



DevOps Institute

# DevOps Foundation®

01mars2021

## Dites-nous qui vous êtes...

Dites-nous en quelques mots qui vous êtes:

- Nom, organisation et rôle
- Expérience DevOps / Agile / Lean / ITSM
- Pourquoi vous suivez cette formation
- Ce que vous souhaitez apprendre



Quelle est votre définition ou votre perception de DevOps?

# Objectifs de la formation de DevOps Foundation

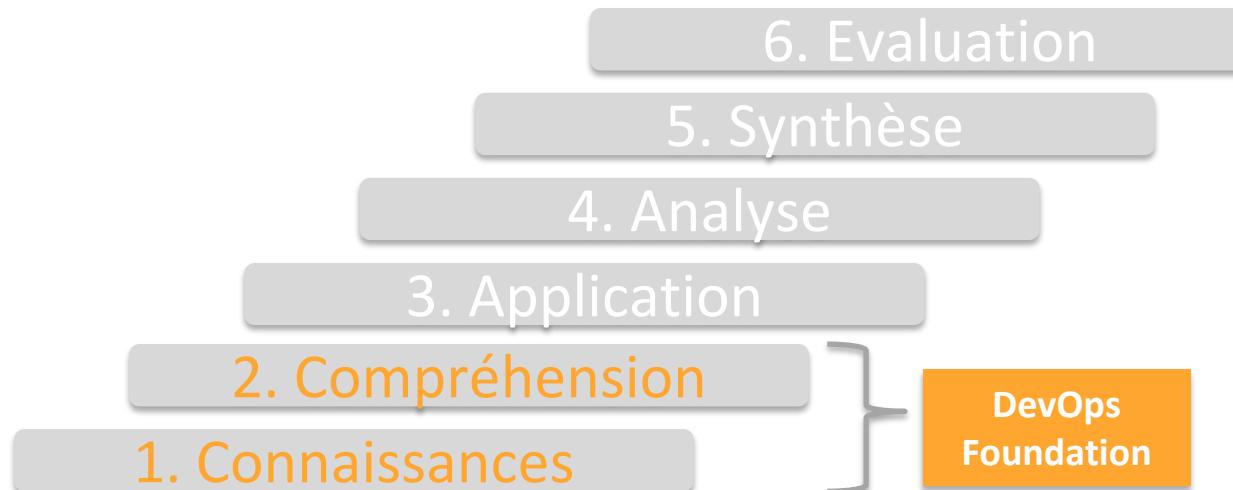
- En savoir plus sur DevOps
- Comprendre son vocabulaire de base, ses principes, ses pratiques et son automatisation
- Entendre et partager des scénarios réels
- S'amuser !



L'examen DevOps Foundation :

- 40 questions à choix multiples ;
- 60 minutes ;
- Obtention à partir de 65% ;
- Accrédité par DevOps Institute ;
- Obtention de votre badge numérique.

# La taxonomie de Bloom



La taxonomie de Bloom est utilisée pour la classification des objectifs pédagogiques et, à partir de là, l'évaluation des acquis.

## Le DevOps Institute



Le DevOps Institute est la communauté d'apprentissage continu pour les pratiques émergentes DevOps.

La vision du DevOps Institute est de créer une communauté où apprendre est un mode de vie et où les membres ont accès au contenu, aux formations et aux certifications les plus novateurs, inspirants et transformateurs de DevOps. Nous nous efforçons de fournir un contenu qui inspire à la discussion, la collaboration et la transformation.

# Contenu de la formation DevOps Foundation

Jour 1		Jour 2	
Bonjour et bienvenue		Jeu d'échauffement	
Module 1	Explorer DevOps	Module 5	<b>Culture</b> , comportements et modèles opérationnels
Module 2	Les principes fondamentaux de DevOps	Module 6	<b>Automatisation</b> et architecture des toolchains DevOps
Module 3	Les principales pratiques de DevOps	Module 7	<b>Mesures</b> , indicateurs et reporting
Module 4	Frameworks business et technologique	Module 8	<b>Partage</b> , observation et évolution
Révision examen blanc		Examen	

# MODULE 1

# Explorer DevOps

# Module 1 : Explorer DevOps

- Définir DevOps
- Pourquoi DevOps est-il important ?
- Du point de vue Business
- Du point vue IT

Composant	Module 1 - Contenu
Vidéo	Un bref historique de DevOps avec Damon Edwards
Etude de cas	Banque ING, Pays-Bas
Discussion	DevOps : mythes et réalités
Exercice	Votre “Pourquoi” organisationnel

# Définir DevOps

## Le corpus de connaissances collectif DevOps

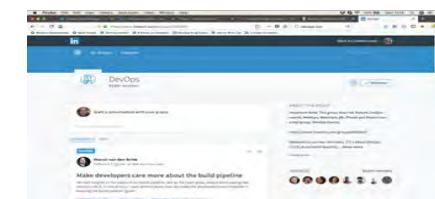
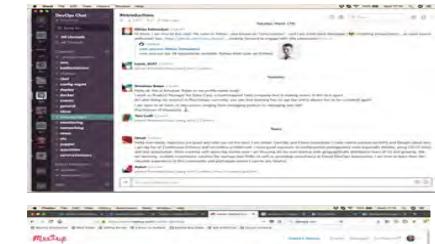
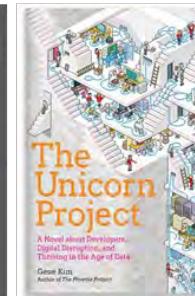
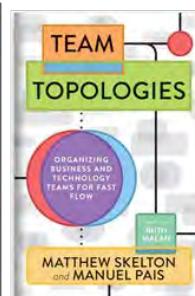
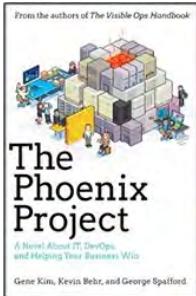
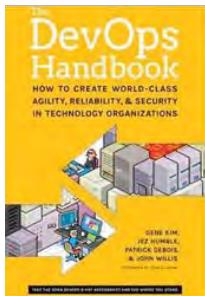
Fidèle à ses valeurs fondamentales, DevOps émerge à travers un ensemble de connaissances partagées et collectives (CBok) comprenant :

- Publications ;
- Conférences ;
- Groupes MeetUp ;
- Canaux Slack ;
- Groupes LinkedIn ;
- Vidéos et webinaires ;
- Blogs et articles ;
- Etudes de cas ;
- Prix et récompenses ;
- Expertises spécialisées.

DevOps Institute recherche et influence activement les pratiques émergentes de DevOps afin de créer des formations et certifications importantes.

Module 1 : Explorer DevOps

## Le corpus de connaissances collectif DevOps (2)



DevOps Institute © Tous droits réservés

Module 1 : Explorer DevOps

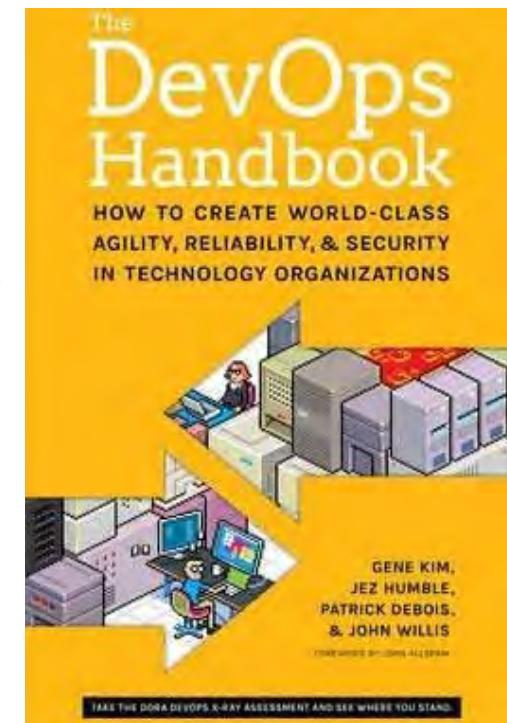


The Short History of DevOps  
with Damon Edwards (11:47)



## Qu'est-ce que DevOps ?

« Imaginez un monde où les product owners, développeurs, responsables qualité, opérationnels IT et responsables sécurité travaillent ensemble, non seulement pour s'entraider, mais également pour garantir le succès de l'ensemble de l'organisation. En travaillant ensemble vers un objectif commun, ils permettent la mise en production rapide du travail planifié, tout en assurant une stabilité, fiabilité, disponibilité et sécurité de premier ordre. »



« En un sens, DevOps consiste à mettre en place une chaîne de livraison de valeur - un pipeline rationalisé sans gaspillage grâce auquel la valeur peut être livrée à l'entreprise avec une durée de cycle courte et prévisible. »

*Mark Schwartz  
'The Art of Business Value'*



## Ce que DevOps n'est pas

- Un titre
- Une équipe à part
- Un outil
- Seulement une culture
- Seulement une automatisation
- L'anarchie
- Une stratégie unique pour tous



DevOps prend vie grâce à de nouvelles pratiques qui apportent une vraie valeur à de vraies organisations.

<http://devops.com/2016/03/17/what-devops-is-not/>

## Pourquoi DevOps est-il important maintenant ?

- Les entreprises se retrouvent concurrencées par de petites start-up agiles
- Le développement logiciel agile et l'infrastructure cloud sont en progression
- L'IT ne peut plus fonctionner dans une culture en silo
- De plus en plus d'organisations migrent vers le cloud
- Les consommateurs ont des habitudes et des attentes issues des apps (disponibles sur les mobiles)
- Les données disponibles pour l'entreprise sont de plus en plus nombreuses
- Le time to value doit être plus rapide

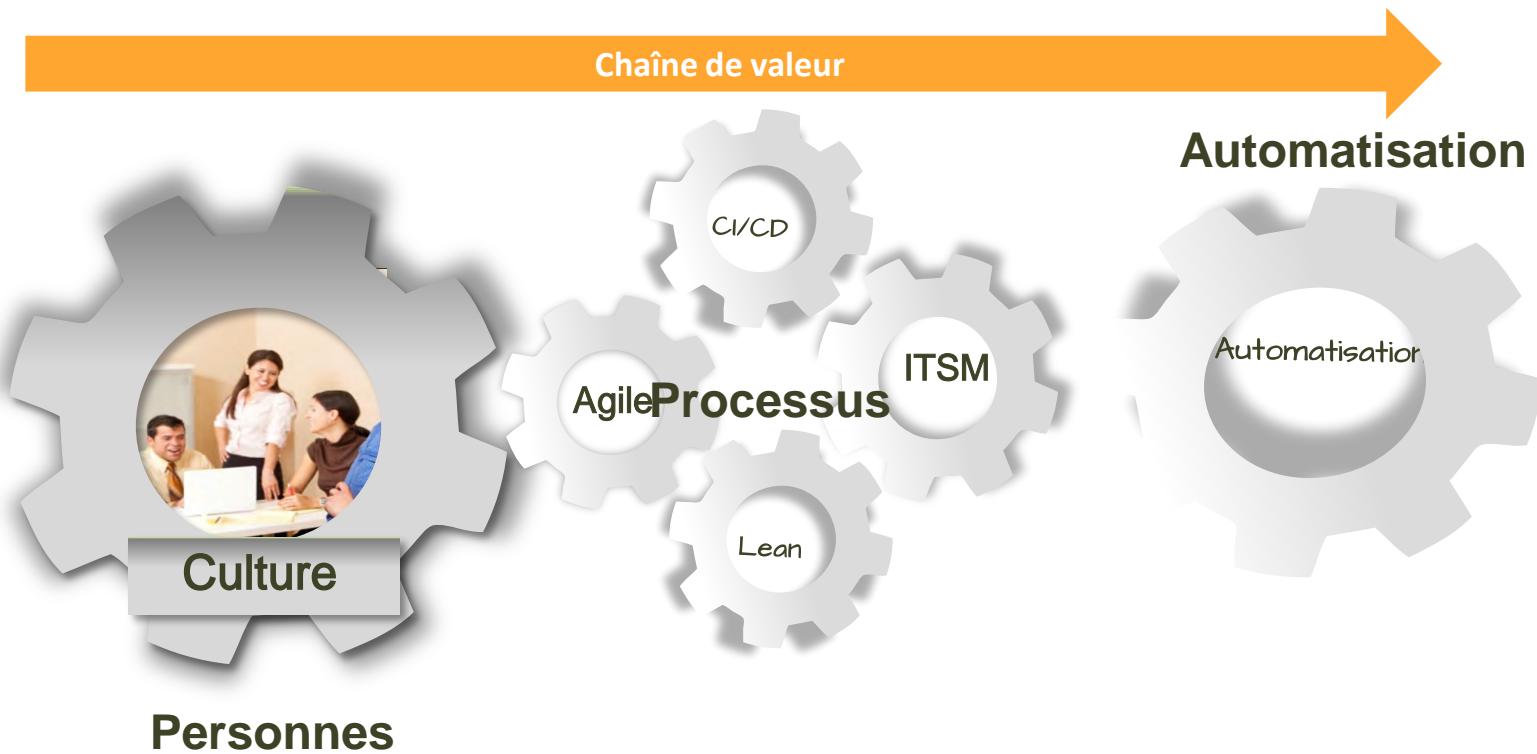
Pour faire face à ces conditions changeantes, l'IT doit adapter sa culture, ses pratiques et son automatisation pour être plus « C »

## Qu'est-ce qui rend DevOps si unique ?

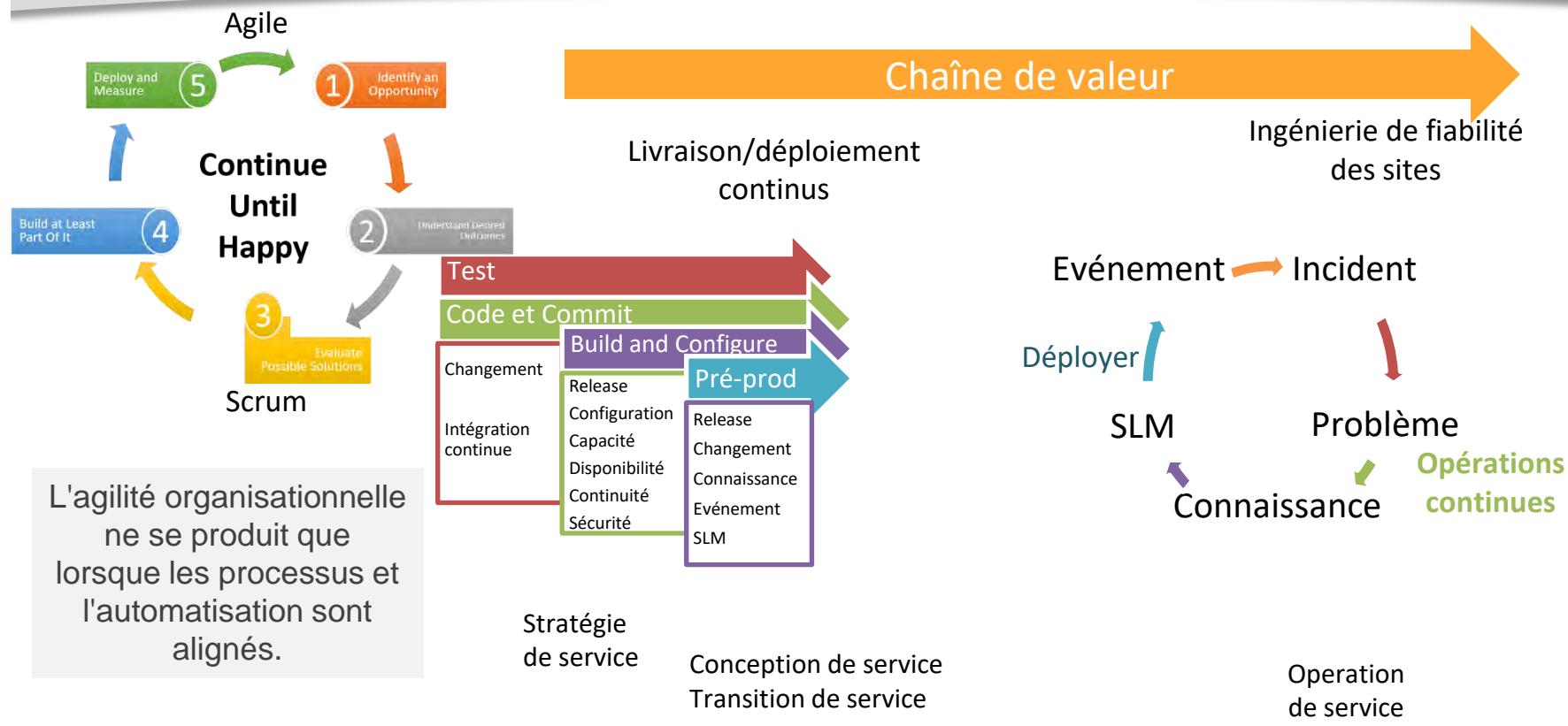
- Comment le comparer avec Scrum pour améliorer le flux de travail des développeurs ?
- Comment le comparer avec les pratiques de cybersécurité ?
- Comment le comparer au Lean pour augmenter l'efficacité de l'IT ?
- Comment le comparer avec ITIL® pour la gestion des services ?
- Comment le comparer avec les techniques de Gestion du changement organisationnel pour la culture ?
- Comment le comparer avec les outils, les technologies et l'automatisation ?

Chacun de ces frameworks et approches a permis d'apporter un certain avantage, mais aucun n'a apporté une amélioration informatique globale et complète.

# DevOps applique la pensée systémique à l'ensemble du spectre IT



# L'IT est un système de systèmes



ITIL® is a registered trade mark of AXELOS Limited.

# DISCUSSION

## DevOps : mythes et réalités

## Objectifs DevOps

- Des versions plus petites et fréquentes
- Réduction des efforts et des risques
- Réduction du coût des itérations et des retards
- Une culture de communication et de collaboration
- Cohérence et rapidité grâce à l'automatisation

### Gains dans :

- Time to market/value
- Intégration aux processus business
- Réactivité
- Qualité du code et du déploiement
- Productivité
- Visibilité
- Agilité

## Valeurs DevOps

DevOps est avant tout un mouvement culturel fondé sur des interactions humaines et techniques destinées à améliorer les relations et les résultats.



## L'automatisation est un élément essentiel

L'automatisation permet agilité, cohérence, rapidité et fiabilité.



La prise de décision partagée, l'accès aux toolchains et leur compréhension, ainsi que d'autres automatisations, simplifient la livraison de logiciels et préparent Ops au long terme.

## Parties prenantes DevOps



Dev inclut toutes les personnes impliquées dans le développement de produits et services logiciels, notamment :

- Architectes, représentants métiers, clients, chefs de produits, chefs de projets, responsables qualité, analystes, fournisseurs, etc.

Ops inclut toutes les personnes impliquées dans la fourniture et la gestion de produits et services logiciels, notamment :

- Professionnels de la sécurité de l'information, ingénieurs système, administrateurs système, ingénieurs d'exploitation, administrateurs de base de données, ingénieurs réseau, professionnels supports, fournisseurs, etc.



DevOps s'étend au-delà des développeurs de logiciels et de l'exploitation

« Vous ne changerez rien en luttant contre la réalité.  
Pour changer quelque chose, construisez un nouveau  
modèle qui rend le modèle existant obsolète. »

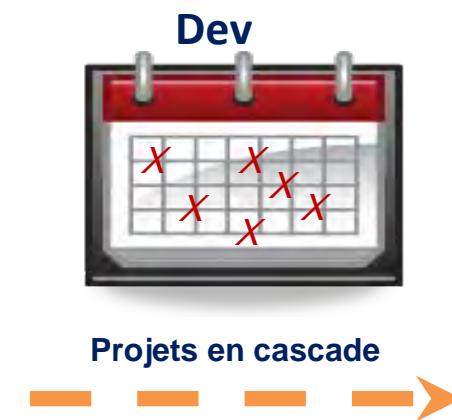
*Buckminster Fuller*



# Pourquoi DevOps est-il important ?

## Notre cadence n'est pas la bonne

Historiquement...



Cadence – le flux ou rythme des événements.

## DevOps améliore la cadence et la vitesse

... des pratiques agiles, Lean et ITSM sont également nécessaires.

L'entreprise



Gagner par  
l'innovation



DevOps Agile/Lean



Livraison continue



## DevOps améliore la rapidité ET la stabilité

Selon le rapport 2019 State of DevOps, les organisations les plus performantes ont :

- Des déploiements de code 208 fois plus fréquents
- Des délais 106 fois plus courts entre le « commit » et le déploiement
- Un délai de rétablissement suite à un incident 2604 fois plus rapide
- Un taux d'échec de changement 7 fois plus faible

"Nos recherches continuent de montrer que les quatre mesures clés du développement et de la fourniture de logiciels, qui sont la norme dans l'industrie, déterminent la performance organisationnelle dans les transformations technologiques. Le rapport de cette année réaffirme les conclusions précédentes selon lesquelles il est possible d'optimiser la stabilité sans sacrifier la vitesse".

« Le rapport Accelerate State of DevOps représente six années de recherche et de données provenant de plus de 31 000 professionnels du monde entier. Il s'agit de la recherche la plus vaste et la plus ancienne de ce type, qui offre un point de vue indépendant sur les pratiques et les capacités qui favorisent de hautes performances. Les résultats nous permettent de comprendre les pratiques qui conduisent à l'excellence dans la fourniture de technologies et à de forts résultats commerciaux. »



## ETUDE DE CAS : Banque ING

« Nous avons voulu créer une culture et un environnement dans lesquels la création, le test et la livraison de logiciels peuvent être réalisés rapidement, fréquemment et de manière plus fiable. Au début de cette périple, nous avons commencé par le plus important : les personnes. Ce périple, a eu un début mais il n'aura pas de fin. Une fin mettrait un terme à la transformation, alors qu'en réalité, vous devez toujours vous assurer que vous continuez à vous améliorer pour le client. »

« L'IT est devenue le cœur de la banque. »



Ron van Kemenade,  
CIO



### Avantages

- A transformé une organisation avec une aversion pour le risque en puissance Agile
- Time to market passé de 13 semaines à moins d'1 semaine
- Processus plus automatisés
- Une nette réduction des transferts
- Une culture de performance collaborative

## Adoption de DevOps



GitLab 2019  
Enquête auprès  
développeurs

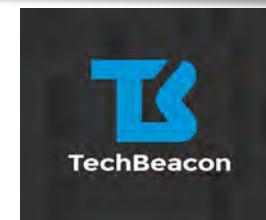
- Près de 60 % se déploient plusieurs fois par jour, une fois par jour ou une fois tous les quelques jours. En hausse par rapport à 45 % en 2019
- Plus de 25 % des entreprises se trouvent dans le "point chaud" du DevOps, c'est-à-dire entre trois et cinq ans de pratique
- 47% des entreprises déclarent que les essais sont la première raison des retards



puppet

Rapport sur  
l'état des lieux  
des  
opérations de  
développement en 2020

- Les entreprises très évoluées sont six fois plus susceptibles de signaler une forte utilisation des plateformes internes que les entreprises à faible niveau d'évolution des DevOps
- Les entreprises très évoluées ont près de trois fois plus de chances d'avoir une gestion du changement très efficace que les entreprises à faible niveau d'évolution des DevOps



21  
Prévisions  
pour 2021

- Les chefs d'entreprise apprécieront de plus en plus les DevOps
- Les équipes de produits hybrides deviendront un pilier de la fourniture de valeur au client
- La sécurité des applications ne sera plus une préoccupation secondaire
- Les DevOps prédictives seront la prochaine transformation qui apportera une valeur ajoutée aux entreprises

« Le véritable objectif (tout à fait réalisable) consiste à aligner les objectifs IT sur ceux du business en supprimant tous les goulots d'étranglement, inefficacités et risques entre une idée business et un résultat client mesurable.

»

**Damon Edwards**

# Du point de vue business

## Pourquoi le business est-il le moteur de DevOps ?

- Toutes les entreprises sont maintenant des entreprises technologiques
- L'IoT progresse rapidement
- Les consommateurs ont développé des habitudes « apps sur mobile »
- Les clients valorisent les résultats, pas les produits
- Le time to value remplace le time to market
- Les données intelligentes doivent déterminer rapidement la direction
- Le plaisir du client est plus important que la satisfaction du client

Votre plus gros concurrent peut être une start-up.

