# Big data

## C’est quoi le big data ?

### Big data : constat

• Explosion des capacités de stockage et de calculs des ordinateurs

• Augmentation exponentielle du nombre de données générées et stockées

• Augmentation exponentielle du nombre de données

• Développement des technologies de virtualisation et du cloud computing

### Big data : loi de Moore

• La loi de Moore stipule que les ordinateurs deviennent au fil du temps plus petits, plus rapides, moins chers, à mesure que les transistors sur circuits intégrés deviennent plus efficaces.

• Aujourd’hui, on est arrivé à une limite physique. On n’arrive plus à réduire les transistors.

### Big data : volume de données

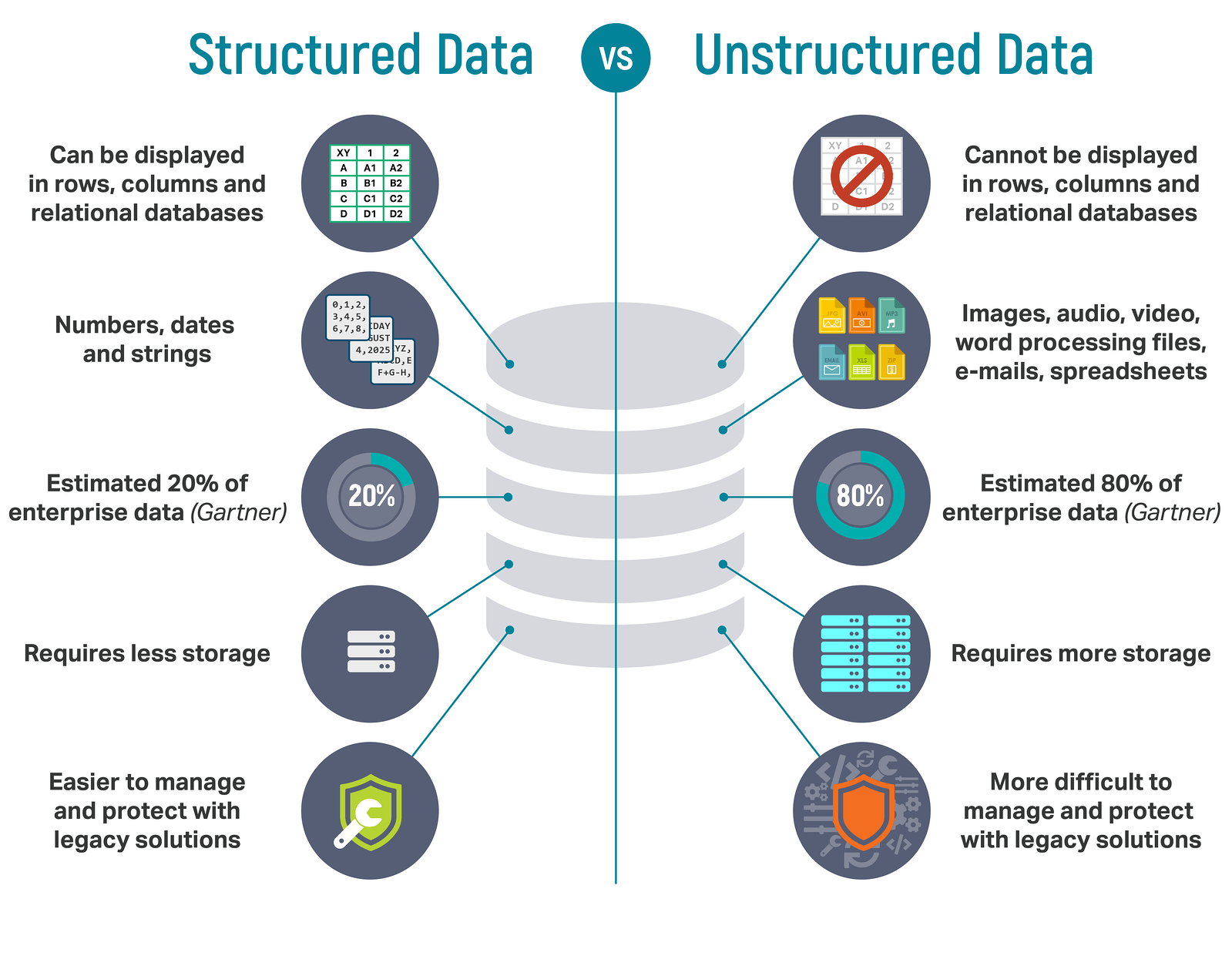
• Génération sans cesse croissante de données

• En 2010, le monde entier avait généré 2 zettaoctets, en 2015 on a multiplié par 8 ce chiffre et en 2020, on est à 181 zettaoctets.

