UMONS
Université de Mons

Faculté Polytechnique

DATA BASES

CHAPTER 7 : BIG DATA

Sidi Ahmed Mahmoudi

POLYTECH MONS

sidi.mahmoudi@umons.ac.be

08 November 2017

PLAN

Introduction

I. History of Big Data

II. Definitions

III. The 4V of Big Data

IV. Big Data technologies

V. The bases of Big Data

- HDFS
- Map Reduce

VI. NoSQL

VII. Big Data example

Conclusion

Introduction

- Big Data is all about (big) data
- Big Data is not a trend but concepts and technologies that are already endorsed

"Big Data: What's Next And How To Handle It All"
Forbes

"Big Data: Rise of the Machines"
New York Times

"Big Data Is Changing the Way We Look at Problems"
News & Observer

"EU Big Data Roadmap Can Provide Jobs Boost"
Irish Times

"How Big Data Is Changing the Whole Equation for Business"
Wall Street Journal

"Big Data Can Change the World"
Los Angeles Times

"Pentagon's Big Data Strategy Taking Shape"
Washington Post

"Shopping or browsing on Main St? India's Big Data Firms Know"
Chicago Tribune

"Big Data Could Cripple Facebook"
TechCrunch

"Big Data: March Madness Math in the Real World"
CNBC

NC STATE UNIVERSITY

History of Big Data

Founder of mathematical abstractions and algorithms

Algorithms, ~800

5 tons : computes 16 digits & 6 orders of difference

Babbage, 1822

167 $m^{2,}$ 50 tons

Computer ENIAC 1943

9 x15 $m^{2,}$ 1 tons

IBM 305 RAMAC, 1956

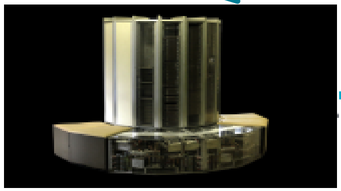
History of Big Data

Algorithms, ~800

Babbage, 1822

Computer ENIAC 1943

IBM 305 RAMAC, 1956



Parallel computing,1975

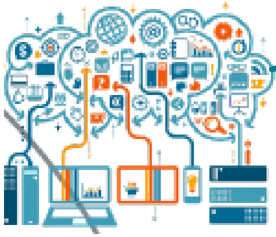
160 MFlops and up
Memory : 1 MB-32 MB

Distributed computing ~1980

Distributed computation & storage

Cloud computing

Data explosion & HPC



Big Data

Transform data to information

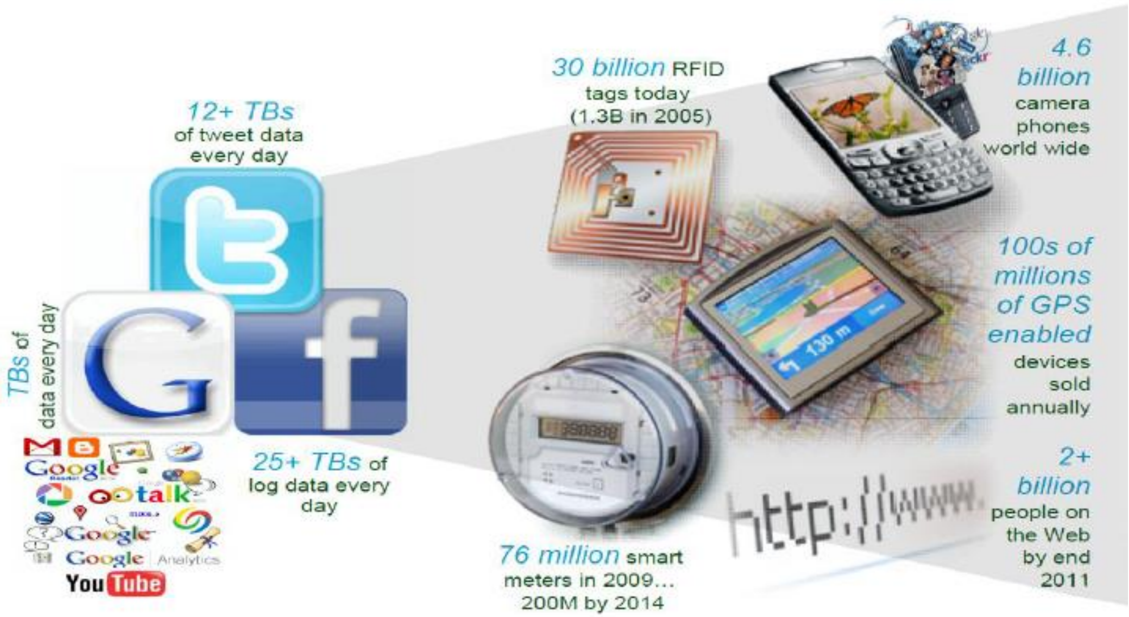
History of Big Data

Volume generation

- Every day, 2.5 trillion bytes of data are generated
- 90% of the data created in the world have been generated in the last 2 years
- Forecast growth of 800% in the amount of data to be processed within 5 years

Diversity of sources:

- sensors, social media, images, videos, online shopping, GPS signals


12+ TBs of tweet data every day
TBs of data every day
25+ TBs of log data every day
30 billion RFID tags today (1.3B in 2005)
4.6 billion camera phones world wide
100s of millions of GPS enabled devices sold annually
76 million smart meters in 2009… 200M by 2014
2+ billion people on the Web by end 2011
http://www.

History of Big Data : Examples

- The company Air Bus generates 10 TB every 30 minutes.
- About 640 TB of data are generated for each flight.

- Facebook generates about 500 TB of data every day.

- Boeing 737 generate 240 TB of data for each flight.

Definitions

Who owns the information, owns the world

Francis Bacon

Literally

Massive volume of structured or non-structured data (Datamasse)

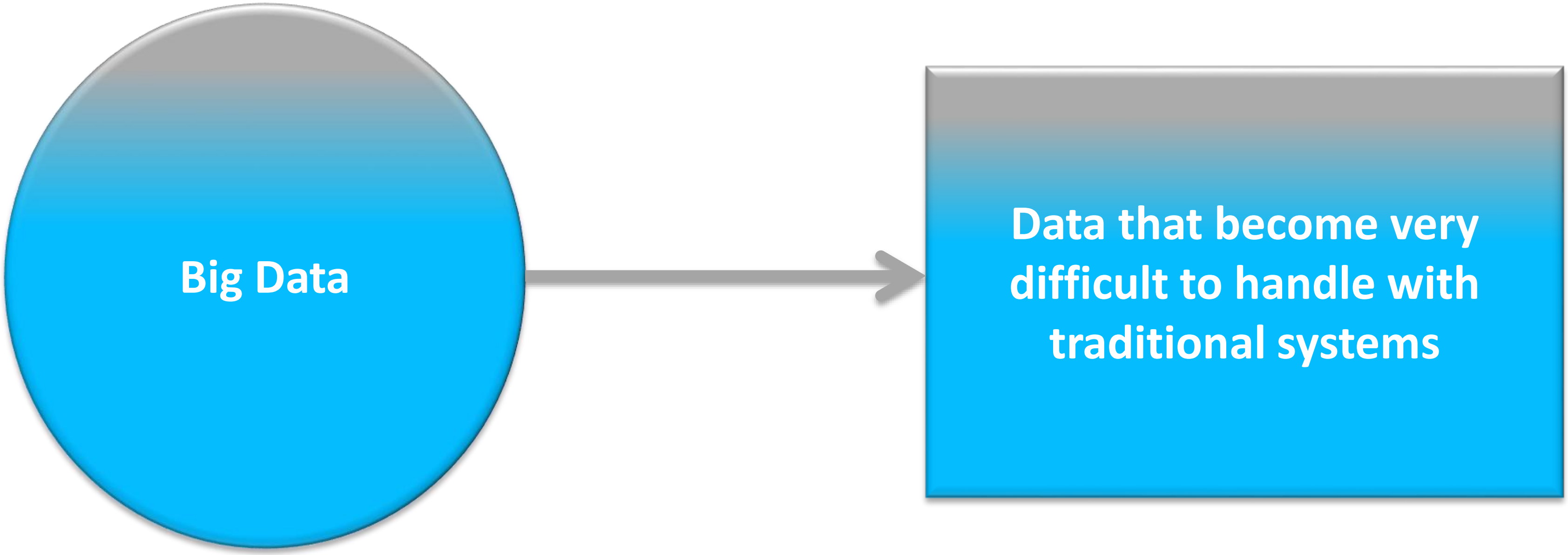
Conceptually

Big Data vulgarizes both the representation of large data volumes and the related infrastructures related to the treatment of these data.

What is Big Data?

Name	Symbole	Value
Kilobyte	KB	10^3
Megabyte	MB	10^6
Gigabyte	GB	10^9
Terrabyte	TB	10^{12}
Petabyte	PB	10^{15}
Exabyte	EB	10^{18}
Zettabyte	ZB	10^{21}
Yottabyte	YB	10^{24}

What is Big Data?



Big Data

Data that become very difficult to handle with traditional systems

Difficult to process by Traditionel System

Unable to send

100 MB Document

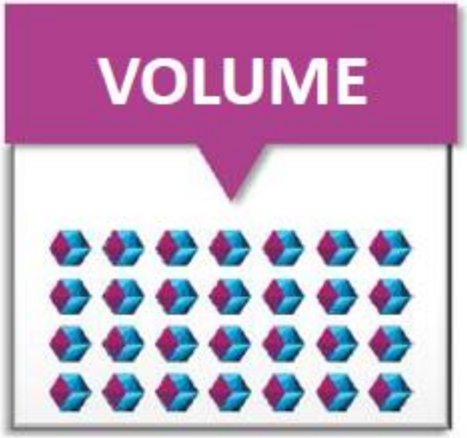
Unable to visualize

100 GB Image

Tous Dépend le Système?

wetransfer

The 4V of Big Data



The 4V of Big Data

Volume :

Continuous growth of data of any type and size (in Terabytes or even in Petabytes)

Variety :

Treatment of structured and unstructured data that require a collective analysis (databases, texts, sensors data, sounds, videos, paths, files, newspapers, etc.)

Velocity:

Use and exploitation of data in real time (exp. detection of fraud, etc.)

