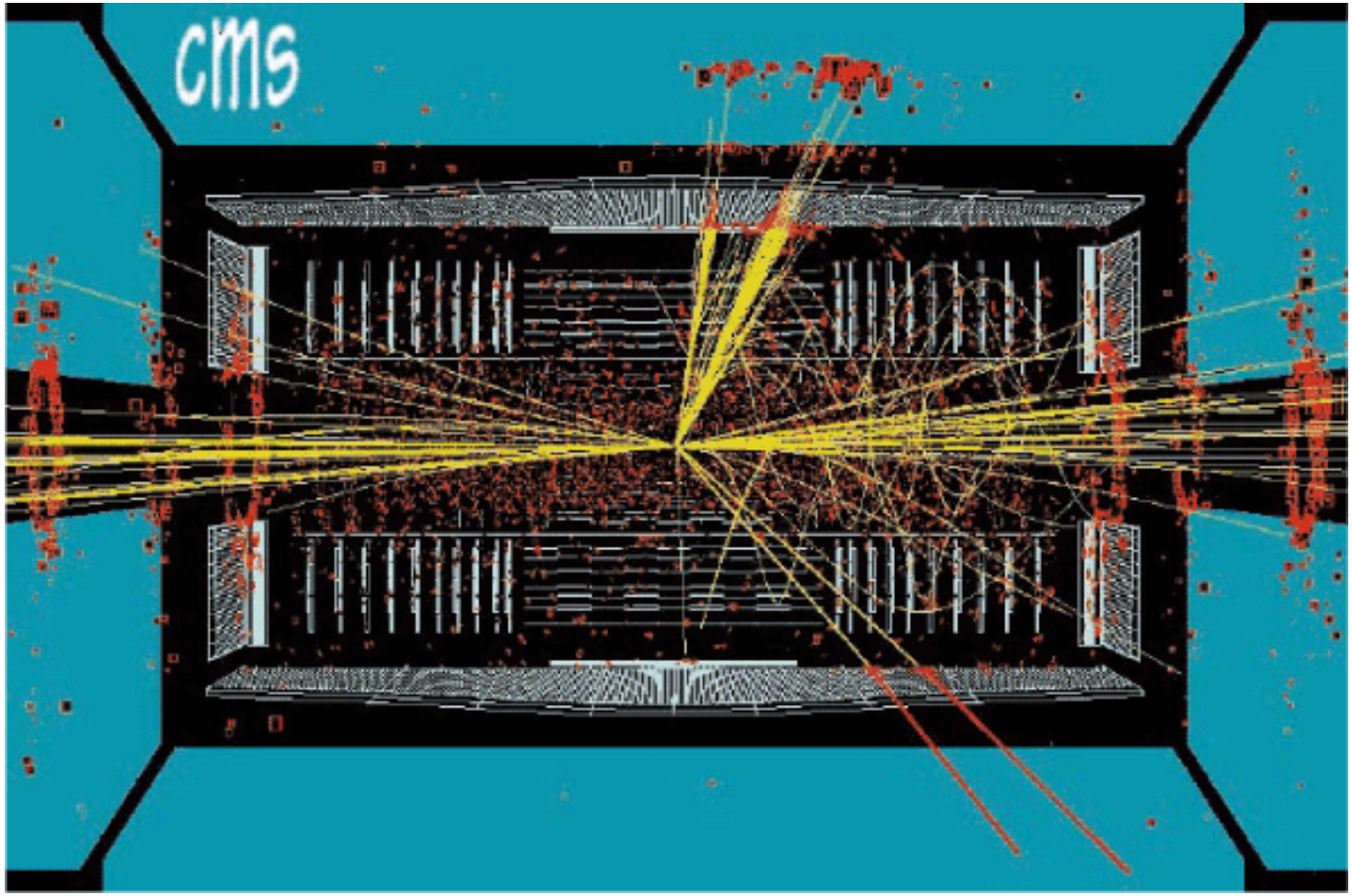
CMS Data Management and CMS Monitoring (emphasis on T2 perspective)

- *CMS data organization*
- *Data names, numbers, flow*
- *Data handling issues at a site*
- *CMS Computing operation monitoring*

Data building block: The Event

- Two protons collide at the center of CMS detector, and millions of electronic channels collect data



- Yellow lines (tracks) e.g., added later on during data processing

The Event

- At the core of the experiment data is "The Event"
- A numerical representation of one proton-proton collision at LHC
  - As seen through CMS detector
  - Sort of a digital picture of the collision
- There are many events ➔ Datasets
- There are many data in one event
  - Data inside each event are described as objects
  - Events data are written in root format (outside CERN at least)
- As data are processed, event content changes ➔ Data Tiers
  - Events with similar content (same objects) are said to belong to same data Tier (more in next slides)

Data processing steps

- MonteCarlo (MC) Production (Simulation)
  - Create data that look like coming from detector
- DAQ/HLT
  - Data AcQuisition and High Level Trigger = online event selection
  - Writes data coming from detector (not all events are recorded)
- Processing and reprocessing (Reconstruction, Production)
  - Add/change data
  - Because of computations made on the data
  - Because of added information (calibration, conditions, geometry…) from DataBases
- Skim
  - Select a fraction of the data
- Analysis
  - All of that, but often: extract one plot, or one number

- Each processing step output is in a well defined format: data Tier
  - GEN, HIT, DIGI, RECO, AOD

Data Tiers and Datasets

- Data Tiers track the story of one event through the CMS computing chain: horizontal organization
- A Data Tier is a collection of events with same objects
  - Detail: each data Tier can be thought as comprising the full CMS data set in that particular representation (list of objects)
  - Hence we talk about
  - The CMS RAW data
  - The CMS RECO data, etc.

- Vertical data organization: All data tiers for events of a particular kind, coming from a specific origin
  - One specific Monte Carlo simulation set of input parameters
  - One particular selection criteria in the CMS Trigger
  - One particular selection in the processing chain, followed by other processing and skim steps
- These are called : datasets

Examples (names you may sort of see)

	Simulation Z-to-ee	HLT 4-leptons	HLT Jet-1200	Simulation JetSim1200
MC Production at Tier2	GEN,HITS, DIGI			GEN,HITS, DIGI
Copy to Tier1	GEN,HITS, DIGI			GEN,HITS, DIGI
DAQ/HLT at CMS (P5)		RAW	RAW	
Prompt Reco at Tier0		RECO, AOD	RECO, AOD	
Reprocessing at Tier1	RECO, AOD	RECO, AOD	RECO, AOD	RECO, AOD
Skim at Tier1	RECO, AOD	RECO, AOD	RECO, AOD	RECO, AOD
Copy to Tier2	RECO, AOD	RAW,RECO, AOD	RAW,RECO, AOD	RECO, AOD

CMS data management

- There are data (events) (KB~MB: size driven by physics)
  - 1PB/year = 10^12 KB
- Event data are in files (GB: size driven by DM convenience)
  - 10^6 files/year *CMS catalogs lists files, not events*
- Files are grouped in Fileblocks (TB: size driven by DM convenience)
  - 10^3 Fileblocks/year *CMS data management moves Fileblocks*

- Fileblocks are grouped in Datasets (TB: size driven by physics)
  - Datasets are large (100TB) or small (0.1TB)
  - Datasets are not too many: 10^3 Datasets (after years of running)
- CMS catalog (DBS) lists all Datasets and their contents, relationships, provenance and associated metadata
- CMS Data Location Service (DLS) list location of all File Blocks
- RAW, DIGI, HITS, RECO, AOD, NTUPLE…. All are handled in this way

In practice: files on servers

- A Tier2 will manage various kinds of data
  - Appear as a set of file blocks from specific datasets
- MC production output
  - Intermediate small files at job output (unmerged)
  - Merged files, O(GB) for transfer to Tier1
- Data for analysis users
  - Skims from larger datasets at Tier1 (all kind of tiers)
  - Data for/from local users from processing of those

- Some data will have backup (on tape) at Tier1/0
- Some not (MC output before transfer, user's data)

- A Tier2 may (or may not) want to use different resources (for space, performance, reliability) for different data
  - How does a Tier2 know which file is of which kind ?

Logical and Physical names

- Each CMS file in a particular data set/data tier is a well defined set of objects, a set of bytes
- This has a name that uniquely defines it
  - But has many physical instances, since it can move around
- So have a Logical File Name: LFN
  - Unique name for a file in the CMS data set
- And we have a Physical File Name; PFN
  - The actual name of a file in a particular site in the format that can be used by an application to act on it
  - One LFN may correspond to many PFN's
  - dcap:/… rfio:/… srm:/.
  - More on Sunday
- Basic concept
  - LFN and PFN space is organized
  - So a site can assign data to resources based on PFN

File Name space organization

- CMS will organize File Name Space (i.e. directory structure in LFN *and* in PFN) so that storage management is easy
- For this to work a contract with the site is needed
  - It is expected,that each site offers storage to CMS via a single Storage Element with a unique uniform name space
  - CMS can cope with multiple Storage Elements as long as each offers the same uniform name space
- The Storage Element must have an SRM server
- The Storage Element must offer Posix-like access, i.e. some protocol to open the file from the analysis application
  - Dcap, rfio, rootd, xrootd….
  - In addition to srm/gsiftp
- dCache, Castor, DPM are all OK

File Names organization

- CMS organize all its data in a unique, hierarchical, LFN space.
- All data live in subdirectories of this common name space.
- This name space organization is the one visible in DBS
- CMS guarantees that the number of files in each directory is limited
- Therefore CMS LFN name space can be trivially mapped to physical name space of any specific Storage Element
- Current layout of the trivial catalog name space is available at https://twiki.cern.ch/twiki/bin/view/CMS/ProdLFNConvention
- CMS will use the leftmost directory/ies of the name space to separate sets of data that may need to be handled differently at sites, as far as physical location (tape, disk, other). Examples
  - /store/unmerged/.... used for temporary outputs that need to be merged in larger files before moving/storing, well suited for disk-only storage
  - /store/production/... used for final production output, to be saved on tape at Tier1's
- Each site will then map those branches of the LFN tree to a specific SE (or piece of) according to desired policy and local technicalities

Data flows: output

- MC Simulation data are created at Tier2 and stay there only for a brief time. Final destination is one Tier1 picked
  - because of its capacity to offer custodial storage for them
  - not because of geography
  - required bandwidth is small, no problem to reach any T1

- Data rate out of a Tier2 dependent on amount of generated data per unit of time (i.e. CPU), usually a few MB/sec (e.g. ~4 for US T2's)

- Imagine this as a steady trickle of data out from the site

- Which data is produced will be under control of central CMS operations, site provides resources but does not control which job is sent

Data flows: input

- What data to import is largely under site control
- See discussion about Tier2 role in previous talk
- How much to import is a combination of local users needs and disk capacity
- In general transfers will be initiated by a person responsible for data management at the site, whom asks PhEDEx to replicate locally a given (fraction of a) dataset, i.e. a list of Fileblocks

- New data are not required every day
- But when they are required for local analysis, tomorrow is not too soon
  - Input traffic has peaks and comes in bursts
  - Will try to saturate network for a while, then stop
  - Few MB/sec average, but up to 100MB/sec peak
- Data may have to come from different T1's
  - AOD are exception, are at all T1's, including best connected one

Data flows summary

- PhEDEx will manage most of the data flow
  - Tutorial on Sunday
- PhEDEx relies on FTS and SRM
  - Tutorial on Saturday

- Input bandwidth much higher then output
  - Beware competition with data access from running jobs
  - May have site specific issues that requires site specific solution depending on actual hardware configuration
- CMS can tune operation to have as much transfers as possible to have the "best connected Tier1" at the other end, but this will not satisfy all needs

- You will need to move data to/from many Tier1
- We start to address/understand details of flows
  - http://lcg.web.cern.ch/LCG/documents/Megatable161106.xls

Summary on Data Management

- Structured data
- Well defined name space for files
- Sites can map CMS data to various hardware, pool policies

- Uniform name space must be offered locally

- CMS's onw products focus on the TB scale (fileblocks, datasets)

- Grid solutions underneath deal with singe files at GB scale
  - FTS, SRM, gsiFtp

- CMS will work with all standard SRM servers currently deployed
  - dCache, Castor, DPM

Monitor

- Monitor the data locations
  - Which data a site hosts
- Monitor the data transfers, data flows
  - Which data should a site receive/send ?
  - Are data moving ? How well ?
- Monitor the running applications
  - Which jobs are running at a site ?
  - How long they wait, run ?
  - Are they successful ?
  - Which data do they access ?
  - How much data do they read ?

