DIRAC – Grid solution for the LHCb community

A.Tsaregorodtsev,
CPPM-IN2P3-CNRS, Marseille

20 October 2009, Webinaire IN2P3

Outline

- Specific issues of large Grid Communities
- DIRAC:
  - Framework
  - WMS with Pilot Jobs
  - Security aspects of the model
  - User interfaces
- LHCb extensions to DIRAC
- Conclusion

HEP applications

- HEP experiments collect unprecedented volumes of data to be processed on large amount of geographically distributed computing resources
  - 10s of PBytes of data per year
  - 10s of thousands CPUs in 100s of centers
  - 1000s of users from 100s of institutions
- However, other application domains are quickly approaching these scales

Large VO issues

- Large user communities (Virtual Organizations) have specific problems
  - Dealing with heterogeneous resources
    - Various computing clusters, grids, etc
  - Dealing with the intracommunity workload management
    - User group quotas and priorities
    - Priorities of different activities
  - Dealing with a variety of applications
    - Massive data productions
    - Individual user applications, etc

General problems

- Overcome deficiencies of the standard grid middleware
  - Inefficiencies, failures
    - Production managers can afford that, users can not
  - Lacking specific functionality
- Alleviate the excessive burden from sites – resource providers – in supporting multiple VOs
  - Avoid complex VO specific configuration on sites
  - Avoid VO specific services on sites

VO solutions

- The complexity of managing the VO workload resulted in specific software layer on top of the standard grid middleware. Among the LHC experiments
  - ***AliEn*** in Alice
  - ***PanDA*** in Atlas
  - ***GlideIn WMS*** in CMS
  - ***DIRAC*** in LHCb

DIRAC Project

DIRAC Community Grid Solution

- DIRAC is a distributed data production and analysis system used by the LHCb experiment
  - Includes workload and data management components
  - Was developed originally for the MC data production tasks
  - Extended to data processing and user analysis
  - The goal was:
    - Integrate all the heterogeneous computing resources available to LHCb
    - Minimize human intervention at LHCb sites

DIRAC: complete chain

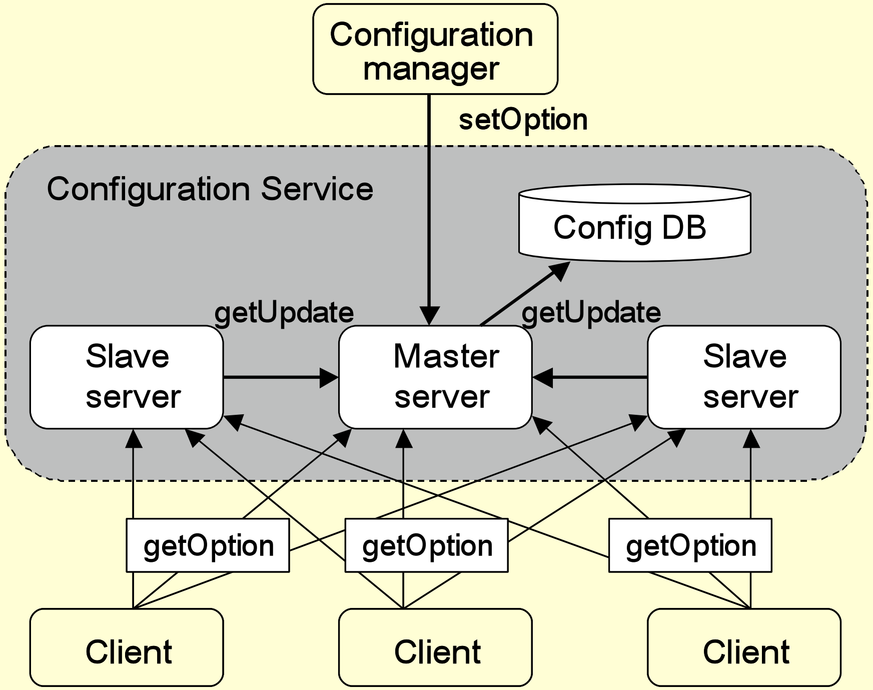
- DIRAC is covering all the LHCb needs in the distributed data processing
  - Data export from the experiment pit to CERN off-line storage
  - Automatic data distribution to Tier-1 centers
  - Automatic creation and submission of the data reconstruction jobs
  - Automatic distribution of the analysis data
  - Full management of the MC data production
  - Full support for the user analysis jobs
- Different subsystems built in the same framework
  - Reuse of technical solutions in different subsystems
  - A concerted team of developers sharing experience

DIRAC Framework

- Services oriented architecture
  - DIRAC systems consist of services, light distributed agents and client tools
- All the communications between the distributed components are secure
  - DISET custom client/service protocol
    - Control and data communications
  - X509, GSI security standards
  - Fine grained authorization rules
    - Per individual user FQAN
    - Per service interface method
    - Per job

DIRAC base services

- Redundant Configuration Service
  - Provides service discovery and setup parameters for all the DIRAC components
- Full featured proxy management system
  - Proxy storage and renewal mechanism
  - Support for multiuser pilot jobs
- System Logging service
  - Collect essential error messages from all the components
- Monitoring service
  - Monitor the service and agents behavior



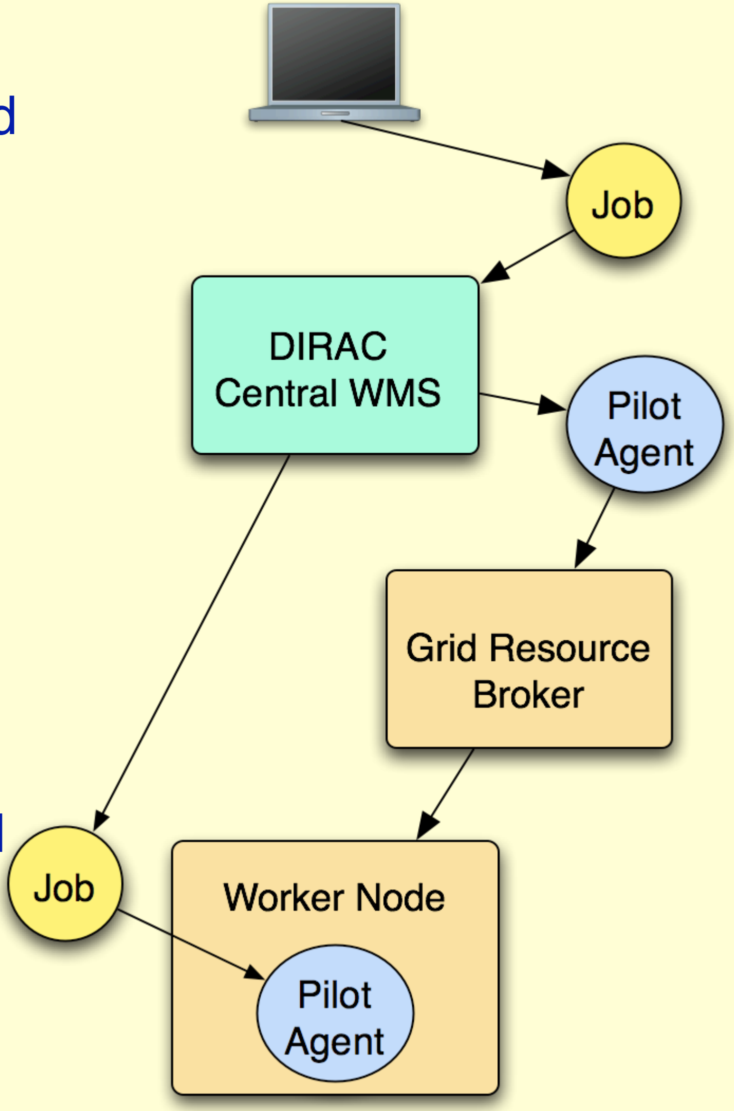
DIRAC development environment

- Python is the main development language
  - Fast prototyping/development cycle
  - Platform independence
- MySQL database for the main services
  - ORACLE database backend for the LHCb Metadata Catalog
- Modular architecture allowing an easy customization for the needs of a particular community
  - Simple framework for building custom services and agents