• Rappel : 1 Zo = 1000 000 000 000 Go = 1021 octets

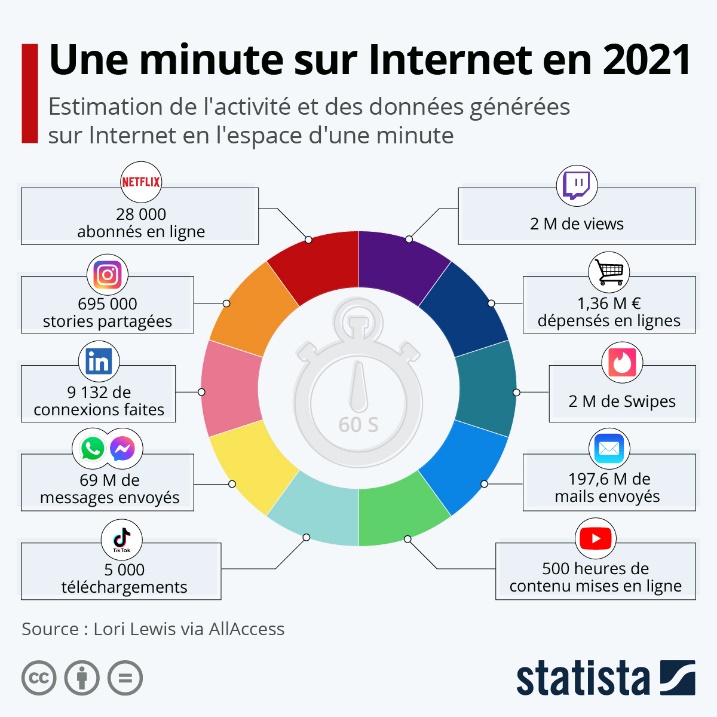
• Jusqu’à 2010, 80% de la data générée dans le monde était à usage humain. En 2020, on est à 18% et ça continue de descendre. Les données qui ne sont pas à usage humain sont pour les machines qui communiquent avec les autres machines.

### Données structurées vs non structurées



• Utilisation de données semi ou non-structurées de plus en plus importante.

### Big data : données connectées



### Limitation des systèmes classiques

• Données trop volumineuses pour être stockées sur un seul serveur

• Données trop variées pour être stockées facilement

• Données changeantes trop rapidement pour être stockées/traitées facilement

• Données trop complexes pour être traitées facilement

### Le big data

• **Big data :** ensemble de technologies et de méthodes permettant de stocker, traiter et analyser des données massives, variées et changeantes.

• Objectif : permettre un traitement efficace de données massives, variées et changeantes **à un coup financier, humain et temporel raisonnable**.

### Big data : historique

• Concept ancien issu des années 70 avec les premiers datacenters

• Le terme serait apparu en 1997.

### Big data : tout est information

• Principe que l’homme et la totalité du monde qui l’entoure peuvent être représentés comme « des ensembles informationnels, dont la seule différence avec la machine est leur niveau de complexité. »

• « La vie deviendrait alors une suite de 0 et de 1, programmable et prédictible.

### Big data : MapReduce et NoSQL

• Concrétisation dans les années 2005 par Google qui déploie un algorithme sur des opérations massives : le MapReduce qui deviendra le projet Hadoop.

• Dans les années 2009, le déploiement de stockage open-source, distribués et non-relationnels : le NoSQL

### Big data : concepts

• Le big data repose sur le concept de **parallélisme** et de **distribution** des algorithmes et de traitement et du stockage des données.

• Diviser pour mieux régner, c’est-à-dire répartir le stockage ou le traitement des données massives sur plusieurs machines.

### Big data : domaines d’application

• L’utilisation du Big Data est très large et touche de nombreux domaines :

— **Commerce :** analyse des données clients, prévision, recommandation, etc.

