Local Filesystems

KELEMEN Péter
CERN IT

HEPiX Storage Day
Rome, Italy

Filesystems

- inodes...
- block-structured... fine for small-to-medium sizes
  - recovery times too long for big sizes
- journaling...
  - (de)allocation times too long for big sizes
- extent-based...
- several competing filesystems
  - ext3, XFS, JFS, ReiserFS, all capable of 2+ TiB

ext3

- block-structured
- journaled (metadata, metadata+data)
- complexity: ~10'000 SLOC (including JBD)
- vanilla inclusion: 2000
- development: yes, RedHat
- RHEL4 status: OK
- very stable, very widely used general-purpose fs

XFS

- extent-based, multiple B+-trees
- journaled (metadata only)
- complexity: ~100'000 SLOC
- vanilla inclusion: 2001
- development: yes, SGI
- RHEL4 status: disabled
- feature-rich, fully 64-bit, widely used general purpose fs geared towards large files

JFS

- extent-based, multiple B+-trees
- journaled (metadata only)
- complexity: ~30'000 SLOC
- vanilla inclusion: 2002
- development: yes, IBM
- RHEL4 status: disabled
- slow but steady progress, not widely used yet

ReiserFS (v3)

- single B+-tree design
- journaled (metadata only)
- complexity: ~30'000 SLOC
- vanilla inclusion: 2001
- development: no, Namesys
- RHEL4 status: disabled
- widely used, designed for small files, but fragile
- Reiser4 as successor, not in vanilla yet

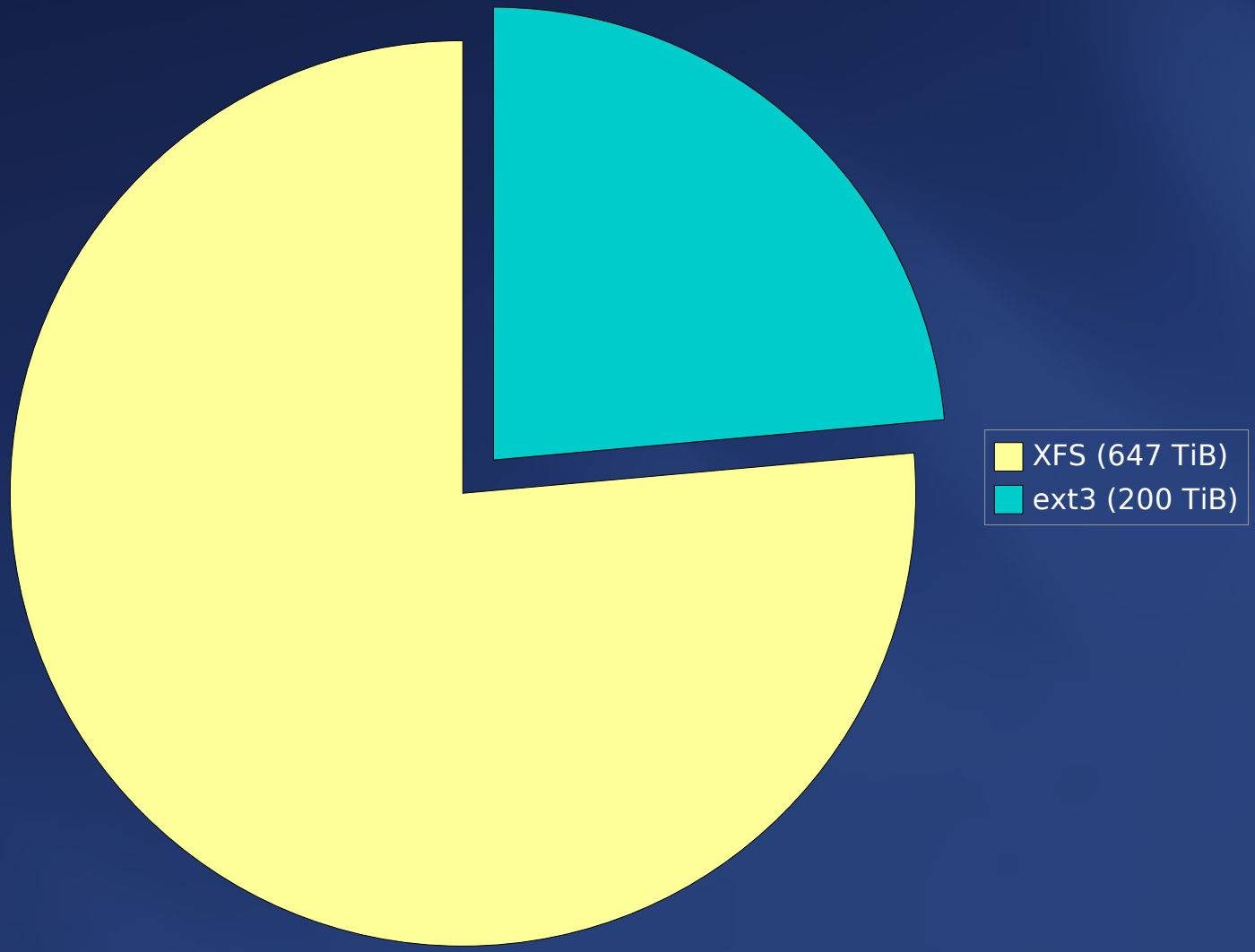
At CERN, we need...

