CEPH
Une solution de stockage distribué Open Source

Journées d'actions Proxmox / Ceph
7-8/06/2022

Clermont-Ferrand

Sommaire

- Types de stockage
- Architecture réseau
- Intégration avec OpenStack, OpenShift, Proxmox
- Placement des données
  - Placement Group (PG)
  - CRUSH
  - Protection des données
  - Ceph Object Storage Daemon (OSD)
- Ceph mirroring
- Recommandations pour les matériels
- GT Ceph de resinfo

Les défis du stockage

- Evolution du volume et de la variété des données (fichiers, vidéos, disques de machines virtuelles ou de conteneurs, des datasets pour l'IA)
- Accélération de la création de contenu
- Supporter les changements et les évolutions de la capacité de stockage
- Intégrer les évolutions matérielles
- Changement des marchés, de gamme ou de constructeurs
- Faire mieux avec le même budget

CEPH

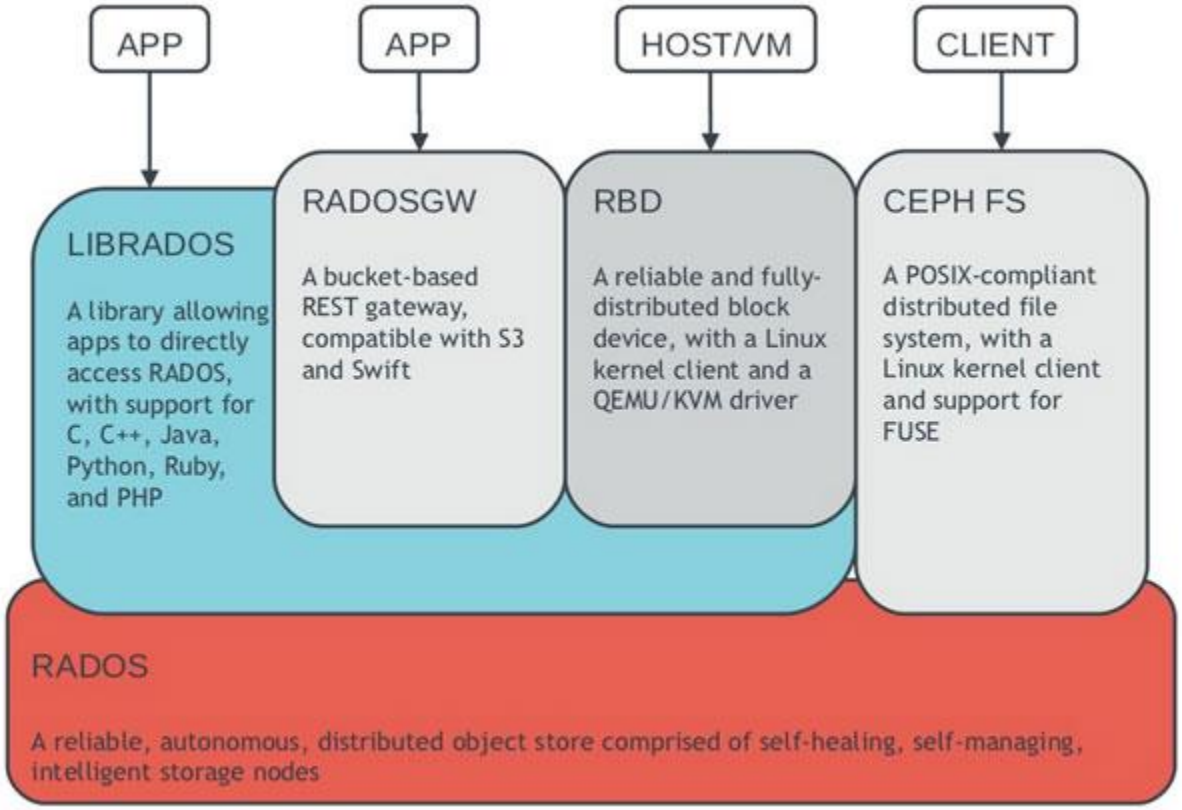
- Solution de stockage distribué
  - Pas de point unique de défaillance : les éléments sont redondés et fonctionnent en mode multi-actif
  - Extensible jusqu'à l'exaoctet
  - Conçu pour s’auto-réparer et réduire les coûts d’exploitation.
  - Offre une évolution dynamique (scale-out)
- Open Source
- Tolérance aux pannes
- Fonctionne avec du matériel standard

Historique des versions

- 12/2007 Sujet de thèse de Sage Weil
- 2011 Création de l'entreprise Inktank pour fournir du support
- 2012 Première version LTS
- 2014 Rachat par Red Hat
- Une version LTS tous les ans depuis 2015

07/2012 Argonaut (v 0.48) LTS
01/2013 Bobtail (v 0.56)
05/2013 Cuttlefish (v 0.61)
08/2013 Dumpling (v0.72) LTS
11/2013 Emperor (v 0.67)
05/2014 Firefly (v0.80) LTS
10/2014 Giant (v0.87)
04/2015 Hammer (v0.94) LTS
11/2015 Infernalis (v9.2)
04/2016 Jewel (v10.2) LTS
01/2017 Kraken (v11.2)
08/2017 Luminous (v12.2) LST
03/2018 Minic (v13.2)
03/2019 Nautilus (v14.2) LTS
03/2020 Octopus (v15.2) LTS
03/2021 Pacific (v16.2) LTS
04/2022 Quincy (v17.2) LTS

CEPH 3 types de stockage



http://www.sebastien-han.fr/blog/2012/06/10/introducing-ceph-to-openstack/

Type de stockage 1/2

Stockage Bloc (RBD)

- Accès à un disque distant (équivalent aux disques iSCSI).
- Support des snapshots, du thin provisionning, de la compression, live-migration
- Fonction de mirorring asynchrone, cache tiering (cache rapide en lecture ou écriture)
- Support via un module kernel ou nbd
- Intègre une Gateway iSCSI pour les systèmes non compatibles
- Pas de gestion d'accès concurrents, pas de déduplication

CEPH FS

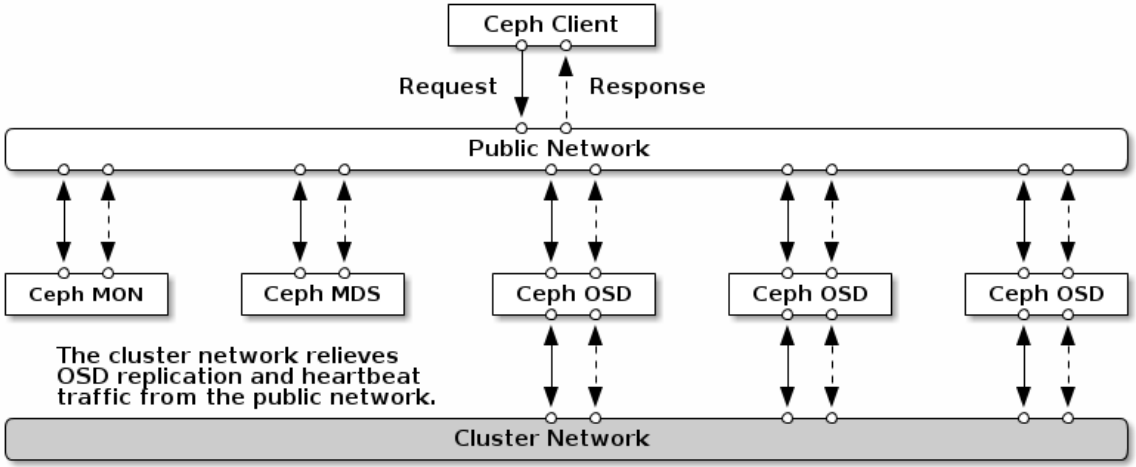
- Accès à un stockage concurrent en mode fichiers compatible POSIX (équivalent à NFS+ACLs)
- Nécessite le service MSD (Meta Data Server)
- Support via un module kernel ou fuse
- Support snapshot par répertoire, quota par répertoire (kernel 4.17 ou fuse)
- intègre une gateway NFS pour les systèmes non compatibles
- Montage depuis Windows via ceph-dokan
- Fonction de mirorring asynchrone par snapschot
- Non recommandé pour les disques virtuels (double écriture, sécurité)

