Un-scratching Lustre

LUG 2019 / Houston

Cameron Harr
(Lustre Ops & Stuff, LLNL)

May 17, 2019

LLNL-PRES-773414
This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

Lawrence Livermore National Laboratory

Abstract

File systems, especially complex, parallel ones, take many years to mature. Given the risky behavior inherent in adolescence, these young file systems are often used as "scratch" file systems, containing non-critical or easily-reproducible data in case of data loss or corruption from unforeseen bugs. With Lustre now starting its third decade of life since conception at Carnegie Mellon University in 1999, it is striving to cast off its teenage years and present itself in a responsible and mature fashion.

As one of the original funders of Lustre and the first user of Lustre in production back in 2003, Lawrence Livermore National Lab's Livermore Computing (LC) has a long and close relationship with the file system and has an interest in seeing Lustre further mature. To that end, in the second half of 2018 and coinciding with the retirement of many older Lustre file systems, LC commenced the "un-scratching" of Lustre: the migration of multiple, production, "scratch" Lustre file systems to persistent, non-scratch, non-purgeable, file systems.

This presentation first addresses the state of Lustre in LC through the first half of 2018. It then further details the rationale behind this change, specifically from a user and an administrative perspective. Next it covers some of the mechanics, results, and yes, even a bit of politics involved in the conversion. The presentation then addresses the current state of the production Lustre file systems in LC, including the implementation of user quotas and the takeaways gathered from that experience. Finally, the presentation will touch on what this change means for the future, specifically in regards to refreshing the hardware underlying Lustre.

Lawrence Livermore National Lab

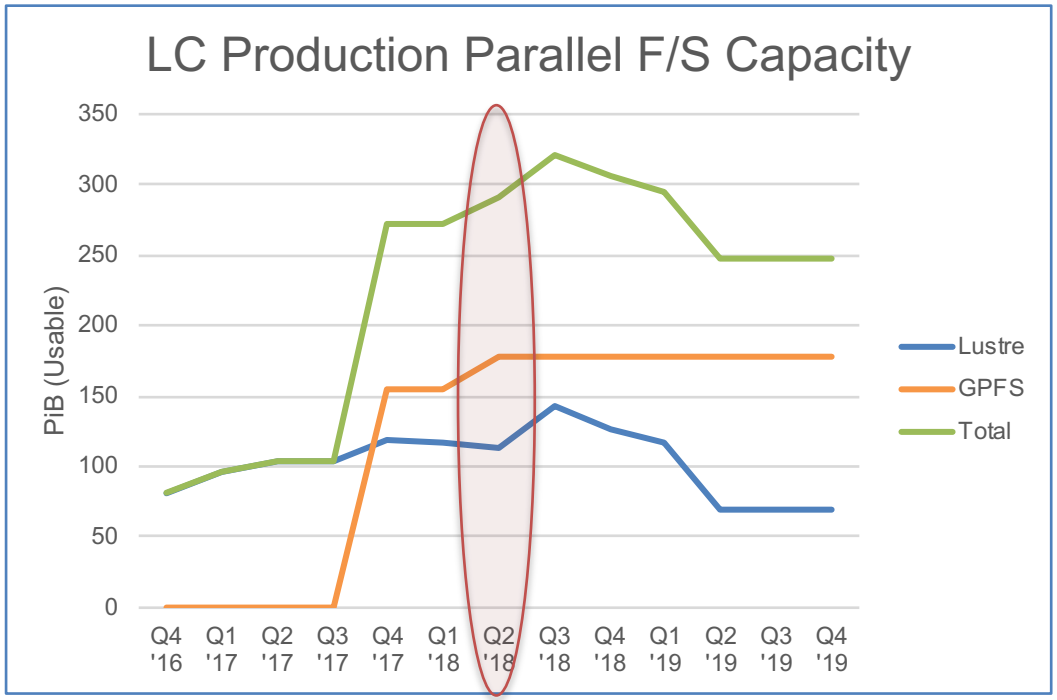
- US DoE / NNSA
  - Missions:
    - Biosecurity
    - Defense
    - Intelligence
    - Science
    - Counterterrorism
    - Energy
    - Nonproliferation
    - Weapons

Livermore Computing (LC)

- Compute
  - Classified: ~151 PF
    - Sierra: 126 PF_{pk}, #2
    - Sequoia: 20 PF_{pk}, #10
  - Unclassified: ~30 PF_{pk}
    - Lassen: 19 PF_{pk}, #11
- 4+ Data centers
  - TSF: 45MW -> 85MW
- 3 *Centers: CZ, RZ, SCF*
- Software Too!
- We're Hiring!

