

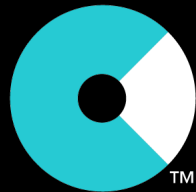
Introducing Pool Quotas

LUG 2019

Sergey Cheremencev, Nathan Rutman, and Cory Spitz

 spitzcor@cray.com

 [@spitzcor](https://twitter.com/spitzcor)



CRAY[®]

ABSTRACT

With the deployment of heterogeneous clusters containing a mix of flash OSTs and disk OSTs, administrators may need to restrict use of higher-performance OSTs on flash. Lustre's OST Pools feature enables the grouping of similar OSTs into performance tiers and the assignment of file layouts into these tiers. However, this feature does not support limits on the usage of more desirable / more expensive / smaller capacity tiers.

Quota controls are the practical solution to impose administrative limits on a cluster's resources. However, currently-available Lustre quotas are limited to filesystem-wide limits on a per-user, per-group or per-project basis.

In this presentation, we describe a new design for pool quotas that extends Lustre's capability to limit allocations within pools. We outline the feature's design and introduce a strategy of using multiple quotas within a cluster.



Why Tiers Within Lustre?

- Tiered storage is rapidly becoming more prevalent
 - 100% flash is not economical... yet
 - Lang's Law: "the more tiers, the more tears"
 - Conclusion: still need a capacity tier, flash will augment disk
- Lustre performs on flash – therefore, flash should exist in the Lustre namespace
 - Lustre can now efficiently perform small I/O; this wasn't always the case
 - Bandwidth at server of NVMe flash + network exceeds disk + network
- Result: heterogeneous storage and tiers within Lustre clusters

OST Pools for Tiers?

- OST Pools already exist to manage groupings of OSTs
- OST Pools **do not** provide administrative controls!
 - OST Pools are *only a convenience* to concisely describe file layouts on OSTs
- OST Pools != tiers, we want permissions and controlled access to tiers
 - However, we don't want or need to invent a new concept or construct
 - Let's try to use OST Pools to manage tiers

```
mgs# lctl pool_new lustre.flash
```

```
mgs# lctl pool_add lustre.flash OST[0-10]
```

```
client$ lfs setstripe -pool flash myflashdir
```

```
client$ dd if=/dev/zero of=./myflashdir/myfile bs=1g count=100
```

```
dd: error writing './myflashdir/myfile': No space left on device 🙄
```

Why Pool Quotas?

- It is too easy to fill a flash OST
 - Flash tiers are in high demand
 - Generally smaller capacity
 - Traditional quotas don't help
 - Sub-directory mounts don't help
 - PFL helps and Self Extending Layouts help yet more
 - But, PFL+SEL doesn't completely solve the problem
- Lustre has no method to limit the usage of desirable OSTs/tiers
- Quotas are the practical solution to impose administrative controls
- Problem: quotas are limited to filesystem-wide limits on a per-user/group/project basis
- Solution: Pool Quotas; that is, quotas for OST Pools