Monitor the data location

DBS data discovery page

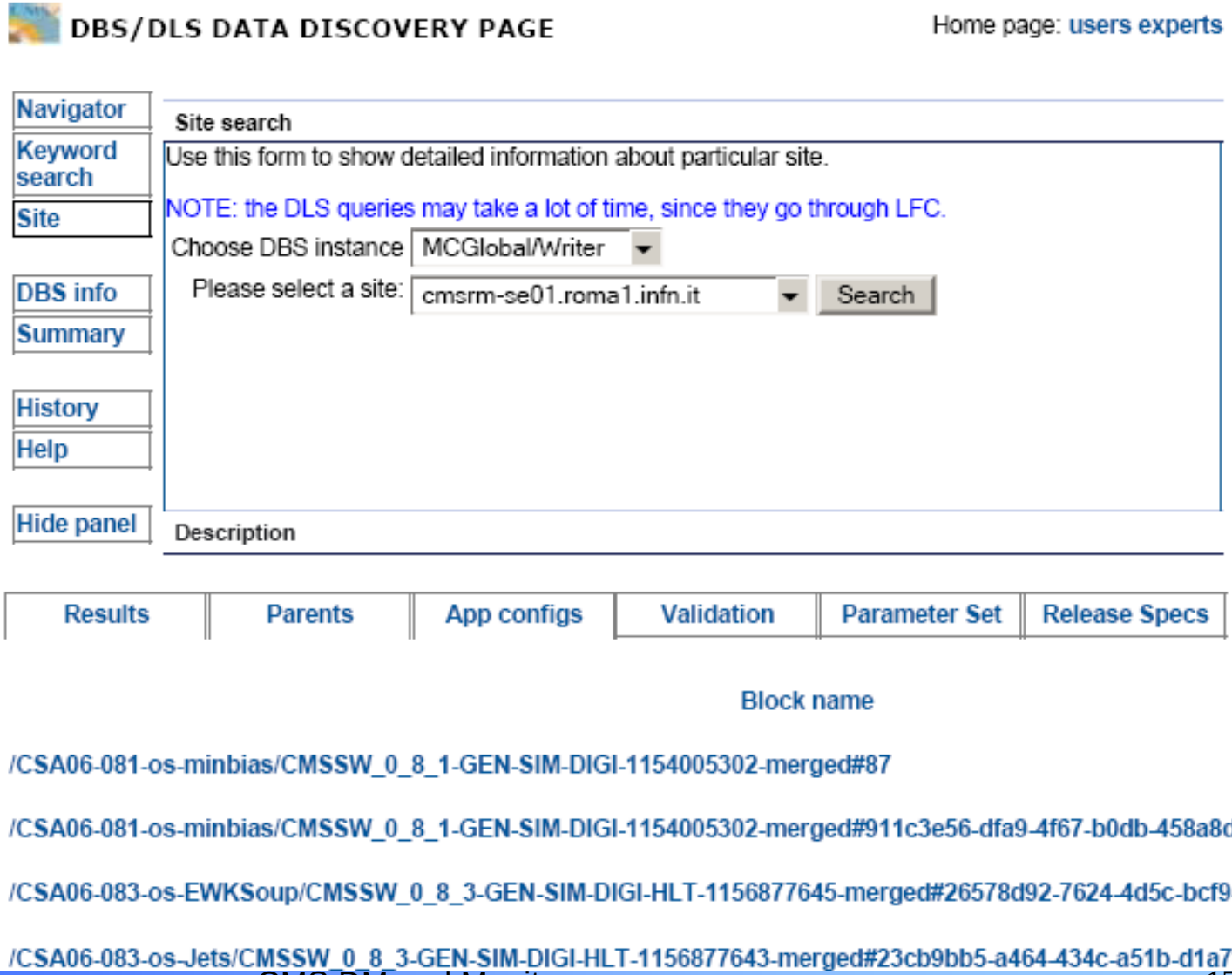
http://cmsdbs.cern.ch/discovery

Integrated DBS/DLS view

Click on file block name to get file list

More Info on Sunday

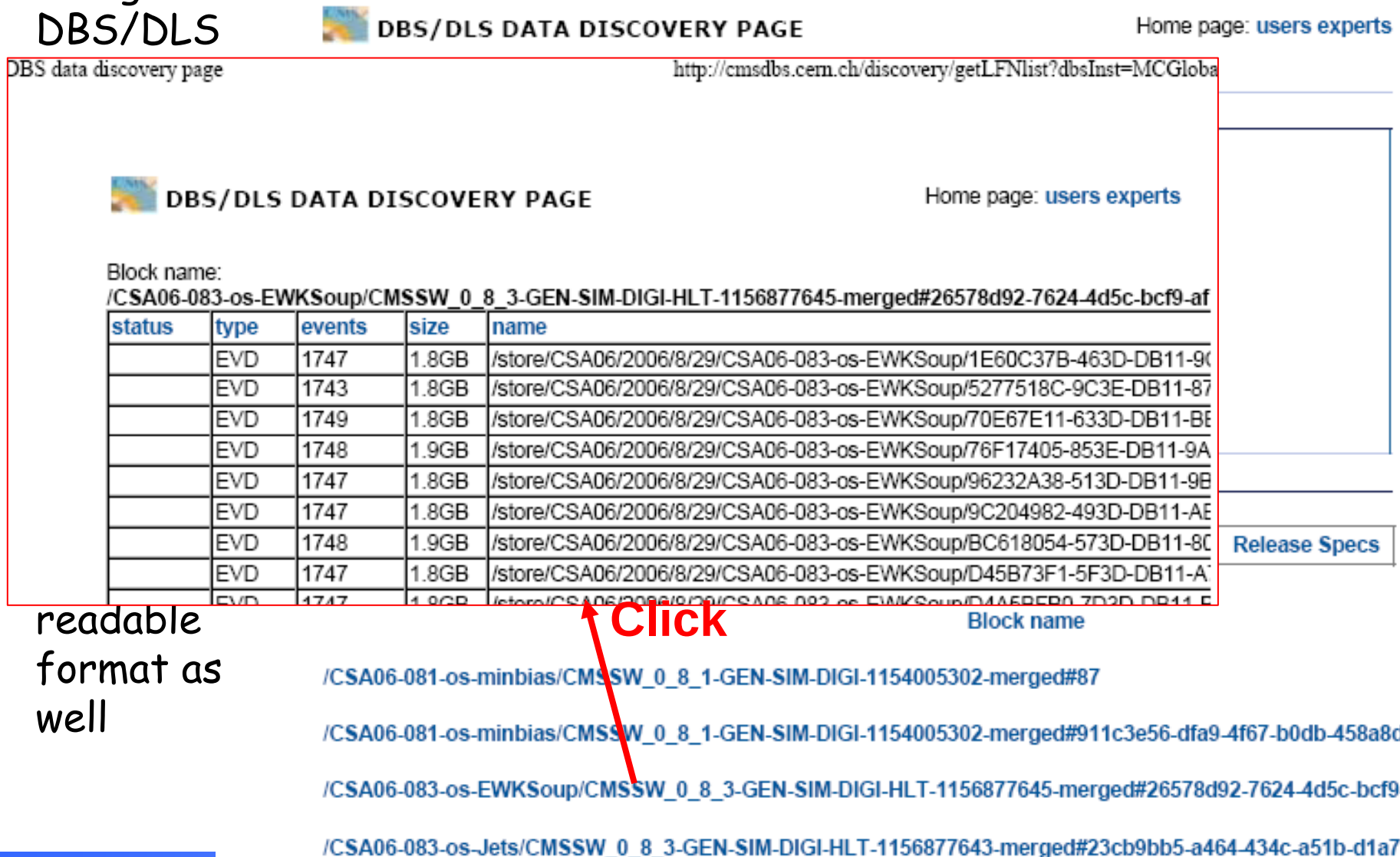
Computer readable format as well



Monitor the data location

DBS data discovery page http://cmsdbs.cern.ch/discovery

Integrated DBS/DLS readable format as well

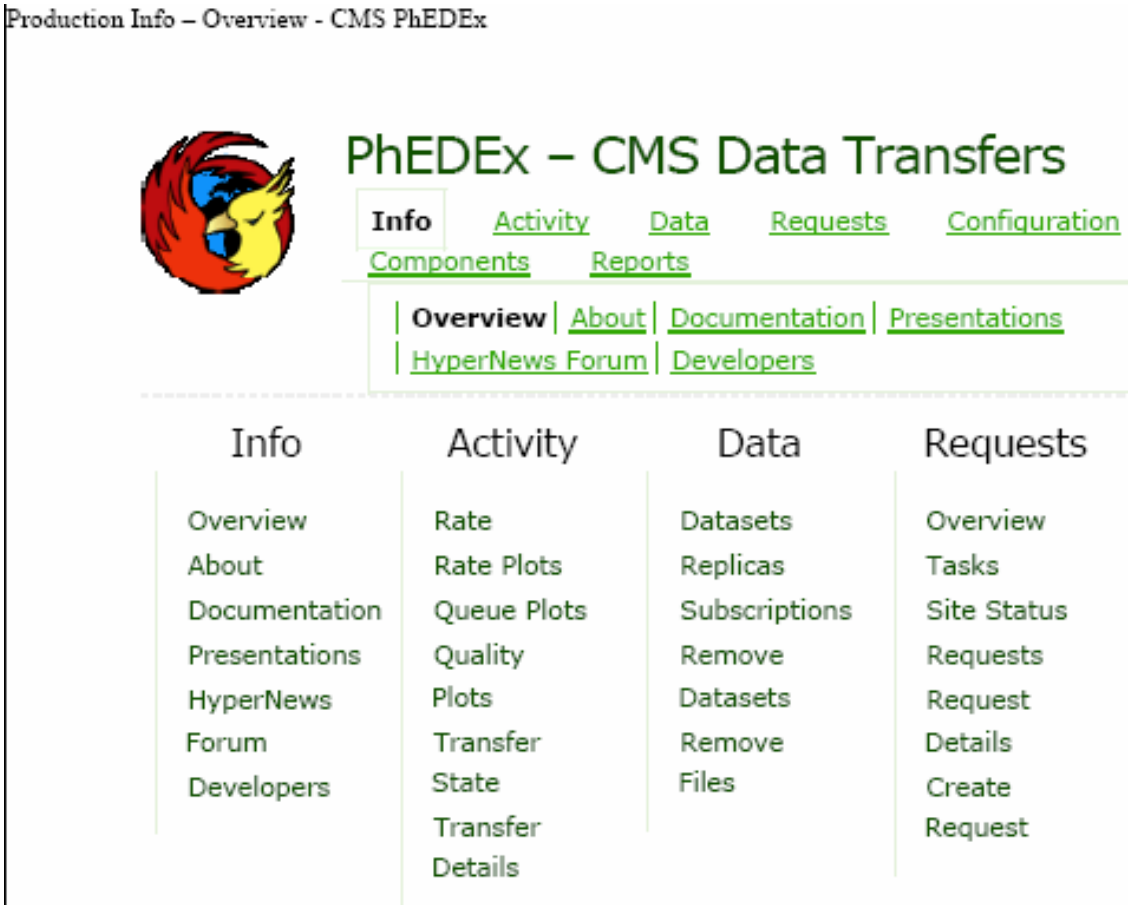


Monitor the data transfers

- Layered set of tools
  - PhEDEx : moves CMS datasets
    - rich set of web pages and graphs
    - more features and new web site in development
  - FTS : moves files, some retries, hides SRM details, implements access policies on "site A to site B" traffic
    - no general monitoring tool yet
    - see FTS tutorial tomorrow
  - SRM/gsiftp : low level single file transfer tool
    - http://Gridview.cern.ch monitoring based on gridFtp logs
    - does not cover all sites in practice at present

