

# DEVOXX France 2024

## Table des matières

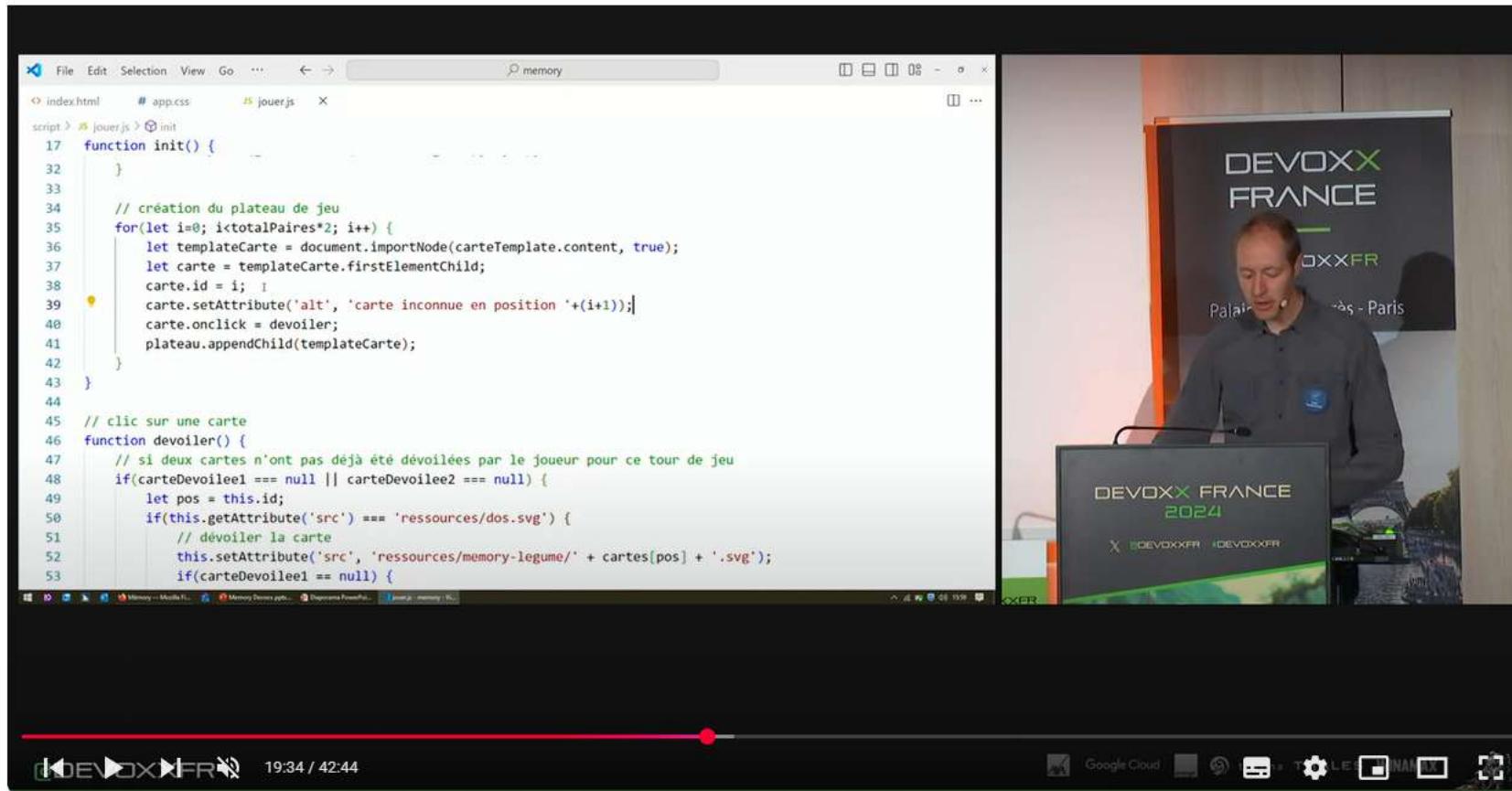
Accessibilité .....	4
Comment rendre une web app accessible à une personne non voyante ? .....	4
Daltonien, mais pourquoi je vois rien ? [*] .....	6
L'IA et le handicap : progrès ou exclusion ? .....	8
Au-delà des heures : La semaine de 4 jours comme levier d'égalité .....	12
Empreinte .....	15
Comment réduire son empreinte carbone grâce à un calendrier de mise à l'échelle automatique ? .....	15
Et si on divisait par 4 l'empreinte carbone du numérique ? [*] .....	19
Green washing ou RSE pragmatique, si on choisissait ? .....	27
Sécurité .....	32
Sécurité du GenAI & des LLM: Une nouvelle ère d'Hacking éthique .....	32
Cybersécurité et cyberdéfense : un sujet géopolitique - Guillaume Poupart .....	34
Quand la cybersécurité n'est pas qu'une affaire de failles .....	35
La fin des mots de passes partagés avec Vault et Boundary .....	38
Tests .....	42
Testcontainers : Simplifiez vos tests d'intégration et vos développements [à tester] .....	42
Je délègue tous mes tests à une IA .....	45

Analyse des erreurs .....	47
Mais non, c'est pas une erreur de CORS .....	47
Conteneur .....	49
De Compose à Kubernetes, Same player shoot again .....	49
Code .....	50
Porter le Flambeau du Clean Code à l'ère du GenAI [livres/articles à lire] .....	50
A Healthy diet for your Java application.....	53
Jeux en ligne, mais comment ça lag pas ? .....	57
L'IA et qualité de code : Construire une synergie avec l'intelligence humaine .....	62
Sauve un dév, écris une doc ! .....	68
Recherche .....	71
IA en médecine : où en sommes-nous ?.....	71
Tout ce que vous n'avez jamais voulu savoir sur les fuseaux horaires .....	75
La recherche à l'ère de l'IA .....	79
Apache Lucene : de l'indexation textuelle à l'intelligence artificielle.....	84
Upgrade.....	89
Bootiful Spring Boot 3.3.....	89
Angular : le renouveau (v17) (Uni) [à reprendre] .....	93
Architecture.....	102
Architecture Hexagonale: Comment venir à bout du code spaghetti ? (Kévin LLOPIS) [pratiquer ?] .....	102

L'Architecture Hexagonale par la pratique, le live coding qui rendra vos applications pl.....	106
L'architecture micro frontend : comment transformer son site web en puzzle géant collaboratif .....	112
Gestion de la mémoire.....	119
Understanding 5 languages memory models (or most of them) .....	119

# Accessibilité

Comment rendre une web app accessible à une personne non voyante ?



The screenshot shows a video player interface. On the left, a code editor displays a portion of a JavaScript file named 'jouer.js'. The code includes functions for initializing a game board and handling card reveals. On the right, a video frame shows a man speaking at a podium. The podium and background banner both feature the text 'DEVOXX FRANCE 2024' and 'Palais des congrès - Paris'. The video player has a progress bar at the bottom, indicating it is 19:34 minutes into a 42:44 minute video. The title 'Comment rendre une web app accessible à une personne non voyante ?' is visible at the bottom of the screen.

```
File Edit Selection View Go ... ↵ → ⌂ memory
index.html # app.css jouer.js X
script > jouer.js > init
17 function init() {
32 }
33
34 // création du plateau de jeu
35 for(let i=0; i<totalPaires*2; i++) {
36     let templateCarte = document.importNode(carteTemplate.content, true);
37     let carte = templateCarte.firstChild;
38     carte.id = i;
39     carte.setAttribute('alt', 'carte inconnue en position +' + (i+1));
40     carte.onclick = devoiler;
41     plateau.appendChild(templateCarte);
42 }
43
44
45 // clic sur une carte
46 function devoiler() {
47     // si deux cartes n'ont pas déjà été dévoilées par le joueur pour ce tour de jeu
48     if(carteDevoilee1 === null || carteDevoilee2 === null) {
49         let pos = this.id;
50         if(this.getAttribute('src') === 'ressources/dos.svg') {
51             // dévoiler la carte
52             this.setAttribute('src', 'ressources/memory-legume/' + cartes[pos] + '.svg');
53             if(carteDevoilee1 === null) {
```

Comment rendre une web app accessible à une personne non voyante ?

## Comment informer l'utilisateur en direct ?

- L'attribut `aria-live` ARIA (Accessible Rich Internet Applications)
  - Valeurs possibles
    - `off`: valeur par défaut
    - `polite`: ajout à liste des éléments à lire
    - `assertive`: lecture prioritaire à faire immédiatement

```
<p id="message"></p>

message.innerText = nbPaires + " sur " + totalPaires + " paires";
```

DEVOXX FRANCE 2024  
Palais des congrès - Paris

DEVOXXFR

24:51 / 42:44

Google Cloud

TOLE

NAN

## Comment rendre une web app accessible à une personne non voyante ?

- Expliciter
- Alt en fonction des évènements
- `aria-live` / `role`

## Daltonien, mais pourquoi je vois rien ? [\*]

DEVOXX FRANCE 2024  
12<sup>e</sup> ÉDITION - DU 17 AU 19 AVRIL 2024

### Approche alternative - ColorADD

15. sopra steria

BLEU JAUNE ROUGE

BLANC NOIR

Bleu Vert Jaune Orange Rouge Violet Marron

Google Cloud Trello Intranet

Daltonien, mais pourquoi je vois rien ?



## Approche alternative - ColorADD



16

sopra steria

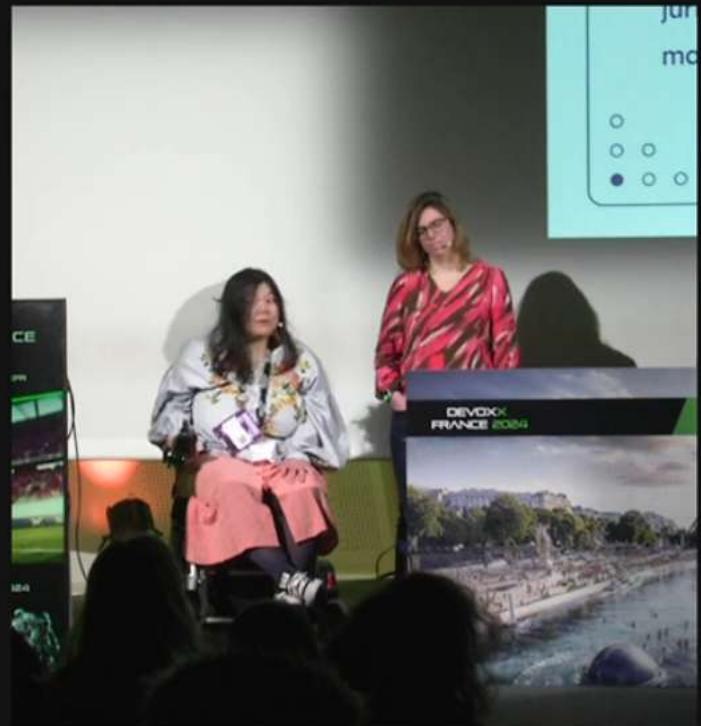
DEVOXXFR 11:42 / 13:10

Google Cloud TOLE INAM

Daltonien, mais pourquoi je vois rien ?

- Contraste
- Forme
- Position

## L'IA et le handicap : progrès ou exclusion ?



## Article 22 de la loi RGPD

"La personne concernée a le droit de ne pas faire l'objet d'une décision fondée exclusivement sur un traitement automatisé [...] produisant des effets juridiques la concernant ou l'affectant de manière significative de façon similaire."



Image générée avec Craiyon



## D'autres innovations qui peuvent être utiles

### [KEIA](#) et [Elio2](#) (IRIS)

Traduisent les contenus automatiquement en LSF.

### [Emoface](#)

Permet de faire reconnaître des émotions.

### [Wiseone](#)

Reformule autrement les textes compliqués.

### [Oticon](#)

Réduit les bruits ambients dans les aides auditives.

### [Glaaster](#)

Transforme les textes pour les personnes dyslexiques.

### [Otter.ai Voice](#)

Prend des notes et rédige des résumés.

### [Speechify](#)

Permet de lire les textes à haute voix.

### [SymboTalk](#)

Permet de communiquer à travers des images et symboles.

### [Sesame Enable](#)

Transforme les smartphones et tablettes en dispositifs mains libres.

### [Waymap](#)

Permet de faciliter le déplacement en donnant des instructions audio détaillées.

### [Mintt](#)

déetecte les chutes en temps-réel et prévient des secours.

### [Rango](#)

Une canne intelligente détectant les obstacles et permettant de se repérer lors des déplacements.

un GPS parlant pour les personnes aveugles et malvoyantes.  
Mintt pour détecter les chutes en temps réel et prévenir



- Problème des biais : visible/invisible, représentation limitée par les médias et la connaissance commune, incompréhension/cliché, non connaissance
- Problème de sous-représentation, qui tend à disparaître par simplification/ « optimisation »
- L'IA va **amplifier les biais** déjà existants : problème majeur de l'IA actuelle, qui amplifie les « trends » / « mainstream » en diminuant la représentation
  - o Particulièrement problématique pour la créativité, la recherche, l'éthique, la santé et la sécurité !!
  - o D'où nécessité de ré-entraînement constant ET contrôlé
- Retranscription audio : KO avec les accents par exemple, et notamment l'accent de personnes malentendantes de naissance
  - o Malgré le gain qu'apporte l'IA, n'étant pas exacte, oblige un effort supplémentaire d'analyse (évident pour les personnes sensibilisées : chercheurs, annalistes, avec handicap, etc, donc plus fatigant mentalement ; moins évident pour d'autres : non informés, « naïfs », etc. qui vont « oublier » d'avoir cette réflexion, donc moins fatigant mais non efficace non plus car risque d'erreur)
- Amélioration/aide qu'apporte l'IA : l'automatisation, la traduction automatique, l'assistance, etc. ; énormément de possibilité, et d'avantage, dans la mesure où on entraîne bien, et contrôle bien ; car l'impact en cas d'erreur peut être sur la santé ou la sécurité, mais aussi sur les moyens personnels (exemple aide lors de la consommation ou investissement)
- Idée vs réalité : contexte, application, diffère de la théorie dans sa tête (comme la reconnaissance du langage des signes, la reconnaissance de signaux sonores qui diffèrent entre les lieux/cultures)
- IA pour mettre les surcouches pour l'accessibilité : actuellement pas prêt, pas recommandé du tout, l'IA n'a ni le contexte ni même la connaissance du aux biais et à la méconnaissance générale, enfin surcouches non maîtrisées augmente très fortement la maintenance (par les devs)

Au-delà des heures : La semaine de 4 jours comme levier d'égalité



Comment la semaine de 4 jours peut venir adresser une grande partie de cette roue ?

- Temps projet personnel (qui contribue à la société)
- Soulagement en cas d'handicap et affilié (ex. maladie chronique)
- Semaine de 4 jours (non 4/5) avec salaire complet et pouvant être flexible sur le 3 jour off de semaine
- Réflexion dans de nombreux pays
- Gain charge mentale (baisse de stress observé), meilleure hygiène de vie amélioré (baisse des arrêt maladie observé)
- Lisser les activités « administratives » et « logistiques » (courses, etc)
- Maladie chronique : 37% : meilleure prise en charge avec le 3e
- Aidants
- Se former, reprendre ses études
- Plus de temps pour s'inclure
- Etude de l'impact environnemental/social/associatif
- « IT Woman by Yeeso »
- Améliore l'employabilité (?) et rentabilité (+1,4%) de l'entreprise
- <https://s4j.fr/> : liste entreprise semaine 4 jrs (attention non sécurisé)
- Mise en place en entreprise ? nécessite une réflexion de chaque acteur de l'entreprise pour prendre les contraintes et améliorations

## Empreinte

Comment réduire son empreinte carbone grâce à un calendrier de mise à l'échelle automatique ?



## Changement climatique – dans l'IT

1-2 % de l'énergie mondiale est consommée par les data centers  
8 % en 2030 ?

0.5 % de l'énergie mondiale est consommée par le minage de Bitcoin

70 % de l'énergie provient de l'énergie fossile

12 % de l'énergie mondiale consommée provient de l'IT

3-4 % de l'empreinte carbone mondiale provient de l'IT

@DEVOXXFR

Google Cloud takima THALES WINAMAX

Changement climatique

Wikipedia • La variabilité du climat correspond à tous les changements du système climatique qui persistent plus longtemps qu'un événement météorologique.

⋮

Comment réduire son empreinte carbone grâce à un calendrier de mise à l'échelle automatique ?



## GreenIT Modèle de référence

<b>Responsabilité de l'entreprise</b>	Stratégie informatique durable	Gouvernance IT	Gestion des fournisseurs et achats informatiques
	Architecture d'entreprise	Gestion du portefeuille applicatif	Information & cybersécurité
	Conception d'application et efficacité du code	Data: Conception, Utilisation et Stockage	Cycle de vie du matériel Informatique
	Gestion du cloud	Déploiement de la plateforme	Utilisation & Scaling
<b>Responsabilité du Data Center ou du Cloud provider</b>	Serveurs	Stockage	Base de données
	Energie	Eau	Gestion circulaire des déchets
	Installation Tech	Connectivité	Bâtiment
		Terrain	
<b>Responsabilité du fournisseur de service</b>	Réseau électrique		
	Sources d'énergie renouvelables	Sources d'énergie de transition	Sources d'énergie de combustibles fossiles
			Refroidissement et perte de chaleur

Betclic  
GROUP

DEVOXXFR 9:54 / 46:12



### ☰ Changement climatique ☰

Wikipedia • La variabilité du climat correspond à tous les changements du système climatique qui persistent plus longtemps qu'un événement météorologique.

