

LE MONOLITH

CE FAMEUX CROQUEMITAINE

Désigne un logiciel qui n'est pas microservice

Dans les faits des logiciels « legacy », mal conçus, mal gérés dans le temps et qui provoquent de l'attrition

Bref de MAUVAIS INVESTISSEMENTS ...

... mais on peut aussi faire des erreurs avec des microservices

Getting the wrong idea from that conference talk you attended



Solving Imaginary
Scaling Issues

At Scale

O RLY?

@ThePracticalDev

CONCEPTION DE SERVICES

AVONS NOUS BESOIN DE MICROSERVICE ? DÉCONSTRUIRE LE MYTHE !

Il n'est pas nécessaire de faire des « micro service web » pour :

Agnostique du langage : la plupart des langages permettent soit de l'interopérabilité de bytecode (JVM, .Net), soit de l'interopérabilité de librairie C (.dll, .so, .a), soit de l'interopérabilité par un pivot de langage (JS) ou un assembly (Wasm)

Agilité : 1 service \sim 1 librairie \Rightarrow 1 équipe ; c'est le fonctionnement des communautés open sources

Résilience : Typage statique + fonctions

Extensibilité : L'« horizontal scale » est parfois plus cher que le « vertical scale » sur le cloud et dans certains cas moins performant (puissance de calcul VS parallélisation)

Tout cela a l'avantage de produire des logiciels valides à la compilation et de réduire l'adhérence au réseau !