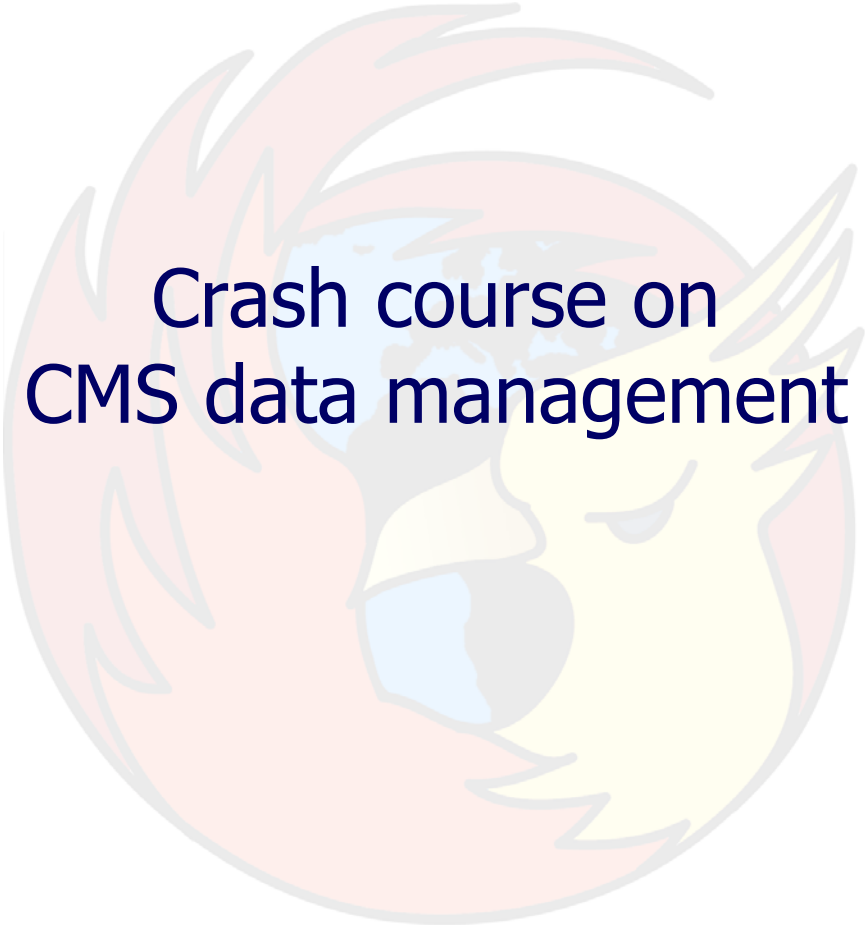
CMS File Transfers and Service Challenge 3

Lassi A. Tuura
Northeastern University

Talk structure

- CMS data management
  - Data management concepts
  - Data transfer component
  - Current operations
- Introduction to PhEDEx
  - Mission and Context
  - Design overview
  - Main components
  - Transfer handshake
  - Current Issues
- Service Challenge 3
  - CMS transfers
  - Site services
  - Schedule



