

## Faut-il stocker les données dans la base de données ou dans un système de fichiers ?

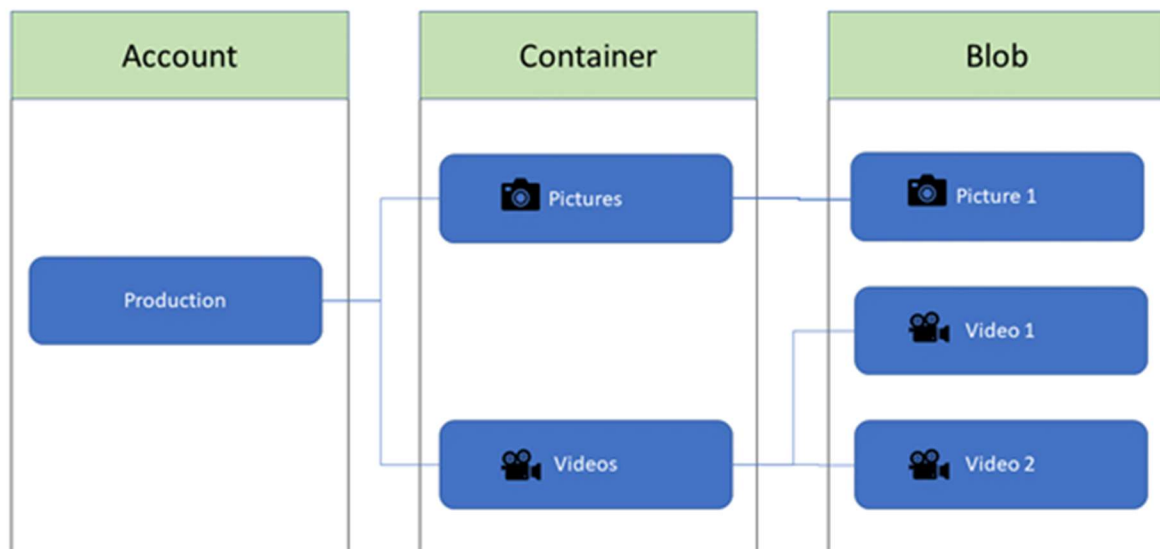
Tout d'abord en termes de données nous allons en avoir de différentes tailles. Il peut y avoir par exemple des BLOBs (Binary Large Object) qui sont de gros objets le plus souvent de type image, vidéo ou son. C'est dans l'industrie du big data que les BLOBs sont principalement utilisés, où les données de visiteurs de site Internet sont collectées en masse. Ils peuvent également être utilisés pour enregistrer les films, émissions, etc. de façon quasiment chiffrée. Il est à noter que les BLOBs sont sauvegardés de différente manière par les différents systèmes de bases de données.

C'est donc surtout pour ce type de donnée qu'il va falloir déterminer où les stocker.

Comment le stockage de BLOB fonctionne-t-il ?

