

Retour d'expérience de l'exploitation de CEPH au sein de la plateforme SCIGNE

JTech Ceph

28 November 2018

LPNHE

Resinfo

 ceph



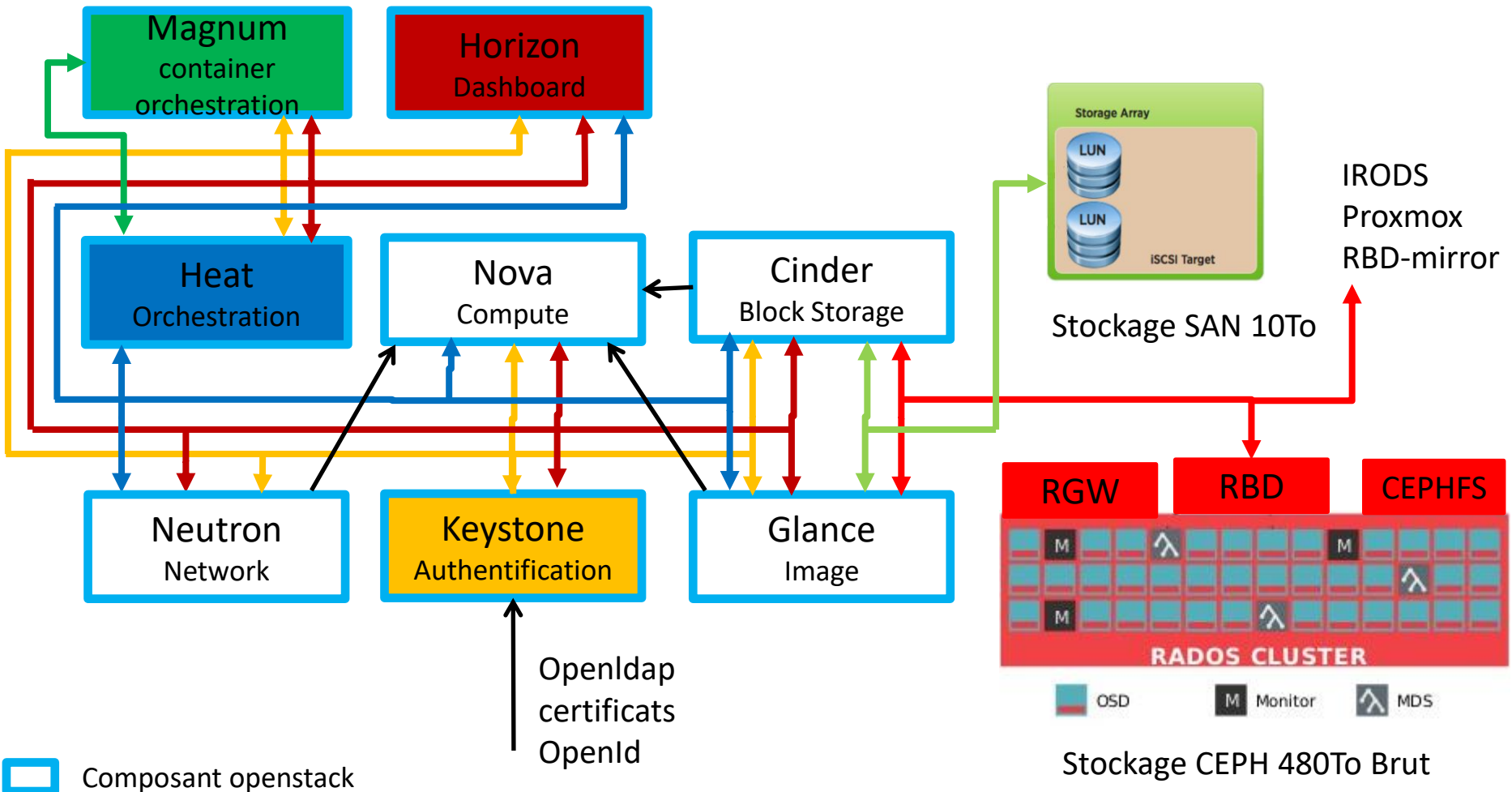
Sommaire

- Présentation de la plateforme SCIGNE
- Test de charge
- Protection des données
- Configuration OpenStack
- Cache Tiering
- Montée de version
- Evolutions

