

# LOG680

## Introduction à l'approche DevOps

### Second Way: Les principes de Rétroaction

DevOps Handbook  
Part I, Chap 3



Francis Bordeleau, 2021

# Three Ways of DevOps

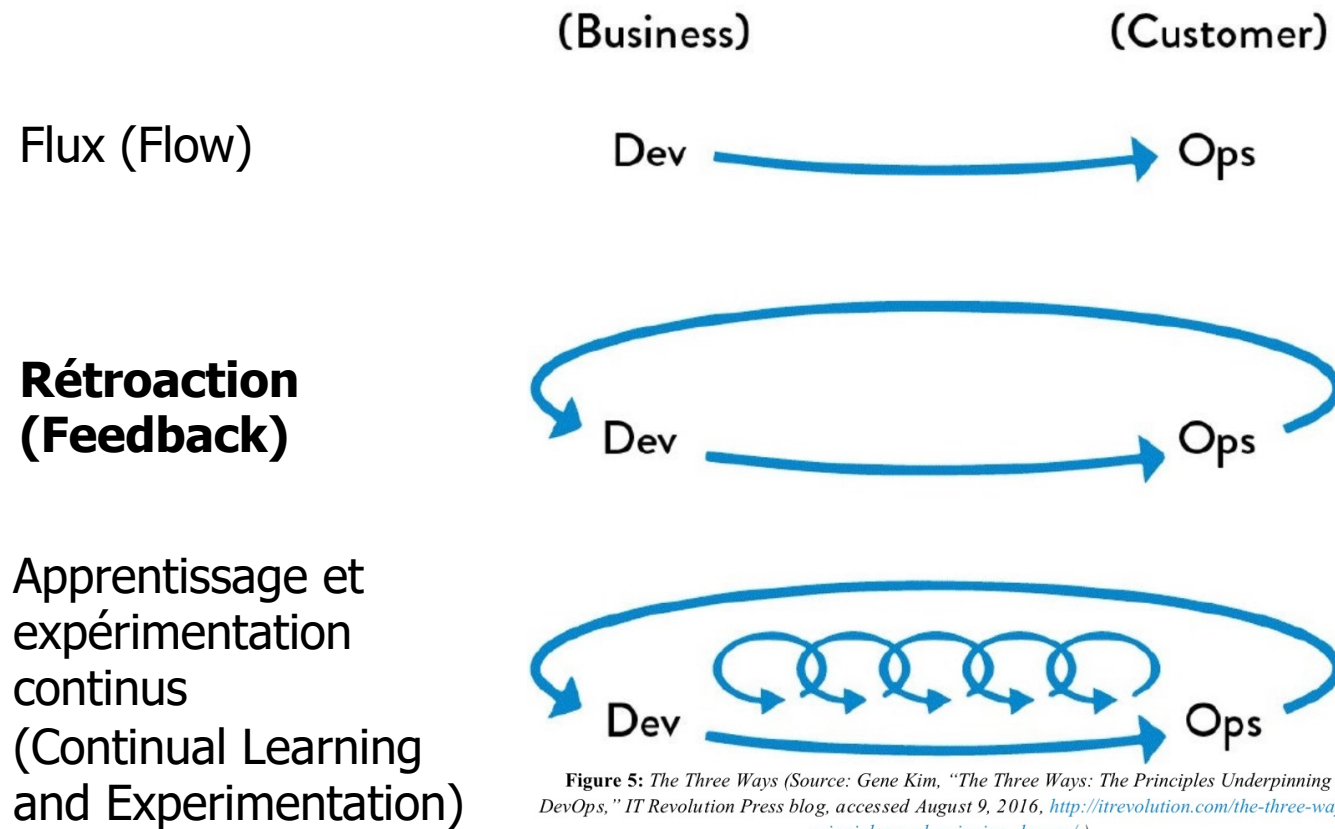


Figure 5: The Three Ways (Source: Gene Kim, "The Three Ways: The Principles Underpinning DevOps," IT Revolution Press blog, accessed August 9, 2016, <http://itrevolution.com/the-three-ways-principles-underpinning-devops/>.)

