

Distributed cluster dynamic storage:
a comparison of dcache, xrootd,
slashgrid storage systems running on
batch nodes

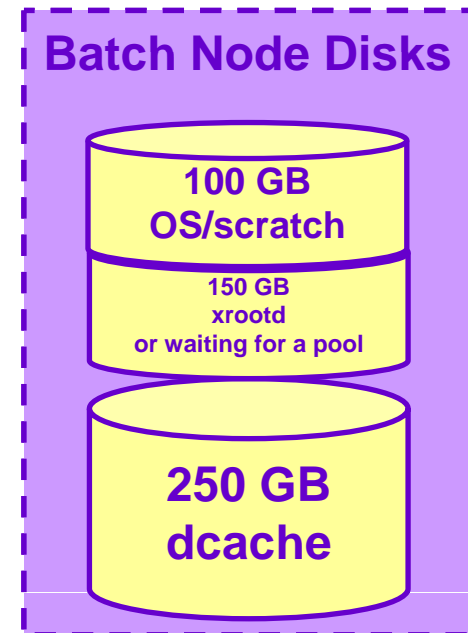
Alessandra Forti
CHEP07, Victoria

Outline

- Hardware
- Storage setup
- Comparison Table
- Some tests results
- Conclusions

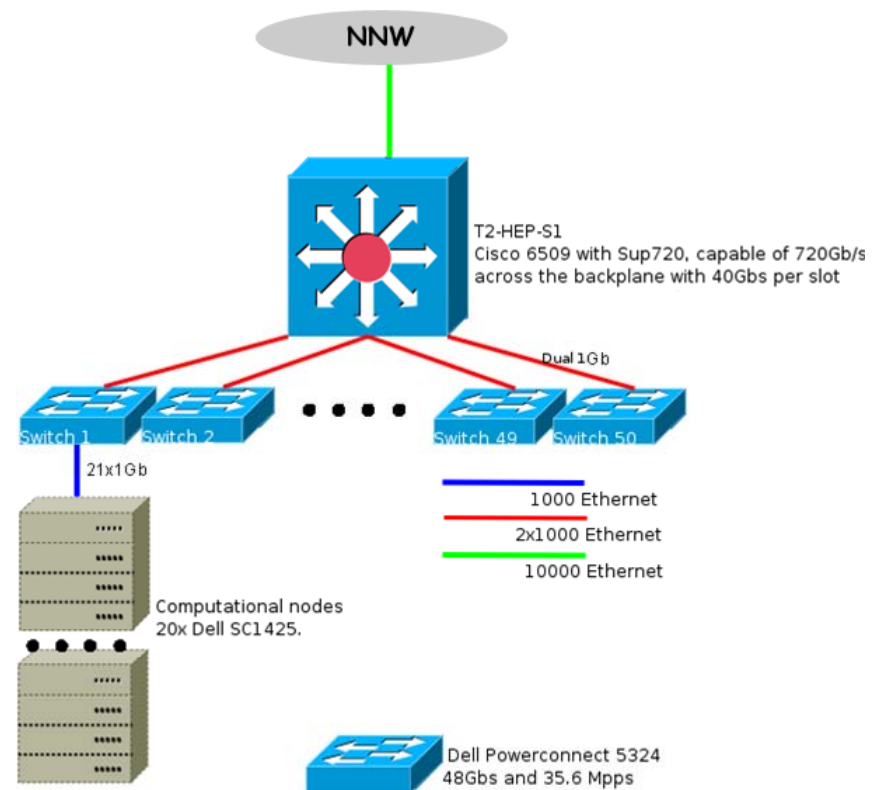
Manchester Nodes

- 900 nodes
- 2X2.8 GHz
- 2X2 GB RAMs
- 2x250 GB disks
 - Total space available for data ~350TB
 - Disk transfer speed ~470 Mb/s
 - specs and benchmarked
 - WN disks are NOT RAIDed
 - disk 1: OS+scratch+data
 - disk 2: data
- No tape storage.
 - Nor other more permanent storage as RAIDed disk servers.
- Nodes divided in two identical, independent clusters.
 - Almost!
 - Head nodes have the same specs as the nodes.



