

Direction



du numérique

Université de Strasbourg

PaaS Unistra

journée “déploiement d'applications”

ESUP

13/06/2024

Guillaume Oberlé - Alain Zamboni

Plan

- Présentation du projet
- Choix des briques logicielles
- Architecture
- Administration des clusters
- Gestions des espaces applicatifs
- Conclusion

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 application :
 - Ansible, docker, outils internes, manuel, etc.
- Investissement dans la mutualisation ESR
 - Équipe développement PC-Scol
 - Offre d'hébergement IaaS (PC-Scol/Pégase, AMUE, etc.)
 - Stockage S3
 - Hébergement SaaS d'applications : Immersup, Campulse

Pourquoi le projet

- Être en adéquation avec les pratiques actuelles
- Rationaliser les initiatives de conteneurisation des équipes applicatives
- Améliorer la démarche d'intégration continue
- Initier une démarche de déploiement continue
- Passage à l'échelle des offres SaaS de l'Unistra
- Offre de service d'hébergement ESR

Objectifs de l'offre de service

- Unistra
 - DNum : Hébergement interne d'applications
 - DNum : Hébergement d'applications en SaaS
 - Composante et labos Unistra
- ESR
 - Namespace a a Service
 - Cluster as a Service
 - Pilote : Unicaen pour offre SaaS Smile
 - Cluster dédié administré par l'Unistra
 - Outils fournis : ArgoCD, Cert-manager, Quay, Vault
 - Unicaen est autonome sur la gestion des namespaces

Plan

- Présentation du projet
- Choix des briques logicielles
- Architecture
- Administration des clusters
- Gestions des espaces applicatifs
- Conclusion

Choix des briques logicielles

- Phase d'étude
 - Compétences transverses : infra, administrateurs d'applis et développeurs
 - Étude théorique exhaustive d'outils
 - Maquettage d'une sélection d'outils
- Choix d'une infrastructure virtualisée sur notre OpenStack
 - plus de souplesse
 - optimisation de l'utilisation des ressources physiques
- Choix d'une distribution Kubernetes avec support

Choix des briques logicielles

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

souligné : félicitations du jury

Bundle OpenShift Platform Plus

 **Red Hat**
OpenShift
Kubernetes Engine

Includes:

- Enterprise Kubernetes runtime
- Red Hat Enterprise Linux CoreOS immutable container operating system
- Administrator console
- Red Hat OpenShift Virtualization

 **Red Hat**
OpenShift
Container Platform

Adds:

- Developer console
- Log management and metering/cost management
- Red Hat OpenShift Serverless (Knative)
- Red Hat OpenShift Service Mesh (Istio)
- Red Hat OpenShift Pipelines and Red Hat OpenShift GitOps (Tekton, ArgoCD)

 **Red Hat**
OpenShift
Platform Plus



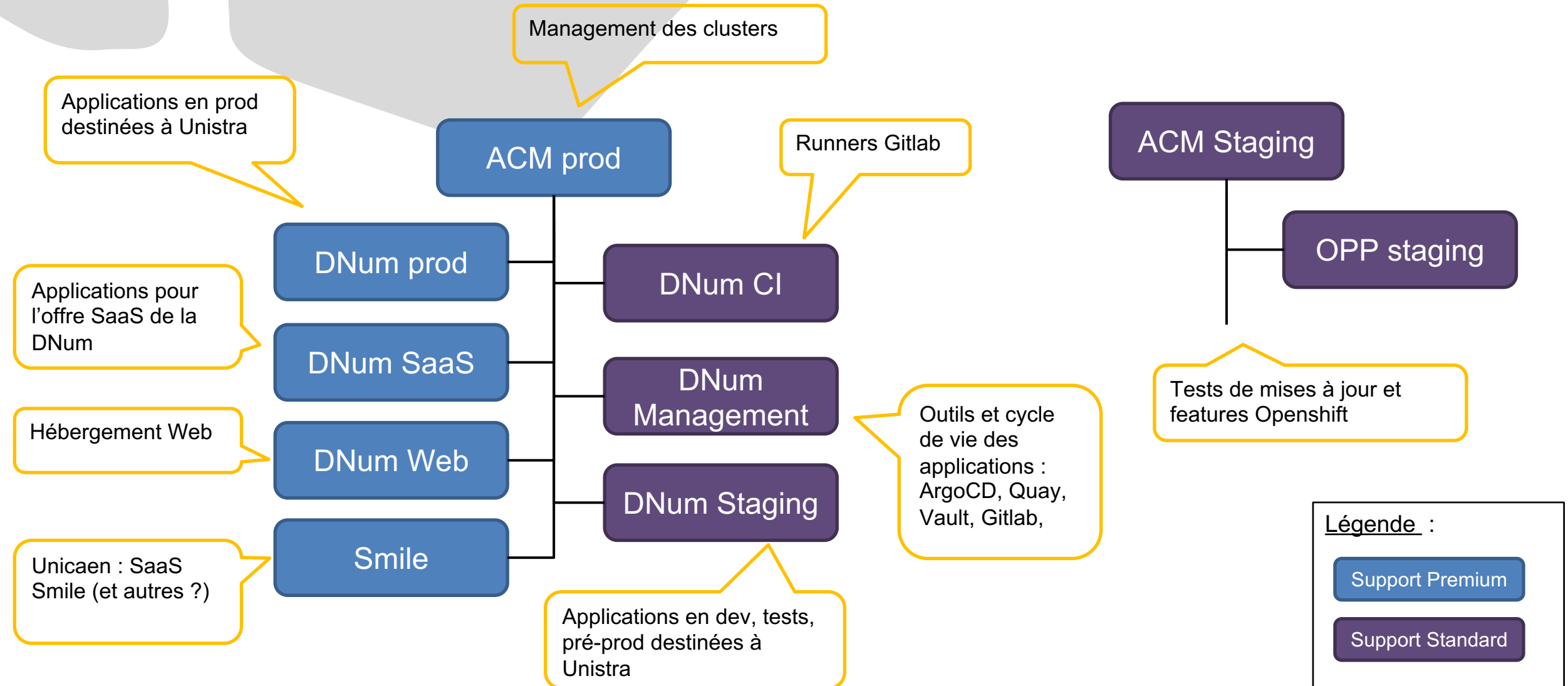
Adds:

- Red Hat Advanced Cluster Management for Kubernetes
- Red Hat Advanced Cluster Security for Kubernetes
- Red Hat Quay

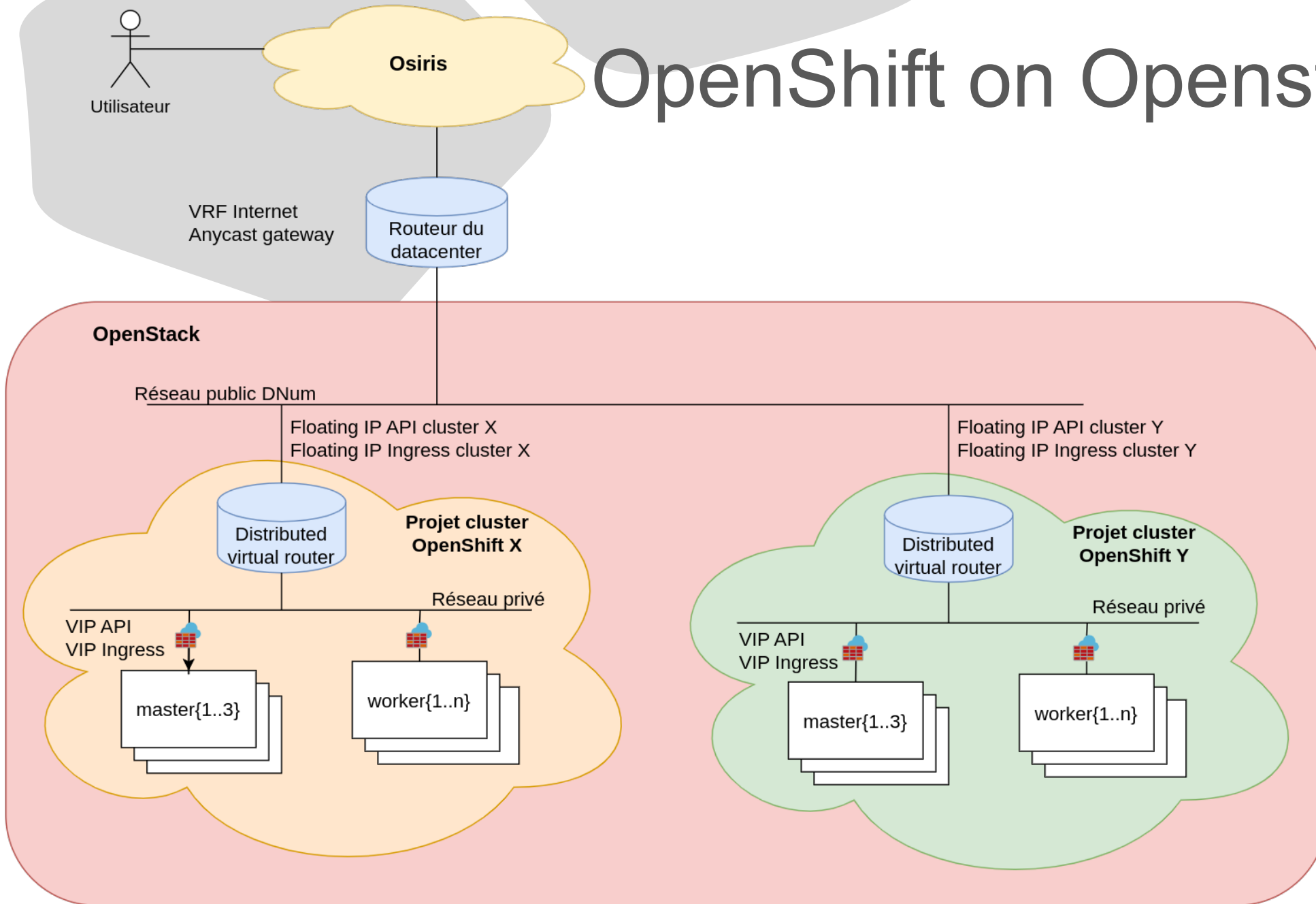
Plan

- Présentation du projet
- Choix des briques logicielles
- Architecture
- Administration des clusters
- Gestions des espaces applicatifs
- Conclusion