# Objectifs d'apprentissage

- Expliquer en quoi consiste la deuxième voie (Second Way)
- Expliquer ce qu'est le Toyota Andon?
- Expliquer ce qui caractérise les systèmes complexes
- Décrire les quatre conditions qui permettent de rendre plus sûr le travail dans des systèmes complexes?
- Donner des exemples de contrôles qualité inefficaces

# Sujets

- Introduction
- Systèmes complexes
- Voir les problèmes lorsqu'ils se produisent
- Investiguer et résoudre les problèmes
- Qualité à la source
- Optimiser pour les centres de travail en aval
- Conclusion

- **Introduction**
- Systèmes complexes
- Voir les problèmes lorsqu'ils se produisent
- Investiguer et résoudre les problèmes
- Qualité à la source
- Optimiser pour les centres de travail en aval
- Conclusion

## Second Way: Rétroaction



# Introduction

- Tandis que la "First Way" décrit les principes permettant la circulation rapide du travail de gauche à droite, la "**Second Way**" décrit les **principes permettant la rétroaction réciproque rapide et constante de droite à gauche à toutes les étapes de la chaîne de valeur.**
- **Objectif: créer un système de travail toujours plus sûr et plus résilient.**
- Particulièrement important lorsque vous travaillez dans des **systèmes complexes**
  - Notre travail se déroule presque entièrement dans des systèmes complexes avec un risque élevé de conséquences catastrophiques.
  - Nous ne découvrons souvent les problèmes que lorsque des défaillances importantes surviennent, e.g. panne de production massive ou une atteinte à la sécurité entraînant le vol de données clients.
- Objectif: Rendre notre système de travail plus sûr en **créant des flux d'informations rapides, fréquents et de haute qualité dans l'ensemble de notre chaîne de valeur et de notre organisation.**

- Introduction
- **Systemes complexes**
- Voir les problèmes lorsqu'ils se produisent
- Investiguer et résoudre les problèmes
- Qualité à la source
- Optimiser pour les centres de travail en aval
- Conclusion



# Questions

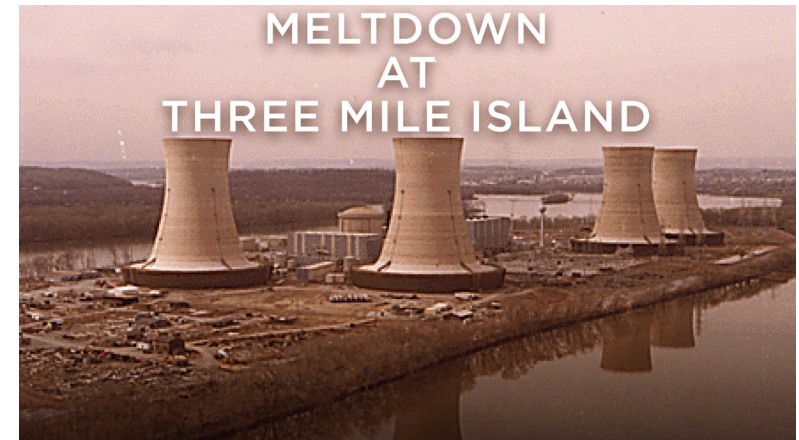
- Qu'est-ce qu'un système complexe?
- Quelles sont les caractéristiques d'un « système complexe »?
- Exemples de systèmes complexes

# Systemes complexes

- Caracteristiques d'un **systeme complexe** :
  - Essentiellement impossible de voir le systeme dans son ensemble et de comprendre comment toutes les pieces s'emboitent et interagissent
  - Niveau eleve d'interconnexion de composants a couplage etroit
  - Niveau eleve de parallelisme et de simultaneite (parallelism and concurrency)
  - Le comportement du systeme est non-deterministe, i.e. faire deux fois la meme chose ne mene pas necessairement au meme resultat.
  - Le comportement du systeme est emergent (emerging behavior) et ne peut pas etre compris simplement en analysant/observant le comportement individuel des composants du systeme.
  - **Pratiquement impossible de comprendre tous les comportements possibles**
- Par consequent, **la defaillance etant inherente et inevitable**
  - Nous **devons concevoir un systeme de travail sur**, ou nous **pouvons effectuer un travail sans crainte, confiants que toutes les erreurs seront detectees rapidement, bien avant qu'elles ne provoquent des consequences catastrophiques.**

# Accident de Three Mile Island

- [Wikipedia](#)
  - 28 mars 1979 dans la [centrale nucléaire](#) de [Three Mile Island](#) (3,3 km<sup>2</sup>)
- Dr Charles Perrow a étudié la crise de Three Mile Island et a constaté que
  - Impossible à quiconque de comprendre comment le réacteur se comporterait dans toutes les circonstances et comment il pourrait échouer.
  - Lorsqu'un problème était en cours dans un composant, il était difficile de l'isoler des autres composants, passant rapidement par les chemins de la moindre résistance de manière imprévisible.



