

SC3 experiences

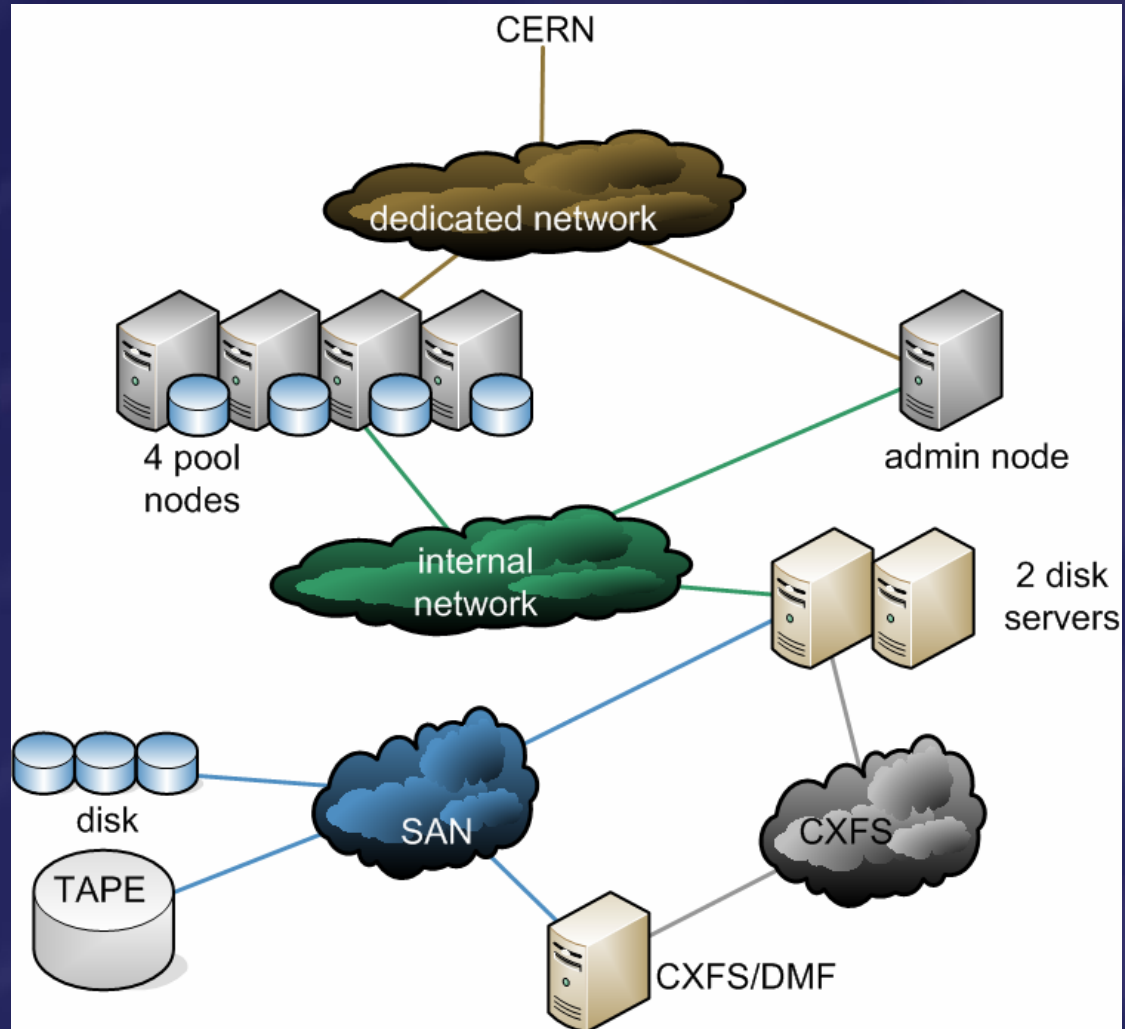
Ron Trompert

SARA

Why dCache?