Monitoring and control in PhEDEx

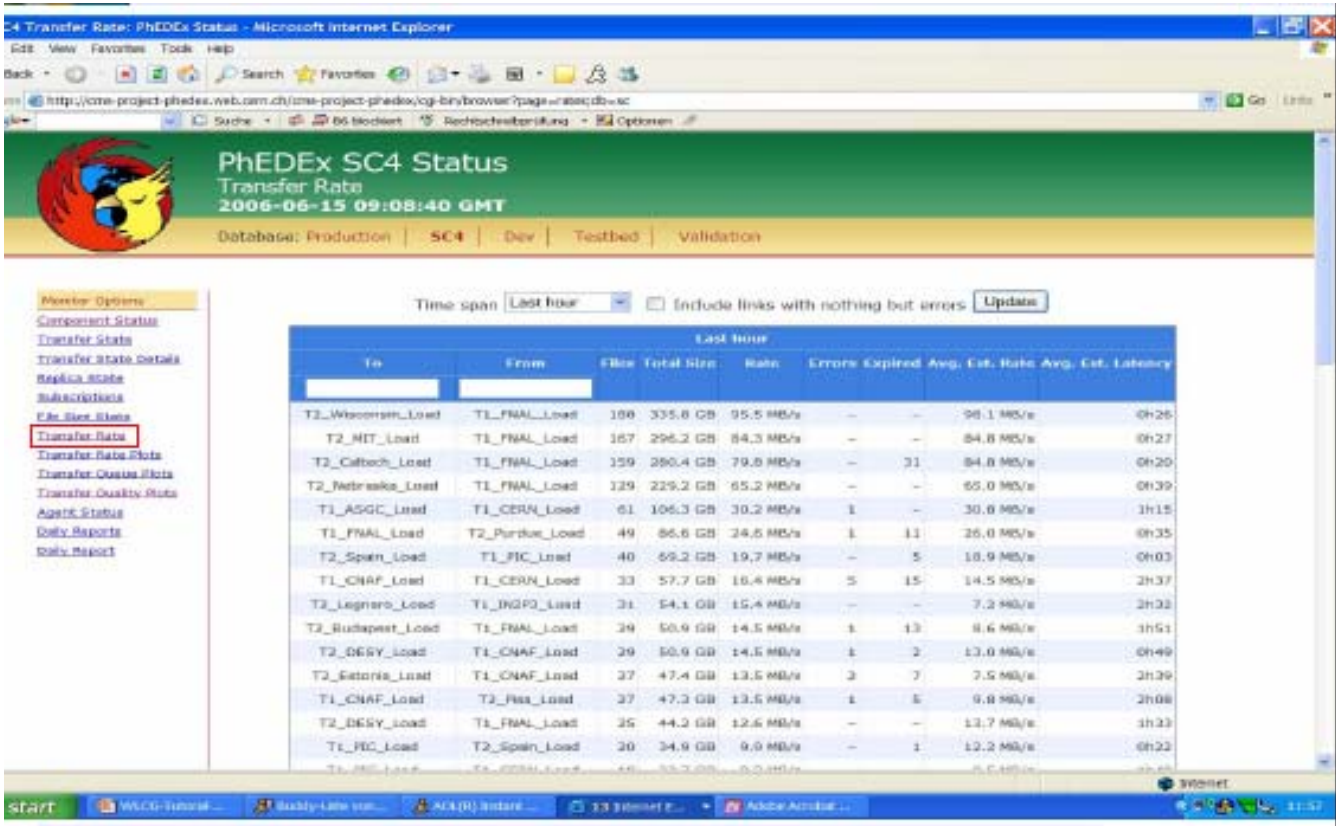
- http://cmsdoc.cern.ch/cms/aprom/phedex/prod/



- See also http://agenda.cern.ch/askArchive.php?base=agenda&categ=a062664&id=a062664s0t12/transparencies

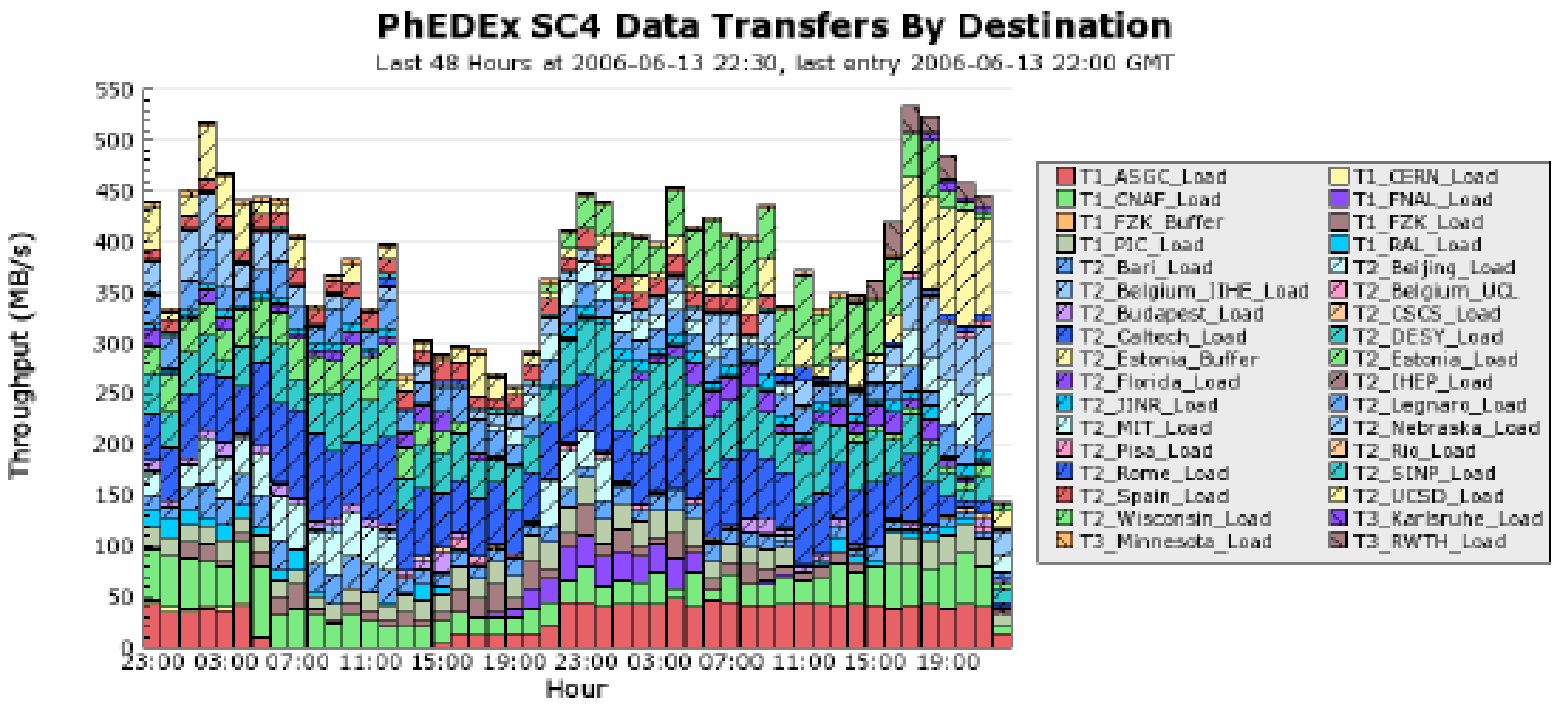
PhEDEx example (more on Sunday)

- Transfer rates



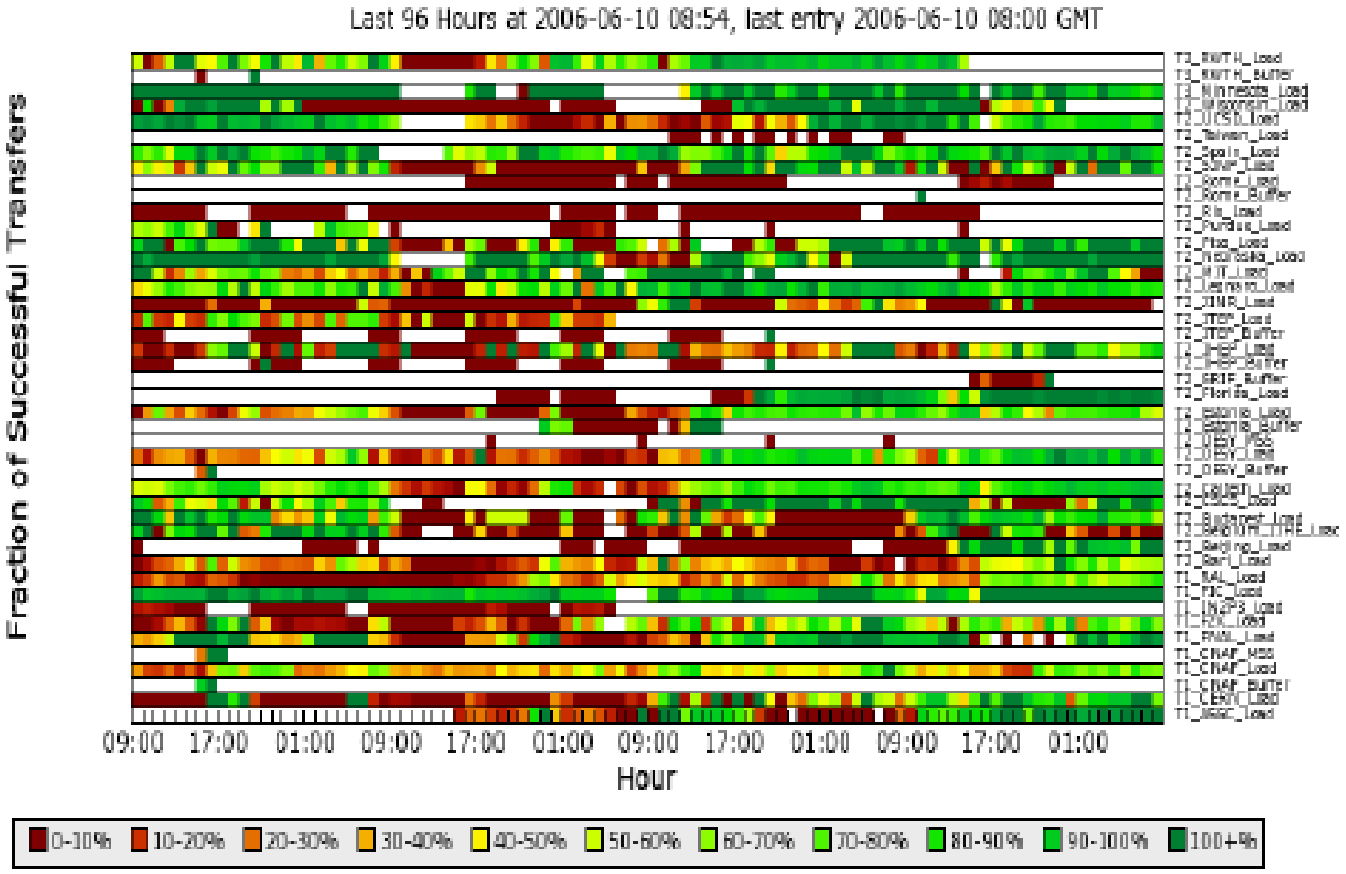
More PhEDEx example

- Can select source, destination, link, time interval..



More PhEDEx example

- Transfer quality plots (fraction of successful trasnfers)
- Can select source, destination, link, time interval..



Monitoring CMS jobs

- There is site monitoring and grid monitoring
- But now we care about application monitoring
  - Correlate/aggregate information based on application specific information that site or grid does not know
    - which data does the job access
    - is this real production or test ?
    - user's analysis or organized physics groups activity
    - was the application (CMS SW) successful ? Why ?