<https://www.pbs.org/wgbh/americanexperience/films/three/>

# Systemes complexes – Dr Steven Spear

- La conception de systemes parfaitement sûrs est probablement au-delà de nos capacités
- Cependant, nous pouvons **rendre plus sûr le travail dans des systemes complexes** lorsque les **quatre conditions** suivantes sont remplies:
  - Le travail complexe est **géré de manière à révéler les problèmes de conception et de fonctionnement**
  - **Les problèmes sont essayés et résolus**, entraînant la construction rapide de nouvelles connaissances
  - Les **nouvelles connaissances locales sont exploitées globalement** dans l'ensemble de l'organisation.
  - Les **leaders créent d'autres leaders** qui développent continuellement ces types de capacités
- Chacune de ces capacités est nécessaire pour travailler en toute sécurité dans un système complexe.

- Introduction
- Systèmes complexes
- **Voir les problèmes lorsqu'ils se produisent**
- Investiguer et résoudre les problèmes
- Qualité à la source
- Optimiser pour les centres de travail en aval
- Conclusion

# Voir les problèmes lorsqu'ils se produisent

- Dans un système de travail sécurisé, nous devons constamment tester nos hypothèses de conception et de fonctionnement.
- Objectif :
  - **Accroître le flux d'information dans notre système à partir du plus grand nombre de contextes possibles, plus tôt, plus rapidement, à moindre coût et avec autant de clarté que possible entre cause et effet.**
    - Plus vite nous pouvons invalider les hypothèses, plus vite nous pouvons trouver et résoudre les problèmes
    - Augmente notre résilience, notre agilité et notre capacité à apprendre et à innover.
- Moyen :
  - Créer des retours d'information (feedback) et des boucles d'information (feedforward) dans notre système de travail.

# Secteur manufacturier

- Dans le secteur manufacturier, l'absence de rétroaction efficace contribue souvent à des problèmes majeurs de qualité et de sécurité.
  - Dans un cas bien documenté de l'usine de fabrication General Motors Fremont, il n'existait aucune procédure efficace permettant de détecter les problèmes lors du processus d'assemblage, ni de procédure explicite indiquant la marche à suivre lorsque des problèmes étaient détectés.
  - En conséquence, il est arrivé que des moteurs soient montés à l'arrière, que des voitures manquent de volants ou de pneus, et que des voitures doivent même être remorquées hors de la chaîne de montage car elles ne démarraient pas.
- En revanche, dans les opérations de fabrication à hautes performances, il existe un flux d'informations rapide, fréquent et de haute qualité sur l'ensemble du flux de valeur: chaque opération de travail est mesurée et surveillée, et tous les défauts ou écarts importants sont rapidement détectés et traités.
  - Ce sont les fondements de ce qui permet la qualité, la sécurité, l'apprentissage et l'amélioration continue.

# Secteur technologique

- Dans le flux de valeur technologique, nous obtenons souvent des résultats médiocres en raison de l'absence de retour rapide.
  - Exemple: projet de logiciel de cascade
    - Nous pouvons développer du code pour une année entière et ne recevoir aucun retour sur la qualité avant le début de la phase de test - ou pire, lorsque nous livrons notre logiciel aux clients.
  - Lorsque les retours ("feedback ») sont aussi tardifs et peu fréquents, ils sont trop lents pour nous permettre d'éviter des résultats indésirables.



