

Déploiement continu, l'agilité maximisée?

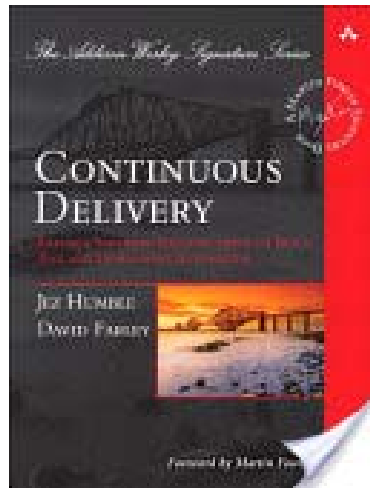
Agile Tour Montréal 2016



- **Pourquoi je vous parle de ce sujet?**
- Qu'est-ce que c'est?
- Est-ce vraiment utile?
- Et les enjeux?
- Comment s'y rendre?
- Et puis...

Qu'est-ce que c'est?

- Approche popularisée depuis 2010
 - Initié par le rappel de

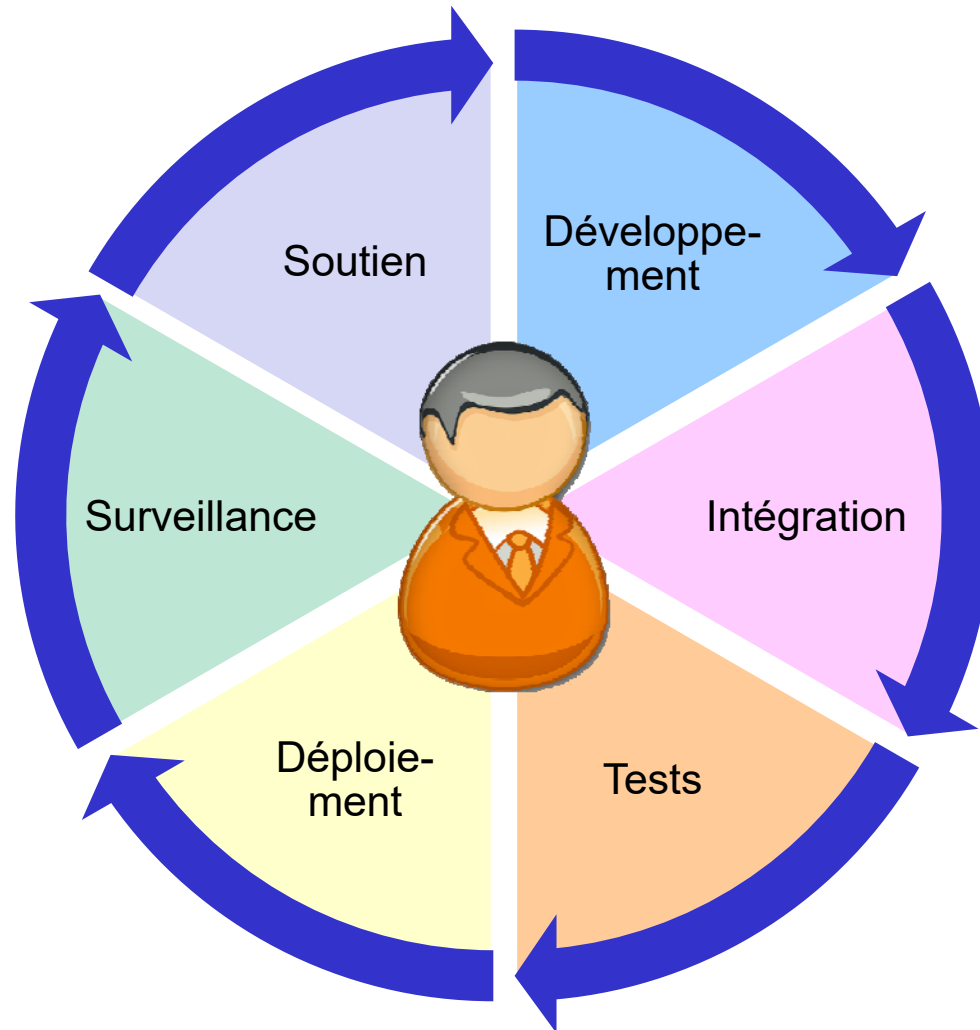


Agile Principle #1

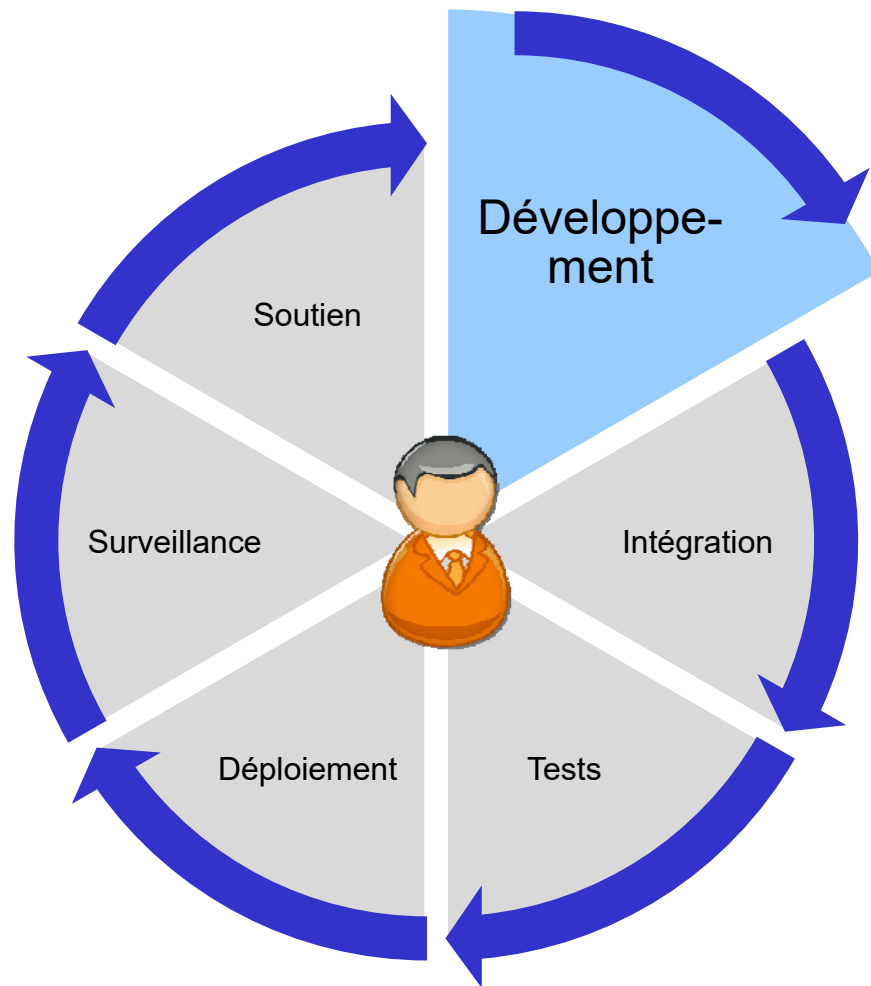
Our highest priority is to satisfy the customer through early and **continuous delivery** of valuable software.