- Strategy
  - Instrument job to report about itself at submission, execution, completion times
  - Via hooks in job management tools (Crab, ProductionAgent)
  - Via hooks in job wrapper
  - Collect all data in central, Oracle based DB
  - Have interactive "dig-in" browser and static plots

CMS Dashboard

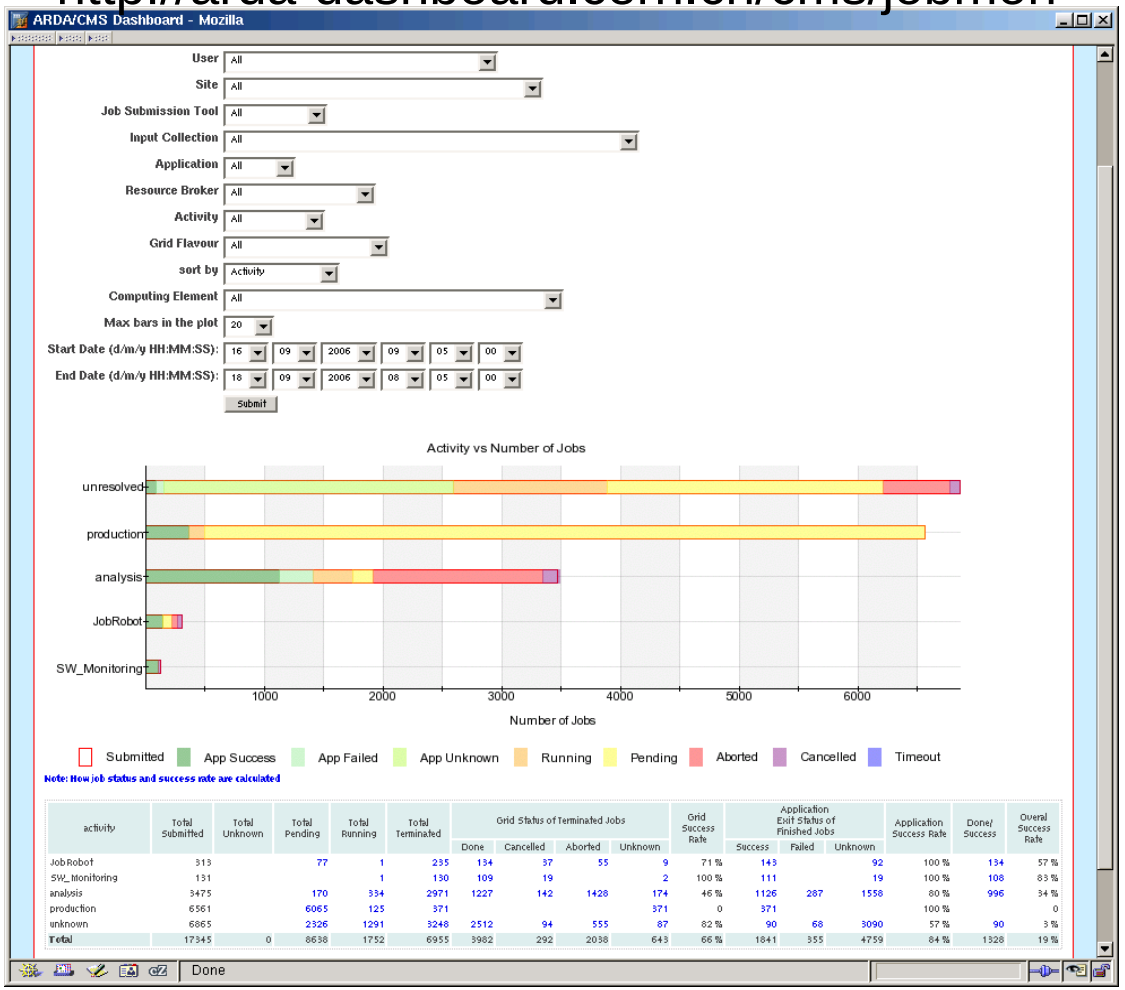
- http://arda-dashboard.cern.ch/cms/
- Explore around, there are quite a few useful things
- Next slides show examples from "Job Monitoring" links
- See also Michael Ernst tutorial at June's T2 Workshop http://agenda.cern.ch/askArchive.php?base=agenda&categ=a062664&id=a062664s0t12/transparencies

CMS Job Dashboard

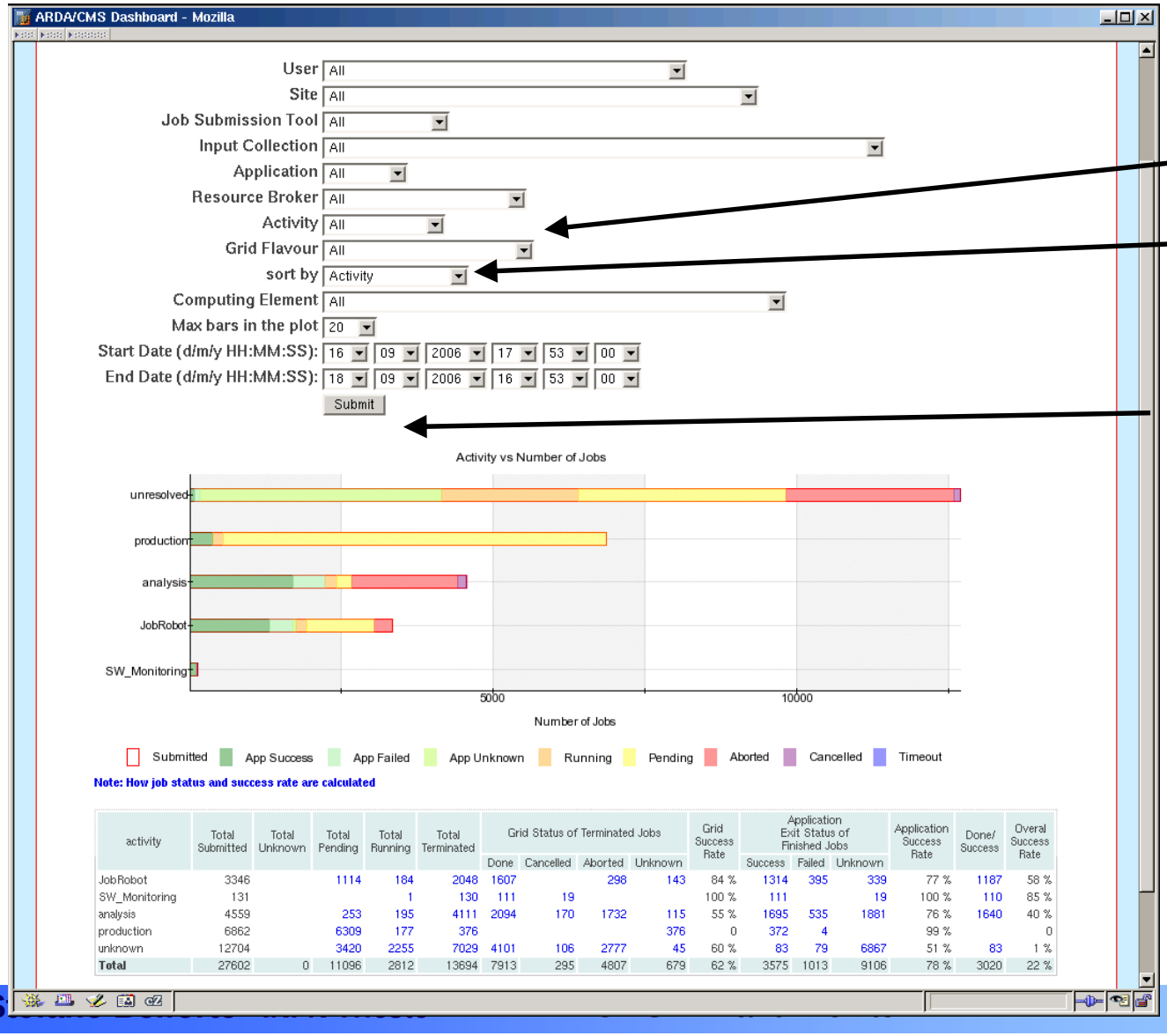
- Tutorial at last CMS week: http://indico.cern.ch/materialDisplay.py?contribId=62&sessionId=4&materialId=slides&confId=5878

http://arda-dashboard.cern.ch/cms/jobmon

- Running Oracle back-end and PHP web UI.
- Reading data from various sources
- Gaining valuable experience running such a service
  - Performance
- Currently we are working towards the next versions of the Dashboard with extended scope
  - Tier-0 (CMS)
  - Grid reliability
  - Service monitoring (SAM, 3D)



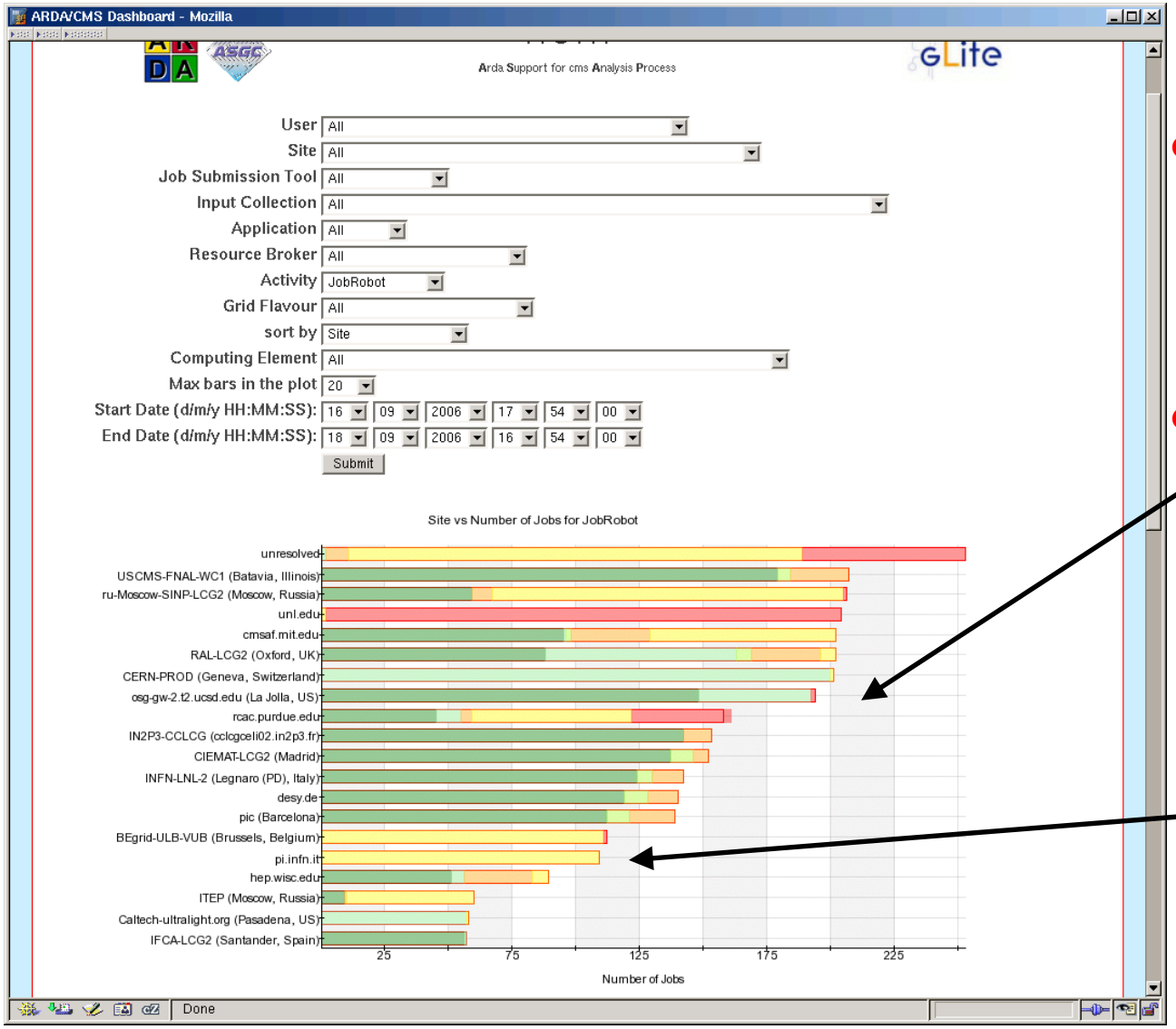
Using the Dashboard for Job Monitoring



activity	Total Submitted	Total Unknown	Total Pending	Total Running	Total Terminated	Grid Status of Terminated Jobs				Grid Success Rate	Application Exit Status of Finished Jobs			Application Success Rate	Done/ Success	Overal Success Rate
						Done	Cancelled	Aborted	Unknown		Success	Failed	Unknown			
JobRobot	3346		1114	184	2048	1607		298	143	84 %	1314	395	339	77 %	1187	58 %
SW_Monitoring	131			1	130	111	19			100 %	111		19	100 %	110	85 %
analysis	4559		253	195	4111	2094	170	1732	115	55 %	1695	535	1881	76 %	1640	40 %
production	6862		6309	177	376				376	0	372	4		99 %		0
unknown	12704		3420	2255	7029	4101	106	2777	45	60 %	83	79	6867	51 %	83	1 %
Total	27602	0	11096	2812	13694	7913	295	4807	679	62 %	3575	1013	9106	78 %	3020	22 %

