In this age,Big Data

Is coming……

贵州科学院
Guizhou Academy of Sciences

- In pioneer days they used oxen for heavy pulling, and when one ox couldn’t budge a log, they didn’t try to grow a larger ox.
- We shouldn’t be trying for bigger computers, but for more systems of computers.

—Grace Hopper

Application and Platform Building of Big Data

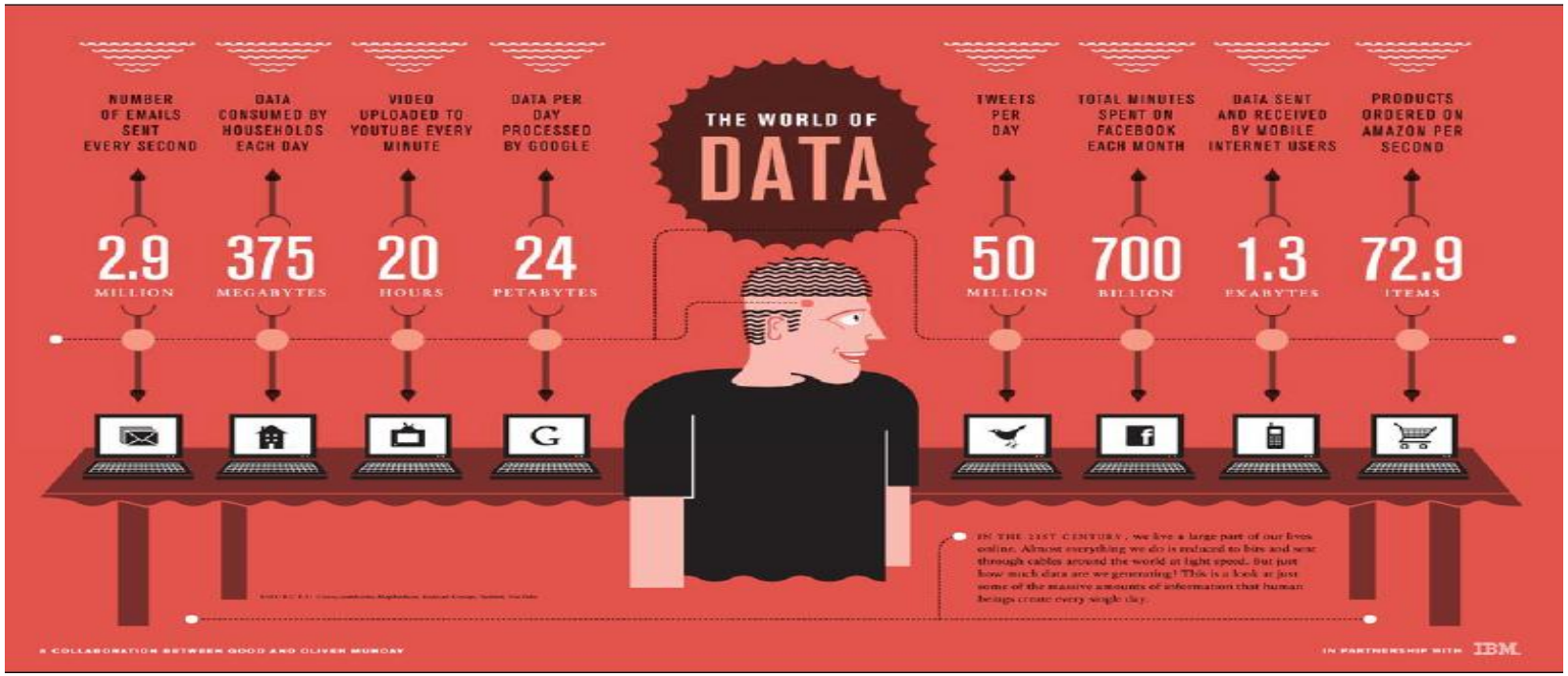
Xinqiang Ma
Assoc. Prof, GAS
E-mail: xinqma@163.com
Study Groups: Dr. Li Danning, Dr.Li Dan, Mr. Chen yuqing

贵州科学院
Guizhou Academy of Sciences

Topics (Outline)

- 1. What is Big Data?
- 2. Application of Big Data
- 3. Big Data opportunities and challenges
- 4. How to Deal with Big Data?
- 5. What's Hadoop/MapReduce?
- 6. Big data players/Software Tools/Platforms
- 7. Examples

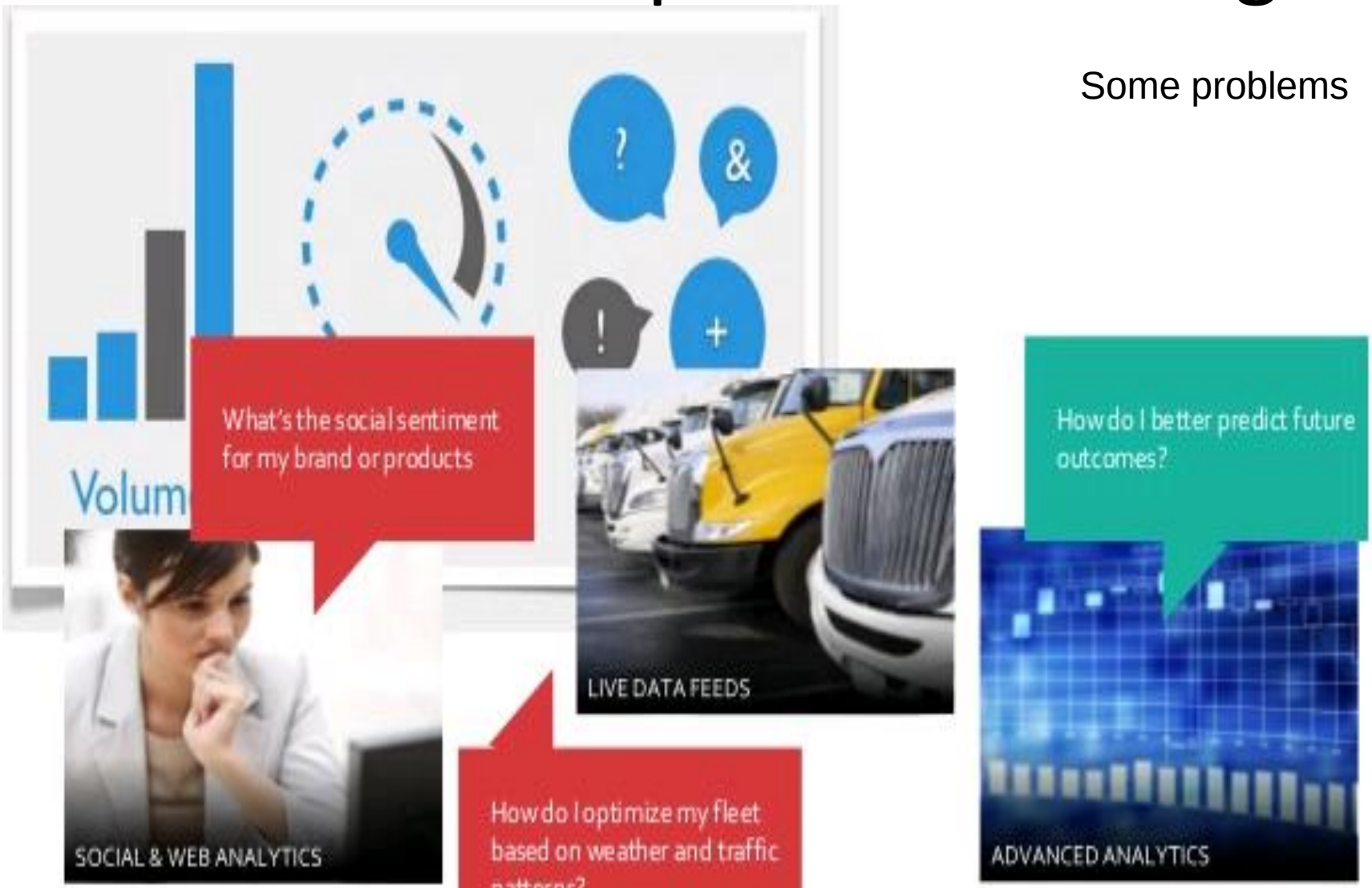
Data!

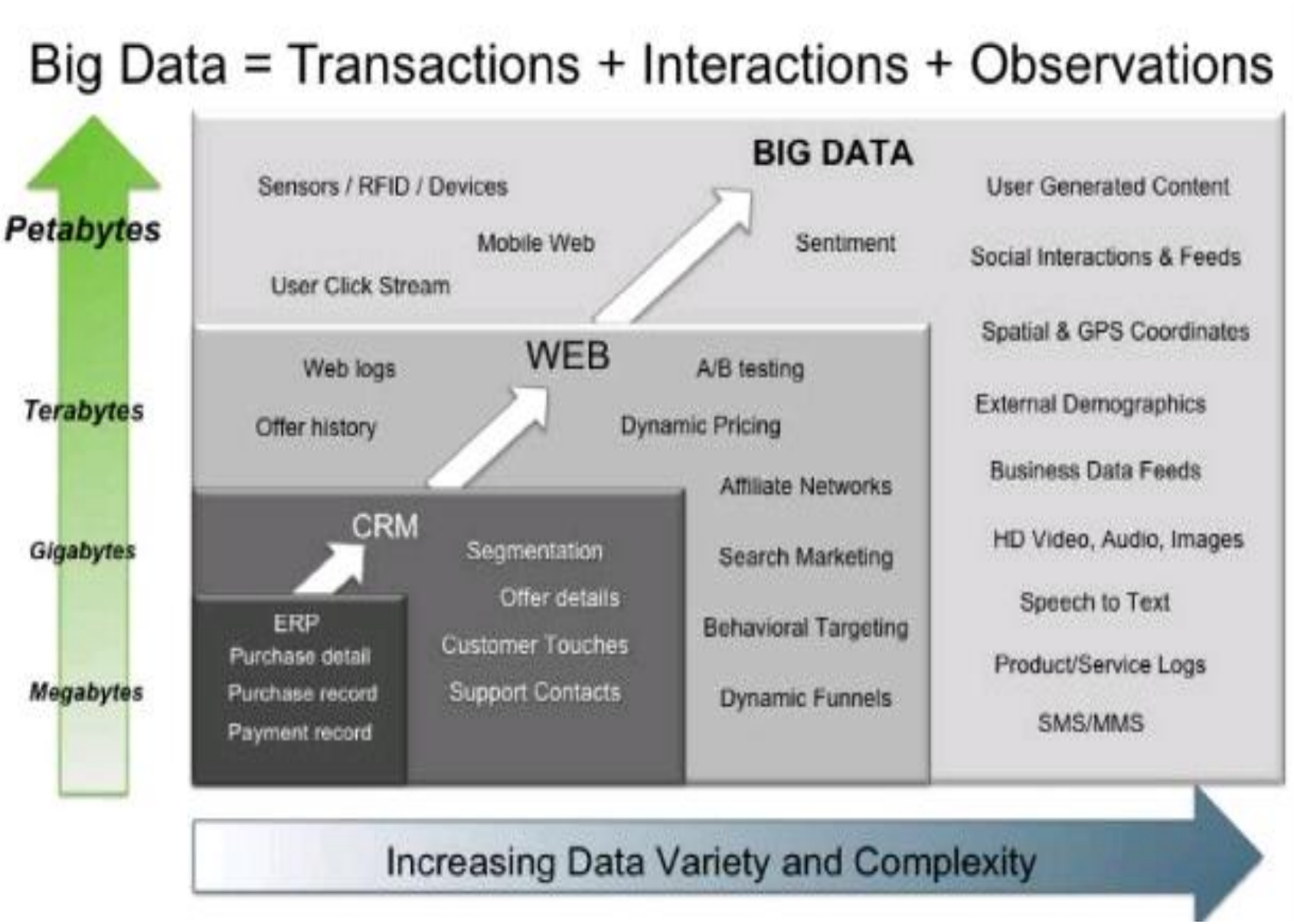


- We live in the data age. It's not easy to measure the total volume of data stored electronically, but an IDC estimate put the size of the "digital universe" at 0.18 zettabytes in 2006 and is forecasting a tenfold growth by 2011 to 1.8 zettabytes.
- The trend is for every individual's data footprint to grow, but perhaps more important, the amount of data generated by machines will be even greater than that generated by people. Machine logs, RFID readers, sensor networks, vehicle GPS traces, retail transactions—all of these contribute to the growing mountain of data.
- The good news is that Big Data is here. The bad news is that we are struggling to store and analyze it.