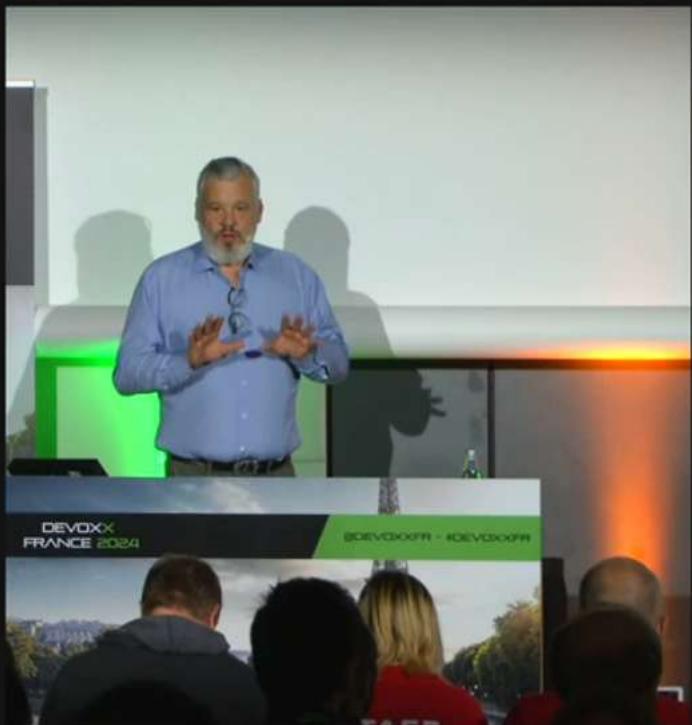
⋮

Comment réduire son empreinte carbone grâce à un calendrier de mise à l'échelle automatique ?

Aide au calcul de l'empreinte carbone : <https://dataviz.boavizta.org/>

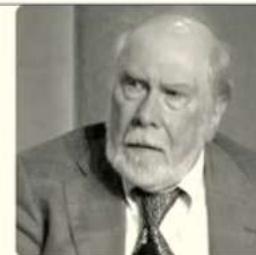
- Préférence pour le serverless & FaaS (déport de la responsabilité, diminution de la consommation passive/dormante)
- Code performant et propre
- Scaling (CPU, Mem, ...) en fonction de l'activité (exemple pic de connexion, pic de transaction)
- Eteindre une partie des machines lorsqu'elles ne sont pas sollicitées (environnement de Staging par exemple à 80-90% hors horaires ouvrées)

Et si on divisait par 4 l'empreinte carbone du numérique ? [\*]

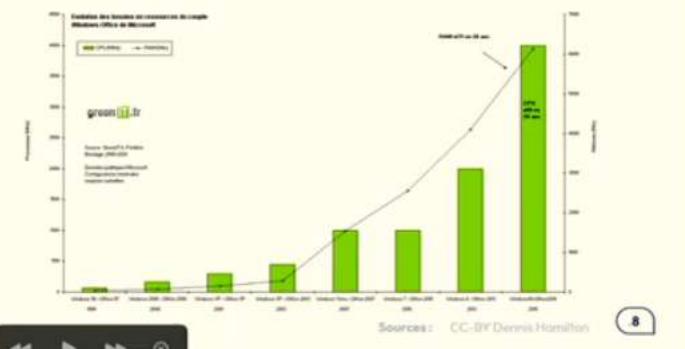


## Quelques exemples de gâchis numérique ces 50 dernières années

**Loi de Wirth**  
**(Presque)**  
Ce qu'Intel vous donne,  
Microsoft vous le retire

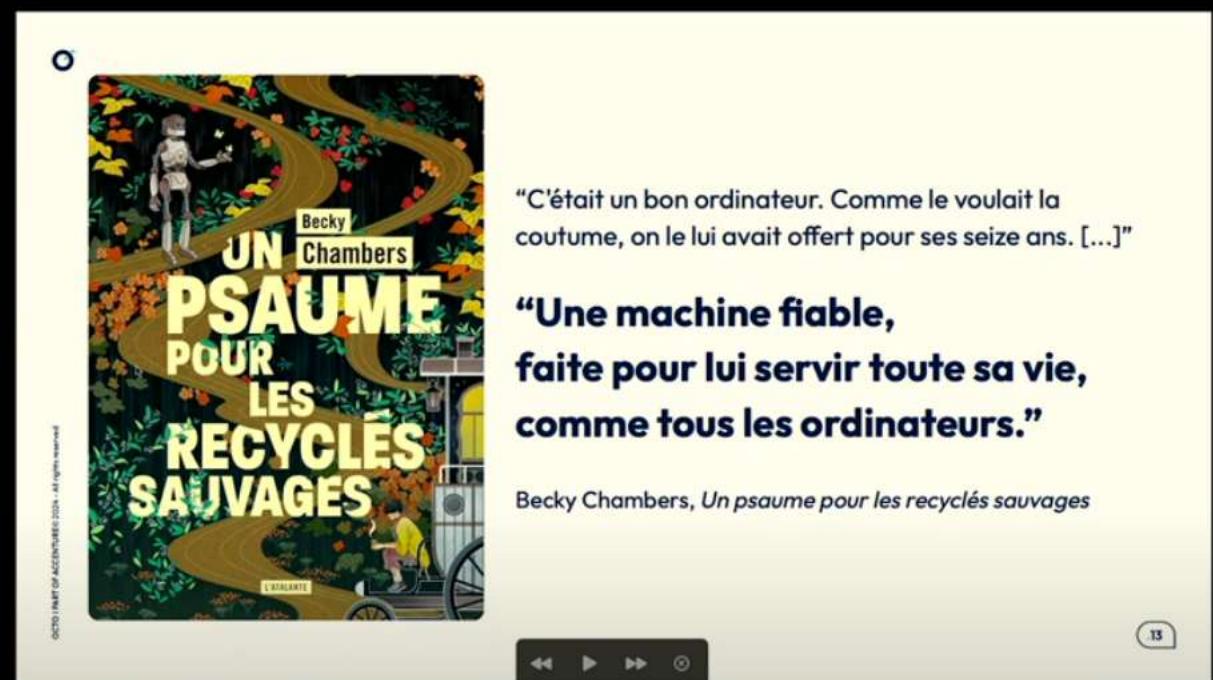


- La taille moyenne d'une page Web a été multipliée par 150 en 25 ans.
- Le temps pour afficher un caractère sur l'écran a été multiplié par 5 entre un Apple II de 1983 et un Lenovo X1 plus jeune de 33 ans.
- Le couple Office + Windows a vu sa configuration minimale de mémoire vive multipliée par 171 en deux décennies.



## Changement climatique

Wikipedia • La variabilité du climat correspond à tous les changements du système climatique qui persistent plus longtemps qu'un évènement météorologique.



17:37 / 45:56



## ☰ Changement climatique ☰



Wikipedia • La variabilité du climat correspond à tous les changements du système climatique qui persistent plus longtemps qu'un événement météorologique.



## Scénarios de réchauffement climatique

Source : <https://ourworldindata.org/future-emissions>

Annual global greenhouse gas emissions  
in gigatonnes of carbon dioxide equivalents

150 Gt

100 Gt

50 Gt

Greenhouse gas emissions  
up to the present

1990 2000 2010 2020 2030 2040 2050 2060 2070 2080 2090 2100

0

No climate policies  
4.1 - 4.8 °C  
→ expected emissions in a baseline scenario if countries had not implemented climate reduction policies.

Current policies  
2.5 - 2.9 °C  
→ emissions with current climate policies in place result in warming of 2.5 to 2.9°C by 2100.

Pledges & targets (2.1 °C)  
→ emissions if all countries delivered on reduction pledges result in warming of 2.1°C by 2100.

2°C pathways

1.5°C pathways

Nous sommes ici

Il faudrait être là

17

@DEVOXXFR



22:39 / 45:56



Changement climatique

⋮

Wikipedia • La variabilité du climat correspond à tous les changements du système climatique qui persistent plus longtemps qu'un évènement météorologique.



O

## Loi d'erooM

Effort Radicalement Organisé  
d'Optimisation en Masse

En optimisant le logiciel d'un facteur 2 tous les deux ans, on libère de la puissance informatique avec laquelle on peut inventer de nouveaux usages.

C'est comme la loi de Moore, mais **sans changer le matériel !**

26



32:16 / 45:56



## EI Changement climatique ☰



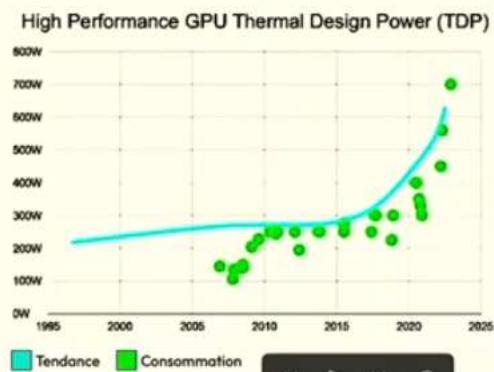
Wikipedia • La variabilité du climat correspond à tous les changements du système climatique qui persistent plus longtemps qu'un évènement météorologique.



### La consommation électrique des GPU augmente



La consommation électrique de chaque transistor, après des années de baisse bienvenue, s'est arrêtée. La densité continue d'augmenter, la consommation de chaque puce s'envole.

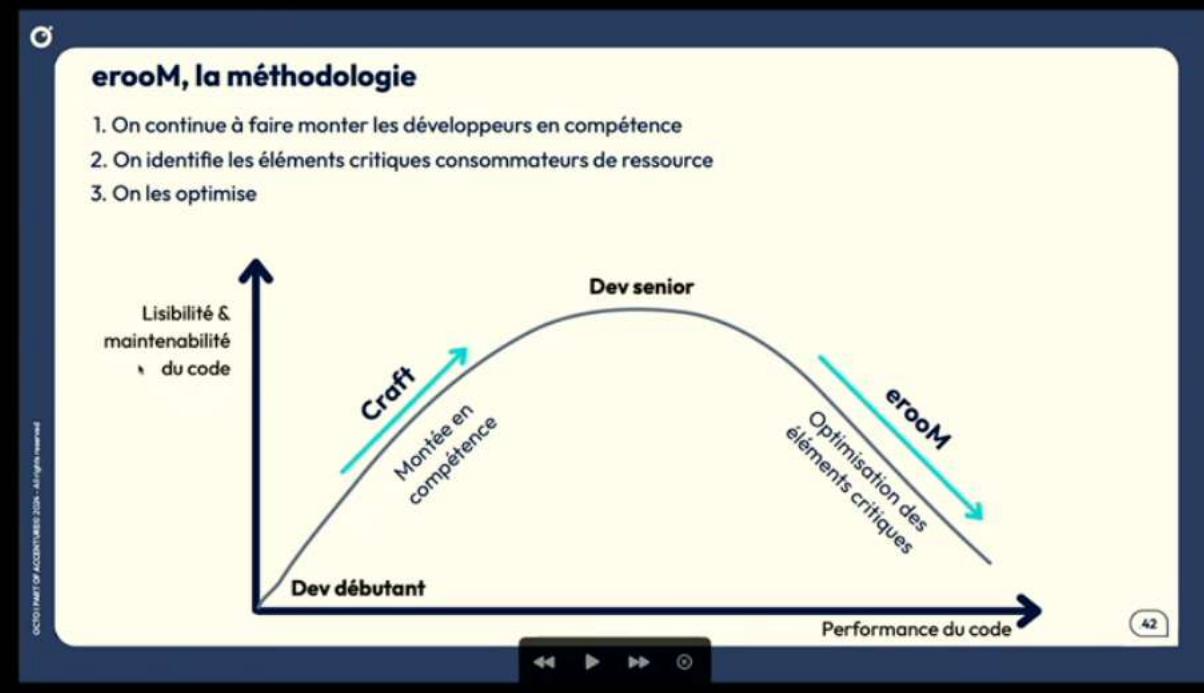


38



### ☰ Changement climatique ☰

Wikipedia • La variabilité du climat correspond à tous les changements du système climatique qui persistent plus longtemps qu'un évènement météorologique.



## EI Changement climatique ☰

Wikipedia • La variabilité du climat correspond à tous les changements du système climatique qui persistent plus longtemps qu'un événement météorologique.



- Loi de Moore (transistor qui double chaque année, volonté d'Intel de faire en sorte que, et ça a marché)  $\Leftrightarrow$  loi de Wirth
- On est poussé à **ajouter des fonctionnalités plutôt qu'optimiser l'existant** (donc + gros et – optimisé)
  - o On est non seulement poussé à ajouter de la fonctionnalité au détriment de l'optimisation, mais en plus, +ou- due à la loi de Moore, de toute façon la techno sera suffisante pour « compenser » la non-optimisation
- Le matériel ne ralentit pas le logiciel, c'est l'ajout de fonctionnalités qui ralentissent dû à la lourdeur
- S'il n'y avait plus de loi de Moore ? => « un psaume pour les recyclés sauvages »
- Optimisation pour accélérer les process, en ayant des processus plus rapide on consomme aussi moins
- <https://www.youtube.com/watch?v=c33AZBnRHks> “Someone improved my code by 40,832,277,770%”
  - o CodeS sur GitHub
- **ET SI on décidait en moyenne d'optimiser d'un facteur 2 tous les 2 ans ?!**
- La loi de Moore stagne depuis 2015, mais les GPU grattent la puissance et donc la consommation par transistor augmente énormément depuis
- Faire du Craft
  - o Attention à trop optimiser, on peut ne plus rien comprendre ou trop complexifier

## Green washing ou RSE pragmatique, si on choisissait ?

DEVOXX FRANCE 2024  
12<sup>e</sup> ÉDITION - DU 17 AU 19 AVRIL 2024

DEVOXX FRANCE  
#DEVOXXFR  
Palais des Congrès - Paris

DEVOXX FRANCE 2024  
X DEVOXXFR X DEVOXXFR

DEVOXX FRANCE 2024  
EDITION AVRIL 2024

“

L'intégration volontaire (*ou pas*) par les entreprises de préoccupations sociales et environnementales dans leurs activités.

”

•

@DEVOXXFR

Google Cloud takima THALES WINAMAX

Green washing ou RSE pragmatique, si on choisissait ?



Extraits choisis de lois en France et en Europe

- 1978 - Le code de l'énergie
- 1987 - Développement durable est inventé !
- 1992 - Sommet de la Terre à Rio
- 2001 - NRE
- 2009 - Grenelle I
- 2010 - Grenelle II
- 2014 - Directive européenne sur le Reporting Non Financier
- 2015 - Loi relative à la transition énergétique pour la croissance verte
- 2017 - Devoir de vigilance
- 2019 - Loi Pacte, impacts sur le code civil et création du statut d'entreprise à mission
- 2020 - Loi anti gaspillage pour une économie circulaire - AGEC
- 2021 - Loi Climat et Résilience
- 2022 - Directive européenne "Women on board"
- 2022 - Directive européenne Corporate Sustainability Reporting Directive (CSRD)

10



Extraits choisis de lois en France et en Europe  
**Des grandes étapes**

- NRE 2003 **Trouver les réflexes** (sur les grandes entreprises)  
créer des habitudes et un savoir-faire
- Grenelle II 2012 **Généraliser et préciser**,  
professionnaliser l'approche, rendre concret
- Devoir de Vigilance 2017 **Créer un ruissellement** (sur toute la chaîne de valeur),  
étendre la responsabilité aux fournisseurs
- Directive EU CSRD 2024 **Étendre à l'Europe**,  
généraliser et contraindre tous les acteurs

13



Agir

### Notre pratique Tech a aussi des impacts...

"Le poids d'une page web a été multiplié par **155** en 27 ans"

"**70%** des fonctionnalités demandées par les utilisateurs ne sont pas essentielles,  
**45%** ne sont jamais utilisées."

"**30%** : la part mondiale des serveurs inutilisés qui consomment de l'électricité en pure perte"

### L'empreinte carbone de nos activités numériques

Estimation des émissions de CO<sub>2</sub> liées à l'utilisation des technologies numériques par personne et par an \*



\* Calculs réalisés en Allemagne en avril 2020.  
Source : Oko-Institut

### Green washing ou RSE pragmatique, si on choisissait ?

- Responsabilité Sociétale des Entreprises
- Petites actions ET gros impacts
- Prendre en compte toutes les possibilités même l'impossible
- La pratique tech

- Nettoyer : nouvelle fonctionnalité utile ? économique « by design » ; code/fonctionnalité mort ; etc.
- Investir les gens/équipes / force de proposition par tous

# Sécurité

## Sécurité du GenAI & des LLM: Une nouvelle ère d'Hacking éthique

The slide is titled "Les Attaques" and features a circular diagram divided into segments representing different types of attacks. The segments are: Malware injection (blue), Prompt injection (GenAI) (dark blue), Model theft (pink), Data theft (light pink), Training data extraction (magenta), Poisoning (cyan), Backdoor (light cyan), and Evasion (lightest cyan). Each segment contains a small icon related to the attack type. The background of the slide is white with three decorative 3D soccer balls (blue, white, and pink) positioned around the circle. The THALES logo is at the bottom left, and a copyright notice is at the bottom right.

DEVOXX FRANCE 2024  
12<sup>e</sup> ÉDITION - DU 17 AU 19 AVRIL 2024

Les Attaques

- Malware injection
- Prompt injection (LLM, GenAI)
- Model theft
- Data theft
- Training data extraction
- Evasion
- Backdoor
- Poisoning

THALES

REF 00000000000000000000000000000000 - Date: 00/00/0000 / Version: 00211168-DOC-GRP-EN-004  
This document may not be reproduced, modified, adapted, published, translated, or any way, in whole or in part or licensed to a third party without the prior written consent of THALES. © 2023 THALES. All rights reserved.  
[THALES GROUP LIMITED DISTRIBUTION]

#DEVOXXFR

Paris Congrès - Paris

DEVOXX FR 2024

8:38 / 43:27

Google Cloud

TOLE

INANIX

Sécurité du GenAI & des LLM: Une nouvelle ère d'Hacking éthique

La Défense

Code audit

Input Transform. (LLM, GenAI)

Model watermarking

Data watermarking

Crypto.