- Select the JobRobot activity
- Set sorting by site
- Submit query

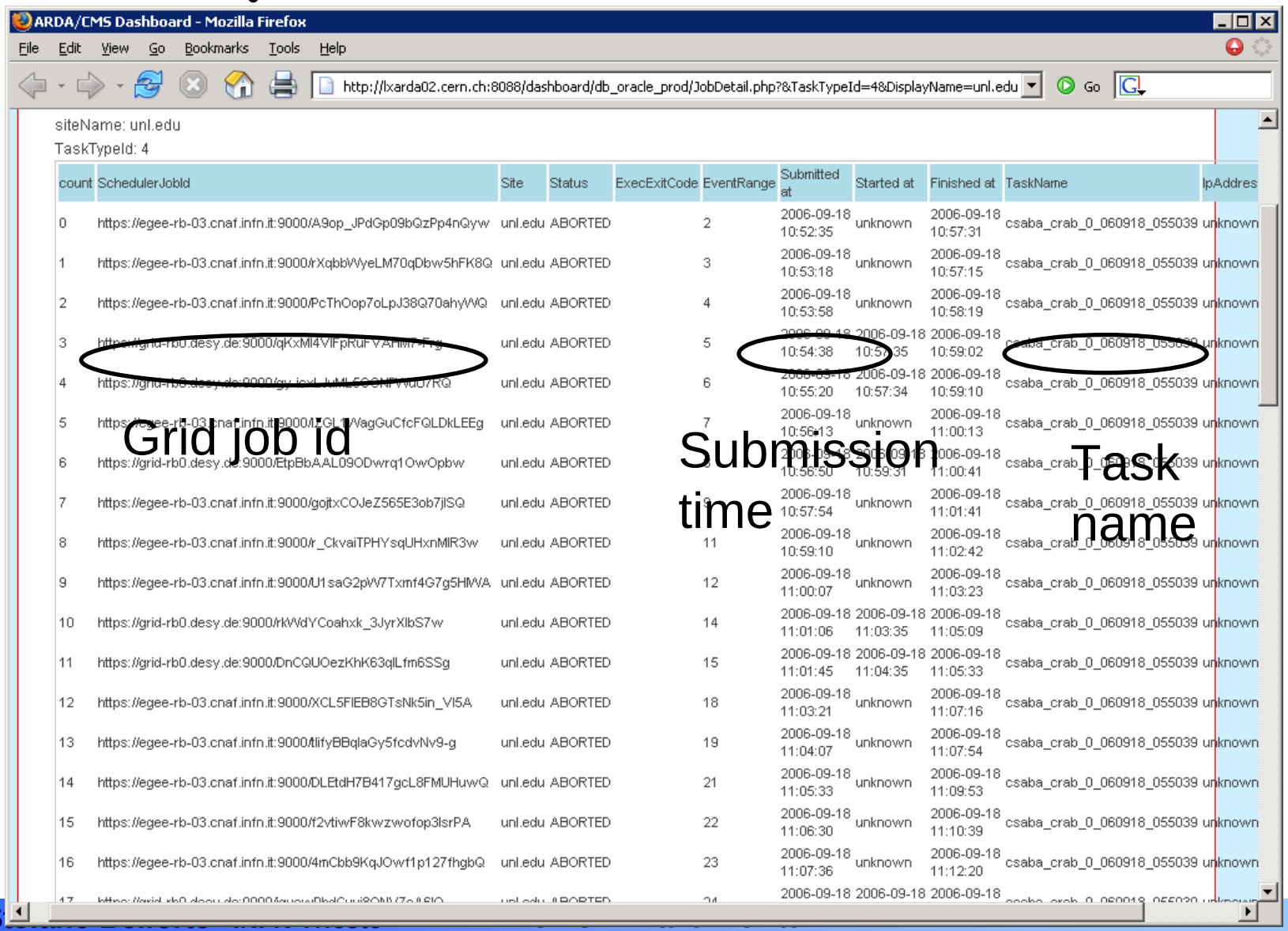
Using the Dashboard for Job Monitoring



- Bar with large amount of red colour implies to a temporary problem with many aborted jobs.
- Green colour indicates successful job execution

Using the Dashboard for Job Monitoring

- Can tell if jobs failed before or after execution start, and when

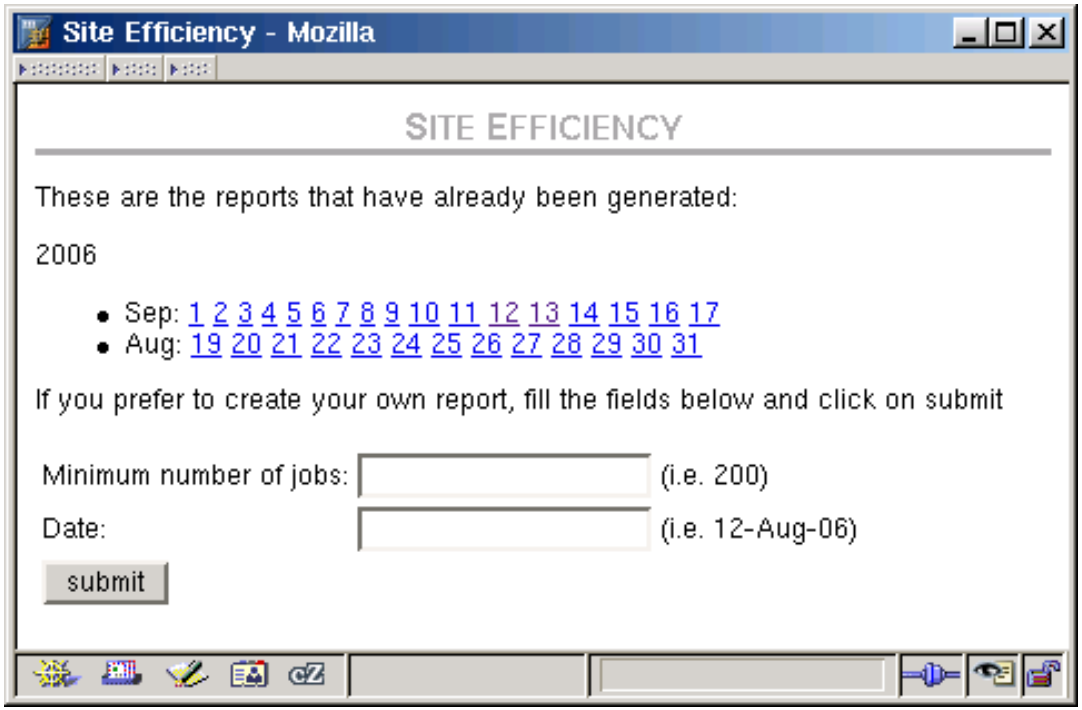


count	SchedulerJobId	Site	Status	ExecExitCode	EventRange	Submitted at	Started at	Finished at	TaskName	IpAddres
0	https://egee-rb-03.cnaf.infn.it:9000/A9op_JPdGp09bQzPp4nQyw	unl.edu	ABORTED		2	2006-09-18 10:52:35	unknown	2006-09-18 10:57:31	csaba_crab_0_060918_055039	unknown
1	https://egee-rb-03.cnaf.infn.it:9000/rXqbbWyeLM70qDbw5hFK8Q	unl.edu	ABORTED		3	2006-09-18 10:53:18	unknown	2006-09-18 10:57:15	csaba_crab_0_060918_055039	unknown
2	https://egee-rb-03.cnaf.infn.it:9000/PcThOop7oLpJ38Q70ahyWQ	unl.edu	ABORTED		4	2006-09-18 10:53:58	unknown	2006-09-18 10:58:19	csaba_crab_0_060918_055039	unknown
3	https://grid-rb0.desy.de:9000/qKxMl4VIFpRuFVAHMF4Tg	unl.edu	ABORTED		5	2006-09-18 10:54:38	2006-09-18 10:57:35	2006-09-18 10:59:02	csaba_crab_0_060918_055039	unknown
4	https://grid-rb0.desy.de:9000/gy-jcxL-JuMLE6GNFVdD7RQ	unl.edu	ABORTED		6	2006-09-18 10:55:20	2006-09-18 10:57:34	2006-09-18 10:59:10	csaba_crab_0_060918_055039	unknown
5	https://egee-rb-03.cnaf.infn.it:9000/MGL1WagGuCfcFQLDkLEEg	unl.edu	ABORTED		7	2006-09-18 10:56:13	unknown	2006-09-18 11:00:13	csaba_crab_0_060918_055039	unknown
6	https://grid-rb0.desy.de:9000/EtpBbAAL09ODwrq1OwOpbw	unl.edu	ABORTED		8	2006-09-18 10:56:50	2006-09-18 10:59:31	2006-09-18 11:00:41	csaba_crab_0_060918_055039	unknown
7	https://egee-rb-03.cnaf.infn.it:9000/gojtxCOJeZ565E3ob7jlSQ	unl.edu	ABORTED		9	2006-09-18 10:57:54	unknown	2006-09-18 11:01:41	csaba_crab_0_060918_055039	unknown
8	https://egee-rb-03.cnaf.infn.it:9000/r_CkvaiTPHYsqUHxnMlR3w	unl.edu	ABORTED		11	2006-09-18 10:59:10	unknown	2006-09-18 11:02:42	csaba_crab_0_060918_055039	unknown
9	https://egee-rb-03.cnaf.infn.it:9000/U1saG2pW7Txmf4G7g5HMWA	unl.edu	ABORTED		12	2006-09-18 11:00:07	unknown	2006-09-18 11:03:23	csaba_crab_0_060918_055039	unknown
10	https://grid-rb0.desy.de:9000/rkWdYCoahxk_3JyrXlbS7w	unl.edu	ABORTED		14	2006-09-18 11:01:06	2006-09-18 11:03:35	2006-09-18 11:05:09	csaba_crab_0_060918_055039	unknown
11	https://grid-rb0.desy.de:9000/DnCQUOezKhK63qlLfm6SSg	unl.edu	ABORTED		15	2006-09-18 11:01:45	2006-09-18 11:04:35	2006-09-18 11:05:33	csaba_crab_0_060918_055039	unknown
12	https://egee-rb-03.cnaf.infn.it:9000/XCL5FIEB8GTsNk5in_VI5A	unl.edu	ABORTED		18	2006-09-18 11:03:21	unknown	2006-09-18 11:07:16	csaba_crab_0_060918_055039	unknown
13	https://egee-rb-03.cnaf.infn.it:9000/tlifyBBqlaGy5fcdvNv9-g	unl.edu	ABORTED		19	2006-09-18 11:04:07	unknown	2006-09-18 11:07:54	csaba_crab_0_060918_055039	unknown
14	https://egee-rb-03.cnaf.infn.it:9000/DLEtdH7B417gcL8FMUHuwQ	unl.edu	ABORTED		21	2006-09-18 11:05:33	unknown	2006-09-18 11:09:53	csaba_crab_0_060918_055039	unknown
15	https://egee-rb-03.cnaf.infn.it:9000/f2vtiwF8kwzwofop3lsrPA	unl.edu	ABORTED		22	2006-09-18 11:06:30	unknown	2006-09-18 11:10:39	csaba_crab_0_060918_055039	unknown
16	https://egee-rb-03.cnaf.infn.it:9000/4mCbb9KqJOwf1p127fhgbQ	unl.edu	ABORTED		23	2006-09-18 11:07:36	unknown	2006-09-18 11:12:20	csaba_crab_0_060918_055039	unknown
17	https://grid-rb0.desy.de:9000/[illegible]	unl.edu	ABORTED		24	2006-09-18	2006-09-18	2006-09-18	csaba_crab_0_060918_055039	unknown

Grid Reliability – Site Efficiency

- The Grid Reliability project was triggered by the experience of the Dashboard project
- The Goal : present real reasons of job failures based on analyzing the R-GMA log files

http://arda-dashboard.cern.ch/cms/jobmon-gr



Grid Reliability – Site Efficiency

Site Efficiency - Mozilla

SITE EFFICIENCY

Click on any Site, and you will have a breakdown of the jobs according to the CEs

If you click on the CEs, you will have a breakdown of the jobs of that CE according to their workflow

Warning!This table does not represent the number of jobs, but the number of job attempts. For example, if a user submits one job, the job lands twice on site A where it fails, and then it lands on site B where it is successful, that job would produce three entries in the following table: two failures for site A, and one success for site B.

