

# MongoDB - Méthode 3: Migration de MMapV1 vers Wired Tiger

## Sommaire

- Qu'est que WiredTiger
- Quel format de donnée votre MongoDB utilise ?
- Quelle procédure utiliser ?
  - Migration sur le serveur de production
    - Avantages
    - Inconvénients
  - Migration des données volumineuses en utilisant un autre serveur
    - Avantages
    - Inconvénients
  - Migration d'un cluster MongoDB
    - Avantages
    - Inconvénients

## Qu'est que WiredTiger

La base de données va se fragmenter au fil des insertions/suppressions d'éléments et le volume des données va devenir plus faible que l'espace disque consommé ( dans `/var/lib/mongo` ).

Après la version 3, MongoDB propose un nouveau moteur de stockage de données appelé WiredTiger, qui permet d'avoir de meilleures performances et un espace disque consommé plus faible avec moins de fragmentation et une compression de donnée.

Il y a **2 situations** ou vous aurez à utiliser la procédure de migration de MMapV1 vers WiredTiger:

- Si vous avez installé votre Shinken à partir d'une version ANTÉRIEUR à la V02.06.00.
  - Le format de stockage était alors MMapV1.
- Si vous faites un shinken-restore d'une configuration sauvegardé sur un Shinken au format MMap1.
  - La sauvegarde des données de MongoDB garde le format de stockage.

## Quel format de donnée votre MongoDB utilise ?

Vérifier que le type de moteur de stockage est bien MMapv1 à l'aide de la commande suivante :

```
mongo shinken --quiet --eval "print(db.serverStatus().storageEngine.name)"
```

Le retour de la commande doit être **MMapv1**.



Si le retour est **wiredTiger**, alors la base de données est déjà migrée et vous n'avez pas besoin de continuer cette procédure.

## Quelle procédure utiliser ?

La migration se fait à froid, avec MongoDB éteint. Shinken devra donc par conséquent être stoppé plus ou moins longtemps durant la migration.

Nous disposons de trois procédures qui permettent de s'adapter à chacun de vos besoins.

Le choix de la procédure se fera donc en fonction de vos contraintes, notamment sur la durée d'interruption de service de Shinken ou la possibilité d'avoir une machine tierce avec une installation Shinken.

### Migration sur le serveur de production

Il est possible de migrer la base directement sur le serveur de production.

- Cette procédure permet de migrer sans transférer de données et ne nécessite pas de machine tierce.
- Néanmoins, Shinken devra être arrêté durant la totalité de la migration.

#### Avantages

- Méthode simple à mettre en œuvre.
- Ne nécessite pas de transfert de données.
- Ne nécessite pas de machine tierce.

## Inconvénients

- Shinken doit être arrêté durant la totalité de la migration

## Migration des données volumineuses en utilisant un autre serveur

Afin de réduire le temps d'arrêt de Shinken, il est possible de migrer les données les plus volumineuses sur une machine tierce (disposant d'une installation de Shinken), permettant à Shinken de continuer à délivrer son service pendant la migration.

- Les données évoluant pendant la migration, il est nécessaire de resynchroniser ces données au moment de l'intégration des données déjà migrées.
- Ces opérations doivent être réalisées dans une seule journée car le calcul des archives SLA est réalisé tous les jours.

## Avantages

- Le temps d'arrêt de Shinken est réduit.
- Possibilité de réduire la quantité des données SLA, si des jours sont stockés inutilement, pendant la migration sur l'autre serveur sans impacter la performance de la production.

## Inconvénients

- Nécessite de transférer les données sur une autre machine.
- La migration doit être effectuée dans la journée.

## Migration d'un cluster MongoDB

Cette procédure s'adresse aux installations utilisant un cluster MongoDB.

La méthode est plus rapide et le temps d'interruption de service se limite au temps d'intervention sur le nœud primaire du cluster.

## Avantages

- Méthode simple à mettre en œuvre.

## Inconvénients

- Ne s'applique que sur les clusters MongoDB.