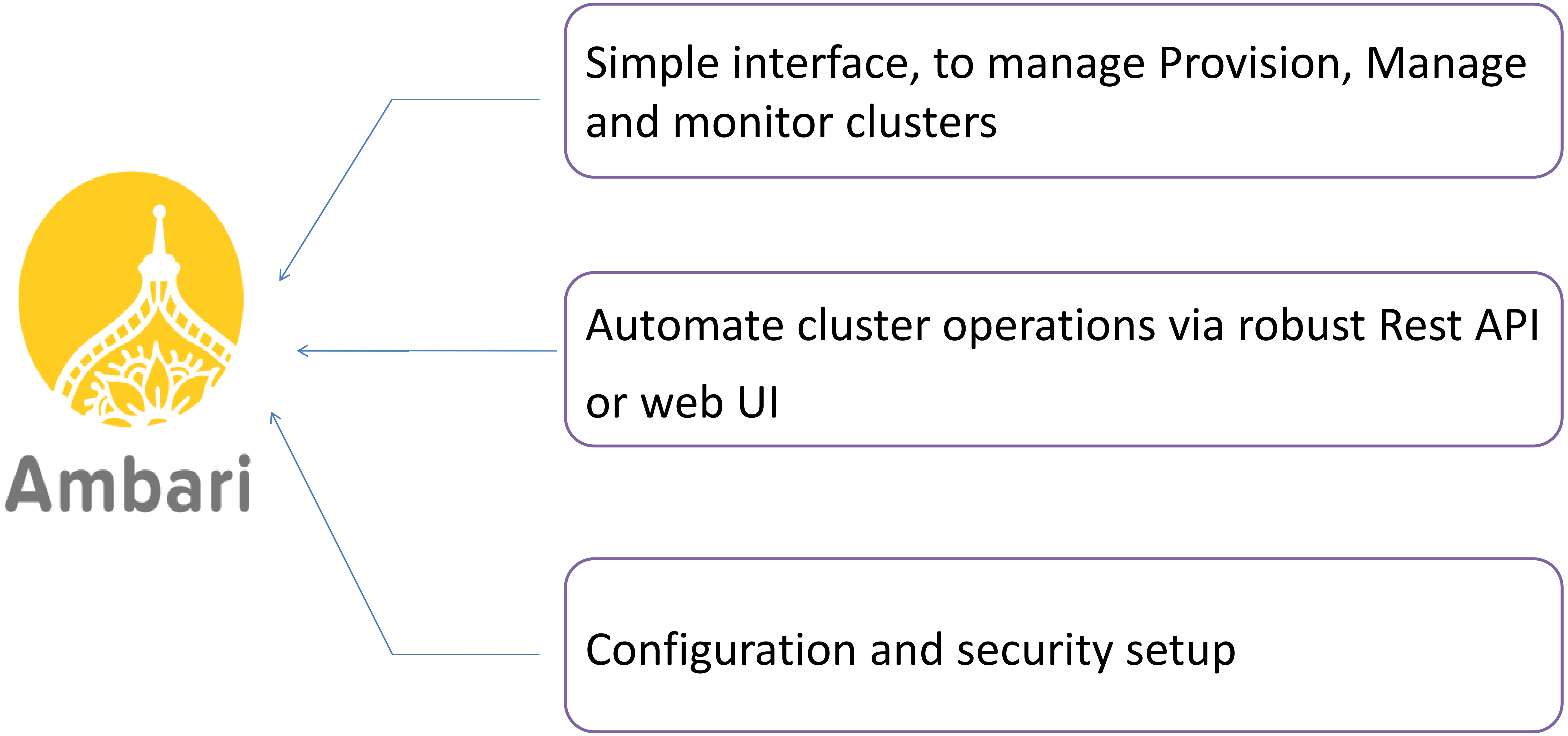
MapReduce – Logical flow



$$(key_1, val_1) \xrightarrow{\text{Map Function}} \text{List}\,(key_2, val_2)$$

$$(key_2, \text{List}\,(val_2)) \xrightarrow{\text{Reduce Function}} \text{List}\,(val_2)$$

Other Components



Simple interface, to manage Provision, Manage and monitor clusters

Automate cluster operations via robust Rest API or web UI

Configuration and security setup

Hive and Pig

❑ Hive: data warehousing application in Hadoop

- Query language is HQL, variant of SQL
- Tables stored on HDFS as flat files
- Developed by Facebook, now open source