# Secteur technologique

- **Objectif: créer des retours d'information (feedback) et des boucles d'information (feedforward) rapides**, quel que soit le travail effectué, à **toutes les étapes** de la chaîne de valeur technologique
  - Incluant gestion des produits, développement, assurance qualité, Infosec et Ops.
  - Cela inclut la création de processus de génération, d'intégration et de test automatisés, afin que nous puissions détecter immédiatement lorsqu'une modification nous amène dans un état problématique (e.g. le système ne fonctionne pas correctement ou ne peut être déployé).
- Créer également une **télémetrie omniprésente** afin de voir comment tous les composants de notre système fonctionnent dans l'environnement de production, afin de pouvoir détecter rapidement quand ils ne fonctionnent pas comme prévu.
  - La télémetrie **permet aussi de mesurer si nous atteignons les objectifs visés** et, idéalement, se répercute sur l'ensemble de la chaîne de valeur afin que nous puissions voir comment nos actions affectent le système dans son ensemble.

# Boucle de rétroaction

- Les boucles de rétroaction permettent non seulement une détection et une récupération rapides des problèmes, mais elles nous **indiquent également comment éviter que ces problèmes ne se reproduisent à l'avenir.**
  - Augmente la qualité et la sécurité de notre système de travail et crée un apprentissage organisationnel.
- Elisabeth Hendrickson (VP Ingénierie chez Pivotal Software, Inc. et auteure de Explore It!: Reduce Risk and Increase Confidence with Exploratory Testing)

"Lorsque je dirigeais l'ingénierie de la qualité, je décrivais mon travail comme étant la "création de cycles de rétroaction".

La rétroaction est essentielle car c'est ce qui nous permet de diriger.

Nous devons constamment valider les besoins des clients, nos intentions et nos implémentations.

Les tests ne sont qu'un type de retour d'information."

- Introduction
- Systèmes complexes
- Voir les problèmes lorsqu'ils se produisent
- **Investiguer et résoudre les problèmes**
- Qualité à la source
- Optimiser pour les centres de travail en aval
- Conclusion

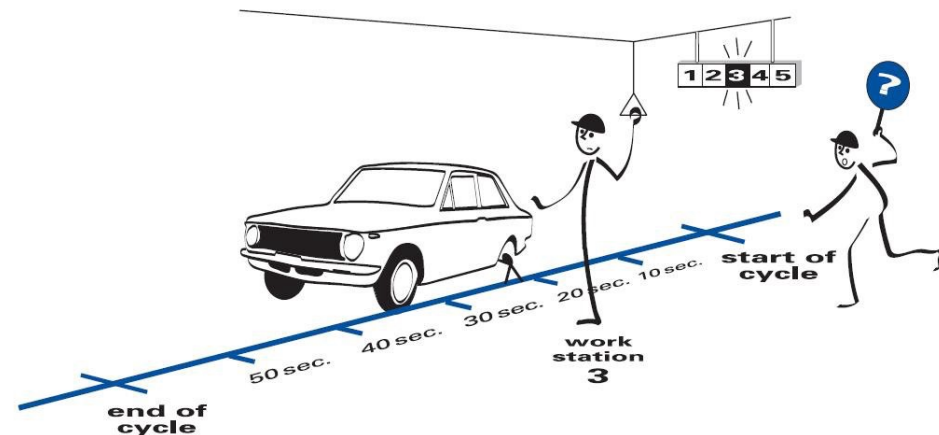
# Investiguer et résoudre les problèmes

- Lorsque des problèmes surviennent, nous devons les investiguer/essaimer ("swarm"), en mobilisant ceux qui en ont besoin, pour les résoudre.
- Selon le Dr Spear, l'essaimage a pour objectif de contenir les problèmes avant qu'ils ne se propagent, ainsi que de diagnostiquer et de traiter le problème afin qu'il ne puisse plus se reproduire.
- Ce faisant, **les équipes développent des connaissances de plus en plus approfondies sur comment gérer les systèmes pour faire le travail, transformant ainsi l'ignorance en connaissance.**

