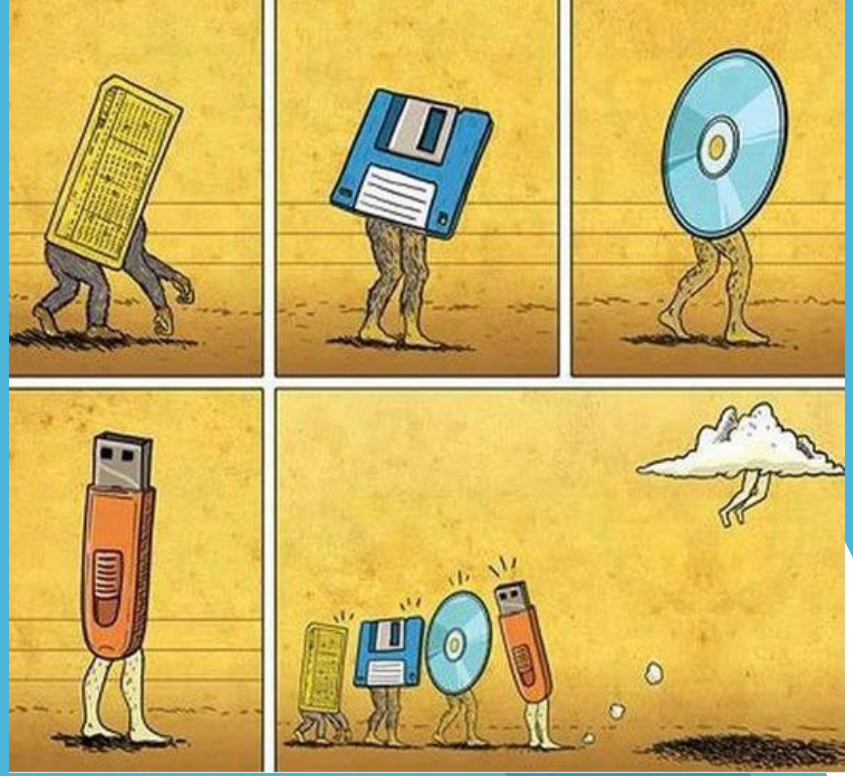
Introduction to Big Data Tools

HBD - 01/2019

Big Data Tools



Information

“Time is money. Information is power.”

- Unknown

“Data! data! data!"
he cried impatiently.
"I can't make bricks without clay.”

Arthur Conan Doyle, The Adventure of the Copper Beeches

Information
is only **useful**
when it can be
understood

Muriel Cooper
(co-founder of the
MIT Media Lab)

Redesign
by Dario
Michelone

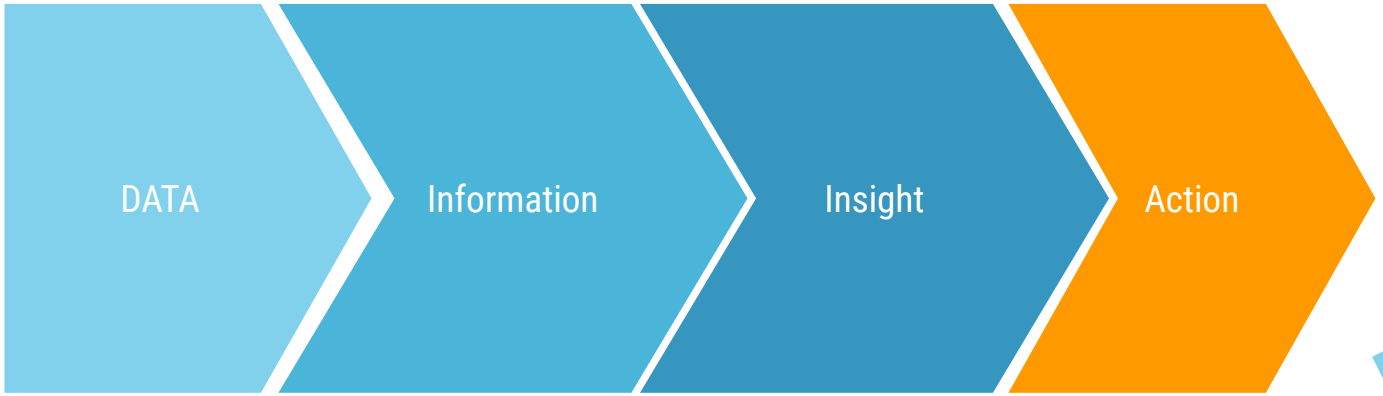
“The goal is to transform data into information and information into insight.”

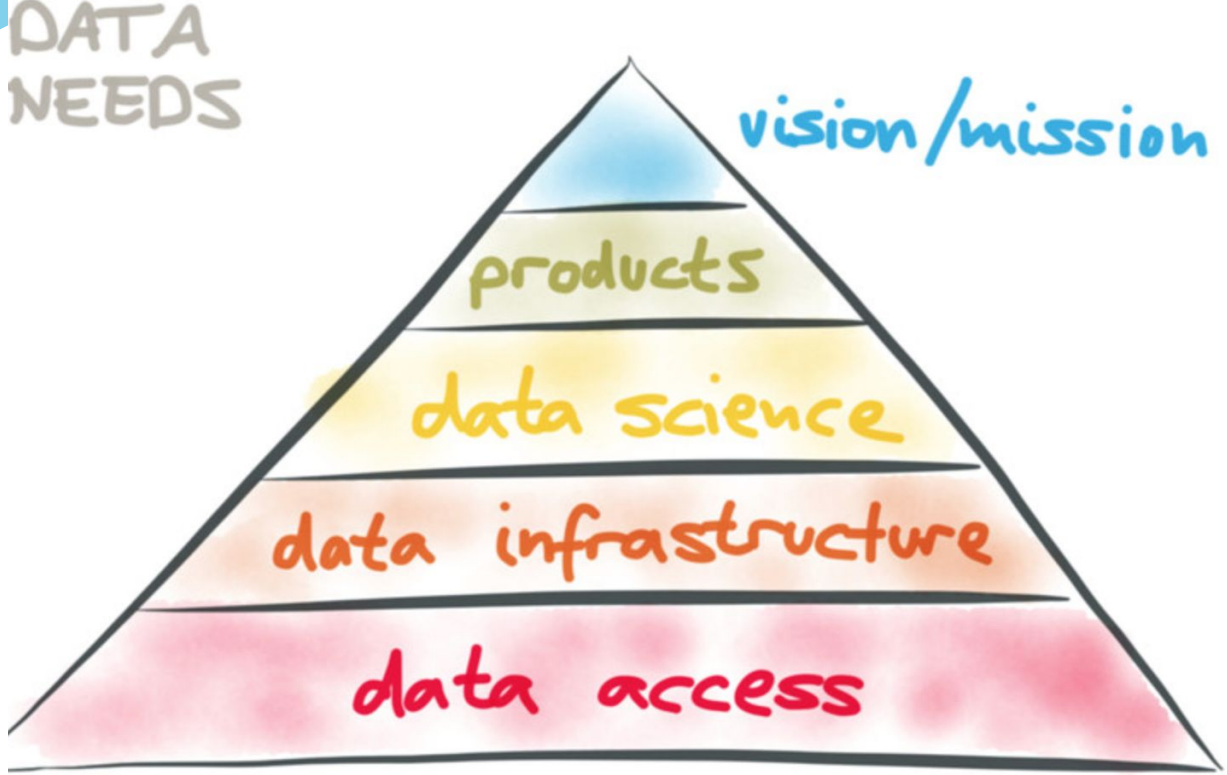
Carly Fiorina
(Ex CEO HP)

https://www.linkedin.com/pulse/8-memorable-quotes-importance-data-web-analytics-chris-bennington

Having insights allows us to form **informed** decisions and actions that should help on the progress of our organization.

Then our process should be:





https://martin.kleppmann.com/2015/05/06/data-agility-at-strata.html

How we transform Data into Information?

Acquisition - Challenge

The main challenge during acquisition is to be able to keep up with the amount of data the organization receives.

In the “pre big data era” the main source of the information for the majority of the organizations was the **organization’s own day to day operations**.

This mean that the acquisition challenge was low and hidden in the “processing challenge”.

Process - What we do?