Type de stockage 2/2

Stockage Objet (RADOSGW)

- Espace d'adressage linéaire, objets avec un identifiant unique, support des métadonnées et du versionning
- Exemple : stockage mail, photos, vidéos, documents
- Pas de fonction de partage, de verrous ou d'arborescence
- Compatible avec Amazon S3 ou OpenStack Swift
  - S3 intelligent tiers (différentes classes de stockage)
  - write once, read many (WORM)
  - Mirroring asynchrone par zones ou par bucket
- Export NFS (pour les backup)

Architecture réseau



https://docs.ceph.com/en/latest/rados/configuration/network-config-ref/

MON : service qui maintient une copie des cartes du cluster. Nécessite un nombre impair de services pour définir un quorum.

MDS(MetaDataServer) : service nécessaire uniquement avec CEPHFS. Assure l'enregistrement des métadonnées POSIX.

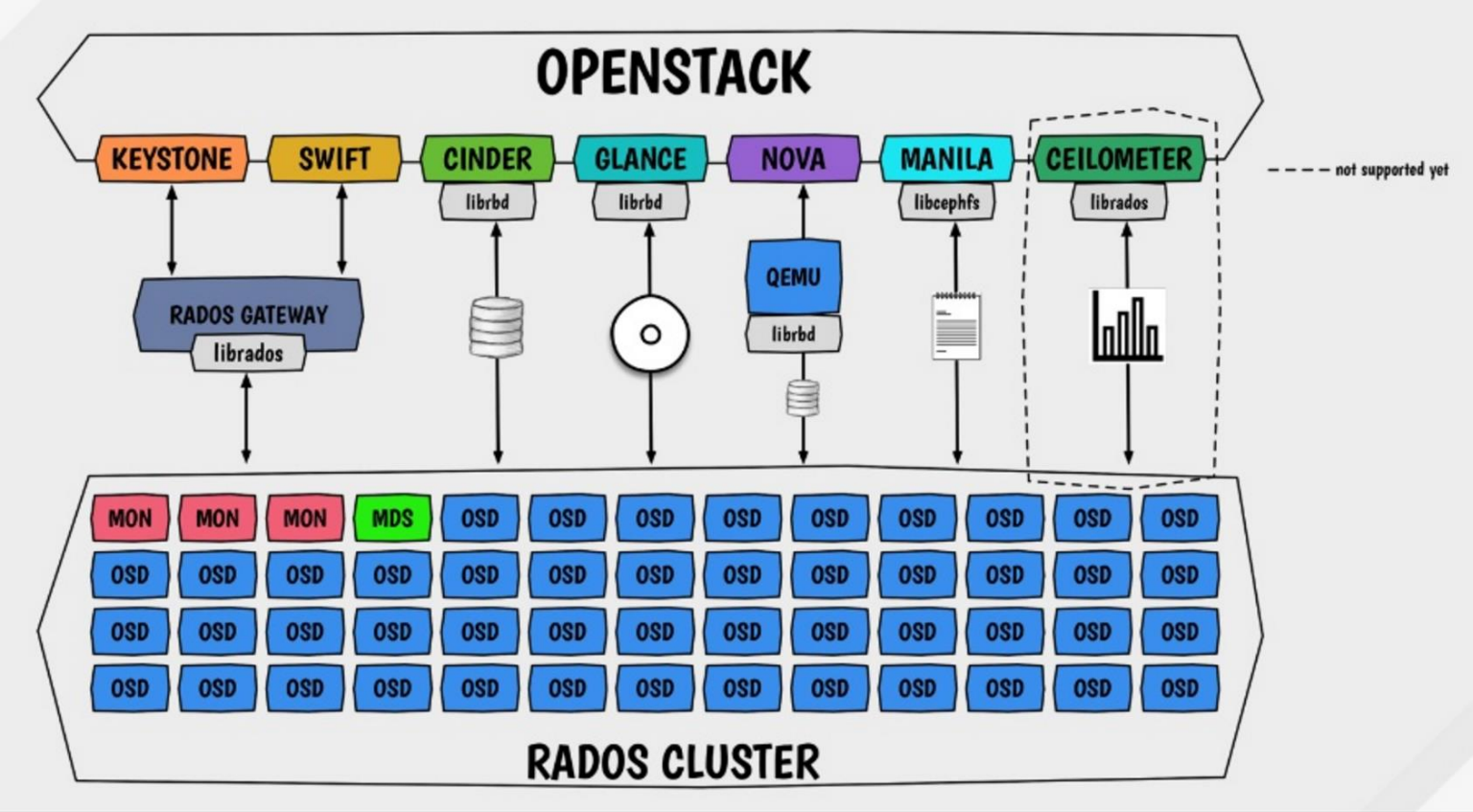
OSD(Object Storage Device) : service de stockage des objets. Utilise les disques locaux du serveur

Réseau publique : réservé aux clients pour l'accès aux MON et aux OSD

Réseau cluster : réservé pour la réplication des informations entre OSD. Réseaux privé non routé.

