DCache at Purdue CMS Tier2

US CMS Tier-2 Workshop
LIGO
March 3, 2009

Fengping Hu, Preston Smith

Purdue University

RCAC Rosen Center for Advanced Computing | ITaP INFORMATION TECHNOLOGY AT PURDUE | PURDUE UNIVERSITY

Outline

- DCache Resources at Purdue
- Performance Results
- Limits & Operational Issues Encountered
- Conclusion

Purdue CMS Tier2 Data Center

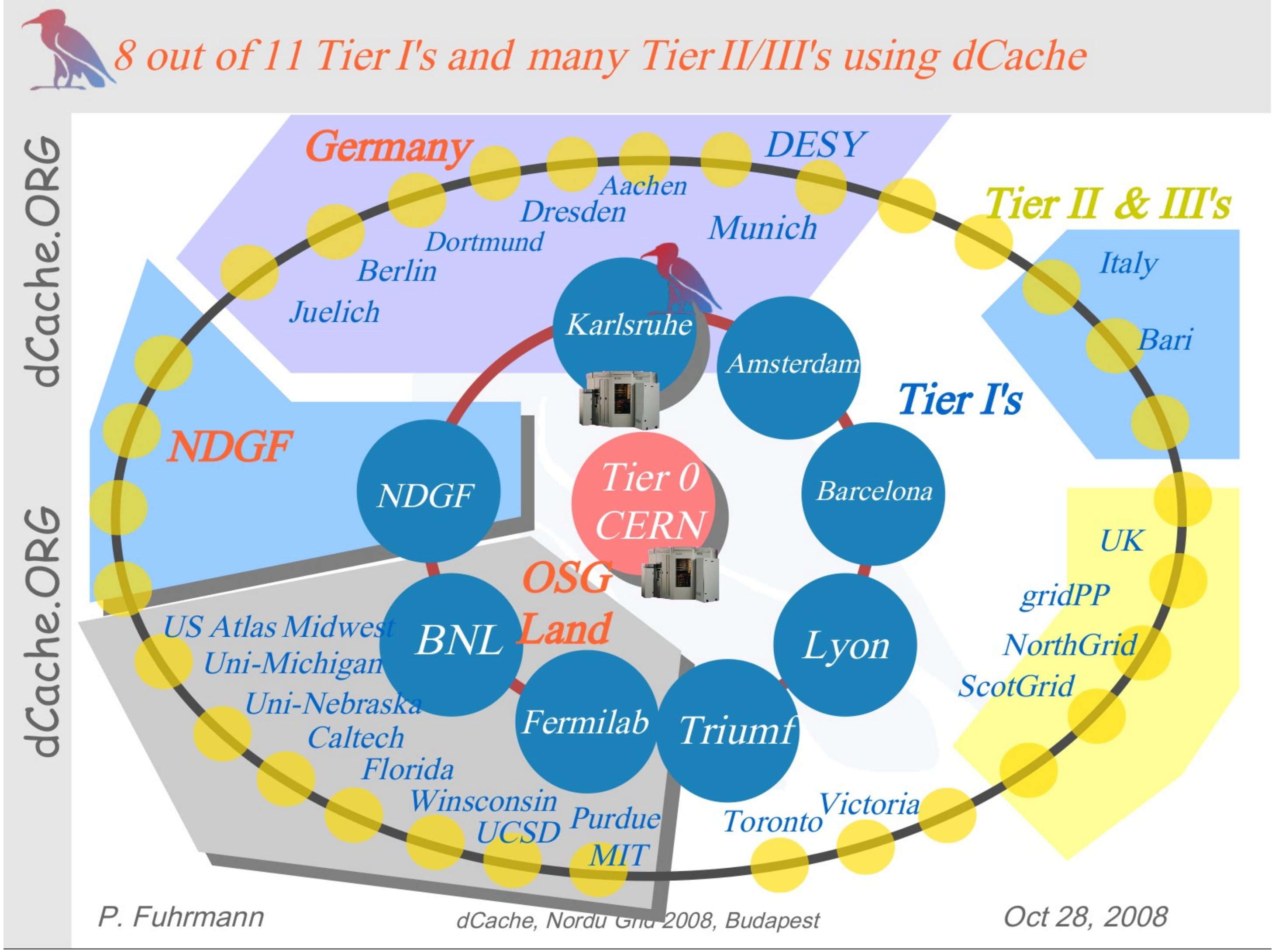
Size: ~ 500 TB (200 TB in Resilient, 300TB in Non-resilient), ~250 nodes