How to data capture and storage?

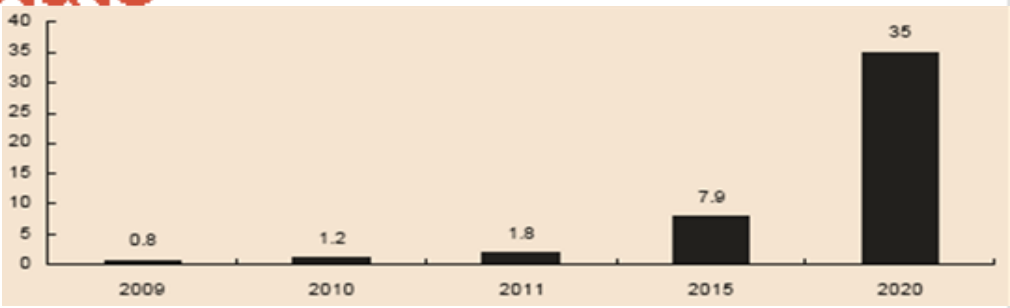
Some problems



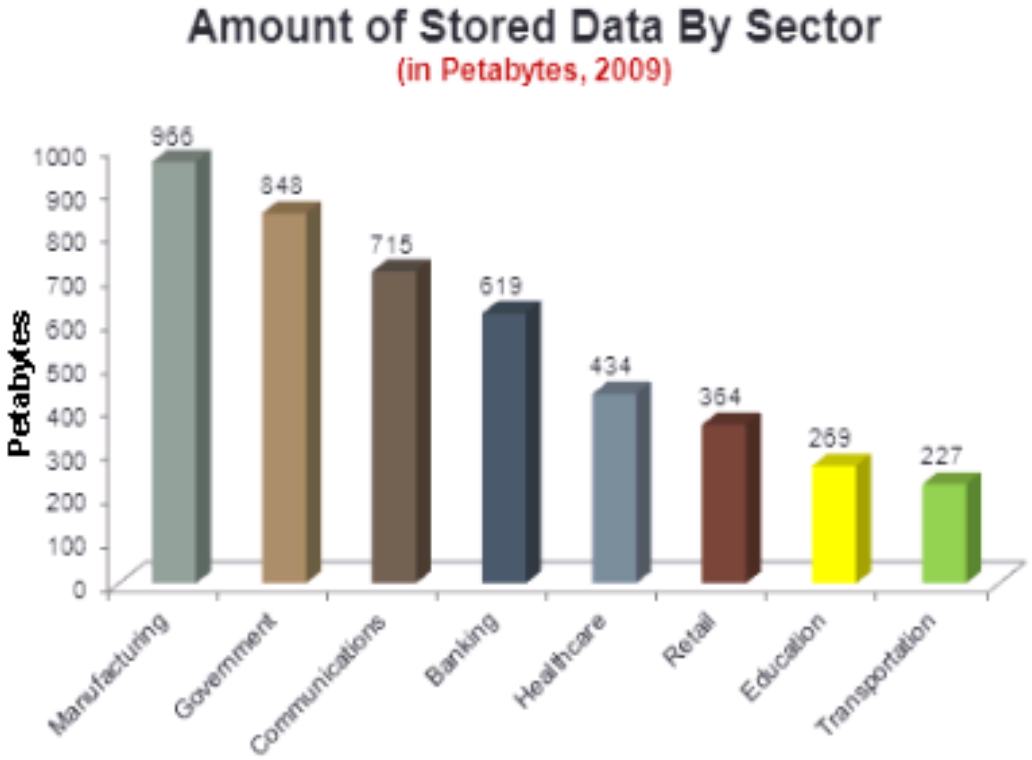


Big data is a broad term for data sets so large or complex that traditional data processing applications are inadequate.

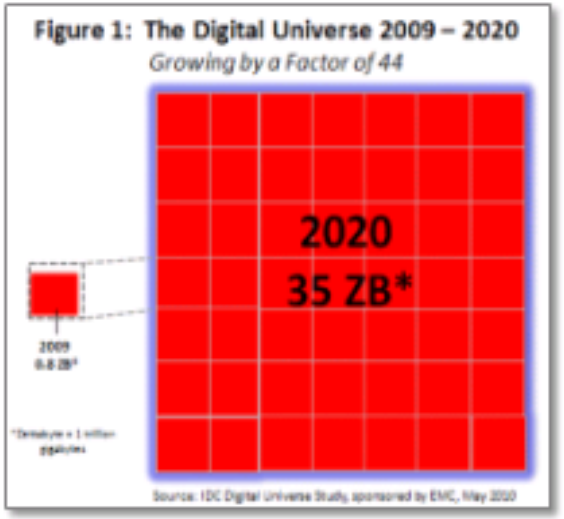
Some Big Data Stats



Rapid growth in data rates



Sources:
"Big Data: The Next Frontier for Innovation, Competition and Productivity."
US Bureau of Labor Statistics | McKinsey Global Institute Analysis



1 zettabyte?
= 1 million petabytes
= 1 trillion terabytes
= 1 quadrillion gigabytes

TB ⇨ PB ⇨ EB ⇨ ZB

What is Big Data?

- **Capturing and managing lots of information**
- **Working with many new types of data**

  Structure/Unstructured
- **Exploiting these masses of information and new data types with new styles of applications**
- **Bigger than Terabytes**

  volume, variety, velocity, variability, value

Big data is a broad term for data sets so large or complex that traditional data processing applications are inadequate.

What is Big Data?

Big data is a broad term for data sets so large or complex that traditiona l data processing applications are inadequate.

1. Volume

Refers to the huge amount of data that are being produced daily and that have gone from MB and GB to PB.

2. Variety

Deals with the many number of types of data, both structured an d unstructured.

3. value

The quality of the data being captured can vary greatly. Accur acy of analysis depends on the veracity of the source data.

4. Velocity

Refers to how fast the data is coming in and how fast it needs to be analysed.

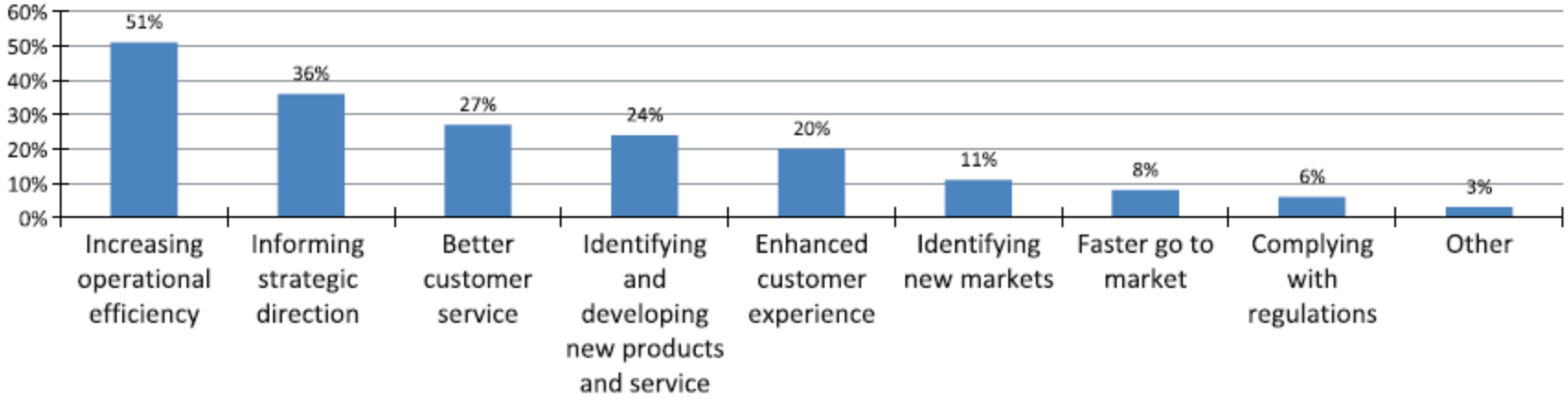
Application of Big Data

What This Means for You

Big Data can help a company do many things:

- Profile customers
- Determine pricing strategies
- Identify competitive advantages
- Better target advertising
- Inform internal research and product development
- Strengthen customer service

Big Data opportunities and challenges

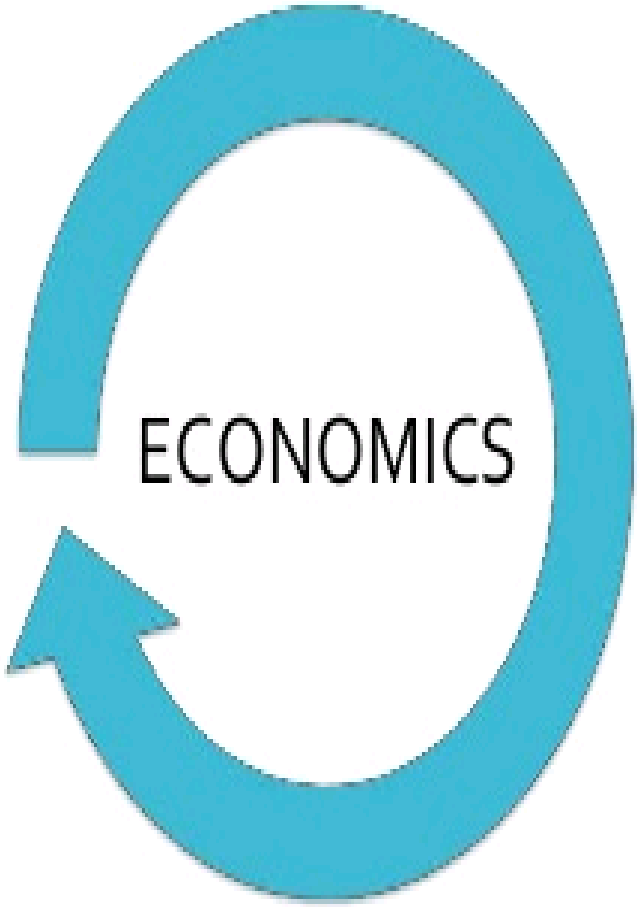


Big Data Opportunities: above 50% of 560 enterprises think Big Data will help them in increasing operational efficiency, etc.

