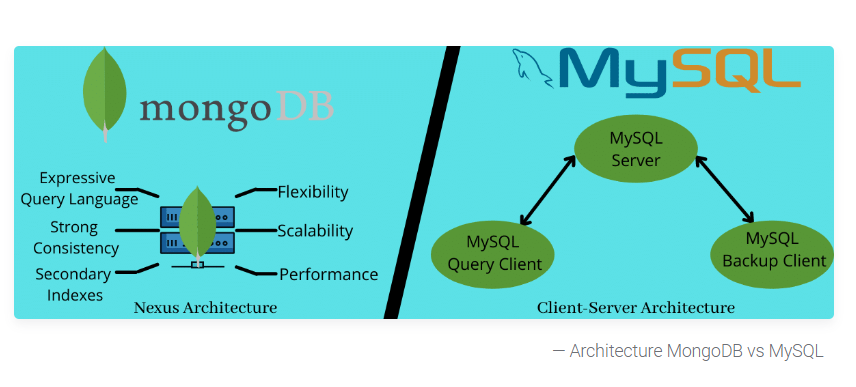
**Comparaison des SGBD MongoDB et MySQL**

Voici quelques paramètres sur la base desquels nous allons comparer MongoDB vs MySQL.

### Architecture



L’architecture constitue la base de tout système et établit le cadre dans lequel toutes les caractéristiques et fonctionnalités peuvent être introduites. Il est donc important de comparer l’architecture de MongoDB et de MySQL et de les comprendre de près afin de déterminer quel sera le meilleur choix pour votre application.

#### MongoDB

Il a pour philosophie de conception l’architecture Nexus, qui combine les fonctionnalités des bases de données relationnelles. Il peut répondre aux besoins des applications modernes en offrant une grande évolutivité, une disponibilité globale et un schéma flexible. Par conséquent, il est assez facile d’apporter des modifications à sa conception.

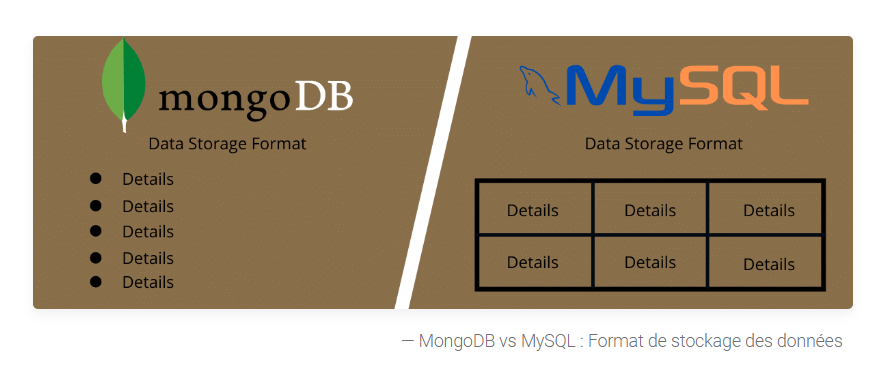
En outre, MongoDB comprend des pilotes officiels pour les principaux environnements de développement comme [AWS](https://kinsta.com/fr/blog/panne-aws/), Azure et Kubernetes, les langages de programmation comme JavaScript, Python, etc. et les frameworks comme Django.

#### MySQL

MySQL, quant à lui, comprend une architecture client-serveur avec un stockage optimisé pour offrir de hautes performances et le multi-threading. Sa documentation présente également quelques techniques d’optimisation des performances qui portent sur la configuration plutôt que sur le réglage fin des mesures SQL.

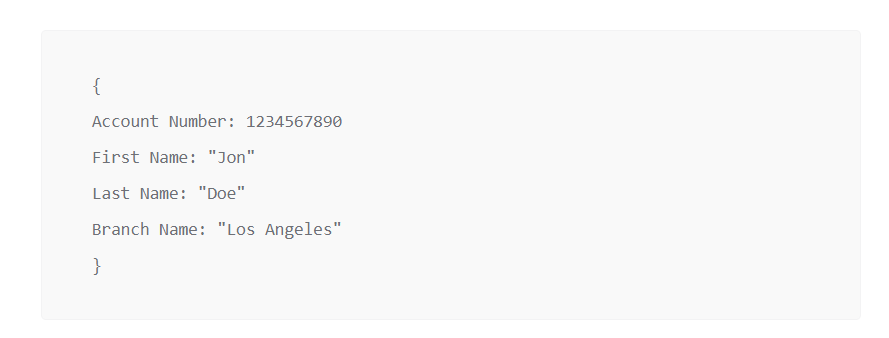
**Gagnant**: C’est une égalité.

