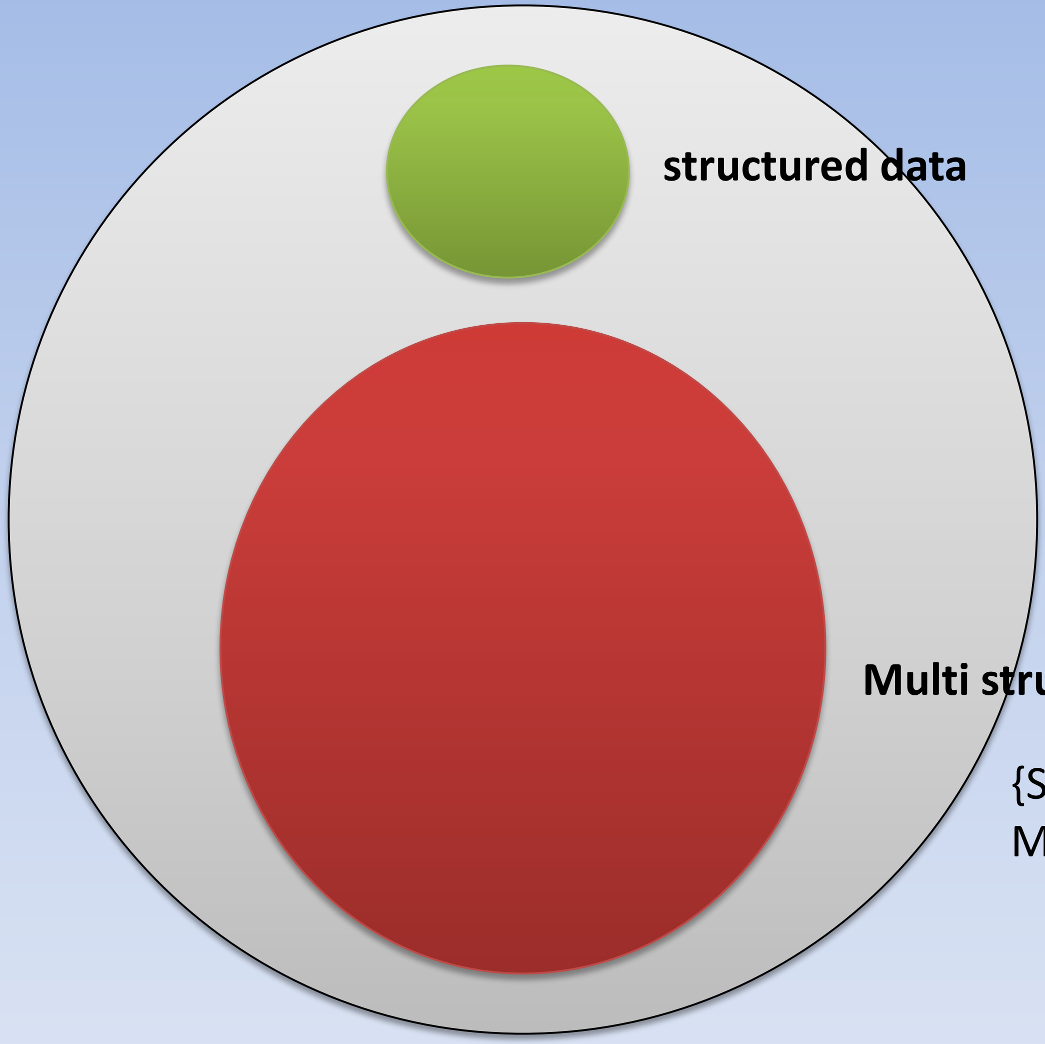
{Python} in Big Data World

Objectives

- ***What is Bigdata***
- ***What is Hadoop and its Ecosystem***
- ***Writing Hadoop jobs using Map Reduce programming***