— **Finance :** analyse des données financières, prévision, etc.

— **Santé :** analyse des données médicales, prévision, etc.

— **Energie :** analyse des données énergétiques, prévision, etc.

— **Transport :** analyse des données de transport, prévision, etc.

— **Agriculture :** analyse des données agricoles, prévision, etc.

### Exemple : en santé

• Recherche sur l’effet d’un médicament dans une maladie

• Collecter des données à partir de milliers/millions de patients

• Sources multiples et variées (dossiers médicaux électroniques, des bases de données d’assurance maladie, des registres et des enquêtes de santé publique).

• Analyses des données

• Identifier tendances, corrélations avec les modèles, efficacité du médicament, facteurs de risque, traitements associés, et les impacts sur la qualité de vie

• Aide à la décision, adaptabilité des traitements, prévision des effets secondaires ou de l’évolution de la maladie

### Big data : domaine technique

• Croisement de nombreuses spécialités techniques :

— Informatique transactionnelle (principe ACID, etc.)

— Informatique décisionnelle (BI, prise de décision, etc.)

— Informatique en temps réel (temps de réponse critique)

— Stockage et tri de données (besoin volume, rapidité, etc.)

— Traitement et analyse de données (catégorisations, synthèse, prédictions, représentations, etc.)

## Caractéristiques du big data

### Les 3 V

• Le Big Data repose sur le principe des 3 V :

— Volume

— Vitesse

— Variété

### Volume

• Volume : masse croissante de données

• Nécessite des besoins spécifiques pour :

— stocker les données

— transporter et échanger les données

— traiter et analyser les données

• Volume : producteurs

— Entreprises commerciales (Google, Amazon, Facebook, Twitter, etc.

— Analyses de données de santé (les données médicales)

— Analyses météorologiques/spatiales

### Vitesse

• Vitesse : rapidité avec laquelle les données sont générées, traitées ou modifiées

— Répond aux besoins des processus chronosensibles (bourses, stream, etc.)

— Risque pour l’Homme de perdre le contrôle sur les données (traitement à la nanoseconde)

### Variété

• Variété : diversité des types de données

— Données structurées (tableaux, fichiers, etc.) : 20%

— Données semi ou non structurées (XML, JSON, textes, images, etc.) : 80%

• Variété : données structurées : données qui a une structure prédéfinie (tableau, fichier, etc.)

— Ce sont toutes les données observables avec un schéma de typage bien définis et fixe

— Généralement, elles sont observables en tableau ou base de données relationnelles traditionnelles (adresses, ventes d’articles, etc.)

• Variété : données non-structurées : données qui sont qualitatives et non quantitatives

— toutes les données qui ne sont pas observables avec un schéma de typage bien définis et fixe

— sont une conglomération de nombreuses données de différents types qui sont stockées dans leurs formats en mode natif (vidéo, audio, etc.)

• Variété : données semi-structurées : données intermédiaires entre les données structurées et non-structurées

— données qui ont une structure mais qui ne sont pas fixes ou qui n’ont pas de structure prédéfinie mais possède des métadonnées typées (xml, JSON, etc.)

### Big data : extension des 3V

• Aujourd’hui, les principes des 3V se sont largement étendus. Par extension, on y trouve d’autres grands V comme : volatilité, valeur, vulnérabilité, véracité, validité, visibilité

### Volatilité

• Notion de durée de vie d’une donnée

— L’estimation de la durée de vie d’une donnée

— De sa prise en charge et son traitement

— De son obsolescence sont à prendre en compte

### Valeur

• Notion de profit à tirer d’une donnée

— Extraire une valeur d’une masse d’information est une problématique majeure

— Notions de regroupement, filtres, classification, hiérarchisation, etc.

### Vulnérabilité

• Notion de sécurité des données

— Un gros volume de données est une cible de choix pour les pirates informatiques

— Induit la nécessité d’une structure sécurisée adaptée

### Véracité

• Notion de fiabilité et confiance des données

— Nécessite des outils de vérification et de validation des données (recoupement, enrichissement, etc.)

