



Use of the gLite-WMS in CMS for production and analysis

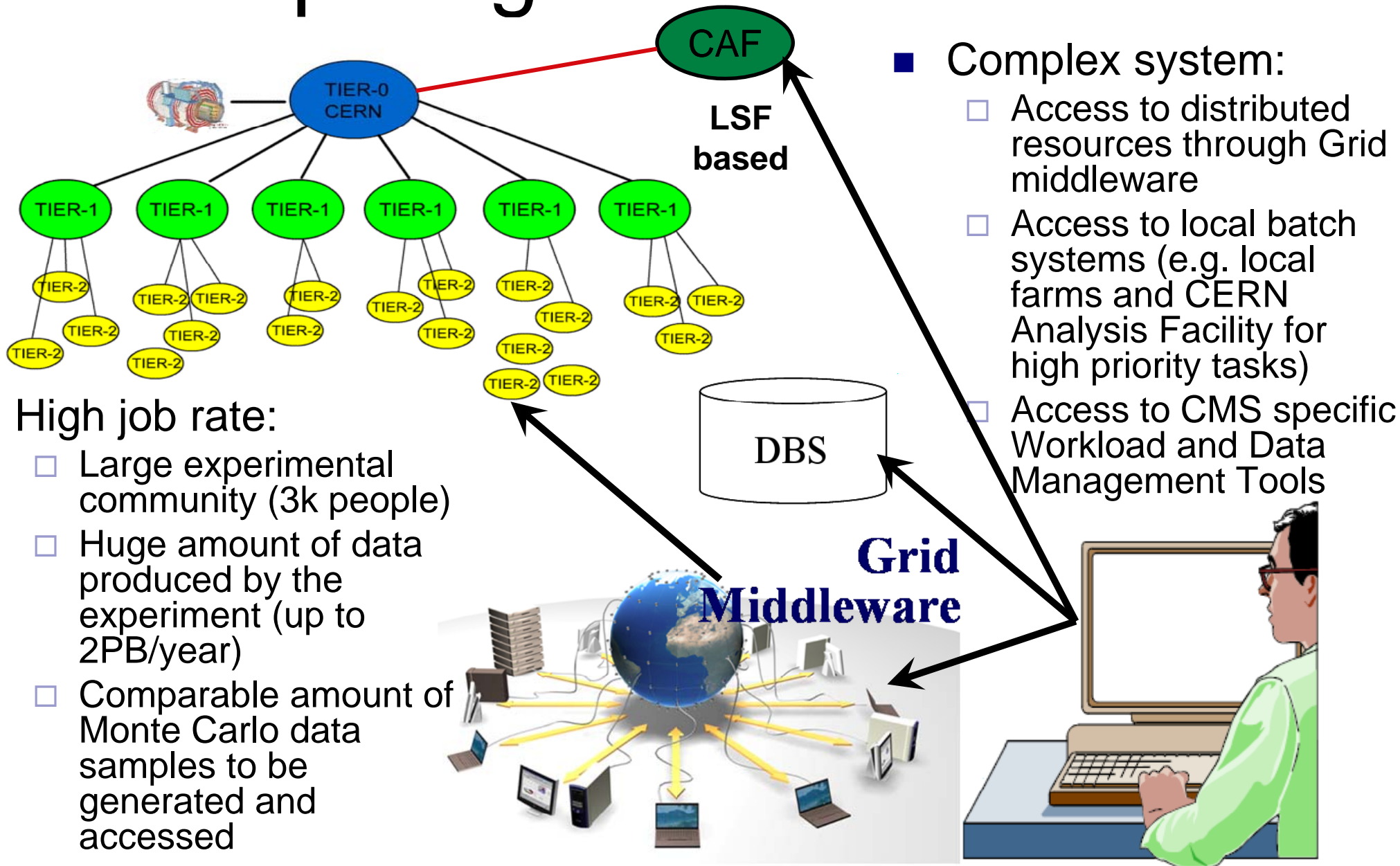
Giuseppe Codispoti

On behalf of the CMS Offline and Computing

Outline

- BossLite: the common interface to Grid and batch systems for the CMS tools
- gLite usage through BossLite
- gLite integration in the CMS tools
- Issues and proposed solutions
- WMS usage in analysis and MC production activities
- Overall performances
- Conclusions