### Format de stockage des données



#### MongoDB

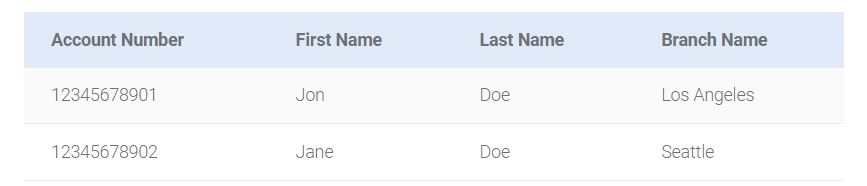
Le format de stockage des données dans MongoDB peut ressembler à ceci :



Vous pouvez voir que le format est de type JSON, et que vous pouvez facilement apporter des modifications aux données, c’est-à-dire que vous pouvez ajouter des données supplémentaires, supprimer certaines informations et modifier des données sans problème. L’exemple ci-dessus montre qu’il n’y a pas de schéma fixe pour la base de données, ce qui introduit plus de flexibilité.

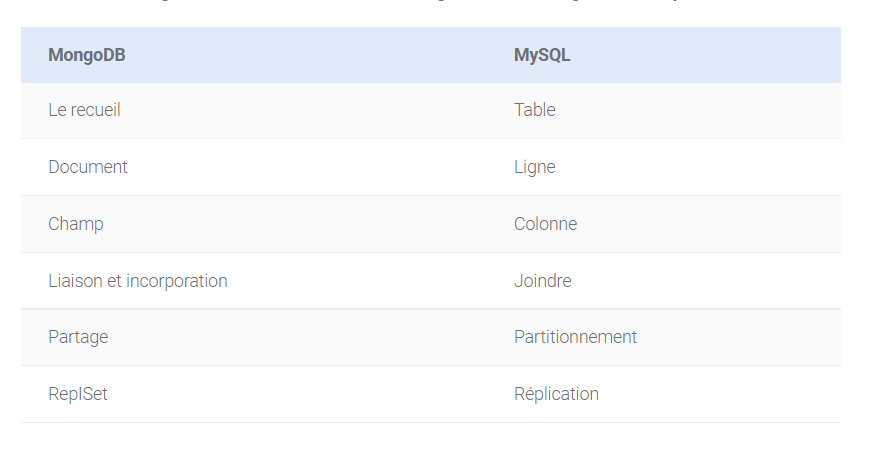
#### MySQL

Le format de stockage des données dans MySQL, d’autre part, ressemble à quelque chose comme ceci :

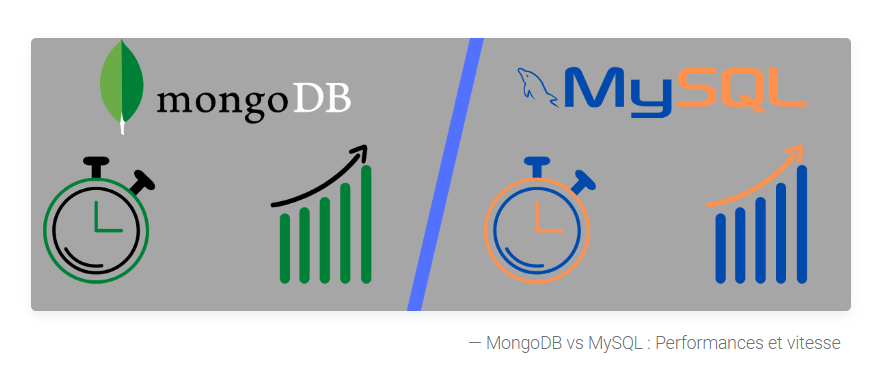


Le tableau ci-dessus montre comment MySQL organise les données sous forme de lignes et de colonnes. Il possède une structure propre et rigide qu’il est difficile de modifier par rapport à MongoDB. C’est parce que vous ne pouvez pas introduire une ligne ou une colonne spate ; vous devez faire la conception de telle sorte que pour chaque ligne, il y a une colonne et vice versa ; sinon, cela violera le schéma. Mais, dans MongoDB, vous pouvez facilement modifier les données.