structured data

{Relational data with well defined schemas}

Multi structured data

{Social data, blogs , click stream, Machine generated, xml etc…}

Trends … Gartner

Mobile analytics

Mobility

App stores and Market place

Human computer interface

Big Data

Personal cloud

In memory computing

Multi touch UI

Advanced Analytics

Flash Memory

Green data centre

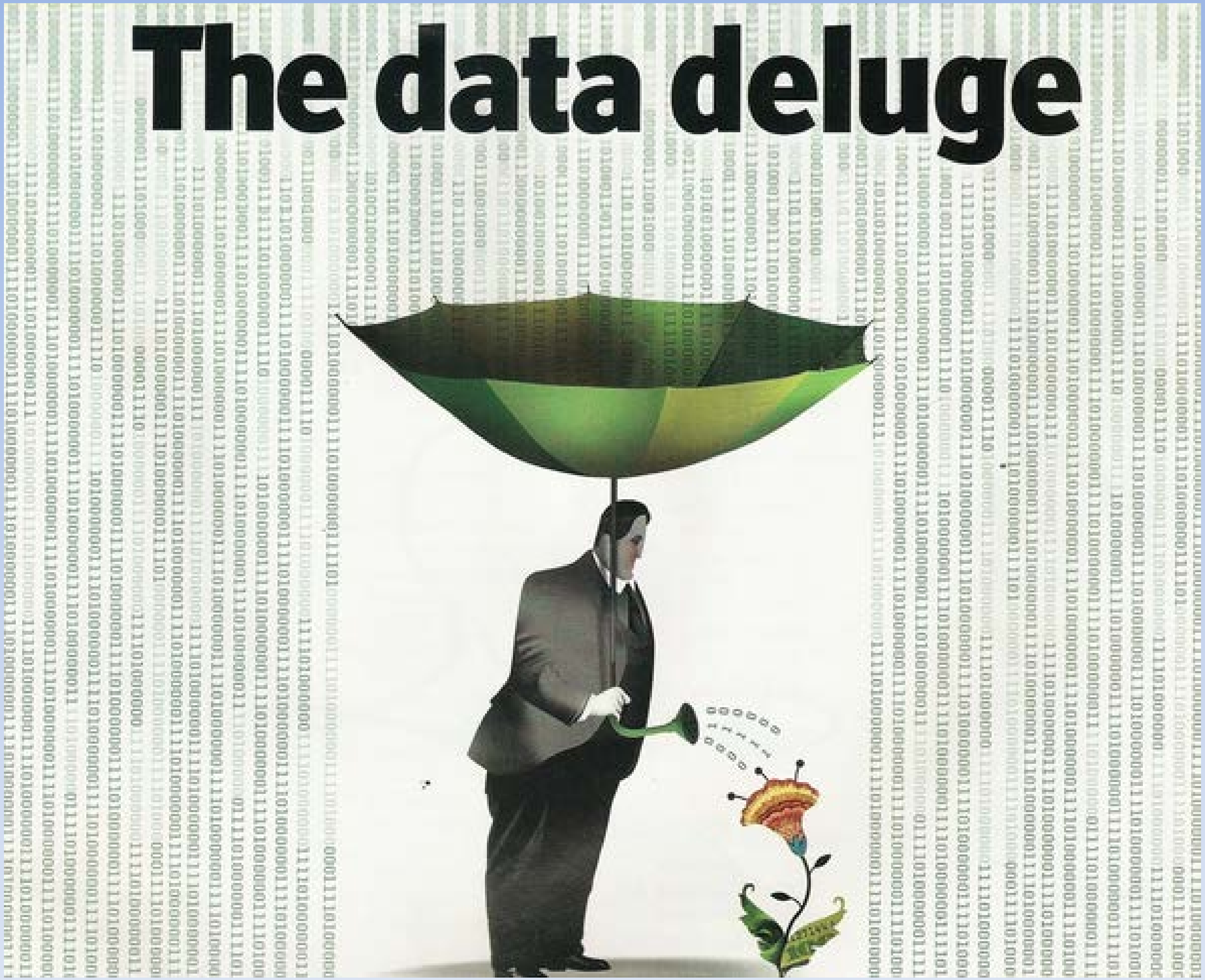
Social CRM

Solid state drive

HTML5

Context aware computing

The Problem…



Source : The Economist

The Problem…

Facebook

955 million active users as of March 2012, 1 in 3 Internet users have a Facebook account

More than 30 billion pieces of content (web links, news stories, blog posts, notes, photo albums, etc.) shared each month.