Computing Model Overview

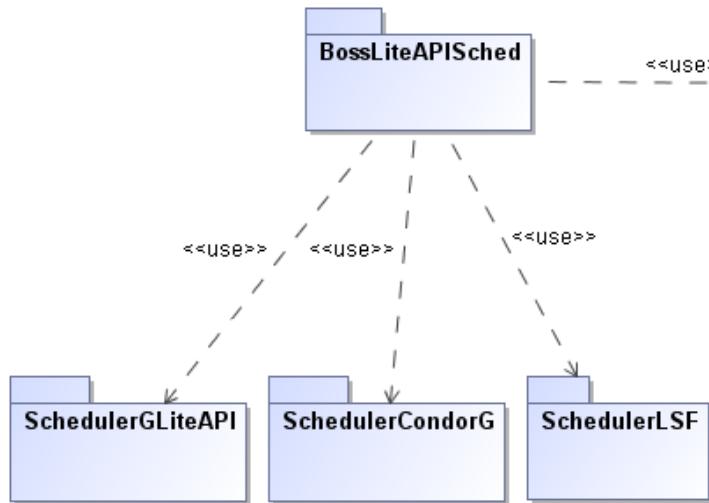


See talk [192] by I. Fisk: [Challenges for the CMS Computing Model in the First Year](#)

BossLite: a common Grid/batch interface with logging facilities

- CMS interface to different Grid [WLCG, OSG] and batch systems [LSF, ARC, SGE...]
- Database to track and log information into an entity-related schema
- Information logically remapped into python objects that can be transparently used by the CMS framework and tools
- Requested high efficiency and safe operation in multithreaded environment
- Database interaction through safe sessions and connections
 - Connections pool
 - Thread safe
 - Focused on connection stability

BossLite Architecture

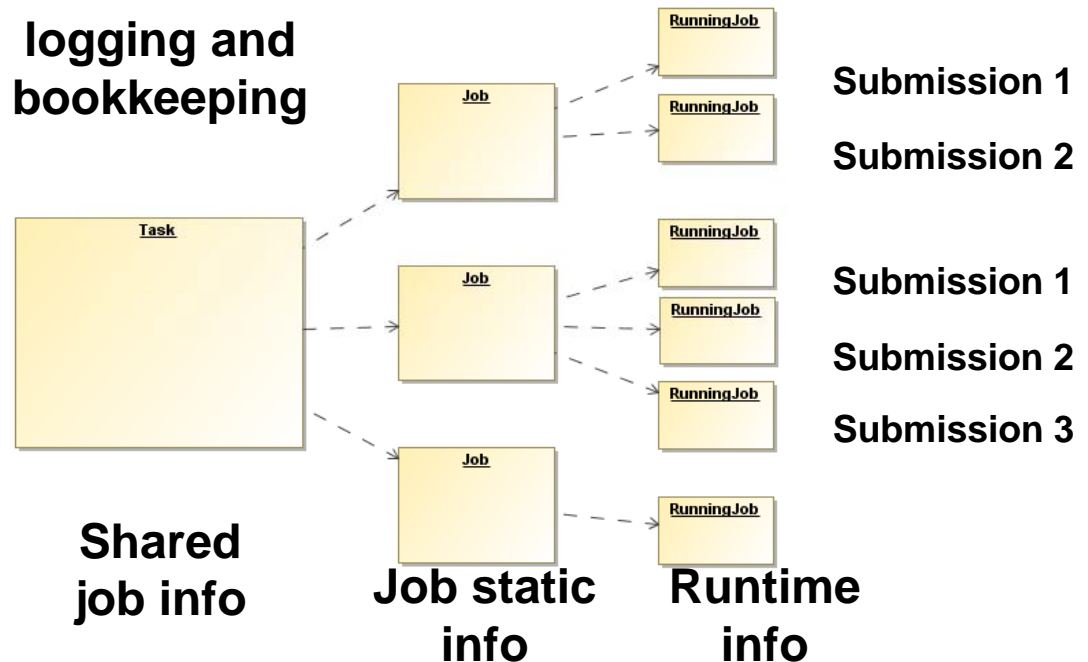


Plugins for transparent interaction with Grid and local batch systems

User Task Description:
 identical jobs accessing different part of a dataset or producing a part of a MC sample

DataBase backend for logging and bookkeeping

Pool of DB connections to be shared among threads



The BossLite interface to gLite

- Bulk submission, bulk match-making, bulk status query
 - Faster, more efficient

- Access through WMPProxy python API
 - Needed to allow the association between BossLite jobs and their grid Identifiers (not trivial through the CLI)
 - No parsing of "human readable" streams
 - But the use of the API is complex and the UI tools (e.g. UI configuration, input sandbox transfers, ...) are lost
 - Exposed to python compatibility issues

- Access to LB information through API
 - Easy and fast check of job status
 - Easy to extract many more useful information at runtime: destination queue, status reason, scheduling timestamps...

- Note: the CMS computing model uses its own data location system: the WMS match making is only used to select among available resources hosting selected data.