Federated Learning

Adversarial training

Backdoor detection

Data sanitization

Code audit

THALES

#DEVOXXFR

Paris Congrès - Paris

DEVOXX 20

DEVOXXFR - #DEVOXXFR

## Sécurité du GenAI & des LLM: Une nouvelle ère d'Hacking éthique

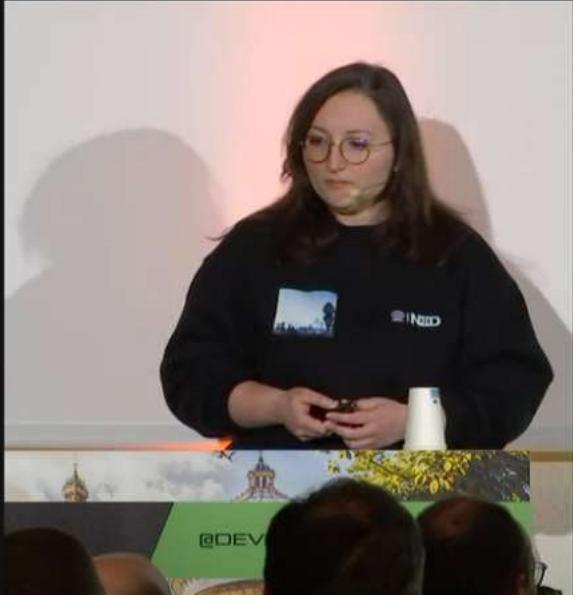
- IA n'est pas encore réglementé : opportunités et risques
- Possibilités d'attaque d'IA nombreuses en trompant l'IA : notamment spécifique à l'IA tel que le training ou l'injection dans le prompt

- Exemple attaque d'injection dans le prompt : l'altération des métadonnées d'une image qui peut être utilisée pour une présentation/information
- Nécessité de limiter la permissivité des outils IA pour diminuer la surface d'attaque
- Plusieurs axes de défense tel que l'entraînement à partir de données saines

## Cybersécurité et cyberdéfense : un sujet géopolitique - Guillaume Poupart

- Cybercriminalité : opportunités partout et facilement, notamment les zones de non droits (où il n'y a pas ou plus d'autorité auquel se fier pour combattre la cybercriminalité), facile d'escroquer
- Attaques les cibles faciles : mal ou peu renseignés face aux attaques, facile à berner/accuser comme les personnes a-tech ou les petites entreprises,
- Espionnage boosté par le numérique, victimes prennent beaucoup de temps à réaliser, et encore plus à être prouver et encore plus à attribuer, en plus les victimes n'en parlent pas (mauvaises images)
- Paralyser des systèmes à cause de la cybercriminalité : numériques, physiques, économiques, démocratiques...
- Etablissement de scénarios, ça ne finit jamais, on finit par arrêter la liste, « le livre blanc de la défense et de la sécurité nationale » (2008) en France = un problème : une loi (ça marche assez bien...)
- Utilisation pour mettre des coups de pressions à très grande échelle, parfois à partir de rien ou presque (exemple de l'attaque de TV5monde)
- Coopération pour trouver les patients 0 pour mieux analyser les attaques
- Tension permanente amplifiée en conflit géopolitique
- La sécurité dans le code première ligne de défense, essentielle et parfois presque suffisante

## Quand la cybersécurité n'est pas qu'une affaire de failles



DEVOXX FRANCE 2024  
12<sup>e</sup> ÉDITION - DU 17 AU 19 AVRIL 2024

# Usage **catachrétique**

Détourner de sa destination initiale un outil ou un instrument affecté à une tâche ou à l'action sur un objet de travail

DEVOXXFR 235 / 40:50

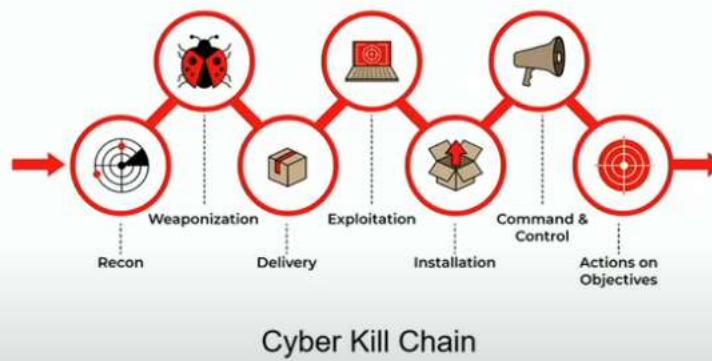
Google Cloud  LES  NAIX  

Quand la cybersécurité n'est pas qu'une affaire de failles



## Conclusion

C'est un peu tiré par les cheveux ton truc non ?



Cyber Kill Chain

### Quand la cybersécurité n'est pas qu'une affaire de failles

- Usage catachrétique (exemple hack meuble ikea)
- CTF (capture de flag) challenge pour récupérer le flag en exploitant faille ou mauvaise configuration, mais pas que, en détournant le fonctionnement, exemple

- DNS (domain name system), ex DNS tunneling (interroger “innocemment” le DNS pour faire connaitre à l’extérieur qu’un system/process est activé)
- Texte et image, ex Stéganographie (dissimulation de secret dans des images ou textes, via l’encodage des pixels ou lettres qu’on va transformer)
- Cryptographie, ex (en fait lié à une faille de sécurité mais hardware plutôt que software ?) Side-channel Attacks en analysant l’implémentation du chiffrement (courbe de consommation CPU, d’où faille hardare) pour « reconstruire/déduire » la clé

La fin des mots de passes partagés avec Vault et Boundary



PROBLÉMATIQUE

**Secret admin partagé**

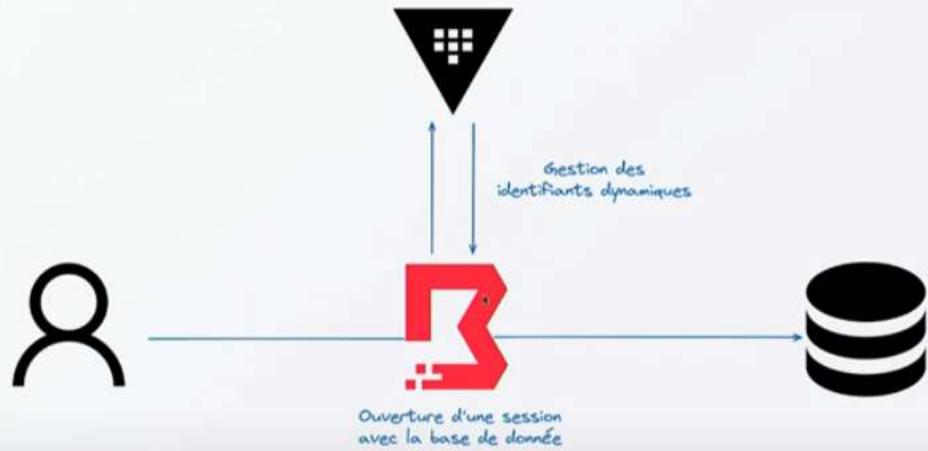
The image is a cartoon from The Simpsons. At the top, it says "STRONG PASSWORD" in large white letters. Below that is a large, multi-tiered wooden castle with a drawbridge. In the foreground, there's a smaller wooden fence. Two characters, Bart Simpson and his sister Lisa, are standing in front of the fence. Bart is on the left, wearing an orange shirt and blue shorts, and Lisa is on the right, wearing a yellow dress. Below them, the text "EVERYONE SHARING IT" is written in large white letters. The background shows a green landscape with trees and a path.

- Risque de fuite du secret
- Tout le monde peut effectuer des actions d'administration sur la base de données
- Il est impossible de savoir qui requête la base de données

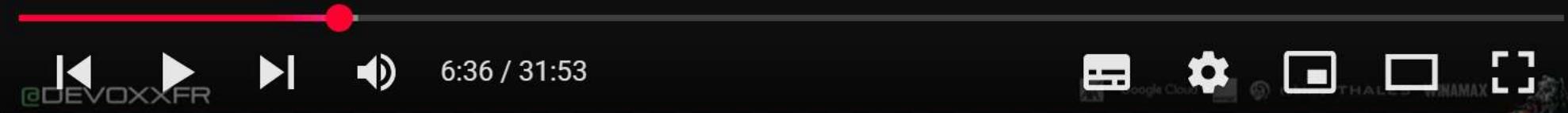


PROBLÉMATIQUE

**Vault et Boundary**



- Vault pour utiliser des secrets dynamiques
- Boundary pour améliorer l'expérience de DevX



**La fin des mots de passe partagés avec Vault et Boundary**

- Dédié à la connexion manuelle, pas de chaîne de connexion (qu'on utiliserait pendant la dev par exemple)

-

# Tests

Testcontainers : Simplifiez vos tests d'intégration et vos développements [à tester]

DEVOXX FRANCE 2024  
12<sup>e</sup> ÉDITION - DU 17 AU 19 AVRIL 2024

FRANCE VOXXFR

Congrès - Paris

Géré par le framework

```
@TestConfiguration(proxyBeanMethods = false)
public class TestContainerConfig {

    @Bean
    @ServiceConnection
    OracleContainer oracleContainer(){
        return new OracleContainer("gvenzl/oracle-xe:21-slim-faststart")
            .withDatabaseName("testDB")
            .withUsername("testUser")
            .withPassword("testPassword")
            .withInitScript("init_schema.sql");
    }
}
```

```
@SpringBootTest
@Import(TestContainerConfig.class)
@Transactional
class MyIntegrationTest {
```

@DEVOXXFR

Google Cloud takima THALES WINAMAX

Testcontainers : Simplifiez vos tests d'intégration et vos développements



@DEVOXXFR

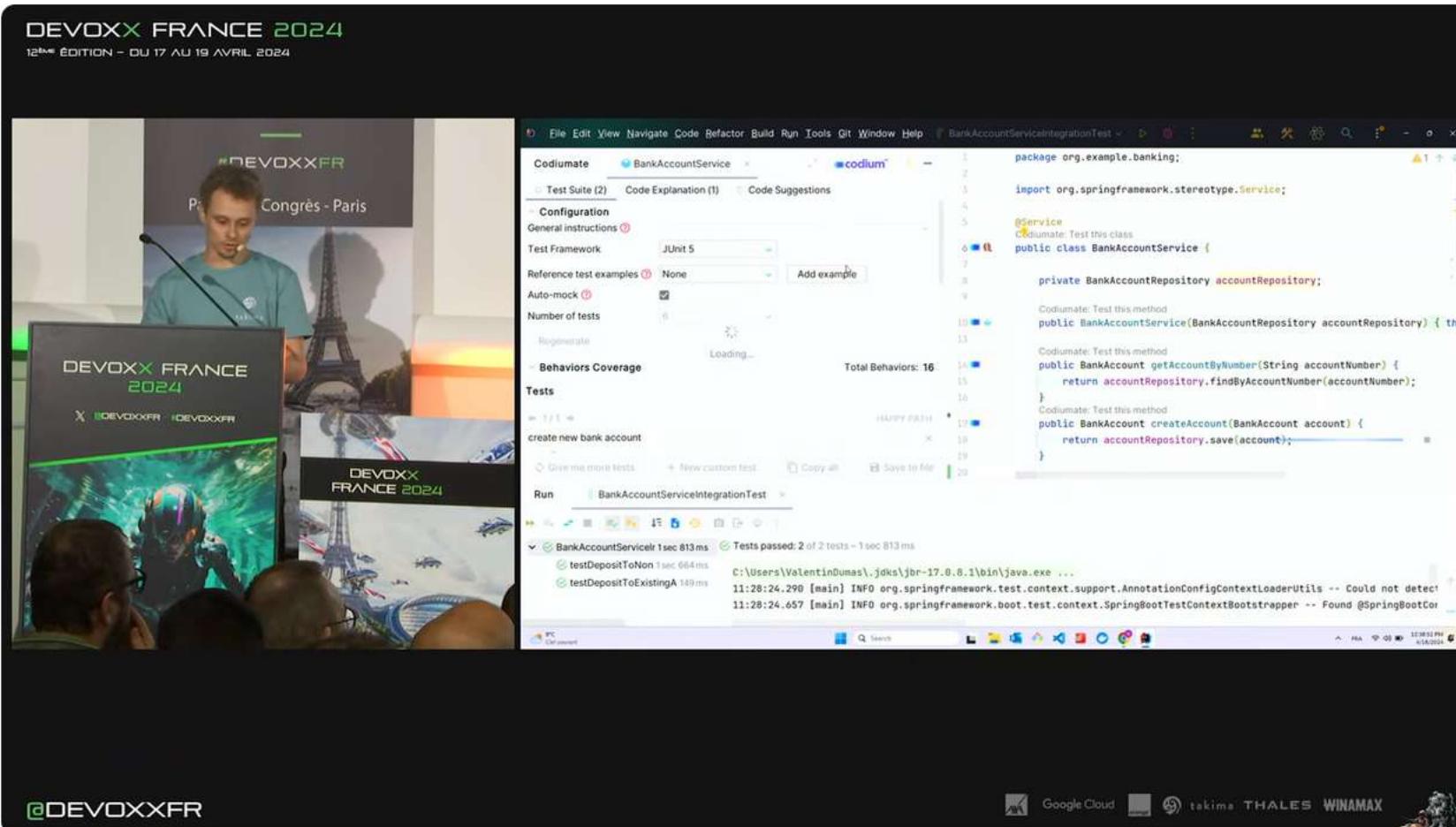
Google Cloud takima THALES WINAMAX

## Testcontainers : Simplifiez vos tests d'intégration et vos développements

- Tests d'intégration
- Isoler (impossible avec des ressources partagées)
- Rejouer
- En fonction de l'environnement (limite des BDD embarquées telle que H2)

- Intégré à Spring (Spring Boot)
- @ServiceConnection (accès aux informations de connexion du testContainer)
- JUnit 5 (@TestContainer, @Container)
- Possibilité de provisionner un container spécifique pour un test/une démo
- ! Fortement couplé à Docker : nécessité d'avoir un Daemon Docker

## Je délègue tous mes tests à une IA



Je délègue tous mes tests à une IA



Live Démo  
**Génération TI sur contrôleur**

“ Generate integration tests for endpoints using rest template.

*Write whole class* ”

```
@SpringBootTest(webEnvironment = SpringBootTest.WebEnvironment.RANDOM_PORT)
Codumate Add more tests
public class BankAccountControllerIT {
    @LocalServerPort
    private int port;
    @Autowired
    private TestRestTemplate restTemplate;
    @Autowired
    private JdbcTemplate jdbcTemplate;
    private String baseUrl;

    @BeforeEach
    public void setUp() {
        this.baseUrl = "http://localhost:" + port + "/accounts";
        jdbcTemplate.execute(sql: "TRUNCATE TABLE accounts");
    }
}
```

14

DEVOXXFR 11:03 / 14:48

Google Cloud TOLE INDEX

Je délègue tous mes tests à une IA

- Codium AI (prompt, analyse du code et du contexte => génération de tests basiques, formate un prompt pour chatBot IA : GPT4)
- Contexte du projet
- Tests unitaires, Tests d'intégration
- Extension IntelliJ : « Codumate »

# Analyse des erreurs

Mais non, c'est pas une erreur de CORS

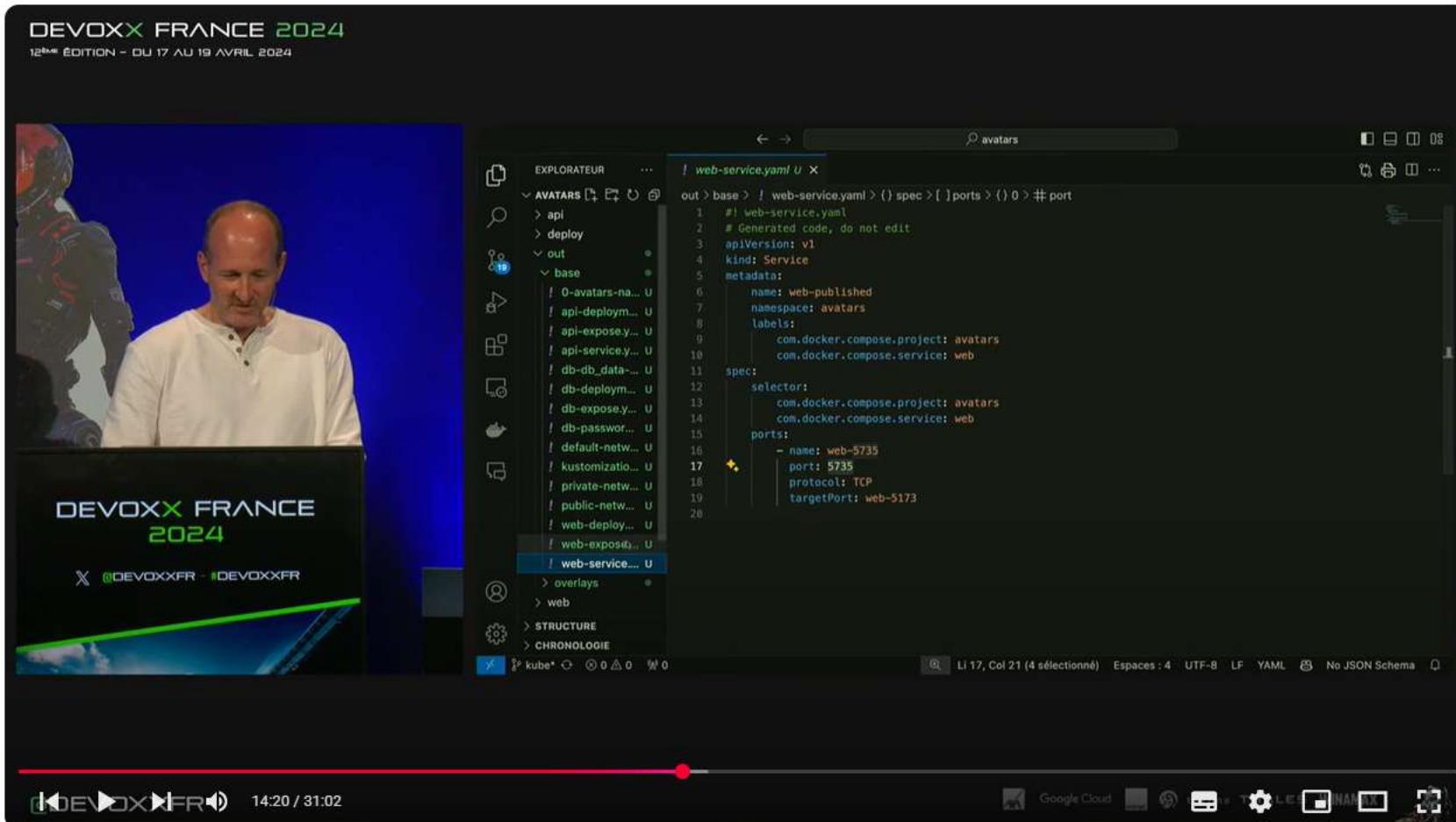
The image consists of two main parts. On the left is a screenshot of a presentation slide titled "EN CONCLUSION". The slide contains a bulleted list of five items: "C'est au service appelé de configurer les en-têtes CORS", "Attention aux intermédiaires", "Regardez le retour effectif de la requête", "Pensez à tous les endroits d'où sont appelés vos services", and "Consultez la doc de votre framework". At the bottom left of the slide is the "DEVOXX France" logo. On the right is a photograph of a man with glasses and a beard, wearing a black t-shirt, standing behind a podium and speaking to an audience. The podium also has the "DEVOXX FRANCE 2024" logo. The background shows a stage set with a large screen displaying the conference's branding.

