

A thick red arrow pointing to the right, positioned to the left of the word "Synthèse".

# Synthèse

# Les 7 principes de DevOps

- 1 - versioning de tous les artefacts

Dev et Ops doivent versionner leur code / IaC, et les produits du flux CI /CD qu'il partagent. Ils doivent utiliser la même techno

- 2 - Intégration et Déploiement Continus

Exécuter les flux CI /CD le plus régulièrement possible sur le trunk du SCM – éviter de laisser du code abandonné dans des branches

- 3 - Automatiser les test d'acceptance

Stopper l'exécution à la moindre erreur, dès les tests unitaires, tout le long de la phase de test

- 4 - Pratiquer la revue des travaux par les pairs

Sur le dev, permet d'accélérer la correction de bug, sur l'ops permet d'aligner les objectifs, améliore la communication

# Les 7 principes de DevOps

- 5 - Culture de la confiance

Encouragée par le principe précédent. Responsabilisation des acteurs et facteur d'innovation

- 6 - Supervision permanente de l'environnement de production

« Shift Right », source privilégiée d'information pour comprendre et améliorer l'expérience utilisateur

- 7 - Relation « Gagnant - Gagnant » entre Dev et Ops

Validation des livrables et automatisation du déploiement par les Dev, intégrité serveur et remontées d'informations par les Ops => profitabilité partagée, confiance rétablie

# Les 7 piliers de DevOps

- 1 - La Culture de la Collaboration: Alignement des objectifs, coopérations des Devs, Ops, Qa... Sur le flux de production
- 2 - Le Design pour DevOps: architecture du produit préférentiellement orientée microservices, IaC déclarative et immutable, Cloud Computing.
- 3 - Intégration continue (CI): minimiser les régressions, les rollbacks, les délais de livraisons.
- 4 - Tests en Continu (CT): Grande variété des sujets (units, e2e, performances, sécurité, qualité... Tirer parti au mieux de la parallélisation et rechercher la couverture maximale, dans le temps le plus raisonnable (risque de goulot d'étranglement).

# Les 7 piliers de DevOps

- 5 - Monitoring en Continu (CM): Remontée des informations à chaque étape du pipeline pour aider à la décision et éviter les goulots d'étranglement.
- 6 - Une Infrastructure flexible: redéfinissable facilement par l'IaC, scalable et contrôlable par orchestration
- 7 - Livraison Continue (CD): automatiser la préparation et le test d'un code à livrer pour augmenter la fréquence de livraison



# Lexique ITIL

- Les principales publication d'ITIL v3(phases du cycle de vie des services)
  - **Service Strategy** : planification, financement, priorisation des services
  - **Service Design** : Production du service. Facilitation de sa livraison
  - **Service Transition** : gestion des builds & déploiement, gestion du changement
  - **Service Operation** : SLA, gestion des évènements, problèmes, incidents...
  - **Continual Service Improvement** (CSI) : Amélioration des processus ITSM
- **Service Provider** : structure qui fournit des services à des clients internes ou externes.
- **Service Request** : demande utilisateur d'accès à un service standard de l'IT
- **Service Desk** : Guichet unique entre le service IT et les utilisateurs. Gestion du cycle de vie des services.
- « **Service Level Agreement** », SLA : Accord écrit entre le service It et ses clients qui définit les objectifs clés d'un service et les responsabilités des deux parties.

# Lexique ITIL

- **Known Error / Known Error Database** : Erreur dont on connaît la cause première des solutions de contournement
- **« Remediation Plan »** : Plan de remédiation ou relance d'un projet adopté dans la foulée de la mise en œuvre de changements non concluants dans l'organisation, ou après une livraison ratée
- Indicateurs ITIL :
  - **Mean Time Between Deploys** : Durée moyenne entre 2 déploiement.
  - **Mean Time Between Failures (MTBF)** : Temps moyen pendant le quel un CI peut fonctionner sans interruption => mesure de la fiabilité
  - **Mean Time to Detect Defects (MTTD)** Temps moyen entre de remontée des bugs.
  - **Mean Time to Repair (MTTR)** : temps moyen requis pour la correction d'un bug (hors reprise de service).
  - **Mean Time to Resolution (MTTRe)** : Temps moyen de résolution d'un problème bloquant
  - **Mean Time to Restore Service (MTRS)** : Temps moyen entre une interruption de service et la reprise de service => mesure de la maintenabilité

# Lexique Scrum

- « **Scrum** » : un cadre méthodologique simple pour assurer une collaboration efficace sur des projets complexes. Scrum fournit un minimum de règles pour générer une structure d'équipe légère centrée sur la production de valeur.
- « **Scrum Team** » : équipe multidisciplinaire auto-organisée utilisant Scrum pour fournir des produit de manière itérative et incrémentale. Comprend le « **Product Owner** », L'équipe de développement, et le « **Scrum Master** ».
- « **Sprint** » : une itération planifiée du projet pendant laquelle une incrémentation du produit est effectuée
- « **Impediment** » : Obstacle. Tout ce qui peut empêcher un membre de l'équipe d'avancer comme prévu.
- « **Product Backlog** » : liste priorisée de besoins fonctionnels et non fonctionnels exprimés sous la forme de User Stories.
- « **Sprint Backlog** » : Part des Backlogs concernant le Sprint courant.