- There are many advantages in business section that can be obtained through harnessing Big Data as illustrated in Fig, including increasing operational efficiency, informing strategic direction, developing better customer service, identifying and developing new products and services, identifying new customers and markets, etc.
- By liberal estimates , Big Data could produce $300 billion potential annual value to US health care, and €250 billion to European public administration. There will be $600 billion potential annual consumer surplus from using personal location data globally, and give a potential increase with 60%. Only in United States, Big Data produce 140,000 to 190,000 deep analytical talent positions and 1.5 million data-savvy managers. Undoubtedly, Big Data is usually juicy and lucrative if explored correctly.

Application of Big Data

BIG DATA
Economics

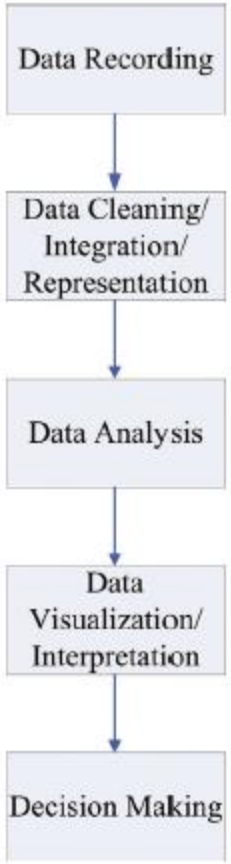


- Commodity hardware compatibility
- Reduction in storage cost
- Open source system
- The Web economy

Big Data Solutions

- Cloudera: Cloudera Enterprise
- Microsoft:Windows Azure HDInsight Service
- Google:BigQuery
- Amazon: DynamoDB
- IBM: InfoSphere Streams/Netezza
- EMC: Greenplum
- TeraData: Aster MapReduce Platform
- Oracle: Hadoop/Mapreduce Big Data connectors

Big Data opportunities and challenges



Knowledge discovery process.

- Opportunities are always followed by challenges. On the one hand, Big Data bring many attractive opportunities. On the other hand, we are also facing a lot of challenges when handle Big Data problems, difficulties lie in data capture, storage, searching, sharing, analysis, and visualization.

Challenges

- **Information growth**
- **Processing power**
- **Physical storage**

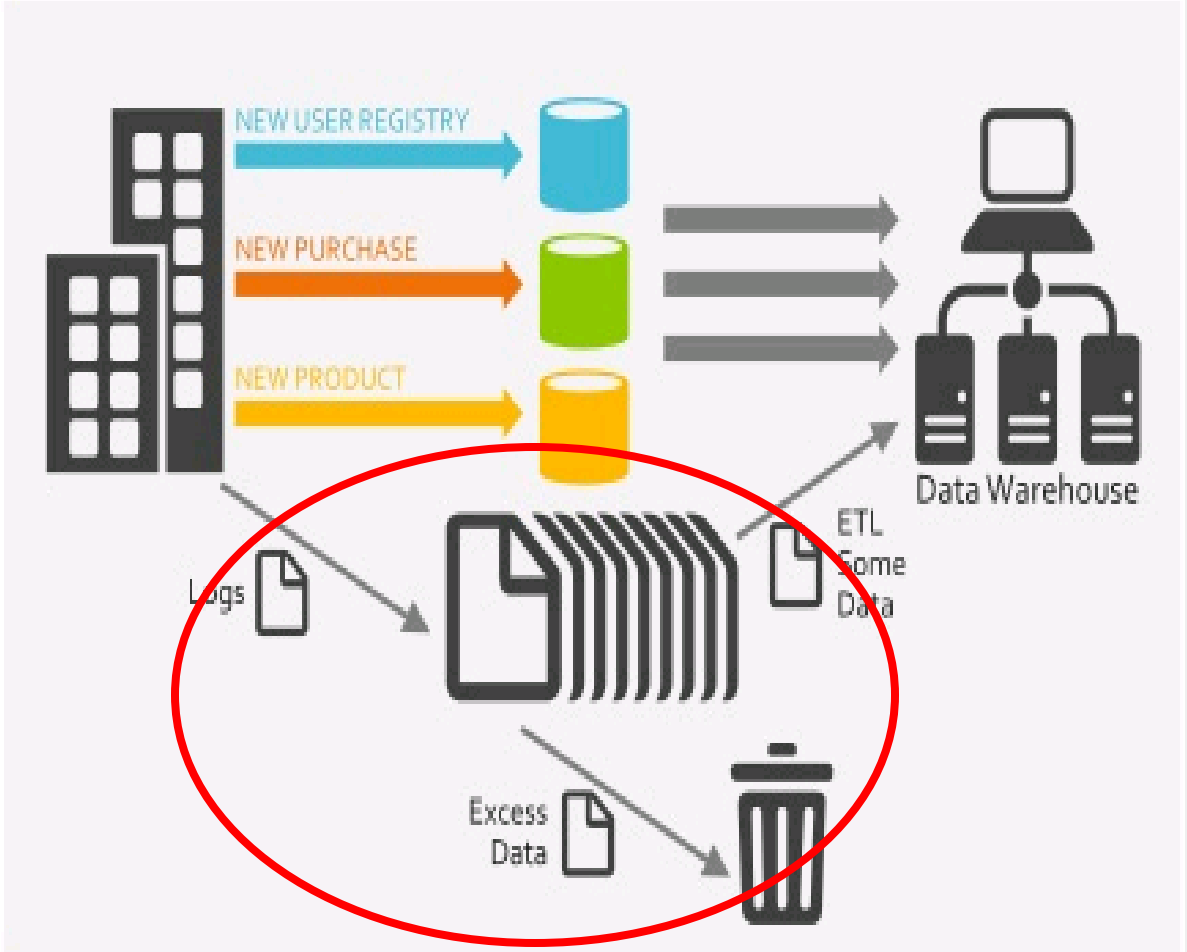
  disk capacity increase dramatically

  100 MB/S read from disk (bottle neck)

  data seeking time is slow than data transferring
- **Data issues**
- **Costs**

OPERATIONAL DATA

With current implementation



Serial Parallel Extract-Transform-Load

OPERATIONAL DATA

With big data implementation

Not serial but parallel example: Hadoop cluster

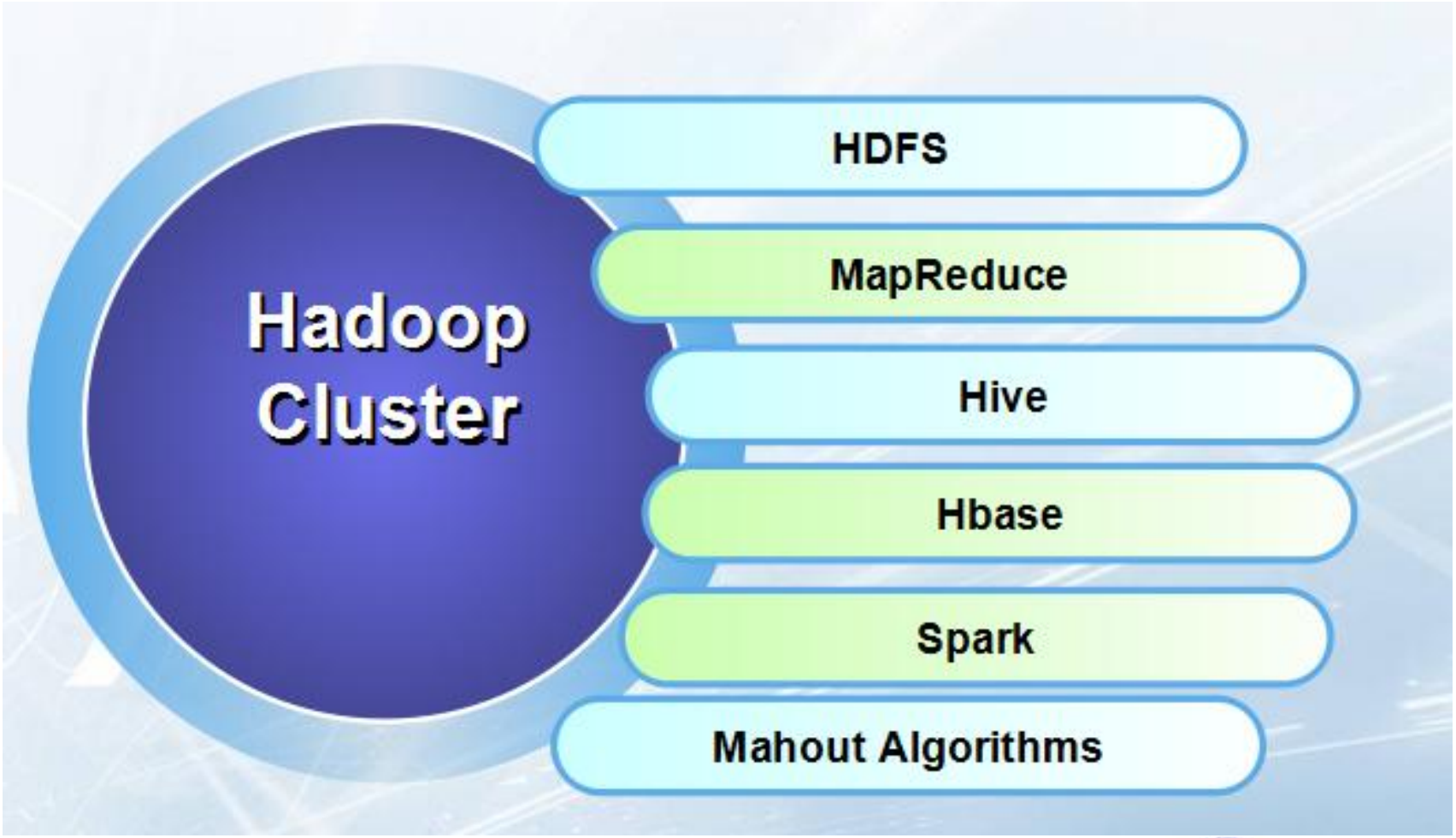
Main steps in adopting an analytical system

- 1. What Will We Analyze?
- 2. Do We Buy or Build?
- 3. Are We Ready to Invest?
- 4. Do We Understand the Impact?

How to Deal with Big Data?

- MapReduce
- Rational Database Management System(RDBMS)
- RDBMS vs MapReduce
- Hadoop
- HDFS

How to Deal with Big Data?



