PhEDEx – reliable and scalable data distribution on the Grid

Tim Barrass University of Bristol
Jens Rehn CERN
Lassi A. Tuura Northeastern University

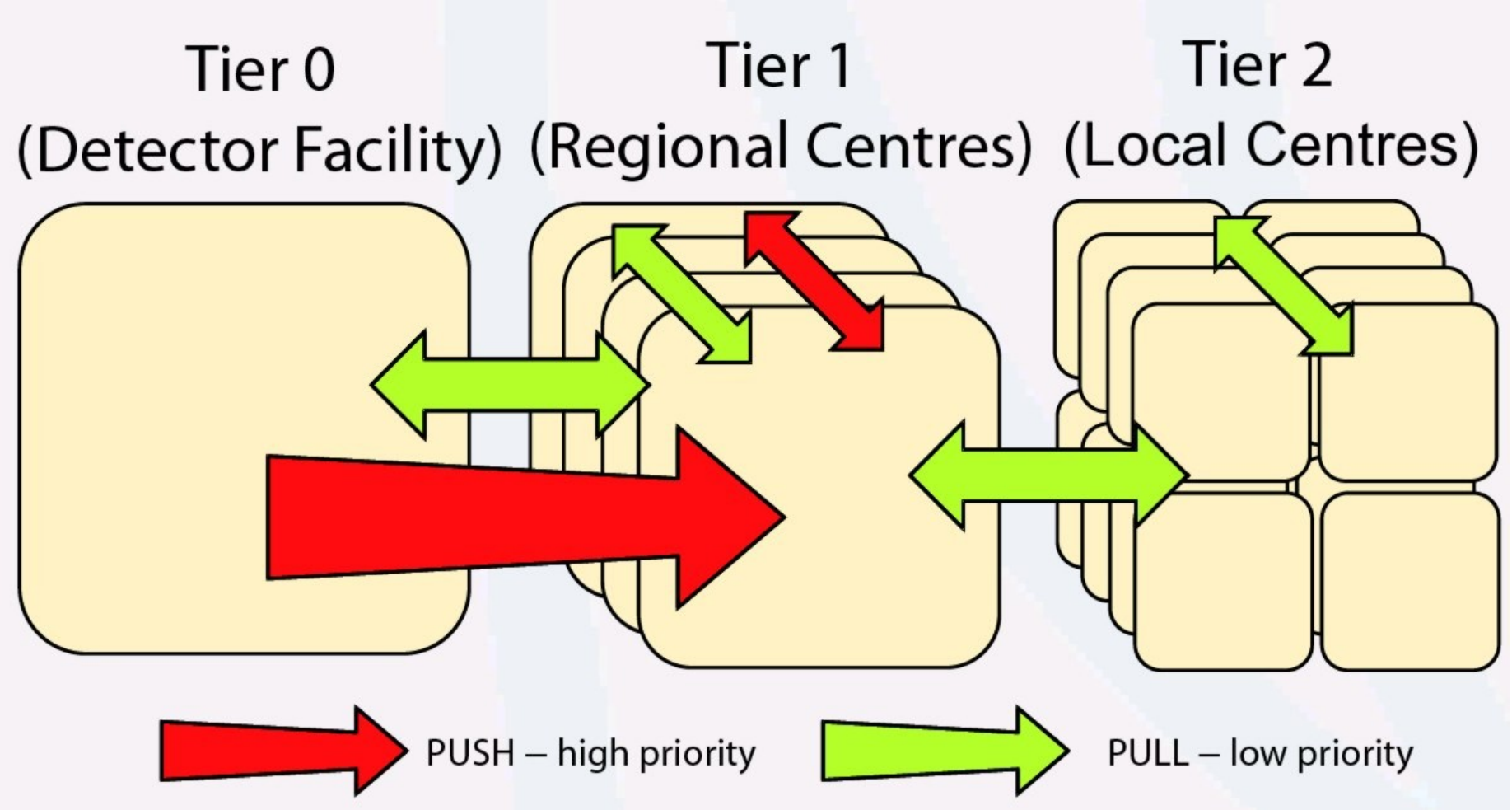
Outline

- Introduction to PhEDEx
  - HEP data transfers
  - Features and functionality
- Operating and monitoring a live PhEDEx system
- Practical examples from the last service challenge
- How to set-up and run PhEDEx

CMS data flow

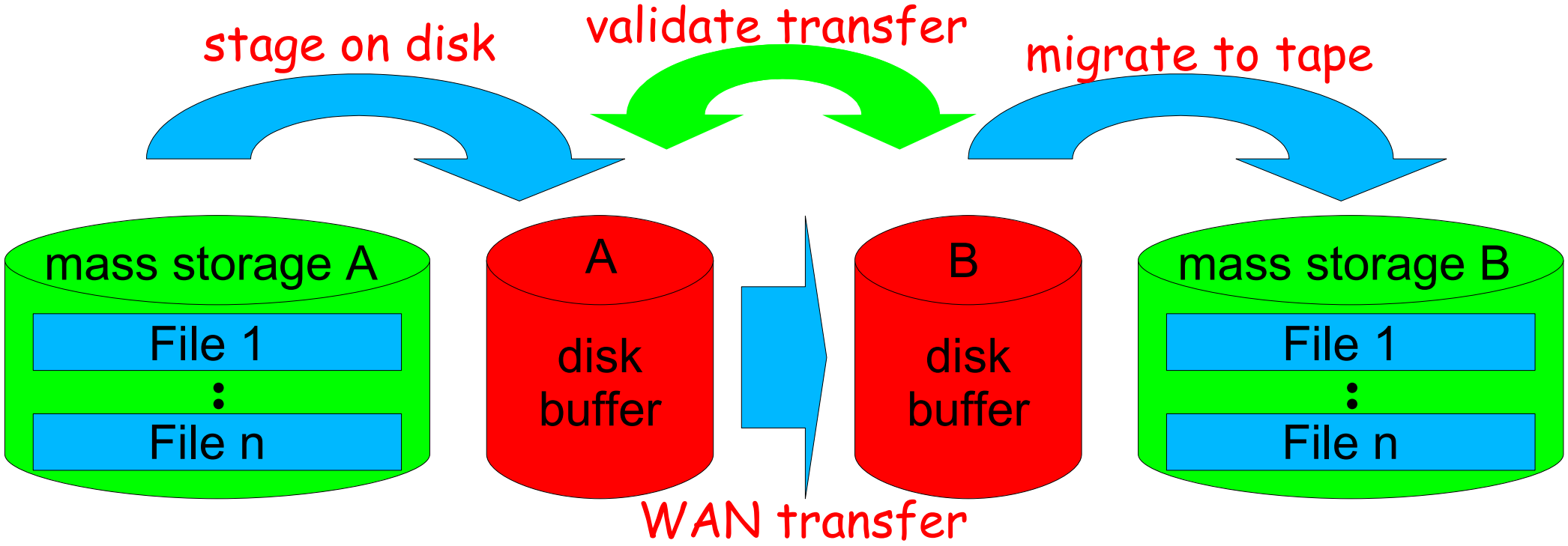
- Detector data distribution @ high priority
  - One copy at Cern; one distributed copy at regional centers
  - Expected transfer volume for 2008: ~ 7 PB ≈ O(10M) files
  - Required transfer speed for 2008: ~ 5 Gb/s
- Simulated data distribution @ low priority
  - Among and between regional and local centers
  - Expected bandwidth utilisation: few Gb/s per regional center
- Data structured in blocks of files
  - dataset, datatiers

Tiered data flow



- Push and pull are logical operations – not tech. implementations

HEP data replication
Traditional workflow



- Each step done by hand
- Manpower intensive
- Feasible only for small amount of files

HEP data replication
PhEDEx workflow

monitoring

management at block level

file level operations

find closest replica

stage on disk

validate transfer

migrate to tape

mass storage A

File 1

File n

A

disk buffer

B

disk buffer

mass storage B

File 1

File n

WAN transfer

HEP requirements for a data distribution system

- Managed & structured data flow
- Reliability
  - Robustness & self-healing
    - Error recovery, automatic back-off, etc
  - Integrity of replicated data
- Flexibility
  - Different transfer models: push and pull
  - Support of common transfer protocols & storage systems
- Monitoring