Architecture - Clusters OpenShift



OpenShift on Openstack



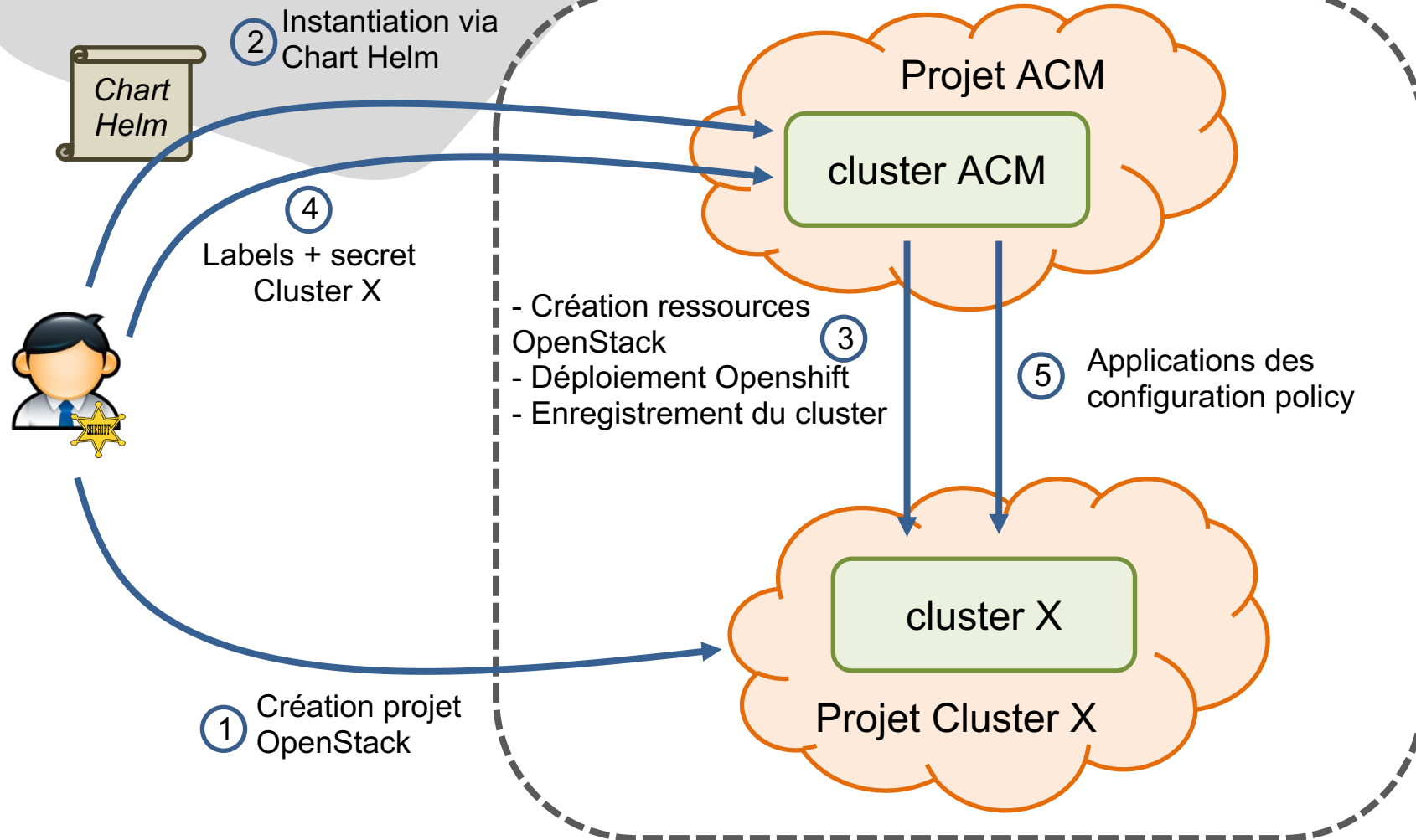
Plan

- Présentation du projet
- Choix des briques logicielles
- Architecture
- Administration des clusters
- Gestions des espaces applicatifs
- Conclusion

Administration de la plateforme

- Utilisation de Advanced Cluster Management (ACM)
 - Cluster ACM d'administration
 - Déploiement automatisé de clusters
 - Configuration des clusters fils par politiques
- Administration des clusters
 - Supervision des clusters via sondes Centreon
 - Métrologie centralisé des clusters via Grafana/Prometheus
 - Authentification 2FA avec Keycloak (OpenID)
- Gestion des accès utilisateurs
 - Provisionnement d'environnements PaaS automatisés

Déploiement de clusters



Configuration des clusters

- Utilisation du mécanisme de politiques d'ACM
- Architecture Hub-Spoke
 - Déclaration de l'état souhaité sur le cluster hub (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, ...
 - Alerte ou remédiation
- Évaluation des politiques toutes les X secondes
- Système de templating : Go templates
 - variables lues sur le cluster hub et managé : configMap, Secret, etc.

Quelques polices mises en place

- Configuration
 - Alerting (alert-manager)
 - Authentication
 - Storage Class
 - Default role binding
 - Supervision
- Appli déployés par polices via opérateurs
 - Cert-manager
 - OpenShift Gitops (ArgoCD)
 - Quay
 - Agent Vault

Plan

- Présentation du projet
- Choix des briques logicielles
- Architecture
- Administration des clusters
- Gestions des espaces applicatifs
- Conclusion

