dCache tape pool performance

Niklas Edmundsson
HPC2N, Umeå University

dCache tape pool performance

This presentation is focused on dCache tape pools as most commonly deployed on the NDGF Tier1, with TSM as the tape provider and ENDIT as the connecting glue.

However, most of the issues touched upon should be of interest for all dCache disk pool deployments.

Problem statement

- Current NDGF Tier1 dCache tape pools are having trouble delivering optimal IO performance during real-life loads.
- Optimal in this case is defined as able to always utilize tape drives fully, ie if a tape drive can do 120MB/s but we're seeing 60MB/s than we're not optimal.
- Main culprit seems to be disk-IO related, but things such as bad tapes etc also exists.
- Todays main focus will be on disk IO.

Problem statement (2)

- In the future we will see an increased bandwidth requirement, roughly a factor 2. Maybe more...

Scalability issues (tape HW)

- LTO tapedrive performance does not seem to follow previously announced roadmaps
  - 2006: LTO4 – 800 GB @ 120MB/s
  - 2010: LTO5 – 1500GB @ 140MB/s
  - 2012: LTO6 – 2500GB @ 160MB/s
  - 2015?: LTO7 – 6400GB? @ 315MB/s?
- Means we'll need to use more tapedrives
- Or change tape technology
  - 2011: IBM TS1140: 4000GB @ 250MB/s
  - 2011: Oracle T10000C: 5000GB @ 250MB/s

Scalability issues (SW)

- ENDIT currently doesn't do multi-tapedrive efficiently
- We were hoping for IBM to include efficient multi-tape retrieve in dsmc, but seems to require rather large effort to convince them.
- We have a plan in place to work around this.
  - Requires exports of volume content lists from TSM server.
  - Will support multiple tapepool machines.

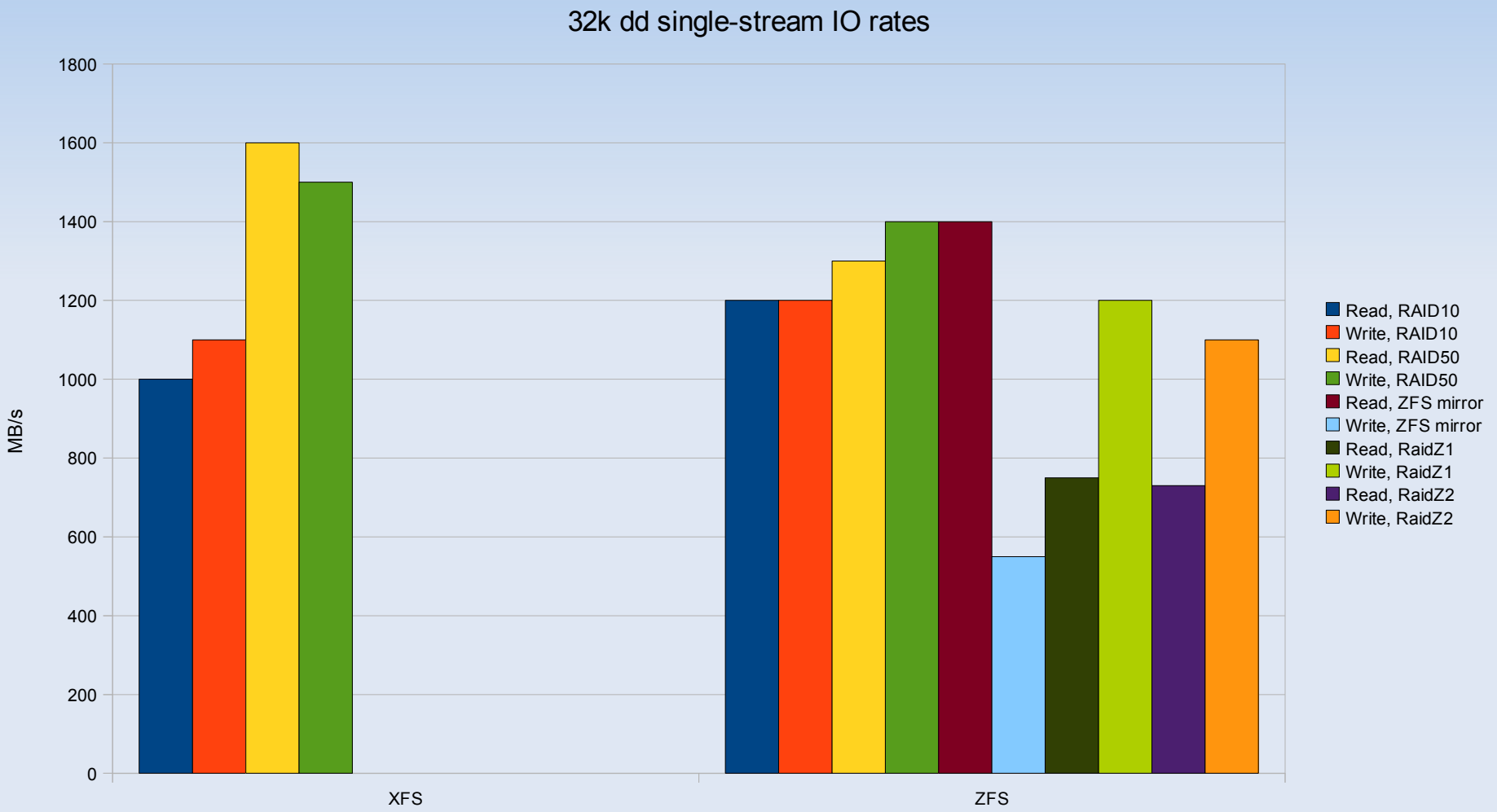
Scalability issues (pool HW)