Veracity:

Management of the reliability and veracity of inaccurate and predictive data

Technologies of Big Data

BIG DATA : Open Source actors

- The major actors of the web such as Google, Yahoo, Facebook, Twitter, LinkedIn … were the first actors confronted with very large volumes of data.
- They were at the origin of the first innovations in this field of Big Data within two types of technologies:
    - NoSql databases
    - Platforms of data development and treatment

The majority of these companies decided to open these internal developments to the Open Source world

Example : Hadoop from Apache foundation integrated to the Big Data offers of IBM, Oracle, Microsoft, EMC …

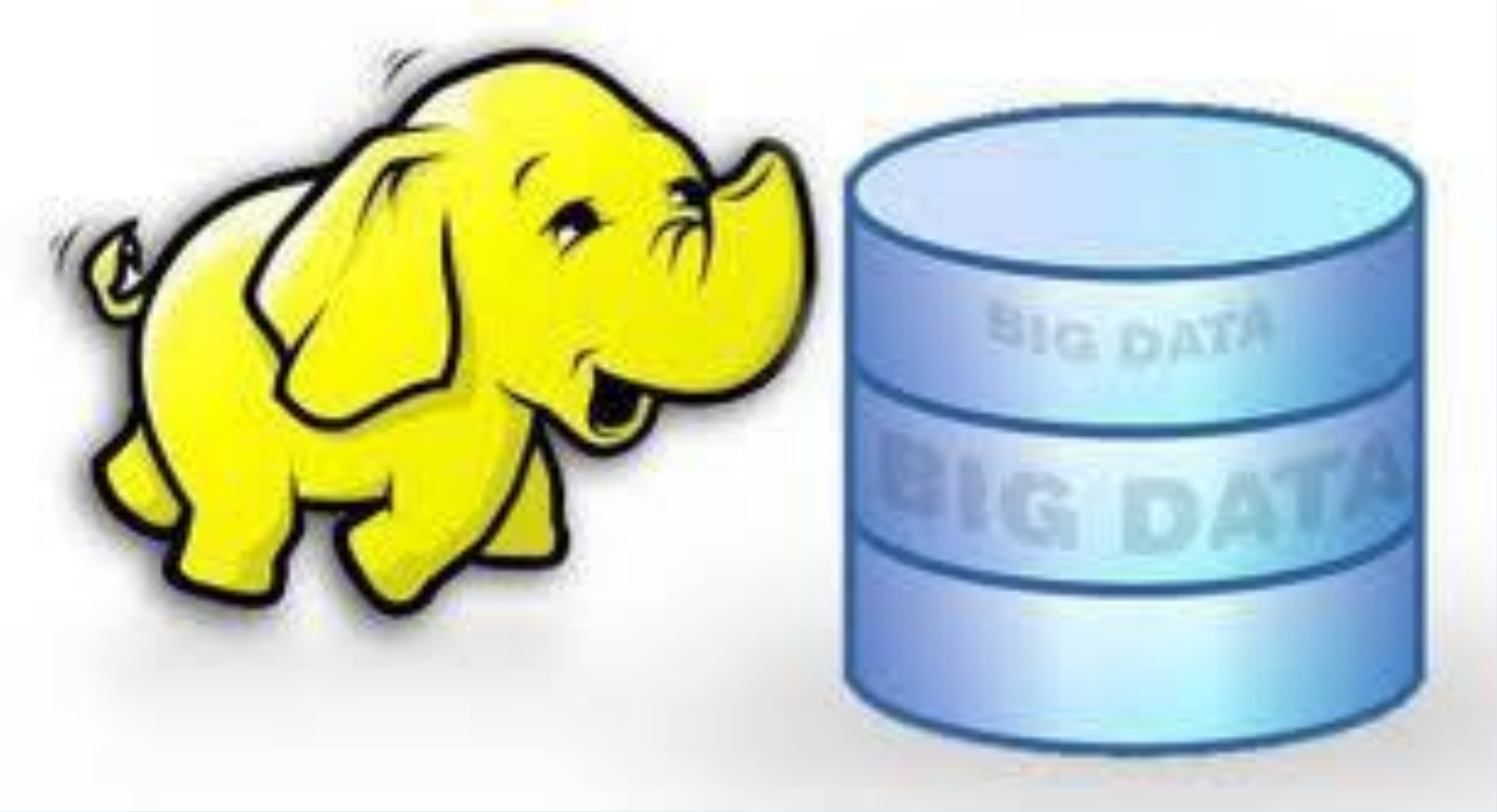
Technologies of Big Data

Company	Developed technology	Type of technology
Google	Big Table	Distributed database system based on GFS (Google File System). Non-open source technology, which inspired the open source HBase.
	MapReduce	Platform of development and distributed treatment.
Yahoo	Hadoop	Java Platform for distributed applications and intensive data management. Originally derived from google Big Table, MapReduce, and Google File System.
	S4	Development platform dedicated to continuous data flow processing applications
Facebook	Cassandra	NoSQL and distributed Database
	Hive	Software of data analysis using Hadoop
Twitter	Storm	Platform of massive data treatment
	FlockDB	Distributed database of type graph
LinkedIn	Kafka	Distributed system of messages management
	SenseiDB	Real time distributed and semi-structured database
	Voldemort	Distributed database for very large volumes

Open source technologies of Big Data [Lavoisier]

The bases of Big Data : HADOOP/MAPREDUCE

Hadoop

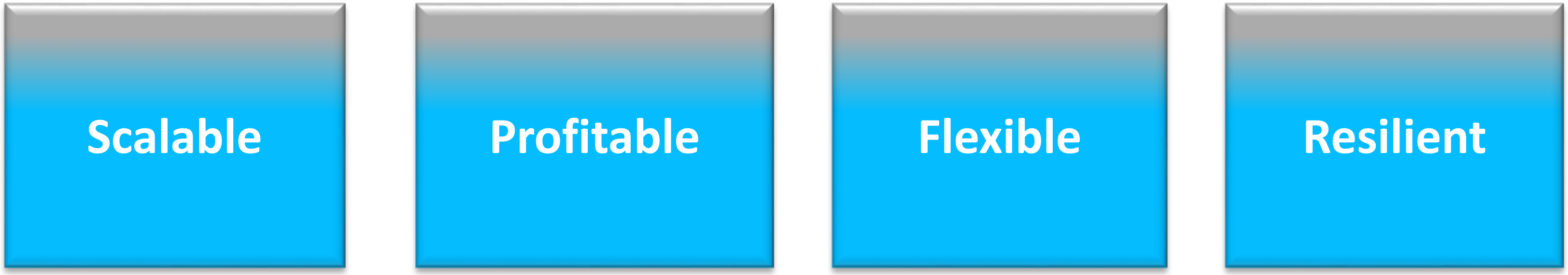


The bases of Big Data : HADOOP

- **Hadoop** is an open source framework known for its power of indexing, transforming, researching or developing models on very large volumes of data.

- **Hadoop** offers distributed treatment of large datasets through clusters of computers using simple programming models.

Characteristic of Hadoop



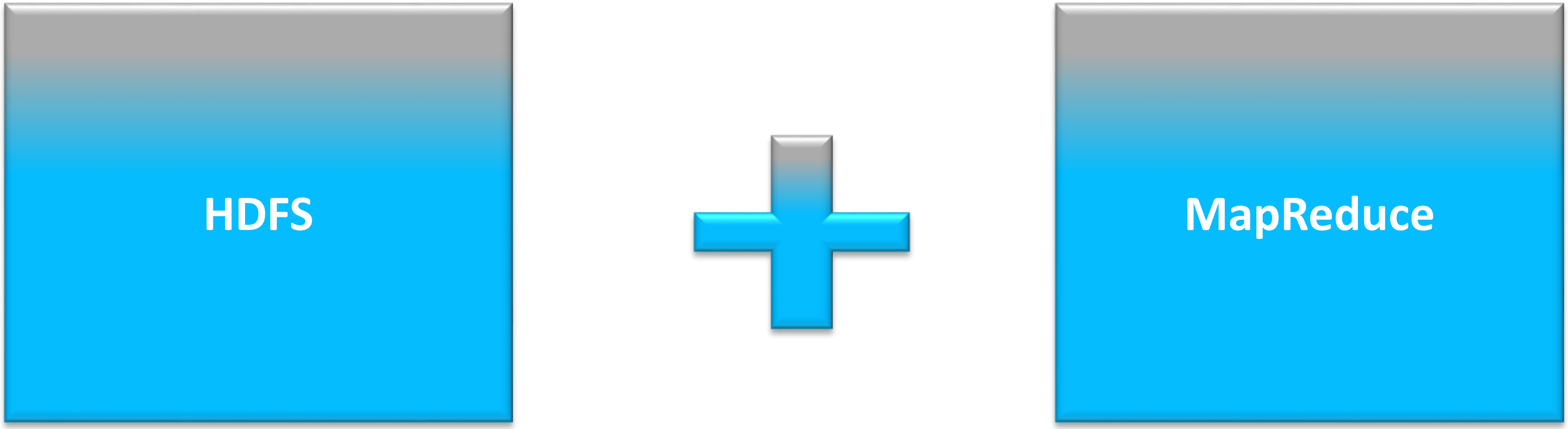
Scalable

Profitable

Flexible

Resilient

Architecture



HDFS

MapReduce

HDFS (Hadoop Distributed File System)

- Distributed, extensible and portable file system developed by Hadoop
- Based on the Map Reduce principle from Google File System
- Written in Java and designed to store large volumes of data on low costly distributed machines (equipped with common hard disks)
- Abstraction of the physical storage architecture in order to manipulate a distributed file system as if is was a single hard disk
- Many companies use Hadoop such as : Adobe, AOL, Bing (Microsoft), eBay, Facebook, Google, IBM, LinkedIn, Twitter, Yahoo, Spotify, etc.

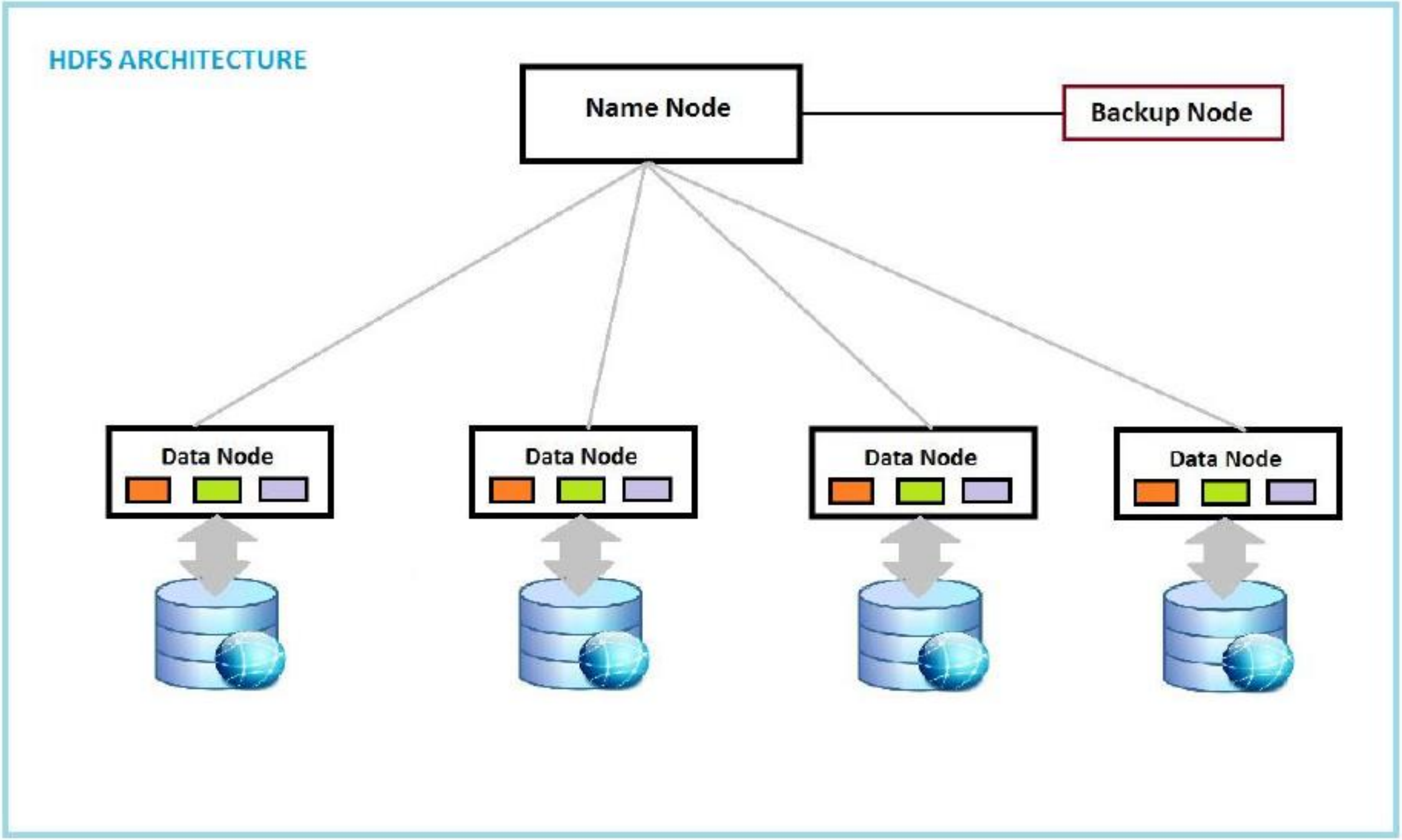
HDFS (Hadoop Distributed File System)