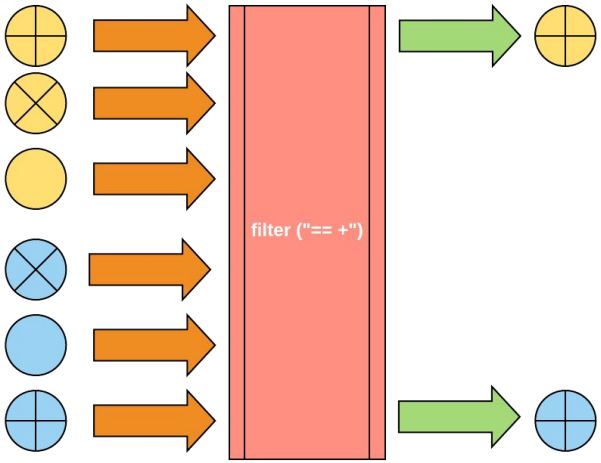
As explained before we need to **transform** the **data** into useful **information**.

This transformation is made by applying a pipeline of operations to our incoming data, to obtain the final result.

Process - Filter

Filter tests each element of the incoming data against a **conditional operation (or predicate)**.

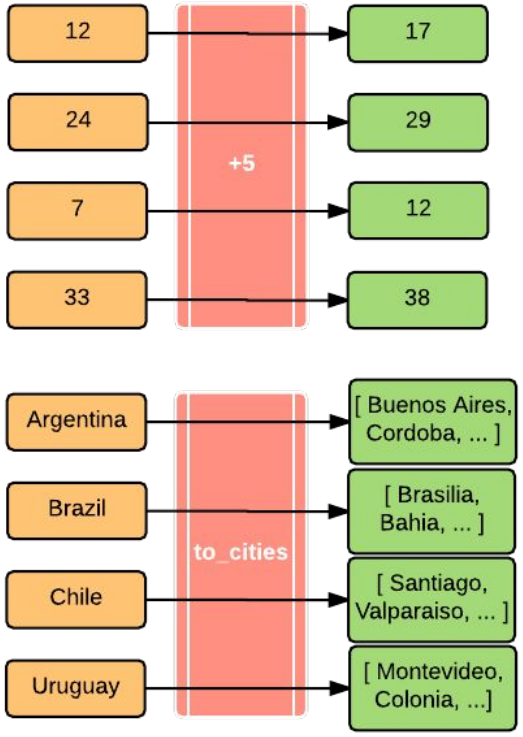
The output data contains only the elements that pass the predicate test (predicate returns true).



Process - Map

Map works by applying a **function (transformation operation)** to each element in the incoming data to obtain each element of the output data.

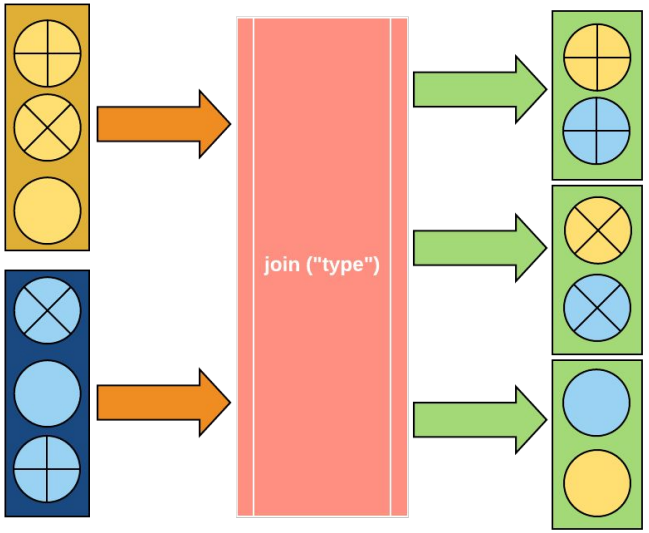
The output element could be a list.



Process - Join

Join combines two sets of input data by creating one element with the combination of one element of each input.

Generally uses a common value between both type of elements to join

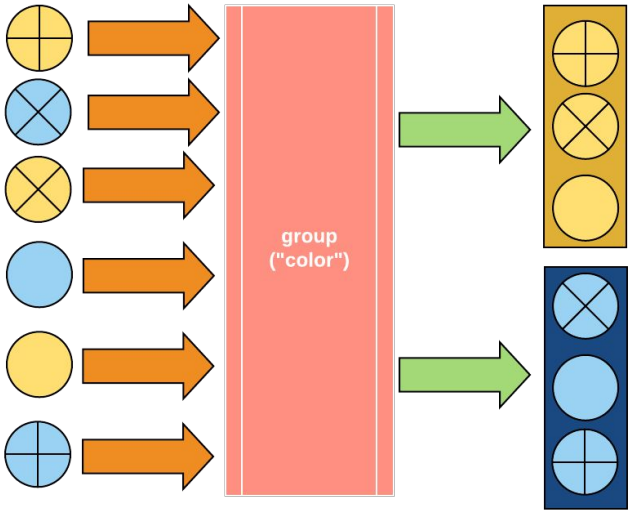


Process - Group

Creates groups of the elements of the incoming data.

The groups are formed by applying an operation that return a value used as key for the group.

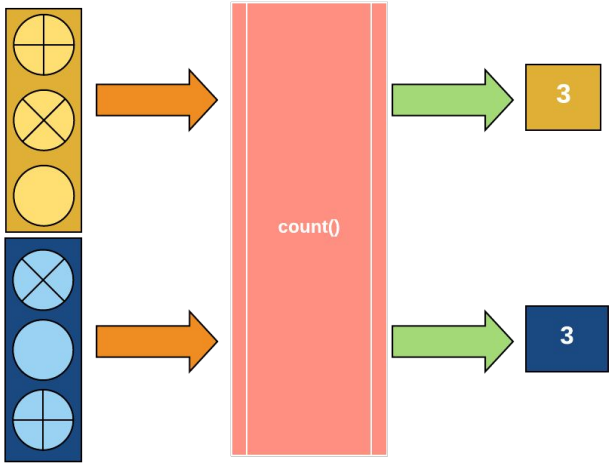
All the elements that return the same key go to the same group.



Process - Aggregate

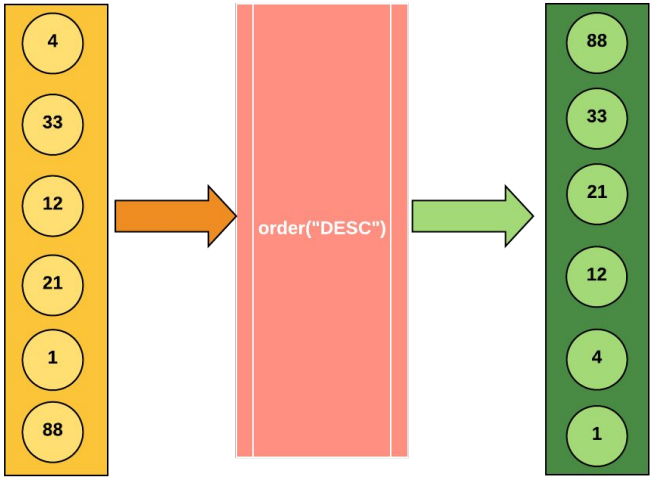
Aggregation Operations are functions applied to a group of related elements of the input data (usually obtained by a group).

Usually transforming a list of element into a value



Process - Order

Order Operations sort the elements of the input data using an ordering criteria.



Process - Using SQL as pipeline

```
SELECT u.id, u.name, sum(i.price) price
FROM users u
    INNER JOIN invoice i on i.user_id=u.id
WHERE
    u.id < 1000
GROUP BY u.id, u.name
HAVING  sum(i.price) < 100
ORDER BY sum(i.price) DESC
```

- Map
- Filter
- Join
- Group
- Aggregation
- Order

Store - Challenges

The necessity is to have the **“correct”** information on request.

In the “pre big data era” this would be have always the last information as soon as the event was produced.

This is why the RDBMS and SQL became necessary as they would provide:

- Schema optimizations: improves storage
- ACID transactions: guaranties “correct” data.
- Indexes: improves search and retrieve.

Show - Challenges

As we talked before on showing the information the challenges are:

- Respond fairly quickly or at least give some "loading feedback".
- Make the information easy to read
- Make the usability as simple as possible.

Evolution of data systems: before 2006

OLAP and Data Warehouses:

- **Inflexible batch processes**.
- **Batch Pipeline**: Extract-Transform-Load (**ETL**)
- **Batch Result:** would be **bulk-loaded** into a Data Warehouse.
- **Batch Reprocess:** as ETLs have daily schedule they could demand **weeks** (or months) for **re-processing**.
- **Proprietary software:** (vendor lock-in) and structured data.
- **Vertical scaling:** Only way to scale is **“bigger machines”**.

