I/O benchmarking

Martin Hellmich
on behalf of the LCGDM development team

Introduction

- New front-ends to DPM
  - HTTP
  - NFS
- Compare with existing protocols
  - RFIO
  - GridFTP
  - Xroot
- Verify modifications

Possible approaches

- Download/upload the full file and get the throughput
  - Playing with multiple streams, buffer size, etc...
- Remote I/O
  - Purely random
  - Real use cases from existing frameworks
- Stress tests for both approaches

Current coverage

- Downloading the full file
  - With and without multiple streams
- Pure random I/O
  - The seed is a parameter, so the same access pattern can be reproduced between different protocols
  - Does vector reads
  - Randomize offsets, number of elements and their size
  - Not completely fair, since real use cases have patterns that can be used to optimize the access
- Real use cases
  - Athena, including stress testing
  - CMSSW benchmark

Perfsuite Structure

- Perfsuite is a test framework that orchestrates and summarizes the execution of one or several tests
  - Which can run independently
- It is able to launch multiple “clients” at the same time
  - Multiple threads
  - Multiple physical machines
- The mentioned tests (except Hammercloud) work with these framework

Test Configuration

DPNS_HOME=/dpm/cern.ch/home/dteam

test_dpns_get(f:200,c:1,v:true,s:true)x10

test_dpns_full(f:500,c:{10 20},s:true)x2

test_dpns_get(f:200,c:1,v:true,s:true)
test_dpns_get(f:200,c:1,v:true,s:true)

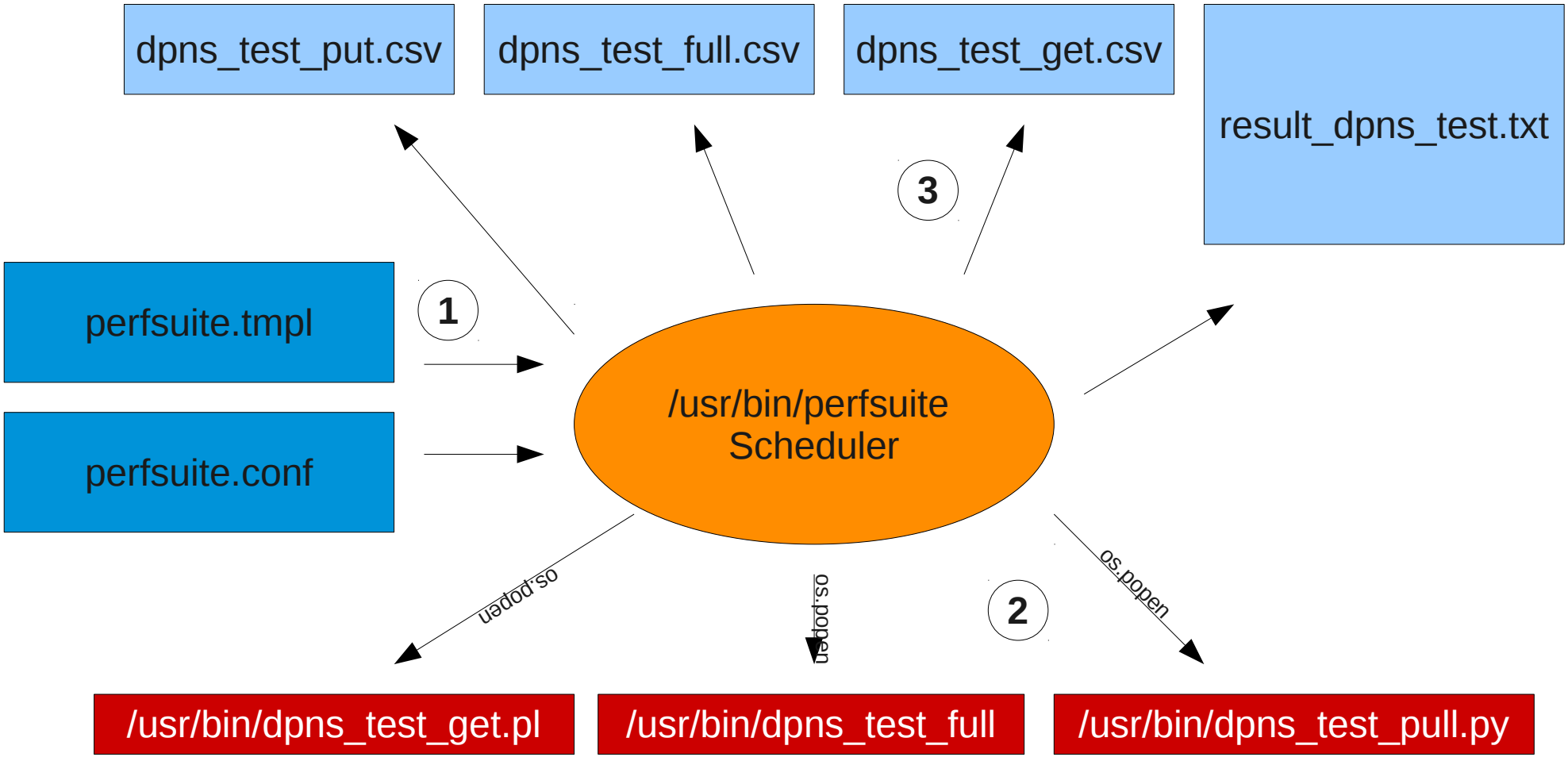
...
10
...

