L'Unistra se dé-PaaS

Alain ZAMBONI

Guillaume OBERLÉ

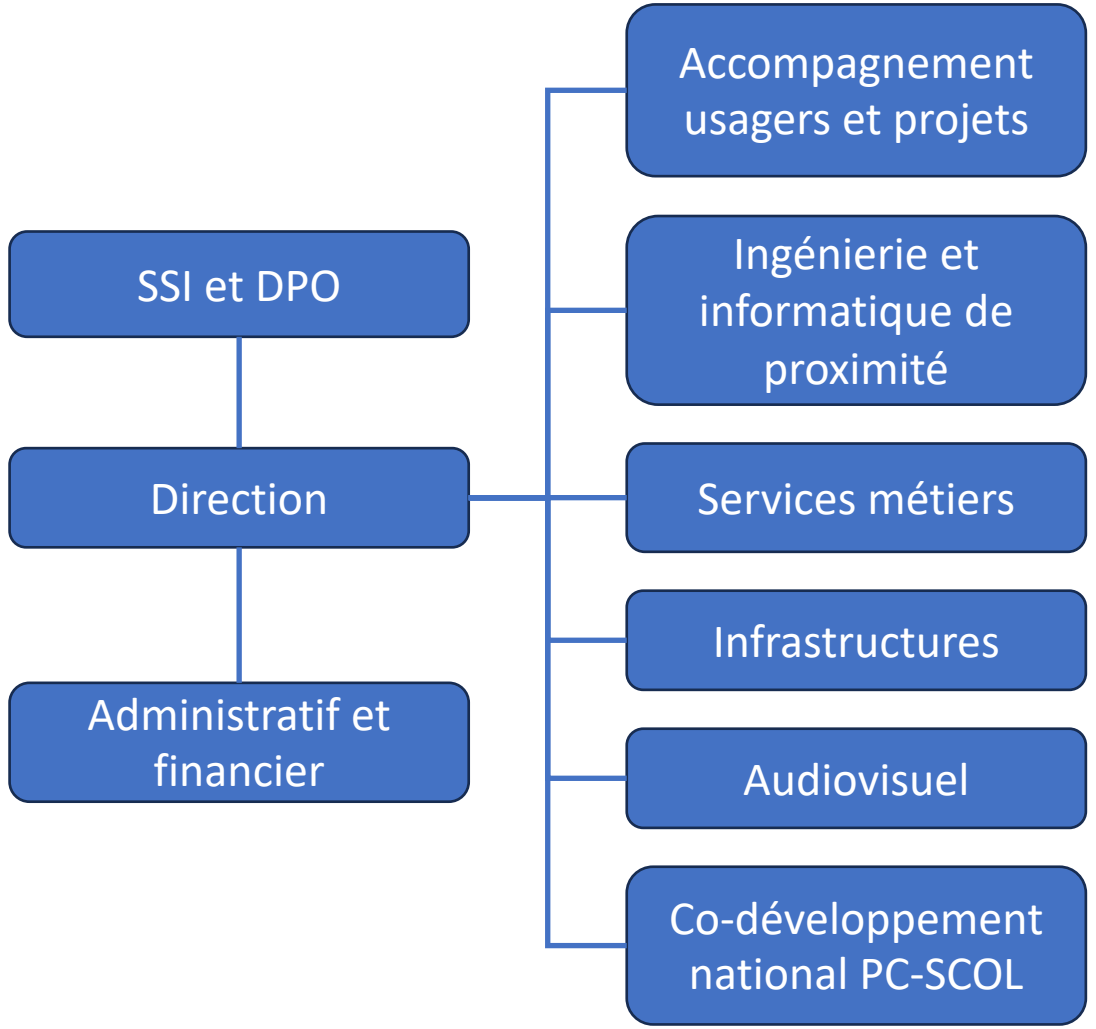
Direction du numérique | DNum
Université de Strasbourg

jres RENNES 2024

Plan

Présentation de l'Unistra

- Quelques chiffres
  - **~55 000** étudiants
  - **~6 100** personnels
  - **40** services/directions
  - **35** composantes de formations
  - **66** unités de recherche
- Direction du Numérique
  - **~160** personnels



Contexte Unistra

- Datacenter construit en 2019
  - Labellisé avec le DC l'université de Lorraine (ADAGE)
- Hébergement d'applications
  - Virtualisation sur RedHat OpenStack / Ceph
  - Déploiement des infras par Terraform
  - Déploiement des applications : Ansible, Docker/Swarm, outils internes, manuel, etc.
- Investissement dans la mutualisation ESR
  - Équipe développement PC-Scol
  - Offre d'hébergement IaaS et S3 (PCScol/Pégase, AMUE, etc.)
  - Hébergement d'applications SaaS : ImmerSup, Campulse

Projet « PaaS-Partout »

- Objectifs
  - Rationaliser les initiatives de conteneurisation des équipes applicatives
  - Être en adéquation avec les pratiques actuelles
  - Améliorer la démarche d'intégration continue et de déploiement continu
  - Passage à l'échelle des offres SaaS de l'Unistra
  - Offre de service d'hébergement ESR
- Débuté en septembre 2022, fin estimée décembre 2025
- Projet transverse aux départements de la DNum

Plan

1. Présentation du projet
2. Étude et choix des outils
3. Déploiement et administration
4. Utilisation de la plateforme
5. Retour d'expérience

Étude et choix des outils

- Phase d'étude
  - Compétences transverses : infra, administrateurs d'applis et développeurs
  - Étude théorique exhaustive d'outils
  - Maquettage des outils sélectionnés
  - Choix des outils basé sur des critères
- Choix d'une infrastructure virtualisée sur notre OpenStack
  - Plus de souplesse
  - Optimisation de l'utilisation des ressource physiques

Étude et choix des outils

Fonction	Retenu	Testés	Raisons principales
Distribution Kubernetes	**OpenShift Plaform Plus (OPP)**	Rancher, KOPS, Ubuntu Charmed	Gestion multi-cluster, intégration avec OpenStack, fonctionnalités pré-packagées (télémétrie, authentification, alerting)
Registry d'images			
Intégration continue			
Déploiement continu			
Gestion des secrets			