Displaying the sites with more than 150 jobs

Displaying the values of the date: 13-Sep-06
If you want a similar report for any other day, click here

SiteName (click on any site)	Successful jobs	Failed jobs	Efficiency
hep.kbfi.ee	110	496	18.15%
unl.edu	43	185	18.86%
cmsaf.mit.edu	128	138	48.12%
CIEMAT-LCG2	107	52	67.30%
RAL-LCG2	211	91	69.87%
INFN-LNL-2	177	74	70.52%
desy.de	202	79	71.89%
FZK-LCG2	401	151	72.64%
pi.infn.it	150	42	78.12%
Taiwan-LCG2	187	51	78.57%
DESY-HH	146	20	87.95%
RWTH-Aachen	143	12	92.26%
USCMS-FNAL-WC1	1995	121	94.28%
hep.wisc.edu	252	14	94.74%
INFN-T1	145	6	96.03%
cern.ch	239	5	97.95%
CERN-PROD	1244	6	99.52%
osg-gw-2.t2.ucsd.edu	213	1	99.53%

Done

Site Efficiency - Mozilla

Displaying the sites with more than 150 jobs

Displaying the values of the date: 13-Sep-06
If you want a similar report for any other day, click here

SiteName (click on any site)	Successful jobs	Failed jobs	Efficiency
hep.kbfi.ee	110	496	18.15%
unl.edu	43	185	18.86%
red.unl.edu:2119/jobmanager-pbs-cms	43	185	18.86%

JobIds	# jobs	Successful?	Error message
See all the jobids...	57	Ignored	Job RetryCount _N_ hit
See all the jobids...	15	Ignored	Job successfully submitted to Globus
See all the jobids...	12	Ignored	Job successfully submitted to Globus
See all the jobids...	58	No	Aborted by user
See all the jobids...	49	No	Cannot download _file_ from _file_
See all the jobids...	19	No	Aborted by user
See all the jobids...	16	No	Aborted by user
See all the jobids...	15	No	Aborted by user
See all the jobids...	9	No	Job RetryCount _N_ hit
See all the jobids...	9	No	Cannot download _file_ from _file_
See all the jobids...	8	No	Got a job held event reason: Globus error 12: the connection to the server failed check host and port
See all the jobids...	1	No	Cannot download _file_ from _file_
See all the jobids...	1	No	10 data transfer to the server failed
See all the jobids...	32	Yes	Job terminated successfully
See all the jobids...	7	Yes	user retrieved output sandbox
See all the jobids...	3	Yes	user retrieved output sandbox
See all the jobids...	1	Yes	user retrieved output sandbox

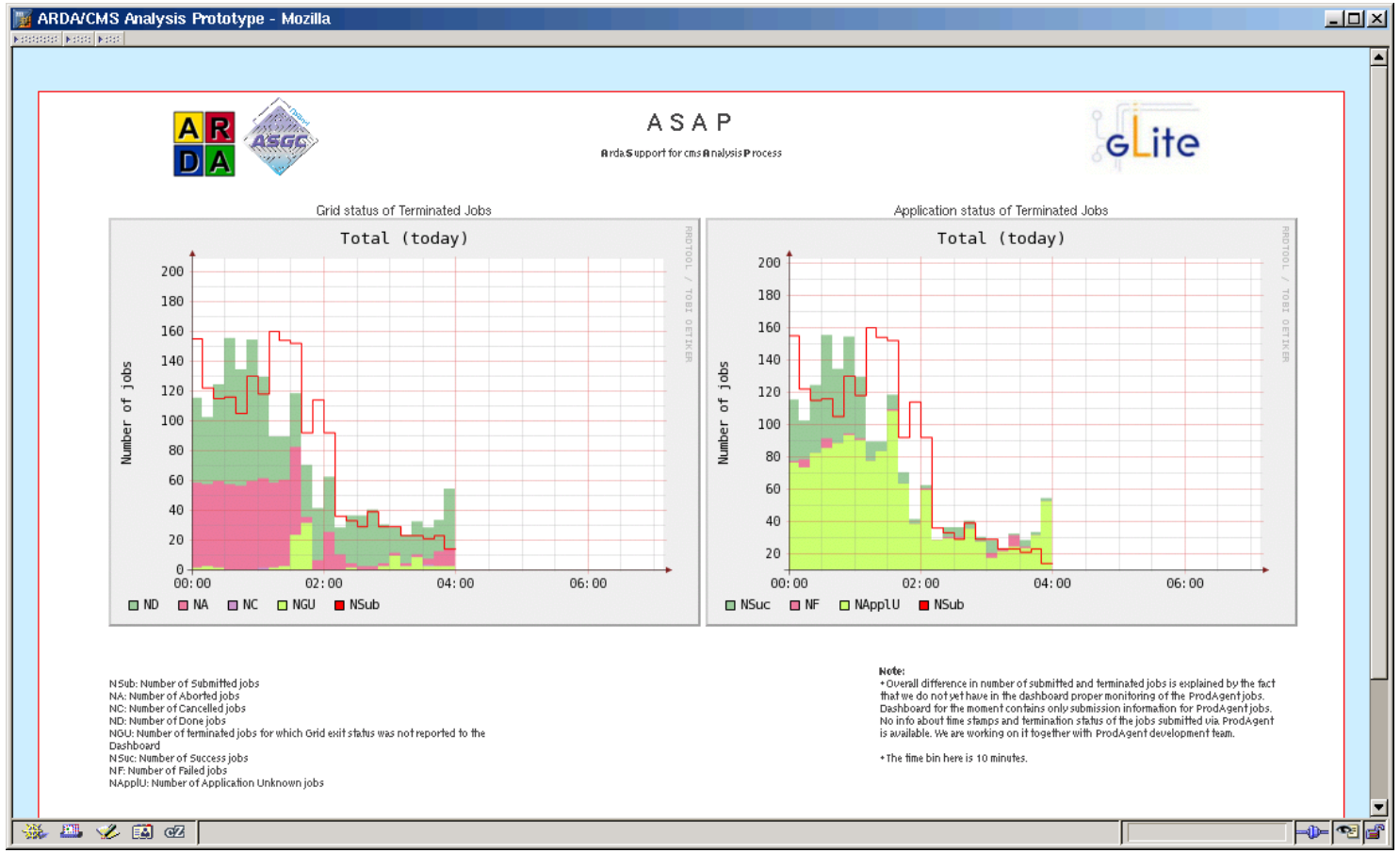
cmsaf.mit.edu	128	138	48.12%
CIEMAT-LCG2	107	52	67.30%
RAL-LCG2	211	91	69.87%
INFN-LNL-2	177	74	70.52%
desy.de	202	79	71.89%

Done

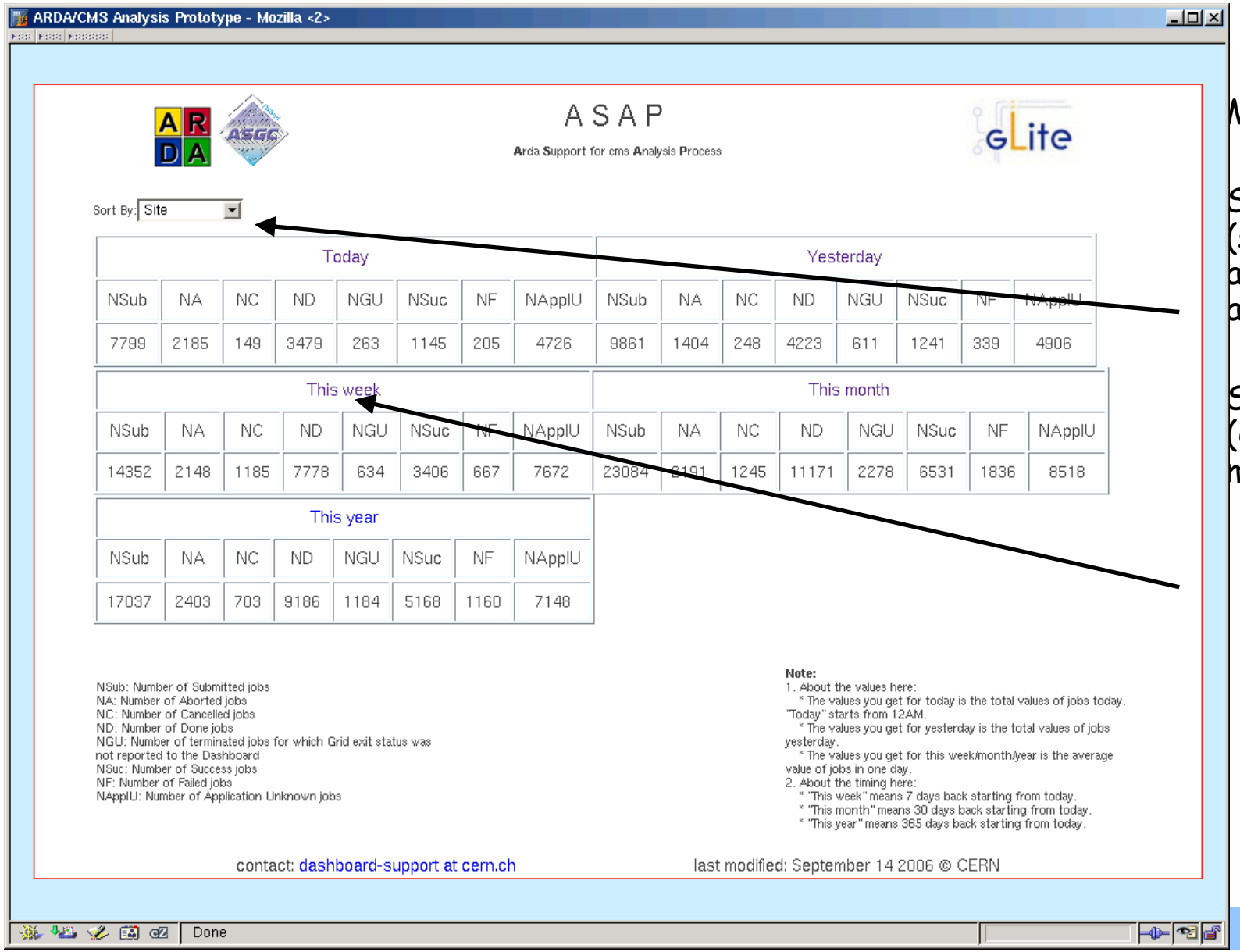
Pre-defined views (“coffee views”)

- Fast plots, by time period, time history
- http://arda-dashboard.cern.ch/cms/jobmon-history

Example of time history



Navigation by table



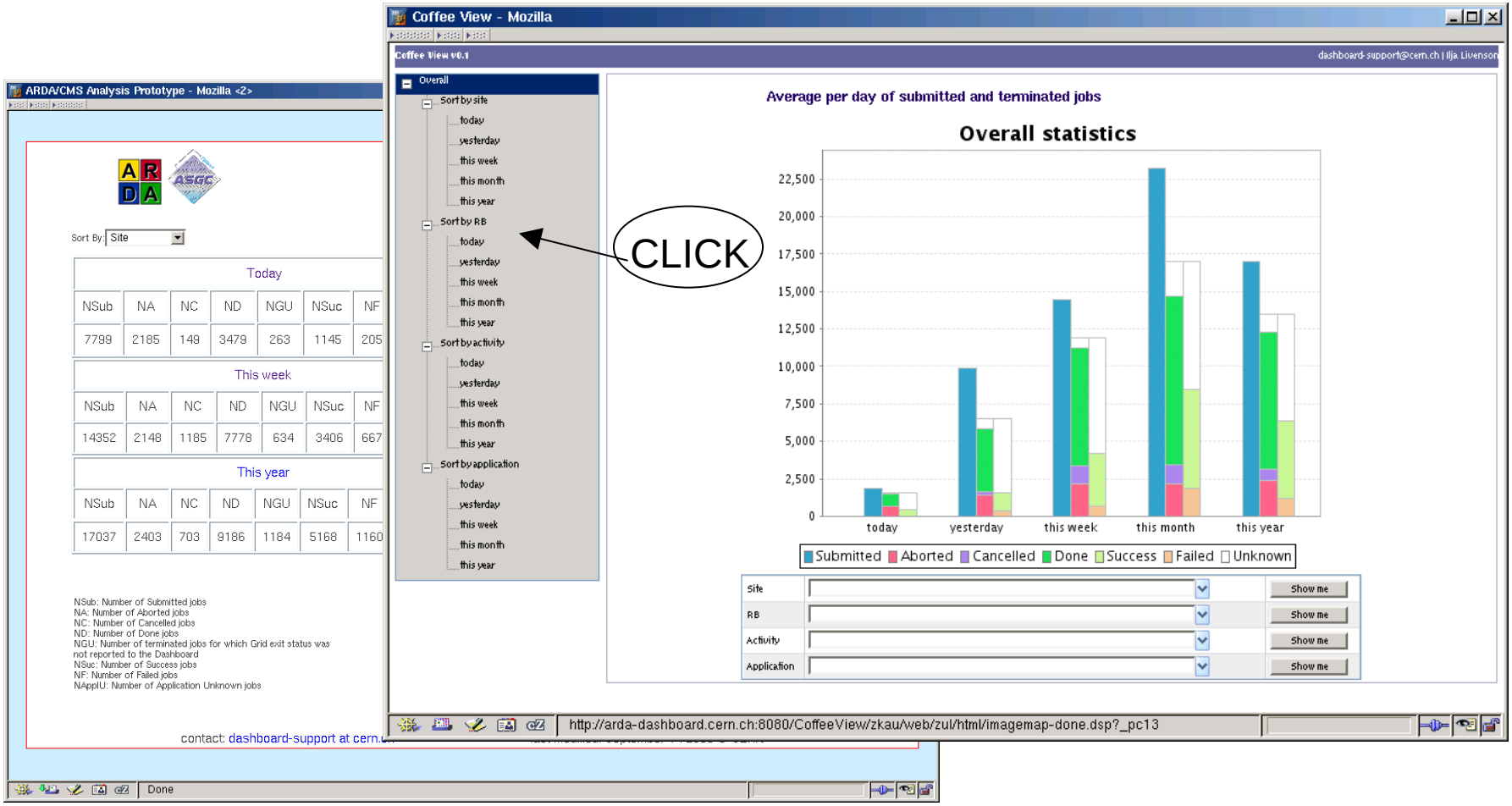
Main page

Select aggregation (site, RB, application, activity)