Workload Management

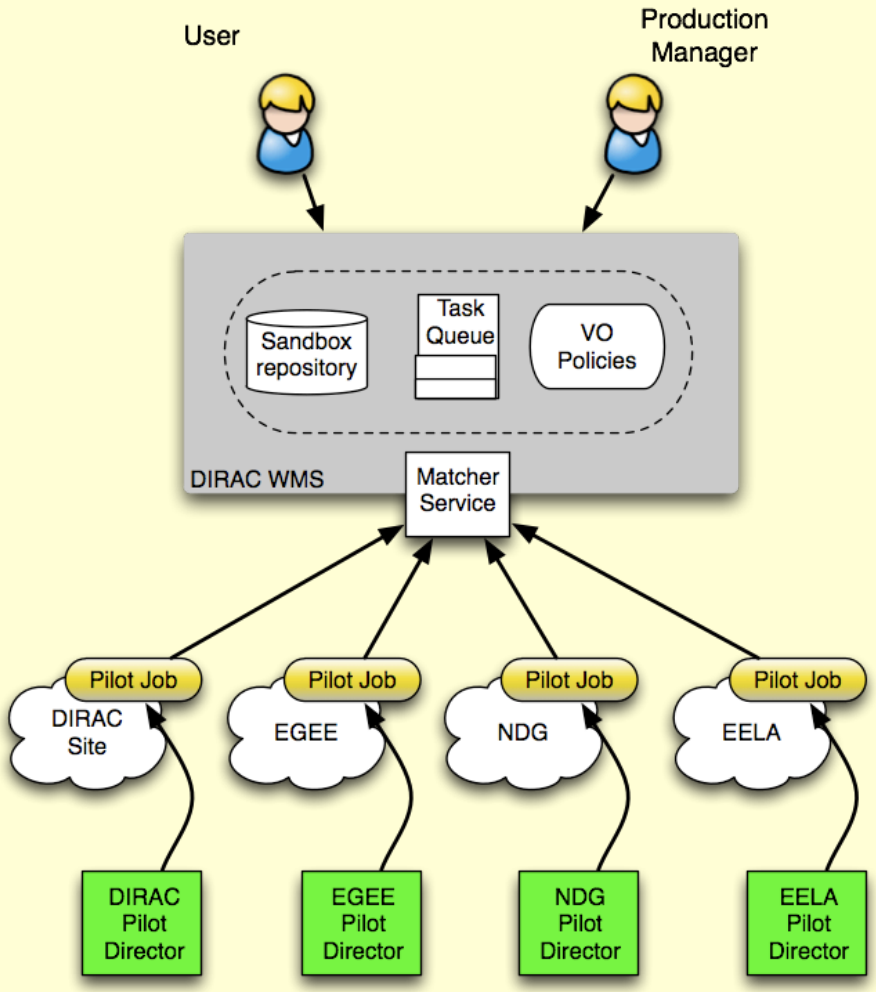
Pilot Jobs in a nutshell

- Pilot agents are deployed on the Worker Nodes as regular jobs using the standard grid scheduling mechanism
  - Form a distributed Workload Management system
  - Reserve the resource for immediate use
- Once started on the WN, the pilot agent performs some checks of the environment
  - Measures the CPU benchmark, disk and memory space
  - Installs the application software
- If the WN is OK the user job is ***pulled*** from the central DIRAC Task Queue and executed
  - Terminate gracefully if no work is available



DIRAC WMS

- Jobs are submitted to the DIRAC Central Task Queue with credentials of their owner (VOMS proxy)
- Pilot Jobs are submitted by specific Directors to a Grid WMS with credentials of a user with a special Pilot role
- The Pilot Job fetches the user job and the job owner's proxy
- The User Job is executed with its owner's proxy used to access SE, catalogs, etc

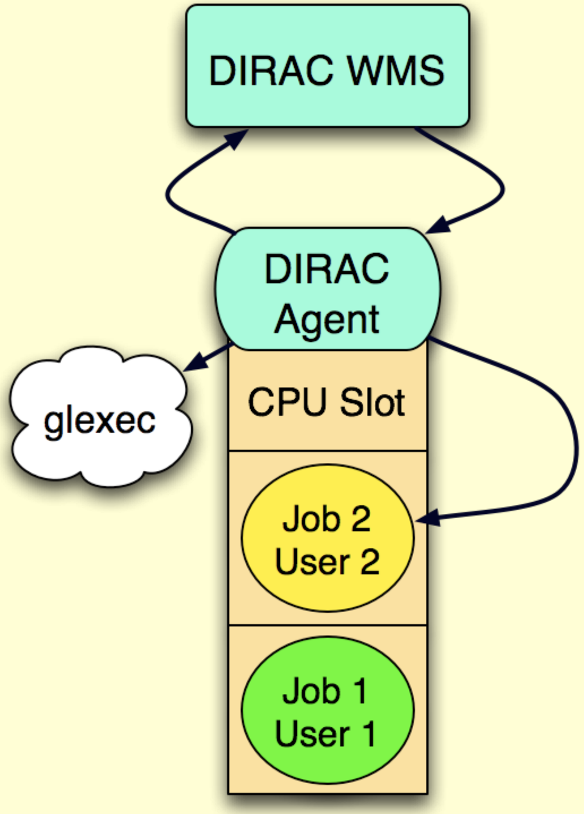


User Job efficiency

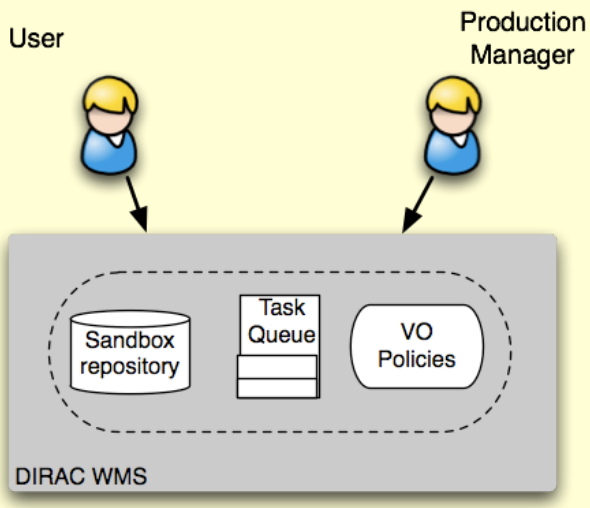
- Improved visible reliability due to pilot agents
  - ~96% efficiency for DIRAC jobs vs 70-90% efficiency for the WLCG jobs
- If some resources are failing, it is just seen as a reduced pool of resources for the users
- An excess of Pilot Jobs over User Jobs just to cover inefficiencies of Computing Resources or Grid middleware
  - it is normal that computing resources are failing but
  - it is not normal that users are suffering from that

Workload optimization

- Pilot Agents work in an optimized 'Filling Mode'
  - Multiple jobs can run in the same CPU slot
  - Significant performance gains for short, high priority tasks
  - Also reduces load on LCG since fewer pilots are submitted
  - Needs reliable tools to estimate remaining time in the queue
- Considering also agents in a "preemption" mode
  - Low priority task can be preempted by a high priority tasks
    - Low priority, e.g. MC, jobs behave as resource reservation for analysis jobs



