UNCLOUD, de 0 à 10 000 utilisateurs en 1 an

Arnaud Abélard
DSIN de l'université de Nantes
2 rue de la Houssinière
44322 Nantes cedex 1

Matthieu Le Corre
SCI Sciences – Faculté des sciences
Université de Nantes
2 rue de la Houssinière
44322 Nantes cedex 1

Mots-clefs

CLOUD, NEXTCLOUD, ONLYOFFICE, CEPH RADOS GATEWAY , COLLABORATIF

1 Introduction

En 2017, l'université de Nantes a lancé UNCloud un projet de services numériques devant faciliter les échanges entre les populations universitaires mais également avec les collaborateurs extérieurs. Plate-forme de stockage et outil collaboratif, proposant 100Go à ses 70 000 personnels et étudiants, UNCloud est devenu, en quelques mois, une pierre angulaire de l'établissement. Bien que basé sur l'outil Nextcloud, celui-ci, bien connu et largement déployé dans la communauté, ne sera pas l'objet principal de cette présentation. Celle-ci se focalisera sur l'ampleur du projet et les spécificités politiques et techniques ayant mené à son succès et son adoption par 10 000 utilisateurs dès la première année.

2 Genèse du projet

En 2015 la Direction des Systèmes d'Information et du Numérique (DSIN) a lancé son premier schéma directeur du numérique. L'un de ses enjeux est de proposer de nouvelles pistes d'interactions entres les acteurs aussi bien internes qu'externes. L'explosion des problématiques autour de la donnée et de son usage à l'université de Nantes, a entraîné une réflexion sur la mise à disposition de ces dernières. Pour répondre à ces problématique, les projets suivants ont été définis : "*Disposer d'outils de communication / collaboratifs à l'état de l'art pour les étudiants, les enseignants et les agents*", "*Mettre en place un espace unique et sécurisé pour les étudiants*", "*Développer les infrastructures numériques pour renforcer le travail collaboratif au niveau du territoire*" et "*Permettre l'accès aux services et ressources numériques de l'Université de Nantes en mobilité*". Ceux-ci visaient à adresser une série de problématiques grandissantes au sein de l'établissement :

Centraliser le stockage de données afin de faciliter les échanges : jusqu'en 2018, l'université ne proposait que du stockage traditionnel, majoritairement axé sur des serveurs de fichiers de type Samba. Chaque composante possède son propre serveur de fichiers, celui-ci permettant à ses utilisateurs de partager des documents entre eux. Les serveurs de fichiers sont principalement destinés à l'hébergement des données institutionnelles, les documents personnels de travail restant bien souvent sur les postes utilisateurs. On constatait alors une fuite des données vers des services de partage grand public tels que Dropbox ou Google Drive.

Simplifier l'organisation et le cycle de vie des données d'établissement : les serveurs de fichiers sont traditionnellement désorganisés. Avec le temps et la mobilité des personnels, la multiplication des partages complexifie l'usage des données partagées. En cas de changement d'affection d'une personne,

l'absence de centralisation implique une perte d'accès à ses données personnelles. Celles-ci deviennent ainsi orphelines sur le serveur.

Faciliter la mobilité du personnel et des étudiants : la nécessité d'utiliser le VPN de l'établissement pour accéder aux serveurs de fichiers est un frein important, notamment dans le cas des terminaux mobiles.

Simplifier les échanges et la collaboration entre les populations : le partage de données entre les étudiants et le corps enseignant se limitait surtout au courrier électronique, à la plate-forme d'enseignement numérique basée sur Moodle et à l'usage de comptes personnels des étudiants sur les services grand publics. Ceux-ci ont par ailleurs popularisé la rédaction collaborative de documents.

Garantir la confidentialité de l'information et sa conformité légale : le recours de plus en plus fréquent aux services d'hébergement de sociétés américaines expose les données d'établissement à la législation internationale. Parmi les textes de loi notables qui s'appliquent aux données externalisées sur ce type de service nous pouvons citer :

- Le **USA Patriot Act**, datant de 2001, donne des pouvoir élargis aux agences américaines. Il leur permet d'accéder aux données utilisateurs des services grand publics américains.
- **La directive conjointe des ministères de la Culture et de l'Intérieur d'avril 2016** à destination des administrations territoriales françaises restreint l'exportation de tout document numérique, en leur donnant le statut de potentiels "trésors nationaux".
- **Le Cloud Act** (Clarifying Lawful Overseas Use of Data Act) est un texte de loi américain datant de 2018 sur la surveillance des données personnelles. Cette loi donne accès aux agences américaines à toute donnée personnelle gérée par une société dont le siège est aux USA et ce quel que soit l'endroit ou est hébergée l'information.
- Enfin, **le RGPD** (Règlement Général sur la Protection des Données), un texte européen obligeant les sociétés de services numériques à communiquer à l'usager sur comment celles-ci vont exploiter ses données et qui peut y avoir accès. Le RGPD impose un droit à l'oubli.

3 Projet

UNCloud est, à l'origine, un projet principalement axé sur le partage entre les populations universitaires et l'extérieur de l'université. Toutefois lors de la phase de test, nous nous sommes rendu compte que les utilisateurs avaient de fortes attentes en matière de travail collaboratif. Il fut alors décidé d'élargir le périmètre du projet et d'y adjoindre des fonctionnalités supplémentaires tels l'édition collaborative en ligne de documents, la gestion de projet, le support des discussions instantanées et par la suite la visioconférence.

3.1 Choix

3.1.1 Choix de la plate-forme de partage de documents

Au vu des problématiques exposées ci-dessus, nous avons donc étudié les solutions suivantes :

- **Google Suite :** Google Suite est un service très populaire parmi la population universitaire. Sa messagerie Gmail est la référence dans le domaine, elle attire notamment un grand nombre de chercheurs. Après avoir contacté le service commercial de Google celui-ci nous a opposé la directive d'avril 2016. Ainsi ils nous ont indiqué ne plus collaborer avec les administrations françaises, ne pouvant pas garantir l'hébergement des données en Europe. Leur respect du RGPD fait également débat [1] .
- **Microsoft Office 365** : le service grand public de Microsoft est l'autre service le plus populaire parmi les populations universitaires, tout particulièrement chez les étudiants. Au

moment où nous avons fait nos choix, Microsoft ne proposait pas encore d'hébergement garanti en Europe. De même que Google, leur respect du RGPD est encore sujet à débat [2].

- **Renater Partage** : étant conforme à la législation en vigueur puisque qu'hébergé en France, Partage a été très sérieusement envisagé. Toutefois son offre de partage de fichiers était alors limitée au porte document de Zimbra. La roadmap concernant le partage de fichiers et l'édition collaborative était encore trop floue. De plus l'ergonomie proposée par la solution Zimbra ne correspondait pas au niveau des standards actuels.

- **Seafile** : Seafile est une solution opensource de partage de fichiers, qui a la particularité de ne pas être dépendante d'un serveur web et directement codée en C. Elle se distingue par ses temps de réponse sans égal. Malheureusement, son écosystème était, à l'époque, en retrait comparé aux autres projets étudiés.

- **Pydio** : Pydio est un projet opensource écrit en php proposant une interface utilisateur originale. Malheureusement cette originalité est aussi son point faible, trop éloignée des standards existants, sa prise en main est la plus complexe des produits testés. Son écosystème bien qu'offrant une solution d'édition collaborative était relativement réduit.