Networking : 1 Gb/s to Internet 2, 10 Gb/s to TeraGrid, 10 Gb/s to FNAL via StarLigh

Why dCache Was Selected

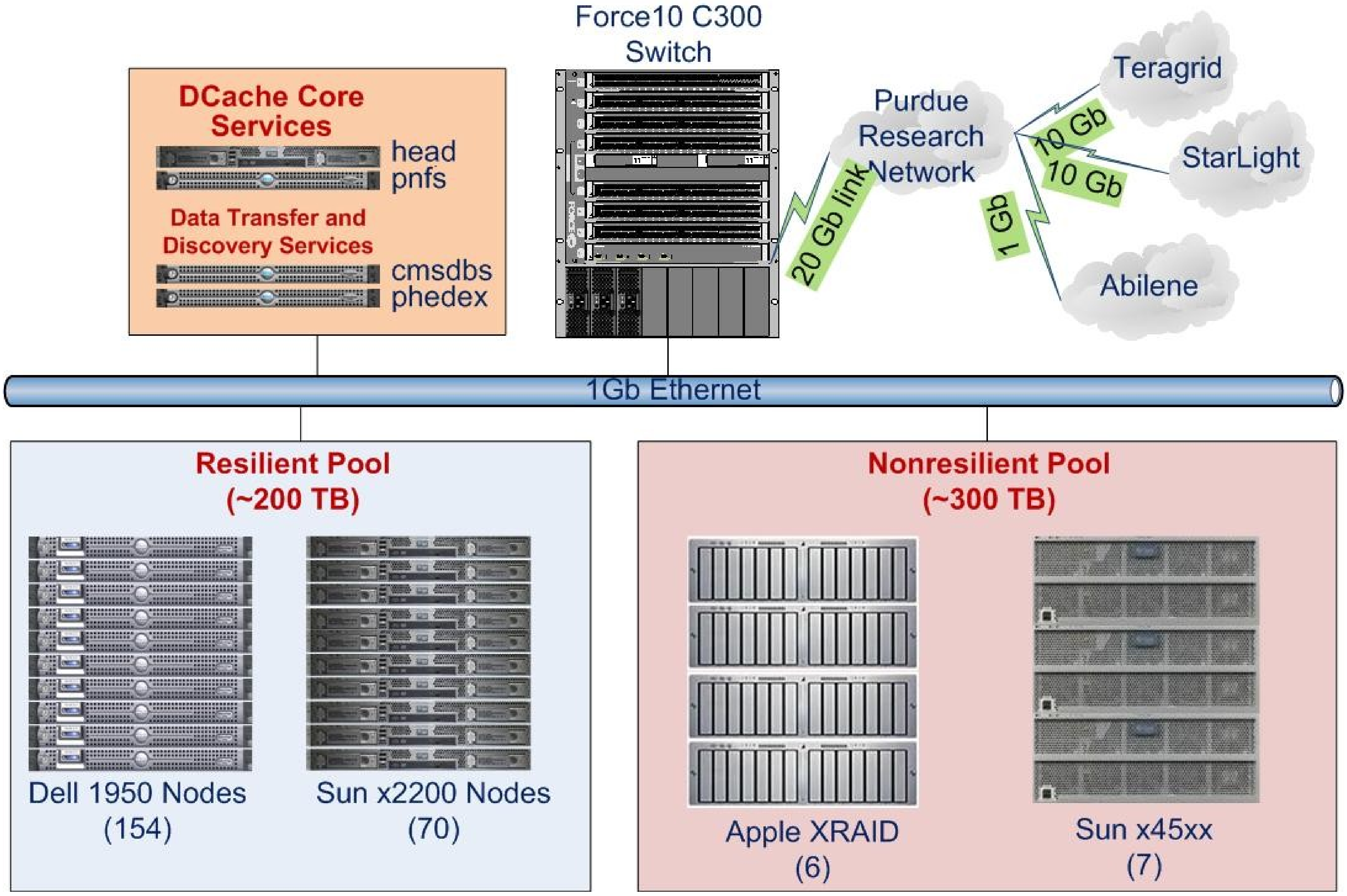
- Used by lots of OSG sites – best choice at the time
- A distributed storage solution that exploits the capabilities of each server
- Designed to be scalable, reliable
  - Fault tolerance by separation of the namespace and data repositories
  - Resilience with the Replica Manager
  - Load balancing with custom cost metrics
- Grid-aware
  - Grid-aware user authentication
  - Interoperability through SRM
  - Information Service
- Fancy features
  - Hot-spot migration (increase throughput by duplicate popular files)