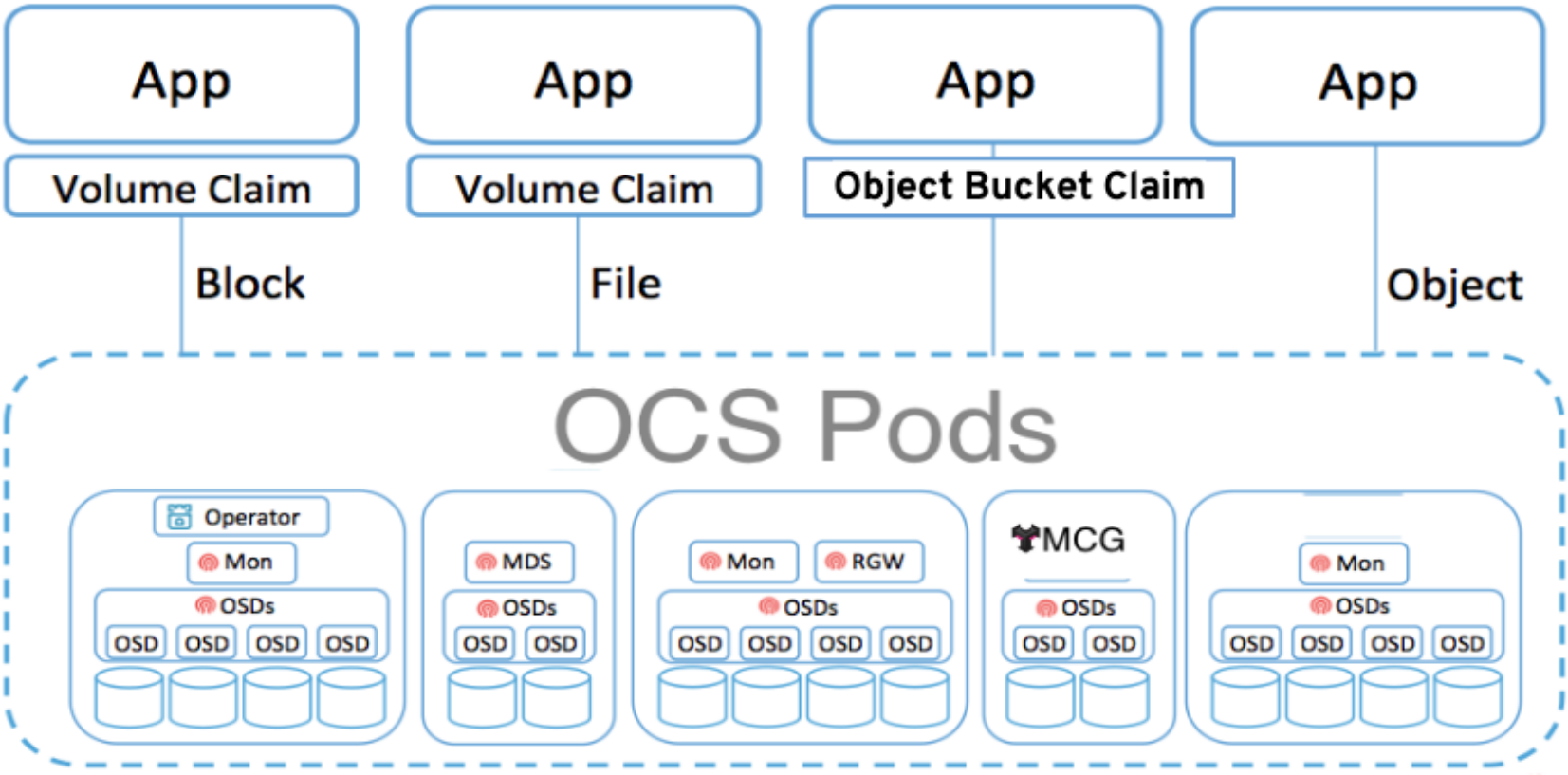
Les clients doivent seulement avoir accès au réseau public du cluster.

Intégration avec OpenStack



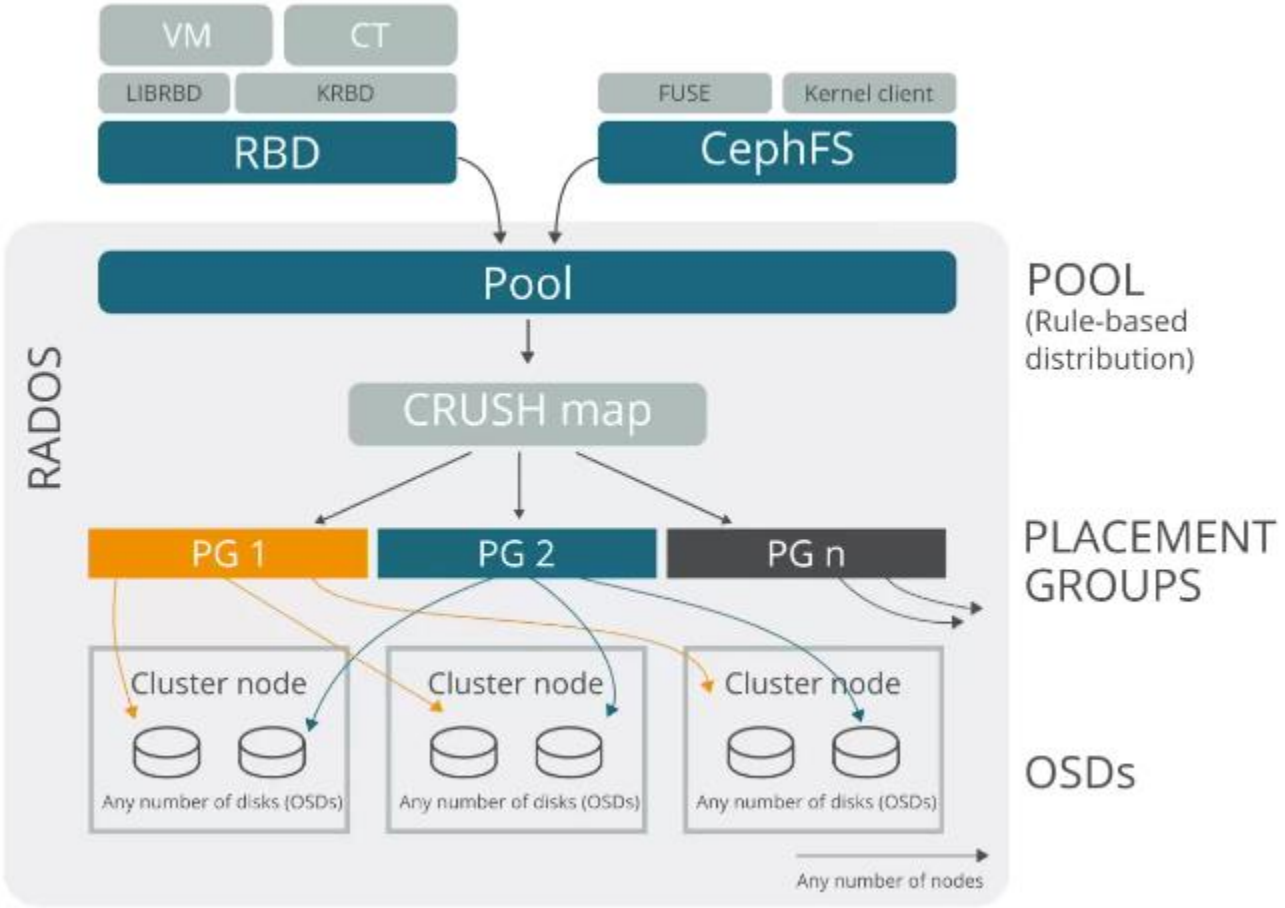
https://www.sebastien-han.fr/blog/2016/05/16/The-OpenStack-Ceph-Galaxy/

Intégration avec OpenShift



https://cloud.redhat.com/blog/deploying-your-storage-backend-using-openshift-container-storage-4

Intégration avec Proxmox



Placement des données

- Le stockage des objets se fait sur du matériel standard, sans utiliser des contrôleurs RAID.
- Permet une meilleure évolution du matériel (disques de différentes capacités, vitesses, technologies HDD, SSD, NVME, Optane)
- Reconstruction en parallèle =>Temps de reconstruction diminué
- Reconstruction commence sans attendre l'ajout d'un nouveau disque
- Pas besoin d'avoir des disques «hot-spare»

- Pools : Groupe logique pour stocker les objets.
  - Résilience : défini le type de réplications et le nombre de copies/réplicats d'un objet
  - Placement Group (PG) : agrégat d'objets qui permet de déterminer rapidement leurs états (accessibles, valides ou corrompus)
  - Snapshots : état des données à un instant donné. Permet un historique des objets
  - Quota : nombres d'objets ou volume maximum
  - Authentification : règles d'accès en lecture ou écriture pour les clients (service)

Placement Group (PG)

Les PG sont des fragments d'un pool.

Ils sont composés d'un groupe de daemons OSD qui se surveillent entre eux.