CMS use cases

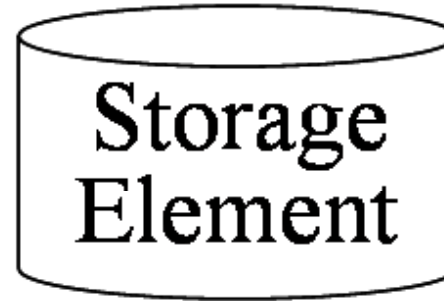
- Monte Carlo Production:
 - Automatic, parallelized system for huge data samples simulation/reconstruction

- Basic analysis tasks (single user):
 - Transparent usage of the Grid infrastructure as well as local batch system, integrated with the CMS workload management system

- Regime Analysis and intensive analysis tasks
 - Centralized system dealing with huge tasks, automating the analysis workflow, optimizing Grid usage
 - High concurrency system for multiuser environment

Production Agent

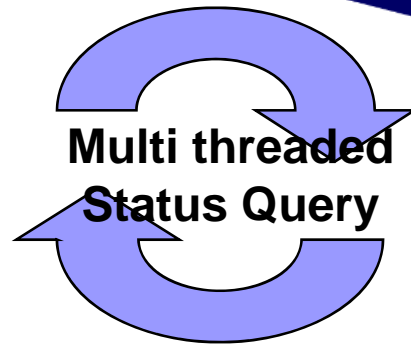
ProdAgent



Files limited in size, produced directly in CMS destination sites (merged later through ad hoc jobs)

Sequential Job Submission (collections)

Multi Threaded Output Retrieval for log files and production report

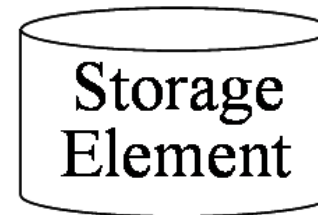


See Poster [82] by F. Van Lingen:

[MC production and processing system – Design and experiences](#)