Holds 30PB of data for analysis, adds 12 TB of compressed data daily

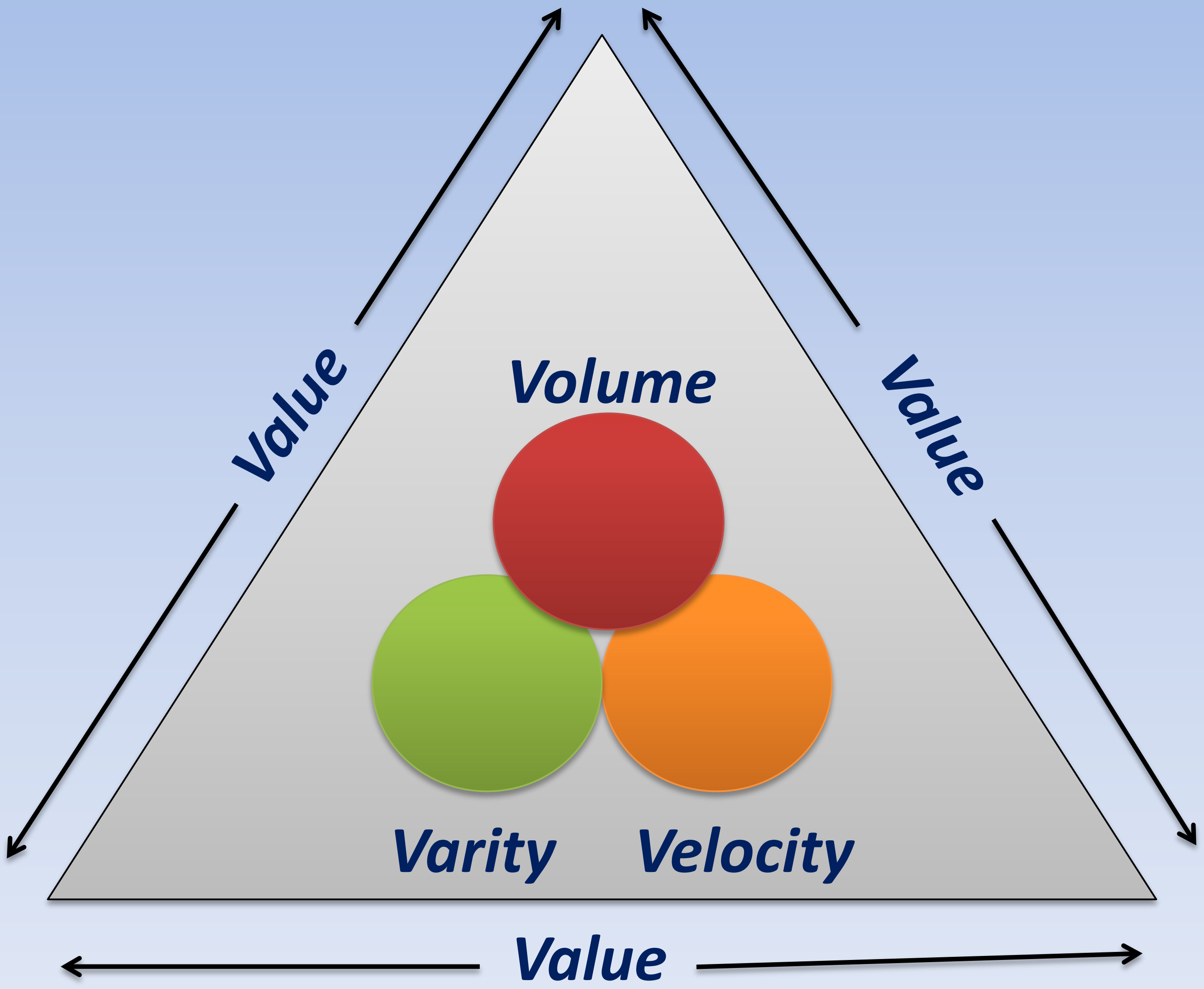
The Problem…

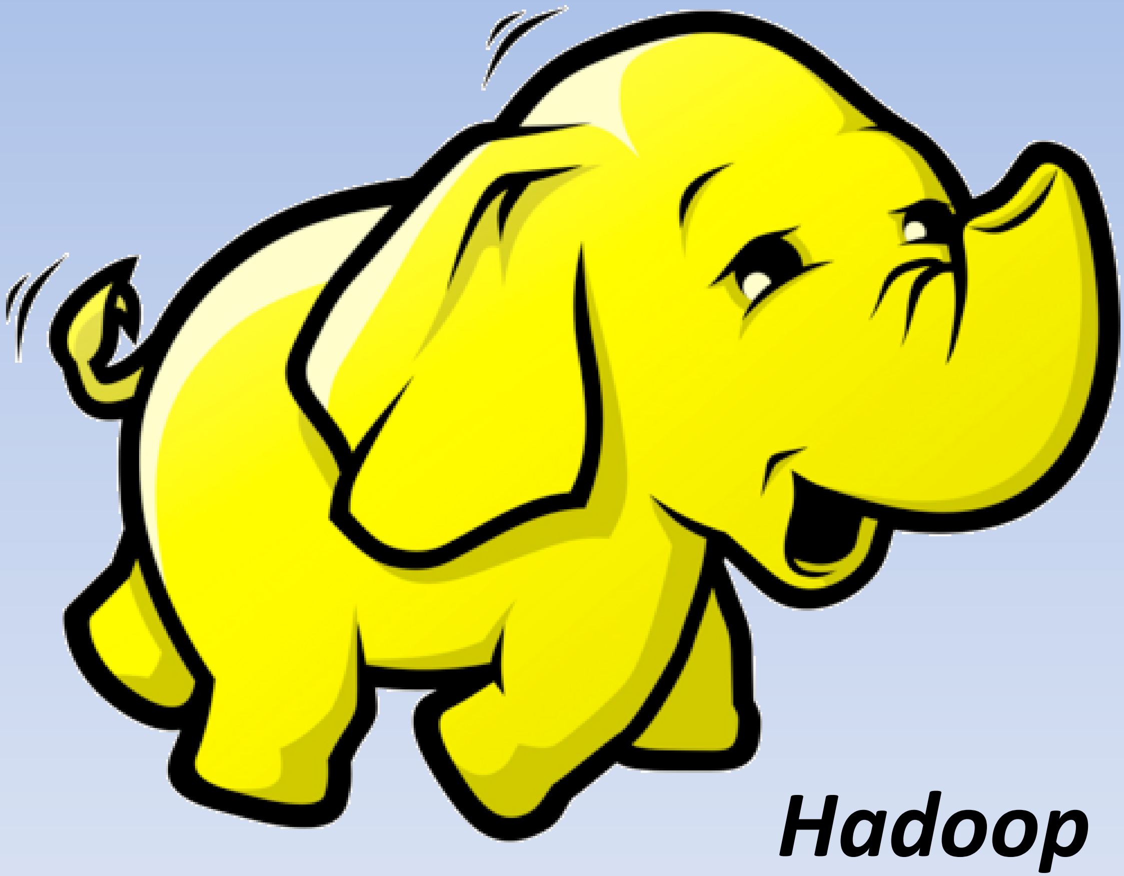
Twitter

500 million users, 340 million daily tweets
1.6 billion search queries a day
7 TB data for analysis generated daily

Traditional data storage, techniques & analysis tools just do not work at these scales !

Big Data Dimensions (V3)





Hadoop

What is Hadoop …

Flexible and available architecture for large scale distributed ***batch*** *processing on a network of commodity hardware.*

hadoop

Apache top level project

http://hadoop.apache.org/

500 contributors

It has one of the strongest eco systems with large no of sub projects

Yahoo has one of the biggest installation Hadoop
Running 1000s of servers on Hadoop

Inspired by …

{Google GFS + Map Reduce + Big Table}

Architecture behind Google's

Search Engine

Creator of Hadoop project

Use cases … What is Hadoop used for

Big/Social data analysis

Text mining, patterns search

Machine log analysis

Geo-spacitial analysis

Trend Analysis

Genome Analysis

Drug Discovery

Fraud and compliance management

Video and image analysis

Who uses Hadoop … long list

- ***Amazon/A9***
- ***Facebook***
- ***Google***
- ***IBM***
- ***Disney***
- ***Last.fm***
- ***New York Times***
- ***Yahoo!***
- ***Twitter***
- ***Linked in***

