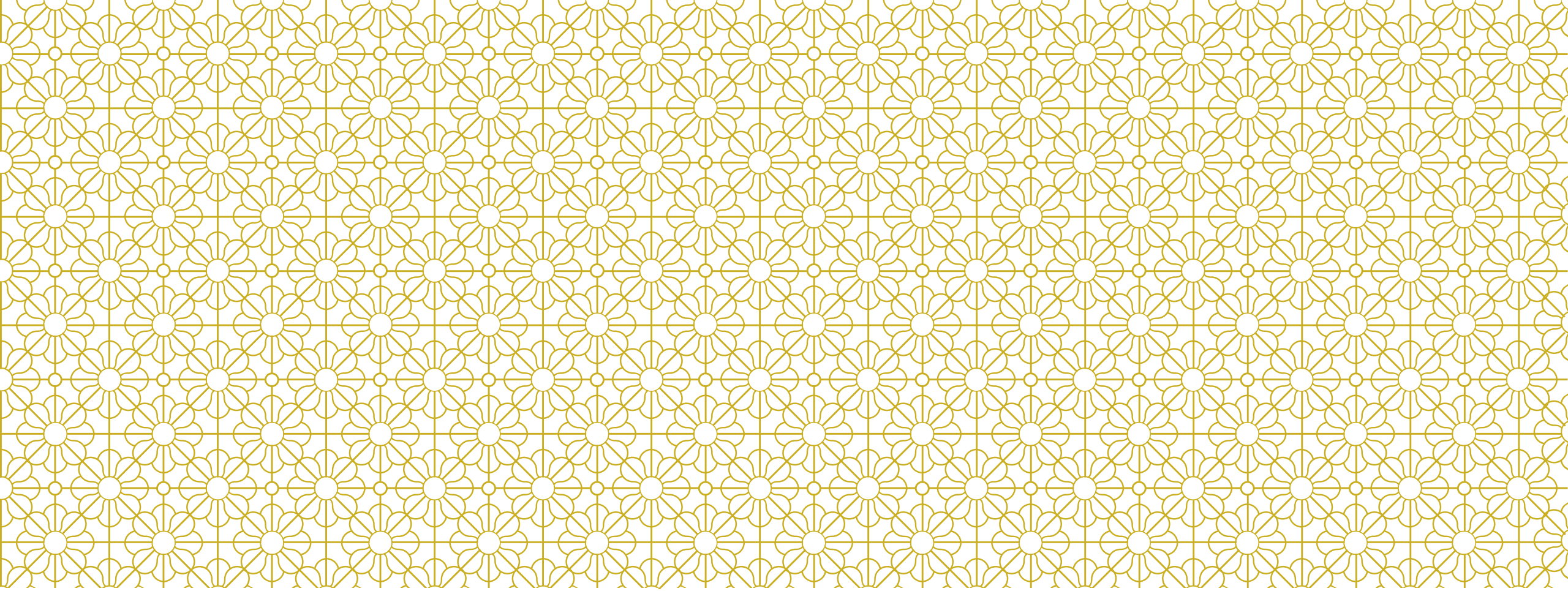


CI CD

Abderrahman CHEIKH AHMEDOU
SOFTWARE ARCHITECT AT GTI

CI CD

Accéler le rythme de déploiement des applications

Historiqument le rythme de mise en production est de 2 à 3 versions majeurs par an, chaque version apporte des nouvelles fonctionnalités