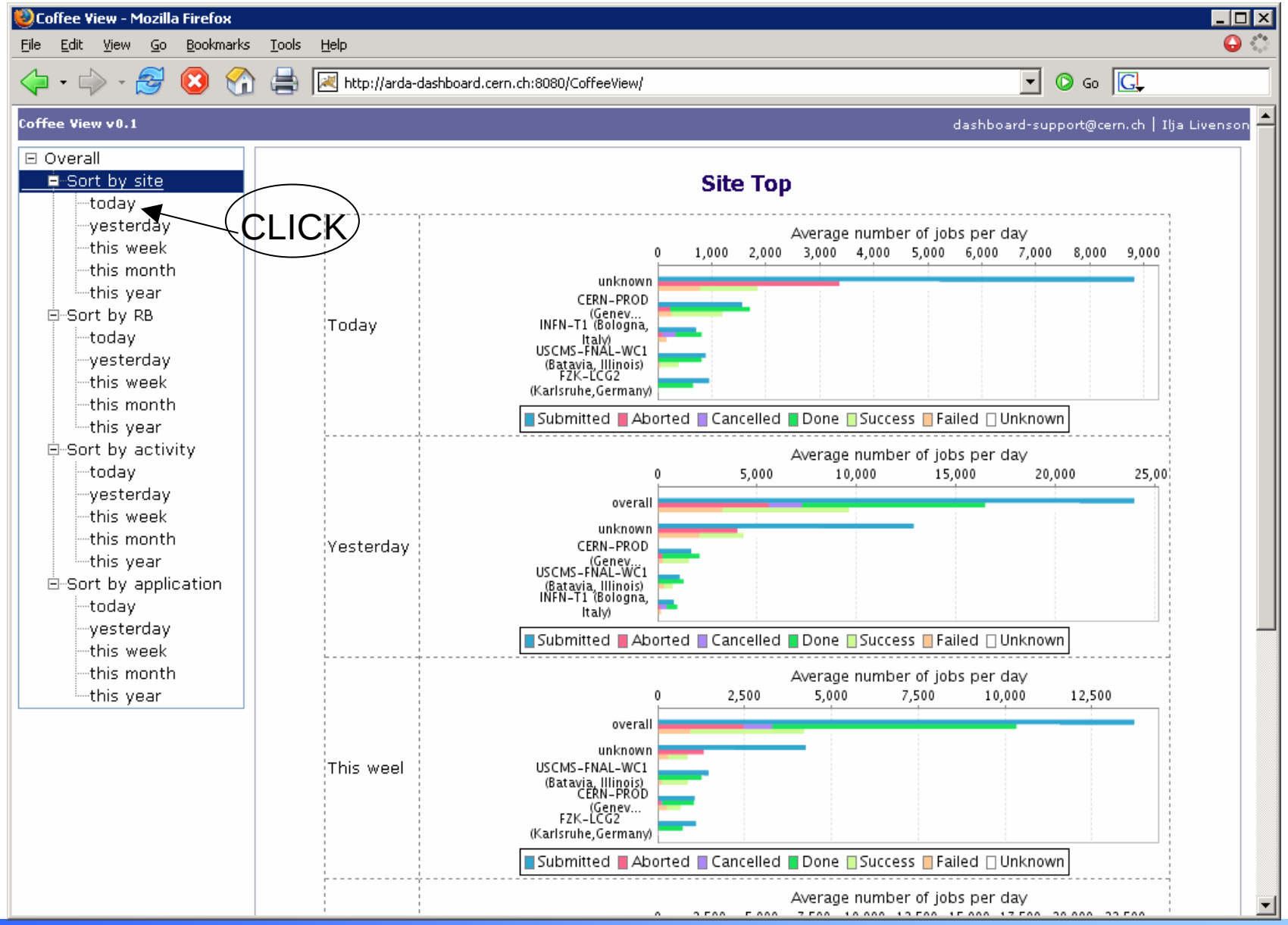
Select time frame (day, week, month,year)

Navigation by click on pictures

- Graphical interface as an entry point for the time history
- http://arda-dashboard.cern.ch:8080/CoffeeView/

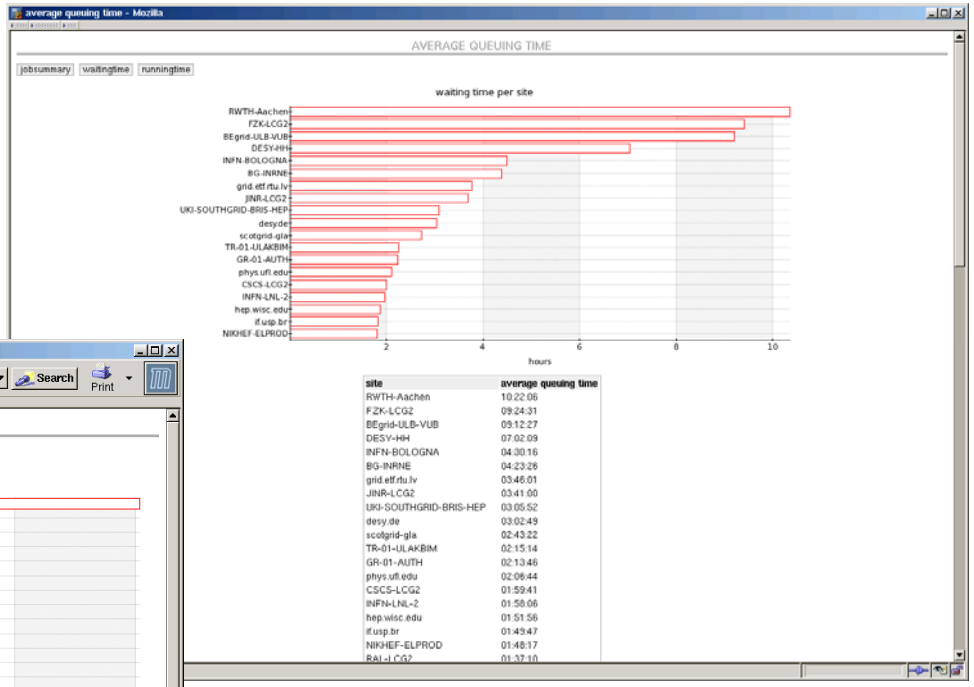


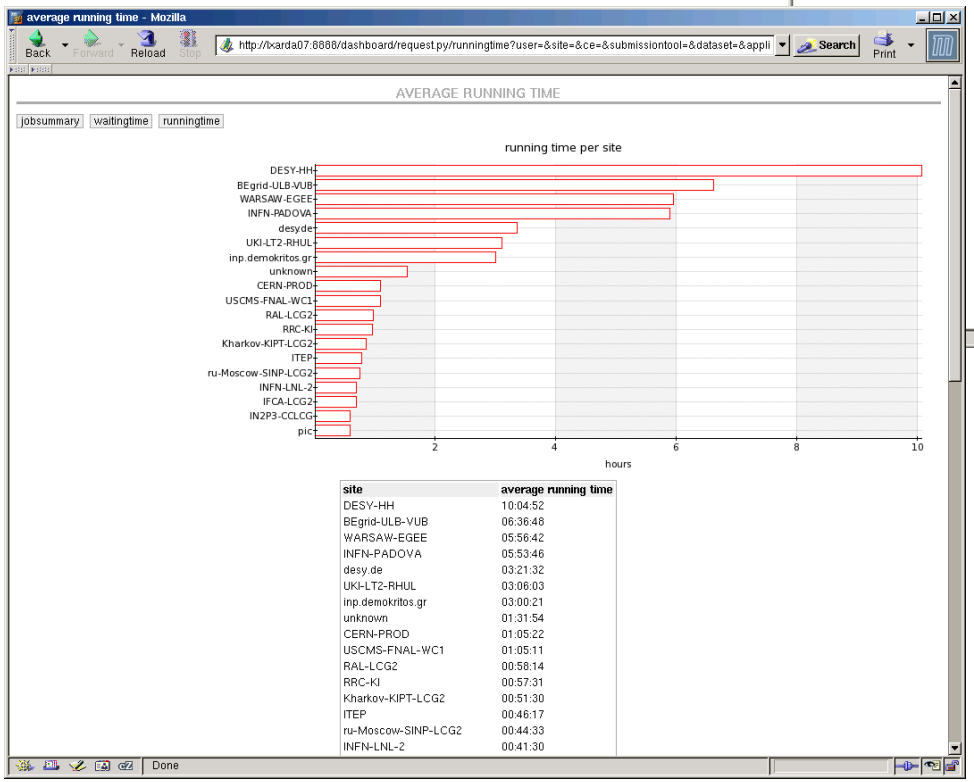
Keep clicking to discover more plots



Jobs waiting and running times

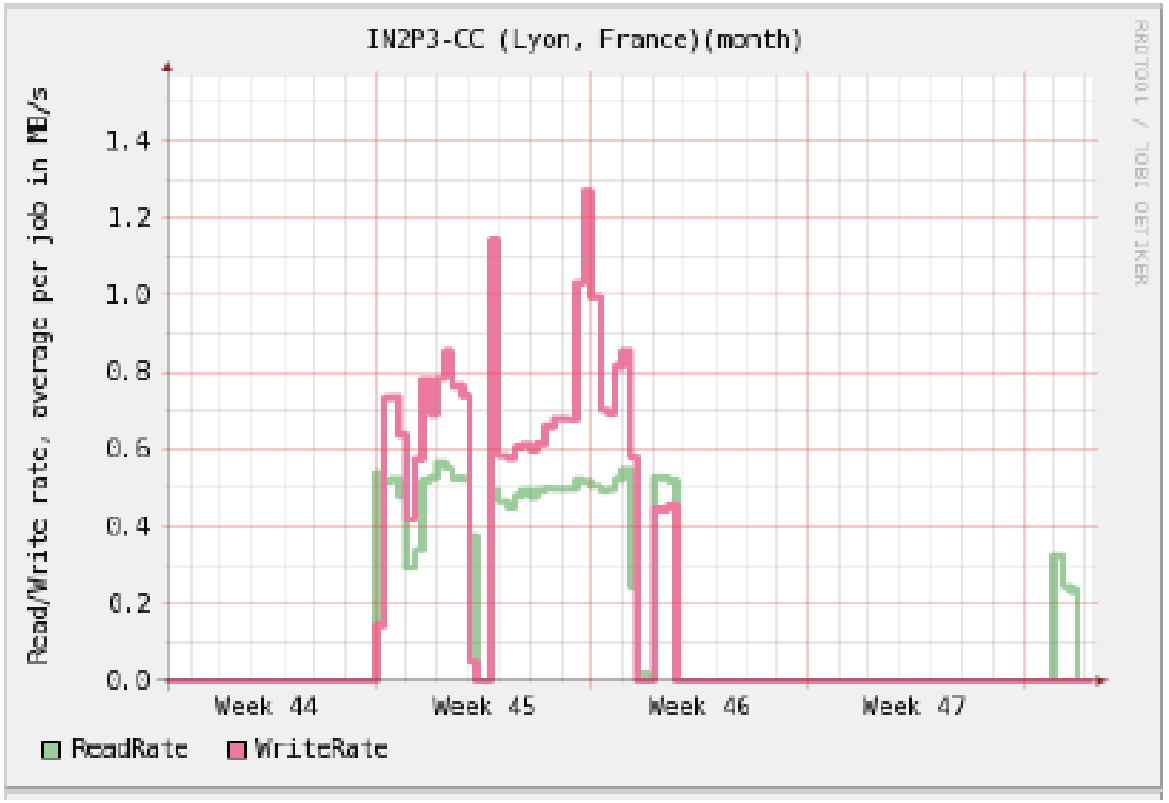
- Per site
- Per application type
- Per user (group)





Data access rates at the site

I/O rate in MB/s



To be used toghether with local monitoring, dashboard gets this from job summaries, and can aggregate for application, task within CMS. Site monitoring usually sum all usage by possibly several experiments.