What is Hadoop

- Hadoop is a distributed computing framework with two main components:
  - a distributed file system and
  - a map-reduce implementation.

- Imagine you have a cluster of 100 computers. Hadoop's distributed file system makes it so you can put data "into Hadoop" and pretend that all the hard drives on your machines have coalesced into one gigantic drive.

- Under the hood, it breaks each file you give it into 64- or 128-MB chunks called blocks and sends them to different machines in the cluster, replicating each block three times along the way.

MapReduce

- MapReduce is a programming model for data processing. The model is simple, yet not too simple to express useful programs in.
- Most important, MapReduce programs are inherently parallel, thus putting very large-scale data analysis into the hands of anyone with enough machines at her disposal.
- Highly fault tolerant

  nodes are expected to fail • Every data block (by default) replicated on 3 nodes (is also rack aware)

How does MapReduce work

MapReduce works by breaking the processing into two phases: the map phase and the reduce phase. Each phase has key-value pairs as input and output, the types of which may be chosen by the programmer.

The programmer also specifies two functions: the map function and the reduce function.

Format and Types

- MapReduce model in detail, and, in particular, how data in various formats, from simple text to structured binary objects, can be used with this model
- MapReduce uses *key/value pairs*.

(Traditionally using rows and columns)

map: (K1, V1) → list(K2, V2)

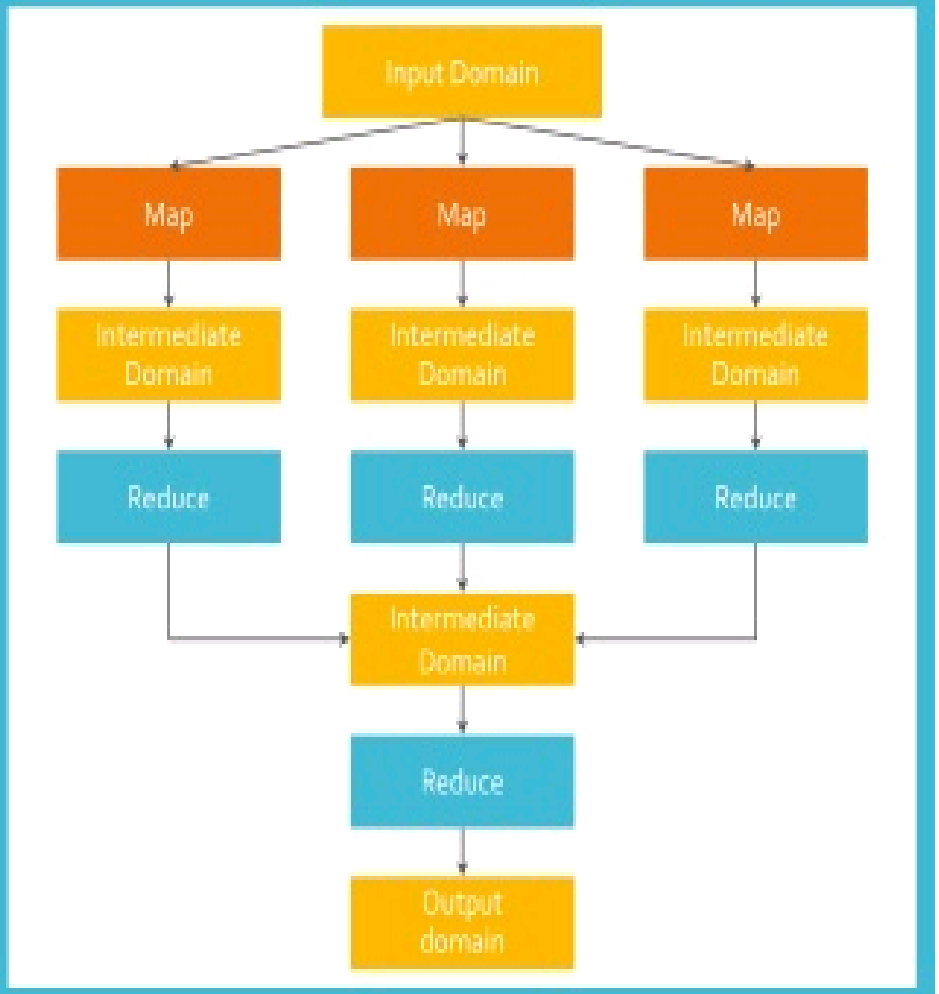
reduce: (K2, list(V2)) → list(K3, V3)

MapReduce – Workflow

A MapReduce *job* usually splits the input data-set into independent chunks which are processed by the *map tasks* in a completely parallel manner

The framework sorts the outputs of the maps, which are then input to the *reduce* tasks

The framework takes care of scheduling tasks, monitoring them and re-executes the failed tasks



Map

- MapReduce uses *key/value pairs*.
  (Traditionally using rows and columns)

  Example: last name/chen
  withdraw amount/20
  transaction date/06-23-2013

Reduce

- all the intermediate values for a given output key are combined together into a list.
- The reduce() function then combines the intermediate values into one or more final values for the same key.

Example: A Weather Dataset

- For our example, we will write a program that mines weather data. Weather sensors collect data every hour at many locations across the globe and gather a large volume of log data, which is a good candidate for analysis with MapReduce because it is semistructured and record-oriented.
- The data we will use is from the National Climatic Data Center (NCDC, *http://www.ncdc.noaa.gov/*).

Example: A Weather Dataset

- Example: it is a small script to calculate the maximum temperature (Fahrenheit) for each year.
- Our map function is simple. We pull out the year and the air temperature because these are the only fields we are interested in. In this case,
- the map function is just a data preparation phase, setting up the data in such a way that the reducer function can do its work on it: finding the maximum temperature for each year.
- The reduce function is also a good place to drop bad records: here we filter out temperatures that are missing, suspect, or erroneous.

Example: A Weather Dataset

- To visualize the way the map works, consider the following sample lines of input data (some unused columns have been dropped to fit the page, indicated by ellipses):

```
0067011990999991950051507004...9999999N9+00001+99999999999...
0043011990999991950051512004...9999999N9+00221+99999999999...
0043011990999991950051518004...9999999N9-00111+99999999999...
0043012650999991949032412004...0500001N9+01111+99999999999...
0043012650999991949032418004...0500001N9+00781+99999999999...
```

These lines are presented to the map function as the key-value pairs:

```
(0, 0067011990999991950051507004...9999999N9+00001+99999999999...)
(106, 0043011990999991950051512004...9999999N9+00221+99999999999...)
(212, 0043011990999991950051518004...9999999N9-00111+99999999999...)
(318, 0043012650999991949032412004...0500001N9+01111+99999999999...)
(424, 0043012650999991949032418004...0500001N9+00781+99999999999...)
```

Example: A Weather Dataset

The keys are the line offsets within the file, which we ignore in our map function. The map function merely extracts the year and the air temperature (indicated in bold text), and emits them as its output (the temperature values have been interpreted as integers):

```
(1950, 0)
(1950, 22)
(1950, -11)
(1949, 111)
(1949, 78)
```

The output from the map function is processed by the MapReduce framework before being sent to the reduce function. This processing sorts and groups the key-value pairs by key. So, continuing the example, our reduce function sees the following input:

```
(1949, [111, 78])
(1950, [0, 22, -11])
```

Each year appears with a list of all its air temperature readings. All the reduce function has to do now is iterate through the list and pick up the maximum reading:

```
(1949, 111)
(1950, 22)
```

This is the final output: the maximum global temperature recorded in each year.

Fahrenheit

MapReduce logical data flow

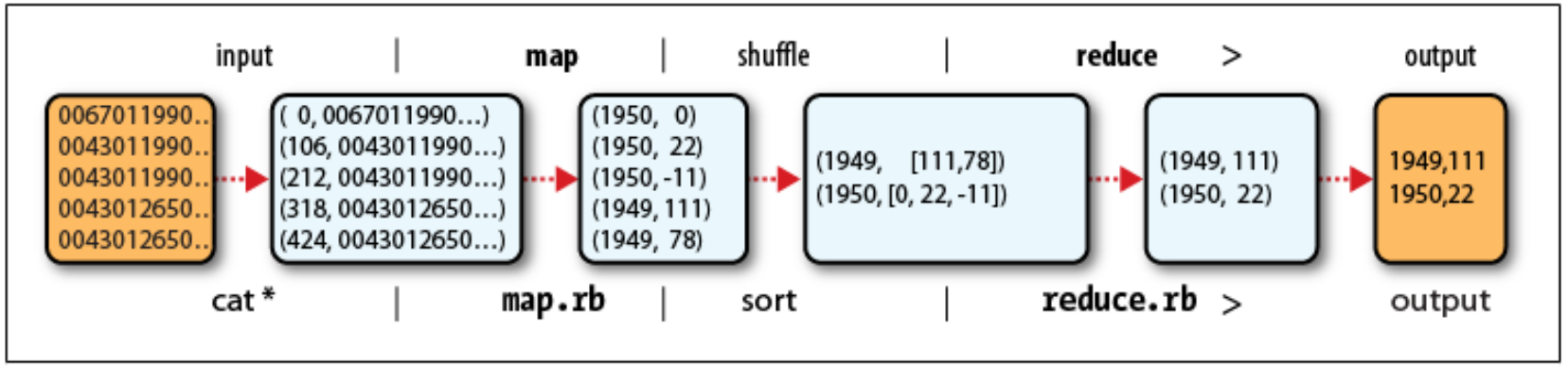


Figure. MapReduce logical data flow

The whole data flow is illustrated in Figure. At the bottom of the diagram is a Unix pipeline, which mimics the whole MapReduce flow and which we will see again later in this chapter when we look at Hadoop Streaming.

Mapreduce Special Feature

- Counter
- Sorting
- Joins
- Shuffle

MapReduce guarantees that the input to every reducer is sorted by key. The process by which the system performs the sort—and transfers the map outputs to the reducers as inputs – Shuffle

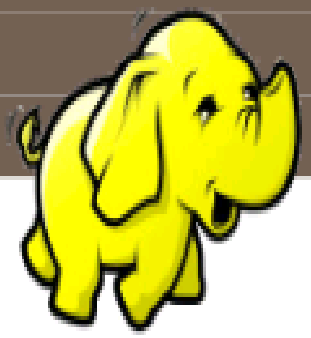
RDBMS

- fixed-schema, row-oriented databases with ACID properties and a sophisticated SQL query engine.
- The emphasis is on strong consistency, referential integrity, abstraction from the physical layer, and complex queries through the SQL language.
- easily create secondary indexes, perform complex inner and outer joins, count, sum, sort, group, and page your data across a number of tables, rows, and columns.