- **Nextcloud/Owncloud :** Nextcloud est un autre projet opensource de partage de fichiers utilisant php. Il est né d'un fork de Owncloud. Ce dernier a été exclu rapidement, son modèle premium/community ne le favorisant pas comparé à Nextcloud qui nous permettait d'exploiter l'intégralité des fonctionnalités sans surcoût, tout particulièrement l'accès au stockage objet. L'interface de Nextcloud, de par son rapprochement avec celles de Google Document, a tout de suite été appréciée. Son architecture pensée pour la haute disponibilité est tout à fait adaptée à nos besoins.

3.1.2 Choix de la suite d'édition collaborative

Il existe actuellement deux grandes suites d'édition collaborative auto-hébergeables :

- **OnlyOffice** : OnlyOffice de la société Ascencio Systems a été la première suite Office opensource en ligne utilisable en production. Elle utilise comme format natif le standard OpenXML (docx, xlsx, pptx) compatible avec la suite Office de Microsoft, solution la plus utilisée à l'université. OnlyOffice a fait le choix d'une génération d'affichage des documents coté client, allégeant ainsi l'infrastructure serveur nécessaire. OnlyOffice annonce officiellement 1000 éditions simultanées par serveur. Son interface est extensible par un système de plugins. Certains sont proposés en standard tel un éditeur d'images, un assistant d'import de vidéos, etc.

- **Collabora Office** : Collabora Office est une solution opensource qui repose sur LibreOffice. Son format de document est donc le standard OpenDocument (odt, ods, odp). Elle génère les documents coté serveur et le client n'est utilisé que pour l'affichage du rendu final. Les serveurs s'en trouvent fortement sollicités et nécessitent plus de ressources. Son interface utilisateur a également été jugée un peu moins accueillante que celle d'OnlyOffice.

3.2 Phase de test et appel aux volontaires

L'ampleur du projet et sa nature structurante pour l'établissement nous ont amené à mettre en place une phase de test à l'échelle de l'université. Le principe retenu pour cette phase a été de passer par un appel à volontaires et de retenir un premier panel de 100 utilisateurs reflétant les différentes populations universitaires. Nous avons ensuite offert à chaque participant la possibilité de parrainer 4 nouveaux utilisateurs. Ce parrainage devant favoriser l'utilisation du système de partage. Le succès de ce test a été immédiat démontrant ainsi le besoin de tels outils au sein de l'établissement.

- **Déroulement** : cette phase a débuté par une réunion d'information afin d'impliquer les gens, de leur expliquer les objectifs et les enjeux du projet ainsi que les fonctionnalités de base de

la plate-forme. Une animation régulière du groupe de testeurs avait lieu sous forme d'une lettre d'information hebdomadaire et d'un forum permettant d'orienter l'attention vers les fonctionnalités que nous jugions nécessaire de tester. Nous avons été très réactifs aux bugs durant cette période.

- **Durée** : initialement prévue pour 3 mois, la plate-forme de test a été maintenue pendant 1 an jusqu'à la mise en service de la version de production.

- **Accueil** : en dehors du succès de l'outil en lui même il faut souligner l'enthousiasme des gens vis à vis du projet. De manière générale le fait d'avoir été consulté et impliqué dans la mise en place a été très apprécié. C'était la première fois qu'une telle opération de test était mise en route a l'échelle de l'université.

- **Retour de la phase de test :** très rapidement les testeurs se sont approprié l'outil qui leur été proposé et en ont dégagé de nouveaux usages qui n'avaient pas été forcément imaginés à l'origine, tels que, par exemple, le dépôt de devoirs ou la gestion de projet. Par ailleurs, l'édition collaborative a été très fortement sollicitée et globalement, le panel de test considéraient la plate-forme de partage de documents comme un portail voire comme un Environnement Numérique de Travail (ENT).

D'un point de vue technique, plusieurs choses ont émergé de la phase de test, la première concerne le stockage en mode fichiers : celui est un point faible et induit une section critique dans l'architecture. Difficilement redondable et scalable, il a été remplacé par du stockage objet dans la version finale du projet. La base de donnée s'est avérée être finalement le point le plus critique de l'application et le plus demandeur en ressources.

De la phase de test ressort aussi l'importance toute particulière que revêt l'organisation de l'annuaire LDAP. L'université de Nantes possède plusieurs annuaires LDAP présentant l'information de manières différentes. Il avait été décidé d'utiliser l'annuaire dont l'organisation reflétait l'organigramme administratif de l'université car celui-ci contenait aussi les groupes d'utilisateurs. Il s'est avéré que, dans Nextcloud, le "Distinguished Name (DN)" qui reflète donc son affectation administrative à l'université, sert par défaut à identifier les comptes utilisateurs. Or, au sein de l'université les changements d'affectation sont courants. Un utilisateur changeant de composante change aussi de DN et devient ainsi, pour Nextcloud, un nouvel utilisateur et n'a plus accès à ses fichiers. Nous avons ainsi changé d'annuaire LDAP pour opter pour une organisation des utilisateurs à plat, ne reflétant plus les affectations.

4 Technique

4.1 Principes

Faute de pouvoir précisément prévoir l'enjouement que le projet allait générer, l'objectif était d'obtenir une infrastructure entièrement répartie et redondée, privilégiant ainsi le mode multi master et si possible nous assurant de pouvoir rajouter des nœuds en cas de besoin. Nous utilisions déjà le couple IPVS/Keepalived pour le rester de notre infrastructure. Nous avons donc pu facilement y intégrer les services nécessaires à l'infrastructure de notre nouveau projet. Une autre alternative aurait été HAProxy mais, celui-ci ne supportant que le protocole TCP, nous avons privilégié IPVS afin de nous assurer de pouvoir fournir de la haute disponibilité et une répartition des protocoles UDP pour parer à toute éventualité.

La base de notre infrastructure dédiée à UNCloud est composée de 4 serveurs physiques dotés de 784Go de RAM, 2x20 cœurs et 400Go de SSD. Sur ces 4 serveurs nous hébergeons les différents nœuds de notre infrastructure, virtualisés avec KVM . Les nœuds sont alors synchronisés entre eux à l'aide du couple lsyncd et csync2.