An HDFS machine architecture (or HDFS cluster) is based on two major components :

a. **NameNode**

b. **DataNode**

HDFS (Hadoop Distributed File System)



HDFS (Hadoop Distributed File System)

NameNode

- This component manages the namespace, the file system tree, metadata and directories.
- It centralizes the location of the data blocks, distributed in the cluster.
- It is unique but has a secondary instance that handles the history of changes in the file system (backup role).
- This Secondary NameNode allows the Hadoop cluster to continue functioning in the event of a failure of the original NameNode

HDFS (Hadoop Distributed File System)

DataNode

- This component stores and restores (restitutes) the data blocks.
- During the process of reading a file, the NameNode is queried to locate all of the data blocks.
- For each of them, the NameNode returns the address of the most accessible DataNode, ie the DataNode which has the highest bandwidth.
- The DataNodes periodically communicate to the NameNode the list of data blocks that they host.

HDFS (Hadoop Distributed File System)

- Each DataNode serves as a data block on the network using an HDFS-specific protocol.
- The file system uses the TCP/IP layer for communication. Customers use the Remote Procedure Call to communicate with each other.
- The HDFS stores large files on multiple machines. It achieves reliability by replicating data across multiple hosts and therefore does not require RAID storage on hosts.
- With the default replication value, the data is stored on three nodes: two on the same support and the other on a different support.
- DataNodes can communicate with each other to rebalance data and maintain a high level of data replication.

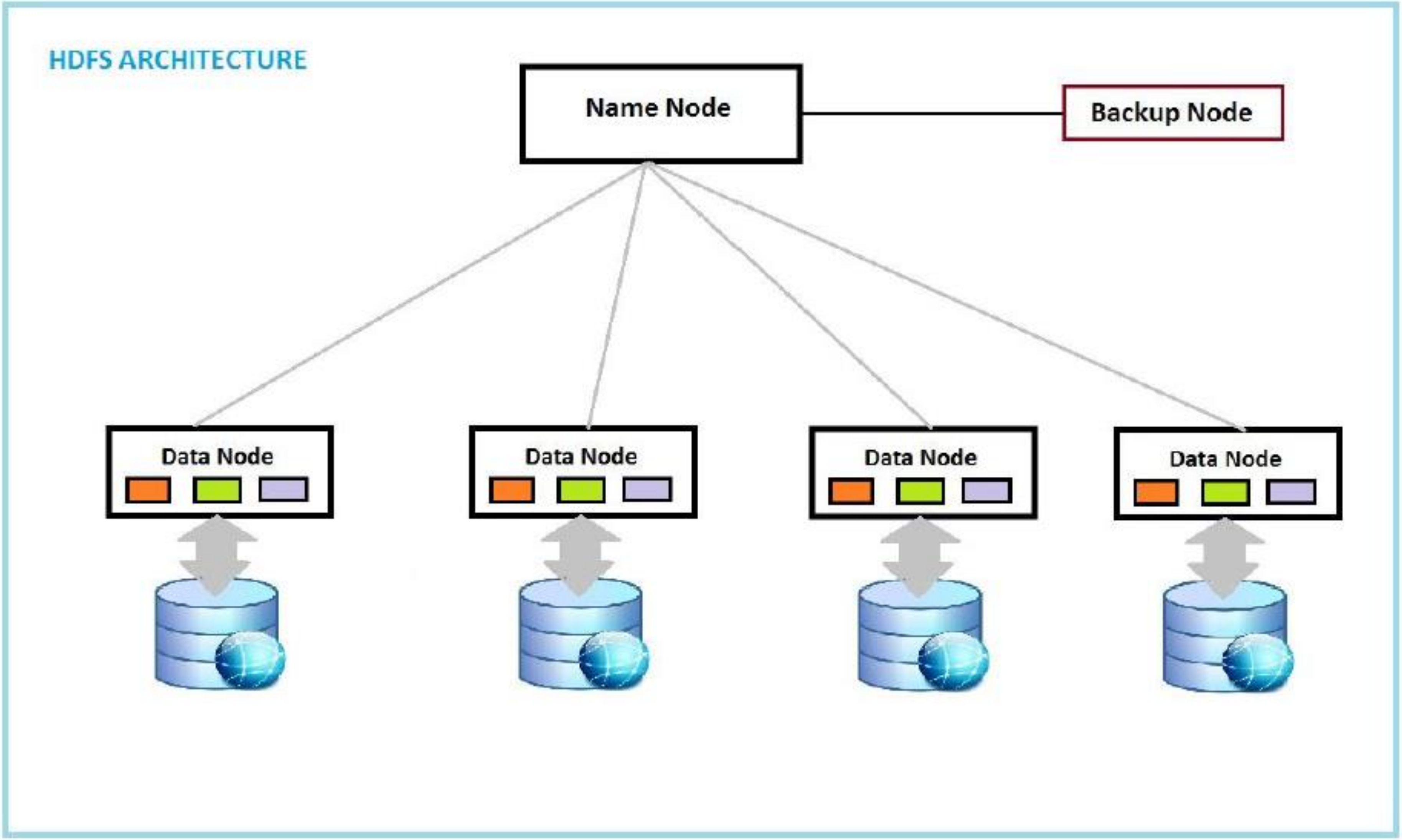
HDFS (Hadoop Distributed File System)

- HDFS has recently improved its capacities of high availability, which allows the primary metadata server to be manually switched to a backup in case of failure.

- Since the NameNodes are the unique point for storing and managing metadata, they can be a bottleneck to support a large number of files, especially when they are small.

- By accepting multiple namespaces served by separate NameNodes, the HDFS limits this problem.

HDFS (Hadoop Distributed File System)

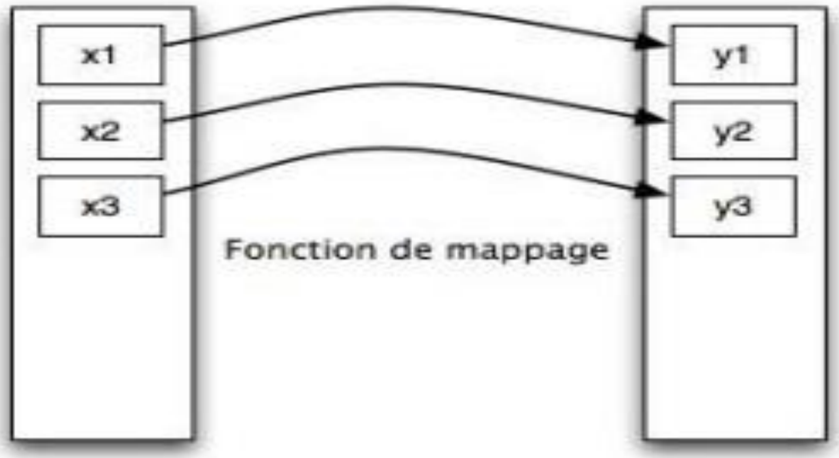


Map Reduce

- It presents a major role in the processing of large amounts of data.
- The distribution of data within servers allows the parallel processing of several tasks, each one dealing with its files.
- The **Map** function performs a specific operation on each element.
- The **Reduce** operation combines the elements according to a particular algorithm, and provide the result.
- The delegation principle can be recursive: the nodes to which tasks are entrusted can also delegate operations to other nodes.

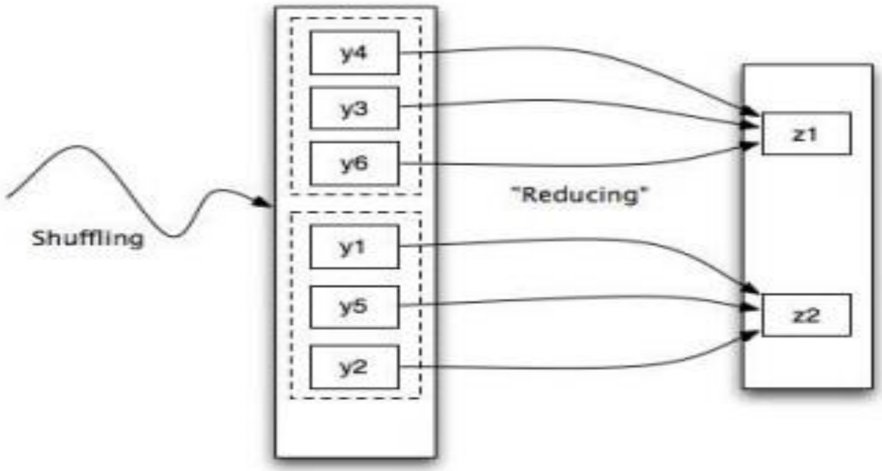
Map

Map: this step performs a specific operation on each element of the input list. From a list in the form (key, value), it generates an output list in the same form



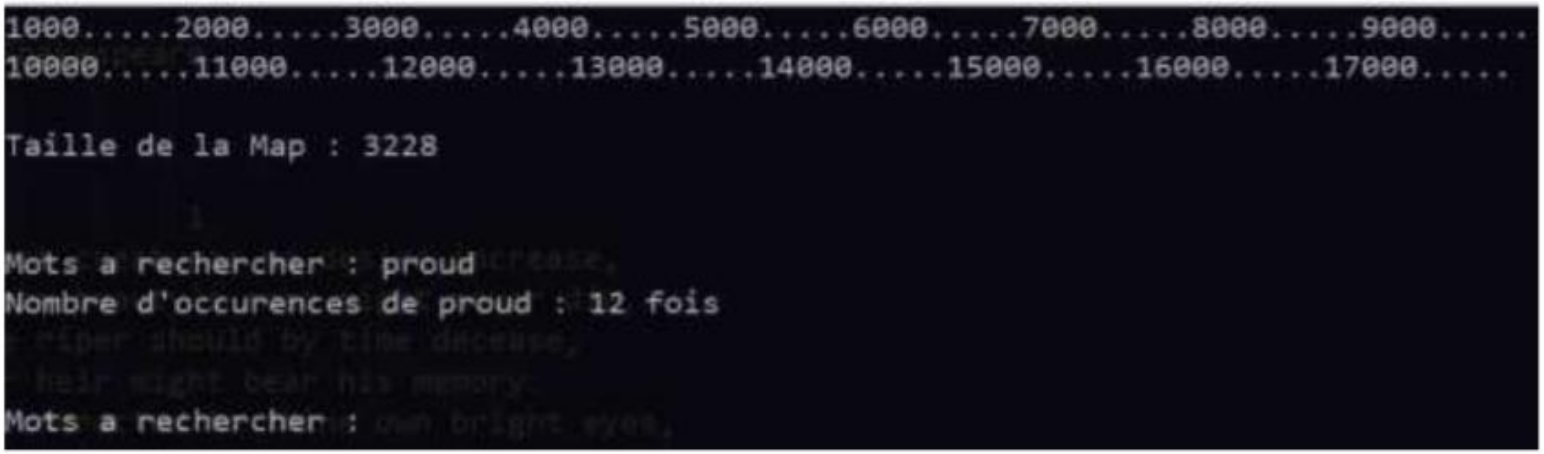
Reduce

Reduce: the operation between the Mapping and the Reducing is called the Shuffling, and rearranges the items in the list to prepare the Reducing. The desired processing is then carried out, giving the following final output



