BeeGFS dopé aux SSDs

David Sanchez - LCPQ

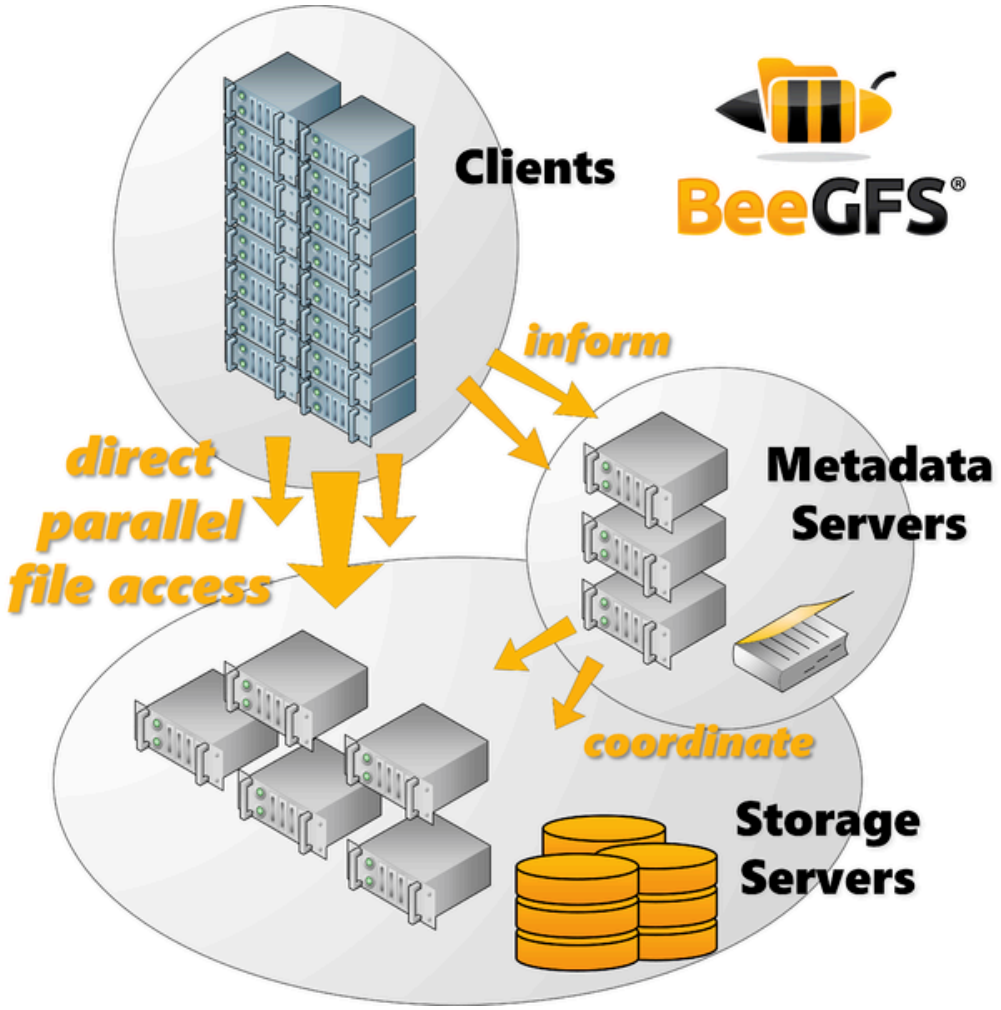
Capitoul - 29 février 2024

Historique stockage distribué au LCPQ

- 1 cluster de calcul doté de Lustre (2009 - 10To) : forte attache au kernel, quelques crashs système...
- 1 cluster avec BeeGFS doté de BeeGFS (2015 - 30To) : super remplaçant, **0 crash** système sur la période 2015 - 2023 !
- 1 nouveau cluster (2023 - 60To) : toujours avec BeeGFS, objectif : améliorer les performances

C'est quoi BeeGFS ?

- Un système de fichier parallèle
- Le code source ouvert
- 4 services de base
  - métadonnés
  - stockage
  - clients
  - management
  - monitoring (facultatif)
- Version actuelle : 7.4.1



Architecture BeeGFS au LCPQ

- 3 baies
- 1 switch Aruba 8360 par baie : 32 ports 25G + 4 ports 100G
- 46 noeuds (Dell & HPe - openSUSE Leap)
- 1 serveur management / metadata (HPe - Debian 12)
- 2 serveurs de stockage (HPe - Debian 12)

Specs serveurs BeeGFS - mgmt & meta

Métadonnées : 1x HPe DL325 gen10+ v2

- AMD Epyc 7313P : 16 coeurs phy @ 3GHz, turbo @ 3.7GHz, TDP 155-180W
- 128Go de RAM
- 2 SSDs système (mdadm / raid 1)
- 2 SSDs 800Go SAS Write Intensive pour /beegfs-meta (mdadm / raid 1 - ext4)
- 1 cartes raid, disques en JBOD
- 1 carte x 4 ports 25G