- ▣ dCache provides an srm I/F
 - ▣ We use DMF in our HSM environment for which there is no SRM implementation
- ▣ dCache provides flexibility with respect to HSM backends
 - ▣ If we need to switch to another HSM setup for some reason
- ▣ dCache provides functionality promised by the SRM standard but not supported by DMF
 - ▣ File pinning

SC3 Infrastructure






SC3 infrastructure

- ▣ Pool nodes
 - ▣ 4x dual Opteron's, 4GB memory, 2x 1GE
 - ▣ 2TB disk cache, 12x 250GB SATA, 3ware RAID controller, disk I/O 200MB/s RAID0 (used during SC3) and 100MB/s RAID5, XFS
- ▣ Admin node
 - ▣ dual Xeon, 4GB memory, 2x 73GB internal disk, 2x 1GE
- ▣ MSS gateway nodes (disk servers)
 - ▣ 2x dual Xeon, 4GB memory, 2x 73GB internal disk, 2x 1GE, dual HBA FC, 1.6 TB CXFS filesystem (SAN shared filesystem)
 - ▣ runs CXFS client, read/write data directly to/from CXFS filesystem
 - ▣ and rfiio daemon to put/get data to/from pool nodes
- ▣ MSS server (CXFS/DMF)
 - ▣ 4 cpu R16K MIPS, 4GB memory, 12x FC, 4x GE, 2x 36GB internal disk, 1.6 TB CXFS filesystem (SAN shared filesystem), 3x STK 9940B tape drives
 - ▣ CXFS MDS server, regulates access to CXFS filesystem
 - ▣ DMF (Data Migration Facility = HSM system), migrates data from disk to tape and back
- ▣ Network
 - ▣ dedicated 10GE network between CERN – Amsterdam
 - ▣ GE internal network between pool nodes and MSS gateway nodes









dCache configuration

dCache 1.2.2-7-3

Admin node

-  ia32
-  SL304 with 2.4.21-32.0.1 kernel
-  Runs pnfs server, srm and gridftp door

Pool nodes

-  amd64
-  Debian (sarge) 2.6.8-10 kernel
-  Pool node s/w is in java
-  j2sdk 1.5.0
-  Got source rpm of CASTOR client to build rfio
-  Three minor issues encountered installing the dCache pool S/W.
-  Pools with XFS filesystem
-  Run gridftp door

dCache configuration

▣ The default number of I/O movers was 100

▣ Leads to very high loads
We have set it to approximately 5

▣ The default heartbeat was 120

▣ Leads to poor load balance over the pools.

A single transfer request with multiple files would dump all transfers on a single pool.

A pool may get a considerable number of transfers before other pools are taken into account

We have set it to 10

Number of streams per transfer

- Using the Globus gridftp server on a dedicated 10 Gb link with a small number of streams (1-2) is optimal (1 GB file, 50 MB/s, 1 stream)
- A 1 stream transfer with a dCache gridftp server leads to 1.6 MB/s for a 1 GB file. This performance scales almost linearly with the number of streams (1-10 streams -> 1.6 MB/s-16 MB/s)
=> probably an implementation issue and not networking.
- Bad for transparency which is desirable in a Grid environment.
- Kept it at 10 which is the default.

Tuning of kernel parameters on pool nodes

- ▣ `vm.lower_zone_protection = 200`
`vm.dirty_expire_centisecs = 250`
`vm.dirty_writeback_centisecs = 250`
`vm.dirty_ratio = 10`

SC3 Results

 Disk2disk: 100-110 MB/s

 Problems with stability of the nodes

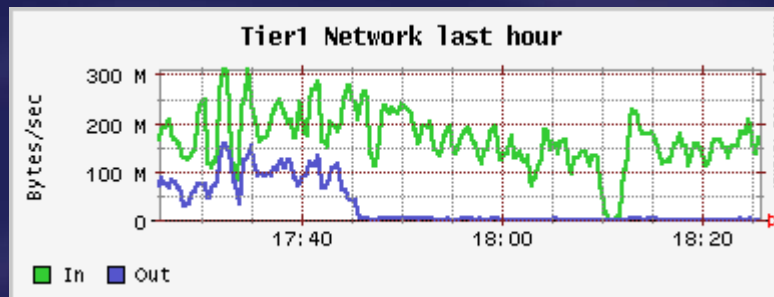
 Disk2tape: 50 MB/s

 Not enough bandwidth, SAN not dedicated




- srmPut requests lead to gridftp doors receiving files and passing them on to pool nodes.
 - Puts unnecessary load on nodes
 - Uses bandwidth which can be used for useful transfers.
 - FTS (srmPut) => 100-110 MB/s
srmcp (srmCopy) => 180 MB/s

SC3 observations

▣ Left srmPut, right srmCopy



Timeouts in returning turls.