Cluster Network

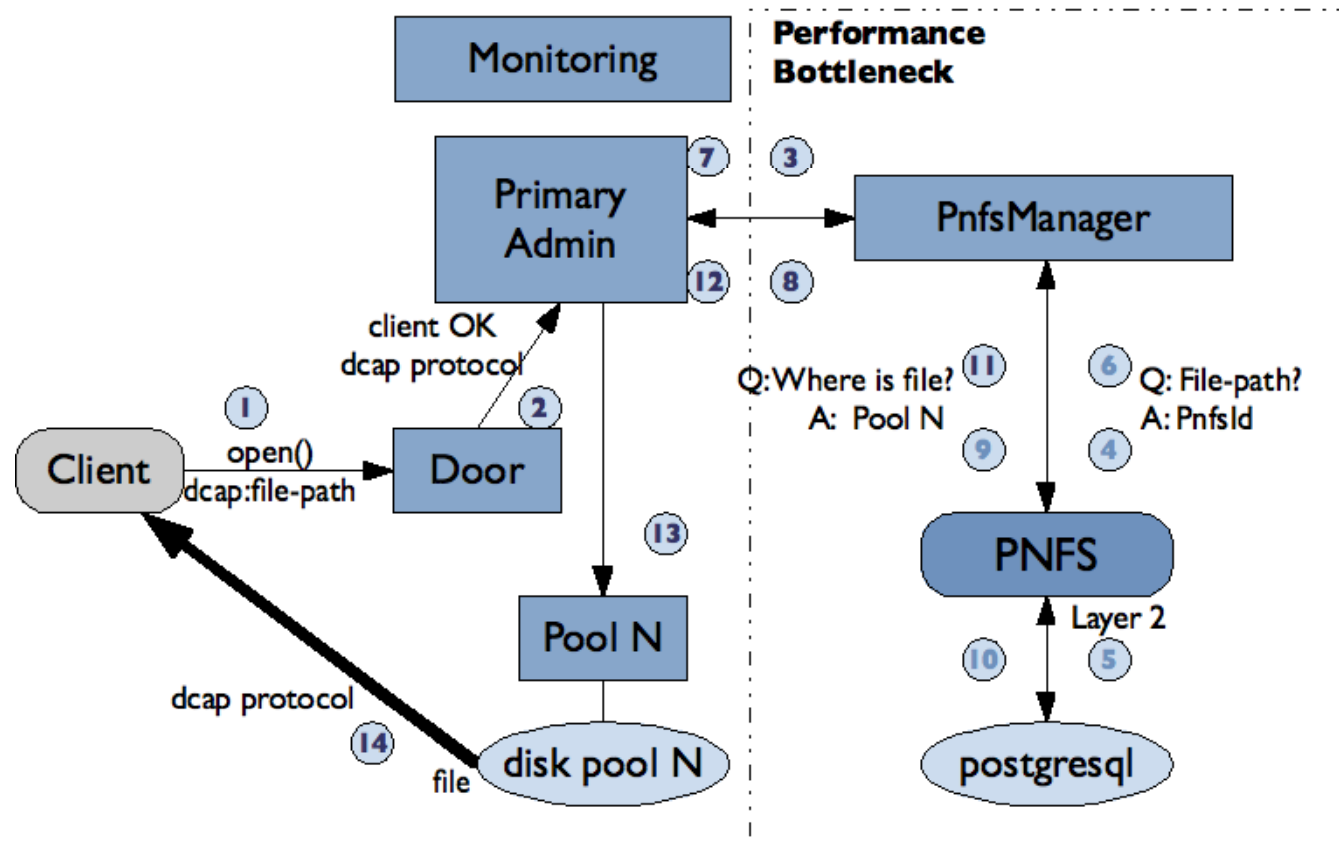
- 50 racks
 - Each rack
 - 20 nodes
 - 1 node 2x1 Gb/s
 - 1 switch 2x1 Gb/s to central switch
- Central switch
 - 10 Gb/s to regional network



dCache communication

OSG twiki

Client Reads File In dCache



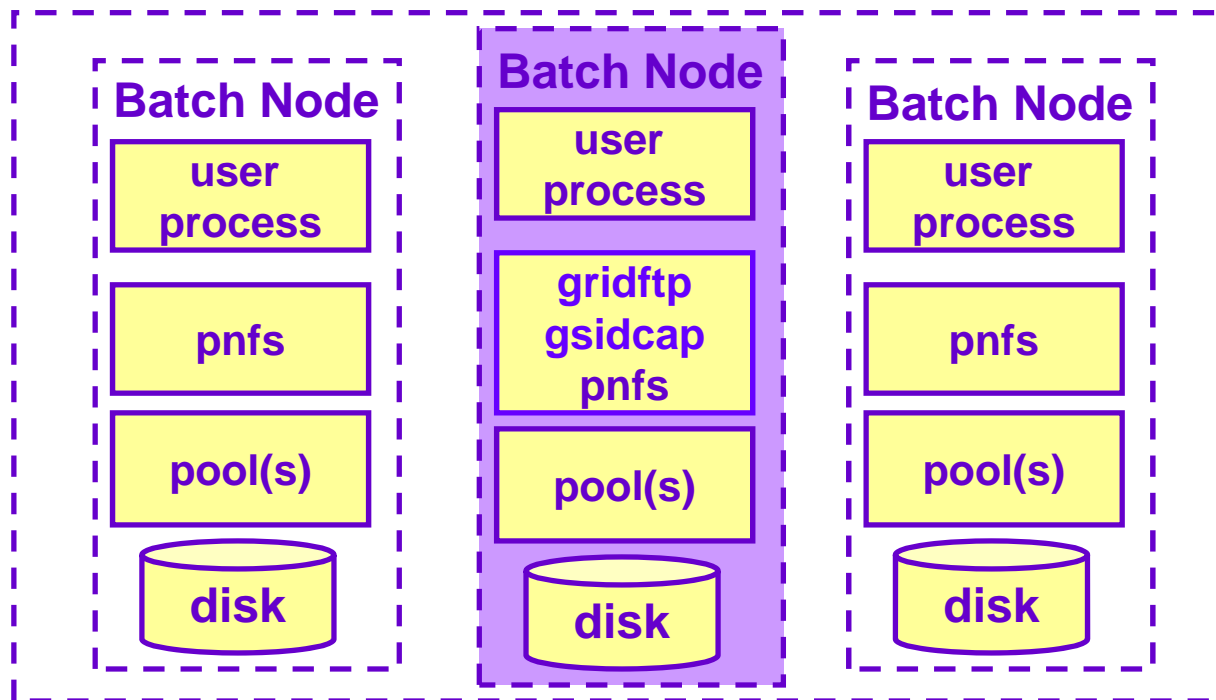
dcache setup

- dcache on all the nodes
- at least 2 permanent open connections on each node with the head node = ~900 connections per head