Specs serveurs BeeGFS - stockage

Stockage : 2x HPe DL345 gen10+

- AMD Epyc 7313P : 16 coeurs phy @ 3GHz, turbo @ 3.7GHz, TDP 155-180W
- 256Go de RAM
- 2 SSDs système (mdadm / raid 1)
- 5 SSDs de 3.8To Value SAS Mix Use
- 14 HDDs de 2.4To @ 10ktpm
- 2 cartes raid, disques en JBOD
- 2 cartes x 2 ports 25G

Mixer SSD et HDD dans BeeGFS

- 2 silos dans notre volume BeeGFS : fast pour SSD, slow pour HDD.
  - nécessite licence entreprise (€€€€€)

Ou alors…

- On utilise les SSDs comme cache des HDDs !
  - ZFS (OpenZFS)
  - bcacheFS (stable kernel linux 6.7, janvier 2024)
  - LVM ?!

LVM

On peut utiliser des disques comme cache pour des volumes LVM.

On peut ainsi utiliser un SSD pour 2-3 HDDs afin d'encaisser les IOPS !

Il existe 2 implémentations :

- dm-cache (lecture/écriture, création/activation/destruction en online)
- dm-writecache (écriture seulement, création/activation/destruction offline)

LVM - suite

2 modes de fonctionnement :

- writeback : l'écriture est validée quand elle est faite sur le disque de cache seulement
- writethrough : l'écriture est validée quand elle est faite sur le disque de cache ET le disque final

2 types de cache :

- cachepool : 2 volumes pour le cache (cache + metadata cache)
- cache volume : cache + metadata cache dans un seul volume

LVM - cli

/dev/sdc : SSD

/dev/sd{d,e,f} : HDD

```
# lvmisation dse disques
pvcreate /dev/sdc /dev/sdd /dev/sde /dev/sdf

# création d'un pool contenant les 4 disques
vgcreate apool /dev/sdc /dev/sdd /dev/sde /dev/sdf

# création d'un volume par HDD
lvcreate -n lv_aphy1 -l 100%PVS apool /dev/sdd
lvcreate -n lv_aphy2 -l 100%PVS apool /dev/sde
lvcreate -n lv_aphy3 -l 100%PVS apool /dev/sdf

# création d'un volume cache sur SSD (sdc) et rattachement à un volume HDD
lvcreate -L 1.1T -n aphy1_cache apool /dev/sdc
lvconvert -y --type cache --cachemode writeback --chunksize 2M --cachevol aphy1_cache apool/lv_aphy1

lvcreate -L 1.1T -n aphy2_cache apool /dev/sdc
lvconvert -y --type cache --cachemode writeback --chunksize 2M --cachevol aphy2_cache apool/lv_aphy2

lvcreate -L 1.1T -n aphy3_cache apool /dev/sdc
lvconvert -y --type cache --cachemode writeback --chunksize 2M --cachevol aphy3_cache apool/lv_aphy3

mkfs.ext4 /dev/mapper/apool-lv_aphy1 && mount /dev/mapper/apool-lv_aphy1 /beegfs/aphy1
```