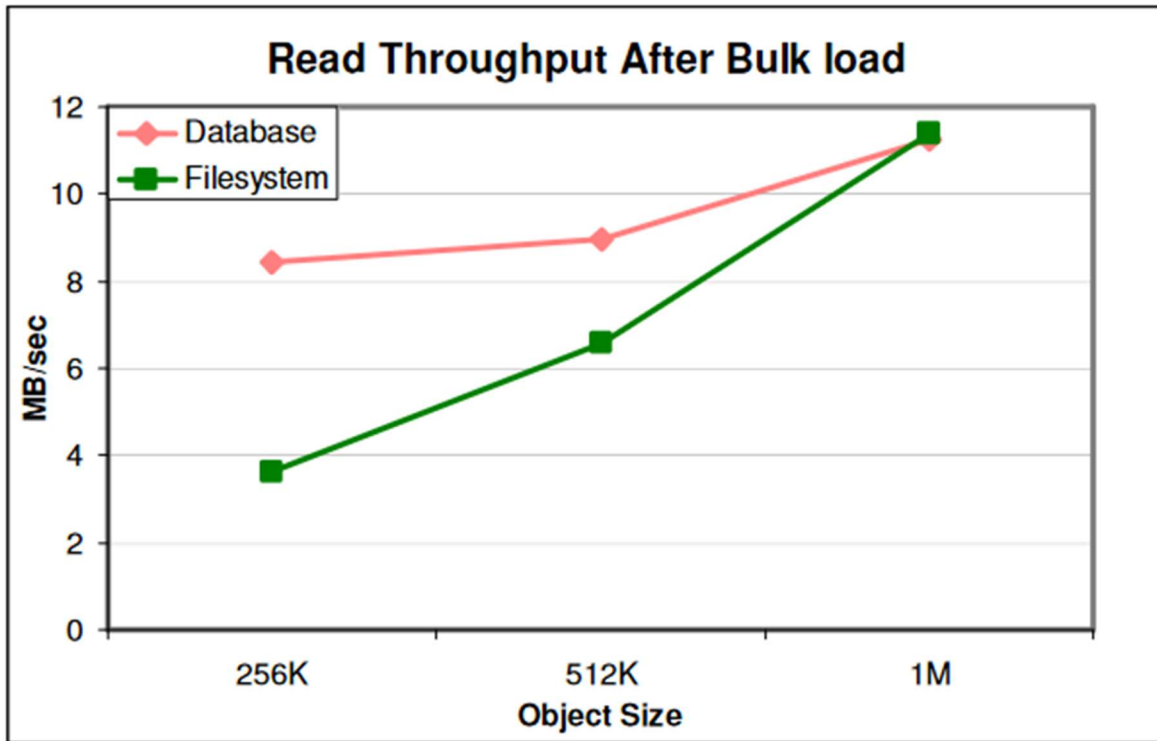
Dans le cas de Azure BLOB Storage, trois ressources différentes sont utilisées :

1. Le compte de Storage (« stockage »)
2. Le conteneur du compte Storage
3. Le blob dans le conteneur



Un conteneur BLOB s'apparente à un répertoire dans un système de fichiers qui organise les BLOBs. Un compte de Storage peut avoir un nombre illimité de conteneurs, et un conteneur peut stocker un nombre illimité de BLOBs.