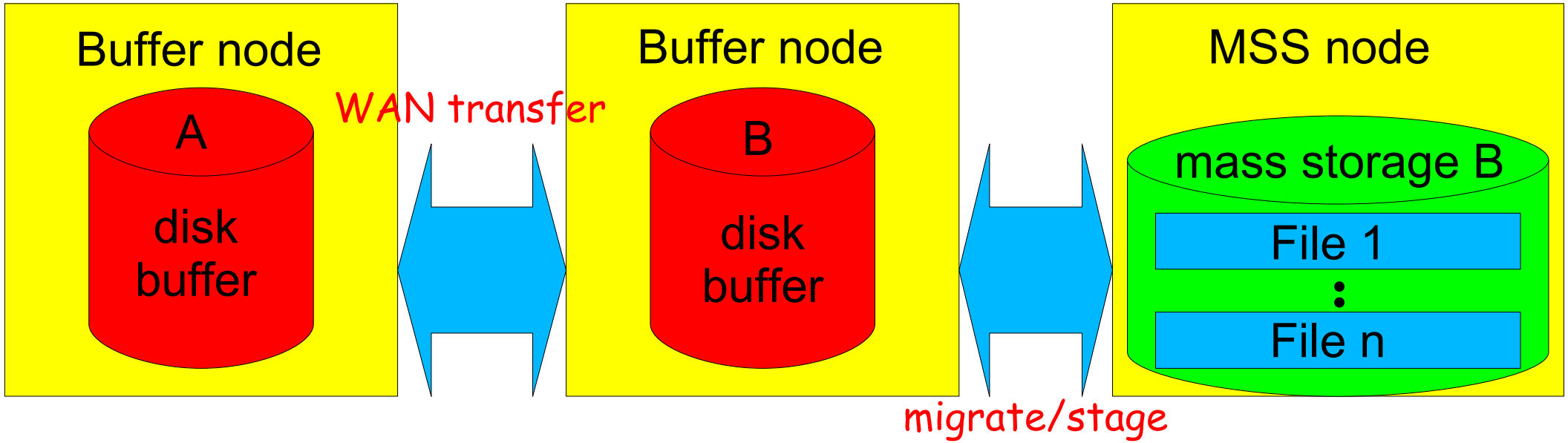
PhEDEx – design
Key features (1)

- Reliability
  - Transfer status monitored
  - Filesize check after each replication
  - Cksum for every file in TMDB available for further checks
  - Automatic cool off for failed transfers
  - Self-throtteling: limits amount of parallel transfers
  - Designed under assumption: any operation might fail
- Monitoring
  - Status web page: http://cern.ch/cms-project-phedex

PhEDEx – design
Key features (2)

- Flexibility
  - Push and pull models supported: logical implementation
    - Push: data subscribed to destination by site hosting replica
    - Pull: destination site subscribes data to itself
  - Automatic protocoll matching: G-U-C, srmcp, dccp, rfcp, lcg-cp
  - Intelligent routing with fall-back mechanism
- Operability
  - Easy to handle deployment
  - Linux *inetd* like start/stop mechanism for agents
  - Try to provide easy to understand log messages

PhEDEx – design
Transfer nodes



- Logical storage units called nodes
  - Buffer node: disk based storage
  - MSS node: tape based mass storage
  - Gives flexibility to sites to organise data storage

PhEDEx – design
Agents & blackboard

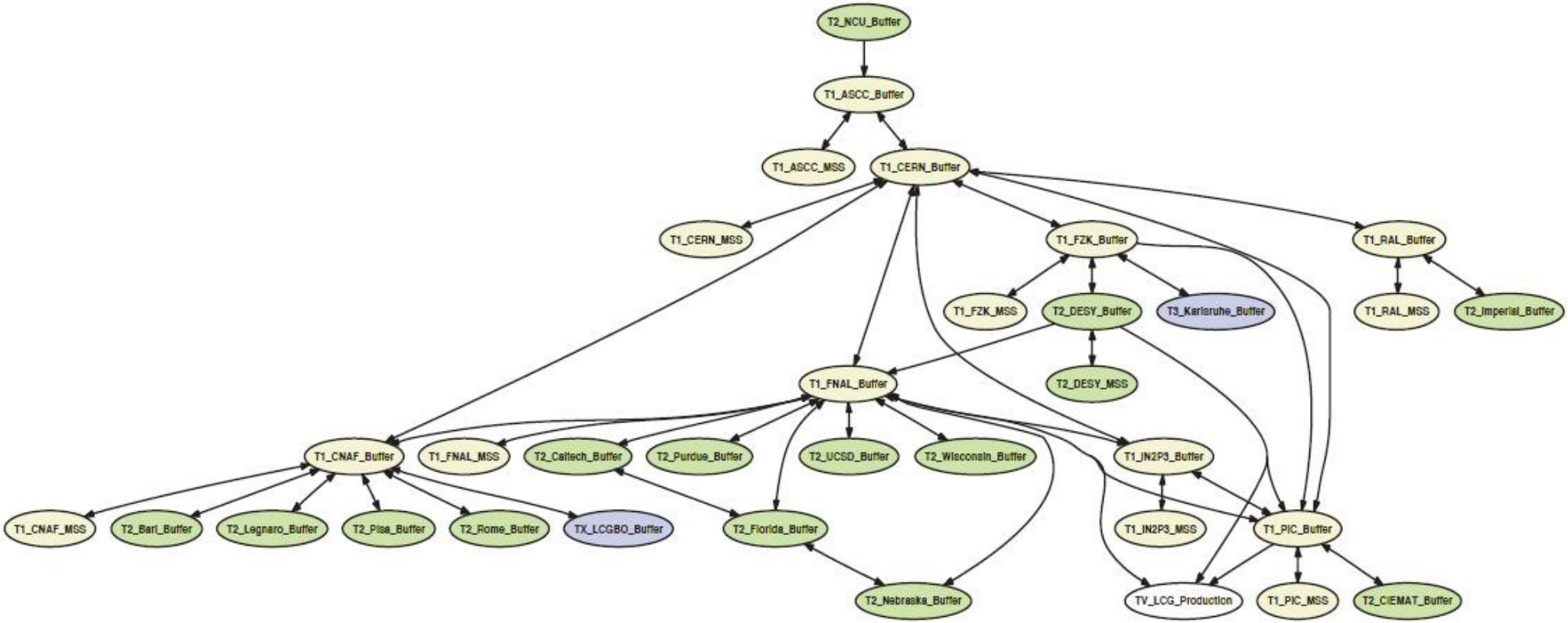
Each site runs a set of specialised agents:

- Agents designed to
  - fullfill a specific „simple" task in a reliable way
- Site specific agents: routing, replication & mass storage

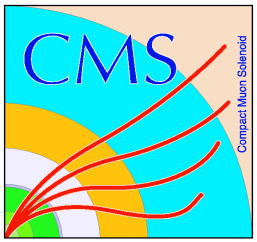
Agents communicate with central blackboard:

- Block replica location & file mapping
- Block subscription and allocation
- File metadata information (filesize, cksum, etc)
- Transfer state (at node; in transfer; wanted; available)

PhEDEx – design
Distribution network



Currently we have one T0, 7 T1s, 16 T2s or smaller sites



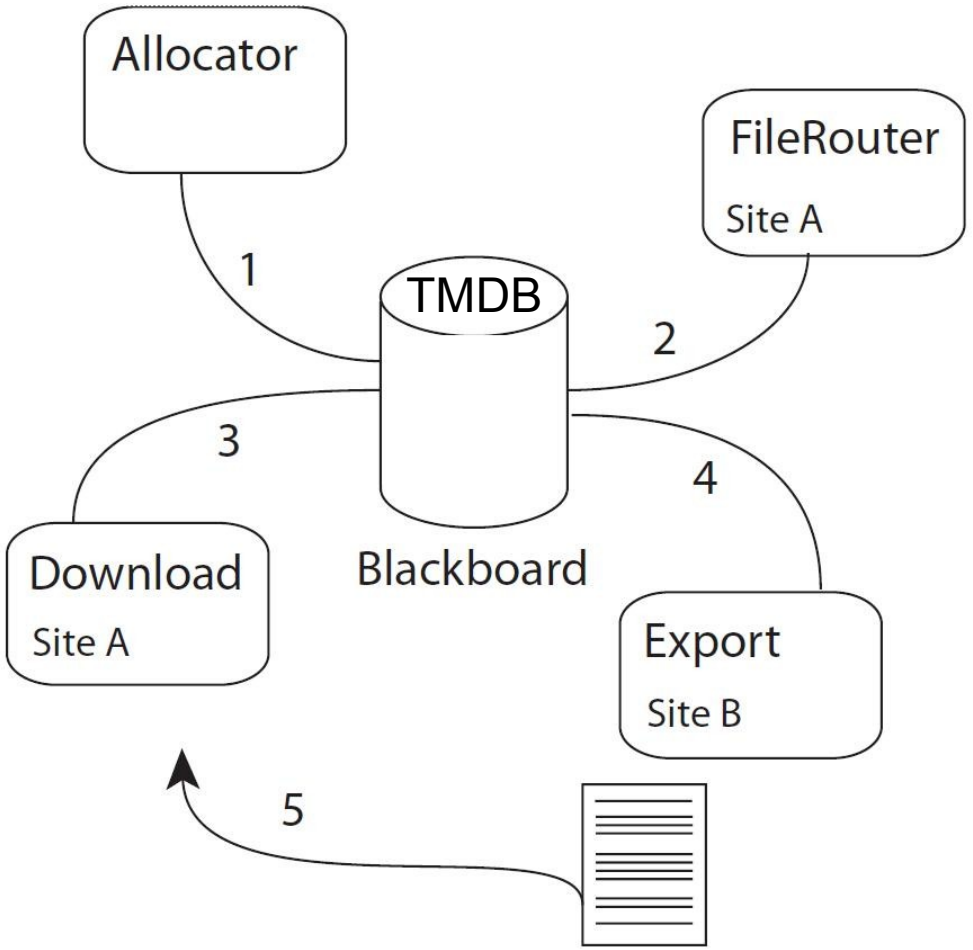
PhEDEx – in practice
Monitoring & subscription