POWER O  
POWER O  
EMAIL  
INTERNET  
WORLD  
NO  
SEARCH

## La valeur business de DevOps

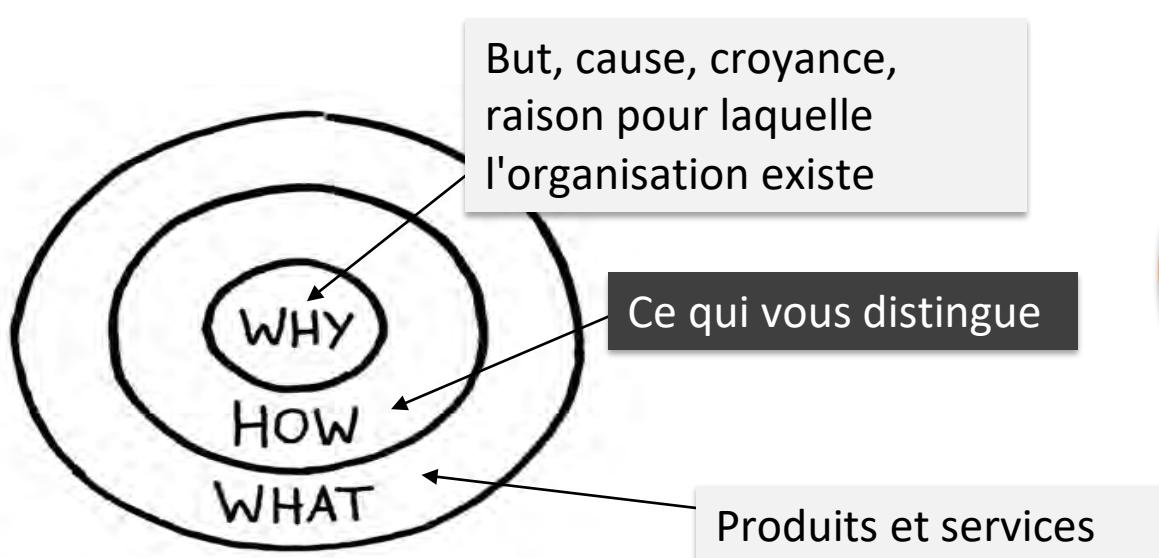
Les objectifs conjugués, commerciaux et non commerciaux, incluent :

- Rentabilité
- Productivité
- Part de marché
- Nombre de clients
- Quantité de produits ou de services
- Efficacité opérationnelle
- Satisfaction client
- Qualité des produits ou services fournis
- Atteinte des objectifs de l'organisation ou de la mission



"La livraison rapide, fiable et sûre de logiciels est au cœur de la transformation technologique et de la performance organisationnelle. Nous constatons que la vitesse, la stabilité et la disponibilité des logiciels contribuent à la performance organisationnelle (notamment la rentabilité, la productivité et la satisfaction des clients). Nos plus performants ont deux fois plus de chances d'atteindre ou de dépasser leurs objectifs de performance organisationnelle".

## Commencez avec “Why” – Le cercle d’or



© 2013 Simon Sinek,  
Inc.

Quel est le  
« pourquoi »  
de votre  
entreprise  
?

Module 1 : Explorer DevOps



Start With Why  
with Simon Sinek (5:00)



# EXERCICE

## Votre “Pourquoi” organisationnel

« L'Agile a permis aux équipes de développement de regagner la confiance du business, mais en laissant involontairement les opérations IT derrière. DevOps est un moyen pour restaurer la confiance de l'entreprise vis-à-vis de toute l'organisation IT. »

**Clyde Logue - Fondateur de StreamStep**



# Du point de vue IT

## Pourquoi l'IT est-il le moteur de DevOps ?

- Toutes les entreprises sont maintenant des entreprises technologiques
- L'IoT augmente rapidement
- Les consommateurs ont développé des habitudes « apps sur mobile »
- Les clients valorisent les résultats, pas les produits
- Le time to value remplace le time to market
- Les données intelligentes doivent déterminer rapidement la direction
- Le plaisir du client est plus important que la satisfaction du client

Reconnaissez-  
vous ces  
moteurs ?

POWER O  
POWER O  
EMAIL  
INTERNET  
WORLD  
NO  
SEARCH

## Le défi IT

DevOps doit continuellement produire des résultats en établissant des ponts et en améliorant presque tous les aspects de l'IT.

### Défis IT internes :

- L'IT doit aller de plus en plus vite sans impact sur la qualité
- Les investissements antérieurs ne fournissent pas une valeur de bout en bout :
  - Le développement logiciel agile c'est bien mais ça n'apporte pas la pleine valeur ;
  - Les processus ITSM c'est bien, mais ça n'apporte pas la pleine valeur ;
  - Une nouvelle automatisation c'est bien, mais ça n'apporte pas la pleine valeur.
- La culture IT en silo limite la chaîne de valeur

L' IT n'a plus besoin de s'aligner ou de s'intégrer au business, elle est le business

## Le mur de la confusion (1)

Qu'en est-il de la sécurité, de la gouvernance, de la gestion des risques et de la conformité ? Que veulent-ils ?



Dev veut  
des  
change-  
ments



Ops veut  
de la  
stabilité



## Le mur de la confusion (2)

Accent sur le  
changement

Quels sont les dangers ?

Accent sur  
la stabilité



Que veut le business ?  
Les trois !

## La culture en silo de l'IT

Dev



Sécurité  
Assurance qualité

Ops



Service Desk

Des silos IT isolés peuvent favoriser les stéréotypes et idées fausses entre Dev, Ops et d'autres équipes IT.

Cela peut également affecter le flux de travail et la capacité de l'IT à innover en permanence.

## Module 1 : Quiz (1)

1. Que signifie CALMS ?
  - a) Culture, Automatisation, Lean, Management, Solidarité
  - b) Collaboration, Automatisation, Lean, Métriques, Solidarité
  - c) Culture, Automatisation, Lean, Mesure, Solidarité
  - d) Collaboration, Automatisation, Lean, Mesure, Solidarité
  
2. Qui a inventé le mot « DevOps » ?
  - a) Gene Kim
  - b) Patrick Debois
  - c) John Willis
  - d) Damon Edwards

## Module 1 : Quiz (2)

3. Qui est devenu le partenaire de DORA dans Accelerate State of DevOps Reports en 2018 ?
  - a) Puppet
  - b) Chef
  - c) Google
  - d) Amazon Web Services
4. En 2015, Gartner avait annoncé que d'ici 2016, le pourcentage sur une base de 2000 organisations dans lesquelles DevOps passerait de l'état de « niche » à l'état de statut de « stratégie principale », serait de :
  - a) 100%
  - b) 50%
  - c) 20%
  - d) 10%

## Module 1 : Quiz (3)

5. Qui est à l'origine du Cercle d'or ?

- a) Simon Sinek
- b) Alan Alter
- c) Ron van Kemenade
- d) Jez Humble

# Module 1 : Réponses au quiz

- 1 Que signifie CALMS ?  
a) Culture, Automatisation, Lean, Management, Solidarité  
b) Collaboration, Automatisation, Lean, Métriques, Solidarité  
**c) Culture, Automatisation, Lean, Mesure, Solidarité**  
d) Collaboration, Automatisation, Lean, Mesure, Solidarité
- 2 Qui a inventé le mot 'DevOps' ?  
a) Gene Kim  
**b) Patrick Debois**  
c) John Willis  
d) Damon Edwards
- 3 Qui est devenu le partenaire de DORA dans Accelerate State of DevOps Reports en 2018 ?  
a) Puppet  
b) Chef  
**c) Google**  
d) Amazon Web Services
- 4 En 2015, Gartner avait annoncé que d'ici 2016, le pourcentage sur une base de 2000 organisations dans lesquelles DevOps passeraient de l'état de « niche » à l'état de statut de « stratégie principale », serait de :  
a) 100%  
b) 50%  
**c) 25%**  
d) 10%
- 5 Qui est à l'origine du Cercle d'or ?  
**a) Simon Sinek**  
b) Alan Alter  
c) Ron van Kemenade  
d) Jez Humble

# MODULE 2

# Les Principes fondamentaux

# de

# DevOps

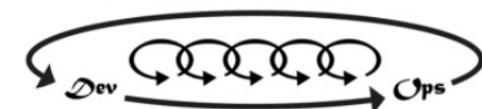
## Module 2 : Principes fondamentaux de DevOps

- Les Three ways
- Chaos Monkey
- Encourager une culture d'apprentissage

Composant	Contenu du Module 2
Vidéo	Gene Kim définit les trois voies du projet “Phoenix”
Etude de cas	Ticketmaster
Discussion	Surmonter les contraintes
Exercice	Faire vivre les Three ways

# Les Three ways (3 voies)

## Les Three ways



Première voie	Deuxième voie	Troisième voie
Flux	Feedback	Experimentation et apprentissage continu
Comprendre et améliorer le flux de travail (de gauche à droite)	Créer des boucles de feedback permettant une amélioration continue (de droite à gauche)	Créer une culture qui favorise : <ul style="list-style-type: none"> <li>L'expérimentation, la prise de risques et tirer des leçons des échecs</li> <li>Comprendre que la répétition et la pratique est la condition préalable à la maîtrise</li> </ul>

Module 2 : Les Principes fondamentaux de DevOps



Gene Kim Defines The Three Ways  
of The Phoenix Project (3:31)



## La première voie : le flux



- Comprendre le flux de travail
- Améliorer le flux en identifiant puis en supprimant les contraintes
- Ne jamais transmettre un défaut connu vers l'aval
- Ne jamais laisser une optimisation locale entraîner la dégradation globale
- Atteindre une véritable compréhension de l'ensemble du système

La première voie vise à fluidifier le travail afin qu'il s'écoule rapidement de gauche à droite.

## Théorie des contraintes

Une approche permettant d'identifier le plus important facteur limitant (c.à-d. la contrainte) qui retarde l'atteinte d'un objectif et de systématiquement optimiser cette contrainte jusqu'à ce qu'elle ne soit plus le facteur limitant.

La théorie des contraintes reconnaît que :

- Chaque processus a au moins une contrainte ou un goulot d'étranglement qui affecte sa capacité à atteindre son objectif.
- Le processus pourra au maximum atteindre la capacité de ses contraintes et pourra être efficace à la limite de son maillon le plus faible.
- L'optimisation des contraintes est le moyen le plus rapide et le plus efficace d'améliorer l'ensemble du processus ou du système

La théorie des contraintes a été introduite dans le livre  
'The Goal' par Eliyahu M. Goldratt.

## Contraintes courantes

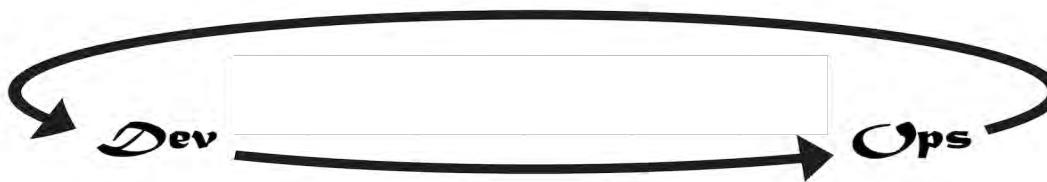
- Délais de développement
- Crédit d'environnement (test, simulation, production, etc.)
- Déploiement de code
- Mise en place et exécution des tests
- Évaluations de sécurité ou de QA
- Dépendance à l'architecture
- Gestion des produits
- Processus complexes ou bureaucratiques



# DISCUSSION

## Surmonter les contraintes

## Deuxième voie : Feedback



- Comprendre et répondre aux besoins de tous les clients - internes comme externes
- Raccourcir et amplifier toutes les boucles de feedback
- Créer et intégrer des connaissances là où elles sont nécessaires

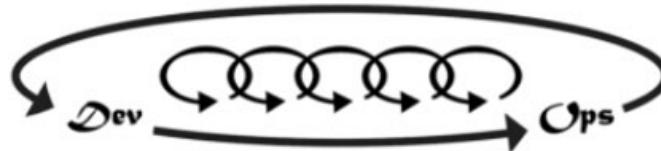
La deuxième voie vise à raccourcir et amplifier les boucles de feedback de droite à gauche afin que les corrections nécessaires puissent être apportées en permanence.

## Exemples de boucles de feedback



- Tests automatisés
- Revue par les pairs des changements en production
- Monitoring
- Tableaux de bord
- Logs de production
- Indicateurs de processus
- Post-mortem
- Partage d'informations entre les pilotes d'exploitation
- Données de changements, incidents, problèmes et gestion connaissances

## La troisième voie : Experimentation et apprentissage continu



La troisième voie encourage une culture qui favorise :

1. L'expérimentation continue, prendre des risques et apprendre des échecs
2. Comprendre que la répétition et la pratique sont les conditions préalables à la maîtrise

- Allouer du temps pour l'amélioration du travail quotidien
- Créer des rituels qui récompensent l'équipe de la prise de risques
- Introduire des défauts dans le système pour augmenter la résilience
- Prévoir du temps pour des expériences et des innovations en toute sécurité (hackathons)

## Chaos Engineering

- Le concept « Simian Army » a été adopté pour la première fois par Netflix comme un service mettant fin de manière aléatoire à une instance de production.
- La réponse aux attaques permet de créer des compétences pour restaurer l'environnement de production suite à des défaillances inévitables.



« Chaos Monkey est un robot qui désactive de manière aléatoire nos instances de production afin de nous assurer que nous puissions surmonter ce type de défaillance ordinaire sans aucun impact sur les clients. Le nom vient de l'idée de libérer un singe sauvage avec une arme dans votre centre de données (ou région de cloud) qui casserait les instances et rongerait des câbles au hasard - tout en continuant de servir nos clients sans interruption. En activant les Chaos Monkeys pendant les jours ouvrables dans un environnement monitoré avec soin par des ingénieurs prêts à résoudre tout problème, nous pouvons toujours tirer les leçons des faiblesses de notre système et mettre en place des mécanismes de reprise automatique.

Ainsi, la prochaine fois qu'une instance échouera à 3 heures du matin le dimanche, nous ne le remarquerons même pas. »  
Netflix

## ETUDE DE CAS : Ticketmaster

« Au fil des ans, nous avons extrait ces technologies traditionnelles en ajoutant des API pour moderniser l'interface de nos moteurs et plates-formes de billetterie. Nous voulions les mettre en production rapidement. Pour ce faire, nous devons toucher de nombreux systèmes. Cela nous a conduit à DevOps. Pour nous, tout a vraiment commencé avec DevOps. Une partie de notre transformation a été d'axer nos efforts sur la création plus rapide de la valeur ajoutée pour l'entreprise et la rapidité avec laquelle le produit était mis sur le marché. »



Justin Dean, VP  
TechOps



“L’accent est mis sur le résultat et non plus sur la quantité de travail réalisée”

### Avantages

- Suppression des goulots d'étranglement et des contraintes hérités
- Amélioration de la rapidité de commercialisation
- Axés sur les résultats et la valeur business
- Autorité distribuée, plus grande autonomie
- Friction réduite grâce au libre service

## Encourager une culture d'apprentissage

- Encourager l'apprentissage quotidien et le partage des connaissances
- Créer des plans de formation et d'éducation basés sur les compétences
- Incorporer l'apprentissage dans les processus
- Utiliser la technologie pour accélérer l'apprentissage
- Faire en sorte que le travail soit éducatif par le biais d'expérimentations, de résolutions de problèmes et de démonstrations
- Autoriser et utiliser des erreurs comme sources d'apprentissage
- Rendre les résultats de l'apprentissage visibles

« Soit vous êtes une organisation en apprentissage, soit vous perdez au profit d'une qui l'est. »

Andrew Shafer dans 'Beyond the Phoenix Project'

## EXERCICE

Faire vivre les Three ways

## Module 2 : Quiz (1)

1. Laquelle des Three ways concerne les boucles de feedback tels que les résultats des tests automatisés ?
  - a) La première voie
  - b) La deuxième voie
  - c) La troisième voie
  - d) Les Three ways
  
2. Comment les principes de DevOps encadrent-ils le traitement de l'échec ?
  - a) Comme une occasion d'apprentissage
  - b) Comme un élément à ignorer
  - c) Qu'il devrait être puni
  - d) Que nous ne pouvons pas nous protéger contre

## Module 2 : Quiz (2)

3. Quel est le préalable à la maîtrise ?

- a) Talent & capacité
- b) S'entraîner et étudier
- c) Orientation et coaching
- d) Répétition & entraînement

4. Nous devons :

- a) Ne jamais transmettre un défaut connu en aval
- b) Rechercher une vue d'ensemble du système
- c) Autoriser l'optimisation locale à dégrader l'optimisation globale
- d) Optimiser tous les liens d'un processus

## Module 2 : Quiz (3)

5. Quelle organisation a créé le Chaos Monkey en open source ?
  - a) Facebook
  - b) Etsy
  - c) Google
  - d) Netflix

## Module 2 : Réponses aux quiz

- |   |  |  |
|---|--|--|
| 1 | Laquelles des Three ways concerne les boucles de feedback tels que les résultats des tests automatisés ? | a) La première voie<br><b>b) La deuxième voie</b><br>c) La troisième voie<br>d) Les Three ways   |
| 2 | Comment les principes de DevOps encadrent-ils le traitement de l'échec ?                                 | a) <b>Comme une occasion d'apprentissage</b><br>b) Comme un élément à ignorer<br>c) Qu'il devrait être puni<br>d) Que nous ne pouvons pas nous protéger contre   |
| 3 | Quel est le préalable à la maîtrise ?  | a) Talent & capacité<br>b) S'entraîner et étudier<br>c) Orientation et coaching<br><b>d) Répétition &amp; entraînement</b>   |
| 4 | Nous devons  | a) <b>Ne jamais transmettre un défaut connu en aval</b><br>b) Rechercher une vue d'ensemble du système<br>c) Autoriser l'optimisation locale à dégrader l'optimisation globale<br>d) Optimiser tous les liens d'un processus |
| 5 | Quelle organisation a créé le Chaos Monkey en open source ?  | a) Facebook<br>b) Etsy<br>c) Google<br><b>d) Netflix</b>   |

# MODULE 3

# Les principales pratiques de

# DevOps

## Module 3 : Les principales pratiques de DevOps

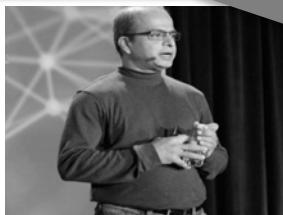
- Tests en continu
- Intégration continue, livraison continue, déploiement continu
- Site Reliability Engineering (SRE)
- DevSecOps
- ChatOps
- Kanban

Composant	Contenu du Module 3
Vidéo	GitHub Professional Guides: Intégration et livraison continues
Etude de cas	Capital One
Discussion	Pourquoi trop de WIP n'est pas bon
Exercice	Evaluez votre capacité CI/CD

## ETUDE DE CAS : Capital One

« Les outils sont une partie importante des approches Agile et DevOps actuelles. Un projet typique traite d'outils de gestion de projet Agile, contrôle de source, outils d'intégration continue (CI), outils de test, outils d'analyse de code statique et d'analyse de sécurité, outils de déploiement et de surveillance, pour en nommer quelques-uns. Les grandes entreprises et les systèmes complexes utilisent parfois plusieurs outils de CI, de test et d'analyse. Chacun bénéficie de beaux tableaux de bord affichant les informations clés stockées. Mais ce qui manque, c'est une vue unique et complète de l'état d'un pipeline de livraison en temps quasi réel. Chez Capital One, nous pensons que si les outils, l'automatisation et la collaboration sont très importants, une boucle de feedback continue est essentielle au succès de DevOps. »

“Axé sur les données, la technologie et la science des données.”



Topo Pal,  
Director & Platform  
Engineering Fellow



DevOps Institute © Tous droits réservés

- Code « committé » des centaines de fois par jour
- Intégration passée d'1 fois par mois à toutes les 15 minutes
- AQ passée d'1 fois par mois à 4 fois par jour
- Déploiement passé de manuel à complètement automatisé
- Mise en production mensuelle / trimestrielle à une fois par sprint

# Continu:

- Essai
- L'intégration
- Livraison
- Déploiement

## Tests en continu

Les tests en continu consistent à exécuter des tests automatisés dans le cadre du pipeline de déploiement pour obtenir un retour immédiat sur les risques business associés à une version du logiciel.



	Low	Medium	High	Elite
Automated build	64%	81%	91%	92%
Automated unit tests	57%	66%	84%	87%
Automated acceptance tests	28%	38%	48%	58%
Automated performance tests	18%	23%	18%	28%
Automated security tests	15%	28%	25%	31%
Automated provisioning and deployment to testing environments	39%	54%	68%	72%

Le « décalage vers la gauche » vise à renforcer la qualité du processus de développement logiciel en intégrant des tests en amont et fréquemment. Lorsque vous décalez vers la gauche, il y a moins de ruptures en production, car les anomalies sont détectées et résolus plus tôt.

## Integration continue

L'intégration continue (CI) est une pratique de développement qui oblige les développeurs à « committer » du code dans un référentiel partagé (maître/tronc) au moins une fois par jour.

- Chaque archivage est validé par
  - Une version automatisée
  - Des tests unitaires, d'intégration et de réception automatisés
- Dépend de normes de codage uniformes
- Nécessite des référentiels de contrôle de version et des serveurs CI pour collecter, construire et tester le code « committé »
- Fonctionne dans des environnements semblables à la production
- Permet une détection précoce et une correction rapide des erreurs dues aux modifications de code avant de passer en production

Bien qu'elles soient principalement associées au développement logiciel agile, les approches en cascade peuvent également tirer parti de l'intégration continue et des pratiques du test-driven development (TDD).

## Module 3 : Les principales pratiques de DevOps

# Les compétences de la chaîne d'outils CI/CD sont essentielles pour les opérations de développement

"En bref, les chaînes d'outils CI/CD contribuent à la rapidité et à la qualité du code, permettent une meilleure collaboration entre les équipes et automatisent de nombreuses étapes, tâches et processus, ce qui réduit le risque et le coût du développement de logiciels".

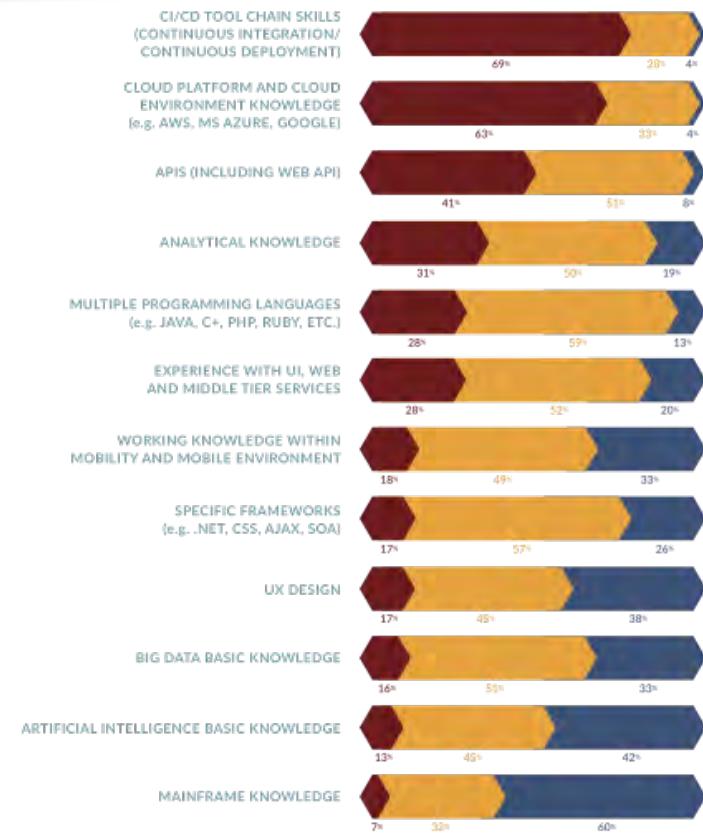
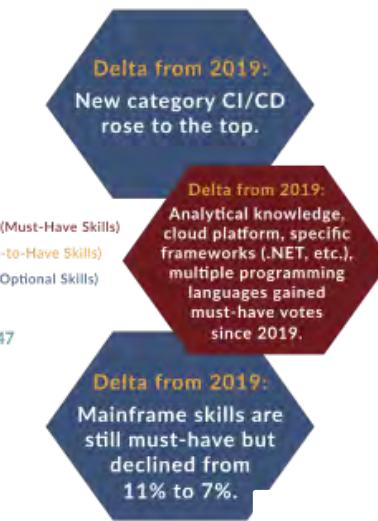


Figure 13: Technical Skills For The DevOps Human CI/CD Toolchain, Cloud Platform And Understanding APIs Are The Top 3 Must-have Technical Skills

Q How would you rate the importance of the following technical skills for your DevOps team members?

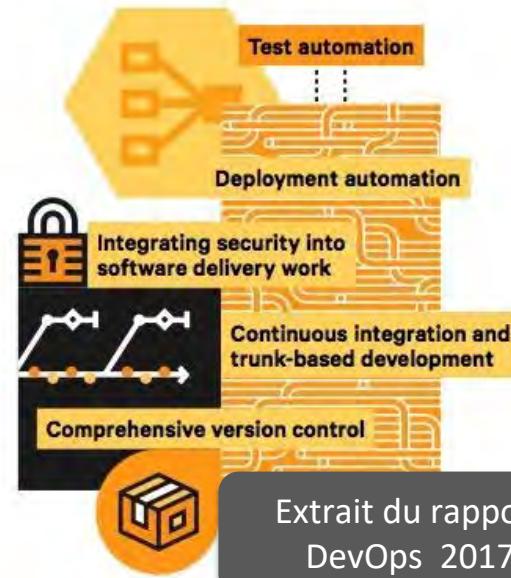


## Livraison continue

Le livraison continue (CD) est une méthodologie qui vise à s'assurer que le logiciel est **toujours dans un état livrable** tout au long de son cycle de vie.

