

# LES LANGAGES RÉSOVENT DE PROBLÈMES

## QUELLES SONT VOS PRIORITÉS

Être résilient par compilation : Haskell, OCaml, F#, Scala, Rust ... dans une certaine mesure Kotlin, Swift

Être résilient par parallélisation : Erlang / Elixir

Valider les états (ADT) : Kotlin, Scala, F#, Swift, Rust, OCaml, Haskell

Valider les transitions (GADT) : OCaml, Haskell, Scala

Écrire des DSL génériques (Higher Kinded Types) : Haskell, Scala

Faire de l'embarqué : Go, Rust, OCaml

Faire des jeux vidéos AAA+ : C#, C++ ; AA+ : Lua, Haxe, Javascript, Rust

Faire du calcul parallèle : Rust, C++, Erlang/Elixir

Faire du micro-frontend : F#, C#, Elixir, Typescript

Faire un OS : C, OCaml, Rust

Faire une blockchain : OCaml, Haskell, Rust

Faire de la data science : python, julia, lua, C++, swift

Faire du prototype rapide : javascript, Ruby

« Tournevis multifonction » : F#, Kotlin, OCaml ... Tout langage avec un bon compilateur Javascript et FFI C

MIAGE M2 - QUALITÉ DU SI - THOMAS HAESSLÉ

# TAKE AWAY

## NOUS AVONS VU

Vous devriez maîtriser au moins un langage pour chacun des 6 idiomes principaux

Par composition d'idiome, l'apprentissage de nouveaux langages est facilité

Pas beaucoup d'innovation en 40 ans mais ...

... Les langages s'influencent les uns, les autres, la plupart des langages modernes sont multi-paradigmes

Plus que l'idiome c'est le système de types qui prime pour la qualité, et l'écosystème / les ressources pour la diffusion

Vous devez savoir dire pourquoi vous avez choisi un langage pour un projet !