Étude et choix des outils

Fonction	Retenu	Testés	Raisons principales
Distribution Kubernetes	**OpenShift Plaform Plus (OPP)**	Rancher, KOPS, Ubuntu Charmed	Gestion multi-cluster, intégration avec OpenStack, fonctionnalités pré-packagées (télémétrie, authentification, alerting)
Registry d'images	**Quay**	Harbor, Nexus Repository	Fonctionnellement complet et support intégré au bundle OPP
Intégration continue			
Déploiement continu			
Gestion des secrets			

Étude et choix des outils

Fonction	Retenu	Testés	Raisons principales
Distribution Kubernetes	**OpenShift Plaform Plus (OPP)**	Rancher, KOPS, Ubuntu Charmed	Gestion multi-cluster, intégration avec OpenStack, fonctionnalités pré-packagées (télémétrie, authentification, alerting)
Registry d'images	**Quay**	Harbor, Nexus Repository	Fonctionnellement complet et support intégré au bundle OPP
Intégration continue	**Gitlab-CI**	Tekton, Jenkins	Déjà utilisé et donne satisfaction, rapport bénéfices/coût de migration insuffisant
Déploiement continu			
Gestion des secrets			

Étude et choix des outils

Fonction	Retenu	Testés	Raisons principales
Distribution Kubernetes	**OpenShift Plaform Plus (OPP)**	Rancher, KOPS, Ubuntu Charmed	Gestion multi-cluster, intégration avec OpenStack, fonctionnalités pré-packagées (télémétrie, authentification, alerting)
Registry d'images	**Quay**	Harbor, Nexus Repository	Fonctionnellement complet et support intégré au bundle OPP
Intégration continue	**Gitlab-CI**	Tekton, Jenkins	Déjà utilisé et donne satisfaction, rapport bénéfices/coût de migration insuffisant
Déploiement continu	**ArgoCD**	FluxCD, Fleet, Spinnaker, Keptn	Fonctionnellement complet, modèle centralisé et support intégré au bundle OPP
Gestion des secrets			

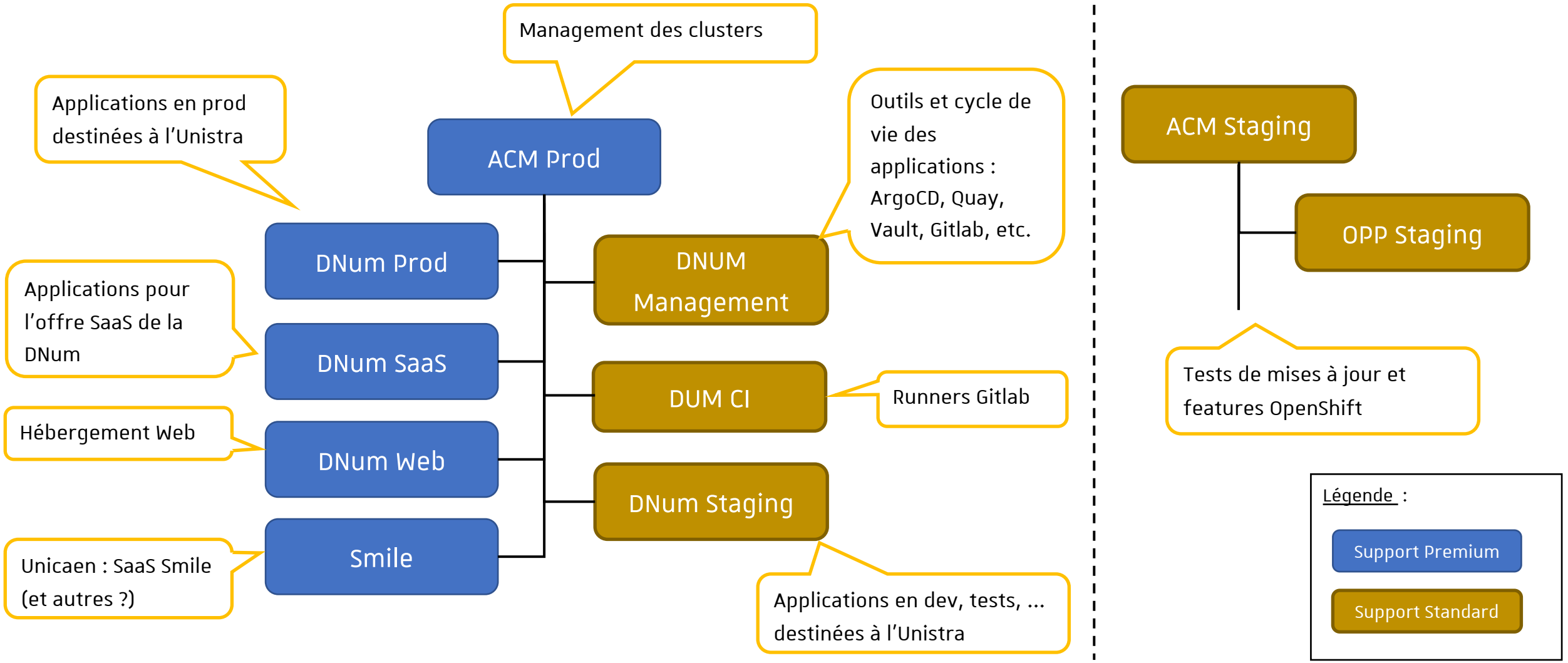
Étude et choix des outils

Fonction	Retenu	Testés	Raisons principales
Distribution Kubernetes	**OpenShift Plaform Plus (OPP)**	Rancher, KOPS, Ubuntu Charmed	Gestion multi-cluster, intégration avec OpenStack, fonctionnalités pré-packagées (télémétrie, authentification, alerting)
Registry d'images	**Quay**	Harbor, Nexus Repository	Fonctionnellement complet et support intégré au bundle OPP
Intégration continue	**Gitlab-CI**	Tekton, Jenkins	Déjà utilisé et donne satisfaction, rapport bénéfices/coût de migration insuffisant
Déploiement continu	**ArgoCD**	FluxCD, Fleet, Spinnaker, Keptn	Fonctionnellement complet, modèle centralisé et support intégré au bundle OPP
Gestion des secrets	**Hashicorp Vault**	Sealed Secrets, CyberArk Conjur	Centralisation des secrets et polyvalence fonctionnelle (même sans support)

