Retour d'expérience du déploiement et de l'exploitation d'une PaaS sur plusieurs IaaS de l'ESR à destination d'une communauté scientifique

Philippe Depouilly, Sylvain Allemand, Romain Théron, Damien Ferney, Henri Massias

Retour d’expérience du déploiement et de l’exploitation d’une solution PaaS sur plusieurs IaaS de l’ESR à destination d’une communauté scientifique

Philippe Depouilly
IMB
351 cours de la Libération
33 405 Talence

Damien Ferney
LMBP
3 Place Vasarely
63 178 Aubière

Romain Théron
Institut Denis Poisson
Rue de Chartres
45 067 Orléans

Sylvain Allemand
IMB
351 cours de la Libération
33 405 Talence

Henri Massias
XLIM
123 avenue Albert Thomas
87 060 Limoges

Résumé

Nous présentons une expérience originale qui avait pour objectif d’intégrer au sein d’une plateforme unique l’ensemble des services et des applications destinés à la communauté mathématiques française. En effet, en tentant d’exploiter au mieux les aspects multi-tenant de la distribution Openshift, nous souhaitions proposer une solution PaaS intégrant les applications d’infrastructure, des services nationaux proposés à notre communauté voire à l’ensemble de l’ESR ainsi que les développements et les applications des utilisateurs. Dès le départ, nous avons déployé notre solution en impliquant les utilisateurs, en leur mettant à disposition, en connaissance de cause, une solution que nous apprenions tout juste à maîtriser pour progressivement arriver à une offre de service que nous souhaitions riche et innovante. Cet article a pour objet de réaliser un retour d’expérience de cette mise en œuvre, en particulier des technologies autour de l’orchestration de conteneurs, de Kubernetes et Openshift. Nous y aborderons les différentes étapes, ce qui a motivé nos choix techniques, ce qui a été réalisable ou non et surtout ce que nous en avons retiré.

Mots-clefs

Kubernetes, Opérateurs, Openshift, OKD, IaaS, Openstack Ceph

1 Les premiers pas

En 2017, naissent, entre des membres du Groupe Calcul et du réseau métier Mathrice, de nouveaux échanges sur la volonté de collaborer autour de projets communs et de voir s'il est possible de déployer des plateformes communes pour la mise à disposition d'instances JupyterHub pour la communauté mathématique et celle du réseau Calcul. Une initiative s'est mise en place sur le campus d'Orsay sur les serveurs du LAL[1] pour un service JupyterHub académique tandis que la PLM proposait une instance JupyterHub monoserveur qui n'avait pas vocation à monter en puissance. En échangeant sur les technologies émergentes, dont le retour d'expérience de l'instance JupyterHub déployée au LAL intégrant KubeSpawner, l'idée de tester cette technologie émergente

1 Laboratoire de l’Accélérateur linéraire qui fait maintenant parti du Laboratoire de physique des deux infinis Irène Joliot-Curie (IJCLab)

apparaît rapidement au sein de la PLMTeam[2]. En cherchant rapidement dans l'état de l'art de l'époque sur comment proposer Kubernetes pour notre communauté, trois options apparaissent : Kubernetes Vanilla, Rancher ou Openshift Origin. La troisième option est retenue avec la version communautaire d'Openshift : OKD. Openshift proposait l'intégration autour de Kubernetes de nombreuses fonctionnalités : une interface web utilisateur assez complète et conviviale, l'intégration et l'authentification dans le système d'information, l'exposition simple des services sur Internet, des règles de sécurité renforcées.

Dès lors, démarre un travail de découverte et d'intégration de nouvelles pratiques (Openshift Origin se déploie à l'époque essentiellement autour de recettes Ansible sur les serveurs préalablement configurés) et même pour des ASR déjà engagés dans le mouvement DevOps[3], il a fallu apprendre les technologies liées aux conteneurs, à Kubernetes et à Ansible. Il a fallu aussi beaucoup d'abnégation et de patience afin de constituer la première plateforme expérimentale présentée lors des Journées Mathrice de Montpellier (Mars 2018)[4].Lors de ces journées, l'initiative d'une plateforme Kubernetes avec la distribution Openshift, appelée PLMshift, est présentée essentiellement dans l'optique de renouveler l'offre d'hébergement de sites web institutionnels pour la communauté mathématique. En effet, ce service alors déployé sur le site du LAREMA (Laboratoire de Mathématiques d'Angers) et constitué essentiellement de scripts « maison » pour construire des virtualhost Apache était vieillissant et posait des problèmes de passage à l'échelle, de suivi de versions des middlewares (PHP, MYSQL, APACHE) et de sécurité. L'idée de pouvoir proposer un service sur étagère (PaaS) pour les chercheurs nous semblait pertinente. Ensuite, la solution a été proposée comme support de TP pour une action de formation en décembre 2018[5] afin de montrer le déploiement d'une pile Big Data basée sur Apache Spark. Enfin, PLMshift a été présenté sous le point de vue ASR en mars 2019 (IHP Paris)[6] puis pour le calcul scientifique et la formation en octobre 2019 (Toulon)[7].