Plateforme SCIGNE

- *Scientific Cloud Infrastructure in Grand Est* : Plateforme labellisée par l'IN2P3 proposant des services de calcul et de stockage pour des données scientifiques.
 - Cluster de calcul HTC (*High-Throughput Computing*) relié à la grille EGI et WLCG
 - Cloud Computing (OpenStack) offrant un service de serveurs de calcul et de conteneurs à la demande relié EGI, IFB
 - Système de stockage et gestion de données massives basé sur iRODS
 - Archivage de données sur bande
- 2017 Evolution stockage
 - Stockage CEPH
 - Version Luminous, capacité : 480To brut
 - 5 Dell R730xd
 - 12 disques 8To SAS
 - Connexion 10Gbs
 - 3 moniteurs M630
 - Connexion 1Gbs
 - Clients
 - 25 Nodes openstack
 - IRODS nodes
- 2018 CEPH Mise en production

Schéma d'infrastructure



Test de charge

- Cluster Ceph Luminous
- 3 clients Transtec
 - Os CentOS 7.4 , 64Go de RAM
 - kernel 4.14 (mainline stable)
 - Connexion 10Gbs
 - CEPH client version Luminous
- Réseaux: 10Gbs
- Outils : rados bench, fio, iftop

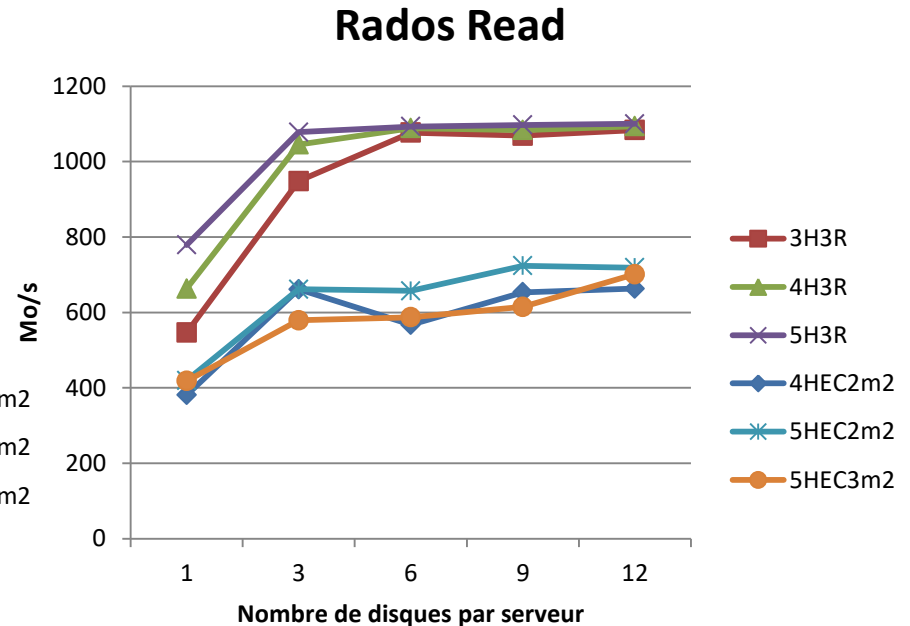
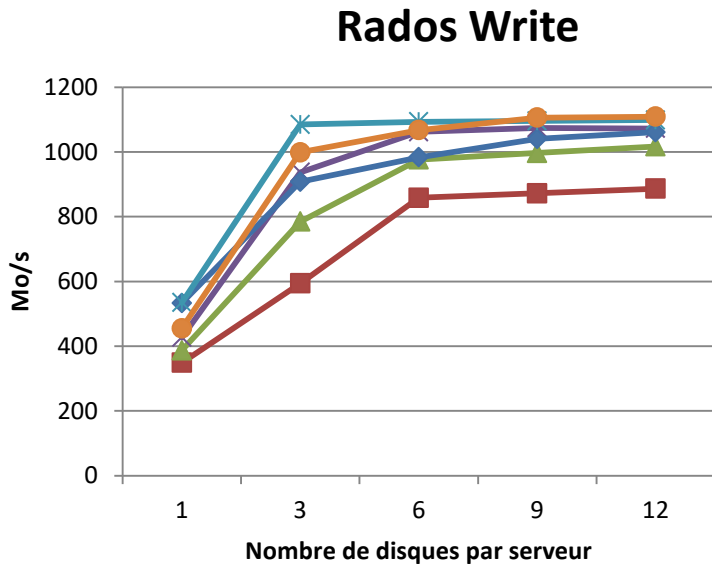
Protection des données

- Réplication
 - 3R => 3 copies, volume net : $1/3$
- Erasurecoding
 - EC3m2j : k=3 m=2 plugin=jerasure
 - EC3m2i : k=3 m=2 plugin=isa
 - EC3m2t : k=3 m=2 plugin=jerasure mode tiering
 - K=3 m=2, volume net : $3/5 = 0.6$
 - K=4 m=2, volume net : $4/6 = 0.66$