Head Node

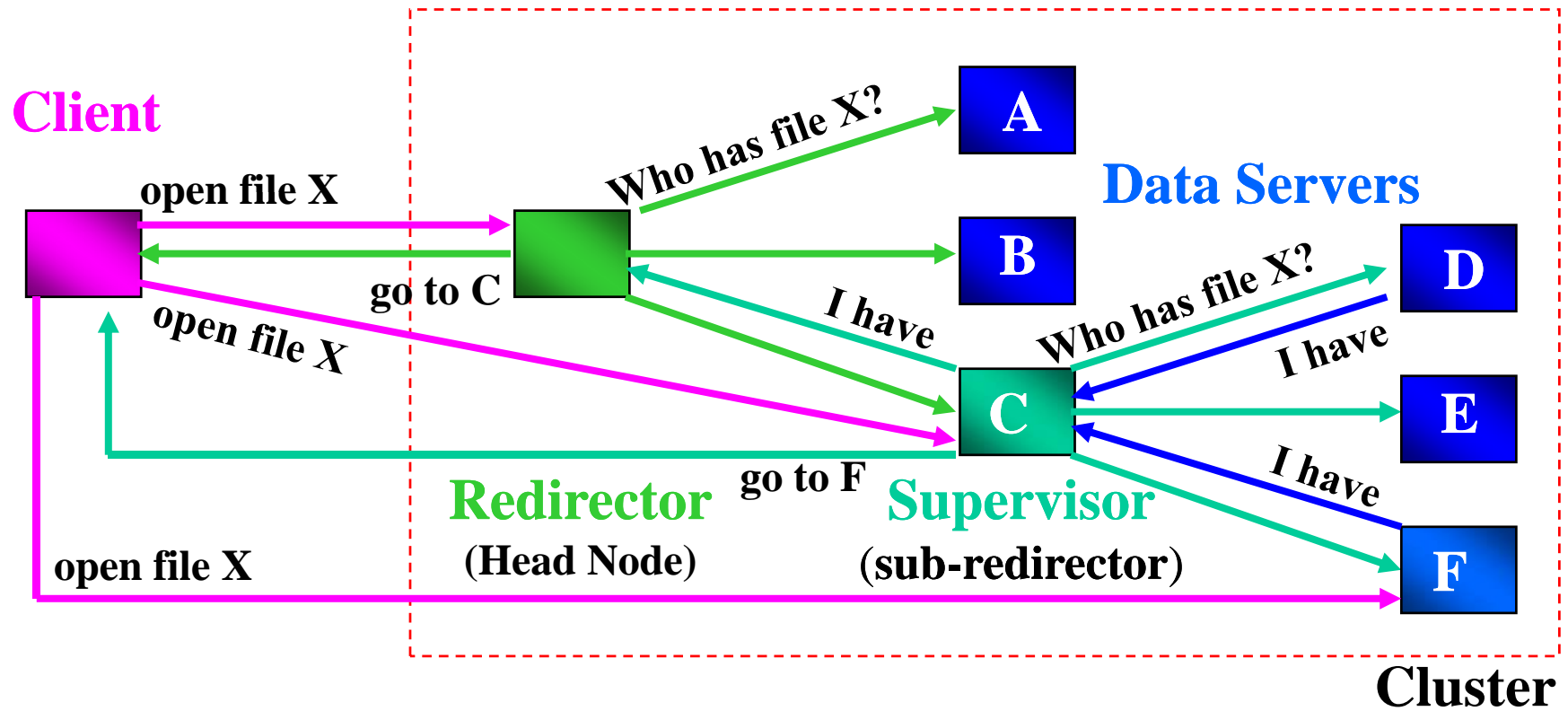
srn
pnfs server
admin

Rack1



Xrootd communication

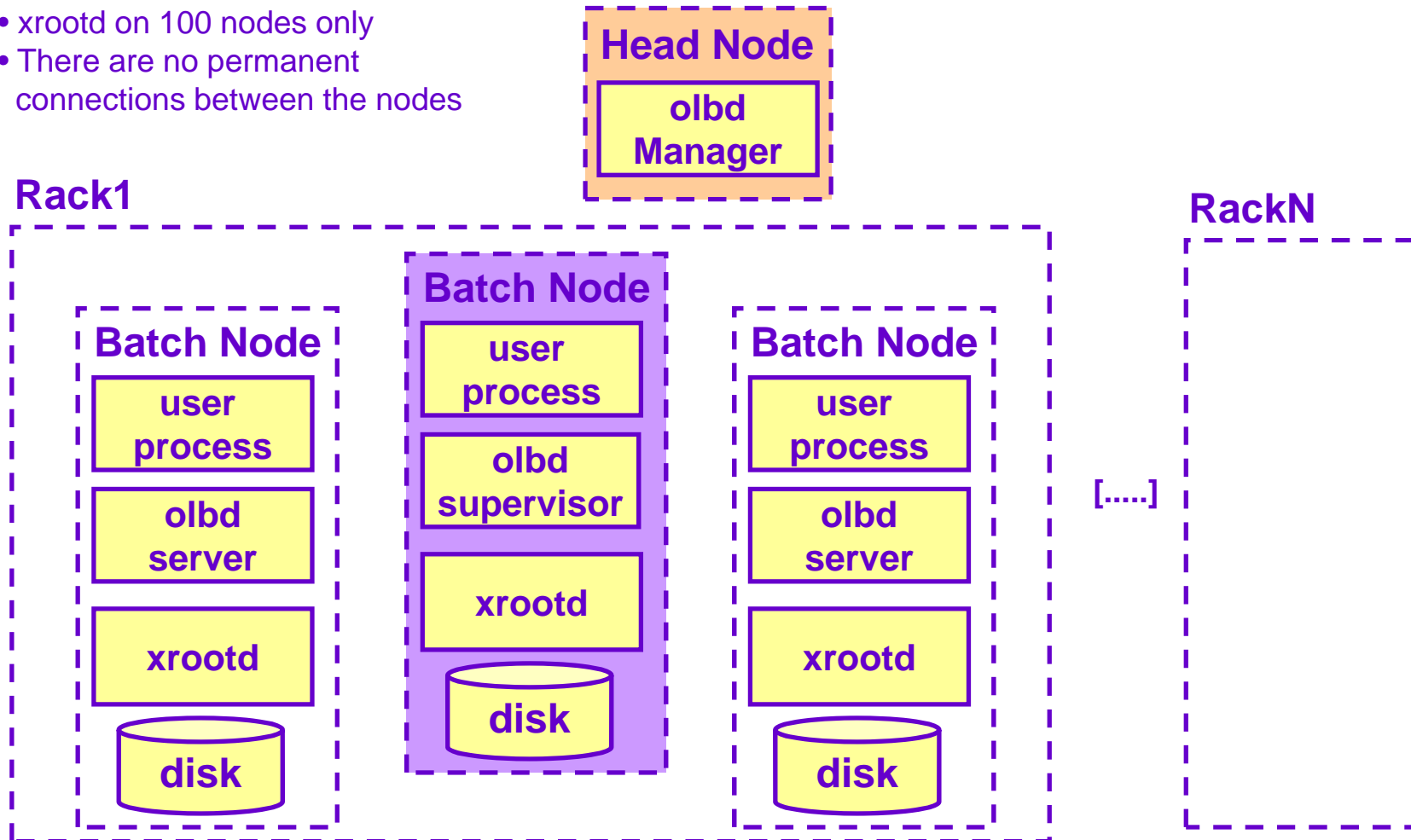
A. Hanushevsky



Client sees all servers as xrootd data servers

Xrootd setup

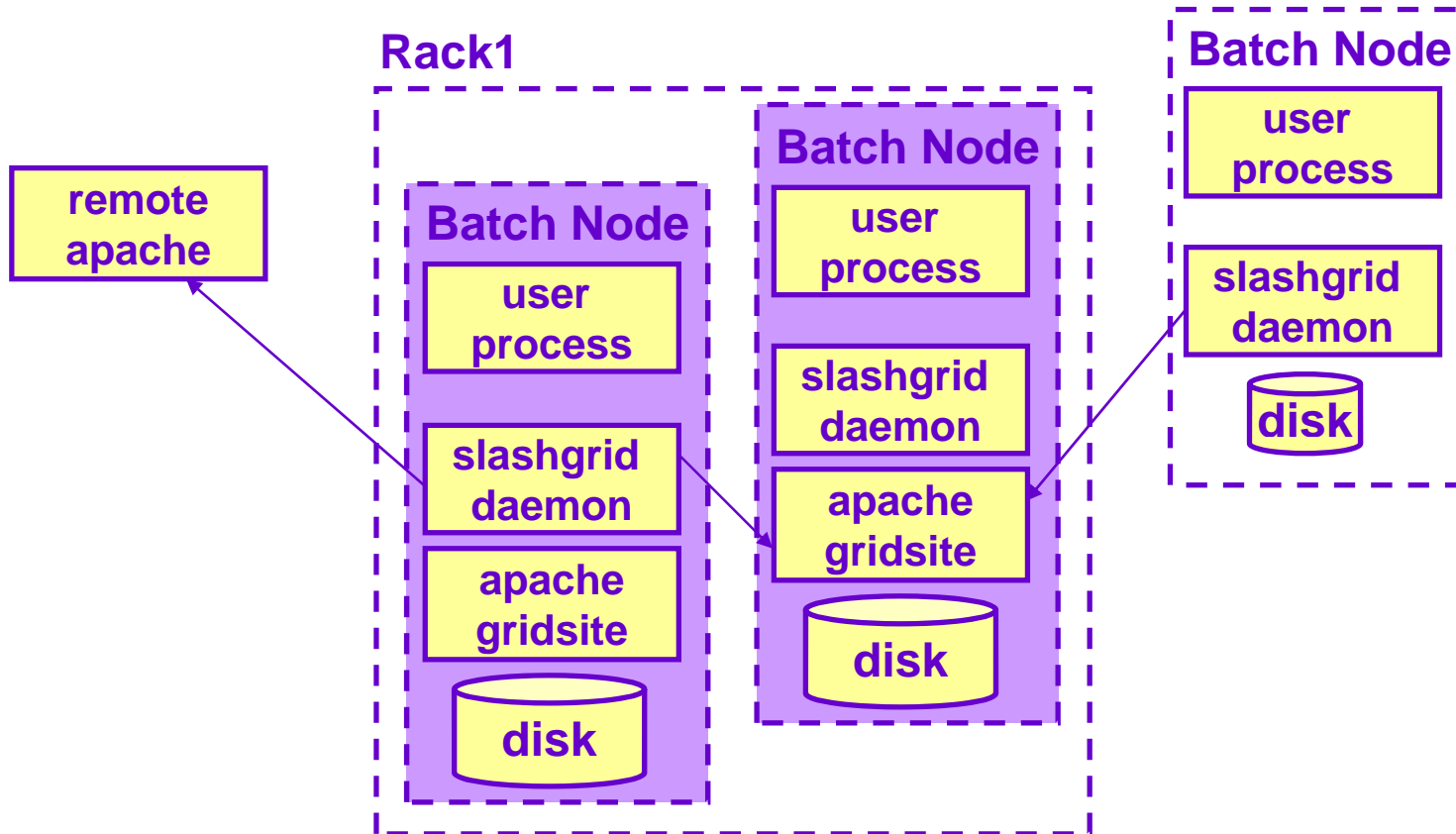
- xrootd on 100 nodes only
- There are no permanent connections between the nodes



Slashgrid and http/gridsite

- slashgrid is a shared file system based on http.
 - ls /grid/<http-server-name>/dir
- Main target applications are
 - Input sand box
 - Final analysis of small ntuples
- It was develop as a light weight alternative to afs.
 - It's still in testing phase and is installed only on 2 nodes in Manchester
- For more information see poster
 - <http://indico.cern.ch/contributionDisplay.py?contribId=103&sessionId=21&confId=3580>
- Although there is the possibility of an xrootd like architecture, in the tests it was used in the simplest way.
 - client contacting data server directly without any type of redirection.
- Transfer tests didn't involve /grid but htcp over http/gridsite
- User analisys test where done reading from /grid