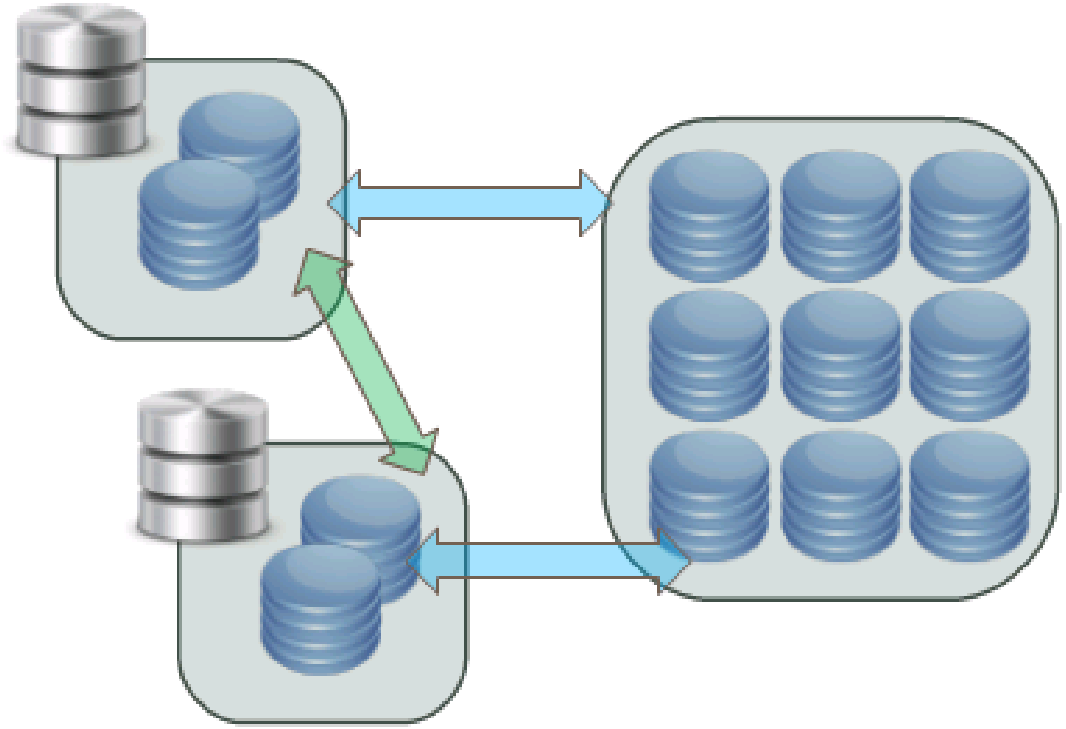
RDBMS vs MapReduce

	Traditional RDBMS	MapReduce
Data Size	Gigabytes *(Terabytes)*	Petabytes *(Hexabytes)*
Access	Interactive and Batch	Batch
Updates	Read / Write many times	Write once, Read many times
Structure	Static Schema	Dynamic Schema
Integrity	High (ACID)	Low
Scaling	Nonlinear	Linear
DBA Ratio	1:40	1:3000

Reference: Tom White's Hadoop: The Definiti



Comparing RDBMS and MapReduce



Traditional RDBMS: Move Data to Compute

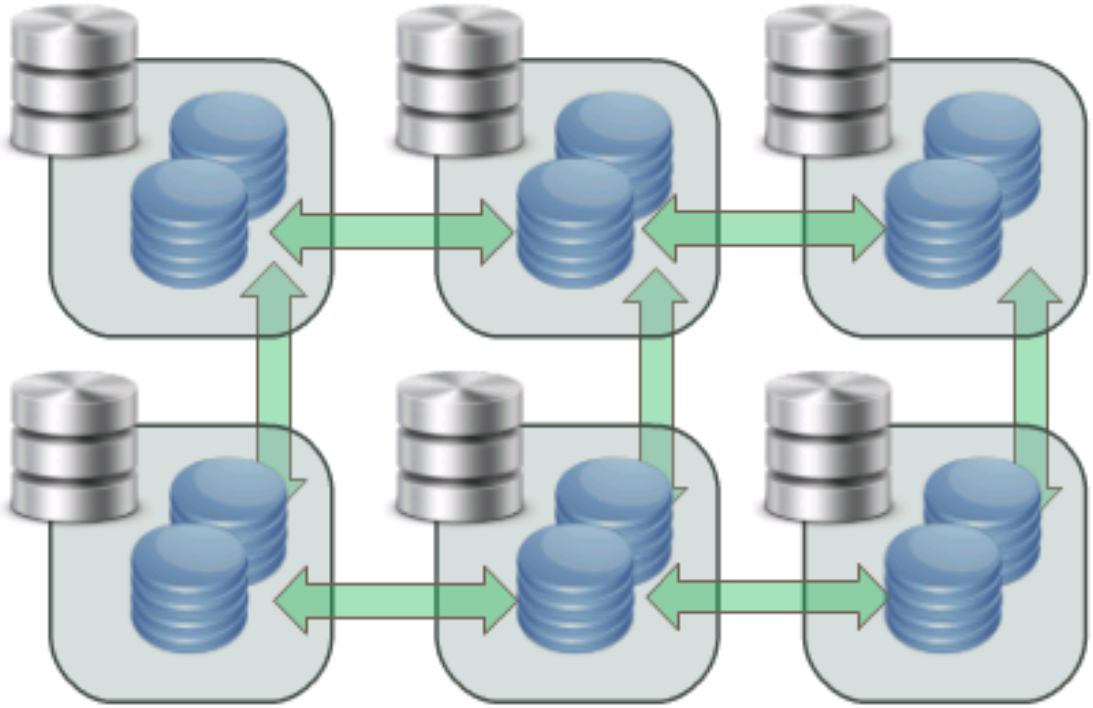
As you process more and more data, and you want interactive response

- Typically need more expensive hardware
- Failures at the points of disk and network can be quite problematic

It's all about ACID: atomicity, consistency, isolation, durability

Can work around this problem with more expensive HW and systems

- Though distribution problem becomes harder to do



Hadoop: Move Compute to the Data

Hadoop (and NoSQL in general) follows the Map Reduce framework

- Developed initially by Google -> Map Reduce and Google File system
- Embraced by community to develop MapReduce algorithms that are very robust
- Built Hadoop Distributed File System (HDFS) to auto-replicate data to multiple nodes
- And execute a single MR task on all/many nodes available on HDFS

Use commodity HW: no need for specialized and expensive network and disk

Not so much ACID, but BASE (basically available, soft state, eventually consistent)

Architechures

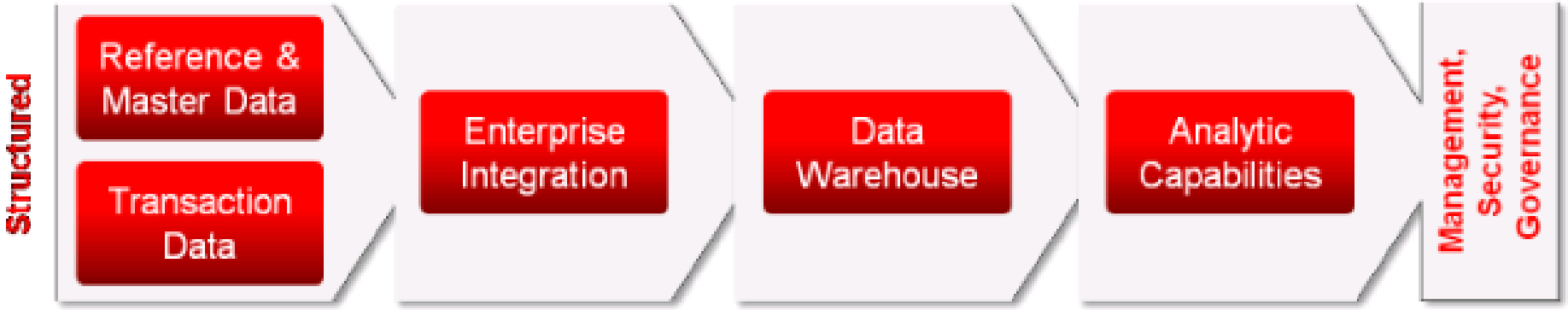


Figure 1: Traditional Information Architecture Capabilities

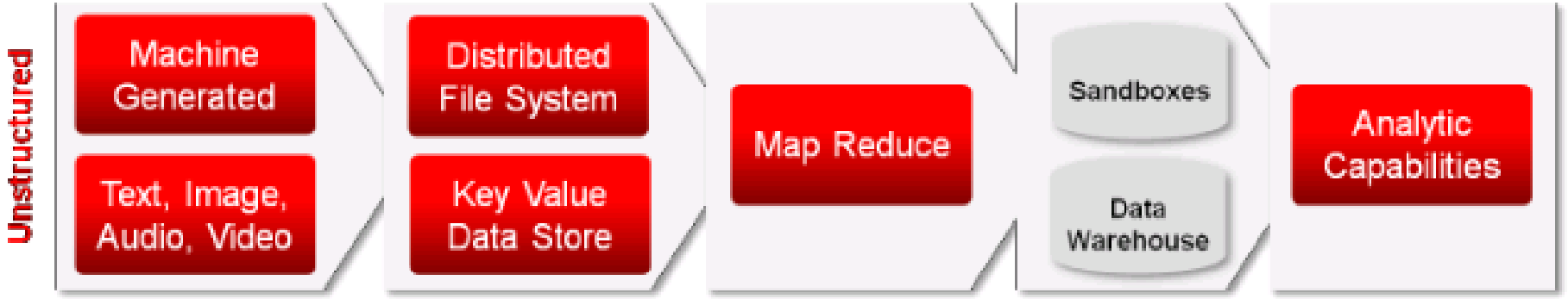
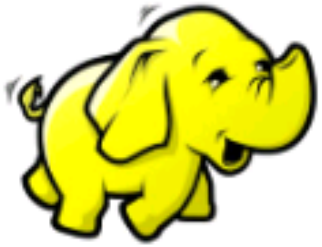


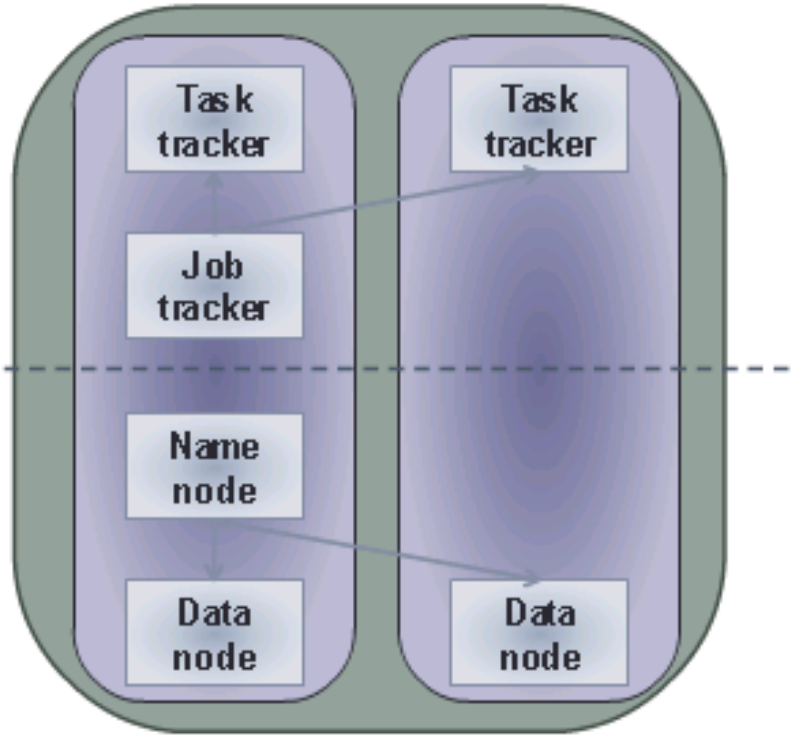
Figure 2: Big Data Information Architecture Capabilities

贵州科学院
Guizhou Academy of Sciences



Map Reduce Layer

HDFS Layer



Reference: http://en.wikipedia.org/wiki/File:Hadoop_1.png

What is Hadoop?