Parallel FS @ LC (2018)

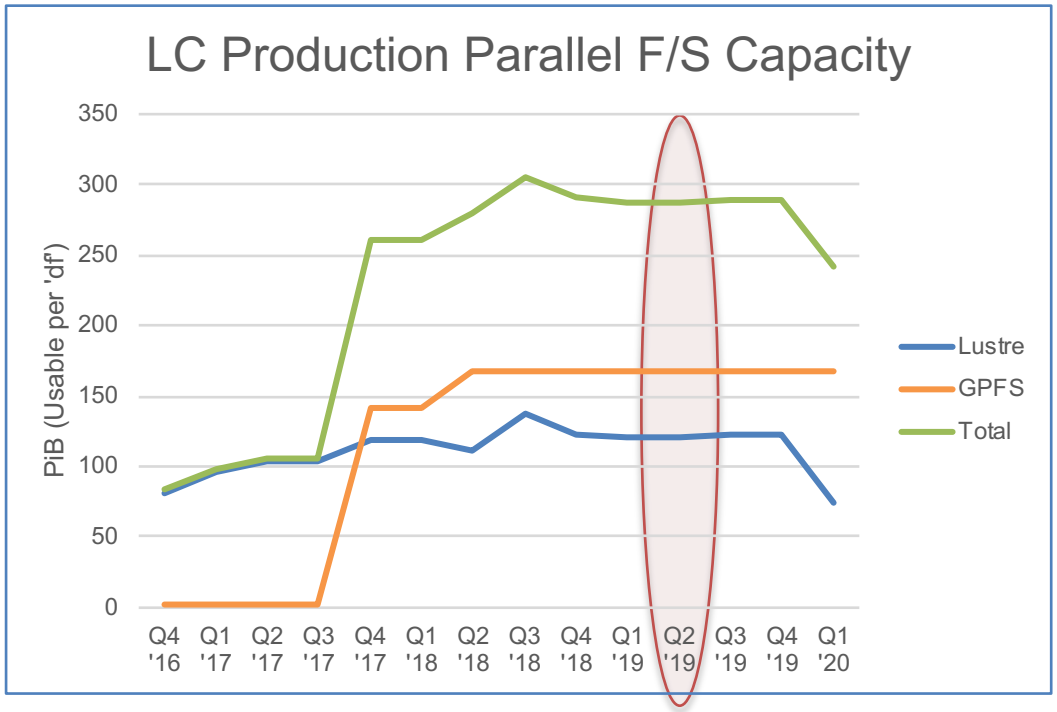
- Production Lustre
  - 13 production file systems
  - >118 PiB (useable)
  - ~15B files
- Multi-generation
  - Lustre 2.5 (NetApp/Cray)
    - 1 MDS
    - ZFS 0.6
  - Lustre 2.8 (RAID Inc.)
    - JBODs
    - 4-16 MDS
      - DNE v1
    - ZFS 0.7



filesystem	Used Space in TB	Percent Full	Millions of files	Average File Size in KB
/p/lscratchd	4294	78%	1035	4456
/p/lscratche	3759	69%	1130	3573
/p/lscratchf	1679	77%	809	2229
/p/lscratchh	6807	41%	3363	2173
/p/lscratchrza	5734	69%	1207	5099
/p/lscratchrzb	461	42%	361	1371
/p/lscratchv	3659	56%	1638	2399

Parallel FS @ LC (2019)

