

Distributed Data Management in CMS



Preparation for
Computing, Software, and Analysis Challenge
in September 2006
A.K.A. CSA06

Lee Lueking
CMS Activities Report

May 16, 2006



CMS Computing model

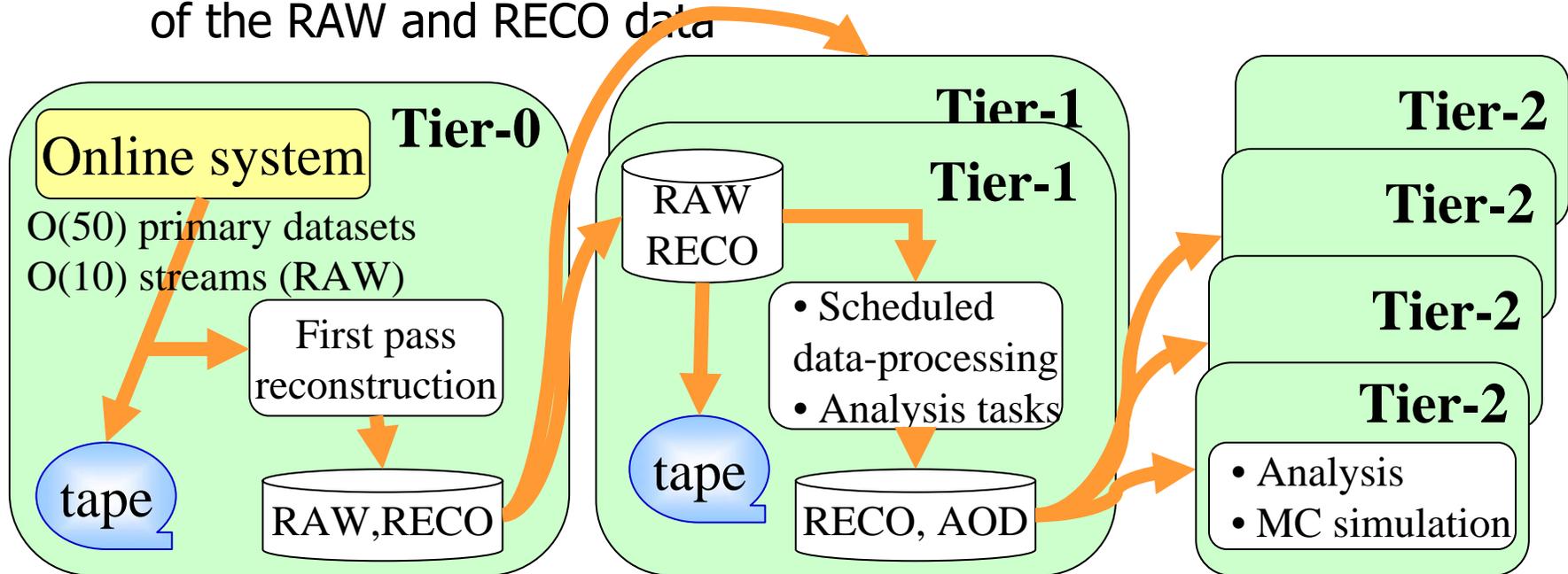


► CMS Computing model as described in CTDR

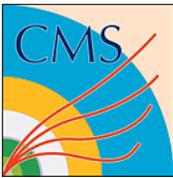
* Online system

- ◆ RAW events are classified in $O(50)$ primary datasets depending on their trigger history
- ◆ Primary datasets are grouped into $O(10)$ online streams

* Distributed model for all computing including the serving and archiving of the RAW and RECO data



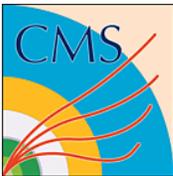
→ Data Management: provide tools to discover, access and transfer event data



General principles



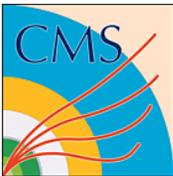
- ▶ Optimization for read access
 - ✱ written once, never modified and subsequently read many times
- ▶ Optimise for the large bulk case
 - ✱ management of very large amounts of data and jobs in a large distributed computing system
- ▶ Minimise the dependencies of the jobs on the Worker Node
 - ◆ Application dependencies local to the site to avoid single points of failure for the entire system
 - ◆ asynchronous handling of job output relative to the job finishing
- ▶ Site-local configuration information should remain site-local
 - ✱ flexibility to configure and evolve the local system as needed without synchronisation to the rest of the world



Data organization



- ▶ CMS expects to produce large amounts of data (events)
 - ✱ O(PB)/year
- ▶ Event data are in files
 - ✱ average file size is kept reasonably large (\geq GB)
 - ◆ avoid scaling issues with storage systems, catalogues when dealing with too many small files
 - ◆ foresee file merging
 - ◆ $O(10^6)$ files/year
- ▶ Files are grouped in Fileblocks
 - ✱ group files in blocks (1-10TB) for bulk data management reasons
 - ✱ 10^3 Fileblocks/year
- ▶ Fileblocks are grouped in Datasets
 - ✱ Datasets are large (100TB) or small (0.1TB) : size driven by physics



