NoSQL

- Soulage certaines contraintes des bases relationnelles pour faciliter la **scalabilité horizontale**.
- Quatre grandes catégories : **Clé–Valeur**, **Colonnes**, **Documents**, **Graphes**.
- **Avantages :** Grande flexibilité, rapidité, très adapté aux applications distribuées.
- **Inconvénients :** Perte de certaines garanties ACID, requêtes parfois moins simples à formuler.

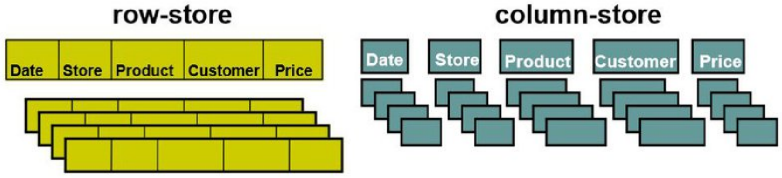
Principales familles de NoSQL

Bases Clé–Valeur

- Stockage minimaliste basé sur des paires (*clé, valeur*).
- Opérations typiques : `get(key)` et `put(key, value)`.
- Parfait pour des cas d'utilisation simples (cache, sessions).
- Exemples : Redis, Amazon DynamoDB.

Bases orientées colonnes

- Stockage des données colonne par colonne (et non ligne par ligne).
- Optimisées pour les lectures/aggrégations d'une même colonne.
- Exemples : Cassandra, Apache HBase.



Bases orientées graphes

- Représentation des données sous forme de nœuds et d'arêtes.
- Extrêmement utiles pour les relations complexes (réseaux sociaux, recommandations).
- Exemples : Neo4j, ArangoDB.

Bases orientées documents

- Stockage de documents **semi-structurés** (JSON, BSON, XML).
- Flexible, chaque document peut avoir des champs différents.
- Exemples : MongoDB, Couchbase.

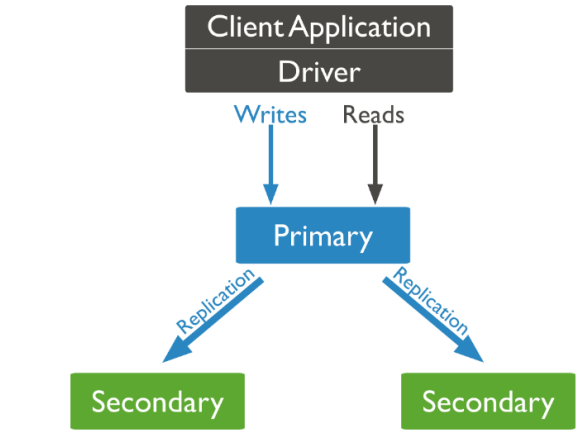
MongoDB : un exemple de base NoSQL orientée documents

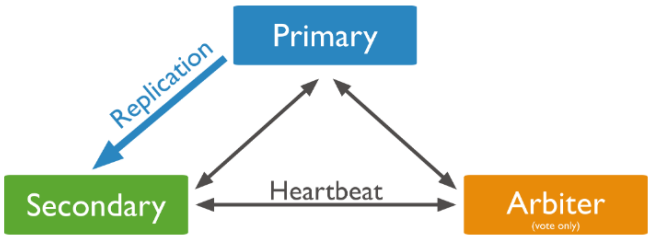
Définition

- Base de données **open-source**, orientée documents (stockés en **BSON**).
- Fournit des API multiples (C/C++, Java, Python, etc.) et **hautement scalable**.

Modèle de données

- Les documents sont regroupés en **collections**.
- Chaque document possède un **_id** unique (généré automatiquement ou défini par l'utilisateur).
- Avantage : Schéma **flexible** (des documents dans la même collection peuvent avoir des champs différents).

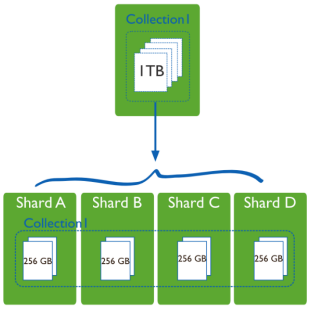


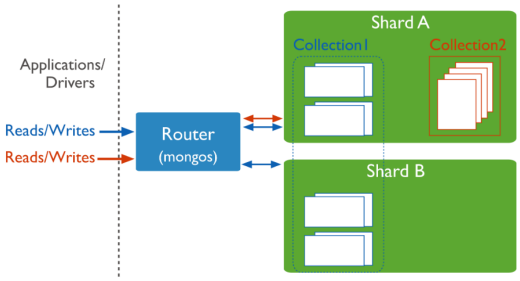


Réplication (Replica Sets)

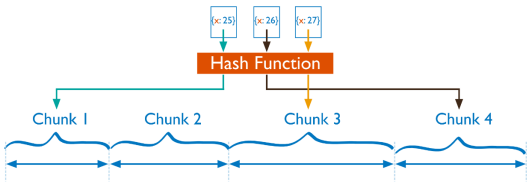
- Un **Replica Set** contient plusieurs instances `mongod`.
- **Nœud primaire** : reçoit les écritures; **Nœuds secondaires** : répliquent le journal des opérations (*oplog*).
- En cas de panne du nœud primaire, les secondaires élisent un nouveau primaire.
- Garantit la **haute disponibilité**.

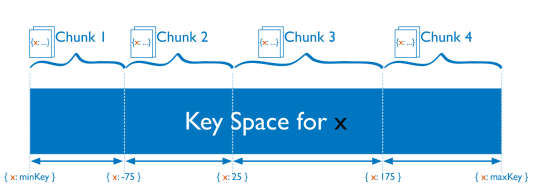
Sharding





- Technique de **partitionnement horizontal** des données en plusieurs morceaux (*chunks*).
- **mongos** : routeur qui fait le lien entre l'application cliente et les différents shards.
- Chaque shard contient une partie du jeu de données (**distribution**).
- **Hashed sharding** : basé sur une clé de hachage pour répartir uniformément les documents.
- **Ranged sharding** : partitionnement par plages de valeurs contiguës pour des requêtes ciblées mais potentiellement déséquilibrées.





Exemple pratique

Gérer des données universitaires massives : informations étudiantes, cours, notes, remarques, etc. un RGBD relationnel multiplierai les tables et donnerai une complexité et des volumes élevés rendant le système peu scalable.

Solution NoSQL

- Chaque étudiant = 1 document avec ses informations, ses cours, ses notes, ses remarques.
- **Avantage** : Regroupement des informations dans un seul document (plus besoin de jointures complexes).
- **Adapté** à de fortes volumétries et à une évolution fréquente du schéma.

Hugo Tondenier