- Accélération de bout en bout de la chaîne de développement
 - + De l'utilisateur au développeur à l'utilisateur

Qu'est-ce que c'est?

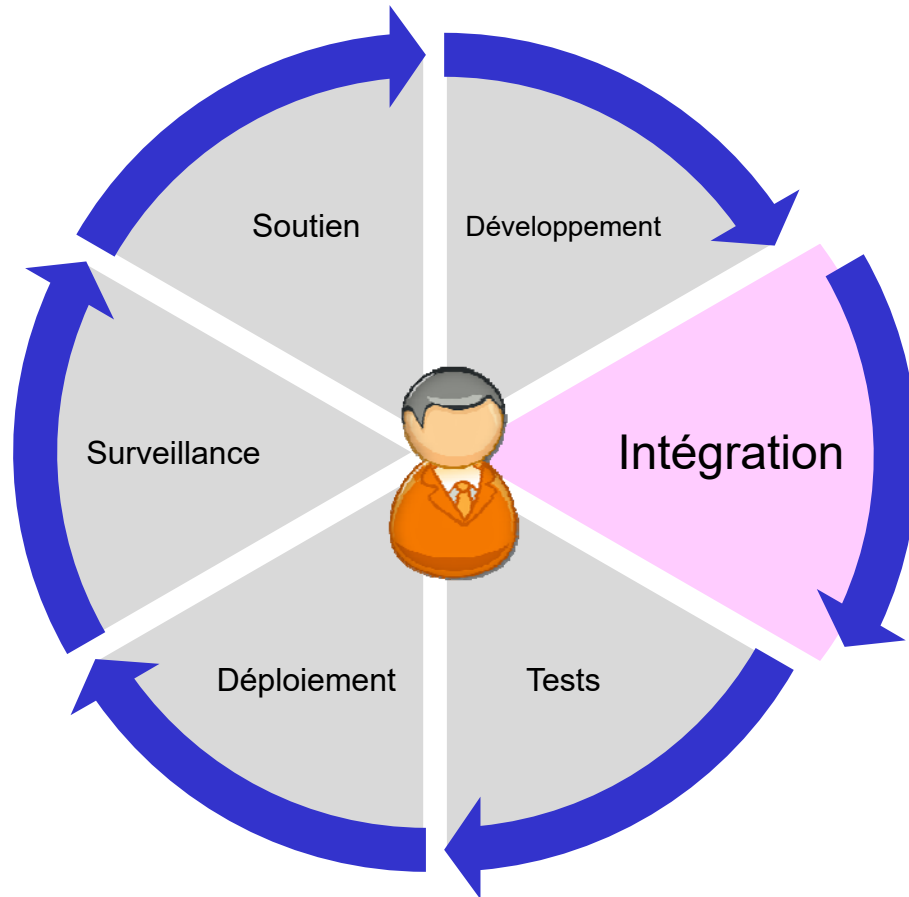


Qu'est-ce que c'est?



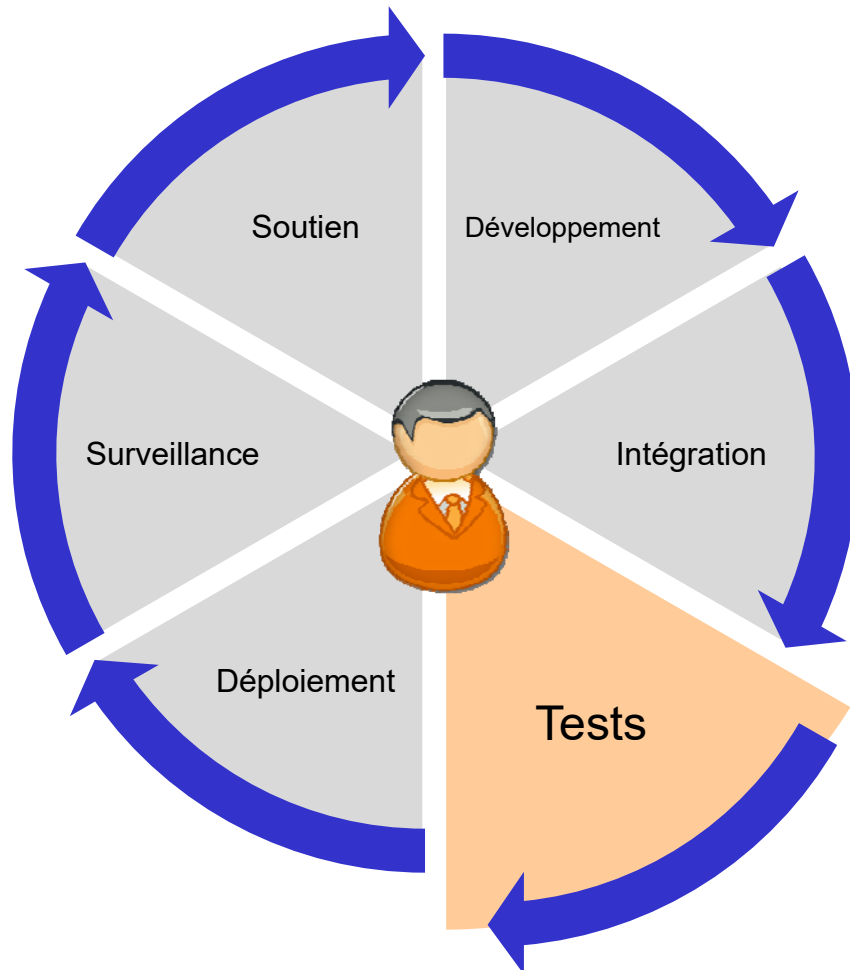
- Analyse
- Architecture
- Cas de tests
- Cueillette des besoins
- Documentation
- Priorisation
- Programmation
- Tests unitaires

