



Big Data, Hadoop & Spark

ekito



Alexia Audevert

Data & Enthusiasm

@aaudevert



Guillaume Eynard-Bontemps

Responsable du Centre de Calcul du CNES
Spécialiste en traitement de données distribuées
5 ans sur Hadoop/Spark, 3 ans sur Dask.

guillaume.eynard-bontemps@cnes.fr
@guillaumeeb sur github

SOMMAIRE

1

Big Data & son écosystème

- Introduction
- Ecosystème
- Data Lake
- Cas d'utilisation

2

Hadoop

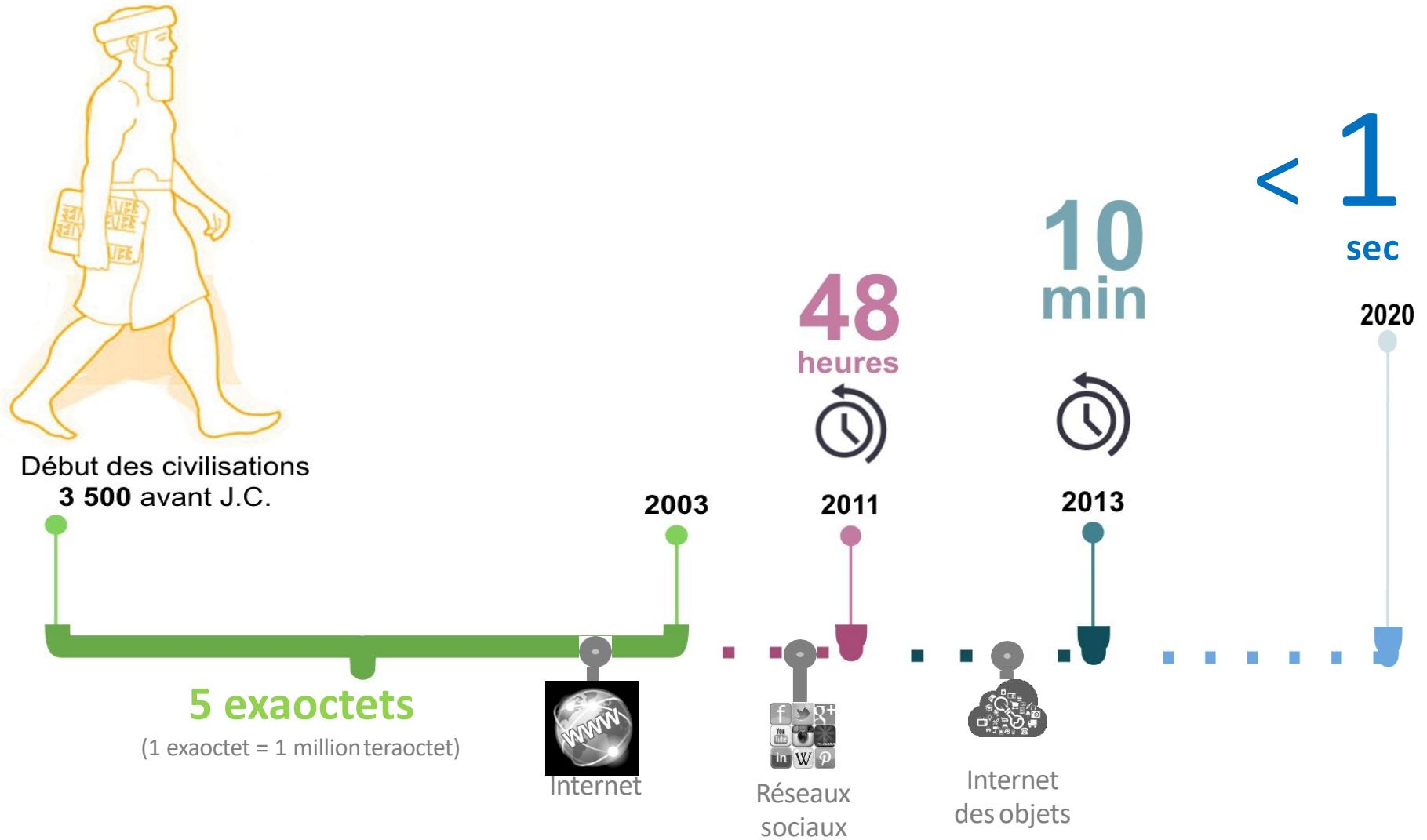
3

Spark

4

Conclusion & Questions

L'ÉVOLUTION DES DONNÉES



4,000,000
MESSAGES PROCESSED

486,000
PHOTOS

26
NEW REVIEWS
POSTED ON YELP

120
NEW ACCOUNTS
OPENED ON
LINKEDIN

MORE THAN
140
SUBMISSIONS
ON REDDIT

MORE THAN
2,315,000
SEARCHES

 **3,125,000**
 **243,055**

MORE THAN
21,000,000
MESSAGES SENT

MORE THAN
195,000
MINUTES OF AUDIO CHATTING
ON WHATSAPP

70,000
VIDEO MESSAGES
SHARED



MORE THAN
69,500
HOURS OF
VIDEO WATCHED
ON NETFLIX

NETFLIX



MORE THAN
48,000
APPS DOWNLOADED
ON IPHONE



IN
60
SECO
ECONDS

GO-Globe™
CUSTOM WEB DEVELOPMENT

YouTube

MORE THAN
430,000
TWEETS SENT

AROUND
56,000
PHOTOS
UPLOADED

9,800
ARTICLES PINNED
ON PINTEREST

MORE THAN
28,000
SNAPS SENT
ON SNAPCHAT

MORE THAN
100
NEW DOMAINS
REGISTERED

14 NEW
SONGS ADDED
ON SPOTIFY

MORE THAN
2,700,000
VIDEO VIEWS AND
139,000 HOURS
OF VIDEO WATCHED

MORE THAN
300 HOURS
OF VIDEO ARE UPLOADED



Les challenges du Big Data

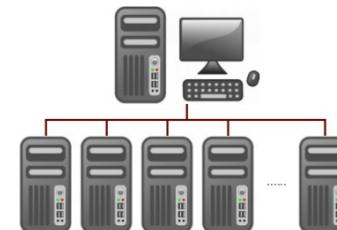
Traiter différentes VARIETES de données :

- Structurées
- Semi-structurées
- Non structurées



Traiter de gros VOLUMES de données :

- Plus de capacité de stockage
- Plus de puissance de calcul



Traiter les données plus rapidement (VELOCITE) :

- Fréquence plus importante de création, collecte et analyse des données
- Quasi temps réel / Temps réel

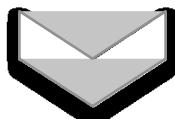


QUE SE CACHE DERRIÈRE LE BUZZWORD BIG DATA ?

Une variété de sources de données...



...des nouvelles technologies et des outils pour exploiter et analyser ces données

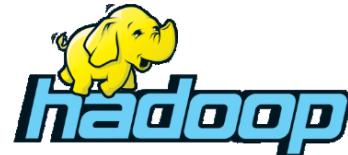


..et des outils & technologies pour les visualiser et les utiliser

Internal & External



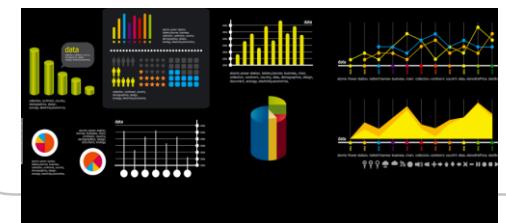
Calculators, Storage... Big Analytics



Platforms & Apps



Visualisation Interfaces



DÉFINITION DU BIG DATA

Le BIG DATA n'est pas une technologie



Mais la capacité de collecter, stocker, traiter, *valoriser*, *rapidement à moindre coût* de gros volumes de données où la taille unitaire d'une donnée est insignifiante.

Big Data vs HPC

High Performance Computing

- Niche market
- Highly optimized hardware
- Centralized data access
- Big Data handling (100s PB)
- “Bring data to the compute”

Big Data

- Widespread adoption (web industry)
- Commodity hardware
- Distributed data access
- Huge Data handling (EB)
- “Bring compute to the data”

Performance
Scalability

High
Performance
Data Analytics

Nouveaux Buzz : Cloud et IA

Cloud :

- Mutualisation de ressources Compute et Stockage → virtualisation
- Ressources vues comme infinies d'un point de vue utilisateur
- Stockage séparé du compute, mais forte proximité entre les deux
- Scalabilité horizontale des ressources.
- Kubernetes et stockage objet

IA ou Data Science :

- En fait, apprentissage machine
- Deep Learning de plus en plus
- Pas forcément beaucoup de données
- Big Data utile pour pré traitement souvent (Data wrangling)

