Comparaison MongoDB, MySQL, Oracle(un peu)

|  |  |  |
| --- | --- | --- |
| SGBD | **MongoDB** | **MySQL** |
| Type | [orientée documents](http://fr.wikipedia.org/wiki/Base_de_donn%C3%A9es_orient%C3%A9e_documents) | Objet |
| Schéma | pas de schéma prédéfini des données | schéma relationnel |
| Stockage | données stockées au format JSON | Lignes, colonnes |
| Manipulation | Collection | Table |

Oracle:

Permet de manipuler une grosse quantité de données mais présente les même désavantage que mySql(schéma relationel, non dynamique etc...) et de surplus est lourd.

**Pourquoi utiliser MongoDB?**

* Besoin d’une seule table, pas de relation.
* système de requêtage évolué : mots clés (retrouver si une question a déja été posée?)
* Performance:

Pas de jointures et fournit d'excellents outils pour l'analyse de la performance

→ apporte un gain de rapidité, mieux que une base de données relationnelle . (Besoin d’indéxer les données)

* Ecriture   
  MongoDB par défaut préfère un taux élevé d'insertion en contrepartie de la sécurité.

→ Besoin de charger des tonnes de lignes de données (avec les millions d’utilisateurs)

* La base de donnée peut atteindre une taille importante (et Shard vos données)  
  → Dégradation de mySQL si ça dépasse les 5\_10GB par table

→ MongoDB intègre une solution facile qui permet de Shard les données donc de les répartir sur différents disques dur   
Bases de données mise à l'échelle est difficile ( une seule représentation de table MySQL va se dégrader en traversant la 5-10GB par table) . Si vous avez besoin pour partitionner et Shard votre base de données , MongoDB a intégré dans une solution facile pour cela.

* L’ensemble des données peut atteindre une taille importante (à partir de 1 Go ) et le schéma n’est pas stable( question sans réponses?)  
  Obligation d’ajouter des colonnes en SGBDR = dégradation des performances lorsque la taille est supérieur à 1GO  
  Avec MongoDB , l'ajout d'un nouveau champ n’affecte pas les anciennes lignes (ou documents ) et sera instantanée . Autre avantage, pas besoin d'un DBA de modifier votre schéma lorsqu’il ya des changements dans l’application .
* Bonus : MongoDB est rapidement adopté par les applications web complexe conjointement avec nodejs . C’est plus performant car mongodb est basé BSON donc pas besoin d' opérateur JOIN pour interroger plusieurs tables . MongoDB est populaire et facile à utilisé en raison de sa simplicité et est “javascript friendly”, “json friendly”.