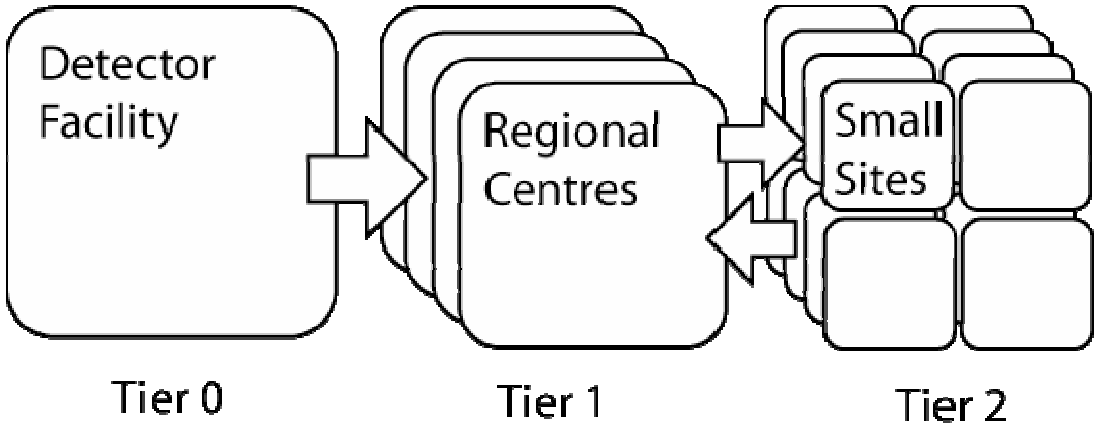
Crash course on CMS data management

Data management concepts

- Logical data organisation: online stream — primary dataset — dataset / data tier — event collection
  - Input to an application: a subset of dataset's event collections
  - Data management edge is event collection, data processing applications are required to look into details smaller than that
  - Placement of bulk data driven by policy and subscriptions
- Physical data organisation: site — block — file
  - Datasets broken down to blocks of O(5-10 TB)/O(1k-10k) files
  - Basic unit of experiment-wide data / storage management
- Main components
  - Dataset bookkeeping system: data organisation
  - Data location index: index of blocks at sites
  - Data transfer system: *this presentation*

CMS data management

Data flows



- Overall data flow
  - Detector data to Tier 1s, safe storage on tape, large-scale processing
  - Processed data to Tier 2s, smaller-scale analysis
  - Simulation and analysis results from Tier 2s cached at Tier 1s
- Overall infrastructure
  - Core infrastructure is a stable set of Tier 0, Tier 1 and Tier 2 sites
  - Dynamic infrastructure typically Tier 2 and smaller sites that are transient — each associating with a larger site

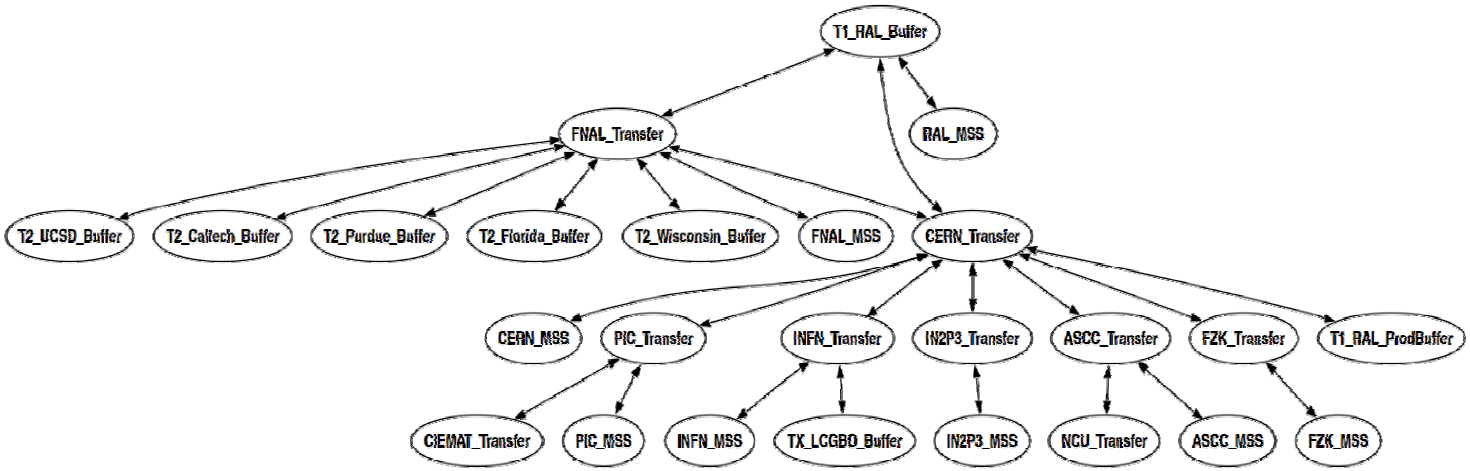
CMS data management

Data transfer component

- PhEDEx is CMS component for data transfers
  - Manages transfers from multiple sources to multiple destinations
  - Provides cost/latency/rate estimates for scheduling
- Main characteristics
  - Oriented for dataset blocks, not just files
  - Asynchronous transfers by agents
    - Not by hand, bulk, or on-demand by job access
  - Based on storage overlay network
    - Tape and disk storage nodes in a transfer graph
    - Factor in transfer policy using routing
    - End-to-end transfers, not just single hop
  - Grid- and other technology agnostic