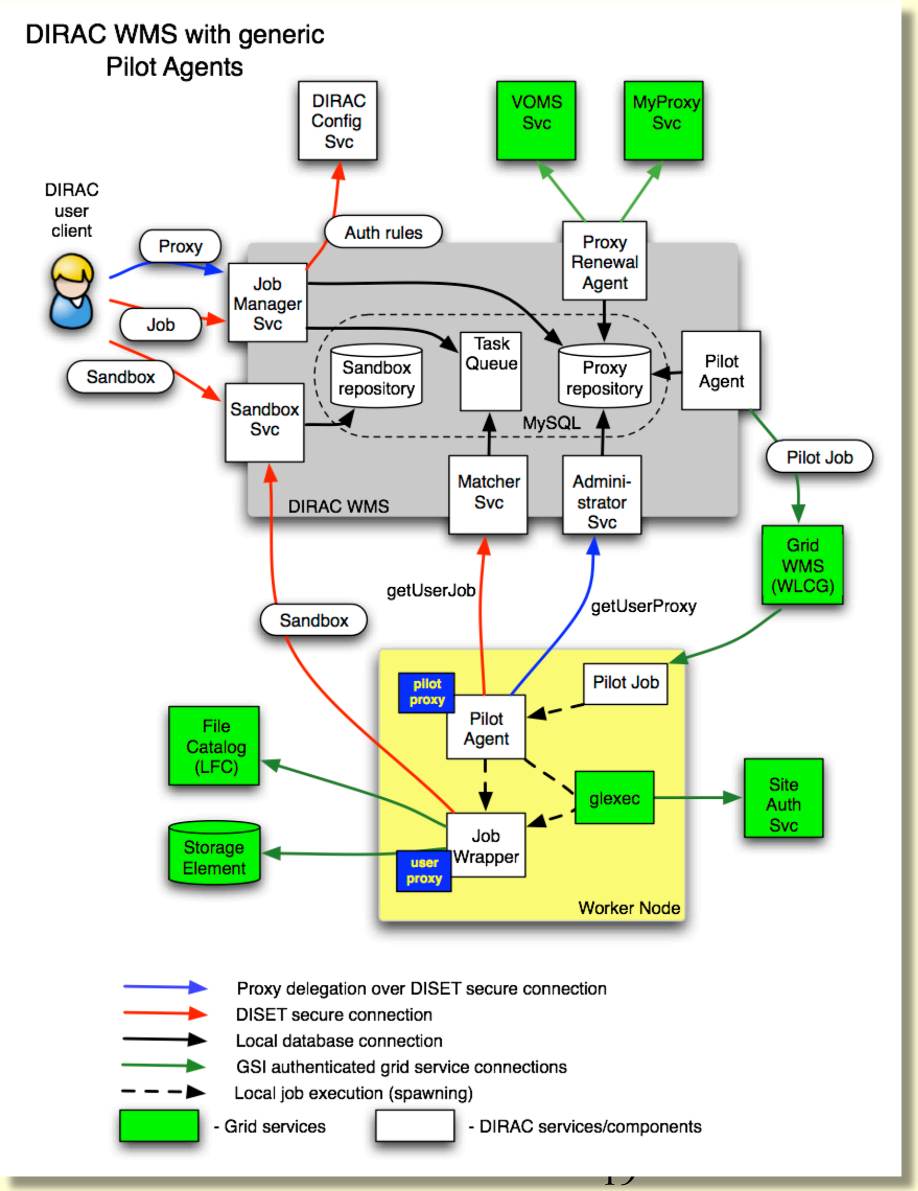
WMS: applying VO policies



- In DIRAC both User and Production jobs are treated by the same WMS
- This allows to apply efficiently policies for the whole VO
  - Assigning Job Priorities for different groups and activities
  - Static group priorities are used currently
  - More powerful scheduler can be plugged in
    - demonstrated with MAUI scheduler
- The VO policies application in the central Task Queue dictates the use of Multiuser Pilot Agents
  - Do not know apriori whose job has the highest priority at the moment of the user job matching
- DIRAC fully supports this mode of operation
  - Multiuser Pilots Jobs submitted with a special “pilot” VOMS role
  - Using glexec on the WNs to track the identity of the payload owner

Security issues of the model

- The VO WMS must be as secure as the basic grid middleware
  - User job submissions using grid security standards: GSI
  - Secure proxy storage in the WMS repository
- The VO WMS takes over the user proxy renewal
  - Limited user proxy
  - Limited number of proxy retrievals per pilot
- Sites still retain the full right to control which individuals are accessing their resources
  - SCAS/glexec facility to authorize user workload execution on the worker node

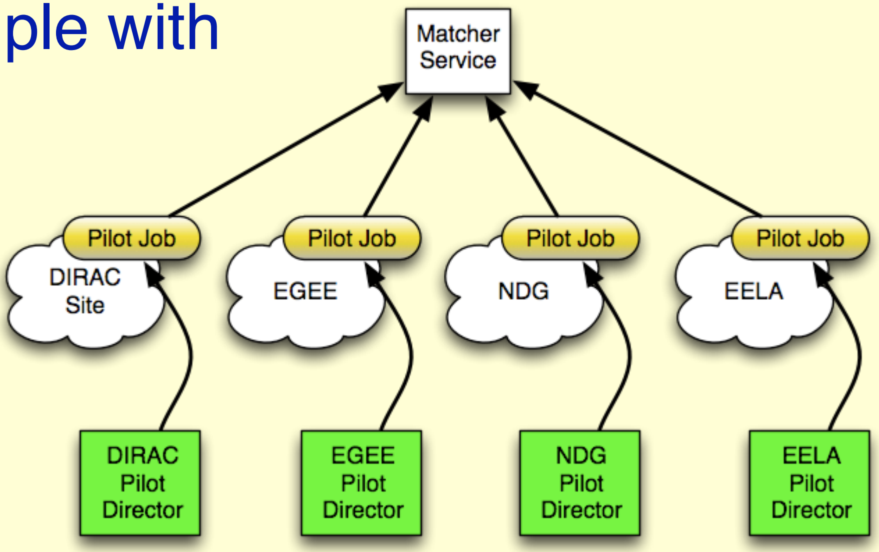


Advantages for site resources providers

- No need for a variety of local batch queues per VO
  - One long queue per VO would be sufficient
  - 24-48 hours queue is a reasonable compromise
    - Site maintenance requirements
  - Reduced number of grid jobs
- No need for specific VO configuration and accounting on sites
  - Priorities for various VO groups, activities
  - User level accounting is optional
- In the whole it can lower the site entry threshold
  - Especially useful for newcomer sites

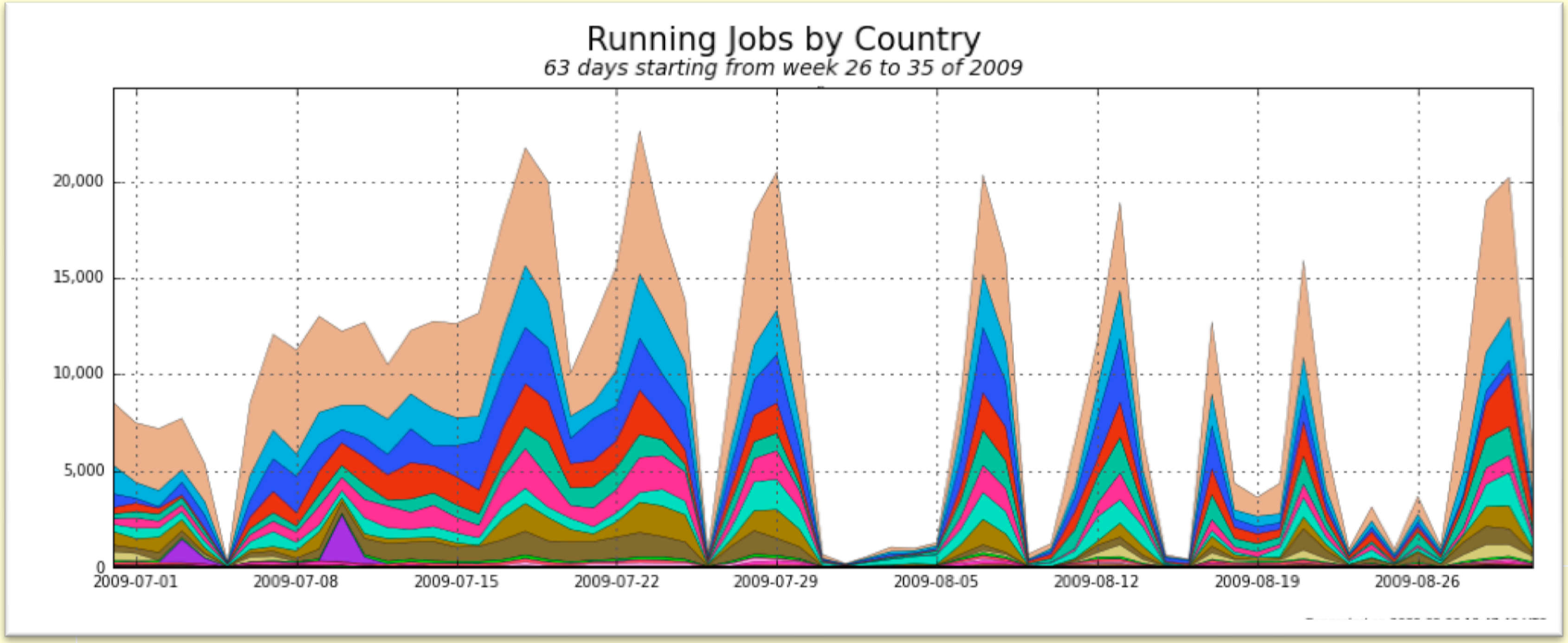
WMS: using heterogeneous resources

- Including resources in different grids and standalone clusters is simple with Pilot Jobs
  - Needs a specialized Pilot Director per resource type
  - Demonstrated with NDG and EELA grid sites
  - Users just see new sites appearing in the job monitoring