Les PG permettent :

- De monitorer le placement d'objets et leurs métadonnées
- De faciliter l'équilibrage des données dans le pool
- De vérifier l'interconnexion entre ces OSD (~30s)
- La reconstruction des données en cas de panne d'un ODS

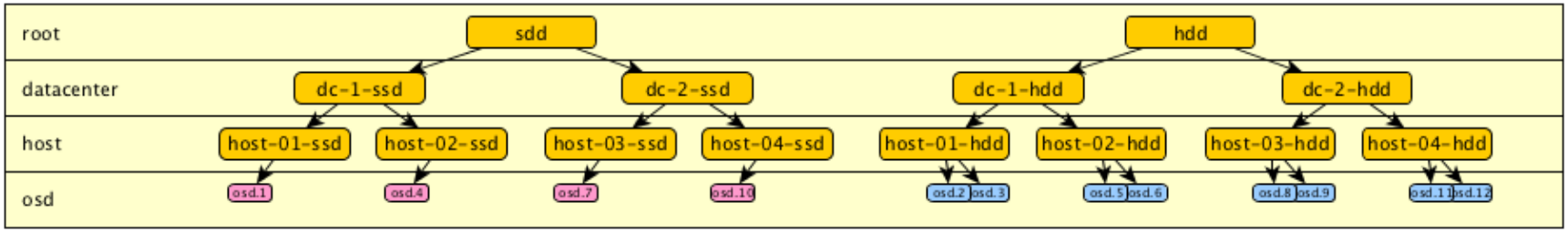
Il est possible d'augmenter et de diminuer le nombre de PG d'un pool.

Conseils :

- Utiliser un nombre de PG qui soit un multiple de 2.
- Activer l'autoscaling des pg par pool
- Calcul du nombre de PG : PG=(OSD*100)/poolsize
  poolsize=nombre de réplicats

CRUSH

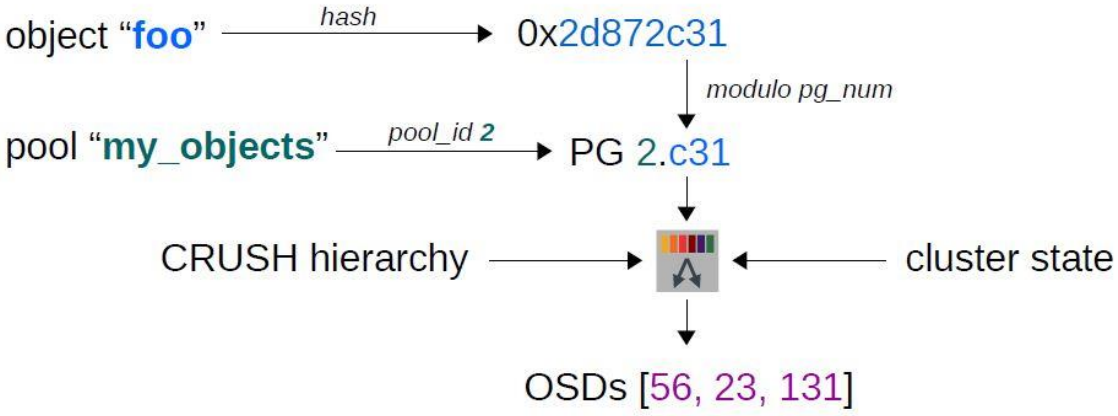
- CRUSH signifie «Controlled Replication Under Scalable Hashing»
- Algorithme de placement pseudo-aléatoire
- Calcul rapide, pas de boucle de recherche, déterministe
- Assure la distribution uniforme des informations sur les OSD
- Définit la topologie de l'infrastructure (nœuds de stockage, racks, rangées, Datacenter)
- Définit un poids à chaque OSD et une classe (SATA, SAS, SSD)
- Les clients ne connaissent que les OSD, pas les serveurs ou racks…
- Exemple : matériel avec un mixte en HDD et SSD



http://cephnotes.ksperis.com/blog/2015/02/02/crushmap-example-of-a-hierarchical-cluster-map

CRUSH

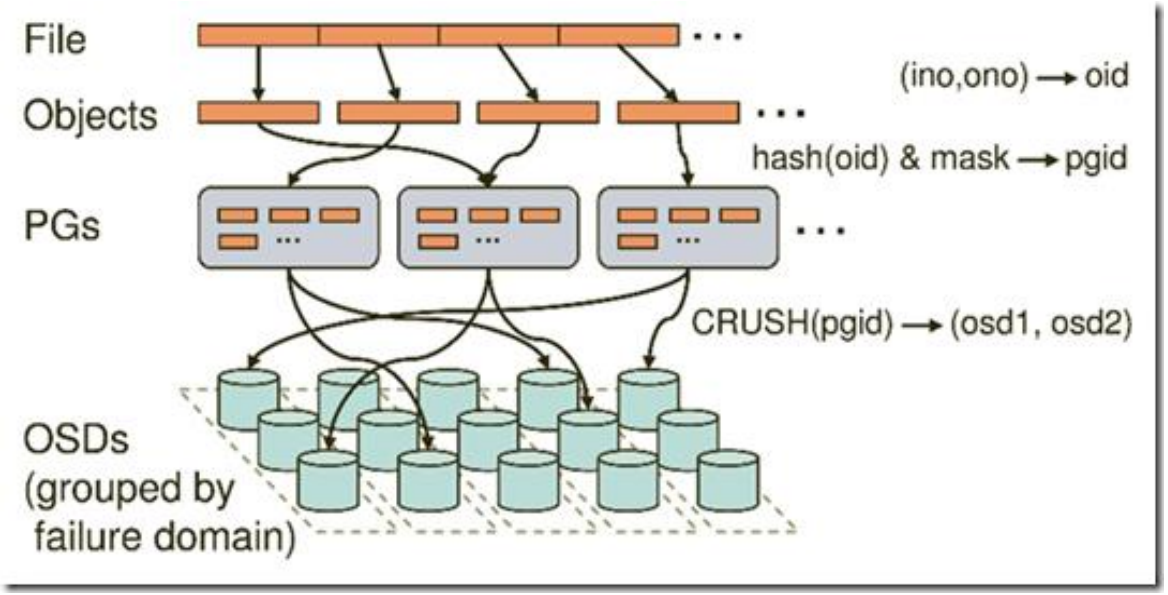
- Détermination de l'OSD en fonction du nom de la ressource



http://www.linux-mag.com/id/7744/

CRUSH

- Exemple avec un pool, une réplication de 2 et une tolérance de panne par serveur.



http://www.linux-mag.com/id/7744/

Protection des données

Depuis la version 0.80 Firefly Ceph propose deux types de protection de données :

- **Réplication** : multiples copies de la source
  - Type de réplication le plus utilisé
  - Réplication minimum par 3 pour réparer automatiquement les données détériorées
  - Reconstruction rapide sans nécessité de calcul de parité
- **Erasure-code** :
  - Les données sont divisées en K fragments, puis combinées avec M fragments de contrôle. L'ensemble est réparties sur différents OSD du cluster.
  - Protection équivalente à RAID6 : K=2, M=2 (Diminution de l'overhead à 50%)
  - Différents algorithmes disponibles : Jerasure, ISA-l, LRC, shec
  - Utilisation du CPU pendant le processus de codage et la reconstruction des données
  - Les morceaux sont répartis sur des OSD de différents serveurs => Nécessite au minimum 4 serveurs pour débuter