Plan

1. Présentation du projet
2. Étude et choix des outils
3. Déploiement et administration
4. Utilisation de la plateforme
5. Retour d'expérience

Architecture des clusters OpenShift



Déploiement des clusters

- Utilisation de Advanced Cluster Management (ACM)
  - Outil de gestion multi-cluster
    - Déploiement, configuration et supervision de clusters managés
  - Cluster OpenShift dédié déployé manuellement
- Déploiement des clusters managés
  - Automatisé via l'utilisation de CustomResourceDefinition (CRD) fournis par ACM
  - Création d'un chart Helm pour instancier les clusters
  - Un fichier de secrets par cluster

Déploiement des clusters

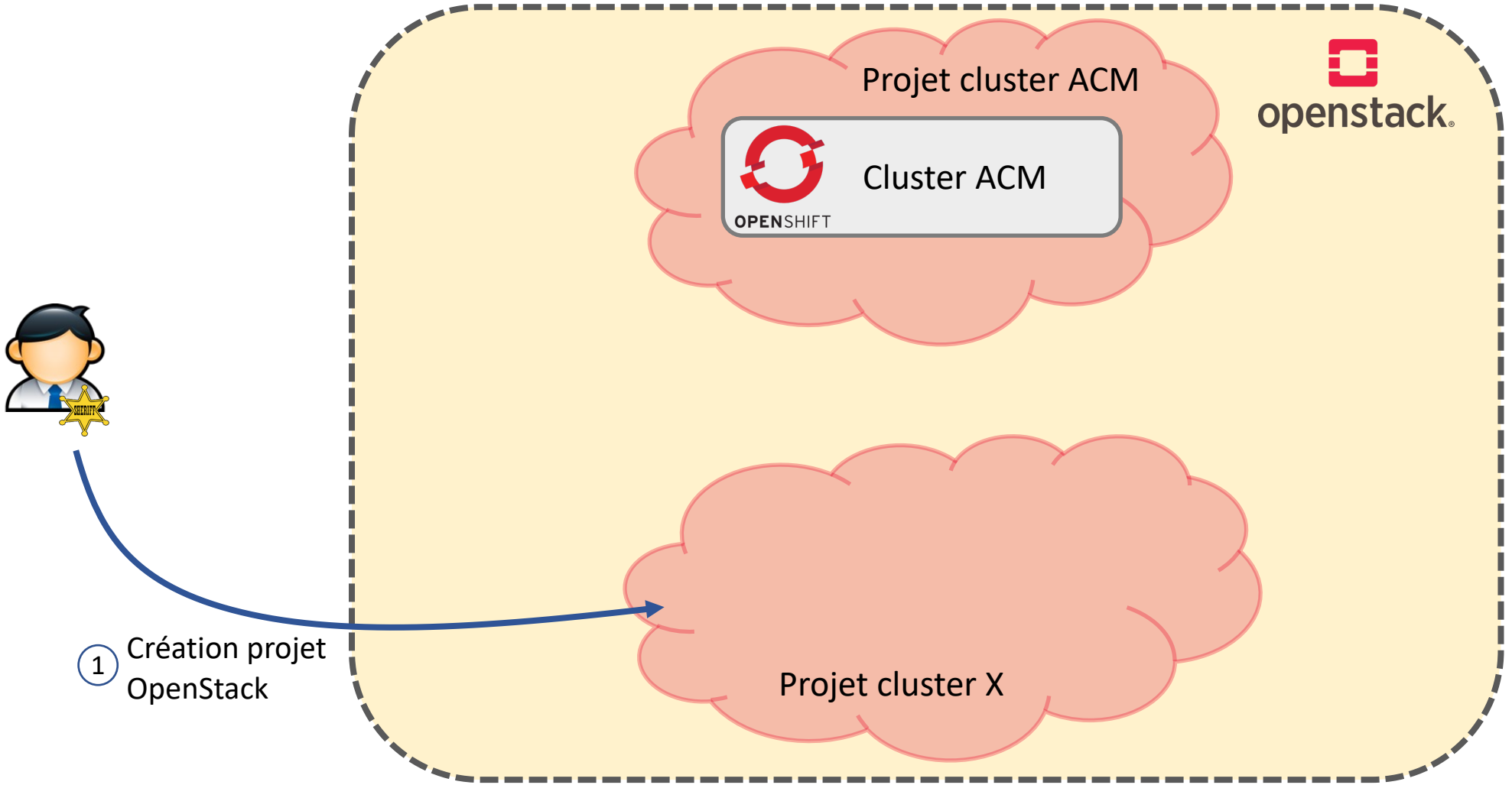
Projet cluster ACM

OPENSHIFT

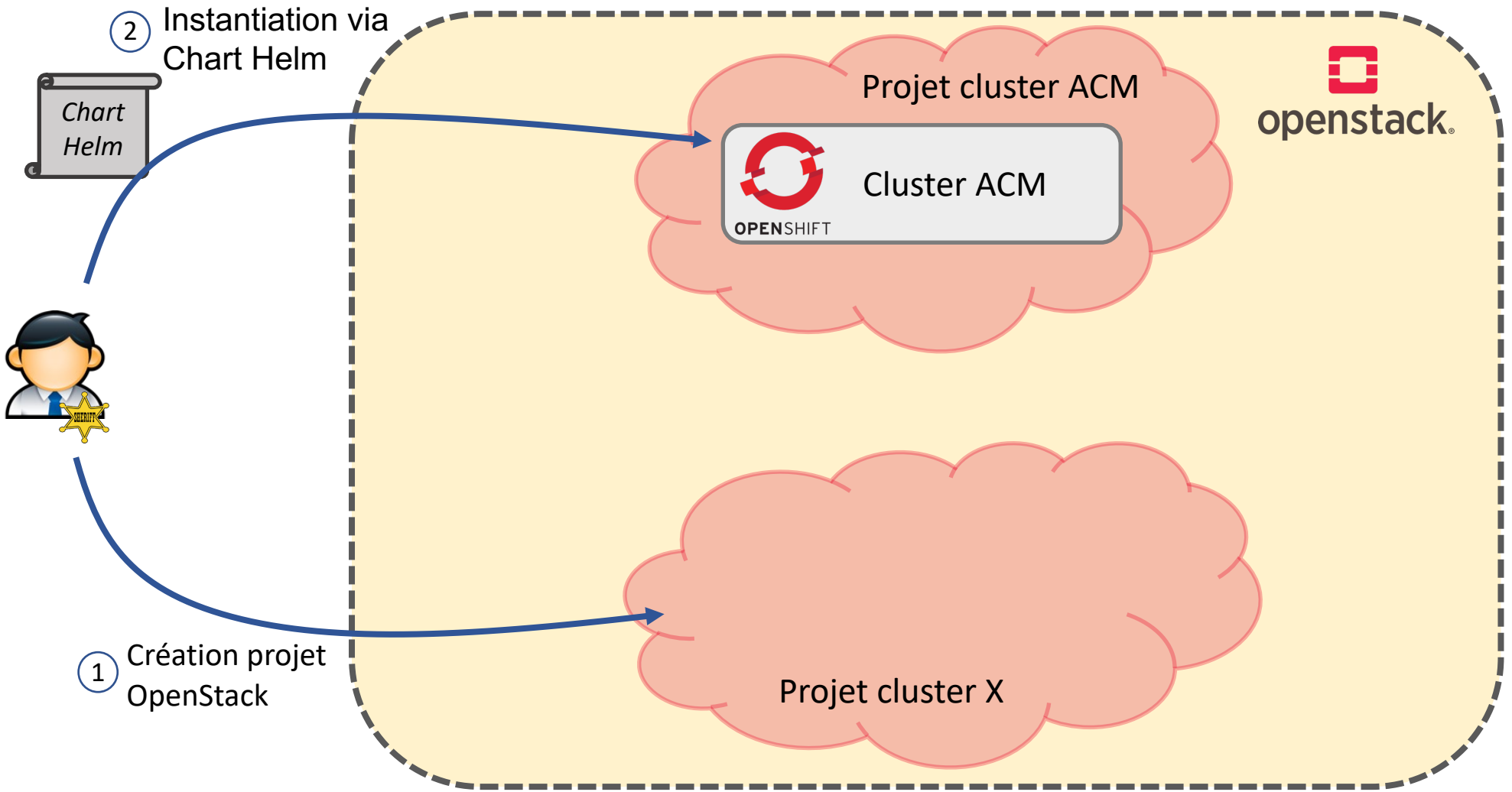
Cluster ACM

openstack.