# Toyota Andon



<https://www.leansixsigmadefinition.com/glossary/andon/>



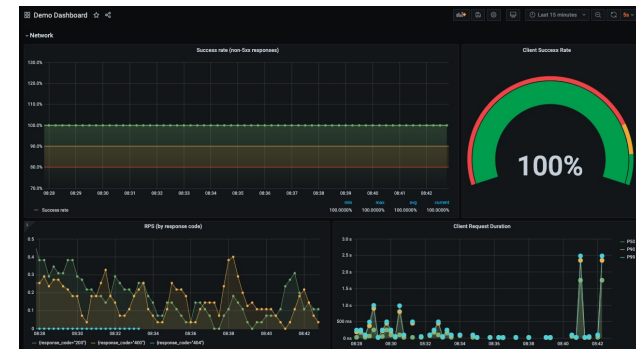
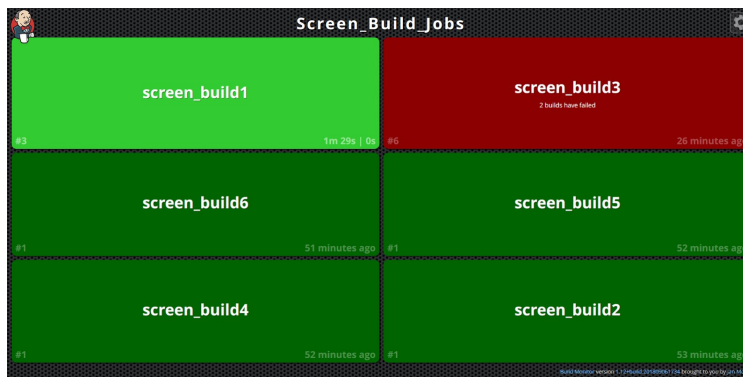
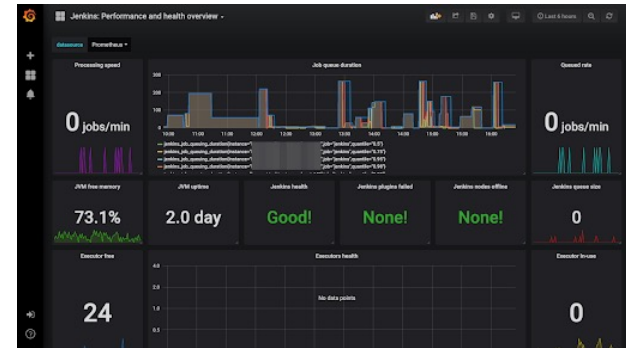
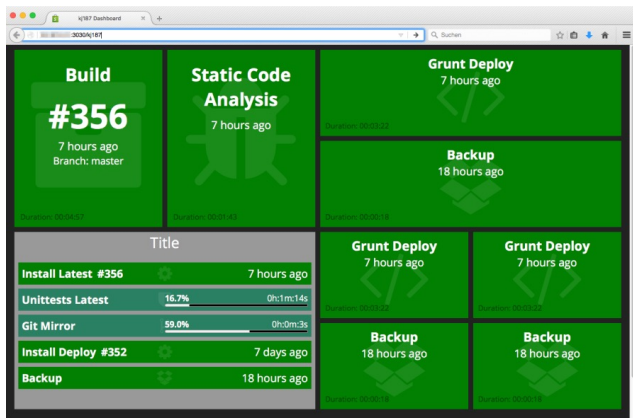
Fixed-position stop system.

<https://medium.com/serious-scrum/andon-processes-can-help-your-team-value-fire-prevention-over-fire-fighting-23fdeee504f5>

# Toyota Andon

- Dans une usine Toyota, au-dessus de chaque centre de travail se trouve un cordon que chaque travailleur et chaque responsable est formé à tirer en cas de problème
  - par exemple, lorsqu'une pièce est défectueuse, quand une pièce requise n'est pas disponible ou même lorsque le travail prend plus de temps que documenté.
- Lorsque le cordon Andon est tiré, le chef d'équipe est alerté et s'efforce immédiatement de résoudre le problème.
- Si le problème ne peut pas être résolu dans un délai spécifié (par exemple 55 sec), la chaîne de production est arrêtée afin que toute l'organisation puisse être mobilisée pour aider à la résolution du problème jusqu'à la mise au point d'une contre-mesure efficace.
- Au lieu de contourner le problème ou de programmer un correctif «quand nous aurons plus de temps», nous essaierons de le régler immédiatement - c'est presque le contraire du comportement de l'usine GM Fremont décrit précédemment.