DCache Usage



8 out of 11 Tier I's and many Tier II/III's using dCache

P. Fuhrmann dCache, Nordu Grid 2008, Budapest Oct 28, 2008



Force10 C300 Switch

DCache Core Services

head
pnfs

Data Transfer and Discovery Services

cmsdbs
phedex

Purdue Research Network

20 Gb link

Teragrid

StarLight

Abilene

10 Gb

10 Gb

1 Gb

1Gb Ethernet

Resilient Pool (~200 TB)

Dell 1950 Nodes (154)

Sun x2200 Nodes (70)

Nonresilient Pool (~300 TB)

Apple XRAID (6)

Sun x45xx (7)

Data Resilience

- Resilient pool
  - Spare disk space in computing farm
  - Replica Manager ensures data integrity and availability by keeping replicas of logical files on different nodes
- Non-Resilient pool
  - RAID disks
  - PFM tool address availability issue by replication
- Disk failure in the past year
  - 4 failures out of ~1000 disks (2 in Resilient pool, 2 in NonResilient Pool)

gPlazma Authorization

- Grid-aware PLuggable Authorization Management
  - kpwd
  - Grid-mapfile
  - gplazmalite-vorole-mapping
  - Saml-vo-mapping (GUMS)
- Configurations deployed
  - kpwd+gplazmalite-vorole-mapping
    - grant read access but restrict write access
    - manually add individual DNs to implement access control on /store/user directory
  - Plan to move to gplazmalite-vorole-mapping + saml-vo-mapping