Méthode de mesure

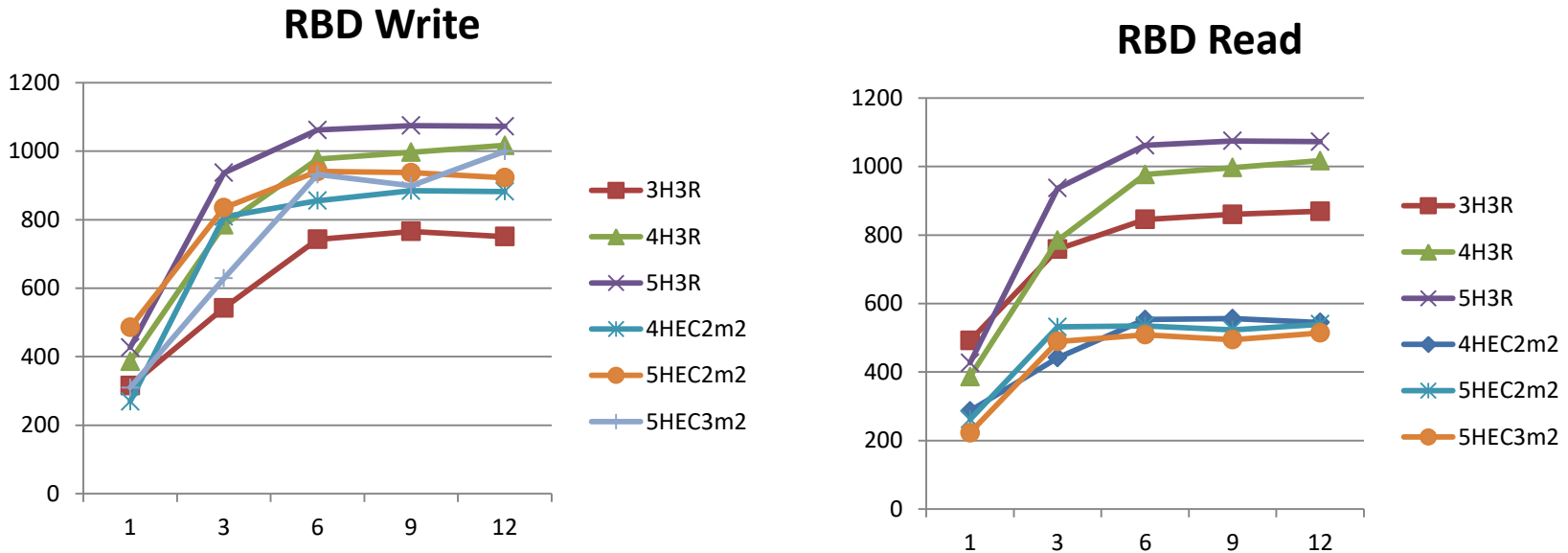
- Création des différents pool Réplication, Erasurecoding
- RADOS:
 - Utilisation de l'outil rados bench fourni par ceph
 - Ecriture et lecture de bloc de 4M et 16 threads pendant 120 secondes
- RBD
 - Création des volumes de 200Mo
 - Monter les volumes sur les clients et formater en Ext4
 - Utilisation de fio 2.8 accès direct
 - Lecture et écriture bloc de 4M et 8 ou 64 threads pendant 120 secondes
 - Destruction des volumes
- CEPHFS
 - Utilisation de fio 2.8 accès direct
 - Lecture et écriture bloc de 4M et 8 ou 64 threads pendant 120 secondes
- Cache client
 - Purge des caches sur le client puis test d'écriture
 - Purge des caches sur le client puis test de lecture
- Benchmark d'un disque 8To SAS
 - READ Seq 4Mo : 218545KB/s
 - WRITE Seq 4Mo : 225283KB/s

Rados Seq Write and Read



```
rados bench -p pool 120 write --no-cleanup -b 4M -t 16  
rados bench -p pool 120 seq -b 4M -t 16
```