SOMMAIRE

1

Big Data & son écosystème

- Introduction
- Ecosystème
- Data Lake
- Cas d'utilisation

2

Hadoop

3

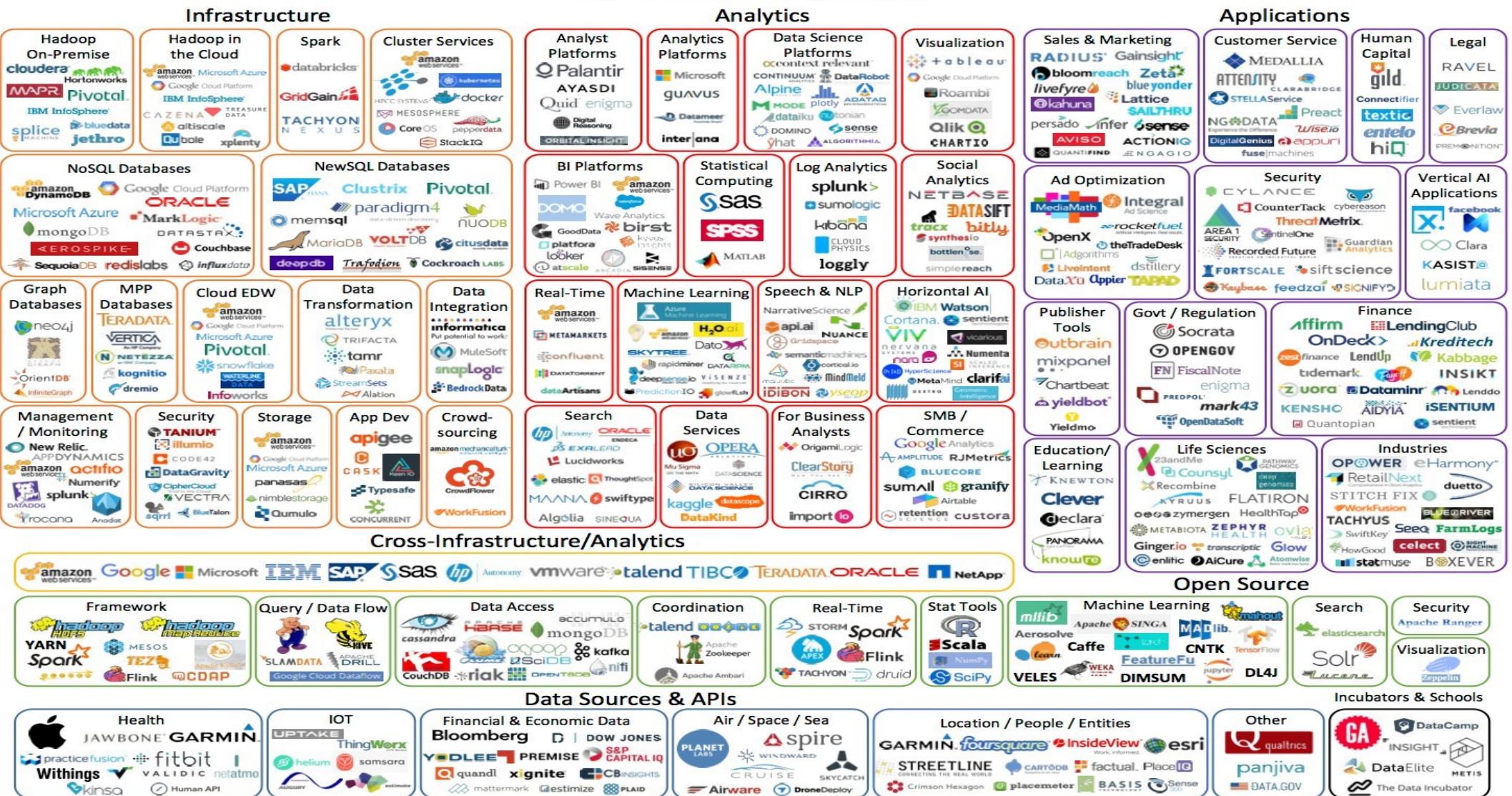
Spark

4

Conclusion & Questions

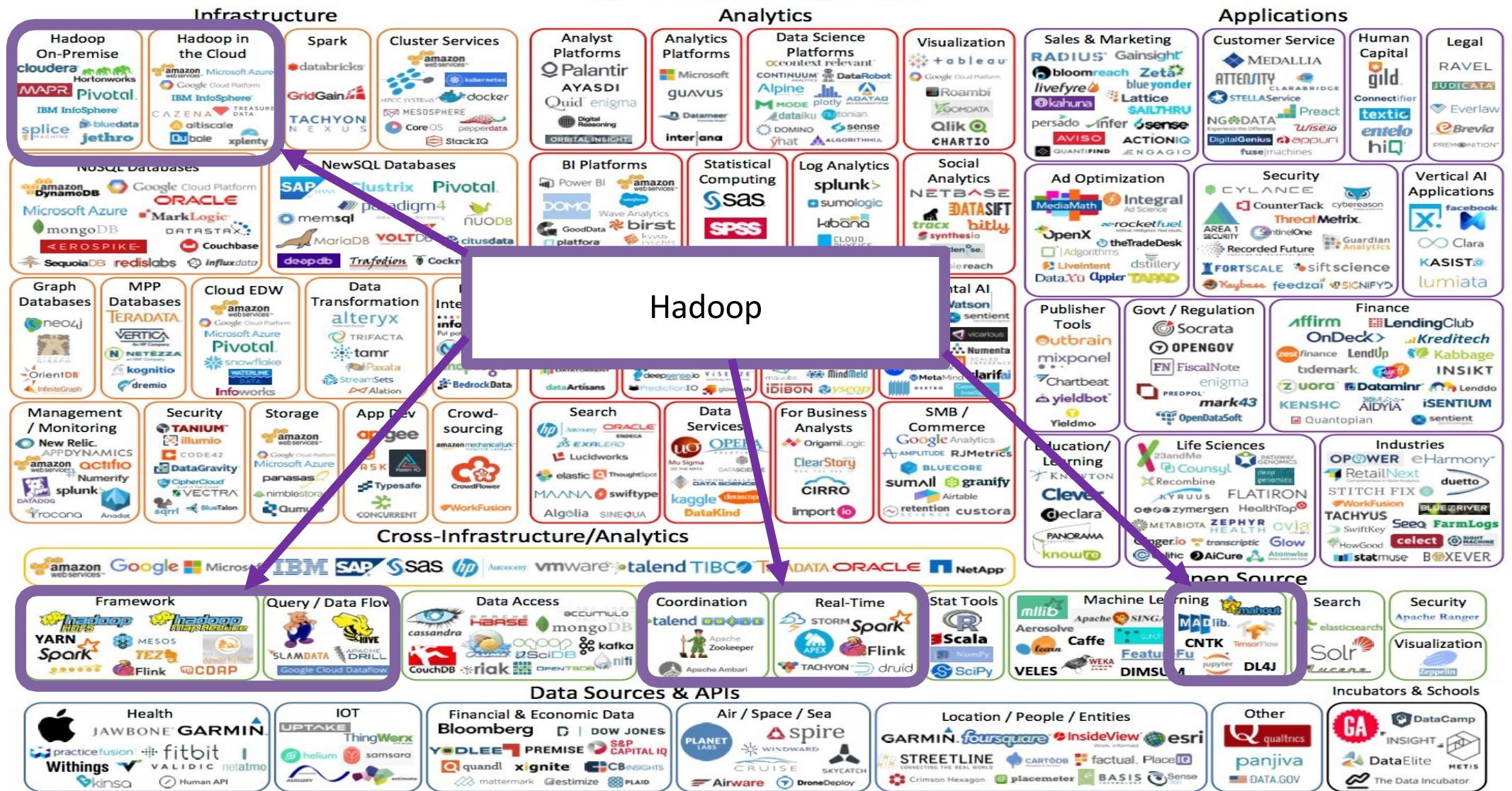
L'ÉCOSYSTÈME BIG DATA

Big Data Landscape 2016

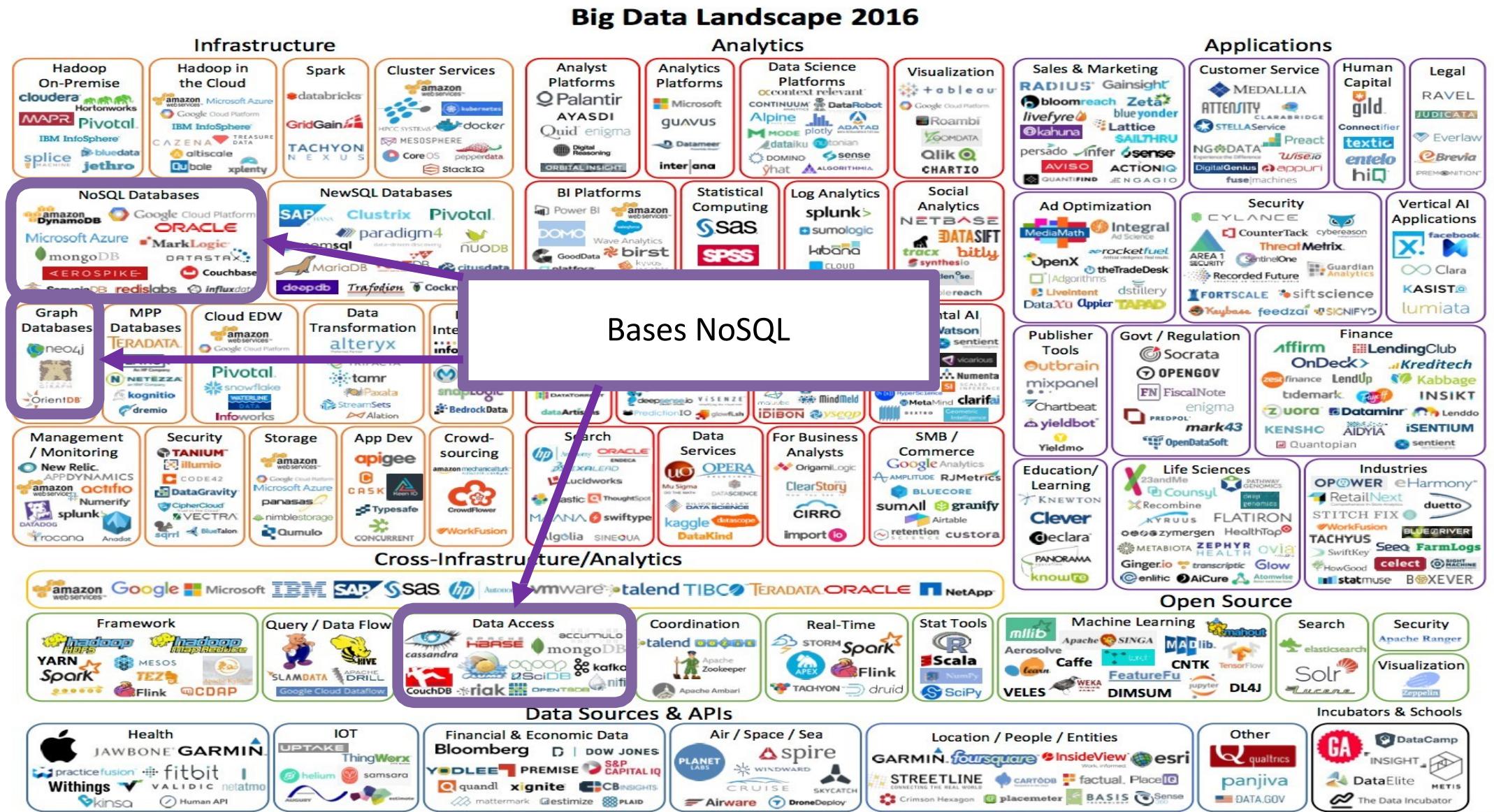


L'ÉCOSYSTÈME BIG DATA

Big Data Landscape 2016

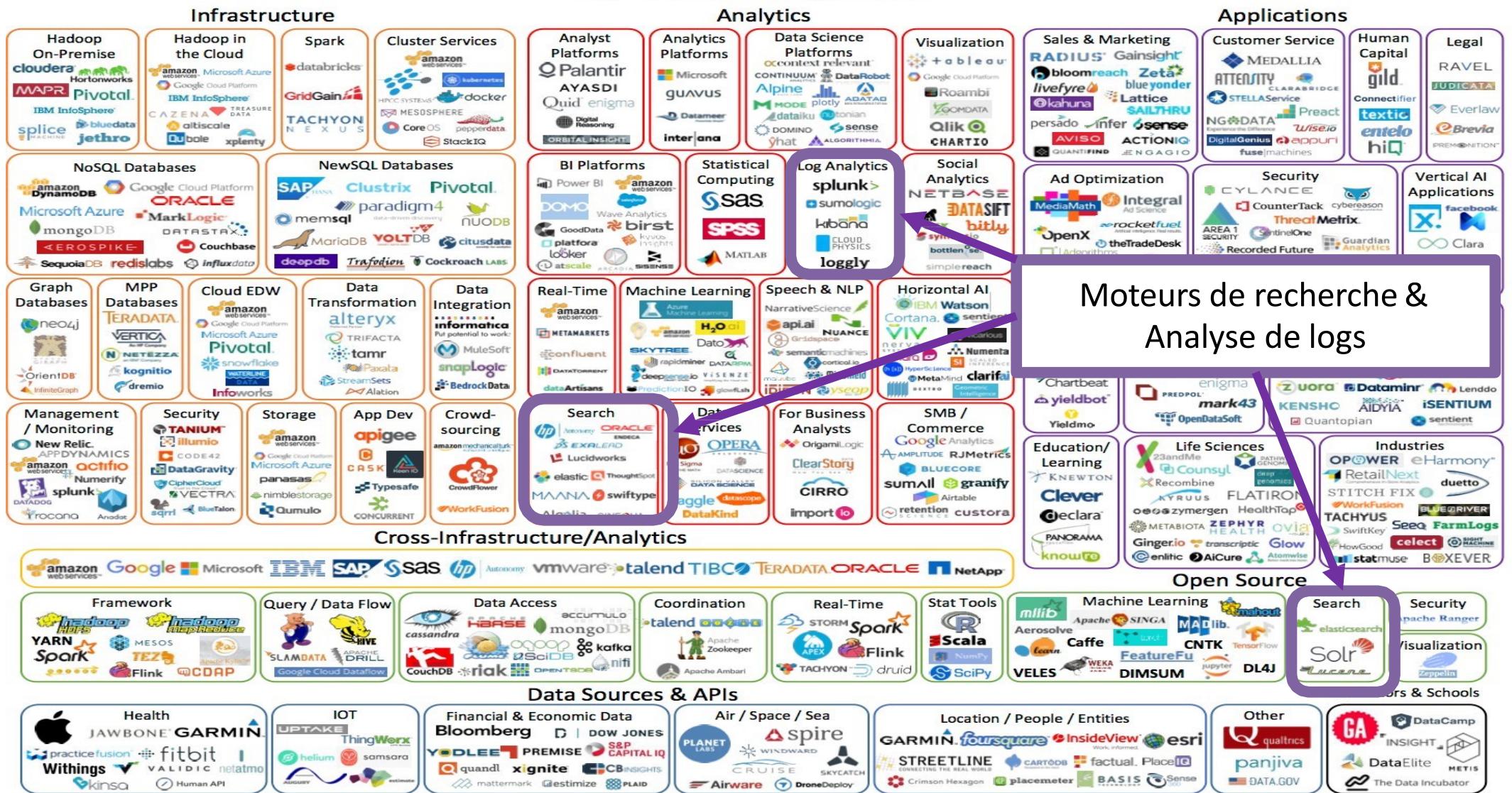


L'ÉCOSYSTÈME BIG DATA



L'ÉCOSYSTÈME BIG DATA

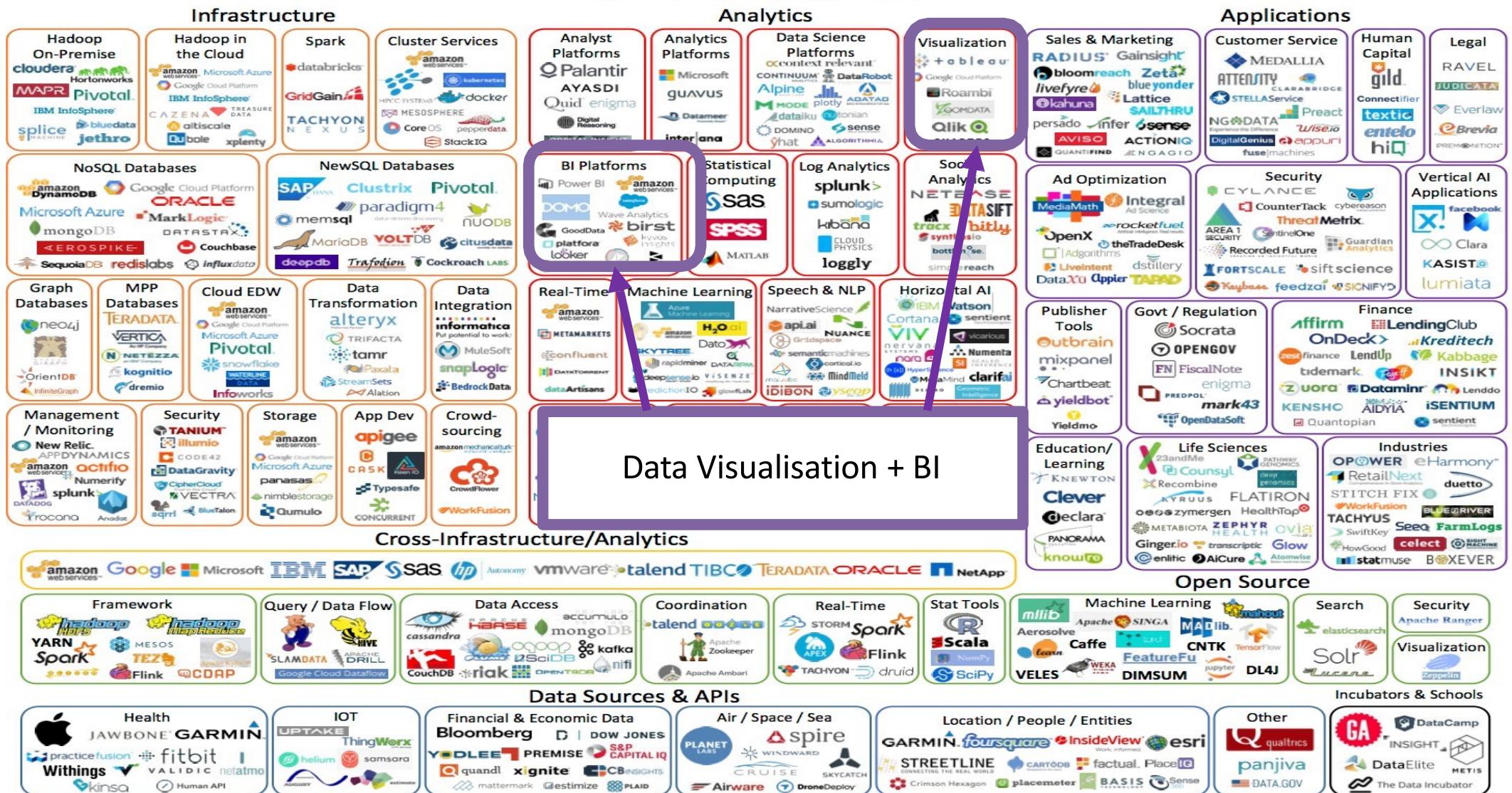
Big Data Landscape 2016



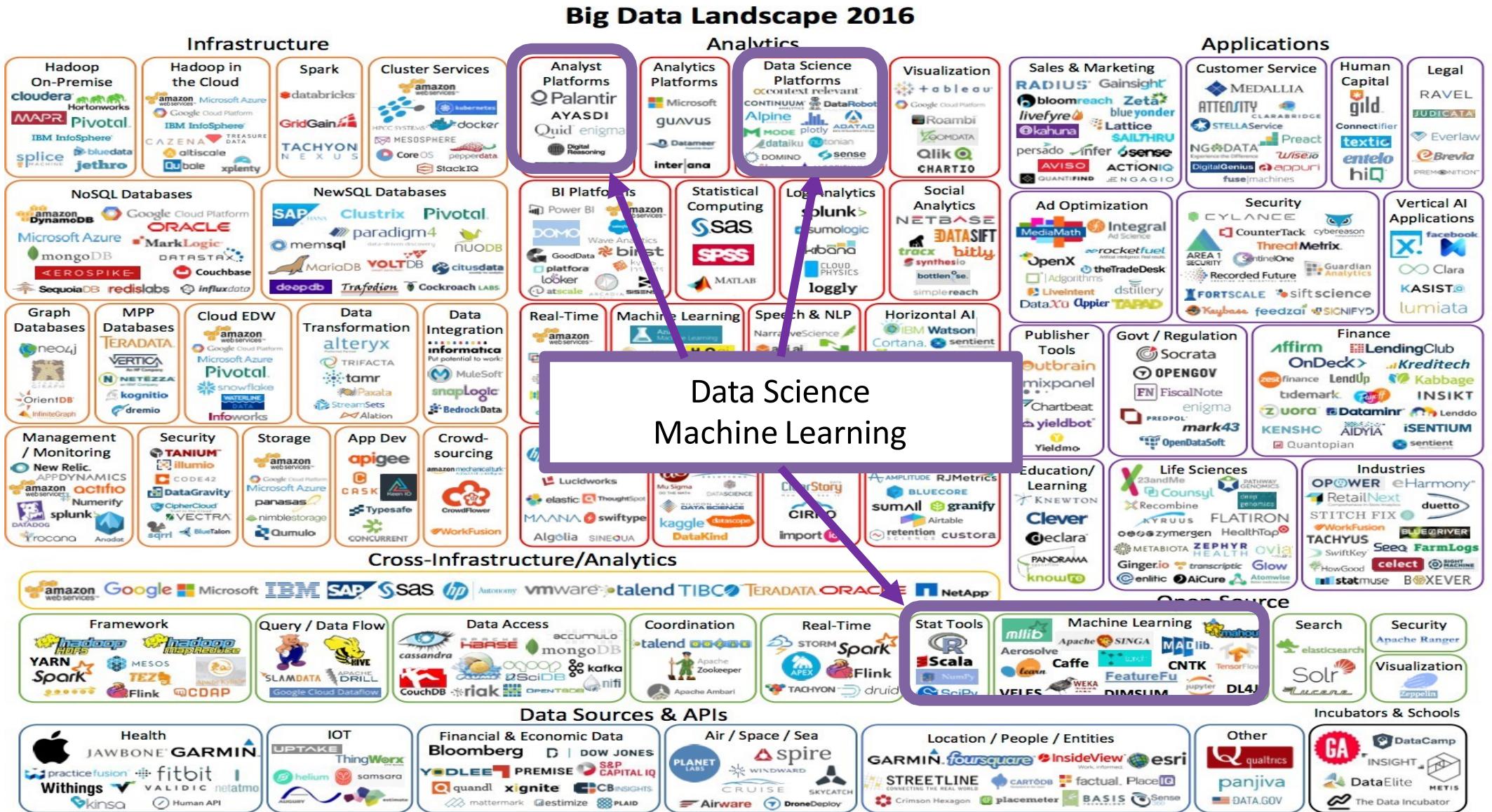
Moteurs de recherche & Analyse de logs

L'ÉCOSYSTÈME BIG DATA

Big Data Landscape 2016