- Production Lustre
  - 8 production f/s
    - 13 - 8 + 3
  - ~120 PiB (useable)
- Multi-generation
  - 3x NetApp
    - 2x 2.5
    - 1x 2.10
  - 5x RAID Inc.
    - 3x 2.10
    - 2x 2.8
  - 2.8/2.10 clients



filesystem	Used Space in TB	Percent Full	Millions of files	Average File Size in KB
/p/czlustre1	2739	17%	310	9492
/p/czlustre2	4058	25%	733	5943
/p/czlustre3	1417	26%	546	2788
/p/lustre1	1955	24%	322	6516

Lustre Naming Scheme (2018)

- What's in a name?
  - *lscratchX*, where *X* is
    - [a-z] on unclassified *Centers* (*CZ, RZ)*
    - [1-*n*] on classified *Centers* (*SCF*)
  - LC's "*lscratchX*" naming caused consternation for users
  - "Scratch" implies:
    - Short longevity
      - Files will self-destruct (purged)
    - Reduced stability

https://images.app.goo.gl/DpxfGgQzUMsQmjom6

Lustre Scratch Purge Policy (2018)

- Official policy: Files > 60 days can be purged
  - Bad for users as losing one file can destroy a large dataset
  - Small users and early-alphabet users purged disproportionately
- Effective policy: purge @ ~80% after cleanup
  - Target top-10 users (files or capacity)
  - Ask users to clean up, then use *lpurge* as last resort on select users
  - Pros
    - Saves small users from suffering from the actions of power users
    - Enables greater utilization of f/s
  - Cons
    - Still requires overhead/time from admins and LC Hotline
    - Delays from users can cause uncomfortable levels of usage
    - Users don't clean up unless forced to

Un-scratching Lustre

- Name change
- Persistent data
- Stability
- Features

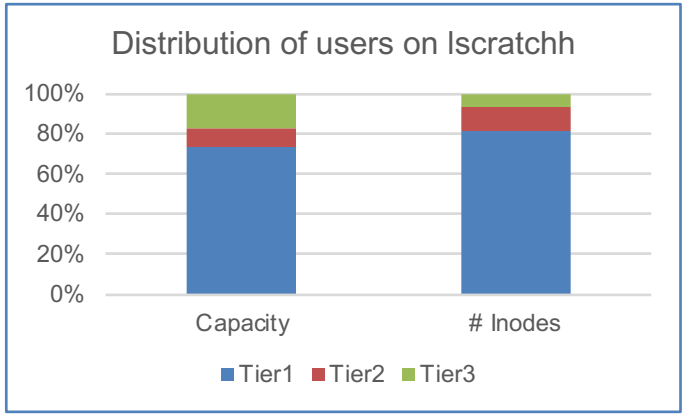
Lustre Naming Scheme (2019)

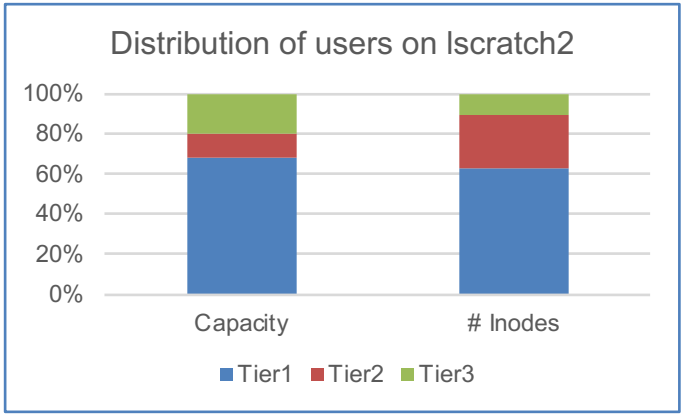
- From scratch to not-scratch
  - From *lscratch[a..z | 1..n]* to *lustre[1..n]* everywhere
    - Every *center* has a *lustre1.*
    - A *center* may have a *lustre2* or *lustre3* as well.
  - Causes less consternation for users
    - Porting code/scripts easier
  - But …
    - Need to always specify LC *center*
    - Name conflict on nodes that mount both *RZ* and *CZ* Lustre

Lustre Purge Policy (2019)

https://techcrunch.com/wp-content/uploads/2010/12/aol.jpeg

Lustre Quota Policy (2019)