Current operational sites

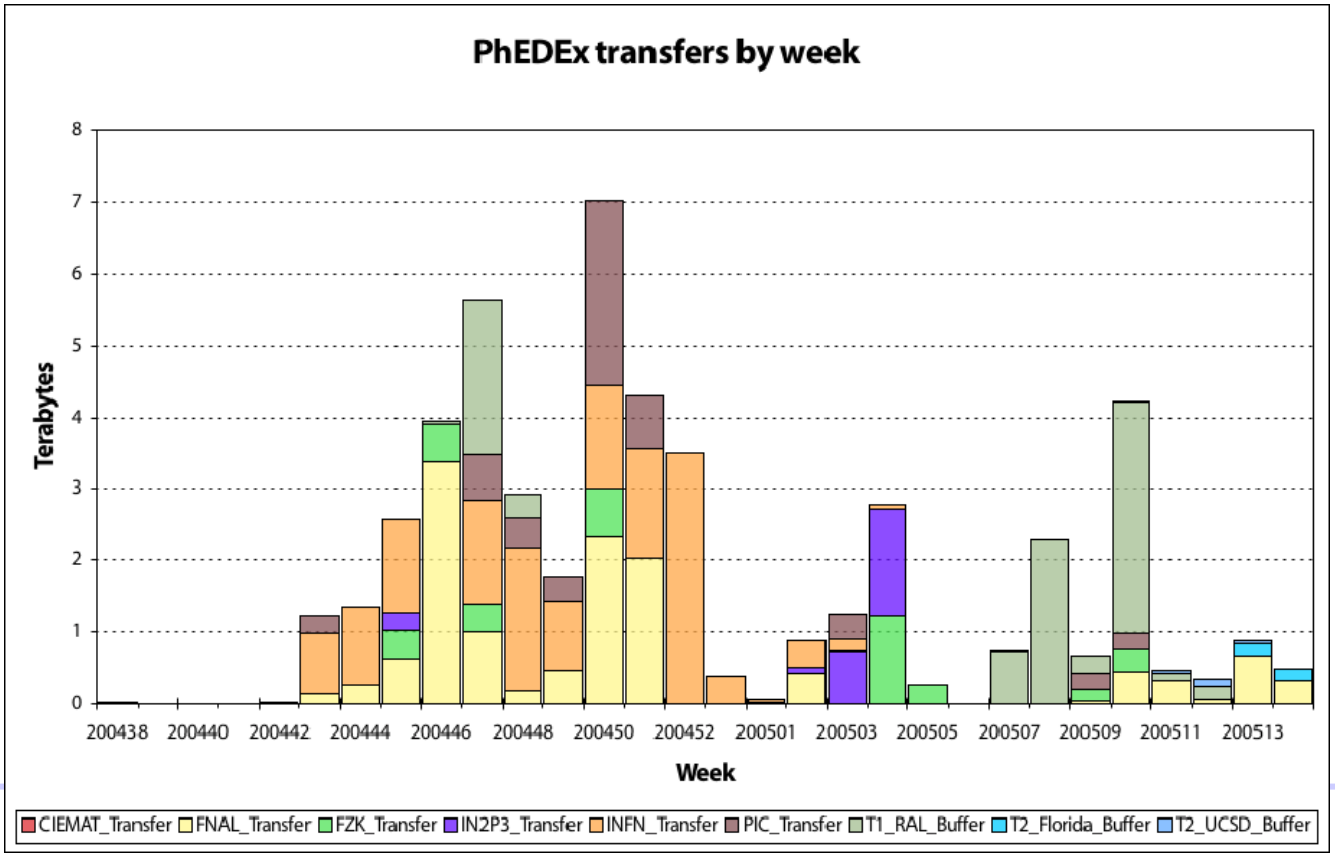
- 7 large sites: FNAL, CERN, INFN-CNAF, PIC, RAL, FZK, IN2P3; ASCC (Taiwan) coming onboard soon
  - Inbound transfers for all, export from CERN, FNAL, others testing
- Number of Tier-2 and other smaller sites, some testing
  - Spain (CIEMAT), Italy (Bari, Bologna), U.S. (UCSD, Florida, Wisconsin, Caltech, Purdue, MIT), U.K. (Imperial), NorduGrid (Finland, Estonia), Pakistan (NCP), Taiwan (NCU)