C'est en présentant ce service, puis en découvrant des recettes de déploiement dédiées à Openshift, que l'idée de le mettre à disposition de la communauté devient évidente. En revanche, des freins apparaissent rapidement, que nous allons lister brièvement :

- le processus d'installation demande une préparation importante de la plateforme, les *playbooks* Ansible sont conséquents et prennent en compte un très grand nombre de paramètres ;
- cette instance est déployée sur une plateforme dite "Bare Metal" (machines physiques) et sur des serveurs virtualisés via KVM, qui elle aussi, demande des adaptations importantes ;
- le maintien en conditions opérationnelles (MCO) dans la durée nécessite de nombreux prérequis pour l'équipe PLMTeam, qui a déjà fort à faire avec le choix fait quelques années auparavant d'utiliser Puppet pour gérer les configurations ;
- un prérequis en termes de connaissances de l'écosystème des conteneurs est indispensable à la compréhension de l'utilisation d'une solution PaaS, ce qui n'est pas du tout évident en 2018, encore moins pour les utilisateurs ;
- la question du stockage persistant n'est pas encore tranchée, depuis les volumes pré-réservés sous NFS, jusqu'aux volumes ZFS en passant par GlusterFS, aucun d'eux ne semble pouvoir

2 La PLMTeam est une équipe de collègues informaticiens issus pour la majorité des laboratoires de mathématiques qui œuvrent de façon volontaire, de la conception au support, aux services de la Plateforme en Ligne pour les Mathématiques (la PLM)

3 https://plmteam.pages.math.cnrs.fr/devops/

4 https://indico.math.cnrs.fr/event/2820/contributions/1573/

5 https://indico.math.cnrs.fr/event/3550/timetable/#18-bd1-introduction-aux-techno

6 https://indico.math.cnrs.fr/event/4309/contributions/

7 https://indico.math.cnrs.fr/event/4598/contributions/4034/

passer à l'échelle, les conteneurs pouvant générer beaucoup d'I/O (base de données standard ou Prometheus par exemple). Seul Ceph semble être la bonne solution, mais son intégration demande une montée en compétence non négligeable.

En progressant sur le déploiement de notre plateforme, nous constatons aussi qu'en déployant un cluster Kubernetes complet, nous embarquons un ensemble de briques complémentaires (Cert-Manager pour les certificats, Prometheus pour le monitoring, ElasticSearch/Kibana pour l'agrégation des logs, etc.) et que ces outils nécessitent eux aussi une montée en compétences et également une puissance de traitement intrinsèque. Nous nous retrouvons rapidement dans un décalage entre les collègues en charge d'une administration plus classique des systèmes et celle en charge de l'administration du cluster Kubernetes.

A cette étape de notre projet, nous faisons donc les constats suivants :

- en 2019, le projet de proposer une solution PaaS pour les chercheurs n'est pas si évident. Nos collègues chercheurs et ingénieurs n'ont pas la disponibilité pour une montée en compétences qui paraît élevée ;
- le passage à l'échelle demande encore du travail d'intégration. En l'état, Openshift Origin souffre des performances de GlusterFS.

Ce dernier point commence à être traité fin 2019 avec le test de l'opérateur Rook-Ceph qui intègre l'installation et l'administration d'un cluster Ceph dans une plateforme Kubernetes. Aussi, grâce à la confiance et le soutien financier du Groupe Calcul, la PLM pourra consolider PLMshift en faisant l'acquisition d'hyperviseurs capables de supporter la charge de l'orchestration Kubernetes.[8]

8 le projet PLMshift n'est pas vraiment financé à l'époque, il reste à l'état de preuve de concept qui n'était pas validé par l'équipe. Ce soutien reviendra à une discussion sur la pertinence de la solution et à l'émergence d'un besoin.