- Other resources soon to be included
  - LHCb Online Farm (4K cores, no batch system)
  - Commercial computing clouds (e.g. Amazon EC2)

WMS performance



- DIRAC performance measured in the recent production and FEST'09 runs
  - Up to 25K concurrent jobs in ~120 distinct sites
  - One mid-range central server hosting DIRAC services
  - Further optimizations to increase capacity are possible
    - Hardware, database optimizations, service load balancing, etc

User Interfaces

DIRAC user interfaces

- Easy client installation for various platforms (Linux, MacOS)
  - Includes security components
- JDL notation for job description
  - Simplified with respect to the « standard » JDL
- Command line tools
  - à la gLite UI commands
  - e.g. `dirac-wms-job-submit`
- Extensive Python API for all the tasks
  - Job creation and manipulation, results retrieval
    - Possibility to use complex workflow templates
  - Data operations, catalog inspection
  - Used by GANGA user front-end

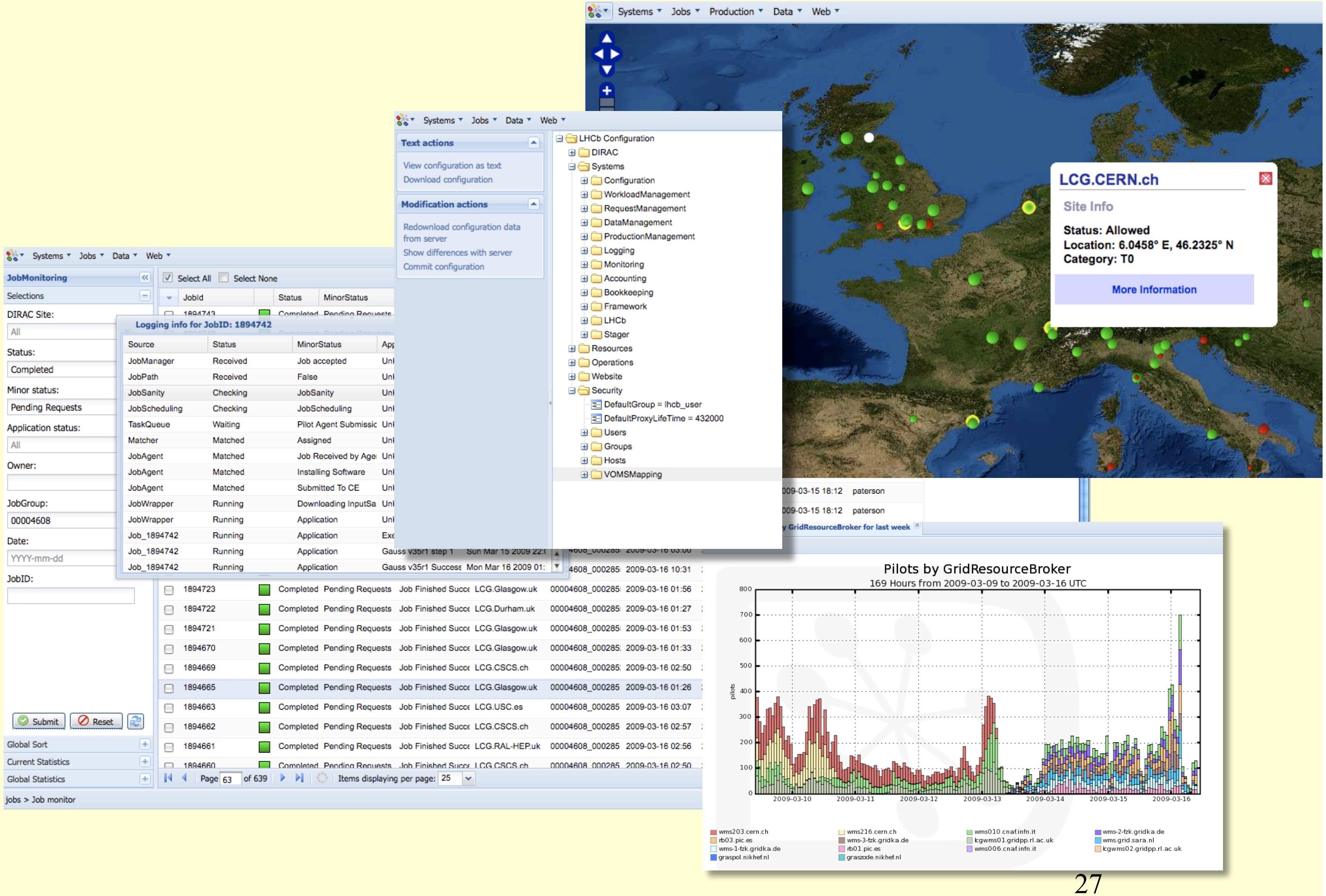
Example job submission

```
from DIRAC.Interfaces.API.Dirac import Dirac
from Extensions.LHCb.API.LHCbJob import LHCbJob
…
myJob = LHCbJob()
myJob.setCPUTime(50000)
myJob.setSystemConfig('slc4_ia32_gcc34')
myJob.setApplication('Brunel','v32r3p1','RealDataDst200Evts.opts','LogFileName.log')
myJob.setName('DIRAC3-Job')
myJob.setInputData(['/lhcb/data/CCRC08/RAW/LHCb/CCRC/420157/420157_0000098813.raw'])
#myJob.setDestination('LCG.CERN.ch')
dirac = Dirac()
jobID = dirac.submit(myJob)
...
dirac.status(<JOBID>)
dirac.parameters(<JOBID>)
dirac.loggingInfo(<JOBID>)
...
dirac.getOutputSandbox(<JOBID>)
```

DIRAC: Secure Web Portal

- Web portal with intuitive desktop application like interface
  - Ajax, Pylons, ExtJS Javascript library
- Monitoring and control of all activities
  - User job monitoring and manipulation
  - Data production controls
  - DIRAC Systems configuration
- Secure access
  - Standard grid certificates
  - Fine grained authorization rules

Web Portal: example interfaces



LCG.CERN.ch

Site Info

Status: Allowed
Location: 6.0458° E, 46.2325° N
Category: T0

More Information

Text actions

View configuration as text
Download configuration

Modification actions

Redownload configuration data from server
Show differences with server
Commit configuration

```
⊟ LHCb Configuration
  ⊞ DIRAC
  ⊟ Systems
    ⊞ Configuration
    ⊞ WorkloadManagement
    ⊞ RequestManagement
    ⊞ DataManagement
    ⊞ ProductionManagement
    ⊞ Logging
    ⊞ Monitoring
    ⊞ Accounting
    ⊞ Bookkeeping
    ⊞ Framework
    ⊞ LHCb
    ⊞ Stager
  ⊞ Resources
  ⊞ Operations
  ⊞ Website
  ⊟ Security
    DefaultGroup = lhcb_user
    DefaultProxyLifeTime = 432000
    ⊞ Users
    ⊞ Groups
    ⊞ Hosts
    ⊞ VOMSMapping
```