Requirements for Pool Quotas

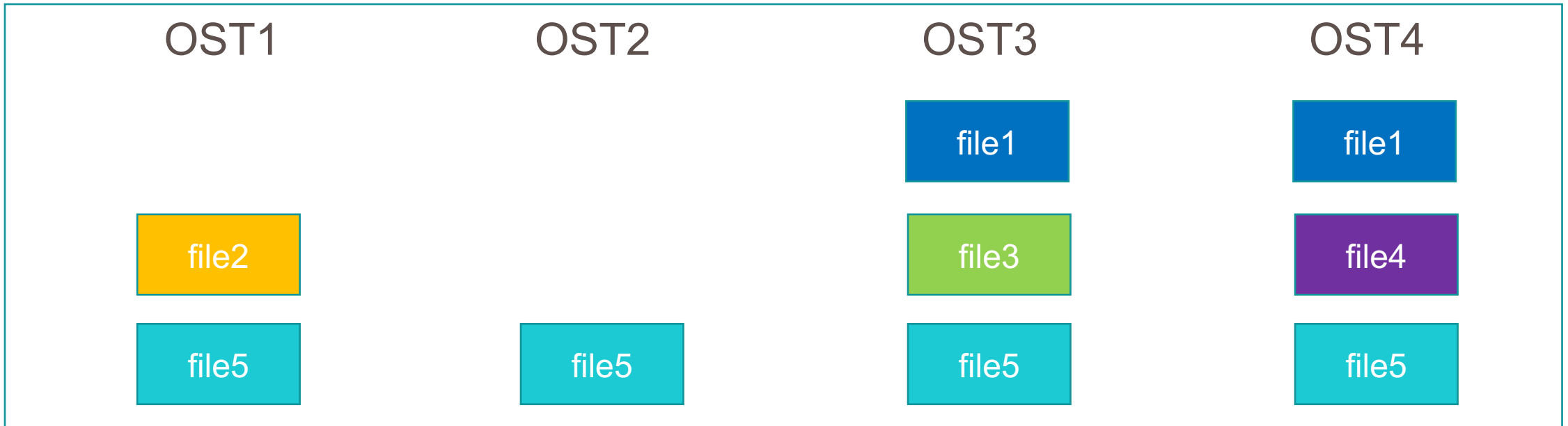
- Basics: provide per-pool user/group/project capacity quotas
 - File quotas don't make sense for OST Pools
- Changes in pool definitions should dynamically affect the remaining pool quotas
 - OST may be added or removed from a pool at any time
- OST may be part of multiple pools, each with different pool quotas set
- Quota limit should be the minimum of all applicable limits
 - Of multiple OST Pools
 - Includes filesystem-wide quotas too
- Leverage existing quotas administrative interfaces, e.g.:
 - `lfs setquota -u bob --pool flash --block-hardlimit=2T /mnt/lustre`

Quotas vs. Pool Quotas

Create a pool of OST3 & OST4, add pool quota

FS Quotas apply

Pool Quotas apply



Implementation Approach

- Start with the existing quotas implementation
- OSTs already request quota grants from quota master
- Current grant is determined on a filesystem-wide basis
- Aggregate resource used is tracked by each OST
- Teach the quota master to consider OSTs within pool definitions!
 - Grant requests to each OST must satisfy all limits
 - Simply the $\min()$ of any fs-wide or per-pool limits

Additional Details

- Design approach means code changes only affect the MDS!
 - Files and objects do not "belong to" pools; OSTs belong to pools
 - No need to store pool information with each object/file
 - OSTs don't need to understand which pool(s) they may be part of
 - Data written to an OST without a pool layout is still accounted for
- Easily cope with changes to pool definition
 - Pool definition changes can cause pool quota limit to be exceeded
 - In which case no new quota will be granted, but existing grants can continue to be used