- Increased tape drive bandwidth means 10GigE (or LanFree, but we think it's not worth the hassle)
- Worst-case IO becomes 10GigE dCache in/out while at the same time streaming at (multiple) tape speed.
  - Remember, not fully using the tape speed will require more tape drives to compensate.
- Current investigations shows that disk IO becomes a big bottleneck.

Scalability issues (pool HW) (2)

- Will be expensive to solve the disk IO issue by going the SSD route.
  - Tape pool needs to be able to cope with incoming data even if tape not available.
  - When doing a tape mount: write a good sized chunk at a time, consider 1h worth of tape IO a minimum.
- Even going for 10k/15kRPM HDDs might be too expensive.

But look at the benchmarks



But look at the benchmarks (2)

- Problem is, benchmarks like these seldom resemble real-life load.
- We're actually doing a poor job on having benchmarks that matches real-life loads.
- Which leads to us architecting solutions either using a bad method of evaluation or just on a whim.
- Usually leads to equally questionable results.
- There is a reason that Mattias W requires Ganglia on all dCache pools.

Evaluating tape pool performance

- Initiated when we realized that Solaris was not a reasonable future plan (thanks Oracle)
- According to Mattias W, the HPC2N+PDC tape pools are the only tape pools in the NDGF Tier1 that performs as expected today.
- Probably a result of them running Solaris+ZFS, but we needed some way of evaluating this.

Eval tape pool - tools

- We needed some way of emulating tape drive IO
  - Hacked a simple script using rsync –bwlimit
  - Copies a 1GB data file to/from tmpfs
- seq|xargs to do parallel file creates
- ls -1|sort -R|xargs to do // file reads

Eval tape pool - how?

- It Depends<tm>.
- These first efforts focused on behaviour when maxing out the performance available
  - Find out how many LTO6 ”tapedrives” can be handled.
  - Find out what happens when competing ”opposite” IO happens, ie 10GigE rate incoming/outgoing IO.
  - Lessons learned along the way, so some data points incomplete

