

The background of the slide is a dark blue field filled with a complex, glowing network of lines and nodes. The nodes are small dots, and the lines are thin, creating a web-like structure that resembles a data network or a molecular model. The overall effect is a sense of connectivity and digital complexity.

Introduction to Big Data

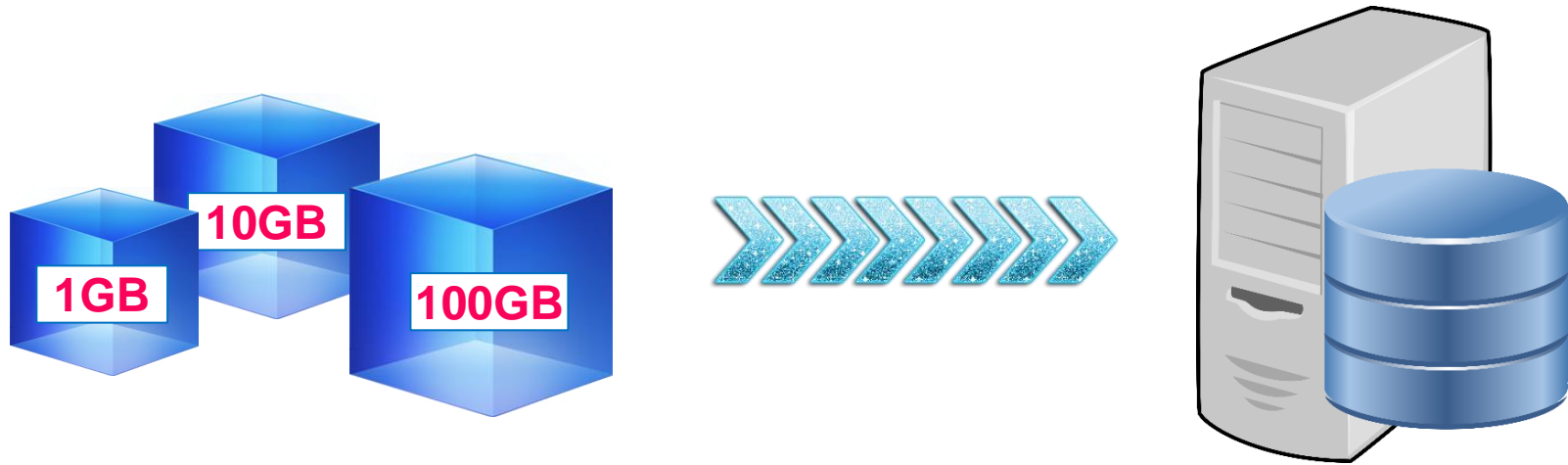
APACHE HADOOP

Le Ngoc Thanh – Nguyen Ngoc Thao
{lnthanh, nnthao}@fit.hcmus.edu.vn

Outline

- An introduction to Apache Hadoop
- When to use and not to use Hadoop
- Hadoop installation
- Hadoop: Now and Future

Imagine a scenario where...



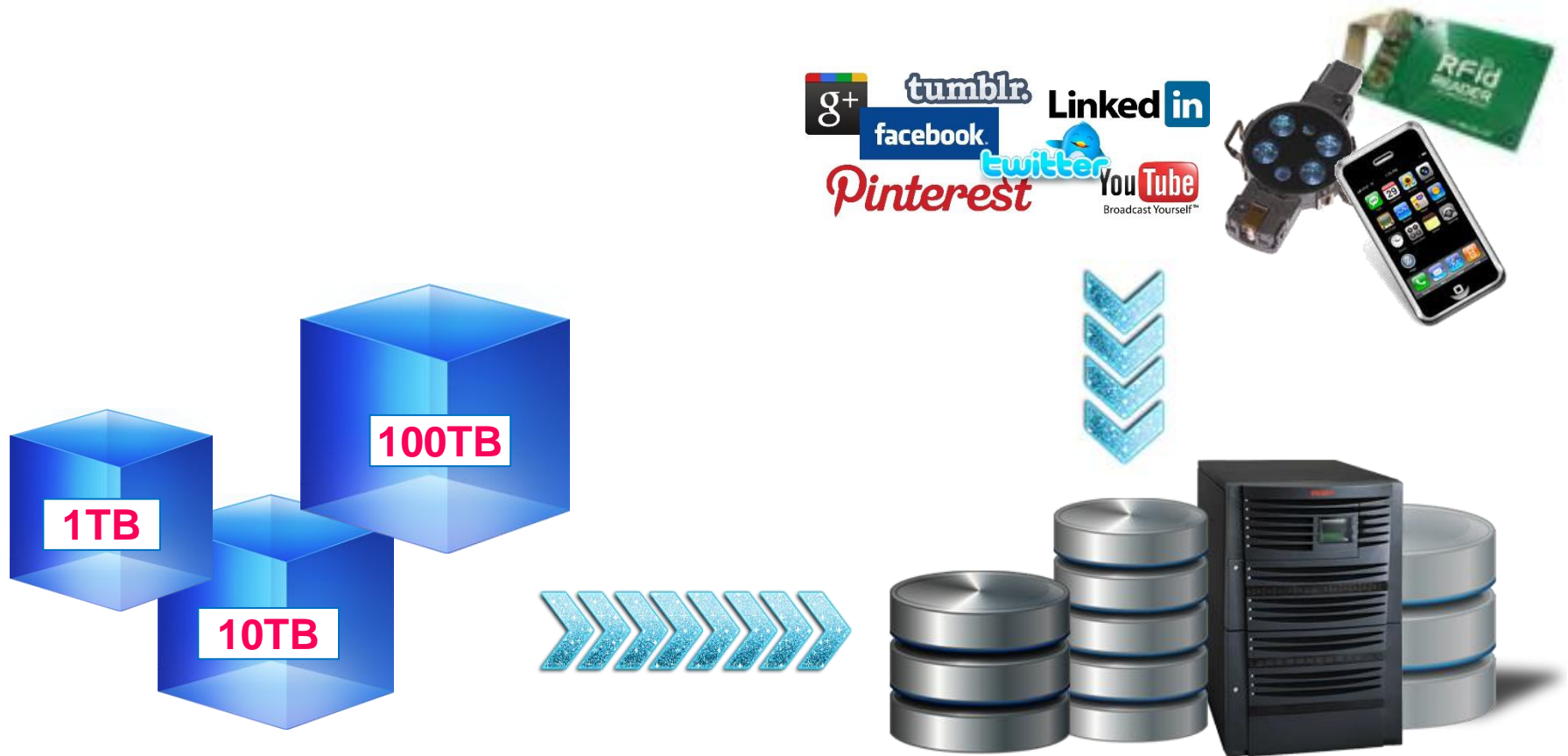
You have 1GB of data to process.... No problem!

Your company starts growing very quickly, and that data arises to 10GB

And then 100GB....

You start to reach the limits of your current desktop computer.

Imagine a scenario where...



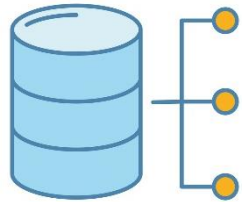
Therefore, you scale-up by investing in a larger computer, but...

the data increases quickly in a few months

it is required to feed the application with unstructured data

Imagine a scenario where...

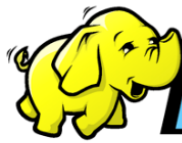
You may want to



Derive information from data of various types, not limited to relation data only.