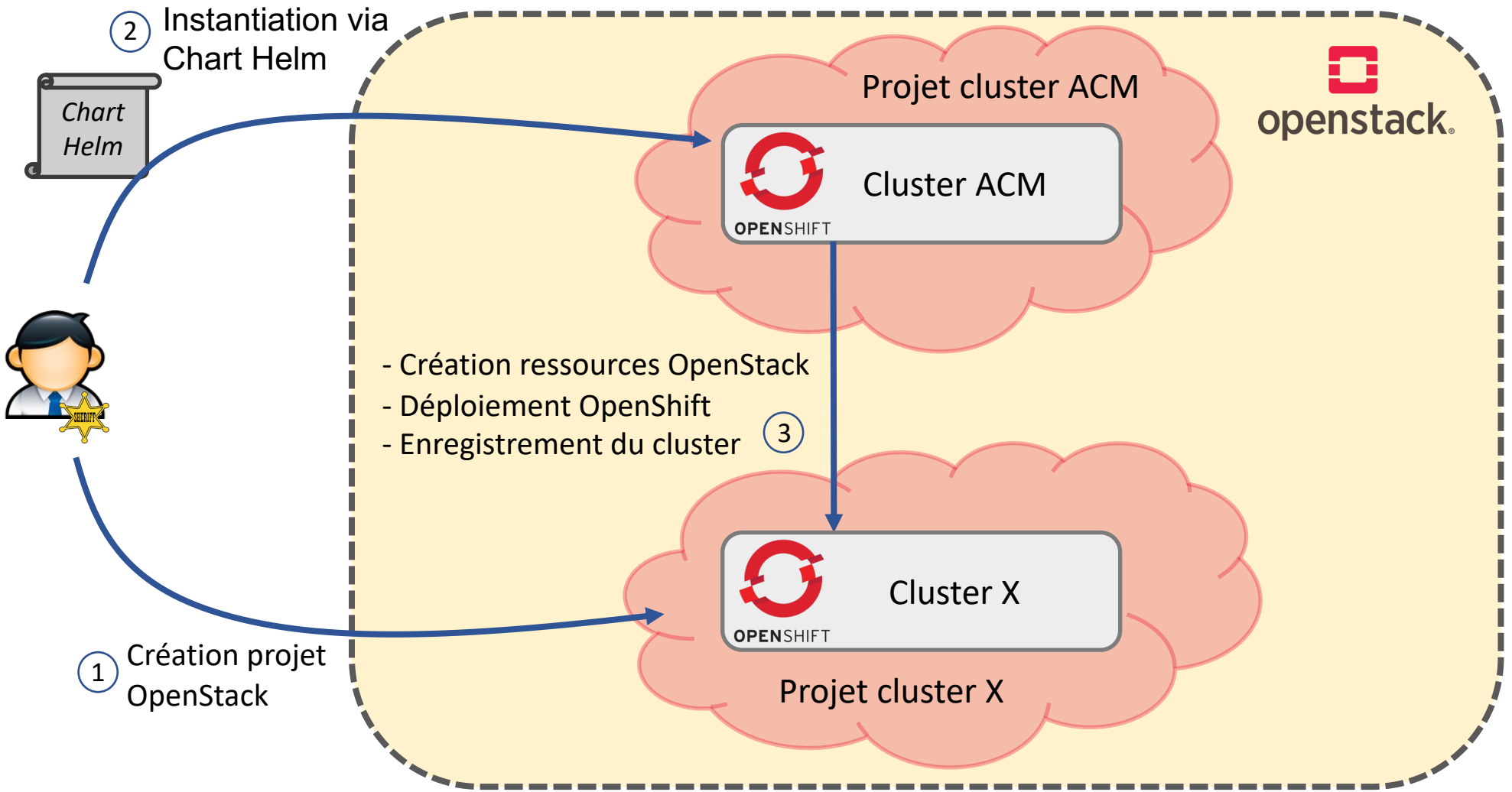
Déploiement des clusters



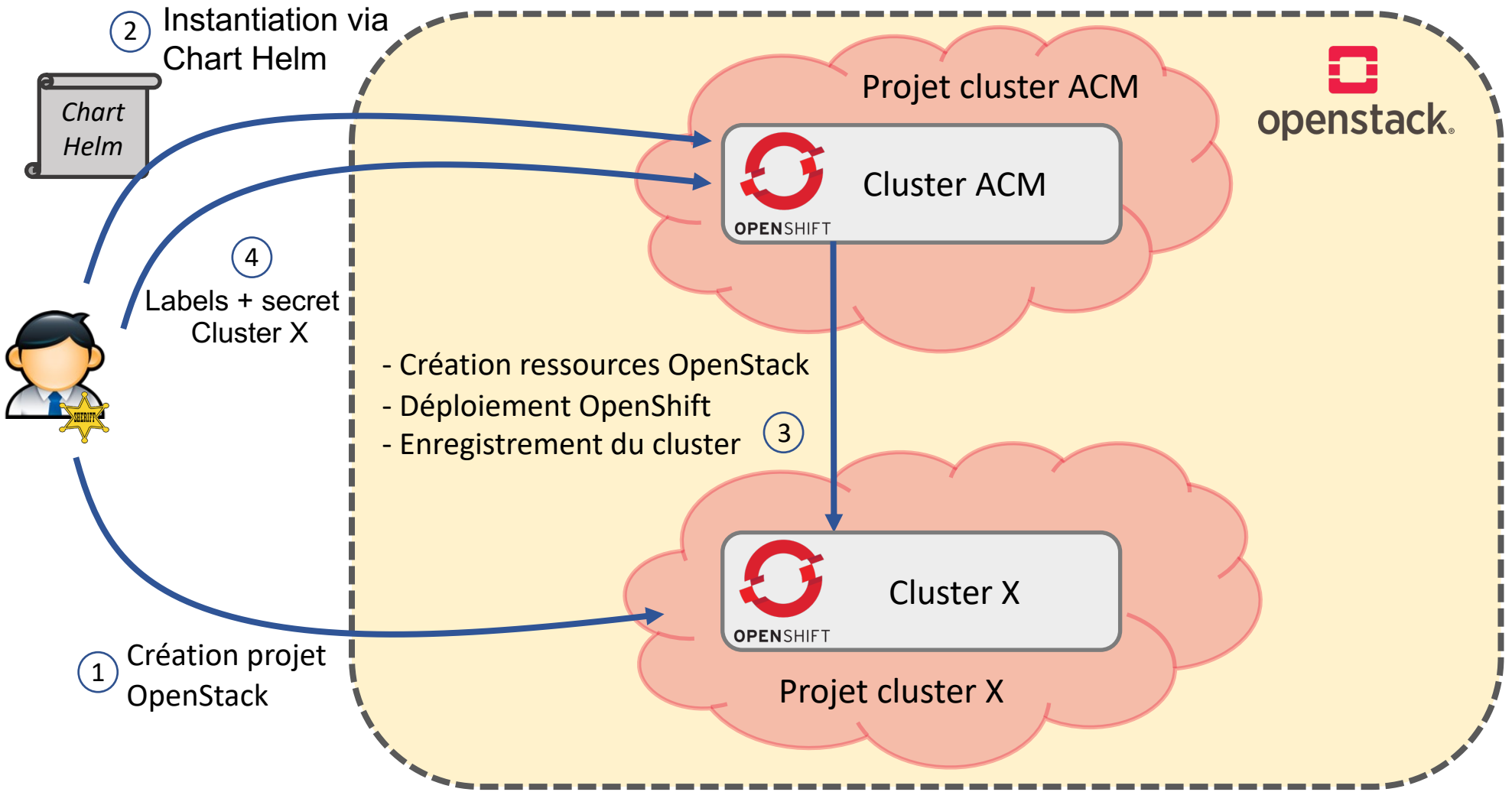
Déploiement des clusters



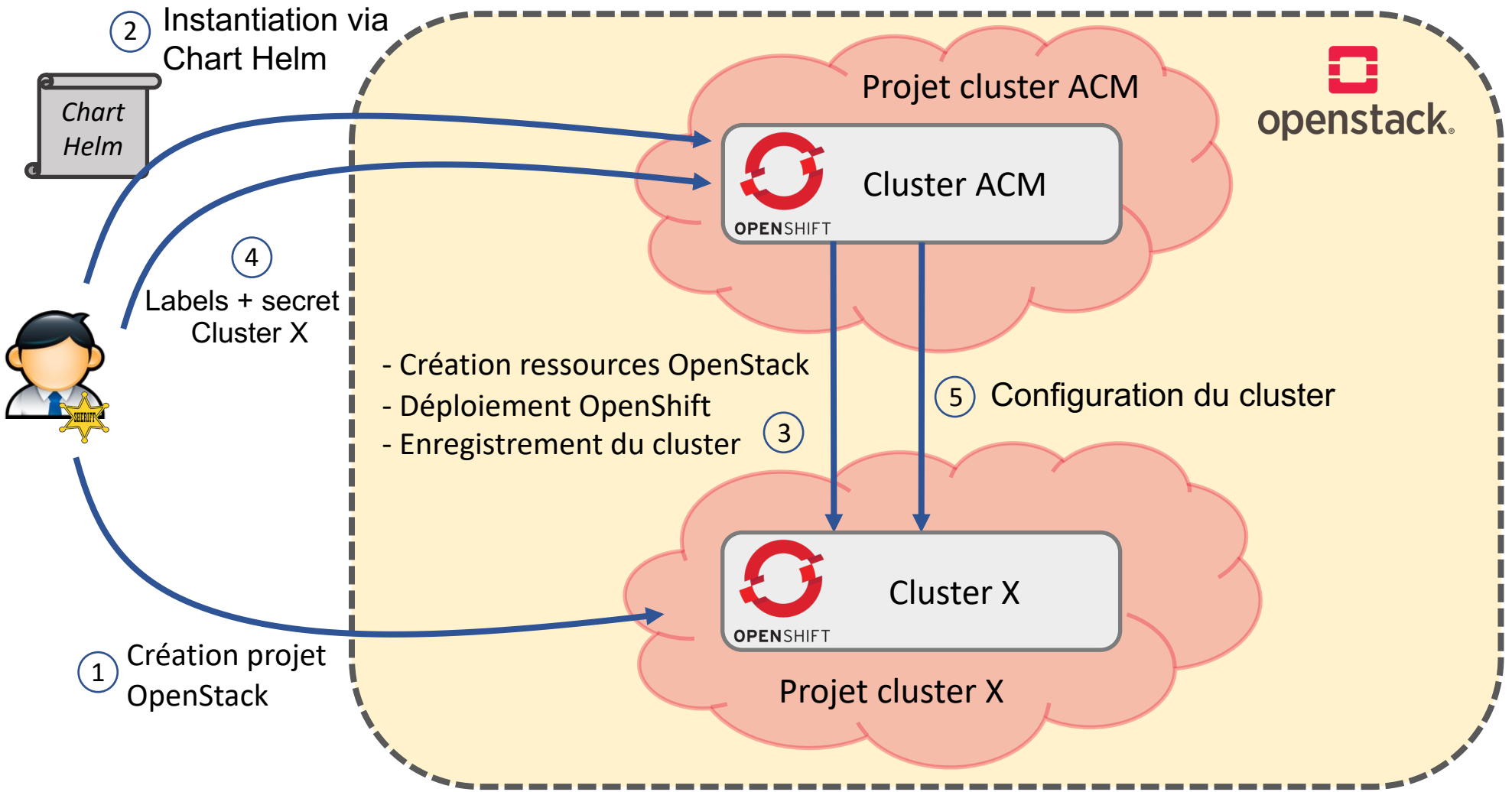
Déploiement des clusters



Déploiement des clusters



Déploiement des clusters



Configuration des clusters

- Utilisation du mécanisme de policies d'ACM
  - Déclaration de l'état souhaité sur le cluster ACM (via des objets Kubernetes avec une CRD spécifique)
  - Mise en conformité par l'agent sur le cluster managé