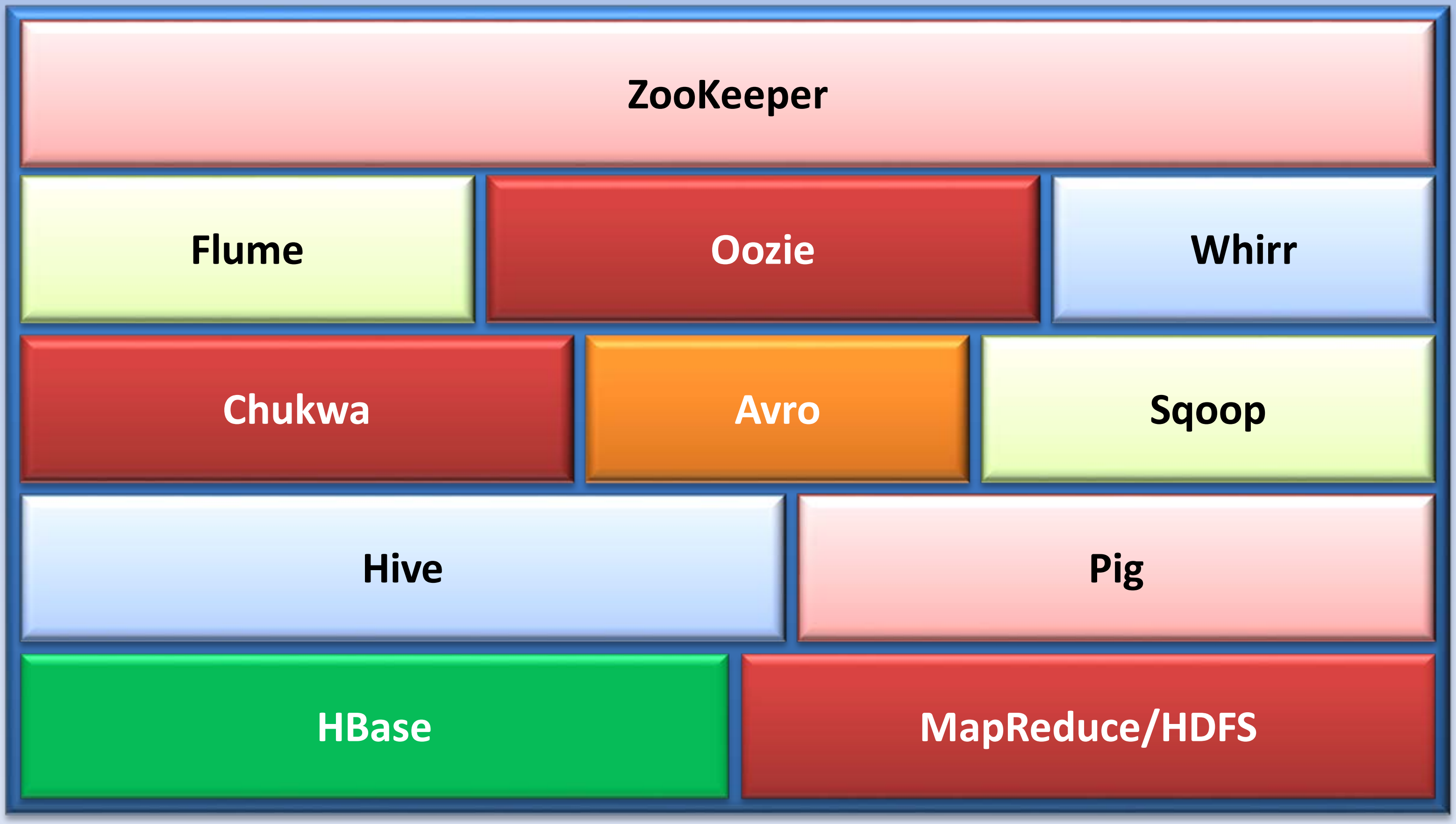
What is Hadoop used for?

- *Search*
  Yahoo, Amazon, Zvents
- *Log processing*
  Facebook, Yahoo, ContextWeb, Last.fm
- *Recommendation Systems*
  Facebook , Disney
- *Data Warehouse*
  Facebook, AOL , Disney
- *Video and Image Analysis*
  New York Times
- *Computing Carbon Foot Print*
  Opower.

Our own …

ADDHAAR uses Hadoop and Hbase for its data processing …

Hadoop ecosystem …



ZooKeeper

Flume

Oozie

Whirr

Chukwa

Avro

Sqoop

Hive

Pig

HBase

MapReduce/HDFS

Hive: *Datawarehouse infrastructure built on top of hadoop for data summarization and aggregation of data more like in sql like language called as hiveQL.*

Hbase: *Hbase is a Nosql columnar database and is an implementation of Google Bigtable. It can scale to store billions of rows.*

Flume: *Apache **Flume** is a distributed, reliable, and available service for efficiently collecting, aggregating, and moving large amounts of log data*

Avro: *A data serialization system.*

Sqoop: *Used for transferring bulk data between Hadoop and traditional structured data stores.*