Obtain the derivation as soon as possible.



hadoop may be your answer!!!

What is Hadoop?



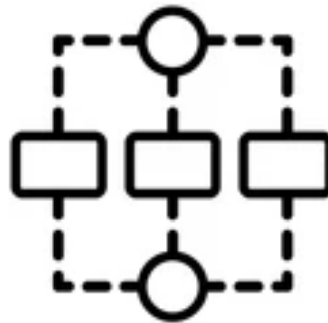
Written in **Java**



ASF project



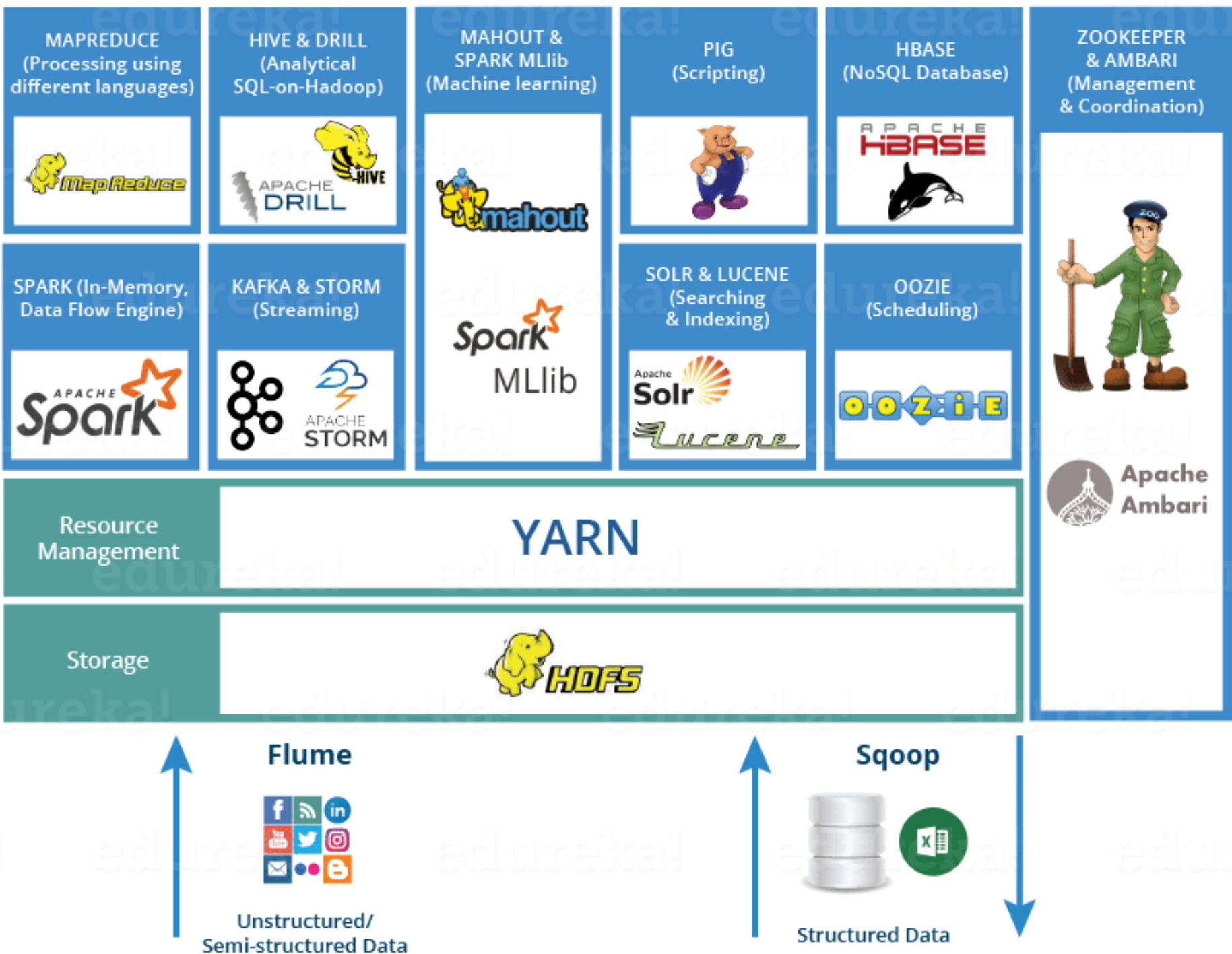
Distributed storage



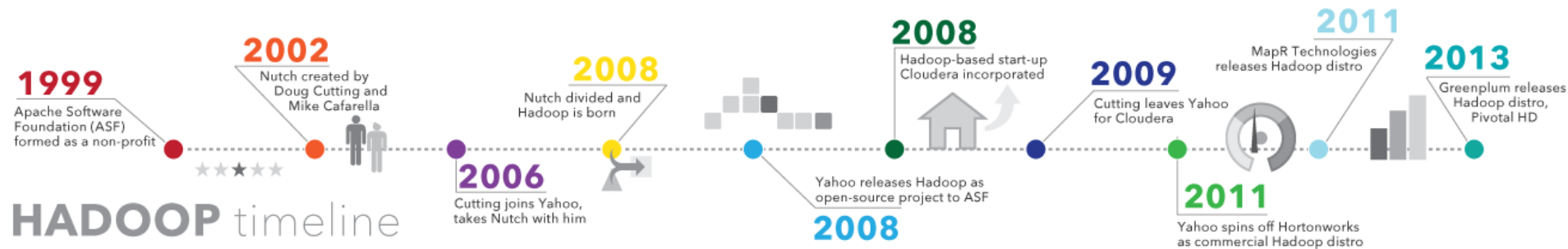
Massive parallel
processing



Reliable replication



A brief history of Hadoop

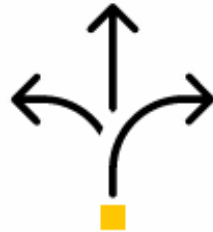


- 2002 – Dough Cutting and Mike Caferella created the Nutch project.
- 2006 – Cutting joined Yahoo and the Nutch project was divided.
 - The web crawler portion remained as Nutch.
 - The distributed computing and processing portion became Hadoop, which was later released as an open-source project.
- 2008 – Yahoo released Hadoop as an open-source project.
- Today, Hadoop's framework and ecosystem of technologies are managed and maintained by ASF

Hadoop framework: Benefits



Fast



Flexible



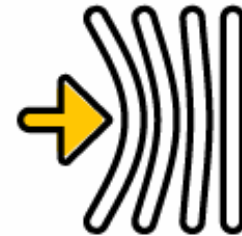
Scalable



Cost-effective



High throughput



Resilient to
failure

Hadoop framework: Benefits

- **Fast:** Hadoop makes data processing hassle-free and faster.
 - It has been found that Hadoop can process terabytes of unstructured data in just a few minutes, while petabytes in hours.
- **Flexible:** Hadoop helps gather data from different sources, of various types, and get valuable insights for many purposes.
 - Data sources: social media, emails, etc.)
 - Data types: structured, semi-structured, or unstructured.
 - Purposes: log processing, market analysis, fraud detection, etc..
- **Scalable:** Businesses can store and distribute large data sets from hundreds of servers that operate parallelly.