- Fait passer l'intégration continue au niveau supérieur
- Fournit un feedback rapide et automatisé sur l'état de préparation à la production d'un système
- Donne la priorité au maintien du logiciel livrable / déployable par rapport au travail sur les nouvelles fonctionnalités
- S'appuie sur un pipeline de déploiement permettant des déploiements à la demande en un clic
- Réduit les coûts, les délais et les risques liés aux modifications incrémentielles

Factors that positively contribute to continuous delivery:

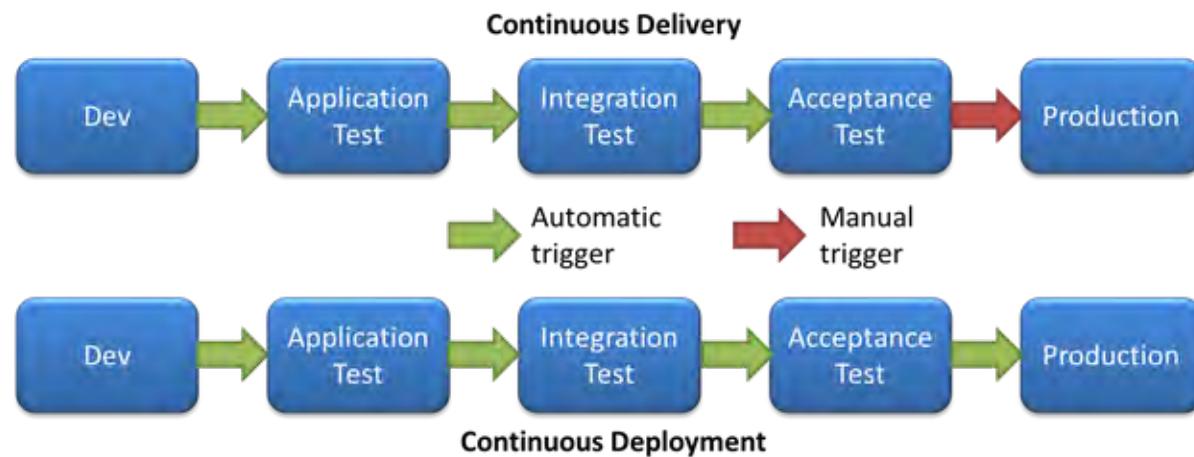


Extrait du rapport  
DevOps 2017

# EXERCICE

## Evaluez votre capacité CI/CD

## Livraison continue et déploiement continu



L'intégration continue est la pratique permettant la livraison continue de la valeur entre les mains des utilisateurs

Extrait de Mirco Hering : [notafactoryanymore.com](http://notafactoryanymore.com), auteur de 'DevOps for the Modern Enterprise'

Module 3 : Les principales pratiques de DevOps



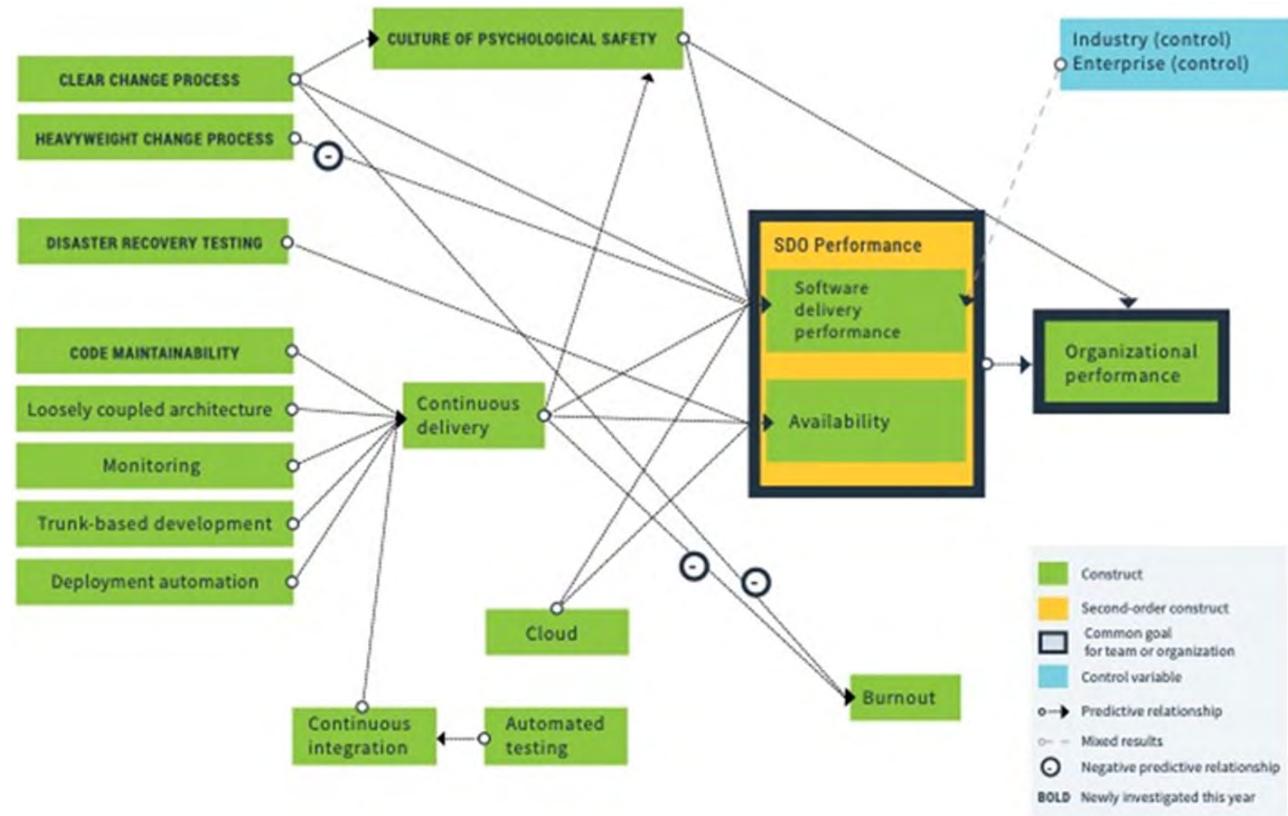
# GitHub Professional Guides



GitHub Professional Guide:  
Continuous Integration & Delivery (6:00)

Module 3 : Les principales pratiques de DevOps

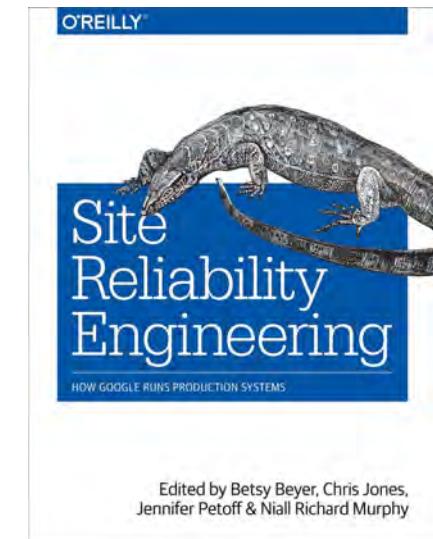
# Un déploiement continu peut conduire à de meilleures performances IT & Business



DevOps Institute © Tous droits réservés

## Site Reliability Engineering - SRE

- « Que se passe-t-il lorsqu'un ingénieur logiciel est chargé de ce qu'on appelait autrefois les opérations ? » Ben Treynor, Google
- Les objectifs sont de créer des systèmes logiciels ultra-évolutifs et extrêmement fiables
- 50% de leur temps de travail lié aux opérations (problèmes, gardes et interventions manuelles)
- 50% de leur temps consacré aux tâches de développement telles que les nouvelles fonctionnalités, la mise à l'échelle ou l'automatisation



Google compte désormais plus de 1500 ingénieurs SRE

## Ingénierie de la résilience

Capacité intrinsèque d'un système à adapter son fonctionnement avant, pendant ou après les changements et perturbations, afin de pouvoir poursuivre les opérations requises dans des conditions attendues ou inattendues.

- L'ingénierie de la résilience examine le fonctionnement de l'organisation dans son ensemble
- La meilleure défense est une bonne attaque
- Adopter une vue agressive, irréprochable et systémique après un incident
- Considérer à la fois les éléments humains et techniques
- Les systèmes doivent être plus forts que leur maillon le plus faible

"L'échec est le côté pile du succès" Eric Hollnagel

## DevSecOps

DevSecOps a pour but de s'appuyer sur le principe selon lequel « tout le monde est responsable de la sécurité » afin de répartir les prises de décision en matière de sécurité au niveau des personnes qui possèdent les compétences les plus adéquates.

[www.devsecops.org](http://www.devsecops.org)

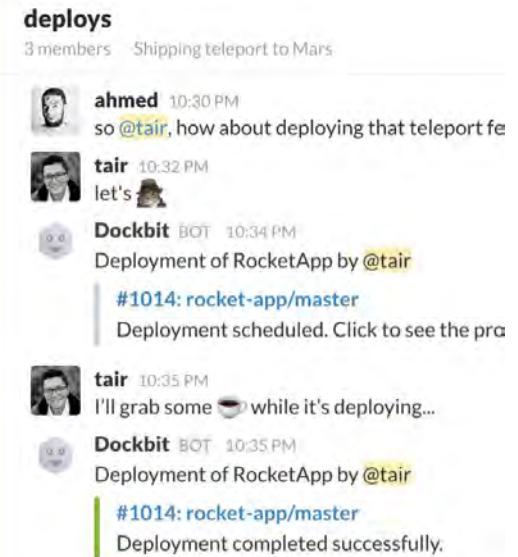
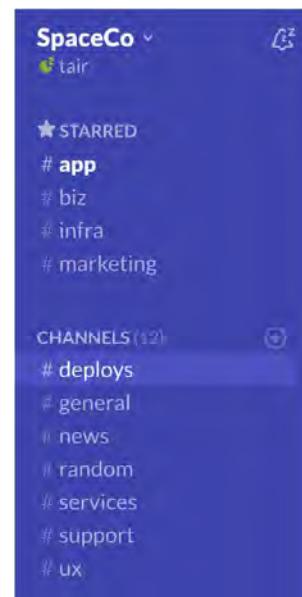
### Who turned out the lights?



- Introduit la sécurité sous forme de code
- Adopte la stratégie de test « décalage vers la gauche »
- Utilise l'automatisation pour la résilience, les tests, la détection et l'audit
- Brise la contrainte de sécurité

## ChatOps

Group chat client + chat bots = développement, livraison et support axés sur la conversation



La transparence de ChatOps raccourcit les boucles de feedback, améliore le partage d'informations, améliore la collaboration en équipe et permet une formation transversale. Il peut également être utilisé pour diminuer le MTTR.

## Kanban

Kanban est une approche de visualisation du travail qui permet de mieux « tirer » le flux de travail au sein d'un processus à un rythme raisonnable.

- Visualise et gère le flux de travail
- Tire le travail pour les équipes quand elles sont prêtes
- Permet aux gens de travailler en collaboration pour améliorer le flux
- Mesure la vélocité de l'équipe (quantité de travail effectué lors d'une itération)
- Réduit les temps morts et les pertes dans un processus



- Rend le travail visible
- Rend les politiques explicites
- Limite les travaux en cours (WIP) à la capacité

# DISCUSSION

## Pourquoi trop de WIP (Work in Progress) n'est pas bon

## Module 3 : Quiz (1)

1. Lequel des éléments suivants n'est pas nécessaire pour l'intégration continue ?
  - a) Les développeurs « commettent » du code au tronc/maître au moins quotidiennement
  - b) Déploiement en un clic.
  - c) Tests unitaires, d'acceptation utilisateur et d'intégration.
  - d) Normes de codage cohérentes.
  
2. Lequel des éléments suivants n'est pas un test non fonctionnel ?
  - a) Performance
  - b) Unitaire
  - c) Sécurité
  - d) Capacité

## Module 3 : Quiz (2)

3. Combien y-a-t-il d'ingénieurs de la fiabilité des sites chez Google ?

- a) 15
- b) 150
- c) 1,500
- d) 15,000

4. Laquelle de ces plateformes n'est pas une plateforme ChatOps ?

- a) Jira
- b) Slack
- c) Stride
- d) Teams

## Module 3 : Quiz (3)

5. Lequel de ces points n'est pas vrai à propos de Kanban ?

- a) Rend le travail visible
- b) Pousse le travail
- c) Applique les limites de WIP
- d) Mesure la vitesse de l'équipe

## Module 3 : Réponses au quiz

- 1 Lequel des éléments suivants n'est pas nécessaire pour le Continuous Integration ?
  - a) Les développeurs «committent » du code au tronc/maître au moins quotidiennement
  - b) Déploiement en un clic.**
  - c) Tests unitaires, d'acceptation utilisateur et d'intégration.
  - d) Normes de codage cohérentes.
- 2 Lequel des éléments suivants n'est pas un test non fonctionnel ?
  - a) Performance
  - b) Unitaire**
  - c) Sécurité
  - d) Capacité
- 3 Combien y-a-t-il d'ingénieurs de la fiabilité des sites chez Google ?
  - a) 15
  - b) 150
  - c) 1,500**
  - d) 15,000
- 4 Laquelle de ces plateformes n'est pas une plateforme ChatOps ?
  - a) Jira**
  - b) Slack
  - c) Stride
  - d) Teams
- 5 Lequel de ces points n'est pas vrai à propos de Kanban ?
  - a) Rend le travail visible
  - b) Pousse le travail**
  - c) Applique les limites de WIP (encours)
  - d) Mesure la vitesse de l'équipe

# MODULE 4

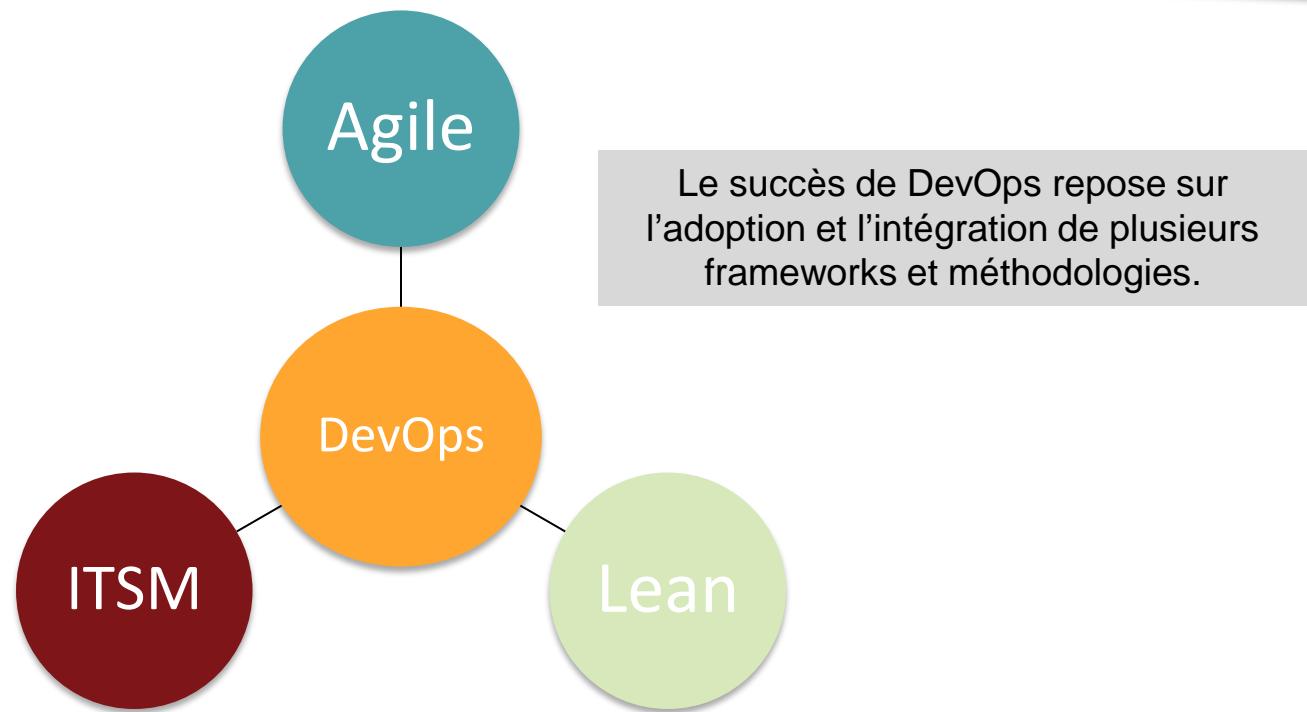
# Frameworks business et technologique

## Module 4 : Frameworks business et technologique

- Agile
- ITSM
- Lean
- Les outils du Lean :
  - Value Stream Mapping (VSM)
  - Improvement Kata
  - Culture de sécurité
  - Organisations apprenantes
  - Financement en continu

Composant	Module 4 : Contenu
Vidéo	Spotify Engineering Culture Part 1
Etude de cas	Alaska Air
Discussion	L'agilité dans les opérations
Exercice	Identifier et éradiquer le gaspillage

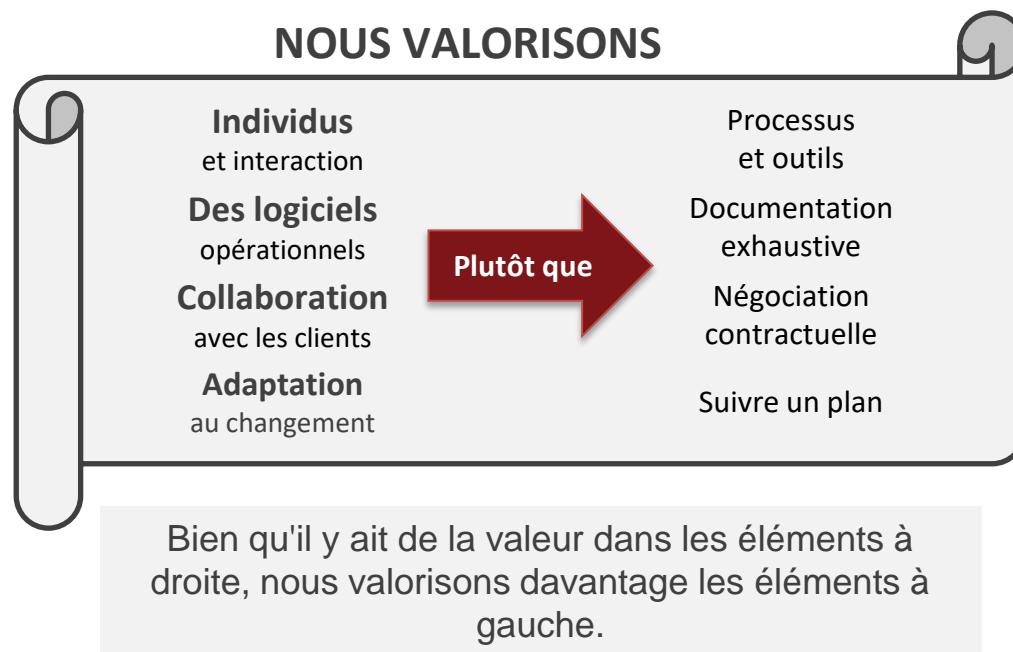
## DevOps ne peut s'en sortir seul



# Agile

## Le Manifeste Agile

Les concepts sous-jacents au développement logiciel agile ont été exposés pour la première fois dans le Manifeste Agile.



Il est plus important d'**ÊTRE** agile que de **FAIRE** de l'agile

- Être centré sur le client
- Être Lean
- Être collaboratif
- Être communicatif
- Être capable de s'adapter
- Être mesurable
- Être cohérent
- Être axé sur les résultats
- Être introspectif

# Scrum

Cadre simple permettant la collaboration efficace au sein de l'équipe sur des projets complexes. Scrum définit un ensemble de règles qui crée « **juste assez** » de structure pour que les équipes puissent concentrer leur innovation sur la résolution de ce qui, sans cette structure, pourrait être un problème insurmontable.

Scrum.org

Scrum est :

- La pratique de développement logiciel agile la plus couramment appliquée
- D'une simplicité trompeuse et difficile à maîtriser
- Ni un processus ni une technique pour les construction de produits

Scrum augmente la capacité de mises en production plus fréquentes.

<http://www.scrumguides.org/>



## Rôles

- Product Owner
- Scrum Master
- Équipe de développement



## Artefacts

- Product Backlog
- Sprint Backlog
- Incrément

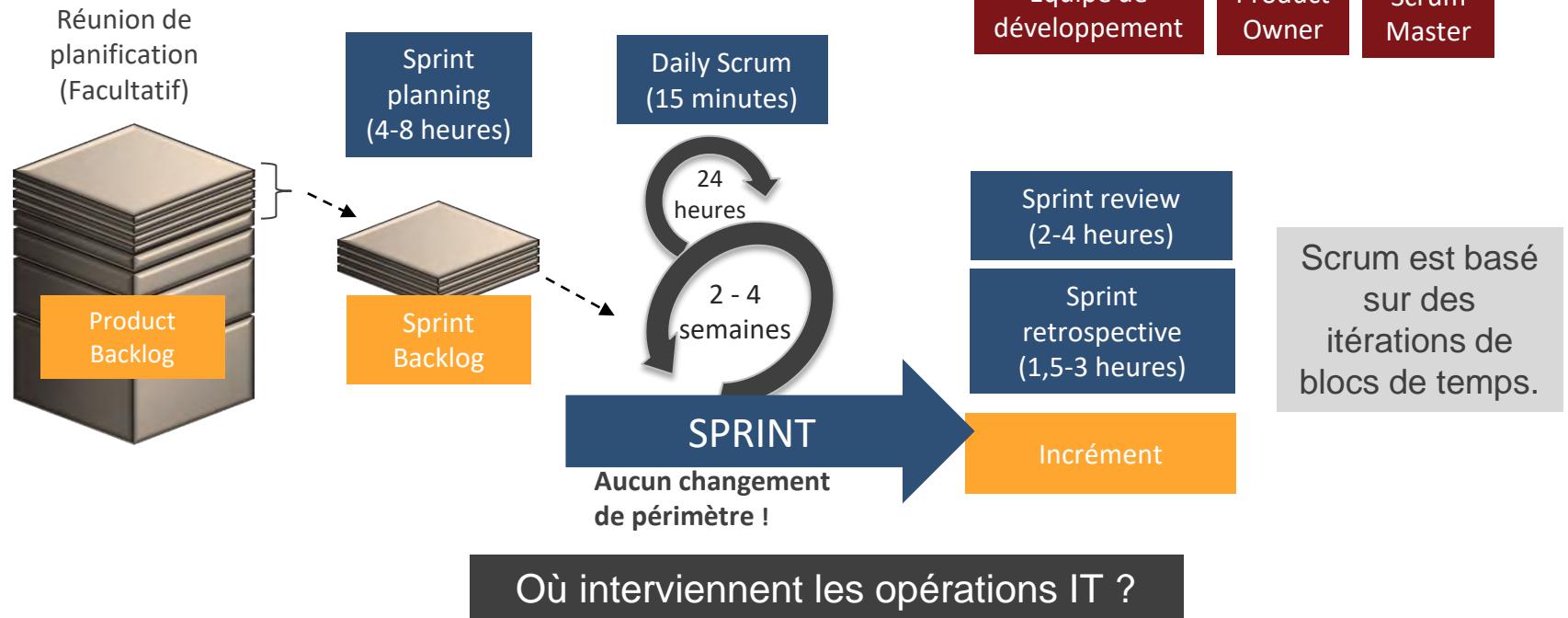


## Événements

- Sprint
- Sprint planning
- Daily Scrum
- Sprint review
- Sprint retrospective

## Scrum en bref

Scrum = 3 Rôles + 3 Artefacts + 5 Evénements



## Le Product Backlog

- **Stories**, également appelées « User stories » sont de courtes exigences ou demandes rédigées du point de vue utilisateur final.
- Les **Epics** décrivent de grandes User stories qui seront décomposées en de nombreuses Stories.
- Les **Initiatives** regroupent des Epics qui ont un objectif commun.
- Les **Thèmes** sont des domaines d'intérêt qui couvrent toute l'organisation. Ils permettent de relier des Stories ayant des points communs, y compris lorsqu'ils appartiennent à des Epics différents.



## Scaled Agile Framework® (SAFe™)

Scaled Agile Framework (SAFe) est un cadre éprouvé et accessible au public pour l'application de principes et pratiques Lean-Agile à l'échelle de l'entreprise.

