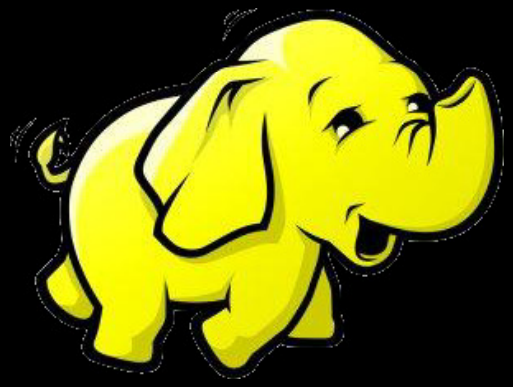
Elephant is Here……

Apache Hadoop

Introduction



What for Today……

Let's Ride the Elephant Today, and
Tame it Tomorrow ☺

- Why?? – You will ask….why Elephant Today, I m good in my own Car?
- Safari with this Elephant – A brief introduction of Hadoop Ecosystem

Big Data : Reality or Hype?

- *Information that can't be processed or analyzed using traditional tools or processes.*
- Growth of Unstructured information
  - Unstructured information is growing 15x faster than structured info
- Coming at speed of monster
  - 90% of the data in the world today has been created in the last two years alone
- CPUs power is growing
  - CPUs are growing so fast that a commodity server today is close to the power of a supercomputer 5 years ago
- Storage costs going down
  - Storage costs on spinning disks is approaching free:
  - 1981: $700 / MB 2001: $0.01 / MB 1995: $10 / MB 2011: 0.002c / MB (so 2 cents per GB)

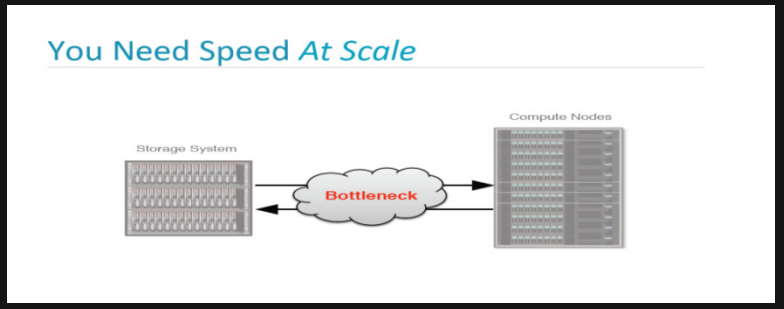
Big Data: Characterstics

- Volume – Amount of data growing from GBs to PBs
  - Eg. Twitter turns 12 TB of tweets a day into improved product sentiment analysis
    - So, you are saying, we need Some Big Storage – Distributed
- Velocity – Speed of data in/out
  - Banks analyze nearly 5 million transactions a day to analyze fraud
    - That means, your data will keep on growing – Scalable
- Variety – Range of data types and sources
  - Eg. Social, phone sensors, documents, email, video, still images…..
    - And for this you need Flexible storage

Let's Bring our Destination Close

- But we already have good Distributed Storage systems– **Go and use them**
- Let's take a look– Issues with current SAN
  - Moving data to code – n/w bandwidth is bottleneck

So let's try Moving code to data



- No Ferrari
  - Moving code to data – Easier said than done -- --- Why not done earlier – Data nodes not have much computing capacity
  - Let's not increase the cost – Try to use Commodity servers
  - When we talk about commodity servers – Fault Tolerance

What's d Plan

- What we are planning to do with such huge amount of data?
  - Storing – certainly not
  - We need processing – what type
    - Realtime or Batch → Batch Processing

- What problem u have with my favorite RDBMS --
  - Have fix schema
  - You loose lots of important info which doesn't fits in schema
  - Expansion of schema is not easy
  - Growing amount of unstructured data which doesn't fit there
  - Let's rid away from fix schema – Flexible storage format – Raw form

It's too much of wishes!!!

- Big Storage – Distributed
- Scalable
- Flexible storage
- Moving code to data
- Commodity servers
- Fault Tolerance
- Batch Processing

This is what HADOOP is ☺

But probably u want formal definition as well, Here u go -

- Apache Hadoop is an open-source, reliable, scalable, cost-effective, flexible distributed file system solution to store millions of large files meant for batch processing (large streaming reads/writes).