- large filesystems (1 TiB .. 5 TiB)
- large files (0.5 GiB .. 5 GiB)
- streaming I/O (RFIO)
- fast massive cleanup operations in batches (rm -rf)
- extras?
  - delayed allocation
  - space reservation (preallocation)
  - online defragmentation
- ...XFS was selected

XFS at CERN

- introduced in 2003/2004
- not available in RHEL, CERN includes XFS in SLC
- ~647 TiB XFS in production (!)
- various operational problems in SLC3
  - 4+ GiB AGs are problematic under load
  - v2 log replay has memory allocation difficulties
- with care, still excellent performance and stability
- majority of problems due to faulty hardware

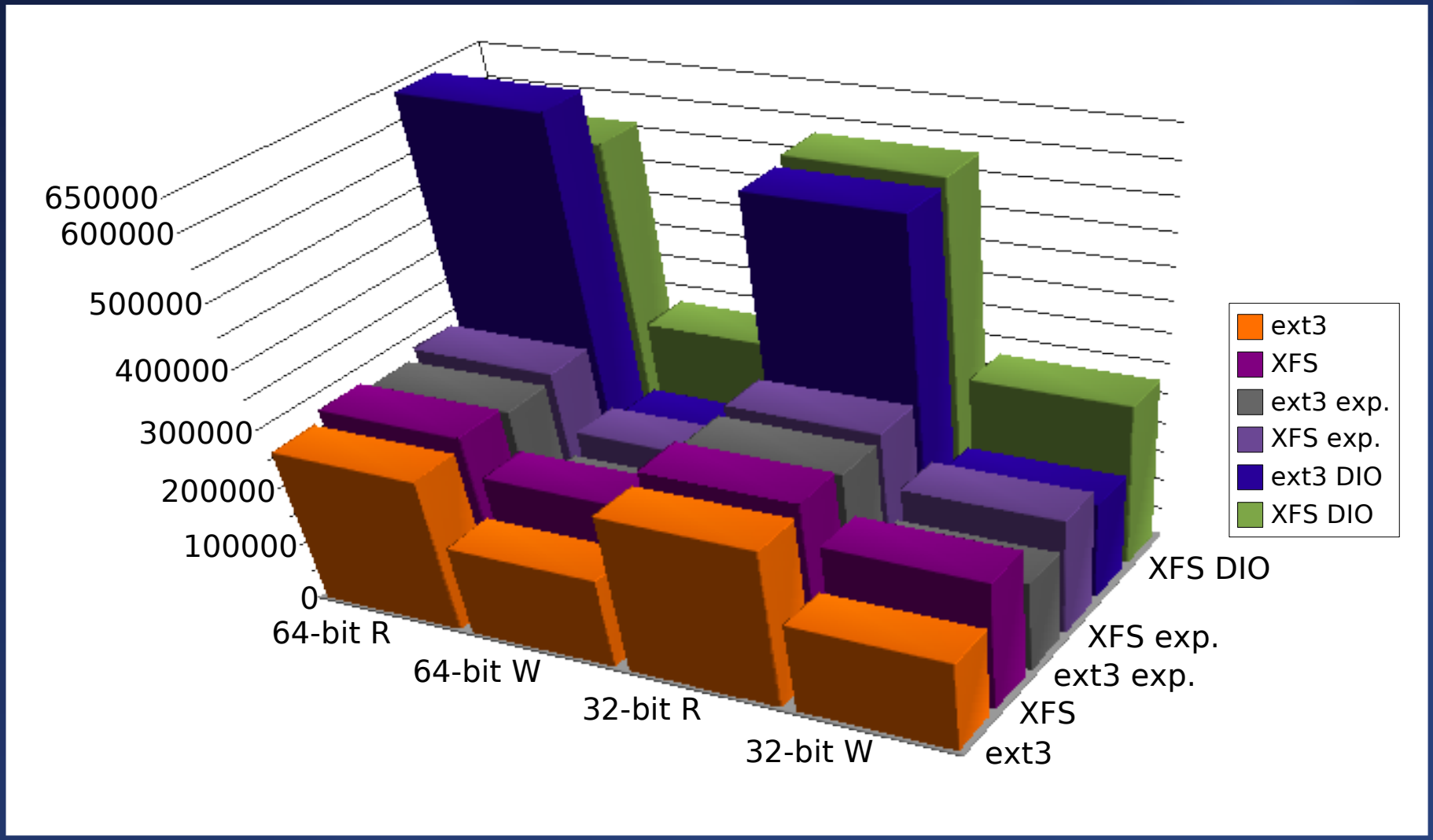
CERN production filesystems



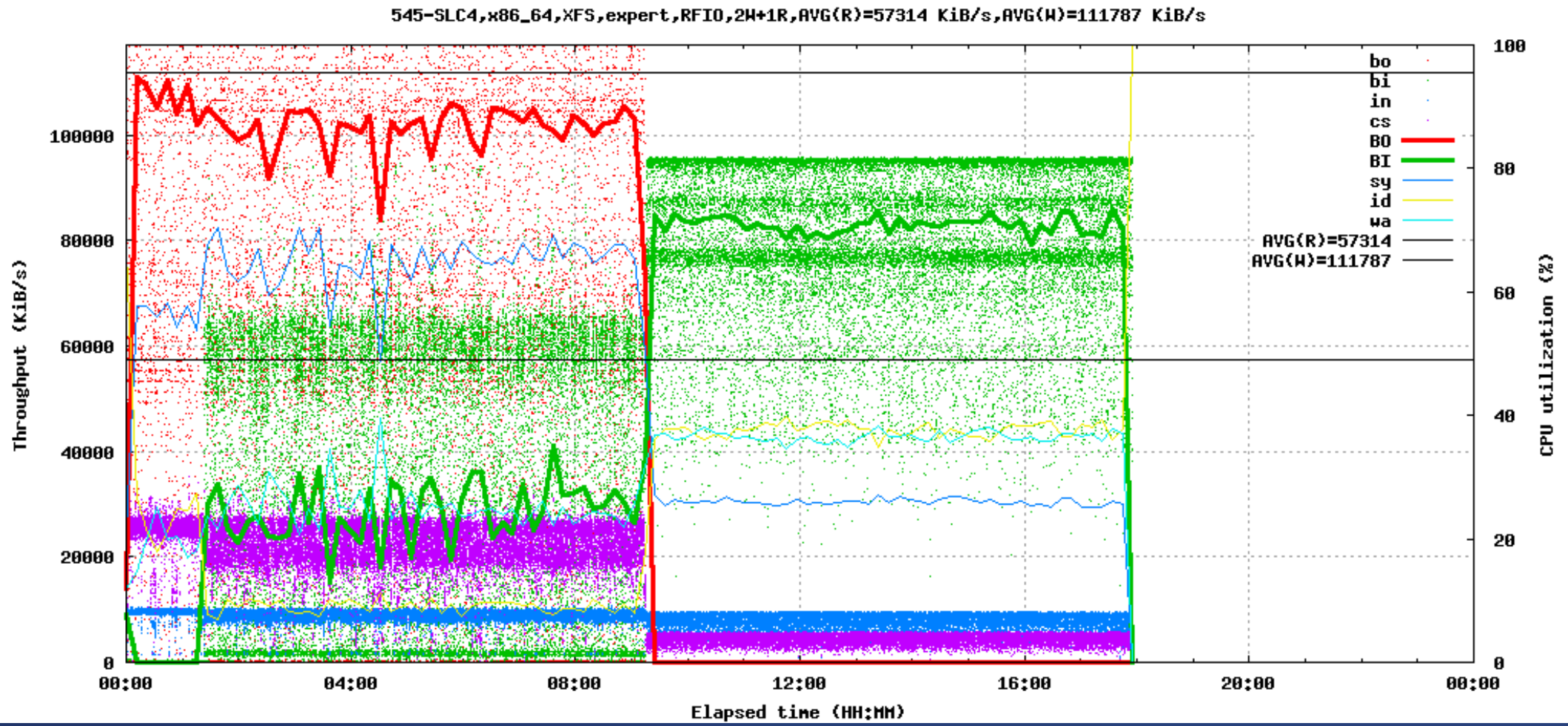
XFS in SLC4?