Hadoop distribution …

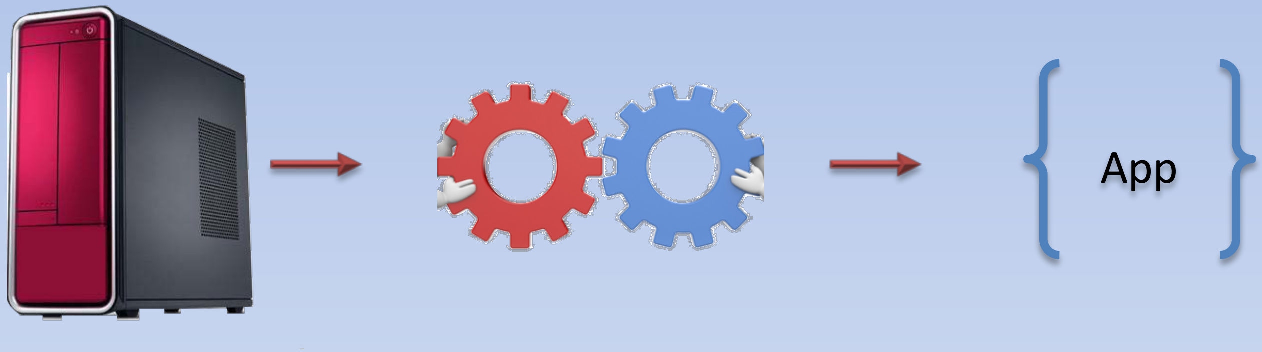
cloudera

Hortonworks

InfoSphere BigInsights

MAPR®
TECHNOLOGIES
EASY. DEPENDABLE. FAST.

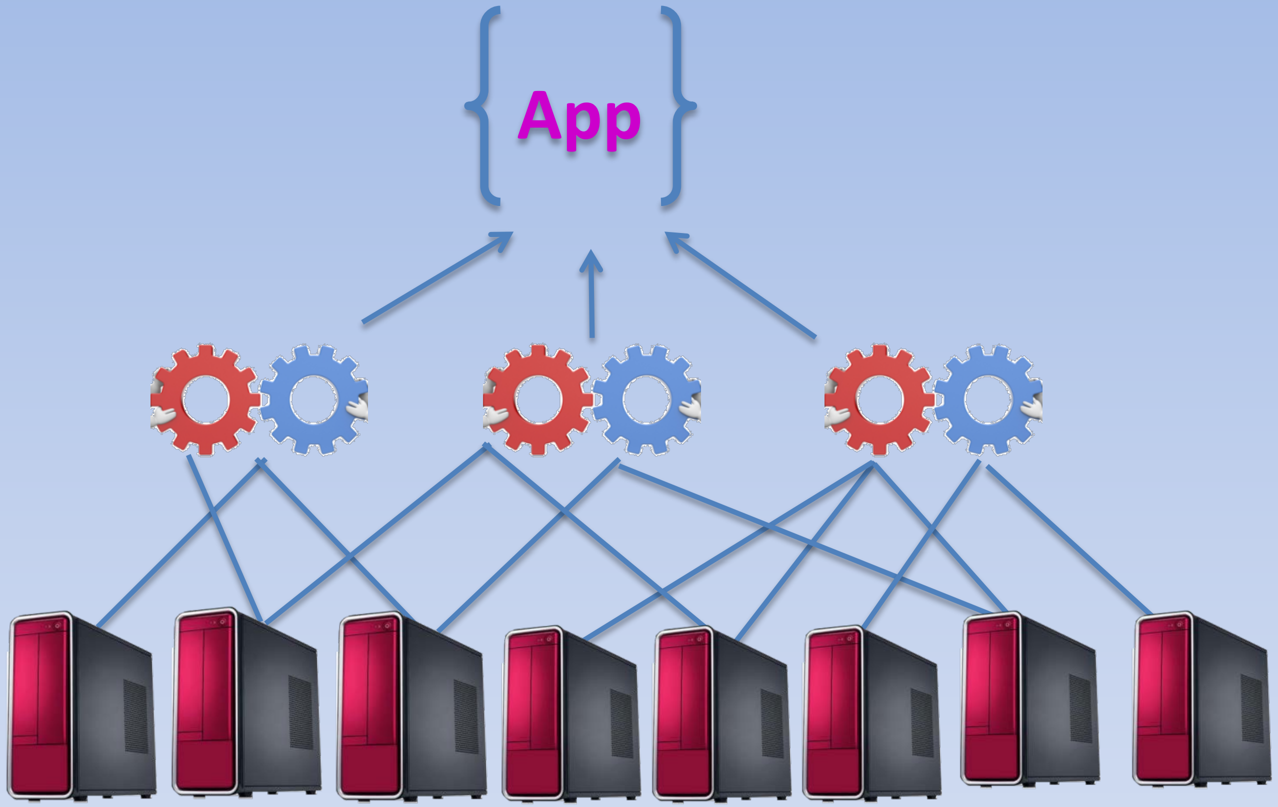
Scale up



App

Traditional Databases

Scale out



Hadoop distributed file system

Why Hadoop ?