### Validité

• **Validité :** notion de conformité et de précision des données

— Notion parallèle à la véracité

— 60% du temps d’un scientifique est consacré au nettoyage de ses données avant de pouvoir effectuer une analyse

### Visibilité

• Visibilité : notion de visibilité des données

— Plus il y a de données, plus il est difficile d’en avoir une vision précise et une représentation claire.

— Nécessite des outils spécialisés avec des multiples représentations complémentaires.

La couleur et la forme dont je vais représenter les données vont avoir un impact sur la façon dont est perçue l’analyse de la donnée.

## Big data : technologies

### Les technologies du Big data

• Ensemble de techniques et stratégies permettant de répondre aux contraintes des grands V du Big Data.

### Paradigmes du Big data

• Pour répondre aux besoins du Big data, des systèmes permettant de dépasser les limites des systèmes traditionnels ont été développés.

— Architectures scalables

— Solutions de stockages pour les données non structurées

— Architectures distribuables et massivement parallèles

— Algorithmes distribués et parallèles

### Scalabilité

• Adapter la taille et/ou la puissance d’un système informatique pour répondre aux changements de la charge de travail.

• Il existe deux types de scalabilité :

— verticale : augmentation des ressources internes

— horizontale : augmentation des ressources externes

• Scalabilité verticale : augmentation de la puissance (processeur, RAM, stockage) d’un système

— C’est la solution la plus simple et la plus rapide à mettre en œuvre.

— Fréquemment utilisée dans les systèmes traditionnels.

• Limites de la scalabilité verticale : l’augmentation de la puissance d’un système informatique est limitée par :

— Cout du matériel (augmentation exponentielle)

— Faible adaptabilité aux changements de la charge de travail

— Problèmes en cas de pannes (single point of failure)

• Scalabilité horizontale : augmentation du nombre de machines de faible puissance pour augmenter la puissance globale.

— C’est la solution économique la plus adaptée.

— Pseudo linéarité des performances.

— Découpages et réplications des données.

• Limites de la scalabilité horizontale : l’augmentation du nombre de machine de faible puissance est limitée par :

— augmentation exponentielle des échanges

— architectures réseaux complexes (cluster)

— synchronisation des données

### Modèles de stockage

• L’avènement du Big Data a nécessité l’adaptation du des systèmes de stockage pour s’adapter à la quantité et aux types de données

— Théorème de CAP et Big Data

— Modèles NoSQL

### Théorème de CAP

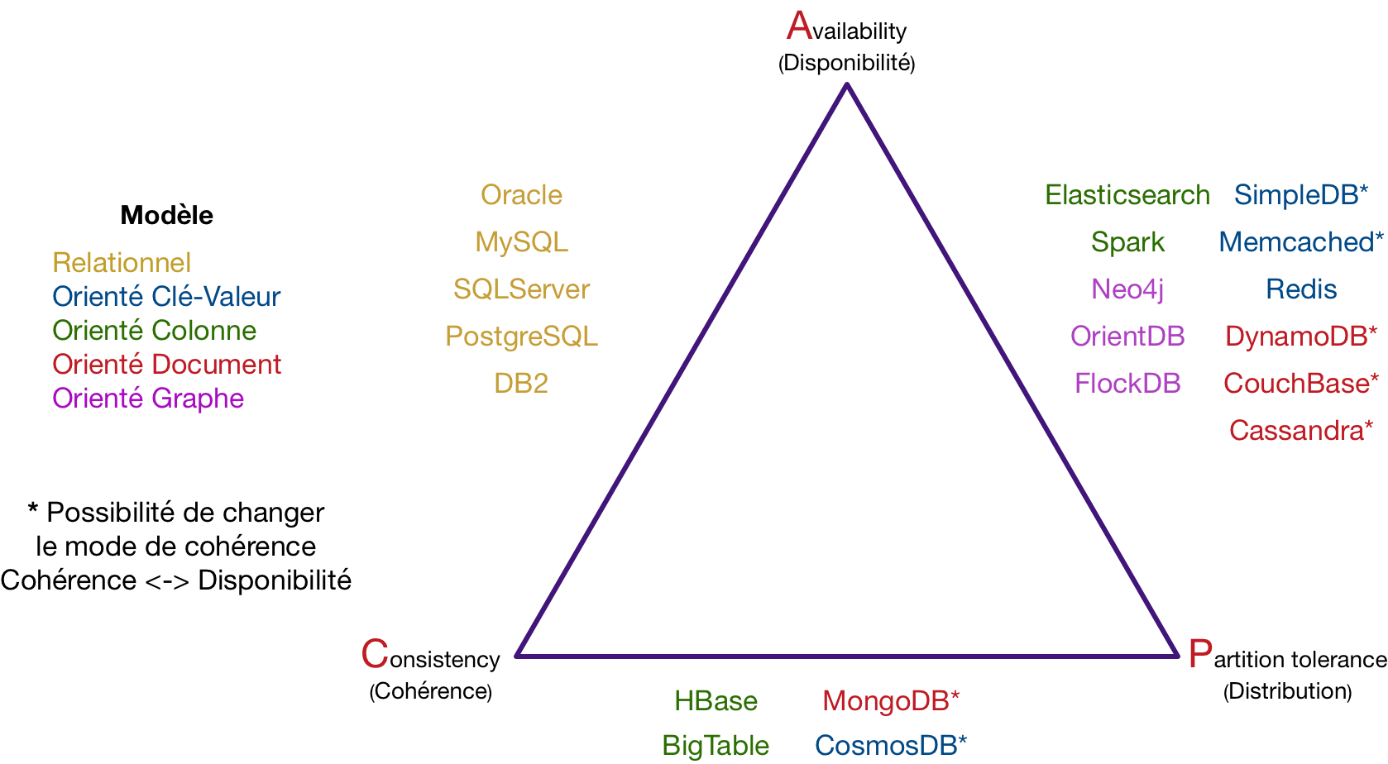
• Tout système ne peut garantir que 2 des 3 propriétés suivantes :