CMS Remote Analysis Builder

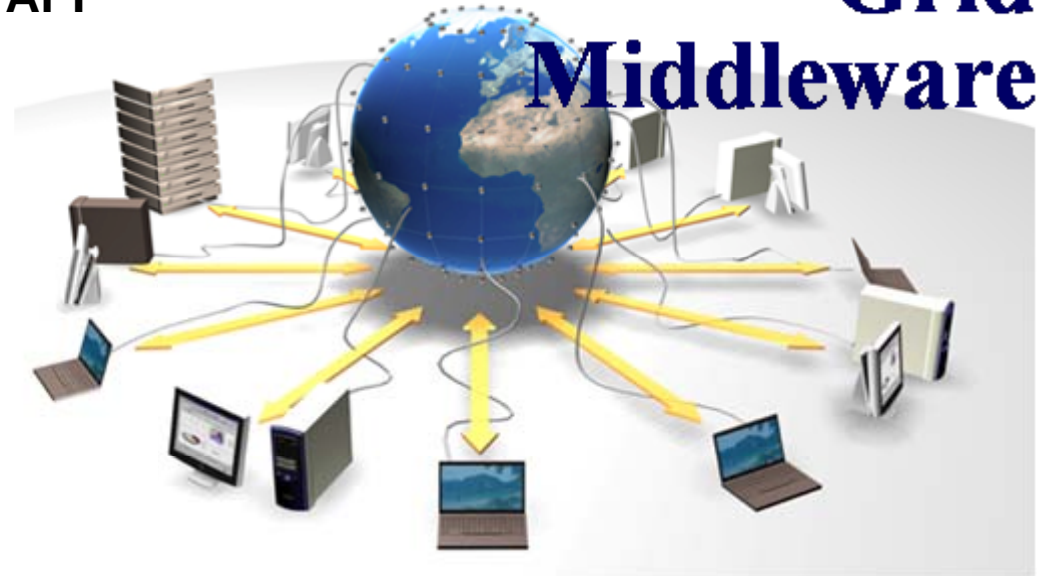
CRAB client



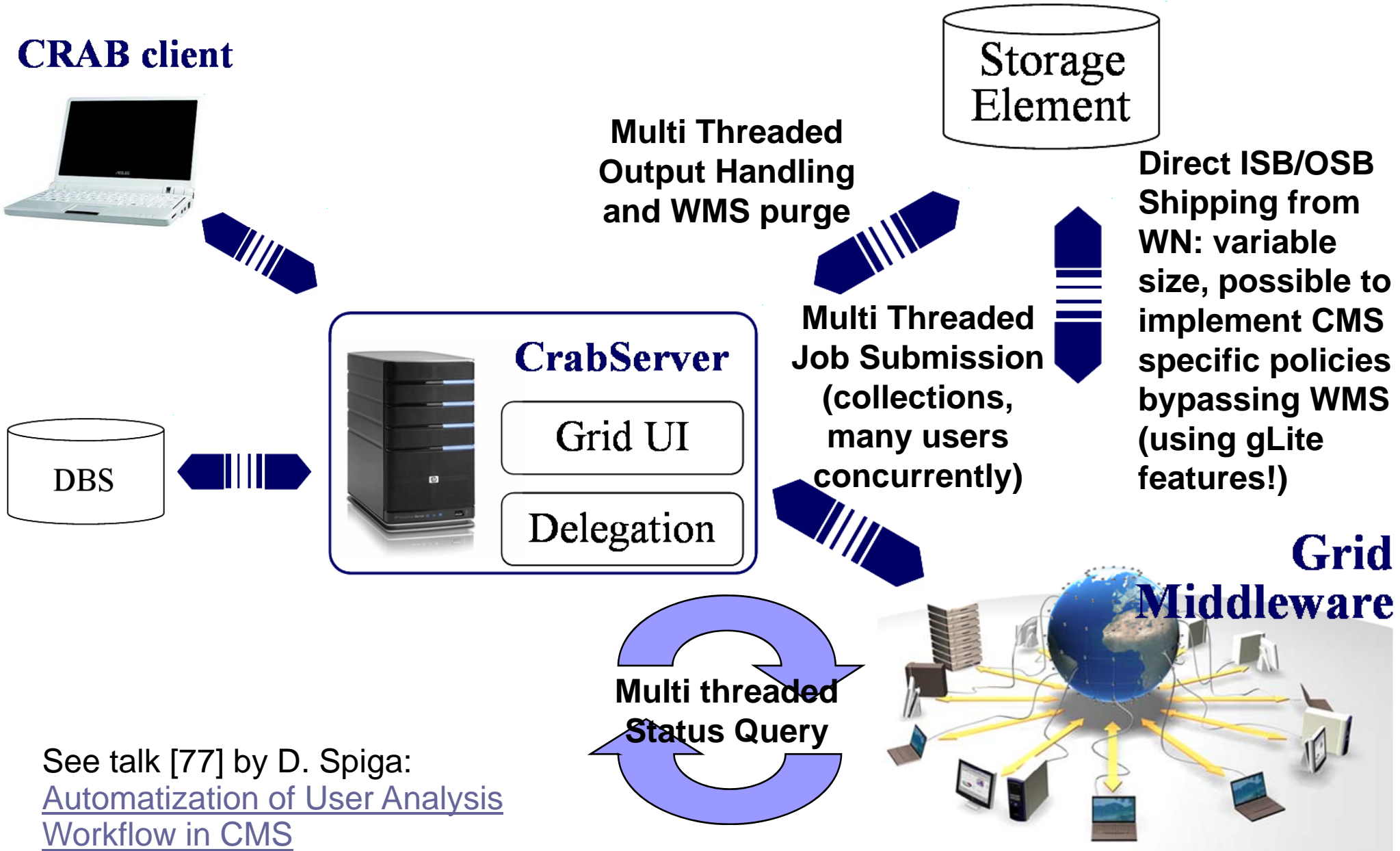
All UI functionalities wrapped with WMPProxy/LB API



Grid Middleware



CRAB Analysis Server



See talk [77] by D. Spiga:
[Automatization of User Analysis Workflow in CMS](#)

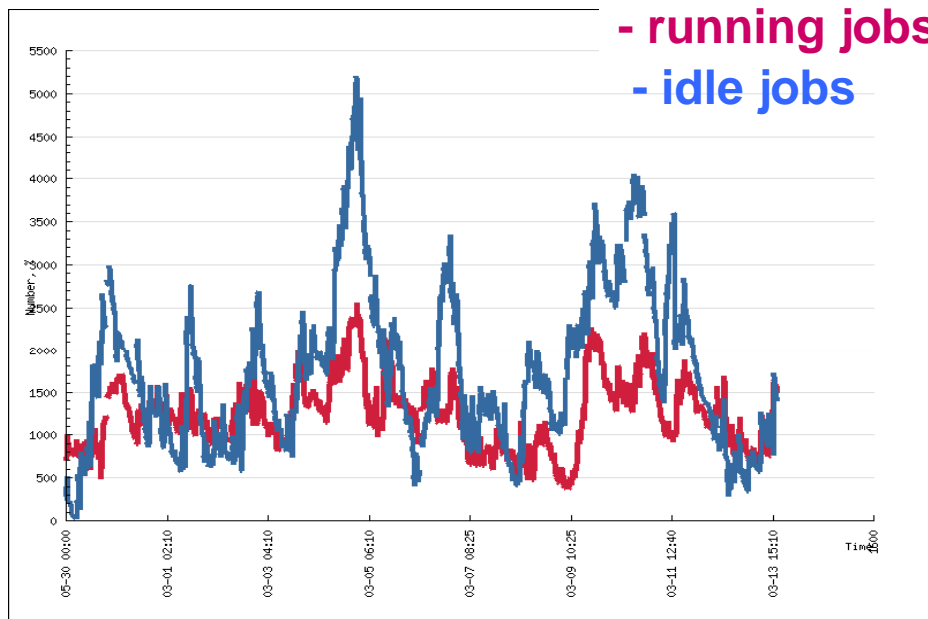
Evolution

- Fruitful collaboration with gLite developers to fix bugs and implement the features CMS needs