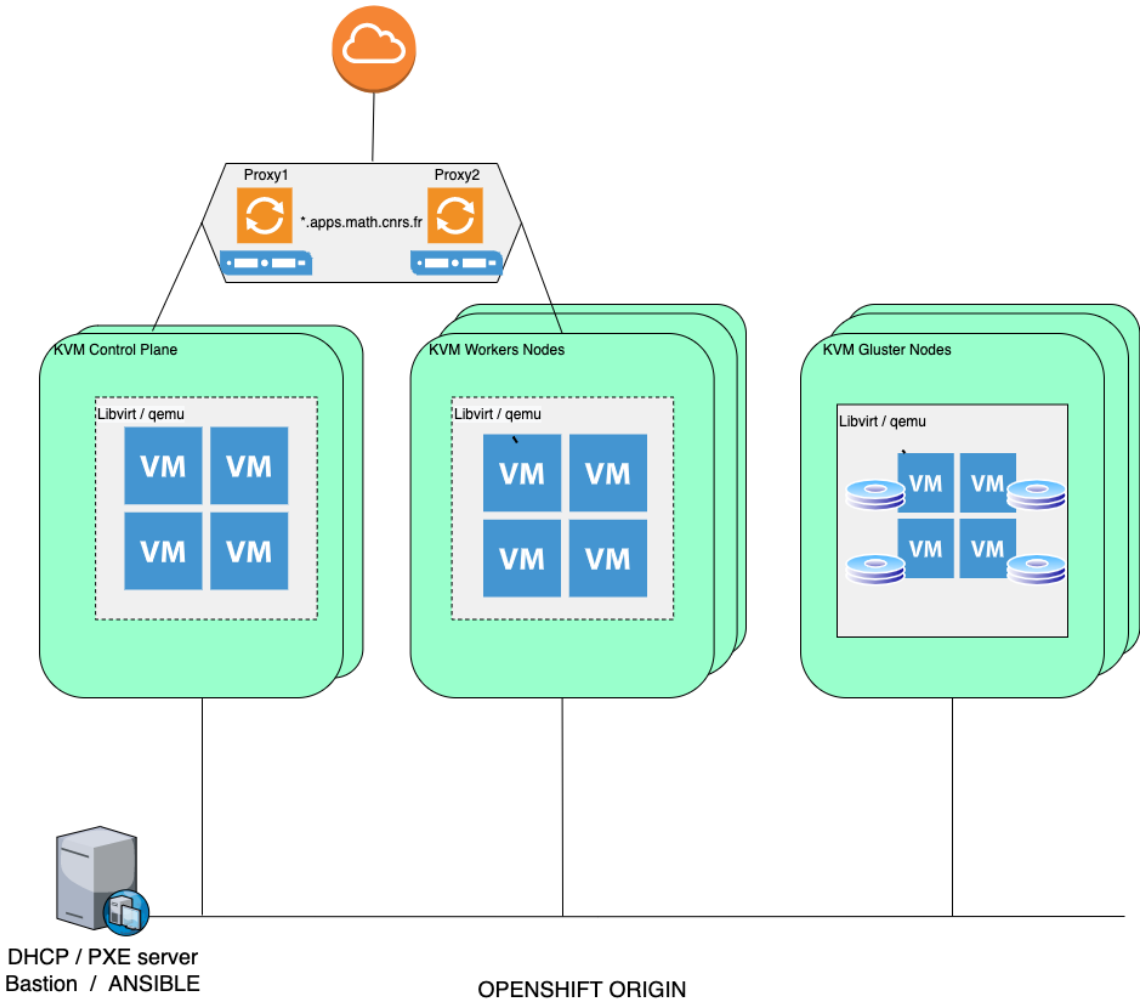

Figure 1 – Notre première instance OKD mise à disposition de nos utilisateurs

2 Accélération

En janvier 2020, afin de lever le verrou du stockage, toujours dans une démarche de prospective, l'équipe valide le remplacement du stockage GlusterFS par Ceph grâce au très prometteur projet Rook-Ceph. Les performances semblent au rendez-vous. Un mois plus tard, le GDS Mathrice organise une grande journée pour les 20 ans du réseau... et patatras, 2 jours avant l'événement, le confinement se met en place. Nous rentrons chez nous, dans nos chambres, nos bouts de bureaux, à la maison. Dans ce contexte, les chercheurs se tournent massivement vers les services de la PLM, les campus ayant du mal à ouvrir leurs services à des communautés élargies. Cela leur semble évident, notre communauté étant déjà bien habituée à utiliser notre offre de services. En revanche, nous ressentons bien la montée en charge de l'utilisation des ressources de notre infrastructure.

Dans un même temps, l'INSMI sollicite la PLMTeam afin de mettre en place en quelques semaines une plateforme scientifique COVID19 [MODCOV19]. Grâce à l'énergie de l'équipe et au travail de montée en compétence, nous arrivons à déployer dans un temps record l'ensemble des briques demandées. Cela sera possible grâce à l'utilisation de l'orchestration Kubernetes qui nous a radicalement simplifié la tâche en reprenant des services conteneurisés à disposition dans les projets Open Source. Très rapidement, des instances dédiées de Codimd, de Limesurvey, un intranet et un

extranet basés sur une sélection d'utilisateurs à travers la Fédération RENATER avec notre solution d’authentification OpenIDConnect sont mis en place. En quelques semaines, l'apport de l'orchestration n'est plus remis en question au sein de l'équipe, le fait que nous ayons anticipé ce projet nous a permis de répondre immédiatement à une demande nationale.[9]

3 Le nirvana ou comment ne pas accepter que nous sommes dépassés

La plateforme en 2020/2021 n'est pas encore optimale en termes d'intégration des ressources, les nœuds de stockage Ceph s'appuient sur des matériels proches de la sortie de garantie. Nous déplaçons le stockage Ceph sur nos nœuds d'infrastructure fraîchement renouvelés et maintenus dans le temps avec un réseau 10G dédié et des disques NVMe. Cette opération sera réalisée efficacement grâce aux fonctionnalités de Rook-Ceph.

La société CoreOS (rachetée en 2018 par RedHat) a apporté deux technologies essentielles pour OpenShift : la distribution Linux CoreOS et les opérateurs. L'opérateur est un couple Contrôleur/Custom Resource Definition (CRD) qui permet d'enrichir simplement les API standards de Kubernetes. Ce principe offre la capacité de pouvoir déclarer à travers les CRD des classes d'objets Kubernetes custom qui seront traitées par le contrôleur. Ce contrôleur traduira ensuite ces CRD en ressources de base de Kubernetes (Deployment/StatefullSet/DaemonSet, Pods, Services, Ingress, PVC). Le gros avantage est de pouvoir concentrer une configuration complexe d'un service sous forme d'une ressource minimale et de la déployer de façon fiable sur le cluster. Ensuite le contrôleur a la capacité de gérer les mises à jour, le MCO du service voire le PRA. Les contrôleurs peuvent se baser sur différents SDK, par exemple la pile Ansible, pour gérer ces fonctionnalités.

9 Cette avance de phase et cette réactivité particulière ont été reconnues par l'attribution d'un cristal collectif. Au CNRS, le cristal collectif distingue des équipes de femmes et d’hommes, personnels d’appui à la recherche, ayant mené des projets dont la maîtrise technique, la dimension collective, les applications, l’innovation et le rayonnement sont particulièrement remarquables.