-  getRequestStatus timeouts
-  Restarting dCache did not help by itself. Cleaning out the postgres db and restarting dCache did.
-  Happens when transfers are going on for a couple of days.

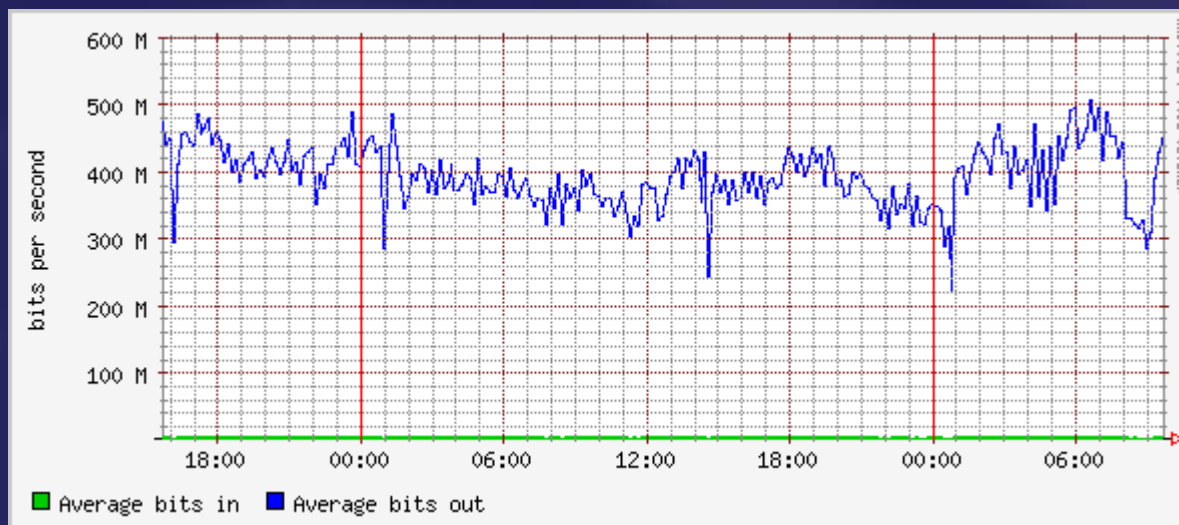
SC3 observations

- ▣ With a full disk pool everything kept running happily

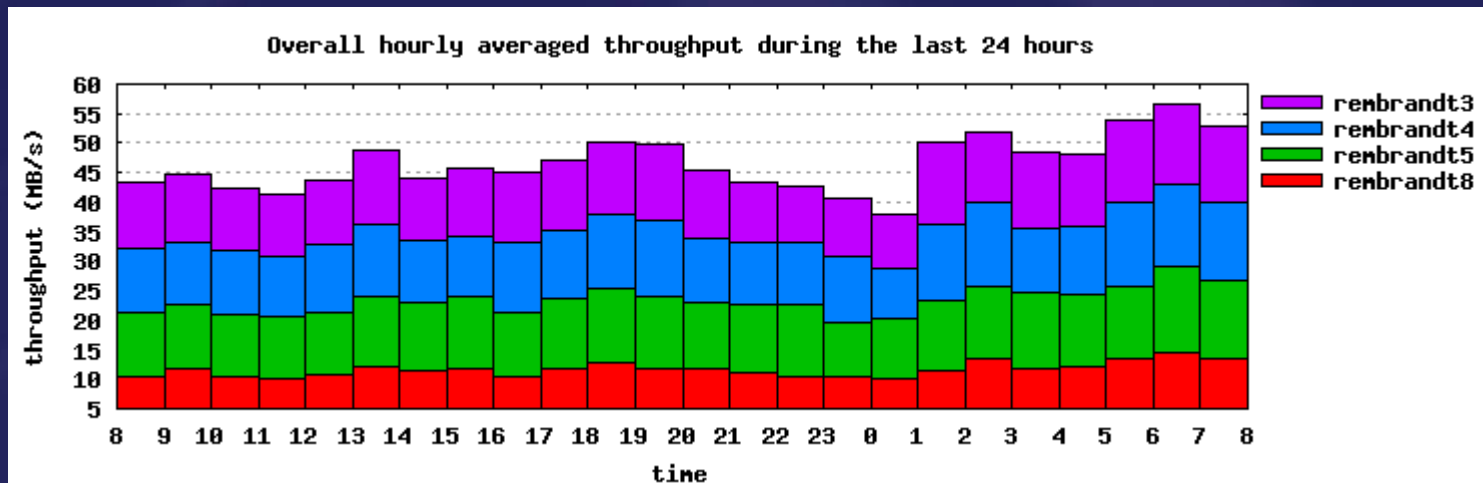


Dips in network traffic

- dCache if fine. There is no relationship with events in the gridftp logs and srm logs with the dips. Also no relation with dips and failed transfers

