La Minute CLOUD de JULES & LÉA

Azure Cache for Redis #1

Aujourd'hui, nous allons évoquer le service **Azure Cache for Redis**.

Vu le nom, je suppose qu'on s'appuie sur la solution Redis ?

Exactement.

Azure Cache for Redis est un service managé de Microsoft qui s'appuie sur Redis, qui est un **système de stockage** de données clé/valeur **en mémoire**.

On l'appelle également **magasin de données (Datastore)**.

L'un de ses rôles les plus connus, est d'**accélérer les résultats d'une recherche** sur un site web.

Azure Cache for Redis propose soit Redis Open Source (**OSS Redis**), soit un produit commercial, **Redis Entreprise**.

Pourquoi les données sont stockées en mémoire ?

Stocker les données en mémoire, permet de les transmettre plus rapidement que si celles-ci étaient stockées en base de données (DB) ou sur un disque.

C'est donc un système de stockage de données à faible latence ?

Tout à fait.

Ce qui en fait un service très utilisé par les applications qui nécéssitent une **faible latence**, et un **haut débit de stockage**.

Et peut-on le coupler avec un système de base de données ?

Sans aucun problème.

Il peut être utilisé en tant que **service autonome** ou **en parallèle** avec d'autres services de DB comme SQL Databases par exemple.

suivante

La Minute CLOUD de JULES & LÉA

Azure Cache for Redis #2

Azure Cache for Redis **améliore les performances des applications** grâce aux différents cas d'usage qu'il propose.

Evidemment, seules les données les plus utilisées, doivent être stockées, essentiellement pour des raisons de coûts.

Sachez que l'on peut également cacher du **contenu statique** qui change rarement.

Mais aussi, **des sessions utilisateurs** comme pour les paniers d'achats d'un site en ligne.

Il faut donc bien identifier le type d'informations à stocker dans Azure Cache for Redis.

Tu as tout à fait raison.

Ce service est aussi utilisé pour stocker **des travaux** ou **des messages en files d'attente**, ou bien des **transactions distribuées**.

Je suppose qu'Azure propose différents niveaux de service ?

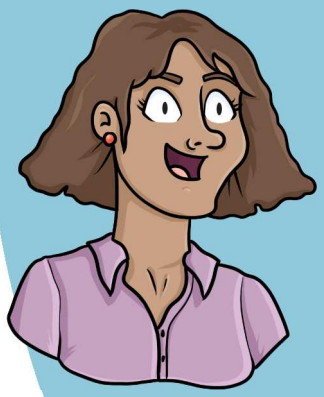
Evidemment.

Tu as le niveau de **Base** où OSS Redis tourne sur une VM, sans SLA, parfait pour les phases de développement ou de tests.

Le niveau **Standard** où cette fois-ci OSS Redis tourne sur 2 VM répliquées.

Le niveau **Premium** où OSS Redis tourne sur des VM plus puissantes pour offrir de meilleures performances.

Le niveau **Entreprise** avec l'édition Redis Entreprise qui offre des modules Redis supplémentaires et une disponibilité encore plus importante.



Et enfin le niveau **Entreprise Flash**, avec Redis Entreprise, qui offre de plus grandes capacités de stockage sur des disques SSD pour réduire le coût du service.

suivante

La Minute CLOUD de JULES & LÉA

Azure Cache for Redis #3

Il faut donc savoir ce que l'on souhaite faire avant de choisir son niveau de service.

Comme toujours, il faut y réfléchir un peu avant de se lancer.

Tu peux retrouver sur le site de Microsoft* un comparatif entre les différents niveaux de service pour t'aiguiller en fonction de ton cas d'usage.

Oui, je pense que c'est plus prudent d'y jeter un oeil.

Sachant que tu peux faire évoluer ton niveau de service à tout moment, par contre tu ne peux pas passer à un niveau inférieur, par exemple d'un Premium à Standard.

Autre contrainte, il est **impossible de passer** d'une version **OSS Redis** à une version avec **Redis Entreprise**.

Cela me semble plutôt logique mais tu fais bien de le préciser.