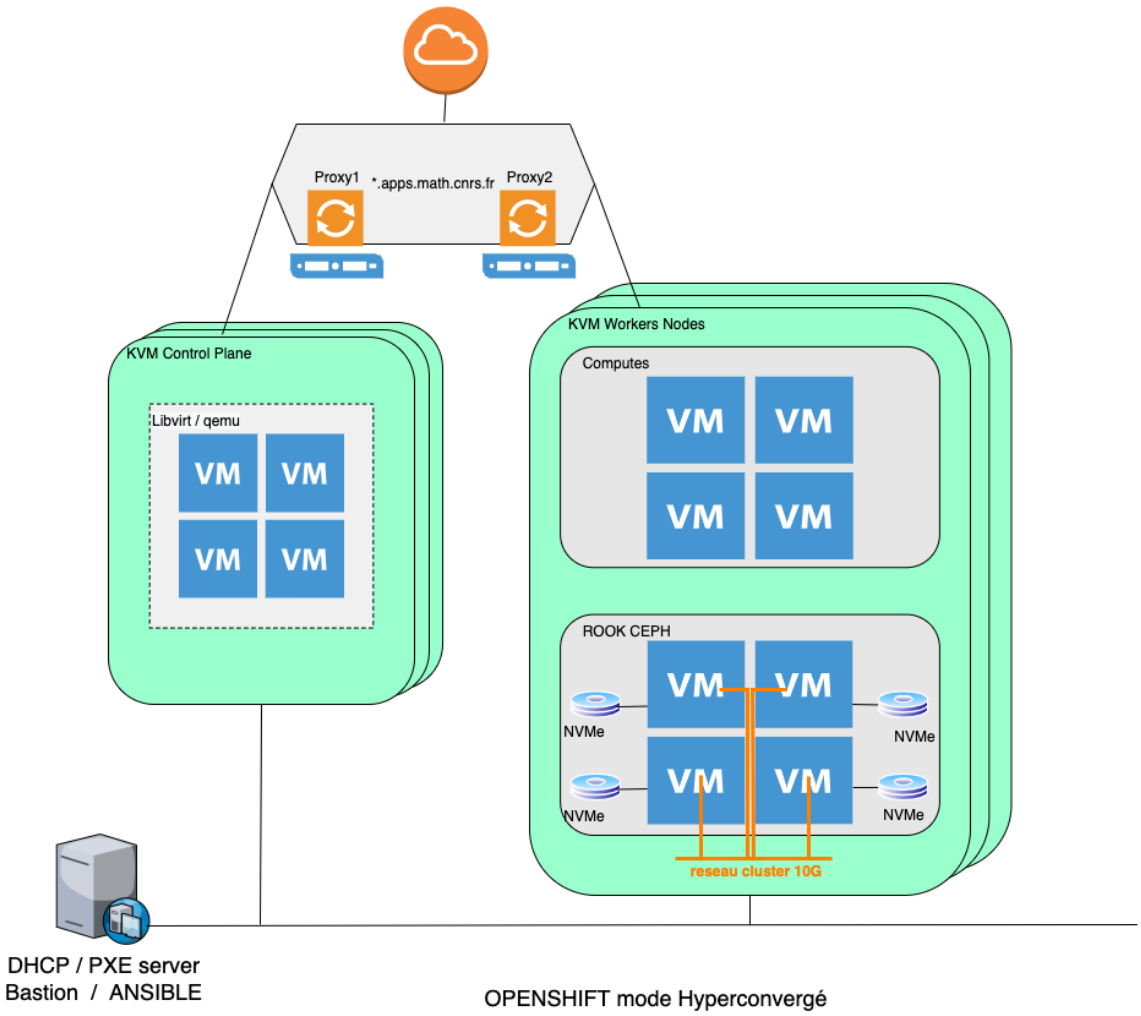

Figure 2 –Version hyperconvergée du cluster avec stockage NVMe dédié

À partir de là, tout un ensemble de projets se raccroche à notre cluster qui accueille aussi bien de l'hébergement Web que des applications scientifiques (Notebook, ShinyR, JupyterHub à la demande ou national, Plmlatex, etc.). Nous commençons ainsi à mettre en œuvre l'architecture cible qui nous tenait tant à cœur : proposer une solution unifiée, multi-tenant pour des solutions PaaS ou SaaS à notre communauté. En parallèle, nous observons l'émergence de solutions Kubernetes de site déployées par les DSI de campus, hors du périmètre des utilisateurs. Par ailleurs les installations sauvages de cluster Kubernetes plus ou moins maîtrisées par des utilisateurs sur des IaaS nous semblaient peu productives, car il faut savoir que l'empreinte d'un cluster Kubernetes n'est pas négligeable quand on le souhaite fonctionnel (routage, gestion des certificats, supervision, monitoring, sauvegarde, etc.).[10] Fin 2020, toujours grâce au soutien du Groupe Calcul et de l'INSMI, nous avons pu, en monopolisant des ressources non utilisées pour cause de Covid19, acquérir des nœuds fortement dimensionnés avec des cartes GPUs performantes. Il devenait alors possible d'accueillir des applications scientifiques à travers une approche "service à la demande" (SaaS).

L'année 2021 a démarré en trombe pour l'équipe PLMTeam, avec la volonté de capitaliser sur cette

10 Un cluster Kubernetes nu représente une trentaine de processus dont moins d'une dizaine par nœud utilisateur (worker node), si on ajoute le nécessaire pour un bon fonctionnement, on peut monter facilement à une centaine de processus. Pour un cluster complet, il faut compter au moins 300 processus (ou 300 Pods sur Openshift, car l'intégralité du fonctionnement du cluster est orchestrée). On passe alors de quelques vCPU et quelques dizaines de Go de RAM à au moins de 48 vCPUs et 128 Go de RAM pour un cluster complet, sans les workers nodes.

offre de service. L'ouverture aux chercheurs et ingénieurs expérimentateurs demandait un accompagnement conséquent, que nous avons surtout mené à travers le canal PLM-Support de notre service Mattermost. Ce qui a été un changement radical dans notre organisation, car jusqu'à présent, nous proposions exclusivement un service de tickets à travers RT. Les utilisateurs pouvaient nous solliciter directement et interagir, ce qui a été très pertinent pour encadrer cette offre de service représentant une rupture technologique. Ces échanges nous ont permis aussi de comprendre que concevoir un catalogue de service est une chose, le rendre accessible, fonctionnel et à jour, accompagné d'un support de qualité, en est une autre. C'est donc une année qui a vu la charge de travail croître de façon importante avec un accompagnement conséquent des utilisateurs sur une technologie nouvelle et complexe.