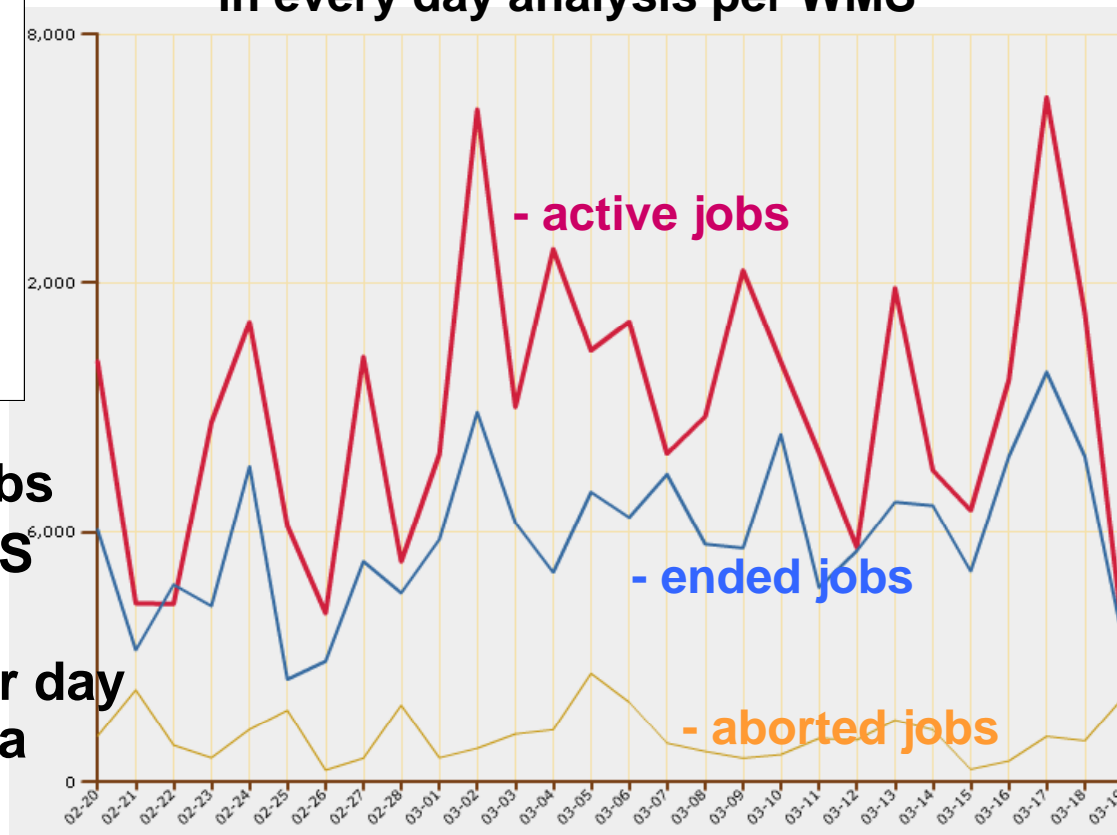
- Proposed xml/json output to CLI commands
 - Same level of detail for job association
 - Reusing all UI functionalities, everything is already there, needs just to be made accessible
 - Specific error logging
 - Accessible through simple subprocess, encapsulating environment/compatibility issues
 - Simpler intermediate layer
 - Reuse&simplify!

Single WMS usage in every day activities

- MC production and analysis jobs are balanced over many WMS: currently 7 for the analysis, 4 for the MC production