How to monitor transfers

How to subscribe to CMS datasets

http://cern.ch/cms-project-phedex

PhEDEx – design
Data replication

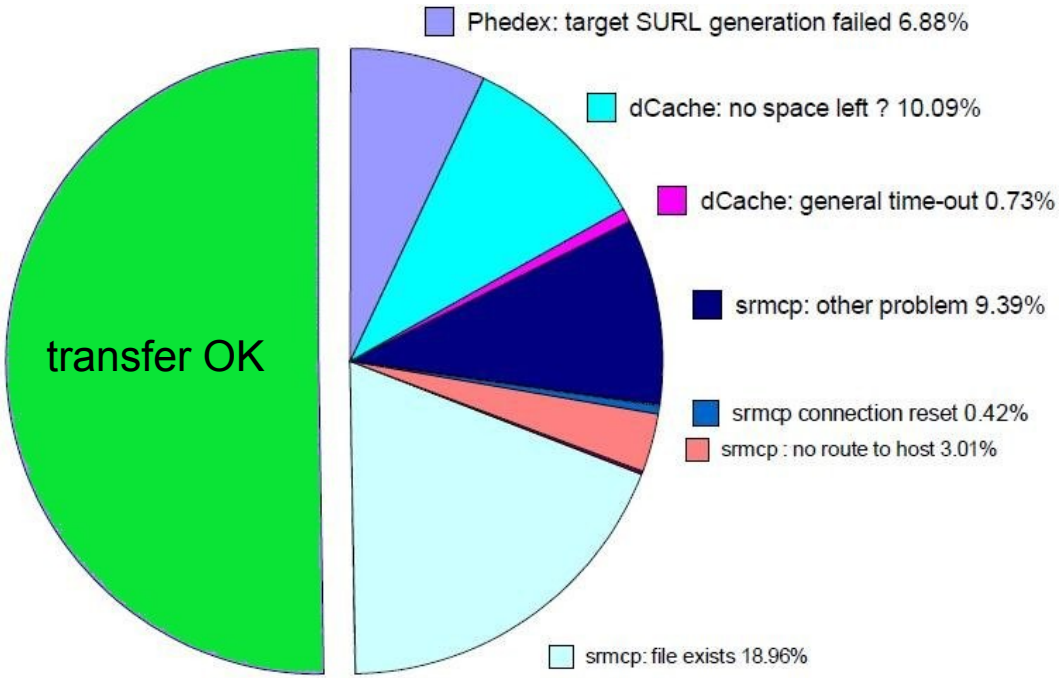


1. Allocator: allocate files to destinations
2. FileRouter: maintains & determines best routes
3. Download: marks files „wanted" from site B
4. Export: initiate staging & provide contact information
5. Download: transfer file

PhEDEx – design
Intelligent routing

- Routing agent determines best route: source → destination
- Routes are ranked automatically
  - According to amount of intermediate nodes: hops
  - Hops can be weighted
- IP-like routing to route files to destination
  - In case of outage, fallback routes chosen via other nodes
  - Unavail. or dead nodes noticed by neighbours; no heartbeat

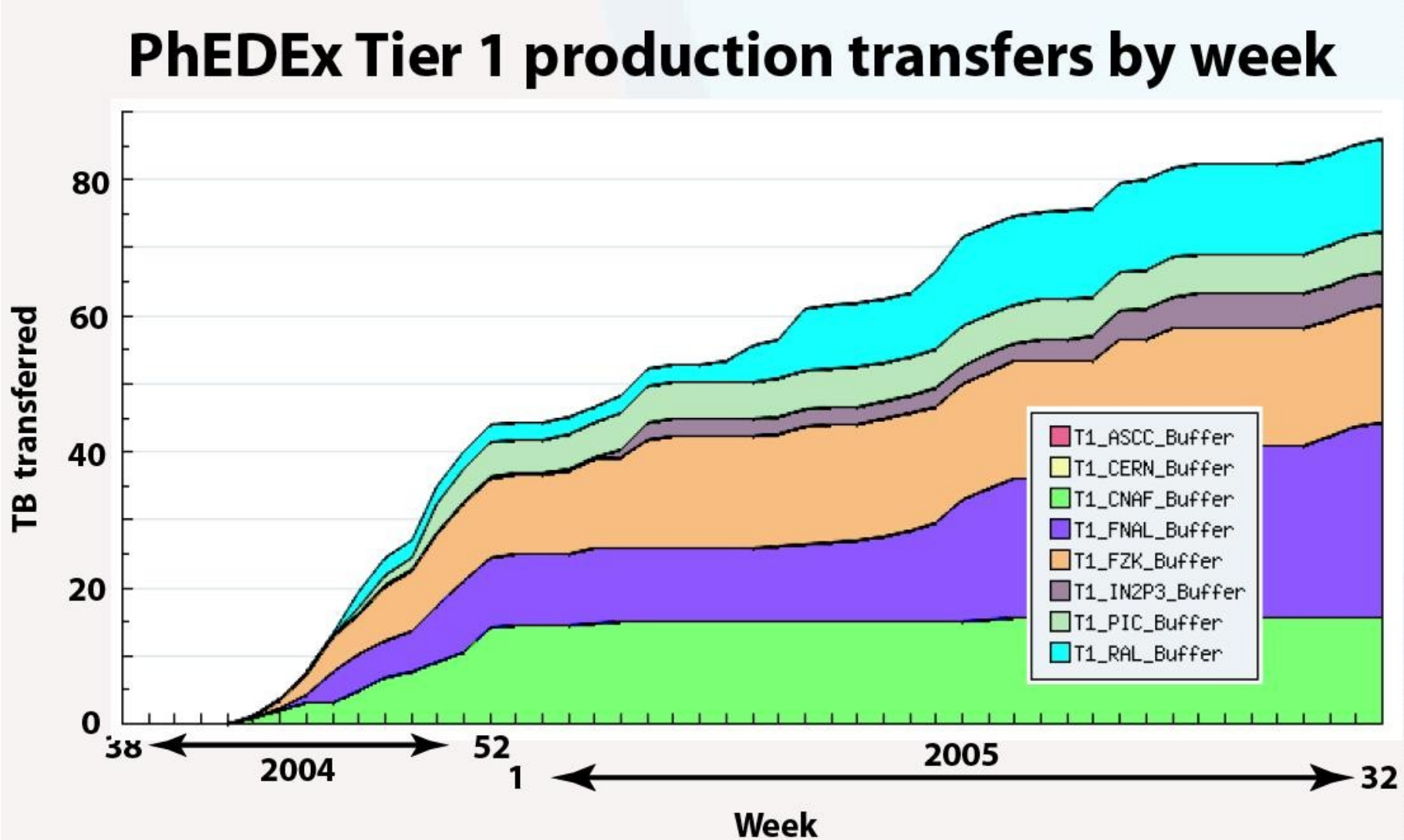
PhEDEx – design
Reliable file replication