- Vérification de la présence (ou non) d'objets Kubernetes
  - Règle de placement pour cibler plusieurs clusters : cluster-sets, noms, labels, etc.
  - Alerte ou remédiation
- Évaluation des policies toutes les X secondes
- Système de templating : Go templates
  - variables lues sur le cluster hub et managé : configMap, Secret, etc.

Configuration des clusters

- Quelques exemples de configurations via des policies
  - Métrologie et alerting (Prometheus, Grafana, Alert-manager)
  - Authentification 2FA avec Keycloak (OpenID)
  - Storage class
  - Default role binding
  - Supervision du cluster (sondes Centreon)
- Applications déployées par policies via opérateurs
  - Cert-manager
  - OpenShift Gitops (ArgoCD)
  - Quay
  - Agent Vault

Plan

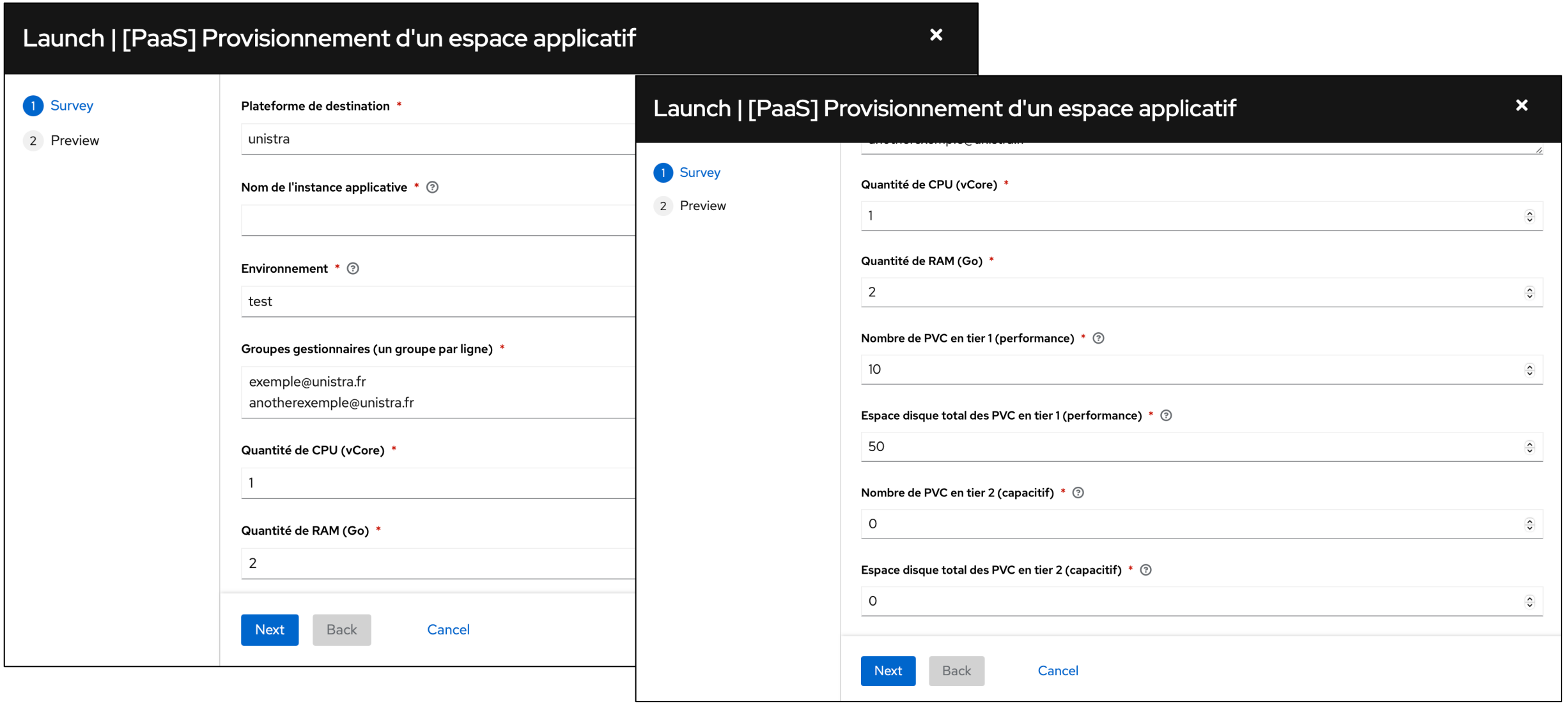
1. Présentation du projet
2. Étude et choix des outils
3. Déploiement et administration
4. Utilisation de la plateforme
5. Retour d'expérience