❑ Pig: large-scale data processing system

- Scripts are written in Pig Latin, a dataflow language
- Developed by Yahoo!, now open source
- Roughly 1/3 of all Yahoo! internal jobs

❑ Common idea:

- Provide higher-level language to facilitate large-data processing
- Higher-level language “compiles down” to Hadoop jobs

Pig takes care of…

- ❑ Schema and type checking
- ❑ Translating into efficient physical dataflow
  - (i.e., sequence of one or more MapReduce jobs)
- ❑ Exploiting data reduction opportunities
  - (e.g., early partial aggregation via a combiner)
- ❑ Executing the system-level dataflow
  - (i.e., running the MapReduce jobs)
- ❑ Tracking progress, errors, etc

HBase

Distributed column-oriented data store built on top of HDFS

scale to hundreds or thousands of nodes

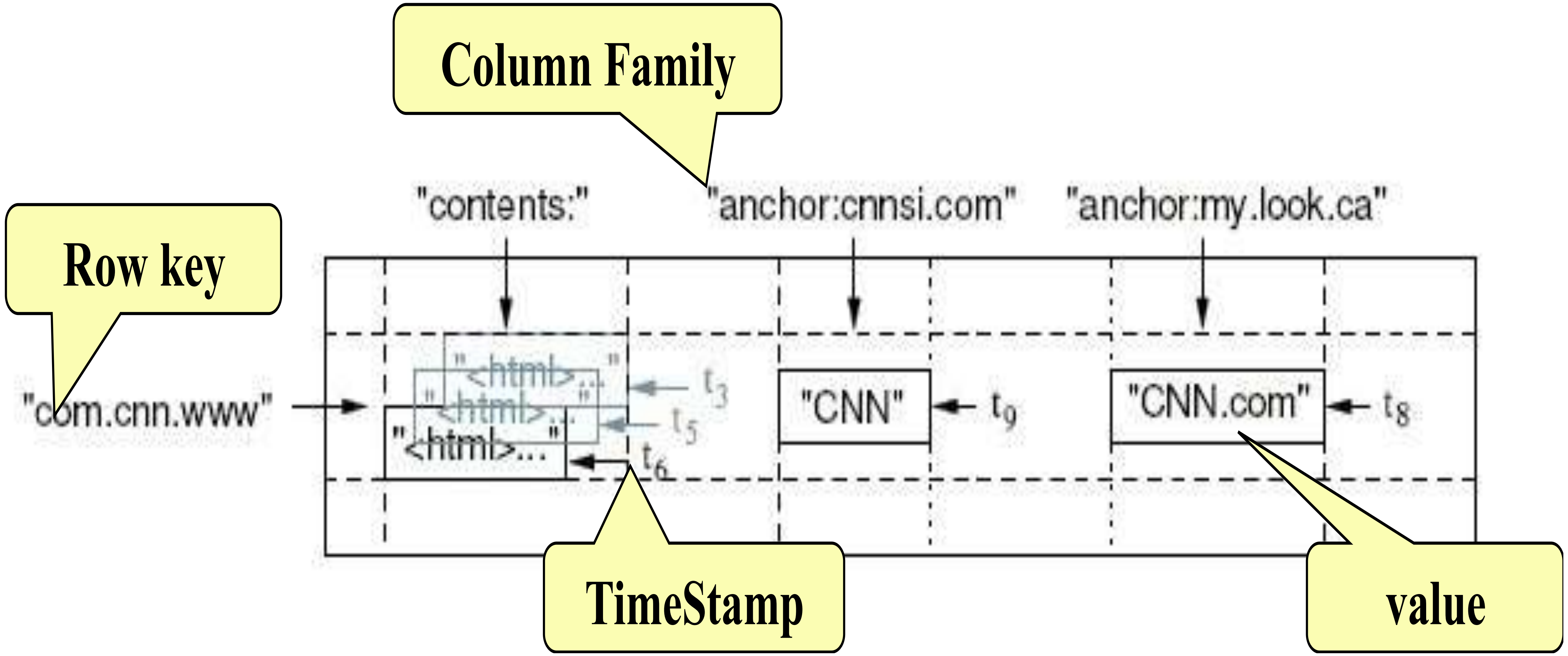
Data is logically organized into tables, rows & columns