**Mais non, c'est pas une erreur de CORS**

- Access-Control Headers
- curl ne renvoie pas les CORS, un front peut avoir un pb CORS même si curl fonctionne

# Conteneur

## De Compose à Kubernetes, Same player shoot again



De Compose à Kubernetes, Same player shoot again

# Code

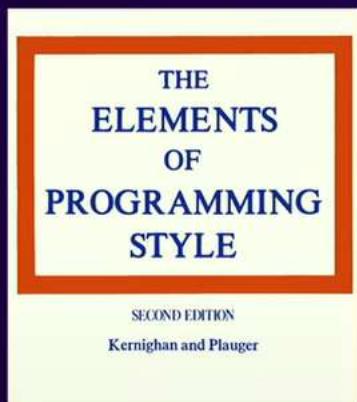
[Porter le Flambeau du Clean Code à l'ère du GenAI \[livres/articles à lire\]](#)

The collage includes the following elements:

- Top Left:** DEVOXX FRANCE 2024 logo with "12<sup>e</sup> ÉDITION - DU 17 AU 19 AVRIL 2024".
- Top Center:** A timeline graphic showing four book covers with their publication years: 1996, 1999, 2002, and 2006. The books are:
  - 1996:** "SMALLTALK BEST PRACTICE PATTERNS" by Kent Beck.
  - 1999:** "extreme Programming explained: EMBRACE CHANGE" by Kent Beck.
  - 2002:** "REFACTORING: IMPROVING THE DESIGN OF EXISTING CODE" by Martin Fowler.
  - 2006:** "TEST-DRIVEN DEVELOPMENT BY EXAMPLE" and "IMPLEMENTATION PATTERNS" both by Kent Beck.
- Top Right:** A photograph of a speaker at a podium during a presentation. The podium has "DEVOXX FRANCE 2024" and "#DEVOXXFR" branding. The background shows a banner for "Palais des congrès - Paris".
- Bottom Left:** The Twitter handle "@DEVOXXFR".
- Bottom Right:** Logos for Google Cloud, takima, THALES, WINAMAX, and a small robot icon.

[Porter le Flambeau du Clean Code à l'ère du GenAI](#)

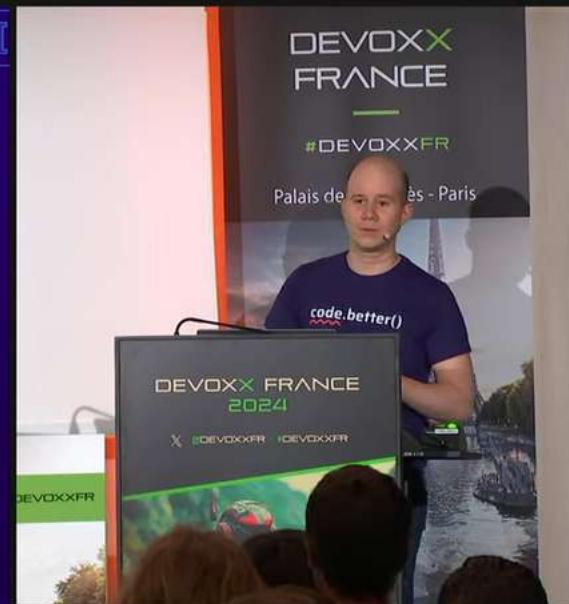
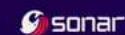
## 1974 : Kernighan & Plauger



- Formatage, noms de variables, control flow, gestion d'erreurs, etc, etc
- Exemples de code conforme et non-conforme
- Phrases:
  - "Each module should do one thing well"
  - "Make it right before you make it faster"
  - Etc...

The Elements of Programming Style, Brian W. Kernighan and P. J. Plauger, 1974. McGraw-Hill, Inc.

©2024, SonarSource S.A., Switzerland.



@DEVOXXFR

Porter le Flambeau du Clean Code à l'ère du GenAI



Google Cloud



takima



THALES



WINAMAX



## Clean Code Taxonomy



DEVOXXFR 28:20 / 45:08

Google Cloud DEVOXXFR LE NAMIX

### Porter le Flambeau du Clean Code à l'ère du GenAI

- Kent Beck “Xtreme Programming Xplained” & “Refactoring”
- Kernighan & Plauger “The Element of Programming Style”
- <https://www.sonarsource.com/blog/what-is-clean-code/>

## A Healthy diet for your Java application

The image is a composite of two screenshots. On the left, a woman is speaking at a podium during the Devoxx France 2024 conference in Paris. She is wearing a dark t-shirt with the text "code better". Behind her is a large screen displaying the "#DEVOXXFR" logo and the text "Paris Congrès - Paris". On the right, an IntelliJ IDEA interface shows a Java code editor with the following code:

```
import ...  
...  
@RestController @author Margarita Nedzelska  
public class ExampleController3 {  
    private List<BigObject.SmallObject> smallObjects = new ArrayList<>(); 1 usage  
    ...  
    @GetMapping("/example3") @author Margarita Nedzelska  
    String hello() {  
        BigObject.SmallObject smallObject = new BigObject().new SmallObject();  
        smallObjects.add(smallObject);  
        ...  
        return smallObject.hello();  
    }  
    ...  
}
```

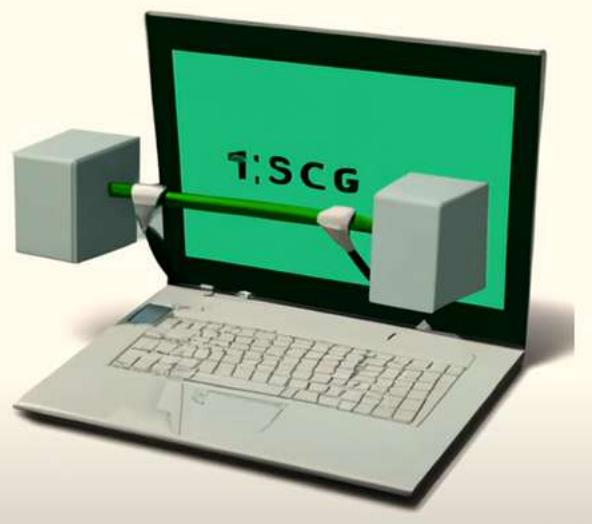
The IntelliJ interface also includes a Problems tab showing several warnings related to collection usage and field visibility.

A Healthy diet for your Java application



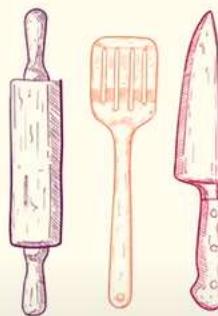
### How to make your app “fit”?

- Is there any problem?
- Let's measure
- Let's analyze
- Find the root cause
- Fix it
- Improve
- Prevent





## A healthy diet for your Java application



- Avoid statics and mutability
- Clean ThreadLocals
- Free resources
- Avoid Non-static inner classes
- If still need them, be careful
- Reduce the scope of references
- Monitor memory usage

@DEVOXXFR 39:33 / 45:14

Google Cloud TALENTINAMAX

### A Healthy diet for your Java application

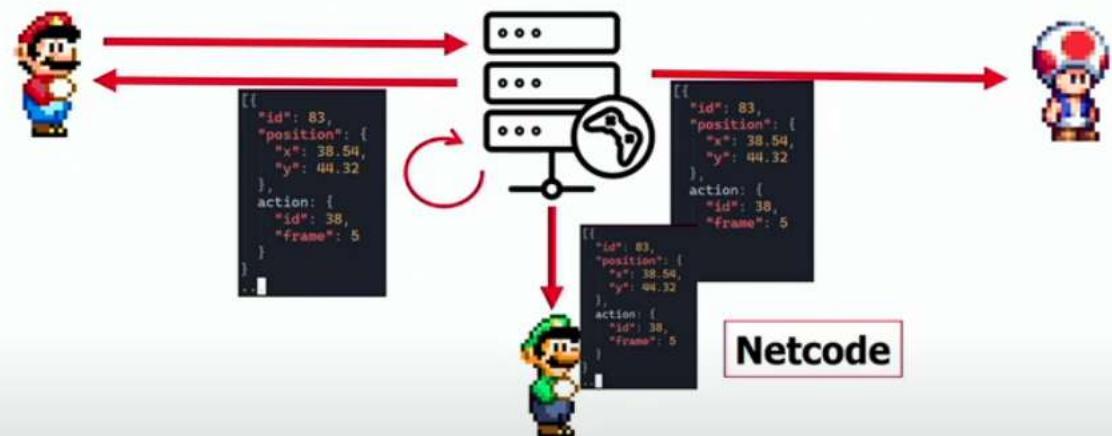
- Memory leaks in Java
- Still referenced elements cannot be GC
- OutOfMemory -> Stacktrace is not useful anymore (since the memory does loose the data/logs which can be useful) so don't trust them in that case

- Static members can hardly be GC, avoid If not necessary, use following right rules: refer to (Sonar) s2386, s2696
- equals/hashCode (example issue with random generate hashCode which can invalid algorithm based on the equality of 2 objects, for hashMap for example)
  - o <https://jqno.nl>equalsverifier/>
  - o <https://github.com/jqno>equalsverifier/>
- Inner classes: Class in class, the outside class may be able to collected, but not the inside class, so none will be...
  - o Should be static ?
  - o Refer to S2694
- Memory leaks (need to be fixed) / memory footprint (can be improve)
  - o Remove references manually
  - o Custom ClassLoader
  - o Minimize the living scope (better local than local in the app, to be unreferenced the earliest we don't need anymore)
  - o WeakReference
- <https://github.com/openjdk/jol>

Jeux en ligne, mais comment ça lag pas ?



## Qu'est-ce qu'on renvoie dans l'état du jeu?



6 | Jeux en ligne, mais comment ça lag pas?

sopra steria



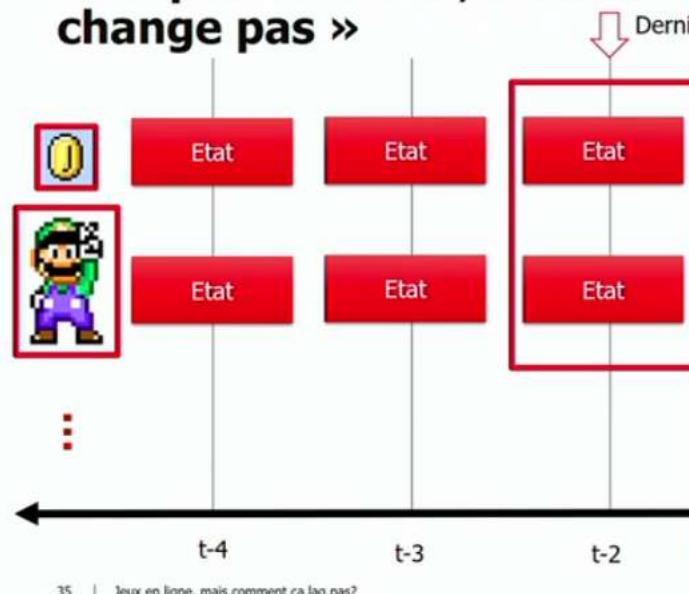
5:51 / 45:12



**Jeux en ligne, mais comment ça lag pas ?**



## « Le plus souvent, l'état des objets change pas »



- 1000 gameObjects
- Etat conservé pendant 3s
- Tickrate de 30 tick/s
- 100 octet/gameObject

$$= 9 \text{ Mo}$$

Jeux en ligne, mais comment ça lag pas ?



Benoit Arnaud

DevOps

# Alive Intelligence

# NodeJS

# Kubernetes

48 | Jeux en ligne, mais comment ça lag pas?

- Gabriel Gambetta - Client-server Game Architecture
- Valve developer community - Source Multiplayer Networking
- Narendra Lakshmana Gowda – How does any MMO Games backend work?

Barnaud/gameClient

Barnaud/gameServer\_poc

[benoit.arnaud@sopra-steria.com](mailto:benoit.arnaud@sopra-steria.com)

sopra steria

@DEVOXXFR



33:36 / 45:12



Google Cloud



THALES



WINAMAX



## Jeux en ligne, mais comment ça lag pas ?

- L'approche “server authoritative”
- “Netcode” gère les cartes (objets statiques) game objects du serveur et game object provenant des autres joueurs
- « tickrate » fréquence de rafraîchissement allège la surcharge
- « protocole » (ici UDP préféré au TCP - qui est strict sur l’ordre des paquets - puisque qu’on veut l’état le plus récent la perte d’un état intermédiaire n’a pas d’importance) + « format de données » (en se basant sur la prédictibilité des types de données) : allège l’utilisation du réseau
- « input lag » lié en partie à l’approche server authoritative, => affichage du jeu pour le joueur en « avance » en se resynchronisant asap « client side prediction », en cas d’une prediction mauvaise => server reconciliation : améliore le jeu pour les joueurs
- « entity interpolation » interprétation des frames intermédiaires pour rendre plus fluide la visualisation du jeu pour les joueurs => créer un frame légèrement en retard mais plus fluide
- Gérer les game objects doit on les renvoyer tout le temps ? (état inchangé entre frame), « delta compression » différence entre l’état d’avant et l’état actuel => améliore la charge réseau
  - o Latence entre joueurs : réutilisation de la connaissance de l’état antérieure pour estimer la latence des joueurs => « lag compensation » tend à gérer la latence
    - Prédictibilité du jeu du joueur : au lieu de compenser la latence (action du joueur en « retard » gérée par le serveur) on va anticiper les actions des autres joueurs (prédition) pour la visualisation de chaque joueur, si la prédition n'est pas bonne le serveur ayant toutes les informations corrige : « rollback based netcode » => améliore les pb de latence, mais ne pourra pas gérer les grosses latences
- Gestion server authoritative et client side prediction (et le reste), nécessite du code des deux côtés server et client (local)
  - o Autre archi connue : peer-to-peer (devient compliqué plus il y a de joueurs pour savoir qui a « raison »)

## L'IA et qualité de code : Construire une synergie avec l'intelligence humaine

The screenshot shows a presentation slide for Devoxx France 2024. On the left, there's a video feed of a speaker on stage, wearing a 'packmind' t-shirt. To the right of the video is a sidebar with a catalog of topics under 'PACKMIND' and 'CATALOGS'. The main panel displays a code editor with several tabs open. A context menu is open over some code, with the 'Identify practice' option highlighted.

```
getExtensionOfPath(path) {
    return path.split('.').slice(-1)[0];
}

getFileName(path) {
    return this.Tools.getFileName(path);
}

async getFilesSortedByDate(userId) {
    const files = await fetchFilesForUser(userId);
    files.sort((a, b) => new Date(b.date) - new Date(a.date));
    return files;
}

Database(userId, sortedFiles);
```

L'IA et qualité de code : Construire une synergie avec l'intelligence humaine

DEVOXX FRANCE 2024

DEVOXX FRANCE 2024

DEVOXXFR

DEVOXXFR

PACKMIND

CATALOGS

- > BDD
- > Clean Code
- > Craft community
- > Front community
- > Hexagonal archite...
- > Html - CSS
- > marketplace web
- > OWASP
- > Tests

Identify Practice

Select space to use

marketplace web

Search a practice by name or category

Functions names should describe side effects

No practices found

Create New Practice

ONGOING SESSIONS

ONGOING BATTLES

USEFUL LINKS

demo-from-develop-saas\*

Google Cloud TALEINANEX

26:11 / 44:24

L'IA et qualité de code : Construire une synergie avec l'intelligence humaine



[\*] packmind

marketplace w...

Tableau de bord

Pratiques

Review

Ateliers d'onboarding

Paramètres

Compte  
Data Board  
Communauté  
Documentation  
Plugins & API Key

## Pratiques > IA suggestions

### Minimize testing multiple methods in a single test case

Unit testing is a method of software testing that verifies the individual units of source code are working properly. A unit test should focus on one feature or method. Testing multiple methods in a single test case can lead to confusion when a test fails and increases the difficulty of diagnosing the problem.