How Hadoop is different from other parallel processing architectures such As MPI, OpenMP, Globus ?

Move compute to data in Hadoop
While in other parallel processing the
data gets distributed to compute.

Hadoop Components …

HDFS

Map Reduce

Job tracker

Task Tracker

Name Node

Python + Analytics

✓ *High Level Language*

✓ *Highly Interactive*

✓ *Highly Extensible*

✓ *Functional*

✓ *Many Extensible libs like*

- *SciPy*
- *NumPy*
- *Metaplotlib*
- *Pandas*
- *Ipython*
- *StatsModel*
- *Ntltk*

to name a few.

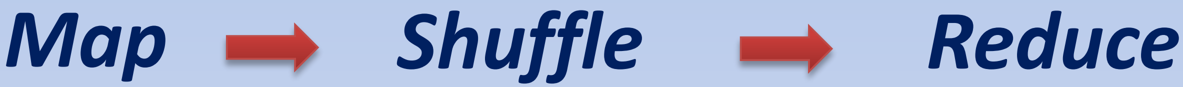
What is common between Mumbai Dabbawalas and Apache Hadoop

Source : Cloudstory.in
Author : Janakiram MSV

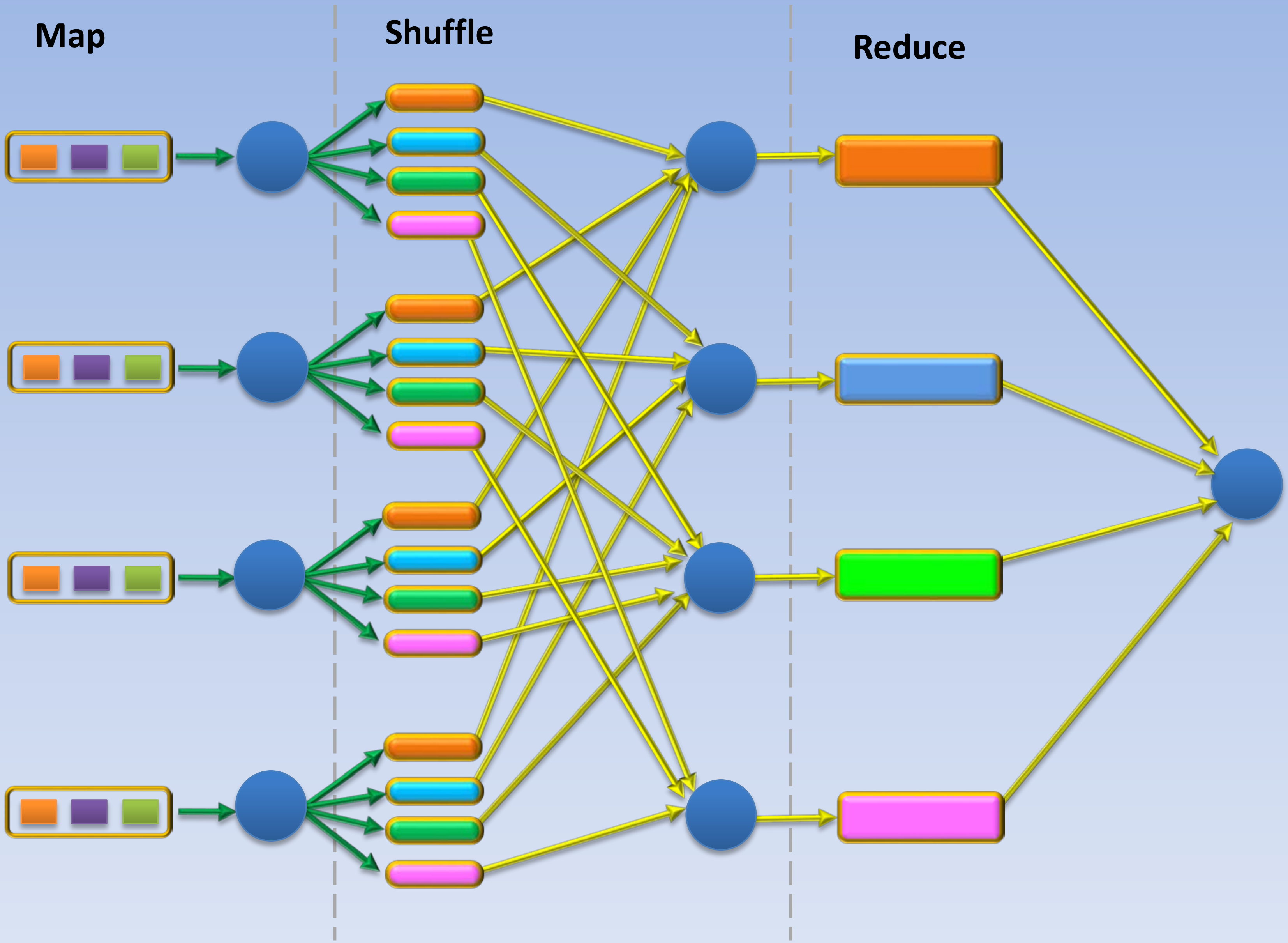
What is MapReduce

MapReduce is a programming model for processing large data sets on distributed computing.

Map reduce steps



Map → **Shuffle** → **Reduce**



Map

Shuffle

Reduce

Map Reduce

- *Java*
- *Hive*
- *Pig Scripts*
- *Datameer*
- *Cascading*
    - *Cascalog*
    - *Scalding*
- *Streaming frameworks*
    - *Wukong*
    - *Dumbo*
    - *MrJobs*
    - *Happy*

Pig Script (Word Count)

```
input_lines = LOAD '/tmp/my-copy-of-all-pages-on-internet' AS (line:chararray);

-- Extract words from each line and put them into a pig bag
-- datatype, then flatten the bag to get one word on each row
 words = FOREACH input_lines GENERATE FLATTEN(TOKENIZE(line)) AS word;

-- filter out any words that are just white spaces
 filtered_words = FILTER words BY word MATCHES '\\w+';

-- create a group for each word
word_groups = GROUP filtered_words BY word;

-- count the entries in each group
word_count = FOREACH word_groups GENERATE COUNT(filtered_words) AS
count, group AS word;

-- order the records by count
ordered_word_count = ORDER word_count BY count DESC;
 STORE ordered_word_count INTO '/tmp/number-of-words-on-internet';
```

Hive (WordCount)

```
CREATE TABLE input (line STRING);
 LOAD DATA LOCAL INPATH 'input.tsv' OVERWRITE INTO TABLE input;

