

En route vers *l'auto-maintenance des tests*

Devenez un·e véritable pilote de la qualité

Sophia Antipolis · ETSI

Avec François-Xavier Le Gal

Une question pour comprendre votre contexte

Quelles ressources dédiez-vous à la maintenance de vos tests automatisés ?

- Pas de personne dédiée, au fil de l'eau
- 1 personne à temps partiel
- 1 personne à temps plein
- Plusieurs personnes régulièrement
- Pas encore d'automatisation

À main levée, sans filtre →

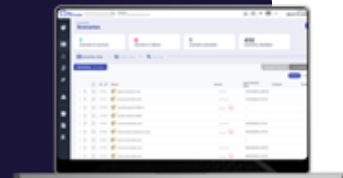
Qui vous parle aujourd'hui



François-Xavier Le Gal

Directeur Général Adjoint

Pilote la vision produit et l'intégration de l'IA dans la plateforme.



Cinq sources, pas des chiffres qui circulent

Chaque donnée citée aujourd'hui remonte à une source primaire vérifiée :

DORA 2025-2026 (Google Cloud) : près de 5 000 pros interrogés, adoption, confiance, taxe de vérification
cloud.google.com/blog (2025) · infoq.com (2026)

METR : organisme indépendant, mesure la productivité réelle des devs avec l'IA
metr.org/blog (juillet 2025, février 2026)

Virginia Tech et Carnegie Mellon (mars 2026) : 22 374 variantes de programmes, huit modèles de langage
Haroon, Khan, Gulzar · arxiv.org/abs/2603.23443 · prépublication

Vathana, Bhatt, Patel, Eisty (juin 2026) : tests IA contre tests humains sur des bugs réels en Python
arxiv.org/abs/2606.08588 · prépublication

Satya Nadella, Microsoft (avril 2025) : jusqu'à 30 % du code de Microsoft écrit par l'IA
cnbc.com · techcrunch.com

Note de synthèse avec liens complets disponible sur demande

En F1, un pit-stop dure *2 secondes.*

Parce que tout le reste a été préparé, mesuré, répété.



Automatiser, c'est devenu standard.
Maintenir, c'est devenu le problème.

5

minutes par mois et par scénario

c'est le coût d'entretien du parc, chez nos clients

Donnée interne Mr Suricate

30%

du code Microsoft

est déjà écrit par l'IA

Source : S. Nadella, avril 2025, formulation exacte : jusqu'à 30 % (CNBC / TechCrunch)

Le coût n'est pas dans l'outil. Il est dans les heures passées par nos collaborateurs.

Du correctif manuel *à l'auto-maintenance intelligente.*



Self-healing :

ce qu'il fait bien, ce qu'il rate.



Ce qu'il fait bien

- Détecte les sélecteurs cassés
- Propose un sélecteur par défaut
- Évite l'échec immédiat du test



Ce qu'il rate

- Ne comprend pas pourquoi le sélecteur a changé
- Risque de faux positifs (le test passe à tort)
- Pas de raisonnement sur la cause profonde

Réparer sans comprendre, c'est fabriquer du vert qui ne veut rien dire. Le vrai problème n'est pas de réparer un test. C'est de savoir s'il faut le réparer.

Diagnostic avant réparation : *quatre causes, une seule bonne réponse.*



On adapte le test

- Changement d'UI légitime : le composant a bougé
- Parcours métier modifié : le produit a évolué
- Dans ces cas : on met à jour le scénario, sereinement



On ne touche pas

- Bug applicatif : le test a raison
- Flaky ou instabilité d'environnement : c'est du bruit
- Dans ces cas : on n'adapte jamais le test pour faire taire l'alerte

Le bon diagnostic, c'est la moitié du travail. Le reste, c'est savoir ne rien faire quand il ne faut rien faire.

Ce qui rend une correction fiable : *l'oracle de test.*



Exécuter ≠ vérifier

Une IA peut atteindre la zone du bug sans détecter le défaut, si son critère de succès est mal posé.



Couverture ≠ détection

Beaucoup de tests et beaucoup de code couvert ne garantissent pas qu'on détecte les bons bugs.



