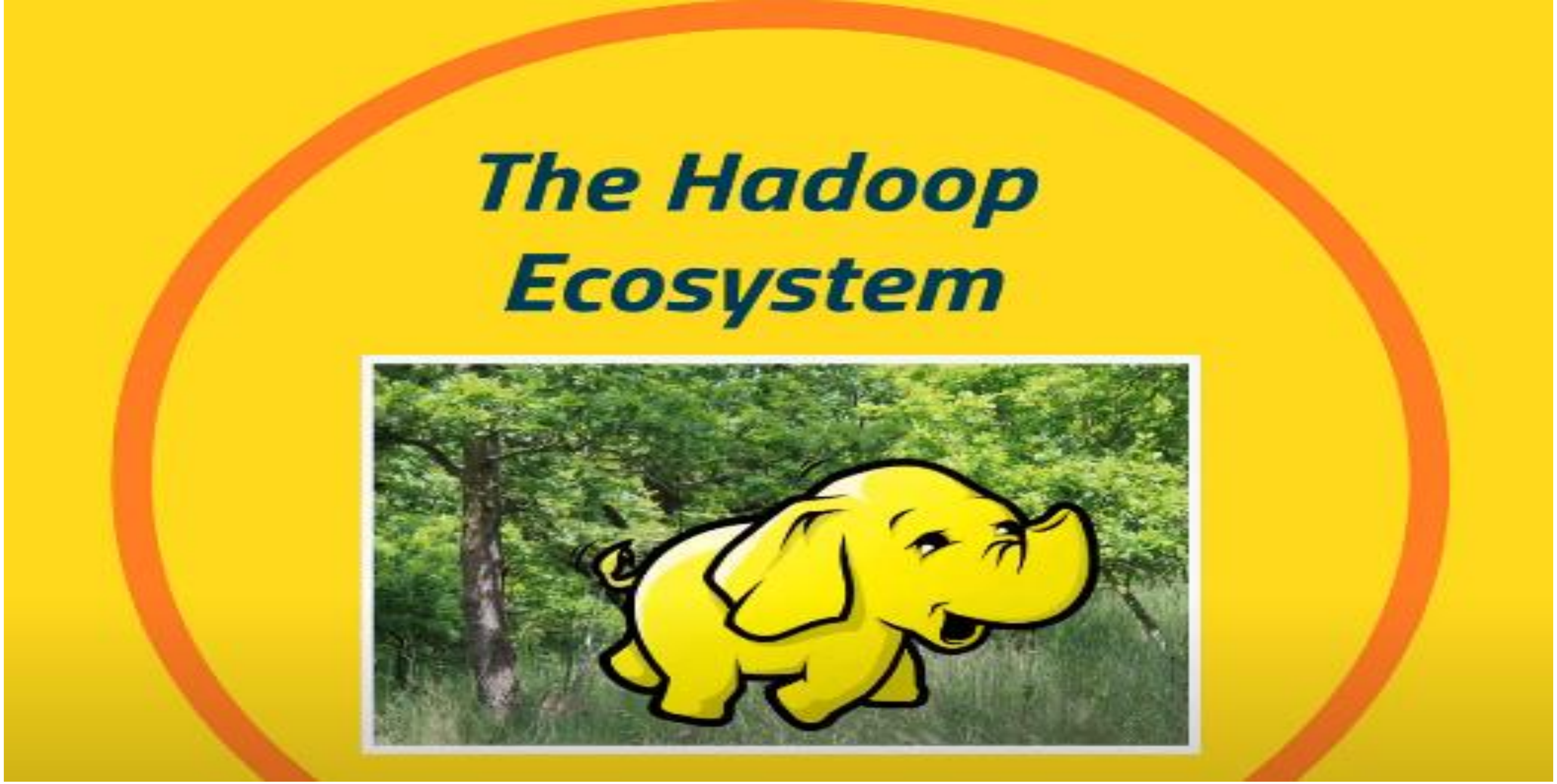
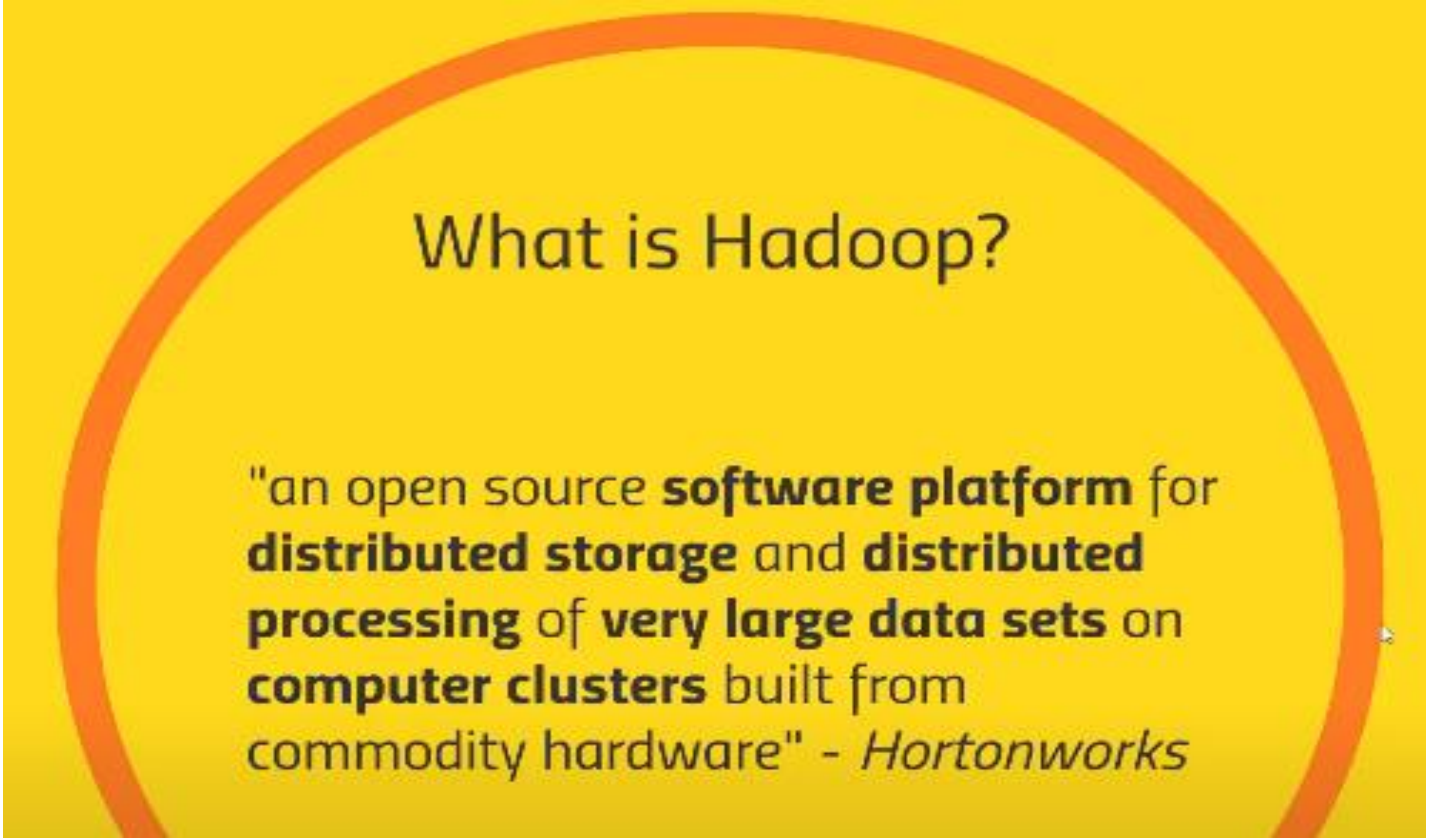