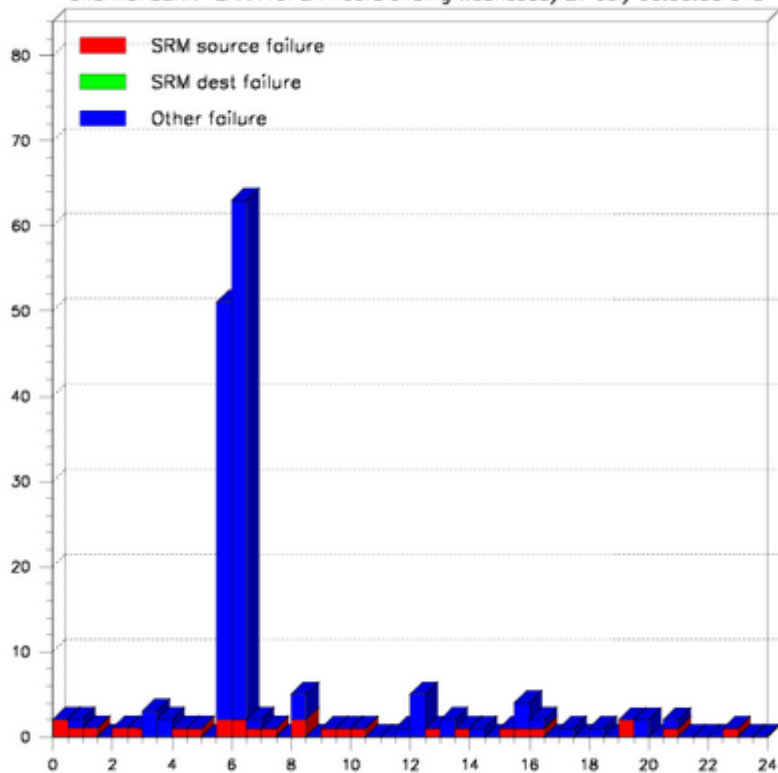


- With a constant number of files no constant throughput

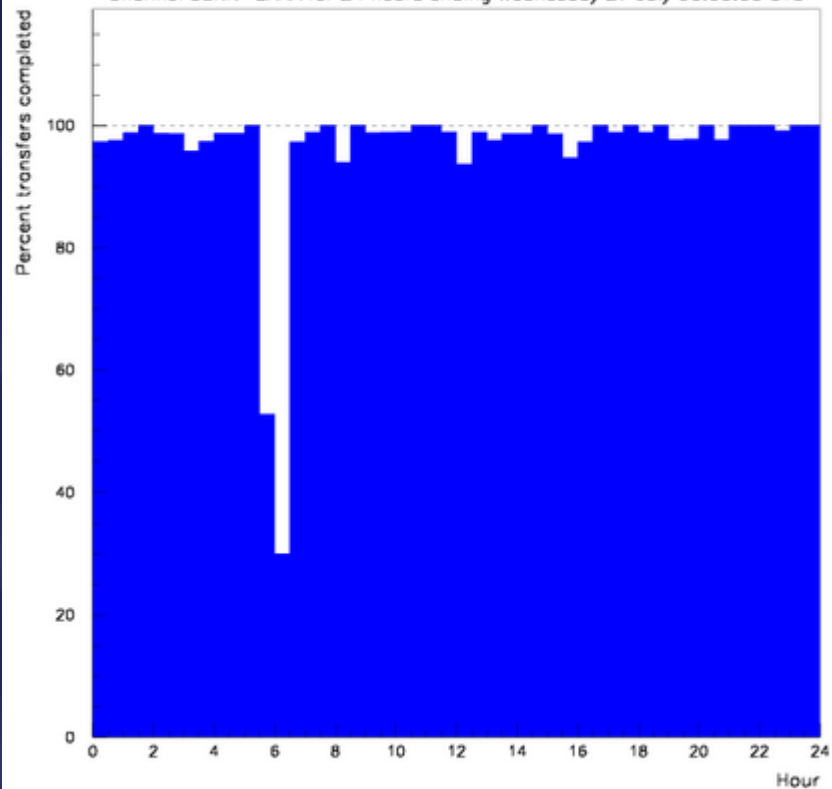


SC3 observations



Channel CERN-SARA for 24 hours ending Wednesday 27 July 06.00.00 UTC



Channel CERN-SARA for 24 hours ending Wednesday 27 July 06.00.00 UTC



Uuid is not always unique

-  Sometimes failed transfers because of attempt to overwrite an existing file
-  Adding a timestamp to the file name solved this

Post SC3 tests

- ▣ dCache 1.6.5-1
- ▣ No gridftp door on admin node

Crash tests

- ▣ dCache 1.6.5-1
- ▣ 5 I/O movers per pool
- ▣ Normal shutdown of one pool node
- ▣ Kill -9 -1 as root on pool node
- ▣ Genuine crash of a pool node due to overload. Thanks to the CERN pe☺ple.
 - ▣ Max. number of I/O movers is 5.
 - ▣ Max. number of files is 6 for the shutdown and nthe kill -9 -1 crash, 20 for the overload crash.