Still Thinking – tik tik

- Take example of Google – motivation for Hadoop
- Just "Google it"
  - How Google is managing such huge amount of data
  - U enter and results r thr – if they r searched realtime – NO
  - Then How – Batch runs and they r indexed

Something more closer to you…..

- Facebook
- Yahoo

- How NSA managing to monitor ???
  - new $2 billion facility in Utah is estimated to store 100 years worth of the worlds' electronic communication

More for you….but won't fit this page

How Big ?

Yahoo!

- RedHat linux 5.x
- 25 PB of storage (70% -> HDFS)
- 4500 machines
- 4 – 12 SATA drives (2 TB each)
- 2 quad core Xeon CPUs @ 2.5 GHz
- 24 GB of RAM per machine
- 1 Gigabit Ethernet

- 70 million files / 90 million blocks

Facebook

- 60 PB of storage
- 2600 machines
- 12 TB per machine
- 8-16 cores per machine
- 32 GB of RAM per machine
- 70+ million files in HDFS
- 30,000 simultaneous clients to HDFS NameNode
- 25,000 MapReduce jobs a day

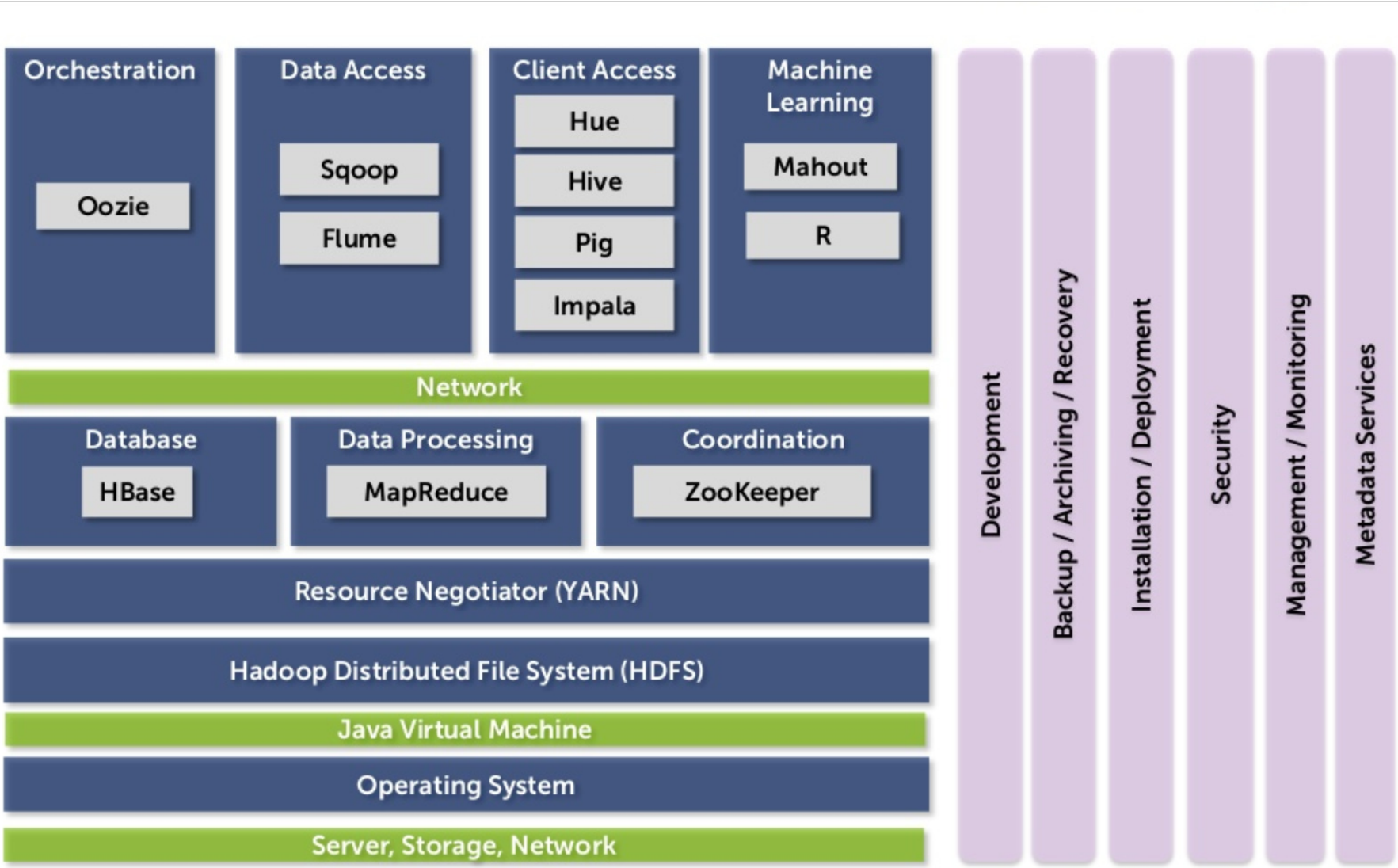
Should I stop using RDBMS ??