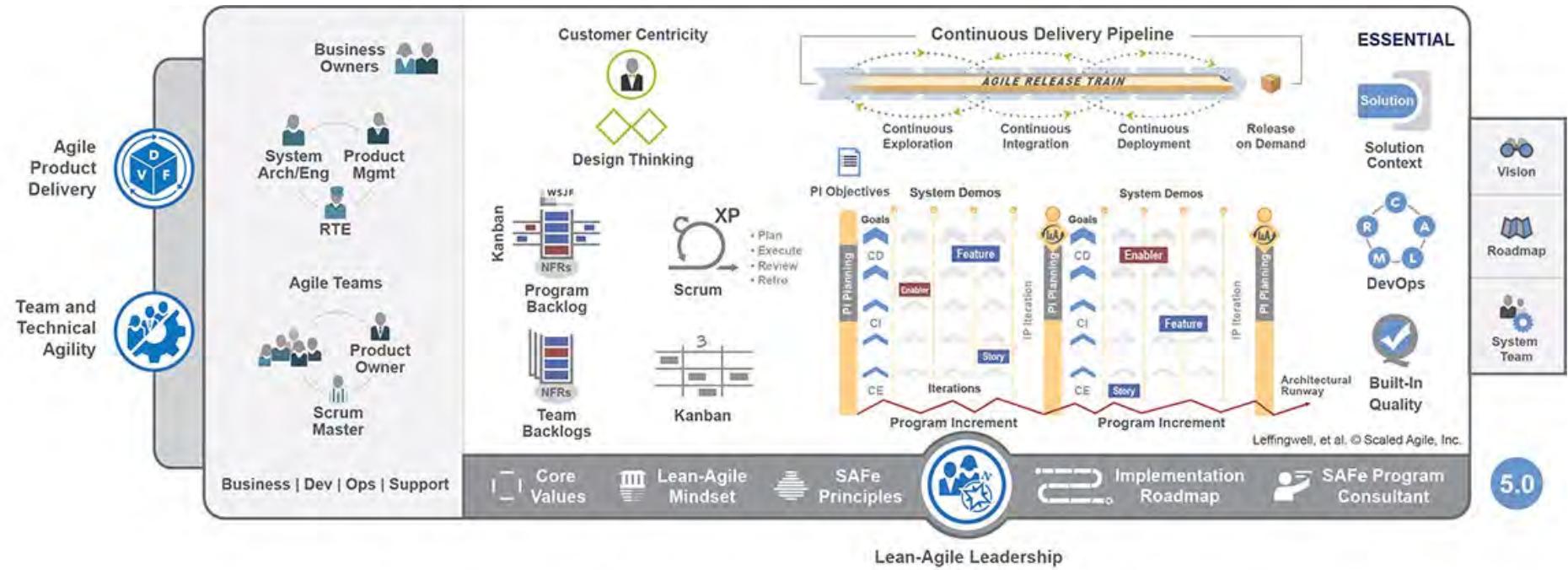
- Intègre la pensée Lean et Agile au développement logiciel
- Se concentre sur le développement itératif et incrémental, le développement agile de logiciels, le flux de développement de produits, la pensée Lean et l'expérience de terrain à l'échelle de l'entreprise
- Peut être appliqué aux organisations avec un grand nombre de praticiens et d'équipes

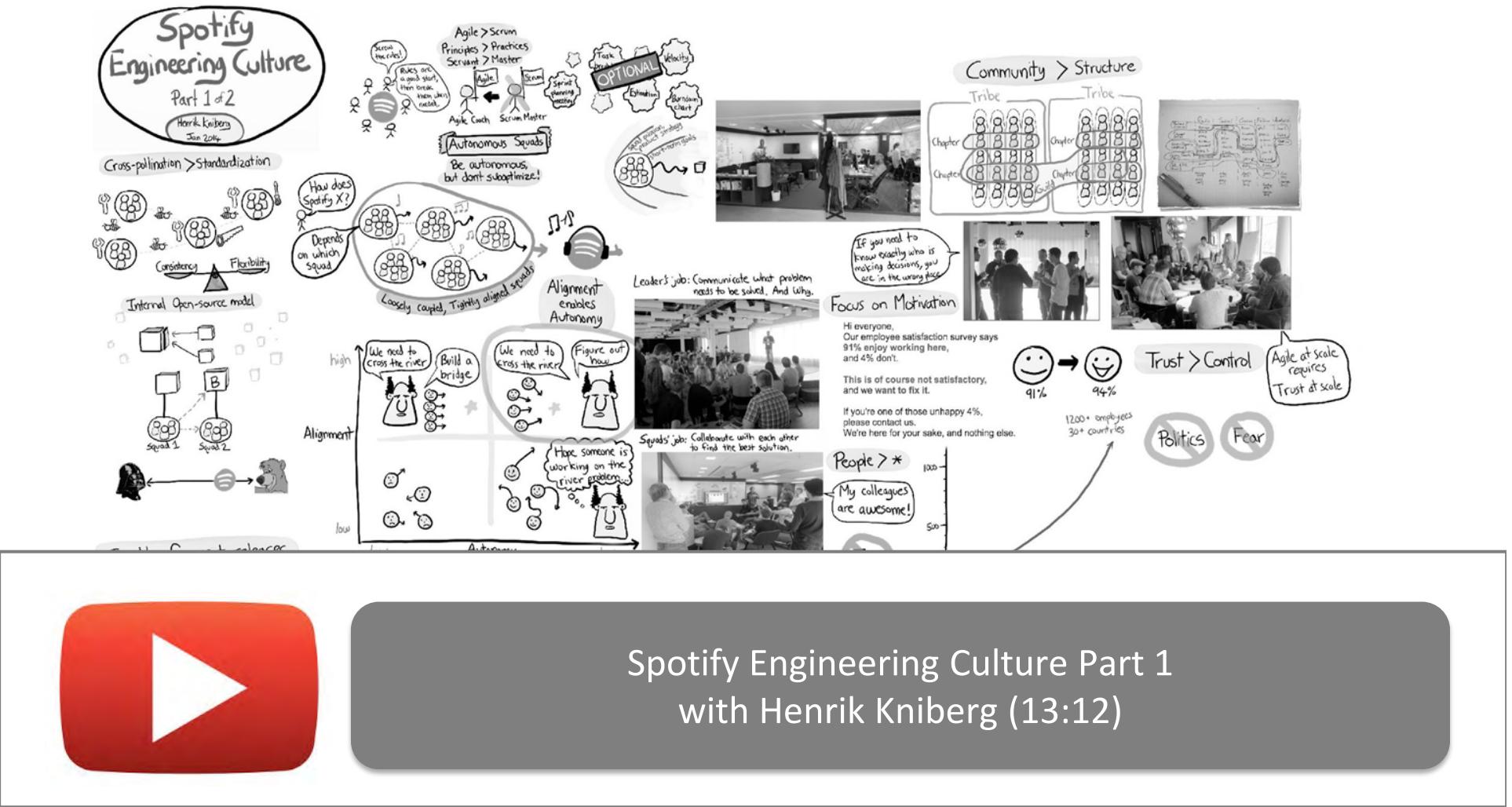


<http://www.scaledagileframework.com/>

## Module 4 : Frameworks business et technologique

# SAFe pour les entreprises Lean



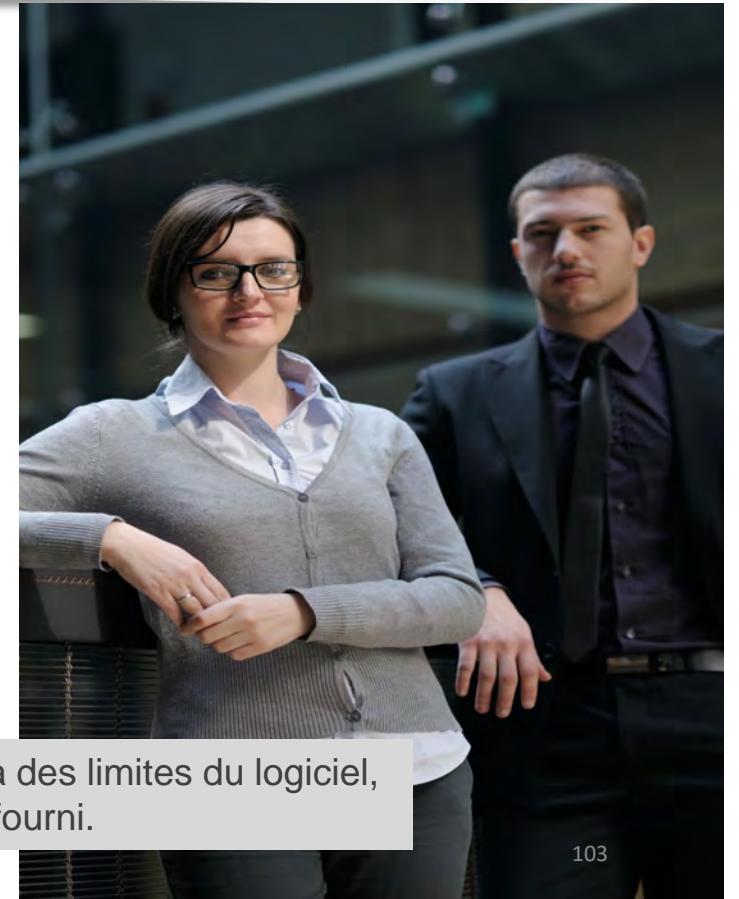


Spotify Engineering Culture Part 1  
with Henrik Kniberg (13:12)

## Contribuer à l'agilité

DevOps contribue à l'agilité en :

- Brisant les silos
- Optimisant les contraintes
- Adoptant une approche unifiée de l'ingénierie des systèmes
- Appliquant des principes agiles aux Devs et Ops
- Partageant des connaissances, des compétences, l'expérience et des données
- Reconnaissant la criticité de l'automatisation
- Déployant plus rapidement avec moins d'erreurs



DevOps étend les principes agiles au-delà des limites du logiciel,  
à l'ensemble du service fourni.

# Gestion des services informatiques (ITSM)

A black and white portrait of Gene Kim, a man with dark hair and a beard, wearing glasses and a striped shirt, smiling at the camera.

« Je suis fermement convaincu que la gestion des services informatiques (ITSM) et le mouvement DevOps ne sont pas contradictoires. Bien au contraire, ils vont parfaitement bien ensemble. »

Gene Kim

## Gestion des services informatiques

La gestion des services informatiques (ITSM) désigne la mise en œuvre et la gestion de services informatiques de qualité répondant aux besoins de l'entreprise.

- Fournit des conseils et une structure aux processus tels que la gestion des changements, de la configuration, des mises en production, des incidents et des problèmes
- Les processus ITSM sous-tendent l'ensemble du cycle de vie des services : stratégie, conception, transition, opérations, amélioration continue et création de valeur
- DevOps a besoin des pratiques ITSM pour atteindre son objectif de déployer des changements plus rapidement sans interruption

Des processus de gestion des services reproductibles - adaptés aux besoins business actuels d'une organisation - peuvent ouvrir la voie à une distribution continue stable et à un flux accru.

## IT Infrastructure Library® (ITIL®)

ITIL® 4 définit un service comme étant un moyen de permettre la co-création de valeur en facilitant des résultats que les clients veulent atteindre, sans qu'ils n'aient à gérer les coûts et les risques spécifiques.

### Les composants clés du référentiel ITIL® 4 sont :

Le modèle à quatre dimensions	Système de Valeur des Services (SVS)
<ol style="list-style-type: none"><li>1. Organisations et personnes</li><li>2. Information et technologie</li><li>3. Partenaires et fournisseurs</li><li>4. Flux de valeur et processus</li></ol>	<ol style="list-style-type: none"><li>1. Les principes directeurs</li><li>2. La gouvernance</li><li>3. La chaîne de valeur des services (SVC)</li><li>4. L'amélioration continue</li><li>5. Les pratiques</li></ol>

ITIL® 4 (2019) modernise ITIL® en repensant une grande partie des pratiques ITSM dans le contexte élargi de l'expérience client, des flux de valeur, de la transformation numérique et adopte de nouvelles façons de travailler avec Lean, Agile et DevOps.

AXELOS® ITIL® et IT Infrastructure Library® sont des marques déposées d'AXELOS Limited.

## Les pratiques ITIL® 4 : DevOps les impactent toutes

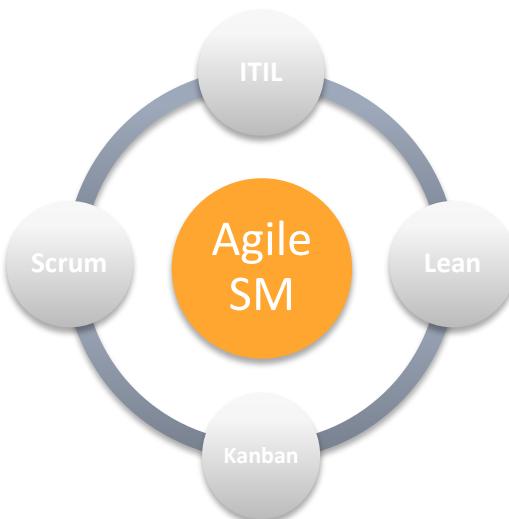
Pratiques de gestion générales	Pratiques de gestion des services	Pratiques de gestion technique
<ul style="list-style-type: none"> <li>• Gestion de l'architecture</li> <li>• Amélioration continue</li> <li>• Gestion de la sécurité de l'information</li> <li>• Gestion des connaissances</li> <li>• Mesure et rapports</li> <li>• Gestion du changement organisationnel</li> <li>• Gestion de portefeuille</li> <li>• Gestion des projets</li> <li>• Gestion des relations</li> <li>• Gestion des risques</li> <li>• Gestion financière des services</li> <li>• Gestion de la stratégie</li> <li>• Gestion des fournisseurs</li> <li>• Gestion des effectifs et des talents</li> </ul>	<ul style="list-style-type: none"> <li>• Gestion de la disponibilité</li> <li>• Business analysis</li> <li>• Gestion de la capacité et de la performance</li> <li>• Habilitation des changements</li> <li>• Gestion des incidents</li> <li>• Gestion des actifs informatiques</li> <li>• Surveillance et gestion des événements</li> <li>• Gestion des problèmes</li> <li>• Gestion des mises en production</li> <li>• Gestion du catalogue des services</li> <li>• Gestion de la configuration des services</li> <li>• Gestion de la continuité des services</li> <li>• Conception des services</li> <li>• Gestion des niveaux de service</li> <li>• Gestion des demandes de service</li> <li>• Validation et tests de services</li> </ul>	<ul style="list-style-type: none"> <li>• Gestion des déploiements</li> <li>• Gestion de l'infrastructure et des plates-formes</li> <li>• Développement et gestion des logiciels</li> </ul> <p>Toutes ces pratiques sont nécessaires, sous une forme ou une autre, dans DevOps (à l'exception peut-être de la gestion des projets).</p>

# DISCUSSION

## L'agilité dans les opérations

## Agile Service Management

Agile Service Management (Agile SM) garantit que les processus ITSM reflètent les valeurs Agile et sont conçus avec « juste assez » de contrôle et de structure afin de fournir des services efficaces qui permettent aux clients d'obtenir les résultats attendus aux moments voulus



- Adapte les pratiques agiles à la conception des processus ITSM
- Implémente la gestion des services par petits incrémentés intégrés
- Assure que les processus ITSM reflètent les valeurs Agiles depuis la conception initiale jusqu'à CSI
- Encourage les processus « minimum viables » et « juste assez » pour augmenter la vitesse et la conformité

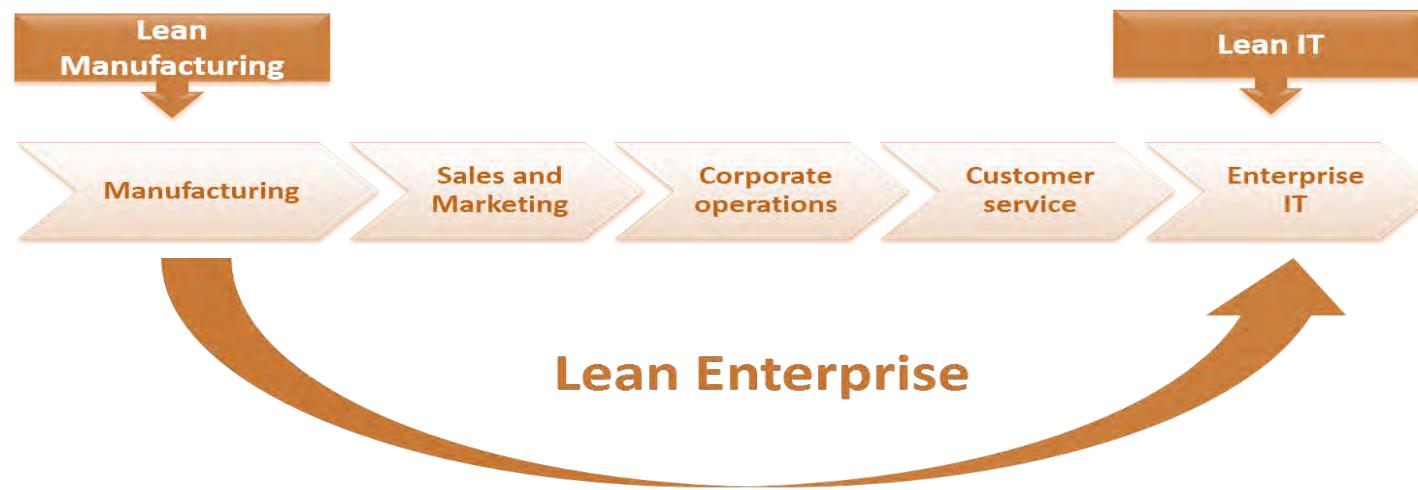
Source : Agile Service Management Guide

Agile Service Management ne réinvente pas l'ITSM - il modernise l'approche.

# Lean

## Perspectives Lean

DevOps a ses racines dans le monde du Lean Manufacturing, qui répond aux problèmes des ingénieurs qui conçoivent des produits que les usines ne peuvent se permettre de fabriquer.



Lean IT applique les idées clés de la production  
Lean au développement et à la gestion de produits  
et services informatiques.

## Sources de gaspillage

L'objectif de la pensée Lean est de créer plus de valeur pour les clients avec moins de ressources et moins de gaspillage. Le gaspillage correspond à toute activité qui n'apporte pas de valeur ajoutée au processus.

Source	Raison	Exemples
<b>Défauts</b>	Dévier des exigences ; erreurs	Défaillances, erreurs connues, désinformation
<b>Surproduction</b>	Produire plus ou plus vite que nécessaire	Trop de documentation ou code
<b>Attente</b>	Délais dûs à attente d'une étape précédente	Décisions, approbations, réponses différées
<b>Non-utilisation</b>	Connaissance ou créativité non utilisée	Compétence, innovation, communication non utilisées
<b>Transport</b>	Déplacer des produits d'un lieu à un autre	Transferts, emails ou réunions nombreuses
<b>Inventaire</b>	Avoir plus de matériel que nécessaire	Logiciels non utilisés, trop d'infrastructure, backlogs ou emails
<b>Déplacement</b>	Déplacer des personnes ou des biens plus que nécessaire	Trop souvent déplacer du code ou infrastructure
<b>Traitements excessifs</b>	Faire plus que nécessaire	Sur-ingénierie, impossible de créer des modèles et autres actifs réutilisables

# EXERCICE

## Identifier et éradiquer le gaspillage



Gene Kim

« Je pense que la plupart des approches DevOps émergent lorsque vous appliquez des techniques telles que Lean, le système de production Toyota, la théorie des contraintes, etc., à la chaîne de valeur informatique.»

## ETUDE DE CAS : Alaska Air

« Nous devons fonctionner comme une marque non-conformiste. nous devons fonctionner de manière fondamentalement différente ; nous devons briser le paradigme de ce que tout le monde pense et imagine du transport aérien. Nous nous sommes donc concentrés sur deux éléments clés 1) gérer efficacement les opérations et 2) encourager l'innovation technologique. »

« Alaska Air se voit comme une entreprise technologique avec des ailes. »



Veresh Sita, CIO



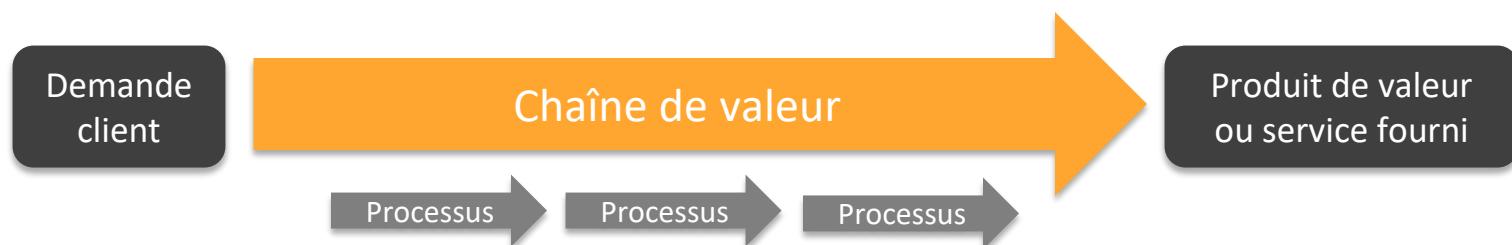
### Avantages

- Peut exposer ses API en toute sécurité à des milliers de services tiers
- Des clients plus heureux, des revenus plus élevés
- Changement d'échelle facile et à moindre coût
- Les sites fonctionnent à un point optimal en performance et coût
- Meilleure productivité, pollinisation transverse et partage des connaissances plus rapides

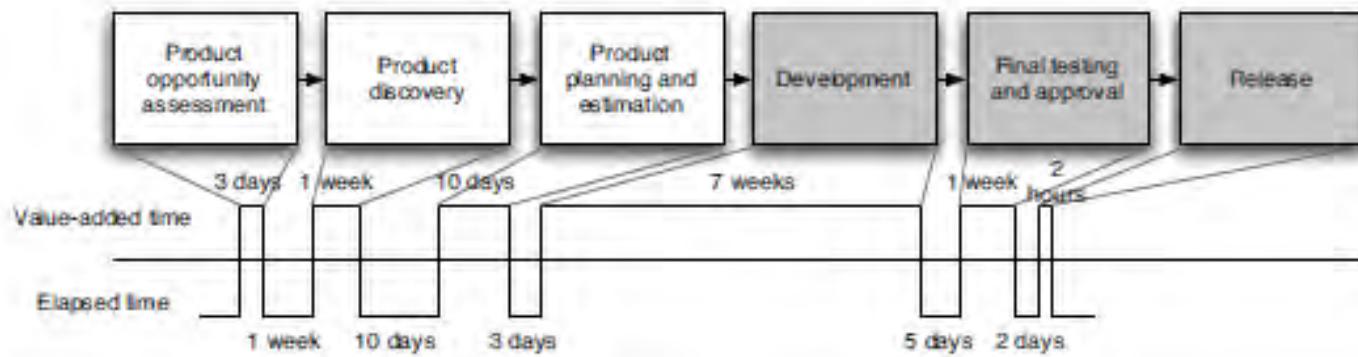
## Value Stream Mapping (VSM)

Value stream mapping est un outil Lean qui décrit le flux d'informations, de matériaux et de travail entre silos fonctionnels, en mettant l'accent sur la quantification et l'élimination du gaspillage, y compris en termes de temps et de qualité.