Designed to efficiently address

- Fast record lookup
- Support for record-level insertion
- Support for updates (not in place

Hbase Data Model

- HBase is based on Google’s Bigtable model
  - Key-Value pairs



Zookeeper

A distributed, highly available coordination service

Runs on a collection of machines and is designed to be highly available

Provides primitives such as distributed locks, used to build a large class of coordination data structures and protocols

Provides an open source, shared repository of implementations and recipes of common coordination patterns (protocols)

Examples include: distributed queues, distributed locks, and leader election among a group of peers

Sqoop

Open-source tool to extract data from a relational database into Hadoop for further processing

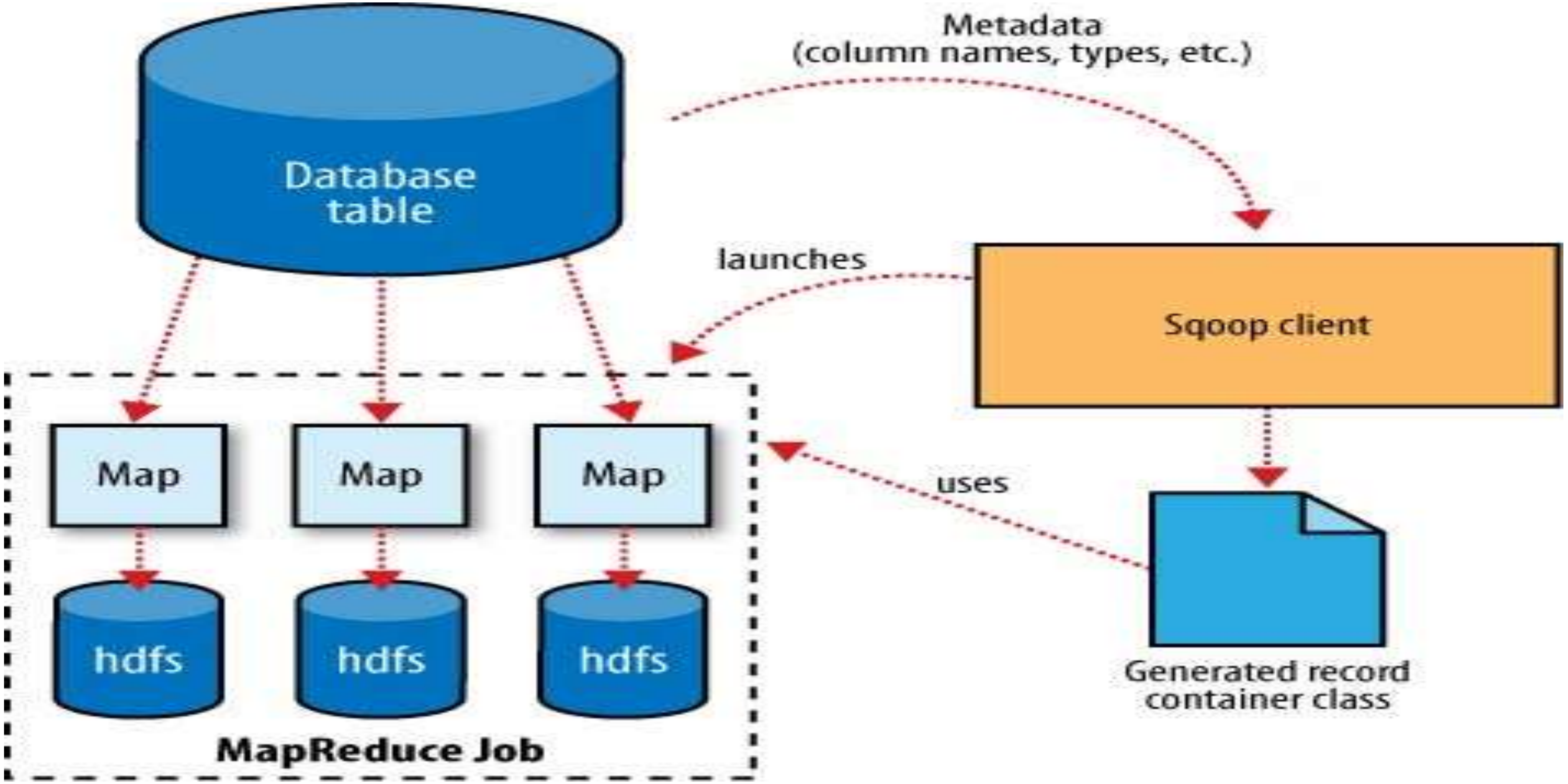
Final results of an analytic pipeline can export back to the database