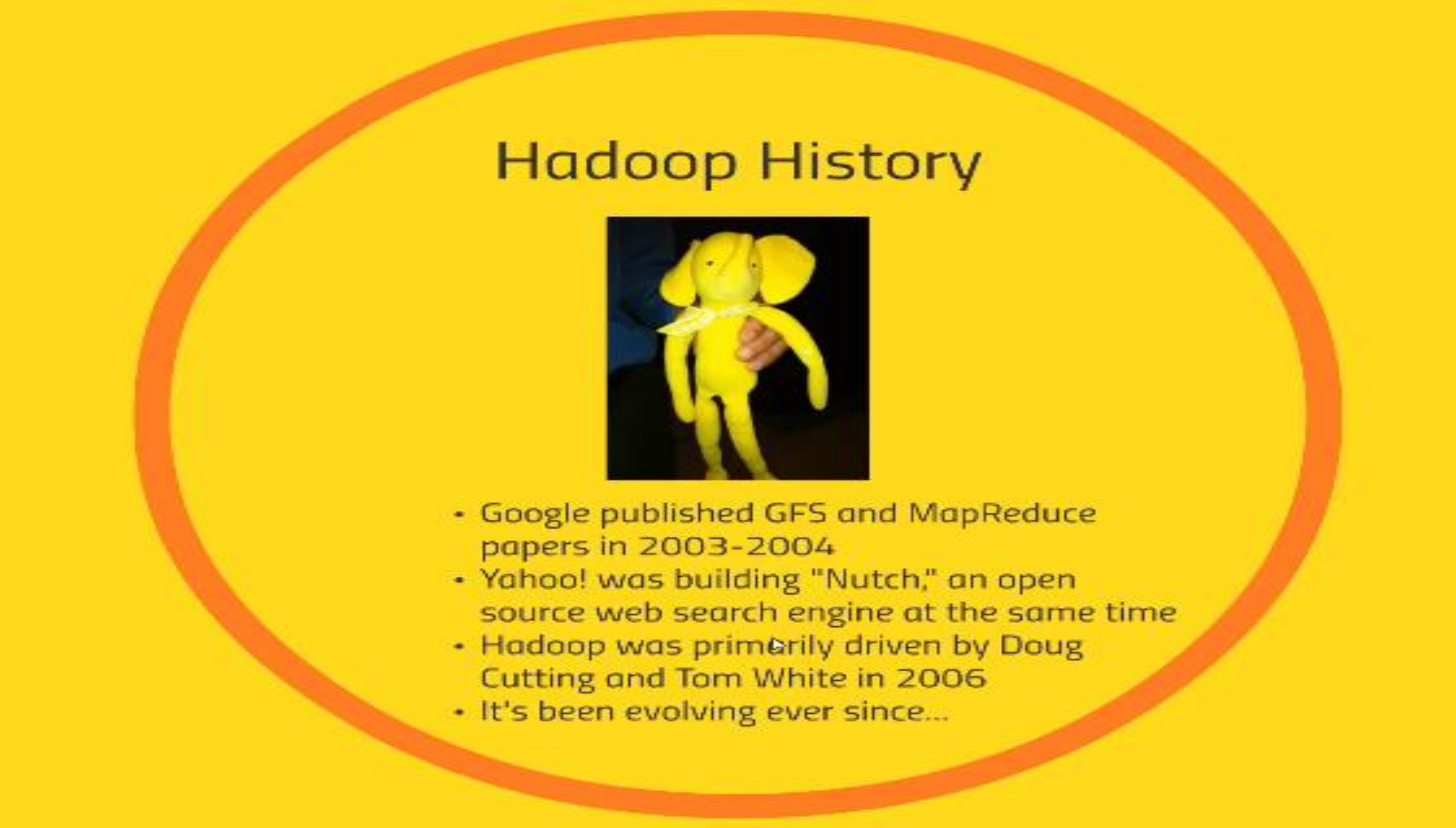
CS555 Big Data Computing

Lec-2



What is Hadoop?

"an open source **software platform** for **distributed storage** and **distributed processing** of **very large data sets** on **computer clusters** built from commodity hardware" - *Hortonworks*

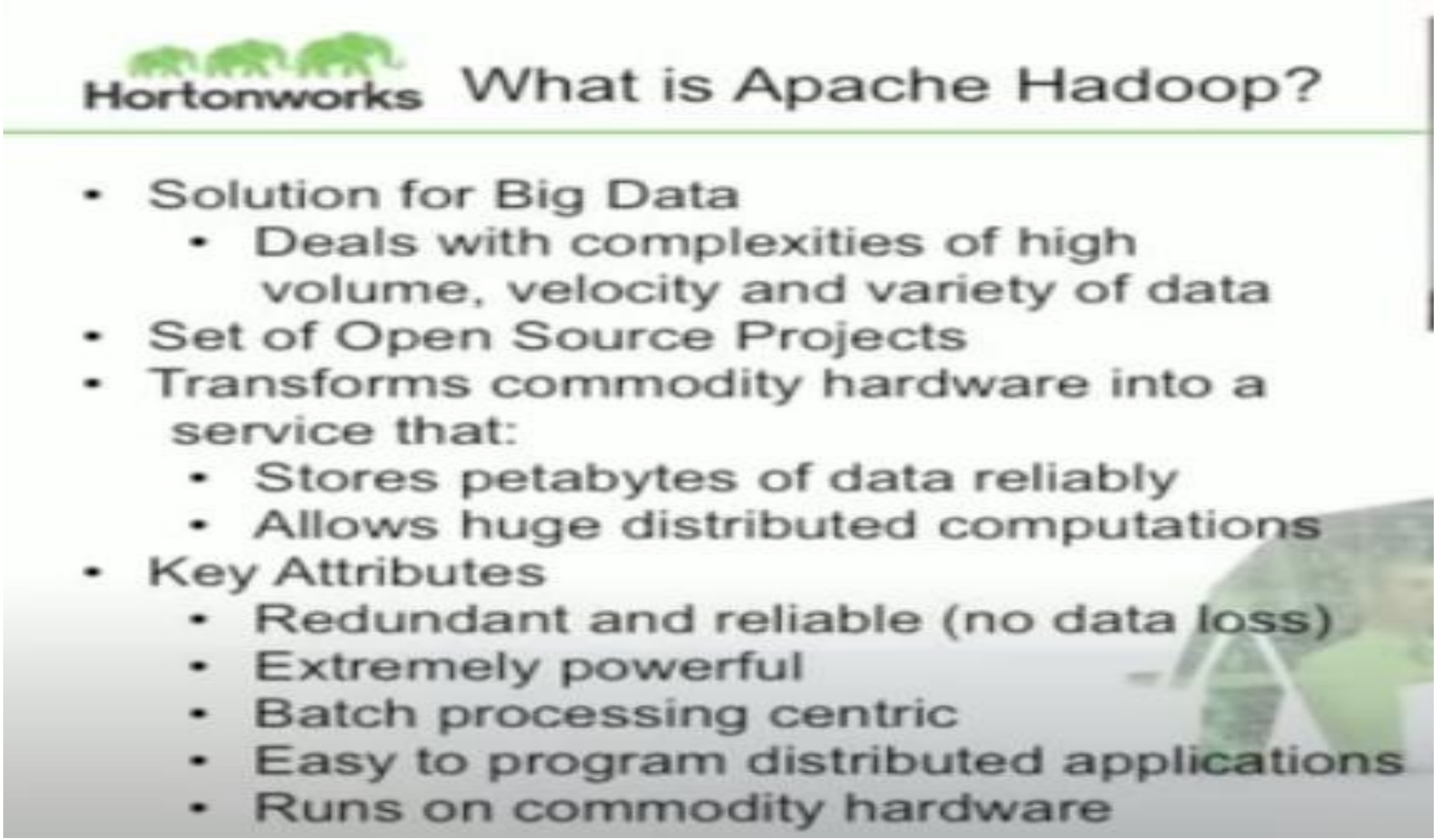


Hadoop History

- Google published GFS and MapReduce papers in 2003-2004
- Yahoo! was building "Nutch," an open source web search engine at the same time
- Hadoop was primarily driven by Doug Cutting and Tom White in 2006
- It's been evolving ever since...

Why Hadoop?