# Glossaire DevOps

- **A3** : Une méthode de résolution de problème basé sur un rapport rédigé sur un document au format A3, qui impose la concision
- **Agile service management** : introduction des principes agiles dans ITIL v4
- **Anti-pattern** : Un anti modèle de résolution de problème.
- **Apprentissage immersif** : expérience en réalité virtuelle ou augmenté dans le but de mémoriser et maîtriser un geste ou une pratique
- **Collective Body of Knowledge(CBOK)** : Etude internationale réalisée par les cabinets d'audit. "Bible" de la gestion de processus pour les responsables qualité, et des managers
- **La gestion des conflits** : c'est le processus consistant à limiter les aspects négatifs d'un conflit tout en augmentant les aspects positifs du conflit. Son but est d'améliorer l'apprentissage et les résultats collectifs, incluant l'efficacité. Les conflits bien gérés peuvent être sources de bénéfices.
- **« Thomas Kilmann Inventory »**, TKI : Instrument Thomas Kilmann, mesure des choix comportementaux des sujets soumis à différentes situations de négociations : collaboration, compétition, compromis, Evitement, capitulation.

# Glossaire DevOps

- **Dette culturelle** : Décision du service IT contraire à la culture de l'entreprise. De telles décisions peuvent introduire des divisions dans les équipes, détériorer la communication et même dégrader l'efficacité du leadership.
- **« Done »** (au sens DevOps) : état final d'une tâche qui assure la valeur ajoutée au client => déployé en production
- **« Done »** (au sens Scrum) : état final d'une tâche dans un sprint qui assure l'acceptation par le Product Owner
- **eNPS** « Employee Net Promoter Score » : indicateur de mesure de l'engagement des employés envers l'entreprise.
- **epics** : Tâches importantes qu'il est préférable de décomposer en différentes « stories »
- **« Golden Circle »** : méthode d'analyse de l'activité qui insiste sur la compréhension de ses causes : se demander "pourquoi faire cela ?" avant "que faire ?" et "comment faire ?"

# Glossaire DevOps

- **SO/IEC 20000** : Standard international pour la gestion de service IT, à des fins d'audit et de certification.
- **Le modèle de Kübler-Ross** : décrit une série de phases auxquelles les individus sont confrontés en cas de changement majeur dans leur vie en vue de son acceptation
- **Machine Learning** : Méthode d'analyse de données qui utilise des algorithmes d'auto apprentissage issu des données.
- **« Minimum Viable Product »** : Produit Minimum Viable, version minimale d'un produit qui peut être déployée et fournir de la valeur ajoutée aux utilisateurs.
- **« Potentially Shippable Product »** : Incrémentation du produit livrable susceptible d'être déployé

# Glossaire DevOps

- **Changement organisationnel** : Efforts entrepris d'adaptation des comportements professionnels à l'intérieur d'une entreprise pour s'acclimater aux nouvelles structures et exigences métier.
- **Performance organisationnelle** : résultat obtenu par rapport aux objectifs, à la stratégie de l'entreprise et/ou aux attentes des parties prenantes. Porte sur la qualité, la quantité, les coûts et le temps.
- **« Non-functional requirements »** : Besoin non fonctionnel, autrement dit Spécification Technique pure.
- **« Root Cause Analysis »**, RCA : analyse des causes racine, méthode d'amélioration continue itérative, qui consiste à remonter successivement aux causes fondamentales du problème, plutôt que de traiter des symptômes.
- **« Rugged Development »** : application du « shift left » à la sécurité, gestion des pratiques de sécurité dès la planification.



# Glossaire DevOps

- **Security as Code** : Automatisation des pratiques de sécurité à partir de configuration, au sein de l'intégration continue
- « **Scaled Agile Framework** », SAFE : cadre Méthodologique d'application des principes et pratiques agiles et lean à l'échelle d'une entreprise.
- **Simian Army/Chaos Monkey** : suite d'outils d'automatisation des tests de résiliences ; simulation de pannes machines
- « **SMART Goals** » : objectifs Spécifiques et Simples, Mesurables, Acceptable et Ambitieux, Réalistes, planifiés dans le Temps
- « **Variable Speed IT** » : Gestion de la coexistence dans le service IT de processus numérisés et non numérisés.
- **Vélocité** : Mesure du nombre moyen de tâches réalisées par unité de temps.
- « **Waterfall Project Management** » : cycle de développement en cascade, linéaire et séquentiel.