Evolution of data systems: 2006-2009

- Multiple Parallel-Processing (MPP) databases
  - Brought **scalability and speed** to the Data Warehouse.
  - **MPP vendors** such as Teradata, Greenplum, Netezza, and Vertica **rose to dominance** and former industry leaders responded with their own solutions such as Oracle's Exadata.
  - Brought **columnar format**s and **multi core parallel** processing.
- **Semi-structured** data (XML, logs, JSON) started appearing.
- First **NoSQL** databases like MongoDB.
- **Size** and pressure of wanting to process **semi-structured** and unstructured data started **putting a strain on ETL processes**.
- **IaaS services:** Help to deploy and maintain the organization's hardware (Amazon AWS, Rackspace, etc).

Big Data

Big Data


CONSULTANTS SAY THREE QUINTILLION BYTES OF DATA ARE CREATED EVERY DAY.
DilbertCartoonist@gmail.com
IT COMES FROM EVERYWHERE. IT KNOWS ALL.
ACCORDING TO THE BOOK OF WIKIPEDIA, ITS NAME IS "BIG DATA."
BIG DATA
BIG DATA LIVES IN THE CLOUD. IT KNOWS WHAT WE DO.
IN THE PAST, OUR COMPANY DID MANY EVIL THINGS.
BUT IF WE ACCEPT BIG DATA IN OUR SERVERS, WE WILL BE SAVED FROM BANKRUPTCY.
LET US PAY.
www.dilbert.com
1-29-12
IS IT TOO LATE TO SIDE WITH EVIL?
SHHHH! IT HEARS YOU.

What is Big Data?

Jon Bruner, Editor-at-Large, O'Reilly Media

"Big Data is the result of collecting information at its most granular level — it's what you get when you instrument a system and keep all of the data that your instrumentation is able to gather."

Ryan Swanstrom, Data Science Blogger, Data Science 101

"Big data used to mean data that a single machine was unable to handle. Now big data has become a buzzword to mean anything related to data analytics or visualization."

http://datascience.berkeley.edu/what-is-big-data/

What is Big Data?

In 1997 NASA researchers Michael Cox and David Ellsworth first mention “the problem of big data”:

“Visualization provides an interesting challenge for computer systems: data sets are generally quite large, taxing the capacities of main memory, local disk, and even remote disk. We call this the problem of big data. When data sets do not fit in main memory (in core), or when they do not fit even on local disk, the most common solution is to acquire more resources.”

What is Big Data?

In 2001 Gartner's analyst Doug Laney coined the **3 Vs definition**.

Volume

The amount of data growing at geometric rate.

Velocity

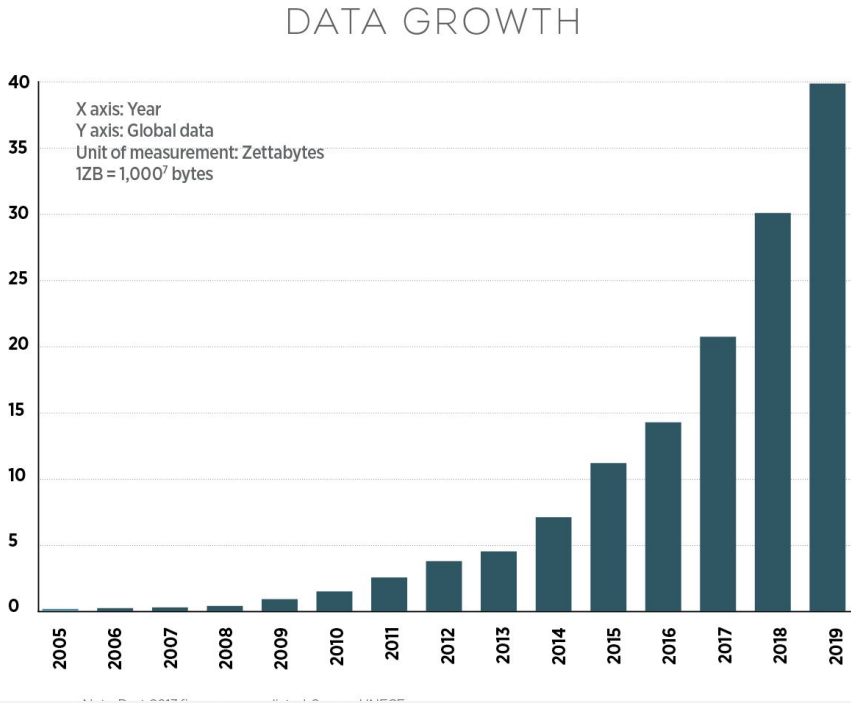
the speed of the generation of data is increasing

Variety

new types of data like semi-structured and unstructured are being stored and analyzed.

Volume

- **2003**: 5 exabytes generated.
- **2004**: first social networks consolidate (Facebook).
- **2006**: 161 exabytes generated. IDC predicts doubling of data every 18 months (= Moore's Law)
- **2007**: first iPhone. Public cloud infrastructure services becoming mainstream.
- **2013-2014**: More data was created than in the entire previous history of the human race.
- **2015:** In August over 1 billion people used Facebook FB -0.09% in a single day.



Internet + Social Media + Internet of Things + Mobile => the data explosion continues

Velocity

- Facebook users sent on average 31.25 million messages and view 2.77 million videos every minute.
- Every minute up to 300 hours of video are uploaded to YouTube alone.
- In 2015, 1 trillion photos werestaggering taken and billions of them will be shared online. By 2017, nearly 80% of photos will be taken on smart phones.
- There are 500 million Tweets sent each day. **That's 6,000 Tweets every second**.
- During the 2014 FIFA World Cup Final, **618,725 tweets were sent in a single minute**.

Variety

Before most of the digital information to be processed would come in a structured way, with a schema

Structured

The information stored in RDBMS with schema, types and indexes.

Semi-structured

Text-based information that have a “schema” or it can be inferred from the data

Un-structured

Information that does not have any schema apparent.

Could be free text like mails, documents; or binary information like pdf, word, documents images, sounds, videos, etc.

3Vs - New Sources

In this era the sources of the data has expanded adding two new “external” type of source:

Organizational data

- Stocks.
- Purchases & Sells.
- Transactions
- Users..

Machine

- Sensor data.
- Satellite data
- Logs
- Internet of things.
  - sport band.
  - intelligent homes.

People

- Social media.
- Pictures
- Videos

3Vs - New Sources

People and **Machine** data are considered "*new sources*", because organizations started to use this information and join it to their own to generate more wisdom and insight.

This two sources are the ones that are generated at fast pace and volume, and specially with variety (3Vs mentioned)

FitBit can produce several gigabytes a day

3Vs - Challenges

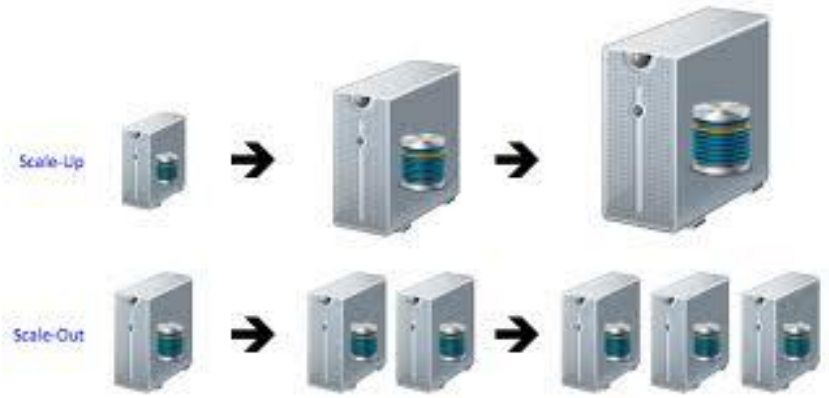
The challenges previous mentioned get exacerbated due to this new attributes of the data so we need to improve each stage of our pipeline

- **Acquisition:** We need to keep up we the amount and speed and probably buffer to prevent
- **Process:** Again try to process as fast as possible to be able to provide current information that is possible.
- **Store:** The volume and variety require new strategies to store, write, and read the data.
- **Show:** New ways to show the data but also be wary of the amount of data sent

We need to Scale

¿But up?

Maybe we can use tools that provide the scaling seamlessly and other features.



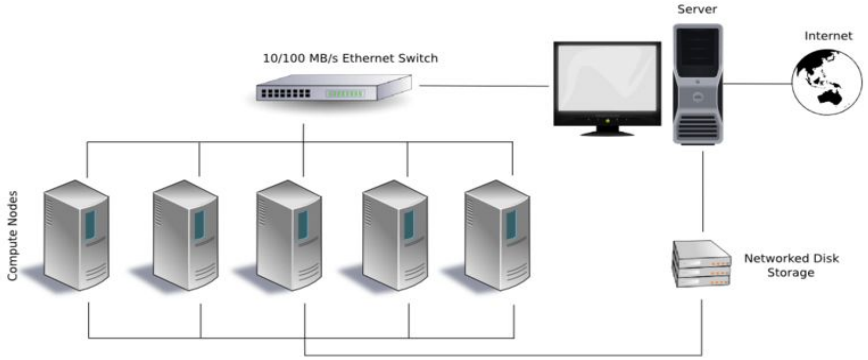
The Key is to scale out!