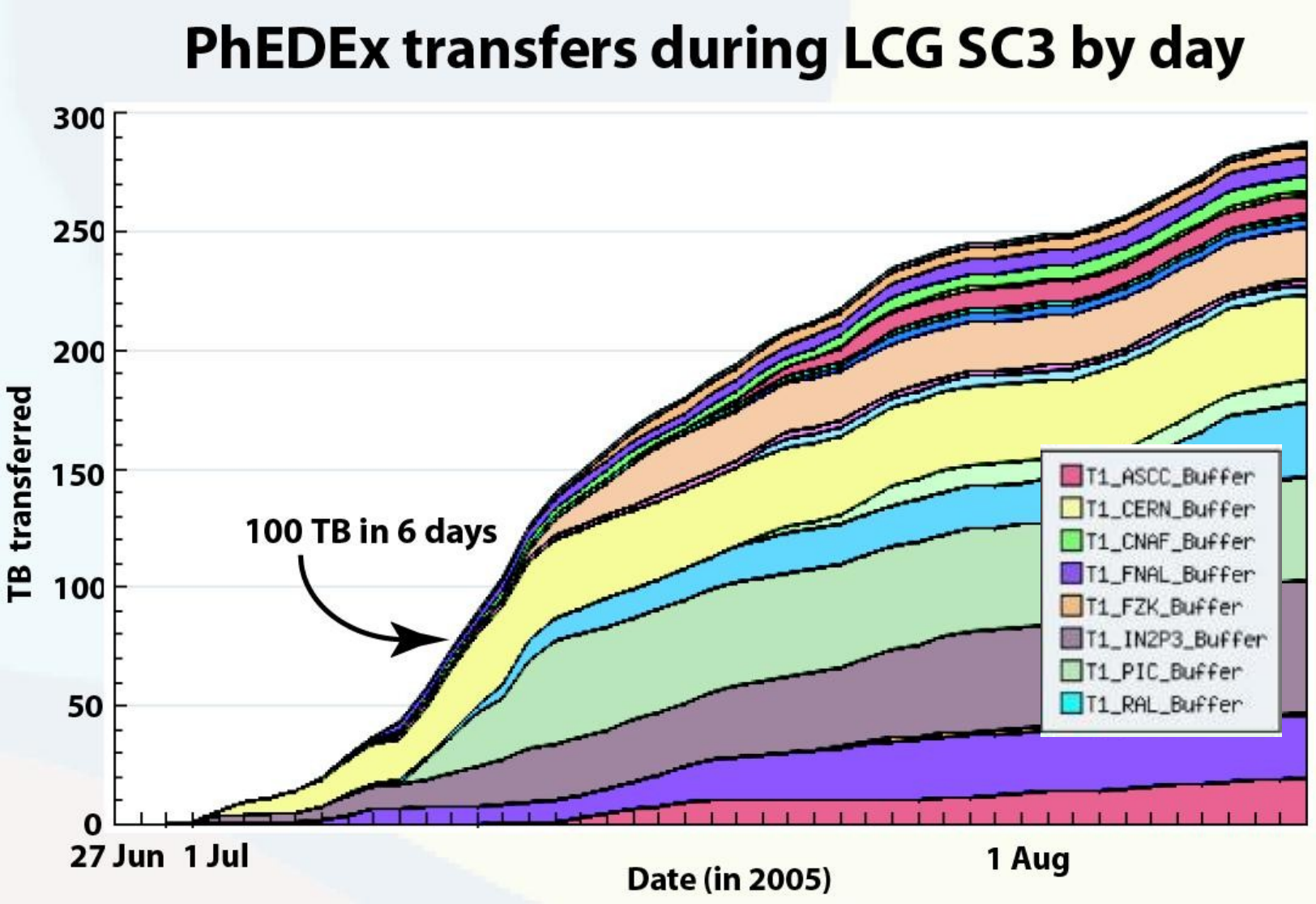
Example of failures experienced during SC3 throughput phase

- Extreme failure rate on new infrastructure
- Only 50% success rate !
  - Failures recovered
  - Files retransfered
  - no data lost :-)
- Recovery by hand not possible for millions of files

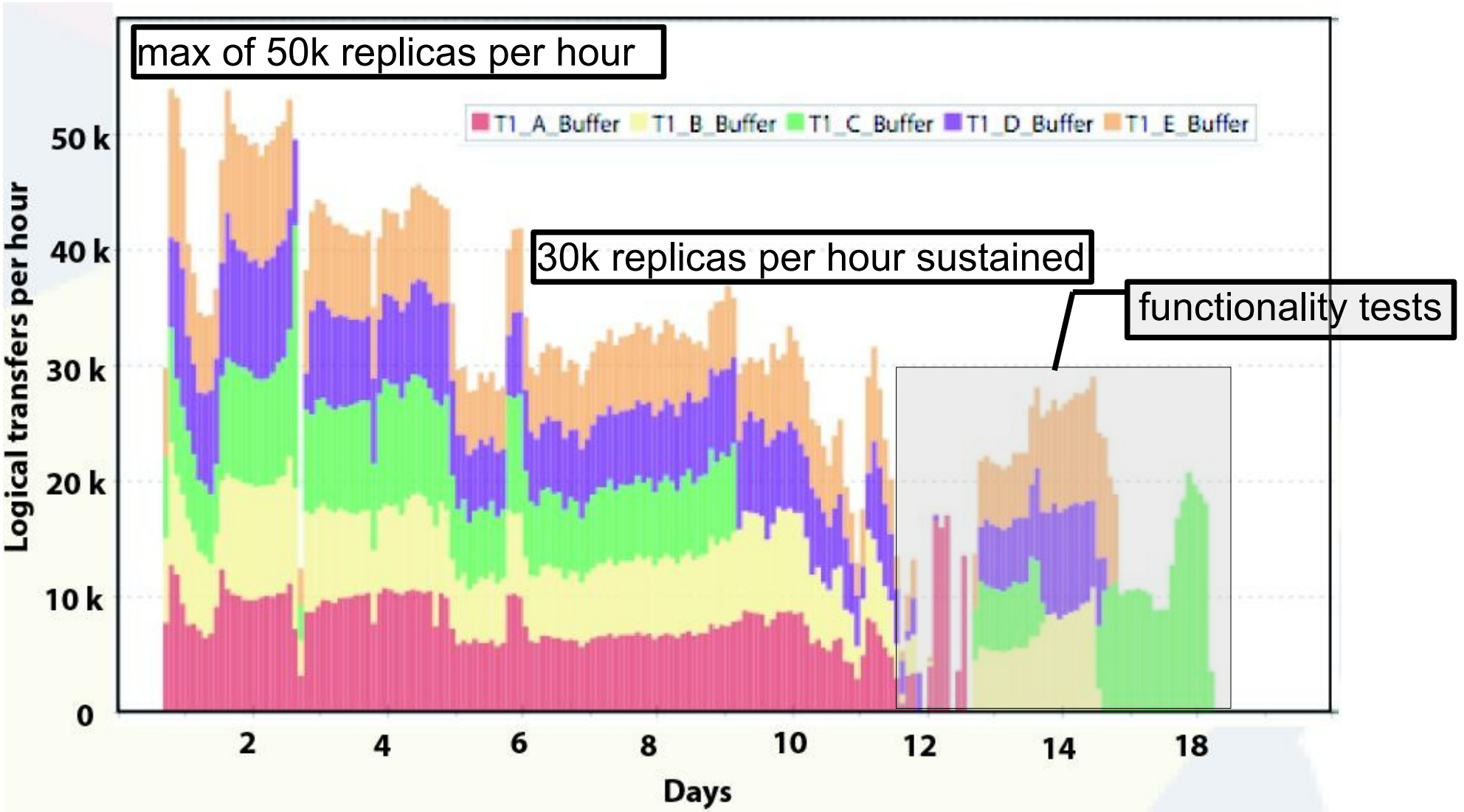
PhEDEx – in practice
Replication performance (1)



PhEDEx – in practice
Replication performance (2)



PhEDEx – in practice
Scalability



PhEDEx – deployment
Overview (1)

- Hardware
  - Machine running PhEDEx agents
  - Disk buffer + tape system (optional)
  - Machine providing catalogue service (**MySQL**, Oracle, LFC)
- Software
  - PhEDEx itself
  - POOL file catalogue tools
  - Oracle client libraries & Perl DBI modules
  - Transfer utilities (**srmcp**, g-u-c, lcgcp, etc)

} managable by XCMSi

PhEDEx – deployment
Overview (2)

- Grid services
  - Site local file catalogue
  - Certificate management (e.g. myproxy)
- Configuration
  - Registration of site nodes in central DB
  - Site local glue scripts, templates provided