Dataset Bookkeeping System (DBS)

Fermilab



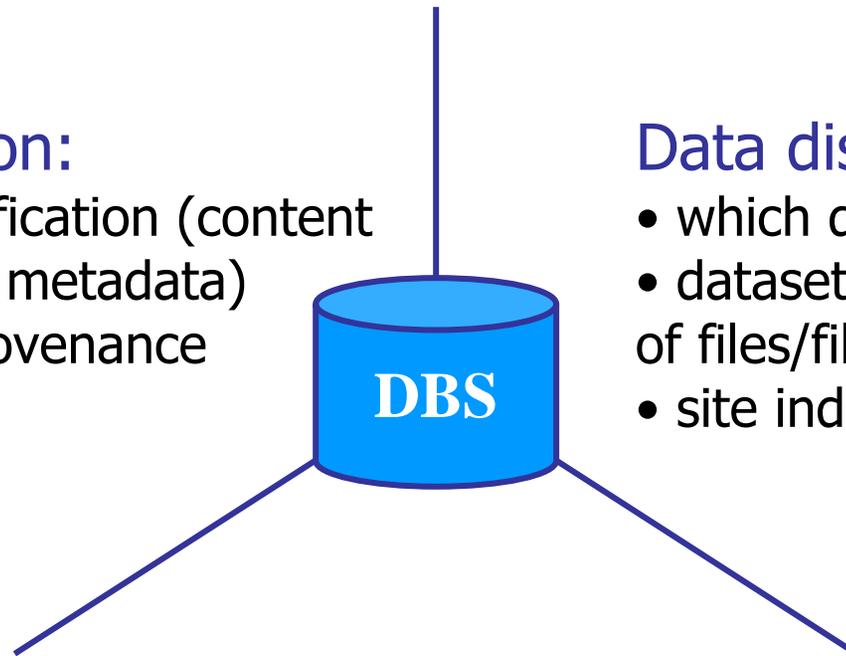
- ▶ The Dataset Bookkeeping System (DBS) provides the means to define, discover and use CMS event data

Data definition:

- dataset specification (content and associated metadata)
- track data provenance

Data discovery:

- which data exists
- dataset organization in term of files/fileblocks
- site independent information

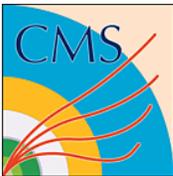


Use:

- Distributed analysis tool (CRAB)
- MC Production system

DBS Wiki has additional info

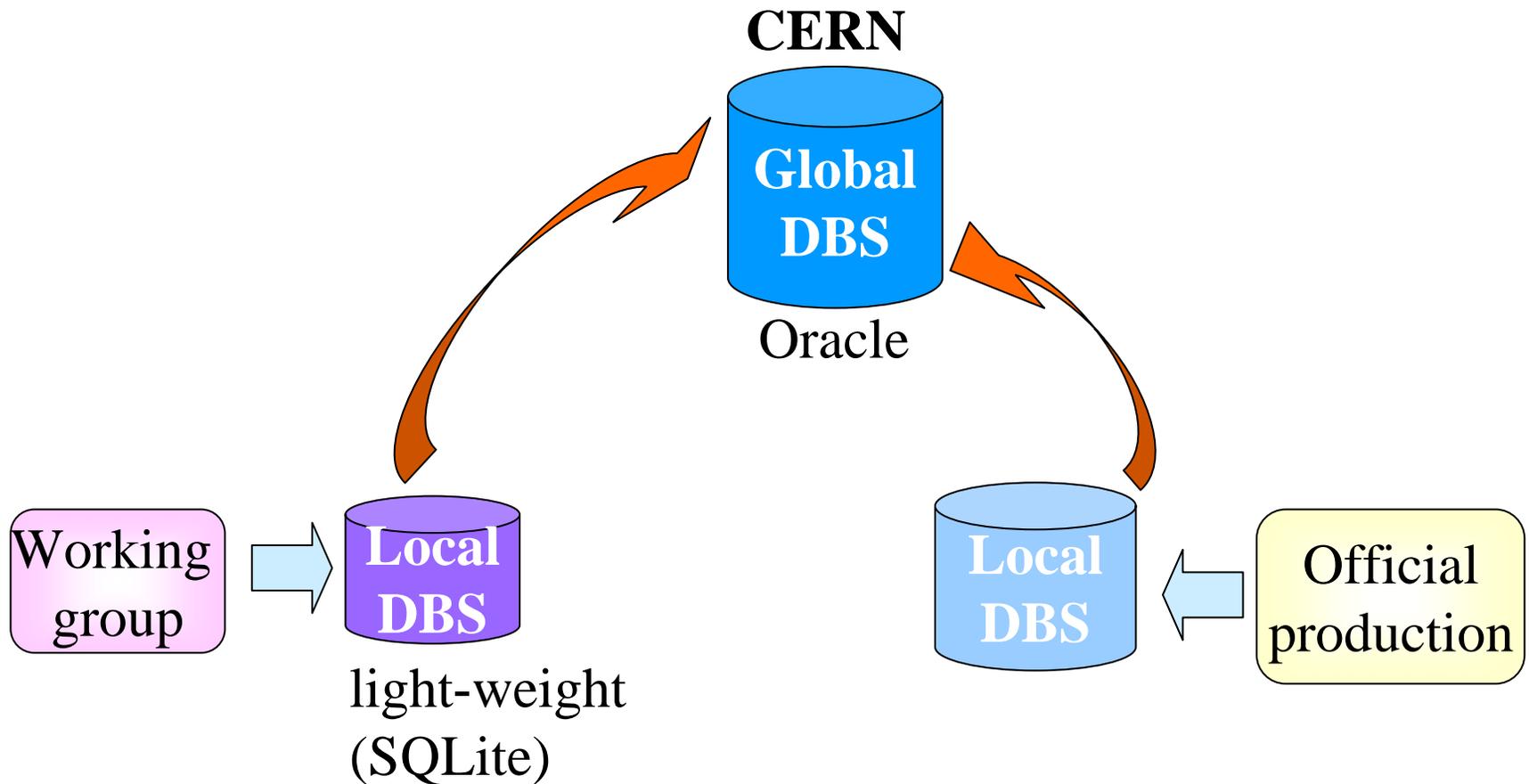
<https://uimon.cern.ch/twiki/bin/view/CMS/DBS-TDR>