Example

3. Réaliser une fonction qui permet d'introduire un mot au clavier et retourner son occurrence dans le texte

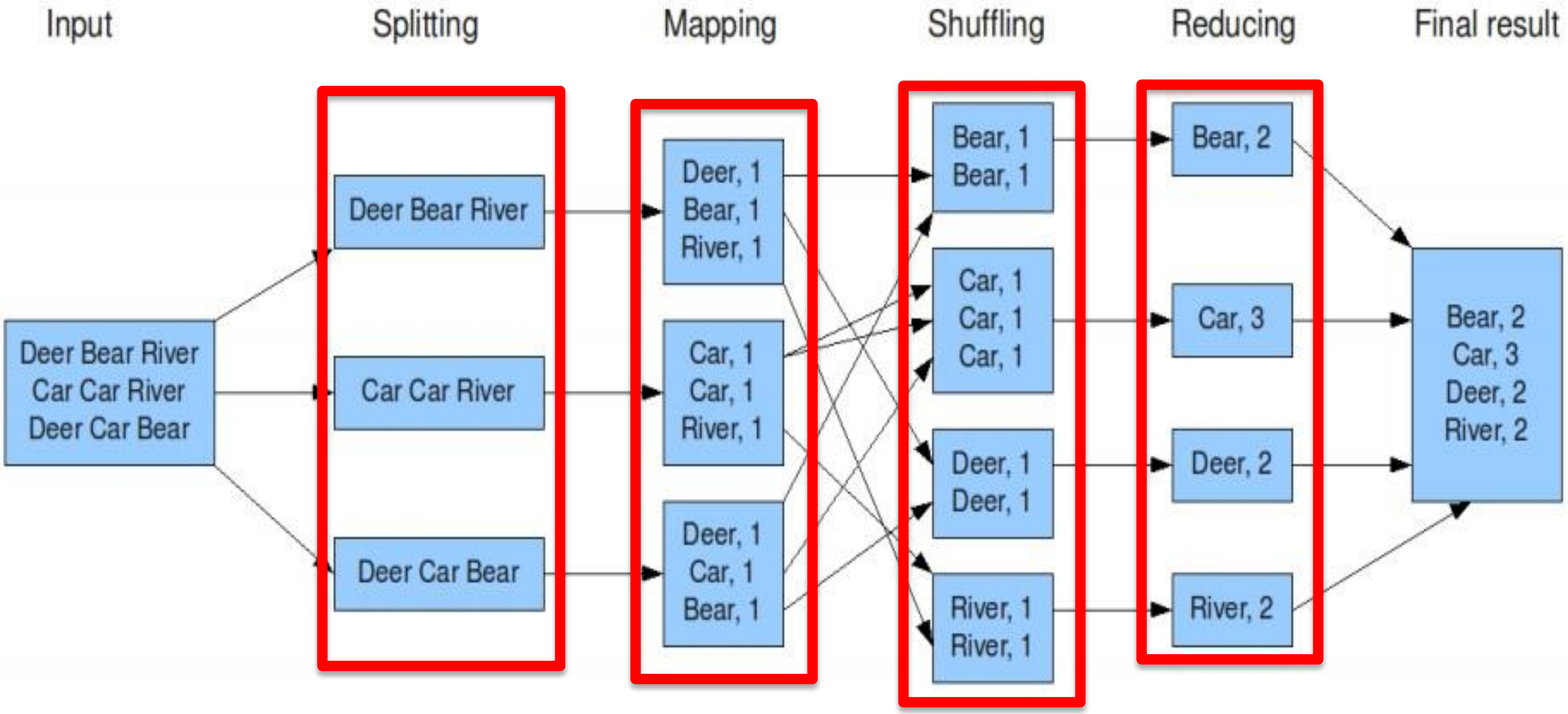


```
1000.....2000.....3000.....4000.....5000.....6000.....7000.....8000.....9000.....
10000.....11000.....12000.....13000.....14000.....15000.....16000.....17000.....

Taille de la Map : 3228


Mots a rechercher : proud
Nombre d'occurences de proud : 12 fois


Mots a rechercher :
```

4. Chercher le mot le plus récurrent dans la totalité des œuvres de William Shakespeare

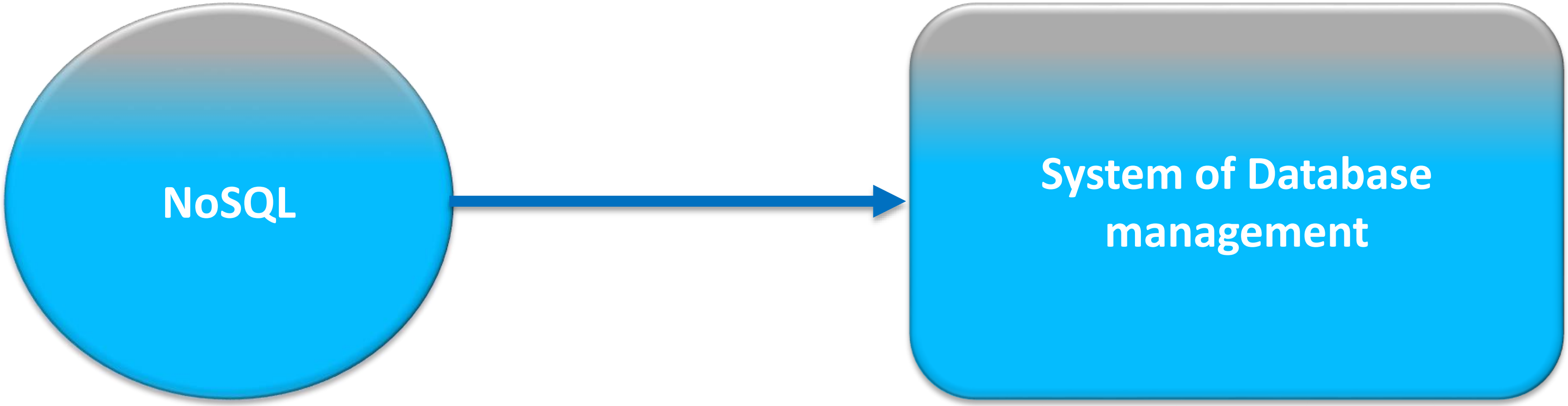
Les Map sont des conteneurs associatifs qui stockent des éléments formés d'une combinaison d'une valeur de clé et d'une valeur mappée, suivant un ordre spécifique.

Example

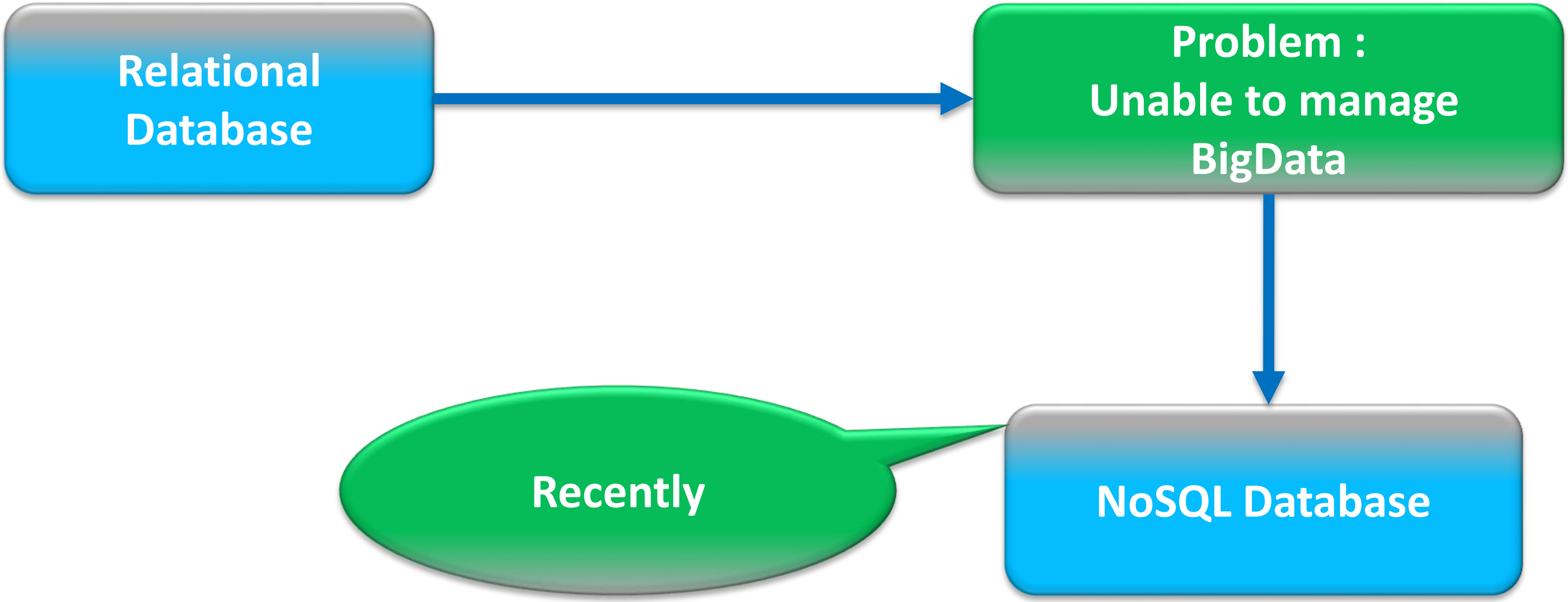

Input
Splitting
Mapping
Shuffling
Reducing
Final result
Deer Bear River
Car Car River
Deer Car Bear
Deer Bear River
Car Car River
Deer Car Bear
Deer, 1
Bear, 1
River, 1
Car, 1
Car, 1
River, 1
Deer, 1
Car, 1
Bear, 1
Bear, 1
Bear, 1
Car, 1
Car, 1
Car, 1
Deer, 1
Deer, 1
River, 1
River, 1
Bear, 2
Car, 3
Deer, 2
River, 2
Bear, 2
Car, 3
Deer, 2
River, 2

