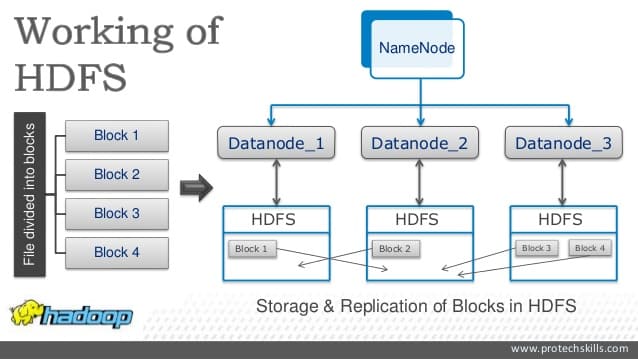
**Introduction au BIG DATA**

* Concept, paradigme, technologie, architecture et procédure
* Pour organiser, capter, stocker, traiter et analyser
* Des données massives hétérogènes et changeantes
* Le big data est caractérisé par les 5V (à la base c’étaient des contrainte qu’on a ensuite intégré dans le concept de big data)
  + **Volume :** 
    - Le volume données à traiter augmente continuellement au fil du temps.
    - De nos jours les Hommes et les machines (connectées) génèrent une très grande quantité de données qu’on ne peut pas traiter avec les outils ‘traditionnelles’.
    - On dit que les données ont une croissance exponentielle.
  + **Variabilité :**
    - On traite des données diverses et variées
    - De nature différentes (fichiers textes, audio, photo, vidéo, …)
    - De différentes sources (internet des homme, internet des machines, …)
  + **Vélocité :**
    - C’est vitesse avec laquelle on traite des données
    - Traitement en streaming (en temps réel) et aussi en batch (traitement périodique automatisé)
  + **Véracité :**
    - C’est la fiabilité des données et donc c’est de leur qualité qu’il s’agit ici
    - A quel point on peut se fier sur des données (degré de certitude)
    - C’est le fait de douter des données de sources externes
    - Exemple : sur les réseaux sociaux on peut récupérer des publications ou commentaires 🡪 comment différencier l’ironie des vraies infos ??
    - Il faut faire attention pour ne pas biaiser les données et analyses)
  + **Valeur**
    - C’est l’objectif des projets big data (notamment du data analyste)
    - Donner des valeurs ajoutées aux données à partir des analyses pour éclairer les décideurs dans l’entreprise
    - Générer des actionnable
* Historiquement, c’est Google qui a été confronté aux problèmes de traitement de données massives pour indexer les pages web 🡪 PageRank (algorithme d’analyse des liens Google pour affiner leur classement)
* Ils ont fait du scale-up de serveur afin d’avoir une capacité de stockage et une puissance de calcul plus conséquentes
* Lorsqu’on parle de ressource, il s’agit de l’ensemble de CPU (processeur), RAM et disques
* En big data, l’idée est de diviser les fichiers (données) en petits blocs de métadonnées pour mieux les traiter.
* Donc la notion de scalabilité est importante
  + Scalabilité verticale (scale-up) 🡪 voir historique Google ci-dessus
  + Scalabilité horizontale (scale-out) 🡪 C’est le cas général en big data, on effectue des traitements distribués sur des données stockées de manière distribuée sur plusieurs petites machines (appelés nœuds) connectées entre elles pour former un cluster.



**Stockage distribué**

* C’est un système de gestion de fichiers distribués au sein d’un cluster Hadoop
* Les fichiers sont divisés en plusieurs machines appelés nœuds (*Node* en anglais) dotés des mêmes services Hadoop.
* Hadoop c’est une plateforme composée de plusieurs services installés sur tous les nœuds.
* Tous les services Hadoop sont des applications distribuées qui travaillent ensemble pour le même but.
* Tolérant aux pannes
  + En cas de panne d’un ou plusieurs nœuds, on peut reconstituer les fichiers et continuer leur traitement
  + On recommence que les traitements sur les blocs ayant subis de panne
  + Les pannes sont généralement matérielles : problème dans les ressources
    - En panne physique
    - Indisponible à cause de surcharge système par exemple
* Réplication des blocs sur d’autres machines avec un facteur de réplication
* Plus le facteur de réplication est élevé plus on augmente la tolérance aux pannes
* Le facteur de réplication est 3 par défaut (2 copies + l’original)

**Hadoop Map Reduce (traitement distribué)**

* Framework de développement distribué
* Conçu pour un système de stockage distribué
* Gère la distibuabilité, trouver les blocs, synchronise, …
* Signification de Map Reduce
  + Map : programmer les traitements à effectuer sur chaque ligne
  + Reduce : les conditions et opérations pour agréger les lignes pour avoir les résultats de traitement
* ****Résiste également aux pannes

**Hadoop YARN**

* Yet Another Resouce Negociator
* Gère les ressources
* Fait en sorte que les jobs qu’on envoie aux nœuds s’exécutent

**ZooKeeper**

* Application qui gère (coordonne) tous les services Hadoop

**Résumé de la plateforme BIG DATA**

* Scalabilité horizontale 🡪 système distribué
* Tolérant aux pannes 🡪 réplications des blocs qui empêche toute perte de donnée
* Facile d’usage 🡪 dans Map Reduce il y a que deux actions à faire
* Capacité à traiter des données variées 🡪 Attention tout est fichier en informatique !
* Schéma à la lecture 🡪 adopte la structure donnée qu’on veut lors qu’on les récupère
* Pas de mise à jour 🡪 les màj génèrent souvent des incohérences de données et des données zombies

**Quelques use cases**

* Dans le marketing on utilise le big data pour des analyses prédictives, des sentiments et des comportements pour avoir un avantage concurrentiel dans le marché
* Dans le sport pour maximiser la performance des joueurs pour avoir une longueur d’avance
* Dans la sécurité publique utilisé à Londres lors des JO pour sécuriser l’évènement

**DataLake**

* DataWarehouse du big data 🡪 base de données dénormalisée (pas relationnelle)
* Une philosophie dans l’entreprise
* Non redondance des flux (n’importe quel métier peut avoir accès au data lake)
* Favorise l’esprit d’innovation car dès qu’on a une idée, on peut l’exploiter grâce au traitement des données dans le datalake
* Accélère le time to market

**Outils d’intégration de données**

* En temps réel (streaming) ou en batch
  + Kafka, Spark Streaming
  + Nifi, StreamingSets, talend
* Outils de reporting
  + mongoDB, Elastic, Cassandra, Solr