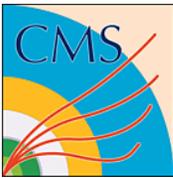


DBS scopes and dynamics



- ▶ A DBS instance describing data CMS-wide (Global scope)
- ▶ DBS instances with more “local” scope





DBS API (Object defs)



- ▶ DbsApplication
- ▶ DbsApplicationConfig
- ▶ DbsEventCollection
- ▶ DbsFile
- ▶ DbsFileBlock
- ▶ DbsParent
- ▶ DbsPrimaryDataset
- ▶ DbsProcessedDataset
- ▶ DbsProcessing
- ▶ DbsProcessingPath

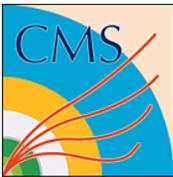
Complete description is on the Wiki at
<https://uimon.cern.ch/twiki/bin/view/CMS/CMS-DMS-DBS-CURRENT-API>



The DBS API (Calls)



- ▶ Create: PrimaryDataset, ProcessedDataset, Processing
- ▶ Insert: EventCollections, InsertFiles
- ▶ Get: DatasetContents, DatasetFileBlocks, DatasetProvenance
- ▶ List: ApplicationConfigs, Applications, ParameterSets, PrimaryDatasets, ProcessedDatasets

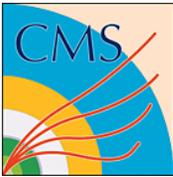


Data Discovery

(example) <http://cmsdoc.cern.ch/~sekhri/Html/query.htm>



- ▶ This is just a simple approach to get us started.
- ▶ Provides web-based access to the API calls.
- ▶ Can select from multiple instances (e.g. production aka RefDB, MCLocal, development)
- ▶ Info for RefDB data is limited so not all queries work.
- ▶ Useful for seeing what is in the database.
- ▶ Your experience may vary depending on which browser you use.



Data Discovery (continued)



Mozilla Firefox

File Edit View Go Bookmarks Tools Help

http://cmsdoc.cern.ch/~sekhri/Html/query.htm

Windows Customize Links Free Hotmail Windows Media

Processed Dataset

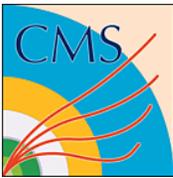
Id	Path
633	/jm03b_qqH600_WW_lnu/Hit/jm_Hit245_2_g133
634	/jm03b_qqH600_ZZ_2l2j/Hit/jm_Hit245_2_g133
635	/jm03b_qqH600_ZZ_4l/Hit/jm_Hit245_2_g133
636	/jm03b_qqH600_ZZ_llvv/Hit/jm_Hit245_2_g133
637	/jm03b_qqH700_WW_lnujj/Hit/jm_Hit245_2_g133
638	/jm03b_qqH700_WW_lnu/Hit/jm_Hit245_2_g133
639	/jm03b_qqH700_ZZ_2l2j/Hit/jm_Hit245_2_g133
640	/jm03b_qqH700_ZZ_4l/Hit/jm_Hit245_2_g133
641	/jm03b_qqH700_ZZ_llvv/Hit/jm_Hit245_2_g133
642	/jm03b_qqH800_WW_lnujj/Hit/jm_Hit245_2_g133
643	/jm03b_qqH800_WW_lnu/Hit/jm_Hit245_2_g133
644	/jm03b_qqH800_ZZ_2l2j/Hit/jm_Hit245_2_g133
645	/jm03b_qqH800_ZZ_4l/Hit/jm_Hit245_2_g133
646	/jm03b_qqH800_ZZ_llvv/Hit/jm_Hit245_2_g133
647	/jm03b_qqH900_WW_lnujj/Hit/jm_Hit245_2_g133
648	/jm03b_qqH900_WW_lnu/Hit/jm_Hit245_2_g133
649	/jm03b_qqH900_ZZ_2l2j/Hit/jm_Hit245_2_g133
650	/jm03b_qqH900_ZZ_4l/Hit/jm_Hit245_2_g133
651	/jm03b_qqH900_ZZ_llvv/Hit/jm_Hit245_2_g133