Hadoop framework: Benefits

- **Cost-effective:** Hadoop runs on commodity hardware.
 - One can easily increase nodes without suffering from any downtime of pre-planning requirements.
- **High throughput:** More jobs can be done in less time.
 - A small job is split into multiple chunks of data in parallel, which are easier to handle.
- **Resilient:** It is possible to recover data whenever any node goes down.
 - It stores replicas of every block at different nodes in the cluster.

Hadoop framework: Limitations



Issues with
small files



Iterative processing



Low security



Higher vulnerability



Support only
Batch processing

Hadoop framework: Limitations

- **Issues with small files:** Hadoop lacks the potential to support random reading of small files efficiently and effectively.
 - A small file is comparatively smaller than the HDFS block size.
 - A vast number of small files may overload the HDFS namespace.
- **Iterative processing:** Hadoop is an unfit choice for machine learning or iterative processing-based solutions.
 - The data flow in Hadoop framework is in the form of a chain, such that the output of one becomes the input of another stage.

Hadoop framework: Limitations

- **Low security:** Security model is disabled by default.
 - Hadoop does not offer encryption at the storage and network levels.
- **Higher vulnerability:** Cybercriminals may easily get access to Hadoop-based solutions.
 - Java is a popular, yet heavily exploited programming language.
- **Support only Batch processing:** MapReduce fails to take advantage of memory to the maximum.

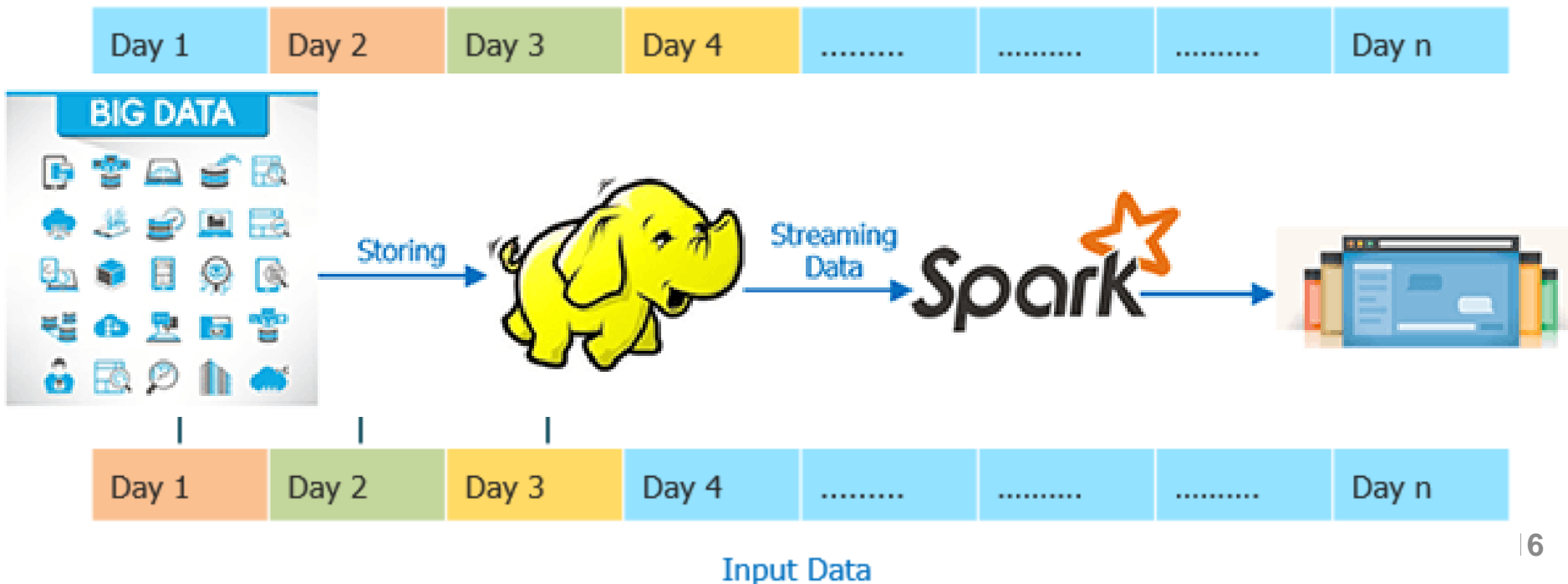


When to use Hadoop?

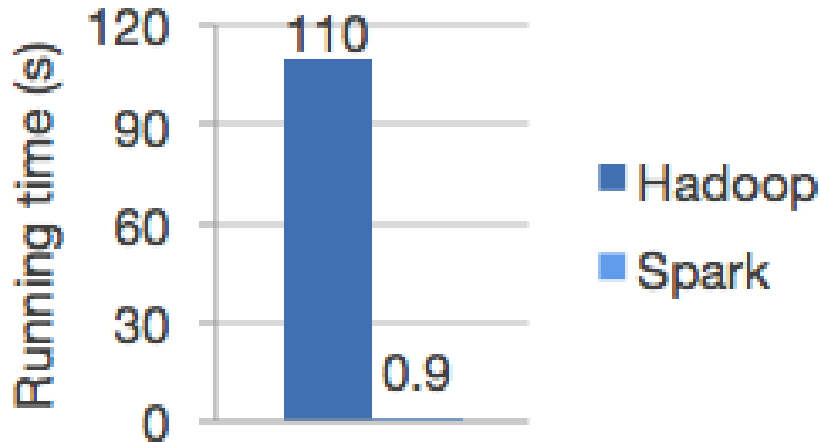
When not to use Hadoop?

- **Real-time analytics:** Results are expected to come quickly.
 - Hadoop works on batch processing → response time is high
 - Alternative: store the Big data in HDFS and mount Spark over it to make the processing real time

Processing Data using MR

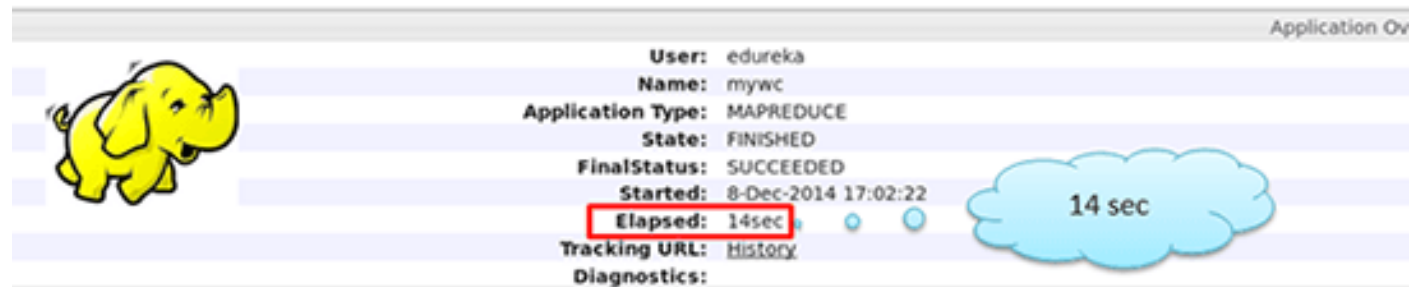


Hadoop vs. Spark: Performance



Spark runs programs up to 100x faster than Hadoop MapReduce in memory, or 10x faster on disk.

