

Aujourd'hui, on va évoquer le service
SQL Managed Instance (SQL MI).

SQL MI est un service de base de données (DB) qui offre
la plus grande compatibilité de moteur de DB SQL.

C'est un service managé ?

Oui à l'image de SQL Database.

SQL MI est parfait pour effectuer des migrations depuis
des environnements On-Premise, en mode lift and shift.

Si je comprends bien, il permet de migrer plus
simplement vers Azure en réduisant les
modifications des applications et des DB ?

Exactement

Cela permet une transition vers le cloud plus simple à
appréhender, tout en modernisant vos applications.

Et la migration n'est
pas trop compliquée ?

Non. L'équipe produit propose **Azure Data Migration
Service (ADMS)** qui est une solution automatisée pour
accompagner les clients dans leur projet de migration.

Comment attribue t-on les ressources à SQL MI ?

Au travers du modèle **vCore** qui permet de
choisir le nombre de coeurs, la quantité de
mémoire et la taille de stockage nécessaire.

Super. Et je suppose que je peux faire évoluer la quantité de
ressource en fonction de ma charge de travail ?

Evidemment.

D'ailleurs, 3 configurations sont proposées : **Standard**,
Premium ou **Premium à mémoire optimisée**.





$$C_4 = C_4^{\alpha}(\alpha - \alpha.)$$



En plus, SQL MI est décliné en deux niveaux de service.

Le 1er, l'**usage général**, conçu pour des applications avec des exigences de performances et une latence standard.

Le 2nd, le **niveau critique**, pour les applications qui nécessitent une très faible latence.

Très bien. Comme cela, en fonction de la criticité de notre application, nous avons le choix.

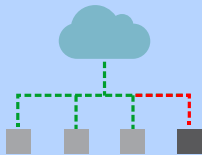
Concernant la partie réseau, un subnet dédié est nécessaire pour l'utilisation de SQL MI.

De plus, un NSG et une table de routage sont automatiquement associés à ce subnet, afin d'optimiser la connexion avec les autres services Azure.



Vraiment cool !

Une autre question, doit-on obligatoirement exposer SQL MI sur internet pour le consommer ?



Non pas du tout.

Tu peux bien sûr choisir de l'exposer sur internet mais pas seulement.

Tu a aussi la possibilité d'utiliser un point de terminaison privé pour des raisons de sécurité.



Top !

Et comment se passe l'authentification à la DB ?



Tu as le choix, soit tu peux utiliser **Azure Active Directory**, soit une **authentification SQL**. Et si nécessaire, tu peux même combiner les 2 modes d'authentification.

C'est génial car cela peut effectivement dépendre de notre cas d'usage.



Tu parlais de migration de données tout à l'heure ?

Oui c'est vrai.

Il existe différents outils pour migrer tes données depuis des DB de type DB2 et Oracle.

Mais aussi depuis des DB hébergées chez d'autres fournisseurs Cloud que cela soit sur des VMs ou sur des services PAAS concurrents.

Excellente nouvelle pour celui qui souhaite centraliser ses données sur Azure.

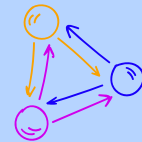
Je suppose qu'il existe aussi une option de sauvegarde ?

Evidemment !

SQL MI offre un système de sauvegardes automatisées pour les **sauvegardes complètes**, **différentielles** et les **journaux de transactions**.

Les données sont sauvegardées dans des comptes de stockage, et sont donc répliquées soit en mode **LRS**, **ZRS**, **GRS** ou **GZRS**, en fonction de ce que tu définis.

Ah effectivement, on avait déjà évoqué les différents types de répliation par le passé.



Et évidemment tu peux définir les rétentions de ton choix, qu'elles soient à court terme (**PITR**) ou à long terme (**LTR**).

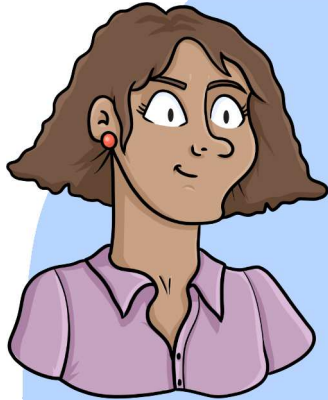
Tu fais bien de le rappeler.

Pour le court terme, la rétention max est de 35 jours, alors que celle à long terme peut aller jusqu'à 10 ans.

Tu as effectivement une bonne mémoire.

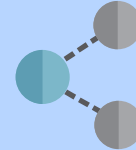


Il existe aussi, la notion de groupes de basculement automatiques, qui permettent de gérer la réplication et le basculement des DB vers une autre région Azure en cas d'incidents.



Comme pour SQL Database.

Et en terme de sécurité, je suppose qu'il y a différentes options ?



A ton avis ?!

SQL MI peut être protégé par **Microsoft Defender for Cloud** pour identifier, par exemple, les vulnérabilités potentielles des DB.



Il peut aussi remonter des alertes par **injection de code SQL** ou une **activité suspecte** au niveau de la DB.

Et en terme de chiffrement ?

Deux options s'offrent à toi :

La 1ère avec le **chiffrement au repos**, ce qu'on appelle **TDE**.

Et la 2nde, **Always Encrypted** qui permet de chiffrer qu'une partie des données dites sensibles, comme des numéros de cartes bancaires ou d'autres données personnelles.

Et bien sûr, tu peux exporter les logs d'audit, vers des comptes de stockage par exemple, qui pourront être exploités par l'équipe sécurité, si nécessaire.

Ainsi, SQL Managed Instance est, entre autres, un excellent candidat lorsqu'on souhaite migrer rapidement nos charges de travail SQL sur Azure.



Exactement, tout dépend de ton cas d'usage et des différentes contraintes de ton application.





Si vous souhaitez continuer à **apprendre**, de façon ludique, sur **l'écosystème Azure**, et ne rater aucune de nos illustrations ...

... N'hésitez pas à vous abonner sur :



<https://aka.ms/grow-una>



<https://www.youtube.com/@grow-una>

Et si le contenu vous plaît, partagez-le ;o)

A très vite !