L'ÉCOSYSTÈME BIG DATA



SOMMAIRE

1

Big Data & son écosystème

- Introduction
- Ecosystème
- Data Lake
- Cas d'utilisation

2

Hadoop

3

Spark

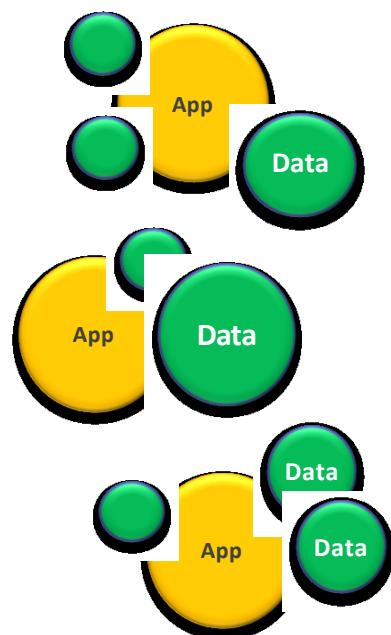
4

Conclusion & Questions

VERS UNE NOUVELLE GESTION DES DONNÉES

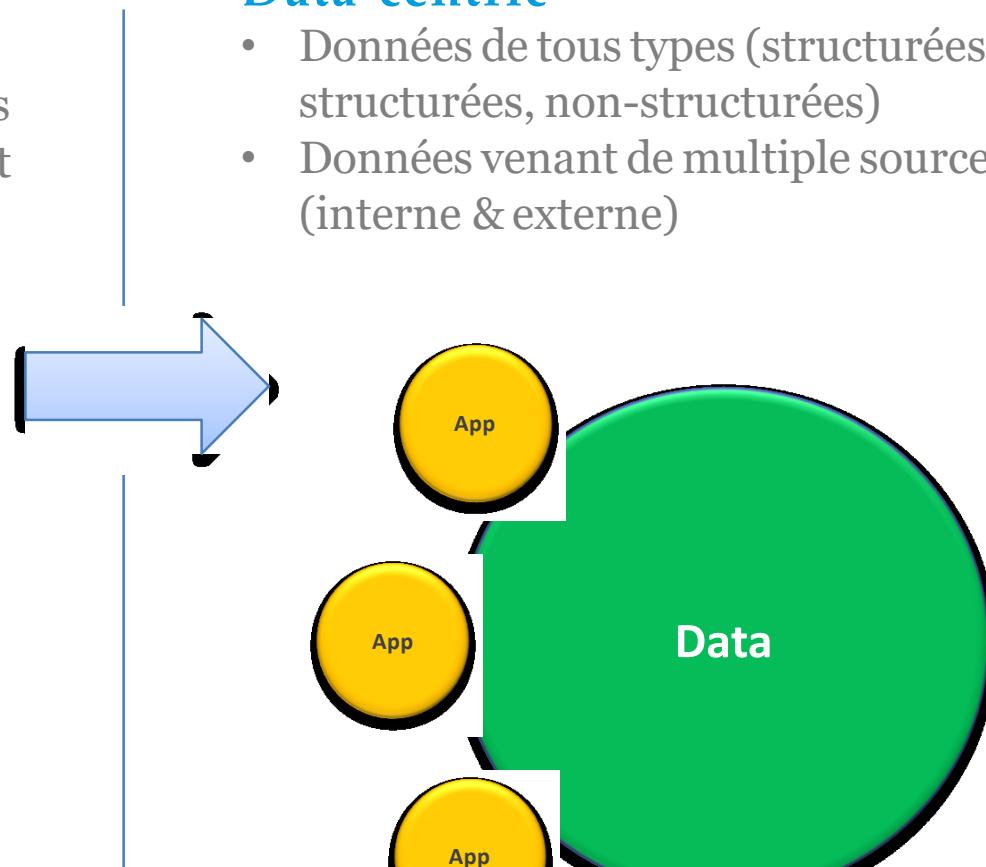
Process-centric

- Données structurées
- Données venant de sources Internes
- Données “importantes” uniquement
- Multiple copies des Données

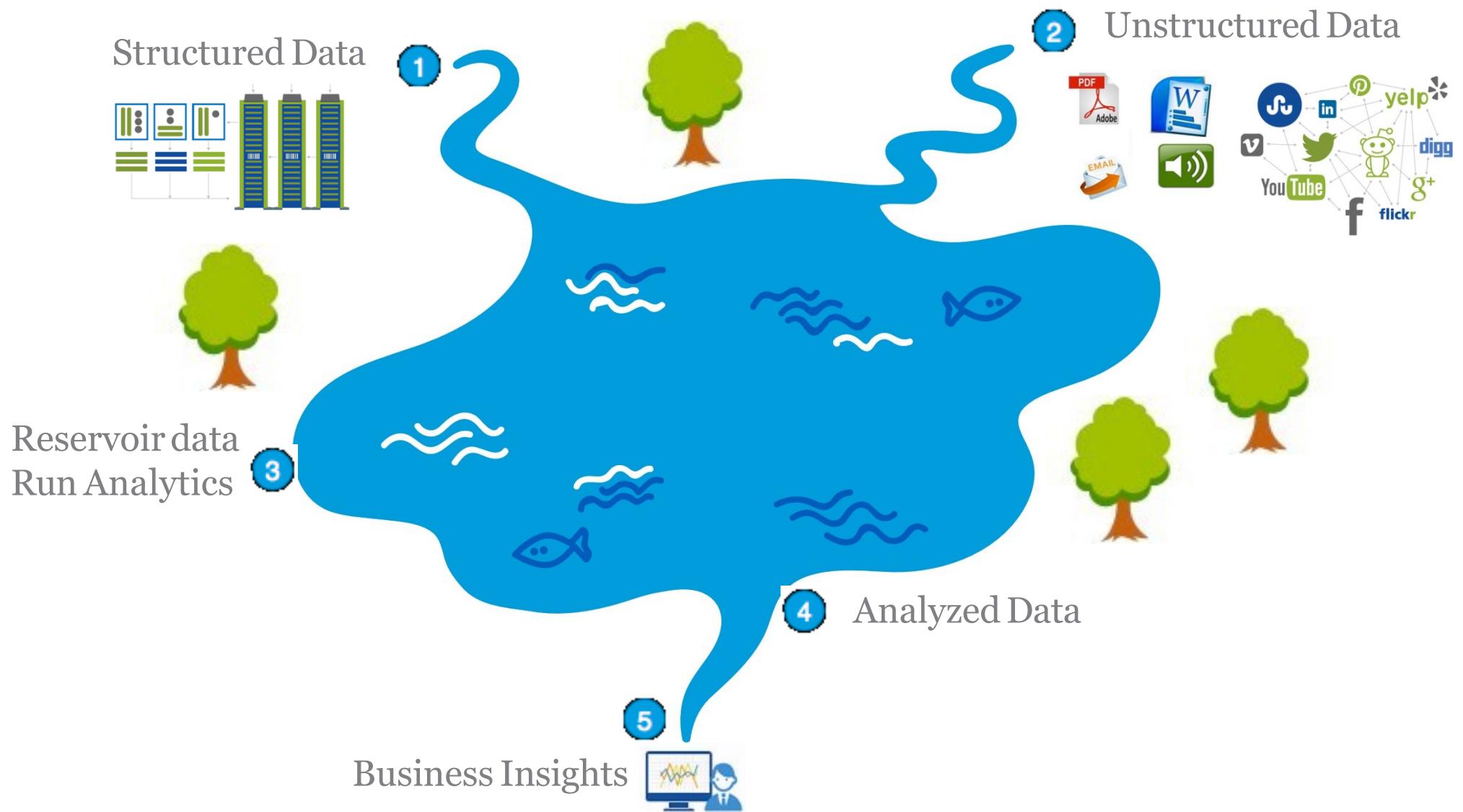


Data-centric

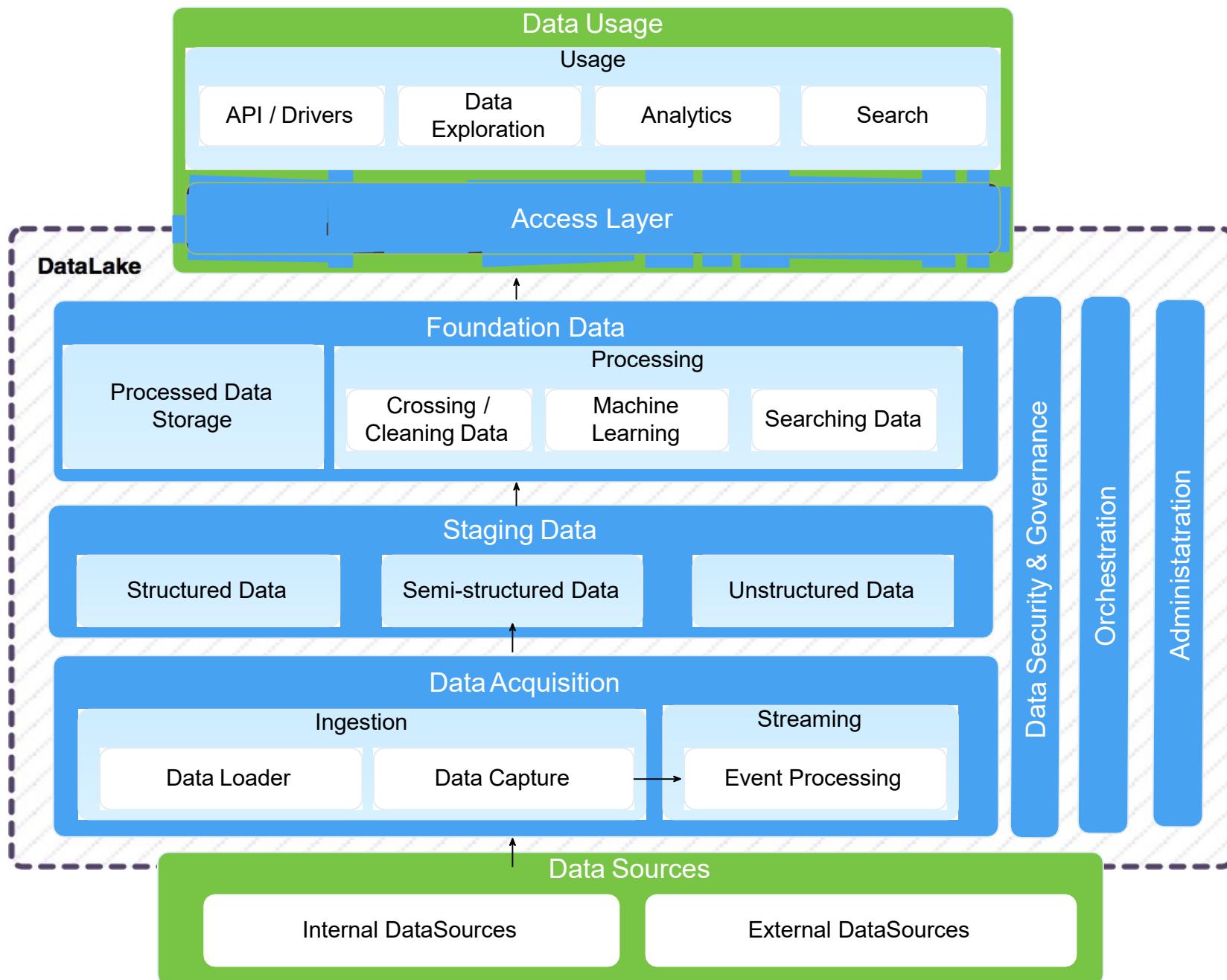
- Données de tous types (structurées, semi-structurées, non-structurées)
- Données venant de multiple sources de données (interne & externe)