Slashgrid setup



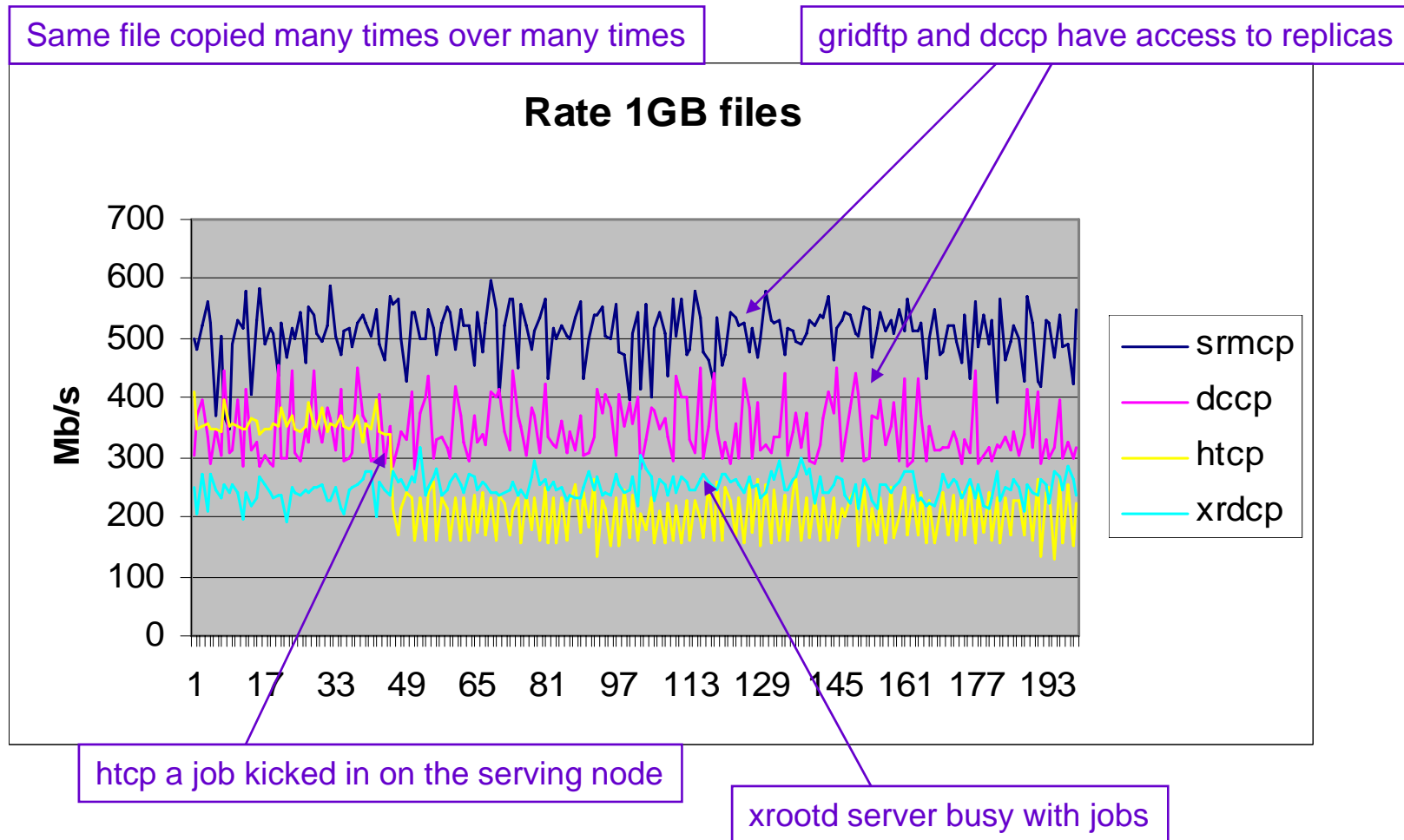
Comparison table

| | dcache | xrootd | slashgrid http/gridsite |
|---------------------------------------|--|-------------------|----------------------------|
| rpms | 2 | 1 | 2+4 |
| config files | 36 | 1 | 1 |
| log files | 20 | 1 | 1 |
| databases to manage | 2 | 0 | 0 |
| srm | yes | no | no |
| resilience | yes | no | no |
| load balancing | yes | yes | yes |
| gsi authentication and VOMS compat | yes | no | yes |
| configuration tools | yaim for EGEE sites VDT for OSG sites | not really needed | not really needed |
| name space | /pnfs | none | /grid |
| number of protocols supported | 5 | 1 | 3 |