A simple example of line processing in Hadoop and Spark



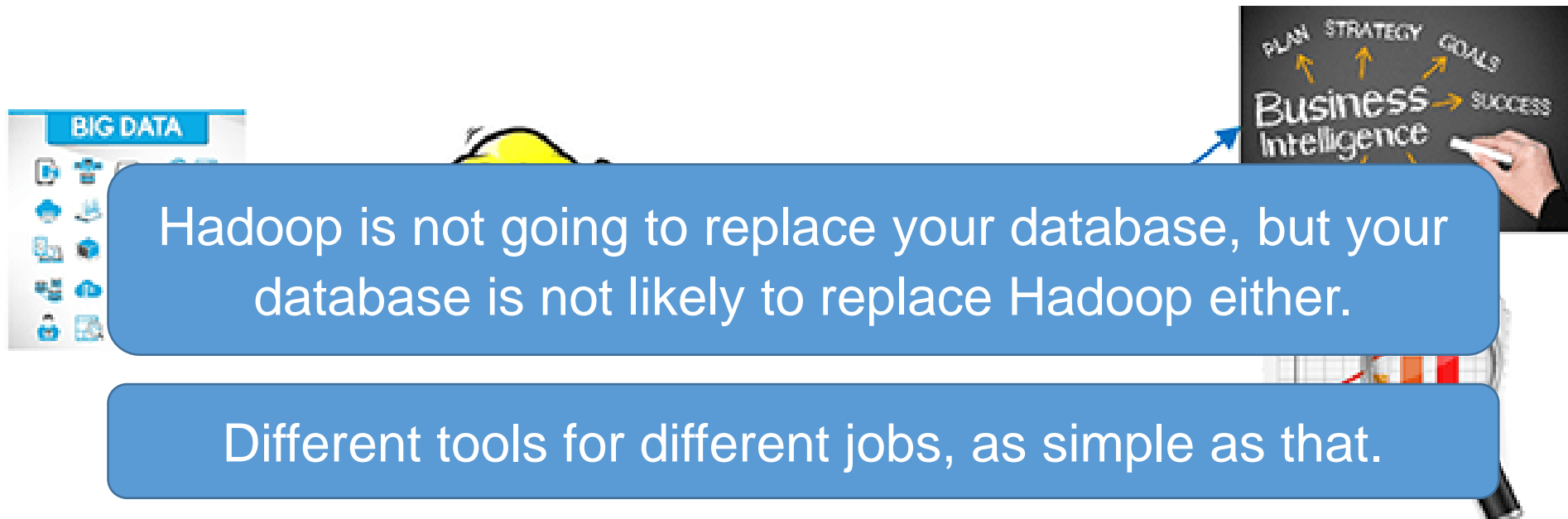
Logistic regression in Hadoop and Spark



```
14/12/08 04:10:06 INFO spark.SparkHadoopWriter: attempt_201412080410_0005_m_000000_5: Committed
14/12/08 04:10:06 INFO scheduler.TaskSetManager: Finished task 0.0 in stage 5.0 (TIO 5) in 296 ms on localhost (1/1)
14/12/08 04:10:06 INFO scheduler.TaskSchedulerImpl: Removed TaskSet 5.0, whose tasks have all completed, from pool
14/12/08 04:10:06 INFO scheduler.DAGScheduler: Stage 5 (saveAsTextFile at <console>:14) finished in 0.279 s
14/12/08 04:10:06 INFO spark.SparkContext: Job finished: saveAsTextFile at <console>:14, took 0.626043309 s
14/12/08 04:10:06 INFO executor.Executor: Finished task 0.0 in stage 5.0 (TIO 5). 826 bytes result sent to driver
wordCounts: Unit = ()
```

When not to use Hadoop?

- An alternative to existing data processing infrastructure
 - Data can be stored in HDFS, processed and transformed into structured manageable data.
 - Formatted data is then sent to RDMBS for BI, reporting, etc.

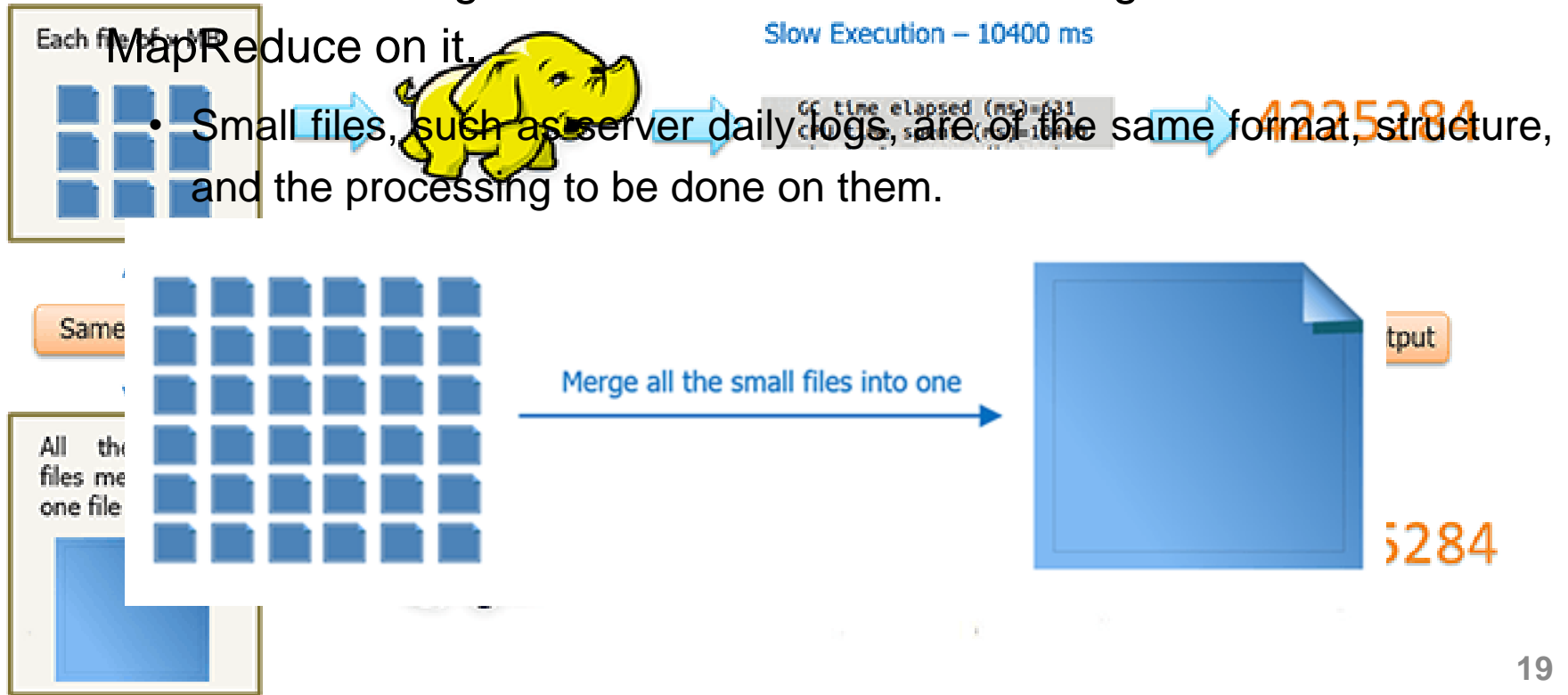


Hadoop is not going to replace your database, but your database is not likely to replace Hadoop either.

Different tools for different jobs, as simple as that.