Utilisation de la plateforme

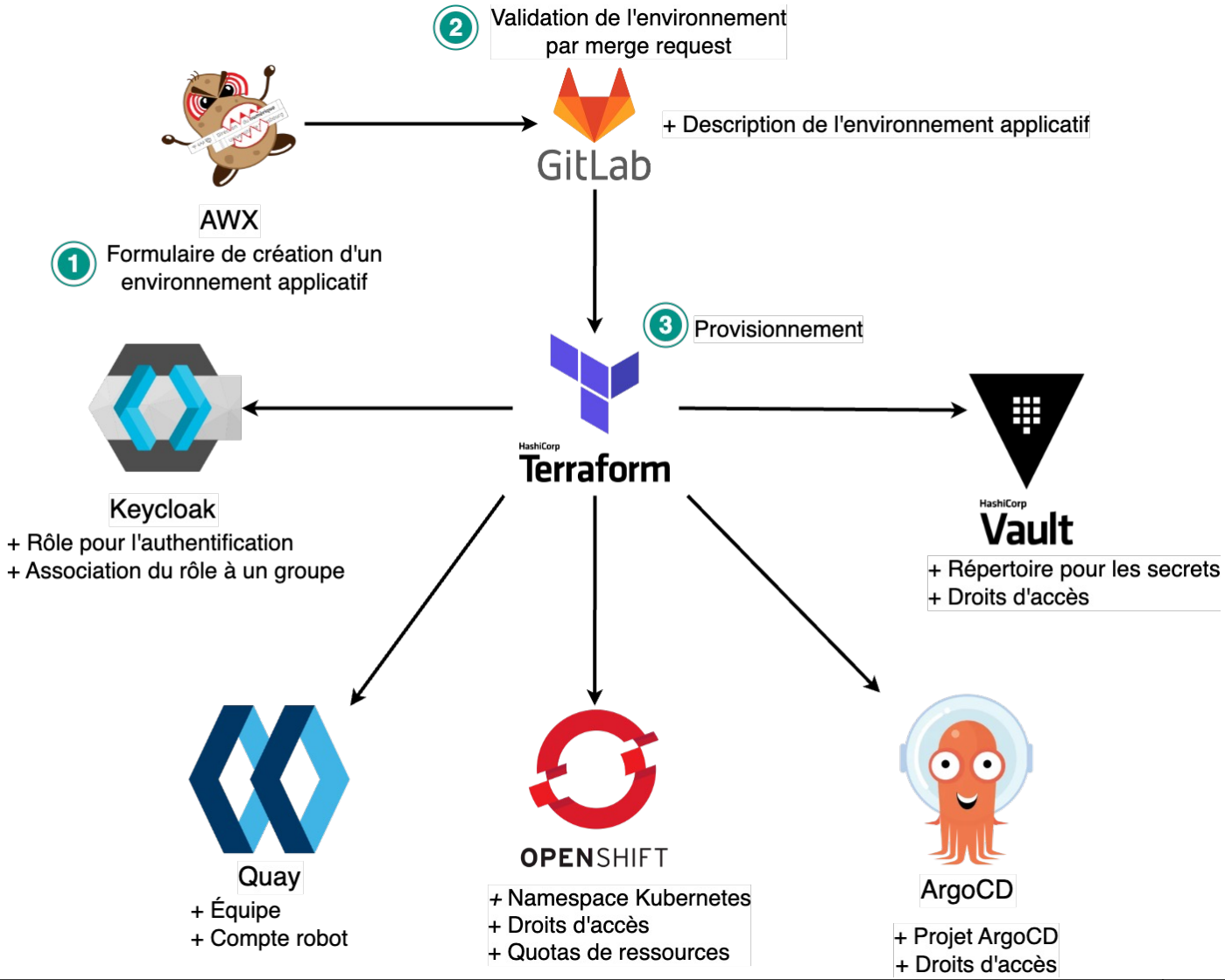
- Clusters partagés par plusieurs équipes applicatives
  - Nécessité de cloisonner les accès aux environnements applicatifs
  - Utilisation des mécanismes multi-tenant des outils

- Automatisation du provisionnement des environnements applicatifs
  - Garantie une mise à disposition rapide et reproductible
  - Permet d'homogénéiser les environnements et les droits d'accès
  - Délégation des demandes de création d'environnements aux équipes applicatives