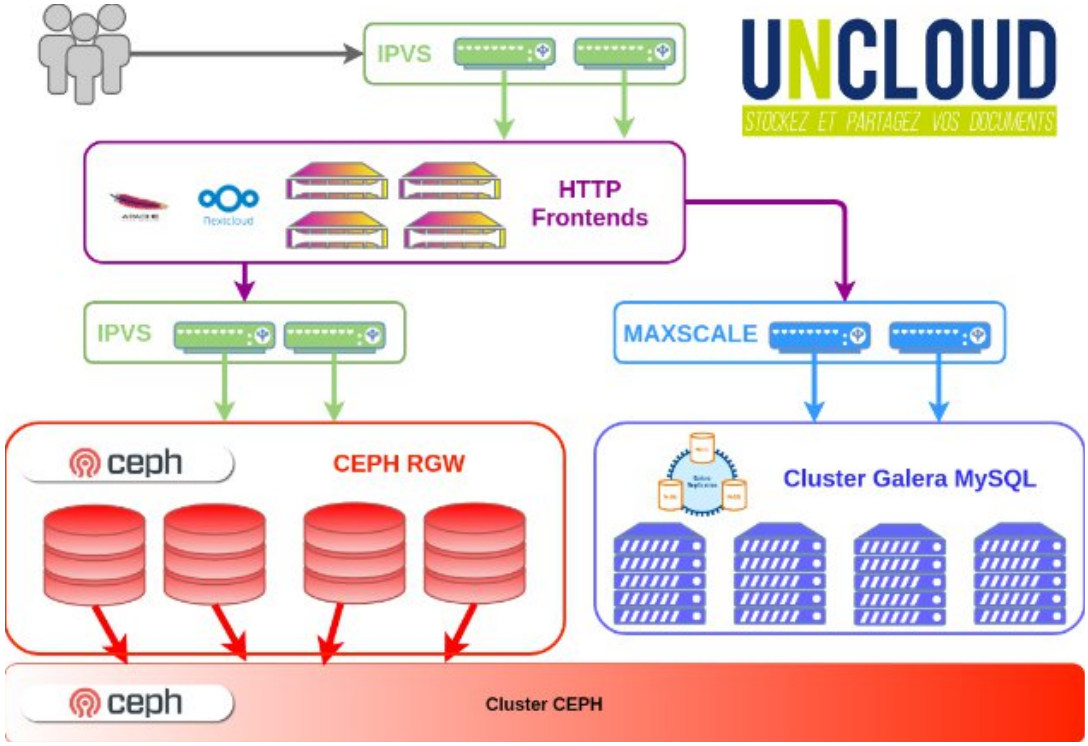

Figure 1 - Aperçu de l'infrastructure en place

4.2 Nextcloud

Par mesure de sécurité, l'infrastructure web de l'université de Nantes (plus de 150 sites) se trouve intégralement sur une DMZ privée derrière un cluster de 4 frontaux HTTP. L'objectif de ces serveurs est d'analyser les requêtes HTTP provenant de l'extérieur, de les transférer vers l'intérieur et de retourner au client la réponse du serveur. Nous utilisons mod_security pour détecter, signaler ou bloquer toute requête ou réponse potentiellement problématique.

Toujours dans l'optique de permettre une haute disponibilité et une répartition de la charge, Nextcloud est ainsi déployé sur 4 nœuds, situés derrières les frontaux mod_security, utilisant Apache2.4 et PHP 7.2.

Le grand challenge des applications web réparties est la persistance des sessions et des uploads entre les nœuds sans avoir à sacrifier la répartition de la charge. Les requêtes HTTP étant réparties aléatoirement sur 4 les nœuds Nextcloud, un utilisateur peut être pris en charge par un serveur puis par un autre à la requête suivante. Si la session est associée à l'hôte où a lieu la requête, celle-ci sera rendue caduque à la requête suivante. De même un fichier uploadé sur un hôte ne pourra pas être traité sur un autre.

Si par défaut PHP utilise un gestionnaire de session basé sur des fichiers locaux, il lui est possible de stocker ses sessions dans une base Redis. Redis étant "clusterisable" cela permet ainsi aux 4 nœuds Nextcloud d'accéder aux sessions PHP peut importe le nœud à l'origine de la session (Fig 2).

```
session.save_handler = "rediscluster"
 session.save_path =
"seed[]=noeud1:7000&seed[]=noeud1:7001&seed[]=noeud2:7000&seed[]=noeud
2:7001&seed[]=noeud3:7000&seed[]=noeud3:7001&seed[]=noeud4:7000&seed[]
=noeud4:7001&read_timeout=2&failover=error&persistent=1"
```

Figure 2 - Configuration Redis

Pour les données d'upload, Nextcloud utilise le mécanisme de "chunked transfer encoding" de la norme HTML 1.1. Chaque upload se fait par fragment de 10Mo. A chaque fin d'upload le fragment est envoyé sur le stockage S3 alors que le fragment suivant est uploadé sur un autre hôte. Une fois tous les fragments constituant le fichier original présents dans la base de stockage, le fichier est reconstitué sur un nœud Nextcloud, uploadé dans le stockage et les fragments sont supprimés. Cette méthode offre le double avantage de pouvoir paralléliser le traitement des fragments tout en répartissant la charge sur les nœuds.

Notre cluster Redis utilisé pour le stockage des sessions PHP est aussi utilisé par Nextcloud comme cache mémoire distribué (distributed memcache) permettant ainsi aux nœuds Nextcloud de partager leurs données en cache et d'accélérer globalement les traitements (Fig. 3)

```
'memcache.distributed' => '\\OC\\Memcache\\Redis',
  'redis.cluster' =>
  array (
    'seeds' =>
    array (
      0 => 'noeud1:7000',
      1 => 'noeud1:7001',
      2 => 'noeud2:7000',
      3 => 'noeud2:7001',
      4 => 'noeud3:7000',
      5 => 'noeud3:7001',
      6 => 'noeud4:7000',
      7 => 'noeud4:7001',
     ),
    'timeout' => 2,
    'read_timeout' => 2,
    'failover_mode' => 1,
  ),
```

Figure 3 - Configuration memcache NextCloud

4.3 Stockage

Ceph est un système de stockage objet opensource, distribué et réparti, basé sur RADOS (Reliable Autonomic Distributed Object Store).

L'université de Nantes, profitant de son implantation géographique multi-sites, exploitait déjà plusieurs clusters Ceph sur 3 de ces sites. Le cluster dédié à UNCloud est composé de 9 serveurs physiques de 16 disques de 10To chacun réparti en 3x3. Chaque plaque géographique recevant un réplicat. ll nous est ainsi possible de perdre un datacenter complet sans impacter l'intégrité des données. Un mécanisme de réplication asynchrone permet d'obtenir une copie du cluster de production en cas d'incident critique nécessitant un plan de reprise d'activité.

Un des atouts de Ceph est de proposer différents mode d'accès compatibles avec notre besoin :

- **cephFS** : représente l'accès en mode fichier. Il aurait possible d'utiliser cephFS, malheureusement celui-ci ne permet pas encore d'obtenir des performances équivalentes à un système S3.
- **Rados Gateway (RGW)** : il s'agit d'un service Ceph permettant d'ajouter une interface S3 à la base de stockage objet. S3 est un protocole de stockage objet spécialement conçu pour le web par Amazon. Ainsi RGW est un serveur web (CivetWeb) permettant l'accès à la base de données objet avec des instructions HTTP classique : GET pour la lecture d'un objet, PUT pour l'ajout un objet, DELETE pour la suppression, etc. L'avantage principal de S3 est qu'il est très simple de faire de la répartition de charge sur du HTTP. De plus le daemon étant léger et la configuration très basique, il est aussi très facilement déployable à la demande. Ce fut donc notre choix principal. Nous savions ainsi qu'en cas de besoin nous pourrions rapidement ajouter des nœuds de manière automatique.

Nous avons donc déployé 4 noeuds RGW dans un cluster IPVS accessible via une adresse HTTP classique. Par exemple http://rados.ha.univ-nantes.prive permet de répartir l'ensemble des requêtes S3 sur les 4 nœuds.

Figure 4 - Débit dans le cluster RGW (écriture en haut, lecture en bas).

4.4 SQL

Nextcloud est compatible Mysql/MariaDB, PostgreSQL et Oracle. Ayant principalement des compétences Mysql, le choix de MariaDB, par sa présence par défaut sur Debian, notre distribution Linux de référence, s'est naturellement imposé. De plus, MariaDB propose deux services qui nous seront utiles : Galera, leur solution de clustering et Maxscale, leur loadbalancer.