PARLER DES VERSIONNING

Version mineurs, majeurs …

ETAPES CI CD

Chaque étape est géré par une équipe et cela prend du temps

POUR ACCÉLÉRER LE PROCESSUS

À chaque étape il faut

Synchroniser les différents acteurs

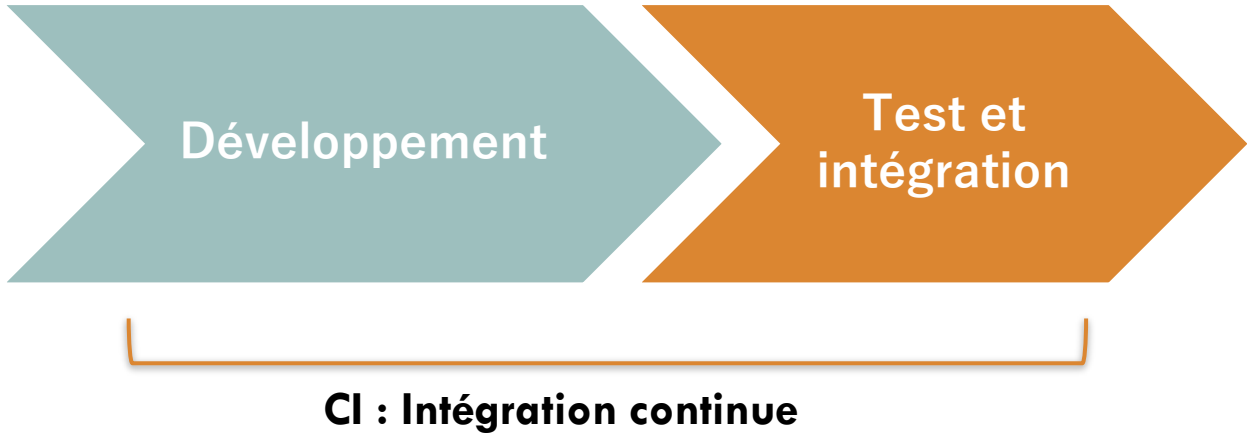
Coordonnées leurs planning

Planifier les actions à mettre en place


Développement
Test et intégration
Recette
Mise en PROD
CI : Intégration continue
CD : déploiement continue

CI

Automatiser les opérations autour du développement



CD

Automatiser les opérations autour du déploiement

PIPELINE CI/CD OU CHAINE CI/CD

Élimine l'intervention humaine et manuelle qui prend du temps et source d'erreur éventuellement

INTÉGRATION CONTINUE

Une équipe composé de plusieurs membre qui travaille sur des micro service et à un moment donné doivent fusionner leur travaille on doit utiliser un SCM : source code management

Exemple de SCM très populaire : **GIT**, GITLAB, GITHUB

Le code va être pusher au serveur d'intégration

SERVEUR D'INTÉGRATION

Composé de plusieurs services

Un service d'orchestration comme Ansible va déclencher une série d'action lorsqu'il détecte un changement dans le code source :

Compiler le code source (build) : un exécutable du code source

Puis lancer les tests pour s'assurer qu'il y aura pas de régression

TYPE DE TESTS

Tests unitaire : codé par le développeur, servent à valider le bon fonctionnement d'une fonction.

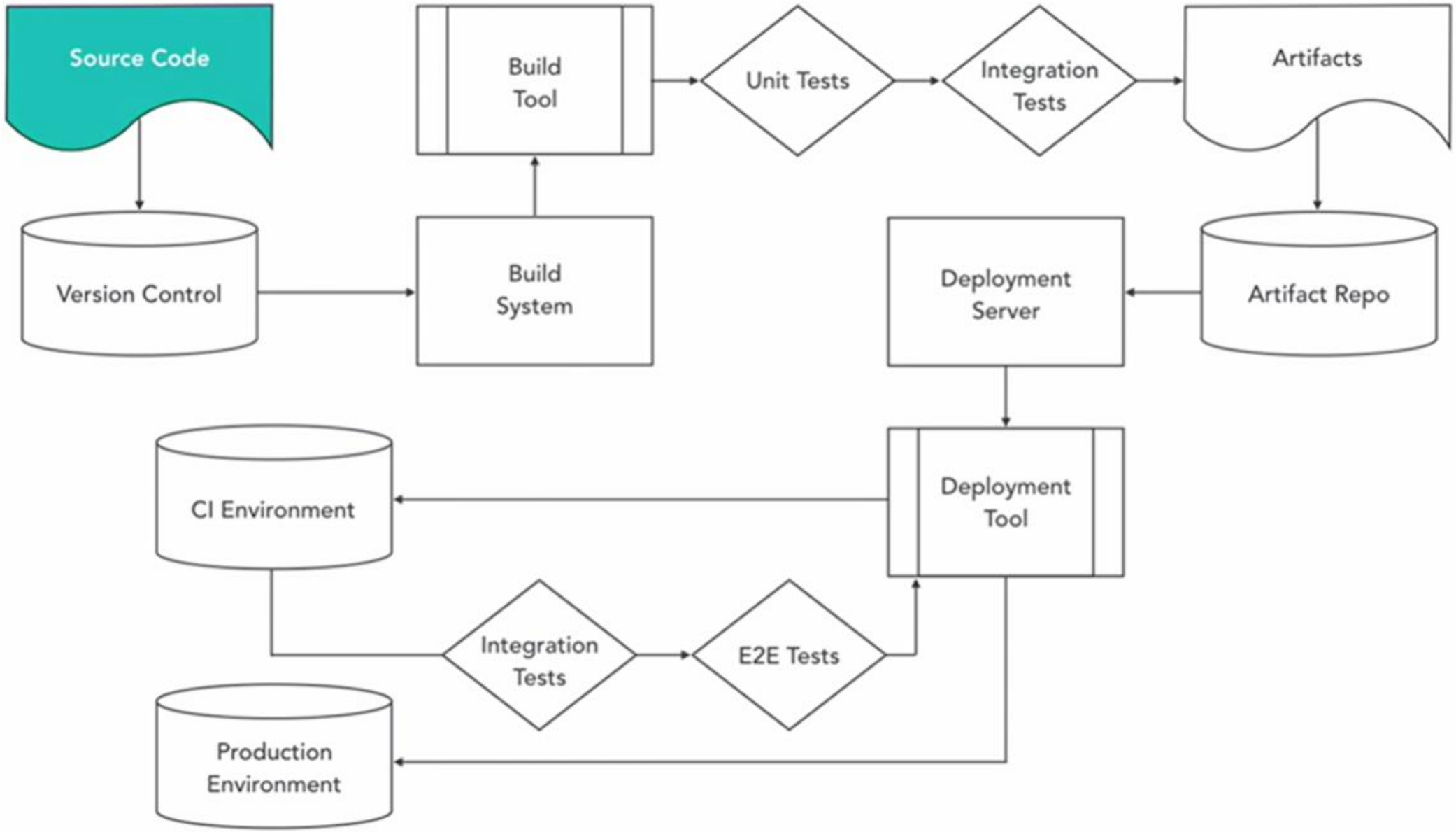
Tests d'intégration système : tests fonctionnels servent à vérifier le bon fonctionnement de l'application dans son ensemble (**Junit** en Java pour ces 2 types de tests)