Qu'est-ce que c'est?



- Assemblage
- Compilation
- Édition de liens
- Gestion des versions
- Modularisation
- Tests d'intégration

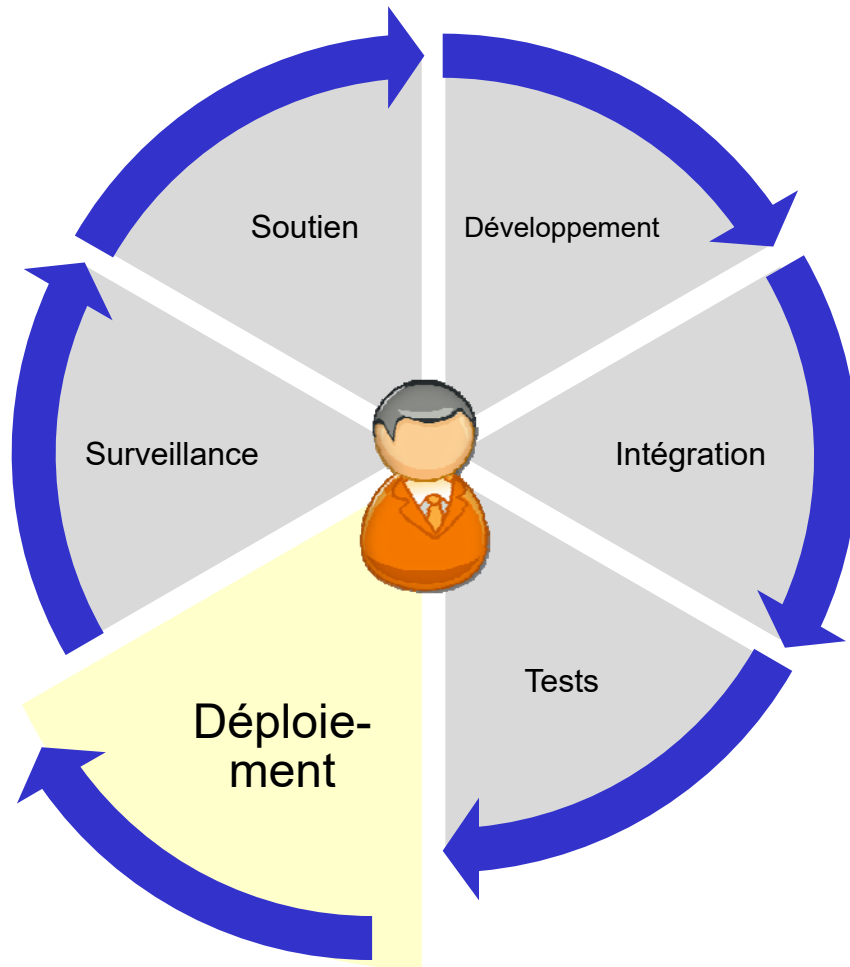
Qu'est-ce que c'est?



- Tests automatisés

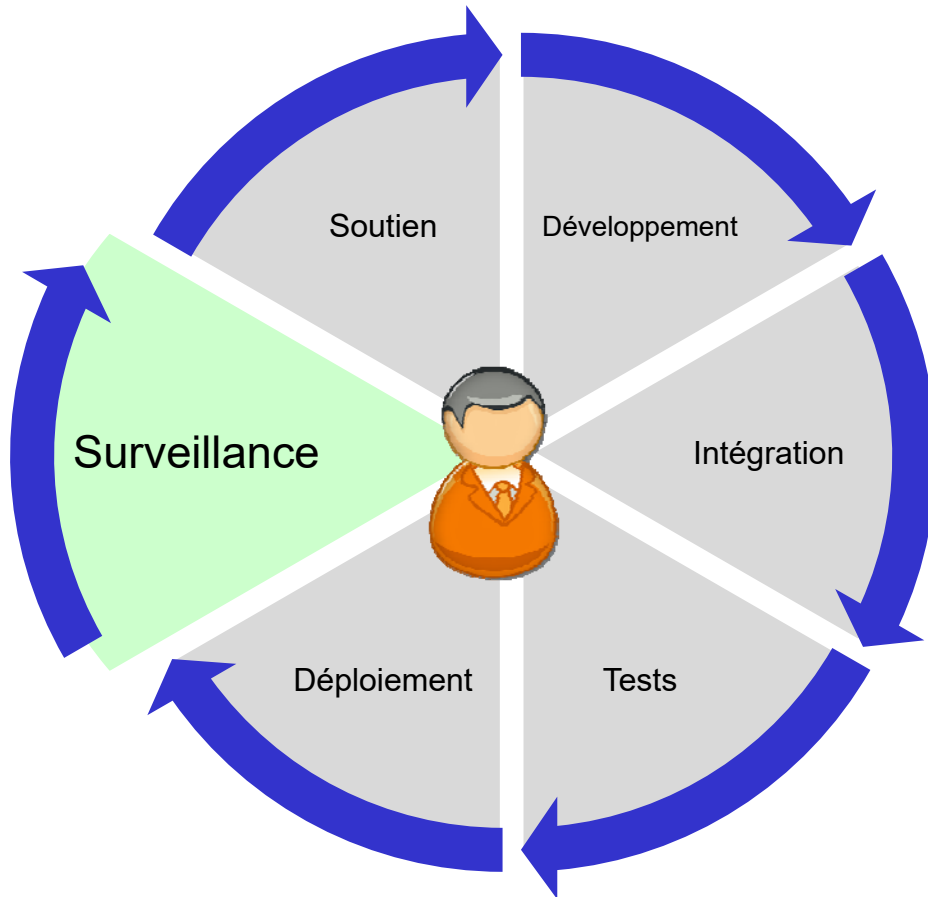
- Acceptation
- Capacité
- Fonctionnels
- Performance
- Régression
- Système
- Vulnérabilité