Création d'environnements PaaS

Lancer | [DEV] Provisionnement Projet PaaS

1 Questionnaire

2 Prévisualisation

Nom du projet *

Type d'hébergement * ⓘ

Environnement à créer *

Groupes gestionnaires pour l'environnement (Un groupe par ligne) *

Quantité de CPU (vCore) demandé pour l'environnement *

Quantité de RAM (Go) demandée pour l'environnement *

Suivant

Retour

Annuler

Lancer | [DEV] Provisionnement Projet PaaS

1 Questionnaire

2 Prévisualisation

anotherexemple@unistra.fr

Quantité de CPU (vCore) demandé pour l'environnement *

Quantité de RAM (Go) demandée pour l'environnement *

Nombres de disques (PVC) en Tier 1 (SSD) pour l'environnement * ⓘ

Taille totaux des disques (PVC) en Tier 1 (SSD) en Go pour l'environnement *

Nombres de disques (PVC) en Tier 2 (HDD) pour l'environnement * ⓘ

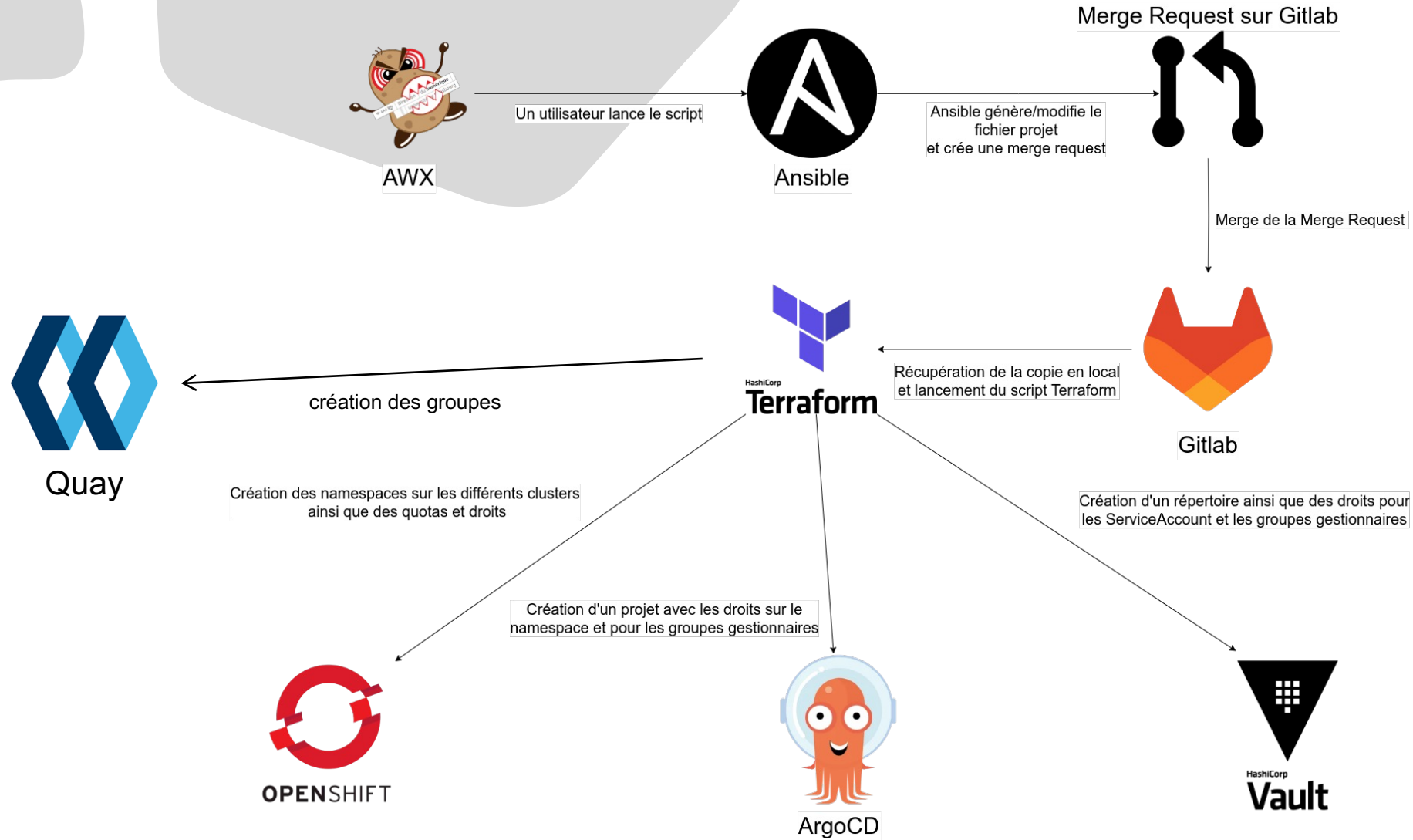
Taille totaux des disques (PVC) en Tier 2 (HDD) en Go pour l'environnement *

Suivant

Retour

Annuler

Création d'environnements PaaS



Habilitation par défaut

- OpenShift
 - Authentification : login/mdp + 2FA (keycloak)
 - Découpage en projet / namespace (1:1) par application et environnement
 - Accès aux projets : groupe de gestionnaires
 - Permissions par défaut sur les projets : lecture/écriture
- ArgoCD
 - Authentification : via OpenShift
 - Découpage en projet par application et environnement