When not to use Hadoop?

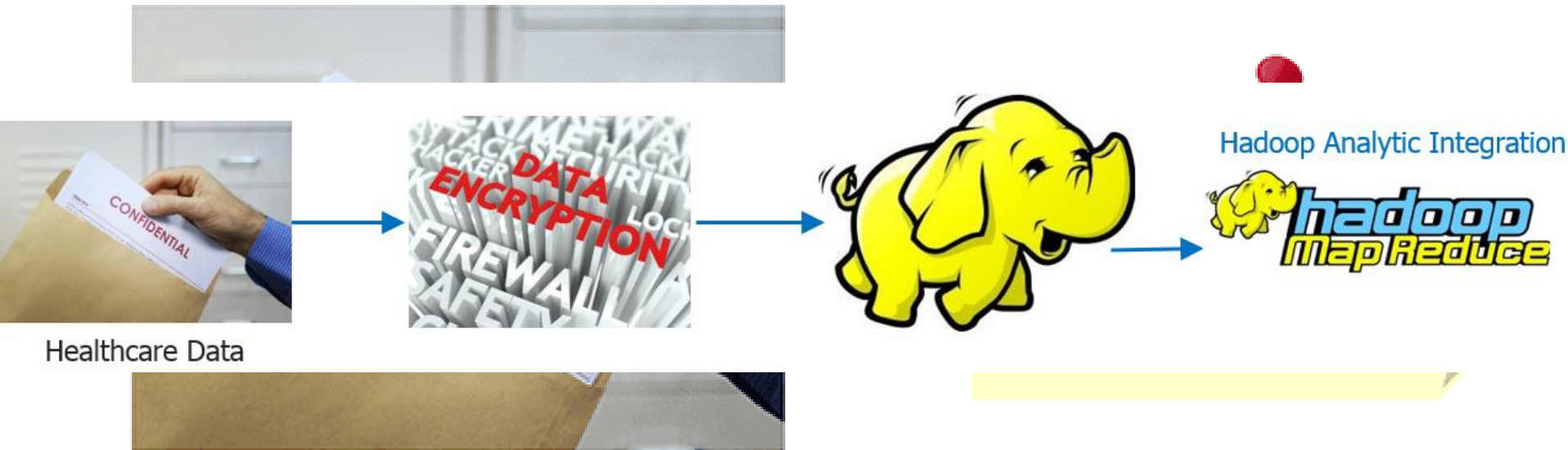
- There are multiple smaller datasets.
 - Hadoop may be costlier than other tools (e.g., MS Excel, RDBMS, etc.) on small-structured datasets.
 - Alternative: merge all small files into one big file and then run



When not to use Hadoop?

- Security is the primary concern

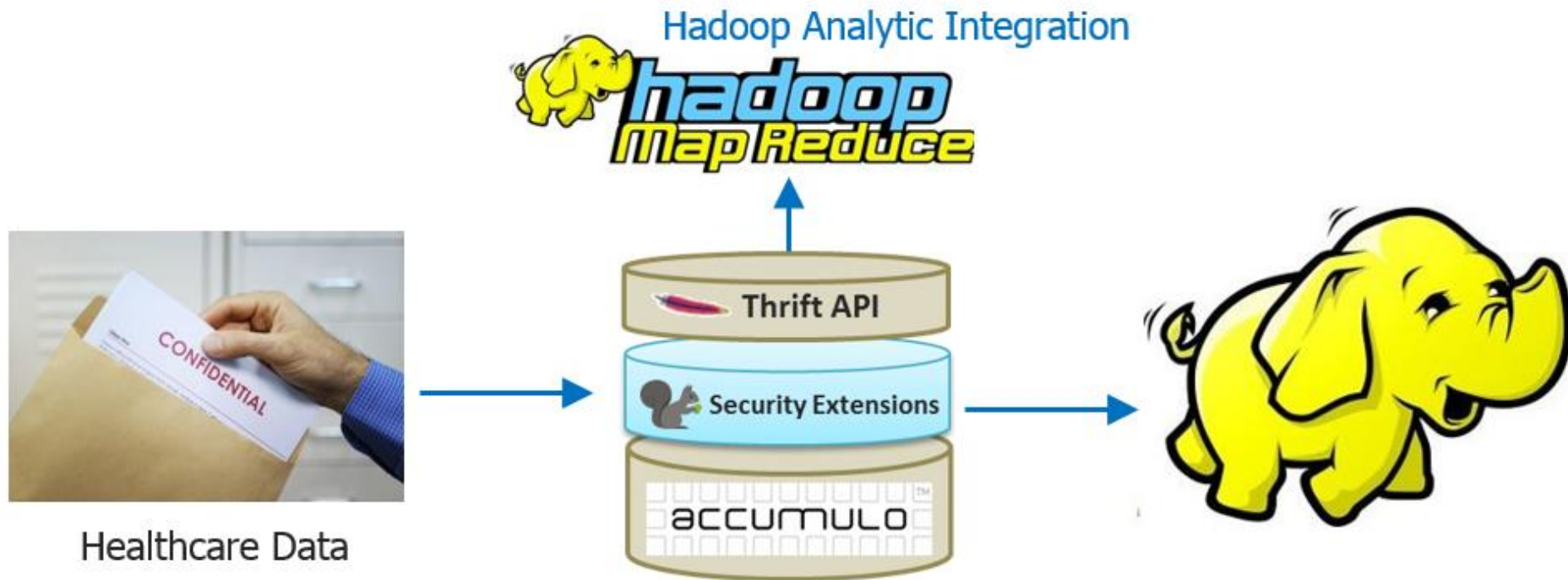
- Enterprises dealing with sensitive data are not able to move towards implementing Big data projects and Hadoop quickly.
- Encrypt the data while moving to Hadoop, then use it for further processing to get relevant insights.



"Example Health-care data used by Insurance companies to calculate premium"

When not to use Hadoop?

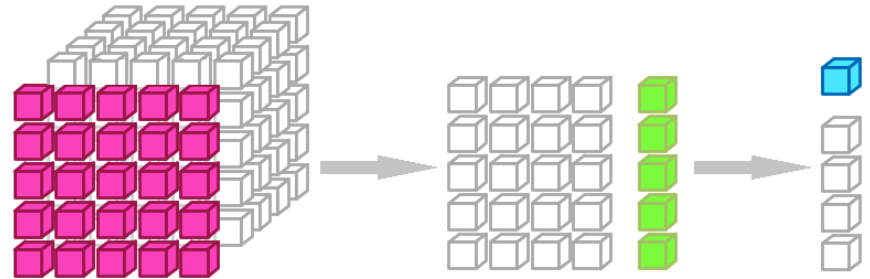
- Security is the primary concern
 - Alternative: use Apache Accumulo on top of Hadoop
 - Accumulo: sorted, distributed key/value store that provides robust, scalable data storage and retrieval; cell-based access control.



In summary, what is Hadoop not for?

- It is good for Big data but not for OLTP or OLAP/DSS.

Hadoop is not a replacement for existing RDBMS technology but complements them.



- Process transactions (random access)
- Process lots of small files
- Works that cannot be parallelized
- Low latency data access
- Intensive calculations with little data

When to use Hadoop?