NoSQL

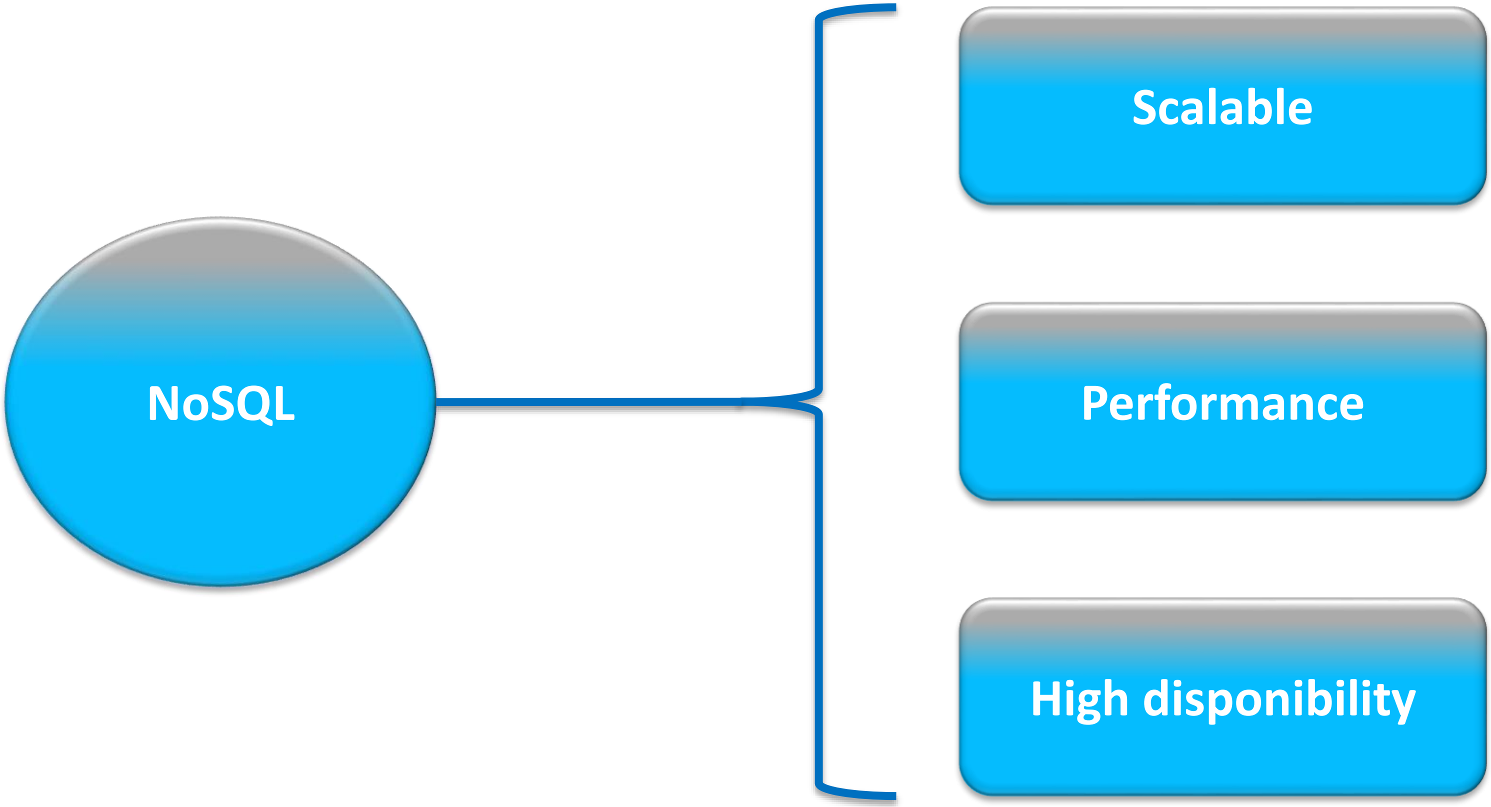
What is NoSQL



NoSQL Databases



Objective



NoSQL

Scalable

Performance

High disponibility

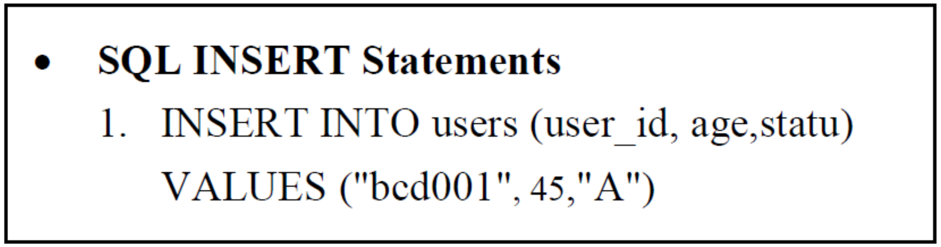
NoSQL

- These data types do not require the table structure and even the joins between these tables

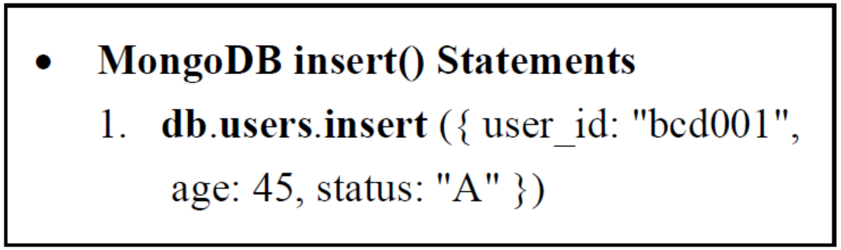
- The NoSQL database system must be non-relational, distributed, open source and horizontally scalable.

- Actually there over 25 types of NoSQL databeses, each with their own caracteristics that are based on different scenarios such as : **HBase, Cassandra, Hypertable, SimpleDB, MongoDB, CouchDB, DynamoDB, Redis, Ne04J, etc.**

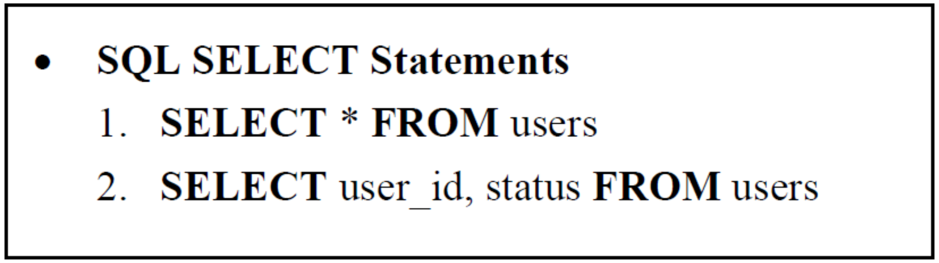
SQL and NoSQL requests



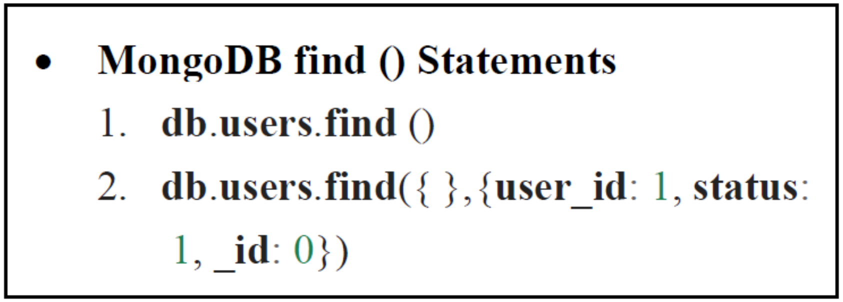
- **SQL INSERT Statements**
  1. INSERT INTO users (user_id, age,statu) VALUES ("bcd001", 45,"A")



- **MongoDB insert() Statements**
  1. **db.users.insert** ({ user_id: "bcd001", age: 45, status: "A" })



- **SQL SELECT Statements**
  1. **SELECT * FROM** users
  2. **SELECT** user_id, status **FROM** users



- **MongoDB find () Statements**
  1. **db.users.find** ()
  2. **db.users.find**({ },{**user_id**: 1, **status**: 1, **_id**: 0})