Qu'est-ce que c'est?



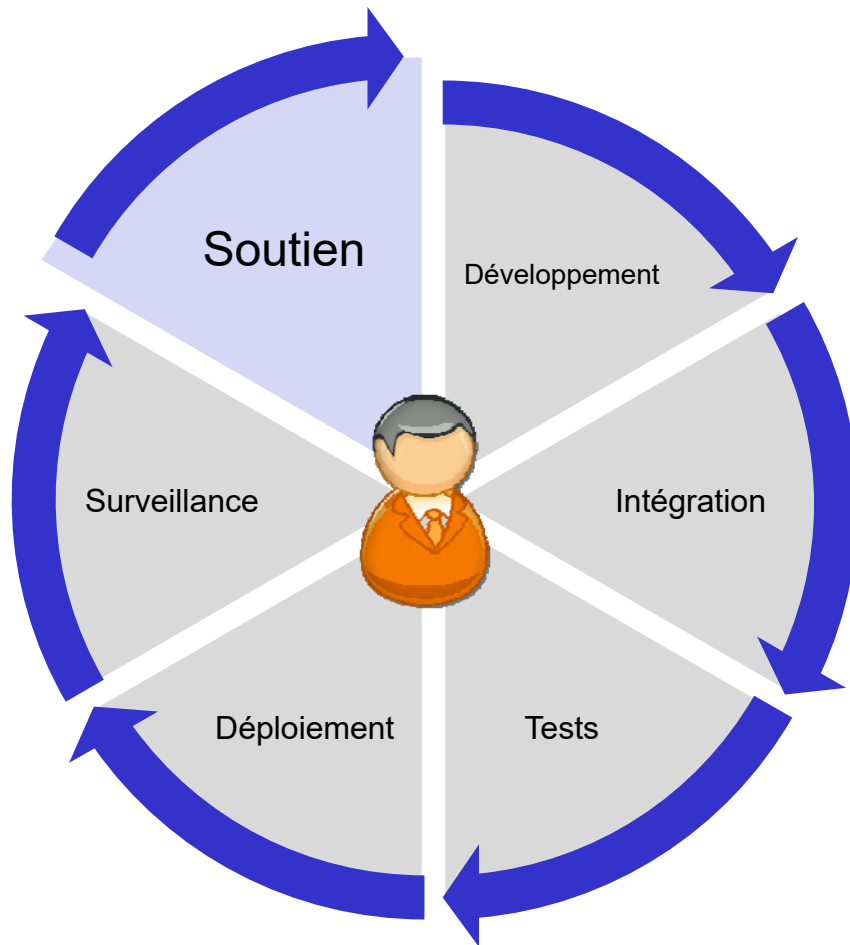
- Gestion de la configuration
- Retour arrière ou de secours
- Versions multiples
 - Canary
 - Dark

Qu'est-ce que c'est?