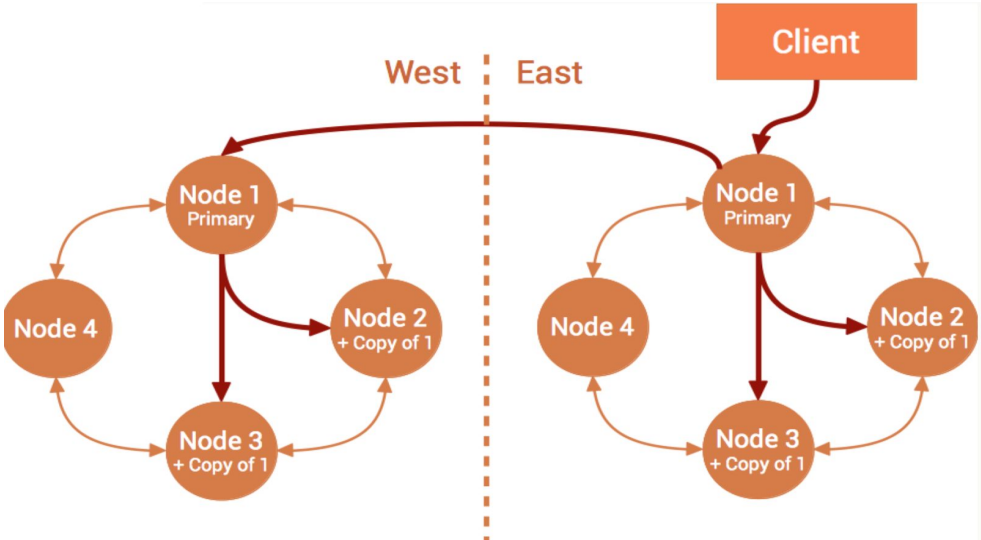
also known as **horizontal scale**.

Horizontal scaling

Is to have **several** computers (or **nodes**) in a **cluster** working together as if they were one (at least for the application that use them).

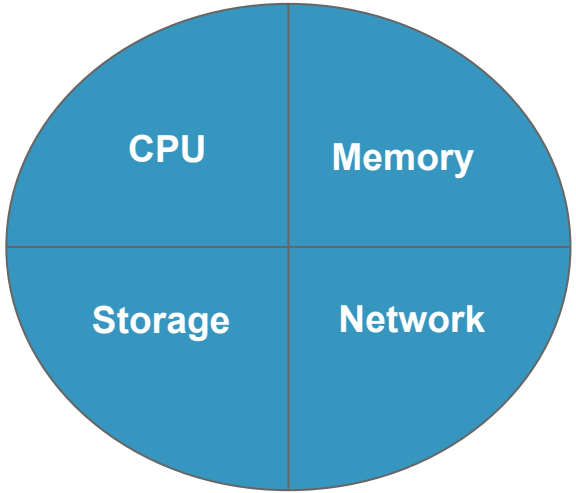
This allow to divide a big task into smaller tasks that can be manageable by one of the nodes of the cluster





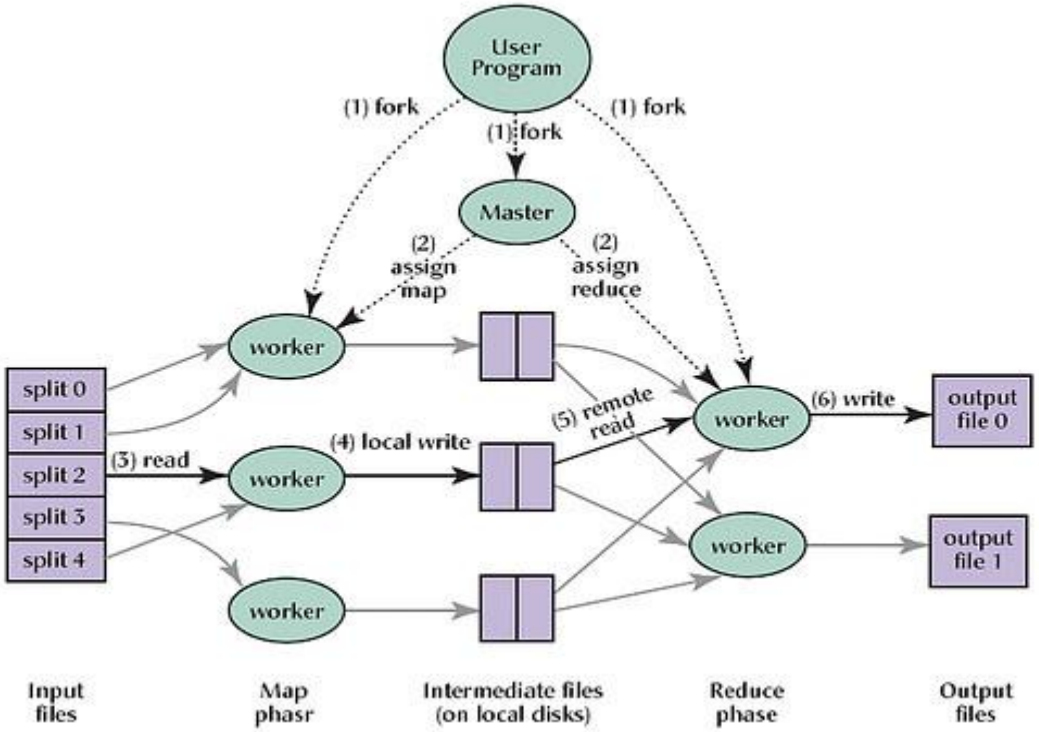
Horizontal scaling - What we scale?

Computers have 4 main resources:



We want to scale:

- **Processing** capability: CPU + Memory
- **Storage** capability: Storage + Memory



Map Reduce distributed processing architecture

Horizontal scaling - Challenges

Before even thinking about your particular problem you would need to provide:

Cluster management:

- Node health.
- Resources management.
- Task distribution and control.
- Data replication.
- High availability.

Error management:

- Network error
- Node error
- Task error
- Data loss

This means the programmer needs only to think on:

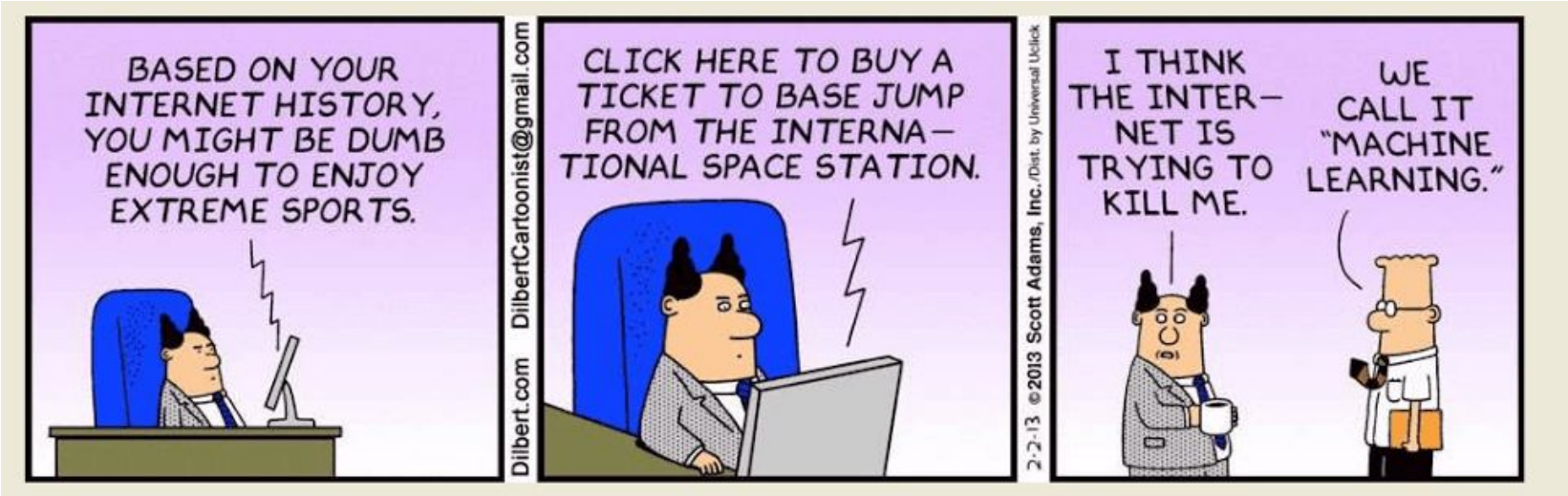
- how make the original problem divisible (with some help or guidance of the tool)
- Potential performance issues (https://gist.github.com/hellerbarde/2843375).

