



Rust vs Go : quel langage pour 2025 ?

Par: francoistonic | mer, 18/06/2025 - 07:45

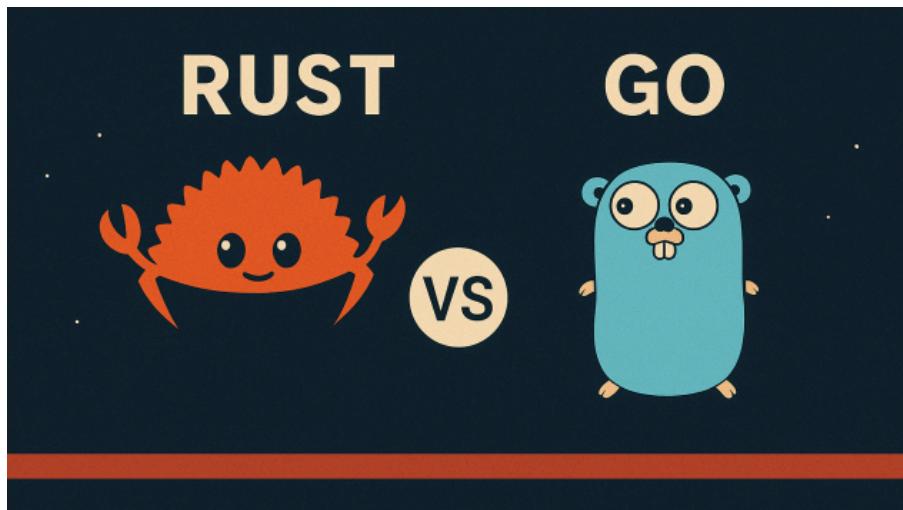
[Rust \(/actualites/rust\)](#) [Go \(/actualites/go\)](#)

Partager 2

Post

Partager

C'est un débat sans fin : quel langage choisir ? Rust ou Go ? Le blog RustRover se propose de nous aider à choisir... ou pas. Ces deux langages connaissent une utilisation en croissance même si nous sommes loin des langages leaders tels que JavaScript, Python, Java, C#, C, C++. Selon State of Developer Ecosystem 2024, Rust est utilisé plus plus de 2,3 millions de développeurs et les migrations de Go vers Rust ne sont pas rares.



La philosophie de Rus est "simple" : la gestion mémoire, une meilleure sécurité, pas de data races, un langage compilé. Rust est poussé par son niveau de sécurité mais il reste un langage difficile à apprendre. Il possède des qualités :

- s'adapte aux IoT
- fonctionne très bien sur WebAssembly
- les grands scaleurs l'utilisent de plus en plus
- langage référence pour créer des CLI
- gestion des ressources et de la mémoire sans garbage collector

Go est aussi un (très)bon langage, plus simple à apprendre. Il aime le cloud, le web, les IoT, les CLI.

Alors, lequel choisir ? Comme souvent, le contexte peut aider. Pour écrire des outils très performances avec un haut niveau de sécurité, Rust est un bon choix. Mozilla l'utilise pour coder le moteur Servo. Go n'est pas mauvais dans la gestion mémoire et les performances dans le web sont très bonnes. Docker l'utilise

comme langage primaire pour les développeurs. Comme Rust, il permet de programmation concurrence.

"Rust et Go ont tous deux des courbes d'apprentissage distinctes et des écosystèmes adaptés à leurs points forts et philosophies respectifs. Le choix doit tenir compte des compétences de chaque membre de l'équipe et des besoins spécifiques du projet." explique le post.

Rust est un langage plus complexe à maîtriser et sa structure peut dérouter. Go apparaît comme plus facile à apprendre avec un syntaxe plus minimaliste. Autre avantage de Go, les goroutines offrent une programmation concurrence plus "naturelle".

"Rust et Go offrent tous deux des outils puissants pour créer des applications concurrentes, mais de manières fondamentalement différentes. Rust offre des garanties de sécurité plus strictes, au prix d'une courbe d'apprentissage plus raide, ce qui le rend idéal pour les applications où la sécurité est essentielle. Go propose une approche plus simple de la concurrence, ce qui le rend plus adapté aux applications nécessitant un développement et un déploiement rapides avec des performances évolutives. Comprendre ces différences devrait vous permettre de choisir l'outil le plus adapté aux exigences spécifiques et à la nature de votre projet." explique le post.

Il s'agit aussi d'une histoire de goût et de pratique. Go est parfois plus riche sur les librairies que Rust. Go aura un usage plus large par sa syntaxe et son apprentissage. Rust connaît une forte croissance dans les développeurs critiques nécessitant sécurité et performances.

Source : <https://blog.jetbrains.com/rust/2025/06/12/rust-vs-go/> (<https://blog.jetbrains.com/rust/2025/06/12/rust-vs-go/>)

Partager 2

Post

Partager

Suivez-nous



Suivre la Page

PROGRAMMEZ !
LE MAGAZINE DES DÉVELOPPEURS

Nefer-IT ©2025. Tous droits réservés
SARL inscrite au RCS de Pontoise, n°797 642 857 siège social : 57, rue de Gisors 95300 Pontoise
Hébergeur : [OVH](#)
Site réalisé par [DzMob](#)