



LHCb glxexec use case

*A. Tsaregorodtsev,
CPPM, Marseille*

Grid Deployment Board meeting, 8 November 2006, CERN

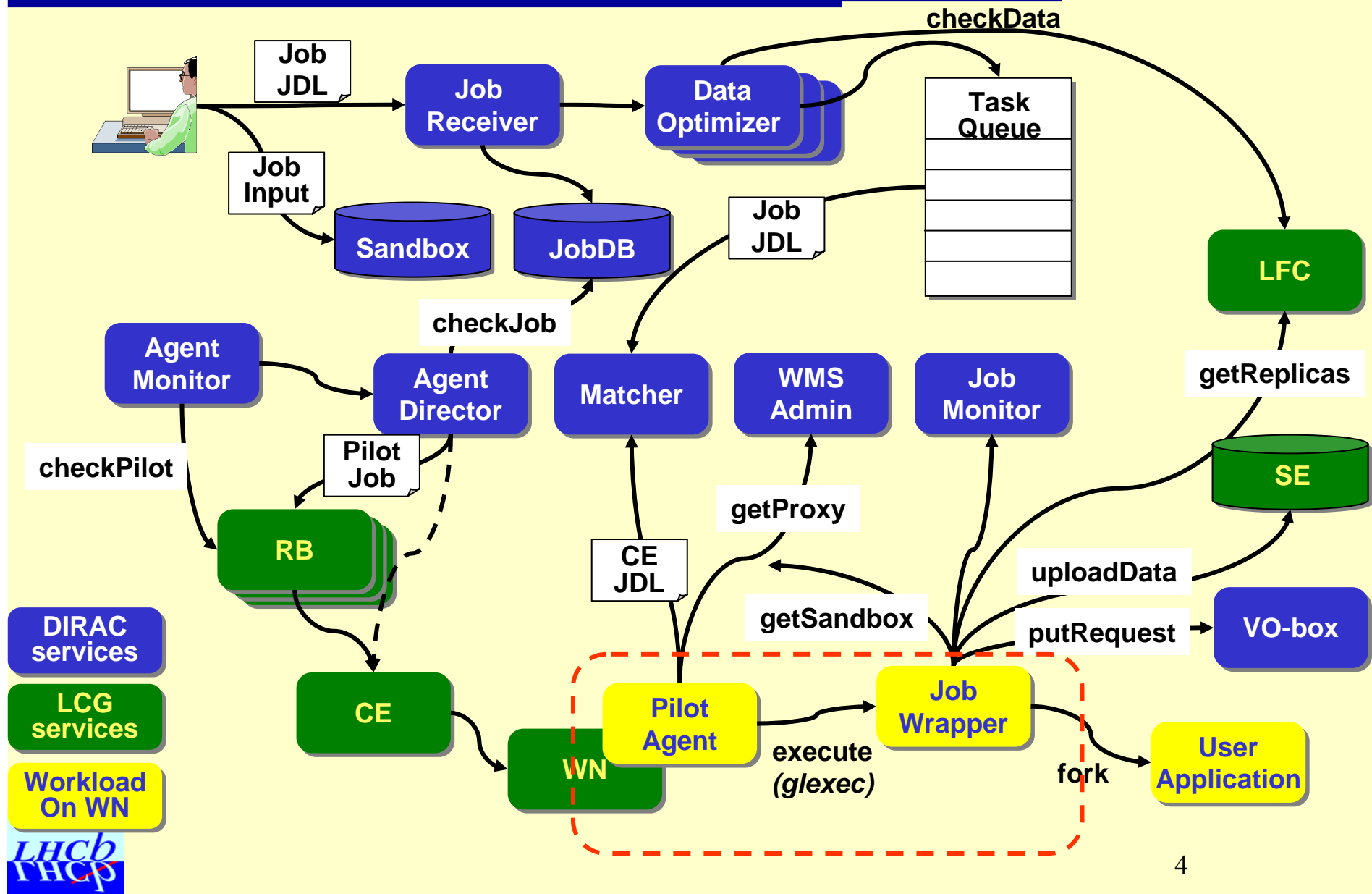
Outline

- ◆ WMS with Pilot Agents
- ◆ Job prioritization problem and solution
- ◆ Use of *glexec* in LHCb
- ◆ Status

Introduction (1)

- ◆ LHCb has its own WMS capable of steering jobs of the LHCb users on different resources
 - ✦ LCG, standalone clusters, PCs
- ◆ The LHCb WMS (DIRAC) uses the Pilot Agent (Job) paradigm
 - ✦ Increased reliability, elimination of the “black holes”
 - ✦ More precise late scheduling
 - ✦ Central VO Task Queue
- ◆ The Pilot Agent paradigm leads naturally to a Job Prioritization schema with generic VO Pilot Agents
 - ✦ Prioritization in the central Task Queue