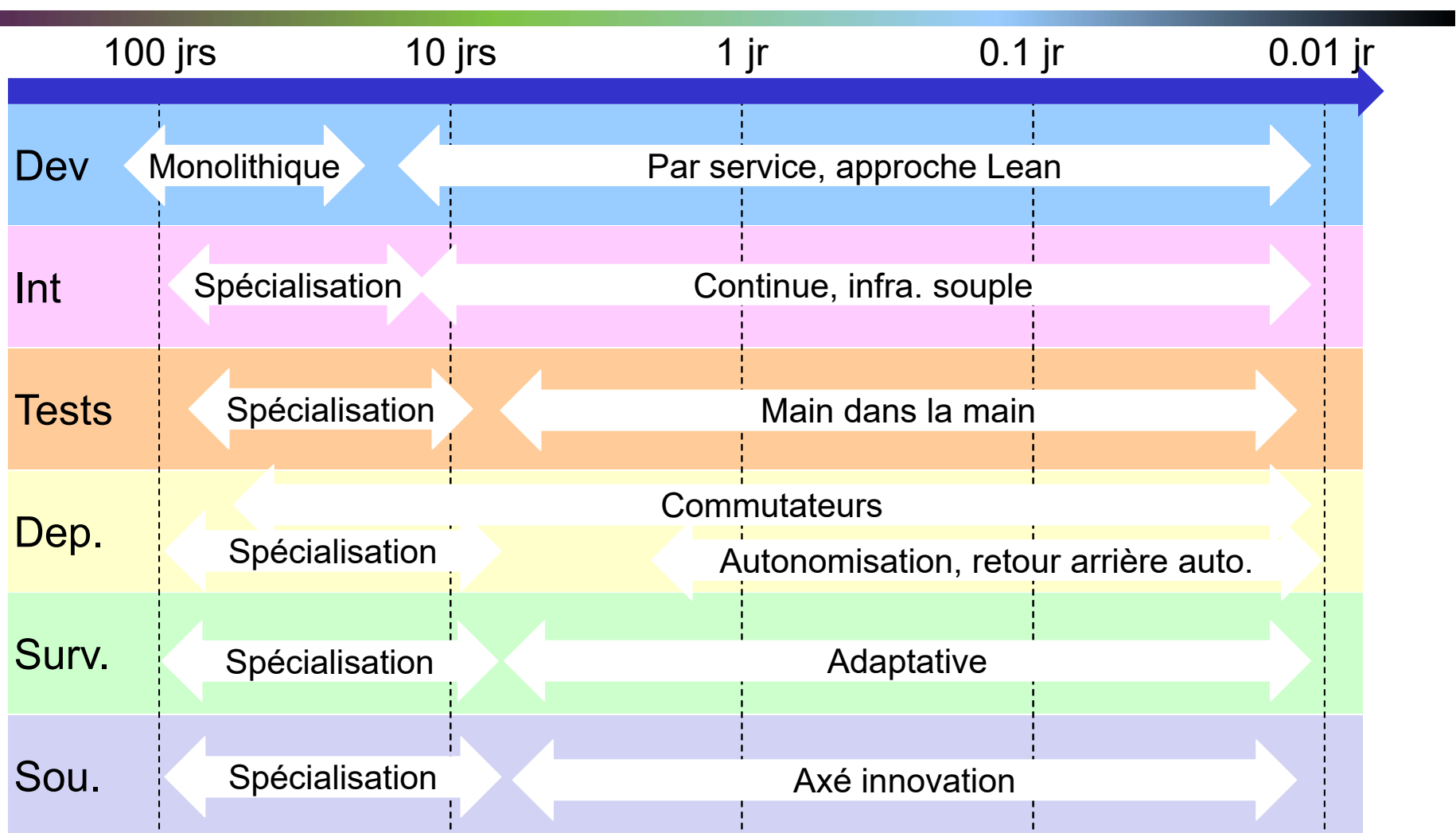
- Gestion de la capacité
- Surveillance
 - Capacité, erreurs, Fraude, intrusion, journaux, performance, plantage, ...
 - Utilisation
- Tests A/B

Qu'est-ce que c'est?

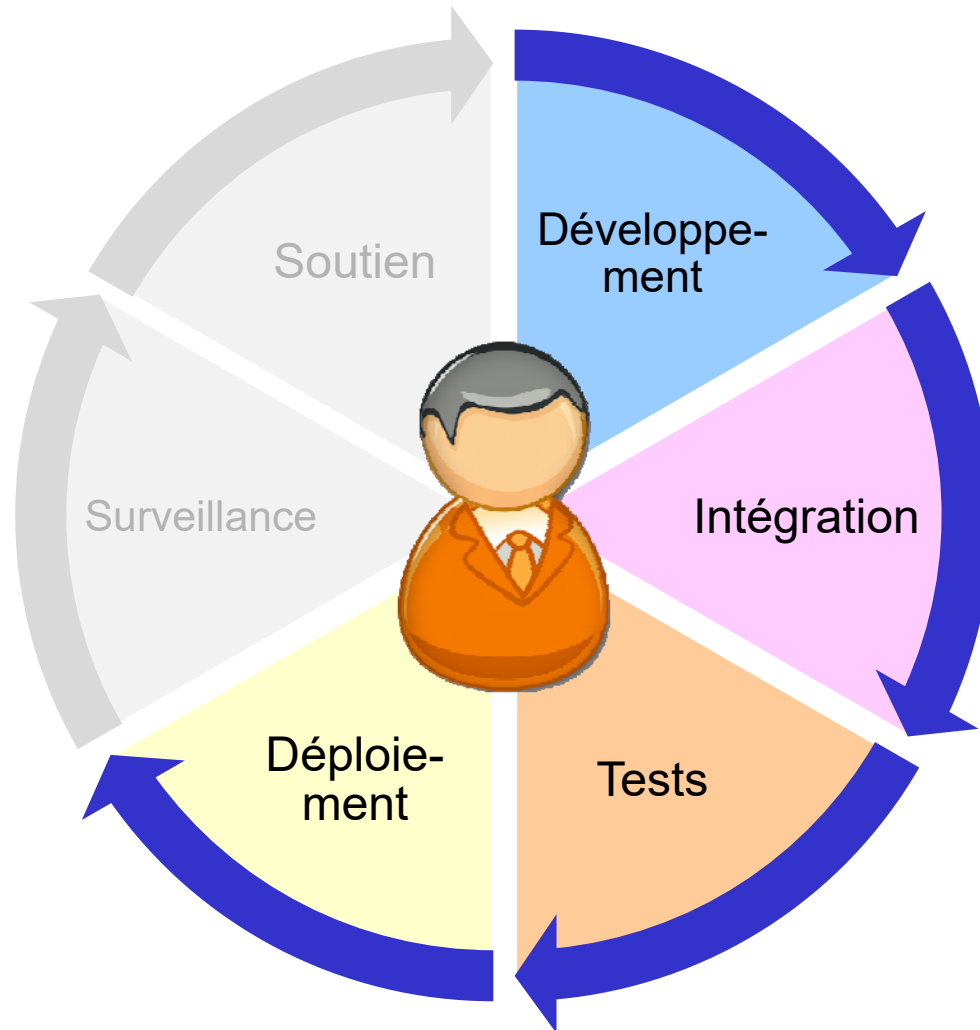


- Autonomisation de l'utilisateur
- Contact omnicanal
- Visibilité des changements