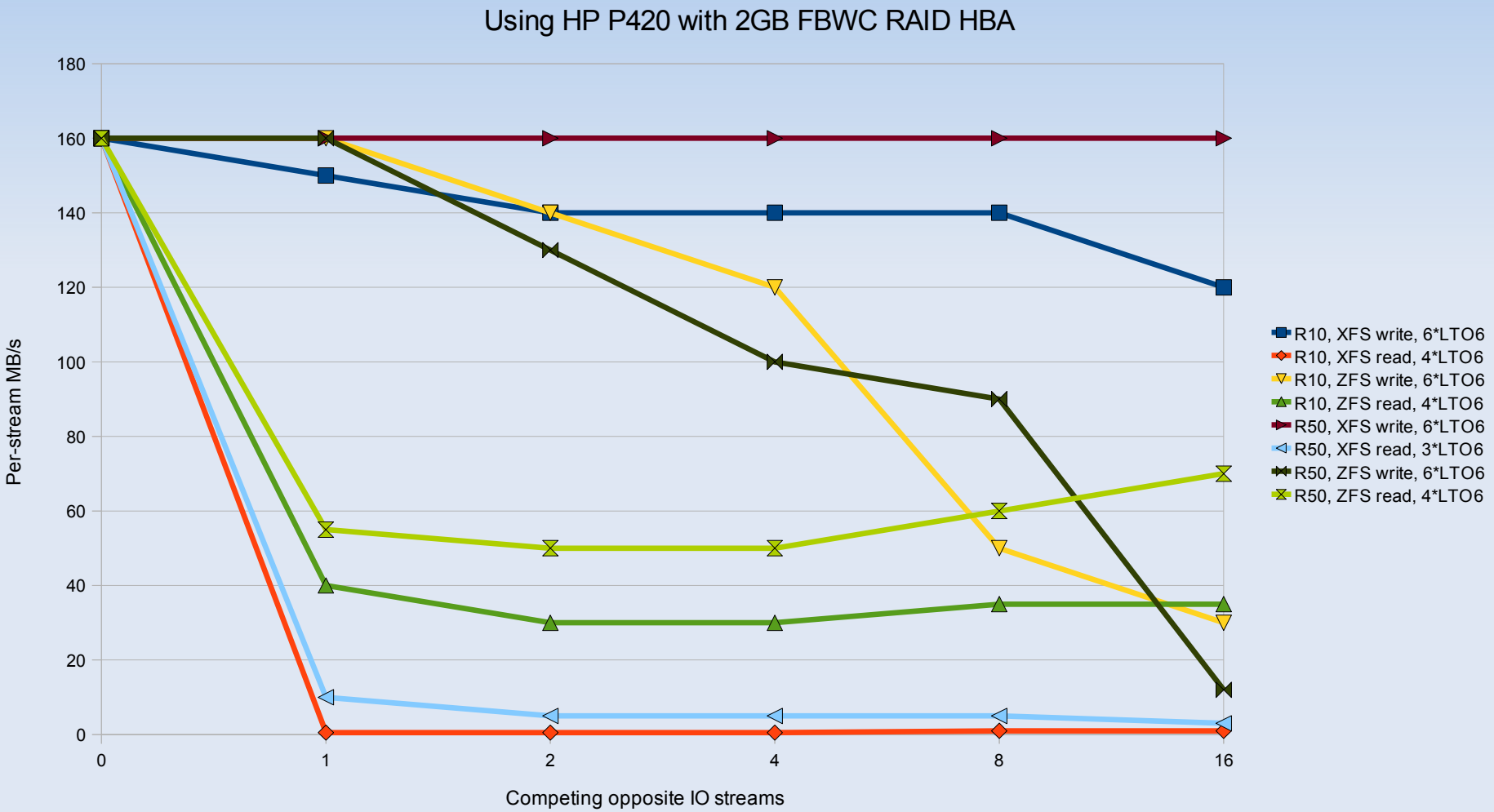
Test hardware

- HP DL380e 668668-421 ”storage server”
  - 25 SFF HDD slots
    - 13/12 split, one 4x6G SAS connector each
  - 25 500G SFF 7kRPM NL SAS HDDs
  - One Intel E5-2420 6core 1.9GHz CPU
  - 36GB RAM
  - HP P420/2G FBWC RAID controller
  - HP H220 SAS HBA
  - HP 530SFP+ 10GigE NIC