# Cordon Andon en DevOps



# Essaimage (“Swarming”)

- L'essaimage est nécessaire pour les raisons suivantes:
  - Il empêche le problème de progresser en aval, où les coûts et les efforts pour le résoudre augmentent de façon exponentielle et que la dette technique peut s'accumuler.
  - Cela empêche le poste de travail de commencer un nouveau travail, ce qui introduira probablement de nouvelles erreurs dans le système.
    - Si le problème n'est pas résolu, le poste de travail pourrait potentiellement rencontrer le même problème lors de la prochaine opération, nécessitant davantage de correctifs et de travail. (Voir annexe 6).



# Essaimage (“Swarming”)

- Cette pratique d'essaimage semble contraire aux pratiques de gestion habituelles
  - Nous laissons délibérément un problème local perturber les opérations à l'échelle globale.
- L'essaimage permet l'apprentissage.
- Il évite la perte d'informations critiques en raison de la perte de mémoire ou de l'évolution des circonstances.
  - Particulièrement critique dans les systèmes complexes, où de nombreux problèmes surviennent à cause d'interactions inattendues et idiosyncratiques de personnes, processus, produits, lieux et circonstances.

# Essaimage (“Swarming”)

- Ce n’est que par l’essaimage de problèmes de plus en plus mineurs découverts de plus en plus tôt dans le cycle de vie que nous pourrions les résoudre avant qu’une catastrophe se produise.
  - En d’autres termes, lorsque le réacteur nucléaire fondra, il est déjà trop tard pour éviter les pires conséquences.
- **Pour permettre un retour rapide dans le flux de valeur technologique, nous devons créer l’équivalent d’un cordon Andon et de la réponse d’essaimage associée.**
- Nécessite de créer une culture qui encourage à tirer le cordon d’Andon en cas de problème
  - Que ce soit lorsqu’un incident de production se produit ou lorsque des erreurs se produisent plus tôt dans la chaîne de valeur
  - Exemple: quelqu’un introduit un changement qui rompt nos processus de construction ou de test continus.

# Essaimage (“Swarming”)

- Lorsque le cordon Andon est tiré, nous essaierons de résoudre le problème et d’empêcher l’introduction de nouveaux travaux tant que le problème ne sera pas résolu.
  - Fournit un retour rapide à tous les utilisateurs de la chaîne de valeur (en particulier la personne à l’origine de la défaillance du système)
  - Permet d’isoler et de diagnostiquer rapidement le problème et d’éviter des facteurs de complication pouvant masquer les causes et les effets.
  - Empêche l’introduction de nouveaux travaux permet une intégration et un déploiement continus, qui constituent un flux monobloc dans le flux de valeur technologique.
- Toutes les modifications qui passent nos tests de construction et d’intégration continus sont déployées en production et toutes les modifications qui entraînent l’échec de ces tests déclenchent notre cordon Andon et sont essaimées jusqu’à ce qu’elles soient résolues.

- Introduction
- Systèmes complexes
- Voir les problèmes lorsqu'ils se produisent
- Investiguer et résoudre les problèmes
- **Qualité à la source**
- Optimiser pour les centres de travail en aval
- Conclusion

# Qualité à la source

- Nous pouvons, par inadvertance, perpétuer des systèmes de travail dangereux en raison de notre façon de réagir aux accidents et aux incidents.
- Dans les systèmes complexes, **l'ajout d'étapes d'inspection et de processus d'approbation augmente en fait la probabilité d'échec futur.**
- **L'efficacité des processus d'approbation diminue à mesure que nous éloignons la prise de décision du lieu où le travail est effectué.**
  - Réduit la qualité des décisions
  - Augmente le temps de cycle
  - Diminue la force de la rétroaction entre cause et effet
  - Réduit notre capacité à tirer des leçons des succès et des échecs
- Lorsque les systèmes hiérarchiques de commandement et de contrôle bureaucratiques deviennent inefficaces, c'est généralement parce que l'écart entre «qui doit faire quelque chose» et «qui fait réellement quelque chose» est trop important.

# Exemples de contrôles qualité inefficaces

- Demander à une autre équipe **d'effectuer des tâches manuelles fastidieuses**, sujettes aux erreurs et pouvant être facilement automatisées et exécutées au besoin par l'équipe ayant besoin du travail exécuté
- **Exiger des approbations de la part de personnes occupées qui sont éloignées du travail**, les obligeant à prendre des décisions sans une connaissance suffisante du travail ou de ses implications potentielles, ou simplement à donner leur aval à leurs approbations.
- **Création de gros volumes de documentation** de détails douteux qui deviennent obsolètes peu de temps après leur rédaction.
- **Transférer de grandes quantités de travail aux équipes et aux comités spéciaux** pour approbation et traitement, puis attendre les réponses.