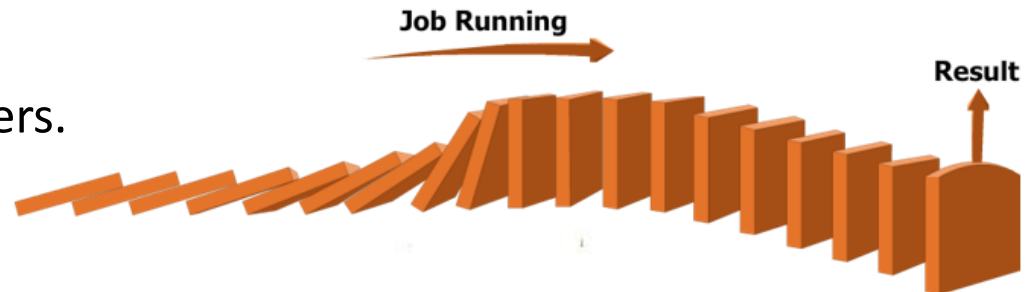
- Data size and data diversity

The data is huge in size, i.e., several terabytes and petabytes.



Data of different types: structured, semi-structured and unstructured.

You are not in a hurry for answers.



When to use Hadoop?

- Future planning

- Build a small or medium cluster for the data available at present and scale up the cluster in future.



When to use Hadoop?

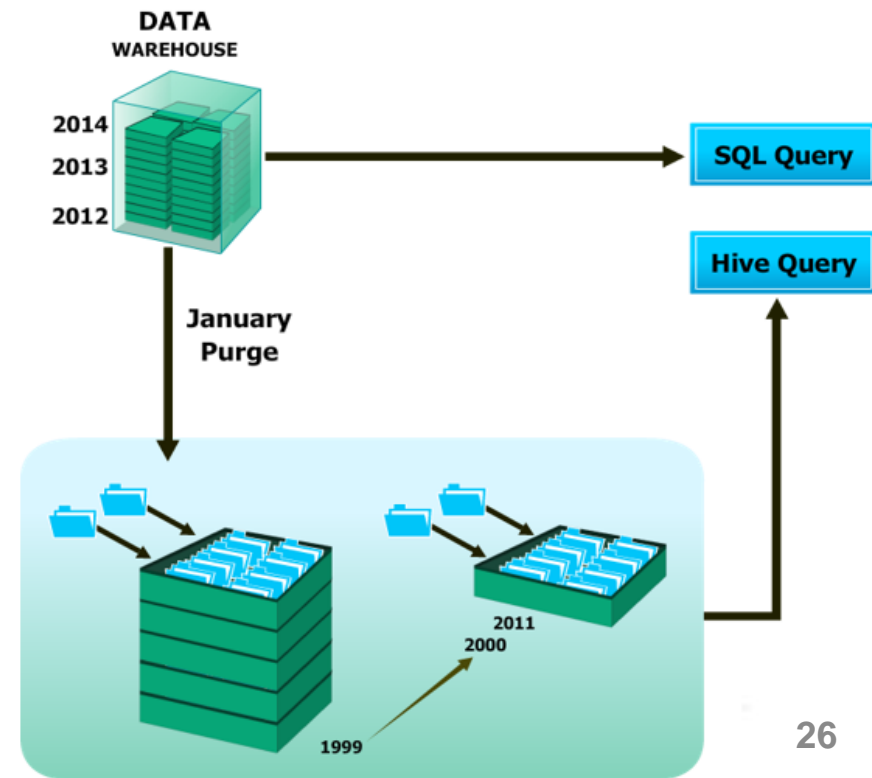
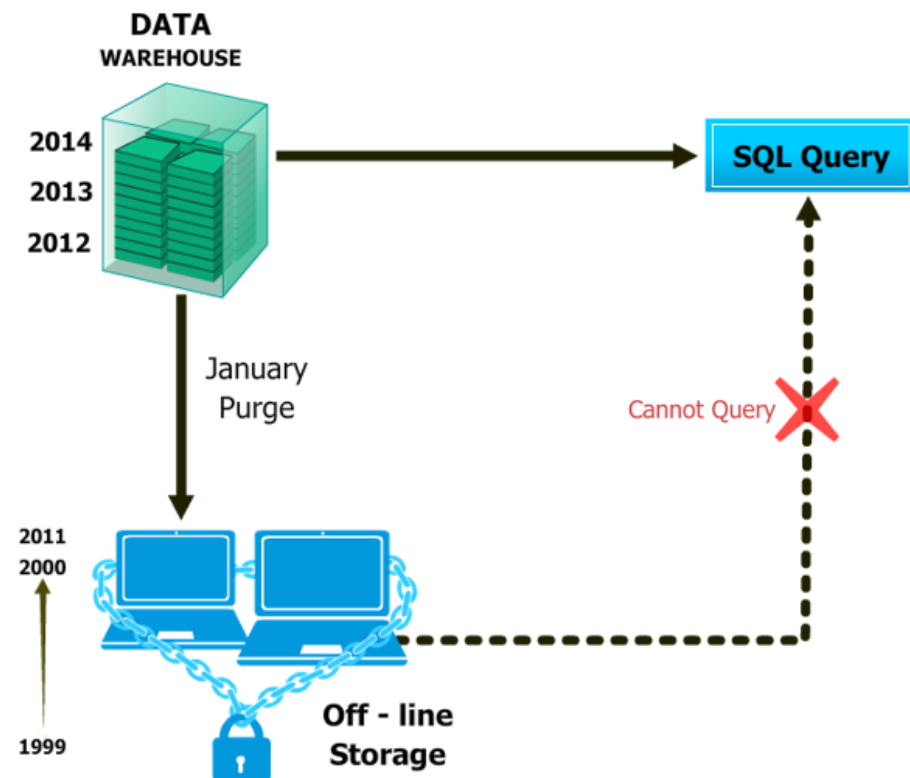
- Multiple frameworks for Big data
 - Can be integrated with multiple analytic tools to get the best out of it
- **Mahout** for Machine-Learning
- **R and Python** for analytics and visualization
- **Spark** for real time processing
- **MongoDB and Hbase** for NoSQL database
- **Pentaho** for BI, etc.



When to use Hadoop?

- Lifetime data availability

- Scalability: the stored data can be live and running forever, the cluster size can be increased unlimitedly by adding nodes to it.



Keys to successfully adopting Hadoop



Business users and analysts have access to as much data as possible

Regulatory requirements like data privacy must still be respected.

Results are accessible through standard tools in an organization

Hadoop developers should expose their logic so that results are easily consumed and reusable.



Governance requirements for the data stored in Hadoop

Data audit for both RDBMS and Hadoop are possible.

Keys to successfully adopting Hadoop



Should not try to find an open-ended problem

This kind of problem has neither clearly defined milestones nor measurable business value.

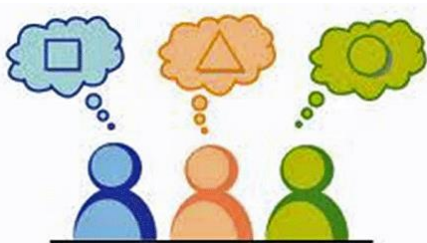
Working with business's leaders

Businesses want to see value from their IT investments, and with Hadoop it may come in a variety of ways.



Examine the perspectives of people and processes that are adopting Hadoop in the organization

Adopters make effort to support data science by fostering experimentation and data exploration





Hadoop case studies

Hadoop in action

- There have several enterprises implementing Hadoop.