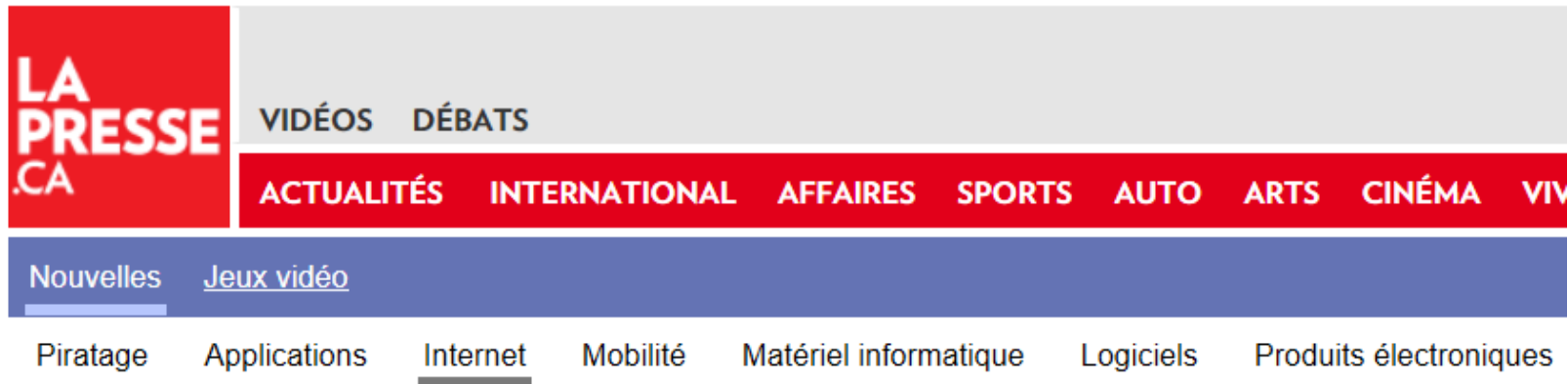
Qu'est-ce que c'est?



Qu'est-ce que c'est?



Est-ce vraiment utile?



[Accueil](#) > [Techno](#) > [Nouvelles](#) > [Internet](#) > Bogue morbide: Facebook s'excuse

Publié le 13 novembre 2016 à 08h42 | Mis à jour à 08h42

Bogue morbide: Facebook s'excuse



Est-ce vraiment utile?

- Marketing accéléré
 - plus facile de comprendre ce que les clients recherchent
- Diminuer le temps de cycle
 - mise en marché accélérée
- Produit plus pertinent
 - rétroaction rapide des utilisateurs
- Productivité accrue
 - automatisation

Est-ce vraiment utile?

- Diminution des risques d'une livraison
 - petits incréments
- Impacts d'une erreur limités
 - erreur rapidement identifiée
- Qualité de l'architecture qui peut se dégrader
 - à surveiller
- Proactivité accrue face aux vulnérabilités

Et les enjeux?

- Certains clients rébarbatifs
 - c'est pas pour tous
- Investissements massifs
 - automatiser c'est pas toujours simple
- Rétrocompatibilité avec logiciels tiers et migration des données
 - architecture à revoir

Et les enjeux?

- Découvertes des nouvelles fonctionnalités par les utilisateurs
 - comment bien le faire?
- Interruption des opérations
 - infrastructure à revoir
- Réglementaires
 - conformité à démontrer

Et les enjeux?

- Automatisation difficile
 - par exemple les tests d'acceptation
- Suivi de la documentation
 - défi de synchronisation et de pertinence
- Arrimage des environnements
- Interventions humaines
 - cela ralentit la cadence

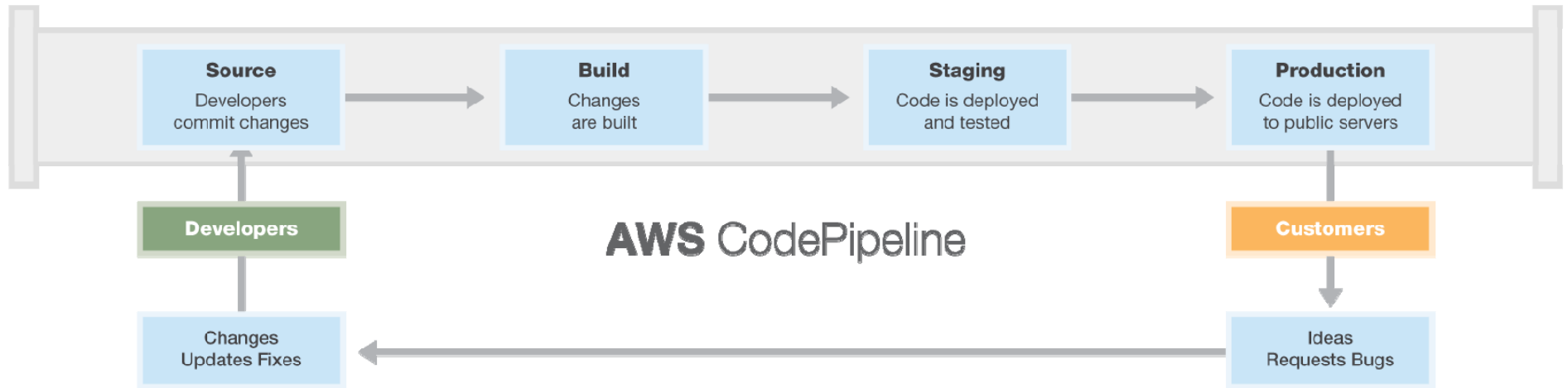
Comment s'y rendre?

- **Gouvernance**
 - Culture
 - + Gestion de changement
 - + Impliquer la haute direction
 - Processus
 - + Accélérer
 - + Automatiser
 - + Simplifier
 - Structure
 - + Aplanir
 - + Client-collaborateur

Comment s'y rendre?

- Les utilisateurs
 - Informer
 - + Blogues, Sondages, Wikis, ...
 - Impliquer
 - + Ciblage
 - + Production participative
 - Former
 - + Webinaire
 - Liens avec développeur?

Comment s'y rendre?