Big Data Tools

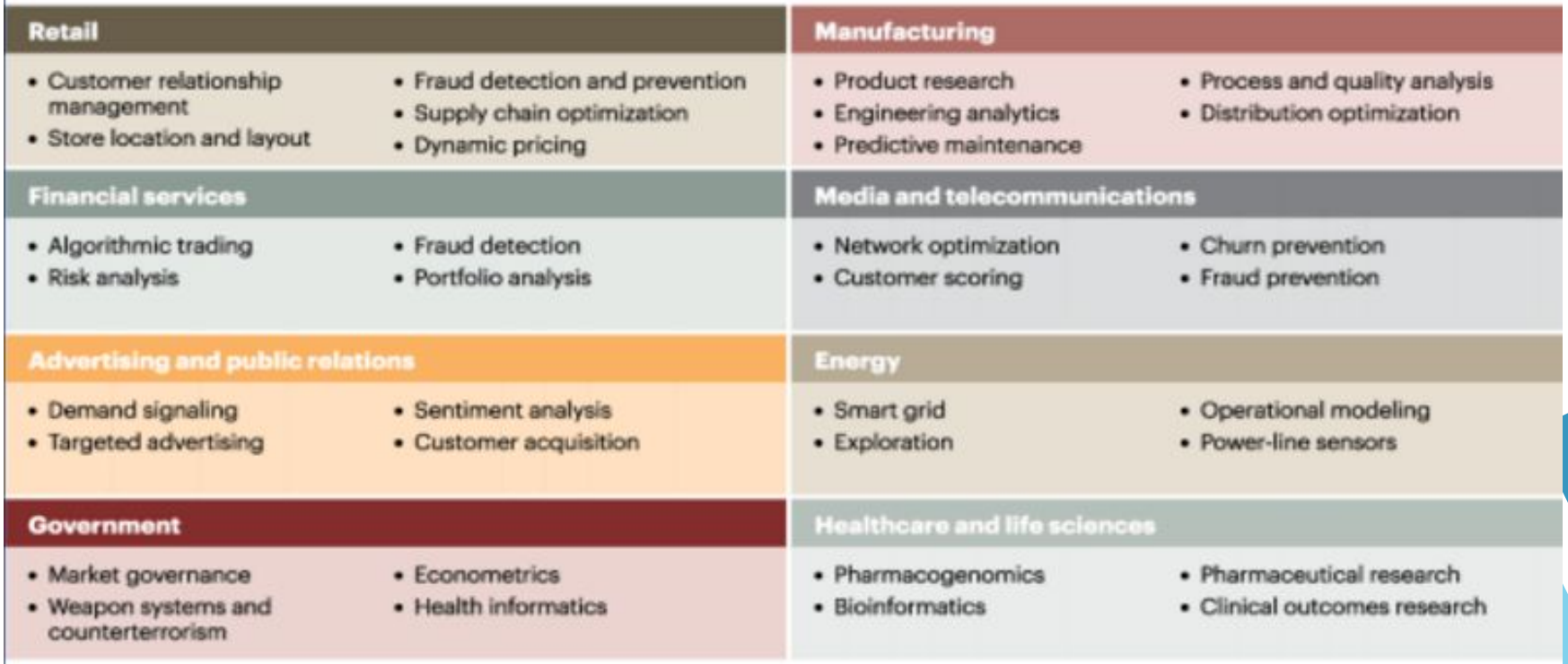
What are big data tools?

Software (applications, frameworks, SaaS and FaaS) that simplifies working with Big Data

What are Big Data tools used for?



What are Big Data tools used for?



Retail

- Customer relationship management
- Store location and layout
- Fraud detection and prevention
- Supply chain optimization
- Dynamic pricing

Manufacturing

- Product research
- Engineering analytics
- Predictive maintenance
- Process and quality analysis
- Distribution optimization

Financial services

- Algorithmic trading
- Risk analysis
- Fraud detection
- Portfolio analysis

Media and telecommunications

- Network optimization
- Customer scoring
- Churn prevention
- Fraud prevention

Advertising and public relations

- Demand signaling
- Targeted advertising
- Sentiment analysis
- Customer acquisition

Energy

- Smart grid
- Exploration
- Operational modeling
- Power-line sensors

Government

- Market governance
- Weapon systems and counterterrorism
- Econometrics
- Health informatics

Healthcare and life sciences

- Pharmacogenomics
- Bioinformatics
- Pharmaceutical research
- Clinical outcomes research

Source: A.T. Kearney analysis

What are Big Data tools good for?

- **historical data analysis**: Analyze and store data in the most granular level.
- Enable processing enormous amount of **data in real-time** (internet of things, fraud detection, real time analysis).
- Effectively analyze and store **semi-structured or unstructured data** (log forensics, text mining, image processing).
- Treat hardware as commodity and **scale horizontally** (lower costs, data lake)

Finally!
The Big Data tools

There are many tools with many objectives

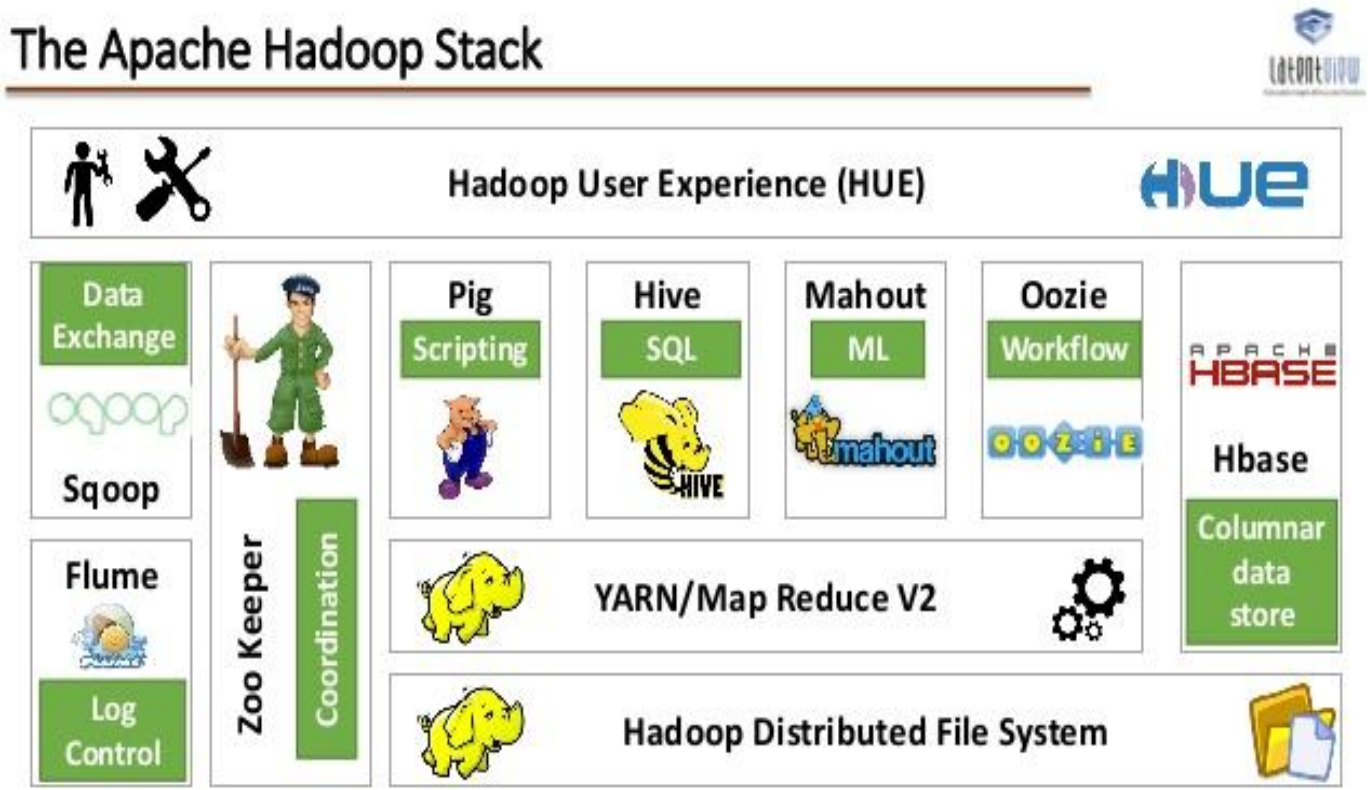
The Big Data technology stack is changing rapidly

Evolution of data systems: 2010-2012

Big Data tools start to be used at Enterprise level

- **Hadoop** became the place where companies could cheaply store and process any type of information.
- Hadoop became the **Enterprise Data Lake:**
  - A single scalable infrastructure for **storing and processing raw data**.
  - Data is **schemaless or schema on read** is used.
- **Hadoop** was being used for **ETL** and data was later loaded to **MPP databases**.