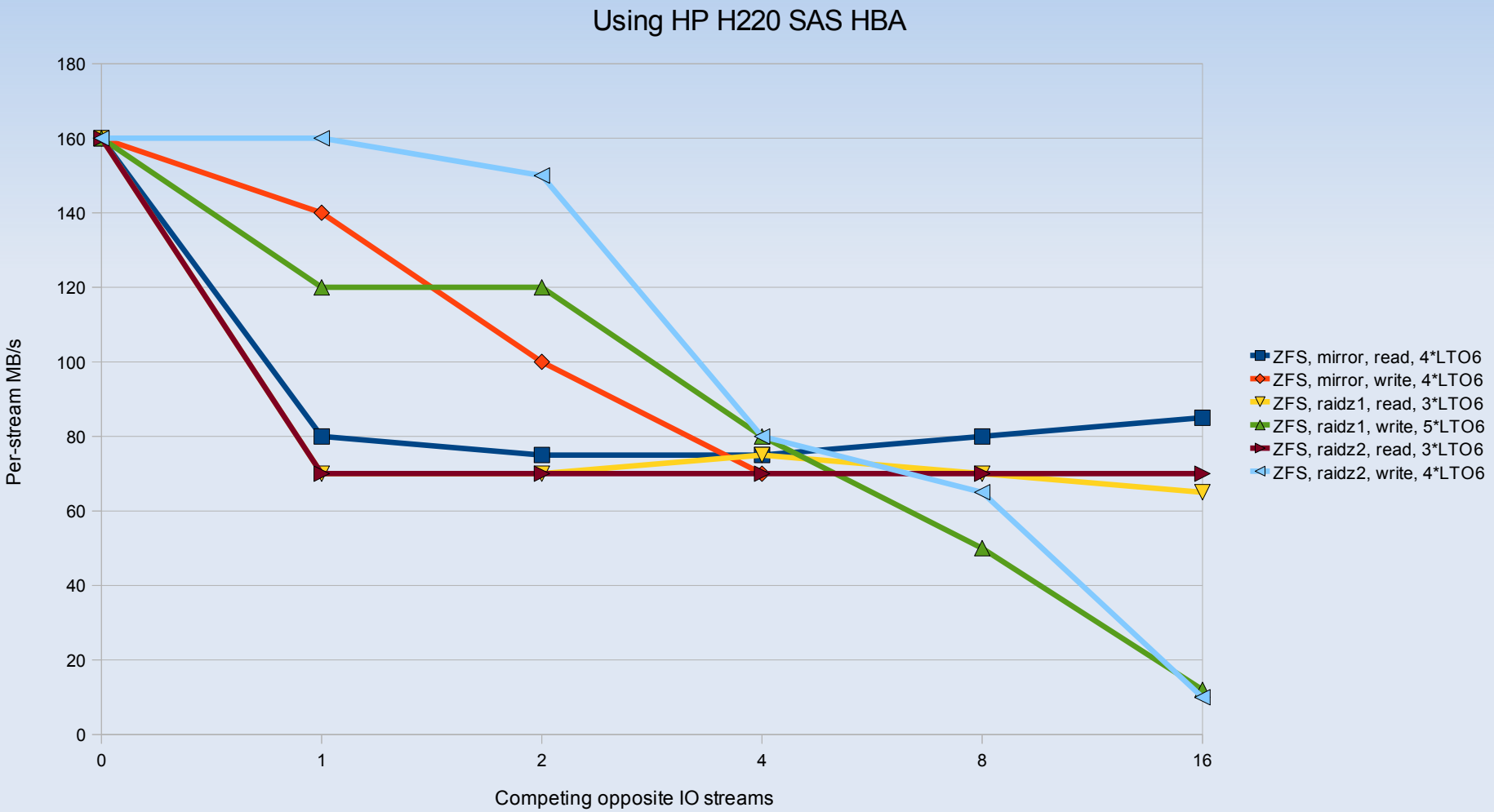
Test hardware (2)

- Aggregated disk IO capacity is approx 2000MB/s measured using the SAS HBA and running one dd for each HDD concurrently.
  - Limit here is most likely the 7kRPM SFF HDDs.
- There were always CPU cores idling during tests.
- In the concurrency tests the competing IO rate was at most on 10GigE level.