JobMonitoring

Selections

DIRAC Site: All

Status: Completed

Minor status: Pending Requests

Application status: All

Owner:

JobGroup: 00004608

Date: YYYY-mm-dd

JobID:

Submit | Reset

Global Sort

Current Statistics

Global Statistics

jobs > Job monitor

☑ Select All ☐ Select None

Logging info for JobID: 1894742

Source	Status	MinorStatus	App	
JobManager	Received	Job accepted	Unl	
JobPath	Received	False	Unl	
JobSanity	Checking	JobSanity	Unl	
JobScheduling	Checking	JobScheduling	Unl	
TaskQueue	Waiting	Pilot Agent Submissic	Unl	
Matcher	Matched	Assigned	Unl	
JobAgent	Matched	Job Received by Age	Unl	
JobAgent	Matched	Installing Software	Unl	
JobAgent	Matched	Submitted To CE	Unl	
JobWrapper	Running	Downloading InputSa	Unl	
JobWrapper	Running	Application	Unl	
Job_1894742	Running	Application	Exe	
Job_1894742	Running	Application	Gauss v35r1 step 1	Sun Mar 15 2009 22:(
Job_1894742	Running	Application	Gauss v35r1 Success	Mon Mar 16 2009 01:

	JobId		Status	MinorStatus				
☐	1894743	🟩	Completed	Pending Requests				
							4608_000285	2009-03-16 03:00
							4608_000285	2009-03-16 10:31
☐	1894723	🟩	Completed	Pending Requests	Job Finished Succ	LCG.Glasgow.uk	00004608_000285	2009-03-16 01:56
☐	1894722	🟩	Completed	Pending Requests	Job Finished Succ	LCG.Durham.uk	00004608_000285	2009-03-16 01:27
☐	1894721	🟩	Completed	Pending Requests	Job Finished Succ	LCG.Glasgow.uk	00004608_000285	2009-03-16 01:53
☐	1894670	🟩	Completed	Pending Requests	Job Finished Succ	LCG.Glasgow.uk	00004608_000285	2009-03-16 01:33
☐	1894669	🟩	Completed	Pending Requests	Job Finished Succ	LCG.CSCS.ch	00004608_000285	2009-03-16 02:50
☐	1894665	🟩	Completed	Pending Requests	Job Finished Succ	LCG.Glasgow.uk	00004608_000285	2009-03-16 01:26
☐	1894663	🟩	Completed	Pending Requests	Job Finished Succ	LCG.USC.es	00004608_000285	2009-03-16 03:07
☐	1894662	🟩	Completed	Pending Requests	Job Finished Succ	LCG.CSCS.ch	00004608_000285	2009-03-16 02:57
☐	1894661	🟩	Completed	Pending Requests	Job Finished Succ	LCG.RAL-HEP.uk	00004608_000285	2009-03-16 02:56
☐	1894660	🟩	Completed	Pending Requests	Job Finished Succ	LCG.CSCS.ch	00004608_000285	2009-03-16 02:50

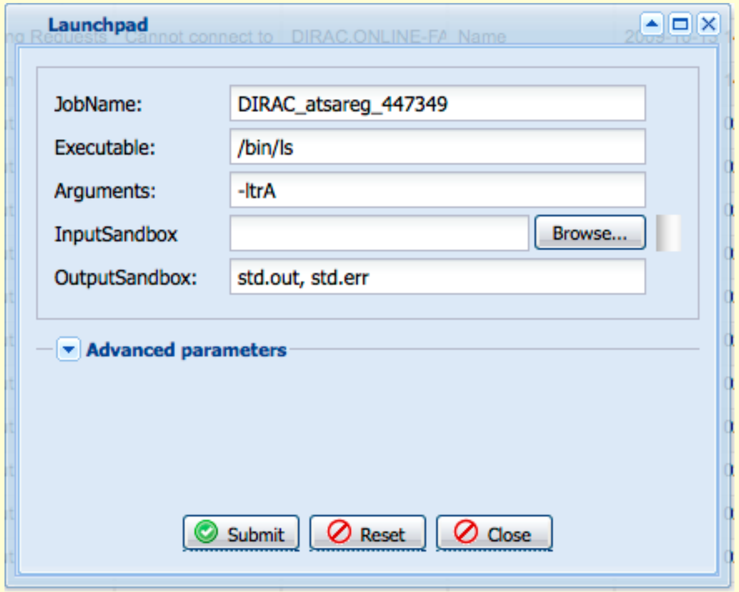
Page 63 of 639 | Items displaying per page: 25

009-03-15 18:12	paterson
009-03-15 18:12	paterson

y GridResourceBroker for last week

Pilots by GridResourceBroker
169 Hours from 2009-03-09 to 2009-03-16 UTC

Web Portal: user tasks



JobName: DIRAC_atsareg_447349

Executable: /bin/ls

Arguments: -ltrA

InputSandbox

OutputSandbox: std.out, std.err

- Data discovery, job monitoring
- Job submission through the Web Portal
  - Full GSI security
  - Sandboxes uploading and downloading
    - Difficult for bulky data files though
  - Generic Job Launchpad panel exists in the basic DIRAC Web Portal
    - Can be useful for newcomers and occasional users
- Specific application Web Portals can be derived
  - Community Application Servers
    - All the grid computational tasks steered on the web
  - VO “formation” DIRAC instance to be deployed at CC/IN2P3