They can be used in as part of a **stack** to provide multiple functionalities to their users.



Evolution of data systems: 2010-2012

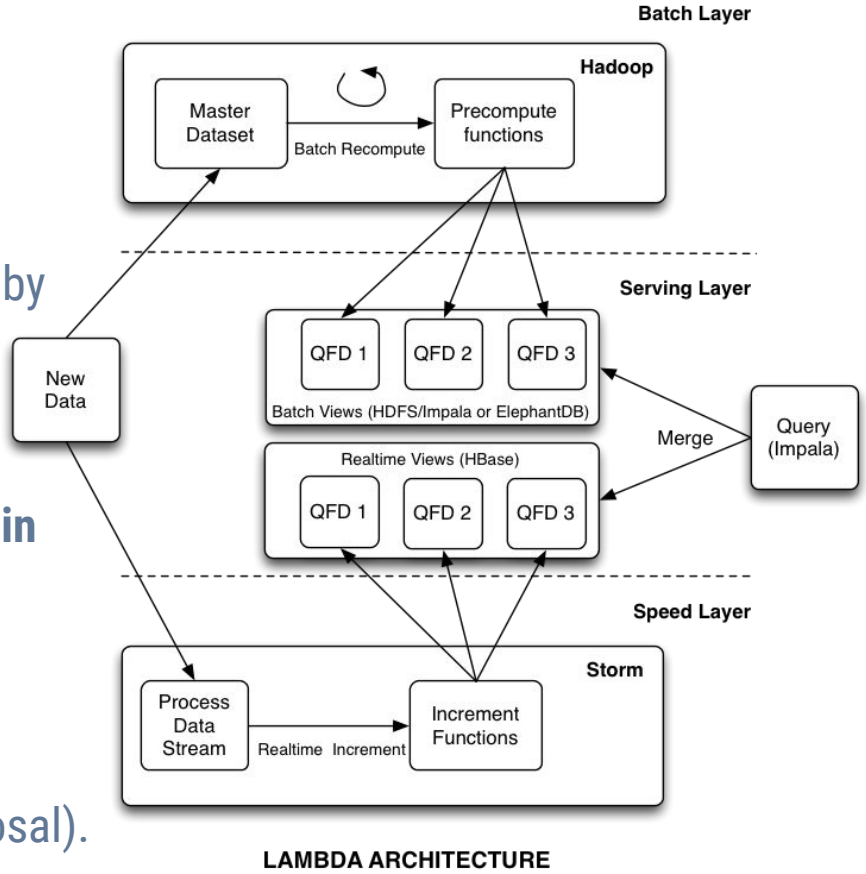
- Business pressure for quicker data loading into MPP databases.
- **Specialized connectors** implemented but lost data locality.
- A requirement for **continuous data processing** (**stream processing**) begins to appear.
- **PaaS** providers allow customers to develop, run, and manage applications without the complexity of building and maintaining the infrastructure typically associated with developing and launching an app (Amazon AWS, Heroku, Google App engine, etc.).

Evolution of data systems: 2012-2014

Rise of the **Lambda Architecture** (concept created by Nathan Marz).

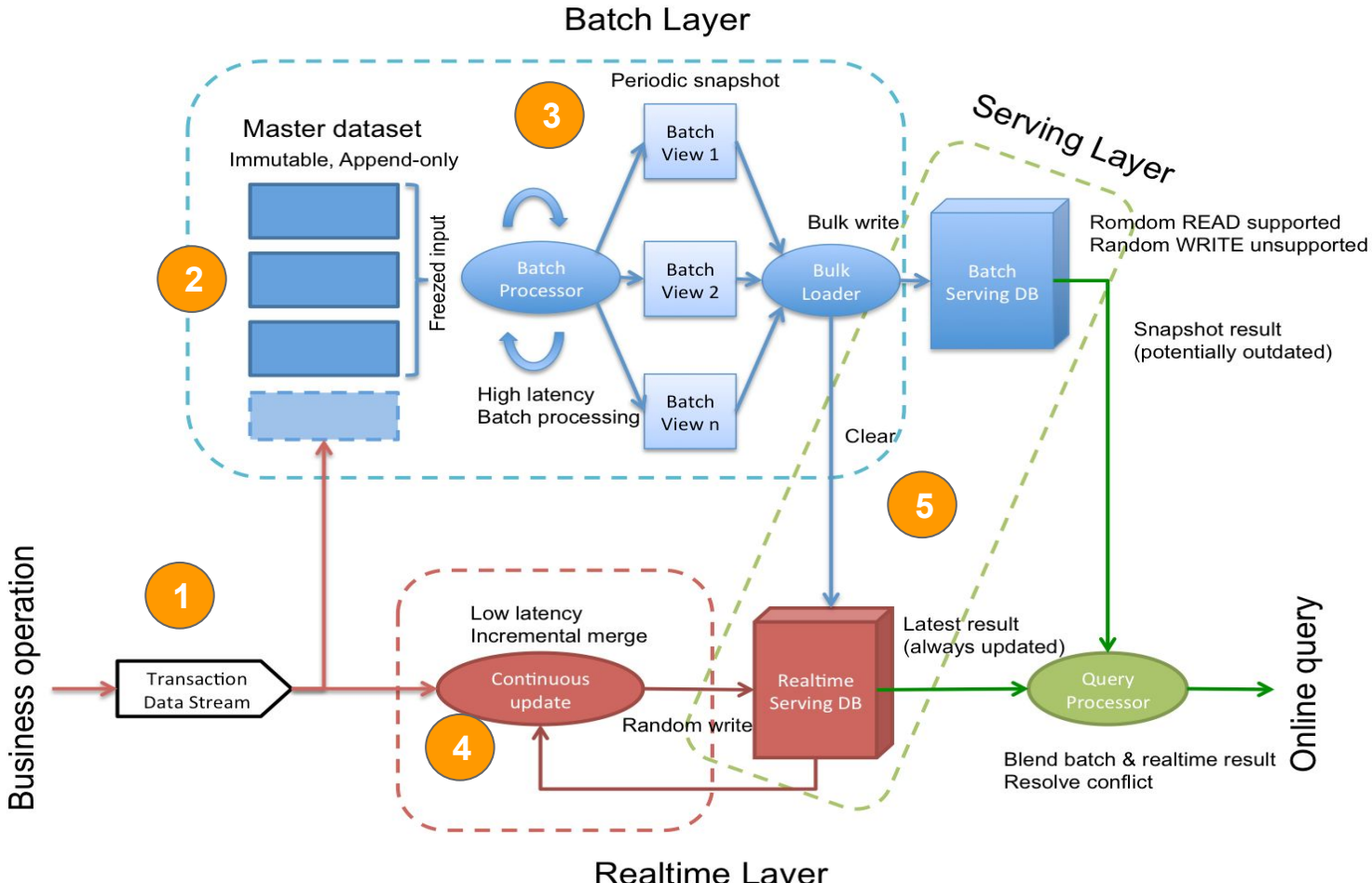
- Fast general processing architecture that combines **batch and stream computation**.
- Enables to ingest and process **large volumes in data in real-time**.
- Usually implemented with:
  - Hadoop for batch processing.
  - Storm for stream processing.

  (As recommended by Nathan Marz proposal).



LAMBDA ARCHITECTURE

Lambda Architecture



Lambda Architecture

1. **Input Buffer:** to support the velocity and volume, generally using an Advanced Message Queuing (RabbitMQ, ActiveMQ) or logs system (Kafka).
2. **Master Dataset:** Immutable set of the input data, append only, contains the full history, known as the "**source of truth**" as allows historical reprocessing.
3. **Batch Layer:** Periodically (hourly?) run through the dataset and recalculate the values from the beginning of time up to that point in time.
4. **Real Time Layer:** Continuously calculates the values since the last batch run till "now", to give "real time feel".
5. **Serving Layer:** Contains the precalculated values by the layers to respond to the query processor.