CMS data management

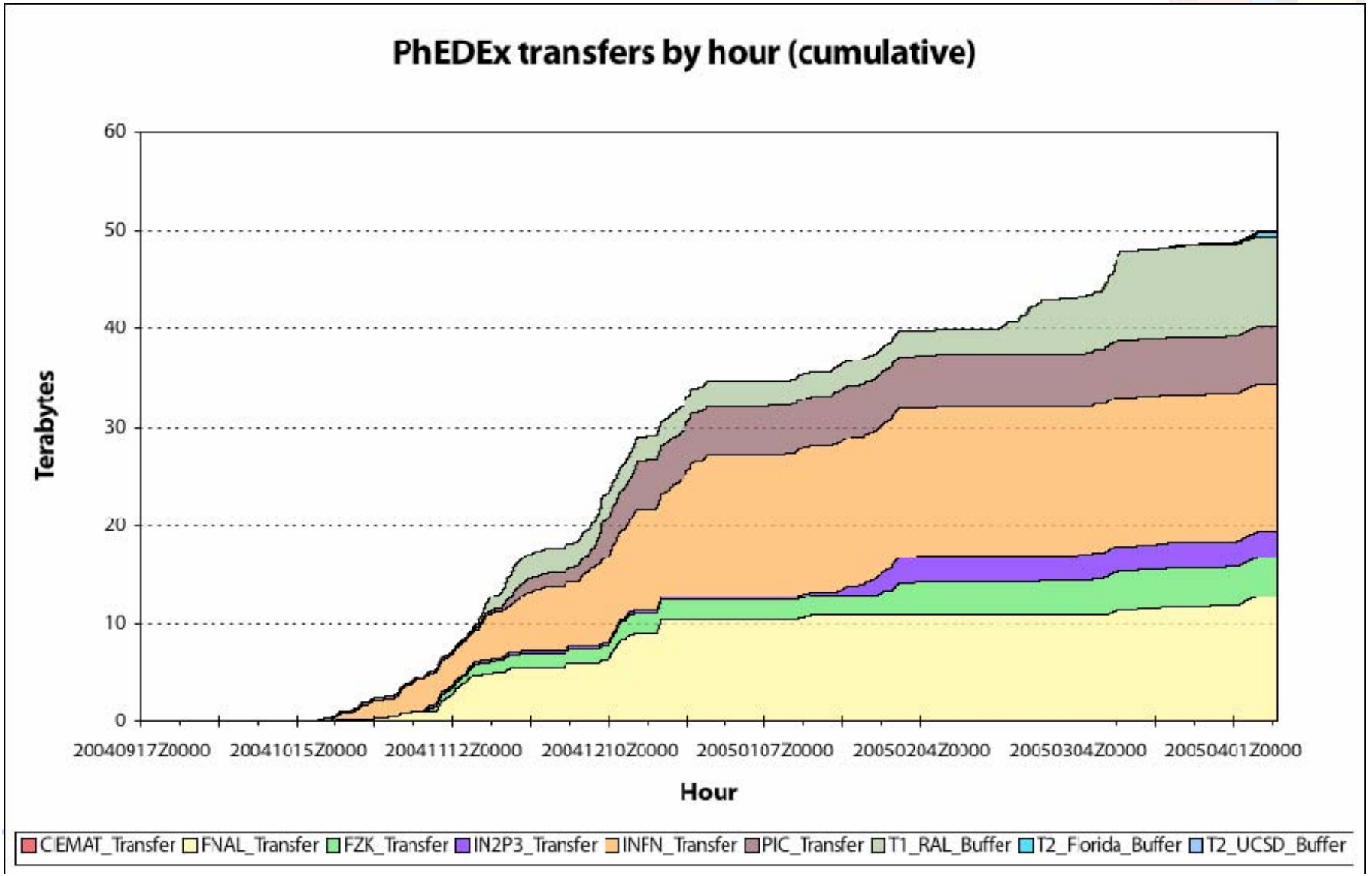
Current operational data

- Production: ~70 TB known, ~150 TB total replicated
- SC2: 1.6 PB — 1.6M replicas of 40 files (!)
- Test instances: 2 x testbed, integration test, castor test