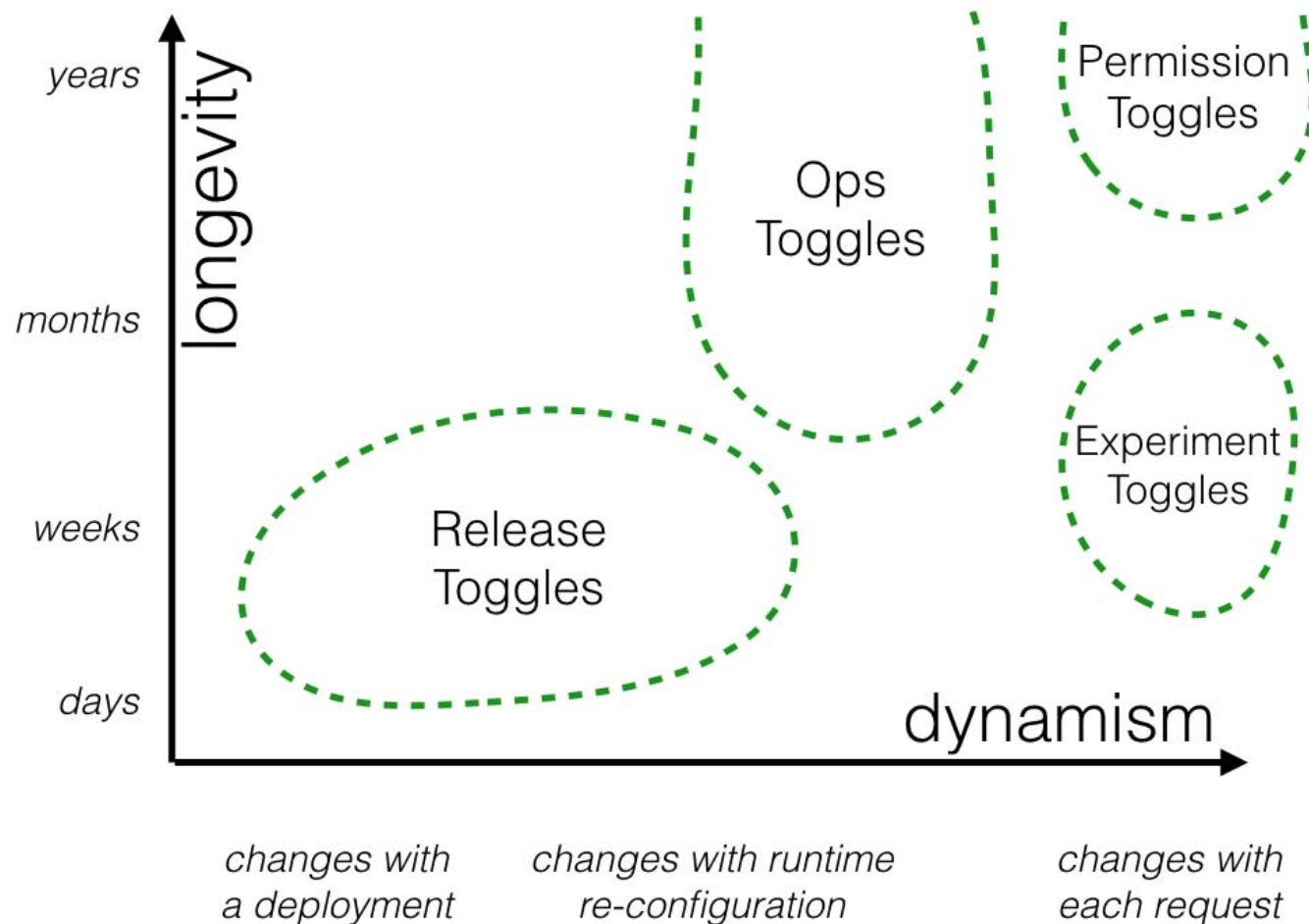
vmware
VMware vCloud® Air™

[Service Offering](#) [Explore vCloud Air](#) [Using vCloud Air](#) [Community](#) [Partner Programs](#)