- XFS codebase as of vanilla 2.6.9 (RHEL4 base)
- major concern: 4K stacks on i386 platform?
  - 4K stack: heavy load triggers the overflow :-(
  - x86_64 is OK, i386+8K: too much divergence
- ext3 is catching up in performance
  - local streaming speed sometimes exceeds XFS!
  - ...deletes are still slow (~90 minutes for 3 TiB)
- O_DIRECT: recent addition to RFIO (XFS/ext3)
  - SLC4: good base, still needs further tuning

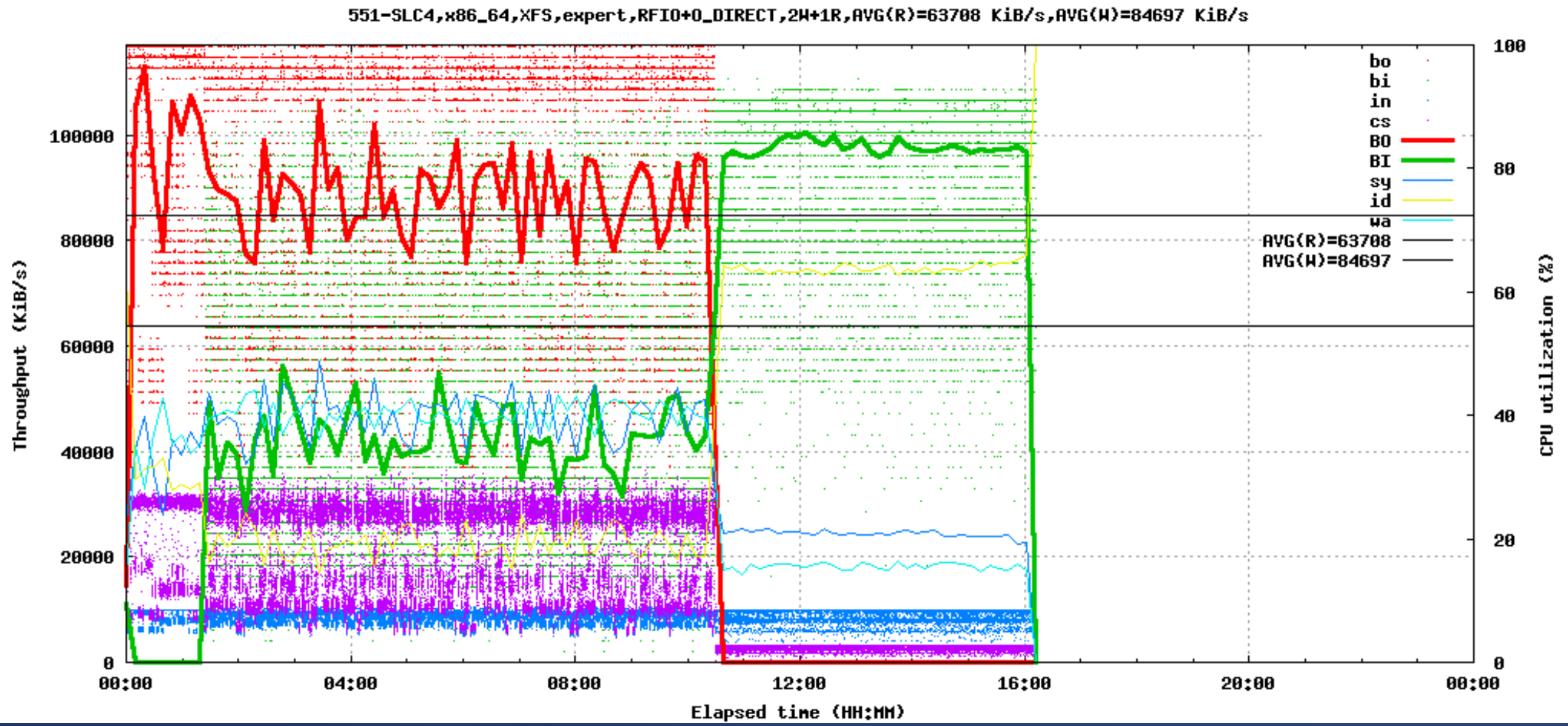
SLC4 Local Tests



Network? Trouble...



Distorted Sampling Example



Conclusion

- raw I/O performance depends on controller/disks
- filesystem choice: diminishing advantages
- system tuning: know your application
- networking changes the whole scenario (as usual)
- PC hardware is still bad at mixing R/W I/O (seeks)
- hardware is changing rapidly
- re-evaluate your environment from time to time
- no silver bullet...
- ...an expert might come in handy

Further reading

- http://olstrans.sourceforge.net/release/OLS2000-ext3/OLS2000-ext3.html
- http://ext2.sourceforge.net/
- http://oss.sgi.com/projects/xfs/
- http://jfs.sourceforge.net/
- http://www.namesys.com/X0reiserfs.html
- http://www.namesys.com/v4/v4.html
- http://en.wikipedia.org/wiki/Comparison_of_file_systems
- https://twiki.cern.ch/twiki/bin/view/LinuxSupport/LocalFsEval
- http://cern.ch/Peter.Kelemen/talk/2004/C5/diskserver/

Questions?

Thank you and have a nice filesystem!