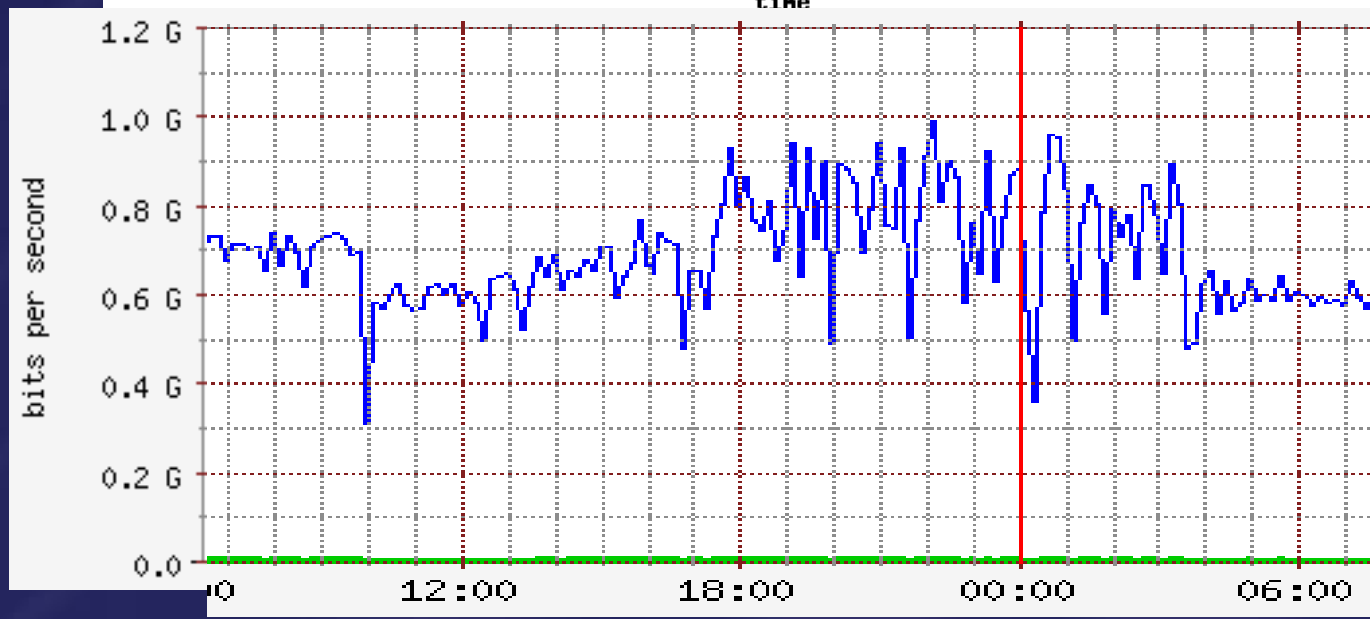
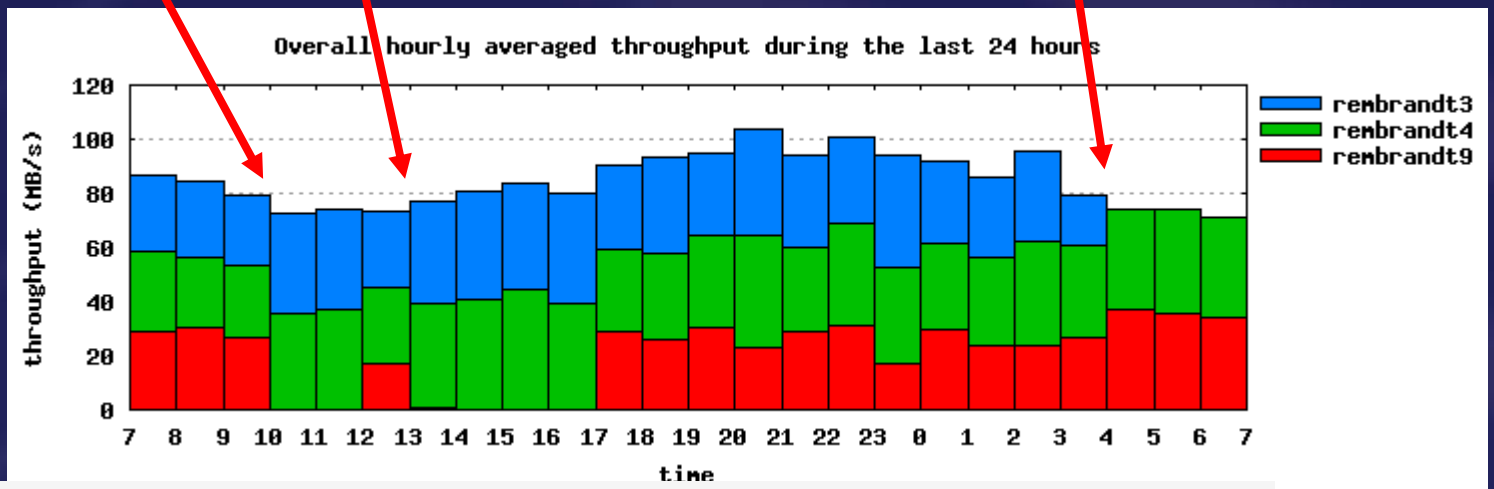


Crash tests

shutdown

