Qu'est-ce que le langage de programmation Move ?

Comprendre le logiciel derrière Diem

Le mois dernier, Meta a annoncé la vente de son stablecoin Diem à Silvergate, la première banque dans le domaine des crypto-monnaies. (Pour un aperçu détaillé de cette vente et de Silvergate, reportez-vous à notre récent rapport). Silvergate gère le réseau d'échange Silvergate (SEN), un service de transfert d'actifs 24 heures sur 24 et 7 jours sur 7 utilisé par de nombreuses exchanges de crypto de premier plan. Silvergate prévoit d'utiliser l'infrastructure Diem qu'elle a acquis pour introduire un stablecoin adossé au dollar.

Le langage de programmation Move est un composant essentiel du package Diem racheté par Silvergate pour près de 200 millions de dollars. Si le projet Diem (anciennement Libra) a pris autant de temps chez Meta, c'est en grande partie à cause de la création d'un langage de programmation propriétaire destiné à la crypto-monnaie. Dans cet article, nous allons examiner les points forts de Move et son utilisation future tandis que Silvergate prépare l'avenir de Diem.

Termes clés

Avant d'aller plus loin, nous devons définir certaines notions importantes. Attention, cet article est assez technique.

Langage de programmation - "Un ensemble de commandes, d'instructions et d'autres syntaxes utilisées pour créer un programme informatique". Move est le langage de programmation de Diem, qui permet à la fois la mise en œuvre des fonctionnalités de base de la crypto-monnaie et la création de futures applications pour Diem.

Blockchain - "Une base de données numérique contenant des informations (telles que des enregistrements de transactions financières) qui peuvent être simultanément exploitées et partagées au sein d'un vaste réseau décentralisé et accessible au public." Les blockchains permettent le fonctionnement des crypto-monnaies, en créant des registres immuables de transactions effectuées. Un consensus doit être trouvé avant que les transactions ne soient inscrites dans le registre.

Le trilemme des blockchains - La difficulté de garantir la sécurité (résistance aux attaques), l'évolutivité (capacité de croître) et la décentralisation (répartition géographique des utilisateurs) des réseaux de blockchains. Jusqu'à présent, les blockchains ont eu du mal à atteindre ces trois objectifs. Bon nombre des meilleures blockchains actuelles, telles qu'Ethereum, sont sécurisées et décentralisées. Cependant, cela entraîne des ralentissements sur le réseau et des frais de transaction élevés, ce qui affecte négativement la scalabilité. Diem, comme la plupart des blockchains, vise à résoudre ce trilemme et Move représente une grande partie de cette solution.

Logique linéaire - Un système de logique mathématique créé par Jean-Yves Girad en 1987 qui met l'accent sur les formules en tant que ressources, plutôt que sur la véracité ou la nécessité de preuves complètes. Le principal changement réside dans le fait que les ressources ne peuvent être utilisées qu'une seule fois, au lieu d'être traitées comme des vérités permanentes (par exemple, "Si tu me donnes A une fois, je te donnerai B une fois", au lieu de "A peut être échangé contre B"). Cela rend la logique linéaire utile en informatique, où restreindre une formule à une ressource au lieu de la traiter comme une vérité universelle permet de créer des systèmes plus polyvalents.

Ressource - L'unité de référence des systèmes de logique linéaire. Dans Move, "une ressource ne peut jamais être copiée ou implicitement supprimée, mais seulement déplacée entre les espaces de stockage du programme". La nécessité pour les tokens, les registres et autres éléments de blockchain d'être permanents est la raison pour laquelle la logique linéaire, en particulier des ressources, a été utilisée pour construire le langage Move.

Qu'est-ce que Move ?

Move est le langage de programmation conçu pour la blockchain Diem. Move et Diem ont tous deux été créés par l'Association Diem, un consortium technologique soutenu par Meta. Récemment, le projet Diem, y compris le langage Move, a été vendu à Silvergate.

Le format d'exécution de Move est un "bytecode de plus haut niveau que le langage assembleur, mais de plus bas niveau qu'un langage source". Le bytecode est examiné on-chain (pour assurer la sécurité des ressources, des types et de la mémoire) par un vérificateur de bytecode, puis exécuté directement par un interpréteur". Ce mécanisme a été pensé pour maximiser la sécurité, sans ajouter le coût de compilation aux transactions, minimisant les frais de gas par rapport à Ethereum. C'est un exemple qui montre que Move tente de résoudre le trilemme par sa conception.

Ressources : Ce qui fait que Move se distingue

Le principal facteur qui sépare Move des autres langages de programmation est son utilisation des ressources, qui est inspirée de la notion mathématique de logique linéaire. Dans la logique linéaire, les formules sont traitées comme des ressources fondamentales qui ne peuvent être utilisées qu'une seule fois. En conséquence, dans Move, "une ressource ne peut jamais être copiée ou implicitement supprimée, mais seulement déplacée entre les espaces de stockage du programme". Move permet aux développeurs d'encoder des types de ressources personnalisées qui sont traitées comme des ressources de "première classe" et ne peuvent être dupliquées ou effacées.