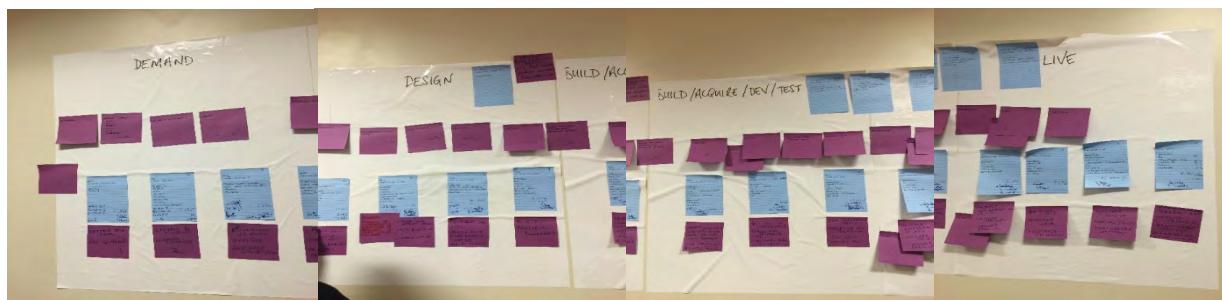
- Une chaîne de valeur est la séquence d'activités requises pour concevoir, produire et livrer un produit ou un service spécifique
- Les chaînes de valeur couvrent généralement plusieurs processus
- La cartographie des chaînes de valeur permet aux équipes fonctionnelles transverses de :
  - Voir tout une chaîne de valeur du point de vue du travail et des flux d'informations
  - Identifier les zones de gaspillage sans valeur qui pourraient être éliminées dans le but d'améliorer la chaîne et de générer une plus grande valeur
  - Identifier, hiérarchiser et mesurer les améliorations



## Exemple de Value Stream Maps



Source :  
Jez Humble - *Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation*

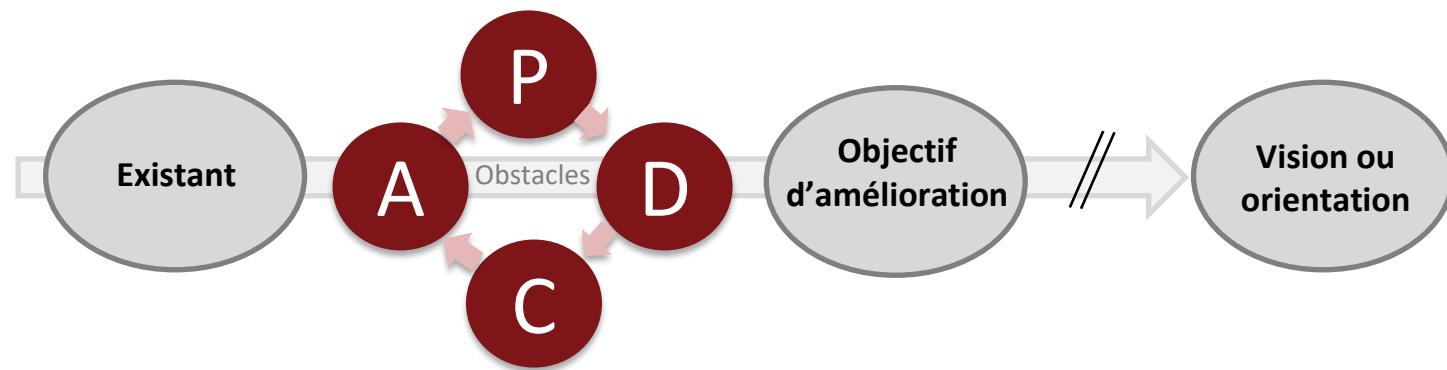


Source : Ranger4

## Improvement Kata

Un kata est une manière structurée de penser et d'agir que vous pratiquez jusqu'à ce qu'elle devienne une habitude.

- 2 Analyser l'existant
- 4 PDCA / expérimenter vers l'objectif
- 3 Définir un objectif d'amélioration
- 1 Comprendre la vision ou l'orientation à long terme



Le Improvement Kata est un processus en quatre étapes qui met l'accent sur l'apprentissage et l'amélioration du travail. Il prend en compte la vision ou l'orientation à long terme de l'organisation.  
Plan > Do > Check > Act (PDCA)

## Culture de la sécurité

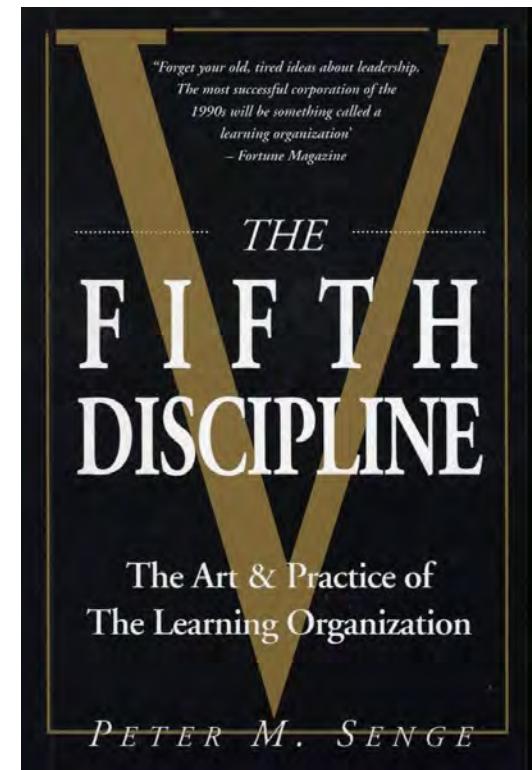
- Attitude, croyances, perceptions et valeurs que partagent les employés en matière de sécurité sur le lieu de travail
- Post-mortem sans reproches
- Evaluer les incidents
- Eviter les points uniques de défaillance (SPOF)
- Le Cordon d'Andon - merci d'avoir créé une opportunité d'apprentissage

« Un incident est un investissement non planifié, et si vous ne le voyez pas comme tel en tant que leader, vous n'obtenez pas le retour sur l'investissement qui a déjà été réalisé pour vous. »

Attribué à John Allspaw par Sidney Dekker dans Beyond the Phoenix Project

## Organisations apprenantes

- S'engager à apprendre
- S'améliorer nécessite d'apprendre quelque chose de nouveau
- Ne pas apprendre génère une dette culturelle
- Les humains aiment l'excellence (et aussi l'autonomie et le partage d'objectifs)
- L'engagement de la direction est essentiel



## Financement en continu

- Le financement traditionnel se fait sur des cycles annuels
- Un financement agile peut être :
  - Coût fixe ou continu
  - Révisé fréquemment
- Financement par produit / équipe
- Financement à risque (ou basé sur le pari)
- Axé sur la mesure du retour

In reply to Helen Beal



Jonathan Smart  
@jonsmart

@HelenRanger4 we're starting to pilot agile investment with qtrly rolling wave instead of annual budgeting.

30/06/2016, 11:44 from Paddington, London



## Module 4 : Quiz (1)

1. Dans le Manifeste Agile, nous valorisons un logiciel opérationnel par rapport à :
  - a) Processus et outils
  - b) Documentation exhaustive
  - c) Negociations de contrats
  - d) Suivre un plan
  
2. Lequel de ces modèles n'est pas un modèle de processus ITSM ?
  - a) Modèle de changement
  - b) Modèle de version
  - c) Modèle d'incident
  - d) Modèle de développement

## Module 4 : Quiz (2)

3. Lequel de ces outils n'est pas un outil Lean ?

- a) A5 thinking
- b) Value Stream Mapping
- c) Improvement kata
- d) Kanban

4. Quelle est la première étape de l'Improvement Kata ?

- a) Analyser l'existant
- b) Définir un objectif d'amélioration
- c) Plan Do Check Act (PDCA)
- d) Comprendre la vision ou l'orientation à long terme

## Module 4 : Quiz (3)

5. Qui a écrit le livre sur « The Learning Organization » ?

- a) Peter Senge
- b) Jonathan Smart
- c) Henrik Kniberg
- d) Gene Kim

## Module 4 : Réponses au Quiz

- 1 Dans le Manifeste Agile, nous valorisons un logiciel qui fonctionne par rapport à :
  - a) Processus et outils
  - b) Documentation exhaustive**
  - c) Negociations de contrats
  - d) Suivre un plan
  
- 2 Lequel de ces modèles n'est pas un modèle de processus ITSM ?
  - a) Modèle de changement
  - b) Modèle de version
  - c) Modèle d'incident
  - d) Modèle de développement**
  
- 3 Lequel de ces outils n'est pas un outil Lean ?
  - a) A5 thinking**
  - b) Value Stream Mapping
  - c) Improvement kata
  - d) Kanban
  
- 4 Quelle est la première étape de l'improvement kata ?
  - a) Analyser l'existant
  - b) Définir un objectif d'amélioration
  - c) Plan Do Check Act (PDCA)
  - d) Comprendre la vision ou l'orientation à long terme**
  
- 5 Qui a écrit le livre sur « The Learning Organization » ?
  - a) Peter Senge**
  - b) Jonathan Smart
  - c) Henrik Kniberg
  - d) Gene Kim

# MODULE 5

## Culture, comportements et modèles opérationnels

# Culture, comportements et modèles opérationnels

- Définir la culture
- La dette culturelle
- Modèles de comportement
- Modèles organisationnels

Composant	Module 5 : Contenu
Vidéo	Spotify Engineering Culture Part 2
Etude de cas	Cible
Discussion	Se positionner sur la courbe de changement
Exercice	Evaluer et améliorer avec le modèle Westrum



Peter Drucker

## Qu'est-ce qu'une culture d'organisation ?



« Vous ne pouvez pas changer directement la culture. Mais vous pouvez changer de comportement, et le comportement devient la culture. »

Lloyd Taylor, VP IT Operations LinkedIn

Les valeurs et les comportements qui contribuent à l'environnement social et psychologique unique d'une organisation.

[www.businessdictionary.com](http://www.businessdictionary.com)

## DevOps aide à surmonter la dette culturelle

La dette culturelle se produit lorsque les considérations culturelles sont ignorées ou différées en faveur de la croissance et de l'innovation.



« Le taux d'intérêt effectif sur la dette culturelle est en général plus élevé que sur la dette technique. » Dharmesh Shah, Founder & CTO



La culture en silo de l'IT et d'autres défis organisationnels résultent directement du non-respect de considérations culturelles en faveur d'une augmentation rapide de la technologie d'entreprise. La date d'échéance est aujourd'hui !

## Caractéristiques d'une culture DevOps

- Vision, objectifs et incitations partagés
- Communication ouverte, honnête et bidirectionnelle
- Collaboration
- Fierté de la qualité de travail
- Respect
- Confiance
- Transparence
- Amélioration continue :
  - Expérimentation
  - Prise de risque intelligente
  - Apprendre et s'entraîner
- Axé sur les données
- Sûr
- Réflexion
- Reconnaissance

La culture organisationnelle est l'un des précurseurs les plus puissants à la fois des performances informatiques et des performances globales de l'organisation.

## Faire évoluer les pensées et les comportements

De

- Centré sur l'IT (de l'intérieur vers l'extérieur)
- Silos
- Commander et contrôler
- Axé sur les tâches
- Faire des reproches
- Réactif
- Contenu
- Résistant
- Faible confiance



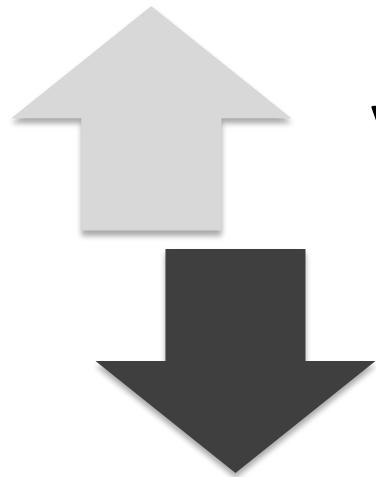
à

- Centré sur le client (de l'extérieur vers l'intérieur)
- Équipes fonctionnelles transverses
- Collaboratif
- Axé sur les résultats
- Responsabilité
- Proactif
- Courageux
- Souple
- Confiance élevée

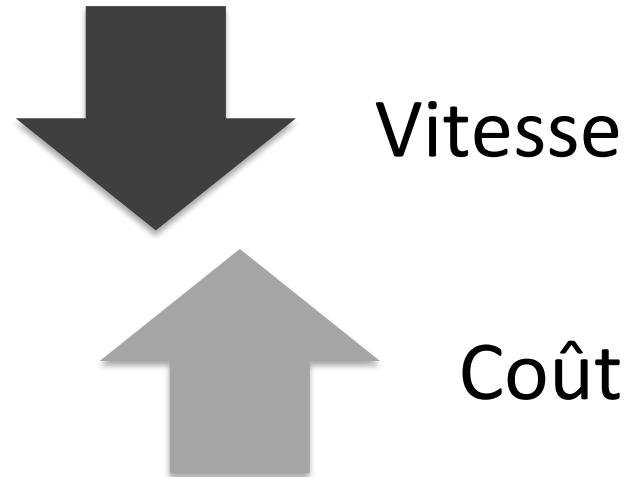
Le vrai changement de culture prend du temps. Il doit être progressif et exécuté à un rythme réaliste.

## Confiance élevée vs. faible confiance

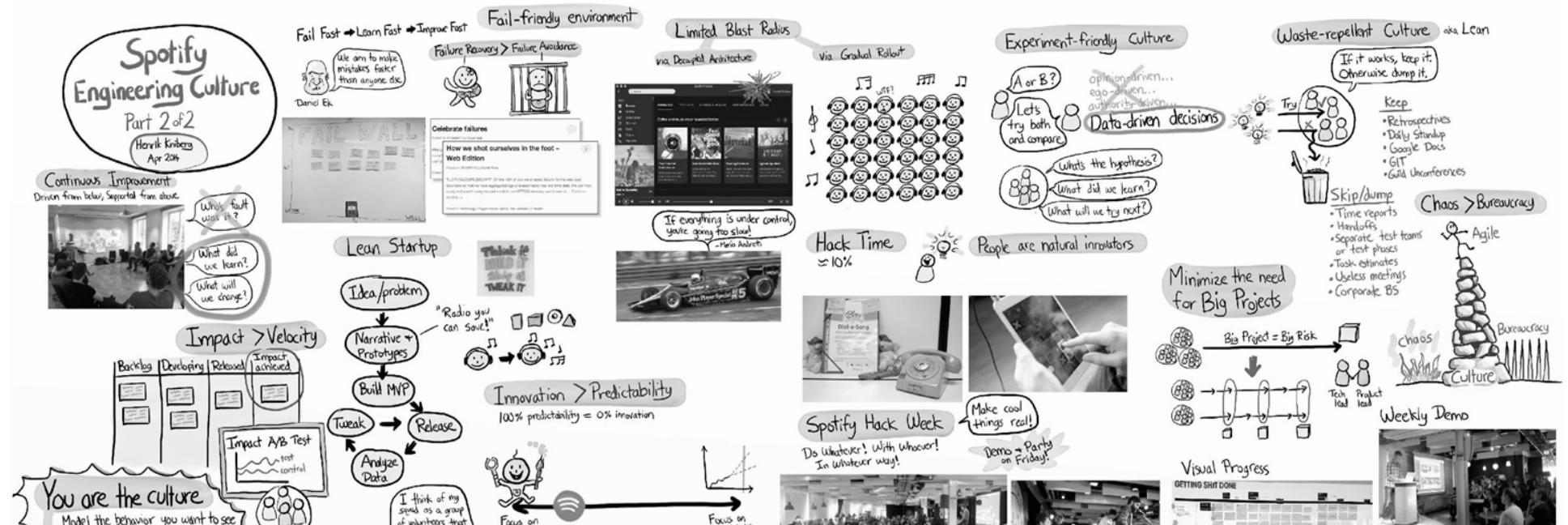
**Confiance élevée**



**Faible confiance**



## Module 5 : culture, comportements et modèles opérationnels



Spotify Engineering Culture Part 2  
with Henrik Kniberg (13:27)

## Culture et flux d'informations

Pathologique (Orientée pouvoir)	Bureaucratique (Orientée règles)	Génératrice (Orientée performances)
L'information n'est pas disponible	L'information peut être ignorée	L'information est activement recherchée
Les messagers sont "fusillés"	Les messagers sont isolés	Les messagers sont encouragés
Les responsabilités sont évitées	Les responsabilités sont compartimentalisées	Les responsabilités sont partagées
Les liens directs sont découragés	Les liens directs sont permis non-encouragés	Les liens directs sont récompensés
L'échec est couvert	L'organisation est juste et pardonne	L'échec entraîne une investigation
L'originalité est bridée	L'originalité crée des problèmes	L'originalité est mise en avant

Source : Westrum, *A Typology of Organizational Cultures*

Les organisations à haut niveau de confiance encouragent la bonne circulation de l'information, la collaboration fonctionnelle transverse, le partage des responsabilités, l'apprentissage à partir des échecs et les idées nouvelles.

# EXERCICE

## Evaluation et amélioration à l'aide du modèle Westrum

« Les gens n'offrent pas de résistance au changement mais à être changés. »

Peter Senge

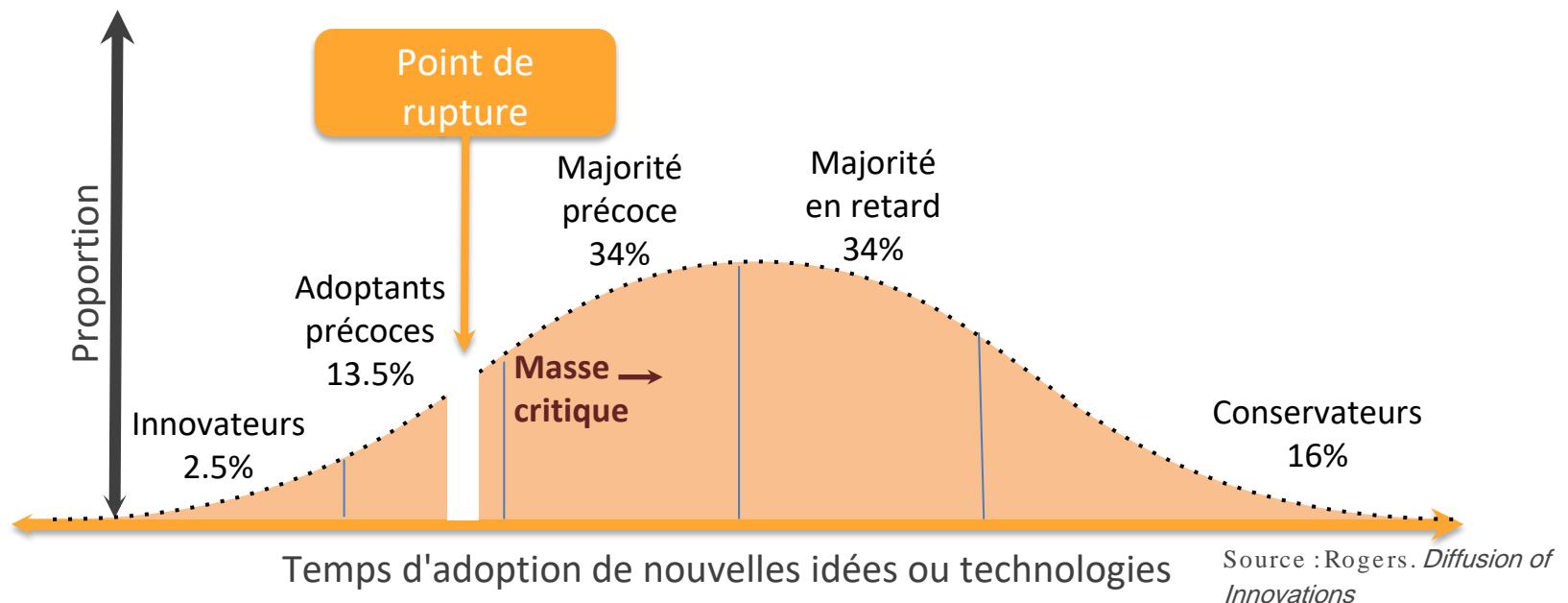


# Changer de culture n'est jamais facile

Les gens n'offrent généralement pas de résistance à leurs propres idées.

- Vous ne pouvez pas changer les gens ; ils sont les seuls à pouvoir le faire
- Le changement prend presque toujours plus de temps et coûte plus cher que prévu
- L'implication des parties prenantes est essentielle
- Les personnes qui participent aux décisions sur quoi et comment changer sont beaucoup plus susceptibles d'accepter le changement

## Adaptation des individus au changement à des rythmes différents

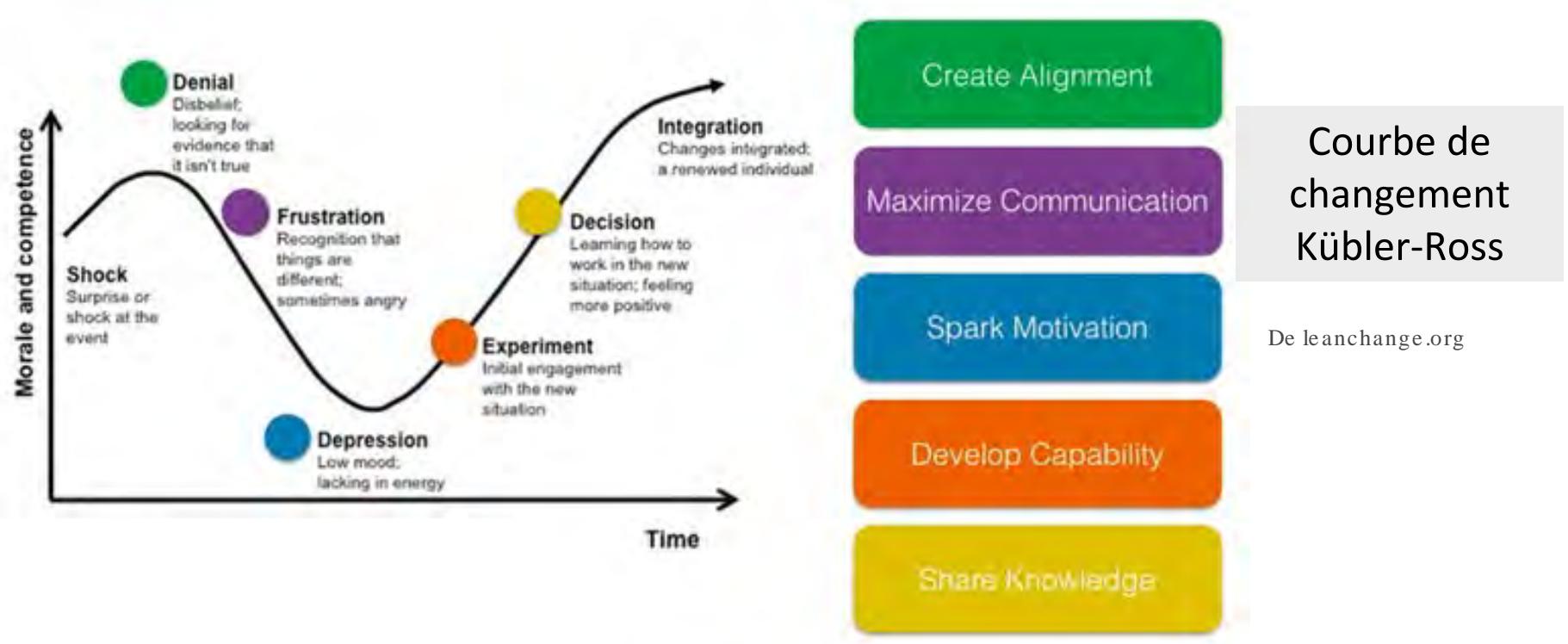


L'adoption signifie qu'une personne fait quelque chose différemment qu'auparavant.

# DISCUSSION

## Se positionner sur la courbe de changement

## Les étapes de l'acceptation du changement



# La communication est essentielle

- Une culture DevOps nécessite une communication rapide et efficace
- Les outils partagés facilitent une communication opportune et significative :
  - Plateformes de chat
  - Gestionnaires de tâches
  - Outils sociaux
  - Outils de gestion des alertes
  - Plateformes de partage de connaissances



## Encourager la collaboration

La collaboration implique des personnes travaillant conjointement avec d'autres dans l'atteinte d'un objectif commun. Dans un environnement collaboratif, la contribution de chacun est valorisée.

- La collaboration :
  - Est volontaire (idéalement)
  - Implique le partage :
    - Responsabilité des résultats
    - Ressources
  - Nécessite de la coopération, du respect et de la confiance

Quelle est la différence entre collaboration et communication ?

- Nécessite la participation :
  - Fournir des feedback
  - Identifier et résoudre les problèmes
  - Apprendre et partager les connaissances et l'expertise
  - Partager et même échanger des responsabilités
  - Prendre et maintenir des engagements réalistes

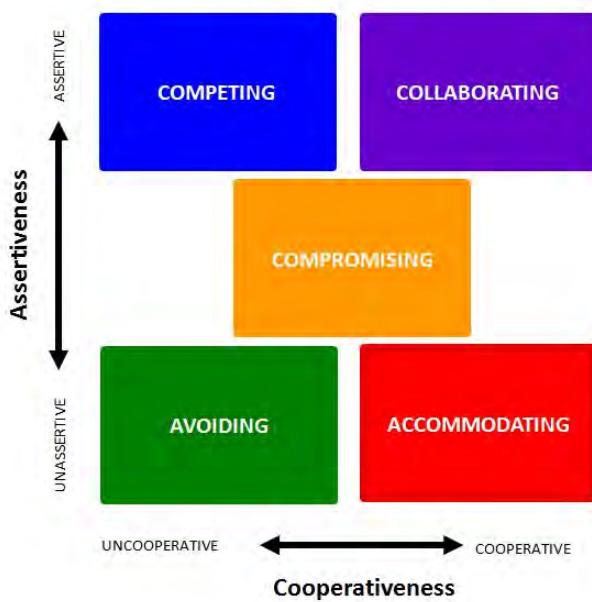
## S'attendre à un conflit : mode conflit de Thomas-Kilmann

Parce que tout le monde a des attentes et des désirs différents, le conflit fait naturellement partie de nos interactions avec autrui. L'inventaire des conflits de Thomas-Kilmann (TKI) mesure les choix comportementaux d'une personne dans certaines situations conflictuelles.