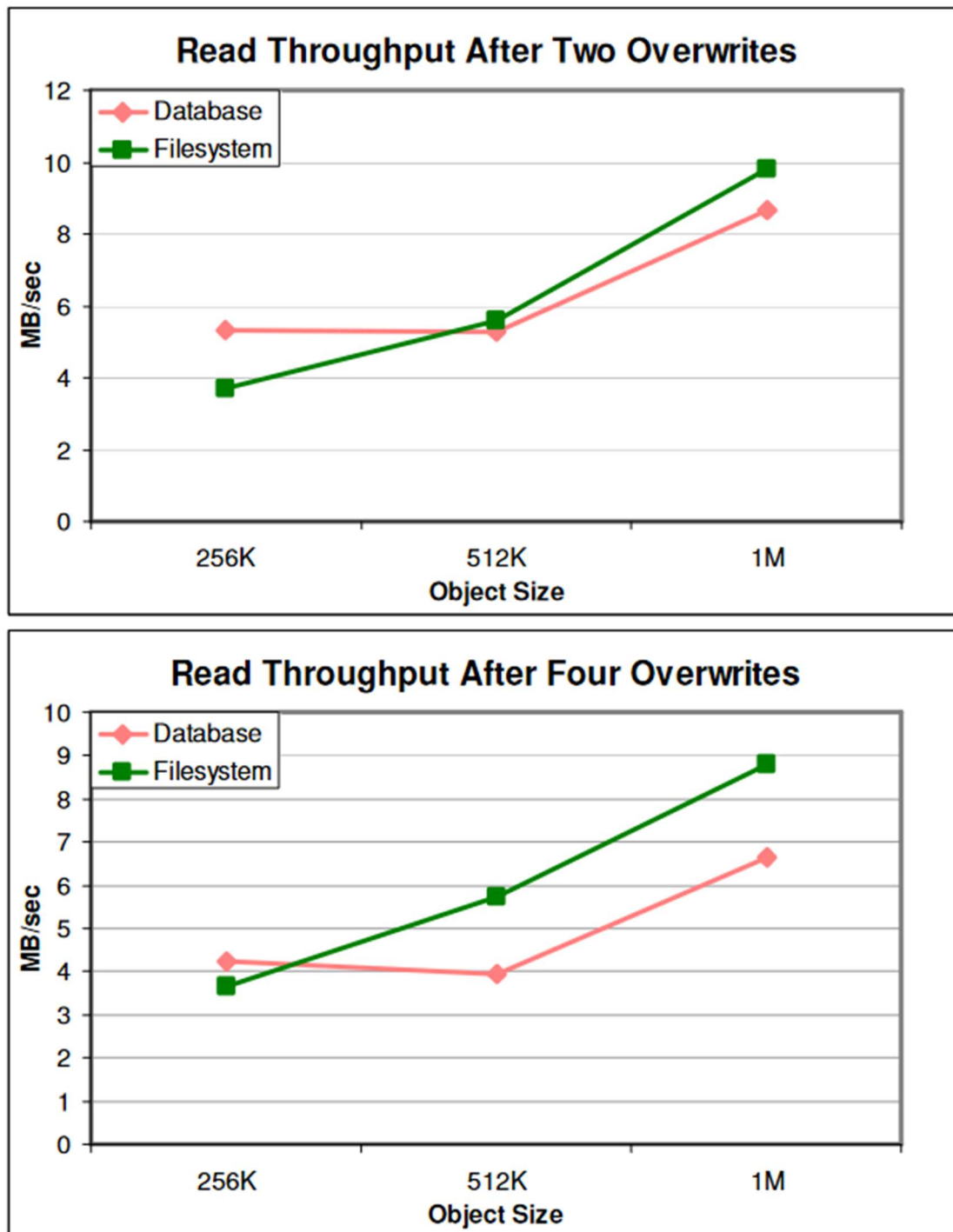
Microsoft a réalisé une étude à propos du stockage de BLOB, qui compare un système de fichier (NTFS) et une base de données (SQL Server) au niveau de leurs différentes performances de lecture et d'écriture des BLOBs, et cela en fonction de différents paramètres (fragmentation, réécriture répétée ou non des objets, taille des BLOBs).<sup>1</sup>



**Graphique 1 : Débit de lecture immédiatement après le chargement des données. Les bases de données sont plus rapides pour les BLOBs de petite taille. Les performances du NTFS s'améliorent au fur et à mesure que la taille de l'objet grandit.**