- Synonymous with the Big Data movement
- Infrastructure to automatically distribute and replicate data across multiple nodes and execute and track map reduce jobs across all of those nodes
- Inspired by Google's Map Reduce and GFS papers
- Components are: Hadoop Distributed File System (HDFS), Map Reduce, Job Tracker, and Task Tracker
- *Based on the Yahoo! "Nutch" project in 2003, became Hadoop in 2005 named after Doug Cutting's son's toy elephant*

Design of HDFS

- Namenodes (The Master)

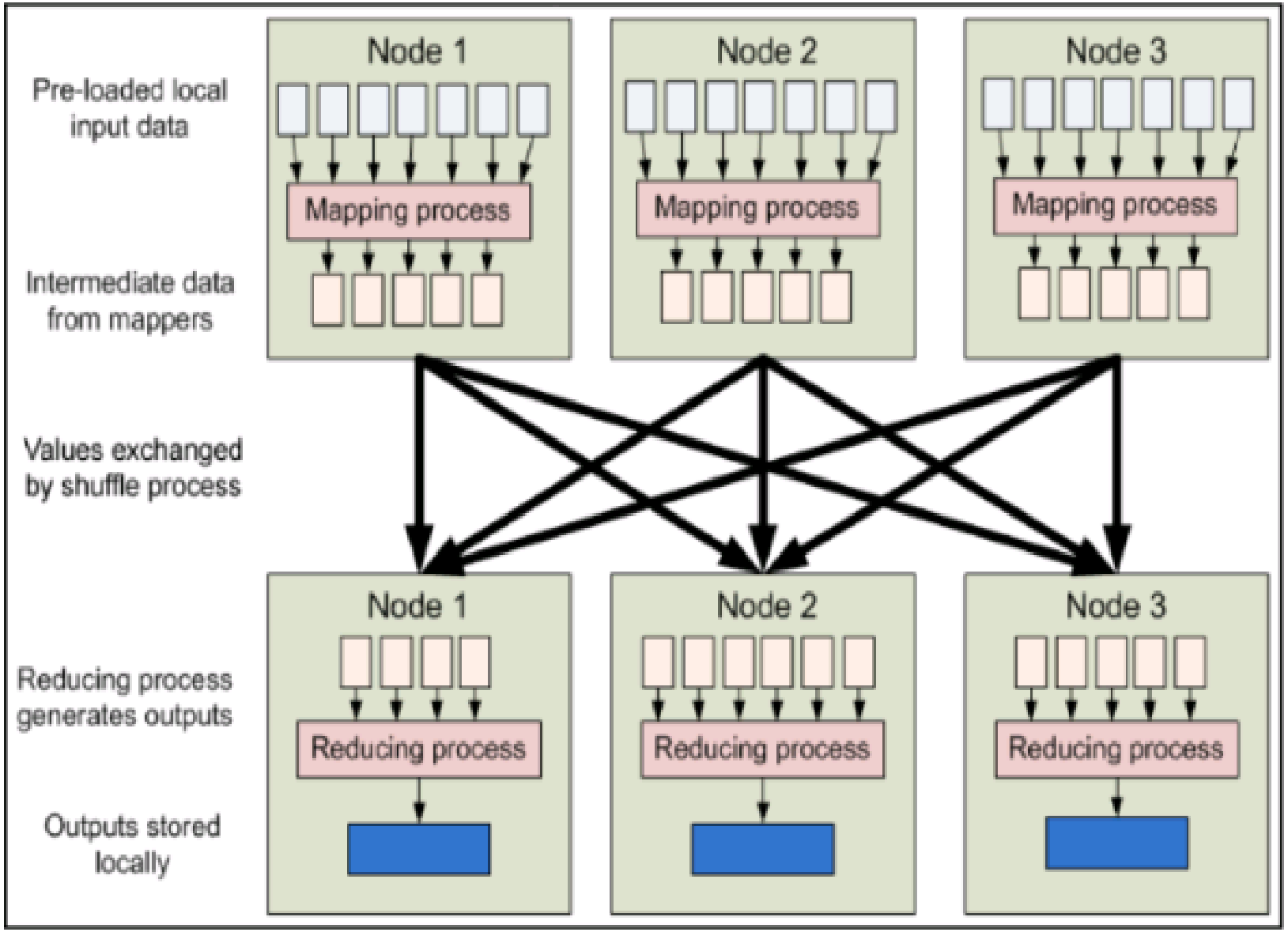
  Manage metadata/file trees
- Datanodes (Workers)

  store/retrieve data block
- Datanodes do not use RAID disk.

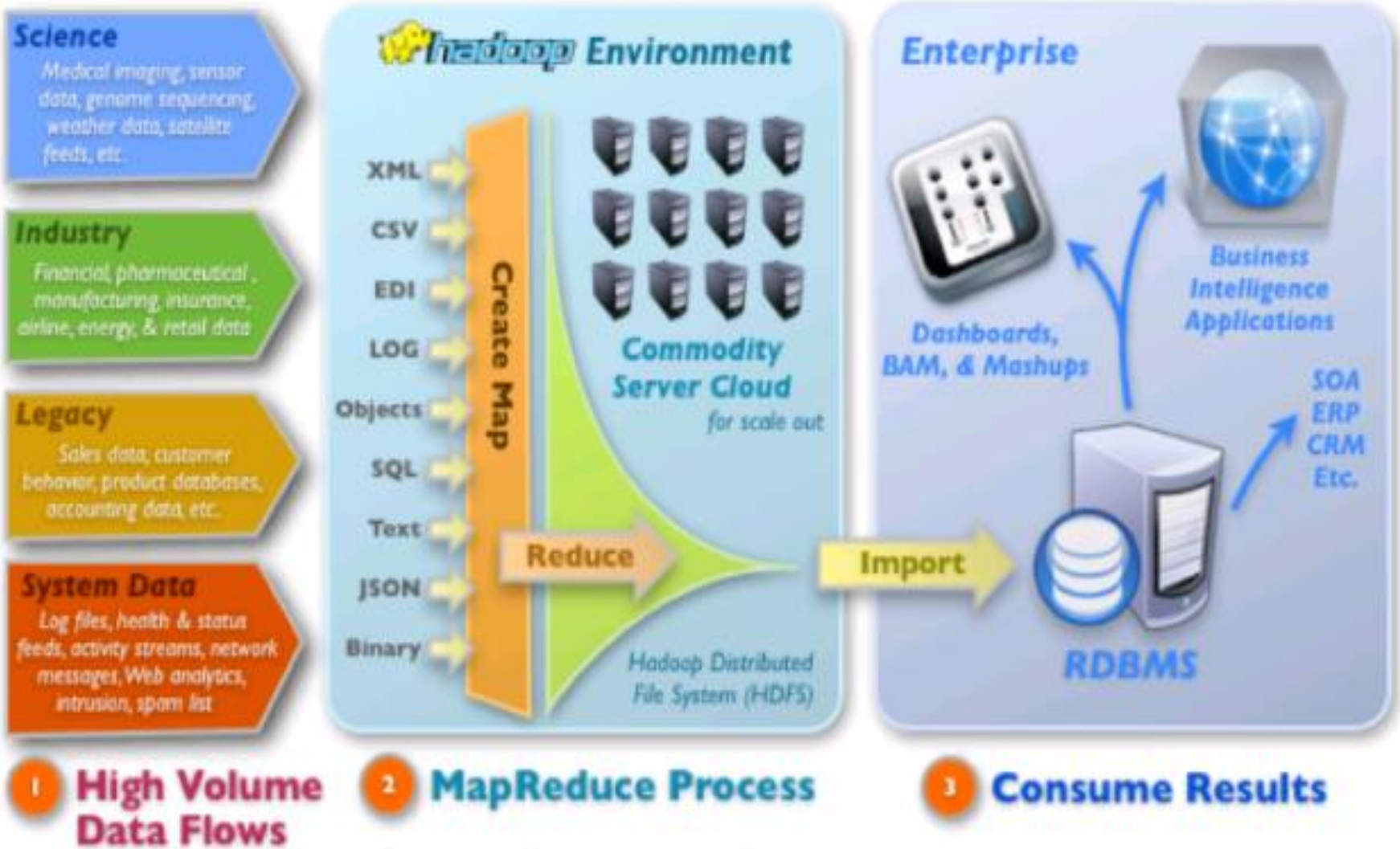
  HDFS round-robins HDFS blocks between all disks. RAID limited by the slowest disk on the array.
- Block