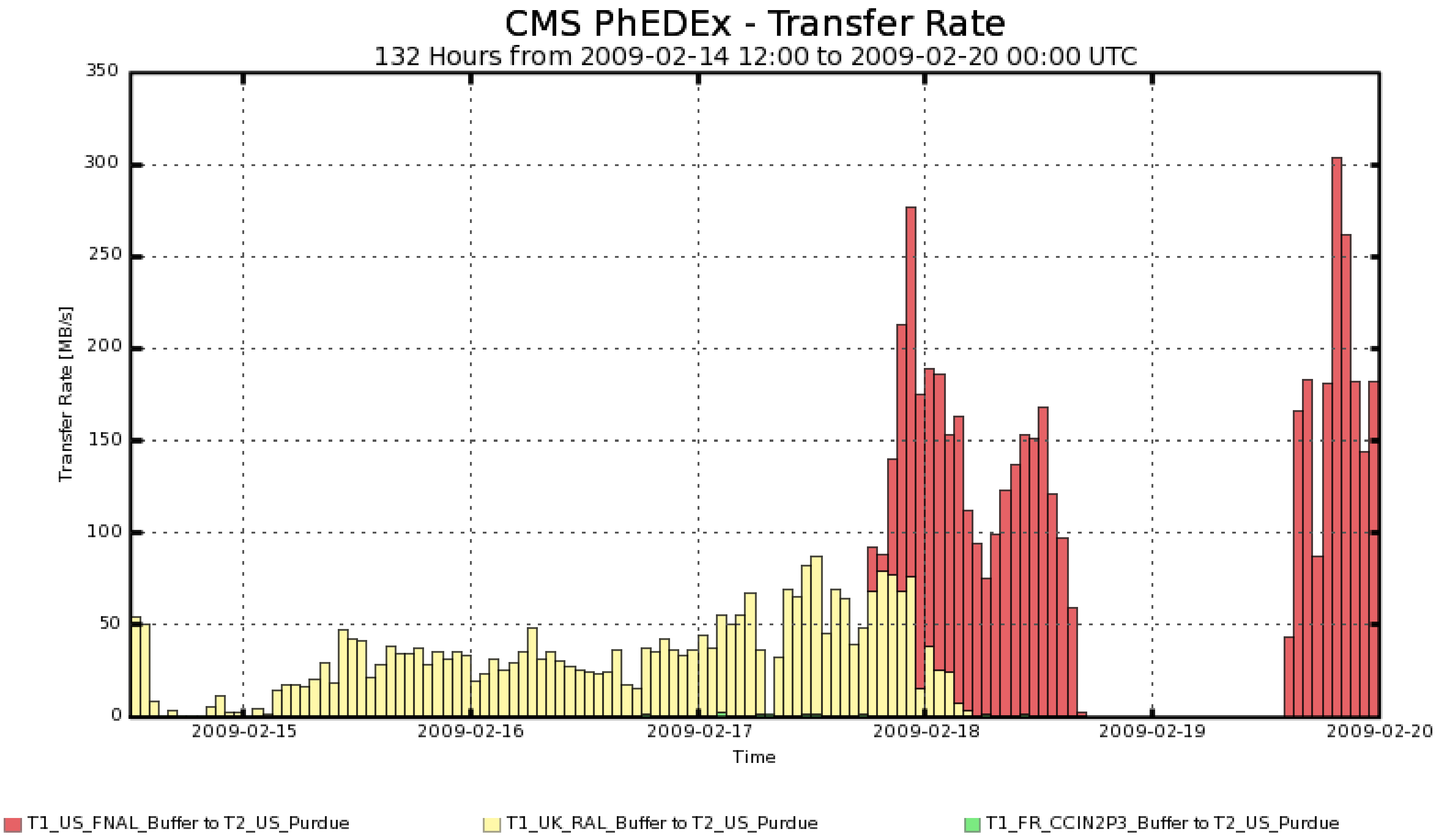
Tunings

- Multiple mover queues in each pool
  - Dcap uses LAN queue, gridftp use WAN queue
  - Slow processing dcap jobs don't clog up the fast gridftp request
  - Limit number of concurrent gridftp transfers to prevent pools from overloading
- Cost module setting left at defaults
- Misc timeouts mostly at defaults
- Network tuning
  - Set TCP buffers according to BDP value (BW * RTT)