Mode conflit	Approche	Résultat
Rivaliser	Assertif et non coopératif	Gagnant/Perdant
Collaborer	Assertif et coopératif	Gagnant/Gagnant
Chercher un compromis	Partiellement assertif et coopératif	Chacun gagne et perd
Eviter	Non assertif et non coopératif	Perdant/Perdant
Céder	Non assertif et non coopératif	Perdant/Gagnant

Source : [www.diagnostics.com](http://www.diagnostics.com)

Thomas-Kilmann Conflict Mode Instrument



## Eviter la fatigue face aux changements

La fatigue face au changement est un sentiment général d'apathie ou de résignation passive envers les changements organisationnels par des individus ou des équipes.

- Voir la résistance au changement comme normale
  - Ecoute, empathie
- Communiquer une vision d'ensemble
  - Expliquer la raison de ce changement
  - Montrer comment les changements sont liés
  - Lier les changements aux stratégies et aux objectifs de l'entreprise
- S'assurer que chaque initiative de changement a un résultat escompté
- Offrir aux gens les moyens de contribuer
- Célébrer les succès (même s'il ne s'agit que de petites réussites)
- Créer des boucles de feedback et d'amélioration visibles

Le degré de fatigue face aux changements ressentie par les personnes est directement liée à la façon dont le changement est géré.

## Valoriser les nouveaux comportements

- Améliorer les pratiques de communication et de collaboration et les outils partagés
- Créer un vocabulaire commun
- Observation en milieu de travail
- Compétences transverses
- Expériences d'immersion
- Team building
- Communautés de pratique
- DevOps Days internes
- Game days (hackathons)
- Simulations
- Partage d'idées style médias sociaux et résolution de problèmes

Le partage entre pairs, organisations et industries est un facteur crucial de la croissance et de l'acceptation de DevOps.

## ETUDE DE CAS : Target

« Quand nous avons demandé la permission, on nous a dit non, mais nous l'avons quand même fait parce que nous savions que nous devions le faire. Nous avons organisé des hackathons sur les outils parallèlement à nos événements internes DevOpsDays et nous avons organisé une tonne de rencontres. Nous avons organisé 6 événements internes DevOpsDays. »

“Nous sommes une entreprise technologique”



Heather Mickman,  
Responsable technologie  
de transformation

Ross Clanton,  
Responsable technique

### Avantages

- Mise en place de changements structurels apportant un soutien ascendant, puis descendant
- Convergence des efforts agiles et DevOps
- Utilisation de la formation, du coaching et des expériences immersives - Dojo énorme !
- Construit une stack complète en quelques minutes au lieu de 3-6 mois
- Création d'un environnement d'empathie et compréhension

## Module 5 : Quiz (1)

1. Qui a dit “ La culture ne fait qu'une bouchée de toute stratégie ” ?
  - a) Peter Drucker
  - b) Gene Kim
  - c) Damon Edwards
  - d) Bill Gates
  
2. Qu'est-ce que vous ne pouvez pas changer ?
  - a) Comportement
  - b) Habitudes
  - c) Culture
  - d) Systèmes

## Module 5 : Quiz (2)

3. Quelle est l'une des caractéristiques de la culture DevOps ?

- a) Accusation
- b) Méfiance
- c) Peur
- d) Courage

4. Que se passe-t-il dans une culture pathologique ?

- a) Les messagers sont fusillés
- b) Les liens directs sont partagées
- c) Un échec lance une investigation
- d) Les innovations sont mises en oeuvre

## Module 5 : Quiz (3)

5. Qu'est-ce que Target n'a pas fait ?

- a) Organiser des hackathons
- b) Mettre en place des Dojos
- c) Obtenir la permission
- d) Alimenter l'empathie

## Module 5 : Réponses au quiz

- 1 Qui a dit " La culture ne fait qu'une bouchée de toute stratégie " ?  
a) Peter Drucker  
b) Gene Kim  
c) Damon Edwards  
d) Bill Gates
  
- 2 Que ne pouvez-vous pas changer ?  
a) Comportement  
b) Habitudes  
c) Culture  
d) Systèmes
  
- 3 Quelle est l'une des caractéristiques de la culture DevOps ?  
a) Accusation  
b) Méfiance  
c) Peur  
d) Courage
  
- 4 Que se passe-t-il dans une culture pathologique ?  
a) Les messagers sont fusillés  
b) Les liens directs sont partagées  
c) Un échec lance une investigation  
d) Les innovations sont mises en oeuvre
  
- 5 Qu'est ce que Target n'a pas fait ?  
a) Organiser des hackathons  
b) Mettre en place des Dojos  
c) Obtenir la permission  
d) Alimenter l'empathie

# MODULE 6

## Automatisation et architecture des DevOps Toolchains

## Module 6 : automatisation et architecture des DevOps Toolchains

- L'automatisation
- Terminologie importante
- Architecture
- Pratiques, communication, amélioration
- DevOps Toolchains

Composant	Module 6 : contenu
Vidéo	The DevOps Toolchain with John Okoro
Etude de cas	Fannie Mae
Discussion	Application de la définition du manuel DevOps
Exercice	Faire l'architecture de votre DevOps Toolchain



## Module 6 : automatisation et architecture des DevOps Toolchains

"DevOps ne se résume pas à l'automatisation de la même façon que l'astronomie ne se résume pas aux télescopes. »

Christopher Little, The DevOps Handbook

## Module 6 : automatisation et architecture des DevOps Toolchains

PERIODIC TABLE OF DEVOPS TOOLS (V3)																		2	En	
1	Os																	2	En	
<b>G1</b> GitHub																		<b>Sp</b> Splunk		
5	Fm	4	En																	
<b>Gh</b> GitHub		<b>Dt</b> Datacal																		
11	Os	12	En																	
<b>Sv</b> Subversion		<b>Db</b> DBMaestro																		
19	En	20	En	21	Os	22	Fm	23	Os	24	Fr	25	Fr	26	Fm	27	En	28	Fr	
<b>Cw</b> ISFW		<b>Dp</b> Dolphix		<b>Jn</b> Jenkins		<b>Cs</b> Codeastic		<b>Fn</b> FitNesse		<b>Ju</b> JUnit		<b>Ka</b> Karma		<b>Su</b> SoapUI		<b>Ch</b> Chef		<b>Tf</b> Terraform		<b>XLd</b> Xebialabs XL Deploy
37	Pd	38	Fm	39	Pd	40	Fm	41	Pt	42	Fp	43	Os	44	Pd	45	En	46	Os	
<b>At</b> Artifactory		<b>Rg</b> Redgate		<b>Ba</b> Bamboo		<b>Vs</b> VETS		<b>Se</b> Selenium		<b>Jm</b> JMeter		<b>Ja</b> Jasmine		<b>Sl</b> SauceLabs		<b>An</b> Ansible		<b>Ru</b> Rudder		<b>Oc</b> Octopus Deploy
55	Pd	56	Os	57	Os	58	Fm	59	Os	60	Fr	61	Fm	62	Pd	63	En	64	Os	
<b>Nx</b> Nexus		<b>Fw</b> Flyway		<b>Tr</b> Travis CI		<b>Tc</b> TeamCity		<b>Ga</b> Gatling		<b>Tn</b> Tentacle		<b>Pe</b> Perfecto		<b>Pu</b> Puppet		<b>Pa</b> Packer		<b>Cd</b> AWS CodeDeploy		<b>Ec</b> ElectricCloud
73	Fm	74	En	75	Fm	76	Pd	77	Fr	78	Os	79	Os	80	En	81	Os	82	Os	
<b>Bb</b> BitBucket		<b>Pf</b> Perforce		<b>Cr</b> CircleCI		<b>Cb</b> AWS CodeBuild		<b>Cu</b> Cucumber		<b>Mc</b> Mocha		<b>Lo</b> Locust.io		<b>Mf</b> Micro Focus UFT		<b>Sa</b> Salt		<b>Ce</b> CFEngine		<b>Eb</b> ElasticBox
91	En	92	Os	93	Fm	94	En	95	En	96	Fm	97	Os	98	Os	99	Os	100	En	
		<b>XLi</b> Xebialabs XL Impact		<b>Ki</b> Kibana		<b>Nr</b> New Relic		<b>Dt</b> Dynatrace		<b>Dd</b> Datadog		<b>Ad</b> AppDynamics		<b>Ei</b> ElasticSearch		<b>Ni</b> Nagios		<b>Zb</b> Zabbix		<b>Zn</b> Zenoss
106	En	107	Pd	108	Fm	109	Fm	110	Fm	111	En	112	En	113	En	114	Pd	115	Pd	
		<b>Sw</b> ServiceNow		<b>Jr</b> Jira		<b>Tl</b> Trello		<b>Sk</b> Slack		<b>St</b> Stride		<b>Cn</b> CollabNet VersionOne		<b>Ry</b> Remedy		<b>Ac</b> Agile Central		<b>Og</b> OpsGenie		<b>Pd</b> PagerDuty
116	Os	117	Fm	118	En	119	En	120	En											
		<b>Sn</b> Smart		<b>Tw</b> Tripwire		<b>Ck</b> CyberArk		<b>Vc</b> Voracode		<b>Bd</b> Black Duck		<b>Sr</b> SonarQube		<b>Hv</b> HashiCorp Vault		<b>Ff</b> Fortify SCA				



Follow @xebialabs

<https://xebialabs.com/periodic-table-of-devops-tools>

## Avantages de l'automatisation

L'automatisation permet :

- Délais plus courts
- Mises en production plus fréquentes
- Mises en production moins houleuses
- Moins d'erreurs
- Meilleure qualité
- Sécurité améliorée et atténuation des risques
- Rétablissements plus rapides
- Satisfaction de l'entreprise et des clients

L'automatisation donne les tâches de routine à l'ordinateur et permet aux individus de :

- Analyser les preuves
- Résoudre des problèmes
- Prendre des décisions en fonction du feedback
- Utiliser leurs compétences, leur expérience et leur jugement

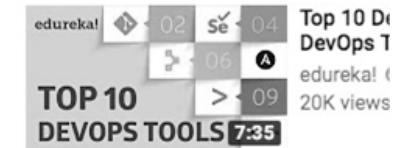
***“Vos outils seuls ne vous feront pas réussir.”***

Patrick Debois

Up next



How Netf  
Coding Te  
58K views



Top 10 De  
DevOps T  
edureka! (20K views)



What is D  
English  
Rackspac  
891K view  
Agile & D  
John Okor  
59K views

## SOURCE CONTROL / VERSIONING



## CONTINUOUS INTEGRATION



DRONE.IO, JENKINS,  
TEAM CITY, TRAVIS CI

ES



GIT, SVN, TFS, CVS,



## DEVELOPMENT ENVIRONMENT



## CONFIGURATION MANAGEMENT



The DevOps Toolchain  
with John Okoro (7:43)



Published on May 28, 2016

SUBSCRIBED 713

5:38

# Terminologie importante

- **Artefact**

Tout élément d'un projet de développement logiciel, y compris la documentation, les plans de test, les images, les fichiers de données et les modules exécutables

- **Interface de programmation d'application (API)**

Ensemble de protocoles utilisés pour créer des applications pour un système d'exploitation spécifique ou comme interface entre des modules ou des applications

- **Microservices**

Une architecture logicielle composée de modules plus petits qui interagissent via des API et peut être mise à jour sans affecter l'ensemble du système.

Ceci est connu comme un couplage lâche

DevOps Institute © Tous droits réservés

- **Virtualisation de système d'exploitation (OS)**

Une méthode pour diviser un serveur en plusieurs partitions appelées "conteneurs" ou "environnements virtuels" afin d'empêcher les applications d'interférer les unes avec les autres

- **Conteneurs**

Une façon de regrouper des logiciels dans des packages légers, autonomes et exécutables, incluant tout le nécessaire pour l'exécuter (code, exécution, outils système, bibliothèques système, paramètres) aux fins de développement, d'envoi et de déploiement.

- **Open source**

Logiciel distribué avec son code source afin que les organisations d'utilisateurs finaux et les fournisseurs puissent le modifier à leurs propres fins

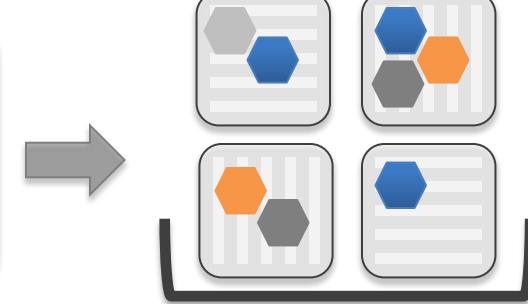
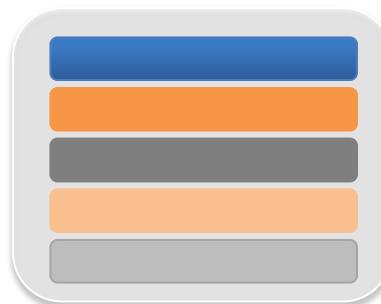
- **Machine learning**

Analyse de données utilisant des algorithmes qui apprennent des données

## Cloud, conteneurs et microservices



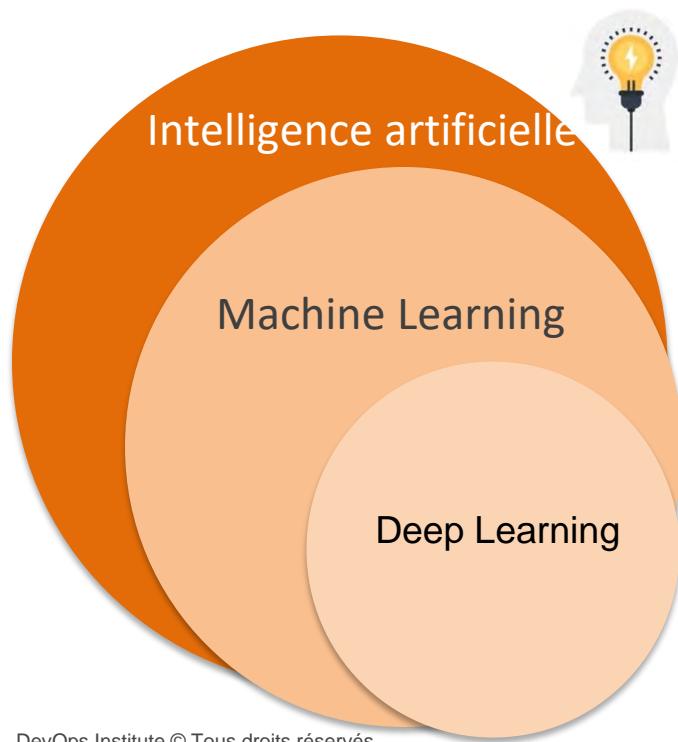
Les équipes qui adoptent les caractéristiques essentielles du Cloud ont 24 fois plus de chances d'être des élites.



**Kubernetes** est un système open source permettant de gérer des applications conteneurisées sur plusieurs hôtes, fournissant des mécanismes de base pour le déploiement, la maintenance et la mise à l'échelle des applications

# IA & Machine Learning

Donner aux ordinateurs la possibilité "d'apprendre" avec des données, sans être explicitement programmé.



- Les organisations collectent plus de données que jamais
- Il est difficile d'extraire pleinement la valeur de ces données.
- La science des données est une discipline de plus en plus populaire
- L'intelligence artificielle et l'apprentissage automatique permettent une analyse prédictive
- Peut trouver des tendances et des corrélations que les humains n'auraient pu trouver
- Augmente la contribution humaine
- Augmente la productivité
- Boucles de feedback automatisées

Définition : Analyse de données qui utilise des algorithmes pour apprendre des données.

## Pratiques d'automatisation DevOps

Une philosophie de toolchain implique l'utilisation d'un ensemble intégré d'outils complémentaires spécifiques aux tâches pour automatiser les processus de livraison et de déploiement de bout en bout.

- Toolchain (par opposition à une solution à fournisseur unique)
- Outils partagés
- En libre service
- Prévoir l'architecture du logiciel de manière à permettre :
  - Automatisation des tests
  - Surveillance
- Infrastructure en tant que code
- Expérimentation

Eviter les outils qui renforcent les silos !

## L'automatisation de la communication et de la collaboration

Des outils et des plates-formes innovants facilitent et accélèrent la communication et la collaboration à travers le spectre Dev et Ops.

Comment	Outils
<ul style="list-style-type: none"><li>• Émettre des alertes et des alarmes</li><li>• Améliorer la réponse</li><li>• Fournir des aperçus de mises à jour de statut</li><li>• Améliorer le flux de travail</li><li>• Améliorer le flux d'informations</li><li>• Permettre la collaboration virtuelle</li><li>• Permettre l'auto-organisation d'équipe cross-fonctionnelle par le partage de compétences.</li></ul>	<ul style="list-style-type: none"><li>• Plateformes de communication</li><li>• Tableaux de bord</li><li>• Tableaux Kanban</li><li>• Groupes de discussion (ChatOps)</li><li>• Flux de travail et outils de gestion de projet</li><li>• Partage de documents</li><li>• Wikis et systèmes de gestion des connaissances</li><li>• Outils ITSM</li><li>• Outils sociaux</li><li>• Backlogs partagés</li></ul>

## Premières étapes pour améliorer l'automatisation DevOps

- Définir l'architecture avant d'automatiser
- Évaluer vos outils existants et vos capacités d'automatisation
- Simplifier d'abord – ne pas automatiser les processus défectueux
- Identifier les lacunes critiques
- Rechercher des fournisseurs qui peuvent répondre à vos exigences
- Automatiser le travail à haute valeur ajoutée, répétitif et sujet aux erreurs
- Optimiser les goulets d'étranglement du flux de travail et la communication
- Améliorer les pratiques de surveillance et de notification automatisées
- S'attendre à un processus itératif - votre toolchain évoluera avec le temps

Ne sous-estimez pas l'effort et le coût de la construction des toolchains à partir d'applications open source.  
L'open source n'est pas nécessairement gratuit. Cela signifie que vous pouvez modifier la source pour répondre à vos besoins.

## ETUDE DE CAS : Fannie Mae

« Nous avons choisi une mise en production application par application. Cela fait maintenant environ un an et demi et nous avons atteint un point de masse critique. Nous pouvons être beaucoup plus flexibles, beaucoup plus dynamiques et fournir à nos clients et partenaires les outils dont ils ont besoin pour interagir beaucoup plus facilement avec nous. Comme tout le monde, nous devons mettre les idées en production beaucoup plus rapidement que ce que nous faisons aujourd'hui. »

“Nous faisons en sorte qu'il soit facile de travailler avec nous”



Jason Anders,  
Responsable IT  
virtualisation

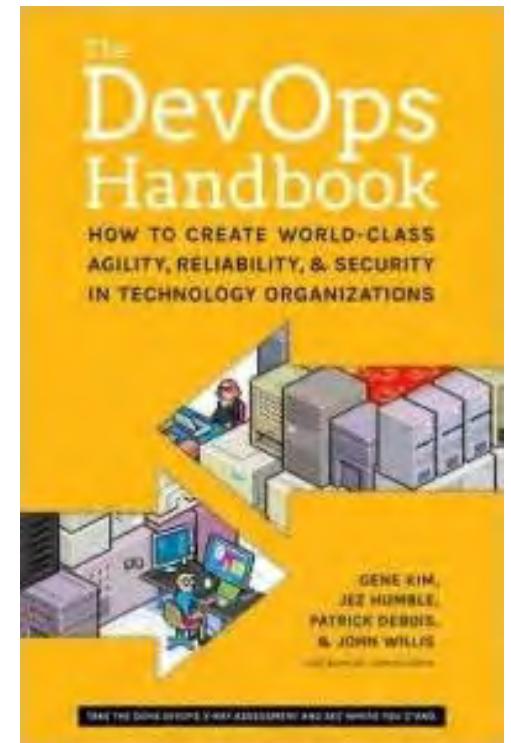
### Avantages

- Déploiement de 3 jours réduit à 45 minutes
- Déployé sept ou huit fois par jour
- 40 à 75% d'économie sur les coûts de stockage grâce à la virtualisation des données

# DevOps Toolchains

« L'un des axes afin d'obtenir des résultats rapides pilotés par le marché consiste, pour les opérations, à créer des plates-formes et des outils pouvant être utilisés par toute les équipes de développement afin de devenir plus productifs... une plateforme qui s'appuie sur un système de gestion de version de référence, associé à des bibliothèques de sécurité vérifiées, un pipeline de déploiement qui exécute automatiquement des outils d'analyse de sécurité et de qualité du code, qui déploie nos applications dans des environnements connus et performants sur lesquels des outils de surveillance de la production sont déjà installés. »

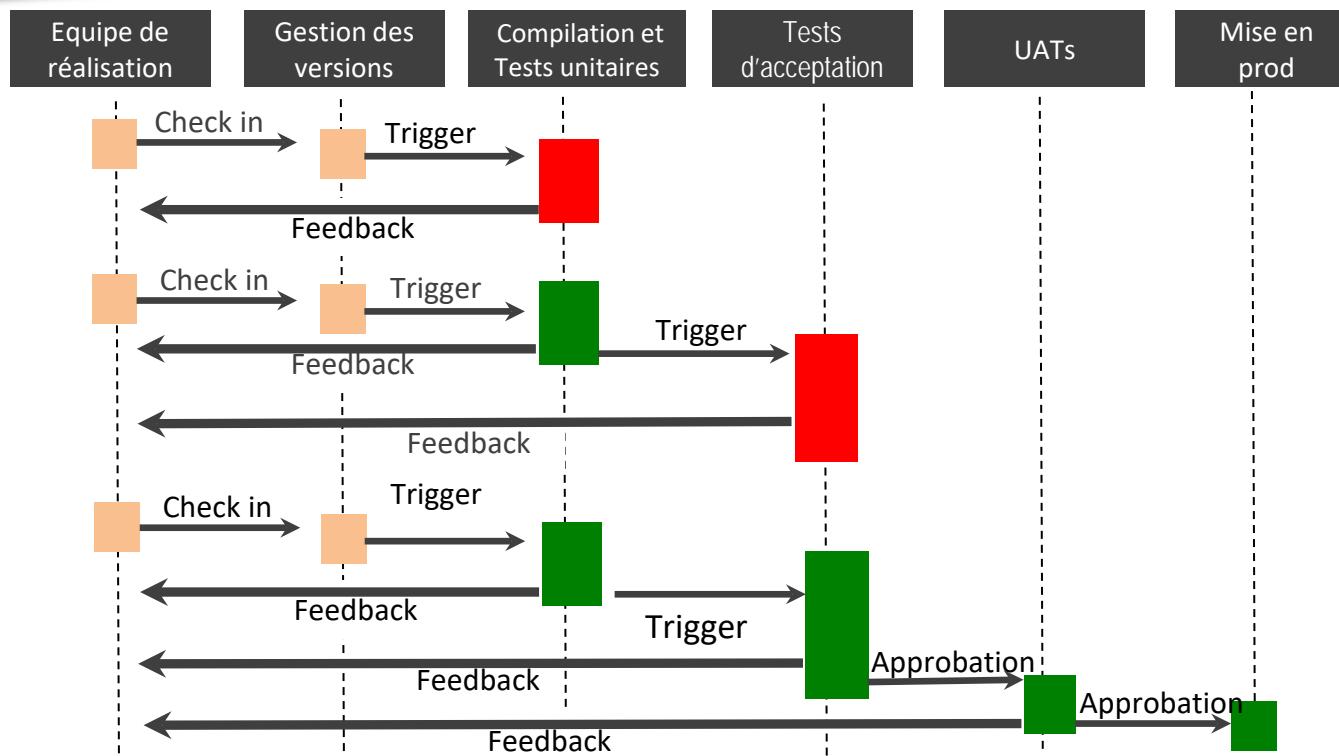
Le DevOps Handbook



# DISCUSSION

## Application de la définition du manuel DevOps

## Le pipeline de déploiement



Le pipeline de déploiement est un processus automatisé permettant de gérer tous les changements, du check-in à la mise en production. Les toolchains couvrent des silos et automatisent le pipeline de déploiement.