Summary on monitoring

- When you need to understand a complicated system (a Tier2 is complicated enough), information is never too much
- There for CMS developed interal monitoring tools to complement, not replace everyhthing else

- PhEDEx and Job Dashboard monitoring allow to look at things from the CMS persective, aggregating information that makes sense with respect to CMS operations
  - CMS datasets
  - CMS applications, tasks
  - CMS users, groups

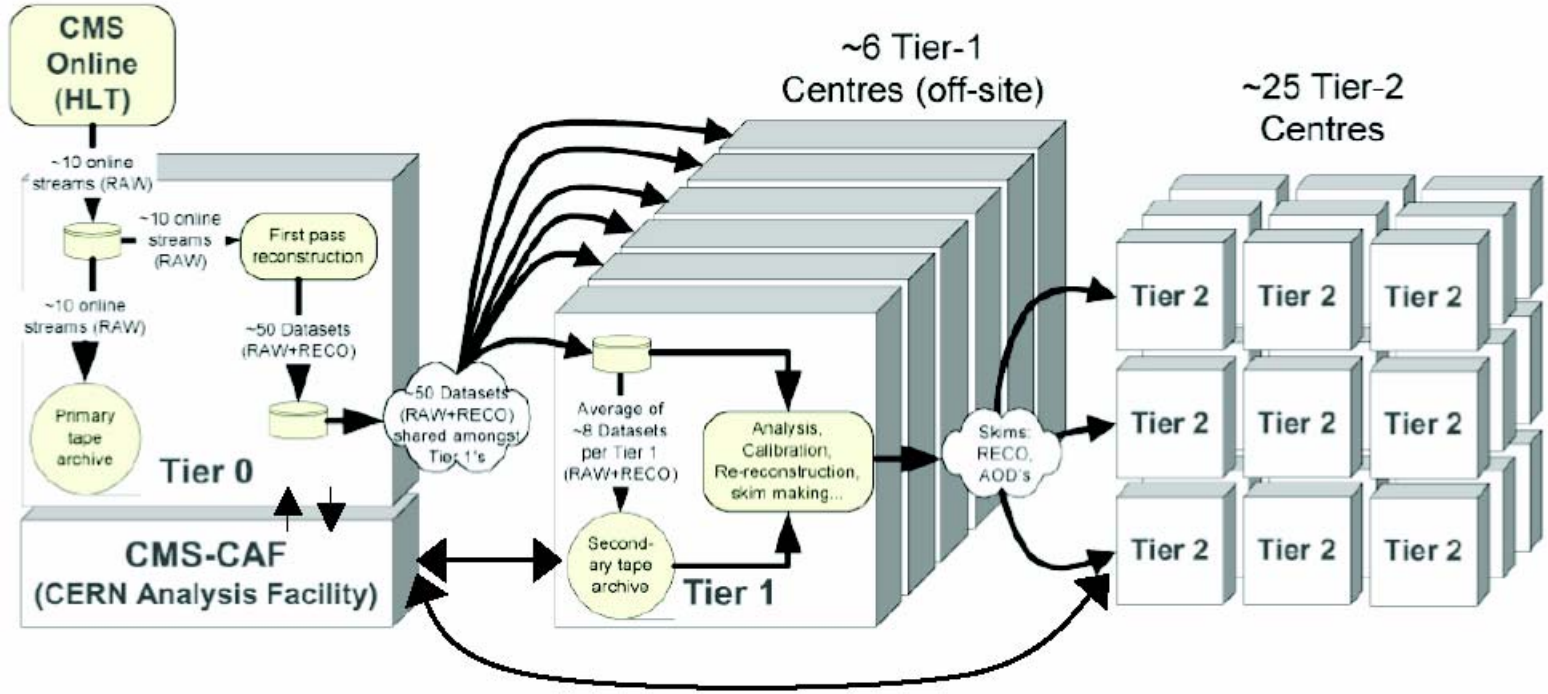
Conclusion

- Questions ?

Spares

- Spare slides follow

Tiered Architecture



Tier-0:

- Accepts data from DAQ
- Prompt reconstruction
- Data archive and distribution to Tier-1's

Tier-1's:

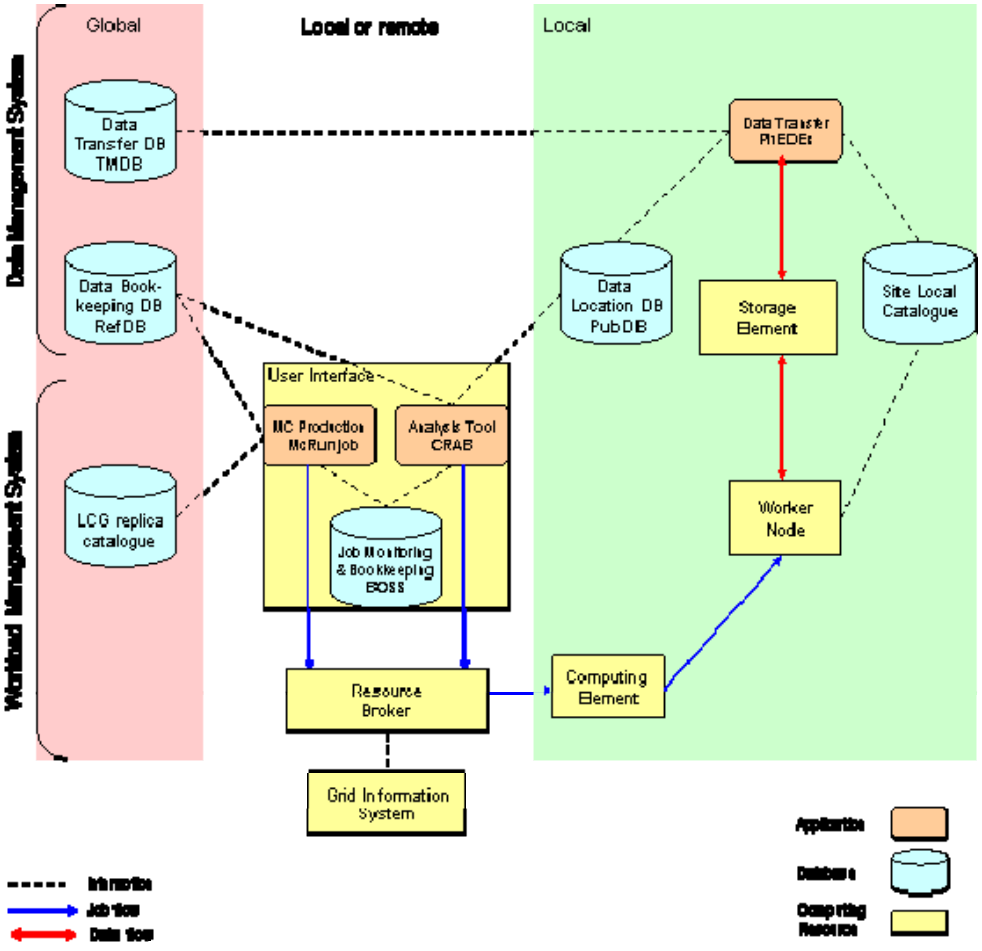
- Real data archiving
- Re-processing
- Skimming and other data-intensive analysis tasks
- Calibration
- MC data archiving

Tier-2's:

- User data Analysis
- MC production
- Import skimmed datasets from Tier-1 and export MC data
- Calibration/alignment

WMS & DMS Services Overview

Current WMS & DMS



Data Management System

- No global file replica catalogue
- Data Bookkeeping and Data Location Systems
  - What data exist & where are locatec
- Local File catalogue
- Data Access and Storage
  - SRM and posix-IO-like
- Data Transfer and placement system
  - PhEDEx

Workload Management System

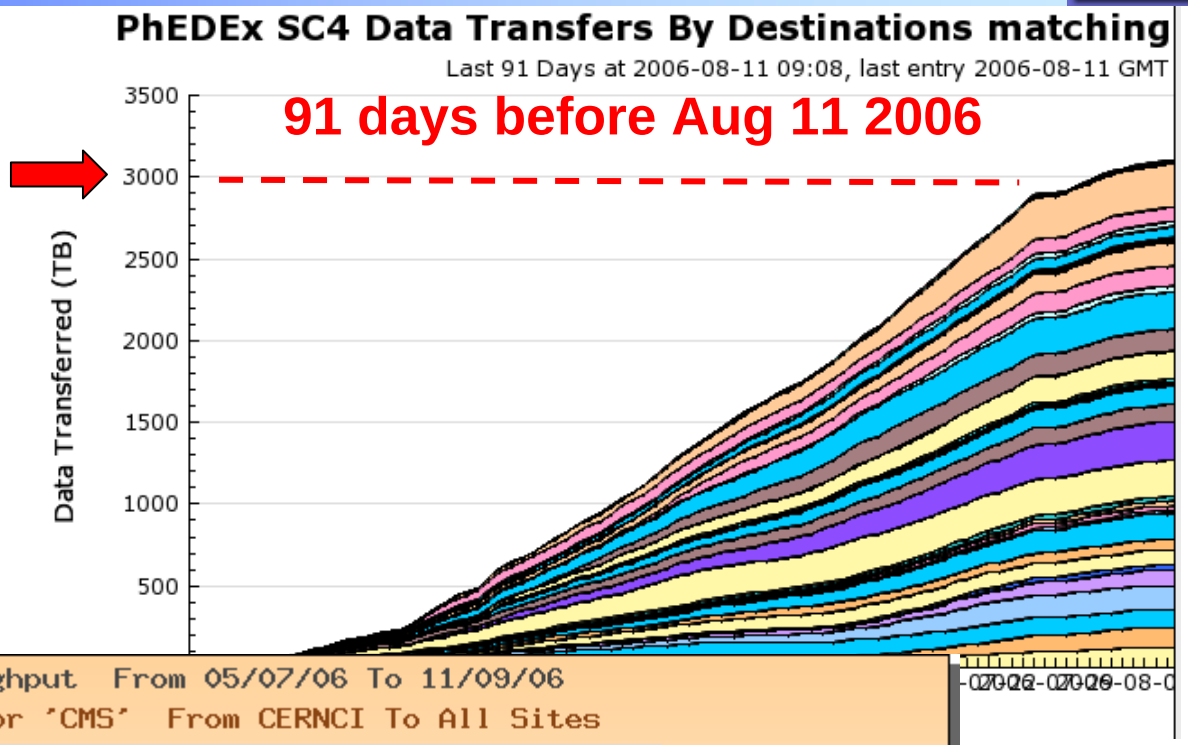
- Rely on Grid Workload Management
  - Reliability, performance, monitoring, resource management, priorities
- CMS-specific job submission, monitoring and bookkeeping tools

Data Processing

- Data are used by workflows. **All workflows the same** (roughly):
  - MonteCarlo, Reconstruction, Analysis…
  - Run application on all files of Dataset D-In (or just N times for Initial MC generation), produce Dataset D-Out
- In practice
  - Access DBS to list Fileblocks for D-In. Access DLS to find locations
  - Split in N jobs to exploit farms. Obtain N output files
    - copy those files to final destination (now or later)
  - Register N files in Dataset D-Out in DBS/DLS
- Special (and VERY common) case: file merging
  - Collect/merge N small outputs in fewer larger files (w/o mistakes)
  - Is still the same workflow: run many jobs, each application instance reads many files to produce a single one.
- CMS ProductionAgent to address this

Test Tler0-Tier1 transfers at 2008 rates

More then 3PB of data transferred by CMS in 3 months



Over 300MB/sec peak from CERN to Tier1's