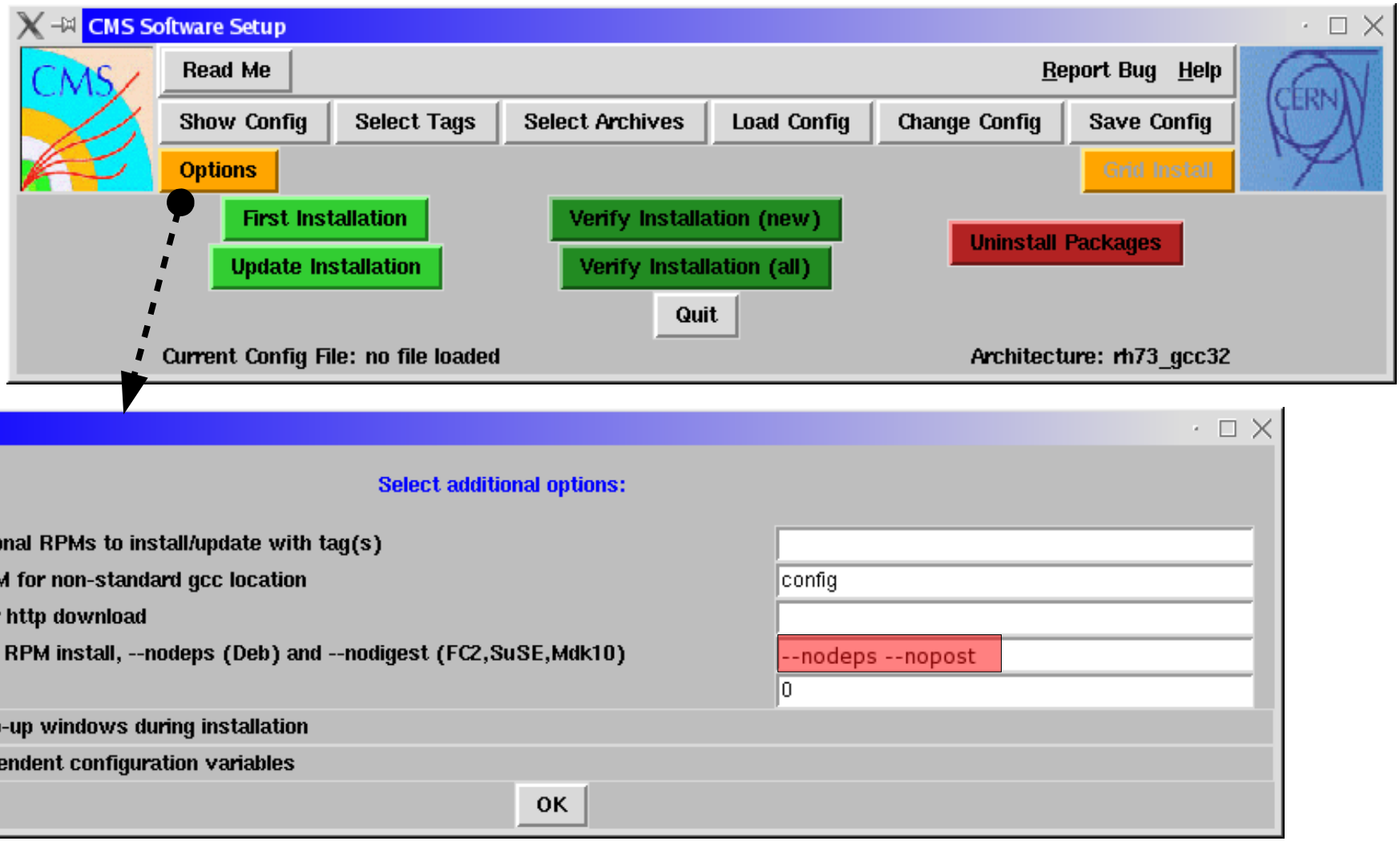
PhEDEx – deployment
Software

- Option 1: checkout PhEDEx from CVS repository
  - CVSROOT=:pserver:anonymous@cmscvs.cern.ch:/cvs_server/repositories/PHEDEX
  - Password: passwd98
  - Execute a series of scripts found in PHEDEX/Deployment
    - Follow PHEDEX/Documentation/README/README-Deployment
- Option 2: use XCMSi
  - User-friendly installation wizard with GUI
  - Most installation steps covered

PhEDEx – deployment
Software via XCMSi (1)

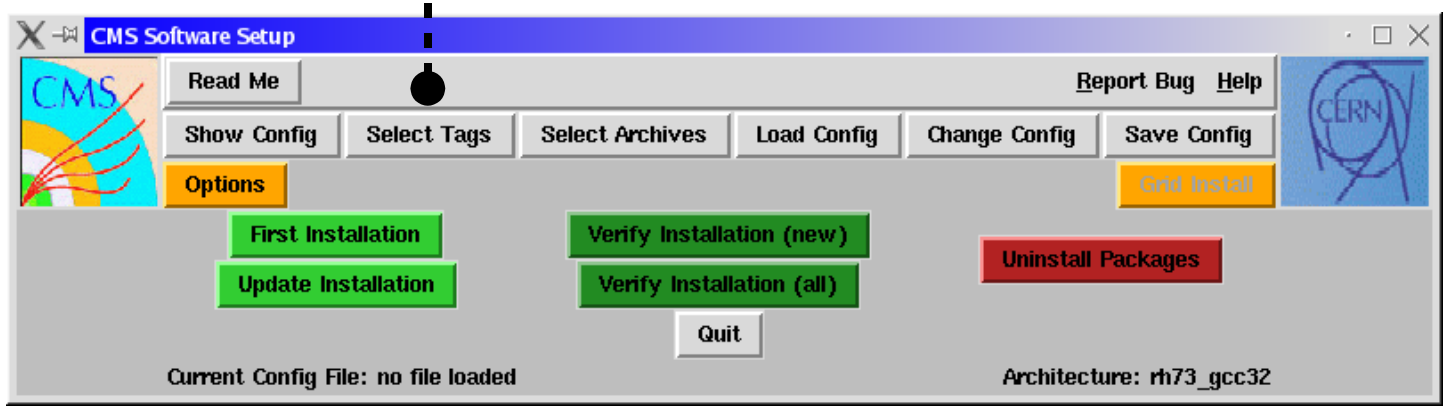
- Decide where to install Phedex
  - Create install dir for XCMSi ($xcmsi-base)
  - Create basedir for PhEDEx ($phedex-base)
- Download packages
  - XCMSi from http://cern.ch/cms-xcmsi
  - Untar XCMSi to $xcmsi-base
  - Get Oracle client libraries (zip): http://www.oracle.com
    - put them in sub-dir $xcmsi-base/ZIPS
- Start the installation GUI
  - cd $xcmsi-base; ./xcmsi.pl

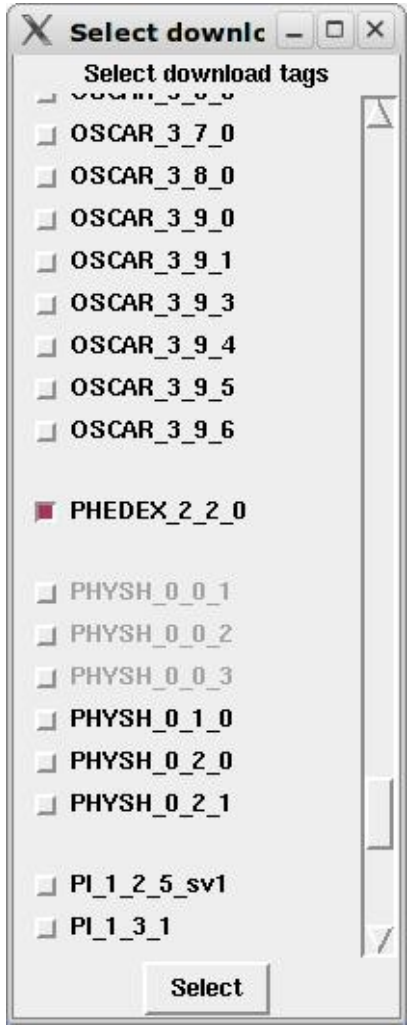
PhEDEx – deployment
Software via XCMSi (2)



No post installation scripts and no dependency checking

PhEDEx – deployment
Software via XCMSi (3)

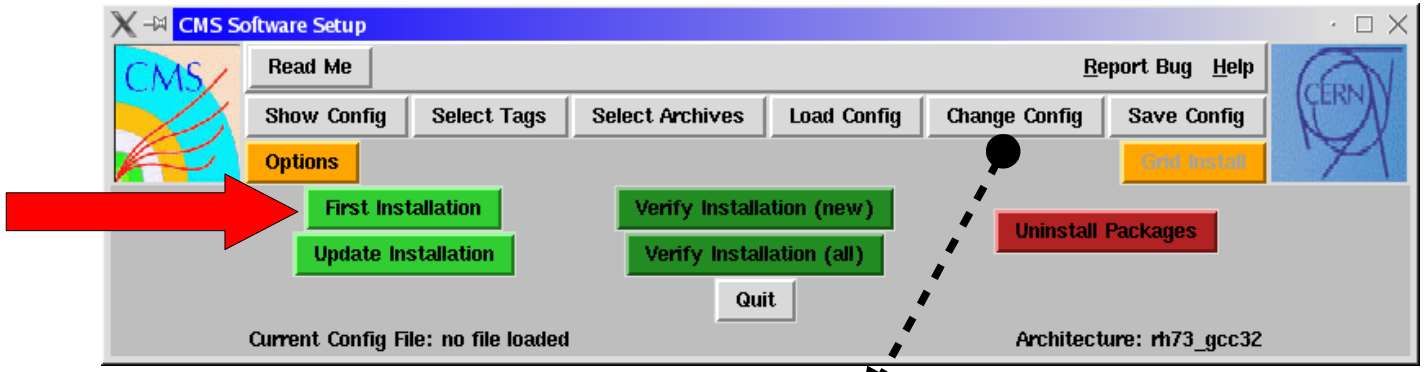


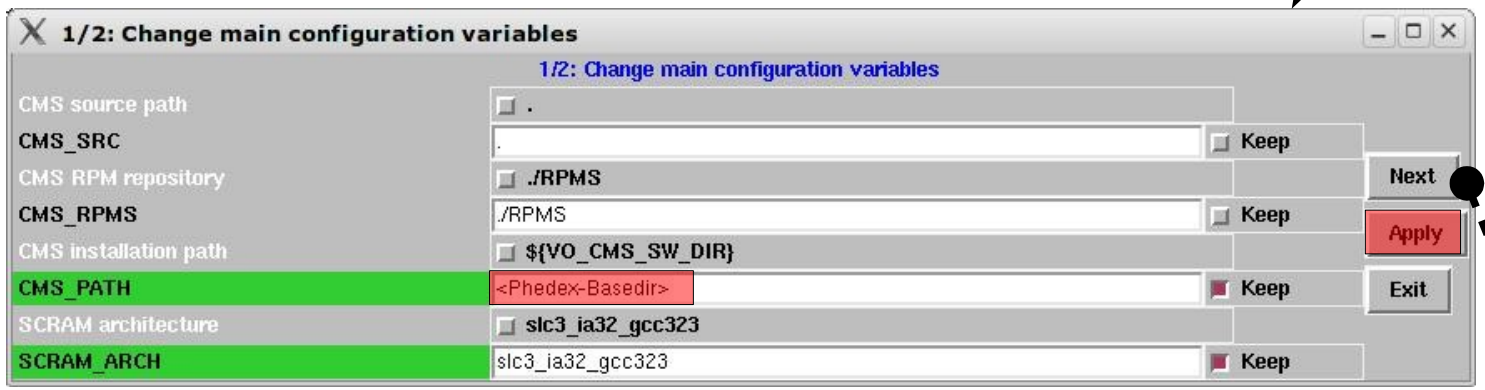


Select PhEDEx version

PhEDEx – deployment
Software via XCMSi (4)