Types of NoSQL Databes

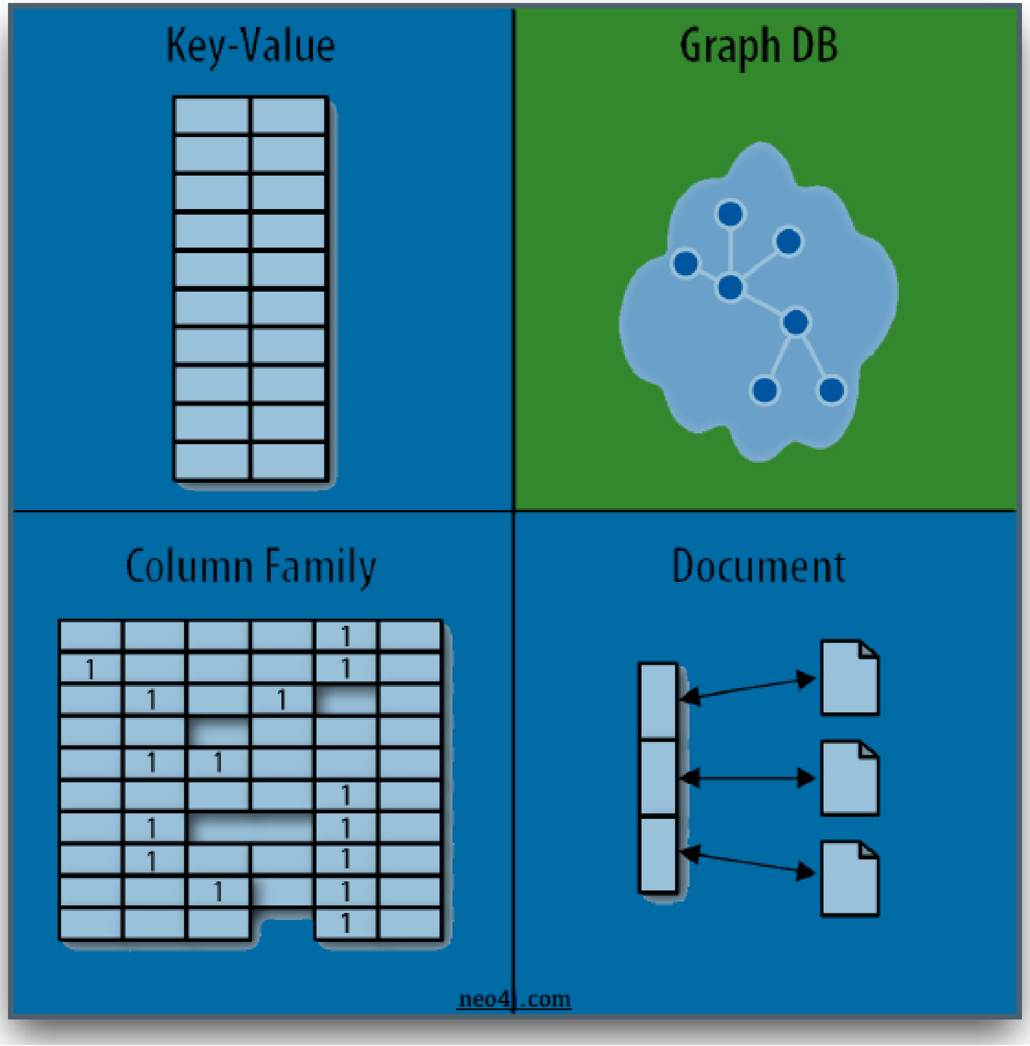
Kay-Value

Document

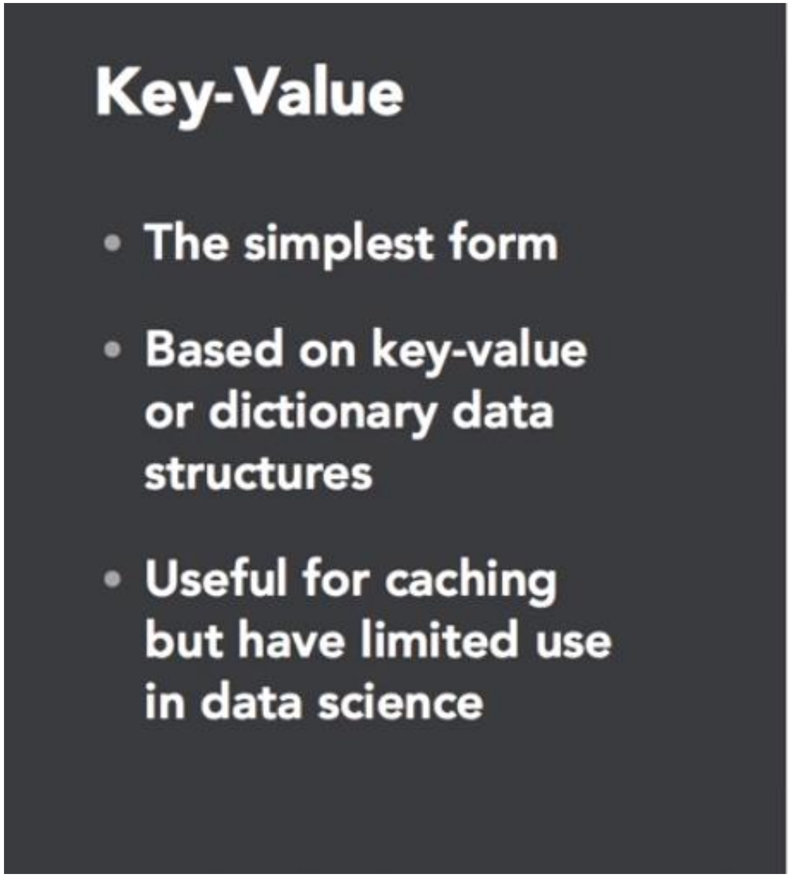
Wide column

Graph

Types of NoSQL Databes

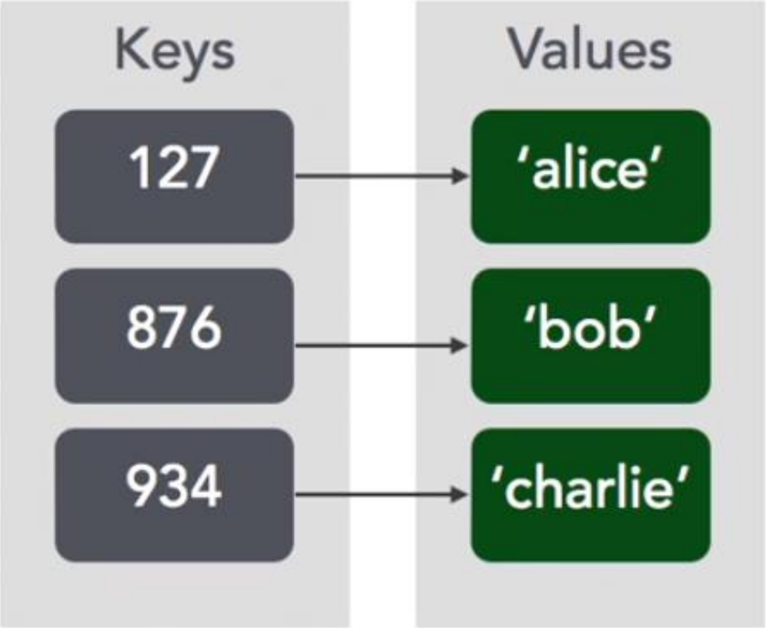


Types of NoSQL Databes : Key-Value



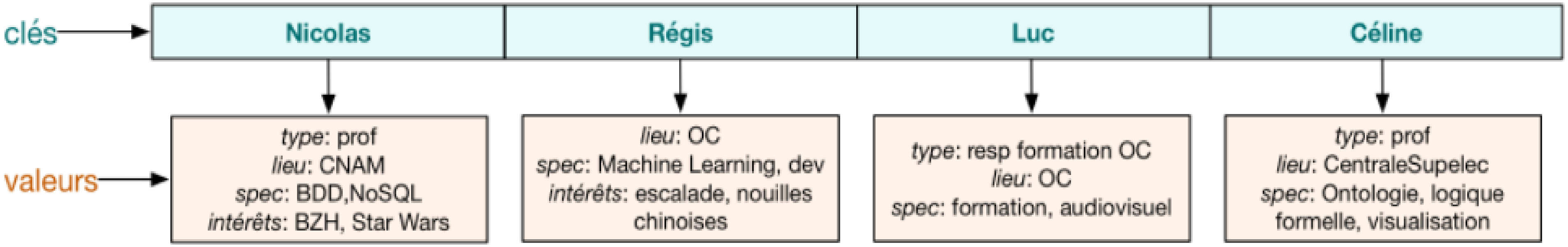
Key-Value

- The simplest form
- Based on key-value or dictionary data structures
- Useful for caching but have limited use in data science



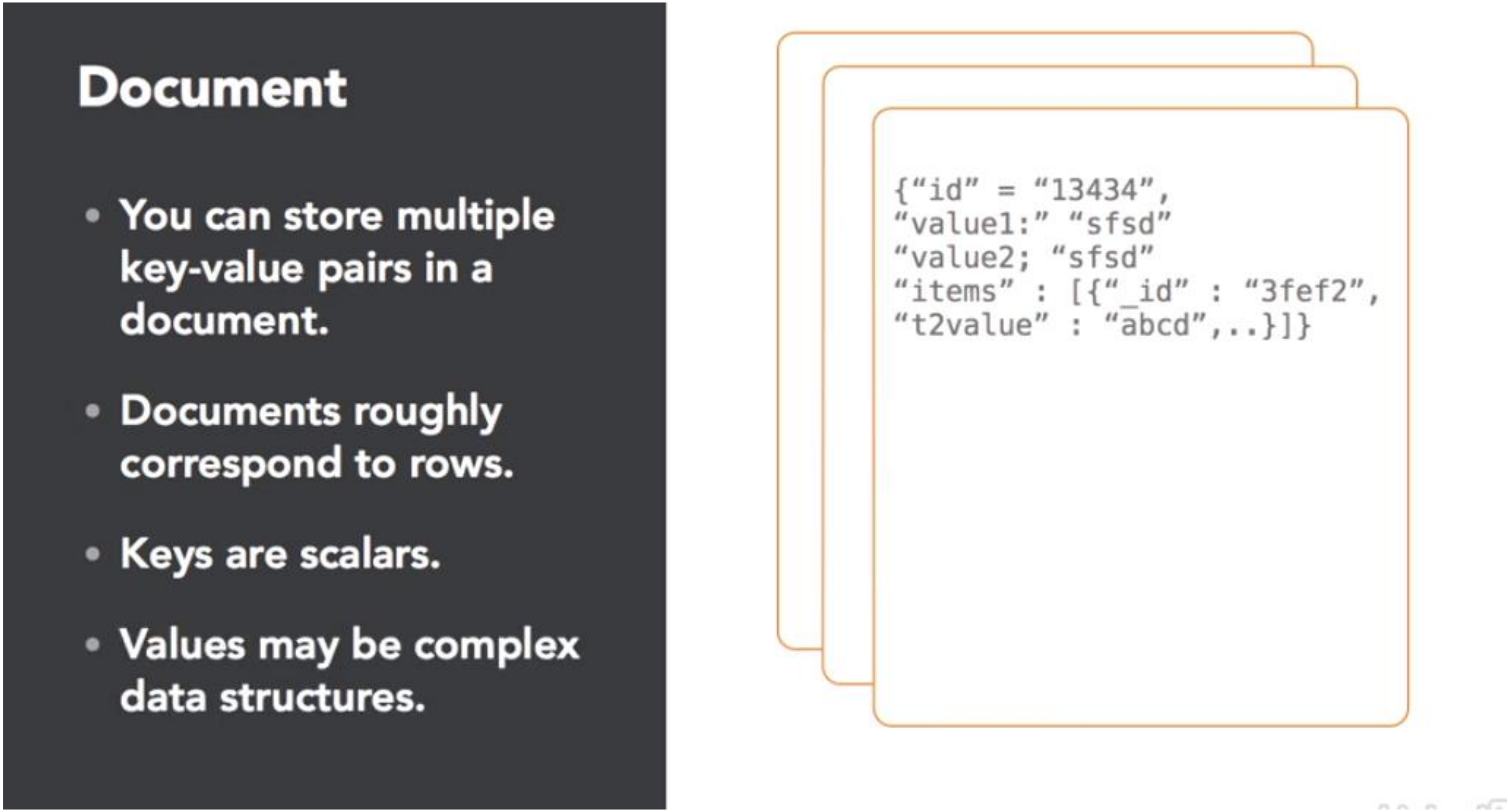
Keys	Values
127	'alice'
876	'bob'
934	'charlie'

Types of NoSQL Databes : Key-Value



clés	Nicolas	Régis	Luc	Céline
valeurs	*type*: prof *lieu*: CNAM *spec*: BDD,NoSQL *intérêts*: BZH, Star Wars	*lieu*: OC *spec*: Machine Learning, dev *intérêts*: escalade, nouilles chinoises	*type*: resp formation OC *lieu*: OC *spec*: formation, audiovisuel	*type*: prof *lieu*: CentraleSupelec *spec*: Ontologie, logique formelle, visualisation

Types of NoSQL Databes : Document

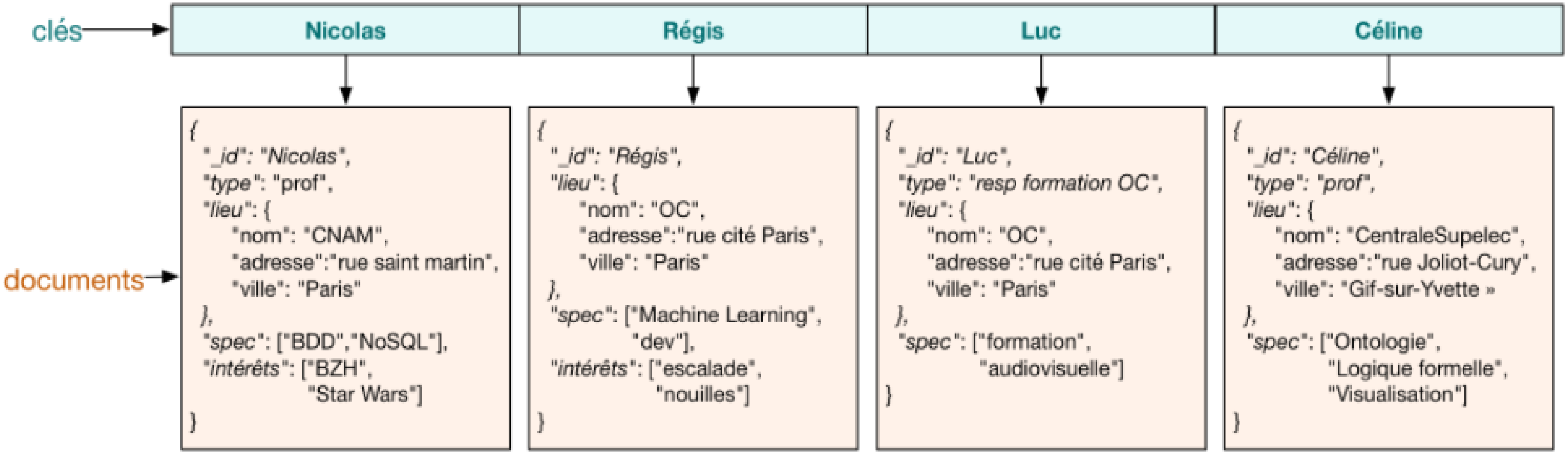


Document

- You can store multiple key-value pairs in a document.
- Documents roughly correspond to rows.
- Keys are scalars.
- Values may be complex data structures.

```
{“id” = “13434”,
“value1:” “sfsd”
“value2; “sfsd”
“items” : [{“_id” : “3fef2”,
“t2value” : “abcd”,..}]}
```