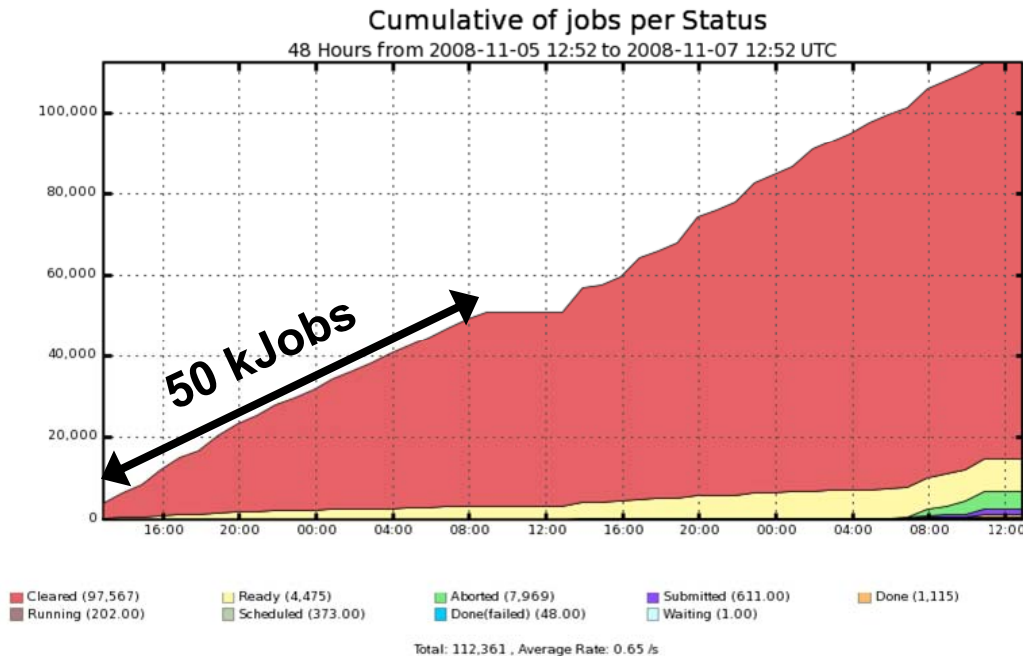


- Typical instantaneous load of a single WMS (jobs running/idle)
 - Up to 5 kJobs simultaneously handled in every day analysis per WMS



- Daily job rate, including ended jobs
- Typical jobs load for a single WMS may already reach 15k jobs
- Stress tests reached 30 kJobs per day without breaking point signal for a single WMS!

Overall Performances of the CMS tools

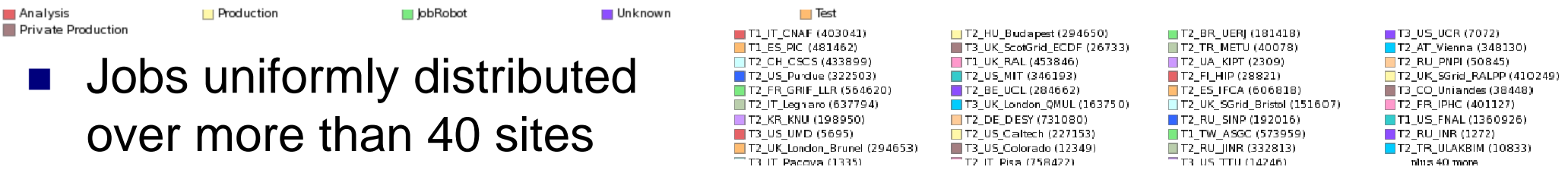
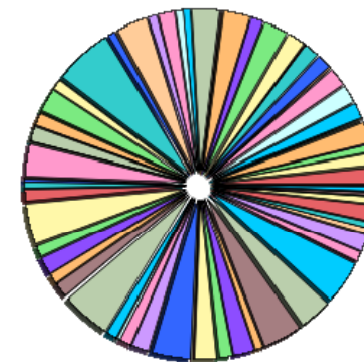
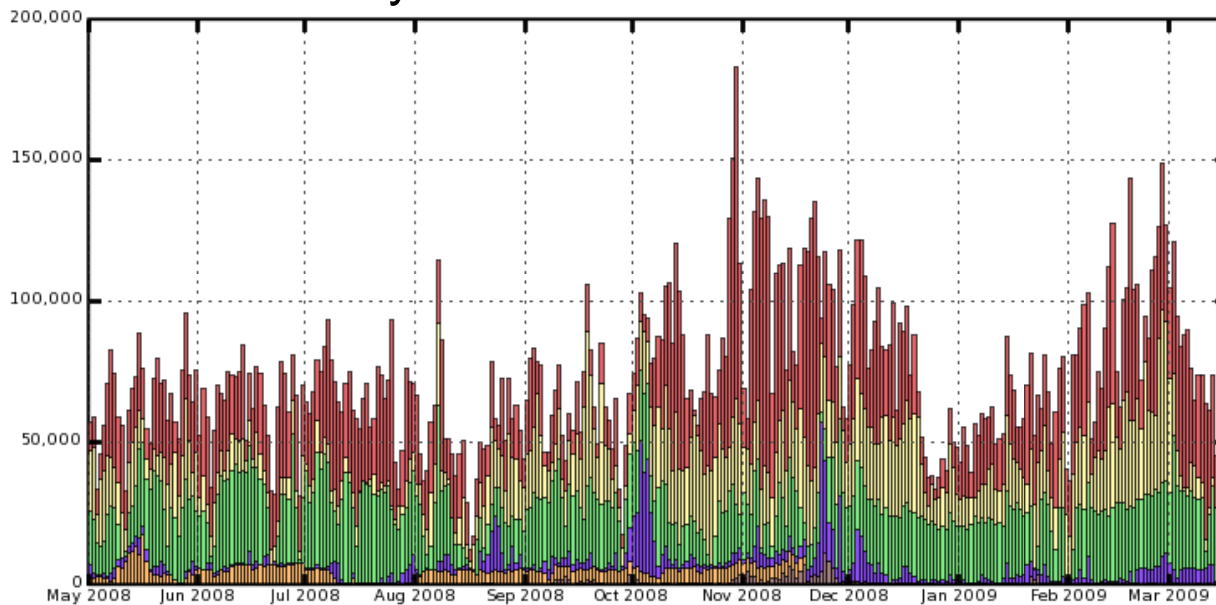