Source : *Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation*

## DevOps Toolchains

La DevOps toolchain est composée des outils nécessaires pour prendre en charge l'intégration continue, la livraison continue ainsi que de mise en production continue et les opérations. (Gartner)

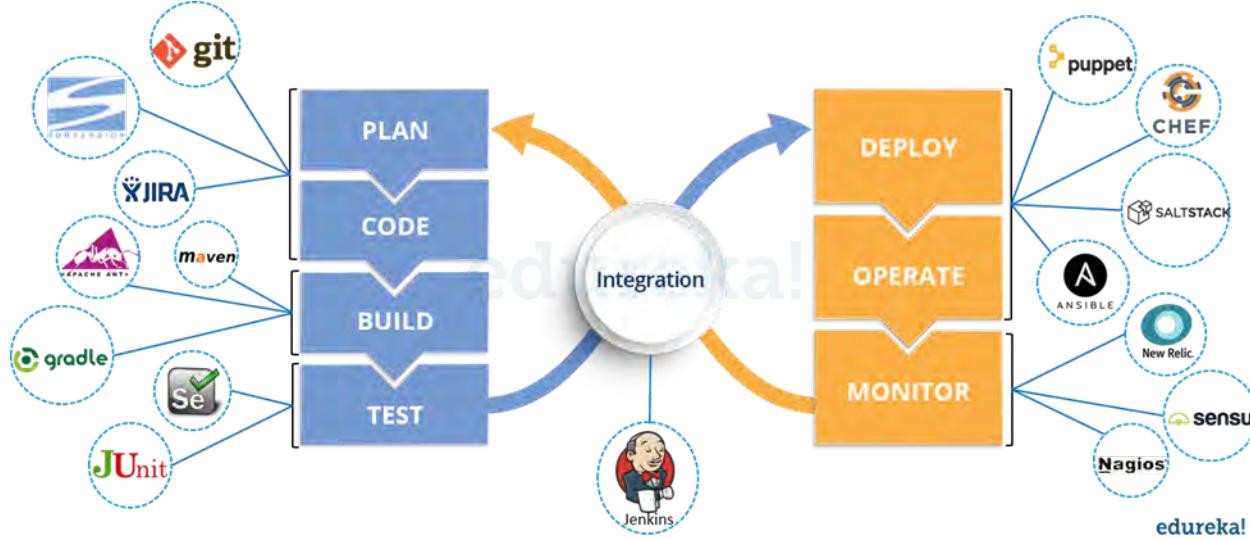
- Les toolchains automatisent les tâches dans le pipeline de déploiement
- Chaque élément de la toolchain sert un objectif spécifique
- Les applications dans les toolchains sont connectées via des API
- Elles ne doivent pas nécessairement être homogènes ou provenir d'un seul fournisseur
- Les toolchains sont généralement construites autour d'écosystèmes sources ouverts
- Exiger une conception architecturale pour assurer l'interopérabilité et la cohérence

Le pipeline de déploiement est un processus automatisé permettant de gérer tous les changements, du check-in à la mise en production. Les toolchains embrassent des silos et automatisent le pipeline de déploiement.

Comment les DevOps toolchains doivent-elles s'interfacer avec des outils opérationnels tels que des applications de surveillance ou de support ?

## Exemple DevOps Toolchain (gouvernement américain - GSA)

Il existe de nombreux outils DevOps open-source ou non avec des écosystèmes dynamiques.



La manière dont ces outils sont adaptés et intégrés dans votre pipeline de déploiement déterminera leur valeur.

## Eléments dans une DevOps Toolchain

- Le pipeline de déploiement divise le cycle de vie de livraison du logiciel en étapes logiques
- Chaque étape fournit :
  - L'occasion de vérifier la qualité des nouvelles fonctionnalités sous un angle différent
  - Du feedback rapide à l'équipe
  - Une visibilité quant au flux des changements
- Les DevOps toolchains fournissent les fonctionnalités nécessaires pour automatiser et accélérer chaque étape

### Eléments Toolchain typiques :

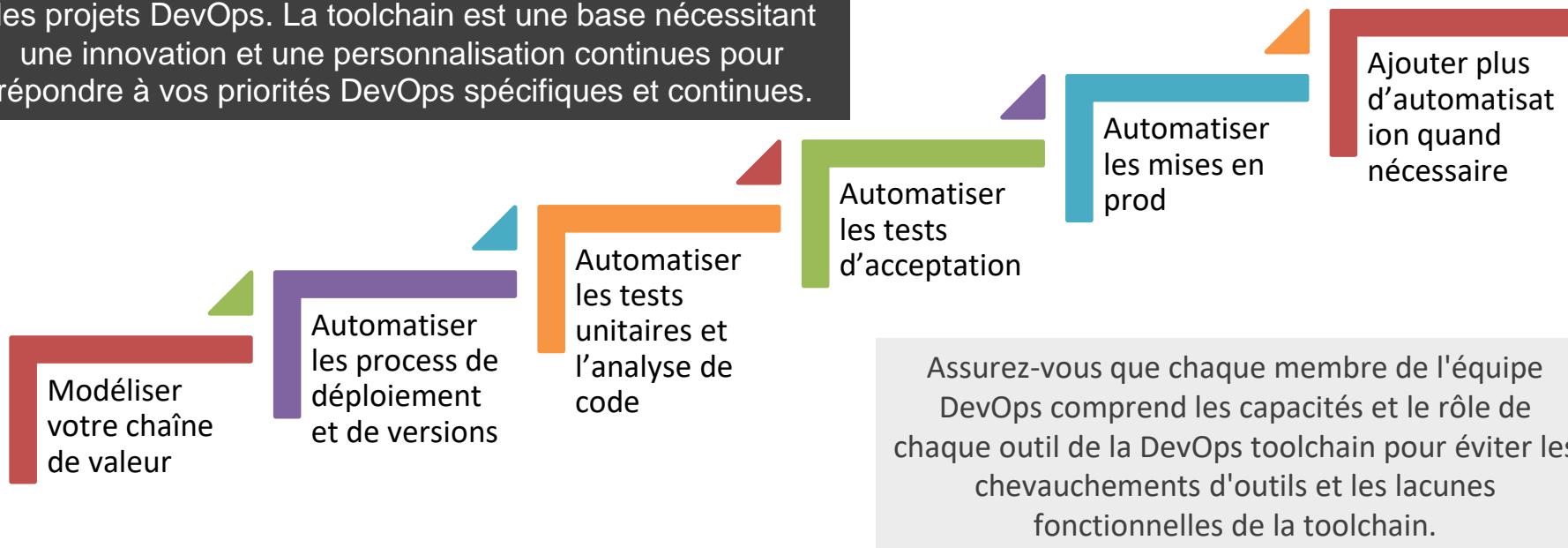
- Gestion des exigences
- Orchestration et visualisation
- Gestion des versions
- Intégration continue et versions
- Gestion des artefacts
- Virtualisation de conteneurs et de systèmes d'exploitation
- Automatisation des tests et de l'environnement
- Configuration et déploiement du serveur
- Gestion de la configuration du système
- Alertes et alarmes
- Monitoring

# EXERCICE

## Définir l'architecture de votre DevOps Toolchain

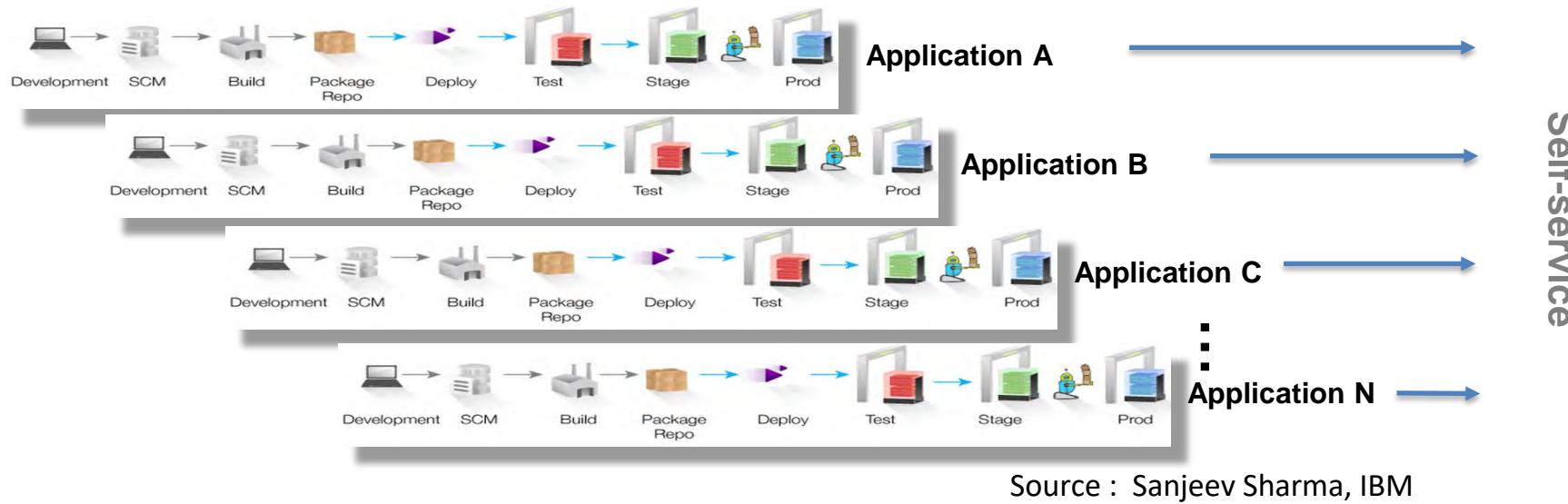
## Construire votre DevOps Toolchain petit à petit

Ne créez pas une toolchain définitive qui s'applique à tous les projets DevOps. La toolchain est une base nécessitant une innovation et une personnalisation continues pour répondre à vos priorités DevOps spécifiques et continues.



Source : Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation par Jez Humble et Dave Farley

## Des applications professionnelles multiples nécessitent des toolchains multiples



Éviter de créer davantage de silos entre les pipelines en adoptant une approche d'architecture d'entreprise

## Module 6 : Quiz (1)

1. DevOps ne se résume pas à l'automatisation de la même façon que l'astronomie ne se résume aux...
  - a) Télescopes
  - b) Etoiles
  - c) Soleil
  - d) Navettes spatiales
  
2. Qui est à l'origine du tableau périodique des outils DevOps ?
  - a) Chef
  - b) Google
  - c) IT Revolution
  - d) Xebia Labs

## Module 6 : Quiz (2)

3. Fannie Mae a réduit le temps de déploiement à 45 minutes. Il était de combien au préalable ?
  - a) 3 heures
  - b) 30 heures
  - c) 3 jours
  - d) 30 jours
4. Qui est le mieux placé pour fournir des DevOps toolchains en tant que service partagé ?
  - a) Le développement
  - b) Le commercial
  - c) L'Infosec
  - d) Les opérations IT

## Module 6 : Quiz (3)

5. Patrick Debois a dit : "Vos outils seuls n'assureront pas votre..."
  - a) Profitabilité
  - b) Compétitivité
  - c) Réussite
  - d) Place de leader sur le marché

## Module 6 : Réponses au quiz

- 1 DevOps ne se résume pas à l'automatisation de la même façon que l'astronomie ne se résume aux...  
**a) Télescopes**  
b) Etoiles  
c) soleil  
d) Navettes spatiales
  
- 2 Qui est à l'origine du tableau périodique des outils DevOps ?  
**d) Xebia Labs**  
a) Chef  
b) Google  
c) IT Revolution
  
- 3 Fannie Mae a réduit le temps de déploiement à 45 minutes. Il était de combien au préalable ?  
**c) 3 jours**  
a) 3 heures  
b) 30 heures  
d) 30 jours
  
- 4 Qui est le mieux placé pour fournir des DevOps toolchains en tant que service partagé ?  
**d) Les opérations IT**  
a) Le développement  
b) Le commercial  
c) L'Infosec
  
- 5 Patrick Debois a dit : "Vos outils seuls n'assureront pas votre..."  
**c) Réussite**  
a) Profitabilité  
b) Compétitivité  
d) Place de leader sur le marché

# MODULE 7

# Mesures, indicateurs et reporting

## Module 7 : Mesures, indicateurs et reporting

- L'importance de mesurer
- Indicateurs DevOps :
  - Vitesse
  - Qualité
  - Stabilité
  - Culture
  - Temps d'attente
- Lignes directrices

Composant	Module 7 : Contenu
Vidéo	4 indicateurs DevOps pour améliorer les performances de livraison
Etude de cas	Société Générale
Discussion	Indicateurs utilisés aujourd'hui
Exercice	Les indicateurs les plus significatifs

## L'importance de mesurer



Première voie	Deuxième voie	Troisième voie
Flux	Feedback	Expérimentation et apprentissage continu
<ul style="list-style-type: none"> <li>Délai de mise en place des changements</li> <li>Temps de cycle des changements</li> <li>Time to value</li> <li>Mise en place de valeur</li> </ul>	<ul style="list-style-type: none"> <li>Résultats de version/test</li> <li>Taux d'échec des changements</li> <li>Monitoring</li> <li>% de retravail / achevé et correct</li> </ul>	<ul style="list-style-type: none"> <li>Journal d'hypothèses</li> <li>Temps alloué</li> <li>Temps passé</li> <li>Maîtrise réalisée et rapportée</li> </ul>
Les mesures nous permettent de trouver des contraintes, de justifier leur suppression et de surveiller les améliorations.	Les preuves créent la confiance et permettent de faire plus – en misant sur l'expérimentation	Les hypothèses nécessitent des résultats quantifiables pour déterminer la prochaine expérience



“Si vous ne  
pouvez pas le  
mesurer vous ne  
pouvez pas  
l'améliorer”

Peter Drucker

The image shows a YouTube video player interface. The main video frame displays a slide from a presentation. The slide features three circular icons representing 'SOFTWARE DEVELOPMENT' (with a code editor icon), 'SOFTWARE DEPLOYMENT' (with a deployment icon), and 'SERVICE OPERATION' (with a server icon). Below these icons are four rectangular boxes labeled 'Lead Time', 'Change Fail', 'Availability', and 'Deployment Frequency'. A horizontal bar at the bottom of the slide is labeled 'FOUR KEY METRICS'. To the right of the slide, a woman in a dark suit is shown speaking on stage. The stage backdrop includes logos for 'OPEN SOURCE SUMMIT NORTH AMERICA' and 'Embedded Linux Conference North America'. On the top right of the YouTube interface, there is an 'AUTOPLAY' button and a section titled 'Up next' showing other video thumbnails.

Up next

AUTOPLAY

Jez Humble - Building and Scaling High Performing...  
Prezi Conference Team  
1.2K views  
44:58

Visual Basic .Net : Search in Access Database -...  
iBasskung  
Recommended for you  
24:59

GOTO 2015 • Agile is Dead • Pragmatic Dave Thomas  
GOTO Conferences  
704K views  
40:39

DebConf 14: QA with Linus Torvalds  
Plen...  
Sessi...  
40:39

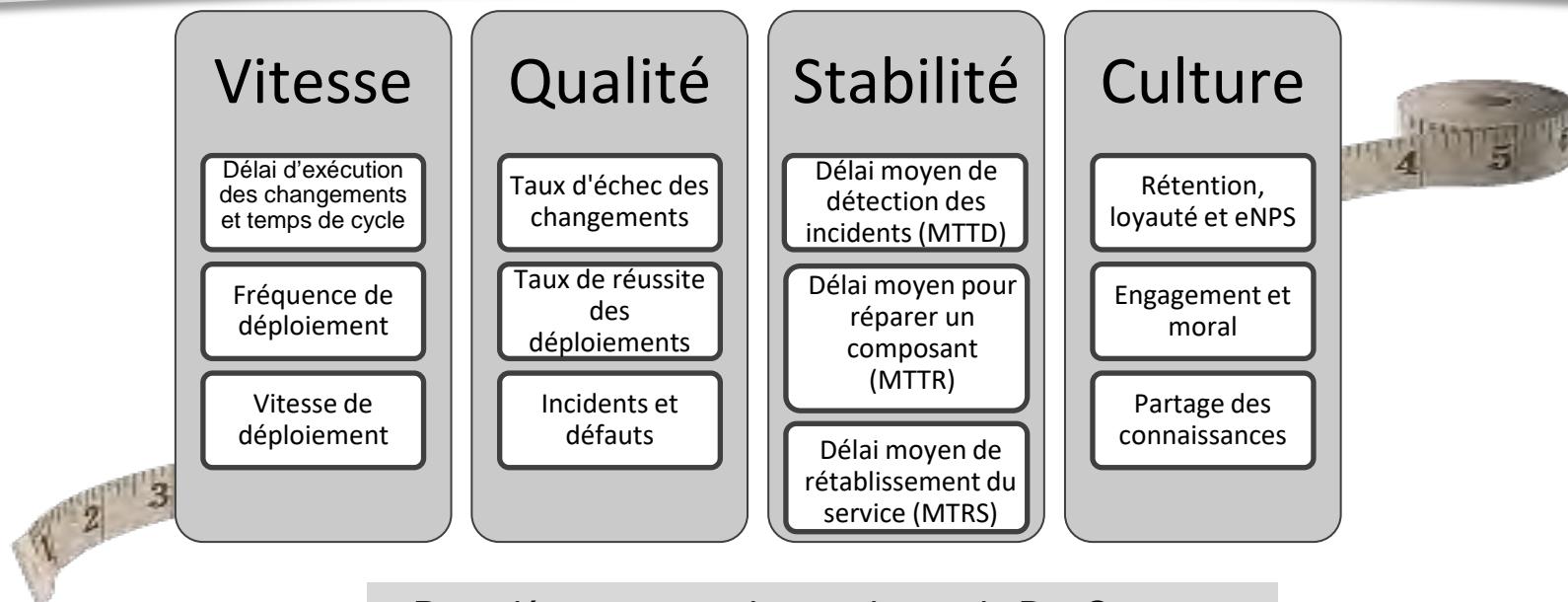
Double the Awesome  
with Dr. Nicole Forsgren (21:46)

The Linux Foundation  
113K subscribers

SUBSCRIBE

36:30

## Mesurer la réussite



Pour démontrer que les pratiques de DevOps sont bénéfiques pour l'entreprise, il est nécessaire d'examiner les facteurs qui influent sur les performances informatiques globales.

Adapté de  
Splunk 2016

# DISCUSSION

## indicateurs utilisés aujourd’hui

## Délais de mise en place / Temps de cycle

Délai de mise en place (Lead time)	Temps de cycle
Temps total écoulé à partir du moment où une user story entre dans le backlog, jusqu'à son heure d'achèvement, y compris le temps d'attente dans le backlog.	Temps qu'il faut pour qu'une histoire passe de « En cours » à Fait.

Le temps d'attente est le délai de mise en place moins le temps de cycle

Source : Accelerate: Dr Nicole Forsgren, Jez Humble & Gene Kim

## Lignes directrices pour mesurer les performances IT

NE PAS MESURER	MESURER
Production, productivité	Résultats, valeur
Maturité	Capacité
Lignes de code, vitesse, utilisation	Délais de mise en place des livraisons, fréquence de déploiement, délais de rétablissement des services, taux d'échec des changements
Individuel ou local	Equipe ou global

### 4 types de travail (IT) :

- Projets business
- Projets IT
- Travail planifié
- Travail non planifié

Accelerate: Dr Nicole Forsgren, Jez Humble & Gene Kim

## ETUDE DE CAS : Société Générale

« Il est important d'établir deux ensembles d'indicateurs. Le premier est la transformation elle-même. En d'autres termes, vous devez mesurer la vitesse à laquelle vous avancez vers la transformation. Le deuxième indicateur concerne la valeur commerciale : quel est le délai de commercialisation de l'idée, depuis l'idée à la production, y compris la vitesse et la qualité du sprint ? »



Carlos Gonsalves, Global  
Chief Technology

« Le retour sur investissement (ROI) de l'effort est extrêmement important pour les autres membres de l'organisation. »

### Avantages

- Passage d'une approche basée sur une cascade à forte charge de travail
- Transformé une base d'utilisateurs insatisfaits
- La livraison continue a vu :
  - Réduction de 45% du délai de mise sur le marché
  - 10% d'économie sur leur budget de fonctionnement (considérable)

## Pyramide des indicateurs DevOps de Gartner



# EXERCICE

## Les indicateurs les plus significatifs

## Module 7 : Quiz (1)

1. Que mettent en évidence les preuves ?
  - a) La douleur
  - b) La suspicion
  - c) La confiance
  - d) La rapidité
  
2. A partir de quels rapports le livre 'Accelerate' décrit-il en détail la recherche ?
  - a) Les rapports annuels DevSecOps
  - b) Les rapports State of DevOps
  - c) Les rapports Meilleurs postes aux États-Unis
  - d) Les rapports Outils DevOps les plus populaires

## Module 7 : Quiz (2)

3. Peter Drucker a déclaré : "Si vous ne pouvez pas le mesurer, vous ne pouvez pas ..."
  - a) Le voir
  - b) Le compter
  - c) L'améliorer
  - d) Le sentir
4. Lequel de ces points doit être mesuré ?
  - a) Maturité
  - b) Capacité
  - c) Production
  - d) Lignes de code

## Module 7 : Quiz (3)

5. Comment est calculé le temps d'attente ?
  - a) Délai de mise en place moins le temps de cycle
  - b) Durée de cycle moins délai de mise en place
  - c) Cadence de cycle moins vitesse
  - d) Vitesse divisée par durée de cycle

## Module 7: Réponses au quiz

- |   |  |  |
|---|--|--|
| 1 | Que mettent en évidence les preuves ?  | a) La douleur<br>b) La suspicion<br><b>c) La confiance</b><br>d) La rapidité   |
| 2 | A partir de quels rapports le livre 'Accelerate' décrit-il en détail la recherche ?  | a) Les rapports annuels DevSecOps<br><b>b) Les rapports State of DevOps</b><br>c) Les rapports Meilleurs postes aux États-Unis<br>d) Les rapports Outils DevOps les plus populaires      |
| 3 | Peter Drucker a déclaré : "Si vous ne pouvez pas le mesurer, vous ne pouvez pas ..." | a) Le voir<br>b) Le compter<br><b>c) L'améliorer</b><br>d) Le sentir   |
| 4 | Lequel de ces points doit être mesuré ?  | a) Maturité<br><b>b) Capacité</b><br>c) Production<br>d) Lignes de code  |
| 5 | Comment est calculé le temps d'attente ?   | <b>a) Délai de mise en place moins le temps de cycle</b><br>b) Durée de cycle moins délai de mise en place<br>c) Cadence de cycle moins vitesse<br>d) Vitesse divisée par durée de cycle |

# MODULE 8

## Partage, observation et évolution

## Module 8 : partage, observation et évolution

- DevOps Days
- DevOps dans l'entreprise
- Rôles
- Leadership DevOps
- Considérations organisationnelles
- Démarrer
- Défis, risques et facteurs critiques de réussite

Composant	Module 8 : contenu
Vidéo	DevOps: A Culture of Sharing
Etude de cas	Disney
Discussion	Quel est votre sujet Open Space ?
Exercice	Rédigez votre plan d'action personnel à titre expérimental

## DevOps encourage une culture du partage

- Les opportunités d'immersion deviennent de plus en plus disponibles dans le but de fournir aux équipes de DevOps un accès à des coachs spécialisés sur des sujets tels que CI, CD, Lean et de conception
  - Dojos (Internes à la cible)
  - Garages (IBM)
  - Lofts (Amazon)
  - ...
- Les simulations et gamefifications DevOps sont également de plus en plus disponibles

<http://target.github.io/devops/the-dojo>

Les jeux, hackathons, espaces de travail communs, simulations et autres innovations contribuent à encourager le partage d'outils, de connaissances, de découvertes et de leçons apprises.

## Module 8 : Partage, observation et évolution

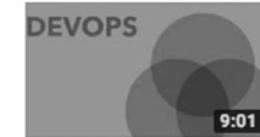
YouTube (2) DevOps: A Culture of Sharing - YouTube - <https://www.youtube.com/watch?v=8cUxWV8sH3c>

youtube.com



Up next

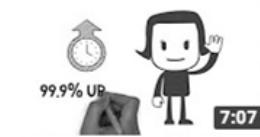
AUTOP



What is DevOps?  
CBT Nuggets  
98K views



The Culture Change of IT  
John Okoro  
2.8K views



What is DevOps? - In Sir English  
Rackspace  
891K views



Charlie Munger Interview 2018



DevOps: A Culture of Sharing  
with Gareth Rushgrove (2:19)



Puppet  
Published on Apr 25, 2016

SUBSCRIBE 12K



Recommended for you

## DevOps Days internes

- Certaines organisations répliquent le modèle DevOps Days lors d'événements internes.
- Les événements DevOps Days donnent aux équipes et aux personnes une opportunité d'apprendre, de partager, de discuter, de dialoguer et de fournir des commentaires et des suggestions

Bien que plus efficace organisés dans un emplacement physique, les DevOps Days internes peuvent être réalisés dans un environnement virtuel.