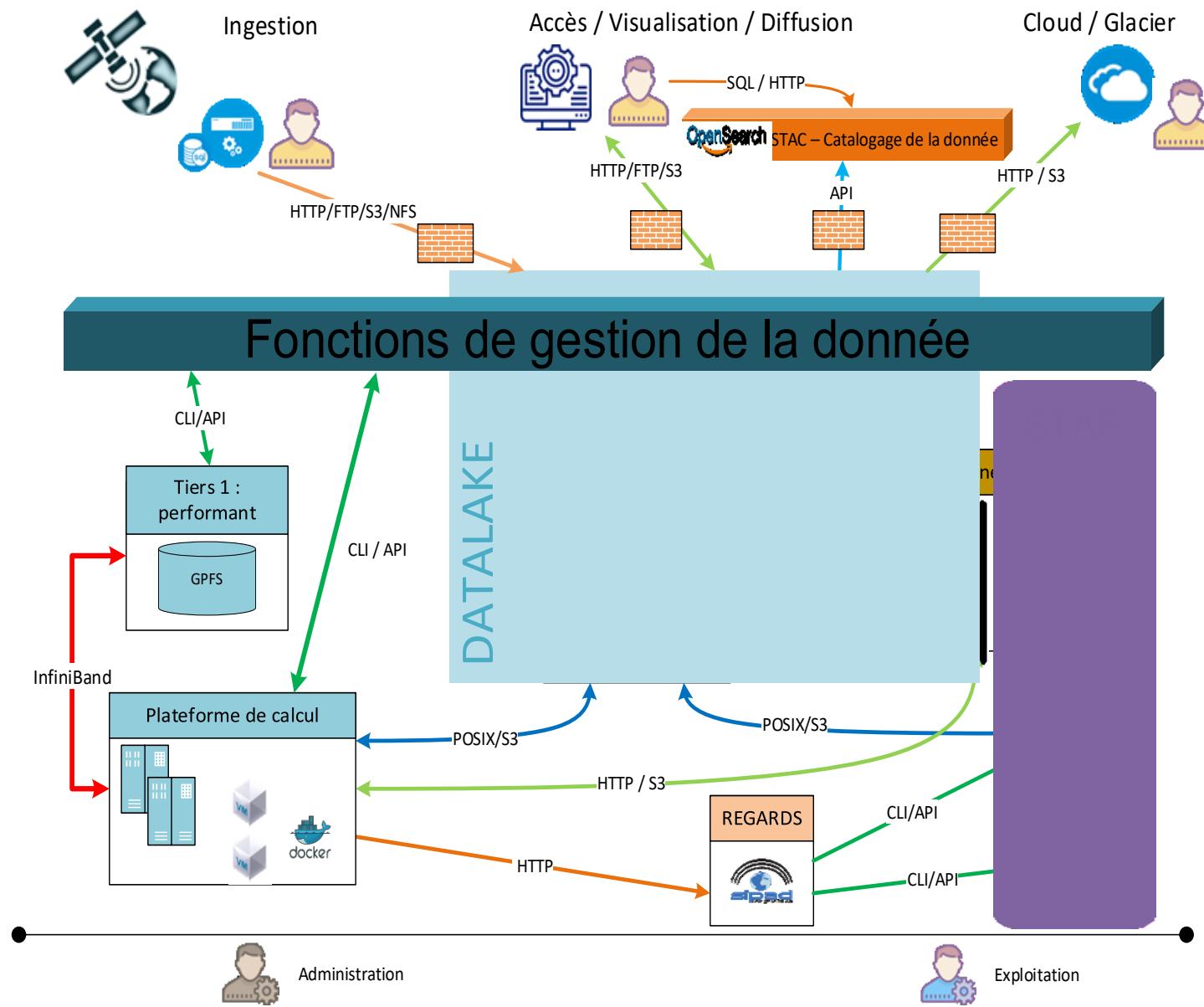
DATA LAKE



DATA LAKE



Exemple de Datalake au CNES



SOMMAIRE

1

Big Data & son écosystème

- Introduction
- Ecosystème
- Data Lake
- Cas d'utilisation

2

Hadoop

3

Spark

4

Conclusion & Questions

QUELQUES CAS D'UTILISATION

1

Réduction des couts:

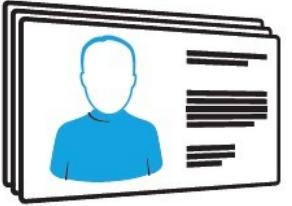
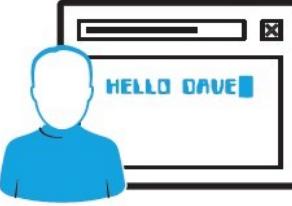
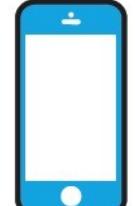
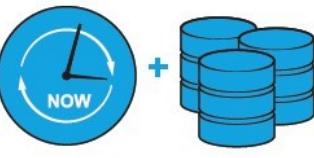
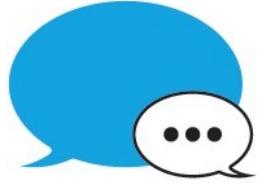
- Archivage
- Déchargement d'entrepôt dedonnées
- ETL (Extract-Transform-Load)
- Fail-Over

2

Elargir le champs des possibles :

- Analyser et tirer de la valeur des données de l'entreprise
- Analyser des données exogènes de l'entreprise et les corréler avec des données internes

QUELQUES CAS D'UTILISATION

<p>Profile Management</p> 	<p>Personalization</p> 	<p>360 Degree Customer View</p> 	<p>Internet of Things</p> 	<p>Mobile Applications</p> 
<p>Content Management</p> 	<p>Catalog</p> 	<p>Real Time Big Data</p> 	<p>Digital Communication</p> 	<p>Fraud Detection</p> 

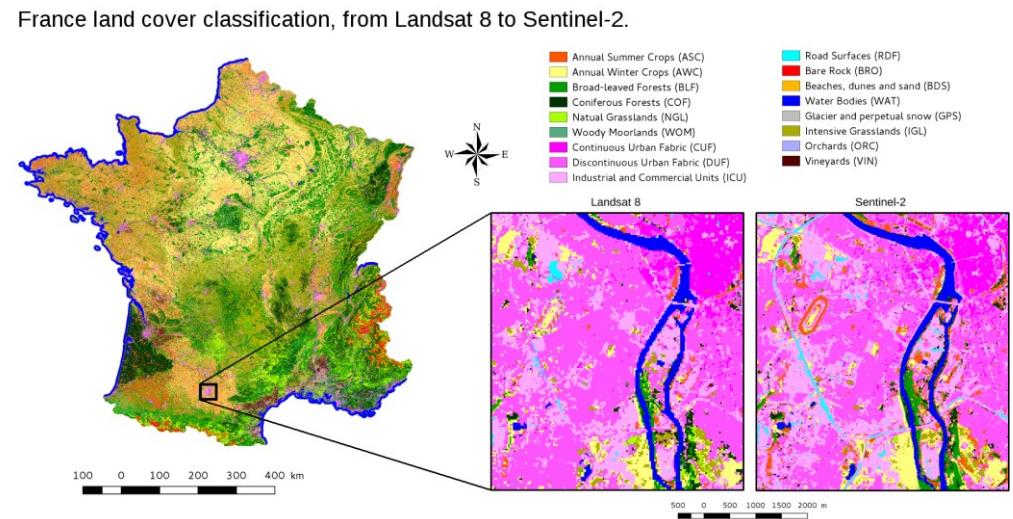
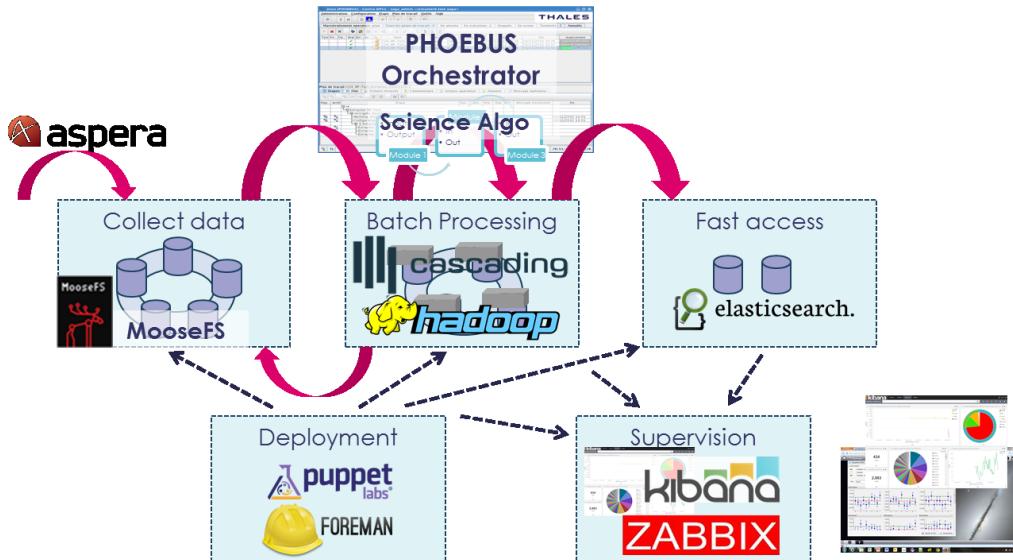
QUELQUES CAS D'UTILISATION

3

Exploration scientifique ou production de données :

- Production de données en flux
- Distribution des traitements sur ferme de calcul
- Exploration ou fouille de données
- Data Science

GAIA : 150TB en entrée, 3PB générés



20T de données d'entrée pour la France

SOMMAIRE

1

Big Data & son écosystème

2

Hadoop

- Introduction
- Composants Primaires
- Écosystème
- Distributions

3

Spark

4

Conclusion & Questions

INTRODUCTION A HADOOP

Framework OpenSource Apache Hadoop

- stocker et traiter de grands ensembles de données
- de façon distribuée (Cluster)
- sur du matériel standard



Composé de nombreux projets Apache Software Foundation

- Répondant à une fonctionnalité bien précise
- Associés à leur propre communauté de développeurs
- Possèdent leur propre cycle de développement



INTRODUCTION A HADOOP

Le projet Hadoop consiste en deux grandes parties :

- Stockage des données: **HDFS** (**H**adoop **D**istributed **F**ile **S**ystem)
- Traitement des données: **Map Reduce**



Principe

- **Diviser** et **sauvegarder** les données sur un **cluster**
- **Traiter** les données directement *là où elles sont stockées*
- **Scalabilité** : possibilité d'**ajouter/retirer** des **machines** au cluster

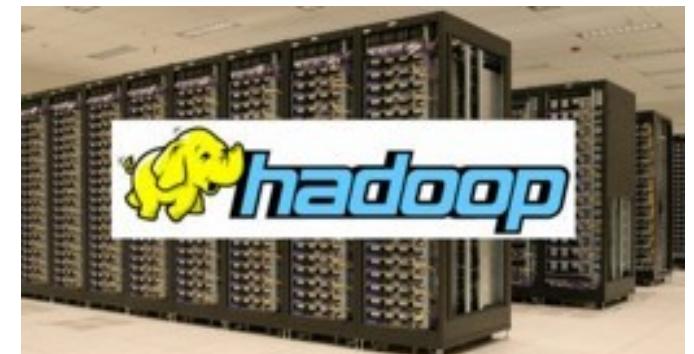
CLUSTER HADOOP

Cluster Hadoop

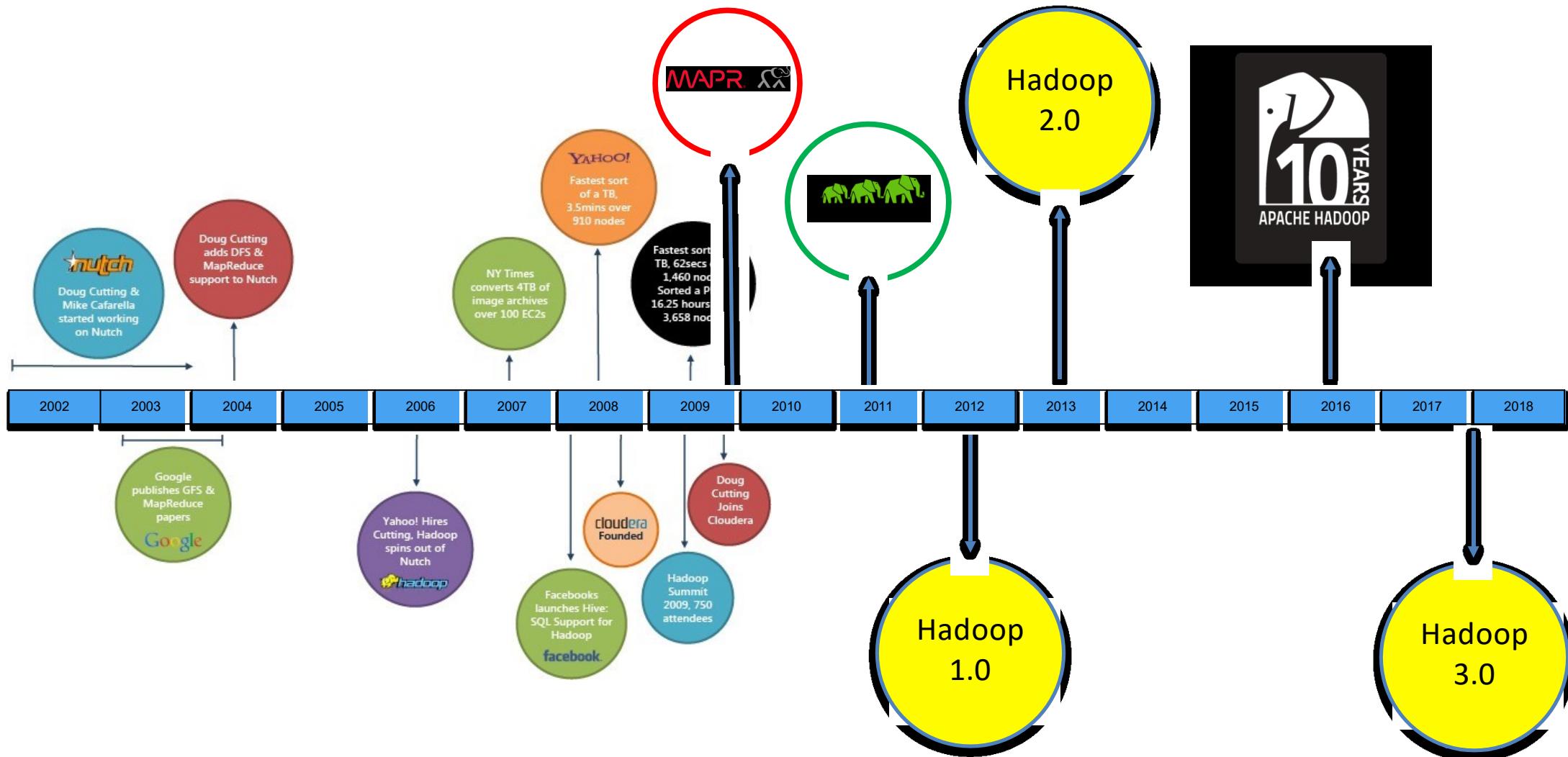
- *Ensemble de machines* : serveurs d'entrée de gamme (commodités)
- Système « *Shared Nothing* » : Le seul élément partagé est le réseau qui connecte les machines
- Une machine est appelé un « *Node* »

Un cluster est composé de :

- *Master Nodes*
 - Gèrent l'infrastructure
- *Worker/Slave Nodes*
 - Contiennent les données distribuées
 - Exécutent les traitements sur les données.



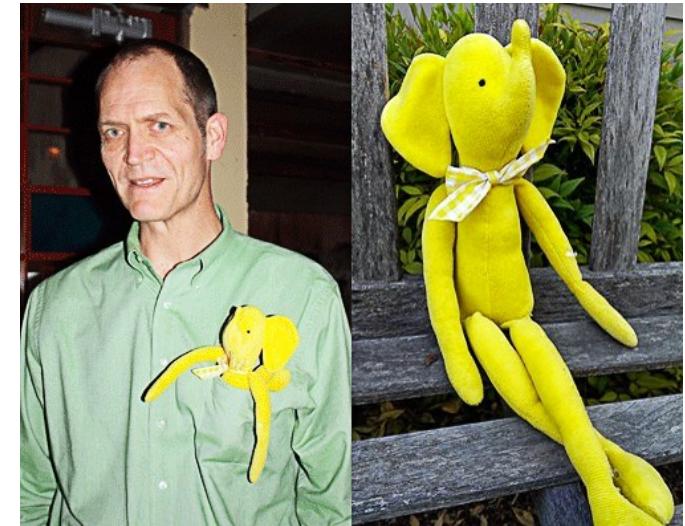
L'HISTOIRE D'HADOOP



QUIZZ!!!

Pourquoi le nom Hadoop ?

Nom de la peluche du fils de DougCutting



Pourquoi le nom Lucène ?

Deuxième nom de sa femme & Prénom de sa grand-mère

SOMMAIRE

1

Big Data & son écosystème

2

Hadoop

- Introduction
- Composants Primaires
- Écosystème
- Distributions

3

Spark

4

Conclusion & Questions

HDFS



HDFS est un système de fichiers distribué, extensible et portable.