So much of mouthful words for Hadoop, so should I stop using my MySQL, Oracle etc…

- When Hadoop is not for you (changing now)—
  - Low latency Access
  - Schema of data to be stored can be easily defined
  - Small files
  - It's not intended to replace RDBMS

From where it came from?

Hadoop EcoSystem



HDFS

- Hadoop Distributed File System
- NameNode -- Master
  - Bookkeeper of HDFS
  - Metadata kept in memory for fast access
  - RAID
- DataNode -- Slave
  - Grunt work of HDFS
  - R+W HDFS blocks to local FS
  - Direclty communicate with Client
  - Replication
  - JBOD

MapReduce

JobTracker

"Word Count!"

Map + Reduce

Map + Reduce

Map + Reduce

Map + Reduce

TaskTrackers

HBase

- Column Oriented database on HDFS
  - Use for random realtime r/w access to data
  - Can Handle billion of rows and millions of columns
  - Based on Google's Big Table
  - Tables can be input/output for MapReduce jobs
  - Drives Facebook new messaging platform

Hive

SQL-like data warehouse infrastructure

- Ideal for capturing and analyzing streams of events (e.g. web logs)
- Targeted for data analysts who are comfortable with SQL and need to do ad hoc queries, summarization and data analysis.

```
INSERT OVERWRITE TABLE user_active
SELECT user.*
FROM user
WHERE user.active=1;
```

facebook

Pig

High-level data flow language

2 main components:
- Pig Latin: high-level data processing language
- Compiler: compiles and runs Pig Latin scripts in a choice of evaluation mechanisms

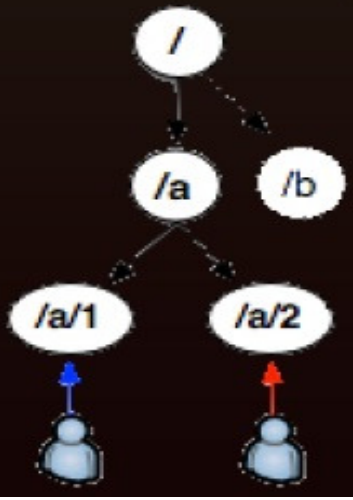
```
log = LOAD 'excite-small.log' AS (user, time, query);
grpd = GROUP log BY user;
cntd = FOREACH grpd GENERATE group, COUNT(log);
DUMP cntd;
```

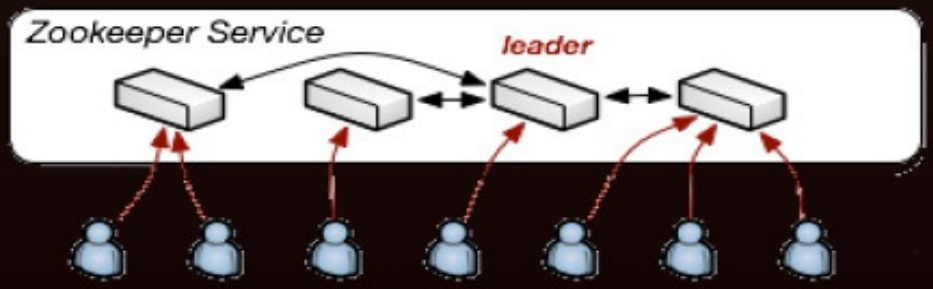
YAHOO!