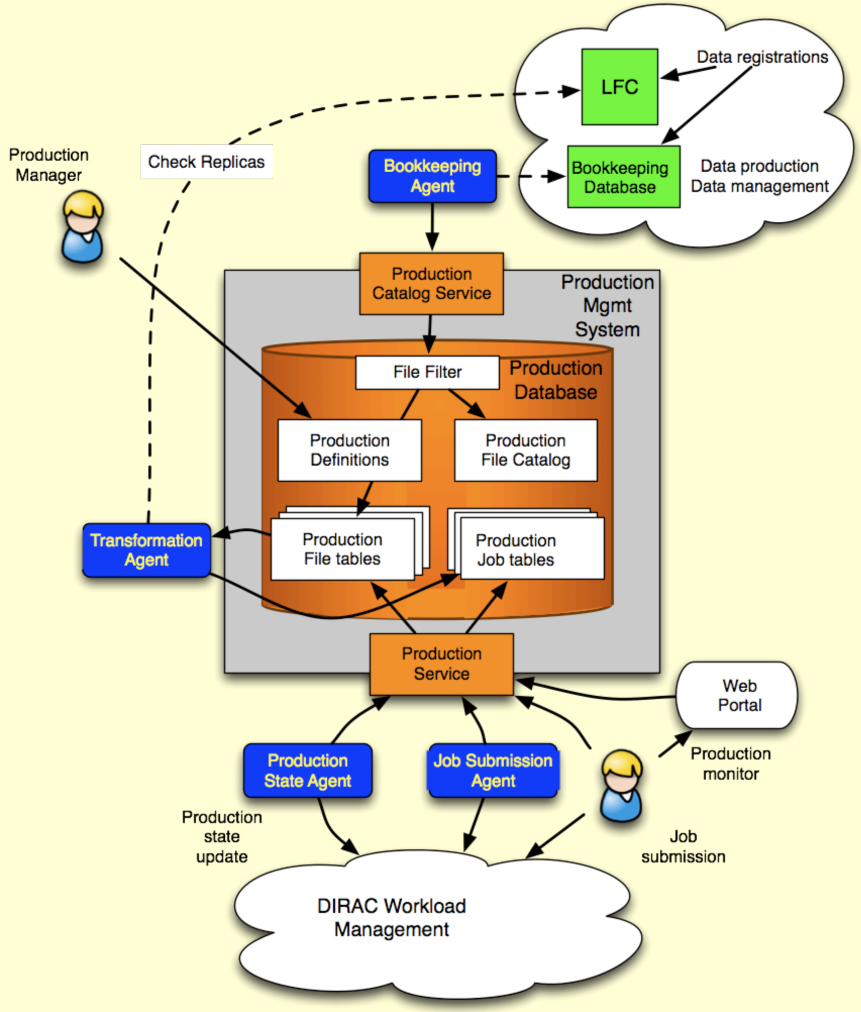
LHCb and other extensions

DIRAC LHCb extensions

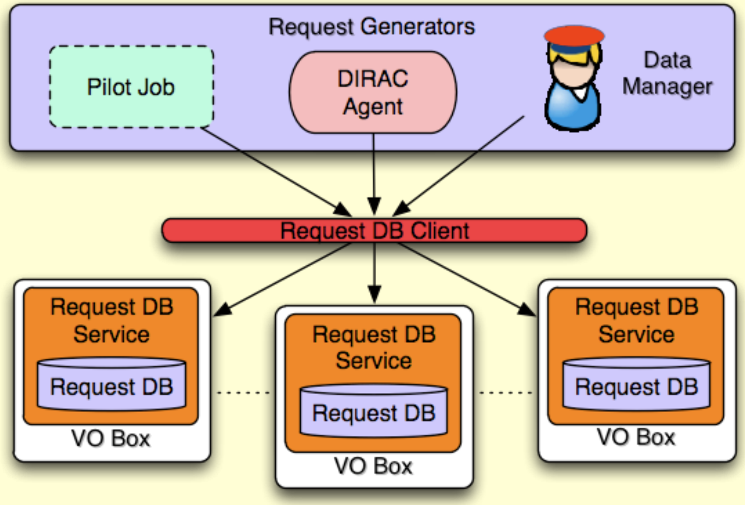
- High level LHCb systems are built in the same DIRAC framework
  - Collaborating services and agents
  - Web based monitoring and controls
  - Detailed authorization rules

Production Management System

- Production Management built on top of the DIRAC WMS and DMS
  - Data requests formulated by users are processed and monitored using Web based tools
  - Automatic data reconstruction jobs creation and submission according to predefined scenarios
  - Interfaced to the LHCb Bookkeeping Database
  - Built using the DISET framework



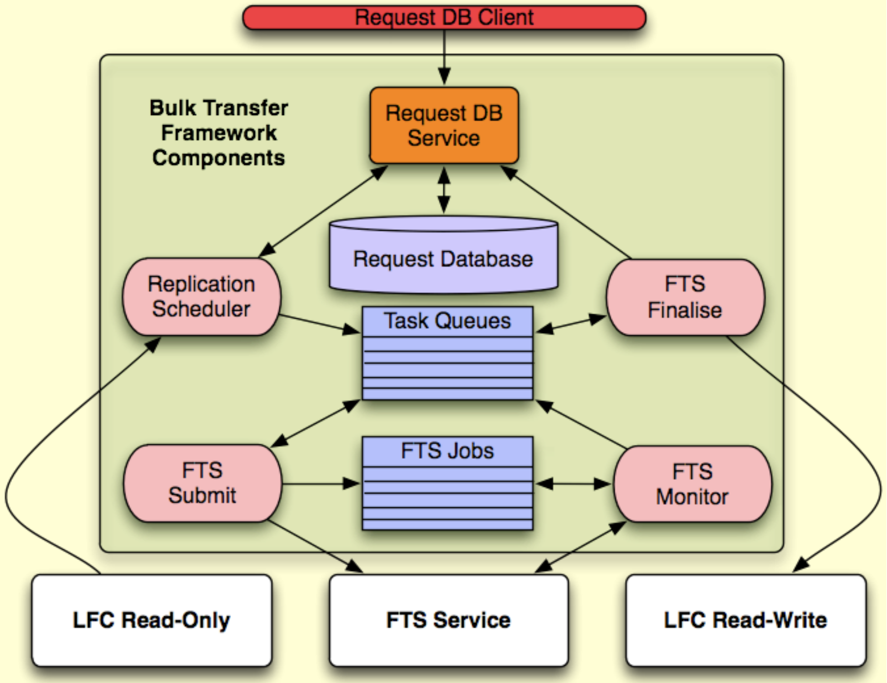
Request Management system



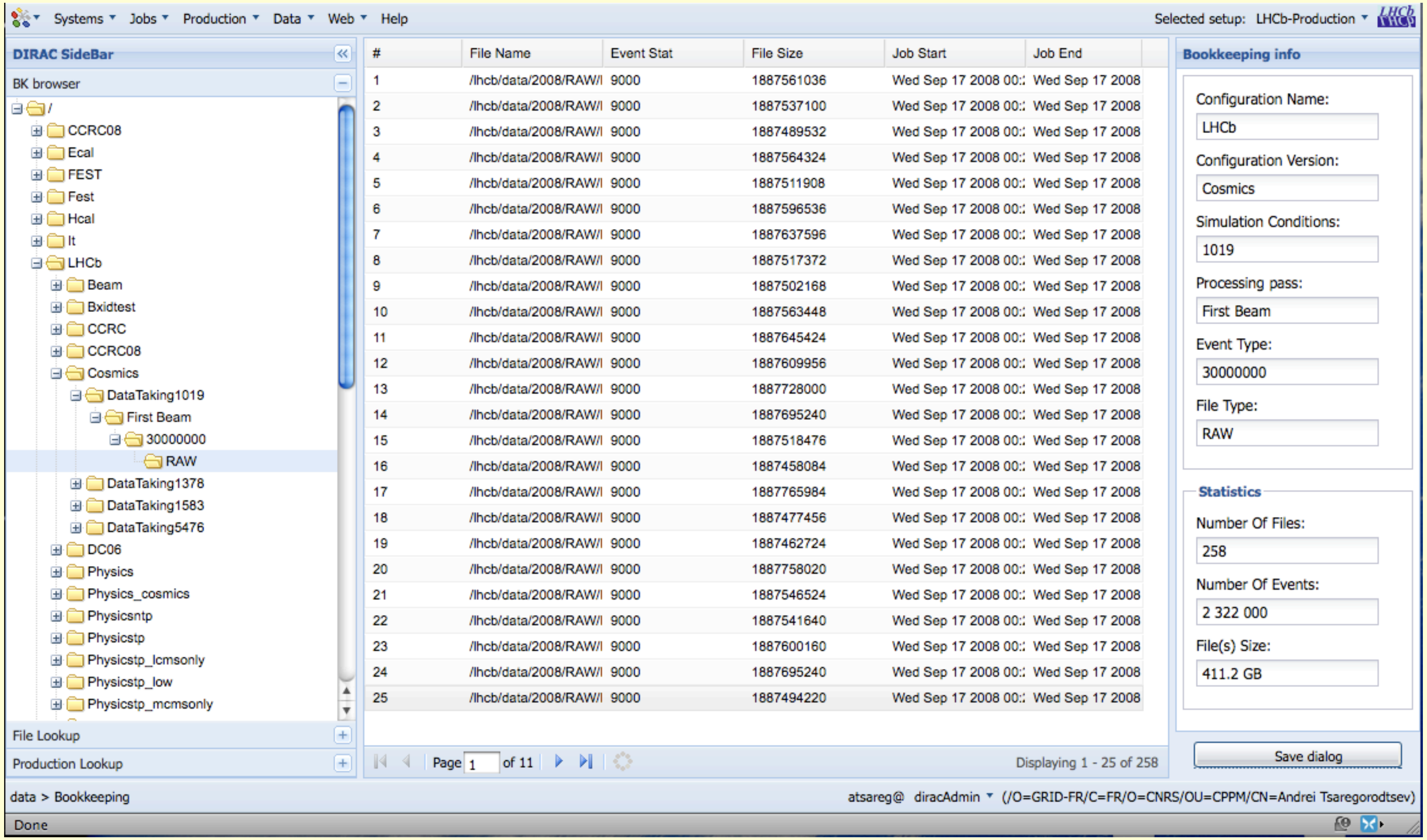
- A Request Management System (RMS) to accept and execute asynchronously any kind of operation that can fail
  - Data upload and registration
  - Job status and parameter reports
- Request are collected by RMS instances on VO-boxes at 7 Tier-1 sites
  - Extra redundancy in VO-box availability
- Requests are forwarded to the central Request Database
  - For keeping track of the pending requests
  - For efficient bulk request execution