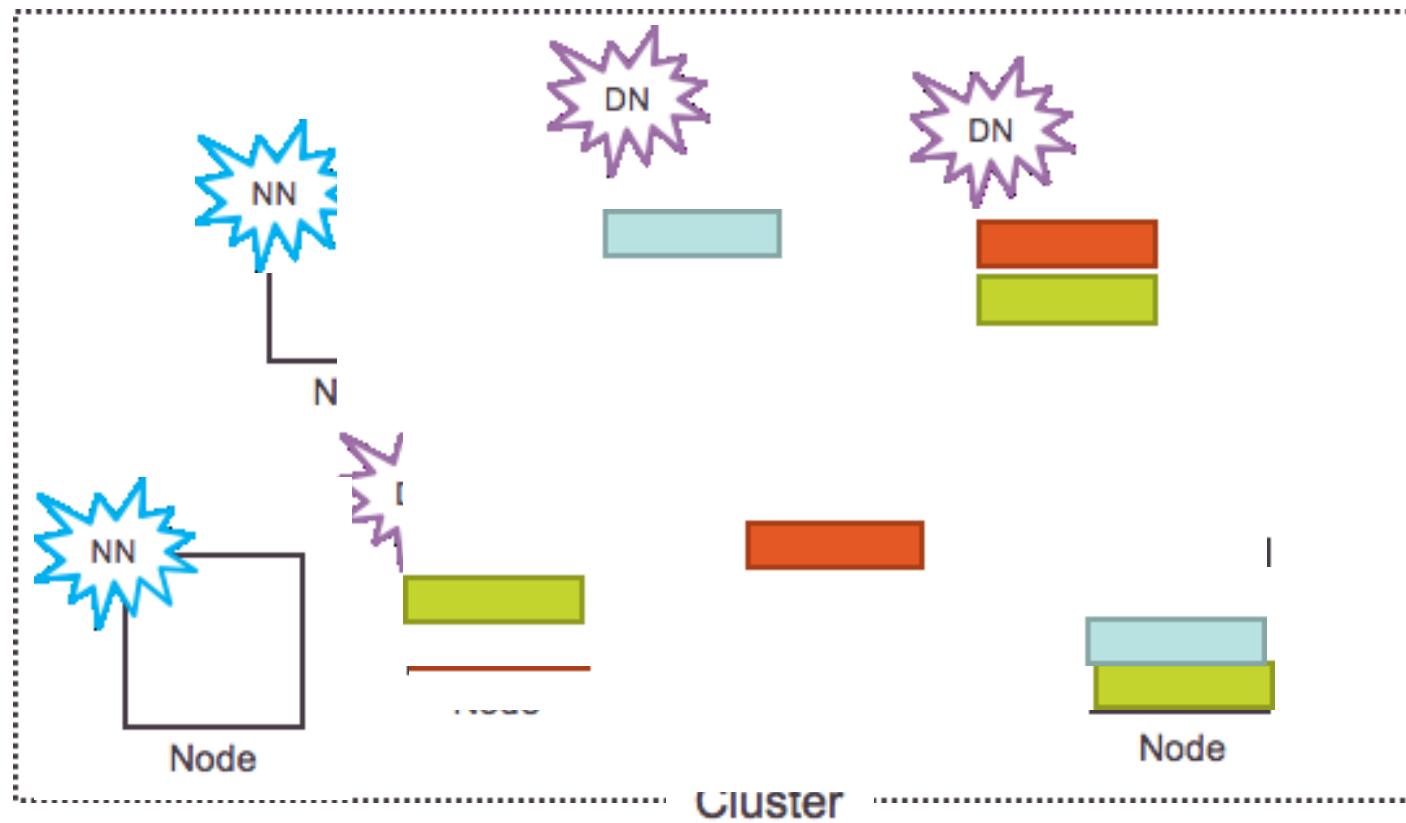
- Ecrit en **Java**
- Permet de **stocker** de très gros volumes de données (données structurés ou non) au sein d'un Cluster

Les données sont **découpées et distribuées** dans un cluster Hadoop :

- **Block Size** : par défaut 128 Mo
- **RéPLICATION Factor** : nombre de copies d'une donnée (par défaut 3 : 1 primaire et 2 secondaires)

Dans HDFS, les données sont de type « **write-once** »

HDFS



HDFS

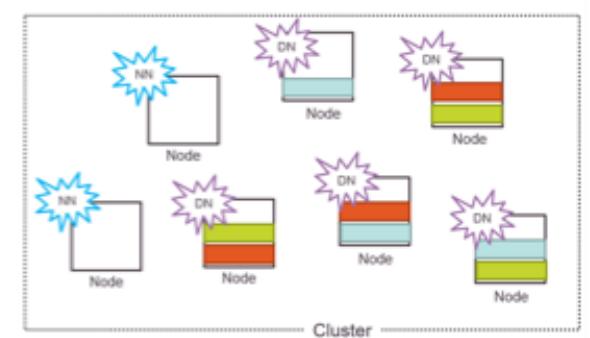


NameNode : Responsable de la localisation des données

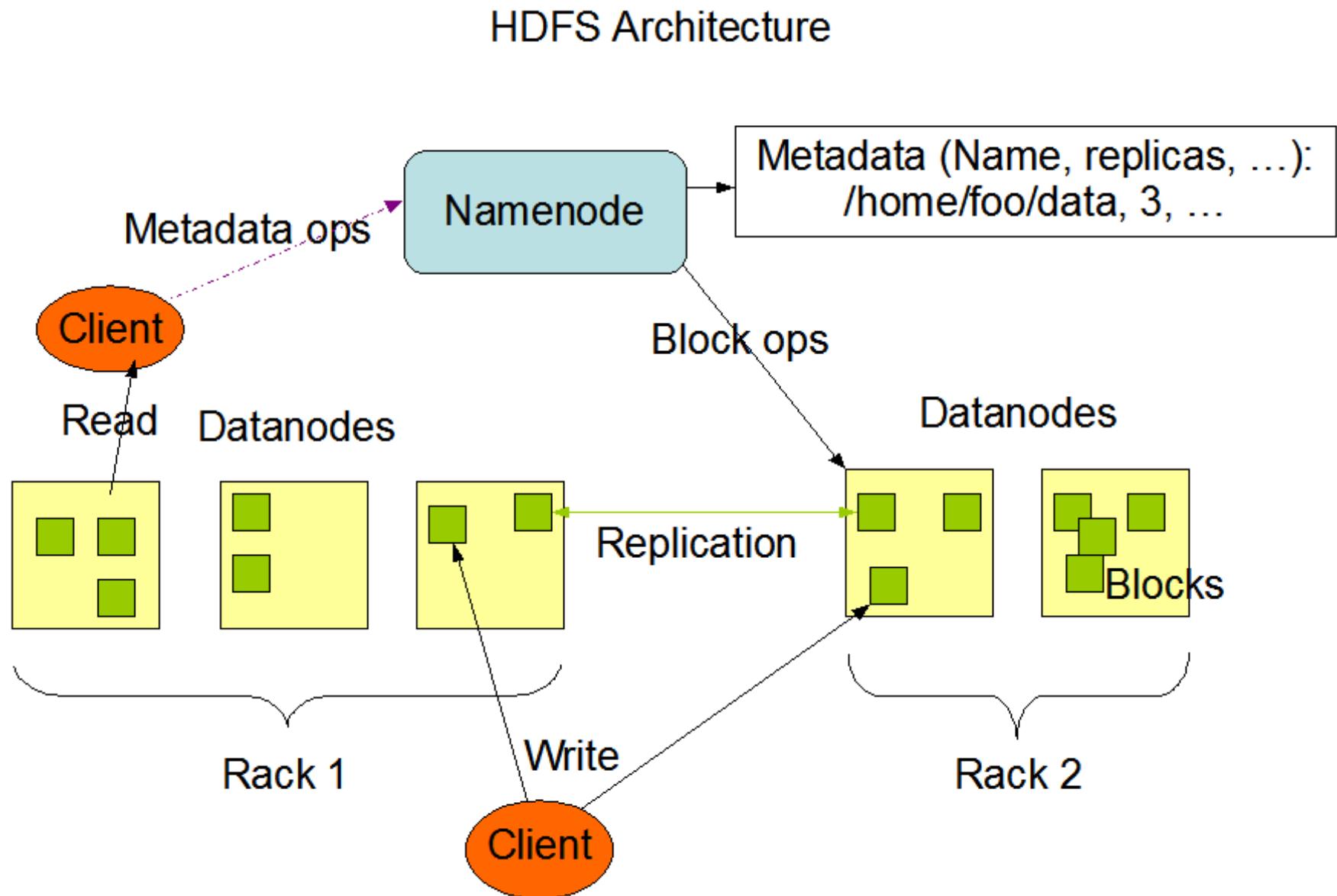
- **Démon** s'exécutant sur une machine séparée
- Contient des **méta-données**, exécute les opération de création, suppression, renommage de fichiers et répertoires.
- Permet de retrouver les nœuds qui contiennent les blocs d'un fichier
- NameNode est **duplicé**, non seulement sur son propre disque, mais également quelque part sur le système de fichiers du réseau (**Secondary NameNode**).

DataNode : Stocke et restitue les blocs de données

- **Démon** sur chaque nœud du cluster
- Opérations de lecture et d'écriture de données
- Opérations de création et suppression des blocks



Architecture HDFS



MAP REDUCE



Map Reduce :

- Concept issu des langages fonctionnels
- Utilisé par Google pour son outil de recherche Web
- *Co-localiser les données & les traitements*
- *Parallélisation automatique des programmes Hadoop*
-> Gestion transparente du mode **distribué**
- Traitement rapide des **données volumineuses**
- **Fault Tolerant** : Tolérance aux pannes basée sur la réPLICATION

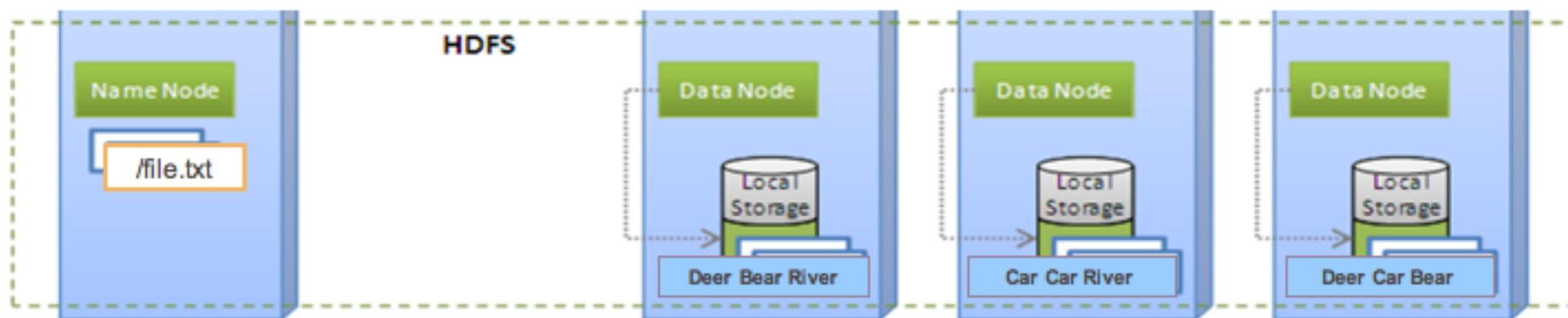
MAP REDUCE :WORD COUNT



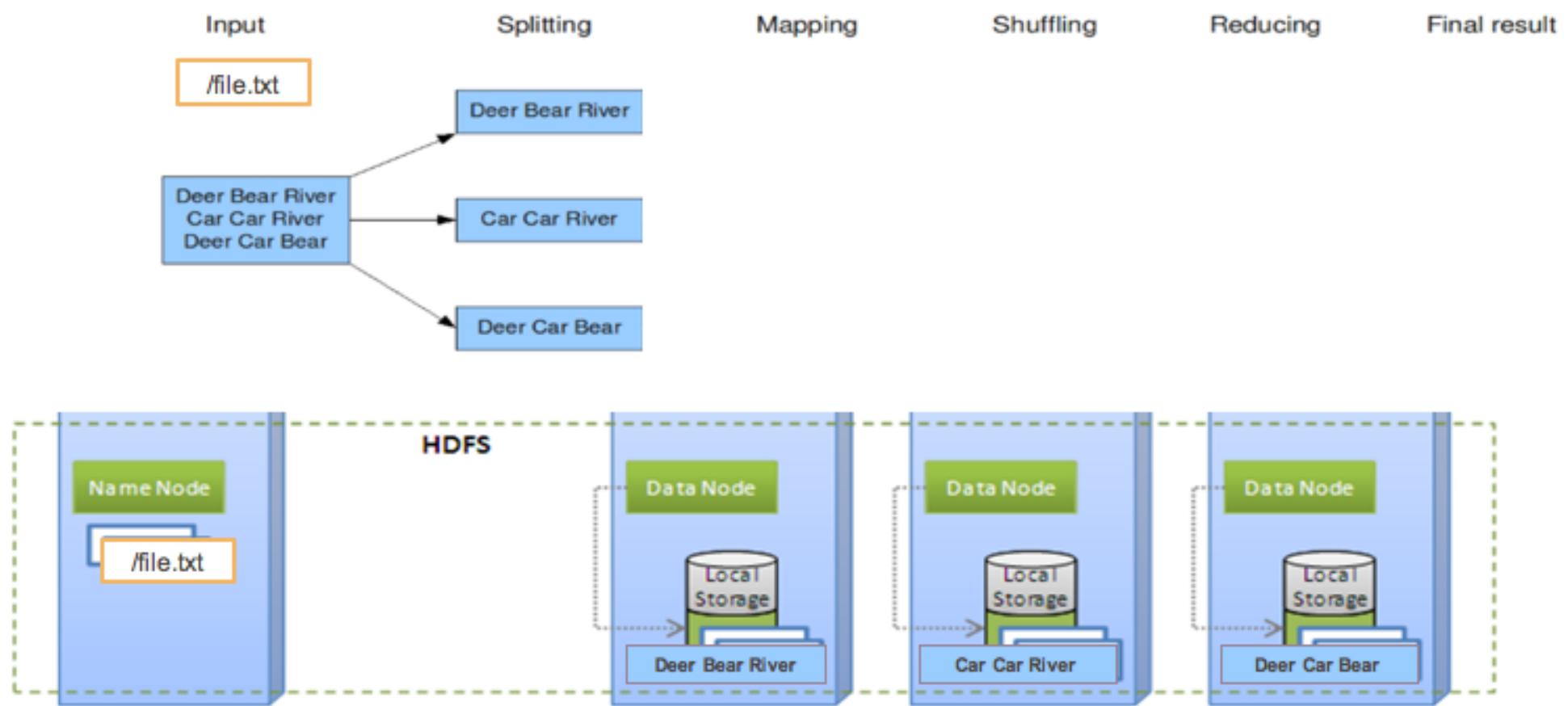
Input Splitting Mapping Shuffling Reducing Final result