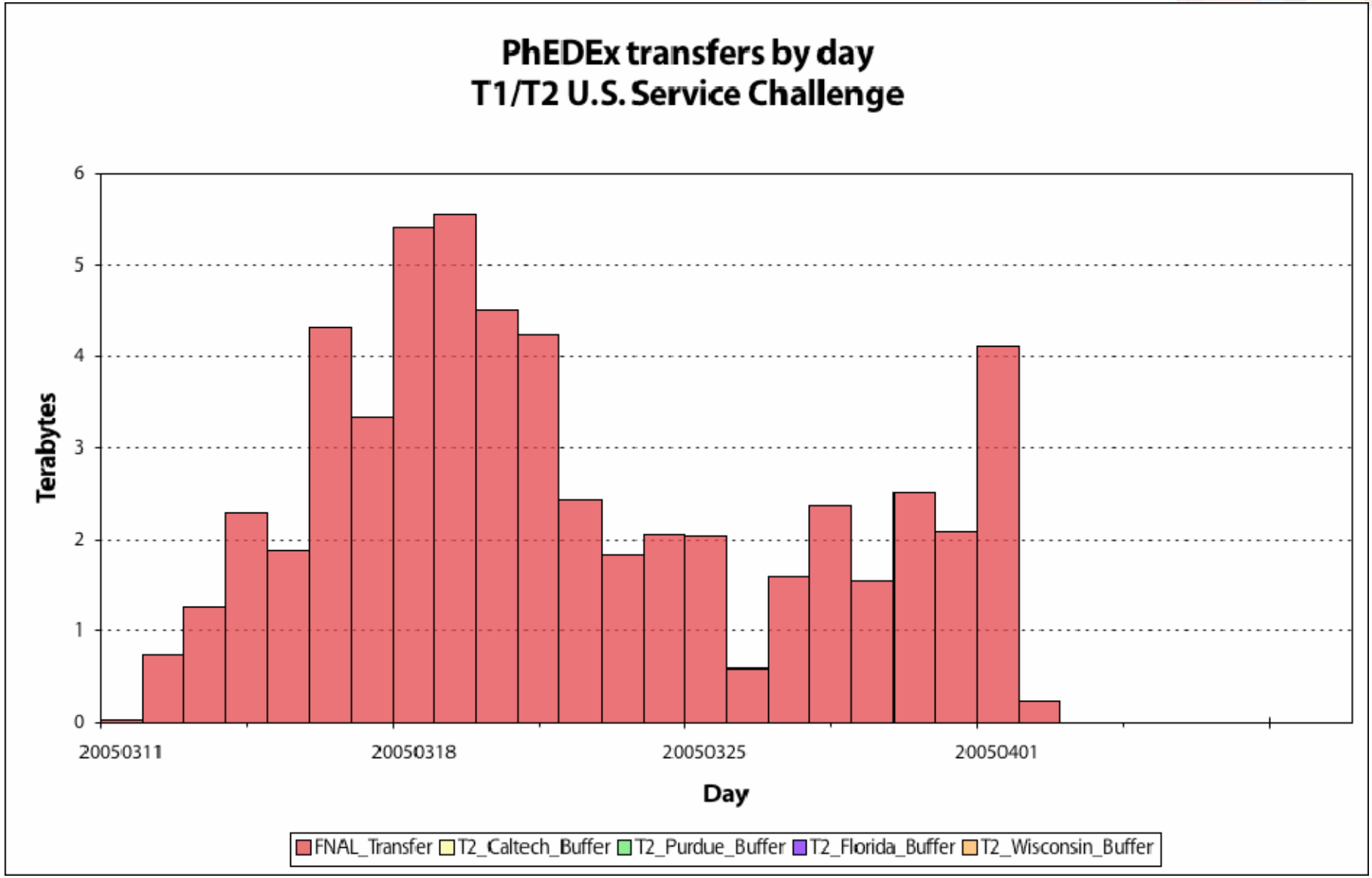
CMS data management

Current operational transfers



CMS data management

Current operational transfers



Introduction to PhEDEx

Mission

- PhEDEx started just before CMS DC04, ~ 1 year go
- Many solutions to data distribution in HEP experiments
  - Nothing directly met CMS requirements
  - Grid- and technology agnostic: respect local choices
    - But avoid solution proliferation too
  - Leverage existing experience and services that really work
  - Retain agility to evolve, replace technology and layers
- Context
  - CMS requirements
  - Other systems