RBD Seq Write and Read

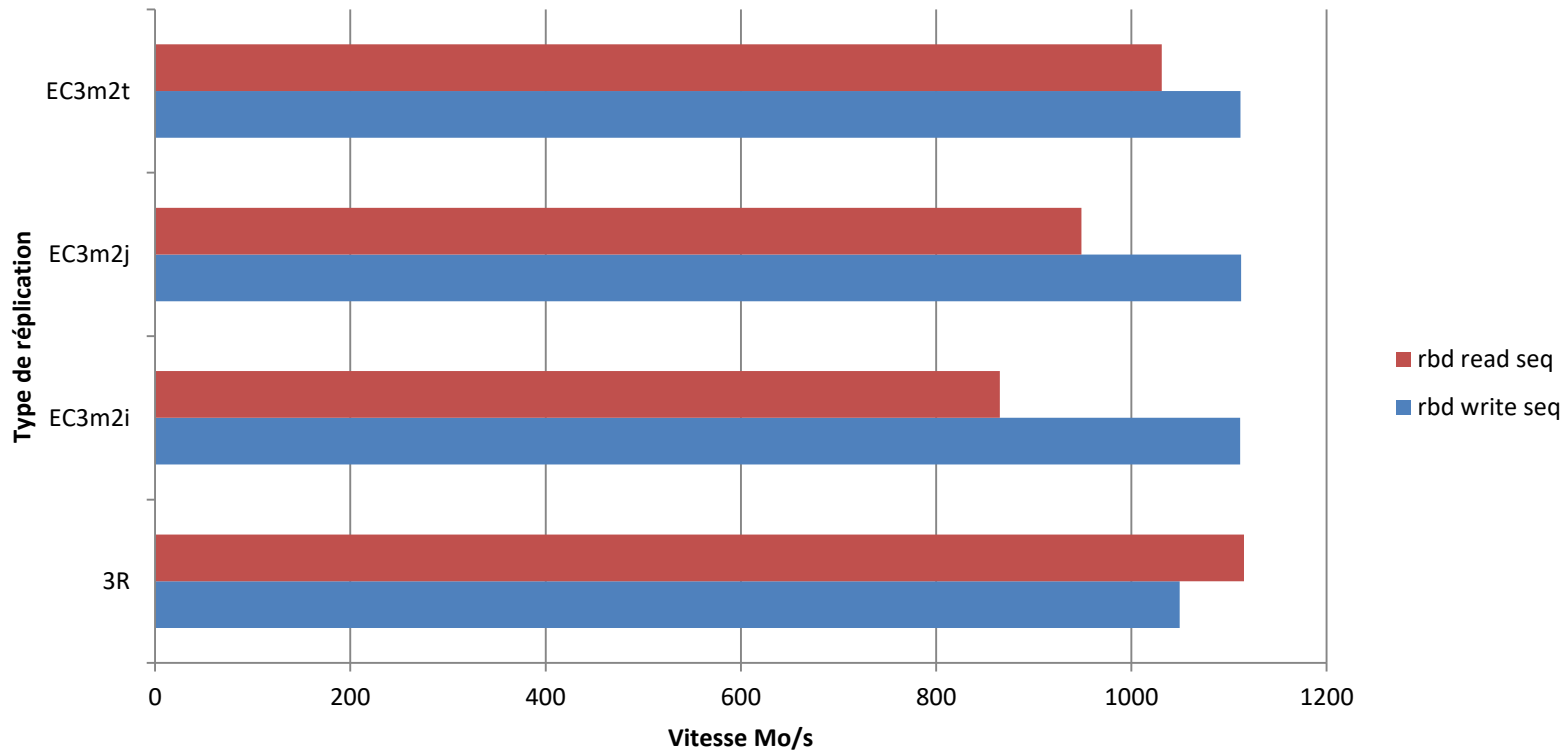


```
fiio --filename=./fioseb --ioengine=libaio --direct=1 --size=100G --bs=4M --iodepth=8  
--numjobs=1 --runtime=120 --ramp_time=20 --rw=write --name=fioseb  
fiio --filename=./fioseb --ioengine=libaio --direct=1 --size=100G --bs=4M --iodepth=8  
--numjobs=1 --runtime=120 --ramp_time=20 --rw=read --name=fioseb
```