/file.txt

Deer Bear River
Car Car River
Deer Car Bear



MAP REDUCE :WORD COUNT



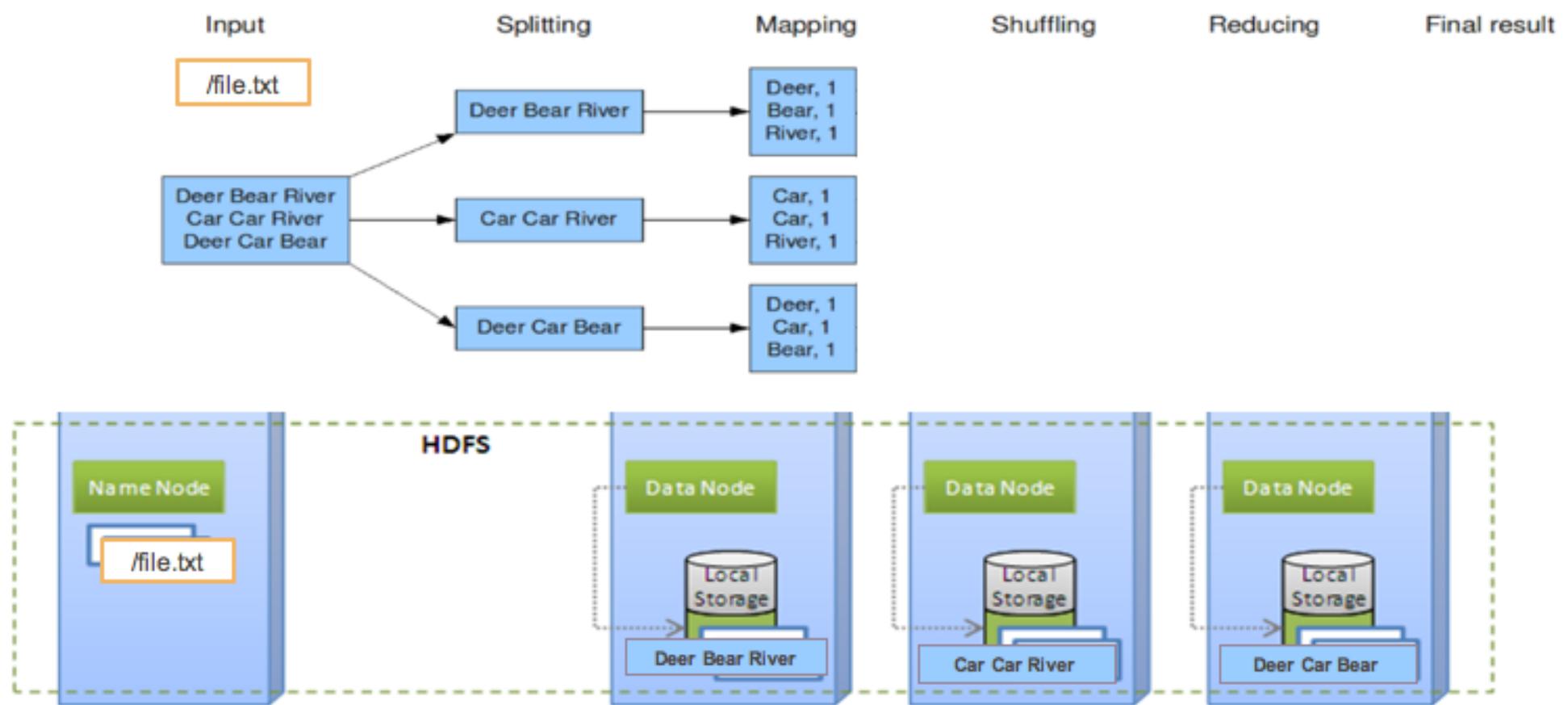
MAP REDUCE



Map: Décomposition d'une tache en un ensemble de tache plus petite produisant un sous ensemble du résultat final

- Composé de **Mappers**
- Fonctionnant en **parallèle**
- **Stockage sur disque** des données en entrée et sortie
- **Sorties** des Mappers = **enregistrements intermédiaires** sous forme d'un couple (clef, valeur)

MAP REDUCE :WORD COUNT



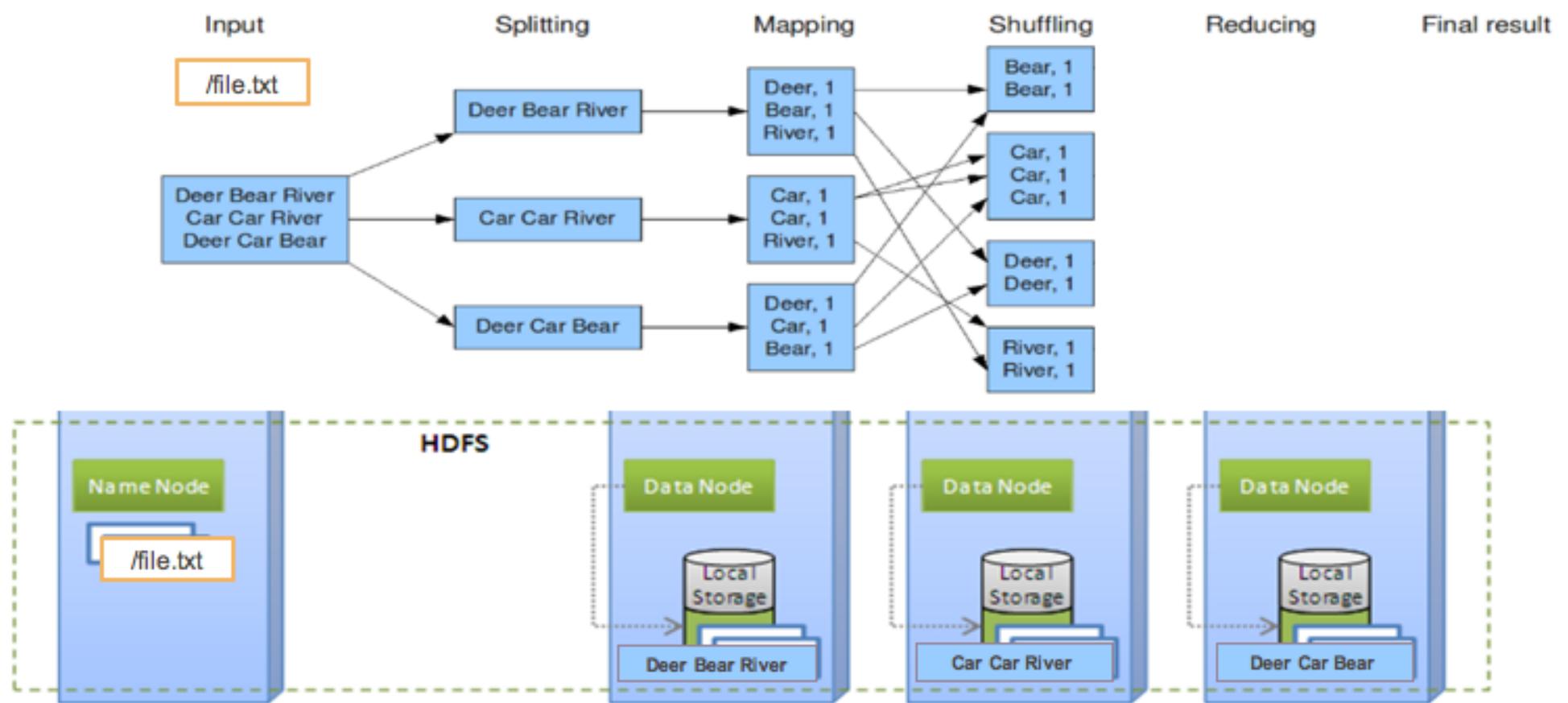
MAP REDUCE



Shuffle & Sort : Mélange et Tri

- *Tri par clef des données intermédiaires.*
- *Envoi des données ayant la même clef vers un seul et même reducer.*

MAP REDUCE :WORD COUNT



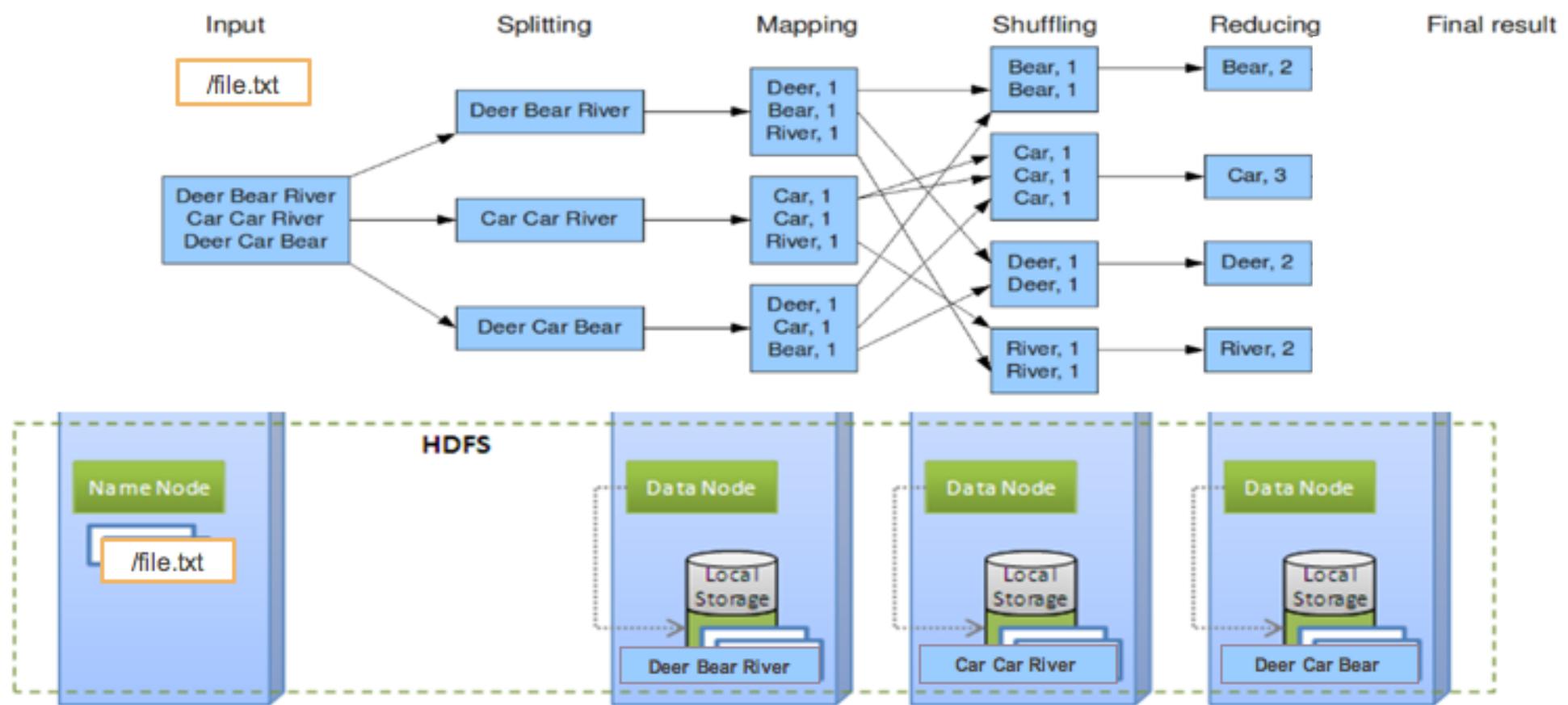
MAP REDUCE



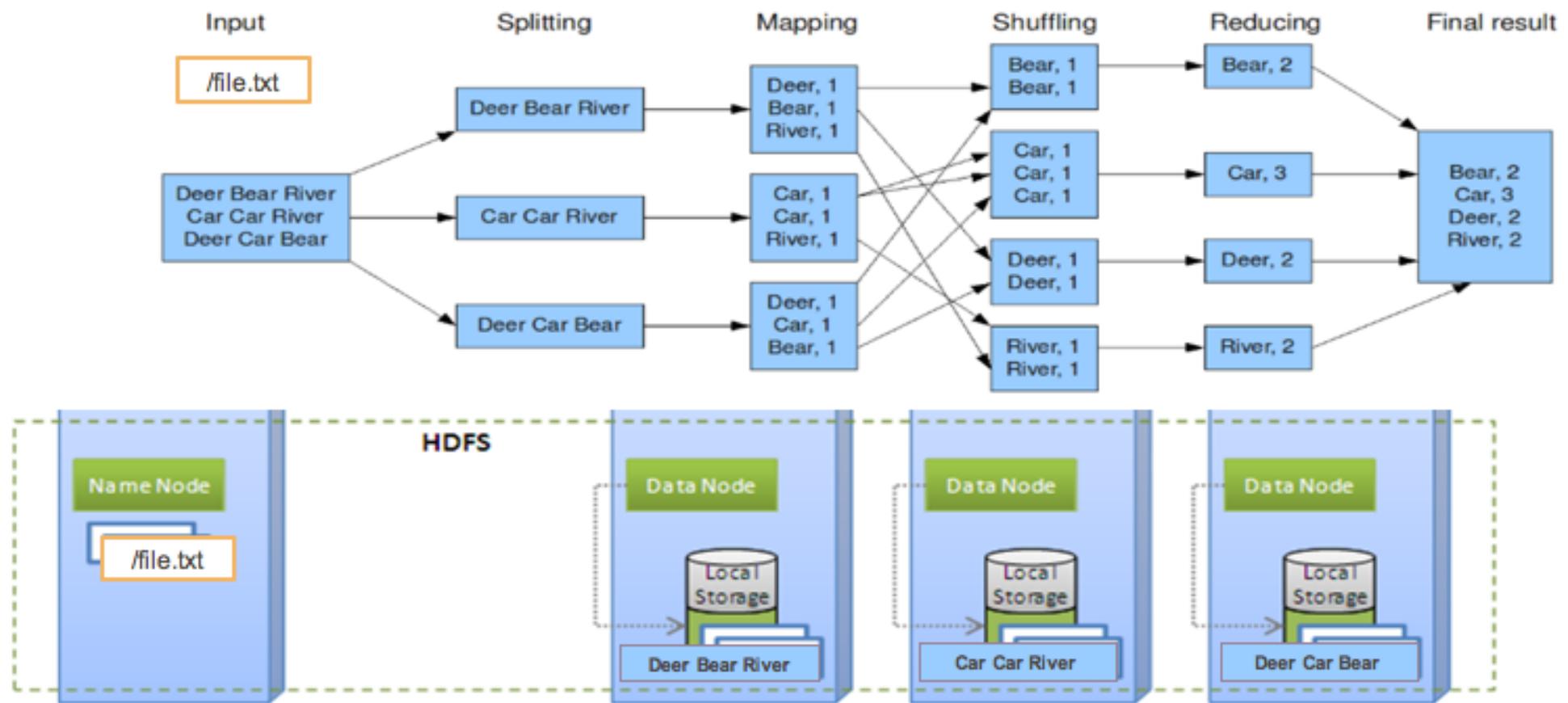
Reduce:

- Consolide (agrégation, filtre) les résultats issus du Mapper.
- Génère les **résultats finaux** et les écrit sur disque.