Création d'environnements PaaS



Création d'environnements PaaS

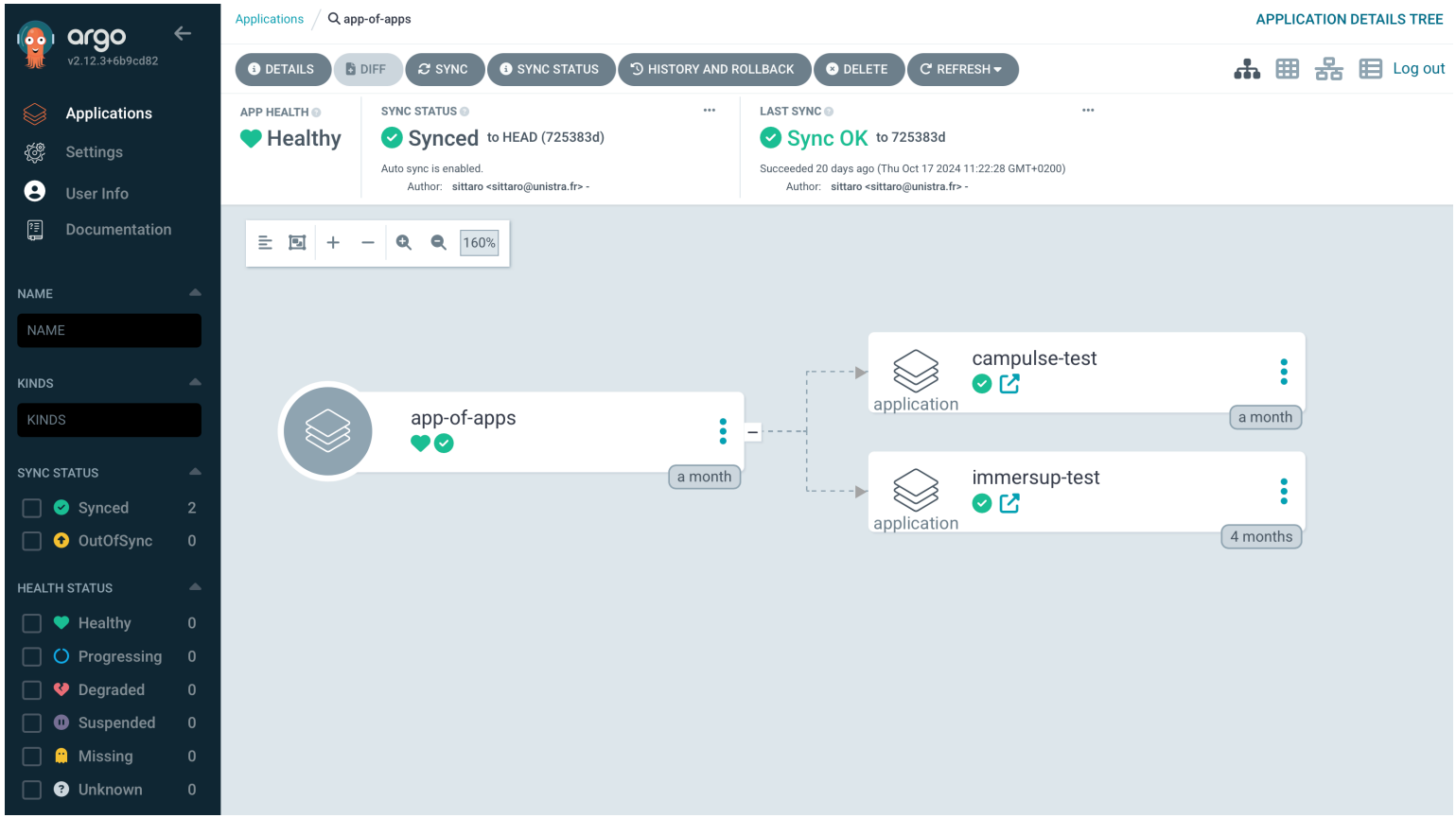


Déploiement des applications

- Utilisation d'ArgoCD et du modèle App-of-Apps

GitLab

```
README.md
app-of-apps.yaml
apps
    campulse
        campulse-test.yaml
    immersup
        immersup-test.yaml
```



Utilisation de la plateforme – les pilotes Unistra

- Choix d'applications pilotes à l'Unistra
  - ImmerSup : immersions des lycéens
  - Campulse : associations étudiantes
- Organisation d'ateliers avec les équipes
  - Création et/ou adaptation des Dockerfile
  - Intégration du build de l'image à la CI
  - Création de charts Helm et intégration à la CD
  - Base de données en conteneurs : tests de l'opérateur CloudNativePG

```yaml
apiVersion: argoproj.io/v1alpha1
kind: Application
metadata:
  name: campulse-test
  namespace: openshift-gitops
spec:
  project: campulse-test
  sources:
    - repoURL: https://xxx.unistra.fr/di/helm/campulse-front.git
      path: .
      targetRevision: 1.0.1
      helm:
        valueFiles:
          - values.yaml
    - repoURL: https://xxx.unistra.fr/di/helm/campulse-back.git
      path: .
      targetRevision: 1.0.1
      helm:
        valueFiles:
          - values.yaml
    - repoURL: https://xxx.unistra.fr/di/helm/campulse-bdd.git
      path: .
      targetRevision: 1.0.1
      helm:
        valueFiles:
          - values.yaml
  destination:
    server: https://xxx.unistra.fr:6443
    namespace: campulse-test
```

Plan

1. Présentation du projet
2. Étude et choix des outils
3. Déploiement et administration
4. Utilisation de la plateforme
5. Retour d'expérience

Retour d'expérience – la plateforme PaaS

- Les points positifs
  - Facilité de déploiement et intégration avec OpenStack
  - Nombreuses briques par défaut (dashboards, métrologie, auth, etc.)
  - Facilité de mise à jour
  - Installation/gestion du cycle de vie d'outils via des opérateurs
  - Sécurité plus ferme que d'autres distributions (conteneurs rootless, UID aléatoire)
  - Documentation complète
  - Outils CI/CD : actuellement satisfaisants, travaux à poursuivre
- Les points de vigilance
  - Distribution Kubernetes assez consommatrice en ressources
  - Complexité de prise en main (ex: ressources auto-gérées par des opérateurs)
  - Tarifs : risque de frein pour certaines structures

Retour d'expérience – la plateforme PaaS

- Prochaines étapes
  - Automatisation et démarche GitOps
    - Déploiement des clusters par ArgoCD
    - Provisionning des environnements applicatifs par ArgoCD
  - Service Mesh et communication multi-cloud pour les applications (Istio, Skupper, Consul, Kuma, etc.)
  - Amélioration de la sécurité
    - Exploration du module « Advanced Cluster Security »
    - Évaluation d'outils tierces de surveillance d'activité (Falco, Neuvector, etc.)

Retour d'expérience – les usages PaaS à l'Unistra

- Une transformation majeure, coûteuse et longue
  - Formations, nouvelles pratiques, adaptations des applications, etc.
- Une démarche indispensable
  - Garantir la modernité de nos infrastructures et pratiques
  - Structurante et fédératrice pour l'organisation
- Un défi accessible
  - L'expérience par la mise en pratique
  - Bonne capitalisation des acquis
  - Résultats encourageants

Retour d'expérience – les usages PaaS à l'Unistra

- Prochaines étapes
  - Mise en production des applications Campulse et ImmerSup
  - Extension sur d'autres applications
  - Évaluation d'OpenShift Serverless (Knative) dans le cadre du projet « DynamicMood »
  - Nœuds GPU et Intelligence Artificielle

Utilisation de la plateforme – les pilotes ESR

- Partenariat pilote avec l'Université de Caen
  - Hébergement d'applications en SaaS dans le cadre d'ESUP
  - Fourniture d'un cluster OpenShift + outils (ArgoCD, Quay, etc.)
- Rationalisation de l'exploitation de la plateforme
- Collaboration fructueuse et enrichissante
- Définition de l'offre standard à effectuer
  - Échange avec des partenaires potentiels (Unistra, régionaux)
  - Services proposés ? Limite de responsabilité ?
  - Définition des tarifs

Questions ?