Zookeeper

Distributed lock service

- Metadata File System
- Centralized configuration service
- Distributed synchronization





Mahout

Machine learning library

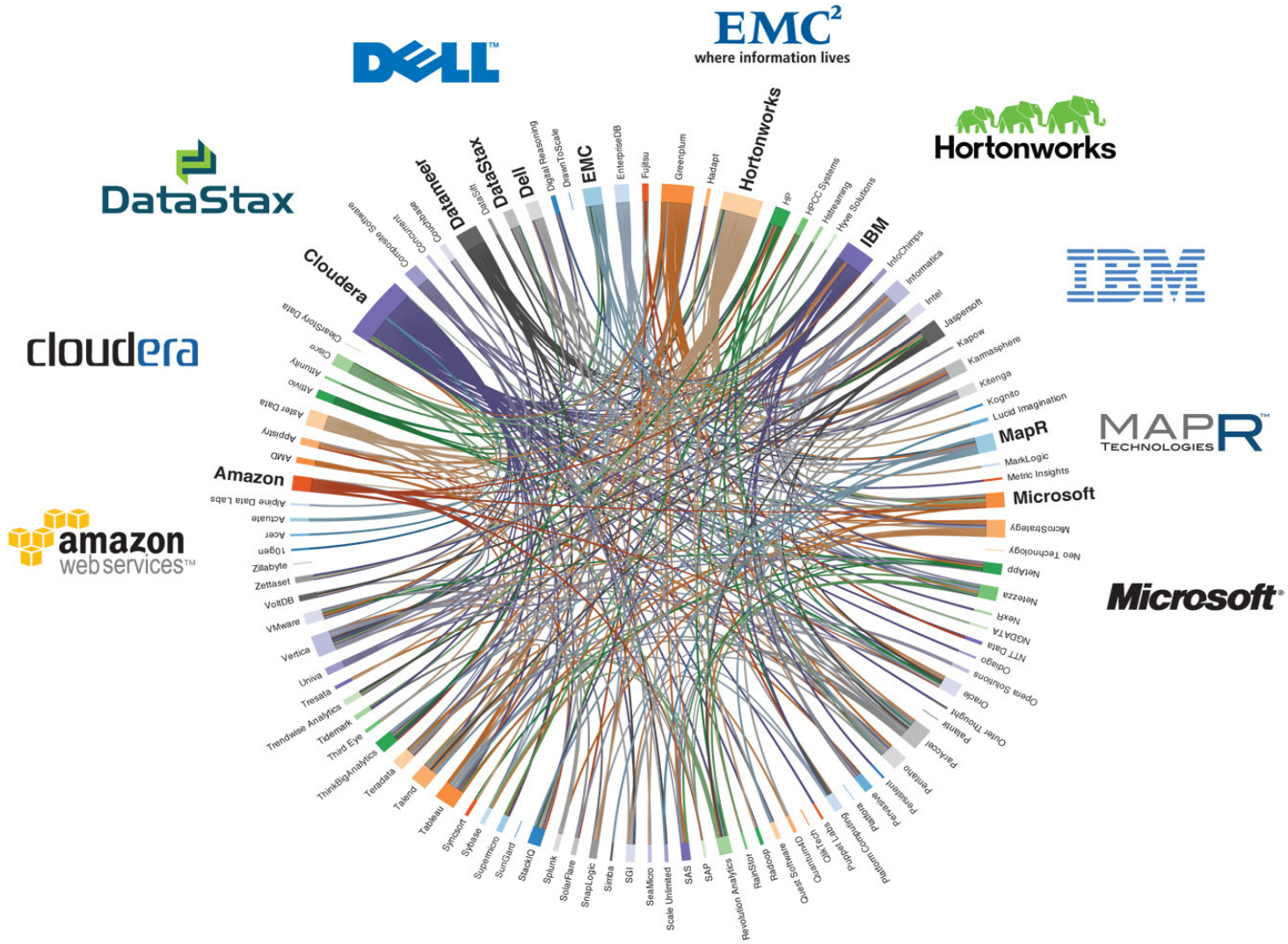
"For today's graduate, just one word: Statistics" – NY Times

"I keep saying that the sexy job in the next 10 years will be statisticians. And I'm not kidding" – Hal Varian, chief economist at Google

4 use cases:

- Recommendation mining
- Clustering
- Classification
- Frequent item set mining

Hadoop Vendors



Thank You