Hadoop case studies in enterprise



10x
The Data

1/5
The Cost

**The
New York
Times**



4TB

in 1.5TB

24Hours

Hadoop case studies in enterprise

NETFLIX

Recommendation engines
and new content decisions
Hadoop, Hive and Pig
together with traditional BI

U.S. XPRESS

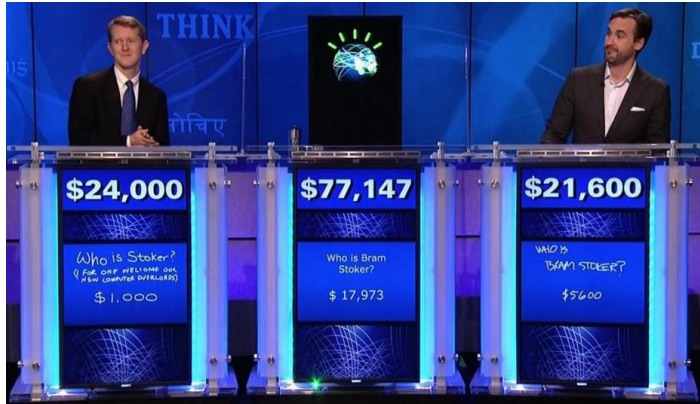
Collect and store 100s of data
points from thousands of
trucks, plus lots of geo data in
a Hadoop cluster



Product recommendations, in-store
shopping experience of customers
Hadoop and NoSQL technologies

More at the following [link](#)

Hadoop case studies in enterprise



About **200 million** text pages

Use Hadoop to distribute the workload for loading the information into memory



Recommender system of pins and boards using GCN

Hadoop2 cluster with 378 d2.8xlarge Amazon AWS nodes

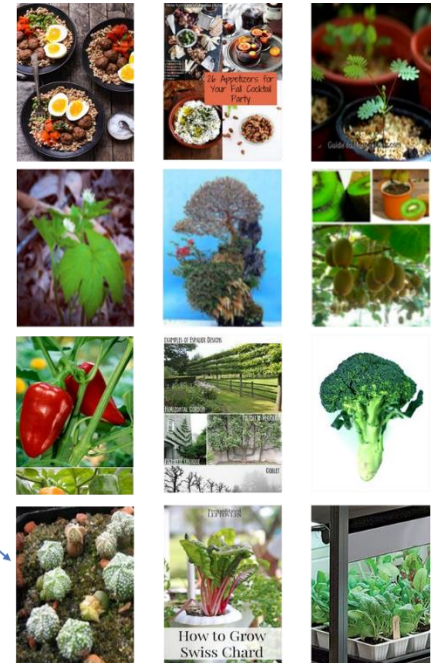


Visual

Annot.

Pixie

PinSage





Hadoop installation

The installation process



Installation modes

- Hadoop cluster can be set up in one of the following three supported modes:



Local (Standalone)
mode

Pseudo-distributed
mode

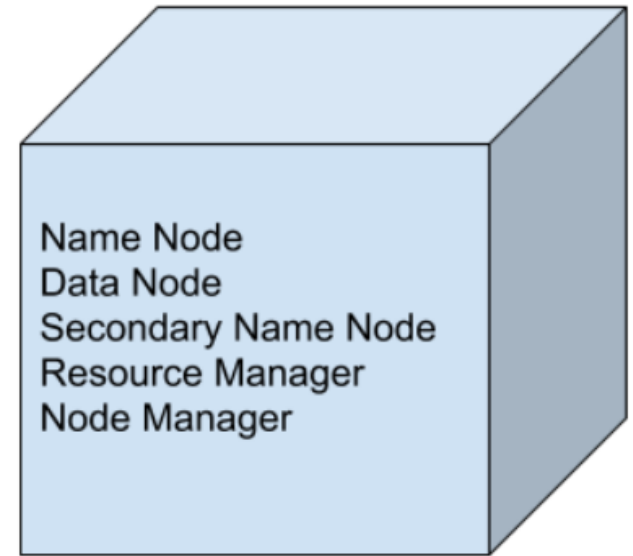
Fully-distributed
mode

Local (Standalone) mode

- Hadoop is **by default** configured to run **all the processes in a single JVM** (Java Virtual Machines).
- This is useful for learning, testing, and debugging.
- There is **no need to configure the xml files**—hdfs-site.xml, mapred-site.xml, core-site.xml for Hadoop environment.

Pseudo-distributed mode

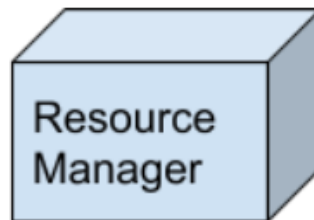
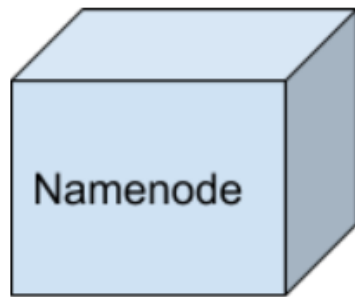
- Use only a single node to simulate the cluster.
- All the processes run independently on separate JVMs.



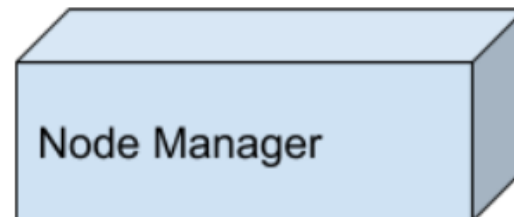
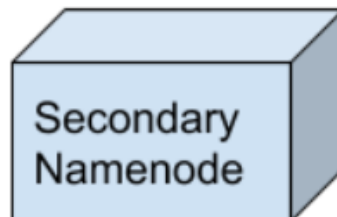
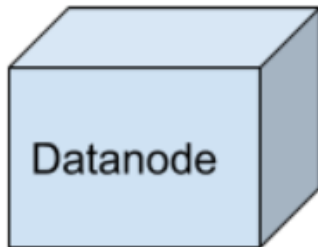
- This is both for development and debugging purposes.
- The configuration files must be specified properly.

Fully-distributed mode

- Hadoop runs on the cluster of machines, each of which plays the role of master daemon or slave daemons.
- The data is distributed across different nodes.



