

# MongoDB - maîtriser l'espace utilisé

## Limiter le stockage des données dans le temps

Les données stockées dans MongoDB peuvent être triées en deux catégories :

- Les données de configurations et des utilisateurs : il s'agit des données du Synchronizer et des paramètres utilisateurs sur l'interface de visualisations ( *Listes, Portails, Favoris, ...* ),
- Les données d'exploitation de la supervision ( *SLA, événements, ...* )

La taille des données de configurations et des utilisateurs va s'accroître lentement dans le temps et restera contenu à l'évolution de votre configuration. En revanche les données d'exploitations de la supervision sont des données qui vont s'accroître tous les jours, et la croissance sera proportionnelle à votre configuration.

Afin de limiter cette croissance dans le temps, il est possible de limiter la rétention de données d'exploitation afin de ne garder que les données pertinentes pour vous ( Voir [MongoDB - Méthode 1 : Ne garder que les données pertinentes.](#) )

Il est également possible de supprimer des anciennes données SLA via l'utilisation d'une commande si vous souhaitez supprimer des données anciennes.

## Récupérer l'espace disque

Durant son fonctionnement, MongoDB va ajouter et supprimer des données. Cela peut entraîner une fragmentation de votre base et il peut être nécessaire de récupérer de l'espace disque que MongoDB a acquis, mais dont il ne se sert plus.

Trois méthodes existent pour le compactage de la base :

- [Compactage de la base in-place](#) ( *nécessite un arrêt de Shinken pendant les opérations* ).
- [Migration de MMapV1 vers Wired Tiger](#) ( *nécessite un arrêt de Shinken pendant les opérations* ).

La première option ne nécessite pas le montage d'une autre base ni de transfert de données.

- C'est globalement plus simple, mais pendant que la base se compacte elle devient indisponible, ce qui peut provoquer un long temps d'indisponibilité.
- De plus, suivant le moteur de base utilisé les contraintes et les résultats sont variables :
  - **MMapV1**: le compactage sera efficace en termes de récupération d'espace, mais pendant le compactage, le double du volume de données sera utilisé, il faut donc prévoir assez d'espace disque.
  - **Wired Tiger**: le compactage est moins efficace, le moteur n'arrivant pas à récupérer tout l'espace perdu, mais par contre il se fait in-place sans consommer plus d'espace disque.

**Il est fortement recommandé de migrer sur le moteur Wired Tiger**, qui permet d'avoir de meilleures performances et un espace disque consommé plus faible ( *moins de fragmentation et compression de données* ).

- Le script de vérification de la fragmentation permet de donner le moteur de données utilisé : [MongoDB - surveillance du taux de fragmentation de la base](#) .
- Par contre, l'opération nécessitera un arrêt de la base MongoDB, donc vous devrez planifier une interruption de votre supervision.