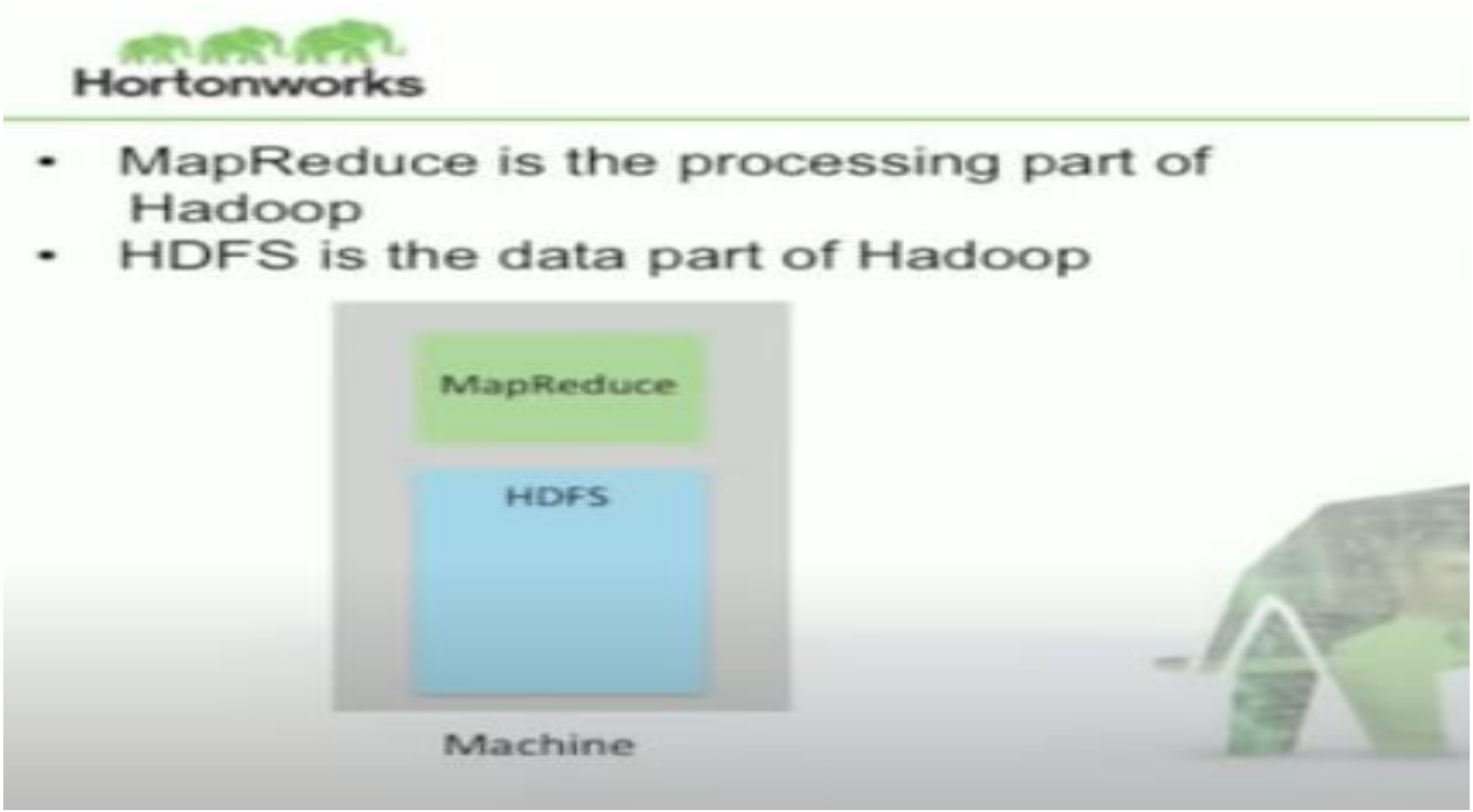
- Data's too darn big - terabytes per day
- Vertical scaling doesn't cut it
  - Disk seek times
  - Hardware failures
  - Processing times
- Horizontal scaling is linear
- Hadoop: It's not just for batch processing anymore



Hortonworks

What is Apache Hadoop?

- Solution for Big Data
  - Deals with complexities of high volume, velocity and variety of data
- Set of Open Source Projects
- Transforms commodity hardware into a service that:
  - Stores petabytes of data reliably
  - Allows huge distributed computations
- Key Attributes
  - Redundant and reliable (no data loss)
  - Extremely powerful
  - Batch processing centric
  - Easy to program distributed applications
  - Runs on commodity hardware