4.4.1 Galera

Galera est la réponse de MariaDB à Mysql Cluster. Une solution de réplication multimaster relativement simple à déployer. Nous utilisons ainsi 4 nœuds Galera. Sur chaque nœud, la configuration d'InnoDB reste traditionnelle. la configuration de Galera se résume à ceci (Fig. 5) :

```
wsrep_on=ON
 wsrep_provider=/usr/lib/galera/libgalera_smm.so
 wsrep_cluster_name="test_cluster"
 wsrep_cluster_address=gcomm://node1,node2,node3,node4
 wsrep_sst_method=xtrabackup-v2
 wsrep_sst_auth="sstuser:monmdpsst"
```

Figure 5 - Configuration de Galera

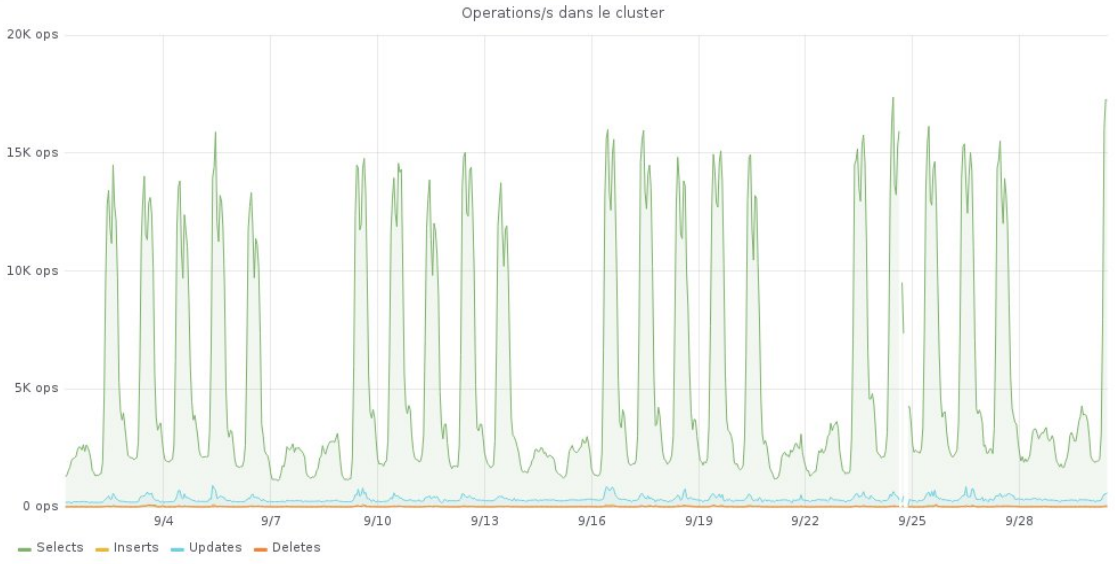

Figure 6 - Opérations dans le cluster Galera en septembre 2019

A l'usage, le nombre de requêtes était tel que régulièrement des verrous (deadlocks) apparaissaient au alentour de 1000 requêtes d'écriture à la seconde. Un deadlock se produit lorsqu'une requête en lecture tente d'accéder à une données qui est en cours de modification. Il a donc été décidé de dédier un nœud à l'écriture et les 3 autres à la lecture. C'est là qu'intervient Maxscale.

4.4.2 Maxscale

Maxscale est un répartiteur de charge SQL conçu par MariaDB. Il offre différents modes de répartition dont un mode dit "readwritespit" permettant d'élire de manière aléatoire un nœud qui sera alors dédié à l'écriture. Toutes les requêtes en écriture seront ainsi envoyées sur ce nœud (Fig. 7) alors que les requêtes en lecture seront réparties sur les autres nœuds (Fig. 8):

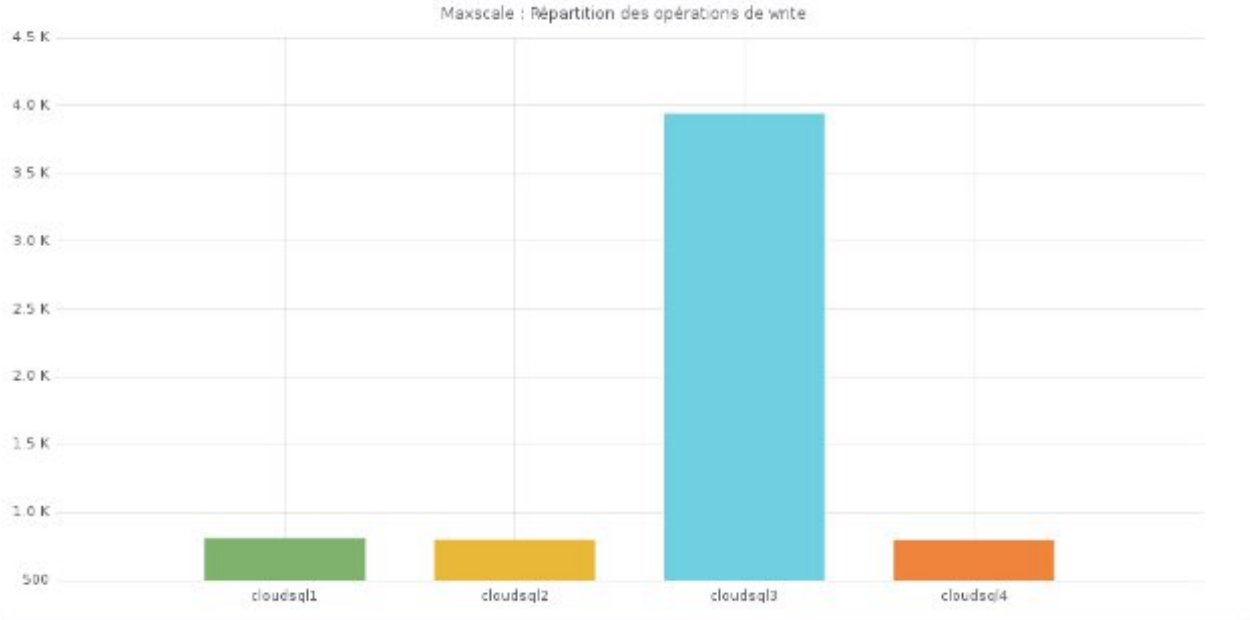

Figure 7 - Répartition des opérations en écriture dans le cluster.