Sun Fire X4500

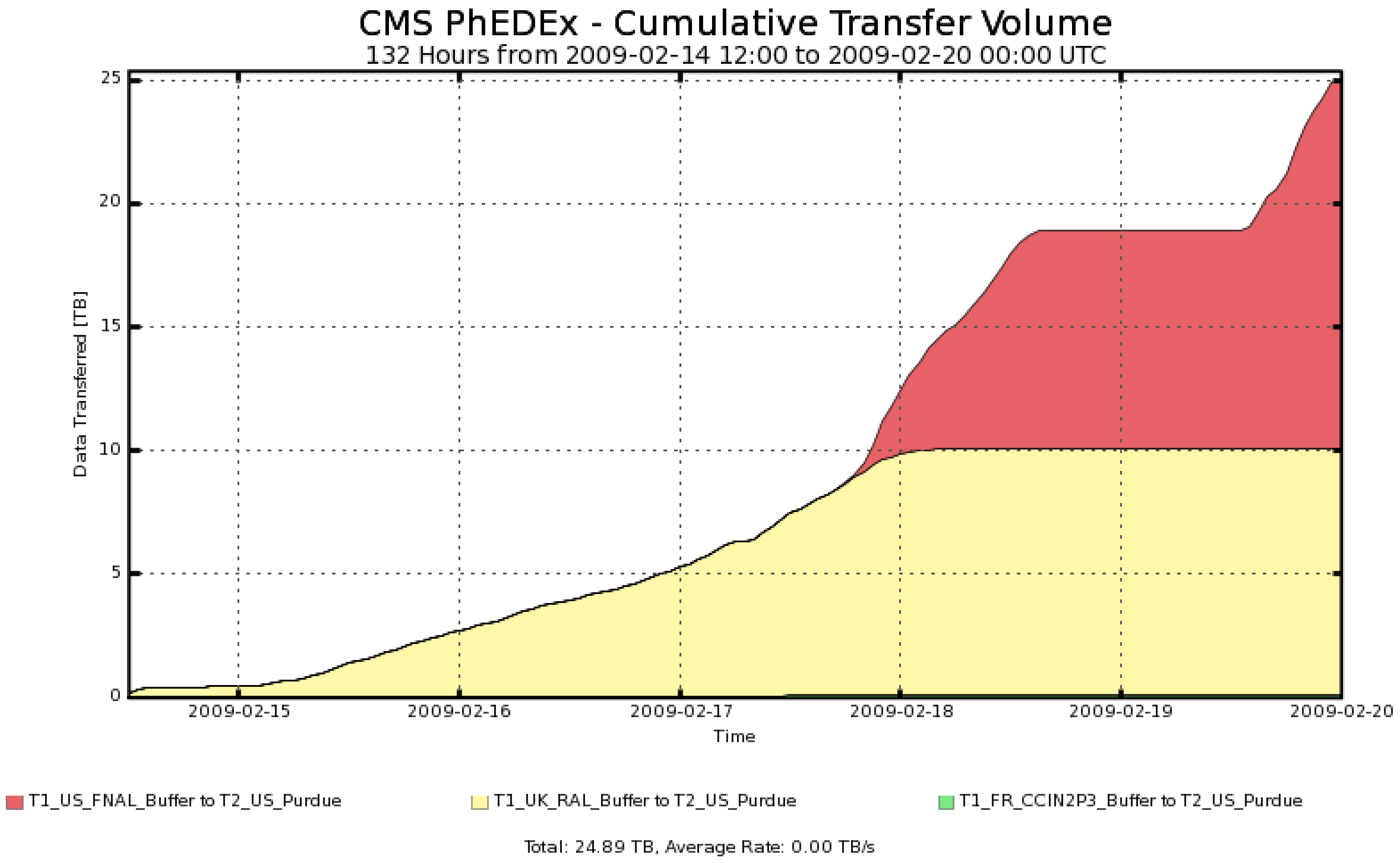
- Linux is currently the OS deployed
- RAID configuration in Linux
  - Create 6 software RAID 5 arrays
  - Concatenate the RAID arrays into 1 super big logical volume
- File system
  - XFS wins over EXT3 in performance (especially for write)
  - XFS and ZFS comparable in performance, and both have their own edges
- Dilemma
  - Solaris/ZFS have slightly better performance than Linux/XFS
  - The cost of adding another flavor in all all Linux environment can't be underestimated

Write Performance

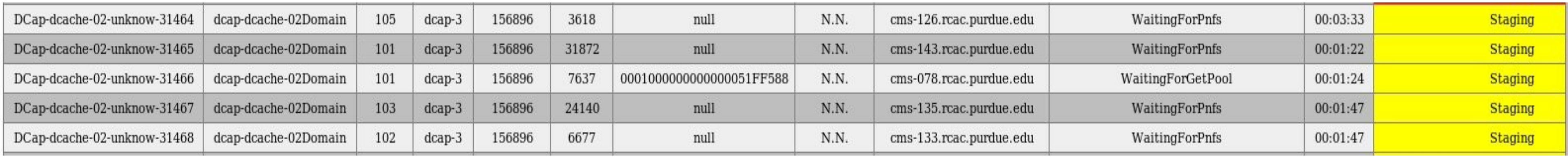


Maximum: 303.81 MB/s, Minimum: 0.00 MB/s, Average: 54.92 MB/s, Current: 181.74 MB/s