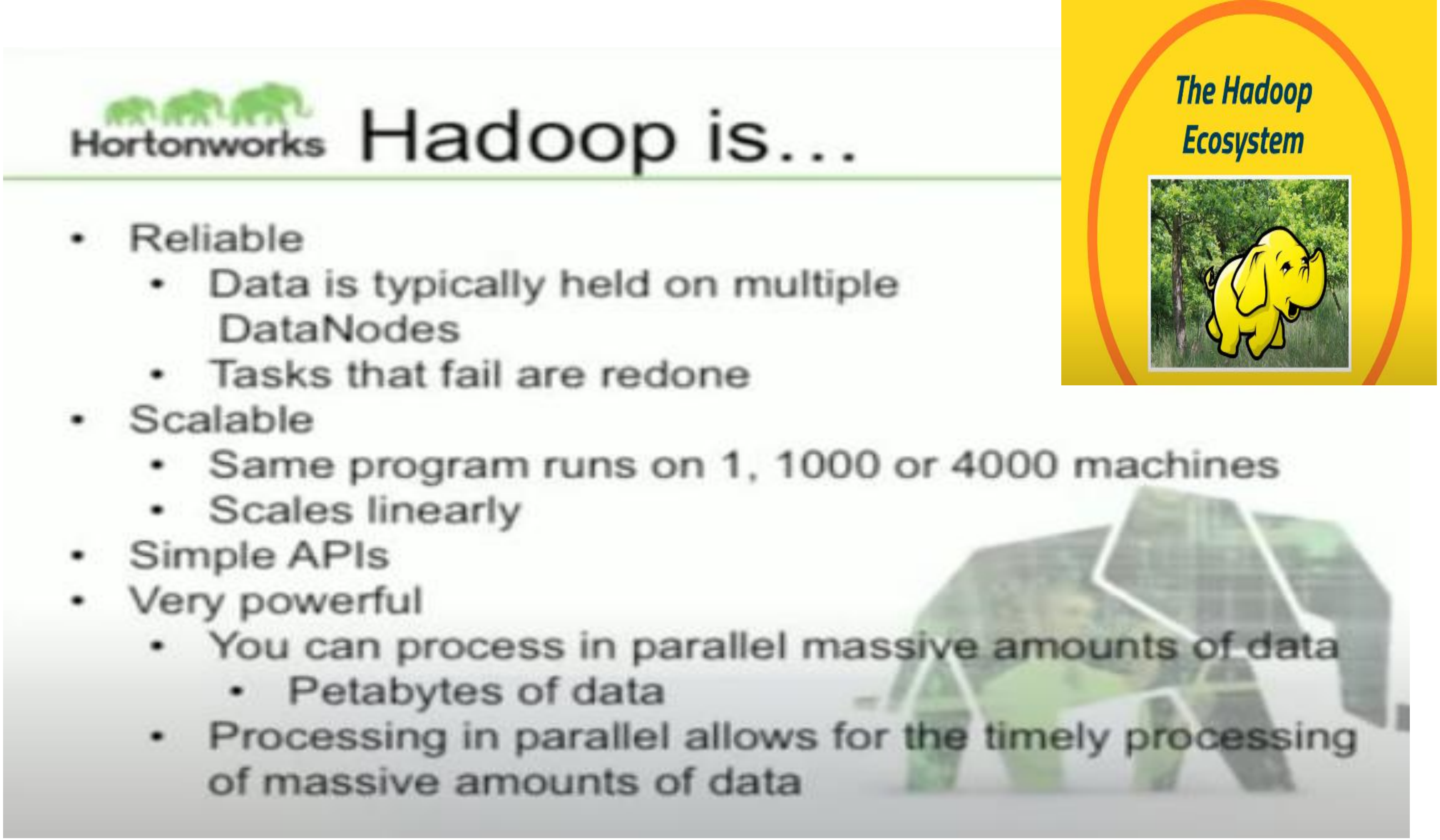
Hortonworks

- MapReduce is the processing part of Hadoop
- HDFS is the data part of Hadoop

MapReduce

HDFS

Machine

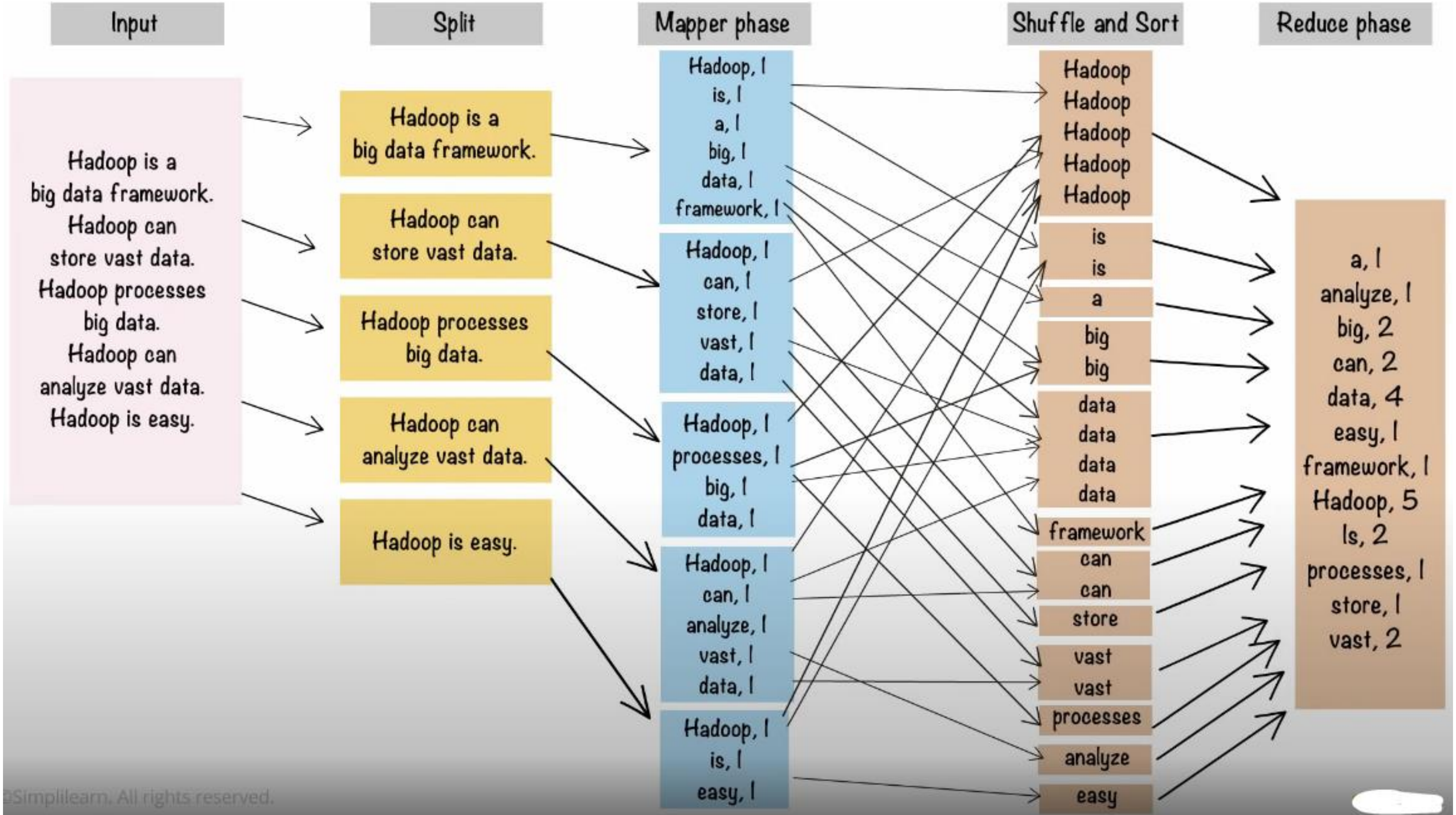


The Hadoop Ecosystem

Hortonworks

Hadoop is…

- Reliable
  - Data is typically held on multiple DataNodes
  - Tasks that fail are redone
- Scalable
  - Same program runs on 1, 1000 or 4000 machines
  - Scales linearly
- Simple APIs
- Very powerful
  - You can process in parallel massive amounts of data
    - Petabytes of data
  - Processing in parallel allows for the timely processing of massive amounts of data



Input

Hadoop is a
big data framework.
Hadoop can
store vast data.
Hadoop processes
big data.
Hadoop can
analyze vast data.
Hadoop is easy.

Split

Hadoop is a
big data framework.

Hadoop can
store vast data.

Hadoop processes
big data.

Hadoop can
analyze vast data.

Hadoop is easy.

Mapper phase

Hadoop, 1
is, 1
a, 1
big, 1
data, 1
framework, 1

Hadoop, 1
can, 1
store, 1
vast, 1
data, 1

Hadoop, 1
processes, 1
big, 1
data, 1

Hadoop, 1
can, 1
analyze, 1
vast, 1
data, 1

Hadoop, 1
is, 1
easy, 1

Shuffle and Sort

Hadoop
Hadoop
Hadoop
Hadoop
Hadoop

is
is

a

big
big

data
data
data
data

framework

can
can

store

vast
vast

processes

analyze

easy

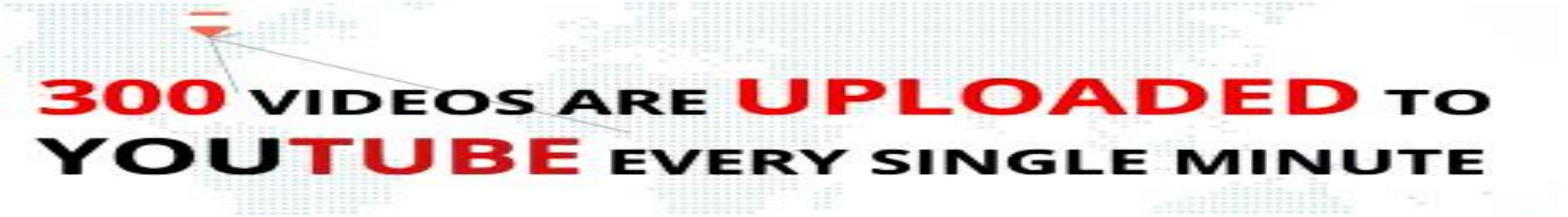
Reduce phase

a, 1
analyze, 1
big, 2
can, 2
data, 4
easy, 1
framework, 1
Hadoop, 5
Is, 2
processes, 1
store, 1
vast, 2

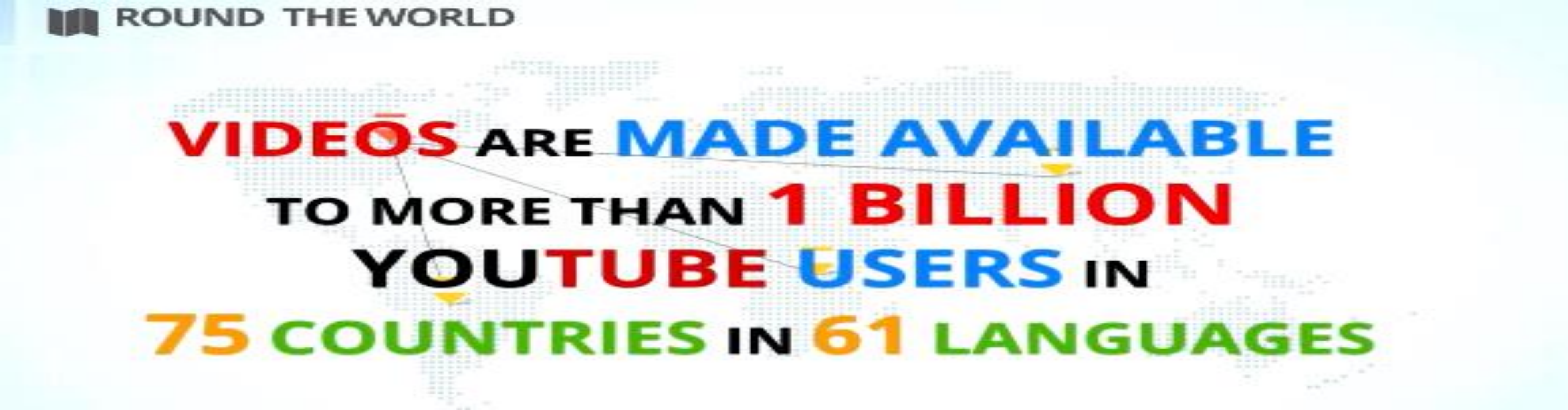
YOUTUBE DATA ANALYSIS

hadoop

Map Reduce Use Case



300 VIDEOS ARE UPLOADED TO YOUTUBE EVERY SINGLE MINUTE



ROUND THE WORLD

VIDEOS ARE MADE AVAILABLE TO MORE THAN 1 BILLION YOUTUBE USERS IN 75 COUNTRIES IN 61 LANGUAGES

Using Data Set Description

YouTube Data is Publicly Available

Powerful Tool for Video Marketers

Let's you Analyze your Competitor's videos too

Column 1: Video id of 11 characters.