#### Pros of this practice:

- Test cases are simpler and more understandable
- Easy to track down the source of failure
- Makes maintenance easier

To avoid this, every test case should be simple and focus on one single behavior. Separate test cases into multiple smaller ones, each focusing on testing only one method. Read more : <https://stackoverflow.com/questions/6094085/nunit-multiple-tests-in-a-single-test-method>

Testing   Readability

Généré le 09/04/2024

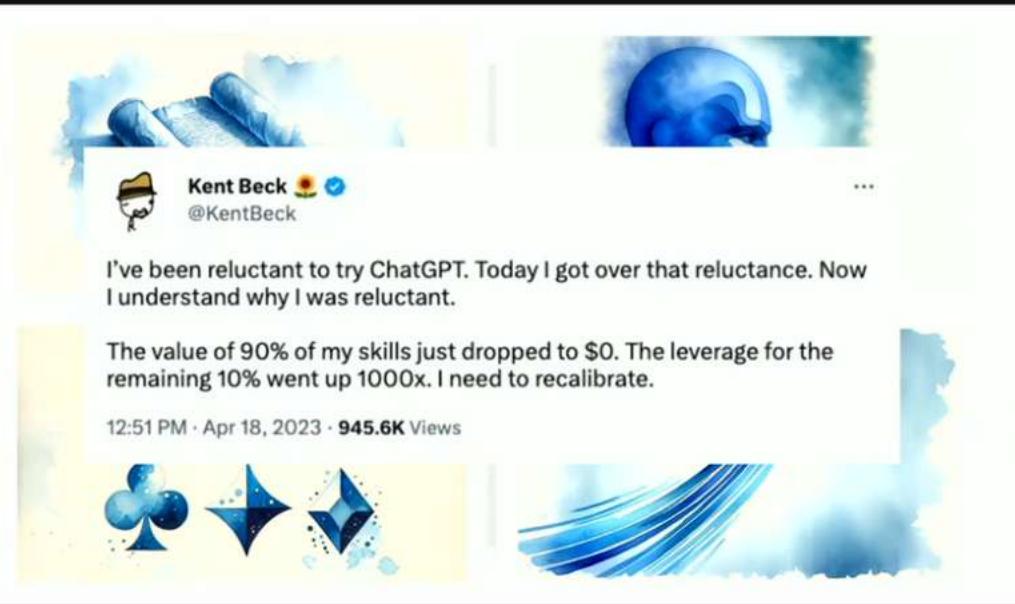
#### Exemples

Supprimer 0

Ajouter à la review +

```
discoveryWorkshopCreationSideMenu.controller.js
43     this.initListener();
44 }
45
46 initListener() {
47     this.$scope.$on('open-discovery-workshop-creation-side-menu', this.openSideNav);
48 }
49
50 Functions names should describe side effects
51
52 const total = await getTransactionsForAccount(accountId);
53 const transactions = await saveGlobalTransactionAmountForAccount(accountId);
54
55
56
57
58
59
60 }
```

L'IA et qualité de code : Construire une synergie avec l'intelligence humaine



@DEVOXXFR

Google Cloud takime THALES WINAMAX

### L'IA et qualité de code : Construire une synergie avec l'intelligence humaine

- Onboarding de l'IA comme un nouvel arrivant, le contexte de l'IA c'est ce qu'on lui donne (fichiers scannés, prompt, ...)
- Responsabilité du code, si on valide aveuglément les propositions, le gain de temps au code peut être perdu en revue de code par non-connaissance de ce qu'on fournit
  - o « code churn » code copié/modifié

- Risque d'augmentation de temps de revue de code, maintenance plus lourde ?
- 5-10% de gain de productivité par individu, pas d'effet au niveau de l'équipe voir complexifie un peu l'orga de l'équipe => il faut que l'équipe soit autant informé et synchrone avec ces nouvelles pratiques
- Risque d'apport de contexte obsolète (exemple des vieux doc confluence qui sont potentiellement plus à jour) l'IA n'est pas capable de savoir si c'est à jour ou non, et prend pour vrai toute info
- Rappel : l'IA n'est pas déterministe => se mettre d'accord dans l'équipe des standards, bonne pratique, distinguer le bon du mauvais, faire la capture ensemble
  - o L'IA ne pourrait elle pas aider à faire cette capture / à s'aider elle-même
  - o **Capturer les pratiques (bonnes comme mauvaises) dans le code et capitaliser l'expérience autour en renseignant le code/le projet et du coup l'IA**
  - o Bonne / Mauvaise pratique peut être mis en draft avant validation de l'équipe
- Diffusion de la bonne pratique, l'IA peut aider
- Attention, on peut très bien avoir des faux positifs et des faux négatifs avec l'IA (non déterministe, une pratique repérée peut être ignoré la fois d'après), l'IA aide mais ne peut pas valider (la qualité du code)
- Lean / regard critique (négatif comme positif) / panel étendue => les gens vont donc pas utiliser de la même manière, et ne doit pas être imposée (exemple TDD, qui est très fortement lié au contexte, peut être difficile voir contreproductive pour une IA, autre exemple document de présentation, certains à l'aise sans, d'autres avec)
- Diminution forte de la valeur liée à la production de code pure / Augmentation beaucoup plus importante de la valeur de notre expertise/contexte/critique/identification des BP
- Gain de productivité plus forte pour un dev junior mais avec une qualité de code moindre et un risque fort de perte de savoir critique : **double tranchant !!**
- Les outils d'aide au code peut identifier ce qui a été généré

Sauve un dév, écris une doc !

DEVOXX FRANCE 2024  
12<sup>e</sup> ÉDITION - DU 17 AU 19 AVRIL 2024

ER  
BEDROCK

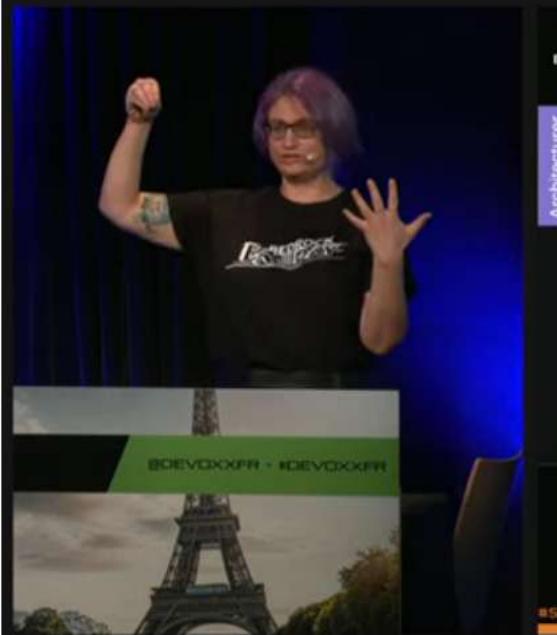
## RÉALITÉ

- Users ne savent pas que le logiciel existe
- Ni à quel problème il répond

@DEVOXXFR - #DEVOXXFR

Google Cloud | Tekina | THALES | WINAMAX

**Sauve un dév, écris une doc !**



Sauve un dév, écris une doc !

- Lecture d'un doc > but connu et attendu : contexte
- Ranger la doc pour mieux s'orienter : contexte, feature, etc.
- <https://diataxis.fr/> : méthode/guide pour faire de la doc

- Notion d'audience
- cory williamson-cardneau “Sticks & Stones: Microaggressions & Inclusiveness”
  - o video <https://www.youtube.com/watch?v=lXXuQQ9xtPc>
  - o slides <https://speakerdeck.com/corywilliamsoncardneau/sticks-and-stones-microaggressions-and-inclusiveness>
- maintenance de la doc
- liens :
  - o <https://www.writethedocs.org/guide/> écrire de la doc
  - o ReadMe Checklist (ddbeck) <https://github.com/ddbeck/readme-checklist>
  - o Feedmerekadmes (lappleapple)
  - o Synonymes : CRISCO <https://crisco4.unicaen.fr/des/> <https://crisco.unicaen.fr/>
  - o Linguee <https://www.linguee.fr/>
  - o Collins Dictionnaire <https://www.collinsdictionary.com/>
  - o Word reference <https://www.wordreference.com/fr/>
- versioning

# Recherche

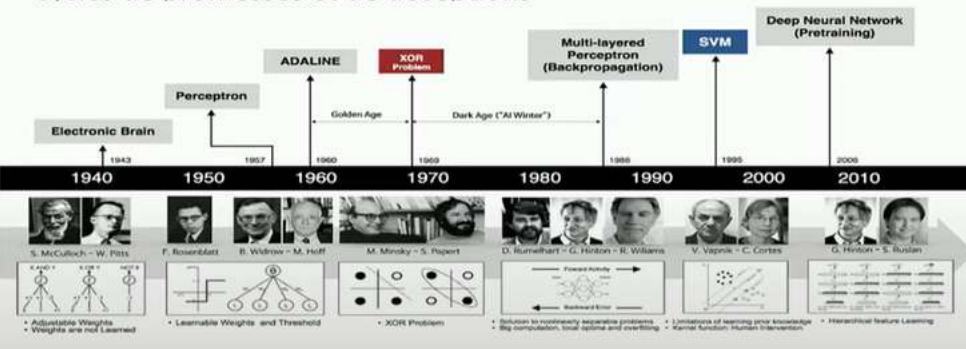
## IA en médecine : où en sommes-nous ?

DEVOXX FRANCE 2024  
12<sup>e</sup> ÉDITION - DU 17 AU 19 AVRIL 2024



### Historique

- 1940 : Premier concept de réseaux neuronaux
- Cycles de promesses et de déceptions



The timeline diagram illustrates the historical development of neural networks:

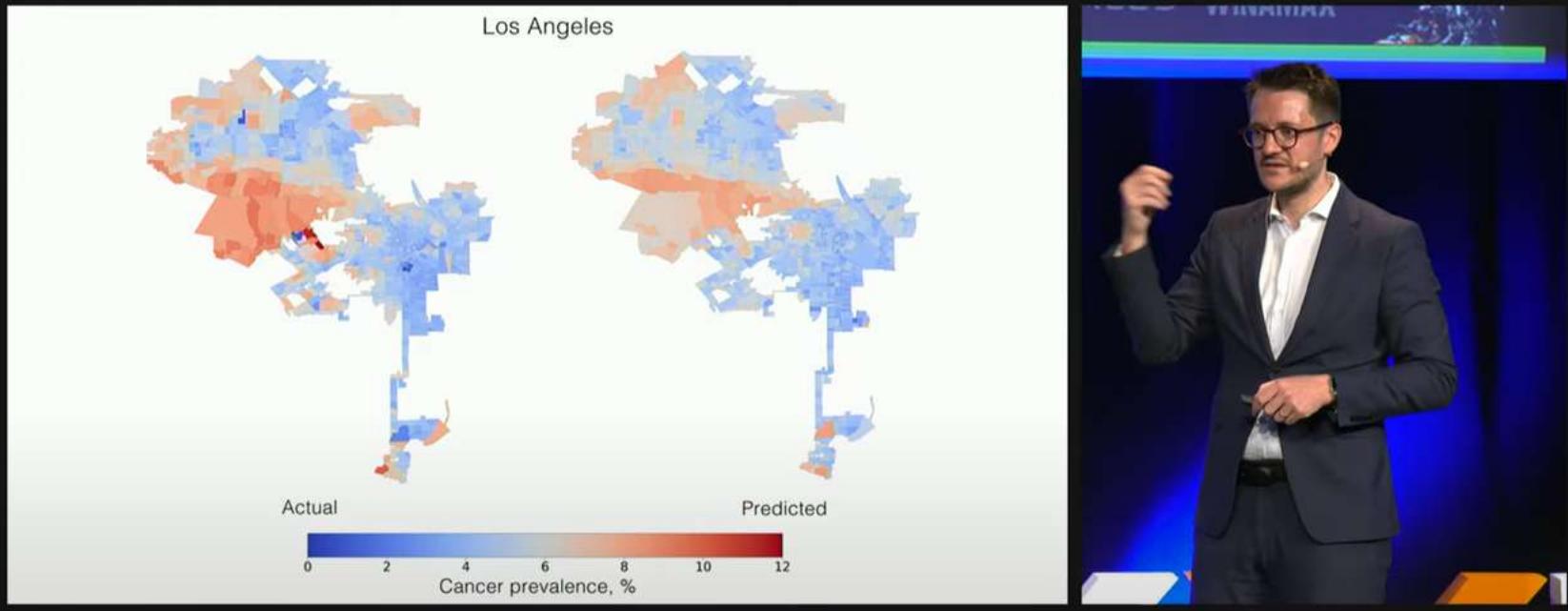
- 1940:** Electronic Brain (S. McCulloch - W. Pitts)
- 1950:** Perceptron (F. Rosenblatt)
- 1957:** ADALINE (B. Widrow - M. Hoff)
- 1960:** Golden Age (M. Minsky - S. Papert)
- 1969:** XOR Problem
- 1970:** Dark Age ("AI Winter")
- 1980:** Multi-layered Perceptron (Backpropagation) (D. Rumelhart - G. Hinton - R. Williams)
- 1986:** Solution to nonlinearity separation problem, Big computation, local optima and overfitting
- 1995:** SVM (V. Vapnik - C. Cortes)
- 2000:** Limitations of learning prior knowledge
- 2006:** Deep Neural Network (Pretraining) (D. Hinton & S. Bengio)
- 2010:** Hierarchical feature Learning

Image source : Andrew Beam

DEVVOXXFR 4:04 / 36:32

Google Cloud TALENT INAMAX

IA en médecine : où en sommes-nous ?



## Ne pas négliger les limites de ces approches en médecine

- Comment en tester la précision future ?
- Comment valider leur utilisation en routine clinique ?
- Essai clinique randomisé ?
- Comment faire évoluer la médecine si tout repose sur l'entraînement d'algorithmes utilisant des données rétrospectives ?

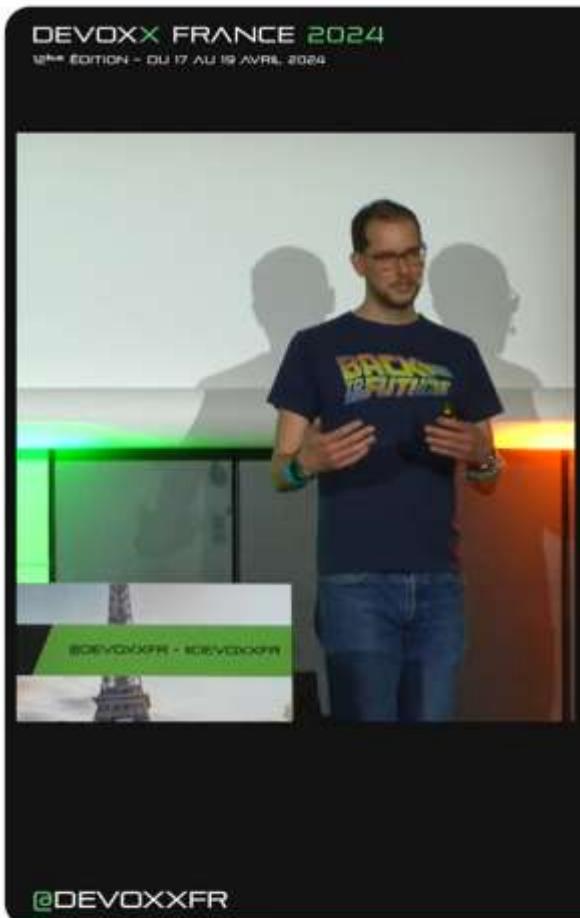


### IA en médecine : où en sommes-nous ?

- Analyse d'imagerie, biais en fonction des informations en dehors de l'image médicale même (exemple nom de l'hôpital qui traite particulièrement telle ou telle maladie)
- Analyse d'image médicale (exemple peau), exemple très réussi d'analyse du cancer de la peau
- Diagnostique pouvant être efficace, notamment sur les maladies rares

- Patient facing IA (notamment l'empathie simulée qui est meilleure qu'un médecin en fin de journée...)
- Pbtq de l'analyse des images à hauts débits (trop d'info sur une image, invisible à l'œil humain comme le bruit d'une photo, peut faussé complètement l'analyse IA) => pb de sécurité et santé lourd
- Pb tels que les données d'entraînement (biais d'hôpitaux, type de peaux, etc.)
- Pb éthique : avec les signaux du réseau neuronale connaître les « pensées »
- Pb majeure de l'IA actuelle, le réentraînement sur les mêmes données en boucle qui elles-mêmes génèrent et utilisent des données simulées
- Une des règles à respecter, utiliser l'IA dans un contexte identique que celui des données d'entraînement (exemple type de peau)
- Nécessité de tests, validations, autant que la validation d'un médicament par exemple avec des essais cliniques
- Nécessité pour ne pas perdre la connaissance de faire de la formation sans aucune aide d'IA (comme le font les pilotes en ligne par exemple)

Tout ce que vous n'avez jamais voulu savoir sur les fuseaux horaires



DEVOXX FRANCE 2024  
12<sup>e</sup> édition - DU 17 AU 19 AVRIL 2024

DES CHANGEMENTS POLITIQUES  
TOUS LES ANS !

- 2022-11-27 DST cancelation in Greenland
- 2022-10-28 Changes in Mexico time zones
- 2022-10-27 Fiji stays on standard time
- 2022-10-05 Permanent daylight saving time in Jordan
- 2022-10-05 Permanent daylight saving time in Syria
- 2022-09-06 Palestine summer time will end one day later than expected
- 2022-08-10 Chile postpones DST one week
- 2022-05-12 Iran cancels daylight saving time
- 2022-03-14 Palestine summer time will start one day later than expected

(Entre 5 et 10 changements chaque année !)

🔗 [https://time.is/time\\_zone\\_news](https://time.is/time_zone_news)

@DEVOXXFR

Google Cloud | Takumi | THALES | WINAMAX

**Tout ce que vous n'avez jamais voulu savoir sur les fuseaux horaires**



## VALEUR PAR DÉFAUT ?

```
new Date('2024-04-17T14:35:00').toString()  
"Wed Apr 17 2024 14:35:00 GMT+0200 (heure d'été d'Europe centrale)"
```

Sur un serveur :

```
new Date('2024-04-17T14:35:00').toString()  
"Wed Apr 17 2024 14:35:00 GMT+0000 (temps universel coordonné)"
```



Tout ce que vous n'avez jamais voulu savoir sur les fuseaux horaires



## DU COUP, ON STOCKE COMMENT ?

value	timezone
2024-04-17T12:35:00	Europe/Paris

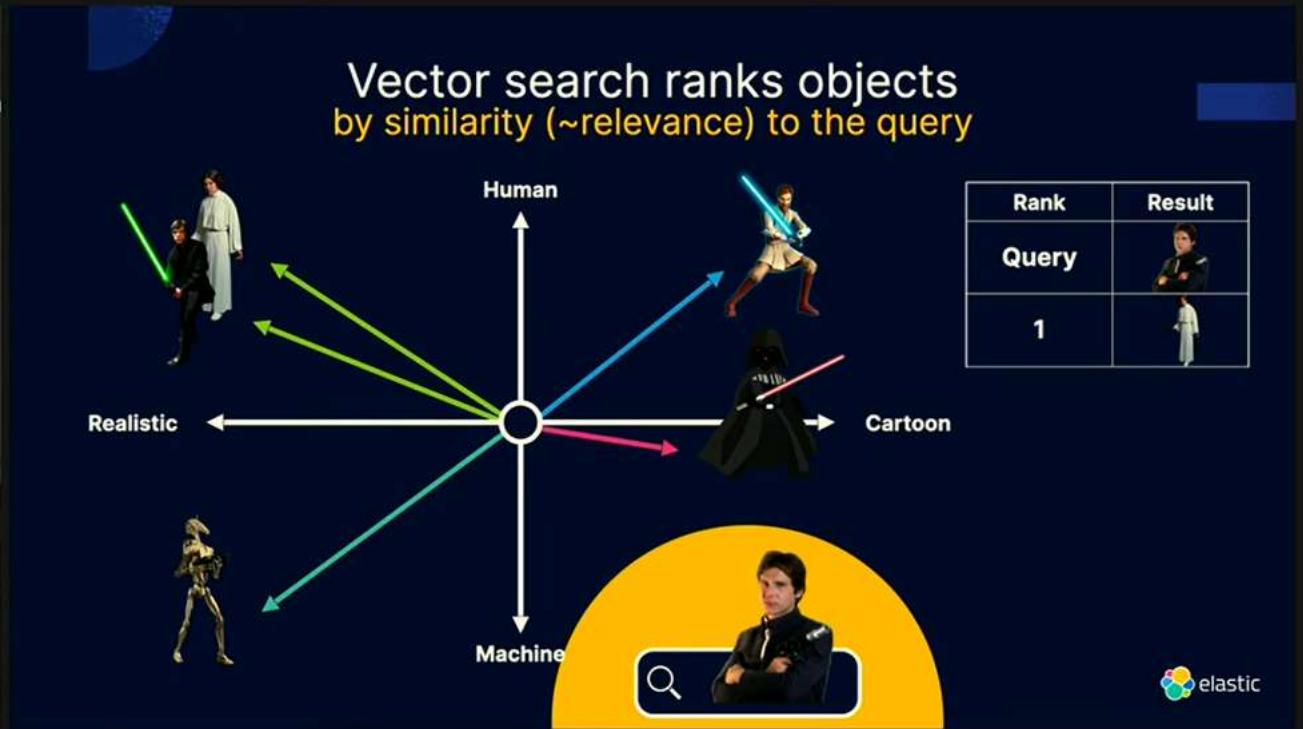
- l'heure locale de la donnée ?
- l'heure UTC de la donnée (et il est donc 14h35 heure de Paris) ?