Ceci est rendu possible par le type de structure statique de Move. Cela signifie que les variables dans le langage de programmation doivent être définies comme étant d'un certain type (c'est-à-dire un numérique ou un texte). Dans Move, ce mécanisme protège la nature de première classe des ressources. Cependant, les ressources peuvent toujours être utilisées de la même manière que tout autre élément moins protégé. Elles peuvent être stockées dans des structures de données ou passées comme arguments à des fonctions (fournies comme valeurs à utiliser pour un calcul).

Tout cela signifie que les ressources dans Move bénéficient d'un niveau élevé de sécurité et de souplesse. Elles sont protégées au sein du code, mais peuvent également être utilisées librement pour toutes sortes d'opérations. Cette association est parfaite pour la programmation de la blockchain, car elle répond à la fois aux aspects de sécurité et d'évolutivité du trilemme. Actuellement, la " monnaie Diem, le traitement des transactions et la gestion des validateurs " sont tous codés en tant que ressources avec Move.

Les ambitions de Move

Le whitepaper Move dresse une liste claire des objectifs qui ont motivé la conception du langage. Ils mentionnent également les défis liés à la création de systèmes de blockchain publics, qui concernent non seulement Move, mais aussi tous les langages de programmation de blockchain.

Ils indiquent notamment que Diem (Libra) doit être un "système ouvert", où tout le monde peut voir l'état de la blockchain et soumettre des transactions. C'est radicalement différent des systèmes traditionnels de gestion d'actifs, comme les services bancaires numériques, qui sont des systèmes profondément fermés. L'accès n'est possible qu'avec des autorisations spéciales (dont il existe de nombreux niveaux) et la transparence vis-à-vis des autres utilisateurs est très limitée.

Mais dans une blockchain, "tous les participants sont sur un pied d'égalité". Cela pose un certain nombre de défis. L'un des principaux est d'empêcher la soumission de transactions non valides, par exemple lorsqu'un utilisateur tente de transférer les actifs d'un autre. En outre, les chaînes de blocs doivent prendre en compte deux aspects de la monnaie traditionnelle qui sont difficiles à exécuter dans un programme. Le premier est la rareté ; la duplication des actifs doit être interdite et la création de nouveaux actifs doit être privilégiée. (La mise en œuvre de ce privilège dans un système égalitaire et ouvert est particulièrement difficile, et limite considérablement la décentralisation d'un tel système). Deuxièmement, l'accès : la capacité d'un utilisateur à contrôler et à protéger ses ressources est d'une importance capitale. L'utilisation de ressources protégées de première classe dans Move constitue en grande partie une tentative de résoudre ces problèmes de rareté et d'accès.

L'équipe d'ingénieurs a listé trois problèmes clés qu'elle a trouvés avec les langages blockchain existants, en particulier Bitcoin Script et Ethereum Virtual Machine (respectivement les principaux langages de programmation de Bitcoin et Ethereum).

1. **Encodage indirect des actifs**

La programmation étant exécutée mathématiquement, de nombreux éléments différents sont représentés sous forme d'entiers. Cela inclut les actifs de la blockchain tels qu'une crypto-monnaie, ce qui est le cas pour le Bitcoin et l'Ethereum. Les ingénieurs de Move ont estimé que cela rendait "maladroit et source d'erreurs" l'écriture de programmes basés sur des actifs cryptographiques. C'est pourquoi Move représente le Diem comme une ressource.

2. **Une rareté inextensible**

Les langues de Bitcoin et d'Ethereum font un bon travail pour protéger la rareté de leur principale crypto-monnaie. Cependant, ils ne sont pas bien conçus pour créer de nouveaux actifs aux caractéristiques rares. C'est un problème particulier pour la machine virtuelle Ethereum, qui permet la création de jetons ERC-20 basés sur la blockchain Ethereum. Dans ce cas, les développeurs doivent concevoir la rareté par eux-mêmes, sans l'aide des langages. Move a voulu changer cela.

3. **Un contrôle d'accès inflexible**

Tout comme dans le cas de la rareté, le Bitcoin et l'Ethereum contrôlent rigoureusement l'accès à leurs principales crypto-monnaies, afin de garantir que la propriété et le transfert ne puissent être altérés. Cependant, il n'est pas facile d'étendre ce contrôle à d'autres éléments, et le niveau ou les moyens d'accès ne peuvent pas être personnalisés. Comme pour la rareté, les ingénieurs de Move ont cherché à rendre ces fonctionnalités plus extensibles et personnalisables afin de rendre le langage plus polyvalent et évolutif.

À propos de Pontem

Pontem Network est la principale plateforme expérimentale destinée à Diem, la blockchain sans permission soutenue par Meta et détenue par Silvergate. Notre testnet incentivé, la plateforme de contrats intelligents Move VM et Pontem Blocks, notre outil de développement, permettent tous de commencer à construire pour Diem. Pontem offre une longueur d'avance aux développeurs qui cherchent à concevoir des projets pour le Silvergate Exchange Network et le métavers de Meta.

Pour rester informé de nos travaux, suivez Pontem sur Twitter, abonnez-vous sur Medium et rejoignez-vous nous sur Telegram.

Titre Tag : Comment fonctionne le langage de programmation Move

Meta Description : Tout sur Move, le langage de programmation conçu pour Diem par Meta.