Kill -9 -1

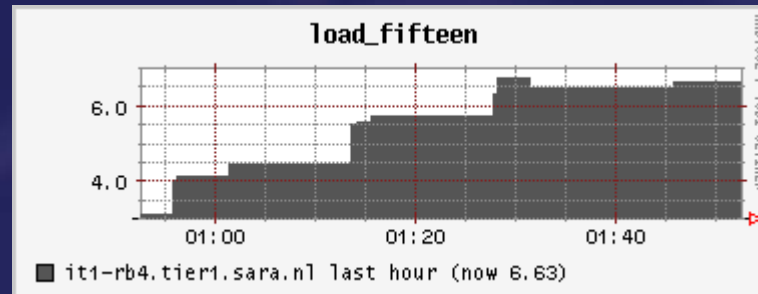
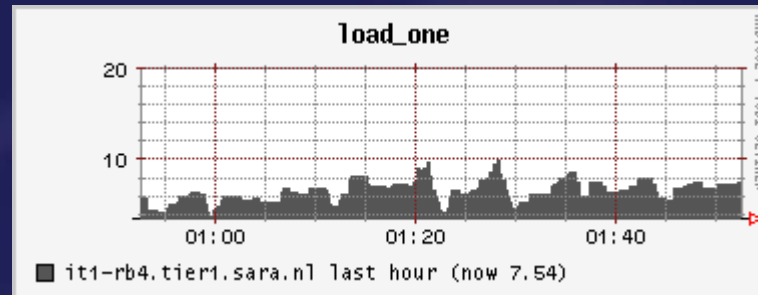
real crash