## Tout ce que vous n'avez jamais voulu savoir sur les fuseaux horaires

- Norme informatique : ISO 8601 / RFC 3339
  - 2025-05-02T10:57:06+02:00 (pb pas de description du lieu !)
  - Toujours pas de consensus complet

- Proposition 2025-05-02T10:57:06+02:00[Europe/Paris]
- Valeur par défaut ?
  - Il n'y en a pas, ça dépend de comment la machine ou le langage fonctionne par défaut (peut être local comme GMT, ou autre)
  - Il faut donc TOUJOURS spécifier le fuseau horaire
- Aucune normalisation entre les langages et techno, certains gèrent les fuseaux (java) d'autres non (js), certains index les mois à partir de zéro (js), d'autres font des mixtes de norme (postgres)
- Bonne pratique usuelle : UTC + lieu

## La recherche à l'ère de l'IA





## Choice of Embedding Model

### Start with Off-the Shelf Models

- Text data: Hugging Face (like Microsoft's E5)
- Images: OpenAI's CLIP

### Extend to Higher Relevance

- Apply hybrid scoring
- Bring Your Own Model: requires expertise + labeled data

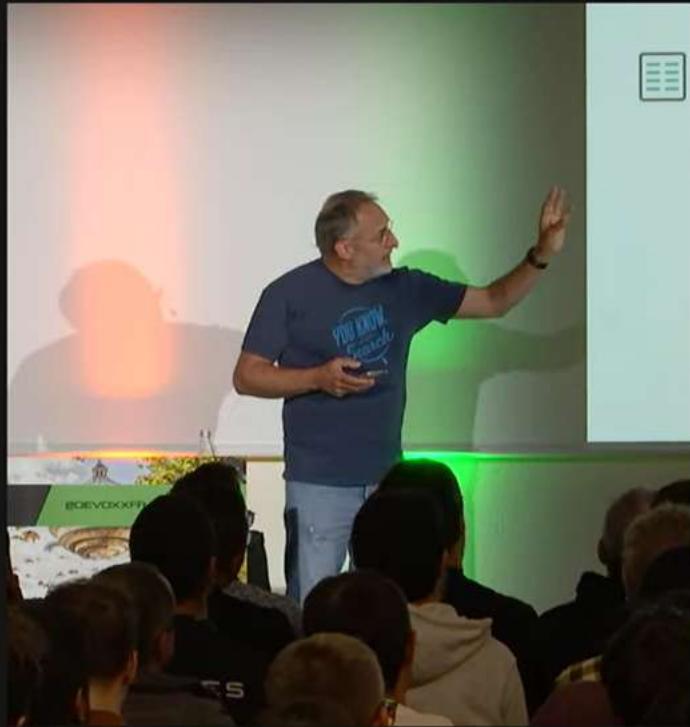


◀ ▶ ⏪ ⏩ ⏴

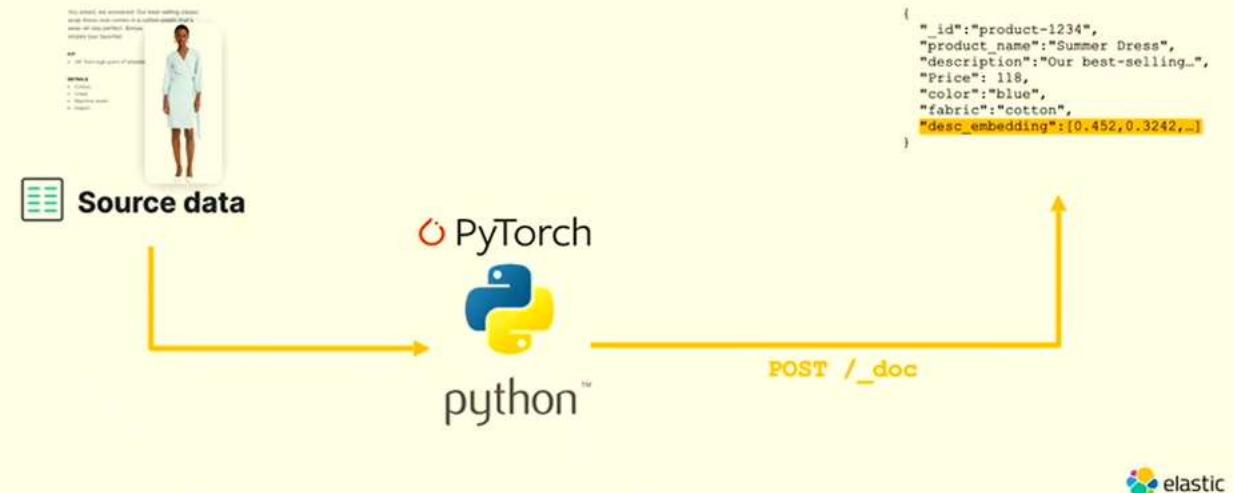
@DEVOXXFR

6:16 / 42:19





## Data Ingestion and Embedding Generation



- Slides (recherche vectorielle dans ElasticSearch) : <https://speaker.pilato.fr/EdLBME/la-recherche-a-lere-de-lia#spJ7XNw>
- There are not the droids you're looking for
  - o Recherche littérale : droid, you, look
  - o Perte de la negation
  - o Recherche littérale  $=/$ = recherche sémantique
- La recherche sémantique => recherche vectorielle
  - o Transformer en Vecteurs
- Modèles (positionnement de Leia, Luc, Darth Vador, Droid, Anakin en cartoon, etc) sur étagère à disposition (pour pouvoir placer Harrison Ford), si insuffisant il faudra l'étendre
- Comment :
  - o D'abord Brut Force (calcul et comparaison avec tous les vecteurs)
  - o Hierarchical Navigable Small Words : en réduisant le spectre / en se rapprochant avec le plus proche : moins précis mais beaucoup plus rapide/efficace
  - o Filtering KNN Vector Similarity : mixte des 2 en fonction du spectre de base
- Reco : ne pas indexer en même temps que la recherche : pour cela utiliser Alias d'index
- Possibilité de filtrer les informations en fonction de qui interroge l'outil

## Apache Lucene : de l'indexation textuelle à l'intelligence artificielle

DEVOXX FRANCE 2024  
12<sup>ème</sup> ÉDITION - DU 17 AU 19 AVRIL 2024

DEVOXX FRANCE  
des Congrès - Paris

DEVOXX FRANCE 2024

DEVOXX FRANCE 2024

DEVOXX FRANCE 2024

30 DEVOXX France 2024 2024-04-19 #DevoxxFR @a2lean adelean

Vectorization dans un vecteur tridimensionnel.

size  
friendliness  
intelligence

Animal	Taille	Amabilité	Intelligence
Chat	0.25	0.85	0.80
Chien	0.30	0.90	0.80
Éléphant	0.90	0.70	0.60
Dauphin	0.60	0.95	0.85
Perroquet	0.15	0.80	0.75

DEVOXXFR 17:45 / 25:06

Google Cloud TOILE INAMAX

Apache Lucene : de l'indexation textuelle à l'intelligence artificielle



Vectorization dans un vecteur tridimensionnel.

Animal	Taille	Amabilité	Intelligence
Chat	0.25	0.85	0.80
Chien	0.30	0.90	0.80
Eléphant	0.90	0.70	0.60
Dauphin	0.60	0.95	0.85
Perroquet	0.15	0.80	0.75

A 3D scatter plot titled "Vectorization dans un vecteur tridimensionnel." showing the vector space of the animals from the table. The axes are labeled "Amabilité" (y-axis), "Intelligence" (x-axis), and "Taille" (z-axis). The points represent the animals: Elephant (top left), Chat (top center), Chien (middle center), Dauphin (bottom center), Perroquet (bottom left), and Dolphin (top right).

31 DEVOXX France 2024-04-19 #DevoxxFR @a2lean adelean

Google Cloud takima THALES WINAMAX

@DEVOXXFR

Apache Lucene : de l'indexation textuelle à l'intelligence artificielle



## Résoudre les deux “problèmes” des LLM

Utiliser un LLM dans le contexte de moteur de recherche doit faire face à deux problèmes majeurs :



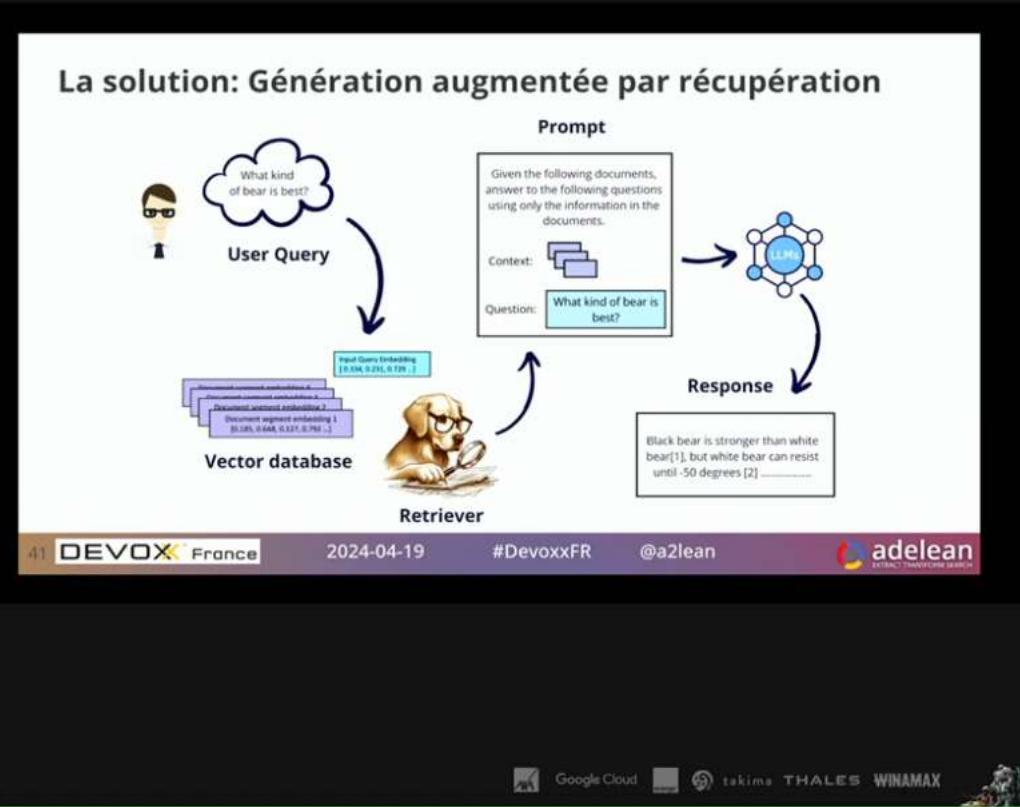
Pas de connaissances en temps réel



Hallucinations

2024-04-19 #DevoxxFR @a2lean





@DEVOXXFR

### Apache Lucene : de l'indexation textuelle à l'intelligence artificielle

- Etat et historique
- « Index inversé » : à partir d'un texte > tokenisation> sélection/tri (> stemming, stopword, etc.) > liste des token
- « analyse de texte » : analyse à l'indexation et analyse à la recherche, autocomplete, ...
- Stockage : « Full text » ; « Stockage colonne » ; « BDD spatiale »

- Recherche vectorielle : comparer la représentation vectorielle d'un objet (image, texte, etc.) indexé avec la représentation vectorielle du recherche
  - o Vecteur sémantique creux (« sparse ») ou dense
  - o « term expansion : ~synonyme exemple : jungle ~ terrain

## Upgrade

[Bootiful Spring Boot 3.3](#)



```
1 # Bootiful Spring Boot 3
2
3 Josh Long
4
5
6 Spring Developer Advocate
7 Kotlin Google Developer Expert (GDE)
8 Java Champion
9
10 Youtube: "Coffee and Software with Josh Long":
11 https://www.youtube.com/@coffeesoftware
12 code: https://github.com/joshlong/bootiful-spring-boot-3
13 email: josh@joshlong.com
14 Twitter/X: @starbuxman
15
16 - Meet an OSS Project: Spring today @ 17:00 / Paris 243 https://www.devoxx.fr/
17 schedule/talk/?id=78001
18 - Password-less apps: implementing WebAuthN avec Daniel Garnier Moiroux https://www.devoxx.fr/schedule/talk/?id=12561
19
```



2:25 / 45:57



## Bootiful Spring Boot 3.3



```
IntelliJ IDEA  File  Edit  View  Navigate  Code  Refactor  Build  Run  Tools  VCS  Window  Help
service  Version control
compose.yaml  application.properties  schema.sql  ServiceApplication.java  data.sql  pom.xml (service)
6
7 @SpringBootApplication
8 public class ServiceApplication {
9
10    public static void main(String[] args) { SpringApplication.run(ServiceApplication.class, args); }
11
12}
13
14
15
16
17 @interface Loan  permits SecuredLoan, UnsecuredLoan{
18}
19
20 final class SecuredLoan implements Loan {
21}
22
23 record UnsecuredLoan(float interest) implements Loan {
24}
25
26 // Data Oriented Programming
27 // - records
28 // - sealed types
29 // - smart switch expressions
```



IntelliJ IDEA interface showing a Java code editor for ServiceApplication.java. The code implements a Loan interface with SecuredLoan and UnsecuredLoan implementations. A tooltip for the variable 'sl' is open, showing options like 'Rename 'sl' to 'ignored'' and 'More actions...'. The status bar at the bottom right shows '20.20 LF UTF-8 4 spaces'.

```
14 }
15 ...
16 class Loans {
17 ...
18     String displayMessageFor(Loan loan) {
19         return switch (loan) {
20             case UnsecuredLoan(var interest) -> "ouch! that " + interest + "% is going to hurt";
21             case SecuredLoan sl -> "good job. well done. nice loan.";
22         };
23     }
24 }
25 ...
26 sealed interface Loan permits SecuredLoan, UnsecuredLoan {
27 }
28 ...
29 final class SecuredLoan implements Loan {
30 }
31 ...
32 record UnsecuredLoan(float interest) implements Loan {
33 }
34 ...
35 // Data Oriented Programming
```

- [Spring Initializr](#)
- Threads : optimisation de la gestion des threads (Java 21)

## Angular : le renouveau (v17) (Uni) [à reprendre]

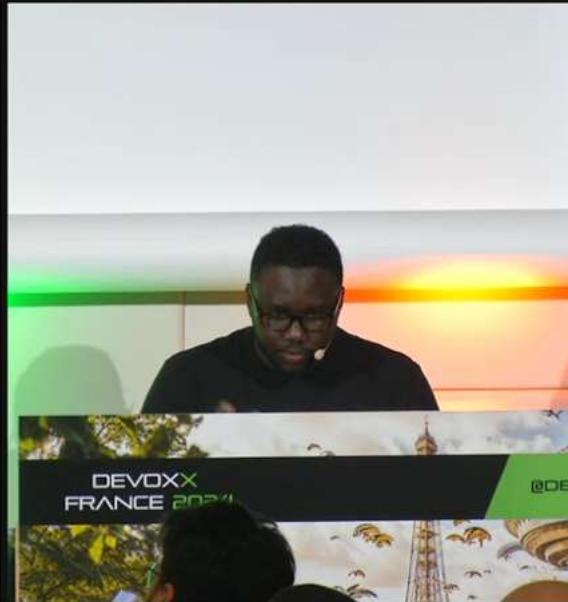
The slide features the Angular v17 landing page on the left, which includes sections for Play-ground, esbuild + Vite, View transitions, Interactive tutorials, Contributors (346), and Deferrable views. It also highlights Hydration, Faster builds for hybrid rendering (Up to 87%), Faster client-side rendering (Up to 67%), and Faster runtime with built-in control flow loops (@for, @if, @switch). The Angular logo is prominently displayed in the center. On the right, a speaker in a black t-shirt is gesturing while speaking. The bottom of the slide shows a video player interface with a progress bar at 4:23 / 2:46:29, a Google Cloud logo, and other presentation controls.

Angular : le renouveau

A screenshot of a code editor (Visual Studio Code) displaying a file named `twitter.pipe.ts`. The code defines a pipe with the name 'twitter' and standalone true. It contains a single method `transform` which returns a URL starting with `'https://twitter.com/@$value'`. The terminal below shows the command `ng generate pipe twitter` being run, creating two files: `twitter.pipe.spec.ts` and `twitter.pipe.ts`.

```
3  @Pipe({
4    name: 'twitter',
5    standalone: true
6  })
7  export class TwitterPipe implements PipeTransform {
8
9    NO usages
10   transform(value: string): string {
11     return `https://twitter.com/@${value}`;
12   }
13 }
14 
```

```
-/D/ng-teams % main ± ng generate pipe twitter
CREATE src/app/twitter.pipe.spec.ts (191 bytes)
CREATE src/app/twitter.pipe.ts (239 bytes)
-/D/ng-teams % main ± 
```



97

Le mécanisme qui permet à Angular de savoir que **le modèle a changé** et qu'il faudrait **mettre à jour la vue** s'appelle le "**change detection**". Il s'appuie sur une bibliothèque qui s'appelle **zone.js**.