test_dpns_get(f:200,c:1,v:true,s:true)

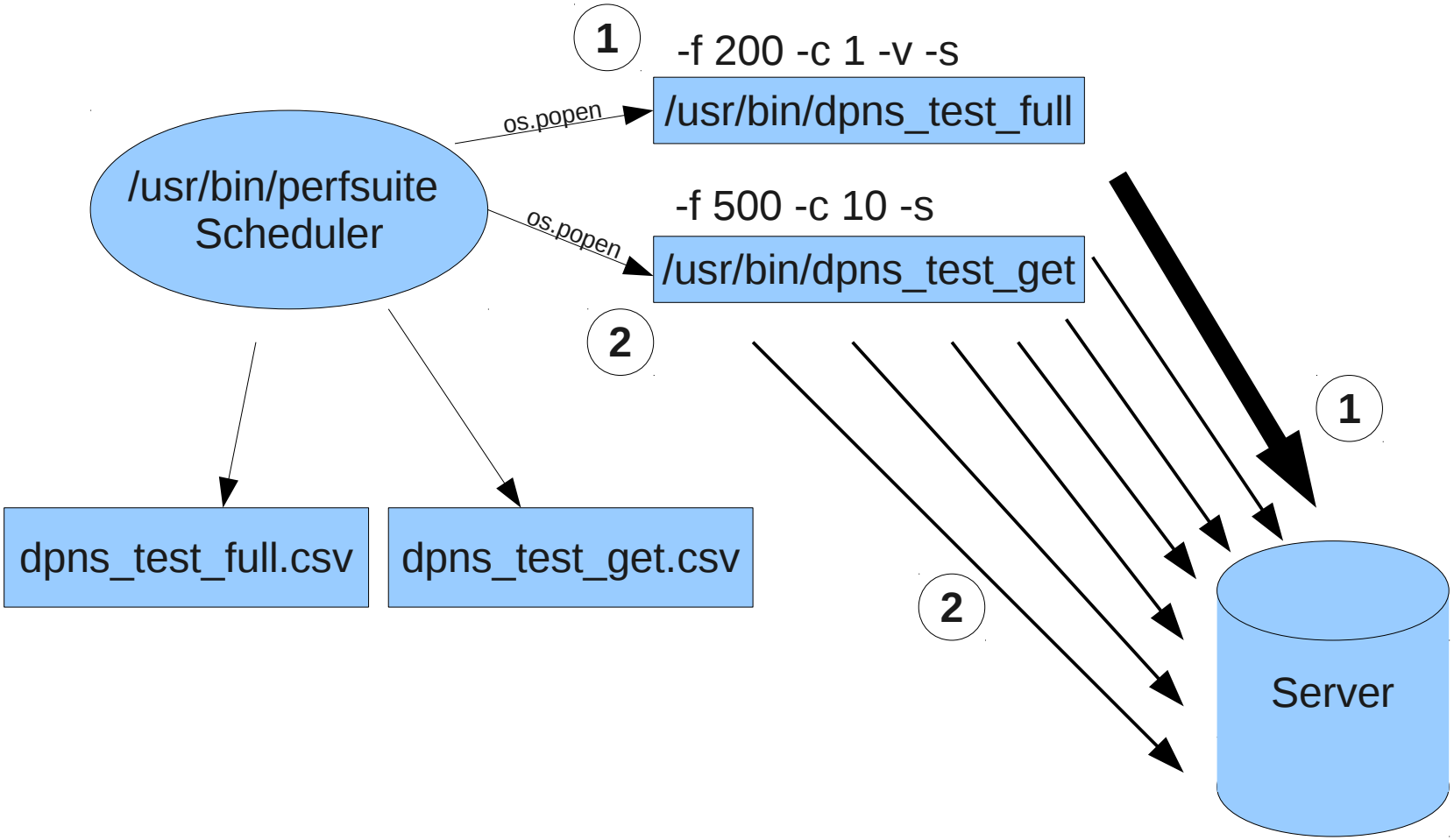
test_dpns_full(f:500,c:10,s:true)
test_dpns_get(f:500,c:10,s:true)