Quota Entries Example With Overlapping Pools

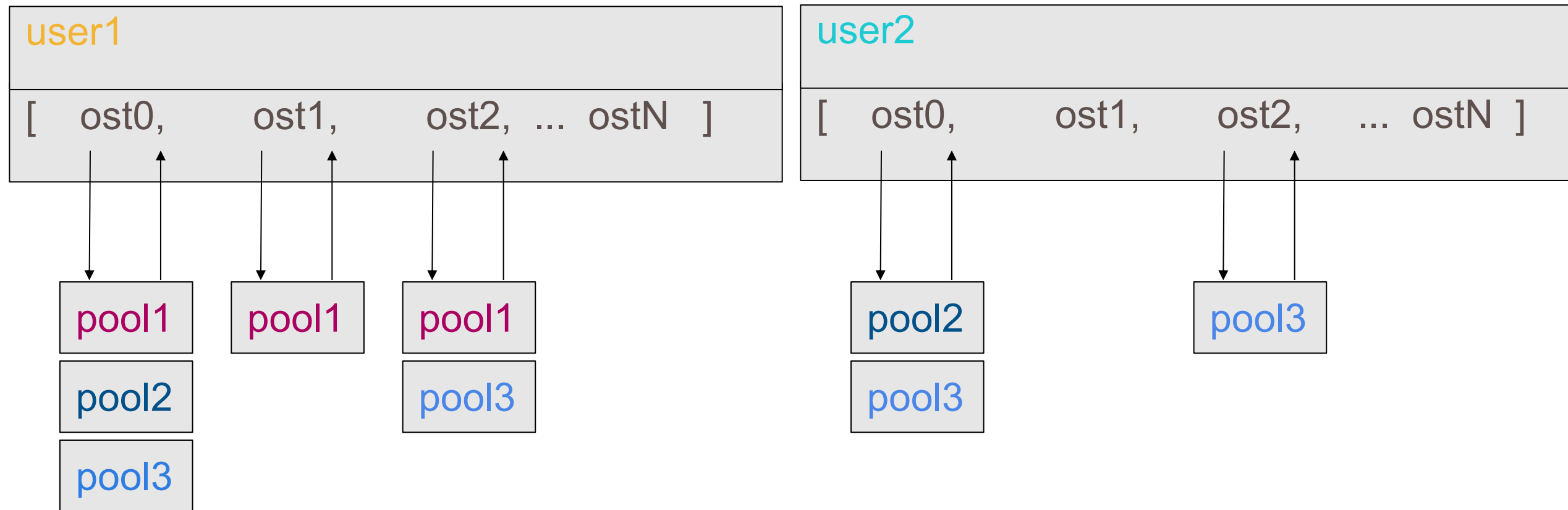
pool1: ost0, ost1, ost2

pool2: ost0

pool3: ost0, ost2

user1: pool1, pool2, pool3

user2: pool2, pool3



Administrator's Example Usage

- Begin with a default Pool Quota

```
# lfs setquota --pool flash -U --block-hardlimit 4M /mnt/lustre
```

- Then create or change Pool Quota as needed

```
# lfs setquota -u bob --pool flash --block-hardlimit 2g /mnt/lustre
```

```
# lfs setquota -p proj --pool flash --block-hardlimit 20g /mnt/lustre
```

```
# lfs setquota -g dev --pool flash --block-hardlimit 200g /mnt/lustre
```

- Disable Pool Quota enforcement

```
# lfs quotaoff --pool flash /mnt/lustre
```

User's Example Usage

- Review quota usage

```
$ lfs quota -h -u bob /mnt/lustre
```

```
Disk quotas for user bob(uid 1579):
```

Filesystem	used	soft	hard	grace	files	soft	hard	grace
/mnt/lustre/		397	0	0	-	3	0	0
flash		100k	1M	...				
disk		510M	1G	...				

- Upon EDQUOT user could have hit filesystem-wide or per-pool limit
- Remember, quota master grants min() of all limits
 - Which quota limit did you exceed?
 - As is today, but Pool Quotas add an extra dimension
 - All quota limits are available in report for inspection

Things to Know

- OSTs may belong to multiple pools and pool membership can change
 - Objects count toward all pool limits and changes may push existing grant over limit
 - Approaching limit on one pool may slow performance for others (as qunit decreases)
- Stripe allocator can allocate to OSTs without enough quota to write
- “No Pool Quota” not possible; Pool Quota of zero means “no limit”
- There are no MDT Pools, therefore Data on MDT space is not limited
- EDQUOT vs. ENOSPC
 - Even with Pool Quotas it is all too easy to get ENOSPC long before EDQUOT
 - Quotas are *not* space reservations