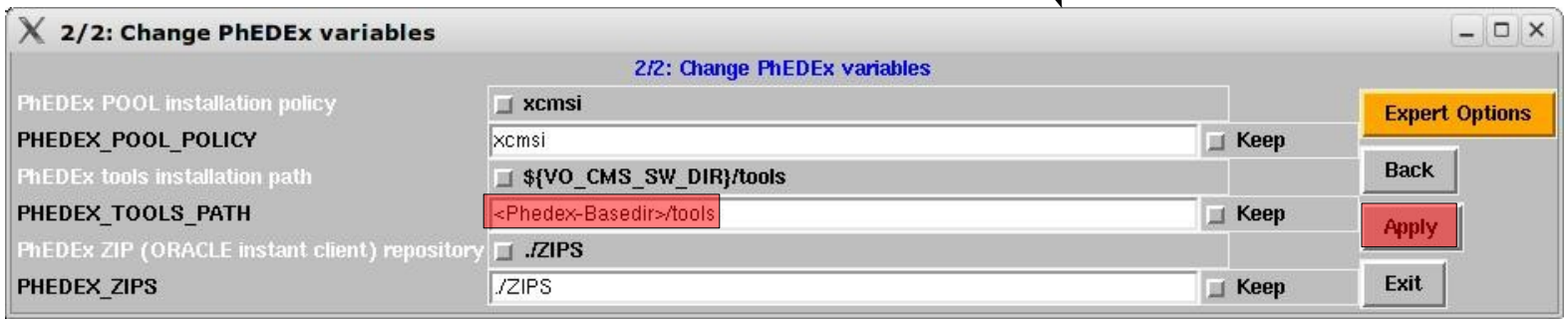
3. Finally start installation





1. Select PhEDEx installation dir

2. Select PhEDEx tools dir



PhEDEx – deloyment
Grid services

- Site local file catalogue – if you don't have one already
  - Any catalogue type is fine
  - MySQL based POOL file catalogue:
    - Helper script PHEDEX/Deployment/SetupPOOLFileCatalogue
- Certificate management
  - Valid Grid certificate proxy: grid-proxy-init
  - Recommended auto-renewal via myproxy

PhEDEx – deployment
Configuration

- Site registration in PhEDEx in central DB
  - Obligatory: Documents/README/README-Deployment
  - Currently: send mail to phedex-developer@cern.ch (CMS only)
- Site local glue scripts
  - Get a copy of templates provided in
    - Custom/CERN
  - Adjust them to meet your site's requirements
    - Remove all not needed agents ! Typically only ~ 5 are needed
- Testing your installation
  - Run Deployment/TestInstallation

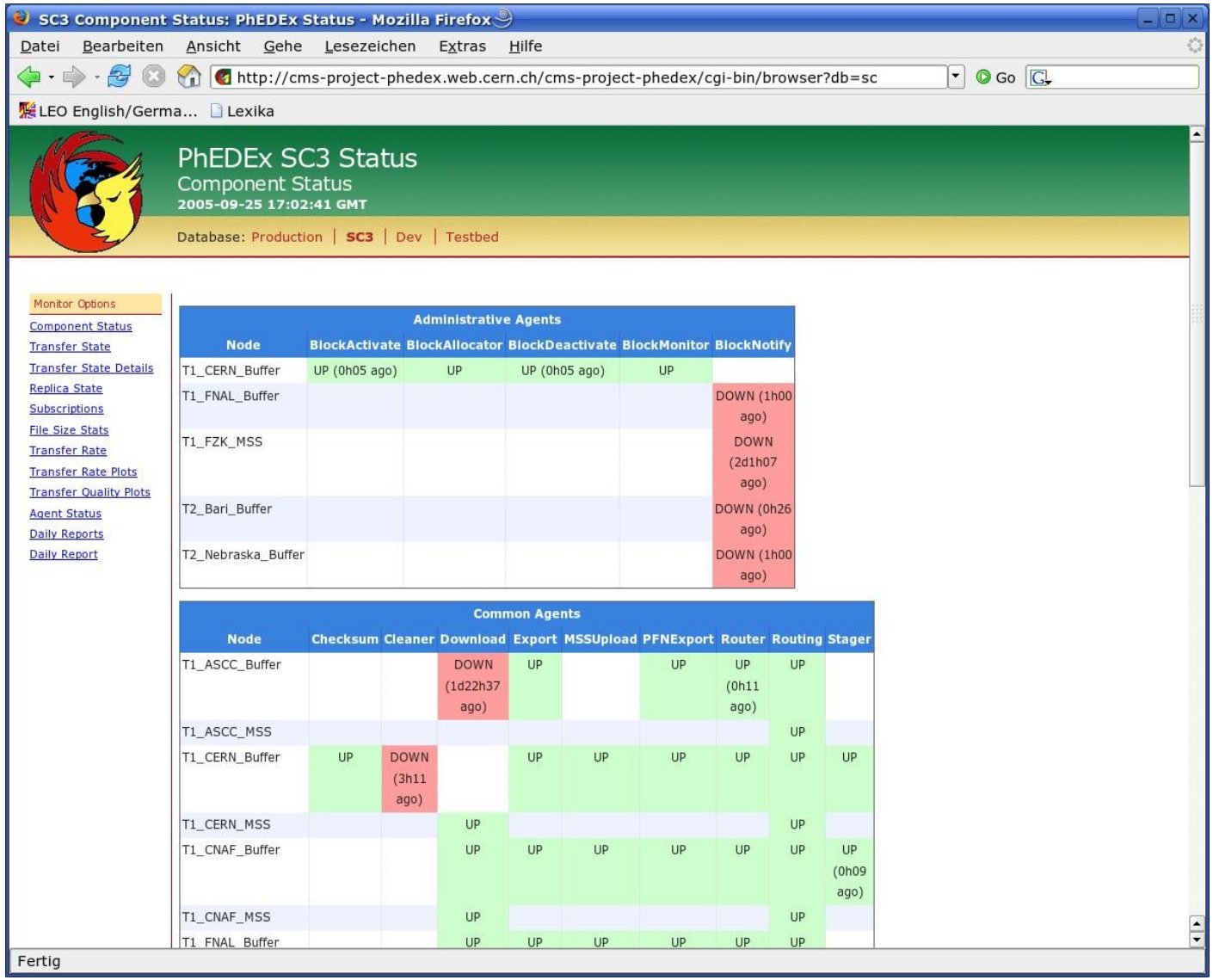
Summary

- PhEDEx provides:
  - Reliable and scalable data distribution on the Grid
  - Flexibility to use any Grid-based replication tool
  - Monitoring through a web server
- Phedex plans:
  - Improve web interface for operations
    - data subscriptions, transfer requests, agent management, deployment
  - Decentralisation of central DB
- Hope to welcome you aboard soon ;-)