Done

Start

Re: Reasonable way to ... Mozilla Firefox DataManagement Microsoft PowerPoint - [...] Netscape

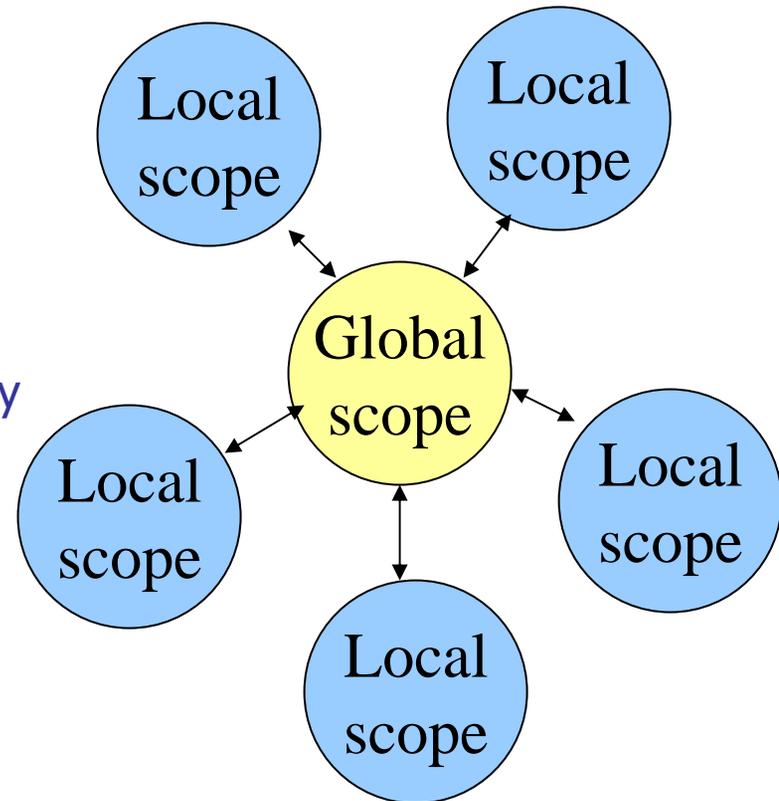
11:04 AM

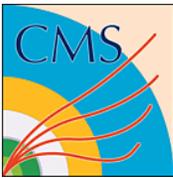


Additional API work in progress



- ▶ Import data from Global-scope DBS to Local-scope
- ▶ Publish data to Global-scope DBS from local-scope
- ▶ Remap merged file parentages
- ▶ Additional needs for PhEDEx, CRAB, EDM/Framework as understood.
- ▶ Schema migration tools to maintain availability of old data.
- ▶ API versioning to maintain backward compatibility with old API.





Schedule for CSA06



▶ Now:

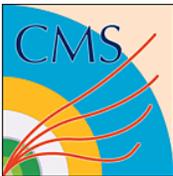
- ▶ * a) Global and local schema instances in place at CERN for testing,
 - ▶ * b) stable client API in Python ready for adding and retrieving data from DB instance,
 - ▶ * c) prototype service using CGI w/ Apache server in place,
 - ▶ * d) web forms-based browser page for data discovery.
- ▶ May 12: File merge remapping API call prototype for testing.
 - ▶ May 26: Global to Local scope import prototype.
 - ▶ June 9: Local to Global scope publishing prototype.
 - ▶ Mid-June to Mid-August: Further integration w/ MC Production, CRAB, PhEDEx. Local scope DB instance w/ MySQL DB.
 - ▶ Additional user documentation. Testing and refinement. Minor schema migrations as needed.



Beyond CSA06



- ▶ Four people from Cornell (~2 FTE equiv.) are helping us develop a more advanced picture of analysis (Dan Riley, Valentin Kuznetsov, Drew Dolqert, Michael Padula) :
 - ✱ Analysis use cases
 - ✱ Testing existing machinery and API
 - ✱ Bringing CLEO analysis experience to CMS
- ▶ Plan to have a small DBS/Analysis workshop in July (possibly at Cornell).
- ▶ Probably not directly applicable to CSA06, but minor improvements would not be excluded.
- ▶ Discussion w/ MTCC (Magnet Test/ Cosmic Challenge) and TB (Test Beam 06) people about possibly cataloging data to gain experience w/ detector data, interfacing w/ storage manager, et cetera.