test_dpns_get(f:500,c:20,s:true)
test_dpns_get(f:500,c:20,s:true)

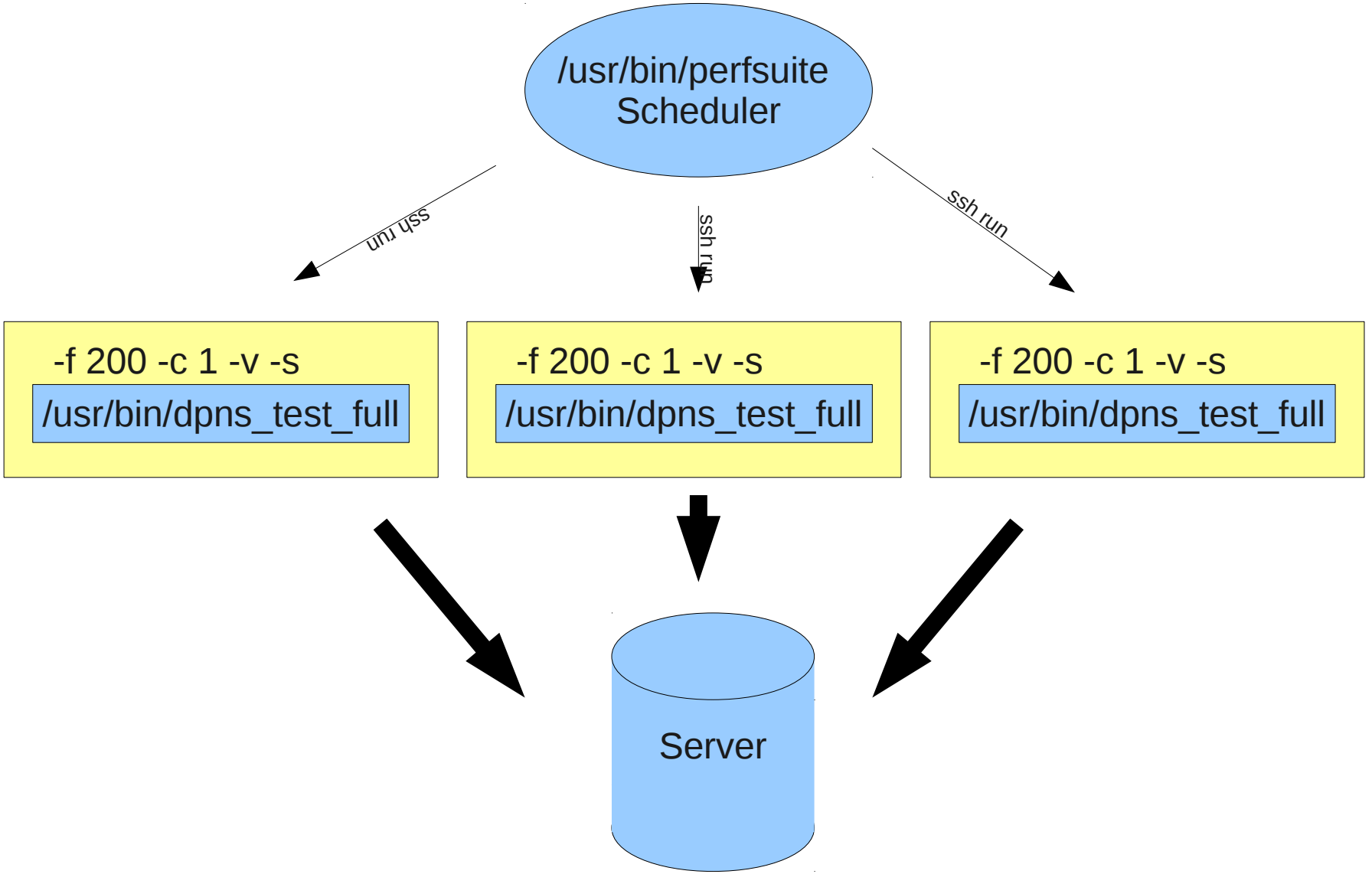
Perfsuite Structure



Perfsuite Local Execution



Perfsuite Distributed



How to get it

ls /etc/yum.repos.d/epel.repo

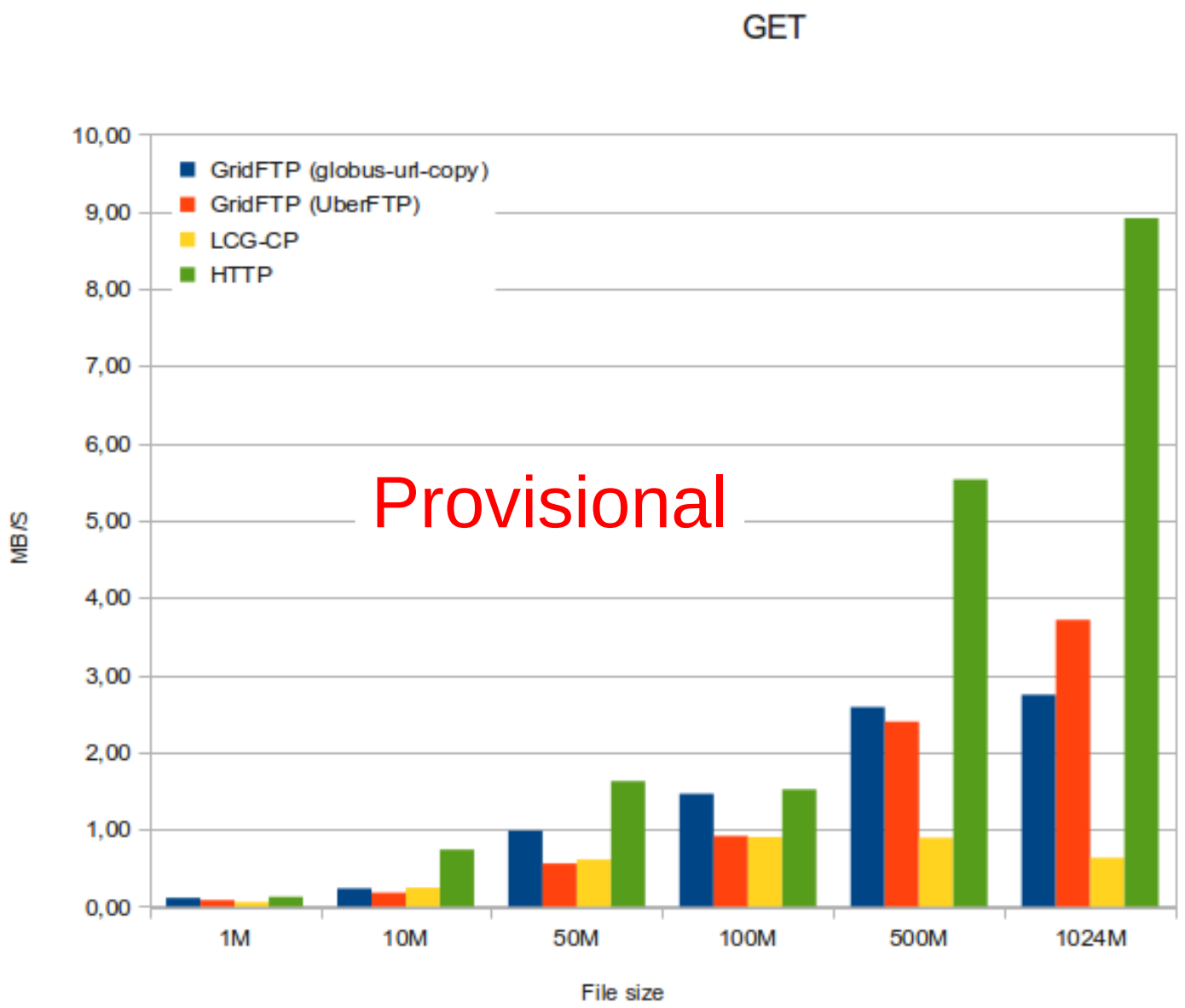
yum search perfsuite

perfsuite.x86_64
perfsuite-debuginfo.x86_64
perfsuite-test-metadata.x86_64
perfsuite-test-randomio.x86_64
perfsuite-test-transfer.x86_64

Infrastructure

- LAN
  - All machines are in the computer center
  - Gigabit Ethernet
- WAN
  - Client: CERN
  - Server: Taipei (ASGC)
  - CERN → ASGC goes through the OPN
    - 10Gbps
  - ASGC → CERN doesn't seem to, though

Results for full file downloading (WAN)



Results for pure random I/O (LAN)

Chunk size: 1024-2048; File size: 2208309933			
Protocol	N.Reads	Read size	Read time
HTTP	500	2,283,351	0.17
HTTP	1000	4,660,892	0.34
XROOT	500	2,283,351	0.11
XROOT	1000	4,660,892	0.22