- Single CRAB Server instance in multi use mode reached 50KJobs per day using 2 WMS
- A single ProdAgent instance reached around 30kJobs per day
 - Lower performance in the output copy from the WMS: we plan to reduce the size and number of the files to be retrieved

- Reached limits are mainly due to tracking and output retrieval/handling
- Some optimizations already in place, other small tweaks possible
- The WMS architecture is such that the system scales linearly with the number of WMSs
 - Add as many WMSs to a CMS service as needed
- The CMS architecture is similar:
 - Deploy as many instances of PA and CRAB Server as needed:
- No scale problems foreseen at the expected rates
 - 50/100 kJobs/day for MC production and 100/200 kJobs/day for analysis

CMS grid activity with gLite WMS

- CMS uses gLite WMS since years
- Increase in the activity during last year
 - May 2008 challenge (CCRC08, see [312],[389])
 - From May 2008 to March 2009
- ~75k per day
 - ~30k analysis
 - ~20k MC production
 - ~25k other activities



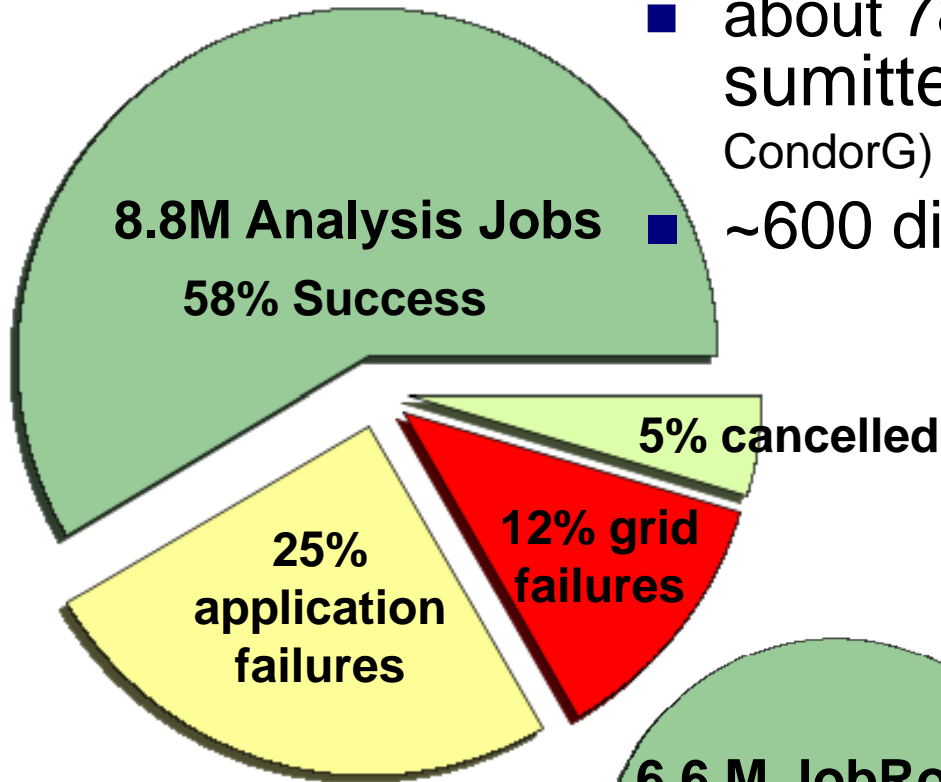
- Jobs uniformly distributed over more than 40 sites

Poster [389]: [CMS results from Computing Challenges and Commissioning of the computing infrastructure](#)
 Poster [312]: [Commissioning Distributed Analysis at the CMS Tier-2 Centers](#)