Quota Tier	Capacity (TB)		# Files		Grace Period (days)
	Soft	Hard	Soft	Hard	
1	18	20	900K	1M	10
2	45	50	9M	10M	10
3	Levels set per justification				10

- Per-file system
- Tier 3:
  - Custom # inodes, TB
  - Max duration: 6 months

Auto-delete

- AutoDelete directories
  - Users would `rm –rf <dir>`
    - And wait
    - … and wait
    - … and wait
  - Now they can `mv <dir> … ` and get on with life
  - drm job, as <user>, removes the files quickly
  - https://github.com/hpc/mpifileutils

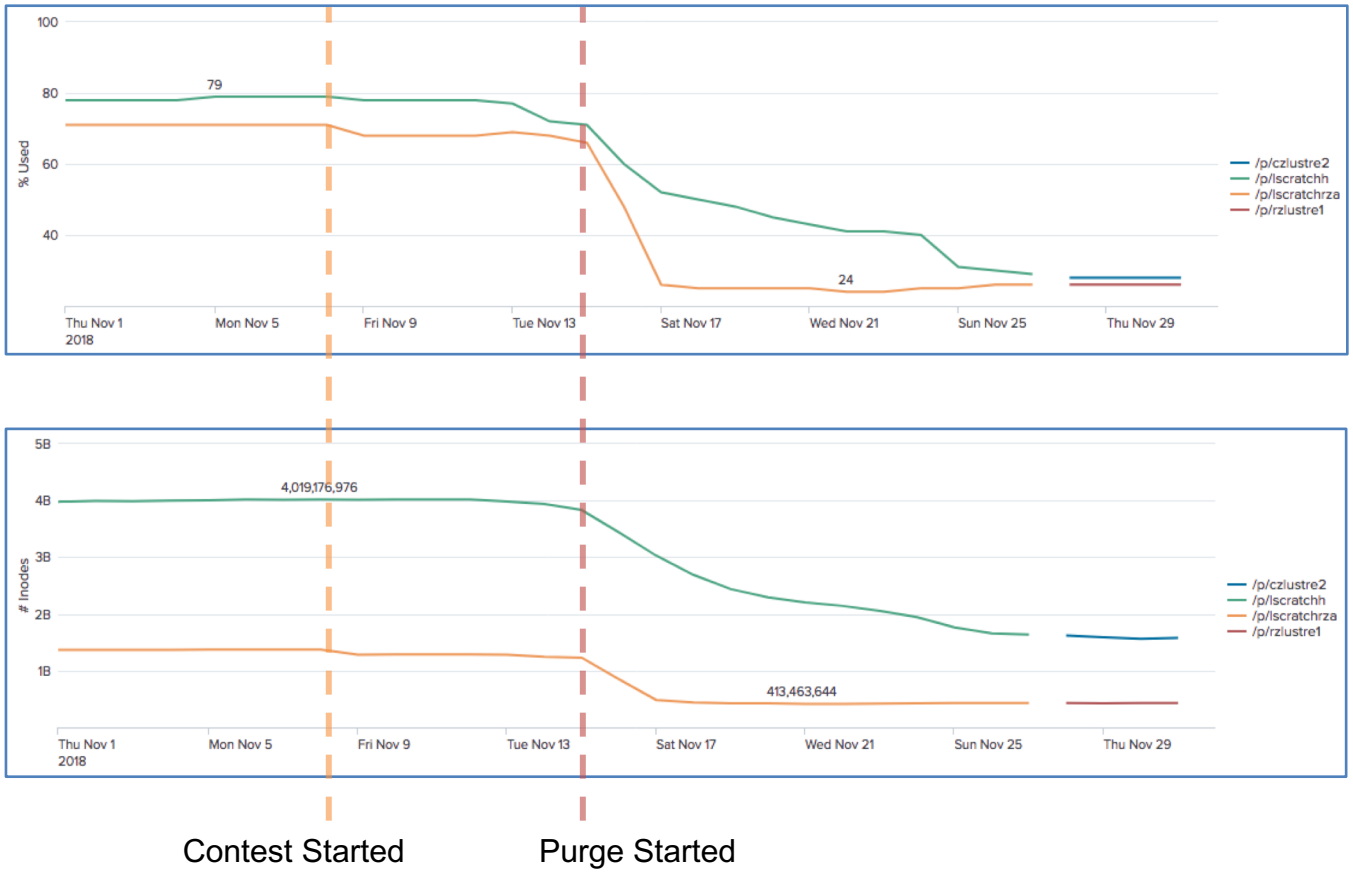
How We Did It

- Stand up new File systems with new policy
- Incentivize clean-up on existing file systems
  - Gift card
  - Exemptions
- One-and-done big purge