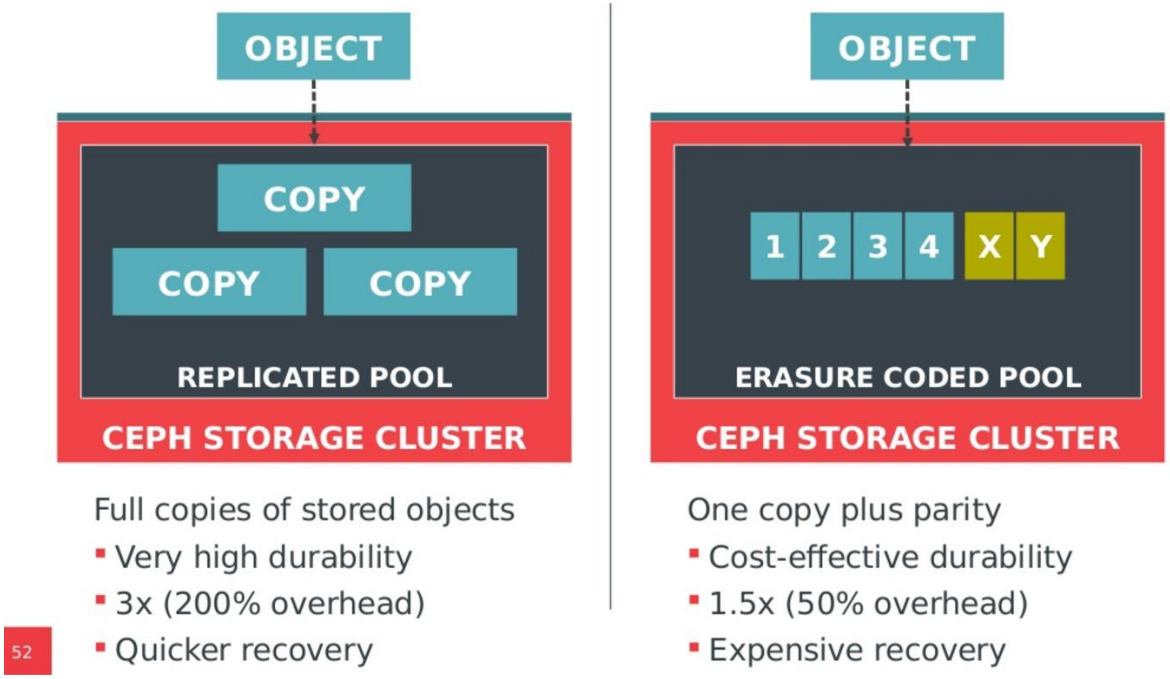
La protection des données est définie pour chaque pool

Les données sont transférées par l'OSD primaire aux réplicats

Réseau dédié à la réplication => meilleures performances

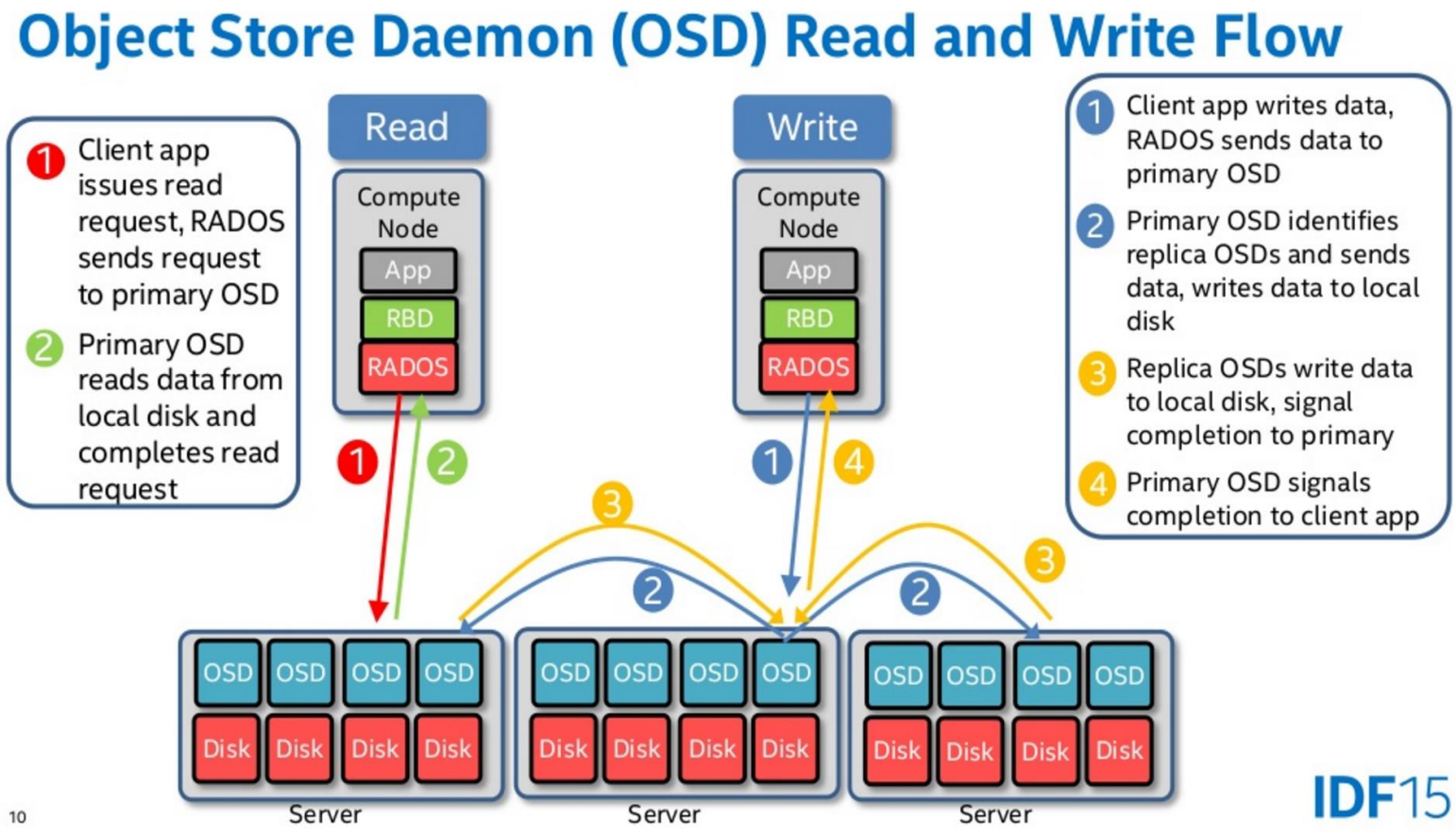
Protection des données

- Différence entre réplication et Erasure coded



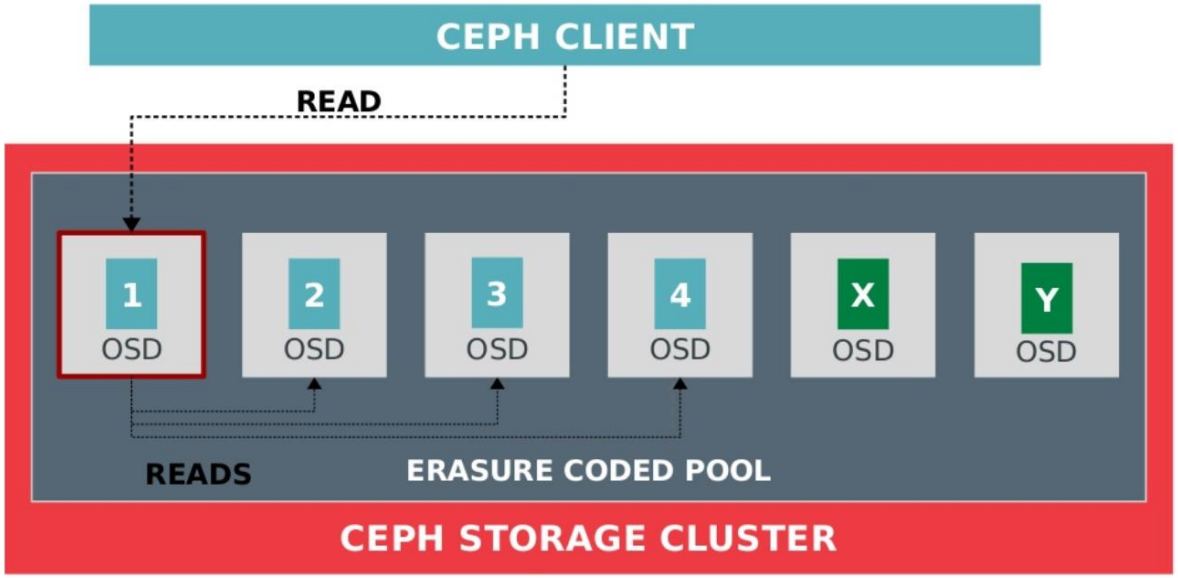
https://www.slideshare.net/sageweil1/20150222-scale-sdc-tiering-and-ec