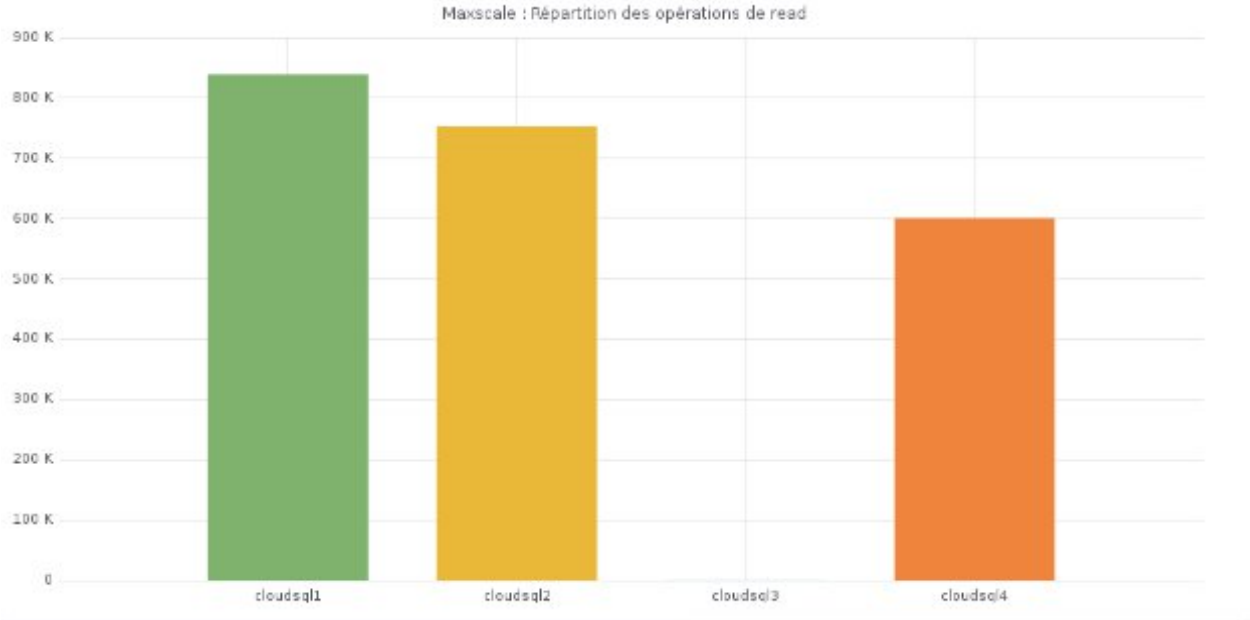

Figure 8 - Répartition des opérations en lecture dans le cluster.

4.5 Redis

Redis est une base de données clés/valeurs en mémoire. Il est prévu avec Redis de répartir les données sur plusieurs nœuds mais la réplication multi-master n'est pas possible, Redis ne sait que répliquer en master-slave. Nous avons donc réparti les clés sur 4 nœuds maîtres. Et pour chacune de ces instances maîtres, une instance slave réplique son sous ensemble de clés sur une machine physique différente.

Chaque machine physique héberge une machine virtuelle recevant ainsi deux instances Redis : une instance master (port TCP 7000) et une instance slave (port TCP 7000). Les instances 1 tournent sur la machine physique n°1, les instances 2 tournent sur la machine physique n°2 etc. Ainsi :

- slave1 réplique le master4
- slave2 réplique le master1
- slave3 réplique le master2
- slave4 réplique le master3

Si une machine physique ou virtuelle tombe, l'intégralité des sous ensembles reste disponible. Le sous ensemble du master absent est disponible via son slave (Fig. 9) [3] [4].

Une autre spécificité de Redis est que l'aiguillage des requêtes vers le nœud hébergeant la bonne clé doit être géré par le client. Ainsi les 8 nœuds composant notre cluster sont explicitement déclarés dans la configuration PHP et de Nextcloud.

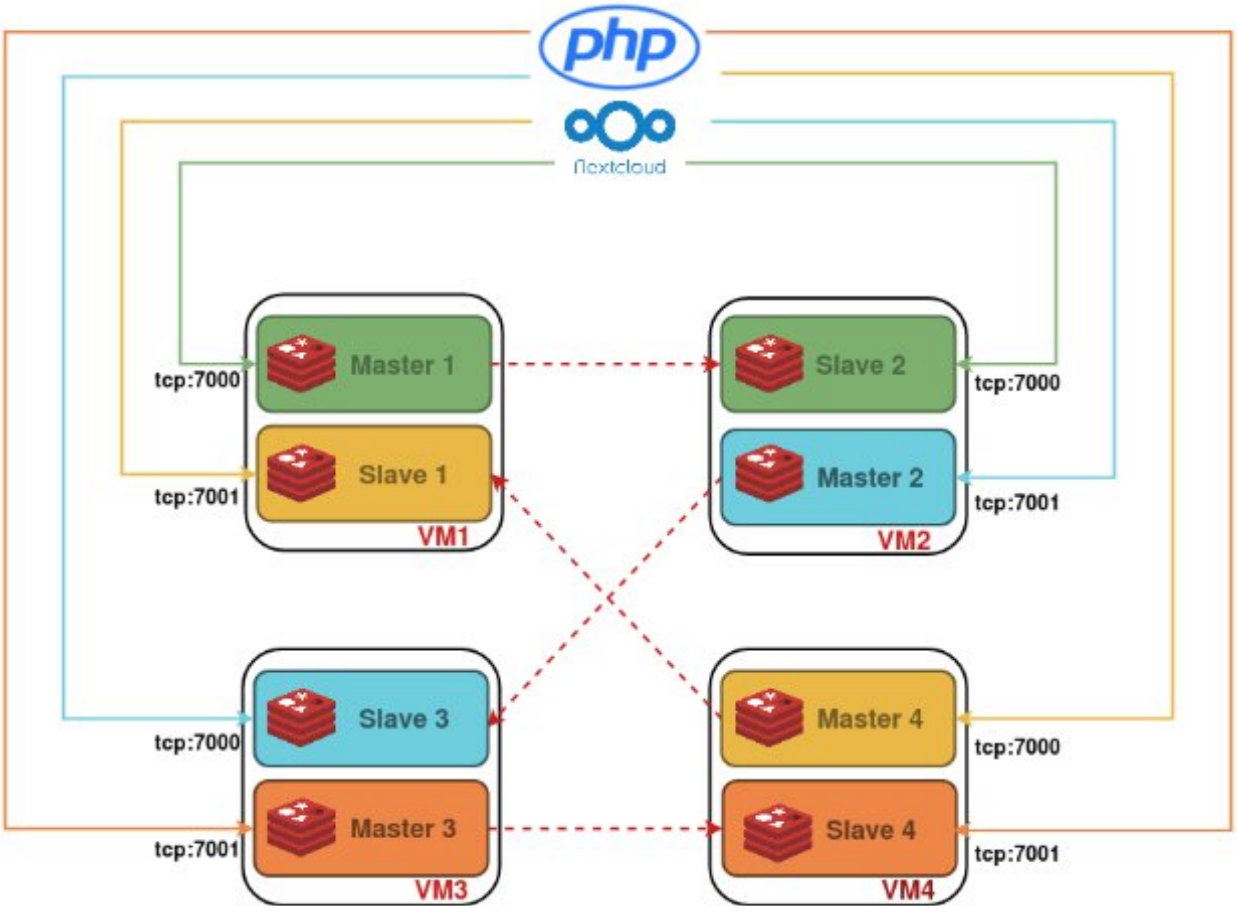

Figure 9 - Répartition et réplication des clés dans le cluster Redis

4.6 OnlyOffice

L'édition collaborative dans UNCloud est réalisée par une plate-forme OnlyOffice. Celle ci est pratiquement disjointe de celle d'UNCloud. La liaison entre les deux est assurée par l'App OnlyOffice pour Nextcloud. Elle consiste en 2 DocumentServeurs OnlyOffice, un cluster RabbitMq qui assure la liaison des message entre les deux, un cluster Redis dédié pour la gestion du cache, un cluster MySQL, ainsi qu'un répertoire temporaire partagé. Ce dernier, exporté en NFS constitue la seule section critique du projet (Fig. 10) . [5]