Column 2: uploader of the video

Column 3: Interval between the day of establishment of Youtube and the date of uploading of the video.

Column 4: Category of the video.

Column 5: Length of the video.

Column 6: Number of views for the video.

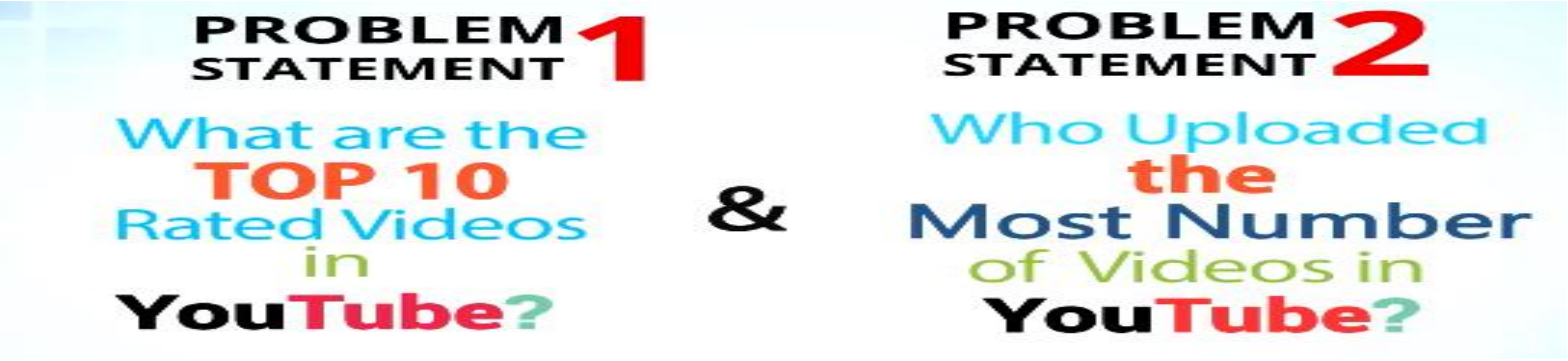
Column 7: Rating on the video.

Column 8: Number of ratings given for the video

Column 9: Number of comments done on the videos.

Column 10: Related video ids with the uploaded video.

How data from YouTube can be Analyzed using Hadoop?



PROBLEM STATEMENT 1

What are the TOP 10 Rated Videos in YouTube?

&

PROBLEM STATEMENT 2

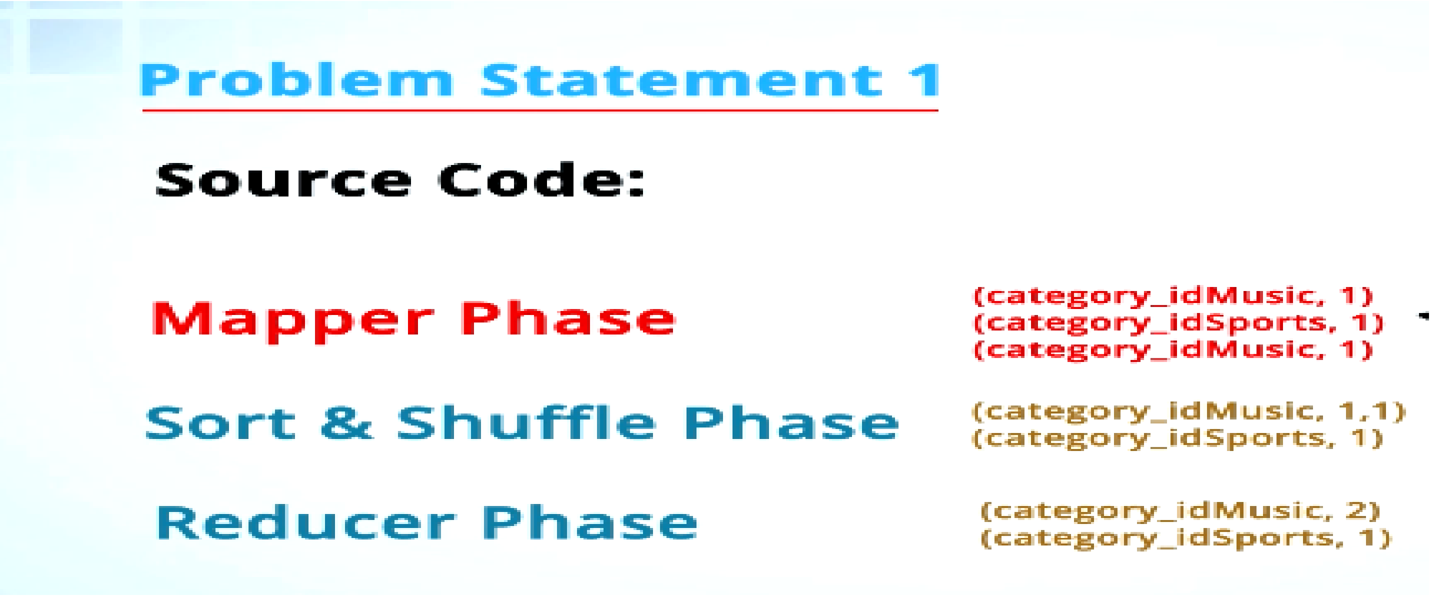
Who Uploaded the Most Number of Videos in YouTube?

PROBLEM STATEMENT 1

Here we will find out what are the top 5 categories with maximum number of videos uploaded.

SOURCE CODE

Now from the mapper, we want to get the *video category as key* and final int value *'1' as values* which will be passed to the *shuffle* and *sort* phase and are further sent to the reducer phase where the aggregation of the values is performed.



Problem Statement 1

Source Code:

Mapper Phase

(category_idMusic, 1)
(category_idSports, 1)
(category_idMusic, 1)