Protection des données



Protection des données

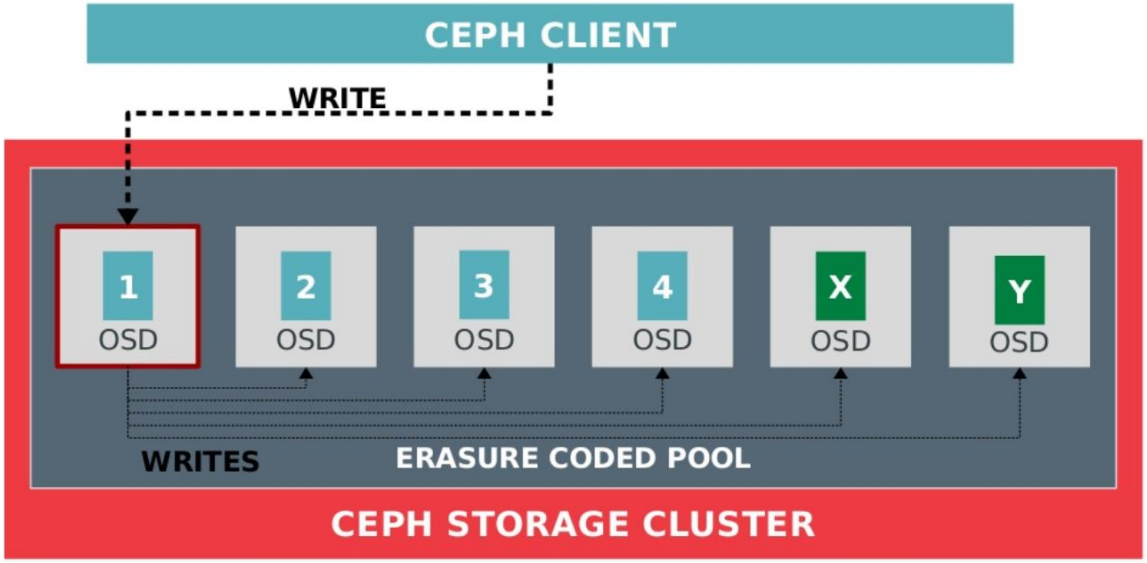
- Cycle de lecture avec l'erasure coding



https://www.slideshare.net/sageweil1/20150222-scale-sdc-tiering-and-ec

Protection des données

- Cycle d'écriture avec l'erasure coding



https://www.slideshare.net/sageweil1/20150222-scale-sdc-tiering-and-ec

Erasure code espace

- K: bloc de données , M: bloc de parités
- % capacité utile = K/(K+M)*100
- Nombre de servers > K+M

K	M=1	M=2	M=3	M=4
1	50	33	25	20
2	67	50	40	33
3	75	60	50	43
4	80	67	57	50
5	83	71	63	56
6	86	75	67	60
7	88	78	70	64
8	89	80	73	67
9	90	82	75	69
10	91	83	77	71

https://docs.ceph.com/en/latest/rados/operations/erasure-code

OSD (Object Storage Device)

Service de stockage des objets, gère la réplication, l'intégrité des données et la récupération si nécessaire

- Les clients CEPH communiquent directement avec les OSD plutôt que par l'intermédiaire d'un serveur centralisé
- Utilisation d'un disque par service
- Evitez l'utilisation de configurations RAID ou de partitionner les disques avec plusieurs OSD
- Différents backend de stockage :
  - FileStore, BlueStore, Seastore
  - Peuvent être mixés dans un même cluster

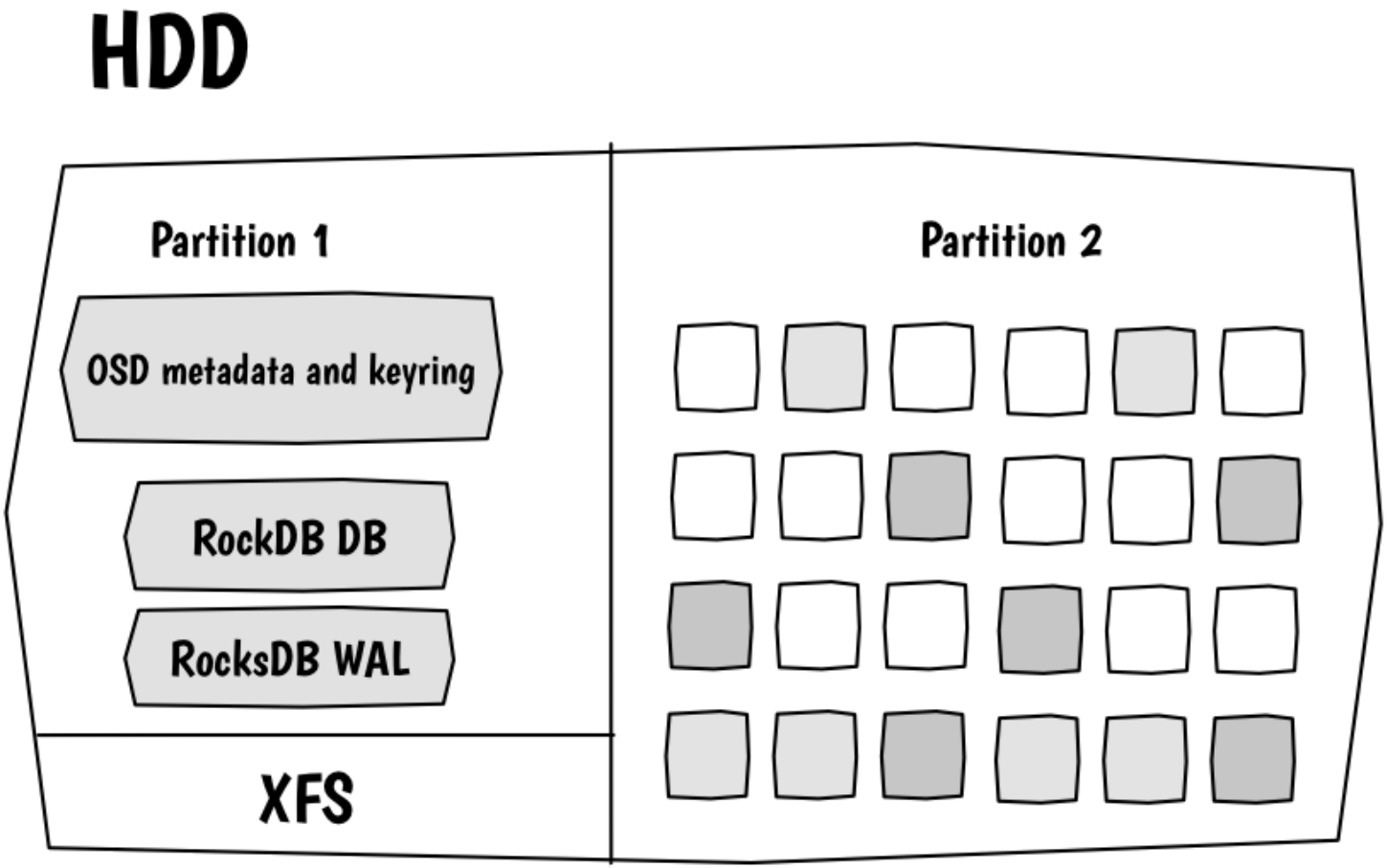
FileStore

- Utilisation par défaut jusqu'à Luminous
- Utilisation en production (bien testé et largement utilisé)
- Utilise un journal et le système de fichiers local (XFS, BTFRS, EXT4, ZFS)
- Écriture synchrone dans le journal, puis en mode asynchrone sur le disque=>Provoque une double écriture !
- Optimisation : possibilité de déplacer le journal sur un disque séparé (SSD) pour augmenter les performances.
- Attention : La perte du disque dédié au journal provoque l'arrêt de tous les OSD concernés. Il est plutôt conseillé d'utiliser le Cache Tiering