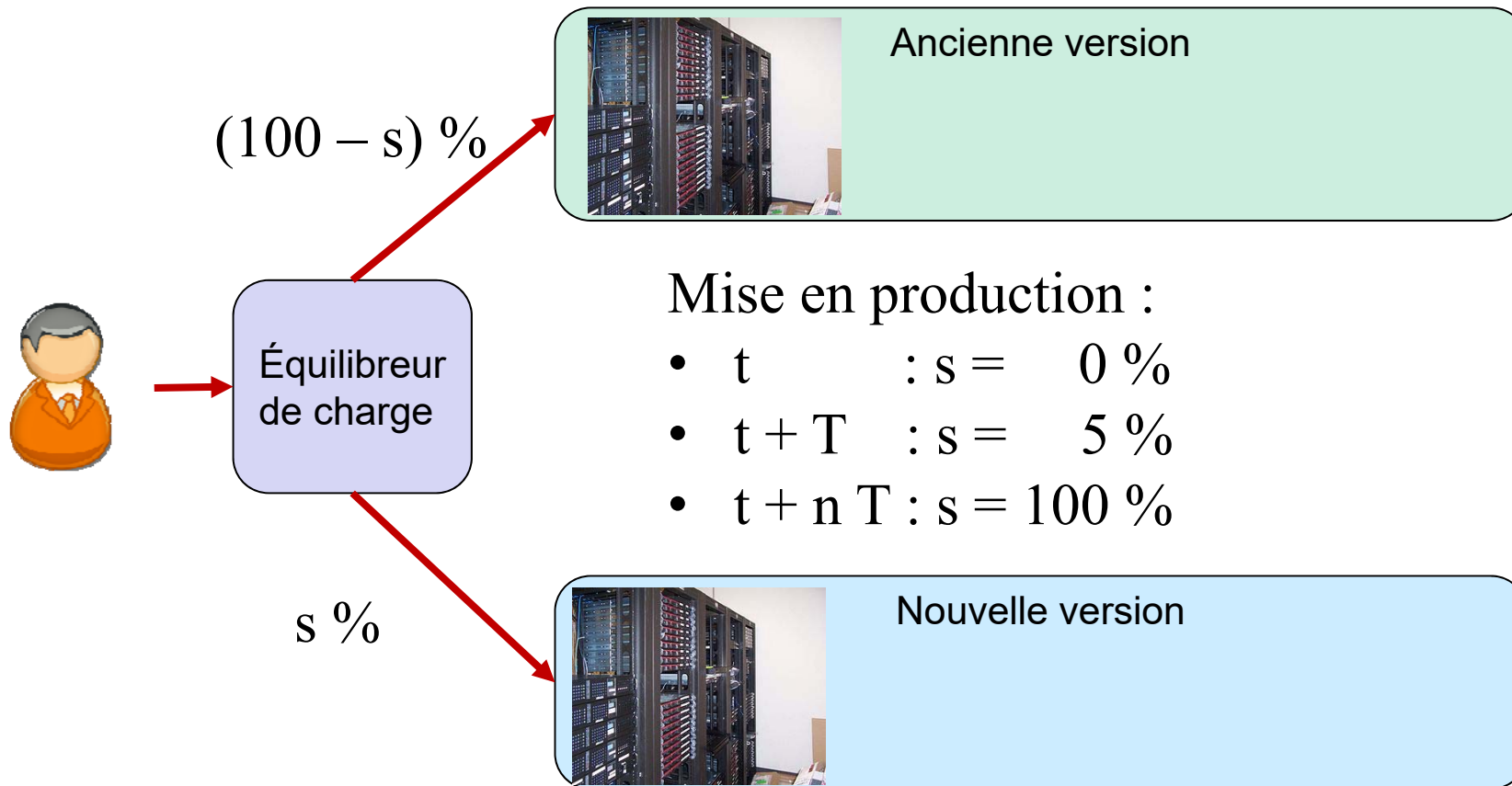


Comment s'y rendre?

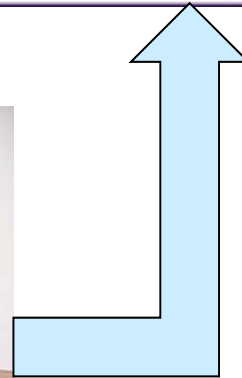
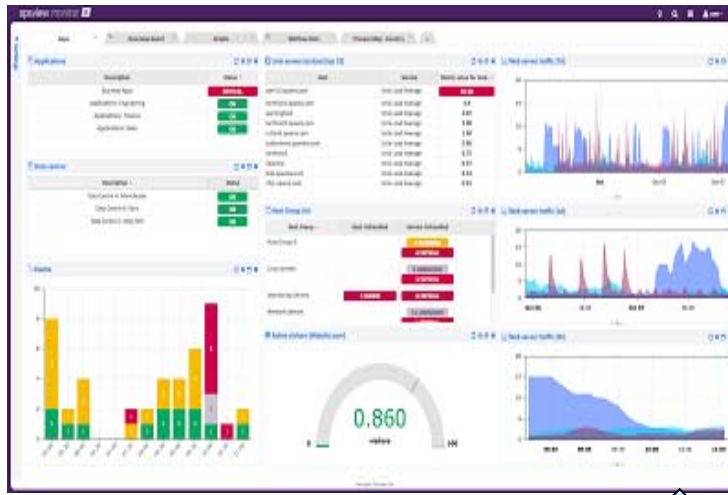
SOURCE: martinfowler.com/articles/feature-toggles.html



Comment s'y rendre?



Comment s'y rendre?



- Bien surveiller le nombre de
 - Bogues (+sévérité)
 - Nombre de dév.
 - Nombre de LoC par dév.
 - Pannes
 - Problèmes rapportés par les clients
 - Tâches de correction
 - Utilisateurs

Comment s'y rendre?

- Applications mobiles
 - Retour arrière et correctifs d'urgence difficiles
 - Stabiliser (5 jours) et figer (3 jours) le code
 - Suivre la cadence du propriétaire de la plateforme (1 semaine Android, 2 semaines iOS)
 - + Bien choisir la journée de livraison, car la qualité diminue le jour de la livraison. FB suggère le dimanche
 - 1 développeur = 1 fichier de code
 - + Le nombre de bogues varie selon le nombre de personnes qui changent un même fichier par livraison

Comment s'y rendre?

- Utilisateur choisit quand il change de version
- Utiliser les commutateurs (*feature toggle*)
- Versions multiples
 - + Appareil
 - + Système d'exploitation (type et version)
 - + Version de l'application

- La fréquence de livraison
 - N'affecte pas le nombre d'erreurs critiques
 - Affecte peu le nombre d'erreurs de moindre priorité
- Le déploiement continu
 - N'affecte pas la productivité des développeurs

Mais surtout...

- N'oublions pas notre client-collaborateur !
 - Appel à tous sur comment, de façon continue, peut-on optimiser:
 - + l'identification des besoins d'une masse d'utilisateurs?
 - + la découverte de nouvelles fonctionnalités?
 - + le transfert de connaissances à nos clients?



Références intéressantes

- + Leppänen, M. et al., The Highways and Country Roads to Continuous Deployment dans IEEE Software, Mars/avril 2016.
- + Rossi, C. et al., Continuous Deployment of Mobile Software at Facebook dans FSE 2016, Novembre 2016
- + Rodríguez, P. et al., Continuous Deployment of Software Intensive Products and Services: A Systematic Mapping Study, Journal of Systems and Software, Janvier 2016
- + Ur Raman, A. A. et al., Synthesizing Continuous Deployment Practices Used in Software Development dans Agile Conference 2015



VisiSYST

CONSEIL EN GESTION ET TECHNOLOGIE DE L'INFORMATION

Dr. Pierre-Martin Tardif, ing., MBA, PMP, CGEIT

pmtardif@visisyst.com

tardifpm



Agile Tour Montréal 2016