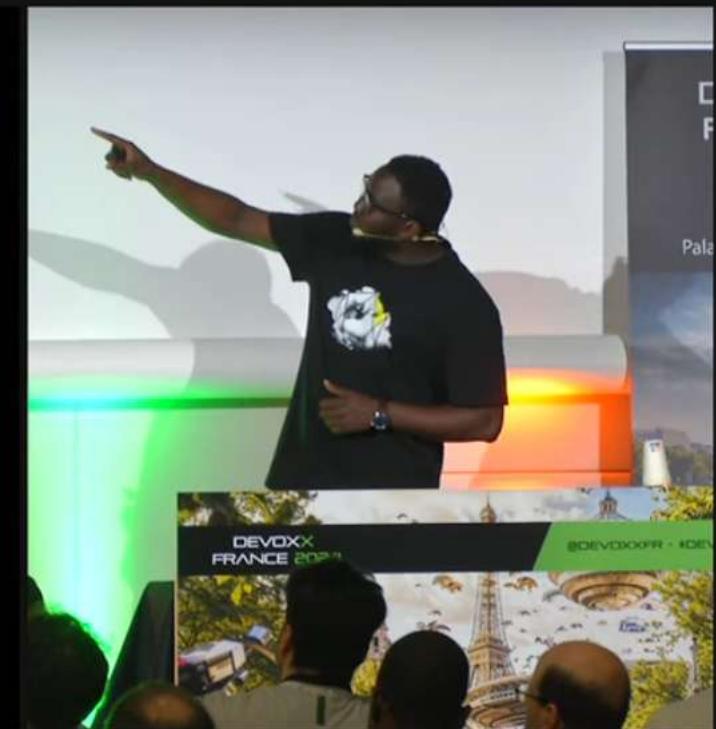
1:31:17 / 2:46:29

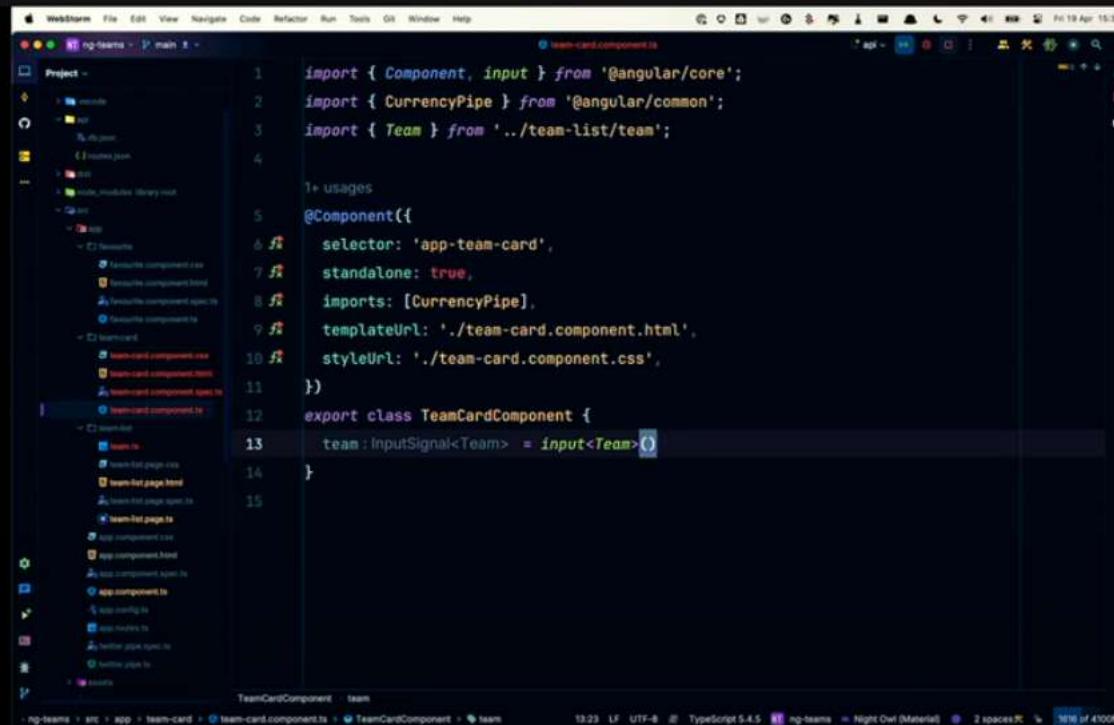


## Angular : le renouveau

## Comment fonctionne le change-detection OnPush ?

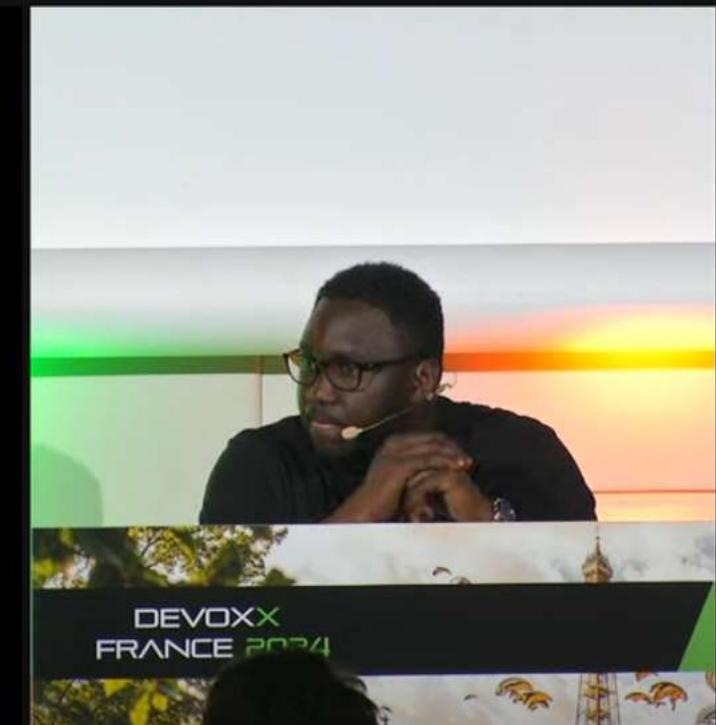
1. La référence d'un @Input a-t-il changé ?
2. Un évènement qui intéresse Angular dans la branche du composant a-t-il été émis?
3. Le composant a-t-il été explicitement marqué dirty ?



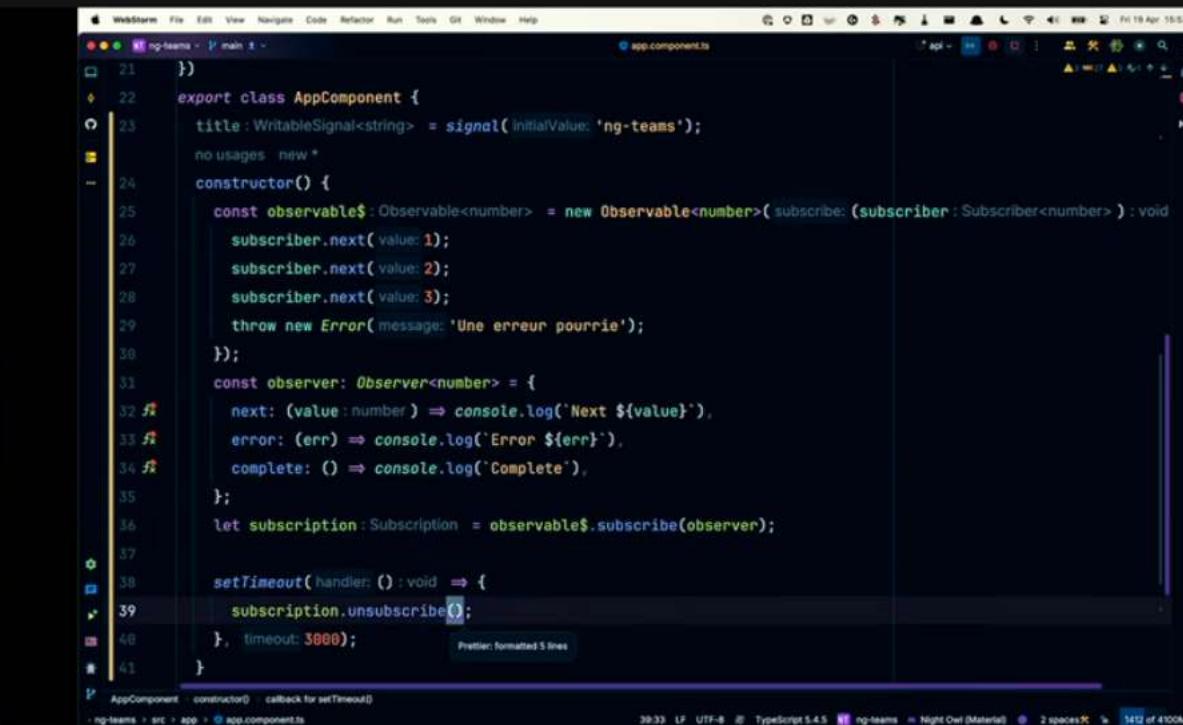


```
import { Component, input } from '@angular/core';
import { CurrencyPipe } from '@angular/common';
import { Team } from '../team-list/team';

@Component({
  selector: 'app-team-card',
  standalone: true,
  imports: [CurrencyPipe],
  templateUrl: './team-card.component.html',
  styleUrls: ['./team-card.component.css'],
})
export class TeamCardComponent {
  team: InputSignal<Team> = input<Team>()
```



## Angular : le renouveau



A screenshot of a developer's workspace in WebStorm. The main window shows a file named `app.component.ts` with the following TypeScript code:

```
21 }
22 export class AppComponent {
23   title: WritableSignal<string> = signal(initialValue: 'ng-teams');
24   no usages new *
25   constructor() {
26     const observable$ : Observable<number> = new Observable<number>(subscriber: Subscriber<number>) : void {
27       subscriber.next( value: 1);
28       subscriber.next( value: 2);
29       subscriber.next( value: 3);
30       throw new Error( message: 'Une erreur pourrie');
31     });
32     const observer: Observer<number> = {
33       next: (value: number) => console.log(`Next ${value}`),
34       error: (err) => console.log(`Error ${err}`),
35       complete: () => console.log(`Complete`),
36     };
37     let subscription: Subscription = observable$.subscribe(observer);
38
39     setTimeout( handler: () : void => {
40       subscription.unsubscribe();
41     }, timeout: 3000);
42   }
43 }
```

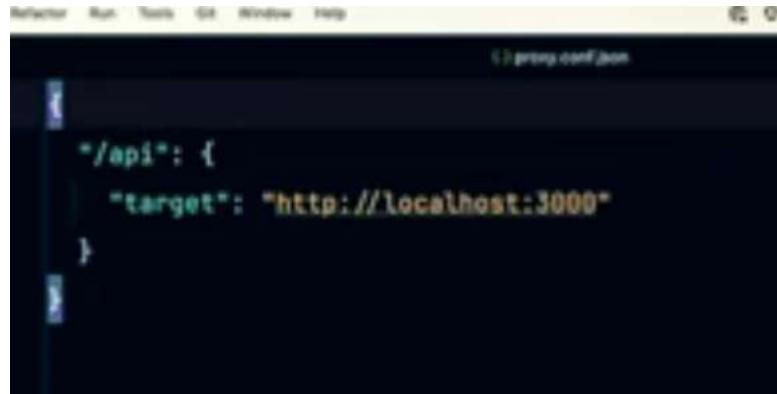
The status bar at the bottom indicates the file is 1412 of 4100M.



- AngularJS v1 / Angular v2-16 (réécriture du framework par rapport à AngularJS) / [pendant Angular v7-14 réécriture du compilateur] / Angular 17 (renouveau sans réécriture du framework)
- Amélioration de la doc / publication de la Roadmap / publication RFC / « developer review »
- ng-update
- framework dogmatique (1 façon de faire)
- nodeJs runtime javascript / npm gestionnaire de paquages
- CLI pour le code et pour le CI/CD
- ng new <projet name>
- ng serve
  - o si << Error: This command is not available when running the Angular CLI outside a workspace >> dans VSC, clique droit sur le projet:
    - Open in terminal
  - o <http://localhost:4200/>
- ng test
  - o execution de tests avec Karma et Jasmine
  - o <http://localhost:9876/>
  - o Le fichier de config de Karma ne sont plus dans le projet
- Les fichiers d'environnement ont aussi disparu, idem pour polyfills
  - o Fichier environnement, n'existe plus par défaut, mais on peut le regénérer ; on peut aussi faire autrement (en buildant par environnement par exemple)
- ng lint n'est plus par défaut, par le ng add on peut l'ajouter rapidement
- rôle du compilateur angular => transformation en Javascript ?
- standalone : avant il fallait utiliser ng module (?)
- property binding : ajout des crochets ex : [src] pour ajouter dynamiquement une propriété
- écouter un évènement : ajout des parenthèses ex : (clic)
- référence avec #

- pipe
  - o pipe pour faire du formatage directement dans le template : << mydate | date: format: 'fulldate' >>
  - o ng generate pipe <nom du pipe>
  - o on peut chainer les pipes
- directive
  - o attribut ; apparence d'un élément
  - o structurel : modifie la structure de l'élément (ex : nglf)
- amélioration/simplification
  - o => @if {} qui est plus naturel, notamment avec le } @else {}
  - o => @switch ... @case
  - o => @for ... track (le track permet à angular de savoir si un objet a changé)
    - ... @empty (pour gérer directement si la liste est vide)
  - o ...
- ng template / ng contener
  - o ng template : bout de code réutilisable
  - o ng contener : permet d'éviter de spécifier une autre balise (ex : div)
- ng generate component team-list --type page
  - o --type : permet de spécifier le type, ainsi au lieu d'avoir team-list.component, on obtient ici team-list.page
- Web component issu d'autres techno pas encore supporté, en cours d'étude (voir roadmap)
- Change detection/zone.js (mise à jour de la vue du à la modification du modèle)
  - o Zone.js patche tout le code asynchrone, divers API async patchés ex. setTimeout, Promise, fetch, FileReader...
  - o Le fonctionnement par défaut, le change detection traverse toutes les possibilités qui ont pu changer
  - o OnPush : pour optimiser le change detection, en skippliant les branches où il n'y a pas eu de changement (référence input ? émission d'évènement qui nous intéresse ? composant explicitement dirty ?)
    - Avantage => plus optimisé que par défaut, pour dev, c'est simple : dirty checking

- Inconvénient => Reste trop fréquent : pb de perf pour les applications telles que la modélisation 3D, tous les projets angular ont zone.js (100kB)
- « Programmation réactive » en utilisant les **signaux** : on encapsule une var avec un signal (computed ?)
  - on « set » un signal « comme » un atomique en java, avec .set()
  - ou avec .update(), avantage par rapport à .set(), est qu'on connaît la valeur avant l'update
- divers moyens d'utiliser les signaux dont input basé sur les signaux
  - on s'épargne le besoin de faire un input et un output avec le signal
- erg(?)js : gestion des observable (=collection de valeurs ou d'évènements qui arrivent au cours du temps)
  - dans la pratique les observables sont rarement voir jamais écrits explicitement (comme en démo) car ils font partie de beaucoup d'élément tel que les requêtes http
- injecter des composants
  - teamService : TeamService = inject(TeamService) ;
  - plus besoin de constructeur
- ex : Back 8080 Front 4200, erreur CORS => utiliser proxy angular



A screenshot of a code editor showing a file named 'proxy.conf.json'. The file contains the following JSON configuration:

```
"/api": {  
  "target": "http://localhost:3000"  
}
```

- 
- **Signal ou Observable ?**
  - Step management => préférable signal
  - Asynchrone => observable
-

# Architecture

[Architecture Hexagonale: Comment venir à bout du code spaghetti ? \(Kévin LLOPIS\) \[pratiquer ?\]](#)

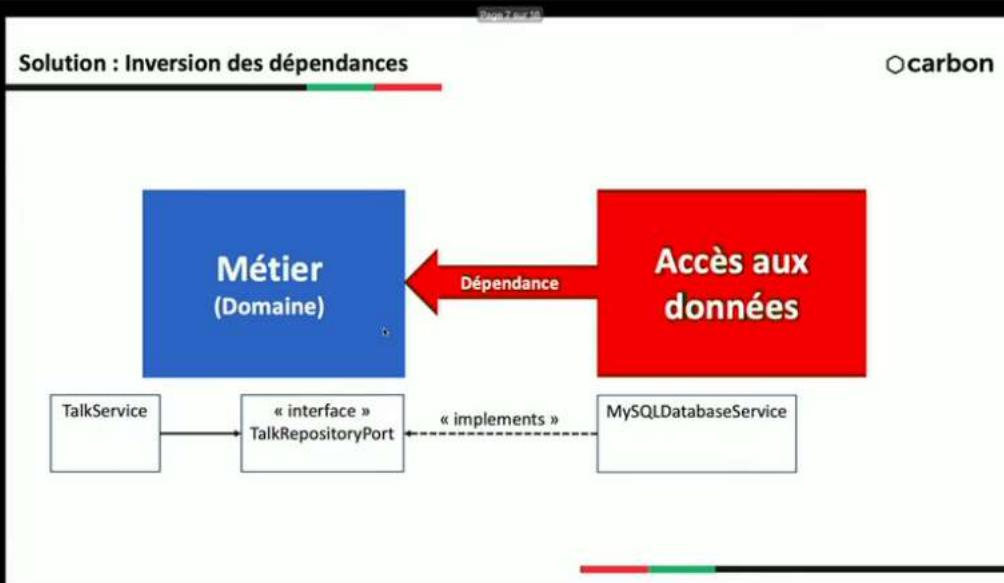
The screenshot shows a video player interface with a presentation slide. The top bar of the video player displays "DEVOXX FRANCE 2024" and "12<sup>e</sup> ÉDITION - DU 17 AU 19 AVRIL 2024". The slide has a title "Problématique : MySQLDatabaseService" and a progress bar at the bottom. The main content is a diagram illustrating a hexagonal architecture pattern:

```
graph LR; subgraph Hexagon [Hexagonal Architecture]; direction TB; A[Métier  
(Domaine)] --> B[Accès aux  
données]; end; C[TalkService] --> D[MySQLDatabaseService];
```

The diagram shows a blue box labeled "Métier (Domaine)" connected by a blue arrow labeled "Dépendance" to a red box labeled "Accès aux données". Below this, a white box labeled "TalkService" has a black arrow pointing to a white box labeled "MySQLDatabaseService". A progress bar at the bottom of the slide is mostly red with a small green segment.