RFIO	500	2,283,351	136.35
RFIO	1000	4,660,892	274.89

Provisional

Results for pure random I/O (LAN) II

Chunk size: 128-102400; File size: 2208309933; N.Read: 5000			
Protocol	Max.Vector	Read size	Read time
HTTP	8	1,166,615,844	12.55
HTTP	16	2,156,423,936	21.02
HTTP	24	3,211,861,800	29.97
HTTP	32	4,226,877,829	38.60
HTTP	64	8,535,839,293	75.20
XROOT	8	1,166,615,844	12.64
XROOT	16	2,156,423,936	22.11
XROOT	24	3,211,861,800	31.59
XROOT	32	4,226,877,829	41.14
XROOT	64	8,535,839,293	79.91

Provisional

Results for pure random I/O (WAN)

Chunk size: 10240-20480; File size: 2208309933			
Protocol	N.Reads	Read size	Read time
HTTP	500	22,773,112	193.88
HTTP	1000	46,027,143	383.32
XROOT	500	22,773,112	143.37
XROOT	1000	46,027,143	283.63

Provisional

Real use cases: CMSSW (WAN)

HTTP

```
real 15m4.518s
user 0m38.449s
sys 0m14.698s
```

XROOT

```
real    11m30.439s
user    0m37.074s