RBD Seq Write and Read

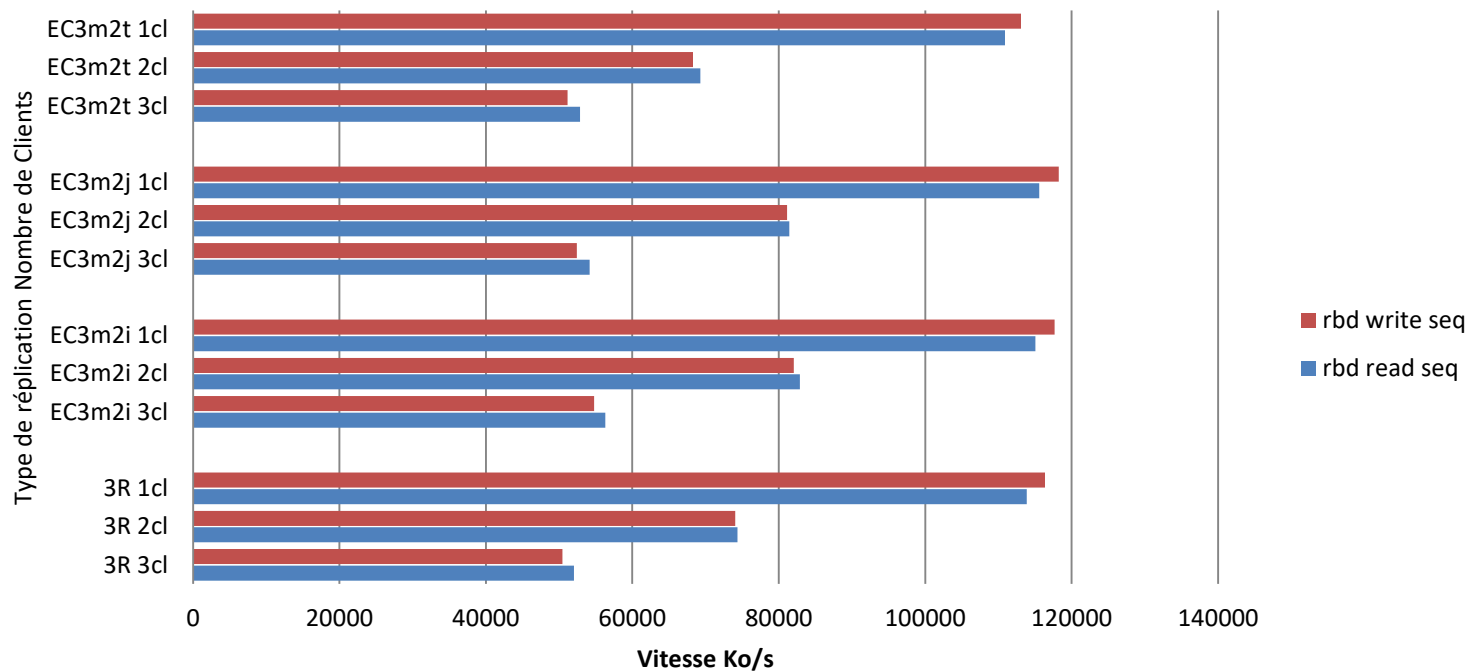
fio idem que précédent

Utilisation de différents pool Réplication, EC isa, jerasure, jerasure avec un cache tiering



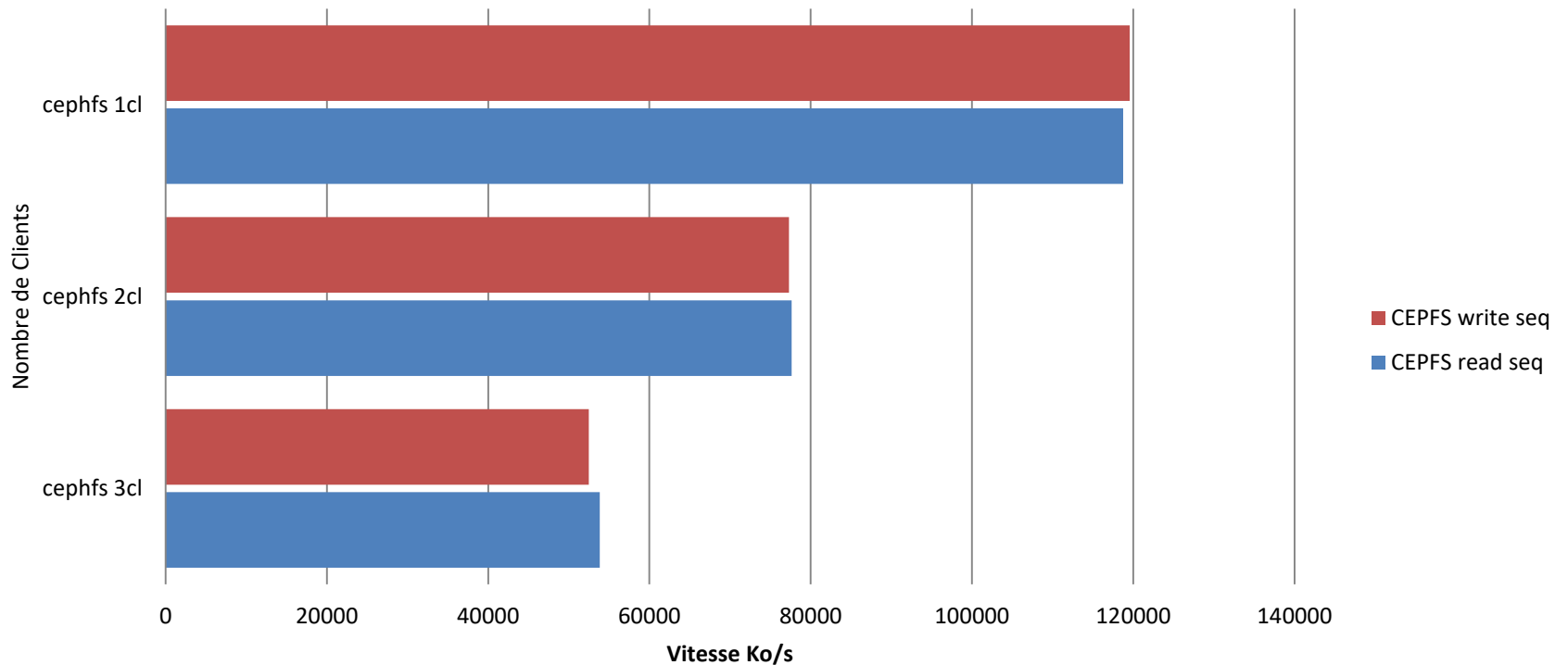
RBD fio Mixed sequential reads and writes

```
fio --filename=./fiosebrw --ioengine=libaio --direct=1 --size=100G --bs=4M --  
iodepth=8 --numjobs=8 --runtime=120 --ramp_time=20 --rw=rw --name=fioseb
```