— Consistency (cohérence) : accès aux mêmes données à tout moment

— Availability (disponibilité) : accès à la lecture et l’écriture à tout moment

— Partition tolerance (tolérance aux pannes)

• Triangle de CAP :



Relationnel + les 4 types NoSQL

### Stockage traditionnel

Les systèmes de stockage traditionnels utilisent des bases de données relationnelles

— Données structurées et formatées

— Requêtes et manipulations simples

— Transaction qui respectent les principes ACID

— Large panel de solutions (longévité des technologies)

### Limites des bases de données relationnelles

• Manque d’adaptabilité des schémas de données

• Limitées aux données structurées (< 20%)

• Dépenses massives en temps et ressources en cas de modifications de la structure des données

### Avènement du NoSQL

• Développer pour pallier aux limitations des bases de données relationnelles

• Prise en charge des données non structurées

• Adaptabilité des schémas de données

• Abandon des principes ACID au profit des principes BASE

### Les principes BASE

• Basically available : notion de disponibilité des données à tout moment (réplication des données)

— Le système doit pouvoir répondre aux requêtes de tout utilisateur même en cas de pannes

— Les requêtes peuvent être obsolètes (cf. Isolation ACID).

*En SQL on privilégie la disponibilité des informations plutôt que leur consistance.*

• Soft-state : notion que les données sont dans un flux d’utilisation constante.

— Les données peuvent être utilisées par plusieurs utilisateurs en même temps.

— La cohérence n’est pas garantie (cf. Cohérence ACID).

• Eventual consistency : notion qu’au bout d’un certain temps, le système sera cohérent

— La synchronisation des données est faite en arrière-plan.

— Les données peuvent être obsolètes (cf. Cohérence ACID).

### Les modèles NoSQL

• À partir ces principes, quatre modèles de stockage NoSQL ont été développés :

— clé/valeur

— colonne

— document

— graphe

• Modèle clé/valeur : stockage sous forme d’un dictionnaire clé/valeur

— Clé : chaine de caractères unique

— Valeur : typage au besoin

Ex : redis (StackOverFlow), riak (GitHub), Memcached (Wikipédia), Voldemort (LinkedIn)

• Modèle colonne : stockage sous forme de table dénormalisée

— Système proche des bases de données relationnelles

— Tableau de clé/valeur groupable en famille (+/- table)

Ex : Cassandra (Nasa), HBase (Facebook, Xiaomi), BigTable (GCP)

• Modèle document : stockage sous forme de document

— Object JSON avec un id et des propriétés clé/valeur

— « Valeur » peuvent être d’autres documents

Ex : MongoDB (SEGA, ThermoFisher Scientific), CouchDB (CERN)