Contrôle qualité : vérifier que la qualité du code n'a pas été dégradé par les nouveaux changement du code (**SONAR QUBE**)

Vérifier que les dépendance ne comprend pas de faille connu (**JFrog**)

si un problème survient durant ces étapes la chaine s'arrête

Si les tests passent, le nouveau build sera mis dans le dépôt et versionné et sera prêt au déploiement


Source Code
Build Tool
Unit Tests
Integration Tests
Artifacts
Version Control
Build System
Deployment Server
Artifact Repo
CI Environment
Deployment Tool
Integration Tests
E2E Tests
Production Environment

Des nouveaux tests vont être lancer à chaque livraison de l'application sur les différents environnements jusqu'à la production

Des **tests automatisé de recette** via un outil comme **selenium** qui permet de dérouler des scénario de tests complet d'une application

Tester l'écosystème applicatif complet des référentiels, des application en amont et en aval (outil UFT)

Réaliser des **tests de charge et de performance** pour s'assurer que les nouvelles fonctionnalités ne dégradent pas le temps d'accès et l'exécution de l'application, le nombre d'utilisateurs simultané est toujours supporté

On contrôle les nouvelles fonctionnalités mais aussi le non régression

DÉPLOIEMENT SUR UN ENVIRONNEMENT

Plusieurs possibilités :

Déploiement **sur un serveur physique ou machine virtuelle** via un script ou un outil de déploiement automatisé

Pour les applications plus moderne : déploiement **via un conteneur avec une image docker**

AVEC CI / CD

Ça devient plus rapide de développer, tester et mettre en production une nouvelle fonctionnalité

Les tests dans la chaine servent à repérer les erreurs le plutôt possible et les corriger

Les itérations plus petites, plus facile à maitriser et les erreurs auront moins d'impact: facile à détecter et à résoudre

PROCESSUS AUTOMATISÉ

TIME TO MARKET EST DÉCISIF

L'un des objectifs des DEVOPS est d'avoir un time to market plus rapide

Time to market est le délais entre l'idée initiale et sa concrétisation sur le marché d'où l'intérêt de l'automatisation des différents étapes

Avec CI/CD les itérations sont plus petites ce qui permet un retour client plus rapide et d'adapter les prochaines itérations

Dans le **contexte concurrentiel des marché le time to market est décisif** pour répondre à la demande du marché et intégrer les retours utilisateurs rapidement et améliorer ses produits