



Hello Jonah, on entend souvent parler des Functions, tu peux nous dire ce que c'est ?

Evidemment Jules.

**Azure Function** est un service **serverless** entièrement **managé** par Microsoft qui te permet d'exécuter du code, par exemple en réponse à un évènement.

C'est quoi exactement du **serverless** ?

C'est un mode de développement qui permet d'exécuter du code sans avoir à gérer l'infrastructure. Ainsi le développeur se concentre uniquement sur son travail à savoir le développement.



Ca semble plutôt sympa !



C'est super. De plus **Azure Function** s'intègre parfaitement avec **les autres services Azure**. Il peut être déployé sur un environnement Linux ou Windows.

L'un des avantages des Functions, est **le nombre de langages et de Frameworks supportés** : .Net, Node.js, Java, PowerShell, Python, ...

Aujourd'hui, 2 modes de déploiement sont proposés :

Le mode **Code**, permet simplement de choisir un Framework et la version.

Le mode **Container** permet d'instancier un container docker avec l'image de ton choix.

J'aime beaucoup le principe du **serverless** qui prend de plus en plus de place de nos jours.



Mais du coup, où est-ce que tu stockes ton code ?

Excellente question.

Le **code**, mais aussi la **configuration** de la Function sont enregistrés dans un **compte de stockage**, ce qui offre un stockage persistant.

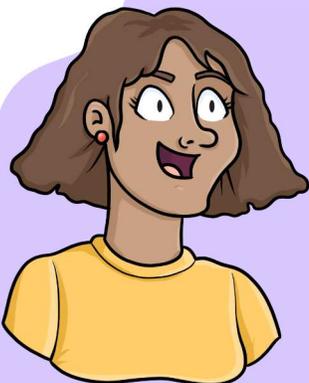
De base, une Function est **sans état (stateless)**, c'est à dire qu'elle ne stocke aucune information liée à son exécution.

Il existe une extension, **Durable Function**, qui permet d'écrire des Functions en gérant l'**état (statefull)**, très utile lorsque tu as un workflow de Functions qui s'enchainent.

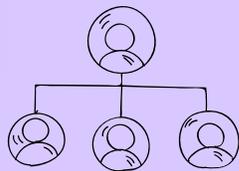


Comme il y a 2 modes de déploiement, comment sont gérées les ressources associées ?

Les ressources dépendent du plan d'hébergement que tu choisis :



Le **plan Consumption** ou **Serverless** où les ressources sont fournies directement par Microsoft. Très utile lorsque tes besoins ne sont pas systématiques ou très courts dans la durée. Tu payes uniquement les ressources consommées lors de l'exécution.



Le **plan Service Plan**, où tu provisionnes un pool de ressources CPU et Mémoire, comme pour les WebApp. C'est pratique lorsque les exécutions sont plus fréquentes avec des temps d'exécution plus longs.

Et enfin, le **plan Premium**, avec des instances plus puissantes fournies par Microsoft. C'est particulièrement utile lorsque les Functions s'exécutent presque en continue. Et en plus tu bénéficies de fonctionnalités supplémentaires comme l'intégration à un **VNET**.

suyvante



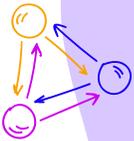
Et la facturation dépendra du plan que tu as choisis ?

Oui, c'est pour cela qu'il faut bien étudier et sélectionner celui qui correspond le mieux à ton **cas d'usage**.

Garde en tête que le plan Consumption n'est pas pris en charge pour le mode Container.

Tu fais bien de le préciser !

Concernant le développement des Fonctions, tu peux le faire sur l'IDE de ton choix, mais **Visual Studio Code** est parfait en plus il propose plein d'extensions gratuites.



Et l'intégration avec les autres services Azure ?

**Azure Function** est souvent utilisé avec **Azure WebApp** ou avec **Azure Logic Apps** qui est aussi un service serverless.

Une Function peut être déclenchée de plusieurs manières. La 1ère par une simple **requête HTTP/S** sur son URL, ce qui rend très simple l'intégration avec les services Azure, mais aussi des services hébergés en dehors d'Azure.

Une Function peut aussi être déclenchée à intervalles réguliers via la configuration d'un **schedule**. Mais aussi être déclenchée par **Event Grid** ou une **file d'attente**.

Une intégration simple avec plusieurs services, des déclencheurs, ainsi que différents frameworks, l'équipe produit à fait un excellent travail !

C'est aussi ce qui explique sa **popularité**.

suivante



Et en terme de **sécurité**, je suppose que Azure Function s'appuie sur les services proposés par Azure ?



Evidemment, il est parfaitement intégré à la solution **Microsoft Defender for Cloud**.

Même si **Azure Function** gère les requêtes HTTP, il est fortement recommandé de **privilégier les requêtes HTTPS**.

En plus on peut même **limiter l'accès** à la Function avec les **clés API**.



Et je suppose qu'il n'y a pas de souci d'intégration avec **Azure Key Vault** ?!



Exact, ce qui est très pratique lorsque tu as besoin de stocker des informations sensibles.



Tu peux aussi l'intégrer avec **Application Insight** pour récupérer les **logs**, les **métriques de performances** mais aussi les **erreurs**.

Le **plan Premium**, propose en plus de déployer Azure Function dans plusieurs **Zones de Disponibilité (ZA)**.

Tu nous as fait découvrir un service que je vais m'empresse d'approfondir, car il répondra à une multitude de cas d'usages que nous avons chez nos clients.



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 LinkedIn à l'adresse :

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

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

A très vite !