• Modèle graphe : stockage sous forme de graphique relationnel

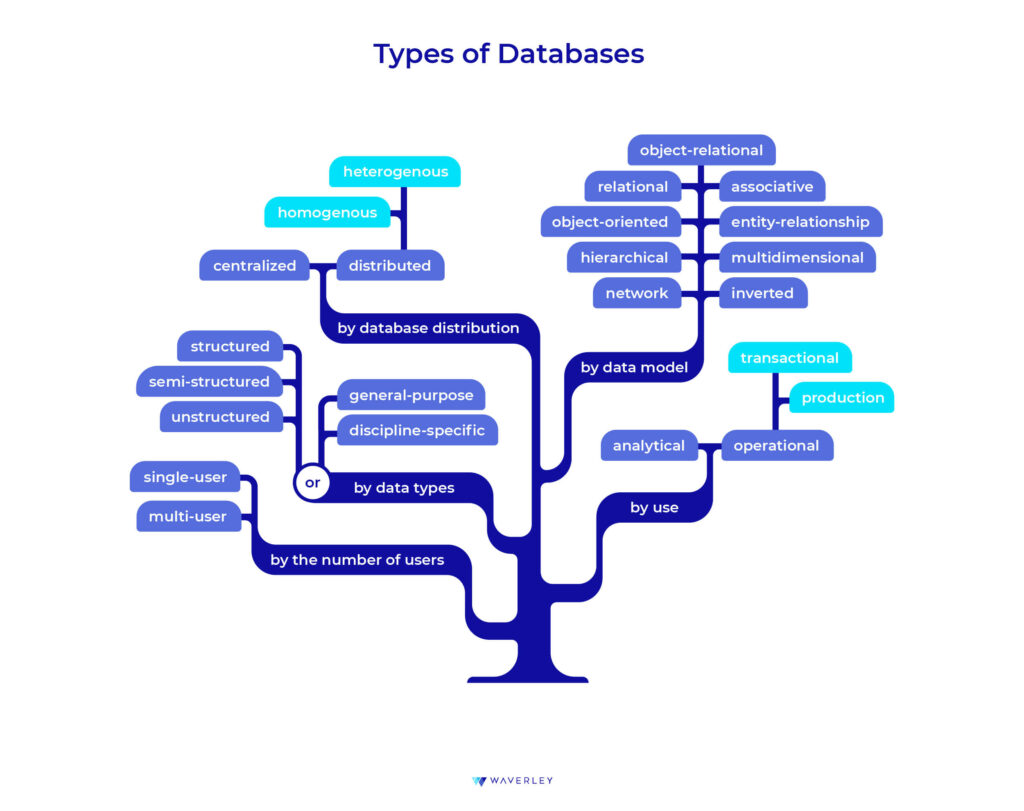
— Nœuds de données de type document clé/valeur

— ARC : relation orientée entre les nœuds porteuse de propriétés

Ex : Neo4j (Orange, Airbus), OrientDB, Titan

### Choix du modèle de stockage

• Critère variable selon besoin



## Big Data : distribution

### Technologies de distribution

• La majorité des technologies du Big Data pour répondre aux besoins des grands V du Big Data sont des technologies distribuées :

— Distribution des stockages

— Distribution des traitements

### Distribution des stockages

• Répartition des données sur plusieurs machines

— La répartition des données permet un stockage plus important qui n’est plus limité à la taille d’une machine.

— Stockage très facilement scalable horizontalement

— Permet réplication et tolérance aux pannes

### Stockage distribué

• Le stockage distribué peut faire intervenir plusieurs types de niveau selon les types et les besoins de stockage

— Réplication du stockage

— Répartition des données

— Sharding

### Réplication du stockage

• Repose sur un principe de dupliquer les données sur plusieurs machines

— Permet d’accroitre l’accessibilité des donnes

— Permet d’être facilement scalable horizontalement

— Assure la tolérance aux pannes

### Approches réplicatives

• Il existe 3 types d’approches de réplication :

— Système maitre esclave (postgreSQL, MongoDB)

— Système multimaster (CouchDB)

— Système de masterless (Cassandra)

## A

## A

A

•

—

Datamart, datalake, datawarehouse, etc etc, datacenter