Data Management System

- All the Data Distribution operations
  - Pit to CERN transfers
  - T0-T1 transfers
  - T1-T1 transfers
- Based on the Request and Production Management Systems
  - Automatic transfer scheduling
  - Full monitoring of ongoing operations
- Using FTS for bulk data transfers
  - Full failure recovery
- Comprehensive checks of data integrity in SEs and File Catalogs



LHCb Web: Bookkeeping page



#	File Name	Event Stat	File Size	Job Start	Job End
1	/lhcb/data/2008/RAW/l	9000	1887561036	Wed Sep 17 2008 00::	Wed Sep 17 2008
2	/lhcb/data/2008/RAW/l	9000	1887537100	Wed Sep 17 2008 00::	Wed Sep 17 2008
3	/lhcb/data/2008/RAW/l	9000	1887489532	Wed Sep 17 2008 00::	Wed Sep 17 2008
4	/lhcb/data/2008/RAW/l	9000	1887564324	Wed Sep 17 2008 00::	Wed Sep 17 2008
5	/lhcb/data/2008/RAW/l	9000	1887511908	Wed Sep 17 2008 00::	Wed Sep 17 2008
6	/lhcb/data/2008/RAW/l	9000	1887596536	Wed Sep 17 2008 00::	Wed Sep 17 2008
7	/lhcb/data/2008/RAW/l	9000	1887637596	Wed Sep 17 2008 00::	Wed Sep 17 2008
8	/lhcb/data/2008/RAW/l	9000	1887517372	Wed Sep 17 2008 00::	Wed Sep 17 2008
9	/lhcb/data/2008/RAW/l	9000	1887502168	Wed Sep 17 2008 00::	Wed Sep 17 2008
10	/lhcb/data/2008/RAW/l	9000	1887563448	Wed Sep 17 2008 00::	Wed Sep 17 2008
11	/lhcb/data/2008/RAW/l	9000	1887645424	Wed Sep 17 2008 00::	Wed Sep 17 2008
12	/lhcb/data/2008/RAW/l	9000	1887609956	Wed Sep 17 2008 00::	Wed Sep 17 2008
13	/lhcb/data/2008/RAW/l	9000	1887728000	Wed Sep 17 2008 00::	Wed Sep 17 2008
14	/lhcb/data/2008/RAW/l	9000	1887695240	Wed Sep 17 2008 00::	Wed Sep 17 2008
15	/lhcb/data/2008/RAW/l	9000	1887518476	Wed Sep 17 2008 00::	Wed Sep 17 2008
16	/lhcb/data/2008/RAW/l	9000	1887458084	Wed Sep 17 2008 00::	Wed Sep 17 2008
17	/lhcb/data/2008/RAW/l	9000	1887765984	Wed Sep 17 2008 00::	Wed Sep 17 2008
18	/lhcb/data/2008/RAW/l	9000	1887477456	Wed Sep 17 2008 00::	Wed Sep 17 2008
19	/lhcb/data/2008/RAW/l	9000	1887462724	Wed Sep 17 2008 00::	Wed Sep 17 2008
20	/lhcb/data/2008/RAW/l	9000	1887758020	Wed Sep 17 2008 00::	Wed Sep 17 2008
21	/lhcb/data/2008/RAW/l	9000	1887546524	Wed Sep 17 2008 00::	Wed Sep 17 2008
22	/lhcb/data/2008/RAW/l	9000	1887541640	Wed Sep 17 2008 00::	Wed Sep 17 2008
23	/lhcb/data/2008/RAW/l	9000	1887600160	Wed Sep 17 2008 00::	Wed Sep 17 2008
24	/lhcb/data/2008/RAW/l	9000	1887695240	Wed Sep 17 2008 00::	Wed Sep 17 2008
25	/lhcb/data/2008/RAW/l	9000	1887494220	Wed Sep 17 2008 00::	Wed Sep 17 2008

Bookkeeping info

Configuration Name: LHCb

Configuration Version: Cosmics

Simulation Conditions: 1019

Processing pass: First Beam

Event Type: 30000000

File Type: RAW

Statistics

Number Of Files: 258

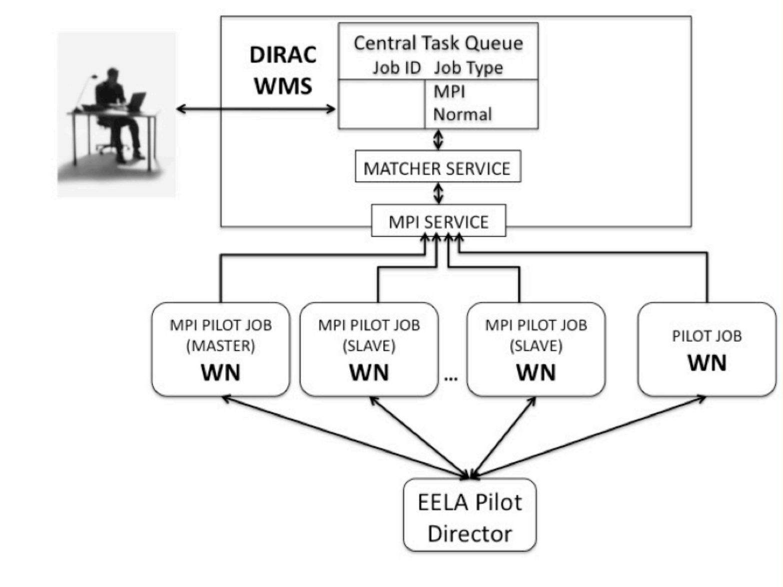
Number Of Events: 2 322 000

File(s) Size: 411.2 GB

- Interface to the LHCb Metadata Catalog
  - Part of the LHCb DIRAC Web Portal

Support for MPI Jobs

- MPI Service developed for applications in the EELA Grid
  - Astrophysics, BioMed, Seismology applications
  - No special MPI support on sites
    - MPI software installed by Pilot Jobs
  - MPI ring usage optimization
    - Ring reuse for multiple jobs
      - ➡ Lower load on the gLite WMS
    - Variable ring sizes for different jobs