Types of NoSQL Databes : Document



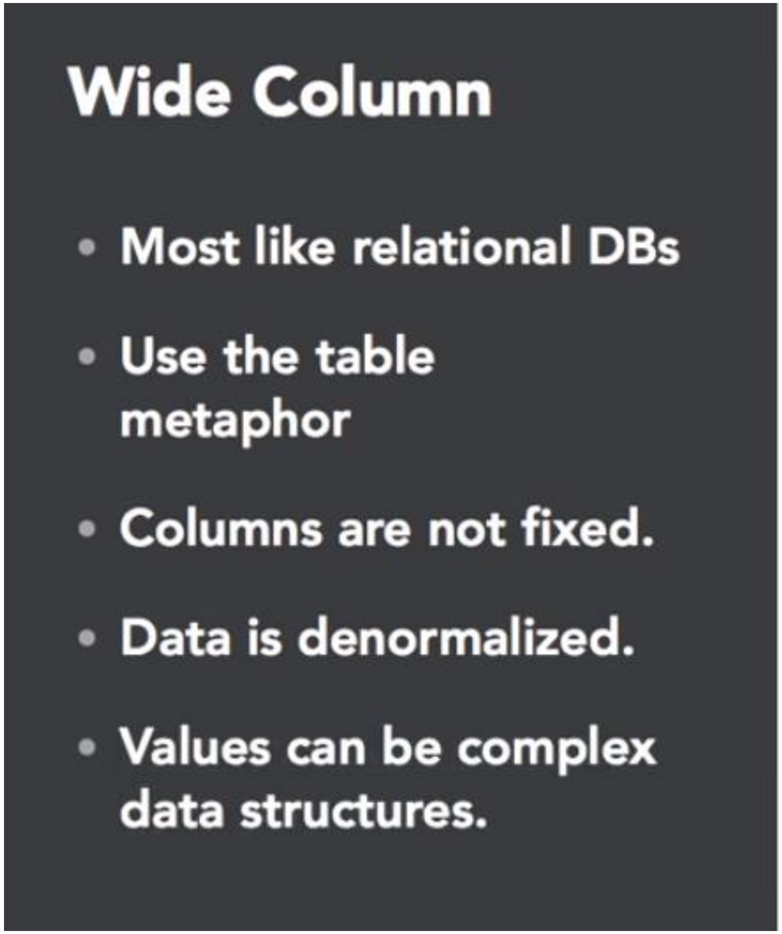
```
{
  "_id": "Nicolas",
  "type": "prof",
  "lieu": {
      "nom": "CNAM",
      "adresse":"rue saint martin",
      "ville": "Paris"
  },
  "spec": ["BDD","NoSQL"],
  "intérêts": ["BZH",
               "Star Wars"]
}
```

```
{
  "_id": "Régis",
  "lieu": {
      "nom": "OC",
      "adresse":"rue cité Paris",
      "ville": "Paris"
  },
  "spec": ["Machine Learning",
           "dev"],
  "intérêts": ["escalade",
               "nouilles"]
}
```

```
{
  "_id": "Luc",
  "type": "resp formation OC",
  "lieu": {
      "nom": "OC",
      "adresse":"rue cité Paris",
      "ville": "Paris"
  },
  "spec": ["formation",
           "audiovisuelle"]
}
```

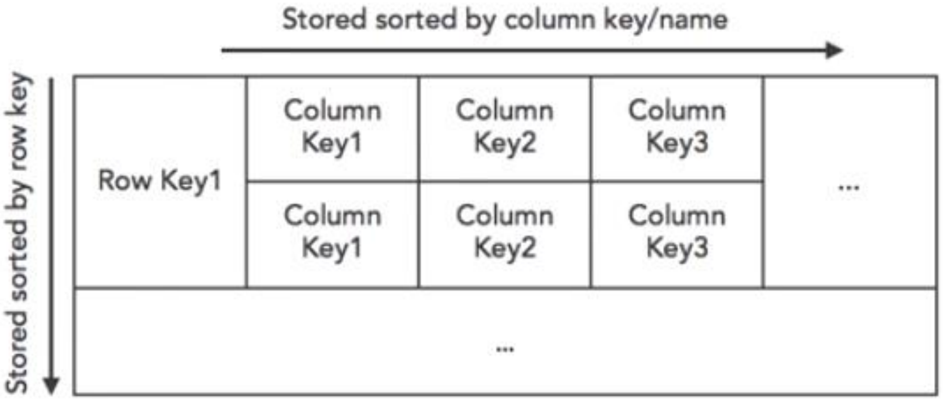
```
{
  "_id": "Céline",
  "type": "prof",
  "lieu": {
      "nom": "CentraleSupelec",
      "adresse":"rue Joliot-Cury",
      "ville": "Gif-sur-Yvette »
  },
  "spec": ["Ontologie",
           "Logique formelle",
           "Visualisation"]
}
```

Types of NoSQL Databes : Wide column



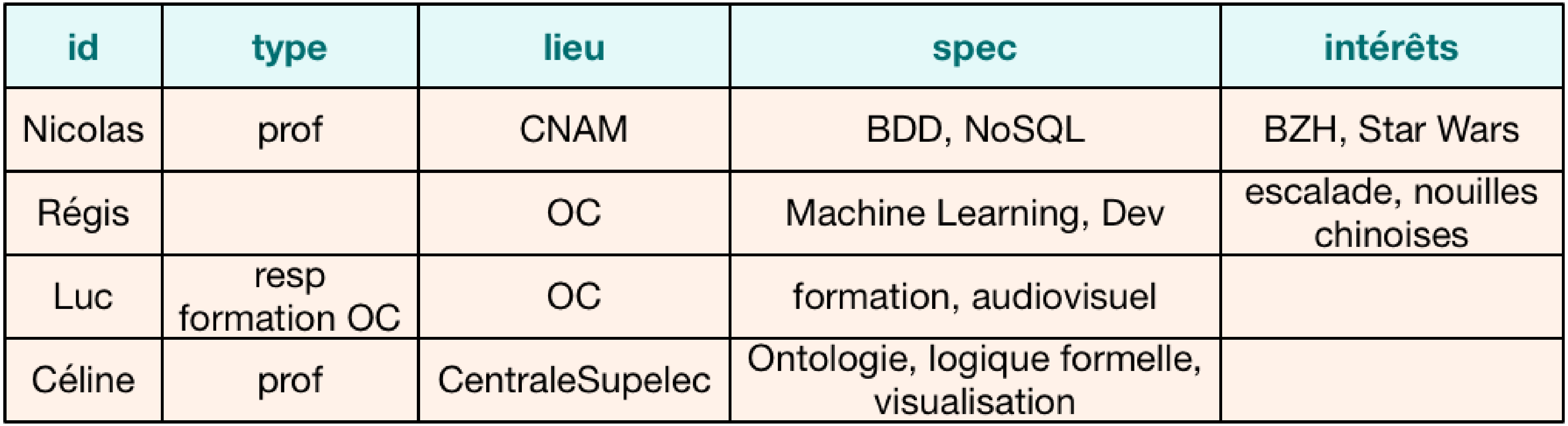
Wide Column

- Most like relational DBs
- Use the table metaphor
- Columns are not fixed.
- Data is denormalized.
- Values can be complex data structures.



Row Key1	Column Key1	Column Key2	Column Key3	...
	Column Key1	Column Key2	Column Key3	
...				

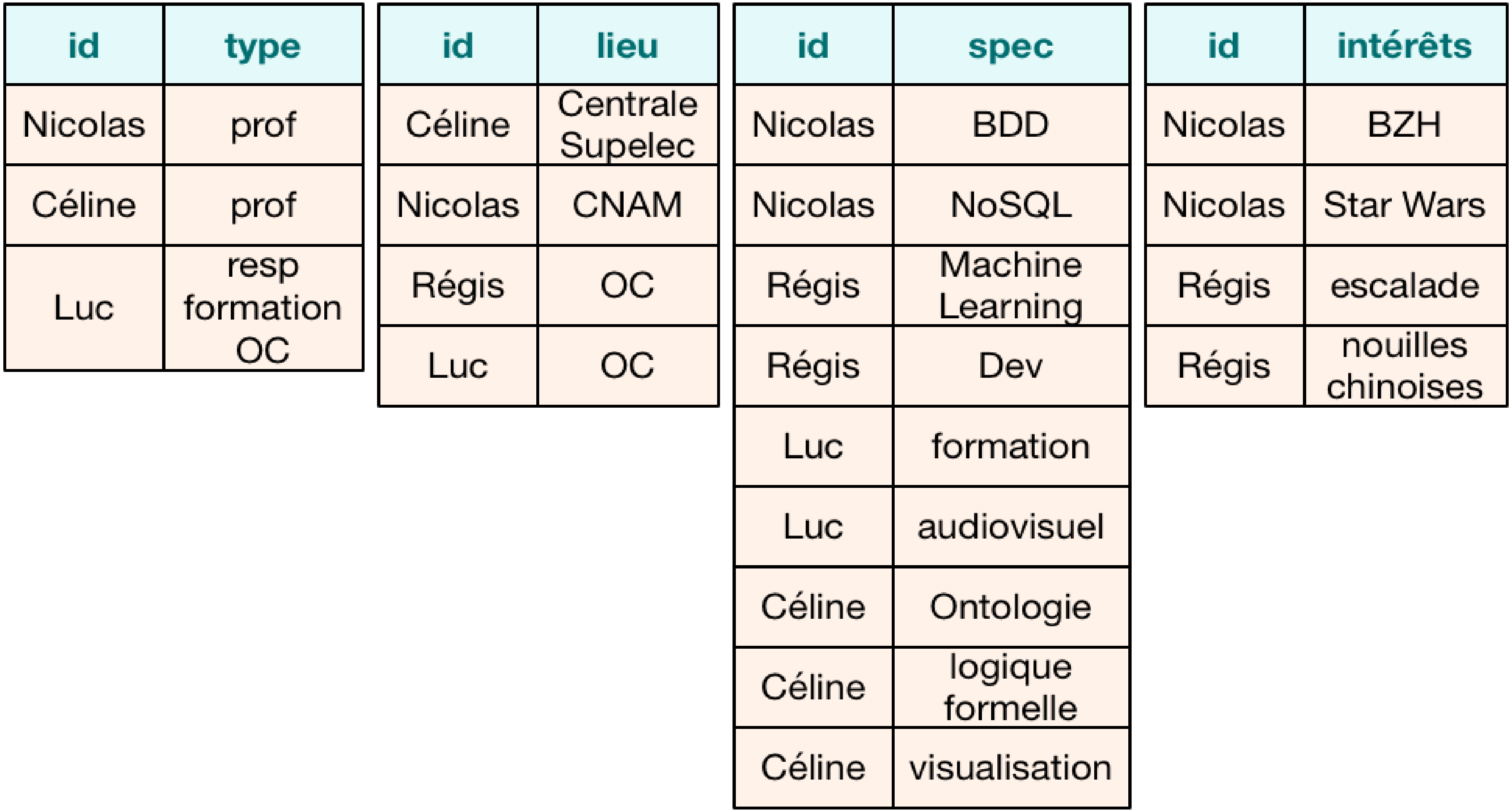
Types of NoSQL Databes : Wide column



id	type	lieu	spec	intérêts
Nicolas	prof	CNAM	BDD, NoSQL	BZH, Star Wars
Régis		OC	Machine Learning, Dev	escalade, nouilles chinoises
Luc	resp formation OC	OC	formation, audiovisuel	
Céline	prof	CentraleSupelec	Ontologie, logique formelle, visualisation	