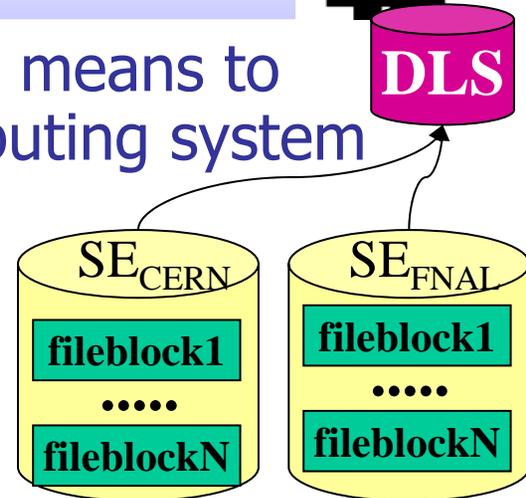
Data Location Service & Local data access

Alessandra Fanfani INFN



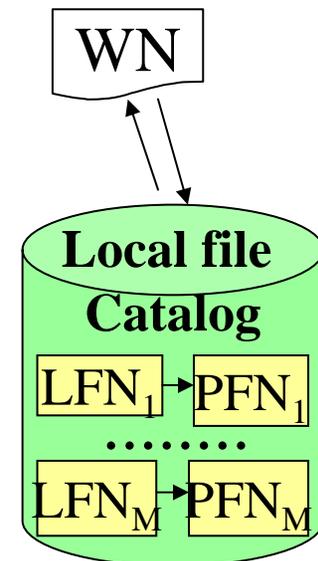
▶ The Data Location Service (DLS) provides the means to locate replicas of data in the distributed computing system

- ✱ it maps file-blocks to storage elements (SE's)
- ✱ few attributes (*custodial* replica = considered a permanent copy at a site)

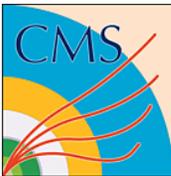


▶ Local data access

- ✱ DLS only has names of sites hosting the data and not the physical location of constituent files at the sites
- ✱ Local file catalogues provides site local information about how to access any given logical file name
- ✱ Support for site local discovery mechanism: discover at runtime on WN the site-dependent job configuration to access data



DLS twiki: <https://twiki.cern.ch/twiki/bin/view/CMS/DLS>



Functionalities available: global DLS



▶ Global DLS server

- ✱ all custom prototype based on MySQL (placeholder for evaluation of Grid catalogues)
 - ◆ python server with MySQL backend + clients tools
 - ◆ no authentication/authorization mechanisms
- ✱ LCG LFC server
 - ◆ LCG LFC server with Oracle backend + CMS custom clients code
 - ◆ Support authentication/authorization
 - ◆ Support Data Location Interface (DLI) for query used by WMS for match-making. No secure connections with the clients \Rightarrow faster query
 - ◆ Few attribute (custodial flag...)
 - ◆ Performance testing satisfy expected DLS requirements
- ✱ Filled with the location of old EDM data to allow CRAB to determine sites hosting the required data