Voici quelques chiffres qui permettent de se représenter la charge de l'accompagnement des utilisateurs. Actuellement, les services de la PLM comptent 7500 utilisateurs actifs dont 3400 personnes invitées (aux interactions avec les mathématiques), notre PaaS compte près de 860 utilisateurs pour plus de 500 projets et 800 Urls exposées. Les services nationaux tels que PLMlatex (authentifiés via la fédération RENATER) sont utilisés par près de 21 400 personnes. Nous sommes une dizaine d'informaticiens de laboratoire à suivre, en parallèle de nos activités quotidiennes, les demandes via notre système de tickets et, depuis 2020, via le Chat de notre instance Mattermost.

4 La panne, ou la première expérience malheureuse de la magie des opérateurs

À L'été 2021, la configuration de Ceph était désormais parfaitement opérationnelle et nous étions en mesure de pouvoir passer à autre chose... sauf qu'un petit paramètre n'avait pas été appliqué correctement. En effet, lorsqu'un cluster Kubernetes héberge des services nécessitant des ressources particulières dédiées, il est indispensable de réaliser trois opérations :

- marquer les nœuds (taint) afin que les applications standard ne puissent pas s'executer dessus ;
- mettre des attributs nodeSelector sur les services devant les utiliser ;
- mettre des tolérances sur ces mêmes services pour qu'ils acceptent la marque (taint).

Nous n'avions pas fait correctement le marquage (taint) par manque de rigueur… à la fin de l'été 2021, les nœuds très capacitifs (GPU) sont tombés en panne et, de façon massive, les applications se sont déversées sur les nœuds d'infrastructure et de stockage Ceph, et bien sûr, en période estivale. Nous perdrons 1 PG Ceph qui malheureusement possédait les métadonnées de notre espace CephFS. La moitié des applications l'utilisait et certaines étaient mal sauvegardées. Là nous constatons notre manque de maturité en termes de gestion sous pression d'un cluster Ceph et nous mettrons 30 jours pour restituer l'intégralité du service. Il nous a fallu deux semaines pour reconstruire l'arborescence du système de fichiers CephFS en analysant les journaux de transactions (via cepfs-journal-tool) via un script Python. Ensuite pour quelques applications nous avons opéré des actions de récupération avec les utilisateurs, application par application.

Grâce au groupe de discussion Ceph, nous prenons contact avec un prestataire qui sauvera notre stockage : nous n'aurons perdu que très peu de données sensibles. En revanche nous en retiendrons les leçons suivantes :

- la configuration des nœuds et des services sur un cluster Kubernetes hétérogène est très importante. Il est indispensable de bien réserver les nœuds aux différents usages ;
- une bonne gestion des configurations devient incontournable pour correctement suivre l'état de notre cluster. Comme Kubernetes est basé sur un principe de configurations déclaratives,

il nous est alors apparu évident de nous appuyer sur les outils et les bonnes pratiques en découlant, comme les pratiques GitOps qui permettent de garantir l'évolution des configurations des clusters Kubernetes ;

- l'hétérogénéité des configurations matérielles génère des déséquilibres de charge qu'il faut pouvoir anticiper (il faut avoir à disposition des nœuds réservoirs capables de récupérer la charge d'une panne de nœud) ;
- la supervision prend une importance particulière, avec la remontée d'alertes préventives. En effet, l'efficacité et la complexité inhérente à Kubernetes peuvent cacher des dysfonctionnements. Dans notre cas cela a été des sauvegardes incomplètes, des services partiellement opérationnels ;
- les opérateurs retirent une charge de développements considérable dans les configurations des services, mais n'excluent pas la montée en compétence sur ces services. Par exemple, la panne de notre stockage Ceph est due à une mauvaise configuration et un manque de maîtrise de la technologie. Pour faire simple, nous avions fait le choix d'un système de fichiers en Erasure Coding avec une configuration inadéquate et trop légère, à la première panne conséquente, une partie des données est devenue inaccessible ou a carrément disparu.

Ce dernier point nous semble important à mentionner, Kubernetes permet de déployer très rapidement des services en cachant la complexité, et il permet d'en déployer un grand nombre. Ainsi sur une même plateforme plusieurs compétences peuvent être mobilisées et les administrateurs n'ont pas nécessairement l’expérience suffisante pour prendre en charge cette multi-complexité. Kubernetes avec la puissance des opérateurs n’exclut pas l'indispensable compétence des personnes en charge de son administration.

5 L'âge de la raison et la découverte des spécificités des IaaS

En plus de cet incident de configuration qui a manqué nous faire perdre nos données persistantes, un autre point est apparu : OKD, la version OpenSource de Openshift, qui nous semblait une excellente alternative à la version commerciale, était basée sur le système d'exploitation Fedora CoreOS (FCOS). Or, ce dernier est un projet parallèle et autonome par rapport à OKD et surtout dépendant de Fedora. Il est arrivé plusieurs fois lors d'une mise à jour d'OKD que des modifications substantielles de FCOS perturbent le bon fonctionnement du cluster comme une régression induite par le changement de comportement par défaut du module noyau Ceph[11] de la Fedora CoreOS. Suite à ces différents incidents et au souhait de notre institut du CNRS d'avoir des services sur une instance Kubernetes plus fiable, nous avons décidé d’utiliser une licence OpenShift sur une infrastructure IaaS. Cela correspondait aussi à notre souhait d'abandonner à terme nos infrastructures Bare Metal pour les services de la PLM vers un hébergement sur des IaaS institutionnelles ([1]). Nous avons donc réservé notre instance PLMshift basée sur OKD et des nœuds bare metal pour l'expérimentation scientifique et opté, grâce aux tarifs RedHat dans le groupe logiciel, pour une instance Openshift sur Openstack pour nos applications institutionnelles. En optant pour des IaaS Openstack, nous nous libérons aussi d’une importante charge de gestion de l’infrastructure matérielle et du stockage.