Il existe une légère différence de terminologie entre MongoDB et MySQL :



### Performance et vitesse



#### MongoDB

Comme MongoDB stocke un grand volume de données non structurées et suit une approche de stockage basée sur les documents, il est relativement plus rapide que MySQL. Cela signifie que MongoDB stocke les données dans un seul document pour une entité et permet une lecture ou une écriture plus rapide des données. Des fonctionnalités comme la réplication peuvent y contribuer. Ses performances sont également meilleures lorsqu’il s’agit d’objets grâce à son stockage d’objets de type Jason.

En outre, MongoDB n’implique jamais de verrouillage du fournisseur, ce qui vous donne la liberté d’améliorer les performances en utilisant des alternatives si vous n’êtes pas satisfait d’un service.

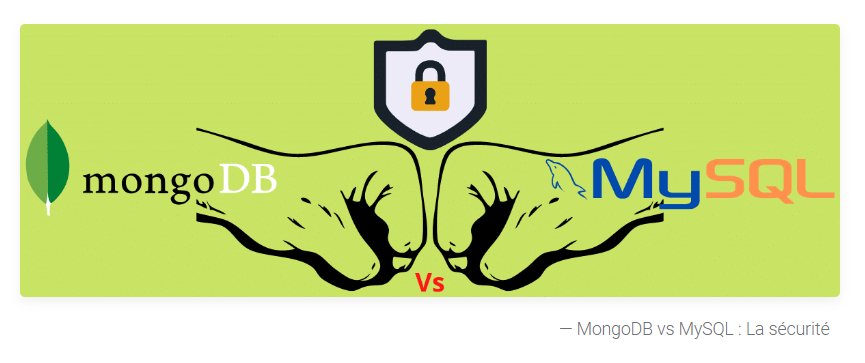
#### MySQL

MySQL peut présenter des performances lentes lorsqu’il s’agit de traiter un énorme volume de données. C’est parce qu’il stocke les tables d’une manière normalisée. Et si vous voulez modifier des données ou les extraire, vous devez passer par de nombreuses tables pour écrire et lire des données, ce qui augmente les charges du serveur et affecte ses performances. Mais vous pouvez opter pour MySQL pour les opérations transactionnelles.

Il nécessite une structure de données définie pour pouvoir ajouter des données dans la base de données. Par conséquent, il n’est pas adapté si vous souhaitez stocker des données non structurées. Et parfois, il devient également difficile de concevoir un schéma approprié lorsqu’il s’agit de données complexes.

**Gagnant**: MongoDB est plus rapide et plus performant.

### Sécurité



MongoDB exploite les contrôles d’accès basés sur les rôles avec des autorisations flexibles pour les utilisateurs et les appareils. Chaque utilisateur se voit attribuer un rôle en fonction duquel il reçoit des autorisations spécifiques pour accéder aux données et effectuer des opérations. Par exemple, les utilisateurs comme les employés de haut niveau ont des niveaux d’autorisation plus élevés, leurs privilèges sont donc plus importants.

C’est une façon de sécuriser votre base de données afin qu’aucun utilisateur non autorisé ou aucune attaque ne puisse accéder à votre base de données et l’exploiter. En outre, MongoDB facilite également le cryptage [TLS (Transport Layer Security)](https://kinsta.com/fr/blog/tls-1-3/) et un protocole de sécurité appelé [SSL (Secure Sockets Layer)](https://kinsta.com/fr/base-de-connaissances/tls-vs-ssl/) pour une sécurité accrue. Vous pouvez également écrire des documents chiffrés dans des collections de données avec une clé maîtresse pour obtenir un chiffrement des données au repos.

#### MySQL

D’autre part, MySQL dispose de contrôles d’accès basés sur les privilèges. Il prend également en charge les facilités de cryptage comme MongoDB avec un modèle d’authentification similaire, comprenant l’autorisation, l’authentification et l’audit. Vous pouvez accorder des rôles et des privilèges aux utilisateurs, leur permettant d’obtenir des autorisations d’accès pour les ensembles de données et les opérations. En outre, vous pouvez également demander TLS et SSL pour plus de sécurité.

Dans sa récente mise à jour, MySQL a également inclus la prise en charge du double mot de passe afin de garantir une plus grande sécurité pour l’accès aux données.

Bien que les deux semblent se livrer une compétition au coude à coude en matière de sécurité, MySQL est considéré comme plus sûr. La raison réside dans son architecture et son schéma rigides, qui offrent une meilleure cohérence et fiabilité des données.