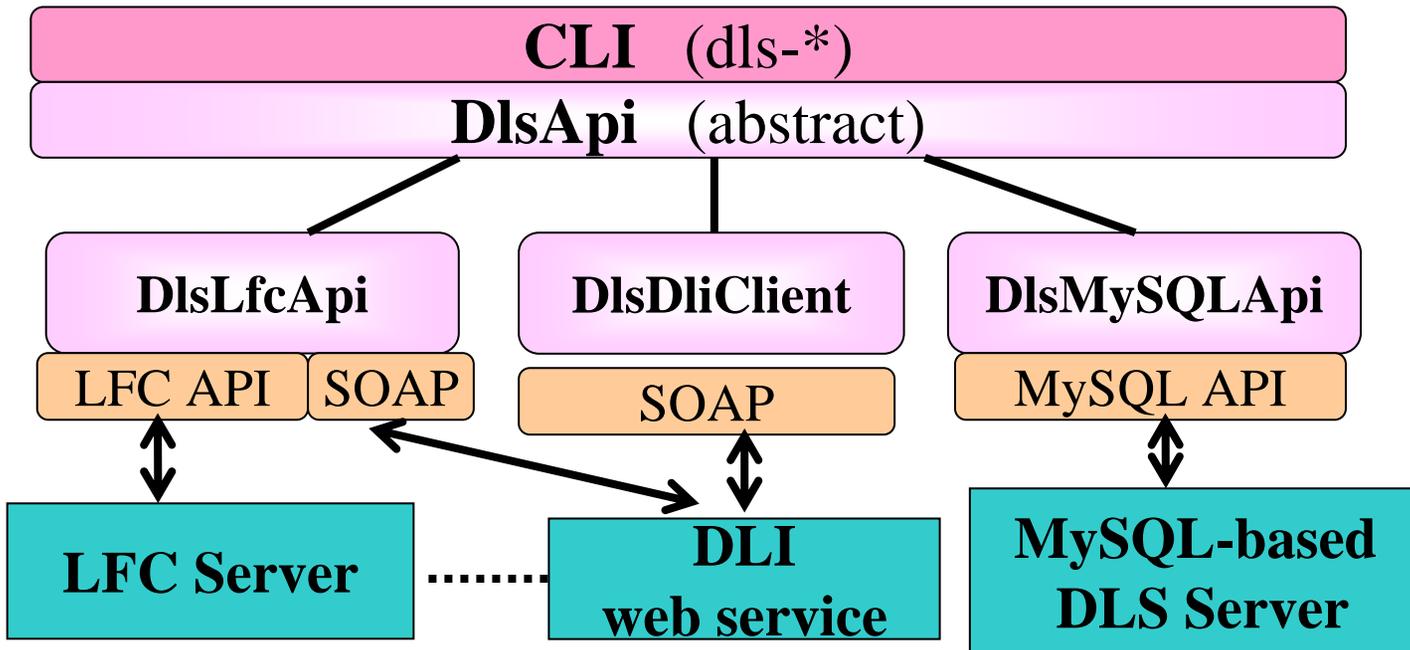


DLS client + stress test suite

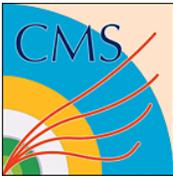


▶ DLS Client API-CLI:

- * query to locate file-blocks, insert/delete fileblock and their location, update attributes
 - ◆ Abstract interface with implementations specific for different DLS server



- ▶ Stress test suite to simulate access patterns, measure performances, stability



Functionalities for CSA06



▶ DLS operations in CSA06

- ✱ Insert produced file-blocks upon data processing at a site
- ✱ Insert file-blocks upon data replication
- ✱ Query to find file-block locations

▶ Functionalities planned for CSA06

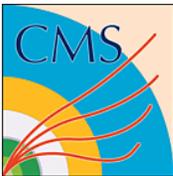
- ✱ Global DLS with the existing functionalities
- ✱ Integration with PHEDEX
- ✱ Integration with MC production: local scope DLS



Performance metrics for CSA06



- ▶ reach 50K jobs a day across all CMS sites (Tier1+Tier2)
 - ✱ DLS query rate:
 - ◆ Unrealistic scenario with 1 query/job the DLS average queries rate : $O(0.6\text{Hz})$
 - ◆ More realistic scenario with queries grouped by file-block, i.e. 100 jobs/file-block $O(0.006\text{Hz})$
 - ✱ DLS insert rate: produced data volume / file-block size
- ▶ data transfer metrics:
 - ✱ Tier-0 → Tier1 : 300 MB/sec
 - ◆ DLS insert rate: $0.0003\text{Hz}/X$ with X = file-block size in TB
 - ✱ Tier-1 → Tier2 : $\sim 10\text{MB/s}$ for all Tier-2s to 100MB/s to best connected Tier-2s
 - ◆ DLS insert rate: $0.00001\text{-}0.0001\text{Hz}/X$ with X = file-block size in TB
- ▶ Preliminary tests show these numbers are easily achieved. Plan for use test suite simulating the CSA06 scenario.

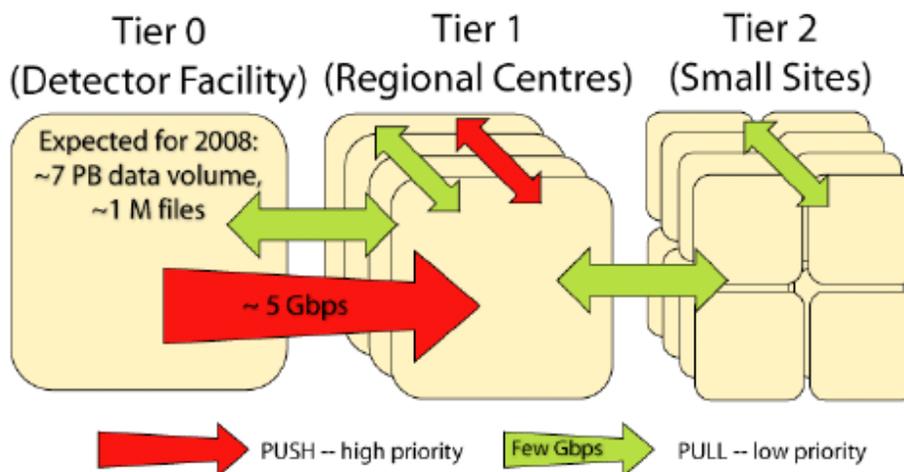


Data placement and transfer (PhEDEx)

Lassi Tuura (USCMS)



- ▶ Data distribution among Tier-0, Tier-1s, Tier-2s sites
- ▶ PhEDEx Data placement
 - * administrative decision of data movement (how many copies and where, which ones are custodial copy,...)
 - * management at file block level
- ▶ PhEDEx Data transfer
 - * Replicating and moving files
 - * Transfer nodes with site-specific replication and tape management