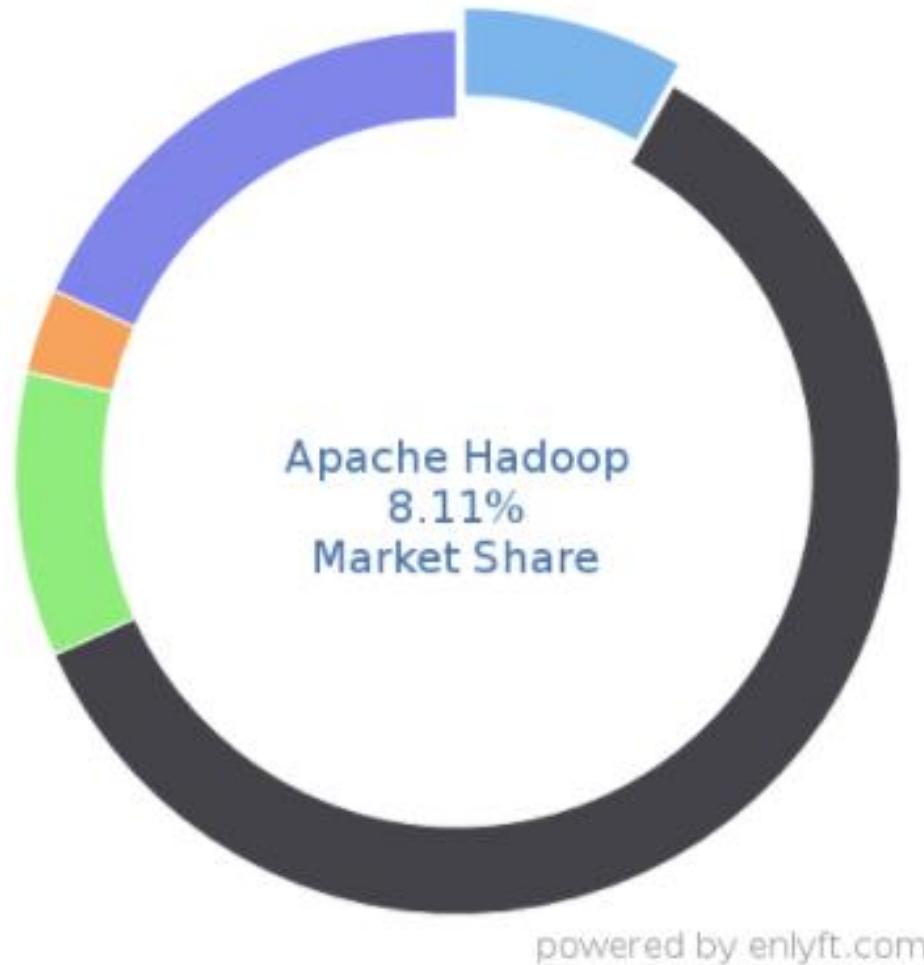
This is the production mode of Hadoop.





Hadoop: Now and Future

Big Data



Apache Hadoop (8.11%)

Snowplow (60.13%)

Informatica (10.41%)

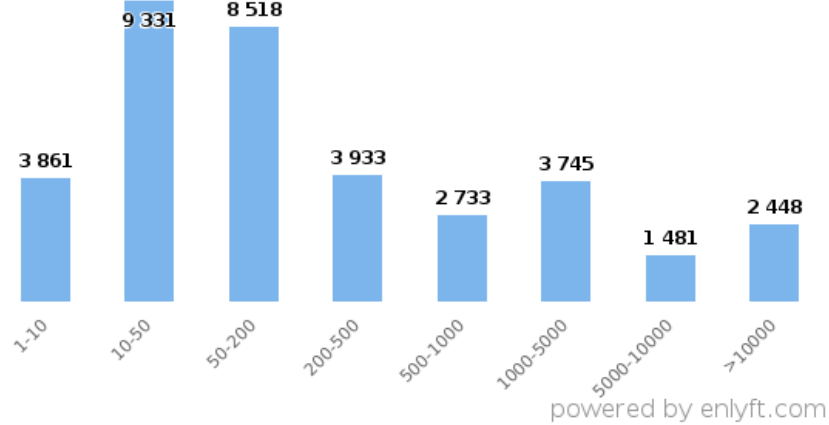
Teradata (3.10%)

[View other alternative products](#)

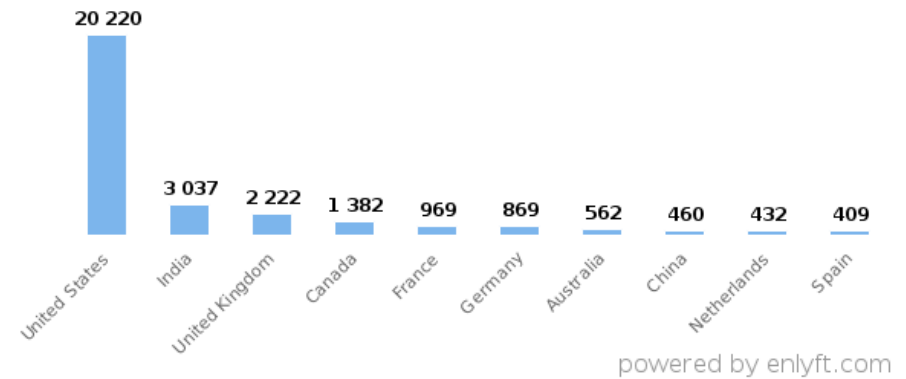
37,031

Companies using Apache Hadoop

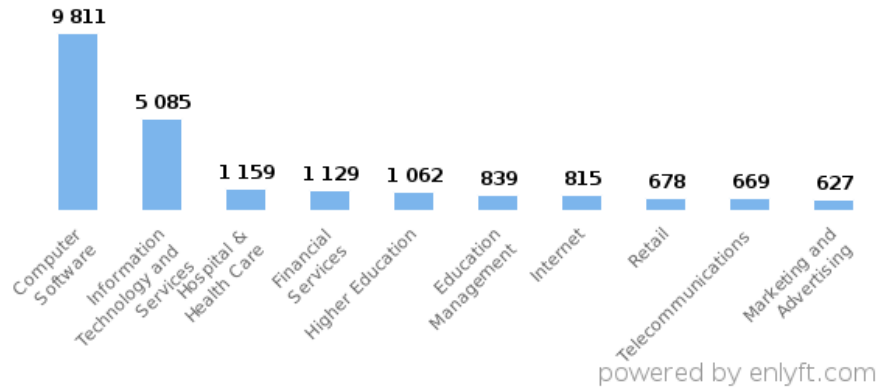
Distribution of companies using Apache Hadoop by Company Size



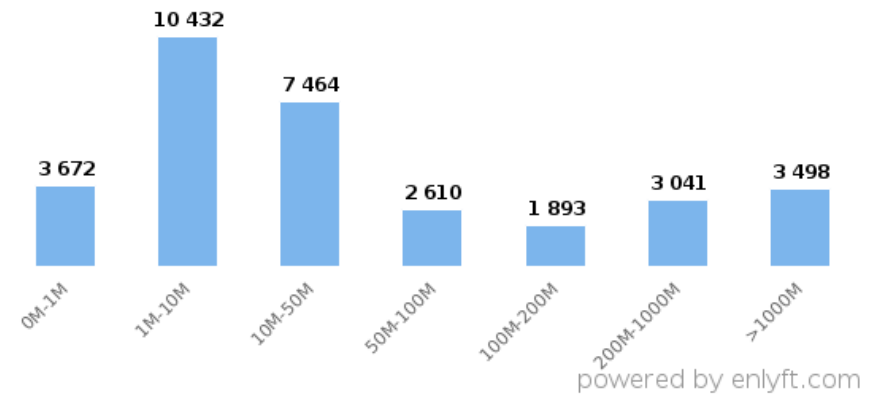
Distribution of companies using Apache Hadoop by Country



Distribution of companies using Apache Hadoop by Industry



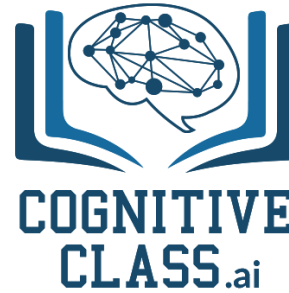
Distribution of companies using Apache Hadoop by Revenue



Top Hadoop Technology companies



Where to
learn
Hadoop?



edureka!



cloudera®
coursera