CMS Requirements

- Managed and structured data flow
  - Not everyone can connect to detector facility, manage resource load
  - Distribution topology not fully connected: hybrid tree-mesh-star
  - Automate more sophisticated Tier 1 roles
    - Permanent safe storage of raw data copy
    - Serving raw and reprocessed data to Tier 2 sites
    - Data custody and caching of data produced or destined elsewhere
  - Higher-level view of multi-step transfers: tape, disk, disk, tape/disk, disk
  - Buffer management: only delete when files safe at destination
  - Ensure files have reached all destinations and custodial storage
  - High-level view of replica processing: stored on tape, checksummed, …
- Different data transfer modes
  - Push from detector to T1 tape; pull for requests, output harvesting
- Autonomous operation without continuous operator attention
- Different actors and systems: manage priority competition
  - Collaboration, physics groups, individuals; Tier 0 to laptop

Other Systems

- SAM(Grid) for CDF, D0
  - Strongly couples many aspects of experimental operation: dataset bookkeeping and auditing, transfers, workload management
  - Large scale data movements handled
  - Moves data in response to user demand
- EDG for LHC experiments and others
  - Much research into optimized on-demand replica management
  - No production-quality automated data management
    - Still only point-to-point, download-your-own
- CondorG + Stork
  - Again, coupled workload and data management
  - No automated data management, no background continuous data flow
- ATLAS Don Quixote + new reliable file transfer service
  - Parallel development with slightly different emphasis in detail?
- EGEE gLite: See later

Design overview

- Separation of data management layers
  - Dataset-level transfer management
  - Data hierarchy means to scale performance
  - Routed multi-hop transfers: topology, replica choice, policy
  - Reliable point-to-point transfers: transfer handshake
    - All transfer tools treated as fundamentally unreliable
- Local information stays local
  - Deletion or other file loss are not local
  - PFN, paths, host names, catalogues are local information
- Agent-based
  - Complex functionality in discrete, lightweight and disposable units
  - Minimal handover between units at clearly specified points
  - Autonomous and peer-to-peer computing benefits
- Two overlay networks: a) *storage overlay* with IP-style routing where node = storage, edge = transfer step, edge state = progress, b) *agent communication overlay,* today via central database

Main components / layers

Request management- dataset level transfers
Scalable management and monitoring of transfer requests.
Automated allocation of files to destinations to fulfill requests.
Dynamic routing alterations to avoid problems.
Automatic harvesting of files; bulk transfer requests for existing data.

Reliable routed, or multi-hop, transfer
Efficient handover of responsibility from node to node in a transfer chain.
Manage clustering of tape stages and migrations.
Determination of closest/ best replica for transfer.

Reliable point-to-point, or single hop, transfer
Failure recovery and retry of transfers.

Unreliable point to point transfers and technologies
srmcp, globus-url-copy, lcg-rep, dccp
srm, gsiftp, dCache

+ Higher levels: transfer request management and tracking

Life of a transfer

- Before the transfer
  - NodeRouters maintain transfer topology, time out dead nodes / routes
  - FileAllocator assigns files to destinations using subscriptions
  - For each file destination assignment, destination FileRouter finds best replica and creates single-hop transfer assignments
- The transfer assignment
  - States: assigned, wanted, available, in transfer, completed, error
  - Everything is a pull: dead sites are ignored (except allocation failover)
  - Wanted = sliding window to allow exporting side plan stage-in
  - Available = exporter tells file on disk, provides transfer URL
  - Configurable number of parallel transfers, can use copyjobs
  - Evaluate transfer success: compare file size, possibly checksum
- After transfer
  - Failed: back off, tick error counts, schedule for later retry
  - Success: hand over locally (CMS: publish to catalogue), route next hop