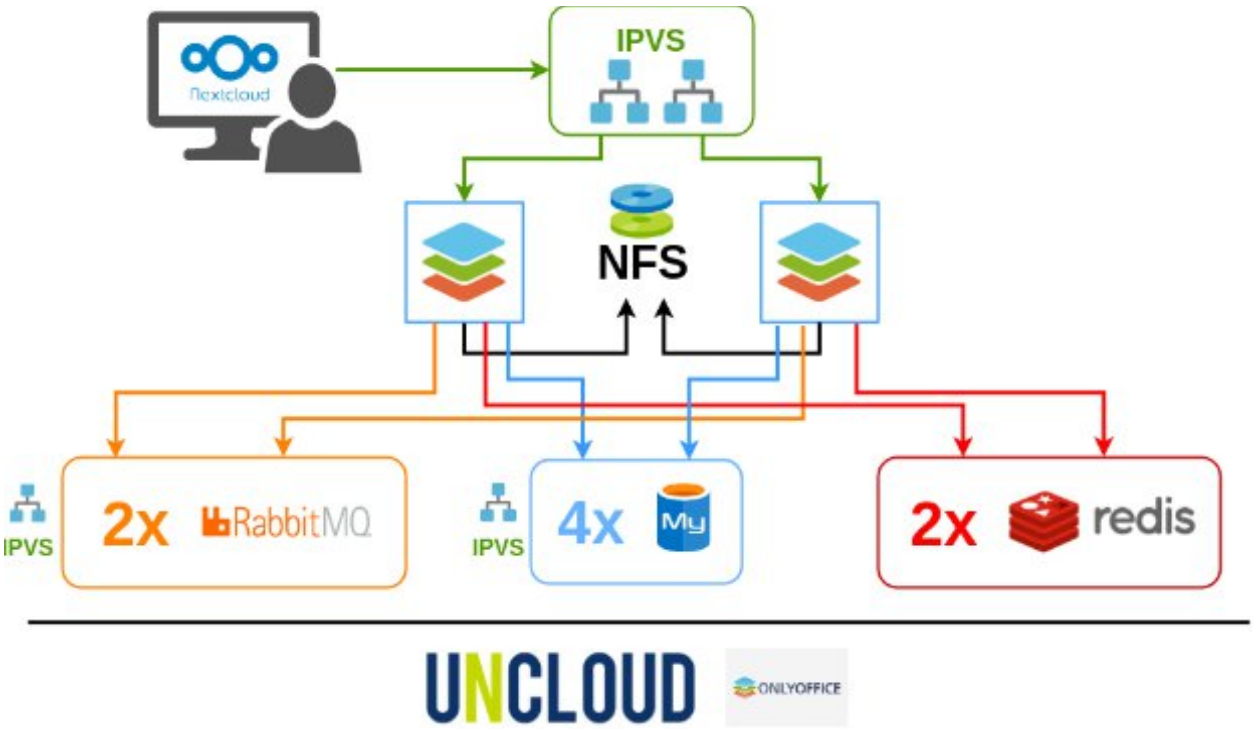

Figure 10 - Fonctionnement du cluster OnlyOffice

La charge des serveurs OnlyOffice est faible, toute la partie intelligence de l'édition se situant au niveau du navigateur. Lors de l'ouverture d'un document celui-ci est converti au format interne d'OnlyOffice avant d'arriver au navigateur, l'opération inverse est effectuée lors d'une sauvegarde.

Comme pour l'ensemble du projet, le cluster OnlyOffice dispose d'une instance de test pour valider les mises à jour.

5 Supervision et indicateurs

5.1 Supervision

La DSIN supervise son infrastructure avec Zabbix. C'est une plate-forme de supervision opensource dont le fonctionnement repose sur 3 concepts :

- la collecte d'items.
- des déclencheurs permettant de générer un événement d'une certaine criticité lorsqu'un ou plusieurs items valident une opération logique.

- des actions à exécuter en fonction des événements déclenchés.

Les items peuvent être collectés par différents mécanismes :

- l'agent Zabbix, un service Linux et Windows collectant les données en local et les communiquant au serveur Zabbix.
- SNMP.
- HTTP agent, un mécanisme permettant d'interroger des APIs et d'en extraire des items.
- scénario web qui permet d'émuler la navigation utilisateur sur un site web. Ce mécanisme est capable de soumettre des formulaires, collecter des valeurs dans les pages pour pouvoir ensuite les utiliser dans un formulaire d'authentification. A chaque page, un scénario web collecte sous forme d'items les temps de réponse, le temps de téléchargement, le code HTTP , etc.

Pour superviser UNCloud, l'agent Zabbix est déployé sur toutes les machines virtuelles impliquées dans l'infrastructure du service. Par son intermédiaire toute la supervision classique de l'usage des ressources est opérée.

Pour chaque service une supervision spécifique est mise en place avec l'agent Zabbix qui contrôle la disponibilité du service global en utilisant le cluster, la disponibilité du service sur chaque nœud du cluster ainsi que des métriques liées au service. Certains de ces items sont des indicateurs et sont regroupés dans des tableaux de bord spécifiques :

- pour le service SQL :
  - le nombre de connexions SQL
  - le nombre et le type de requêtes
  - le temps de résolution de chaque requête
  - la latence dans la réplication
  - le nombre de requêtes identifiées comme lente par le système
- Pour le service de passerelle S3 - Ceph :
  - le nombre de requêtes
  - le débit en lecture et en écriture
  - la latence des accès
  - l'usage du cache
- Pour Redis :
  - le nombre de commandes
  - le nombre de clés stockées
- Pour le service HTTP :
  - le nombre de requêtes
  - le débit HTTP
  - l'état des threads du service apache2
- Pour Nextcloud :
  - L'agent HTTP est utilisé pour collecter à partir de l'API de Nextcloud les métriques suivantes:
  - Le nombre d'utilisateurs enregistrés sur UNCloud (indicateur)
  - Le nombre d'utilisateurs actifs lors des 5 dernières minutes, lors de la dernière heure et lors de la dernière journée
  - le nombre de partage nominatifs et par fédération (indicateurs)
  - la taille de la base de données
  - le nombre de fichiers hébergés (indicateur)
  - Un scénario web permet de quantifier l'expérience utilisateur en reproduisant le scénario suivant :
  - accès à la page d'accueil d'UNCloud
  - authentification sur le service

- chargement de la racine de l'utilisateur
- navigation sur la page des paramètres de l'utilisateur
- déconnexion de l'utilisateur

Pour chaque service, un tableau de bord est généré avec Grafana:

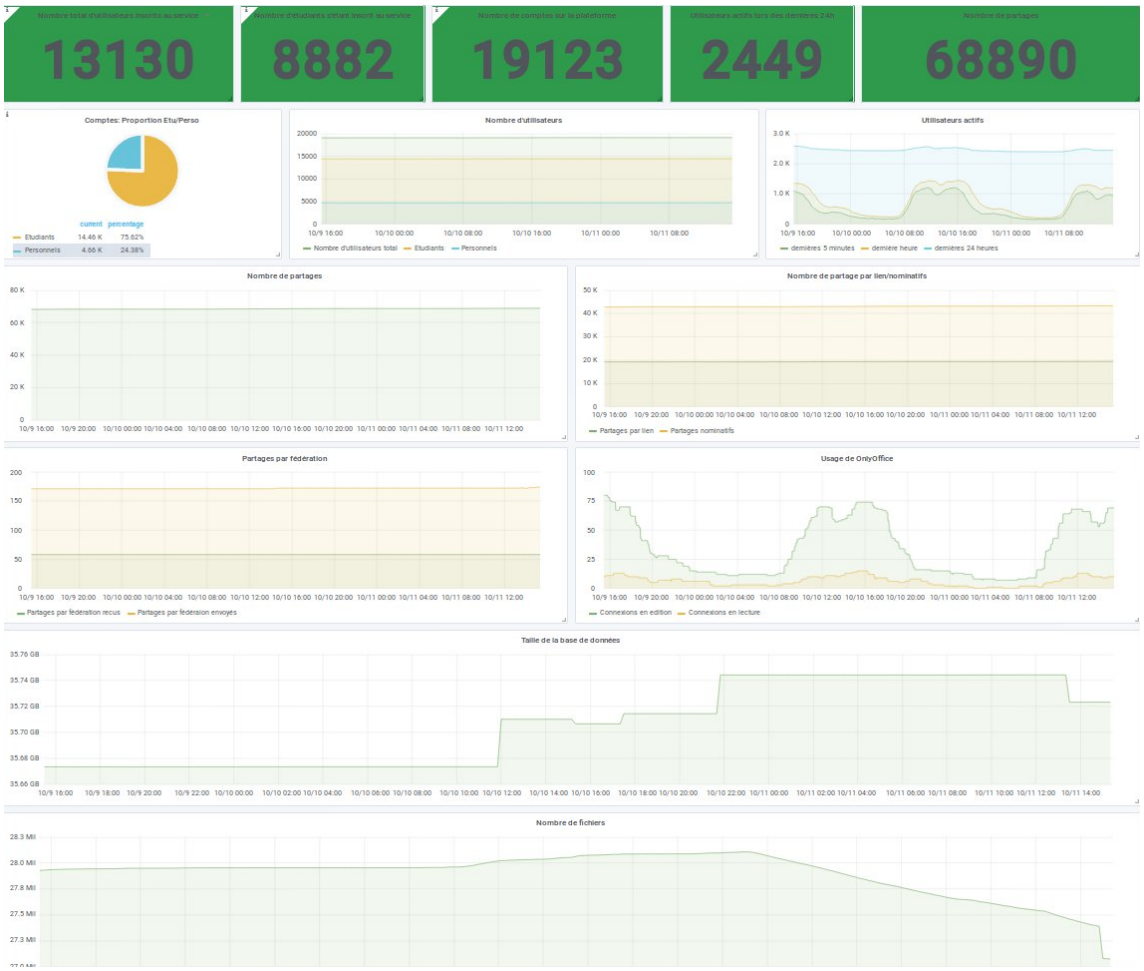

Figure 11 - Tableau de bord "indicateurs"

5.2 Cycle de vie des comptes et des données

Les comptes universitaires sont divisés en deux grandes populations, les étudiants et les personnels. Si dans Nextcloud rien ne différencie un compte étudiant d'un compte personnel, leur activation diffère. Un compte Nextcloud est automatiquement activé à la création d'un compte personnel à l'université. En revanche, les étudiants doivent activer leur accès à UNCloud manuellement dans l'application de gestion de leur compte universitaire. Le but de cette activation manuelle est de maîtriser le nombre de comptes actifs dans Nextcloud afin de limiter le nombre de token de support nécessaires.

Lors de son cycle de vie, si le compte est inactif depuis plus de 6 mois, il est automatiquement désactivé. Ceci est produit par le basculement d'un flag LDAP sur lequel Nextcloud se base pour rapatrier ses utilisateurs. Nous avons ainsi uniquement des comptes actifs de moins de 6 mois et ceci pour des questions coût de support, celui-ci étant lié aux nombre d'utilisateurs actifs. Lorsque le compte est inactif depuis 6 mois supplémentaires, soit au total 1 an, il est définitivement supprimé. Cette dernière procédure

n'est pas encore active sur notre instance. En effet la fermeture définitive d'un compte implique un effacement des données, et une potentielle ré-organisation des partages.

Les données ont un cycle de vie plus complexe que celui des comptes. Elle peuvent en effet être dans un état normal, partagées, versionnées, dans la corbeille ou supprimées.

Le postulat de base que l'on s'est fixé lors de la mise en place d'UNCloud est qu'à la vue du volume de données à traiter et de la technologie de stockage retenue, il ne pourrait pas y avoir de restauration des données de façon individuelle. Pour palier à cet écueil, deux solutions ont été mise en place.

La première, pour ce prémunir d'une modification intempestive d'un fichier, est la gestion des versions. Il permet de résoudre le cas d'un fichier écrasé par un autre de même nom.

Pour gérer l'effacement des fichiers, la corbeille protège naturellement l'utilisateur. Pour éviter que celle-ci ne soit vidée accidentellement par l'utilisateur, toute procédure de vidage de la corbeille par l'utilisateur a été désactivée. Celle-ci purge automatiquement les fichiers présents depuis plus de 60 jours.

Ces deux mécanismes permettent donc de pallier les accidents qui pourraient arriver aux fichiers. Les données étant la partie sensible de la solution, c'est un mode de fonctionnement qui a été soigneusement présenté et validé par les instances de l'Université.

5.3 Sauvegardes

Les sauvegardes ne sont donc pas réalisées dans un but de restauration individuelle des fichiers ou des comptes mais permettent la mise en place d'un plan de reprise d'activité.

La complexité du mode de stockage de l'information réside dans son éclatement en deux endroits différents. Si le contenu des fichiers est lui stocké dans le cluster Ceph, il ne s'y trouve que sous forme de clés S3. La liaison avec le nom et le chemin du fichier se trouve sous forme de référence dans la base de données. Pour avoir une sauvegarde cohérente, il faut donc à la fois sauvegarder le volume S3, mais aussi les tables dans la bases de données.

L'enjeu réside donc dans la production d'un fichier texte le plus simple possible faisant la liaison entre nom de l'utilisateur, nom complet du fichier, et sa clé S3. Il faut également produire, soit un snapshot soit une réplication du bucket S3 à l'instant de la production du fichier.

6 Résultats

6.1 Adoption

Dès la phase de test, le projet a rencontré un franc succès. En effet après les 3 mois prévus de phase de test, la plate-forme a été maintenue jusqu'à la mise en service de la solution définitive. Ceci montre que le besoin d'un tel outil est réel et attendu aussi bien par les étudiants que par les personnels. Le nombre de comptes étudiants augmente régulièrement avec un pic significatif à chaque rentrée. Pour les personnels, la courbe de progression a été plus rapide dans un premier temps et présente maintenant une inflexion beaucoup plus légère. Au bout de 20 mois, la population étudiante représente 73% des comptes sur UNCloud. En septembre 2019 nous avons 2600 utilisateurs quotidiens différents.