Comme présenté lors des JRES 2022 ([2]), le déploiement d'Openshift/OKD sur IaaS est très bien outillé, l'installeur étant un programme en GO utilisant l'API Openstack à travers les bibliothèques Terraform. Il suffit d'un fichier yaml d'une vingtaine de lignes pour déployer un cluster Openshift sur un IaaS depuis son poste de travail. Nous déployons dans la foulée une instance ArgoCD qui

11 Le montage des volumes Ceph devient asynchrone par défaut alors qu'il était synchrone jusque-là, ce qui a mis en évidence une régression lors de la pagination des résolutions des attributs étendus SElinux qui se manifestait par un affichage partiel des attributs selinux au-delà d'un certain nombre de fichiers dans un dossier, aspect qui nous a échappé lors la mise à jour.

consomme un dépôt GÎT. Notre cluster est ainsi configuré et maintenu dans une démarche GitOps et nous maîtrisons le cycle de vie de façon bien plus efficace que précédemment.

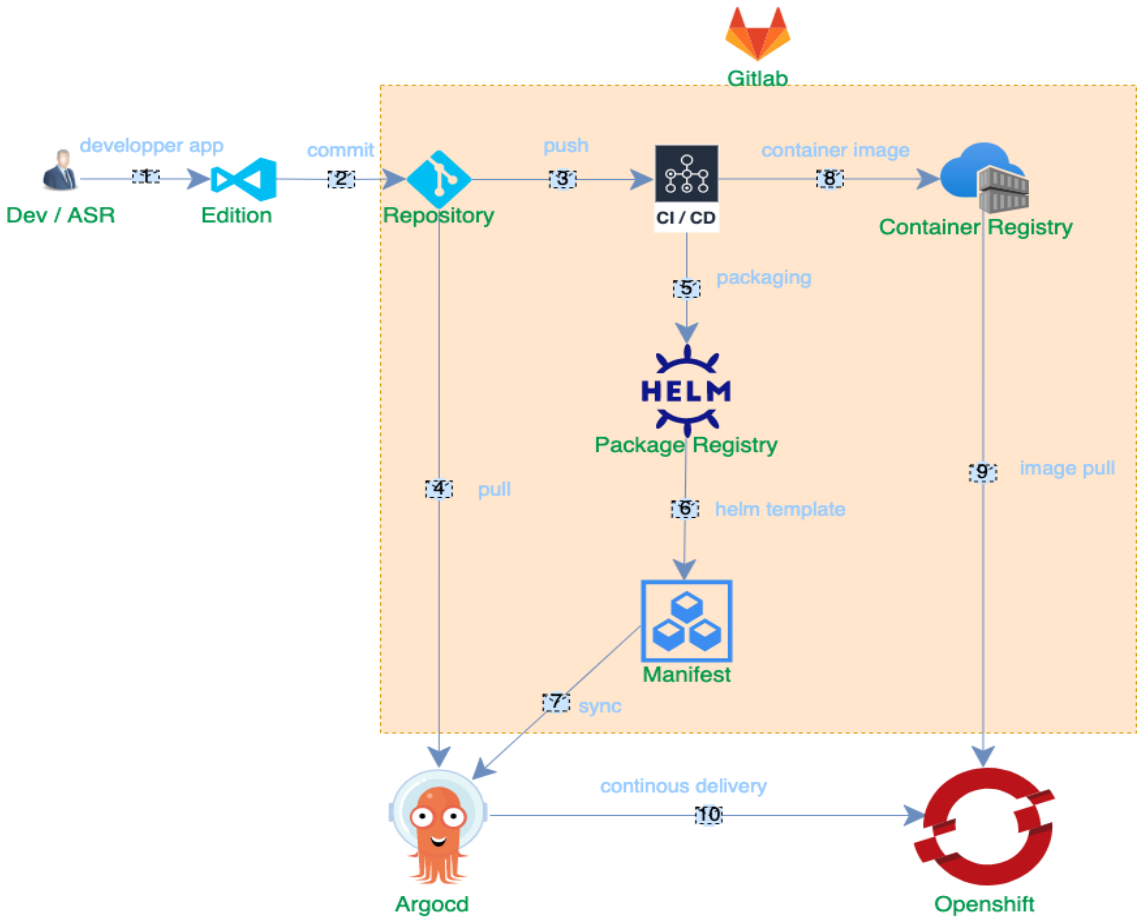

Figure 3 – Organisation opérationnelle de l'administration de nos clusters et nos applications

De la même façon, nous mettons en place ce mode opératoire pour notre cluster Bare Metal, et commençons à déplacer les applications institutionnelles sur le cluster hébergé en IaaS. À cette étape, nous pensons enfin être en mesure de proposer une solution stable et fiable à notre communauté tout en proposant par ailleurs une plateforme expérimentale. Cette migration s'est surtout appuyée sur les sauvegardes réalisées à travers l'opérateur Velero. À partir du moment où nous installons l'opérateur Velero sur l'instance cible avec comme backend storage celui de l'instance source, l'ensemble des backups apparaissent et il est alors possible de créer des ressources de type Restore pour restaurer et en conséquence migrer les applications sur le nouveau cluster. Lors du déploiement sur un IaaS, nous avons eu l'opportunité de nous appuyer sur deux instances Openstack opérées chacune sur une infrastructure différente, donc avec des différences substantielles. Nous ne pensions pas que ces différences allaient impacter le comportement du cluster Openshift.