Some tests results

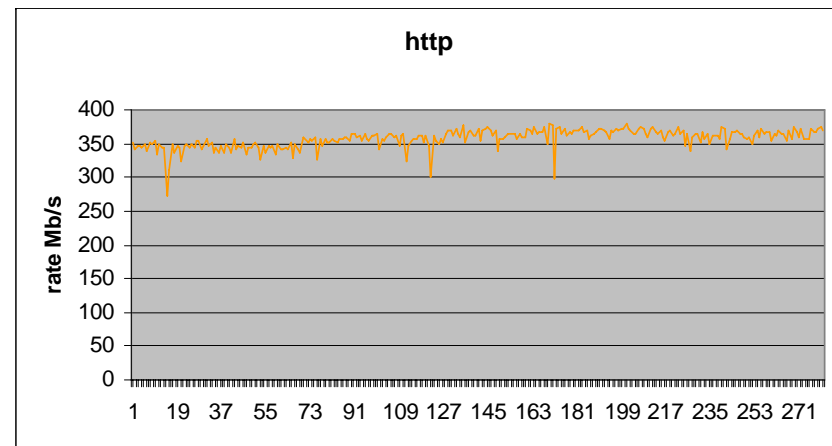
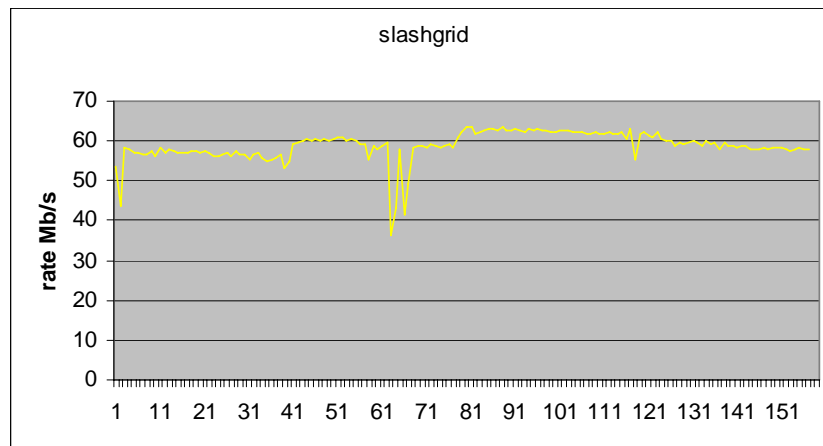
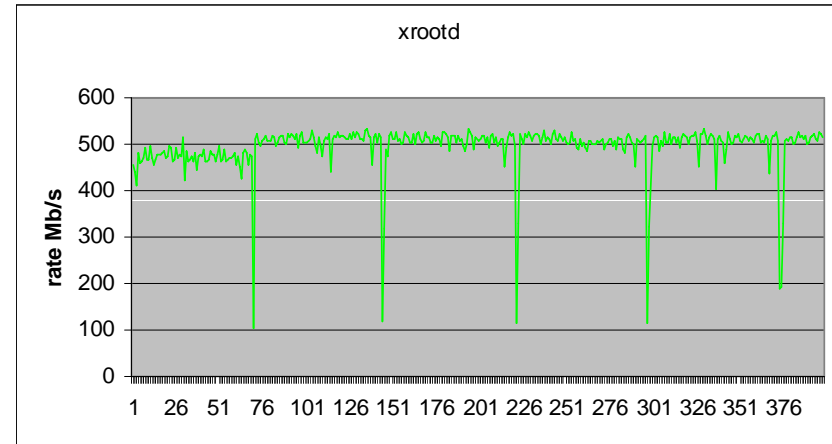
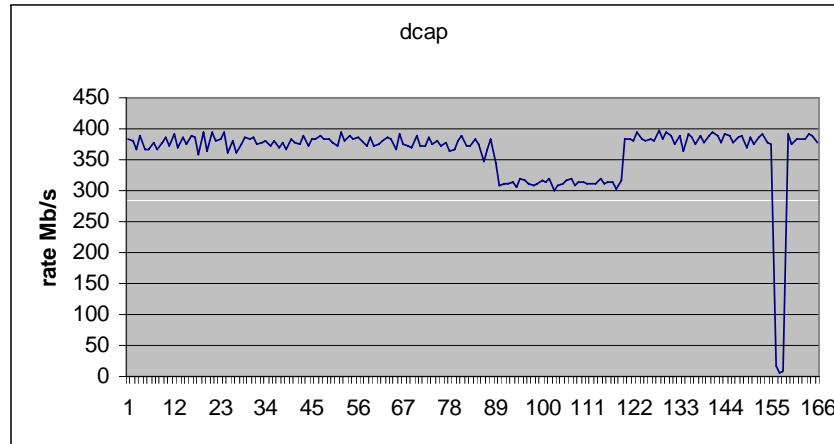
- Basic tests
- Transfers:
 - dccp, srmcp, xrdcp, htcp
- User analysis job over a set of ntuples:
 - dcache, xrootd, slashgrid, root/http, afs
- htcp in different conditions
- srmcp strange behaviour
- BaBar skim production jobs
 - A real life application
- AFS vs slashgrid
 - real time against user time