L'oracle, la vraie difficulté

Pourquoi ce produit coûte 79,90 € ? Pourquoi cette API renvoie 403 ? C'est de la connaissance métier.

Notre approche : 3 étapes, *1 humain au contrôle.*



Une démo Mr Suricate à la fin de cette partie.

Analyse & classification

Avant de réparer, il faut savoir de quoi on parle.



Capture multimodale

Screenshot + DOM + logs + intention du test, capturés simultanément à l'instant de l'échec.



Classification automatique

L'IA distingue : bug réel, élément déplacé, popup parasite, timeout, instabilité réseau.



Taggage avant traitement

Chaque erreur est qualifiée avant tout déclenchement d'action. On ne corrige pas à l'aveugle.

Rejeu de maintenance

Confirmer la cause avant d'alerter qui que ce soit.



Rejeu en environnement isolé

Le test est relancé en conditions contrôlées pour confirmer ou infirmer l'échec initial.



Différenciation des causes

Flaky test ? Vrai bug applicatif ? Changement légitime de l'UI ? Chaque cas est traité différemment.



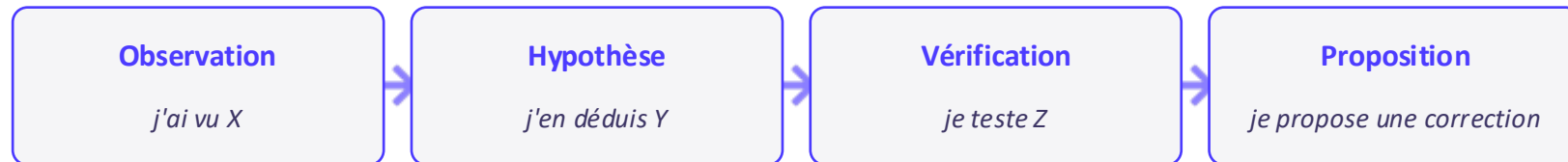
Filtrage des fausses alertes

Quand on alerte un humain, c'est pour une raison qui vaut son temps. Pas pour du bruit.

Correction assistée par IA

L'IA propose. L'humain décide.

Chain of Thought appliquée au diagnostic :



Validation humaine systématique

Chaque proposition est documentée et validée en un clic. Pas de dérive silencieuse.

DÉMO LIVE

Voyons ça dans Mr Suricate.

Démonstration en conditions réelles

Comment on intègre ça *chez Mr Suricate.*



Continuité du no-code

La complexité IA reste invisible pour l'utilisateur. Pas de nouveau langage à apprendre.



Validation humaine intégrée au workflow

Pas un module à part. Le QA garde le contrôle sans changer ses habitudes.



Construit avec nos clients

On ne sort pas une feature en silence. On co-construit avec les équipes terrain.

Le modèle change.

Notre mémoire applicative reste.



Trois idées à emporter

La ressource rare, ce n'est plus le test. C'est la confiance dans ce qu'il vérifie.

1. L'IA déplace le coût, elle ne le supprime pas. Chaque gain de vitesse se paie en vérification.

DORA 2026 : taux d'échec de 5 à 6 %, gains de 35 à 40 % sur du neuf contre 10 % ou moins sur du legacy

2. Réparer sans comprendre, c'est fabriquer du faux vert. Le test redevient vert sans rien vérifier.

22 000 variantes de programmes, huit modèles de langage, plus de 99 % des cas

3. La couverture ne dit rien de ce qu'on vérifie. Ce qui compte, c'est l'oracle.

69 % de bugs détectés par des tests IA contre 17 %, à couverture équivalente

Une dernière question :

Seriez-vous intéressé(e) par une démo plus approfondie de Mr Suricate ?



Levez la main, on compte →

À VOUS LA PAROLE

Des questions ?



Merci !

Et à très vite sur la piste.

Pour aller plus loin

- **Slides disponibles** sur demande, par email : suricate@mrsuricate.com
- **Blog Mr Suricate** mrsuricate.com/blog
- **Restons en contact** www.mrsuricate.com
suricate@mrsuricate.com