BlueStore

- Version stable depuis Luminus. Utilisé par défaut.
- Écrit directement les données sur le disque sans passer par un journal
- Utilisation d'un système de fichier simplifié (bluefs)
- Plus de double écriture
- rockDB pour l'enregistrement des métadonnées (énumération plus rapide)
- Augmentation de la taille des caches par OSD (1GB SATA et 3GB SSD par défaut)
- Gain en vitesse d'écriture x2 sur les disques SATA et plus sur les SSD
- Checksums : vérification de la cohérence à chaque lecture. Permet de diminuer les processus en tâche de fond pour vérifier l'état des PG (scrubing)
- Activation de la compression : lz4, snappy, zlib
- Procédure bluestore-migration pour passer de FileStore à BlueStore http://docs.ceph.com/docs/master/rados/operations/bluestore-migration/

BlueStore



https://www.sebastien-han.fr/blog/2016/03/21/ceph-a-new-store-is-coming/

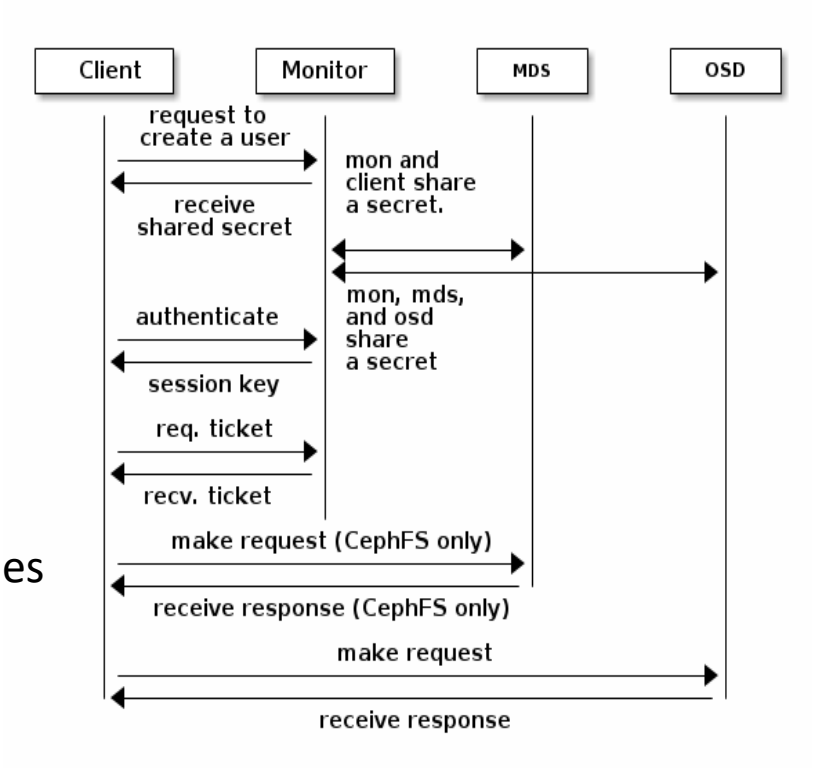
SeaStore

Motivations :

- Evolutions du matériel (Fast NVME, Persistant Memory)
- Alignement sur les segments des NVME
- Pas destiné à être utilisé avec les disques HDD
- Minimiser l'utilisation du CPU
- Programmation mono-thread et IO non-bloquant
- Support du Data Plane Development Kit (DPDK), une librairie pour accélérer les traitements de paquets réseaux en mode utilisateurs

Authentification : CephX

- les MON connaissent
  - les clés de tout le monde
  - les autorisations
- Le client s'authentifie via les MON et demande l'accès aux pools de données
- le MON génère un «secret partagé» limité dans le temps
  RAPPEL : un client écrit uniquement dans l'OSD primaire qui s'occupe de la réplication (ou du dispatch en EC)
- Protection contre l'attaque « man in the middle », et contre les attaques par re-jeu des paquets
- Fonctionnement proche de kerberos
- Limite:
  - N'est pas destinée à l'authentification des humains
  - Gestion des clés clients-services manuelle