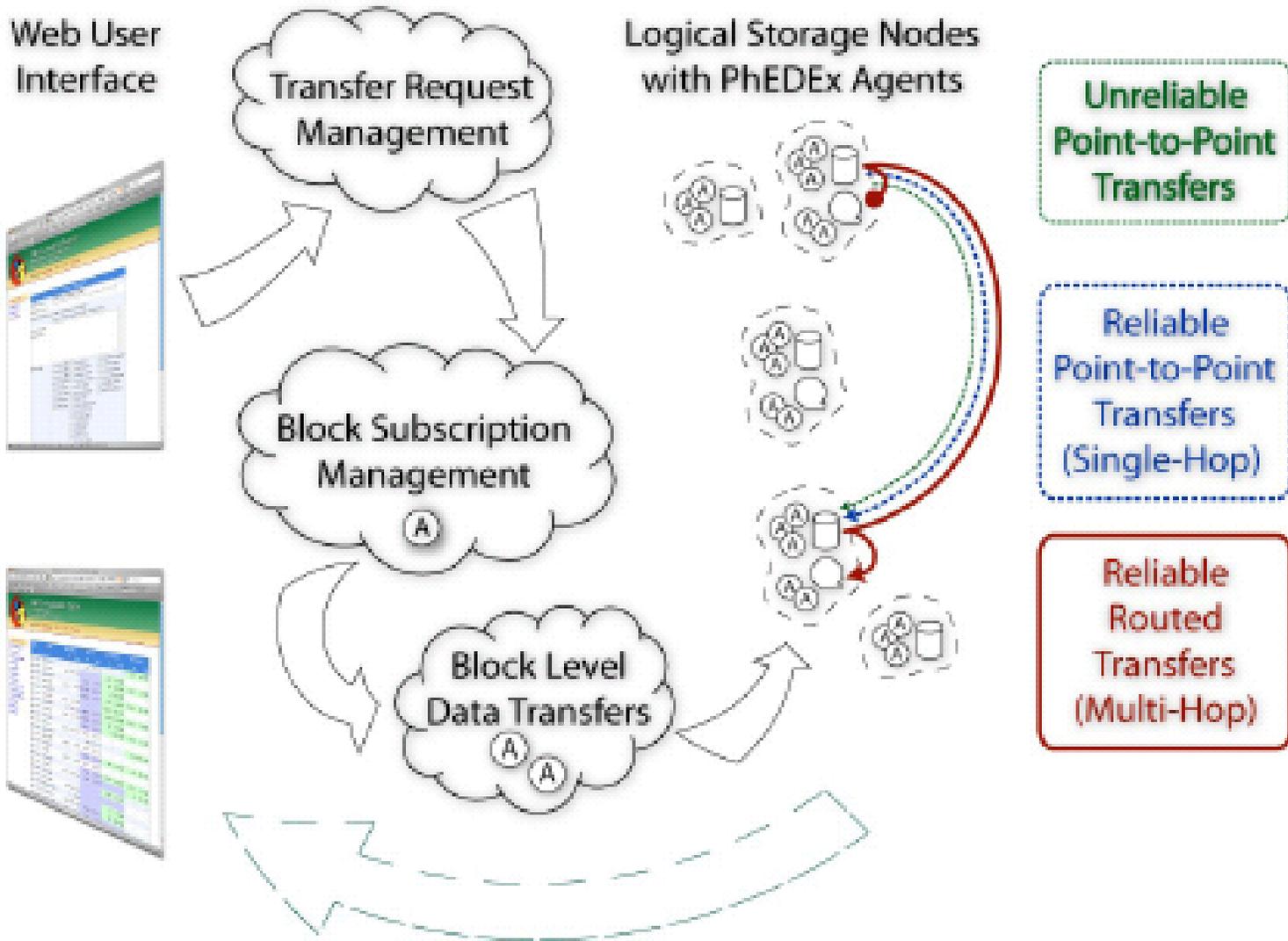


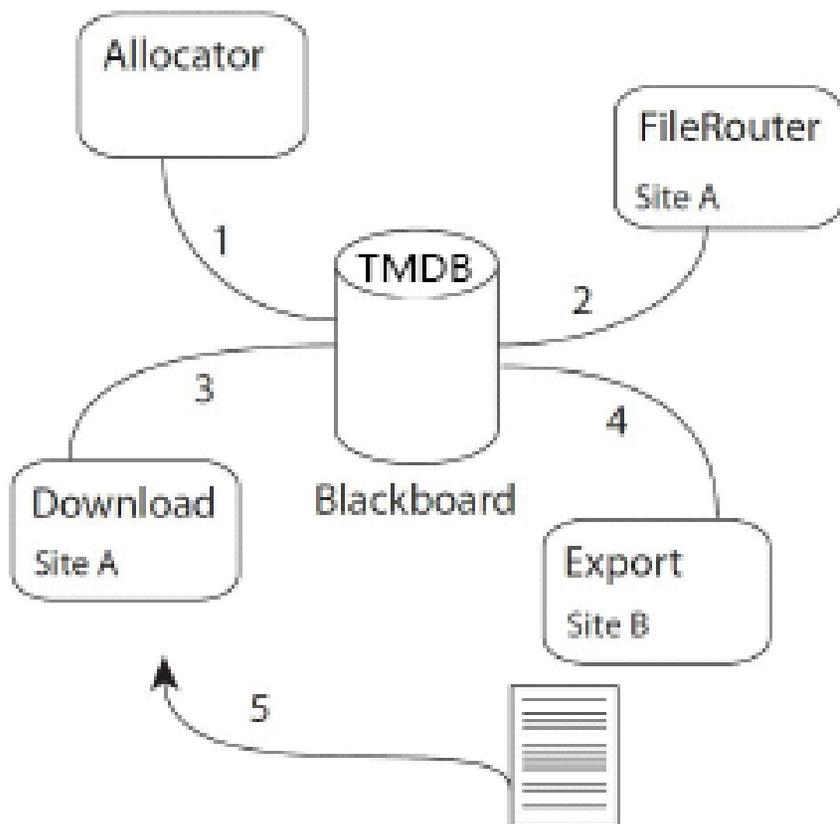


Data placement and transfer (PhEDEx)