Concernant les versions, **Azure Cache for Redis** prend en charge OSS Redis **version 4.0.X** et **6.0.X**.

Je le précise car elles n'offrent pas les mêmes fonctionnalités.

Je peux mettre à jour la version de 4 à 6, si je le souhaite ?

Oui tout à fait, c'est supporté.

En revanche après tu ne pourras pas faire machine arrière.

Entendu.

Concernant les versions Entreprise et Entreprise Flash, elles offrent des modules supplémentaires comme **RedisBloom**, **RedisTimeSeries** et **RedisSeard**.

Merci pour l'info, j'irai me documenter pour savoir ce que cela apporte comme fonctionnalités.

Excellent réflexe !

suivante

* https://docs.microsoft.com/fr-fr/azure/azure-cache-for-redis/cache-overview#feature-comparison

La Minute CLOUD de JULES & LÉA

Azure
Cache for Redis #4

Je suppose qu'Azure offre des mécanismes de **haute disponibilité** et de **récupération d'urgence** ?

Tout à fait.

Ces mécanismes diffèrent en fonction du niveau de service que tu as choisi.

Le mode **Réplication Standard,** réplique les données sur 2 noeuds Redis au sein d'un Datacenter.

Le mode **Redondance de Zone**, réplique les données sur plusieurs noeuds Redis répartis dans différentes zones de disponibilité (ZA).

Le mode **Géo-replication,** réplique les données dans une 2nde région Azure.

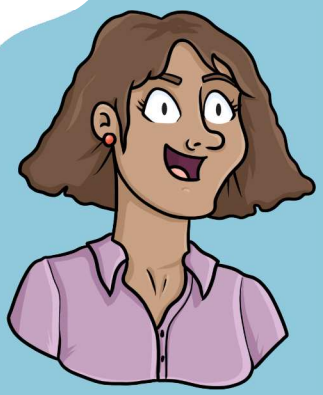
Le mode **Import/Export**, qui comme son nom l'indique permet d'effectuer des imports et des exports de données depuis ou vers un compte de stockage.

Et enfin le mode **Persistance** qui permet de sauvegarder vos données dans un compte de stockage.

Le mode de Persistance est manuel ou automatique ?

Complètement automatique.

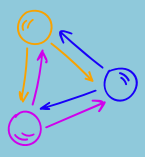
Il existe d'ailleurs 2 options de persistance.



La 1ère, la **persistance RDB** (Redis Database) qui enregistre un snapshot du cache au format binaire. C'est l'utilisateur qui définit la fréquence des snapshots qui va de 15 minutes à 24 heures.

Et la 2nde, la **persistance AOP** (Append only File) qui enregistre chaque opération d'écritures dans un journal, ce qui implique forcément un impact sur les performances de votre Redis, surtout si il est très sollicité.

Très pratique si on souhaite restaurer les données en cas de défaillance matérielle.



Exactement et saches que si besoin, on peut avoir jusqu'à **3 réplicas** Redis, c'est à dire 3 noeuds, sur le niveau **Premium**.

C'est vraiment intéressant pour de la haute disponibilité !

suivante

La Minute CLOUD de JULES & LÉA

Azure
Cache for Redis #5

En plus des niveaux de services qu'on a vu tout à l'heure, tu peux aussi définir la taille de ton cache Redis.

C'est à dire la **taille de la mémoire allouée** à ton instance.

Et sur quels éléments se baser pour définir sa taille et potentiellement la faire évoluer ?

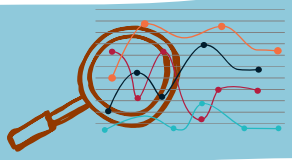
Tu peux te baser sur différents éléments, et si besoin, effectuer une mise à l'échelle :

Tout d'abord la **charge du serveur Redis**. Si elle est souvent élevée, il faut envisager, soit un mode cluster, c'est-à-dire, plusieurs instances pour répartir la charge, ou alors une mise à l'échelle.