CEPHFS fio Mixed sequential reads and writes

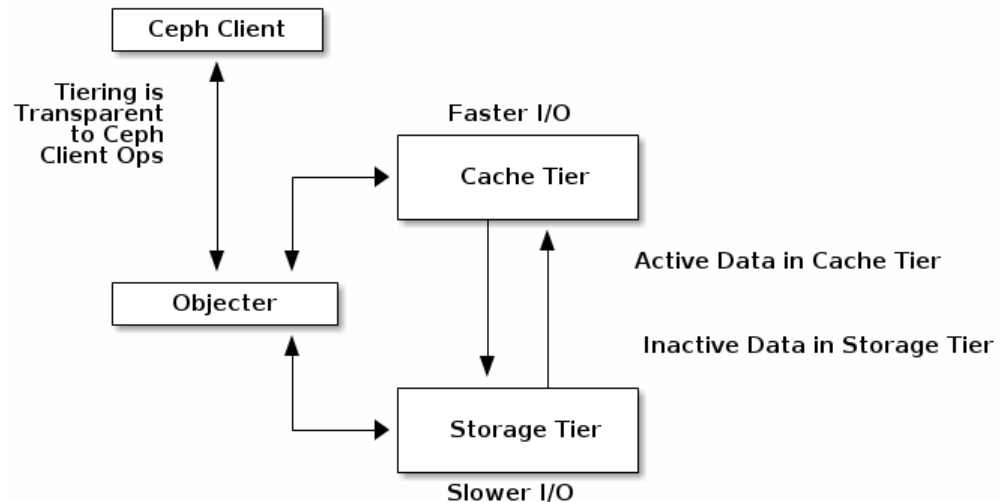
```
fio --filename=./fiosebrw --ioengine=libaio --direct=1 --size=100G --bs=4M --  
iodepth=8 --numjobs=8 --runtime=120 --ramp_time=20 --rw=rw --name=fioseb
```



Cache-Tiering

Le cache-tiering permet de déplacer des données «chaudes» vers les supports haute performance lorsqu'ils deviennent actifs, et les données «froides» vers des supports à faible performance lorsqu'ils ne sont plus actifs.

- Utilisation de disques SSD, NVME
- Les données sont migrées d'un pool à un autre
- Définitions de la politique
 - hit_set_count 2
 - hit_set_period 600 #10 minutes
 - target_max_bytes 100000000 #100MB
 - cache_target_dirty_ratio 0.4
 - cache_target_dirty_high_ratio 0.6
 - cache_target_full_ratio 0.8



<http://docs.ceph.com/docs/master/rados/operations/cache-tiering/>

Temps de reconstruction

- Cluster utilisé à 2,74% 12To sur 427To brut
- 10 pools
- Temps de retour à une situation stable
- Reconstruction automatique des données manquantes

- Perte d'un OSD
 - 54515/4287353 objects misplaced (1.272%)
 - 20Mo/s trafic réseau sur tous les noeuds
 - charge cpu 2%
 - Temps de reconstruction 25 minutes
 - Vitesse de lecture et d'écriture baisse de 8%

Temps de reconstruction

- Perte d'un serveur
 - 792496/4372994 objects degraded (18.123%)
 - 140Mo/s trafic réseau sur tous les nœuds
 - 30Mo/s a 5%
 - Charge cpu 7% (>1% a 5%)
 - Temps de reconstruction 53 minutes
 - Vitesse de lecture et d'écriture baisse de 8%

Configuration OpenStack

- Glance
 - Stockage des images dans un pool RBD
 - Mode RAW, support des snapshots, des clones
 - <https://swamireddy.wordpress.com/2016/04/08/glance-s-quick-uploaddownload-with-ceph/>
- Cinder
 - Stockage des volumes persistants dans un pool RBD 3x
 - Pas d'option du support des pools ErasureCoding
 - Utilisation cache-tiering
 - Autre solution
[client.cinder]
rbd default data pool = volumes_data
https://themeanti.me/technology/2018/08/23/ceph_erasure_openstack.html