Pourtant, voici les différents points auxquels nous avons dû faire face.

Nos hébergeurs s'adressaient plutôt à des besoins HPC et nous proposaient des gabarits inadaptés à l'optimisation des coûts de licence Openshift. En effet la licence étant basée sur le nombre de vCPU des nœuds compute, un ratio plus important de RAM par CPU est plus efficient.

Il peut y avoir un grand nombre de requêtes sur les DNS de site, passant par le cache DNS de la gateway de projet Openstack. Il est pertinent de dérouter ces requêtes vers un cache dans le projet, car la résolution DNS doit être fiable.

Même si les I/O semblent performants, les prérequis pour le cluster ETCD sont très exigeants. Le stockage sous-jacent doit être efficient, car cela impacte directement l'orchestration des conteneurs. Le cluster ETCD est une base de données contenant l'ensemble de la configuration du cluster à un instant donné. L'API s'appuie sans cesse sur cette configuration afin de maintenir l'état du cluster dans l'état souhaité. C'est l'information essentielle à préserver dans le cluster en plus des données des projets. Les recommandations de Redhat préconisent un stockage direct (non distribué à contratrio de Ceph) ou bien avec un stockage distribué sur supports flash (SSD/NVME) rapides. Dans le cas contraire, des timeout peuvent surgir régulièrement, provoquant l'élection d'un nouveau leader du cluster ETCD, ce qui impactera les applications installées. Nous avons constaté ce souci particulièrement sur les clusters Percona, MongoDB et PostgreSQL qui eux aussi changent de leader, car l'API Kubernetes n'est plus suffisamment réactive. Depuis la version Openshift 4.16, RedHat a soulagé cette contrainte en adaptant le comportement du cluster ETCD selon les performances du système de fichier[12]. Dans notre cas, nous étions à la limite basse des performances attendues, donc ce problème se manifestait par intermittence.

Nous avons opté pour utiliser des IaaS dans des Clouds Recherche, en ayant bien conscience du caractère expérimental de ces infrastructures. Les choix de provisionning des hyperviseurs Openstack (over-provisionning ou non) peuvent avoir un impact non négligeable sur le comportement des applications du cluster. Nous le constatons rapidement, avec des lenteurs de notre cluster lorsque les hyperviseurs sont chargés par des VMs tiers ayant une charge CPU importante. Les I/O ne sont pas suffisants pour que le cluster ETCD réponde suffisamment vite et là aussi l'API Kubernetes est impactée.

L'étendue des usages sur notre cluster Openshift a permis de mettre en évidence les partis pris de ces IaaS qui ont été dimensionnés pour des utilisations et des I/O gros grains. Afin de pallier à cela, les sondes de supervision des conteneurs (ReadinessProbe/LivenessProbe) pour des bases de données sont à ne pas négliger et à ajuster, les délais peuvent être insuffisants et le conteneur redémarre avant que la base de données soit correctement initialisée, ce qui la rend inconsistante et empêche le démarrage du service.

De manière générale, les excellentes relations avec les équipes en charge des IaaS sur les deux sites utilisés[13] ont permis des avancées de part et d'autre. En effet, en déployant nos solutions, nous pouvons soit exprimer un nouveau besoin, soit mettre en évidence la nécessité d'ajustements sur des IaaS qui n'accueillent pas de services avec des prérequis d'applications généralistes. Cela permet aux équipes d'infrastructures d'affiner leurs configurations, par exemple envisager d'ajouter du stockage Flash sur les nœuds Ceph afin de garantir un certain niveau d'I/O.

6 La situation actuelle, les projets, ce qu'on veut faire et jusqu'où

Forts de cette expérience, après deux ans de compréhension du fonctionnement d'Openshift sur des IaaS et sur l'accès aux opérateurs avancés de RedHat dont Openshift Data Foundation et Noobaa, nous mettons en place l'architecture [voir Figure 4] afin de proposer les services suivants :

- un hébergement des sites et applications institutionnels avec une résilience satisfaisante y compris l'authentification, l'annuaire de la communauté, le support, le monitoring et la supervision;
- un service de stockage S3 à la demande multi-site répliqué et chiffré pour les unités de l'INSMI ;

12 https://docs.openshift.com/container-platform/4.16/scalability_and_performance/recommended-performance-scale-practices/recommended-etcd-practices.html#etcd-changing-hardware-speed-tolerance_recommended-etcd-practices

13 Nous remercions vivement les structures Virtualdata et Gricad (avec le soutien de FranceGrille) d'avoir bien voulu participer à cette aventure et intégrer dans leurs IaaS Openstack dédiées à la recherche des clusters Kubernetes générant ces nouveaux usages.

– une solution PaaS pour l'expérimentation et la science.

Pour cela nous déployons les clusters suivants :

– un cluster Openshift principal dédié aux applications institutionnelles ;

– un cluster Openshift de secours minimal avec le service de stockage S3 à la demande, disponible pour un PRA du principal ;

– un cluster OKD pour les utilisateurs et l'expérimentation.

Les clusters K8s sont déployés avec une supervision limitée sur 15 jours. Augmenter cette durée nécessiterait de monopoliser des ressources trop importantes sur les clusters au détriment du coût et des performances. Nous souhaitons avoir une mémoire sur un an et une supervision multi-site pour des remontées d'alertes fiables. Pour cela nous avons fait le choix d'utiliser une solution externe à base de Victoria-Metrics et Grafana en soutien des Prometheus locaux aux clusters Openshift.