 - Accès aux projets : groupe de gestionnaires
 - Permissions par défaut sur les projets : lecture/écriture

Habilitation par défaut

- Vault
 - Authentification : login/mdp + 2FA (keycloak)
 - Découpage par dossier avec accès restreint par application et environnement
 - Accès aux secrets : groupe de gestionnaire
 - Permissions par défaut : lecture/écriture
- Quay :
 - Authentification : login/mdp + 2FA (keycloak)

Plan

- Présentation du projet
- Choix des briques logicielles
- Architecture
- Administration des clusters
- Gestions des espaces applicatifs
- Conclusion

Conclusion

- Ce qui est bien
 - Facilité de déploiement et intégration avec OpenStack
 - Nombreuses briques par défaut (dashboards, métrologie, auth, ...)
 - Installation/gestion du cycle de vie d'outils via des opérateurs
 - Documentation complete
- Ce qui est moins bien
 - Distribution Kubernetes assez consommatrice en ressources
 - Complexité de prise en main (ex: ressources auto-gérées par des opérateurs)

Conclusion

- Prochaines étapes
 - Rôder le processus de mise à disposition d'environnements
 - Déployer des applications pilotes
 - Outils à tester
 - Communication multi-cloud entre les applications (Skupper, Consul, ...)
 - Serverless (Knative)
 - Service Mesh (Istio)
 - Advanced Cluster Security
 - Stockage file system partagé (CephFS ?)
 - Préparer une offre de service ADAGE avec l'UL



Questions ?