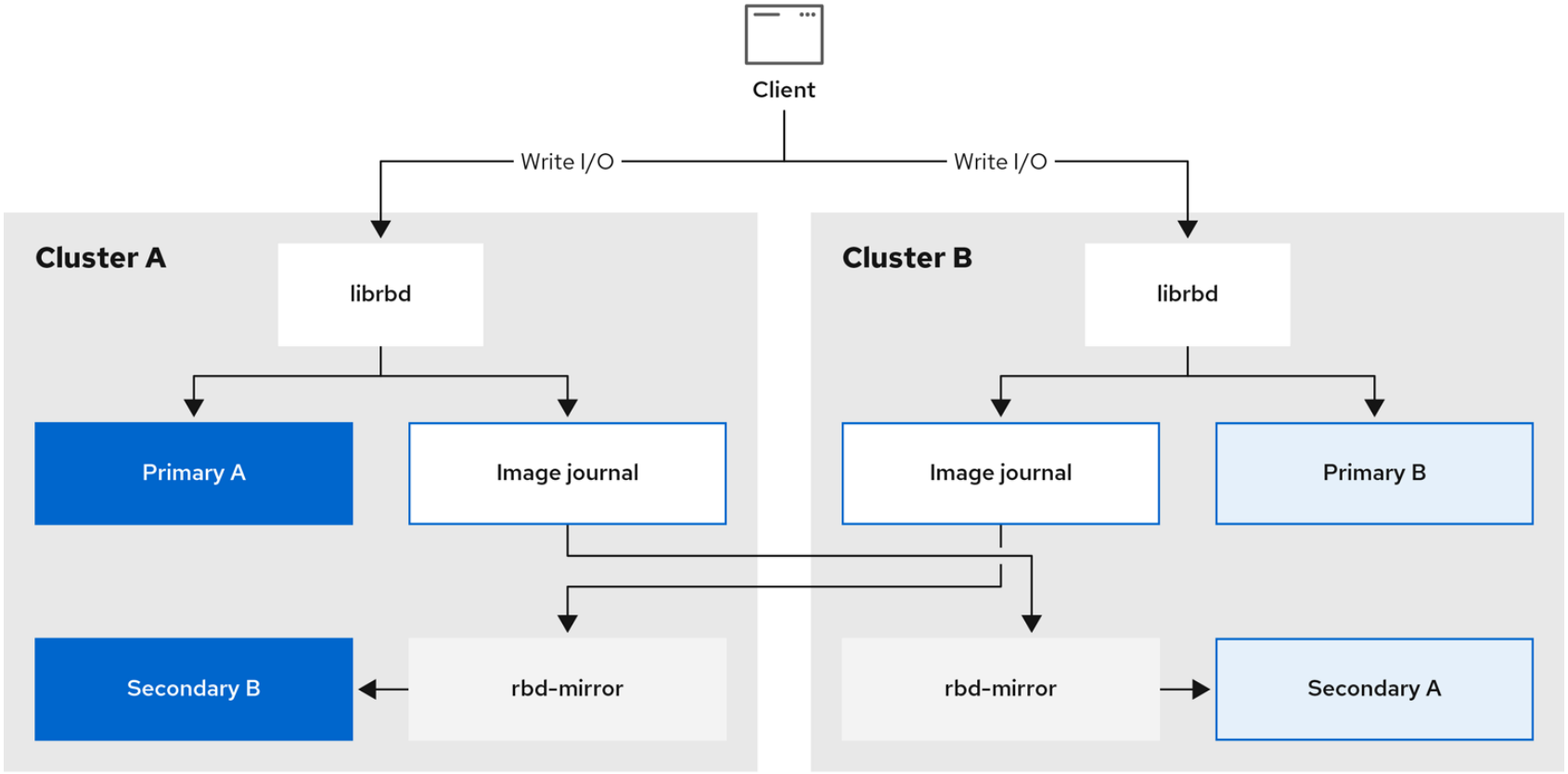


http://docs.ceph.com/docs/firefly/rados/operations/auth-intro/

Ceph mirroring

- Réplication asynchrone
  - Infrastructure différente (stockage ou versions)
  - Protection contre la perte de données ou la défaillance d'un site
  - Diminue le temps de reprise
- Radosgw
  - Par zone (radosgw) ou par bucket (S3-API)
- Rbd-mirroring
  - Réplication active / passive ou active / active
  - Peut synchroniser tout un pool ou sélectionner les images à synchroniser
  - Via un journal ou via des snapshots
- Cepfs-mirroring
  - Par répertoires via l'utilisation des snapshots

Exemple RBD-mirroring



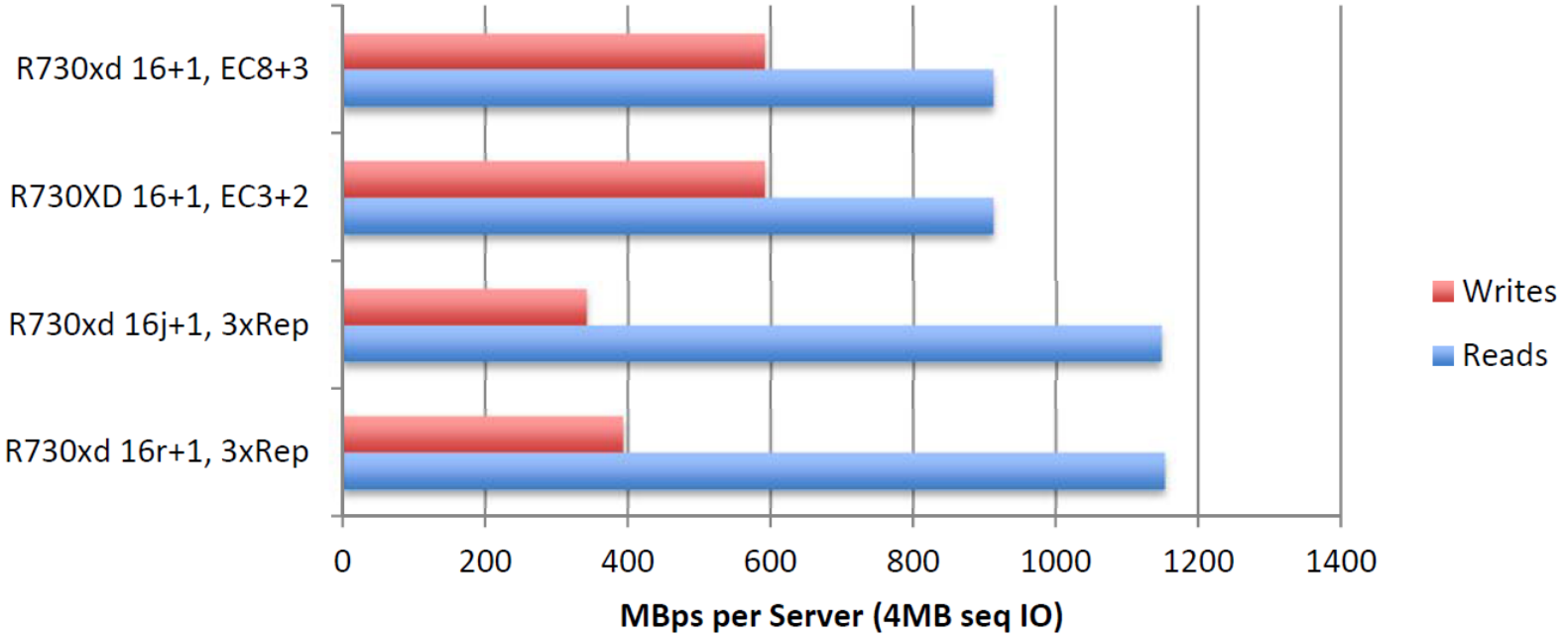
Recommandations pour les matériels

- MON, MSD
  - RAM: 2GB par service , CPU: 2 cœurs
- OSD
  - RAM : 4GB par service , CPU 1 à 2 cœurs
  - Ajouter 20% de RAM à la solution pour lisser les pics.
  - Moins de 20 disques par serveur
  - SSD : utilisé pour les journaux. Ne pas dépasser 5 journaux par SSD
- RadosGW
  - Ram: 64GB , CPU 6 à 8
- Contrôleur de disques
  - Activer la configuration en mode RAID0 pour les disques SAS ou SATA, et JBOD pour les disques SSD
- Réseaux
  - 2*10Gb/s ou 1*25Gb/s
- Documentation
  - http://docs.ceph.com/docs/master/start/hardware-recommendations/
  - Dell PowerEdge Performance and Sizing guide for CEPH Storage http://en.community.dell.com/techcenter/cloud/m/dell_cloud_resources/20442913

Comparaison de performances

Réplication vs. Erasure-coding

- Configuration : disques 4To SAS dans 15 R730xd
- J:JBOD r:RAID0



Dell PowerEdge Performance and Sizing guide for CEPH Storage

GT Ceph @ resinfo

- https://resinfo.org/ceph
- Bonne pratique et entraide
- Liste de diffusion
- Café technique
- Cephlab
- ANF CEPH 2022

Annexes

- CEPH
  http://docs.ceph.com/docs/master/
  https://ceph.io/en/news/blog/
- Ceph pour entreprise
  https://www.youtube.com/watch?v=wLngroWptto
- Dell PowerEdge Performance and Sizing guide for CEPH Storage
  http://en.community.dell.com/techcenter/cloud/m/dell_cloud_resources/20442913
- Retour d'expérience de l'exploitation de CEPH
  https://indico.mathrice.fr/event/143/session/2/contribution/6
- Bluestore performance Scalability (3 vs 5 nodes)
  https://ceph.io/en/news/blog/2019/part-3-rhcs-bluestore-performance-scalability-3-vs-5-nodes/
- Ceph Storage at CERN 2021-12-07 - Dan van der Ster, Pablo Llopis
  https://www.youtube.com/watch?v=pkpTgGsLH_s
- Cephalocon Portland 2022
  https://events.linuxfoundation.org/cephalocon/