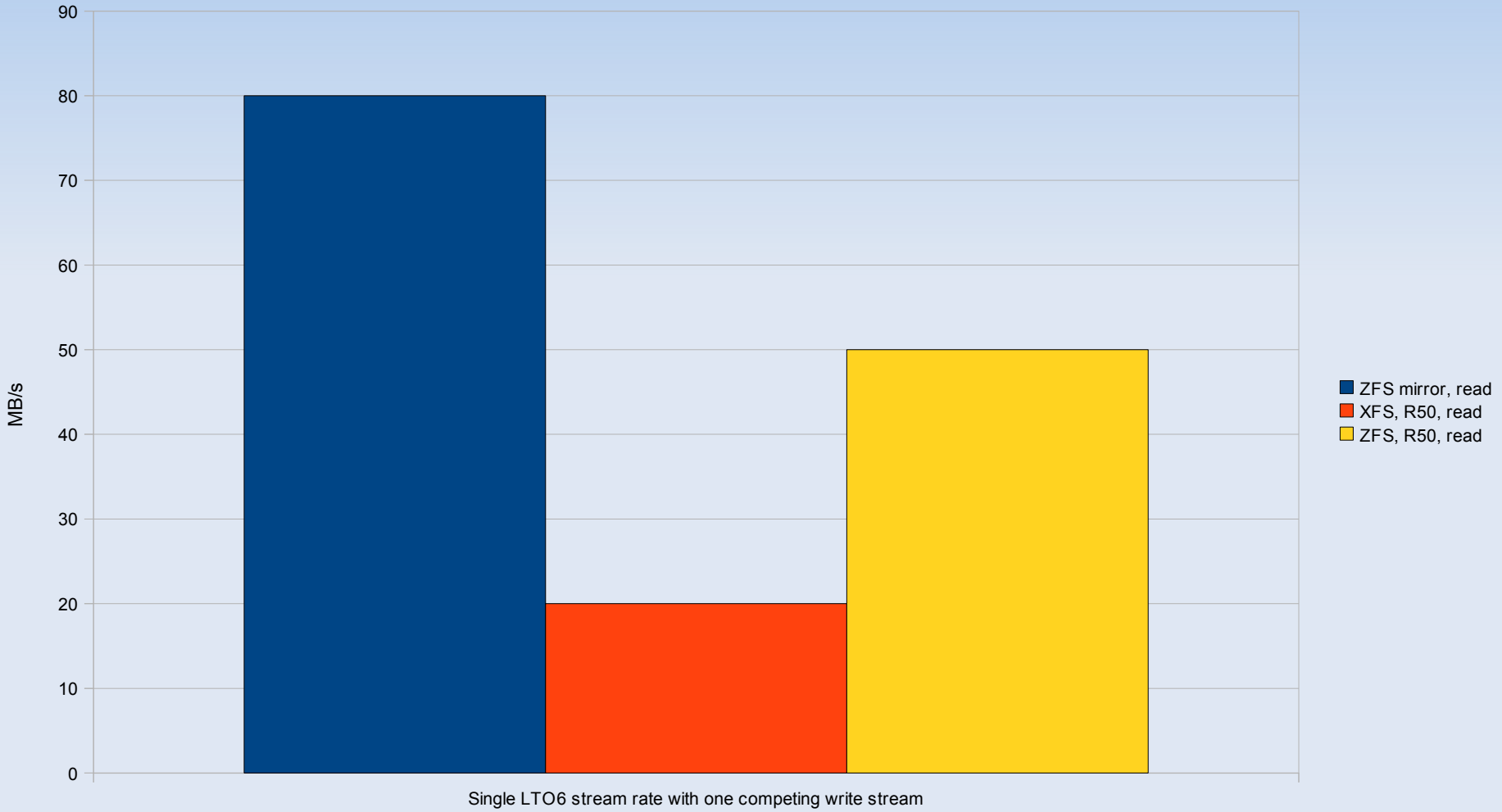
Concurrency disappointment



A bit more balance



Single LTO6 stream



Conclusions?

- It Depends<tm>
- The point I'm trying to make is that this is NOT a trivial problem to solve.
  - Although we could go the regular computer-nerd-route and apply beefier hardware at the problem.
    - Easiest if we get funding :-)
- I have more test data than included in these graphs, hard to present usefully though.
- However, ZFS seems to show better behaviour wrt fairness and consistent results.

Things to consider

- Throttling dCache when doing tape IO to reduce the impact of dCache transfers.
  - cgroup blkio throttle can't throttle buffered writes (yet, patches do exist...).
  - Linux firewalling/iptables probably not suited for high-bandwidth throttling? (no recent tests done)
  - Solve in network equipment?
  - Solve by having GigE connection(s) for dCache transfers and 10GigE for tape/TSM?
- More disk IO capacity
  - Reduces the impact of hitting the wall.

Things to consider (2)

- When upgrading tape drives, tape pools need to be able to keep up
  - Or you can't benefit from increased tape data rate
  - Either size for the future or replace when upgrading
- ZFS on Linux
  - In stabilization phase
  - Performance optimizations to come
  - Worth considering
  - On Solaris, possible to tune to prefer read or write load (not tested on HPC2N pools)