64 MB/128MB (normal disk block 512 KB).

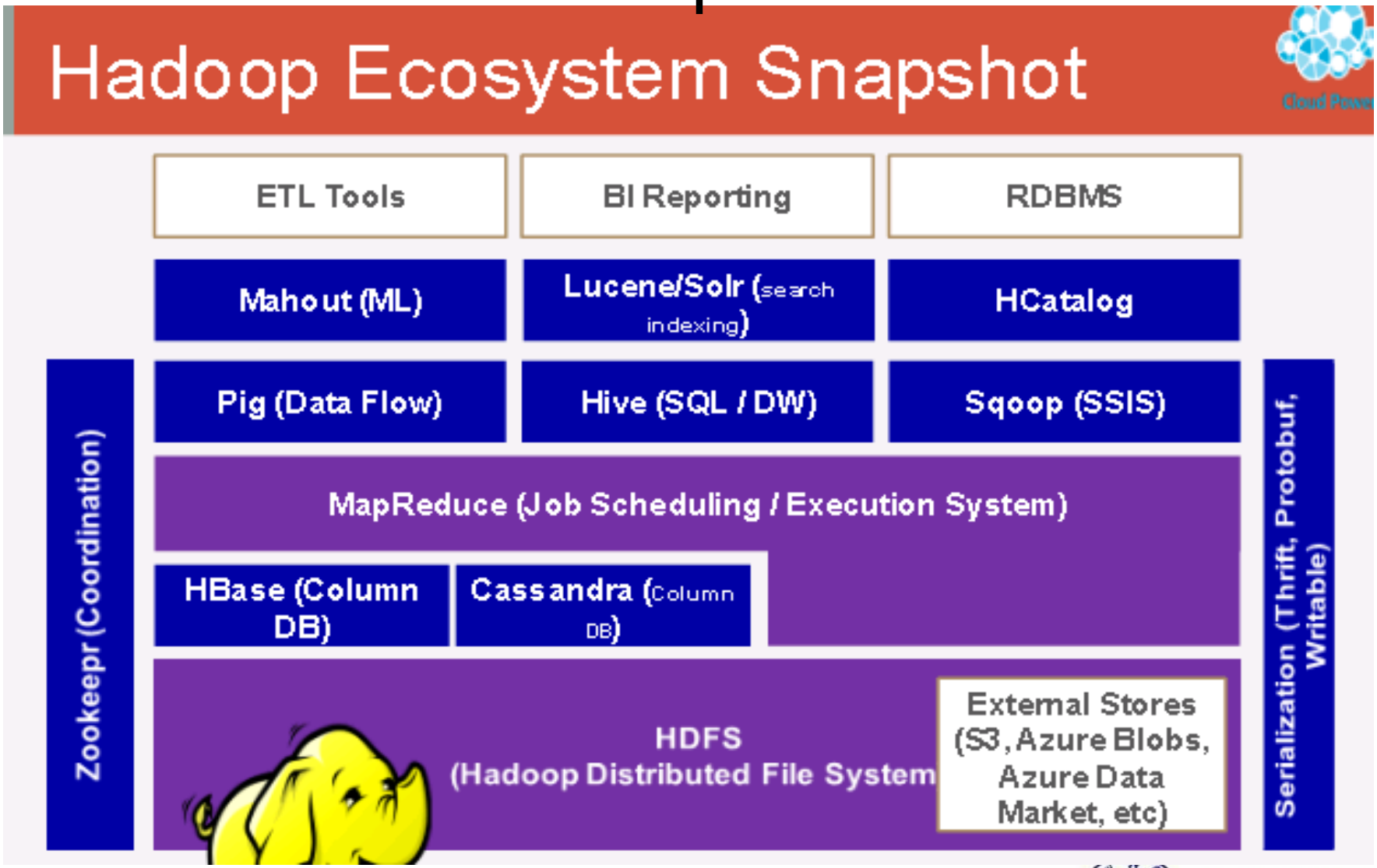
Data flows



Using Hadoop in the Enterprise

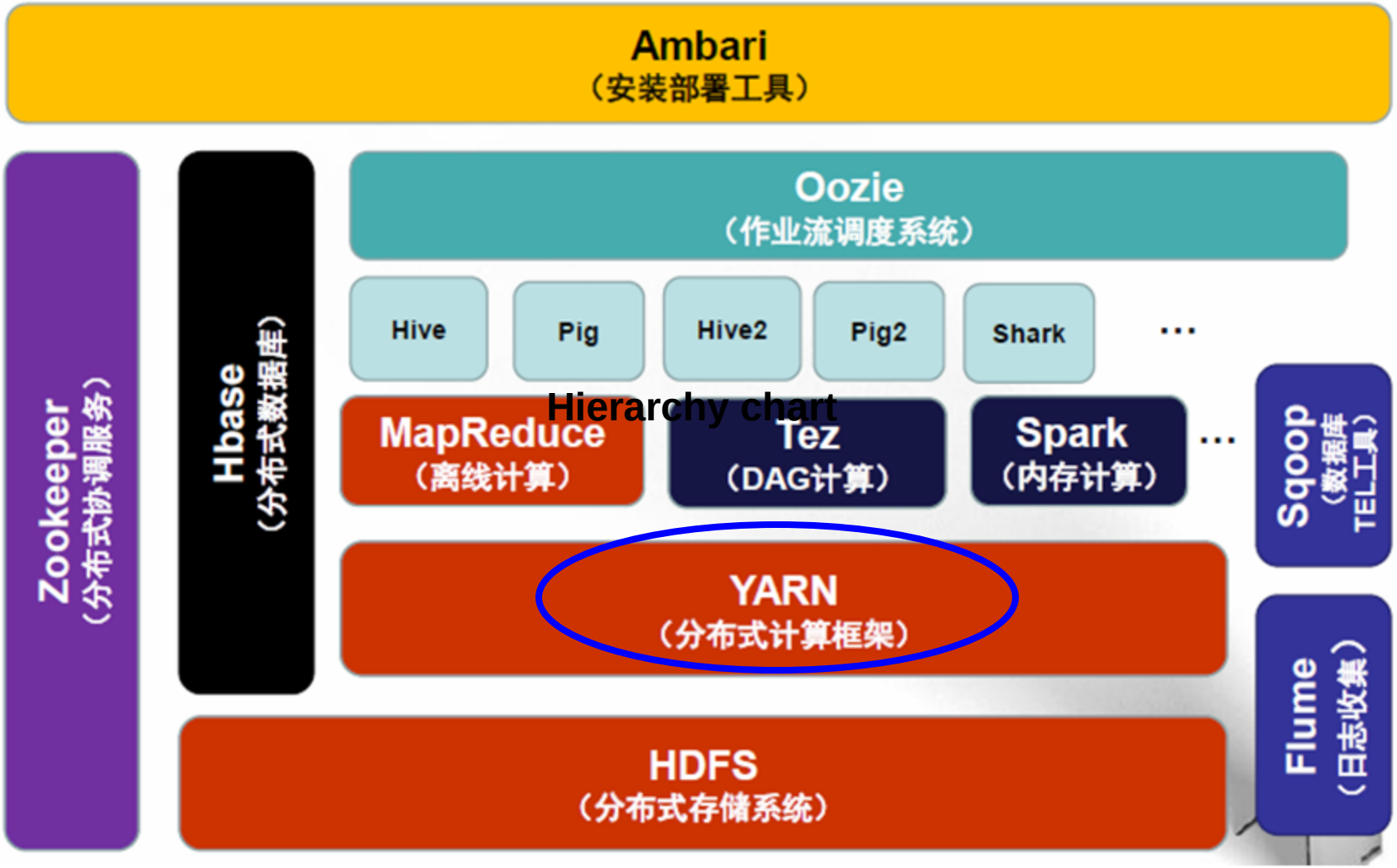


Hadoop 1.0



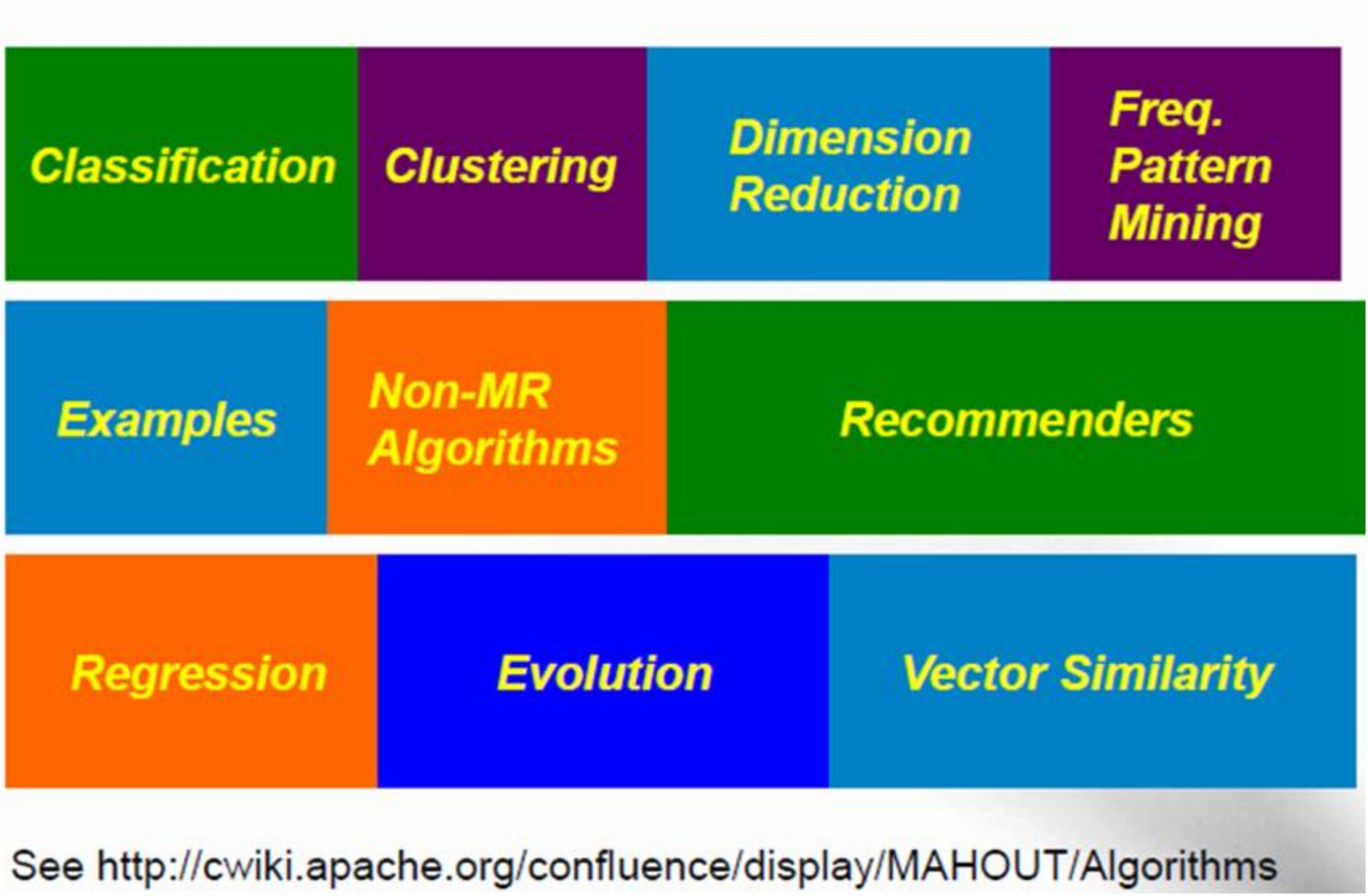
Hierarchy chart

Hadoop 2.0



Hierarchy chart

Hierarchy chart



Install Hadoop

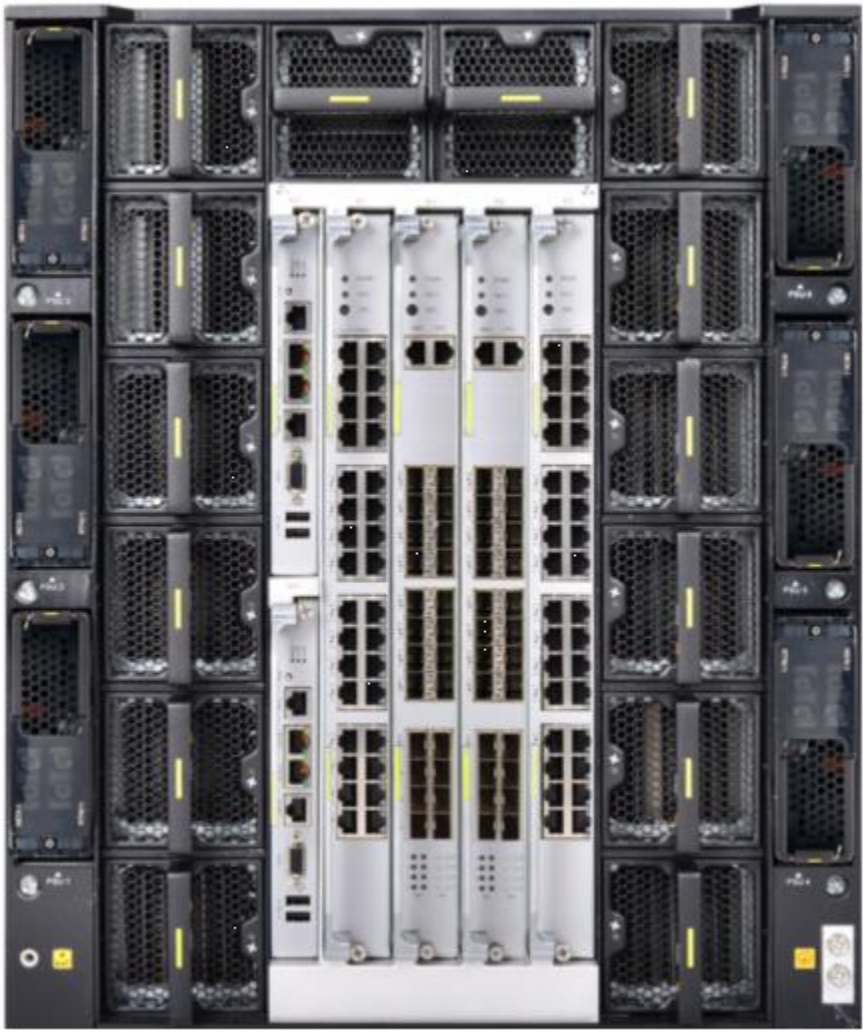
- % cd /usr/local
- % sudo tar xzf hadoop-x.y.z.tar.gz
- change the owner of the Hadoop files to be the hadoop user and group:
- % sudo chown -R hadoop: hadoop hadoop-x.y.z

- http://hadoop.apache.org/

We Start the Big Data Projects

- Considering the infrastructure used, all the experiments were run at the atlas research group cluster. This cluster is composed of 16 nodes, each with two Intel E5-2630 microprocessors (at 2.30 GHz, 128 GB cache) and 2 TB of main memory, connected with 1 Gb/s ethernet. All of them work under Linux CentOS 6.4.

- The cluster is configured with Hadoop and Mahout. One of the nodes is configured as name-node and job-tracker, and the remaining nodes are both datanodes and trask-trackers.The Hadoop version used is 2.0 (Cloudera CDH5.2.0) and the Mahout version is 0.8.



贵州科学院
Guizhou Academy of Sciences

Inferring friendship network structure by using mobile phone data

- 中国联通每天新增1500亿条上网行为记录
- 每条上网行为记录由26个字段组成

Real-Time Urban Monitoring Using Cell Phones

13003759599	8551	55401	128390081 39491	304	2013/11/12 7:25:26	2013/11/12 7:30:36	311	1277	104	1381	0	10.76.254.10	140.206.160.215	0		UNINET	460015358901400	116
13003763810	55082	27612	865521012428430	304	2013/11/12 7:25:29	2013/11/12 7:25:29	1	375	0	375	1	10.51.233.6	140.207.54.36	0		3gnet	460017896601560	220
13003763810	55082	27612	865521012428430	304	2013/11/12 7:25:29	2013/11/12 7:29:30	240	112	316	428	1	10.51.233.6	140.207.54.36	0		UNINET	460015850933188	220
13003765732	55082	34171	866166010129677	304	2013/11/12 7:25:26	2013/11/12 7:25:26	1	56	64	120	1	10.60.181.115	140.206.160.215	0		3gnet	460017575470080	220
13004621608	17941	61666	860312024090890	304	2013/11/12 7:25:28	2013/11/12 7:27:38	130	862	1216	2078	2	10.60.168.125	140.206.160.215	0		3gnet	460011909263237	220
13004621608	17941	61666	860312024090890	304	2013/11/12 7:25:28	2013/11/12 7:25:28	1	68	0	68	2	10.60.168.125	140.206.160.215	0		3gnet	460012212758054	116
13004621608	17941	61666	860312024090890	304	2013/11/12 7:25:28	2013/11/12 7:25:28	;	68	0	68	2	10.60.168.125	140.206.160.215	0		3GNET	460017635914516	220
13004621608	17941	61666	860312024090890	304	2013/11/12 7:25:29	2013/11/12 7:25:29	1	68	0	68	2	10.60.168.125	140.206.160.215	0		3GNET	460010905271381	116
13004719068	55116	3336	356551040012500	304	2013/11/12 7:25:27	2013/11/12 7:29:29	241	52	52	104	1	10.73.141.131	140.206.160.215	0		3gnet	460018925400428	220
13004719068	55116	3336	356551040012500	304	2013/11/12 7:25:29	2013/11/12 7:25:29	1	52	0	52	1	10.73.141.131	140.206.160.215	0		3gnet	460016925039910	220
13004777799	55144	14323	354965057423220	304	2013/11/12 7:25:27	2013/11/12 7:25:27	1	52	128	180	1	10.73.217.102	140.207.54.36	0		3gnet	460015626061273	220
13008929535	55076	17001	353723055111833	304	2013/11/12 7:25:26	2013/11/12 7:25:26	1	56	0	56	1	10.47.158.250	140.206.160.215	0		3gnet	460015866080333	220
13008933263	55082	31101	860308025151310	304	2013/11/12 7:25:30	2013/11/12 7:27:17	108	1152	1408	2560	1	10.60.13.138	140.206.160.215	0		3gnet	460014633022036	220
13008943055	55100	26751	13468003734950	304	2013/11/12 7:25:30	2013/11/12 7:25:30	1	0	1204	1204	1	10.85.155.64	140.207.54.36	0		3gnet	460018275304019	220
13008944728	42257	41688	13354006821460	304	2013/11/12 7:25:28	2013/11/12 7:25:28	1	56	0	56	1	10.84.40.190	140.207.54.36	0		3gnet	460017405614367	220