Line-oriented storage

Types of NoSQL Databes : Wide column



id	type
Nicolas	prof
Céline	prof
Luc	resp formation OC

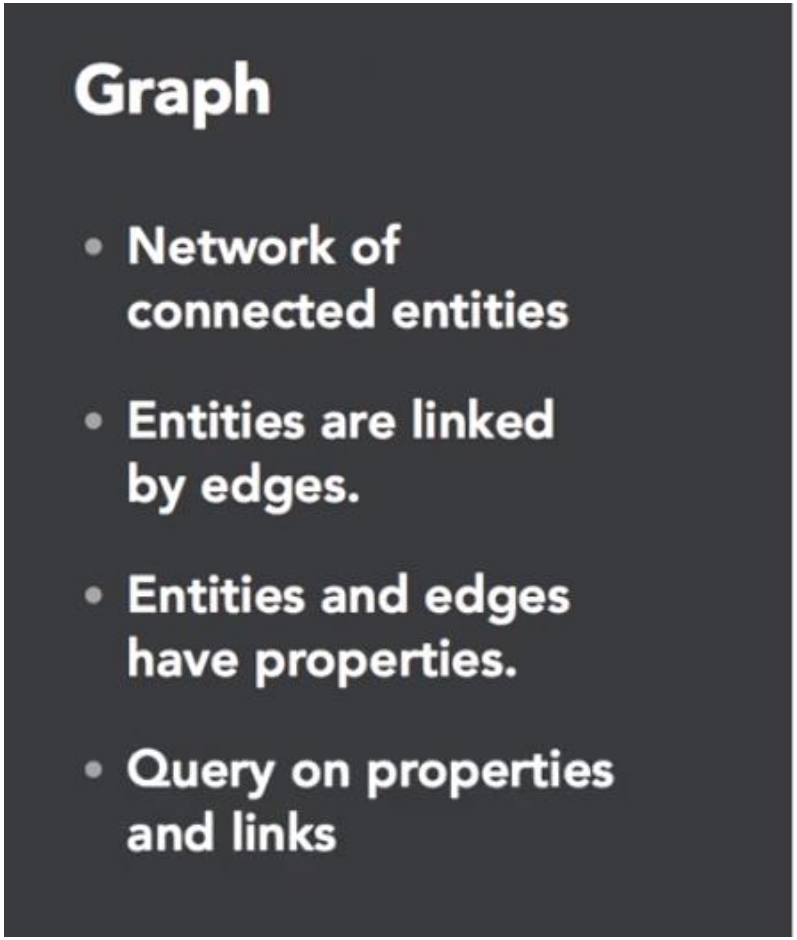
id	lieu
Céline	Centrale Supelec
Nicolas	CNAM
Régis	OC
Luc	OC

id	spec
Nicolas	BDD
Nicolas	NoSQL
Régis	Machine Learning
Régis	Dev
Luc	formation
Luc	audiovisuel
Céline	Ontologie
Céline	logique formelle
Céline	visualisation

id	intérêts
Nicolas	BZH
Nicolas	Star Wars
Régis	escalade
Régis	nouilles chinoises

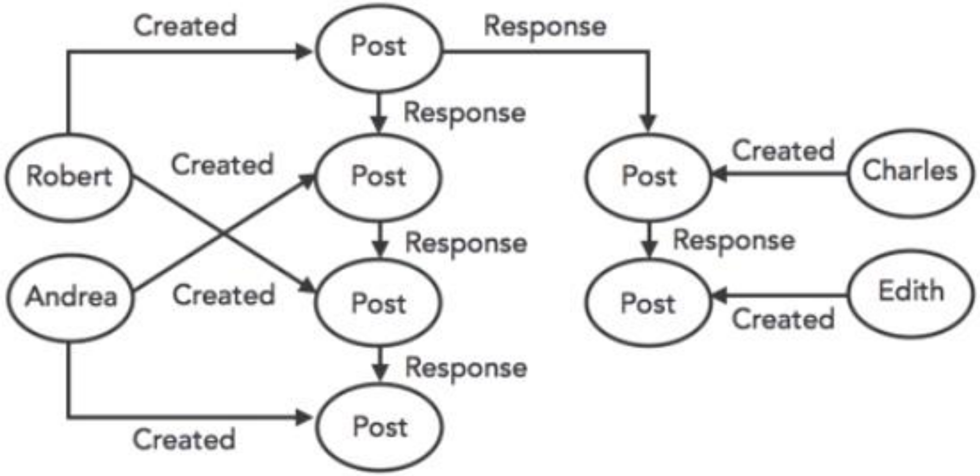
Column-oriented storage

Types of NoSQL Databes : Graph

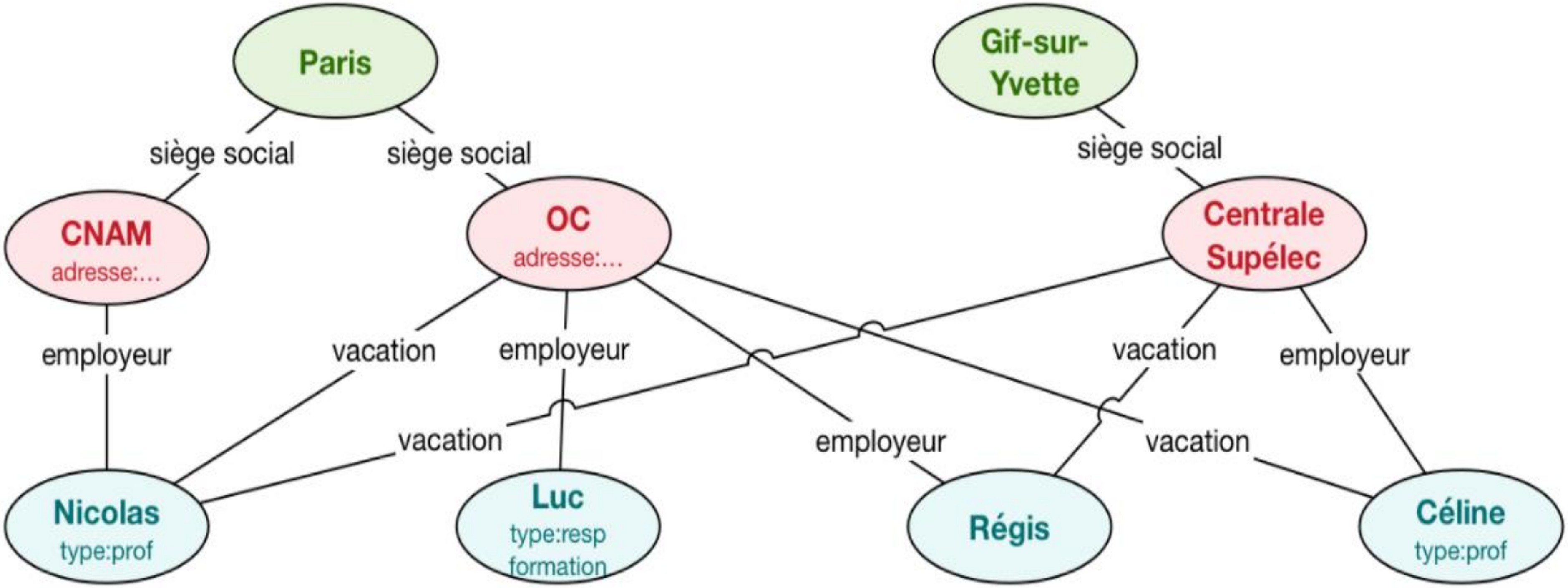


Graph

- Network of connected entities
- Entities are linked by edges.
- Entities and edges have properties.
- Query on properties and links



Types of NoSQL Databes : Graph



When to Use ?

The possibility of storing and recovering large quantities of data

Non structured data or data that change with time

The storage of relationships between elements is not important

Rapid prototypes or applications that need to be developed

SQL to NoSQL

SQL Terms/Concepts	MongoDB Terms/Concepts
Database	Database
Table	Collection/ bunch of documents
Row	Document
Column	Field
Index	Index
Primary key	_i d
Table joins	Embedded documents and linking

Conclusion

- Data management is at the heart of Big Data
- The 3Vs of 4Vs relevant throughout the entire workflow
- No one DB fits all
- Big Data is the future

References

[1] Agrawal, H. M, “**Cloudcv: Large-scale distributed computer vision as a cloud service**”, *In Mobile cloud visual media computing*, pp. 265-290, 2015.

[2] Limare, N. a.-M, “**The IPOL initiative: Publishing and testing algorithms on line for reproducible research in image processing**”, *Procedia Computer Science* , pp. 4:716-725, 2011.

[3] Joseph Redmon and Ali Farhadi, “ **YOLO9000 : Better, Faster, Stronger** ”, *Computer Vision and Pattern Recognition CVPR Conference,* 2017.

[4] Mahmoudi Sidi Ahmed, Belarbi Mohammed Amin, Mahmoudi Said, Belalem Ghalem, "**Towards a Smart Selection of Resources in the Cloud for Low-energy Multimedia Processing**" in Concurrency & Computation : Practice & Experience (2017)

[5] Belarbi Mohammed Amin, Mahmoudi Said, Belalem Ghalem, Mahmoudi Sidi, "**Web-based Multimedia Research and Indexation for Big Data Databases**« , *in CloudTech 2017 : The 3rd International Conference on Cloud Computing Technologies and Applications*, Morocco (2017)

[6] Mahmoudi Sidi, Manneback Pierre, "**Multi-CPU/Multi-GPU Based Framework for Multimedia Processing**« , *in IFIP Advances in Information and Communication Technology. Computer Science and Its Applications,* 456, 2015, 54-65 (2015)

References

[7] Mahmoudi Sidi, Ozkan Erencan, Manneback Pierre, Tosun Souleyman, "**Taking Advantage of Heterogeneous Platforms in Image and Video Processing**" *in Complex HPC book" , Wiley, 978-1-118-71205-4* (2014)

[8] Da Cunha Possa Paulo, Mahmoudi Sidi, Harb Naim, Valderrama Carlos, "**A New Self-Adapting Architecture for Feature Detection**" in *Lecture Notes in Computer Science, 978-1-4673-2257-7, 2012(22) Oslo, 643 - 646, 10.1109/FPL.2012.6339149* (2012)

[9] Mahmoudi Sidi, Manneback Pierre, Augonnet C., Thibault S., "**Traitements d'images sur architectures parallèles et hétérogènes**" *in Technique et Science Informatiques, 31/8-10 - 2012, 8-9-10/2012, 1183-1203, 10.3166/tsi.31.1183-1203* (2012)

[10] Mahmoudi Sidi Ahmed, Manneback P., Augonnet C., Thibault S., "**Détection optimale des coins et contours dans des bases d'images volumineuses sur architectures multicoeurs hétérogènes**"*, in "20èmes Rencontres Francophones de l'Informatique Parallèle , France* (2011)

[11] Mahmoudi Sidi Ahmed, Manneback Pierre, « **Multi-GPU based Event Detection and Localization using High Definition Videos** » *", in "The 4th International Conference on Multimedia Computing and Systems (ICMCS'14)* , Marrakesch, Morocco (2014)

THANK YOU