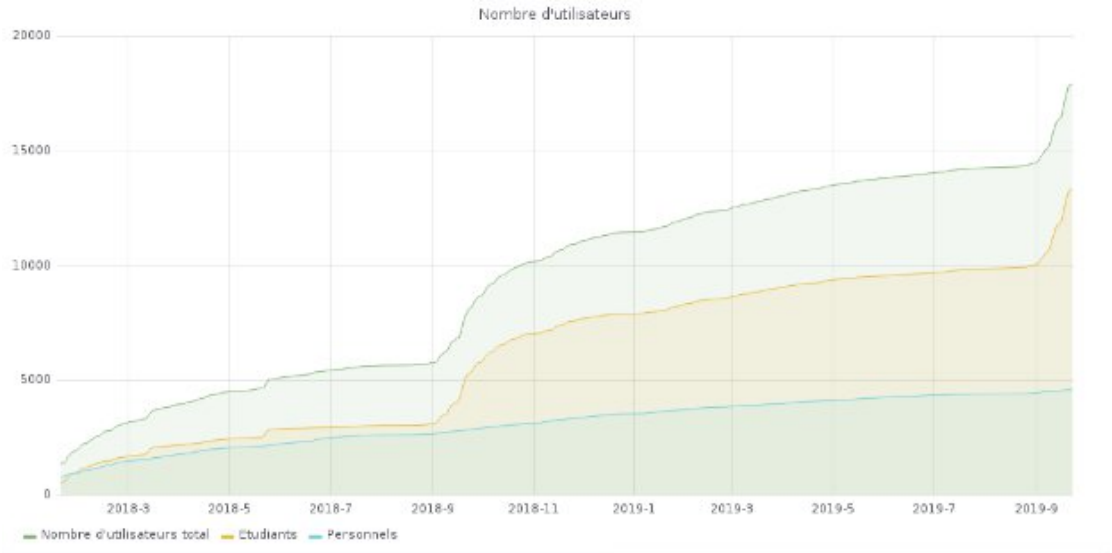

Figure 12 - évolution du nombre d'utilisateurs ayant accès à la plate-forme

6.2 Les évolutions

L'évolution la plus importante mais aussi la plus attendue est sans conteste le lancement de l'édition collaborative en ligne. Le fait de pouvoir éditer un même document de façon simultanée est une avancée majeure dans le mode de travail en équipe. Le choix d'OnlyOffice, même si il ne fournit pas un remplacement à 100% aux solutions client de bureau type MS Office ou LibreOffice, permet dans le cas de documents de complexité moyenne une édition en ligne efficace.

D'autres ajouts, même si ils sont de moindre importance ont été bien accueillis. Le lancement de Deck, gestionnaire de tâche de type Kanban, permet de partager ses listes de tâches entre collaborateurs. Pour une utilisation quotidienne simple il constitue une alternative viable à la plate-forme "Trello". La possibilité d'éditer des diagrammes en ligne via DrawIO et la connexion au service Jabber universitaire directement dans l'interface UNCloud ont également été ajoutées.

6.3 Les prochaines étapes

Prochainement trois briques essentielles au travail collaboratif vont être mises en service, à savoir Contact, Webmail et Agenda. Les fonctionnalités actuellement proposées par ces 3 applications n'étant pas à équivalence avec ce qui est offert aux utilisateurs via les services numériques universitaires, un marché de développement a été passé avec la société Nextcloud GmbH pour augmenter le niveau fonctionnel de ces composants. La réalisation de ces développements devrait permettre la mise en place d'une solution complète de travail collaboratif.

Pour compléter cette mise en œuvre, un solution de visioconférence "légère" devrait être intégrée à la plate-forme UNCloud. Dans un contexte d'augmentation importante du télétravail, cela permettra de faciliter les réunions distantes.

7 Conclusion : retour sur expérience

Le stockage objet nous permet de gagner en performance et en scalabilité mais complexifie la sauvegarde. Les fichiers n'étant plus présent sous forme d'arborescence, il n'est pas possible d'utiliser un mécanisme sauvegarde traditionnel qui indexe le système de fichiers. Une corbeille qu'il n'est pas possible de vider avec une durée de rétention statique de 60 jours permet de réduire le besoin de restauration individuelle de fichier. Un réplicat asynchrone du cluster Ceph nous protège en cas de destruction du cluster de production.

Il est habituel d'identifier les utilisateurs par leur UID. C'est ce qui a été reproduit dans UNCloud. Malheureusement l'UID, à l'université, n'est pas une valeur invariable. Il arrive effectivement que des utilisateurs réclament une modification de leur nom d'utilisateur. Cette opération engendre dans Nextcloud la création d'un nouveau compte et l'utilisateur ne peut plus accéder à ses anciens fichiers. Nextcloud permet désormais de préciser dans la configuration LDAP un attribut différent de l'UID afin d'identifier de manière unique et invariable les utilisateurs. Cette modification n'est malheureusement pas possible une fois les utilisateurs présents dans la base de données.

Lors de la phase de test les utilisateurs se sont très rapidement emparés de l'outil de synchronisation Nextcloud leur permettant de travailler sur les fichiers présents sur leur espace UNCloud depuis leur poste de travail. Malheureusement, les utilisateurs n'ont pas toujours un espace suffisant pour recevoir les 100Go de leur espace UNCloud et n'ont pas forcement les connaissances ou la volonté nécessaire pour la configuration fine de l'outil de synchronisation. De plus, la synchronisation est souvent très complexe et les cas de conflits sont réguliers. Enfin, l'édition collaborative de document entre en conflit avec l'édition locale. En cas de modification simultanée de documents, le dernier qui enregistre le document écrasera la version de l'autre. Pour remédier aux problématiques liées à la synchronisation, Nextcloud travaille à un connecteur de type "drive" qui permettra de connecter son espace UNCloud à son poste de travail sans avoir à recourir à une synchronisation.

UNCloud étant désormais ancré dans les habitudes des utilisateurs, nous constatons que ses données d'équipes ou de services habituellement stockées sur les serveurs de fichiers des composantes migrent progressivement vers les comptes UNCloud personnels des utilisateurs. Cette tendance pose plusieurs problèmes. Notamment parce que des données qui étaient à l'origine indépendantes de tout utilisateur se retrouvent désormais associées à un compte et soumises au cycle de vie de ce compte. Sachant que la tendance est irréversible et que les utilisateurs opteront de plus en plus pour UNCloud plutôt que pour les serveurs de fichiers traditionnels, il est prévu de mettre en place l'application Nextcloud "groupfolder". Cette application permet d'associer un dossier par groupes auxquels l'utilisateur appartient. Tout document déposé dans ce dossier est immédiatement désolidarisé de l'utilisateur, de son cycle de vie et de son quota pour désormais appartenir au groupe.

Bibliographie

[1] CNIL. The CNIL's restricted committee imposes a financial penalty of 50 Million euros against GOOGLE LLC. Janvier 2019; https://www.cnil.fr/en/cnils-restricted-committee-imposes-financial-penalty-50-million-euros-against-google-llc

[2] Nestori Syynimaa and Tessa Viitanen. Is My Office 365 GDPR Compliant. 299-305; https://www.scitepress.org/papers/2018/67706/67706.pdf

[3] CodeFlex.Configuring and Running Redis Cluster on Linux; http://codeflex.co/configuring-redis-cluster-on-linux/

[4] Mauro Morales. Running Multiple Redis Instances. Avril 2106; https://medium.com/@MauroMorales/running-multiple-redis-on-a-server-472518f3b603

[5] Ascensio System SIA. Installing Document Server as a cluster. 2019; https://helpcenter.onlyoffice.com/fr/server/developer-edition/linux/install-cluster.aspx