# Qualité à la source: solution

- **Tous les membres de notre chaîne de valeur trouvent et résolvent les problèmes de leur zone de contrôle dans le cadre de leur travail quotidien.**
  - Ce faisant, nous poussons les responsabilités en matière de qualité et de sécurité et la prise de décision au lieu où le travail est effectué, au lieu de nous fier aux approbations de cadres distants.
- **Utiliser des examens par les pairs** de nos modifications proposées pour obtenir l'assurance nécessaire que nos modifications fonctionneront comme prévu.
- **Automatiser autant que possible le contrôle de la qualité**
  - Généralement effectué par un service d'assurance de la qualité ou de sécurité de l'information.
  - Au lieu que les développeurs aient besoin de demander ou de planifier l'exécution d'un test, ces tests peuvent être effectués à la demande, ce qui leur permet de tester rapidement leur propre code et même de déployer eux-mêmes ces modifications en production.

# Qualité à la source: résultat

- **Faire de la qualité la responsabilité de tous**
  - Par opposition à une responsabilité qui incombe à un seul département.
  - La sécurité de l'information n'est pas simplement une tâche de la sécurité de l'information, de même que la disponibilité n'est pas simplement la tâche d'Operations.
- **Faire en sorte que les développeurs partagent la responsabilité de la qualité des systèmes qu'ils construisent améliore non seulement les résultats, mais accélère également l'apprentissage.**
  - Ceci est particulièrement important pour les développeurs car ils constituent généralement l'équipe la plus éloignée du client.
- Comme le remarque Gary Gruver,  
"Il est impossible pour un développeur d'apprendre quoi que ce soit lorsque quelqu'un crie après quelque chose qu'il a cassé il y a six mois. C'est pourquoi nous devons informer tout le monde le plus rapidement possible, en quelques minutes et non en mois."



- Introduction
- Systèmes complexes
- Voir les problèmes lorsqu'ils se produisent
- Investiguer et résoudre les problèmes
- Qualité à la source
- **Optimiser pour les centres de travail en aval**
- Conclusion

# Optimiser pour les centres de travail en aval

- L'industrie traditionnelle se concentre sur le client externe.
- **Lean définit deux types de clients** que nous devons concevoir
  - **Client externe** – qui paie le plus probablement pour le service que nous fournissons
  - **Client interne** – qui reçoit et traite le travail immédiatement après nous
- Selon Lean, **notre client le plus important est notre prochaine étape en aval.**
  - Optimiser notre travail pour eux nécessite que nous ayons une empathie vis-à-vis de leurs problèmes afin de mieux identifier les problèmes de conception empêchant un écoulement rapide et sans heurts.

# Optimiser pour les centres de travail en aval

- Dans le flux de valeur technologique, nous **optimisons les postes de travail en aval en concevant pour les opérations de manière à ce que les exigences opérationnelles non fonctionnelles** (telles que l'architecture, les performances, la stabilité, la testabilité, la configurabilité et la sécurité) **sont priorisées autant que les fonctionnalités utilisateur.**
- **Ce faisant, nous créons de la qualité à la source**, ce qui aboutit probablement à un ensemble d'exigences non fonctionnelles codifiées que nous pouvons intégrer de manière proactive à chaque service que nous construisons.

- Introduction
- Systèmes complexes
- Voir les problèmes lorsqu'ils se produisent
- Investiguer et résoudre les problèmes
- Qualité à la source
- Optimiser pour les centres de travail en aval
- **Conclusion**

# Conclusion

- La création de rétroactions (feedback) rapides est essentielle pour obtenir qualité, fiabilité et sécurité dans le flux de valeur technologique.
- Nous y parvenons en
  - **déTECTANT les problèmes** au fur et à mesure qu'ils se présentent
  - **essaimant et en résolvant des problèmes** pour acquérir de nouvelles connaissances
  - **rapprochant la qualité de la source**
  - **optimisant en permanence les postes de travail en aval**