JDBC, allows to access data in an RDBMS as well as inspect the nature of this data

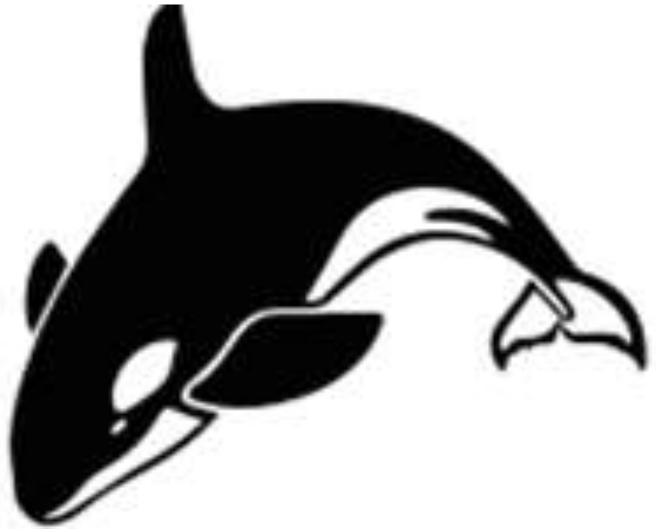
Sqoop uses JDBC to examine the table and retrieve list of cols and data type

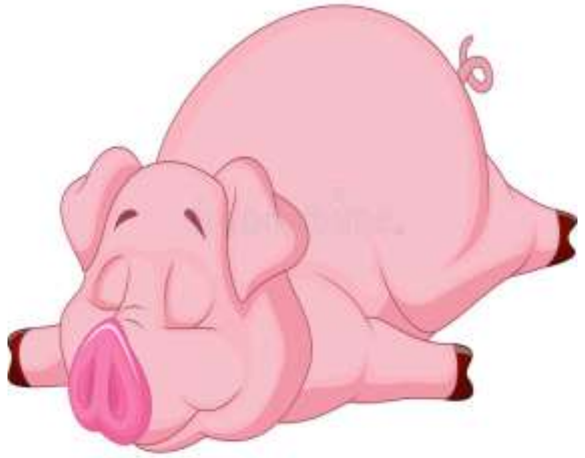
Sqoop's code generator will use this information to create a table-specific class