MAP REDUCE :WORD COUNT



MAP REDUCE :WORD COUNT



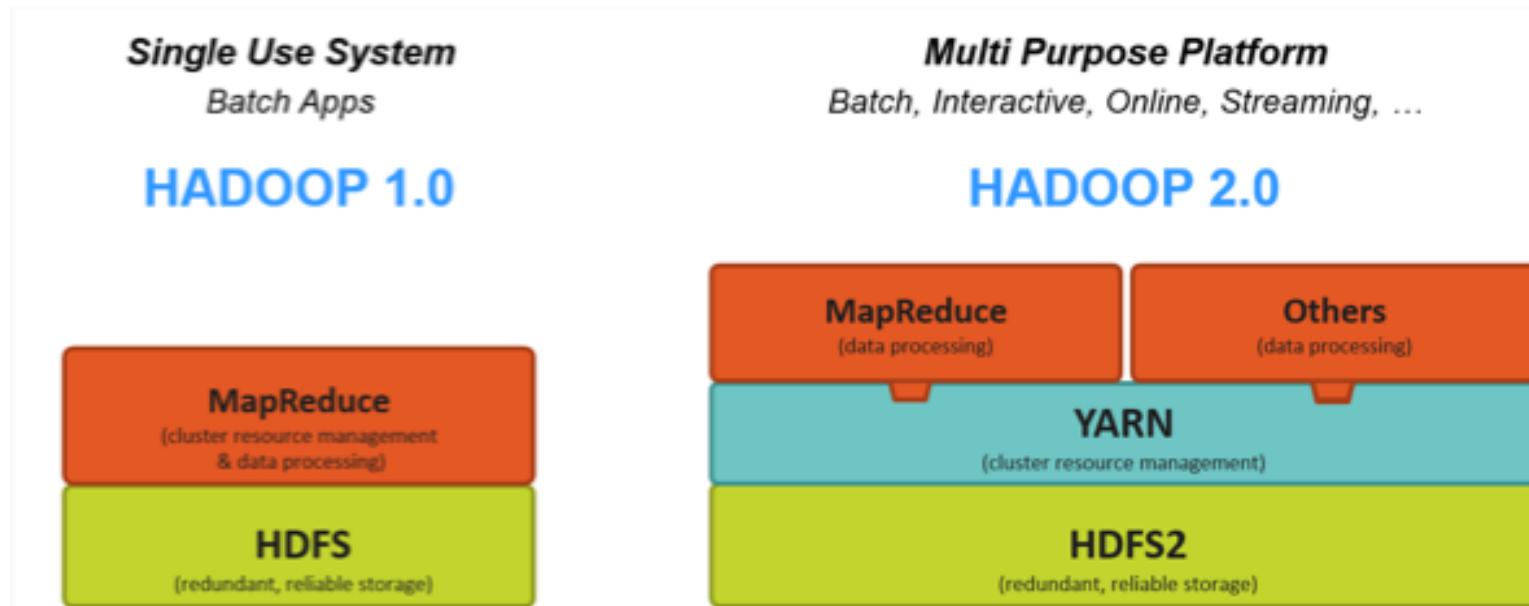
YARN

Yet-Another-Resource-Negotiator

Intégré à **Hadoop** depuis la *v2*

YARN apporte une séparation entre :

- Gestion de l'état du cluster et des ressources.
- Gestion de l'exécution des jobs.



Pour aller plus loin...

Object Store

FILE / BLOCK STORAGE

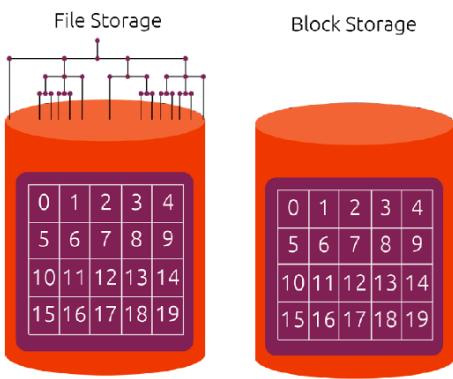
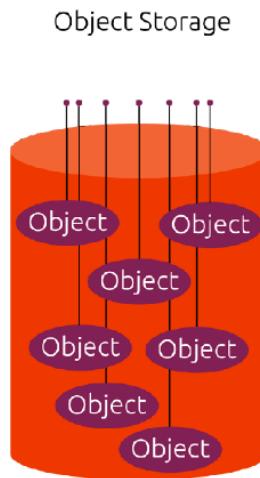


Image credit: <https://blog.ubuntu.com/2015/05/18/what-are-the-different-types-of-storage-block-object-and-file>

- Operating system provides mechanism to read / write files and directories (e.g. POSIX).
- Seeking and random access to bytes within files is fast.
- “Most file systems are based on a block device, which is a level of abstraction for the hardware responsible for storing and retrieving specified blocks of data”

HDFS a été un pont entre le stockage fichier classique et l'object store

OBJECT STORAGE

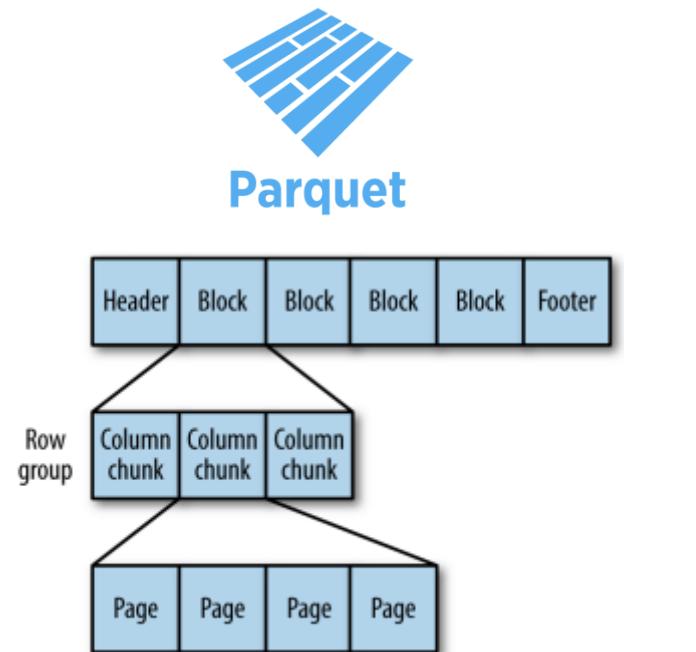
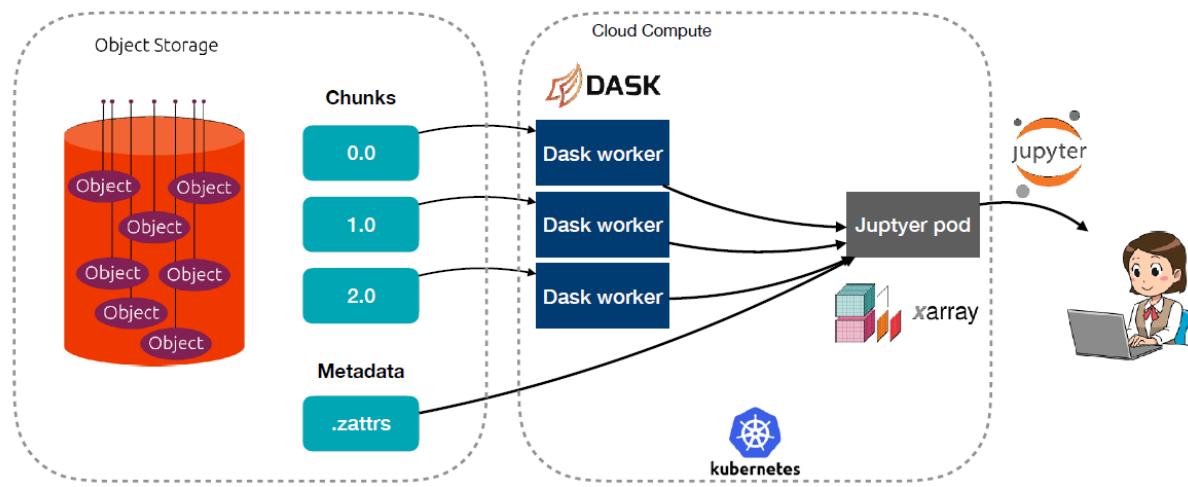


- An object is a collection of bytes associated with a unique identifier
- Bytes are read and written with **http calls**
- Significant latency for each individual operation
- Application level (not OS dependent)
- Implemented by S3, GCS, Azure, Ceph, etc.
- Underlying hardware...who knows?

Image credit: <https://blog.ubuntu.com/2015/05/18/what-are-the-different-types-of-storage-block-object-and-file>

Formats de fichier « Cloud Ready »

ZARR IN PANGEO CLOUD



A Cloud Optimized GeoTIFF (COG) is a regular GeoTIFF file, aimed at being hosted on a HTTP file server, with an internal organization that enables more efficient workflows on the cloud. It does this by leveraging the ability of clients issuing HTTP GET range requests to ask for just the parts of a file they need.



Kubernetes



kubernetes

Kubernetes manages your containers on a cluster of machine while taking care of

- Creation, deletion, and movement of containers
- Scheduling (match containers to machines by resources etc.)
- Scaling of containers
- Serving of containers through unified endpoints
- Monitoring and healing

Kubernetes peut-être comparé à YARN par certains côtés

Frameworks de traitement



SOMMAIRE

1

Big Data & son écosystème

2

Hadoop

- Introduction
- Composants Primaires
- Écosystème
- Distributions

3

Spark

4

Conclusion & Questions

LANGAGE DE REQUÊTAGE

- *Au dessus du MapReduce*

- *Pig*

- Langage de script
 - Développé par Yahoo



- *Hive* :requêtesSQL

- HiveQL :langage SQL –Select only
 - Crée à l'origine par Facebook



- *SQL-on-Hadoop :Impala & Drill*

- Extraction des données directement à partir de HDFS avec SQL
 - Optimisé pour les requêtes à faible latence
 - Requêtes très performantes



BASE NOSQL

HBase

- Base de données NoSQL orientée colonnes
- Distribuée : basée sur Hadoop et HDFS



(Inspirée des publications de Google sur BigTable)

Row Key	Column Key	Timestamp	Value
1	info:name	1273516197868	Gaurav
1	info:age	1273871824184	28
1	info:age	1273871823022	34
1	info:sex	1273746281432	Male
2	info:name	1273863723227	Harsh
3	Info:name	1273822456433	Raman

Annotations pointing to specific features:

- A curly brace on the left side groups the first four rows (Rows 1, 2, 3, 4) and is labeled "Trié selon la clé de la ligne et la clé de la colonne".
- An arrow points from the text "Famille de colonne" to the "info" prefix in the Column Keys.
- An arrow points from the text "Timestamp est entier long" to the timestamp values.
- An arrow points from the text "2 Versions de la ligne" to the second occurrence of Row Key "1" (Rows 3 and 4).
- An arrow points from the text "Nom de colonne" to the "info" prefix in the Column Keys.

ECOSYSTEME HADOOP :CONNEXION A HDFS

Sqoop



- Import des données d'une base de données traditionnelle dans HDFS.
- Développé par Cloudera

Flume

- Collecte d'un ensemble de données (des logs) à partir de plusieurs sources vers HDFS
- Développé par Cloudera



ECOSYSTEME HADOOP

Hue

- Front-end graphique pour le cluster
- Fournit
 - Un navigateur pour HDFS et HBase
 - Des éditeurs pour Hive, Pig, Impala et Sqoop



Oozie

- Outil de gestion de workflow
- Gère et coordonne les jobs Hadoop

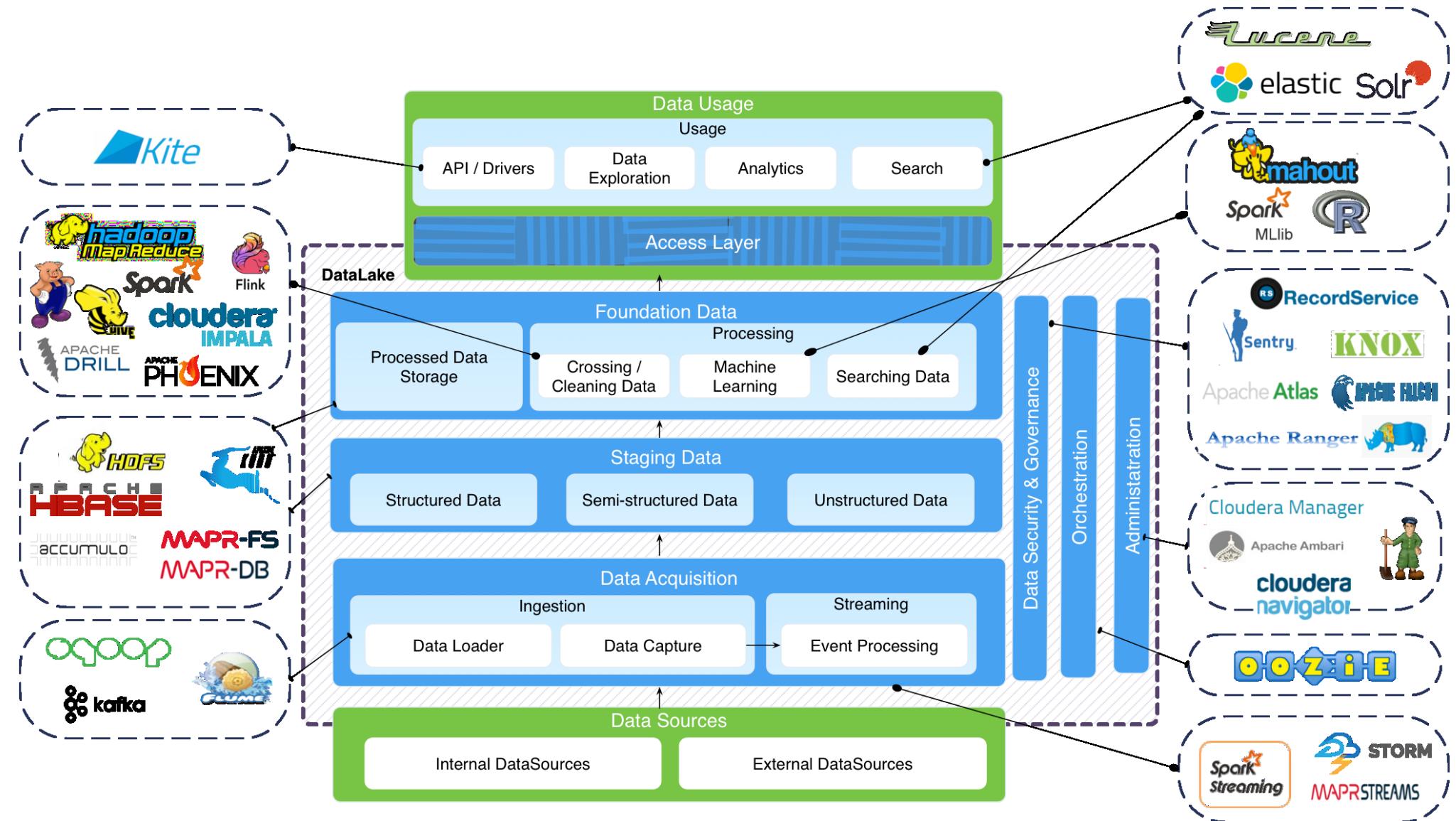


Mahout

- Bibliothèque d'implémentation d'algorithmes d'apprentissage automatique et de datamining



ECOSYSTEME HADOOP



SOMMAIRE

1

Big Data & son écosystème

2

Hadoop

- Introduction
- Composants Primaires
- Écosystème
- Distributions

3

Spark

4

Conclusion & Questions

LES DISTRIBUTIONS D'HADOOP

Open Source

- Apache Hadoop



Pure Players

- Cloudera
- Hortonworks
- MapR



Software Publishers

- Pivotal Greenplum (HDP)
- IBM InfoSphere BigInsights (CDH)
- Oracle Big data appliance (CDH)