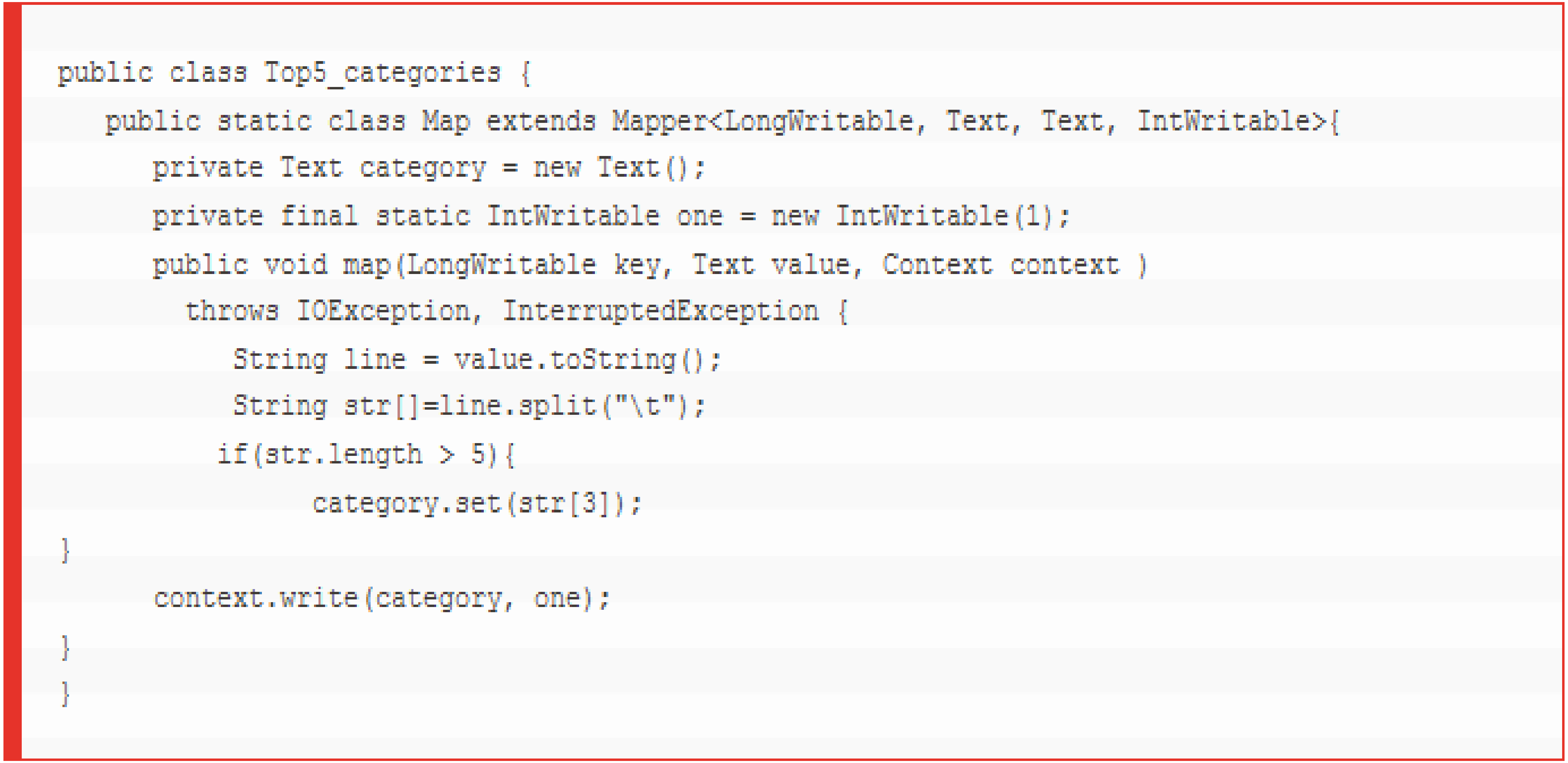
Sort & Shuffle Phase

(category_idMusic, 1,1)
(category_idSports, 1)

Reducer Phase

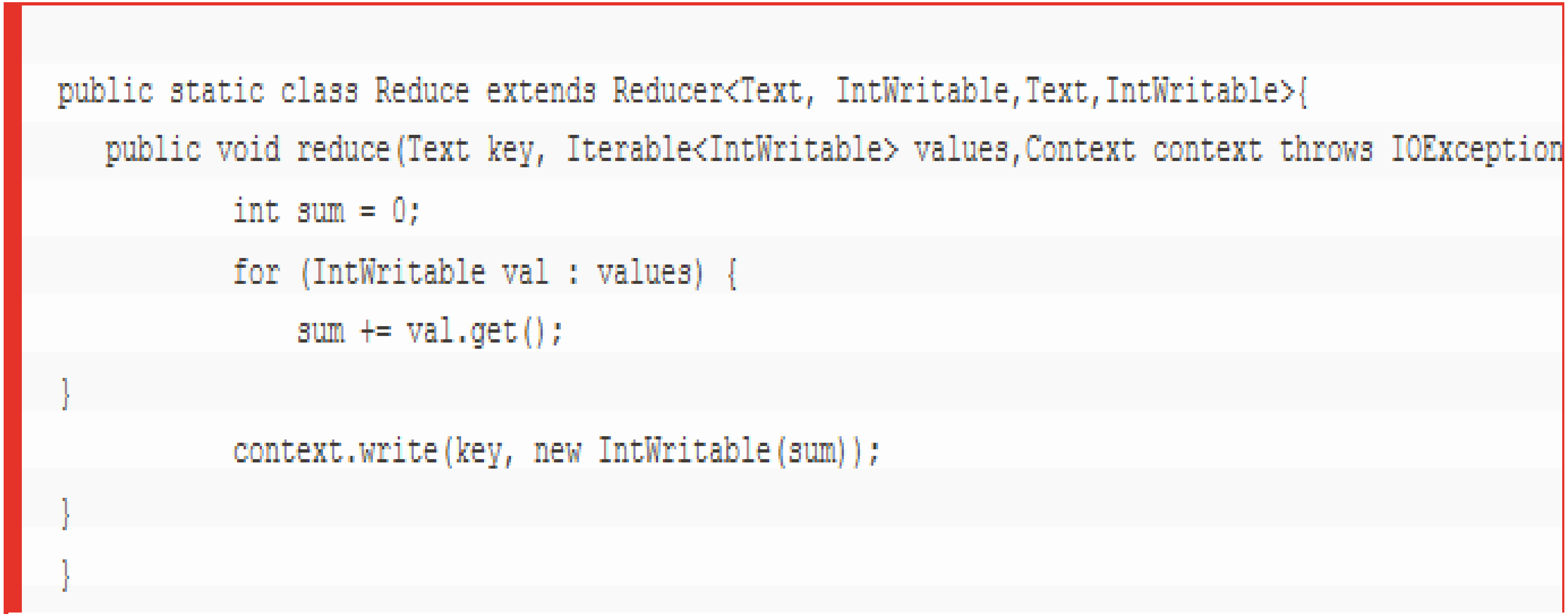
(category_idMusic, 2)
(category_idSports, 1)

MAPPER CODE

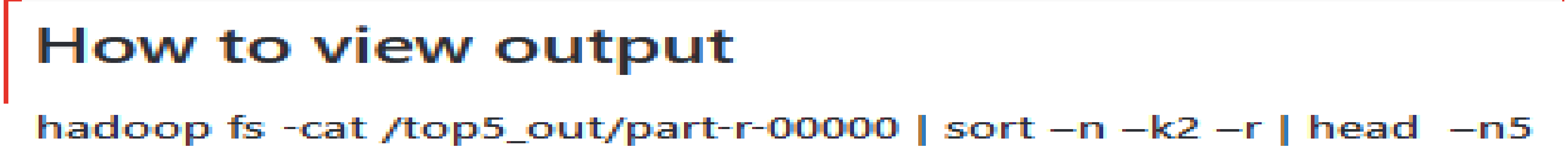


```
public class Top5_categories {
   public static class Map extends Mapper<LongWritable, Text, Text, IntWritable>{
      private Text category = new Text();
      private final static IntWritable one = new IntWritable(1);
      public void map(LongWritable key, Text value, Context context )
        throws IOException, InterruptedException {
           String line = value.toString();
           String str[]=line.split("\t");
          if(str.length > 5){
                category.set(str[3]);
}
      context.write(category, one);
}
}
```

REDUCER CODE



```
public static class Reduce extends Reducer<Text, IntWritable,Text,IntWritable>{
   public void reduce(Text key, Iterable<IntWritable> values,Context context throws IOException
           int sum = 0;
           for (IntWritable val : values) {
               sum += val.get();
}
           context.write(key, new IntWritable(sum));
}
}
```



How to view output

```
hadoop fs -cat /top5_out/part-r-00000 | sort –n –k2 –r | head  –n5
```