The video player interface also includes a progress bar at the bottom left, a timestamp of "4:23 / 14:49", and a set of control icons at the bottom right, including symbols for Google Cloud, volume, and other media controls.

[Architecture Hexagonale: Comment venir à bout du code spaghetti ? \(Kévin LLOPIS\)](#)



## Comment préserver cette architecture dans la durée ?

Avec ArchUnit ✓

```
@AnalyzeClasses(  
    packages = "fr.kevin.llps.conf.event.reminder.domain",  
    importOptions = {DoNotIncludeTests.class})  
public class DomainDevelopmentRulesTest {  
  
    @ArchTest  
    public static final ArchRule DOMAIN_DEVELOPMENT_RULE =  
        classes()  
            .that()  
            .resideInAPackage(DOMAIN_PACKAGE)  
            .should()  
  
    //== 1 - Classes Domain --[Appel champs/méthodes]--> Packages autorisés (Domain, Java) ==  
    //== 2 - Accesseurs autorisés --[Appel champs/méthodes]--> Classes Domain ==  
    .onlyAccessClassesThat()  
    .resideInAnyPackage(ALLOWED_ACCESSED_PACKAGES)  
    .andShould()  
  
    .onlyBeAccessed()  
    .byClassesThat()  
    .resideInAnyPackage(ALLOWED_ACCESSOR_PACKAGES);  
}
```

carbon

DEVOXX  
FRANCE

#DEVOXXFR

des Congrès - Paris

DEVOXX FRANCE  
2024

X DEVOXXFR X DEVOXXFR



## Architecture Hexagonale: Comment venir à bout du code spaghetti ? (Kévin LLOPIS)

- Architecture Hexagonale peut aider à solutionner des problèmes
  - o Problématique 1 : migration technique => **inversion des dépendances** : au lieu d'avoir un service métier qui utilise la dépendance vers un service BDD par exemple, on passe par une **interface**
- Comment inverser les dépendances (exemple) :

- Créer l'interface du service pour la BDD (infra) « TalkRepositoryPort » avec les méthodes qui nous intéressent
  - Dans le service (côté métier/domain) « TalkService » ajouter le Port « TalkRepositoryPort » (final)
  - Implémenter « TalkRepositoryPort » dans l'implémentation « MySQLDatabaseService »
  - Dans le service (côté métier/domain) on peut appeler les méthodes de « TalkRepositoryPort »
  - Créer le Mapper « TalkRepositoryAdapter » qui implemente « TalkRepositoryPort », avec TalkRepository + TalkMapper
  - ...
  - <https://github.com/kevin-lbps/hexagonal-conference-event-reminder>
- Comment faire perdurer :
    - Outil : ArchUnit : exécuter des règles

[L'Architecture Hexagonale par la pratique, le live coding qui rendra vos applications pl...](#)

The image is a composite of two panels. The left panel shows a man with a beard and beret speaking at a podium during a DevOxx France 2024 event. The right panel features a portrait of Yoda with three quotes overlaid: "Coupling is the path to the dark side", "Coupling leads to fragility. Fragility leads to legacy", and "Legacy leads to suffering". The quote "Legacy leads to suffering" is attributed to Yoda.

DEVOXX FRANCE 2024  
12<sup>e</sup> ÉDITION - DU 17 AU 19 AVRIL 2024

EVOXX FRANCE 2024

@DEVOXXFR - #DEVOXXFR

"Coupling is the path to the dark side

Coupling leads to fragility  
Fragility leads to legacy

Legacy leads to suffering

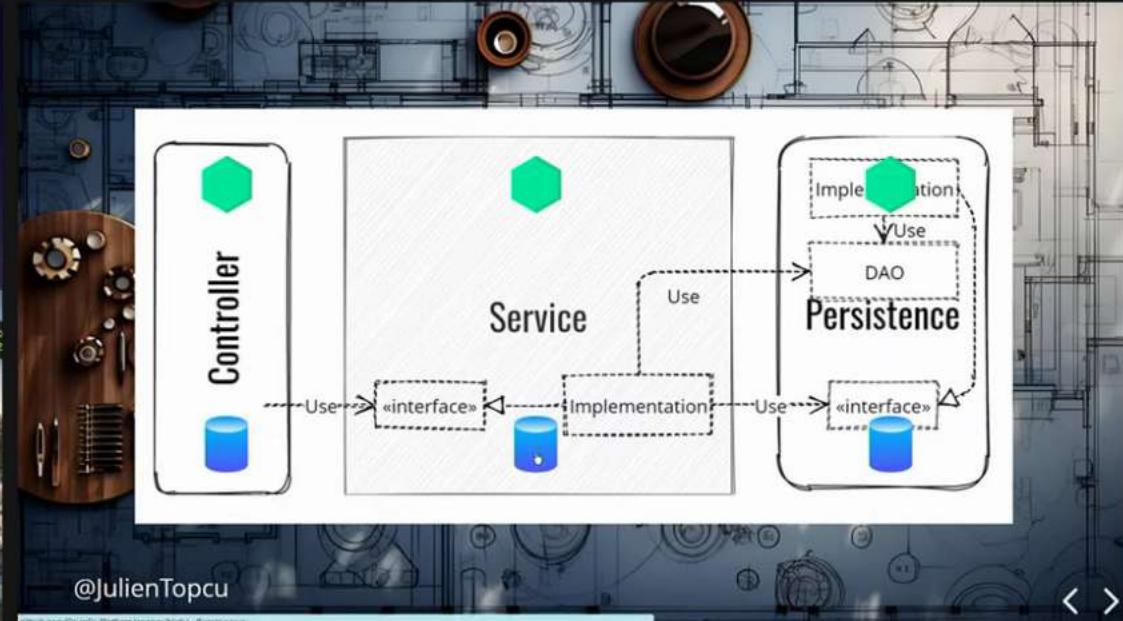
— Yoda

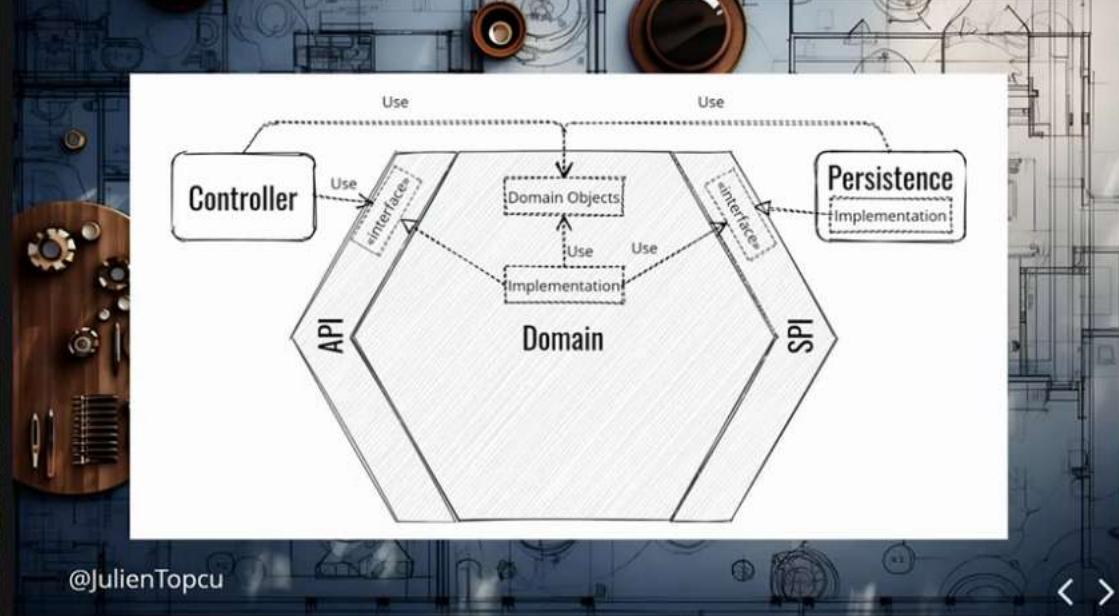
@JulienTopcu

@DEVOXXFR

Google Cloud takim THALES WINAMAX

[L'Architecture Hexagonale par la pratique, le live coding qui rendra vos applications pl...](#)





The diagram illustrates the Hexagonal Architecture pattern. It is divided into two main sections: INFRASTRUCTURE (left) and DOMAIN (right). The INFRASTRUCTURE section contains 'STARSHIPRESOURCE', 'FLEETRESOURCE', 'CONTROLLER', and 'INTEGRATIONTEST'. The DOMAIN section contains 'STARSHIP', 'FLEET', 'ASSEMBLER', and 'STUB'. A central hexagon labeled 'ASSEMBLER' connects the infrastructure and domain layers. External interfaces include 'ANTICORRUPTION LAYER' (dashed line), 'API' (solid line), and 'SPI' (dashed line). Arrows labeled 'USE' indicate dependencies between components.

@JulienTopcu

L'Architecture Hexagonale par la pratique, le live coding qui rendra vos applications pl...

- Problème de l'architecture 1/3 : contrôleur - service - persistance ;
  - o où est le métier ? idéalement dans le service, mais en vrai partout !
  - o idem la persistance est partout
  - o métier couplé au framework

- si on migre ? c'est compliqué
- Architecture
  - Logique Métier => « Domain » (l'hexagone c'est l'intérieur et la logique métier)
  - L'infra => extérieur de l'hexagone
  - Persistance dépend du domaine et par l'inverse
- Comment
  - Mise en place du « domain »
    - Ecriture des tests métiers (TDD)
    - Ecriture des méthodes (idéalement via interface, mais ok si implémentation directe) : dans une couche « api » ou « features »
    - Récupérer des données depuis la persistance ! ne pas appeler/rendre dépendant !
      - => création d'une interface pour requêter l'extérieur du domaine ; dans la couche « spi » (?)
    - « spi » interface pour requêter l'extérieur
    - Mock des données, tests logique interne pas intégration
    - ...
  - Mise en place de l'« infrastructure »
    - (archUnit pour vérifier qu'il n'y a pas d'import interdit entre paquets)
    - Ecritures des tests d'intégration (ITDD)
    - Ecriture d'un contrôleur par exemple
      - Appeler le domaine : injecter l'interface par constructeur
    - Annotation spring dans le domaine : à éviter sinon on couple
      - Il faut que Spring connaisse le domaine et par l'inverse
      - Utilisation d'Annotation custom par exemple
      - Scan des composants du domaine
    - Stub (~« bouchon ») des données pour tester le contrôleur => permet de tester en « vrai » (sur postman/curl/navigateur) avant de finir l'infra, permet de paralléliser le travail

- ...
- Mise en place pour utiliser le référentiel SWAPI (<https://swapi.py4e.com/> « The Star Wars API » / API à utiliser via <https://swapi.dev/> ?) (pour l'exemple)
  - Ecriture des tests pour appeler le SWAPI avec mise en place d'un SWAPI client
  - Mock server pour ne pas utiliser le vrai server qui peut être instable et embarquer un SWAPI dans les tests quoi qu'il arrive
  - Comment gérer le format de l'API avec le format du domaine ? tout l'intérêt des adaptateurs/ports, le domaine ne se demande pas le format des infra : Modèle de SWAPI n'est pas celui de mon Domaine
    - Convertir
    - Faire attention à la qualité de l'infra, les exceptions/problèmes de conversion sont à gérer dans l'adaptateur et non le domaine !
  - ...
- Github : <https://gitlab.com/beyondxscratch/hexagonal-architecture-java-springboot/-/tree/main>
- ressource Rest aligné au domaine peut poser problème, comment les rendre indépendants : Mapper ?

## L'architecture micro frontend : comment transformer son site web en puzzle géant collaboratif

DEVOXX FRANCE 2024  
12<sup>e</sup> ÉDITION - DU 17 AU 19 AVRIL 2024

**DU MONOLITHE AU MICRO-FRONTEND**

Monolithe	Front-Back	Micro-services	Micro-frontends

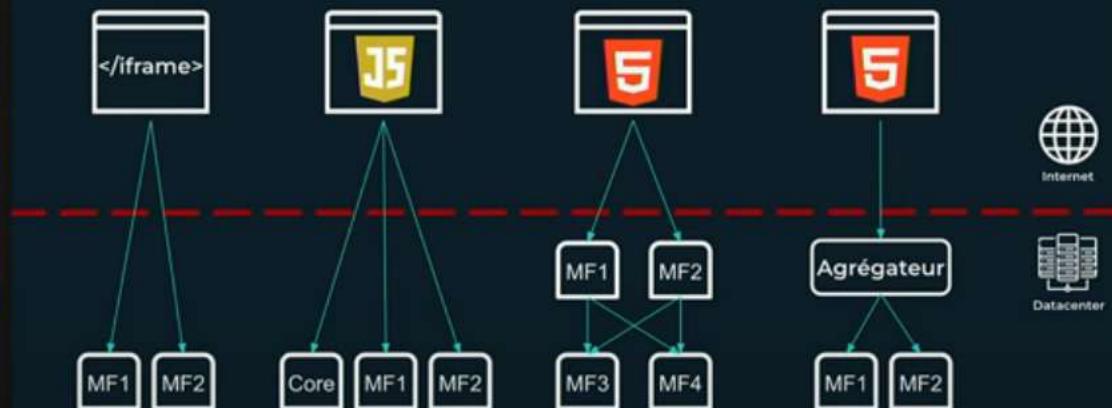
@DEVOXXFR

Google Cloud takim THALES WINAMAX

L'architecture micro frontend : comment transformer son site web en puzzle géant collaboratif



## UNE ARCHITECTURE MICRO-FRONTEND



L'architecture micro frontend : comment transformer son site web en puzzle géant collaboratif

Exemple d'architecture



## UN PEU DE VOCABULAIRE



**Micro Front :**

- application autonome
- protocole d'échange HTTP
- contient des fragments

**Fragment :**

- est un composant métier autonome
- responsable d'aller chercher ses données
- peut être ajouté/supprimé sans impact sur le reste de la page



## RÉSILIENCE

Disponibilité



- Hypothèse : dispo fragment 99.99%



5 fragments  $\Leftrightarrow$  disponibilité **99.95% 99.98%** ( $99.99\%^2$ )

30 fragments  $\Leftrightarrow$  disponibilité **99.7% 99.98%** ( $99.99\%^2$ )



## TECHNO & FRAMEWORK



L'architecture micro frontend : comment transformer son site web en puzzle géant collaboratif

- micro-service à micro-frontend
- comment :
  - o iframe
  - o JS (runtime javascript)

- Système de maillage
- Agrégateur
- Mono repo ou multi repo
- Mono techno ou multi techno
- buts récurrents du choix du micro-front :
  - diminution du « time to market »
  - indépendance des équipes (avec en + multi repo, multi techno par exemple)
- comment les assembler
  - by code
  - by design
- exemple d'architecture
  - agrégateur
  - « registry »/configuration
  - « templating »
  - Gestion des urls « resolver »
  - Backoffice - By design
- Charge liée aux ressources statiques : ressources communes et ressources propres à un micro-front
  - Intérêt : plus facile de supprimer ce qui est obsolète, plus simple à maintenir
- Amélioration gestion de l'impact : alléger les pages web, comment ? > « lazy load » de fragment (en fonction de l'approche view port)
  - Intérêt du micro-front : être plus précis sur la priorisation de chargement et quels éléments en « lazy load »
  - Allégement de page > permet d'utiliser plus longtemps un « device » puisque moins de charge/poids
- A/B tests :
  - mises à dispo de plusieurs versions des fragments en live
  - tests en live avec des jetons
  - permet le live coding (démo)

- Comment migrer ? exemple :
  - o Ajout d'un routeur pour gérer les url afin de rediriger sur le legacy ou le(s) nouveau(x) front(s)
  - o Routeur – agrégateur pour gérer entre autres la cohérence header/footer legacy vs micro-front
- Comment passer à micro-frontend ?
  - o Observabilité impérative pour faire du micro-frontend
  - o Réorganisation impérative pour le passage au micro-front
- Combinaison/interaction, comment ?
  - o Bus d'évènement
  - o Server side rendering
  - o SPA

# Gestion de la mémoire

Understanding 5 languages memory models (or most of them)

DEVOXX FRANCE 2024  
12<sup>e</sup> ÉDITION - DU 17 AU 19 AVRIL 2024

DEVOXX FRANCE

#DEVOXXFR

Palais des Congrès - Paris

ICE

DEVOXX FRANCE 2024

BOEVOXX - KREVOXX

summary

/ SUMMARY

DEVOXX FRANCE 2024

1. The manual model  
C
2. Ownership and borrowing  
Rust
3. Reference counting  
Python
4. Mark and Sweep  
Go
5. Mark and Sweep (Generational)  
Java

@DEVOXXFR

Google Cloud takima THALES WINAMAX

Understanding 5 languages memory models (or most of them)



## / Manual Model

1. Everything goes to the stack
2. You manually ask for heap space
3. Use the heap for dynamic size entities
4. Use the heap for entities that survive the function call



other languages

DEVOXX FRANCE 2024

1. C++: Manual, and ownership or refcounting with smart pointers
2. C# (.NET): Mark and Sweep generational
3. Erlang: Mark and Sweep generational + ref counting for shared heap
4. Fortran: Manual
5. Javascript: Mark and Sweep generational
6. Java (JVM): Mark and Sweep generational
7. Julia: Mark and Sweep generational
8. LUA: Mark and Sweep generational (since 5.4)
9. Pascal: Manual
10. PHP: Reference counting
11. Perl: Reference counting, but Raku Mark and Sweep generational
12. R: Mark and Sweep generational
13. Ruby: Mark and Sweep generational
14. Swift: Reference counting

@DEVOXXFR

Google Cloud takimz THALES WINMAX

### Understanding 5 languages memory models (or most of them)

- Stack & Heap ?
  - o Stack : stocke les fonctions appelées ? statique ?
  - o Heap : besoin de libérer la place après utilisation d'une « fonction » ? dynamique ?

