

# GitLab CI

---

Démo Point de Synchro du 05/04/2024

# Les notions

- Un point d'entrée : `.gitlab-ci.yml`
- Une **pipeline** qui regroupe des **jobs**
- Un **job** = un **conteneur** qui s'exécute sur un **runner**
- Des **templates** et **components** pour réutiliser du code
- Des **variables** et des **inputs** pour paramétrer tout ça

# Objectifs du POC

- **Personne ne veut écrire de pipelines**
- Le moins de configuration possible pour les tâches habituelles
- Intégrer directement un workflow standard
- Mutualiser, rationaliser, simplifier, avoir des *best practices* et documenter

# Les pipelines créées

- Construction d'un conteneur
- Réplication d'un ou plusieurs conteneurs en interne
- Construction d'une chart Helm
- Réplication d'une ou plusieurs charts en interne

# Les outils

- Harbor : registry OCI
- Kaniko : *build* d'images Docker
- Skopeo : copie d'artefacts OCI (conteneurs, charts...)
- Helm : *pull*, *packaging* et *push* de *charts*

# Rappel : CI $\neq$ CD

- Ces actions se limitent à construire et pousser des objets dans la registry !

# Démo : utilisation + explication du workflow

- Exemple avec l'application Keycloak :
  - Mirroring de l'image upstream
  - Construction d'un conteneur pour les thèmes
  - Mirroring de la chart upstream

# Le workflow

1. Itération en DEV
2. Validation pour STG
3. Validation pour PRD

# Le workflow - DEV

- Je développe dans une *branche* git
- Je crée une *Merge Request*, les jobs de DEV s'activent
- Ils s'exécuteront à chaque mise à jour de la *MR* (Push)
- Les artefacts sont traités et envoyés dans la registry DEV
- Je déploie et test en environnement de DEV
- Quand OK, relecture et validation de la MR par un pair
- Merge du code dans la branche *main*

# Le workflows - STG

- Je tague le commit correspondant à la MR précédente
- Les jobs de STG s'activent, les jobs de PRD sont créés mais non déclenchés
- Les artefacts sont traités et envoyés dans la registry STG
- Je suis le process de mise en staging (ticket, validation OPS, etc.)
- Je déploie et test en environnement de STG
- En cas d'échec : procédure de rollback à définir lors du POC de CD

# Le workflow - PRD

- Je déclenche manuellement les jobs de PRD liés à mon tag
- Les artefacts validés sont copiés de la registry STG vers celle de PRD
- Je suis le process de mise en production (ticket, validation OPS, etc.)
- Je déploie et test en environnement de PRD
- En cas d'échec : procédure de rollback à définir lors du POC de CD

# Suite du POC

- Vous faire essayer le workflow et les pipelines
- Avoir un retour concret sur ce qui marche ou non (technique ou pas)
- Itérer
- **#ug\_gitlab-ci**

# Pause

# Partage de Connaissances : implémentation

- Explication du code des composants