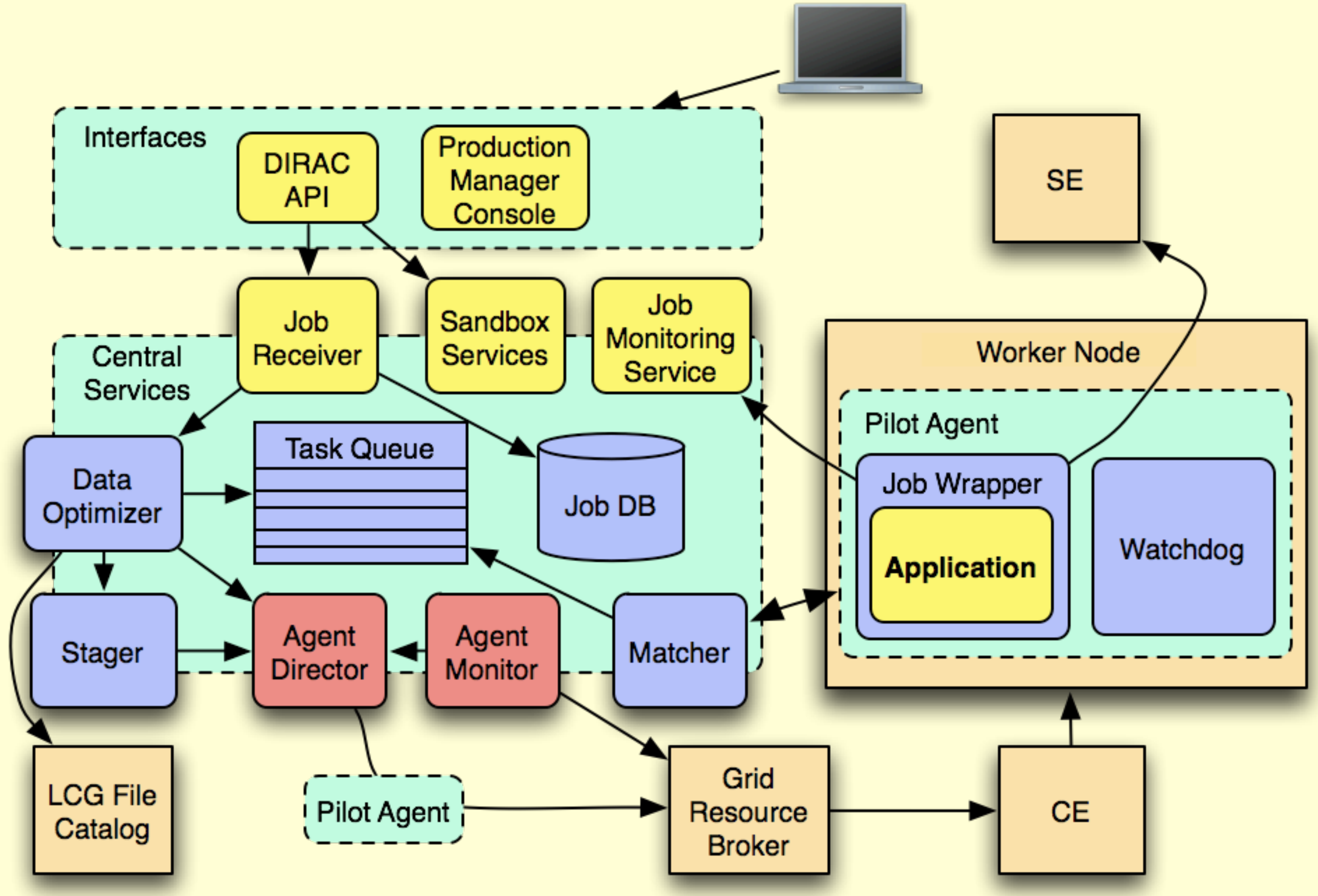
Conclusions

- DIRAC project provides a secure framework for building distributed computing (grid) system
- The WMS with Pilot Jobs addresses (if not solves) multiple problems that large VOs are facing:
  - Heterogeneity of computing resources
  - VO policies, task prioritization
  - Resources and middleware inefficiencies
- The DIRAC Framework can be used to build application specific services and portals
  - Complex LHCb Production Management systems
  - Examples exist also outside the HEP domain

http://dirac.cern.ch

Backup slides

DIRAC WMS components

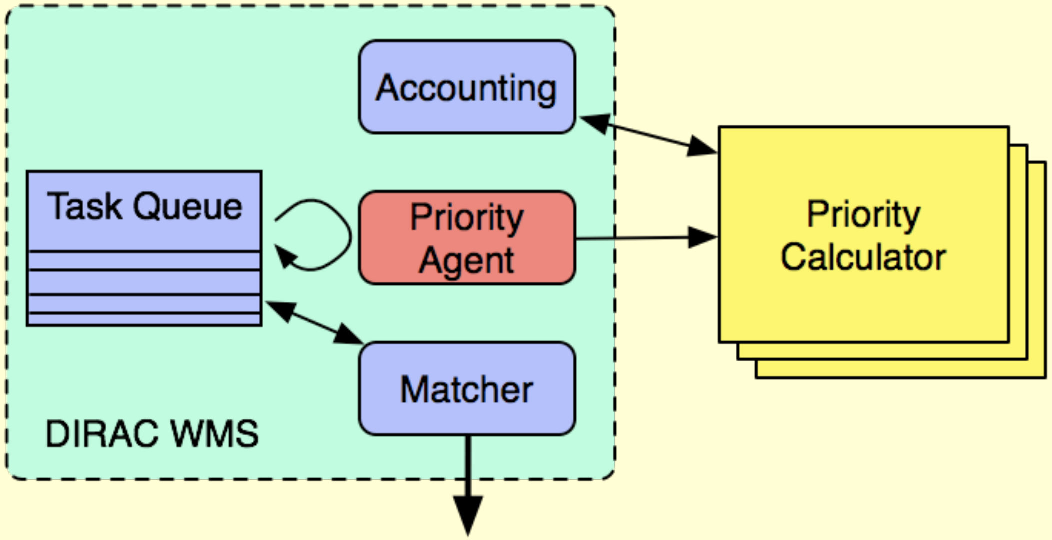


VO policies application: two ways

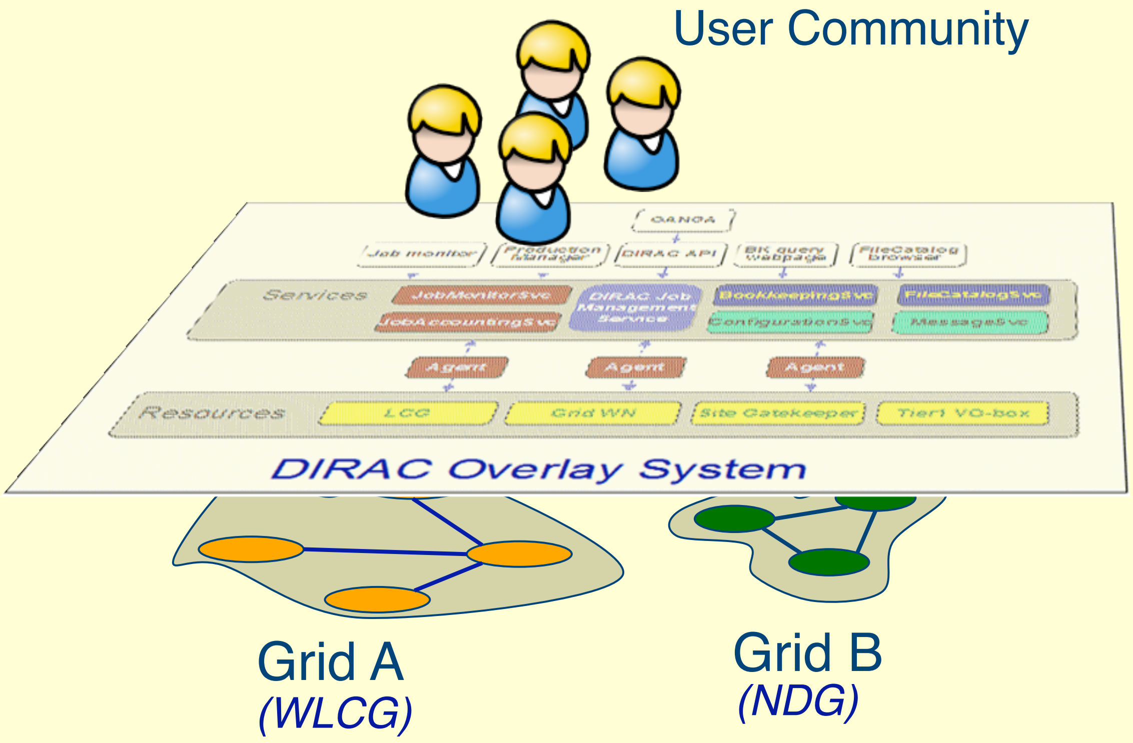
- Define VO policies on each of participating sites
  - Overly complicated to define and to maintain on hundreds of sites
    - Failed so far to provide efficient tools to help this task
  - Imprecise due to latencies in local queues
- Apply VO policies in the central Task Queue
  - Easy to maintain in just one place
  - Precise due to late scheduling
    - Pilot is picking up the highest priority job from the central Task Queue for immediate execution
  - Needs Multiuser Pilot Jobs
    - Pilot Job capable of executing any user’s job

Job Prioritization and VO Policies

- The Matcher service assigns jobs to the requirements presented by Agents that have captured resources
  - Highest priority job dispatched first
- Priority Calculator
  - Static user and group priorities
  - Standard batch system components, e.g. Maui scheduler
  - Others, e.g. “economy models”



DIRAC overlay network



- DIRAC pilots form an overlay network hiding the variety of underlying resources
  - A way for grid interoperability for a given Community
  - Needs specific Agent Director per resource type
- From the user perspective all the resources are seen as a single large “batch system”