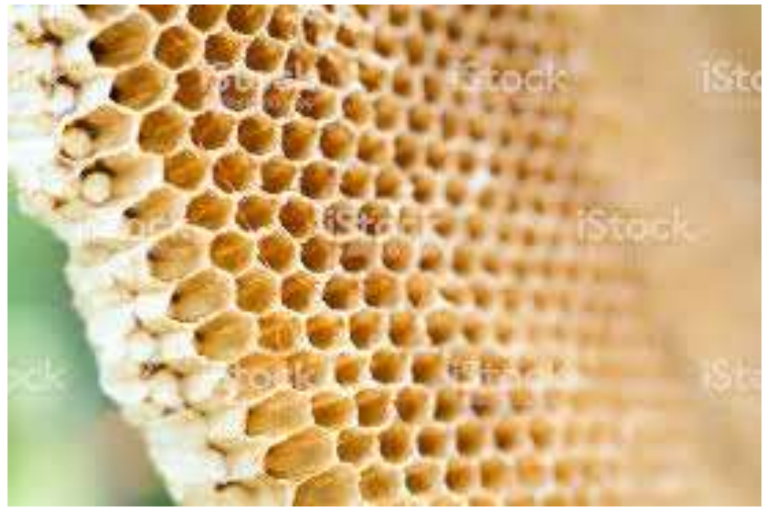
Sqoop

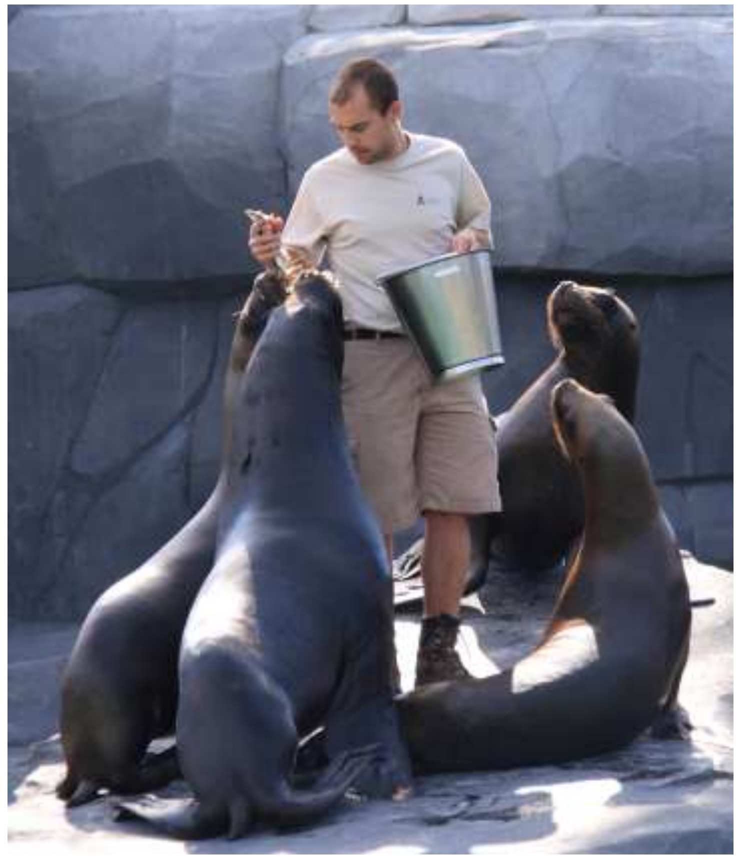

Metadata
(column names, types, etc.)
Database table
launches
Sqoop client
Map
Map
Map
uses
hdfs
hdfs
hdfs
Generated record container class
MapReduce Job

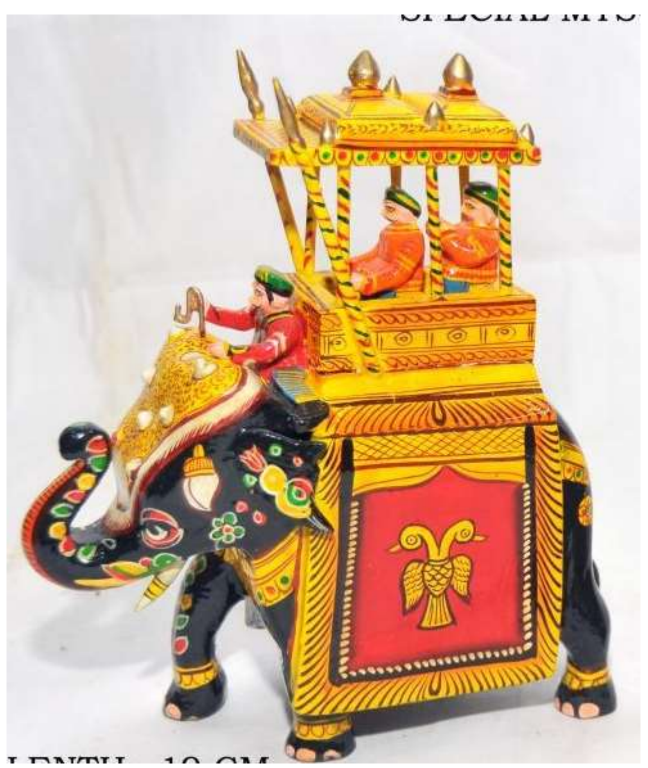
Assessment – Identify the association











References

- ❑ Tom White, “ Hadoop: The Definitive Guide” Third Edition, O’reilly Media, 4th Edition, 2012

Web Resources

- ❑ https://www.geeksforgeeks.org/hadoop-ecosystem/
- ❑ https://www.cloudera.com/products/open-source/apache-hadoop/hdfs-mapreduce-yarn.html
- ❑ https://bigdata-madesimple.com/basic-components-of-hadoop-architecture-frameworks-used-for-data-science/

thank
you!