Stress tests

- ▣ Gradually increase the number of files from 4-40.
- ▣ For each pool:
 - ▣ Max. number of I/O movers = 2
 - ▣ Max. number of store movers = 3

Stress tests



Disk2disk

Test1

- ▶ 4 I/O movers
- ▶ No copies to CXFS disk servers
- ▶ 30 files

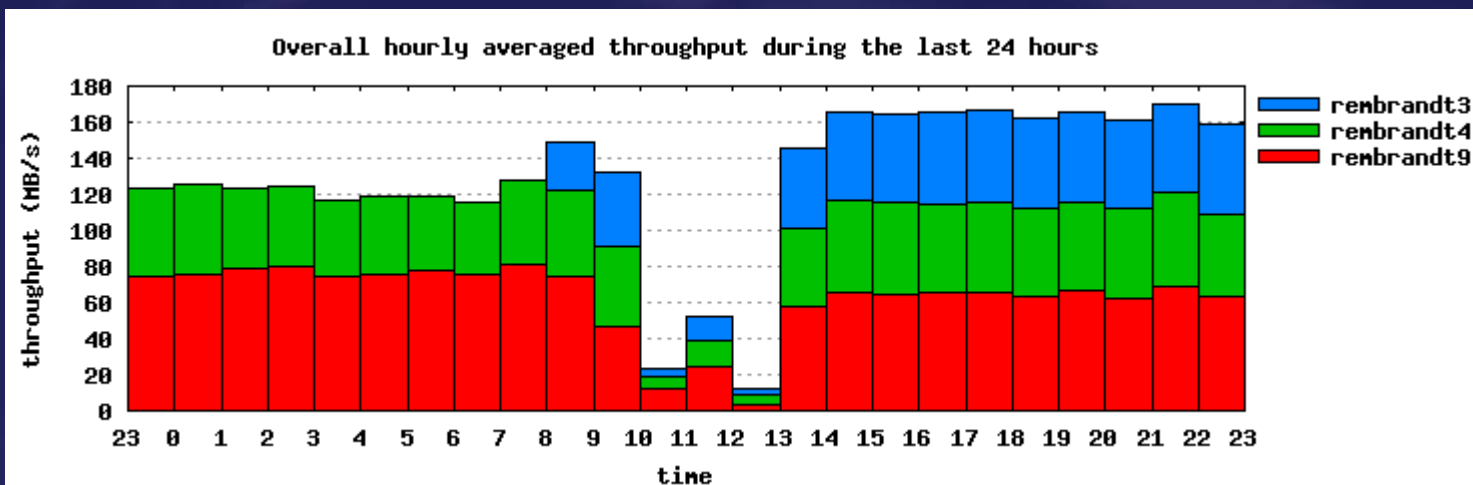
Test2

- ▶ 2 I/O movers, 2 store movers
- ▶ Copies to CXFS disk servers
- ▶ 30 files

Pool Request Queues

CellName	DomainName	Movers		
		Active	Max	Queued
Total		12	32	17
rembrandt3_1	rembrandt3Domain	4	4	4
rembrandt3_2	rembrandt3Domain	0	20	0
rembrandt4_1	rembrandt4Domain	4	4	8
rembrandt9_1	rembrandt9Domain	4	4	5
Total		12	32	17
CellName	DomainName	Active	Max	Queued
		Movers		

Performance tests: test1





Performance tests: test2

Pool Request Queues

CellName	DomainName	Movers			Restores			Stores		
		Active	Max	Queued	Active	Max	Queued	Active	Max	Queued
Total		6	26	20	0	8	0	6	8	10
rembrandt3_1	rembrandt3Domain	2	2	7	0	2	0	2	2	9
rembrandt3_2	rembrandt3Domain	0	20	0	0	2	0	0	2	0
rembrandt4_1	rembrandt4Domain	2	2	7	0	2	0	2	2	0
rembrandt9_1	rembrandt9Domain	2	2	6	0	2	0	2	2	1
Total		6	26	20	0	8	0	6	8	10
CellName	DomainName	Active	Max	Queued	Active	Max	Queued	Active	Max	Queued
		Movers			Restores			Stores		

Performance tests: test2