- ▶ Data distribution among sites with PhEDEx
- ▶ Limited user-requested transfer from now on
- ▶ Focus on development and integration using new EDM data and new Data Management systems before summer
 - ✱ Transfer requests management integrated with DBS/DLS
 - ✱ Data transfer:
 - ◆ Support for trivial catalog
 - ◆ Update DLS upon file-block transfer completion
 - No META data attachment in new EDM \Rightarrow no complex publishing procedure
 - ✱ Support MC production data movement use cases
 - ◆ harvesting files
 - ✱ Support for global and local data management system





1. Allocator: allocate files to destinations
2. FileRouter: determines closest replica
3. Download: marks files „wanted“ from site B
4. Export: initiate staging and provide contact information
5. Download: transfer file



CSA06 Requirements



- ▶ Overall transfer flows (from Darin Acosta)
 - * AOD and some of the FEVT to the Tier-1s
 - ◆ 300 MB/s aggregate to tape
 - * Skims from Tier-1s to Tier-2s: AOD and some of FEVT
 - ◆ 10 MB/s – 100 MB/s depending on site ability, bursty

- ▶ Implied pre-CSA06 requirements for this summer
 - * Integration with the production system
 - ◆ Automatically triggered subscriptions, transfers, notifications
 - * Bandwidth and quality for sustained operation

- ▶ Not required
 - * Tier-1/Tier-1 transfers



PhEDEx functionality



- ▶ The required functionality is covered by the existing plan
- ▶ **PhEDEx 2.3 delivered** bulk of basic technical functionality
 - ◆ Simpler deployment, simpler and more convenient site management
 - ◆ New data management structure (new data model, DBS, DLS)
 - ◆ Trivial site-local file catalogue
 - ◆ Priorities, fair share download balancing, bandwidth caps
 - ◆ Transient transfer requests, move vs. replicate, deletion
 - ◆ More robust file routing and transfer error management
 - ◆ More extensive historical performance data
 - ◆ Improved performance of PhEDEx itself
- ▶ **Porting from RefDB/PubDB to DBS/DLS** main component
 - ◆ Easing site and global data placement
 - ◆ Rewriting transfer request management for new PhEDEx, DBS
 - ◆ Integrating with production system and DLS for completed transfers



Timeline

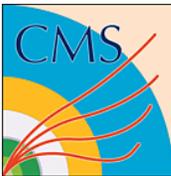


▶ May

- * Deployment of PhEDEx 2.3
- * Transition to using FTS at EGEE sites
- * Transfer request tools using DBS, DLS
- * Test integration with production system

▶ June / July

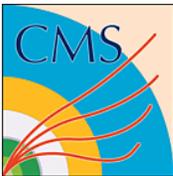
- * SC4 integration testing – pushing the system hard
- * Site validation, performance and quality targets
- * Exercising site data management tools



Prototype for new EDM/MC Production



- ▶ Currently have a DBS system capable of being used for MC production with the new EDM (and configuration of CRAB)
 - ✱ Database on CERN Oracle RAC, Apache server using CGI/Perl
 - ✱ Used to catalog produced data (datasets, files, blocks)
 - ✱ Simple support for users dataset discovery in DBS
- ▶ DLS just handles file-blocks. Foresee the use of local scope DLS within MC production
- ▶ Setup for Local data access is in progress
 - ✱ Baseline is to use a "Trivial catalogue" : use a rule (i.e. append a prefix) to obtain the PFN from a given LFN
 - ✱ Site local discovery mechanism being implemented in the Framework
- ▶ MC production system integrated closely with DM components.



PhEDEx Experience



- ▶ Technically PhEDEx itself is largely ready
 - * Lots of experience both at sites and centrally
 - * Capacity sufficient for LHC production transfers already
 - ◆ Can sustain 200k+ transfers/hour if we have <6M active transfers
 - * Have a very clear idea on how to implement remaining features
- ▶ The interesting question is integration
 - * The full transfer system is *much* more than just PhEDEx
 - * Has been lots of very hard work to get transfers working in practice
 - * Every indication this will remain lots of hard work
 - * Top issues
 - ◆ Transition to WLCG data transfer support
 - ◆ Achieving quick and accurate problem resolution
 - ◆ Transition to FTS at EGEE sites
 - ◆ Keeping SRMs interoperable



Summary



- ▶ The Data Management components of the model described in Computing TDR exist
- ▶ They are being developed and integrated together into a coherent system that will allow managing data production, processing, analysis and transfer in a distributed environment for CSA06
- ▶ Additional manpower is starting to work out deeper analysis needs. This may not be used for CSA06 but will be incorporated into the system for the 2007 version.



Finish