Montée de version

- Utilisation d'un dépôt local
- Spanshot 15 jours avant
- Déploiement via QUATTOR
- Documentation bien détaillée
<https://ceph.com/releases/v13-2-0-mimic-released/>
- Faire les mises à jour par ordre
 1. Ceph Monitors
 2. Ceph OSD Daemons
 3. Ceph Metadata Servers
 4. Ceph Object Gateways
 5. Ceph Clients
- Redémarrer les services après les mises à jour
- Attendre une période de stabilisation avant de poursuivre
- Activez les nouvelles fonctions qu'après avoir mis à jour l'ensemble du cluster
- Vérifier les logs

Evolutions Pool full SSD

- Ajout d'un nouveau serveur R740XD
12 disques SSD mixe use de 480Go
- Opération de redéploiement des disques
 - 2 SSD + 10 SAS par serveur
 - Cache tiering pour pool existant
 - 1 pool Full SSD (~ 1.6To en réplication 3x)
- Migration en production
- Bench de la nouvelle solution
- Définir la méthode de Bench

Evolutions des services

- Stockage 80To IRODS
- Réplication sync ou async France-Grilles
- CephFS pour openstack via manila
- Migration vers Mimic
- Support de la compression

Annexes

- SCIGNE : <https://grand-est.fr/>
- Ceph User Survey 2018
<https://ceph.com/wp-content/uploads/2018/07/Ceph-User-Survey-2018-Slides.pdf>
- ANF CNRS 2017 CEPH
<https://groupes.renater.fr/wiki/ceph/public/formation2017>
- CEPH Une solution de stockage distribué Open Source
http://xstra.unistra.fr/lib/exe/fetch.php?media=stockage:presentation_ceph_geiger_20171005.pdf

Supervision

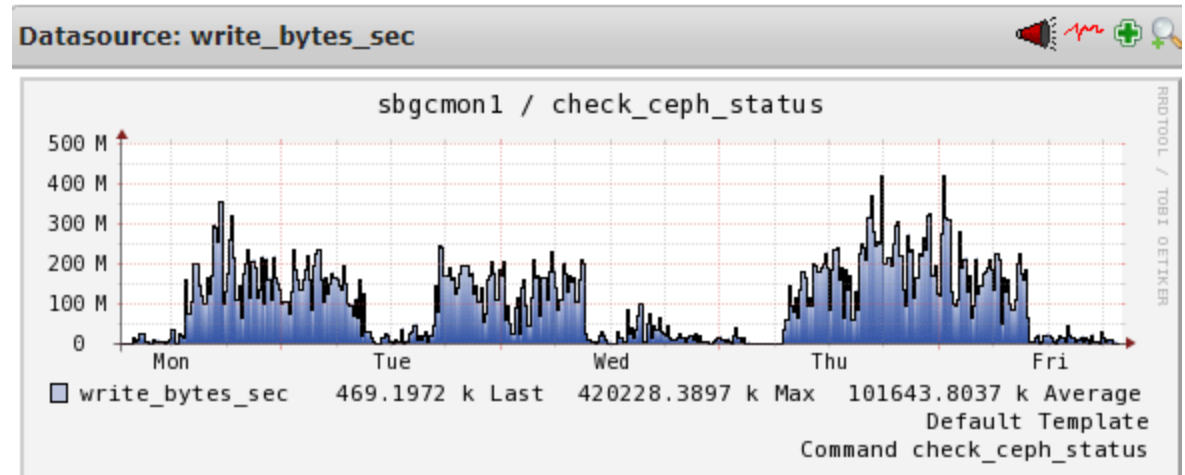
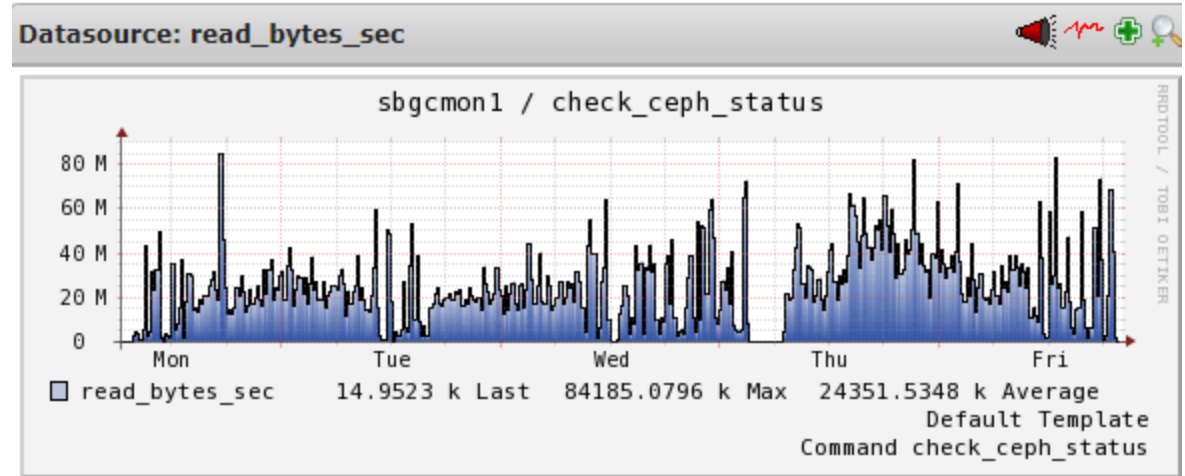
- Commande en ligne
 - ceph health, ceph status, ceph -w, ceph df
 - ceph mon stat, ceph osd stat, ceph osd tree
 - <http://docs.ceph.com/docs/master/rados/operations/monitoring/>
- Dashboard
 - A partir de la version «Luminous»
 - Vue simplifiée des performances Cluster, POOL, OSD
“http://<active mgr host>:7000/“
 - <https://ceph.com/community/new-luminous-dashboard/>