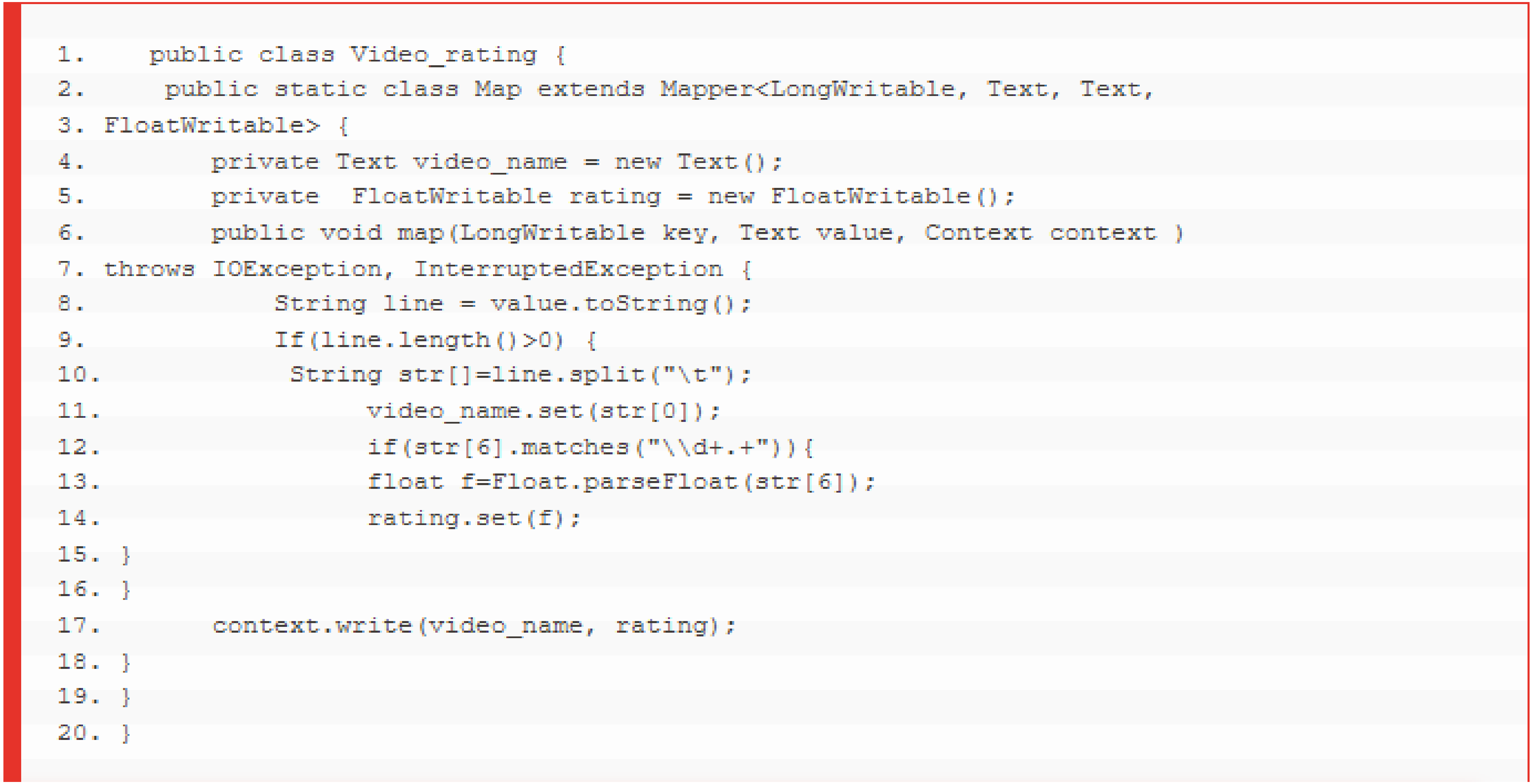
PROBLEM STATEMENT 2

In this problem statement, we will find the top 10 rated videos on youtube.

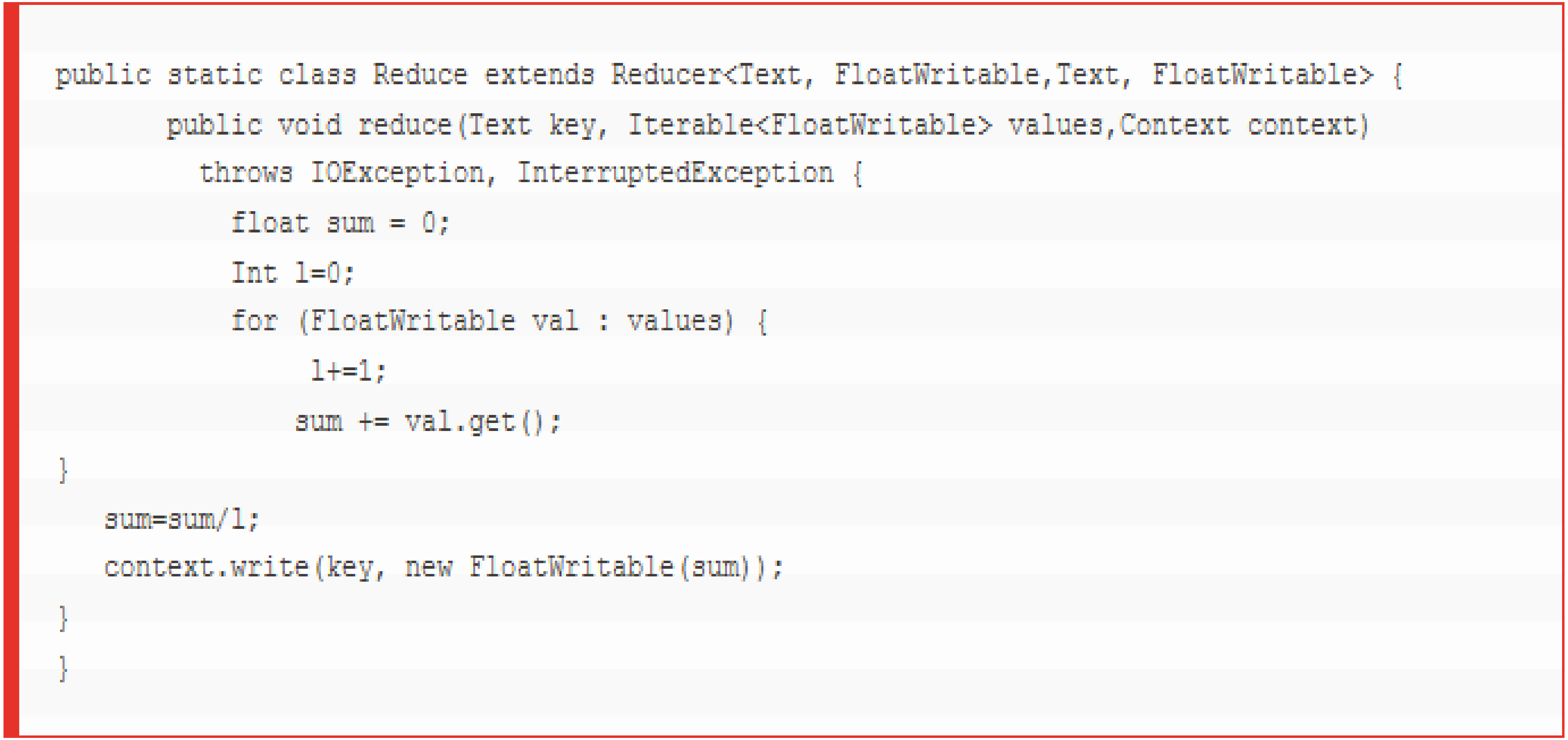
SOURCE CODE

Now from the mapper, we want to get the *video id* as *key* and *rating* as *a value* which will be passed to the *shuffle* and *sort* phase and is further sent to the reducer phase where the aggregation of the values is performed.

MAPPER CODE



```
   public class Video_rating {
    public static class Map extends Mapper<LongWritable, Text, Text,
FloatWritable> {
       private Text video_name = new Text();
       private  FloatWritable rating = new FloatWritable();
       public void map(LongWritable key, Text value, Context context )
throws IOException, InterruptedException {
           String line = value.toString();
           If(line.length()>0) {
            String str[]=line.split("\t");
                 video_name.set(str[0]);
                 if(str[6].matches("\\d+.+")){
                 float f=Float.parseFloat(str[6]);
                 rating.set(f);
 }
 }
       context.write(video_name, rating);
 }
 }
 }
```



```
public static class Reduce extends Reducer<Text, FloatWritable,Text, FloatWritable> {
       public void reduce(Text key, Iterable<FloatWritable> values,Context context)
         throws IOException, InterruptedException {
           float sum = 0;
           Int l=0;
           for (FloatWritable val : values) {
                l+=1;
               sum += val.get();
}
   sum=sum/l;
   context.write(key, new FloatWritable(sum));
}
}
```