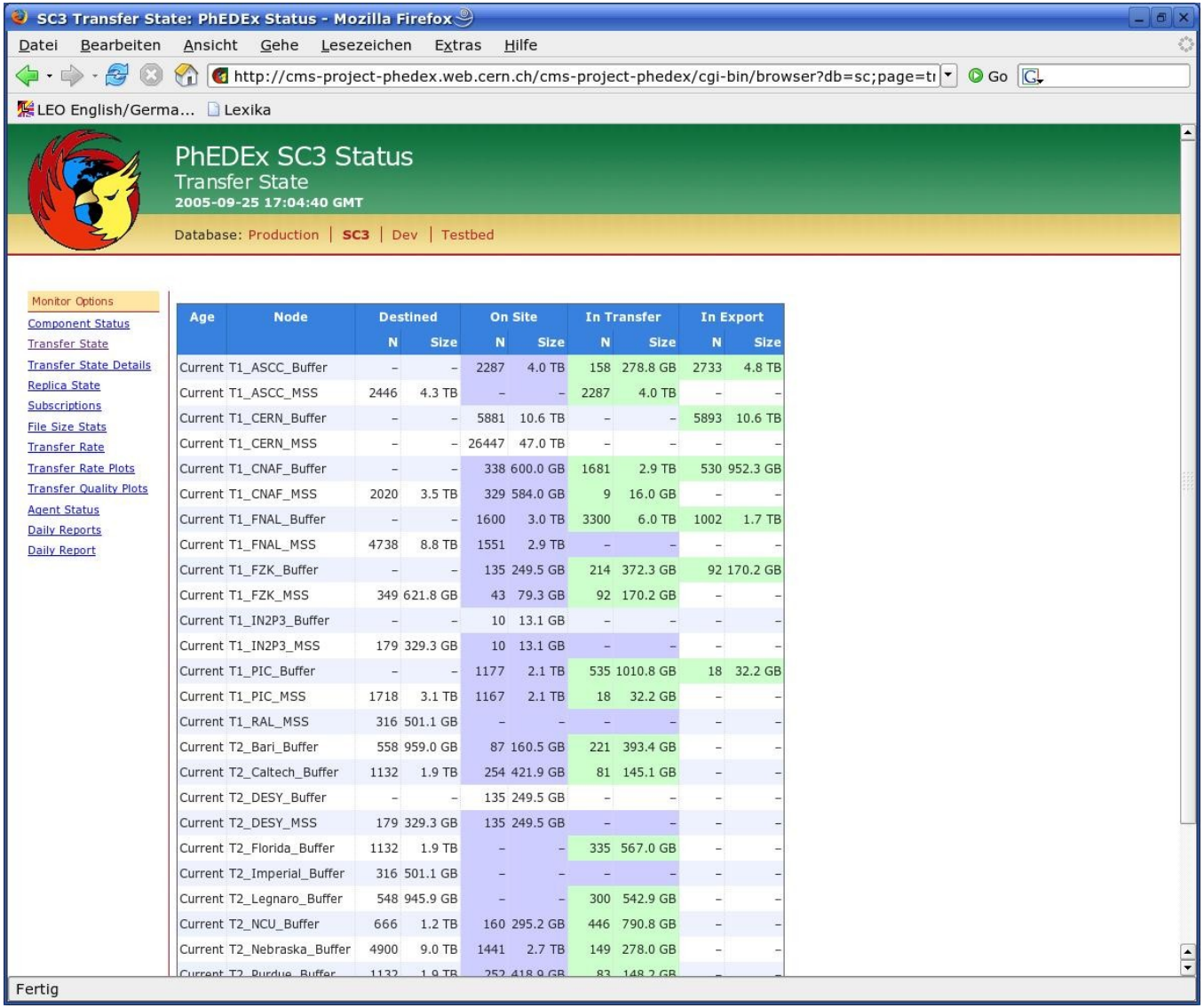
Useful links & contacts

- PhEDEx project web page:
  - http://cern.ch/cms-project-phedex
  - links to documentation, monitoring & CVS repository
- PhEDEx mailing list:
  - cms-phedex-developers@cern.ch