https://www.bulldozer.in/images/solid_waste_%20machine/sd7n_solid_waste_blade_dozer.jpg

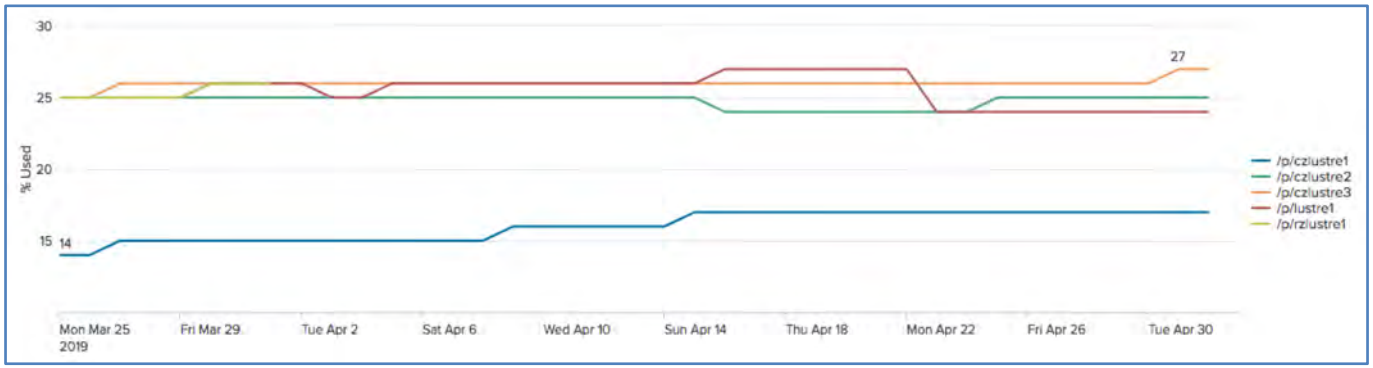
The Purge

- Before Cleanup
  - Capacity:
    - 79% full
    - 13.2 PB
  - Inodes:
    - 4 Bi

