

Hyperledger Fabric : une v2.0 plus souple avec les smart contracts

Paramétrage, installation, exécution, mise à jour... La gestion des *smart contracts* évolue sur Hyperledger Fabric avec le passage en [v2.0](#).

Sur la partie gouvernance :

- Avec les versions précédentes, une organisation pouvait, seule, définir les paramètres applicables à tous les autres membres de son canal.
Ce modèle centralisé reste pris en charge, mais s'accompagne désormais d'un mécanisme décentralisé qui permet de conditionner le paramétrage d'un smart contract à l'accord de plusieurs organisations.
- Dans le même esprit, on peut maintenant exiger qu'un nombre minimum d'organisations approuvent les mises à jour des *smart contracts*.
- Plus besoin de restructurer ou de réinstaller un *smart contract* en cas de mise à jour de certains éléments. En l'occurrence :
 - 1) Les politiques d'approbation
 - 2) Les « collections », destinées à établir des canaux privés de partage de données entre membres d'un canal
- Pour en simplifier l'inspection, les *smart contracts* sont empaquetés sous la forme de fichiers tar.
- Traditionnellement sur Hyperledger Fabric, chaque smart contract a un nom et un numéro de version spécifiés à l'installation.
La v2.0 permet de déployer de multiples versions d'un même *smart contract* sur un même canal – ou sur plusieurs canaux – avec des noms différents.
- Les organisations membres d'un même canal peuvent adapter un *smart contract* à leurs besoins, aussi longtemps que le consensus sur les transactions qui en résultent est atteint.

Smart contracts « as a service »

Sur la partie mise en œuvre :

- Les nœuds n'ont plus nécessairement besoin d'accéder à un démon Docker pour le développement et l'exécution de *smart contracts*.
- L'exécution des *smart contracts* peut se faire dans d'autres environnements que les conteneurs Docker*.
- Il devient possible d'exécuter des *smart contracts* en tant que service externe, par exemple dans un pod Kubernetes.

Hyperledger Fabric introduit aussi des améliorations au niveau du partage de données. Entre autres :

- Des politiques d'approbation au niveau des collections (elles ont priorité sur les politiques définies dans les *smart contracts*)
- Une API pour contrôler l'intégrité des données échangées avec une organisation membre d'un canal mais non membre d'une collection

Le processus de mise à jour vers la dernière version d'Hyperledger Fabric a désormais une [section dédiée](#) dans la documentation du projet.

** Avec la v2.0, les images Docker utilisent Alpine Linux, distribution légère orientée sécurité.*

Logo © Hyperledger