sys     0m1.493s
```

In fact, more data is requested when HTTP is used. The reasons are not clear yet (caching?).

Provisional

Real use cases: HammerCloud

	Remote-HTTP	Remote-HTTP (TTreeCache)	Staging - HTTP	Remote-XROOTD	Remote-XROOTD (TTreeCache)	Staging - XROOTD
		Provisional				
Number of files	40	40	13.3	40	40	13.2
Number of events	199973	199973	66383.3	199973	199973	66173.4
Fetch Panda job	3.8	3.5	4	5.6	3.1	5.1
Set up Software Time	10.8	14.1	16.1	12.5	13.3	17.3
Download input files	127.8	130.2	813.5	128.2	129.2	1620
Athena Running Time	17151	7317.1	1858.5	19931.7	5824.9	1967.1
Output Storage Time	31.3	30.9	33.6	30.9	32.6	34.2
Wallclock	17474.5	7625.5	2874	20256.3	6140.4	3796.1
Completed jobs	597	1376	1537	507	1683	1552
***Events/Athena(s)**	11.8	27.9	37.9	10.1	34.8	35.4
***Eventrate**	11.5	26.8	24.2	9.9	33.1	17.7
***CPU percentage**	28.2	68.2	46	31.8	77	34.5
Job Efficiency	1	0.994	1	1	0.99	0.999