Write Performance

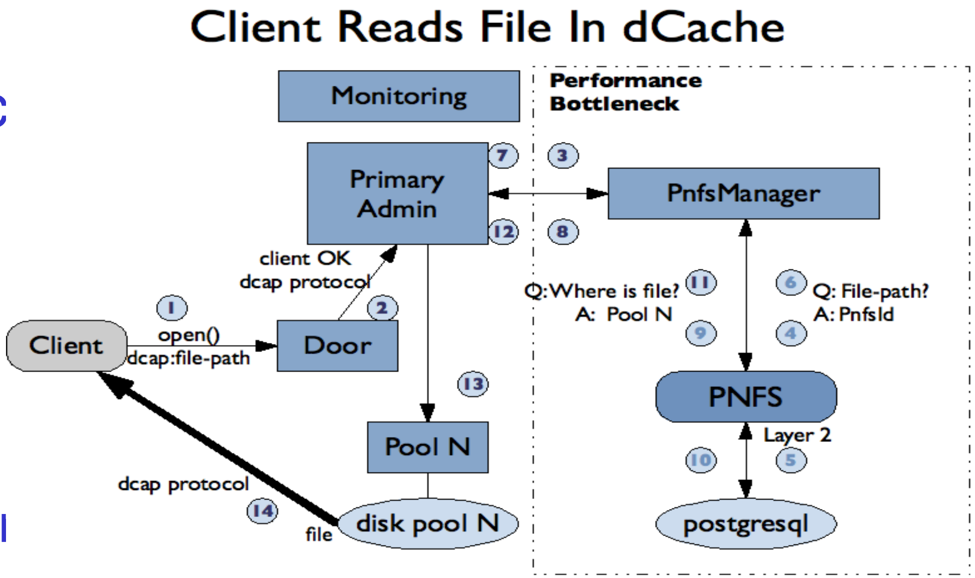


PNFS Bottleneck



DCap-dcache-02-unknow-31464	dcap-dcache-02Domain	105	dcap-3	156896	3618	null	N.N.	cms-126.rcac.purdue.edu	WaitingForPnfs	00:03:33	Staging
DCap-dcache-02-unknow-31465	dcap-dcache-02Domain	101	dcap-3	156896	31872	null	N.N.	cms-143.rcac.purdue.edu	WaitingForPnfs	00:01:22	Staging
DCap-dcache-02-unknow-31466	dcap-dcache-02Domain	101	dcap-3	156896	7637	0001000000000000051FF588	N.N.	cms-078.rcac.purdue.edu	WaitingForGetPool	00:01:24	Staging
DCap-dcache-02-unknow-31467	dcap-dcache-02Domain	103	dcap-3	156896	24140	null	N.N.	cms-135.rcac.purdue.edu	WaitingForPnfs	00:01:47	Staging
DCap-dcache-02-unknow-31468	dcap-dcache-02Domain	102	dcap-3	156896	6677	null	N.N.	cms-133.rcac.purdue.edu	WaitingForPnfs	00:01:47	Staging

- Causes
  - Single access point
  - PNFS server uses global loc on each database
- Fixes
  - Upgrade pnfs node
  - Optimize postgresql
  - Split database so that each stoi a sub-tree of the pnfs fs to parallize the access
  - Replace PNFS with Chimera?



Client Reads File In dCache

PNFS Caveats

- ”rmdir” fails to delete an empty dir
  - ”the number of entries in the directory does not correspond to the actual number of objects in it”--Vladimir (FNAL)
  - ”md3tool modifydirentrycount dbname pnfsid 0”
- A sequece that should be avoided
  1. Create a directory (dir A) and create tags for this directory
  2. Create a subdirectory (dir B) in dir A
  3. Move dir B out of dir A
  4. Remove dir A

Pool Filling Up until Offline

- Set pool size correctly (rule of thumb)
  - *set max diskspace (<Kbytes from 'df -k'> / 1024 / 1024 - 5)g*
- Pool accepts new write request as long as there are space available, without counting in potential space consumption of the current transfer
  - Just my suspicion. If so, it's a bug.
- Makeshift solution deployed
  - A full pool write protect cron job (adapted from WISC)

Miscs

- Suspended RCs
  - Destroy RCs for which file doesn't exist
  - Retry suspended RCs can put them through (quite often)
- Orphaned files
  - File on disk and in the pool, but not in the PNFS name space (out of sync)
  - A cron job run at admin node (2 time/week) delete the orphaned file
- Stuck srmcp myth
  - Restart one of the dcache pool which was online fixed the problem

Conclusion

- DCache prove can support the nominal requirement of a cms tier2
- DCache is quite flexible, reliable and powerful
  - Addressed data reliability and availability on inexpensive commodity disk through file replication
  - Fault tolerance and Load balance features
  - Supports many transfer protocols
  - Grid aware
- DCache has a wide user population in LCG sites, thus good support from peers are available
- DCache needs effort on tuning and configuration
- DCache is not problem-free like anything else that is evolving

Acknowledgement

Ken Bloom (UNL)

Neha Sharma (FNAL)

Norbert Neumeister (Purdue)

Thomas Hacker (Purdue)

Vladimir Podstavkov (FNAL)