Supervision Nagios

- Sonde Nagios
 - <https://github.com/ceph/ceph-nagios-plugins>
 - Script python pour la surveillance générale
 - check_ceph_health, check_ceph_df, check_ceph_mds, check_ceph_mon, check_ceph_osd, check_ceph_rgw
- Informations sur les IO des pools
 - ceph osd pool stats
- Autres vérifications
 - CPU, RAM, NTP
 - Matériel : Openmanage

Exemple Nagios+PNP

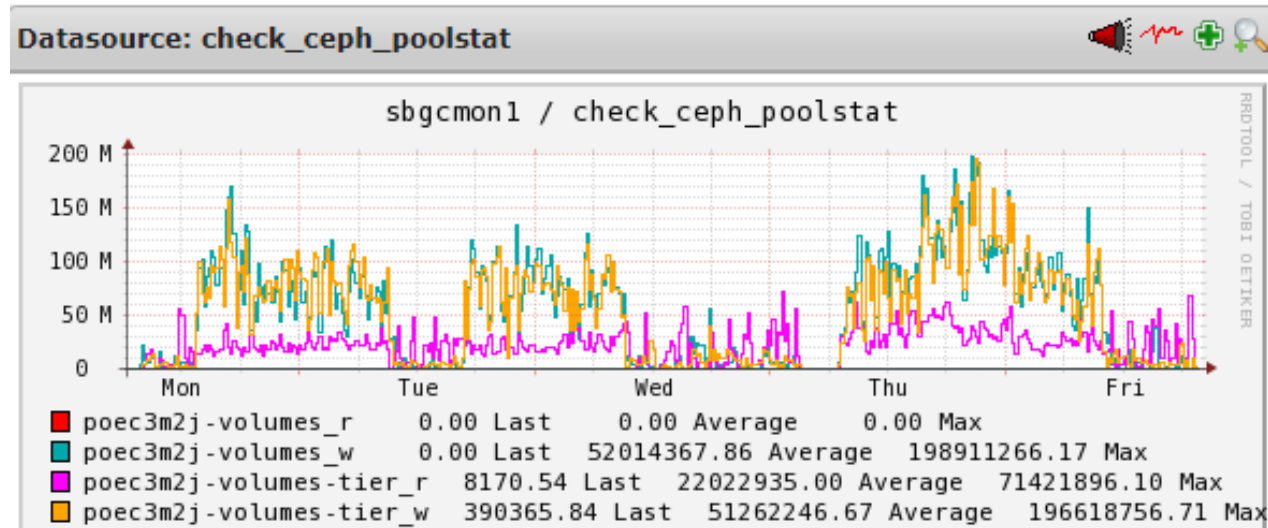
- CEPH_Status
 - read_sec
 - write_sec
 - r_op_sec
 - w_op_sec
 - evict_sec
 - flush_sec
- Ratios write / read
 $101643/24351 = 4.17$



Exemple Nagios+PNP

- PoolStat

- x_volume_r
- x_volume_w



CEPH Manager

- Fournit une surveillance supplémentaire
- Interfaces avec des outils de surveillance ou de gestion externes : Zabbix, Prometheus, Influx, Telemetry, Telegraf, RESTful
- Exemple Prometheus et Grafana
 - ceph mgr module enable prometheus
 - Cluster : Status, Capacity, IOPS, R/W ops,
 - OSD : IN, OUT, PGs, latency,

Exemple de tableau de bord