 -- temporary table to hold words
 CREATE TABLE words (word STRING);

SELECT word, COUNT(*) FROM input LATERAL VIEW explode(split(text, ' ')) lTable
as word GROUP BY word;
```

Hadoop Streaming…

http://www.michael-noll.com/tutorials/writing-an-hadoop-mapreduce-program-in-python/

Map: mapper.py

```
#!/usr/bin/env python
import sys
# input comes from STDIN (standard input)
for line in sys.stdin:
     # remove leading and trailing whitespace
    line = line.strip()
    # split the line into words
    words = line.split()
    # increase counters
    for word in words:
        # write the results to STDOUT (standard output);
        # what we output here will be the input for the
        # Reduce step, i.e. the input for reducer.py
        #
        # tab-delimited; the trivial word count is 1
        print '%s\t%s' % (word, 1)
```

Reduce: reducer.py

```python
#!/usr/bin/env python
from operator import itemgetter
import sys

current_word = None
current_count = 0
word = None

# input comes from STDIN
for line in sys.stdin:
        # remove leading and trailing whitespace
        line = line.strip()

        # parse the input we got from mapper.py
        word, count = line.split('\t', 1)
```

Reduce: reducer.py (cont)

```
# convert count (currently a string) to int
try:
    count = int(count)
except ValueError:
    # count was not a number, so silently
    # ignore/discard this line
    continue

     # this IF-switch only works because Hadoop sorts map
    output
     # by key (here: word) before it is passed to the reducer
     if current_word == word:
         current_count += count
     else:
         if current_word:
             # write result to STDOUT
             print '%s\t%s' % (current_word, current_count)
             current_count = count
             current_word = word
```

Reduce: reducer.py (cont)

```python
# do not forget to output the last word if needed!
if current_word == word:
        print '%s\t%s' % (current_word, current_count)
```

Thank You