Quotas vs. Pool Quotas Review

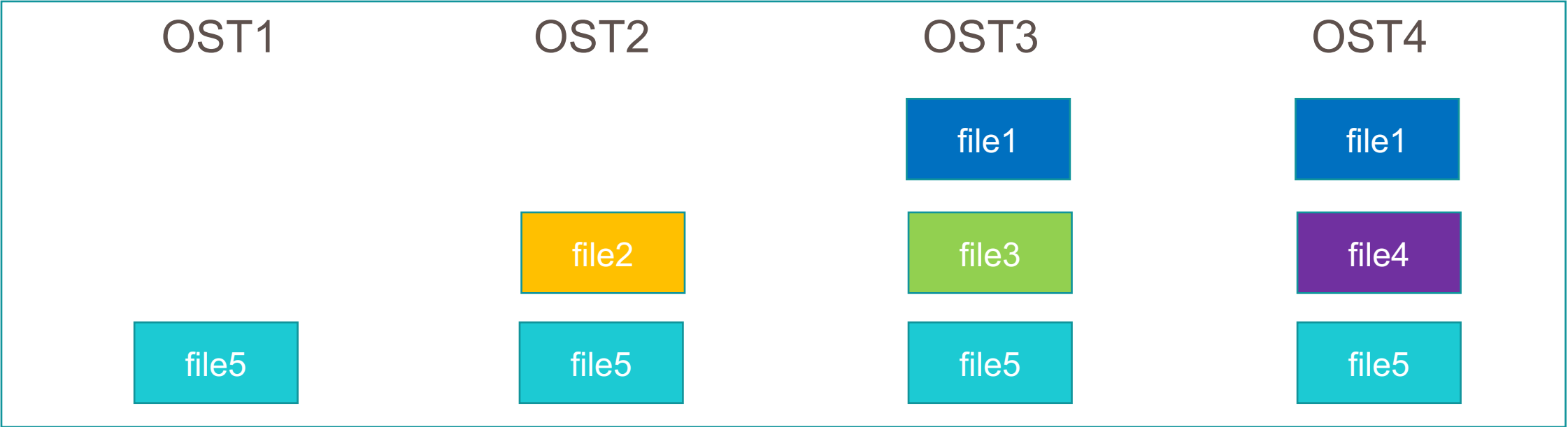


Create a pool of OST 2/3,
OST 3/4, add pool quota

FS Quotas apply

Pool Quotas apply

Pool Quotas apply



Status

- <https://jira.whamcloud.com/browse/LU-11023>
- Targeting 2.14
- [HLD](#) posted and available for review
 - Most (all?) decisions are complete
- Code posted (so far):
 - <https://review.whamcloud.com/34667>
 - <https://review.whamcloud.com/34389>
 - More coming soon
- Test plan: in early stages
 - Luckily, most of the existing quota tests can be applied
- Next steps: ratify HLD, finish implementation, propose/code/execute tests

Future Tiering Enhancements

- Pool Quotas + Self Extending Layouts (SEL) = easy tier administration
- Replicas scheduled/controlled from third-party tiering engines with ties to WLMs
 - Enabled by Lustre-HSM enhancements
- Release FLR mirror upon tier pressure
- Migrate only the minimum components (PFL)
- True space reservations
- Add capability to set default quota to '0'
- MDT Pools + Pool Quotas
 - For inodes in DNE pools
 - For space allocation in DoM pools

SAFE HARBOR STATEMENT


This presentation may contain forward-looking statements that are based on our current expectations. Forward looking statements may include statements about our financial guidance and expected operating results, our opportunities and future potential, our product development and new product introduction plans, our ability to expand and penetrate our addressable markets and other statements that are not historical facts.

These statements are only predictions and actual results may materially vary from those projected. Please refer to Cray's documents filed with the SEC from time to time concerning factors that could affect the Company and these forward-looking statements.



THANK YOU

QUESTIONS?

 spitzcor@cray.com

 [@spitzcor](https://twitter.com/spitzcor)



[cray.com](https://www.cray.com) 

[@cray_inc](https://twitter.com/cray_inc) 

[linkedin.com/company/cray-inc/](https://www.linkedin.com/company/cray-inc/) 