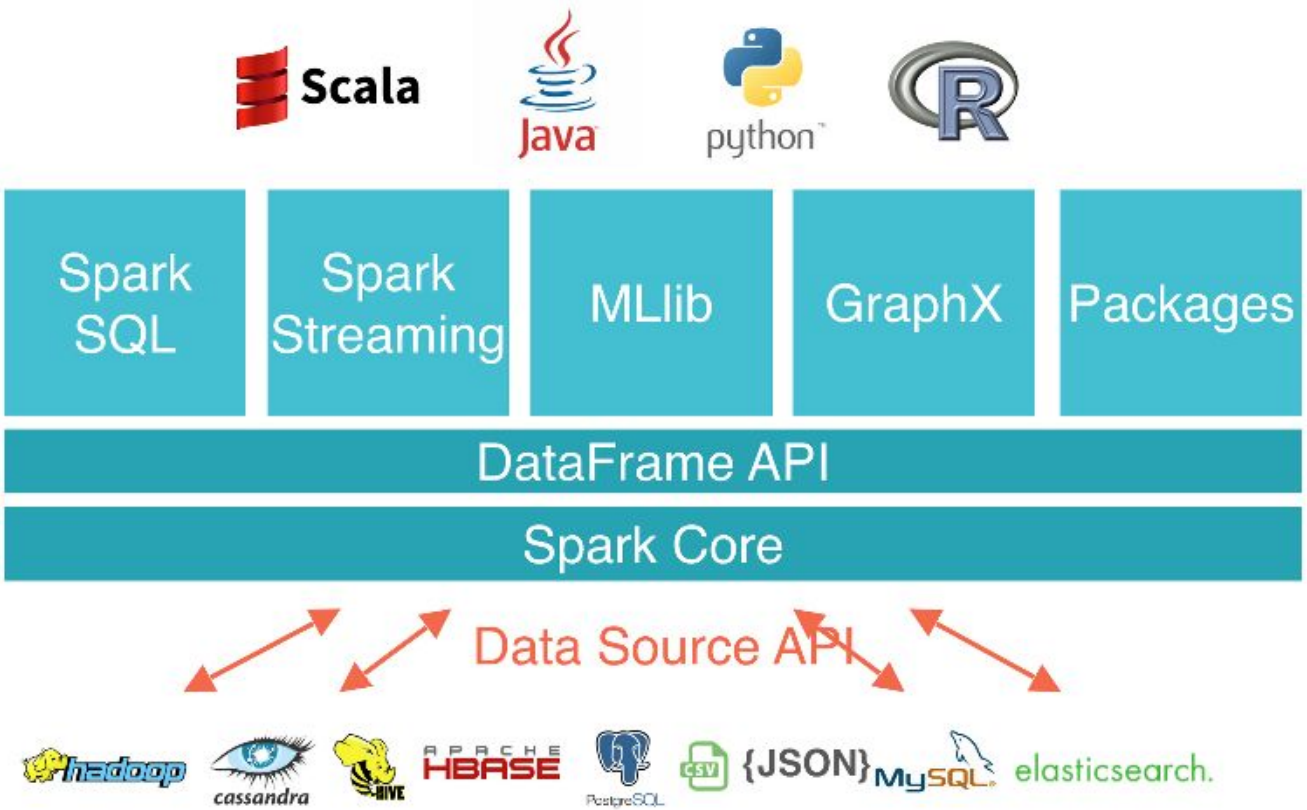
Evolution of data systems: 2012-2014

- Lambda Architecture is criticized because **of duplication of business logic**.
- **Cloudera** consolidates as the **leading Hadoop** vendor followed by **Hortonworks** and later **MapR**.
- Traditional enterprises like **IBM**, **EMC**, **Teradata** start selling their own distributions but gain little traction.
- **LinkedIn** (Kafka and Samza) and **Twitter** (Storm) continue to promote the development of **strong open source stream processing frameworks**.
- **SaaS providers:** like Google BigQuery allow upload and run queries without thinking on deploy and process infraestructures on the clooud

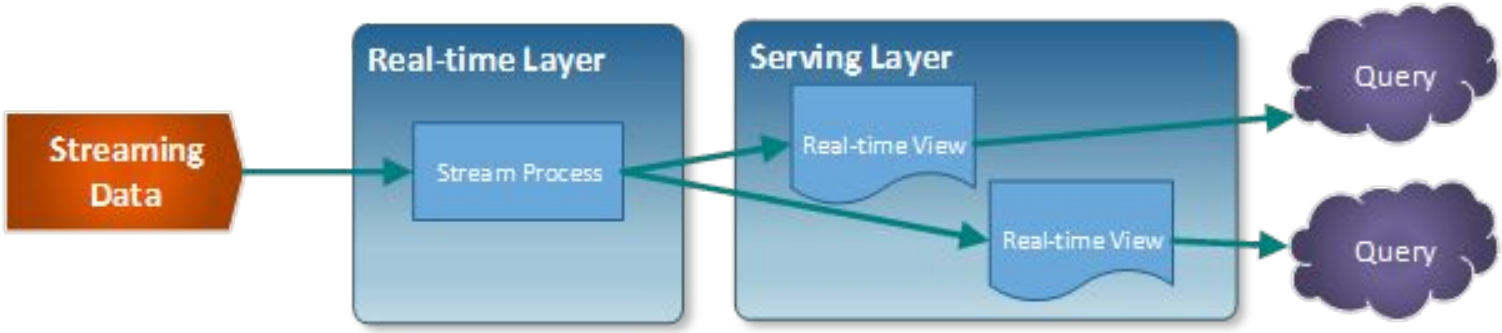
Evolution of data systems: 2015-2017

- **Apache Spark** seems to dominate the data engineering landscape
  - Unified API and infrastructure for batch and stream processing.
  - Expressive language.
  - Runs data processing as DAG in-memory or on disk.
- **Stream and real-time** processing becoming a fundamental part of any data processing system.
- **Apache Kafka** becoming the de-facto standard for fault tolerant, scalable stream management.
- **In-memory** solutions becoming the norm.

Spark Stack


Scala
Java
python
R
Spark SQL
Spark Streaming
MLlib
GraphX
Packages
DataFrame API
Spark Core
Data Source API
hadoop
cassandra
HIVE
APACHE HBASE
PostgreSQL
CSV
{JSON}
MySQL
elasticsearch.