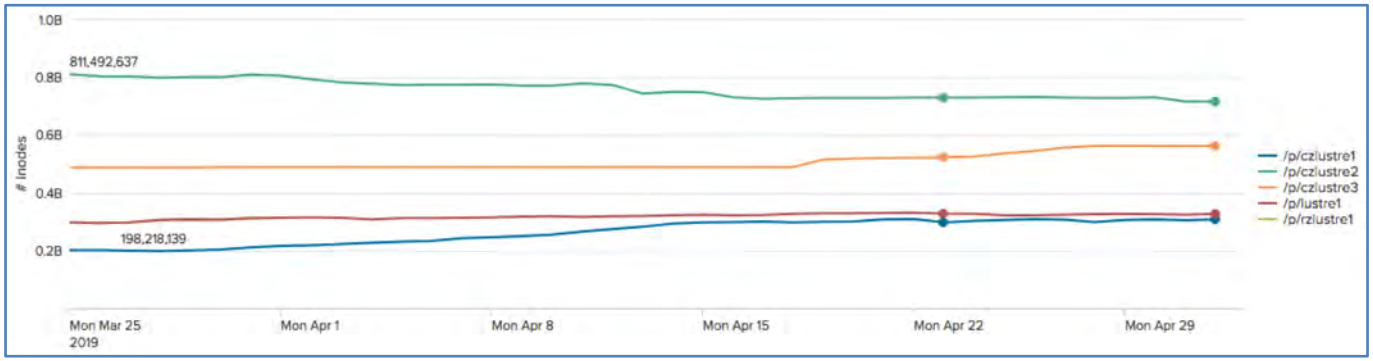


Long-term Results

- Current utilization
  - Capacity:
    - < 30% full

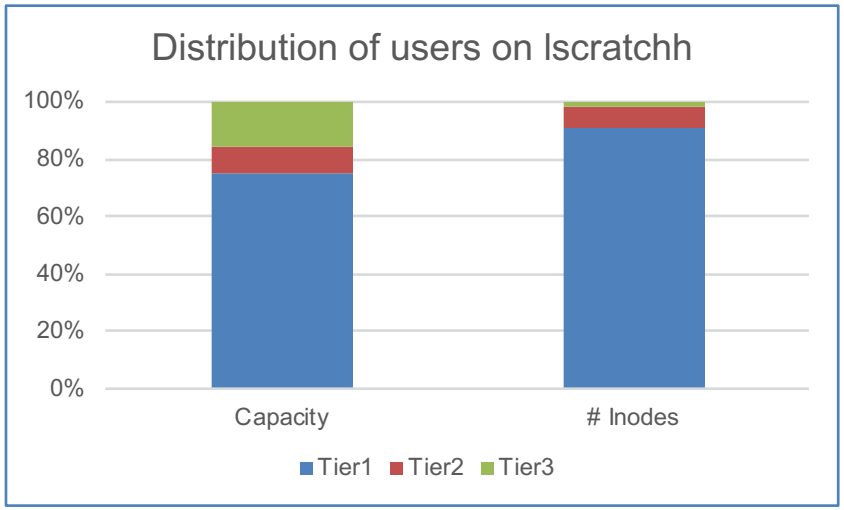


  - Inodes:
    - < 1B files



Long-term Results (cont.)

- Current status
  - Tier 3 allocations (aggregate):
    - 65 users on CZ/RZ
    - 21 users on SCF



- Lessons learned
  - More increases requested than anticipated
    - Enabled LC Hotline to effect the changes
    - Inodes more in demand than expected
      - Bumped Tier 1 to 1M from 500K files
  - Created system to track/check/set/remove Tier 3 allocations

Users' Thoughts

- Current status (cont.)
  - Users mostly pleased with the change
    - Only one user vocally unhappy
    - Paraphrased user responses (per user coordinator):
      - WHAT?!? My files aren't going to disappear?!? That is wonderful! Why didn't I hear about this?
      - 20TB is toooo small for me. Why can't I get more? I can get more?!? You're the best!
      - Ugh. Now I have to figure out what to delete? Why can't LC do that for me based on these rules <insertruleshere>? But they better never delete file X - that is the exception to those rules. Oh. I see now what you mean. That autodelete directory is super nice!
      - Wait. I know you said my files weren't going to disappear, but did you really mean it? I figured that once the system got to a certain point, they would.
      - I reallllllyyyyy like that my files aren't going to disappear.
      - THANK you for emailing me that I am reaching my quota. I wish that it came <more/less> often.
      - While I hate having to clean up after myself, it is WONDERFUL that I am not going to lose any files.
  - Grace period isn't liked
    - "Why can't I use all my allocated storage?"
    - Would like to set infinite grace period
      - LU-12280

Move to 2.10

- Challenges
  - Two old versions
    - 2.5.5, 2.8.2
  - File systems cross-mounted
    - Hard to avoid mix of 3 versions
  - 2.5 won't die!

https://giphy.com/gifs/black-knight-invincible-gif-im-10M2ZnecwCMVGg

Upgrading to 2.10

- Usual test on Dev systems
  - All seems well
- Sync lscratchf -> lustre3
  - Not quite so cute…
  - Tried syncing data 4 months
    - Huge directories
    - Dsync issues
    - Abysmal performance
      - LU-11798
    - Data Corruption
      - LU-11663
    - LFS migrate broken
      - LU-11620

https://tenor.com/view/monty-python-holy-grail-killer-gif-6118842

- Current Status
  - Moved all of RZ (servers, clients) to 2.10
  - All CZ servers at 2.10
    - SCF in June?

Thank you!

Lawrence Livermore
National Laboratory