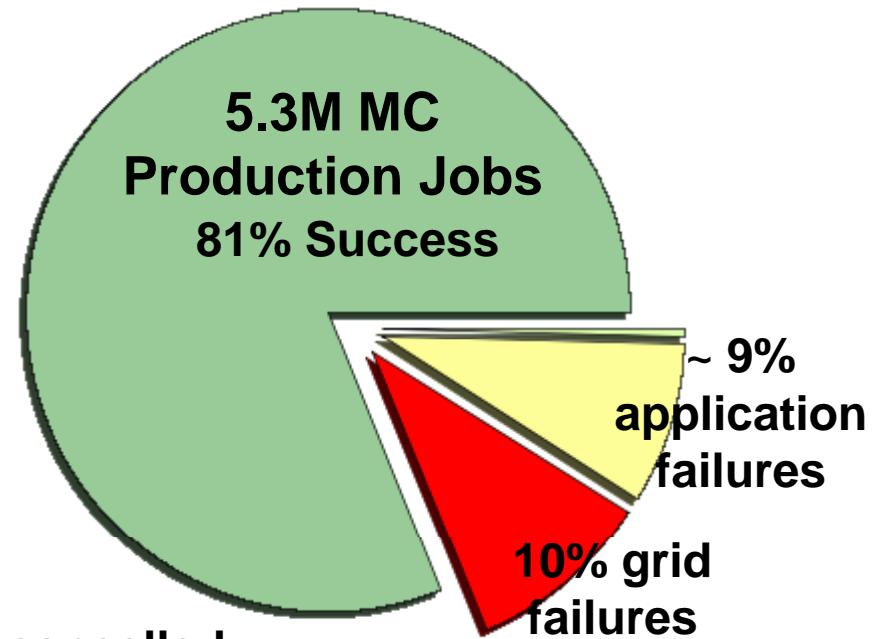
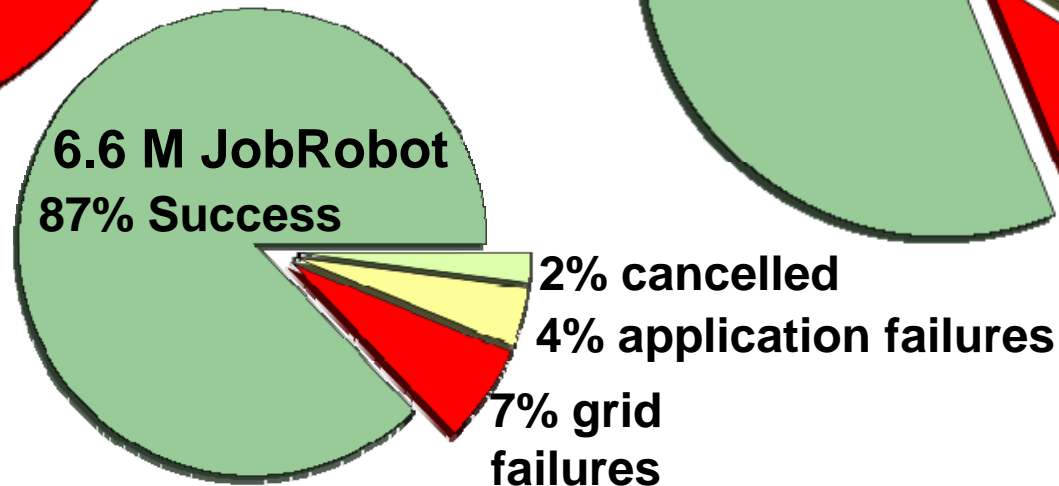
Job distribution per activities

From May 2008 to March 2009 : 23 M total jobs submitted

- about 78% of the total analysis jobs are submitted with the gLite WMS (the rest mainly CondorG) since years!
- ~600 distinct real users in the last 3 months



+ 2,3M jobs other test activities



Conclusions

- CMS successfully uses the WMS in MonteCarlo Production and Analysis tasks
- We are able to reach more than 30kJobs with a single WMS
- Each CMS application service may use in parallel as many WMSs as needed:
 - Up to 50 kJobs from a single CMS server
- We are able to cover CMS requests delivering few instances of CRAB/ProdAgent
- The every day experience and usage allows us to improve the system and provide feedback to WMS/gLite developers since years

Author list

- G. Codispoti , C. Grandi , A.Fanfani (Bo-INFN)
- D. Spiga, V. Miccio , A. Sciaba' (CERN)
- F.Fanzago (CERN,CNAF-INFN)
- M. Cinquilli (Perugia-INFN)
- F. Farina (Mi-INFN,CERN)
- S. Lacaprara (LNL-INFN)
- S. Belforte (Trieste-INFN)
- D. Bonacorsi , A.Sartirana, D. DonGiovanni, D. Cesini (CNAF-INFN)
- S.Lemaitre , M.Litmaath, E.Roche, Y.Calas (CERN)
- S. Wakefield (IC-London)
- J. Hernandez (CIEMAT)