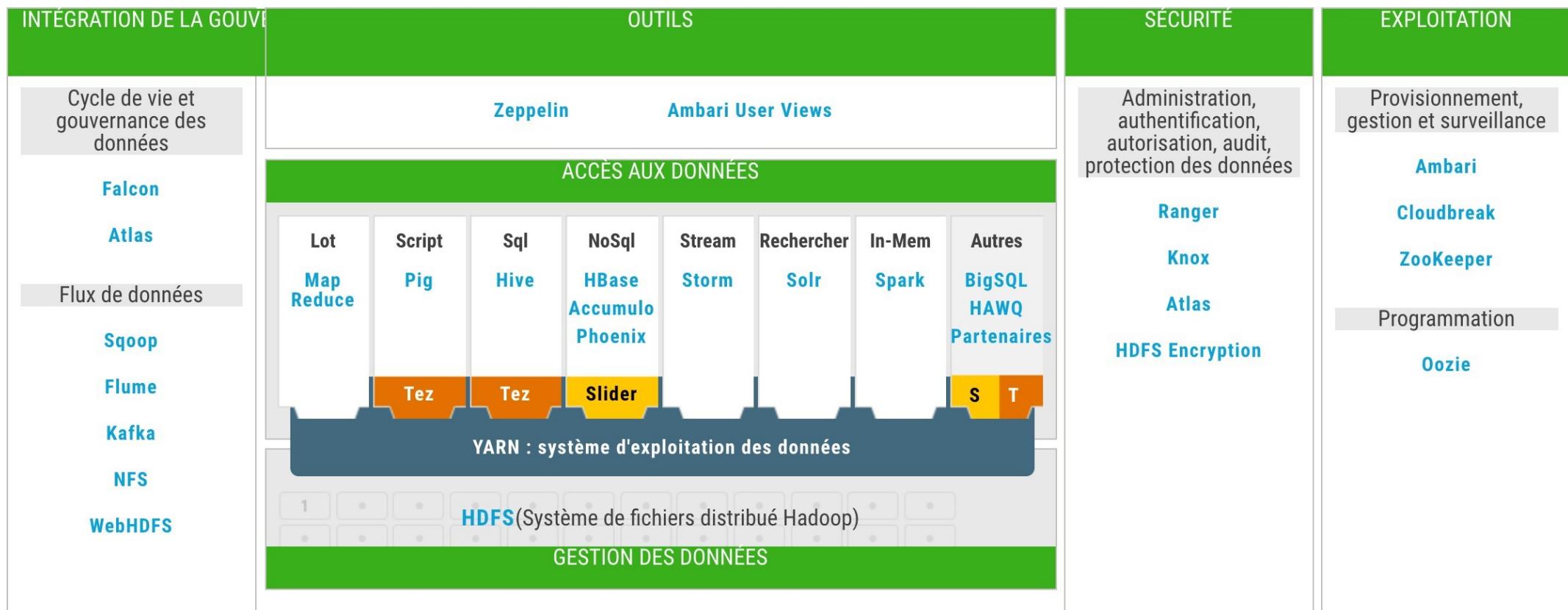


Public Cloud

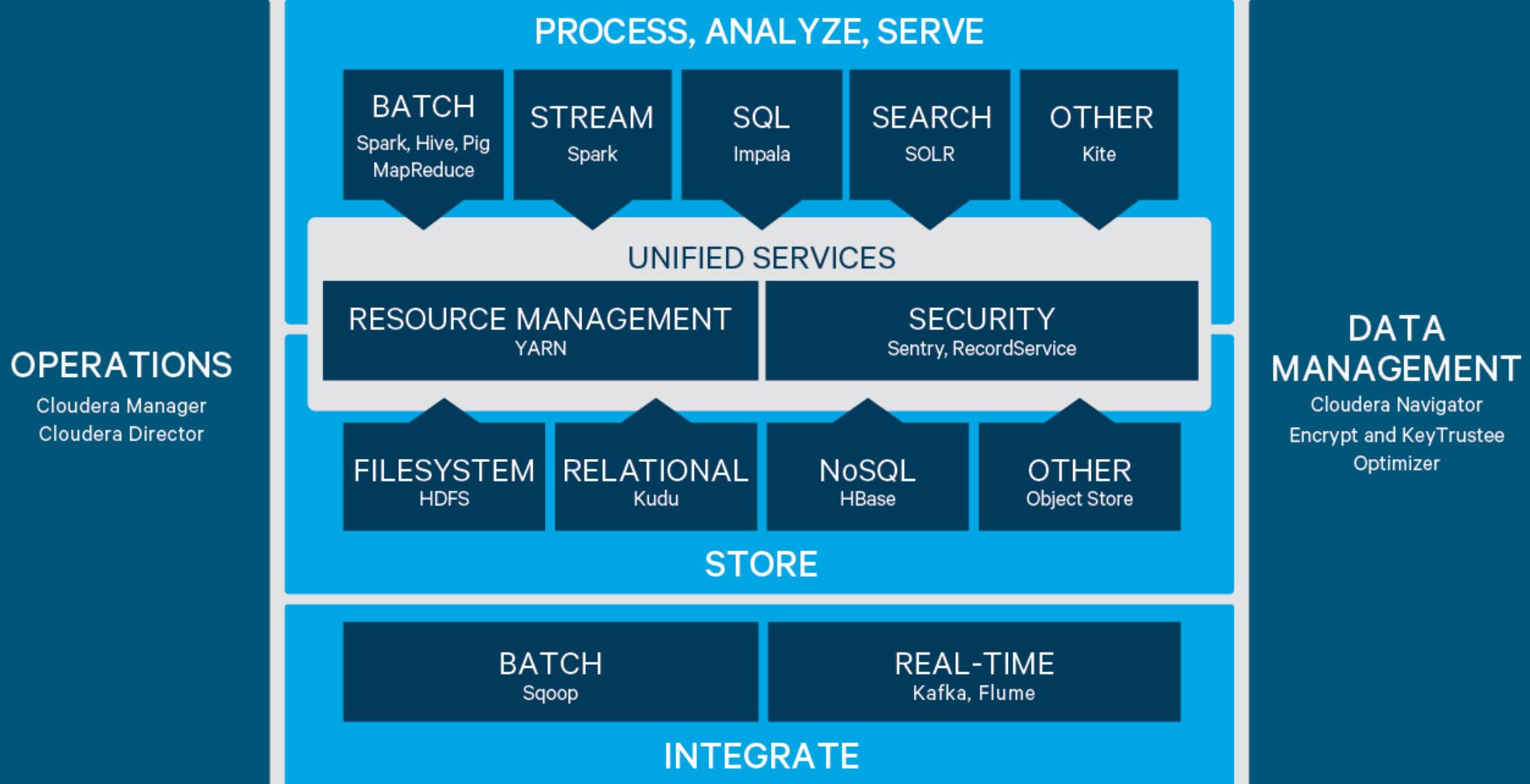
- Amazon Elastic MapReduce (Amazon & MapR)
- Microsoft Azure HDInsight (HW)
- Google Cloud Dataproc



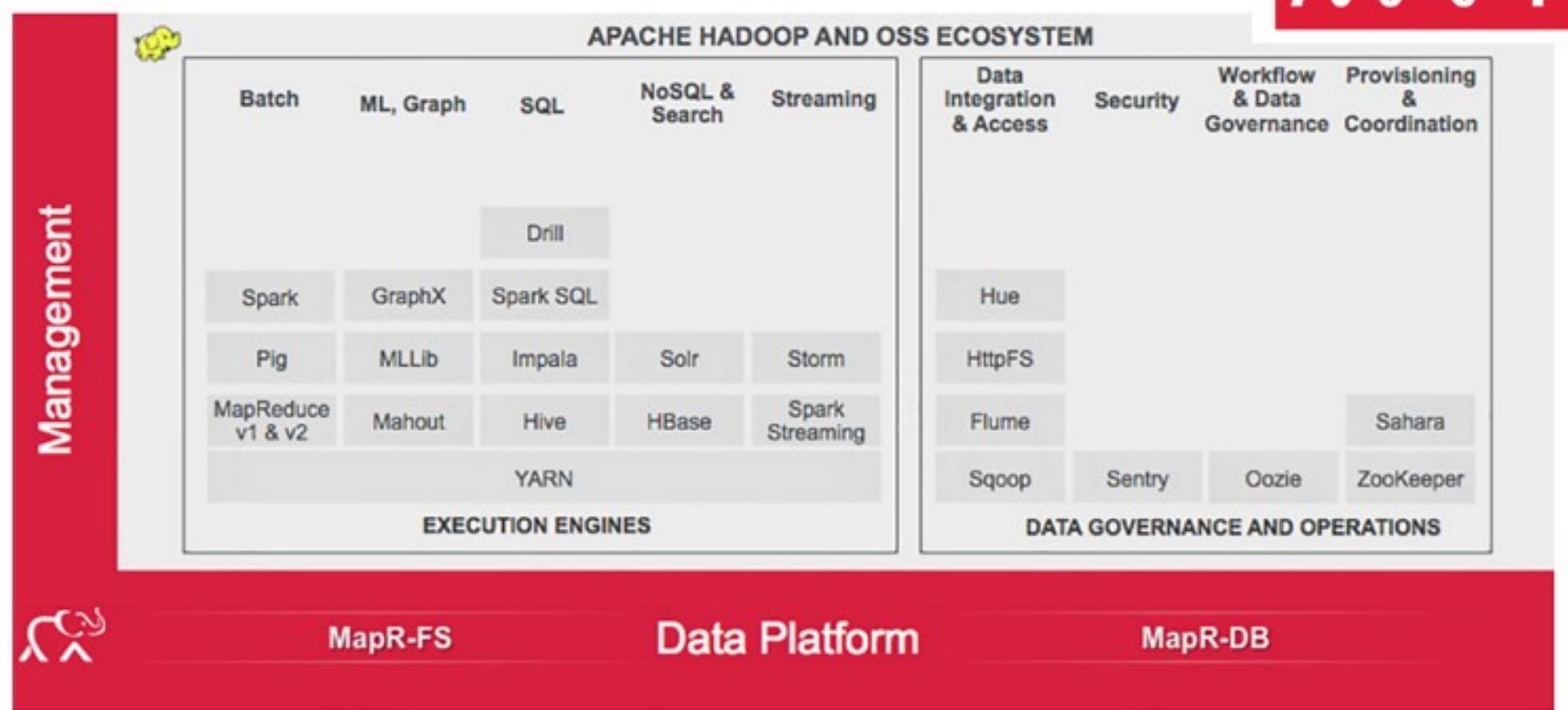
LES DISTRIBUTIONS D'HADOOP: HORTONWORKS



LES DISTRIBUTIONS D'HADOOP: CLOUDERA



LES DISTRIBUTIONS D'HADOOP: MAPR



SOMMAIRE

- 1 Big Data & son écosystème
- 2 Hadoop
- 3 Spark
- 4 Conclusion & Questions

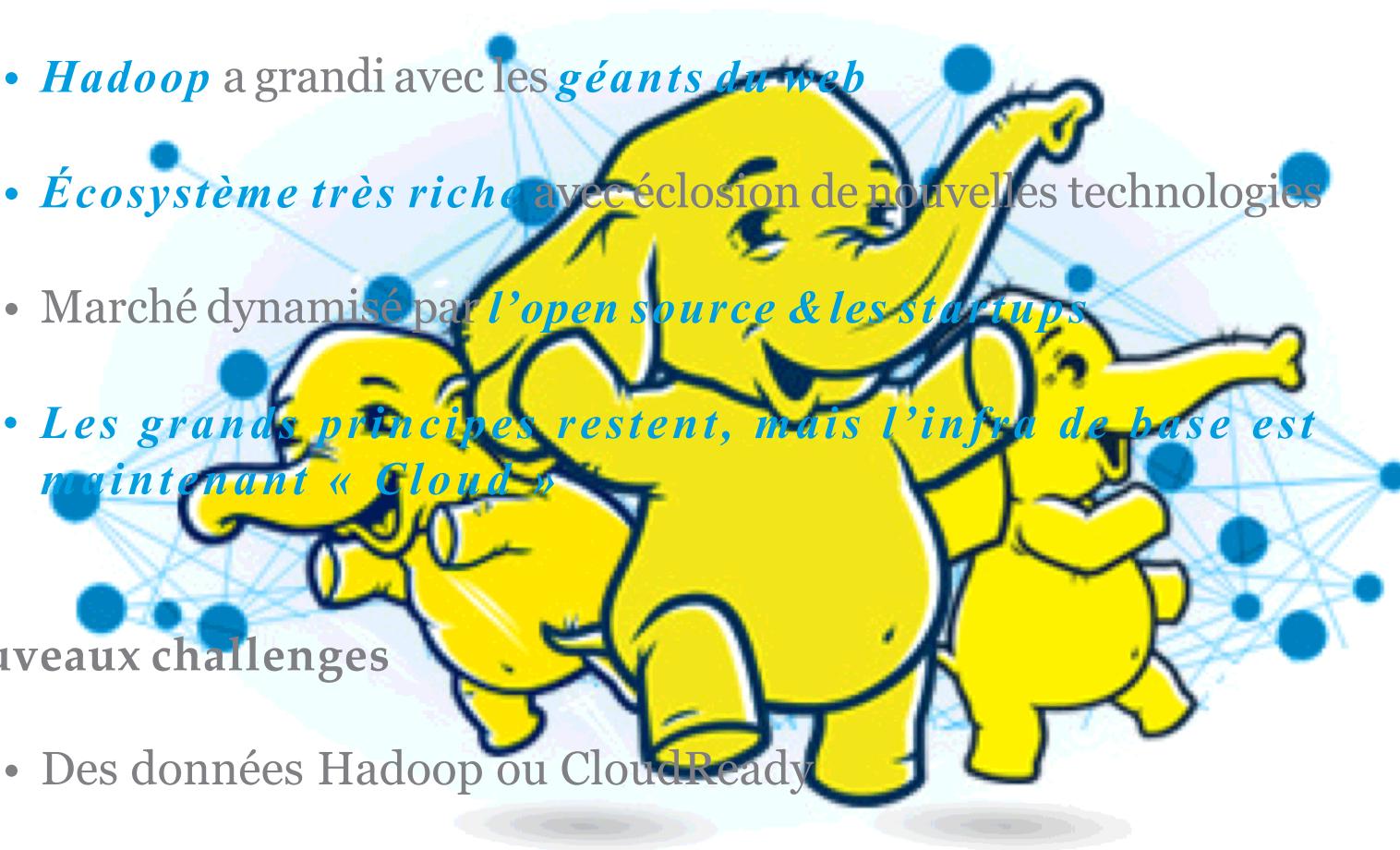
CONCLUSION

Constat

- *Hadoop a grandi avec les géants du web*
- *Écosystème très riche* avec éclosion de nouvelles technologies
- Marché dynamisé par *l'open source & les startups*
- *Les grands principes restent, mais l'infra de base est maintenant « Cloud »*

Nouveaux challenges

- Des données Hadoop ou CloudReady
- La *gouvernance* des données et la *sécurité*
- Machine Learning et Deep Learning



Merci,

Des Questions ?