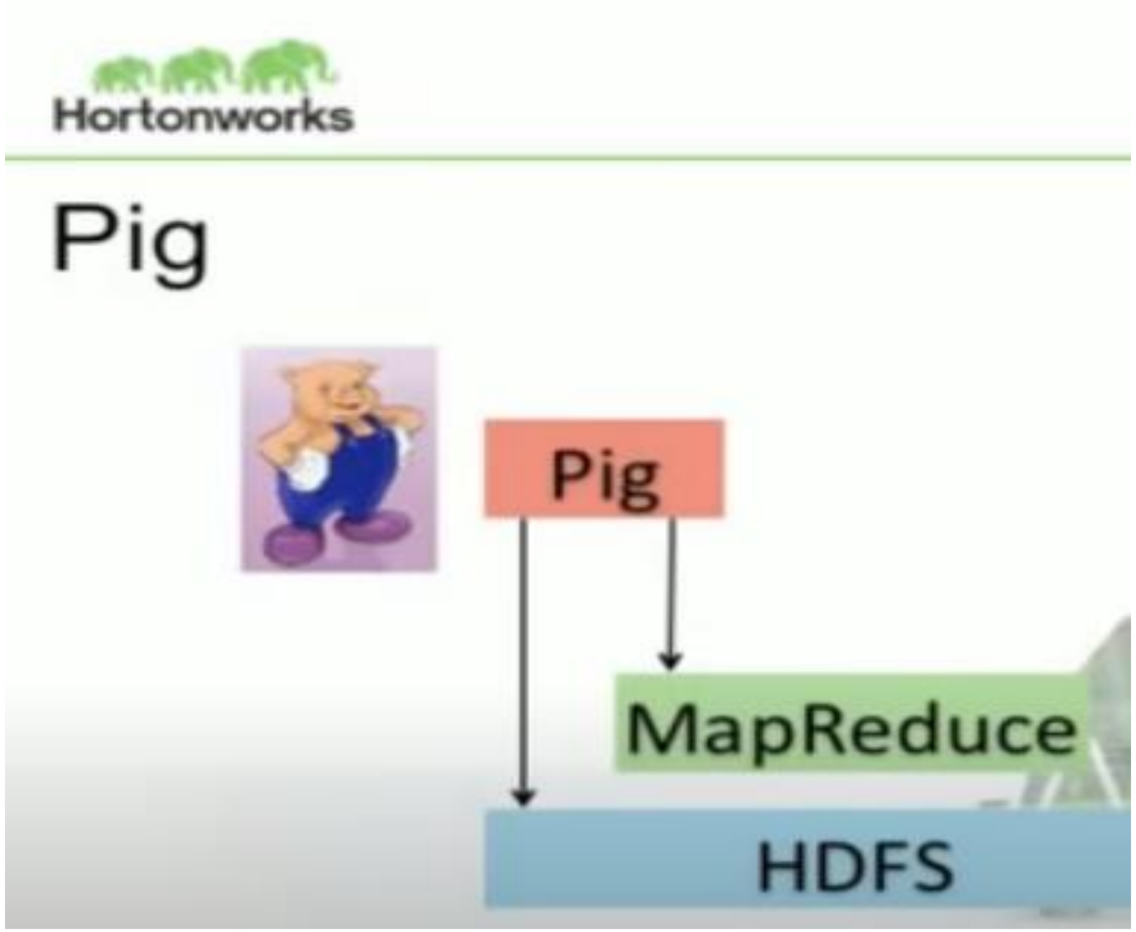
REDUCER CODE

How to view output

hadoop fs -cat /videorating_out/part-r-00000 | sort –n –k2 –r | head –n10

Thank you

Questions?



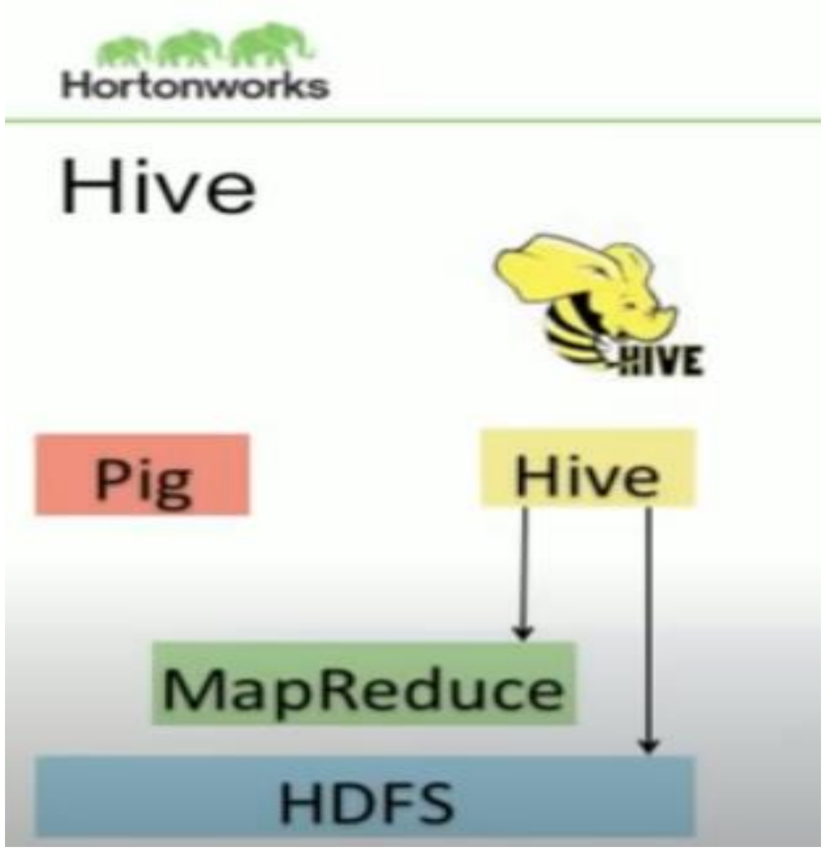
Hortonworks

Pig

Pig

MapReduce

HDFS



Hortonworks

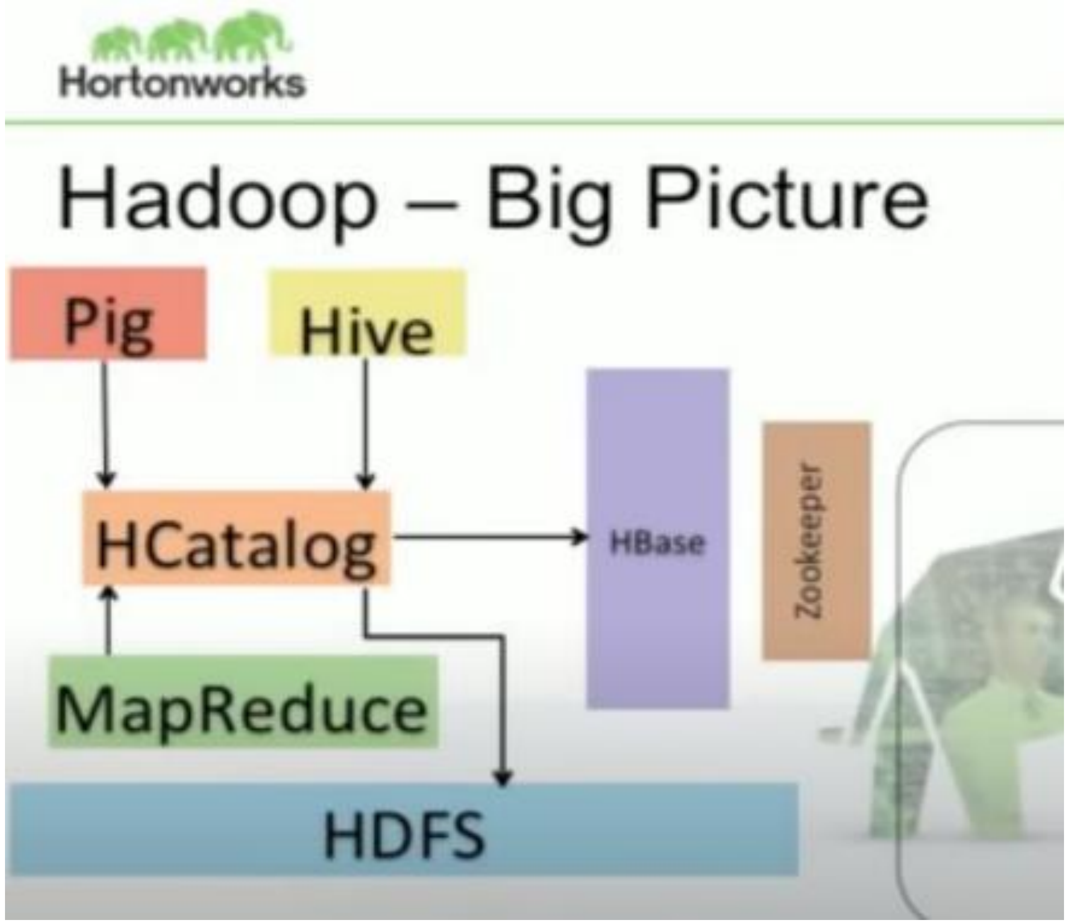
Hive

Pig

Hive

MapReduce

HDFS



Hortonworks

Hadoop – Big Picture

Pig

Hive

HCatalog

MapReduce

HBase

Zookeeper

HDFS

- Thanks