Nous faisons le constat que le projet initial de faire converger les usages sur un même cluster afin d'optimiser les charges d'exploitation n'a pas pu être atteint. Ceci est dû d'une part au choix de partir sur une solution logicielle avec des garanties de l'éditeur, mais qui aurait largement dépassé notre budget si on y avait inséré les serveurs dédiés à l'expérimentation. D'autre part, cela nécessiterait de monopoliser trop de ressources en un seul point avec des soucis de performance, d'efficacité, de disponibilité et de maintenance. En revanche, le fait d'uniformiser les configurations via des dépôts Git très proches opérés par ArgoCD rend l'administration très confortable. Il est alors possible de valider une montée en version d'un côté et la propager ensuite. De manière générale, hormis la partie FCOS, les montées en version d'OKD n'ont pas posé de problème et le passage de FCOS à SCOS (Stream CoreOS) pour les prochaines versions d'OKD ne font que nous conforter dans l'utilisation de OKD pour un large périmètre de services.

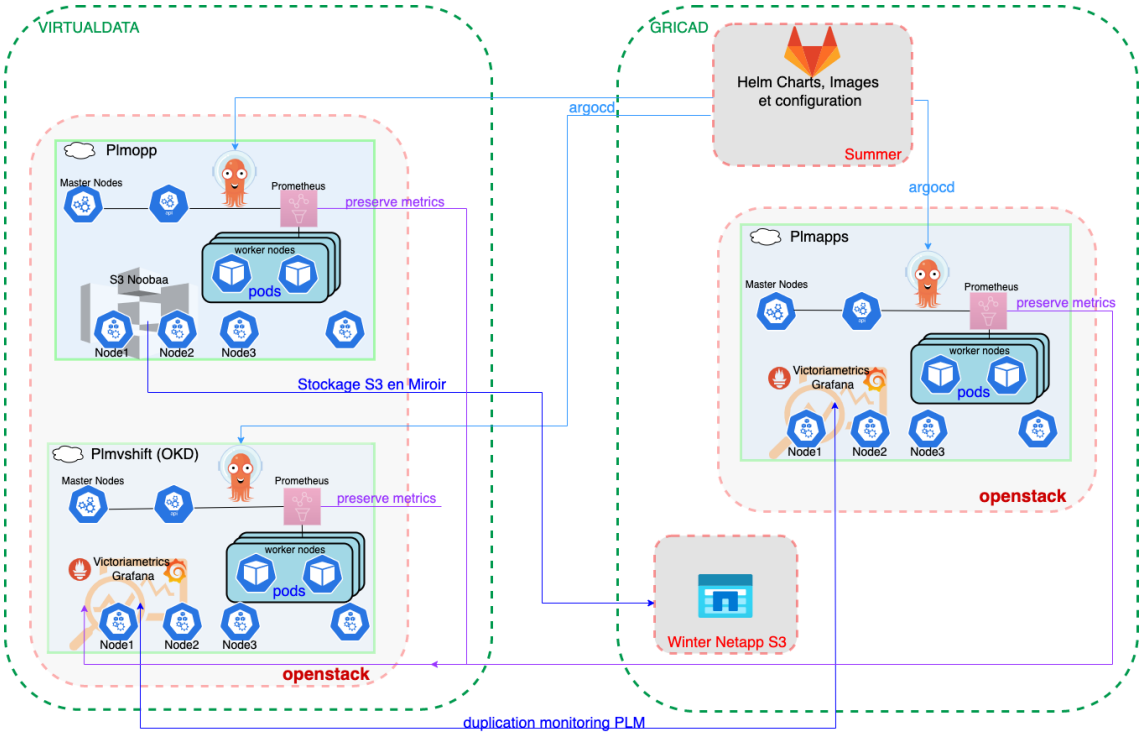

Figure 4 – L'architecture actuelle des cluster OKD/Openshift de la PLM

Un autre point nous confortant dans nos choix est le constat qu'il est possible de proposer à des utilisateurs avancés (des collègues ASR ou développeurs par exemple) l'accès à une plateforme de type OKD afin de simplifier le déploiement d'applications pour les laboratoires. En effet, si les DSI proposent de plus en plus de IaaS à destination des unités pour y déployer des services applicatifs, il devient beaucoup plus efficace de proposer une solution PaaS intégrant tous les services d'infrastructure tel que l'ensemble des outils d'observabilité (monitoring, logs, traces, alertes) pour les utilisateurs ainsi qu'un backup et un PRA avec une collection d'opérateurs et de Charts Helm. Le stockage doit être duplicable à part (via un espace S3 par exemple). En mutualisant cet environnement pour des services internes, la charge d'exploitation est plus efficiente. Nous pensons qu'une bonne partie des services d'unités pourraient être mis en œuvre sous cette forme. Comme nous tentons de le montrer dans nos formations (cf ANF 2024[14]), une telle solution mutualisée (au niveau campus par exemple) permettrait aux ASR de se dédier aux services internes, d'expérimentation ou l'environnement du poste de travail. Ces charges ne pouvant pas passer sur une solution PaaS.

• Bibliographie

[1] Azema L., Delavennat D., Depouilly P., Ferney D., Théron T.. Retour d'expérience sur l'évolution d'une infrastructure à l'ancienne vers des clouds institutionnels, 2022 ; https://conf-ng.jres.org/2021/document_revision_1961.html?download .

[2] Depouilly P., Ferney D.. Déploiement d'une solution complète de type PaaS dans un environnement de Cloud ou de laboratoire, destinée à une large communauté, 2022 ; https://2021.jres.org/programme/

14 « Migration d'applications vers les technologies du cloud » - https://indico.math.cnrs.fr/event/11850/