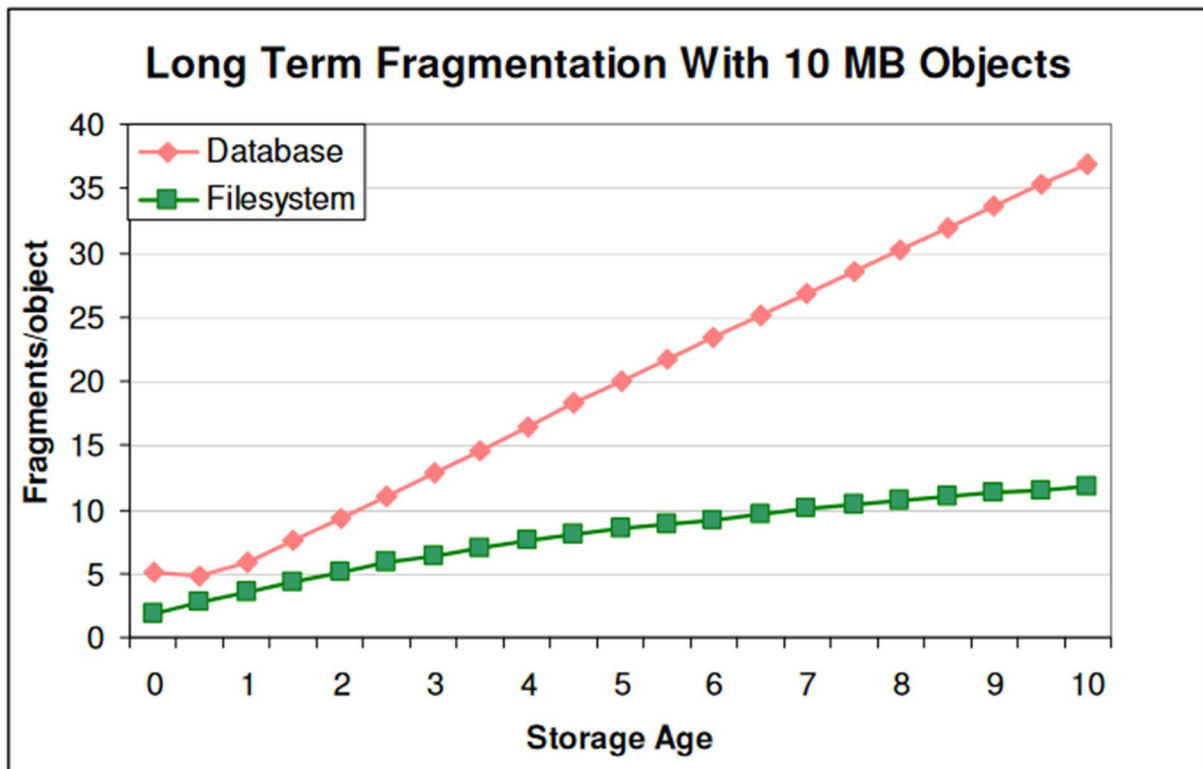
<sup>1</sup> <https://www.microsoft.com/en-us/research/wp-content/uploads/2006/04/tr-2006-45.pdf>

Cependant cette même étude montre aussi que pour des BLOBs dont la taille est supérieure à 256KB, les performances de lecture de la base de données sont inférieures à celle d'un système de fichier à cause de la fragmentation au fil du temps :



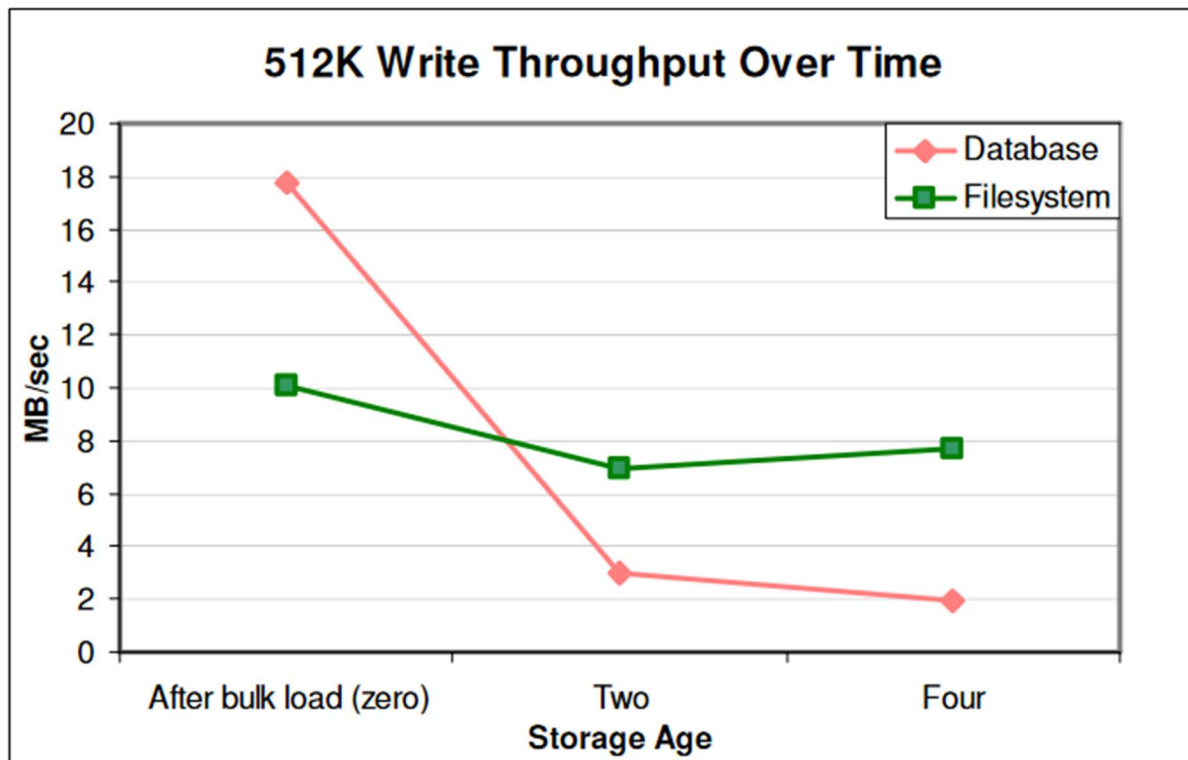
**Graphique 2 : La fragmentation provoque des baisses de performance au fil du temps pour le débit de lecture. Le système de fichier est moins impacté que SQL Server. Sur la durée NTFS surpasse SQL Server quand la taille des BLOBs est supérieure à 256KB.**