PhEDEx – monitoring
Component status

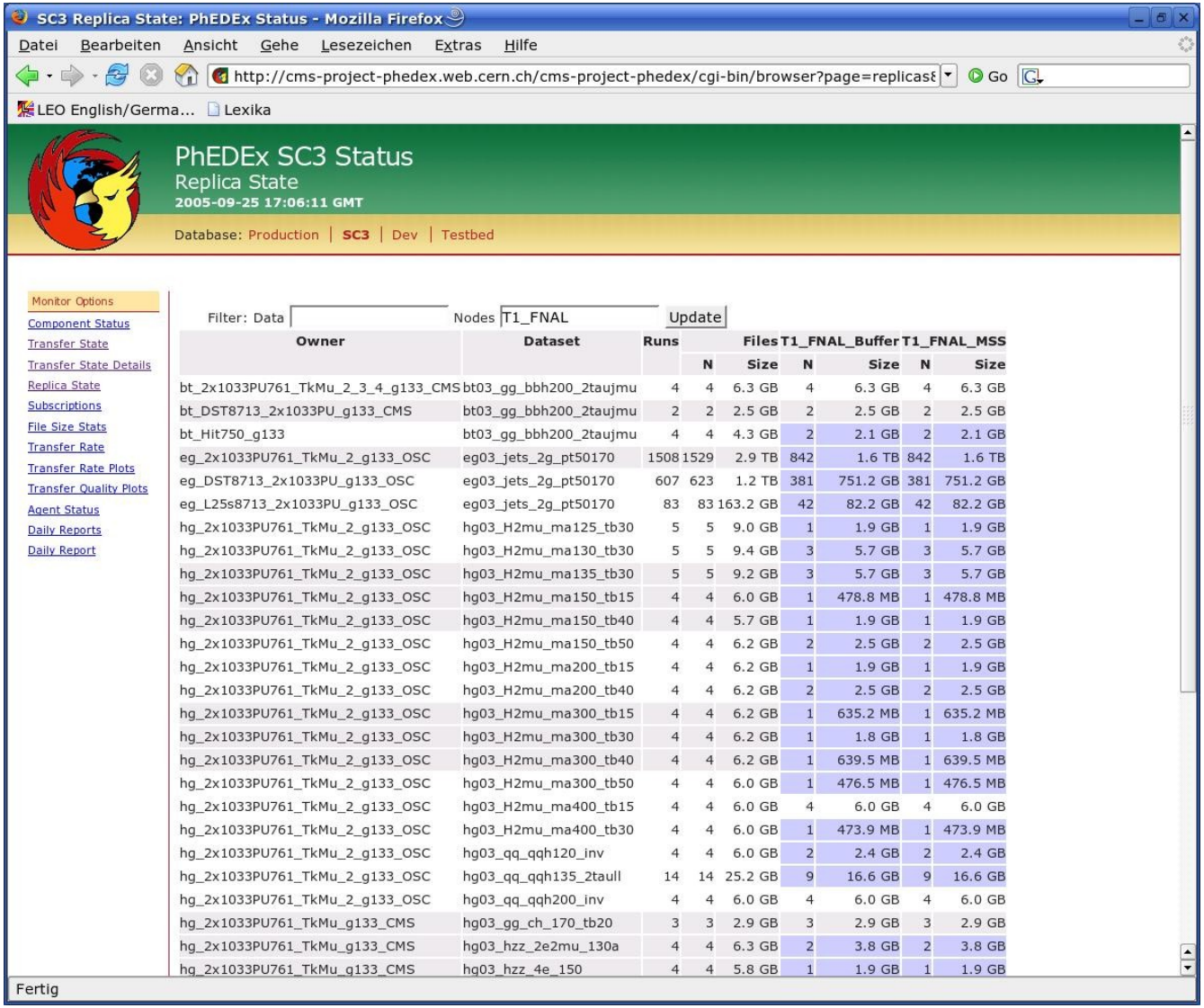


PhEDEx – monitoring
Transfer state



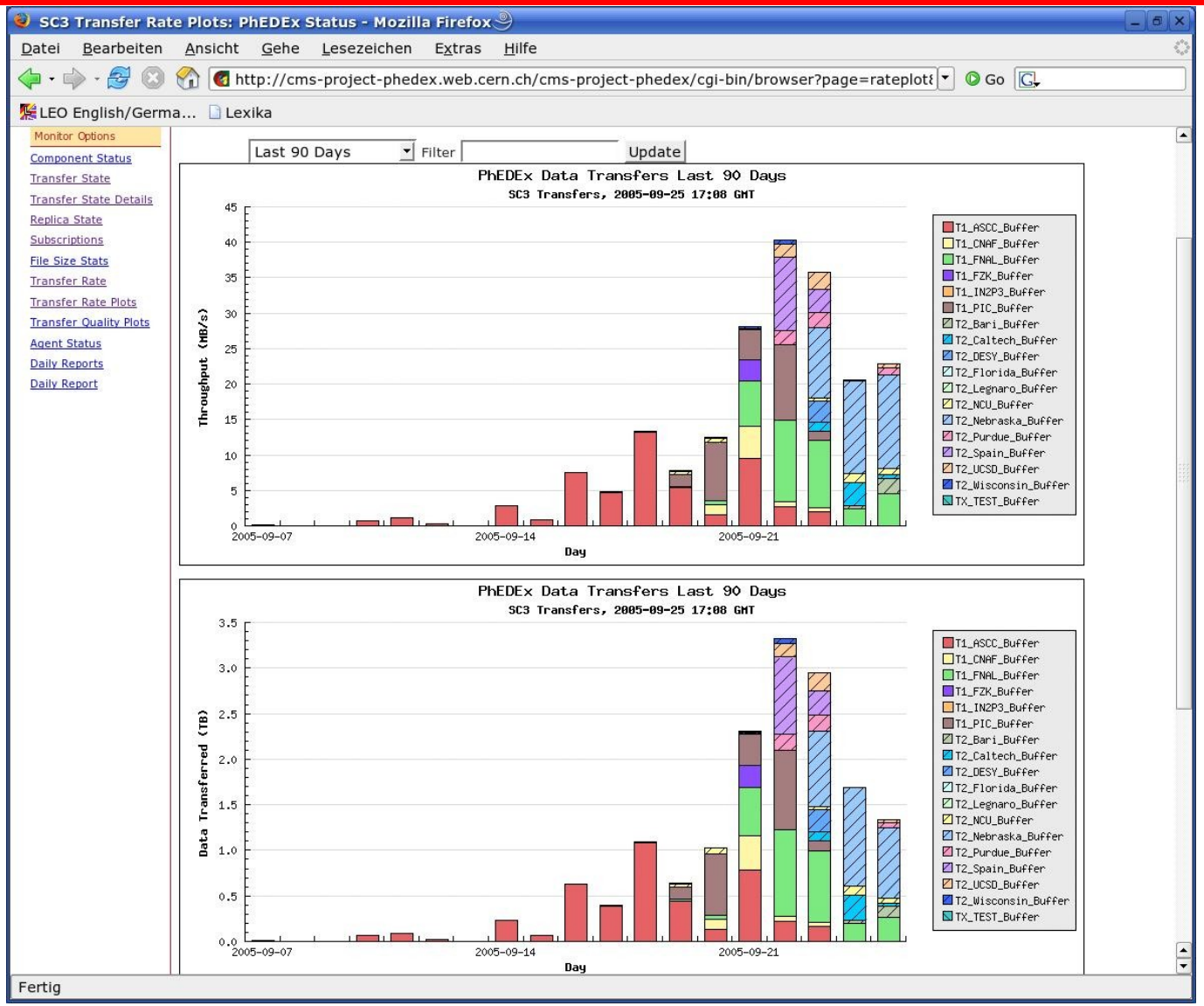
Age	Node	Destined		On Site		In Transfer		In Export	
		N	Size	N	Size	N	Size	N	Size
Current	T1_ASCC_Buffer	–	–	2287	4.0 TB	158	278.8 GB	2733	4.8 TB
Current	T1_ASCC_MSS	2446	4.3 TB	–	–	2287	4.0 TB	–	–
Current	T1_CERN_Buffer	–	–	5881	10.6 TB	–	–	5893	10.6 TB
Current	T1_CERN_MSS	–	–	26447	47.0 TB	–	–	–	–
Current	T1_CNAF_Buffer	–	–	338	600.0 GB	1681	2.9 TB	530	952.3 GB
Current	T1_CNAF_MSS	2020	3.5 TB	329	584.0 GB	9	16.0 GB	–	–
Current	T1_FNAL_Buffer	–	–	1600	3.0 TB	3300	6.0 TB	1002	1.7 TB
Current	T1_FNAL_MSS	4738	8.8 TB	1551	2.9 TB	–	–	–	–
Current	T1_FZK_Buffer	–	–	135	249.5 GB	214	372.3 GB	92	170.2 GB
Current	T1_FZK_MSS	349	621.8 GB	43	79.3 GB	92	170.2 GB	–	–
Current	T1_IN2P3_Buffer	–	–	10	13.1 GB	–	–	–	–
Current	T1_IN2P3_MSS	179	329.3 GB	10	13.1 GB	–	–	–	–
Current	T1_PIC_Buffer	–	–	1177	2.1 TB	535	1010.8 GB	18	32.2 GB
Current	T1_PIC_MSS	1718	3.1 TB	1167	2.1 TB	18	32.2 GB	–	–
Current	T1_RAL_MSS	316	501.1 GB	–	–	–	–	–	–
Current	T2_Bari_Buffer	558	959.0 GB	87	160.5 GB	221	393.4 GB	–	–
Current	T2_Caltech_Buffer	1132	1.9 TB	254	421.9 GB	81	145.1 GB	–	–
Current	T2_DESY_Buffer	–	–	135	249.5 GB	–	–	–	–
Current	T2_DESY_MSS	179	329.3 GB	135	249.5 GB	–	–	–	–
Current	T2_Florida_Buffer	1132	1.9 TB	–	–	335	567.0 GB	–	–
Current	T2_Imperial_Buffer	316	501.1 GB	–	–	–	–	–	–
Current	T2_Legnaro_Buffer	548	945.9 GB	–	–	300	542.9 GB	–	–
Current	T2_NCU_Buffer	666	1.2 TB	160	295.2 GB	446	790.8 GB	–	–
Current	T2_Nebraska_Buffer	4900	9.0 TB	1441	2.7 TB	149	278.0 GB	–	–
Current	T2_Purdue_Buffer	1132	1.9 TB	252	418.9 GB	83	148.2 GB	–	–

PhEDEx – monitoring
Replica state



Owner	Dataset	Runs		Files	T1_FNAL_Buffer		T1_FNAL_MSS	
			N	Size	N	Size	N	Size
bt_2x1033PU761_TkMu_2_3_4_g133_CMS	bt03_gg_bbh200_2taujmu	4	4	6.3 GB	4	6.3 GB	4	6.3 GB
bt_DST8713_2x1033PU_g133_CMS	bt03_gg_bbh200_2taujmu	2	2	2.5 GB	2	2.5 GB	2	2.5 GB
bt_Hit750_g133	bt03_gg_bbh200_2taujmu	4	4	4.3 GB	2	2.1 GB	2	2.1 GB
eg_2x1033PU761_TkMu_2_g133_OSC	eg03_jets_2g_pt50170	1508	1529	2.9 TB	842	1.6 TB	842	1.6 TB
eg_DST8713_2x1033PU_g133_OSC	eg03_jets_2g_pt50170	607	623	1.2 TB	381	751.2 GB	381	751.2 GB
eg_L25s8713_2x1033PU_g133_OSC	eg03_jets_2g_pt50170	83	83	163.2 GB	42	82.2 GB	42	82.2 GB
hg_2x1033PU761_TkMu_2_g133_OSC	hg03_H2mu_ma125_tb30	5	5	9.0 GB	1	1.9 GB	1	1.9 GB
hg_2x1033PU761_TkMu_2_g133_OSC	hg03_H2mu_ma130_tb30	5	5	9.4 GB	3	5.7 GB	3	5.7 GB
hg_2x1033PU761_TkMu_2_g133_OSC	hg03_H2mu_ma135_tb30	5	5	9.2 GB	3	5.7 GB	3	5.7 GB
hg_2x1033PU761_TkMu_2_g133_OSC	hg03_H2mu_ma150_tb15	4	4	6.0 GB	1	478.8 MB	1	478.8 MB
hg_2x1033PU761_TkMu_2_g133_OSC	hg03_H2mu_ma150_tb40	4	4	5.7 GB	1	1.9 GB	1	1.9 GB
hg_2x1033PU761_TkMu_2_g133_OSC	hg03_H2mu_ma150_tb50	4	4	6.2 GB	2	2.5 GB	2	2.5 GB
hg_2x1033PU761_TkMu_2_g133_OSC	hg03_H2mu_ma200_tb15	4	4	6.2 GB	1	1.9 GB	1	1.9 GB
hg_2x1033PU761_TkMu_2_g133_OSC	hg03_H2mu_ma200_tb40	4	4	6.2 GB	2	2.5 GB	2	2.5 GB
hg_2x1033PU761_TkMu_2_g133_OSC	hg03_H2mu_ma300_tb15	4	4	6.2 GB	1	635.2 MB	1	635.2 MB
hg_2x1033PU761_TkMu_2_g133_OSC	hg03_H2mu_ma300_tb30	4	4	6.2 GB	1	1.8 GB	1	1.8 GB
hg_2x1033PU761_TkMu_2_g133_OSC	hg03_H2mu_ma300_tb40	4	4	6.2 GB	1	639.5 MB	1	639.5 MB
hg_2x1033PU761_TkMu_2_g133_OSC	hg03_H2mu_ma300_tb50	4	4	6.0 GB	1	476.5 MB	1	476.5 MB
hg_2x1033PU761_TkMu_2_g133_OSC	hg03_H2mu_ma400_tb15	4	4	6.0 GB	4	6.0 GB	4	6.0 GB
hg_2x1033PU761_TkMu_2_g133_OSC	hg03_H2mu_ma400_tb30	4	4	6.0 GB	1	473.9 MB	1	473.9 MB
hg_2x1033PU761_TkMu_2_g133_OSC	hg03_qq_qqh120_inv	4	4	6.0 GB	2	2.4 GB	2	2.4 GB
hg_2x1033PU761_TkMu_2_g133_OSC	hg03_qq_qqh135_2taull	14	14	25.2 GB	9	16.6 GB	9	16.6 GB
hg_2x1033PU761_TkMu_2_g133_OSC	hg03_qq_qqh200_inv	4	4	6.0 GB	4	6.0 GB	4	6.0 GB
hg_2x1033PU761_TkMu_g133_CMS	hg03_gg_ch_170_tb20	3	3	2.9 GB	3	2.9 GB	3	2.9 GB
hg_2x1033PU761_TkMu_g133_CMS	hg03_hzz_2e2mu_130a	4	4	6.3 GB	2	3.8 GB	2	3.8 GB
hg_2x1033PU761_TkMu_g133_CMS	hg03_hzz_4e_150	4	4	5.8 GB	1	1.9 GB	1	1.9 GB

PhEDEx – monitoring
Transfer rate



PhEDEx – subscription
Create request