Le format peut comprendre :

- Présentations traditionnelles de 30 minutes par des ressources internes et externes
- Sessions « Ignite » (5 minutes) sur des sujets spécifiques
- Discussions en open-space sur des sujets suggérés

# DISCUSSION

## Quel sujet souhaiteriez-vous abordé en Open Space ?

Module 8 : Partage, observation et évolution

# DevOps dans l'entreprise

## ETUDE DE CAS : Disney

« Il n'y a aucun secret à la création de la magie numérique. Nous continuons d'avancer, ouvrons de nouvelles portes, faisons plus de choses parce que nous sommes curieux. »

« L'expansion numérique des entreprises signifie plus de travail et d'incendies à éteindre. »



Jason Cox,  
Directeur Ingénierie  
des systèmes

### Avantages

- 30 minutes pour mettre à jour 100 serveurs au lieu de 8 heures
- Moins de dérive du système
- Livrer continuellement et régulièrement
- Réduire de moitié le coût tout en offrant plus (films)

Module 8 : Partage, observation et évolution

# Rôles

## Développer les compétences DevOps

- La demande importante en ressources DevOps crée un turn-over au sein des entreprises
- Le rythme effréné, auquel les technologies évoluent, rend difficile le maintien des compétences actuelles.
- S'assurer que les personnes possèdent les compétences générales requises et sont bien adaptées à la culture ajoute au défi du recrutement

### Stratégies

- Formation et certification
- Programmes d'immersion / coaching
- Restructuration des salaires et culture d'entreprise
- Compléter les équipes internes avec des talents externalisés
- Bonus de recrutement

Aujourd'hui, les DSI recherchent des ressources capables de s'adapter à l'évolution de la technologie.

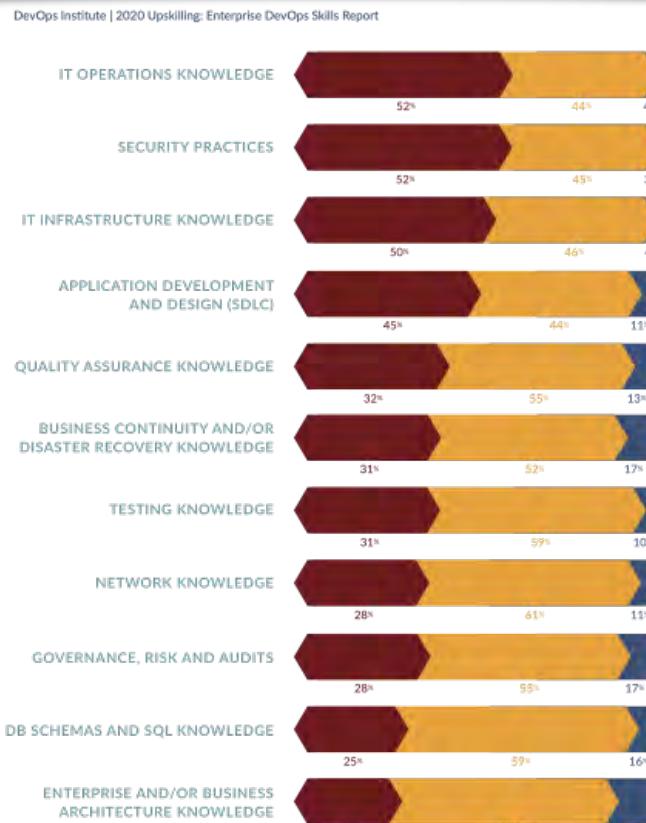
# Compétences en matière d'opérations informatiques, de sécurité et d'infrastructure

L'homme DevOps doit avoir des compétences fonctionnelles qui comprennent les opérations informatiques (ITOps), les pratiques de sécurité, l'infrastructure informatique, le développement et la conception d'applications (AD&D), l'assurance qualité et la continuité des activités/reprise après sinistre (BC/DR)

## UPSKILLING 2020 | Enterprise DevOps Skills Report

DevOps Institute

©DevOps Institute unless otherwise stated



Module 8: Sharing, Shadowing & Evolving

**IT Ops, Security And Infrastructure Skills Are Just As Important As Application Development**

Figure 11: Functional Skills

Q How would you rate the importance of the following functional skills for your DevOps team members?

**Delta from 2019:**  
IT Infrastructure, network, security, GRC, BC/DR and Application Development all gained must-have votes from 2019.

- Very Important (Must-Have Skills)
- Important (Nice-to-Have Skills)
- Not Important (Optional Skills)

# Compétences et caractéristiques d'un professionnel DevOps

## Compétences

- Entreprise - Connaissance des priorités et des processus de l'entreprise
- Techniques - Spécialiste avec de vastes connaissances généralistes (en forme de T)
  - expérience ou au moins un intérêt pour l'écriture de code
- Générales - Communication, collaboration, travail d'équipe
- Autogestion - Initiative, gestion du temps et du stress, motivation personnelle, concentration

## Caractéristiques

- Adaptable
- Centré sur le client
- Artisan
- Curieux
- Axé sur les données
- Engagé
- Empathique
- Transparent

Les connaissances techniques généralistes comprennent une compréhension des pratiques de DevOps, des pratiques modernes d'ingénierie logiciel et des architectures modernes.

## Rôles DevOps

- Evangéliste ou leader DevOps
- Ingénieurs logiciels, développeurs et testeurs
- Gestionnaire des releases
- Architecte automatisation / Livraison continue
- Ingénieur des versions
- Ingénieur sécurité
- Assurance qualité (QA) / Assurance expérience (XA)
- Ingénieur d'exploitation DevOps
- Support IT
- Ingénieur fiabilité des sites
- Agile Service Manager®
- Agile Process owner®

Selon vous, quels autres rôles devraient être impliqués ?

Source : <http://techbeacon.com/7-devops-roles-you-need-succeed>

## Qu'est-ce qu'un ingénieur DevOps ?

- Il n'existe actuellement aucune description de poste ou filière de carrière officielle « reconnue par le secteur » pour un ingénieur DevOps.
- Tout comme pour le concept d'équipe DevOps, le titre a ses avantages et ses inconvénients
- Les caractéristiques générales incluent quelqu'un qui :
  - Veut mettre son talent technique au service d'initiatives pour l'amélioration de l'entreprise et des processus
  - Est à l'aise de collaborer avec d'autres
  - Veut être dans un environnement de travail qui favorise une culture de partage



Module 8 : Partage, observation et évolution

# Leadership DevOps

## Leadership transformationnel

“L’objectif du leadership n’est pas de commander, contrôler, réprimander, intimider et évaluer les collaborateurs au moyen d’un ensemble d’indicateurs artificiels. Le travail des leaders consiste plutôt à aider les organisations à s’améliorer en matière d’auto-diagnostic et d’auto-amélioration, et à s’assurer que les innovations locales peuvent être traduites et converties en améliorations globales.”

Dr Stephen Spear cité par Gene Kim  
dans Beyond the Phoenix Project

### Dimensions of transformational leadership



« Les caractéristiques du leadership transformationnel sont étroitement liées aux performances informatiques et à l'eNPS (employee Net Promoter Score). »

Source : The State of DevOps Report 2017

## Diriger une transformation digitale : selon Jason Cox(Disney)

Ingrédients essentiels	Défis du leadership
<ul style="list-style-type: none"><li>• Collaboration - briser les silos, objectifs mutuels</li><li>• Curiosité - continuer à expérimenter</li><li>• Courage - candeur, défi, sans reproches ni chasse aux sorcières</li></ul>	<ul style="list-style-type: none"><li>• Une directive de management et de contrôle</li><li>• Comment un nouveau leadership peut faire prendre une nouvelle direction à une entreprise</li><li>• La culture du blâme</li></ul>

# Considérations organisationnelles

## Structures organisationnelles DevOps



Certaines organisations :

- Assignent des points de contact Ops aux équipes Dev / Scrum
- Crètent des équipes produits (par opposition à projets) fonctionnelles transversales
- Adoptent des structures matricielles ou axées sur le marché (par opposition à des fonctions)
- Crètent des services Ops partagés prenant en charge plusieurs équipes de développement



Un débat a lieu sur les avantages et les inconvénients des équipes DevOps.

## Equipes DevOps (1)

La création de départements ou d'équipes DevOps est une tendance croissante ; 16% en 2014, 19% en 2015, 22% en 2016. En 2017 et 2018, 27% des personnes ayant répondu à l'enquête State of DevOps ont indiqué qu'elles faisaient partie d'un département DevOps. Elles sont 26% suite à l'enquête 2019 State of DevOps

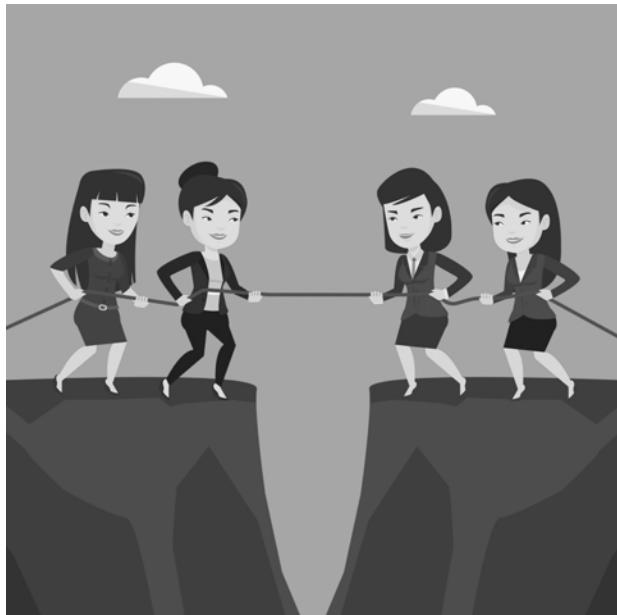
### Les équipes DevOps :

- Développent le concept d'une équipe agile ou Scrum
- Intègrent les compétences en développement et exploitation dans un seul groupe holistique
- Peuvent être temporaires ou dédiées à un produit spécifique
- Peuvent être des « tiger teams » pour des projets à court terme
- Peuvent évoluer pour fournir des services partagés
- Ont des responsabilités partagées
- Devraient adhérer aux normes définies pour le développement, l'automatisation, les risques et la conformité qui s'appliquent à toutes les équipes de DevOp



Il n'existe pas de structure « idéale » pour une équipe DevOps.

## Equipes DevOps (2)



Inconvénients des équipes DevOps dédiées :

- Moins d'engagement dans la chaîne de valeur IT
- Risque d'être un autre silo
- Dev et Ops se lavent les mains des responsabilités
- Les activités DevOps deviennent le problème de quelqu'un d'autre

Quelle que soit la structure, une équipe DevOps doit être homogène, avec un engagement continu et le juste équilibre entre le personnel, les pratiques et les compétences en automatisation.

Module 8 : Partage, observation et évolution

# Démarrer

« C'est un voyage, non une solution miracle, et les dirigeants doivent éviter de se laisser prendre dans la paralysie de l'analyse. Commencez par mettre en œuvre des changements, obtenez des améliorations et laissez l'organisation évoluer. »

Melissa Sargeant



## Démarrez où vous vous trouvez



"DevOps n'est pas votre « pourquoi », ni celui de vos collègues, et certainement pas celui de votre entreprise."

Damon Edwards

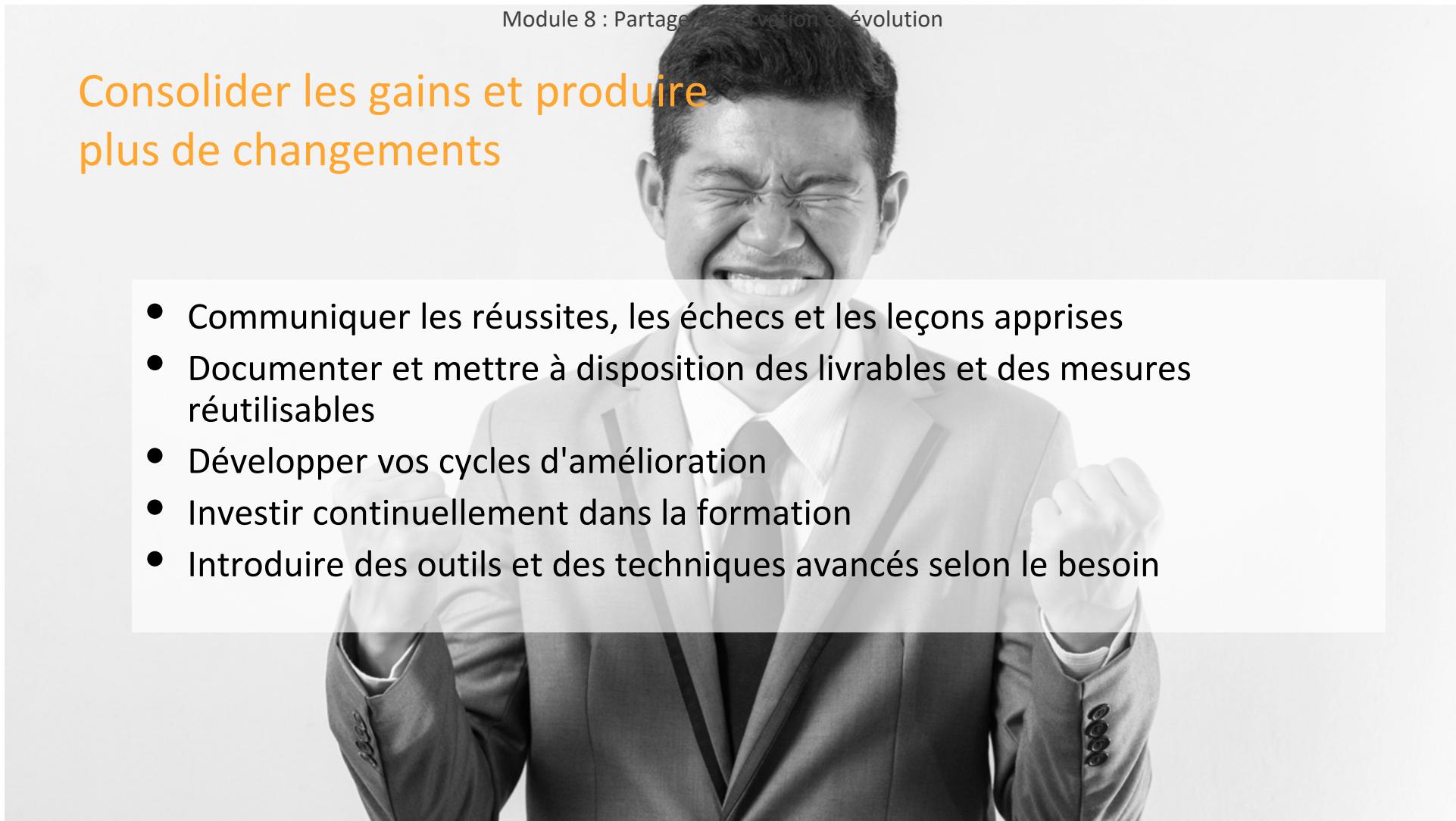
- Être clair sur l'opportunité business - le "Pourquoi?"
- Rassembler les bonnes personnes
- Mettre tout le monde sur la même longueur d'onde
- Investir dans la formation et le développement des compétences
- Construire des capacités qui mènent à un changement durable
- Se concentrer sur les comportements critiques
- Expérimenter et apprendre
- Consolider les gains et produire plus de changement
- Éviter l'inertie

# Apprendre en faisant

- Créer un pilote où vous pouvez maximiser les chances de succès
- Il doit être assez petit pour que :
  - Sa réussite soit évidente et comprise
  - Les conséquences d'un échec ne soient pas si importantes qu'une erreur pourrait bloquer l'ensemble de l'initiative
- Il doit être assez grand pour que :
  - Vous puissiez montrer une preuve d'amélioration
  - Vous obtenez le droit d'apporter des améliorations futures

## Consolider les gains et produire plus de changements

- Communiquer les réussites, les échecs et les leçons apprises
- Documenter et mettre à disposition des livrables et des mesures réutilisables
- Développer vos cycles d'amélioration
- Investir continuellement dans la formation
- Introduire des outils et des techniques avancés selon le besoin



## Ancrer les résultats

- Prouver que la nouvelle façon de faire est meilleure
- Renforcer les nouveaux comportements avec des incitations et des récompenses
- Se préparer à perdre des gens en cours de route
- Renforcer la nouvelle culture avec chaque nouvel employé

*"Le changement persiste quand il devient 'la façon de faire les choses ici'. »*  
John. P. Kotter

# Défis, risques et facteurs déterminants de réussite

## Facteurs determinants de réussite

- Engagement de la direction envers le changement de culture
- Création d'une culture d'apprentissage collaboratif
- Formation et perfectionnement continus des compétences
- Valeurs et vocabulaire communs
- Ingénierie des systèmes englobant Dev et Ops
- Indicateurs significatifs
- Equilibre entre automatisation et interaction humaine
- Application de méthodes Agile et Lean
- Communication ouverte et fréquente



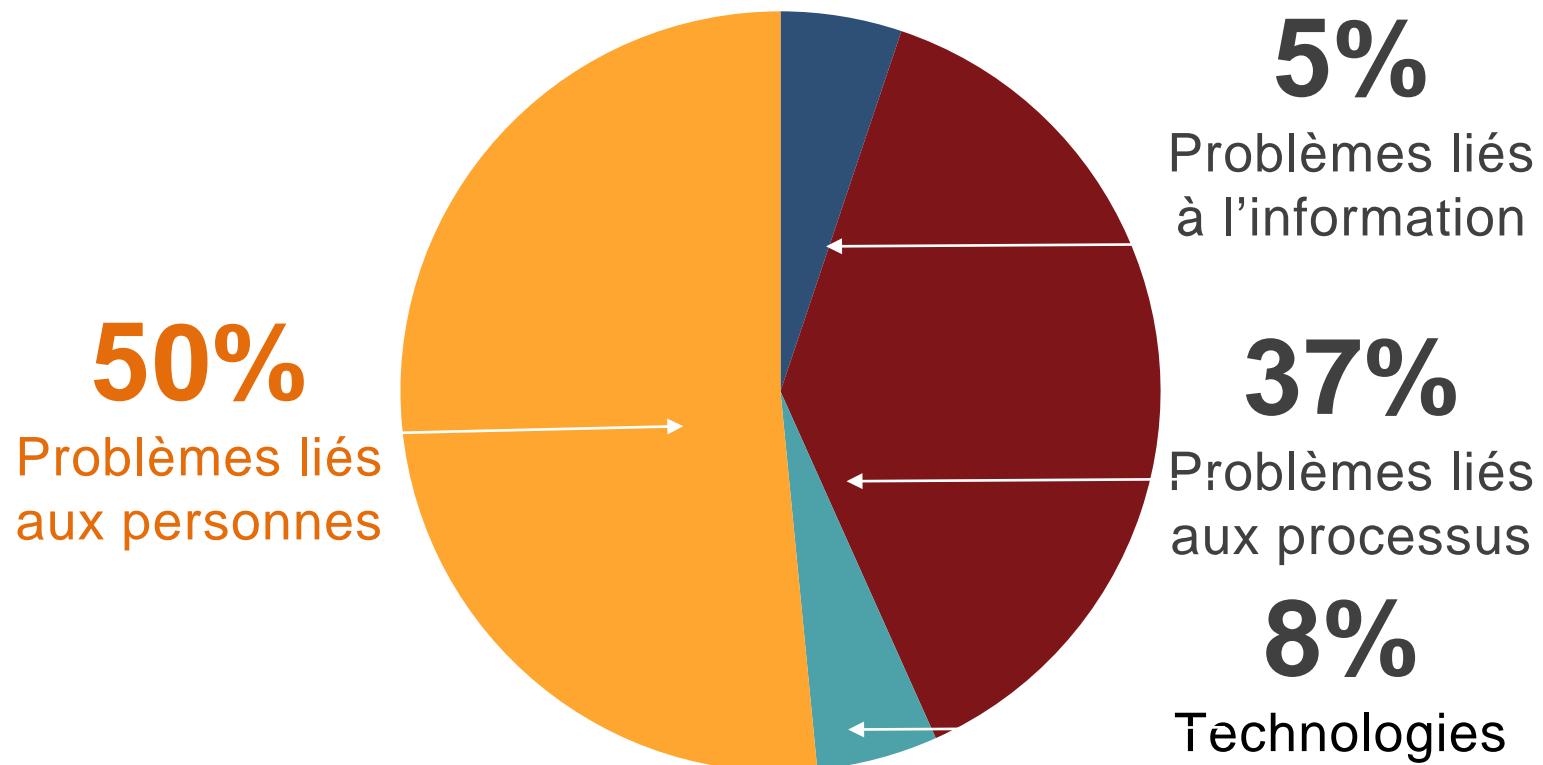
## Défis et risques

Relever ces défis nécessitera un changement organisationnel.



- Manque d'engagement ou de clarté
- Transformer une culture « eux » et « nous »
- Mélanger des équipes géographiquement dispersées, qui se connaissent mal et pouvant inclure des fournisseurs
- Manque d'éducation, de formation et de compétences
- Processus de gestion des services immatures
- Technologies inadéquates
- Mauvaise communication

## Votre plus grand défi pour l'expansion de DevOps ?



# EXERCICE

## Votre expérience DevOps (Plan d'action personnel)

## Module 8 : Quiz (1)

1. Qui fournit des « lofts » en tant qu'expérience immersive ?
  - a) Amazon
  - b) IBM
  - c) Google
  - d) Puppet
  
2. Qu'est-ce qu'un « Ignite » dans le contexte des DevOps Days?
  - a) Un discours inaugural
  - b) Une présentation de 30 minutes
  - c) Une session de 5 minutes sur un sujet spécifique
  - d) Une session en petit groupe

## Module 8 : Quiz (2)

3. Pourquoi Disney continue-t-il à faire plus de choses ? Parce qu'ils sont :
  - a) Impatients
  - b) Compétitifs
  - c) Curieux
  - d) Courageux
4. Quelle est la structure idéale pour une équipe DevOps ?
  - a) Matricielle
  - b) Axée sur le marché
  - c) Il n'y en a pas
  - d) Fonctionnelle transversale

## Module 8 : Quiz (3)

5. Selon Gartner, quel est le plus grand défi pour l'expansion de DevOps dans les organisations ?
  - a) Les personnes
  - b) Les informations
  - c) Les processus
  - d) Les technologies

## Module 8 : Réponses au quiz

- 1 Qui fournit des « lofts » en tant qu'expérience immersive ?  
a) Amazon  
b) IBM  
c) Google  
d) Puppet
- 2 Qu'est-ce qu'un « Ignite » dans le contexte des DevOps Days?  
a) Un discours inaugural  
b) Une présentation de 30 minutes  
**c) Une session de 5 minutes sur un sujet spécifique**  
d) Une session en petit groupe
- 3 Pourquoi Disney continue à faire plus de choses ?  
Parce qu'ils sont:  
a) Impatients  
b) Compétitifs  
**c) Curieux**  
d) Courageux
- 4 Quelle est la structure idéale pour une équipe DevOps ?  
a) Matricielle  
b) Orientée marché  
**c) Il n'y en a pas**  
d) Fonctionnelle transversage
- 5 Selon Gartner, quel est le plus grand défi pour l'expansion de DevOps dans les organisations ?  
a) Les personnes  
b) Information  
c) Processus  
d) Technologie

## Résumé

DevOps permet aux entreprises de fournir de meilleurs logiciels plus rapidement et de manière plus fiable en...

- Améliorant la communication, la collaboration et l'intégration des processus ainsi que des outils dans la chaîne de valeur informatique
- Automatisant le processus de livraison logiciel et des changements d'infrastructure
- Exploitant les pratiques agiles, Lean, ITSM et DevOps



"DevOps n'est pas seulement possible, il est nécessaire dans le nouveau monde d'entreprise."

Forrester Research

# DOI parcours d'apprentissage et communauté



GET DEVOPS FOUNDATION® CERTIFIED  
BECOME A DEVOPS INSTITUTE COMMUNITY MEMBER AND JOIN THE FASTEST GROWING GLOBAL COMMUNITY OF DEVOPS PROFESSIONALS  
ATTEND DEVOPS INSTITUTE SKILLUP® DAYS AND INDUSTRY EVENTS/CONFERENCES



**Why DevOps Institute Certifications Matter?**  
DevOps Institute offers eight certifications based on role-based competencies that are highly desirable when it comes to hiring and retaining talent in today's most competitive organizations.

## Why Get Certified?

- Gain knowledge and understanding of DevOps principles and practices
- Develop your skill set within a key area of DevOps
- Learn how to apply key DevOps principles
- Prove your subject matter expertise
- Enhance your professional credibility
- Stand out from other job applicants

## Certifications

DevOps Foundation®  
SRE Foundation (SREF)™  
Continuous Testing Foundation (CTF)™  
DevOps Leader (DOL)™  
Continuous Delivery Ecosystem Foundation (CDFE)™  
DevSecOps Foundation (DSOF)™  
Certified Agile Service Manager (CASM)®  
Certified Agile Process Owner (CAPO)®



DevOps Institute advances the human elements of DevOps equipping them with Skills, Knowledge, Ideas and Learning (the SKIL Framework).

[DevOpsInstitute.com](http://DevOpsInstitute.com)

DevOps Institute © Tous droits réservés

# DOI parcours d'apprentissage et communauté



## Join The DevOps Member Association

Advancing the Human Elements of DevOps