Transfer tools

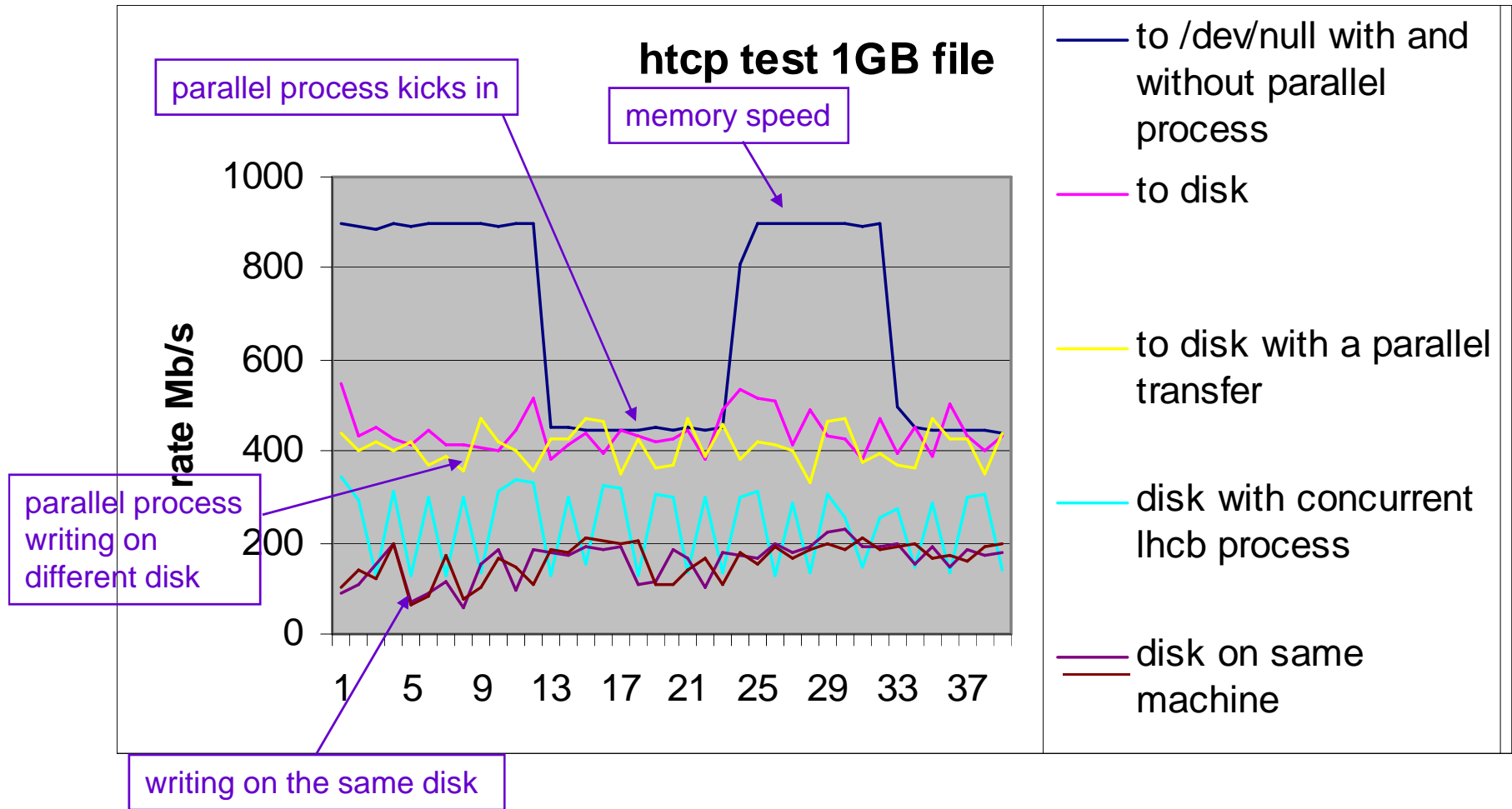


User analysis

Same set of small files copied many times: 29 files ~79MB each

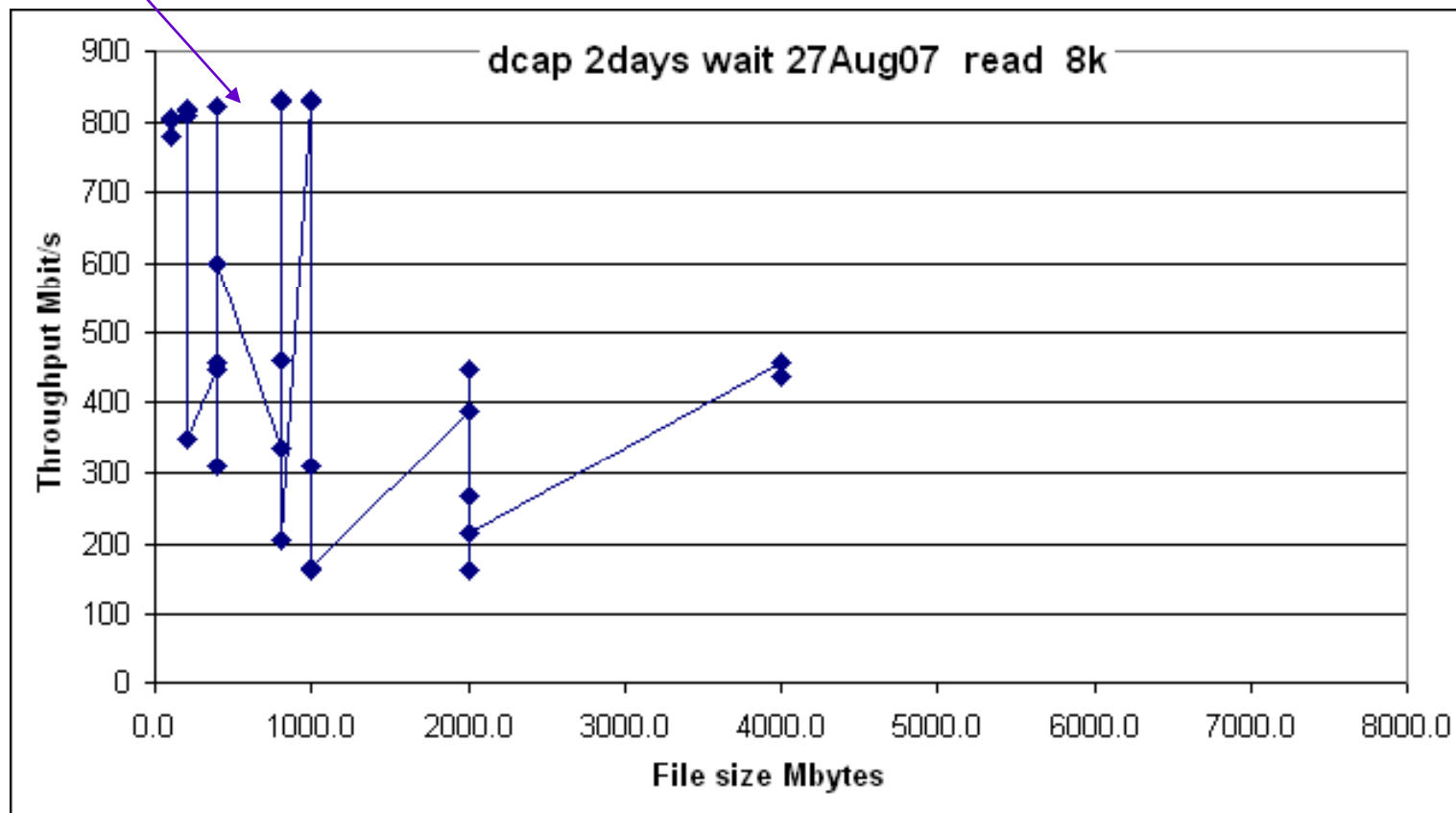


htcp tests 1GB files



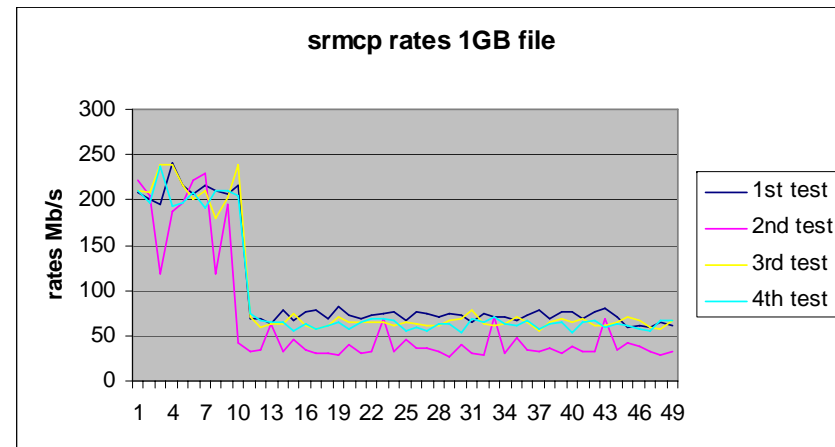
Dcap API

same as http transferring data from memory



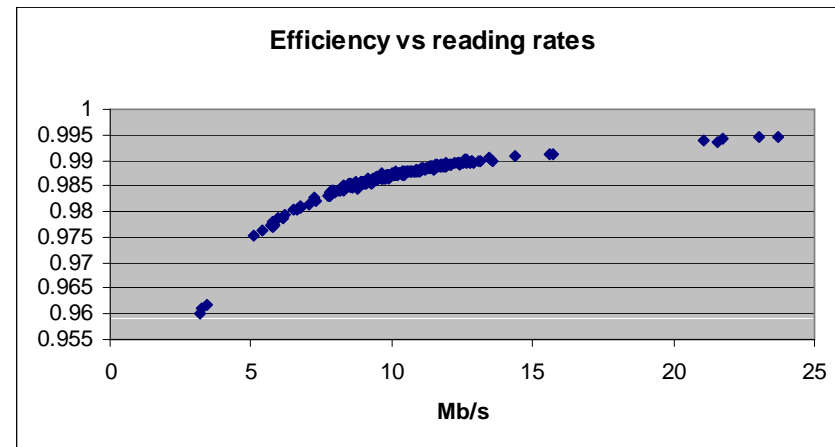
srmcp tests 1GB files

- list of 5 1GB files
 - files on different pools
 - files replicated
- 1st test:
 - copy 10 times the each file for 5 times
- 2nd test:
 - copy each file once for 5 times
- 3rd test:
 - same as the first one
- 4th test:
 - the same as the first one in different sequence
- Drop in efficiency after the first loop in each case
- Cluster empty no concurrent processes to explain the drop.
 - Needs more investigation



BaBar skim production

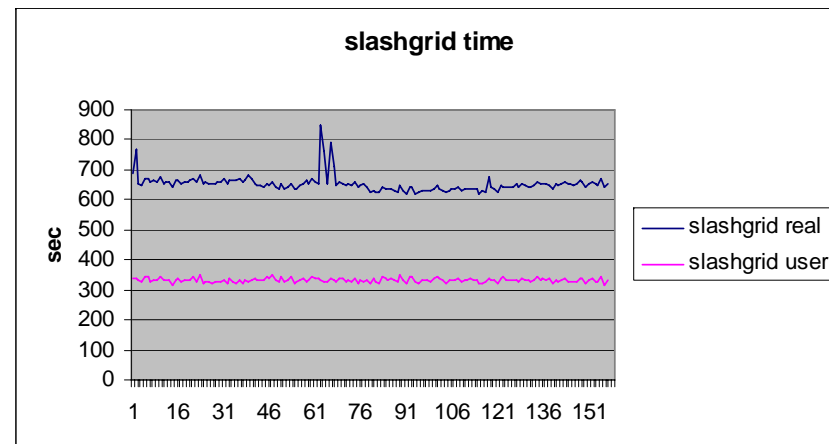
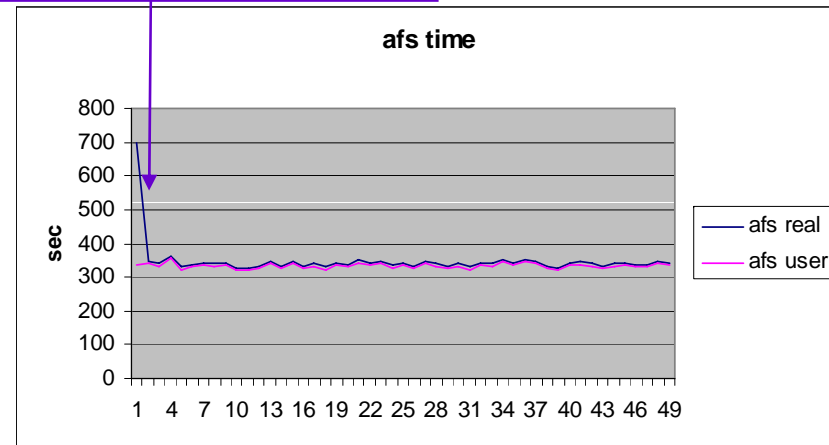
- average reading rate 10 Mb/s
- two files in input
- reading non sequential
 - subset of events
 - jumps from 1 file to another
- average job efficiency (cpu time/elapsed) is 98.6%
- Even increasing the speed of a factor 10 wouldn't change the job efficiency!!



afs vs slashgrid

- Manchester department uses AFS
 - Local experiment software installation
 - User shared data
- AFS measures for comparison with slashgrid.
 - cache smaller than the amount of data read
 - the job was reading from disk after the first time all the same.
- Slashgrid in not designed to do it.

AFS copying from cache



Conclusions

- dcache more complicated, difficult to manage, but has 3 features difficult to beat
 - resilience
 - srm front end
 - 5 different protocols
- xrootd is elegant and simple but all the data management part is in the users/admin hands and the lack of an SRM front end makes it unappealing for the grid community.
- slashgrid could be useful for software distribution
 - for data reading is still too slow.
 - over AFS it has the advantage of being easy to install and maintain.
- Speed within the cluster is comparable among different protocols
 - Applications like BaBar skim production demonstrate that a very high speed is not required.
 - However more measurements are needed to better understand different behavior especially when cluster features enter the equation.