13008955205	55082	41893	13417001487430	304	2013/11/12 7:25:26	2013/11/12 7:31:26	360	1418	1060	2478	1	10.76.237.87	140.207.54.36	0		3gnet	460016170904269	116

13056704641	16000	56140	13189002898501	304	2013/11/12 7:25:53	2013/11/12 7:30:12	259	1000	52	1052	2	10.85.92.83	140.207.54.36	0		3gnet	460016955004060	220
13002601383	16930	12662	12966005976782	304	2013/11/12 7:25:28	2013/11/12 7:28:03	154	1221	31167	32388	2	10.76.173.65	140.207.54.36	0		3gnet	460015870653290	116
13002601383	16930	12662	12966005976782	304	2013/11/12 7:25:28	2013/11/12 7:25:28	1	0	52	52	2	10.76.173.65	140.207.54.36	0		3gnet	460015741979456	220
13002601383	16930	12662	12966005976782	304	2013/11/12 7:25:28	2013/11/12 7:25:28	1	744	0	744	2	10.76.173.65	140.207.54.36	0		3gnet	460016630606833	116
13002601383	16930	12662	12966005976782	304	2013/11/12 7:25:29	2013/11/12 7:28:02	153	1136	0	1136	2	10.76.173.65	140.207.54.36	0		uniwap	460015958963313	220
13002601383	16930	12662	12966005976782	304	2013/11/12 7:25:30	2013/11/12 7:25:30	1	454	0	454	2	10.76.173.65	140.207.54.36	0		3gnet	460012212758530	116
13002627555	16011	4393	355637054708130	304	2013/11/12 7:25:28	2013/11/12 7:25:28	1	56	0	56	2	10.99.206.187	140.207.54.36	0		3GNET	460016071201697	116
13002631930	55220	22443	13183003933891	304	2013/11/12 7:25:30	2013/11/12 7:27:08	98	586	64	650	1	10.98.130.232	140.207.54.36	0		3gnet	460016620650430	116
13002631930	55220	22443	13183003933891	304	2013/11/12 7:25:30	2013/11/12 7:25:30	1	0	52	52	1	10.98.130.232	140.207.54.36	0		uniwap	460015958963313	220
13002631930	55220	22443	13183003933891	304	2013/11/12 7:25:30	2013/11/12 7:25:30	1	56	0	56	1	10.98.130.232	140.207.54.36	0		UNINET	460015756083879	220
13002662838	55136	32392	862966021080617	304	2013/11/12 7:25:29	2013/11/12 7:25:29	1	52	0	52	1	10.98.51.199	140.206.160.215	0		3gnet	460015850695993	220
13002665566	55142	38003	353638057787620	304	2013/11/12 7:25:26	2013/11/12 7:25:26	1	52	0	52	1	10.99.51.242	101.226.76.145	0		3gnet	460013740065529	220
13002682781	18210	31558	352343051868640	304	2013/11/12 7:25:28	2013/11/12 7:25:28	1	0	64	64	2	10.98.231.129	101.227.131.106	0		3gnet	460015800685630	220
13003700875	55080	36601	12958007001551	304	2013/11/12 7:25:27	2013/11/12 7:28:27	180	636	0	636	1	10.85.182.210	140.206.160.215	0		3gnet	460018915400870	220
13003701870	55084	28544	357376058551900	304	2013/11/12 7:25:26	2013/11/12 7:29:59	273	112	0	112	1	10.85.205.28	140.207.54.36	0		3gnet	460017975053326	116

- 2014 Chinese "cloud Guizhou" big data business model contest starts in Guiyang.
- We participated in the contest.

We won the Outstanding Project

贵州科学院
Guizhou Academy of Sciences

2015 Guiyang International Big Data Industry Fair

贵州科学院
Guizhou Academy of Sciences

Business Value

- Guiyang Global Big Data Exchange

Big Data References

Hadoop: The Definitive Guide by Tom White
SQL Server Sqoop http://bit.ly/rulsjX
JavaScript http://bit.ly/wdaTv6
Twitter https://twitter.com/#!/search/%23bigdata

Hive http://hive.apache.org
Excel to Hadoop via Hive ODBC http://tinyurl.com/7c4qjjj
Hadoop On Azure Videos http://tinyurl.com/6munnx2
Klout http://tinyurl.com/6qu9php

Microsoft Big Data http://microsoft.com/bigdata
Denny Lee http://dennyglee.com/category/bigdata/
Carl Nolan http://tinyurl.com/6wbfxy9
Cindy Gross http://tinyurl.com/SmallBitesBigData

Softwares

- http://hadoop.apache.org

- Oreilly: Hadoop. The definitiive Guide 3rd Edition. May 2012.pdf

- Oracle products and price

https://shop.oracle.com/pls/ostore/product?p1=OracleBigDataAppliance&p2=&p3=&p4=&sc=ocom_BigDataAppliance

- Microsoft

http://www.windowsazure.com/en-us/manage/services/hdinsight/

- IBM

http://www-01.ibm.com/software/data/bigdata/platform/product.html

- Cloudera

http://www.cloudera.com/content/cloudera/en/why-cloudera/hadoop-and-big-data.html