Other properties

- General assumption: every operation will fail
  - Surprisingly accurate estimate, innumerable errors exposed in tools
  - Assume most errors are transient: disk full, network down, …
    - Log an alert, back off, retry later
- Designed to be tested
  - Just about every operation and component can be faked out
    - Useful for both testing and what-if analysis
    - Laptop development and testing fully plausible
  - Test everything on developer testbed, then in integration testbed
    - Production system "switched over overnight" after integration
  - Regularly used for validation testing of other components
- Rich amount of tracking information, monitoring
  - Transfer history for rate and progress estimation
  - Agents log output to disk in semi-standard formats for summaries
    - Now also testing distributed access to the logs for remote monitoring

EGEE gLite

Request management- dataset level transfers
Scalable management and monitoring of transfer requests.
Automated allocation of files to destinations to fulfill requests.
Dynamic routing alterations to avoid problems.
Automatic harvesting of files; bulk transfer requests for existing da

Reliable routed, or multi-hop, transfer
Efficient handover of responsibility from node to node in a transfer chain.
Manage clustering of tape stages and migrations.
Determination of closest/ best replica for transfer.

Reliable point-to-point, or single hop, transfer
Failure recovery and retry of transfers.

Unreliable point to point transfers and technologies
srmcp, globus-url-copy, lcg-rep, dccp
srm, gsiftp, dCache

CMS specific management layers

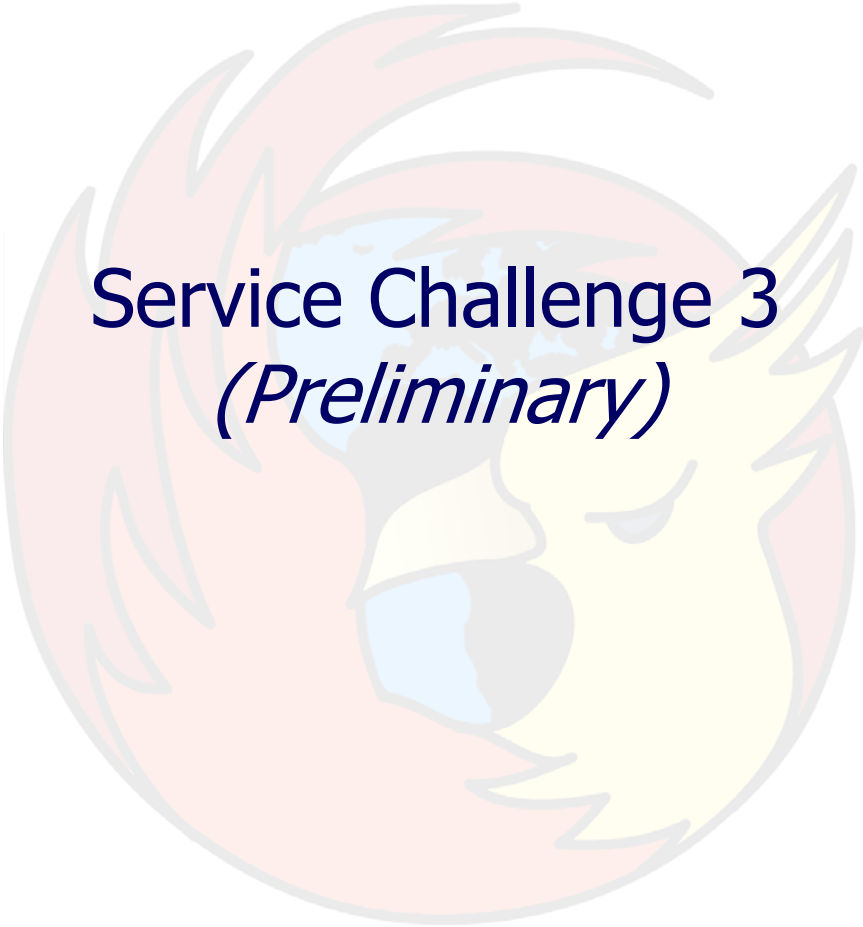
EGEE gLite File Transfer Service?