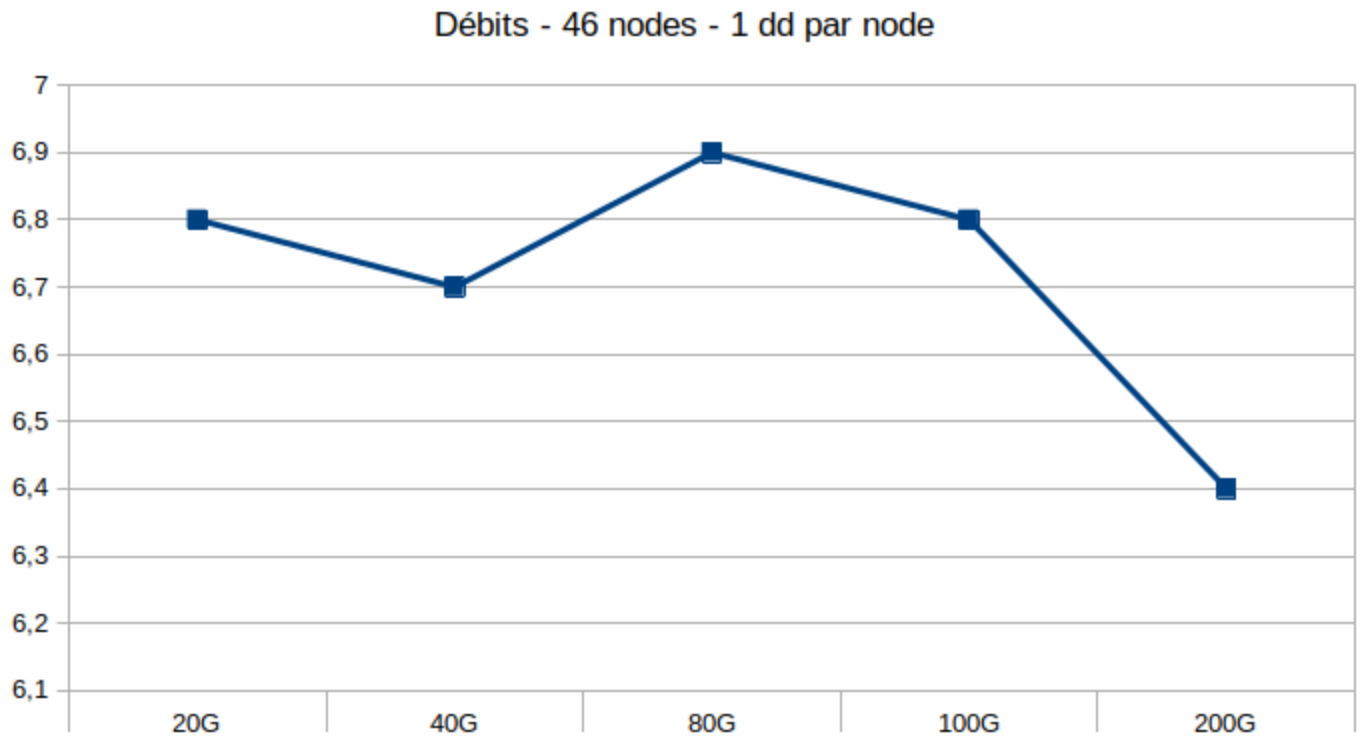
LVM - cli 2

```
oss2:~/bin# ./lvmcache-statistics.sh /dev/mapper/apool-lv_aph1
------------------------------------
LVM Cache report of /dev/mapper/apool-lv_aphy1
------------------------------------
- Cache Usage: 69.7% - Metadata Usage: 21.1%
- Read Hit Rate: 99.0% - Write Hit Rate: 94.0%
- Demotions/Promotions/Dirty: 0/368876/20
- Features in use: metadata2 writeback no_discard_passdown

oss2:~/bin# ./lvmcache-statistics.sh /dev/mapper/apool-lv_aph2
------------------------------------
LVM Cache report of /dev/mapper/apool-lv_aphy2
------------------------------------
- Cache Usage: 63.5% - Metadata Usage: 21.1%
- Read Hit Rate: 98.5% - Write Hit Rate: 93.7%
- Demotions/Promotions/Dirty: 0/328212/1
- Features in use: metadata2 writeback no_discard_passdown

oss2:~/bin# ./lvmcache-statistics.sh /dev/mapper/apool-lv_aph3
------------------------------------
LVM Cache report of /dev/mapper/apool-lv_aphy3
------------------------------------
- Cache Usage: 74.2% - Metadata Usage: 21.1%
- Read Hit Rate: 99.2% - Write Hit Rate: 95.1%
- Demotions/Promotions/Dirty: 0/397081/19
- Features in use: metadata2 writeback no_discard_passdown
```

Performances - 1

Disque	Débit	Opération
SSD (cache)	1.3Go/s	1dd - 40Go
SSD (hors cache)	630Mo/s	1dd - 40Go
HDD	200Mo/s	1dd - 40Go
Rack 1 - 14 noeuds	5.4Go/s	1dd - 20Go
Rack 2 - 19 noeuds	5.1Go/s	1dd - 20Go
Rack 3 - 13 noeuds	5.2Go/s	1dd - 20Go

Performances - 2



IO500

Classement des systèmes de stockage les plus rapides/efficaces au monde.

Test sur 10 machines, 3 processus par machine :

- bande passante : 1.22 GiB/s
- IOPS : 36.16k/s

Système de stockage du LCPQ : https://io500.org/submissions/view/686

Conclusion

Ça marche très bien :-)

On s'est détaché des cartes raid "intelligentes", facilité à changer de constructeur si besoin.

Amélioration : ajouter RDMA / RoCE

Merci pour votre attention !

Questions ?

Sources

https://blog.delouw.ch/2020/01/29/using-lvm-cache-for-storage-tiering/

https://github.com/sphericale/lvmcache-statistics/blob/master/lvmcache-statistics.sh

https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html/configuring_and_managing_logical_volumes/enabling-caching-to-improve-logical-volume-performance_configuring-and-managing-logical-volumes

https://lukas.zapletalovi.com/posts/2019/lvm-cache-in-six-easy-steps/