Avec le graphique suivant on peut voir que la fragmentation pose vraiment problème pour la base de données avec des BLOBs dont la taille est égale à 10 MB :



Graphique 3 : Pour les gros BLOBs, NTFS gère la fragmentation plus efficacement que SQL Server. « Storage Age » correspond au nombre de fois que chaque objet a été remplacé par une nouvelle version. Un objet dans un secteur contigu sur le disque a 1 fragment.

Ensuite, les performances concernant le débit d'écriture de la base de données est lui aussi impacté sur la durée :



**Graphique 4 : Bien que SQL Server écrive rapidement un important volume de données, les performances sont diminuées quand les objets existants sont remplacés.**

Il est à noter qu'en conclusion de cette étude, il a été dit que NTFS, donc un système de fichiers, avait un avantage net lorsque les BLOBs étaient plus larges que 1 MB en moyenne, et que si les BLOBs étaient inférieurs à 256 KB c'était la base de données qui avait l'avantage.

Enfin quand la taille du BLOB varie entre 256 KB et 1 MB, le choix entre la base de données et le système de fichiers dépend d'autres facteurs : combien de fois les objets vont devoir être remplacés / réécrits et si la base de données devra plus effectuer des opérations de lecture ou d'écriture.

Stocker dans la base de données avec un BLOB présente différents avantages et inconvénients.

En termes d'avantages, avoir un BLOB dans la base de données permet de :

- Conserver les propriétés ACID, donc l'intégrité et l'atomicité
- Avoir de meilleures performances pour le stockage d'objets dont la taille est en moyenne inférieure à 1MB comme constaté dans l'étude de Microsoft
- Offrir la possibilité d'intégrer de grosses données binaires pouvant être enregistrées simplement (en effet les BLOBs sont collectés sous forme brute et n'ont donc aucune structure) et référencées sans difficulté.
- Les données sont toutes contenues dans les sauvegardes de la base de données

En termes d'inconvénients :

- Impact sur la performance de la base de données
- Opérations de sauvegarde / restauration très chronophages en raison de la taille des fichiers.
- Prend de l'espace sur la base de données, cette dernière consommant plus de mémoire lors des requêtes
- Toutes les bases de données ne permettent pas l'utilisation de BLOB

En conclusion, la méthode du BLOB directement dans la base de données semble la plus judicieuse étant donné que l'on doit noter cas stocker des documents pour des avocats et que l'intégrité de ces données est primordiale.