Future directions

- Database and agent topology
  - Database deployment improvements
  - Peer-to-peer overlay for data location, transient / small nodes
- Dynamic contractual file routing
  - Request/tender with time validity
  - Choose best replica, handle failing routes, congestion
- Priority and policy
  - Function of collaboration, site and data requestor priorities
  - Overall path priority, local transfer priorities, buffer management
- Semi-autonomy and interaction with fabric management
  - Respond to local conditions and adapt
  - Detect and message on catastrophic failure
- Continued technology testing, what-if analysis

Current issues

- PhEDEx is CMS production data transfer system
  - Maturing now, large-scale transfers are getting simpler
  - Able to sustain TB/day+ transfers, PhEDEx not bottleneck (1%, max 10)
  - Most sites beginning to keep agents up much of the time unattended
  - TMDB only current single point of failure
- Observations, major focus required
  - Underlying infrastructure is maturing slowly
    - At any one time 1/3 of the transfer system is usually down
    - Good news: transfers don't stop, local management possible
  - Exporting data is much harder than importing it
    - Very difficult to play fair with current Castor at CERN
    - SRM-to-SRM transfer incompatibilities
    - Every site has a different infrastructure configuration
  - Exporting, importing and serving data simultaneously painful
    - Poorly understood issues with just importing!
    - Disk-to-disk is only so interesting, we are already doing tape-to-tape…



Service Challenge 3
(Preliminary)

Service Challenge 3

CMS transfers

- PhEDEx will be used for SC3 CMS transfer tests
  - Available to help set up if others have interest
  - Not the only CMS service that needs setting up at the sites
- Transfer features expected to be tested
  - Simultaneous data import, export and serving for local processing
  - Must be representative of real experiment data flow
  - Must use realistic files and realistic storage
    - *This will become the next production service, right?*
    - To/from tape at least on some Tier 1 sites
    - We are working on file size
- Cannot afford to fail
  - Suggest testing a couple of different configurations according to region/site preferences: SRM-SRM transfers, GridFTP only, FTS
  - EGEE FTS not a high priority for CMS, may be for some sites?
    - *Risk for using for all sites is too high, not clear why for e.g. U.S.*

Service Challenge 3

Site services

- Transfers: import and export
  - For CMS tests, using PhEDEx installation at site
- Serving data to bulk data processing applications
  - Simulated and/or real applications
  - This requires several other services to be available
    - Computing element, job submission, output harvesting for transfer, software installation + publishing into the information system, bookkeeping / monitoring databases for production, file catalogue, PubDB or successor
- The above are expected to be available concurrently
  - Throughput phase: concurrent import/export transfers only
  - Service phase: all, but top throughput not required

Service Challenge 3

Schedule

- CMS will participate in the "early phase" (cf. Jamie)
- July: throughput phase
  - T0/T1/T2 simultaneous import/export
  - To and from tape at T1s
  - Real files, real storage
- August: setup phase
  - Agents work while we all enjoy our holidays?
- September: service phase 1 — modest throughput
  - Demonstrate bulk data processing, simulation at T1, T2s
    - Requires software, job submission, output harvesting, monitoring, …
    - Not everything everywhere, something reasonable at each site
- November: service phase 2 — modest throughput
  - Phase 1 + continuous data movement
  - Precursor for next production service

Summary

- Data transfers is a substantial topic
  - The interesting world is beyond "SRM-or-FTS?"…
- CMS has a production data transfer system
  - Many major and smaller sites already involved
  - Handles large scale continuous transfers, mostly on background
  - Relatively low overhead
  - Planning next steps
- Significant amount of work to ramp up everything
  - Major issues remain to be sorted out
  - That's why we are here, doing service challenges
  - That's why CMS visit sites directly for technical contact
  - We have to press on, service challenge or not
  - It's an exciting time, but have to move swiftly :-)

More Information

- PhEDEx
  - http://cern.ch/cms-project-phedex
    - In particular: “Documentation”, “Monitoring”, “Wiki”
    - Want to test? There are deployment and tuning guides…
    - More details in Tim Barrass’ presentation last Monday (April 4)
  - cms-phedex-developers@cern.ch
- CMS data management
  - Project leader: Peter Elmer <peter.elmer@cern.ch>
  - cms-dm-developers@cern.ch