Kappa Architecture

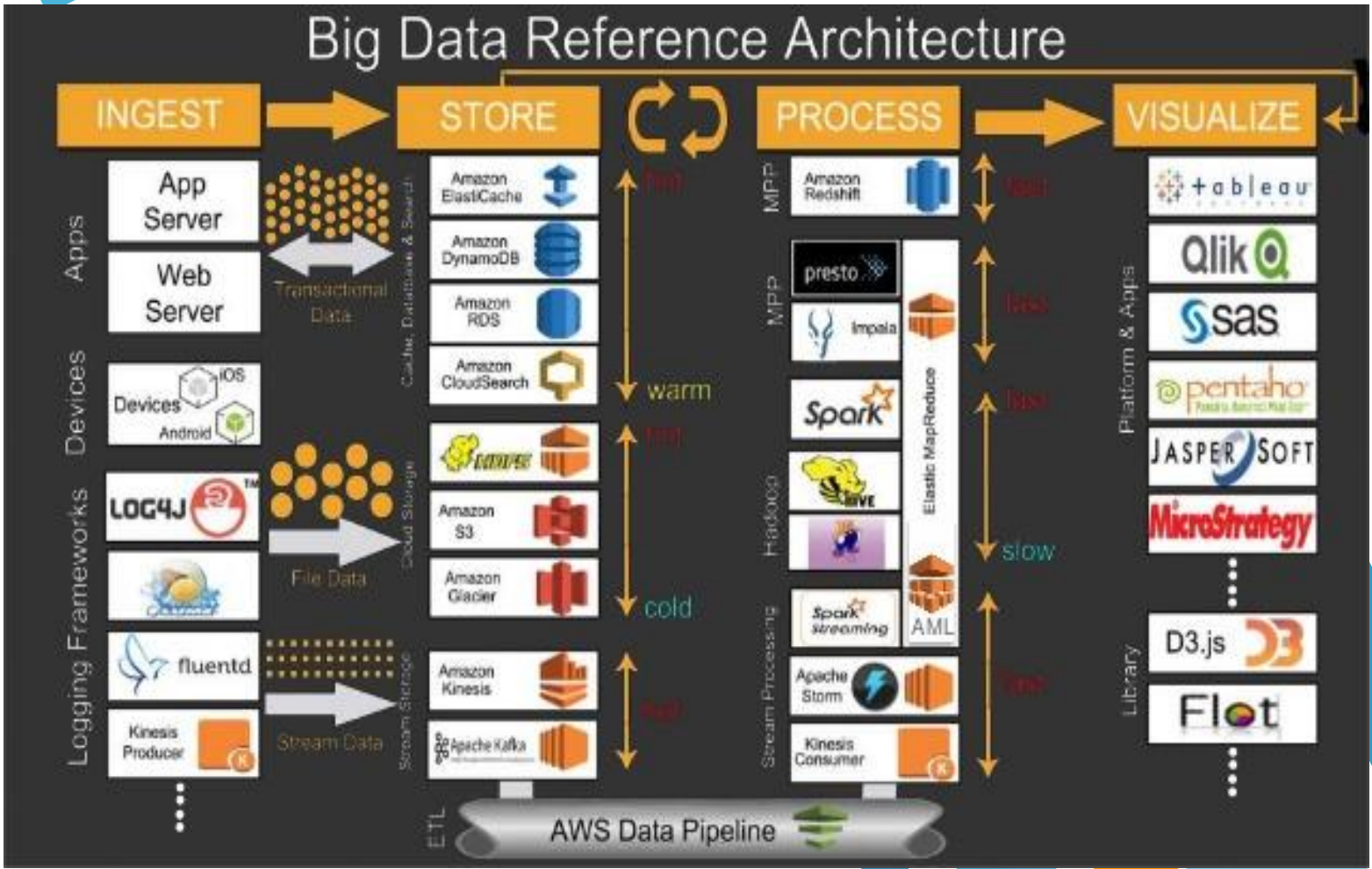


The idea is to handle both real-time data processing and continuous reprocessing in a single stream processing engine. That's right, reprocessing occurs from the stream. This requires that the incoming data stream can be replayed (very quickly), either in its entirety or from a specific position

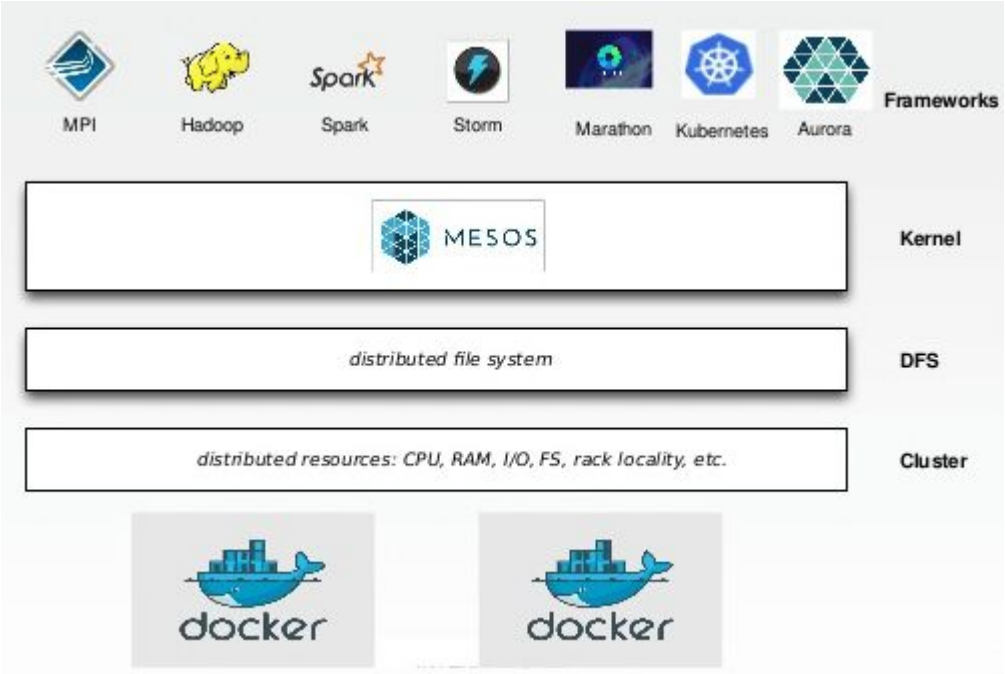
Evolution of data systems: 2017-Beyond

- **Kafka** expands it's services beyond log storage:
  - **Kafka Connect**: allows the community to provide connectors to many sources and sinks.
  - **Kafka Streams:** Provides an API that allow the users to process data inside the kafka cluster.
- **Fass providers** like AWS Lambda, Google Cloud Functions and Microsoft Azure Functions provide even easier elements to process the data into information in simple way.

Cloud services make them **easy to use**

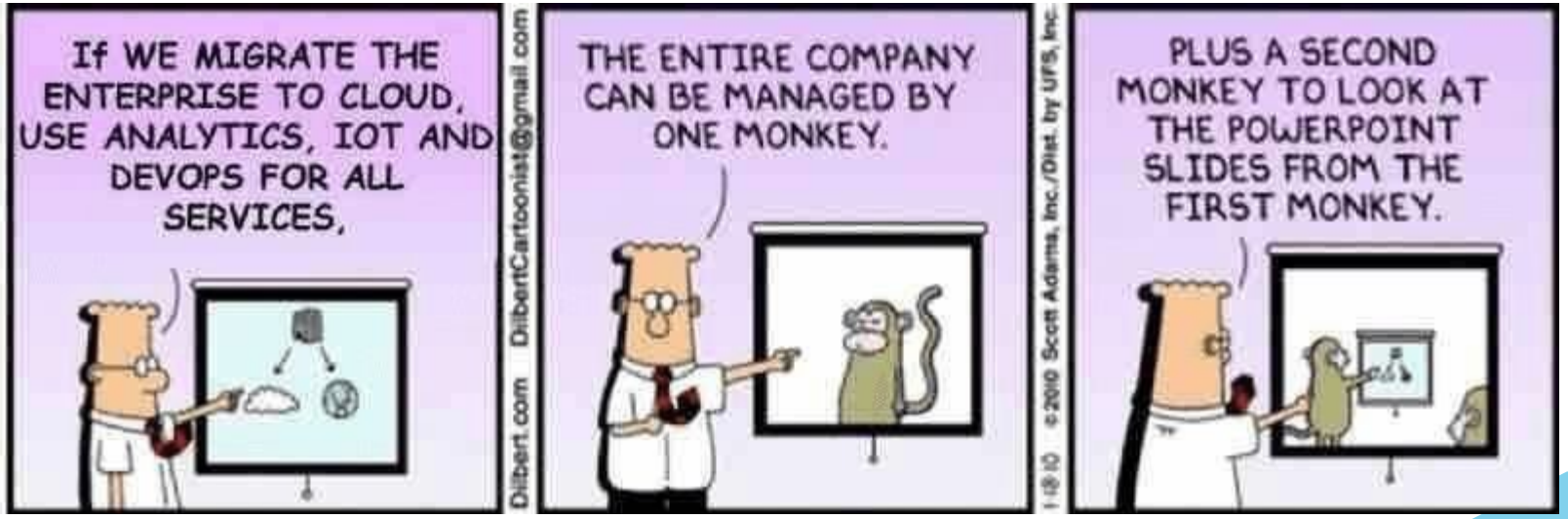


Automation system help too



Final Thoughts

Be aware with new technologies


If WE MIGRATE THE ENTERPRISE TO CLOUD, USE ANALYTICS, IOT AND DEVOPS FOR ALL SERVICES,
THE ENTIRE COMPANY CAN BE MANAGED BY ONE MONKEY.
PLUS A SECOND MONKEY TO LOOK AT THE POWERPOINT SLIDES FROM THE FIRST MONKEY.
Dilbert.com DilbertCartoonist@gmail.com
©2010 Scott Adams, Inc./Dist. by UFS, Inc.

Democratization of Information

Stepping out of the "business" aspect, one key aspect of this Tools is that as they make **the information process more "democratic" and "widely available".**

As this tools run on commodity cheap hardware you don't need to be a "big company" or a country to use them. Also helps the fact that they tend to not requiere strong coding background (at least at first level).

Small companies, ONGs, and small scientific endeavours all benefit from the ability to use this tools to make a more social impact in our society.

A couple of examples:

- Malaria no more project
- San Diego's Wifire project
- save the tigers

LET'S REVIEW SOME CONCEPTS!

Volume, Variety and Velocity of the incoming data requires scaling

Hardware or Cloud computing both getting cheaper and easier to use

Scalable software getting better and freely available

Big Data is the new competitive advantage

CREDITS

Content of the slides:

- Big Data Tools - ITBA - 2018

Images:

- Big Data Tools - ITBA - 2018
- obtained from: commons.wikimedia.org

Special thanks to all the people who made and released these awesome resources for free:

- Presentation template by SlidesCarnival
- Photographs by Unsplash