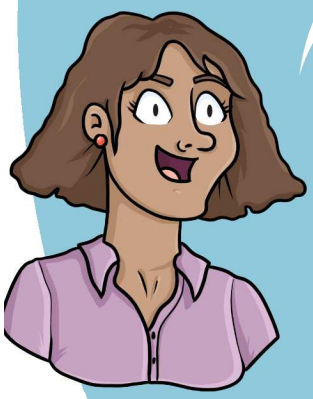
Ensuite, l'**espace de stockage** dont vous allez simplement avoir besoin pour stocker vos données.

Il y a aussi les **connexions clientes**, qui sont limitées par la taille du cache. Au besoin, tu peux donc soit utilisier le mode cluster, soit changer la taille du cache.

Et enfin, la **bande passante réseau** qui peut être insuffisante pour traiter le nombre de requêtes. Il faut alors envisager un changement de la taille du cache.



Et aujourd'hui les mécanismes de mise à l'échelle se font de manière manuelle ?



Oui essentiellement pour des raisons techniques.

Tu peux procéder de 2 manières différentes :

Soit avec un **Scale-up**, c'est à dire que tu passes à une taille de cache plus élevée.

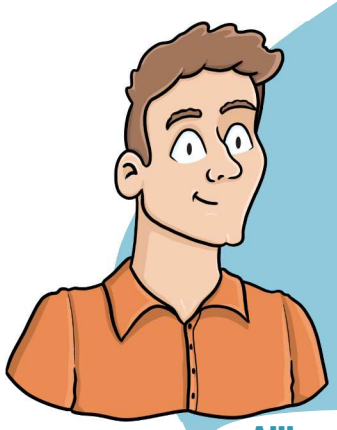
Soit avec un **Scale-down**, c'est lorsque tu ajoutes un réplica à ton cluster Redis.

Super, j'ai tout compris.

suivante

La Minute CLOUD de JULES & LÉA

Azure
Cache for Redis #6



Il existe aussi un mode de **gestion de la mémoire** avant d'envisager un changement de taille du cache.

Redis intègre nativement un mécanisme permettant de supprimer certaines données, on appelle cela **les politiques d'éviction**, il en existe plusieurs :

Noeviction : Les nouvelles valeurs ne sont pas enregistrées lorsque la limite de mémoire est atteinte.

Allkeys-lru : Conserve les clés les plus récemment utilisées et supprime les clés les moins récemment utilisées (LRU)

Allkeys-lfu : Conserve les clés fréquemment utilisées et supprime les clés les moins fréquemment utilisées (LFU)

Volatile-lru : Supprime les clés les moins récemment utilisées avec le champ *Expire* défini sur true.

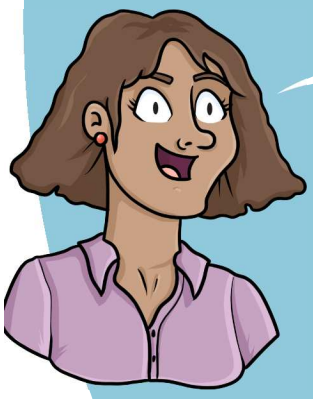
Volatile-lfu : Supprime les clés les moins fréquemment utilisées avec le champ *Expire* défini sur true.

Allkey-random : Supprime aléatoirement des clés pour faire de la place pour les nouvelles données ajoutées.

J' ♡ Redis

Volatile-random : Supprime de manière aléatoire les clés avec le champ *Expire* défini sur true.

En enfin, **Volatile-ttl** : Supprime les clés les moins fréquemment utilisées avec le champ *Expire* défini sur true et la valeur de durée de vie restante (TTL) la plus courte.



Très ingénieux comme mécanisme pour faire du ménage.

En plus des politiques d'éviction, il existe 2 autres paramètres importants dans Redis :

Maxmenory-reserved, qui détermine la quantité de mémoire allouée aux opérations de non cache comme la réplication des données.

Et **Maxfragmentationmemory-reserved** qui détermine la quantité de mémoire allouée à la fragmentation de la mémoire.

Merci Philippe, aujourd'hui j'ai découvert un service qui me plait déjà beaucoup.

La Minute CLOUD de JULES & LÉA

Merci à vous !

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://tinyurl.com/youtube-growuna

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

A très vite !

GROW UNA