DIRAC workload management



Job prioritization task

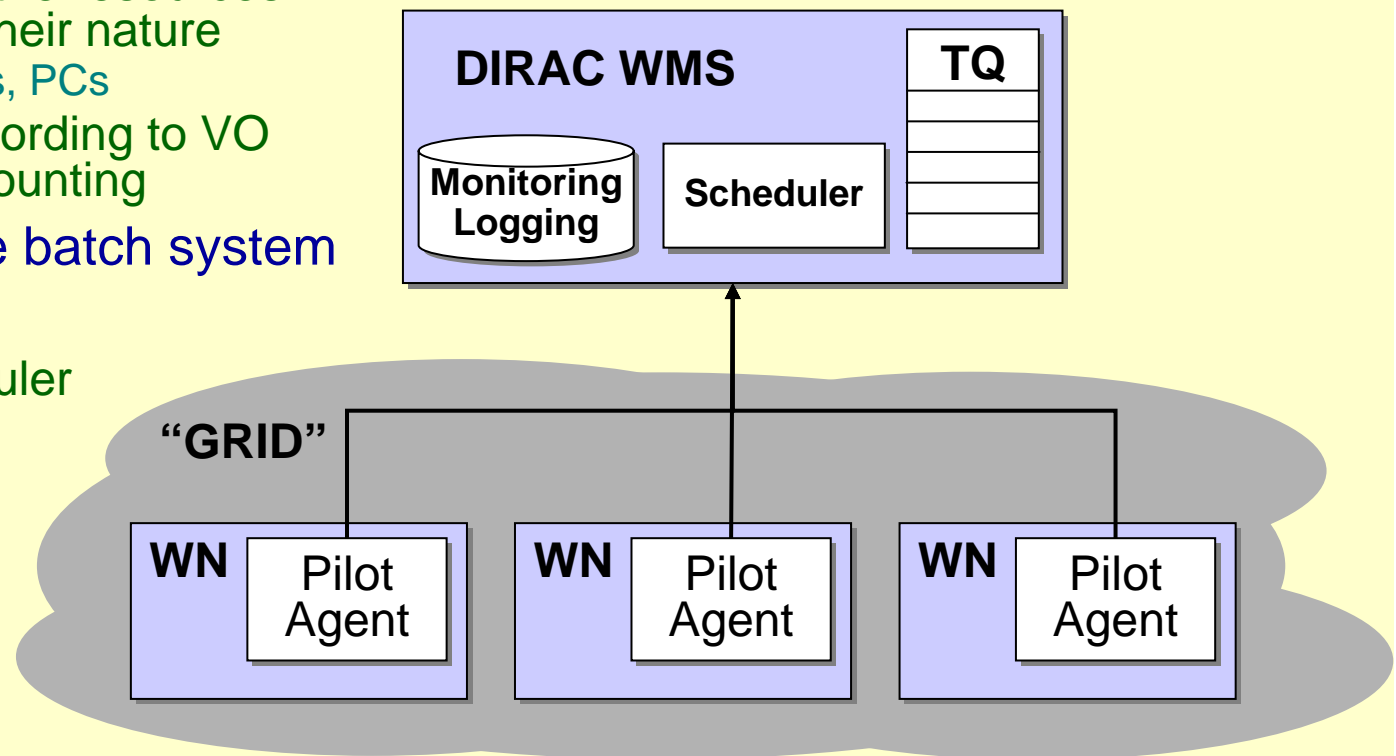
- ◆ LHCb computing activities are varied and each kind of activity has different priority,
 - ✦ Simulation and reconstruction jobs
 - ✦ Production and user analysis jobs
- ◆ LHCb user community is divided into groups of interest and each group will have its well defined share of the LHCb resources
 - ✦ Consumption according to the shares should be ensured
- ◆ Each user activity can consist of different tasks with different importance for the user
 - ✦ User should be able to control the priority of his own tasks

Job Prioritization strategy

- ◆ Submitted jobs are inserted in the Central Task Queue and ordered by their priorities
 - ✦ The priorities are either set by hand or calculated based on the LHCb policies and accounting
 - E.g., using a Maui scheduling engine
- ◆ For each job in the Task Queue a Pilot Agent is sent to LCG
 - ✦ Keeping track of the original job identity now
 - ✦ Generic VO Pilot Agent eventually
- ◆ When the Pilot starts on a WN it asks for a job to the Matcher service
 - ✦ The highest priority suitable job is served
- ◆ “Centralized” prioritization as opposed to the “On-site” prioritization

Community Overlay Network

- ◆ DIRAC Central Services and Pilot Agents form a dynamic distributed system as easy to manage as an ordinary batch system
 - ✦ Uniform view of the resources independent of their nature
 - Grids, clusters, PCs
 - ✦ Prioritization according to VO policies and accounting
- ◆ Possibility to reuse batch system tools
 - ✦ E.g. Maui scheduler



Generic VO pilot agents

- ◆ Central Job prioritization strategy only makes sense if all the LHCb jobs (User and Production) are treated the same way together.
- ◆ Generic VO pilot agents are needed
 - ✦ Each agent should take the highest priority job independently of its ownership
- ◆ Implications
 - ✦ The VO WMS should be as secure as the LCG WMS
 - ✦ User credential delegation
 - ✦ Interaction with the site policy enforcement system
 - ✦ Job owner traceability
- ◆ This is where the **glexec** functionality is necessary

Job submission to DIRAC WMS

- ◆ User jobs are submitted via a JobReceiver service using DSET security mechanism
 - ✦ GSI authentication of the user credentials
 - ✦ User authorization is done based on the DIRAC internal configuration accessible to and managed by the LHCb administrators
 - ✦ The user proxy is passed to DIRAC via an SSL encrypted channel
 - No new private/public key pair generated
 - ✦ The proxies of the users whose jobs are in the system are stored in the MySQL database
 - ✦ LCG proxy cache tools can be incorporated if necessary

Respecting Site policy

- ◆ Generic VO Pilot Agent should not run the jobs of the users in a site black list
- ◆ How to enforce it:
 - ✦ **glexec** utility
 - Consults site policy enforcement box
 - Changes the user identity for running the user application
 - Executes the user application
- ◆ Accounting:
 - ✦ LHCb does not require finer grained accounting than the VO level on the sites
 - Group or user level accounting is done by the LHCb VO
 - ➔ Already exists

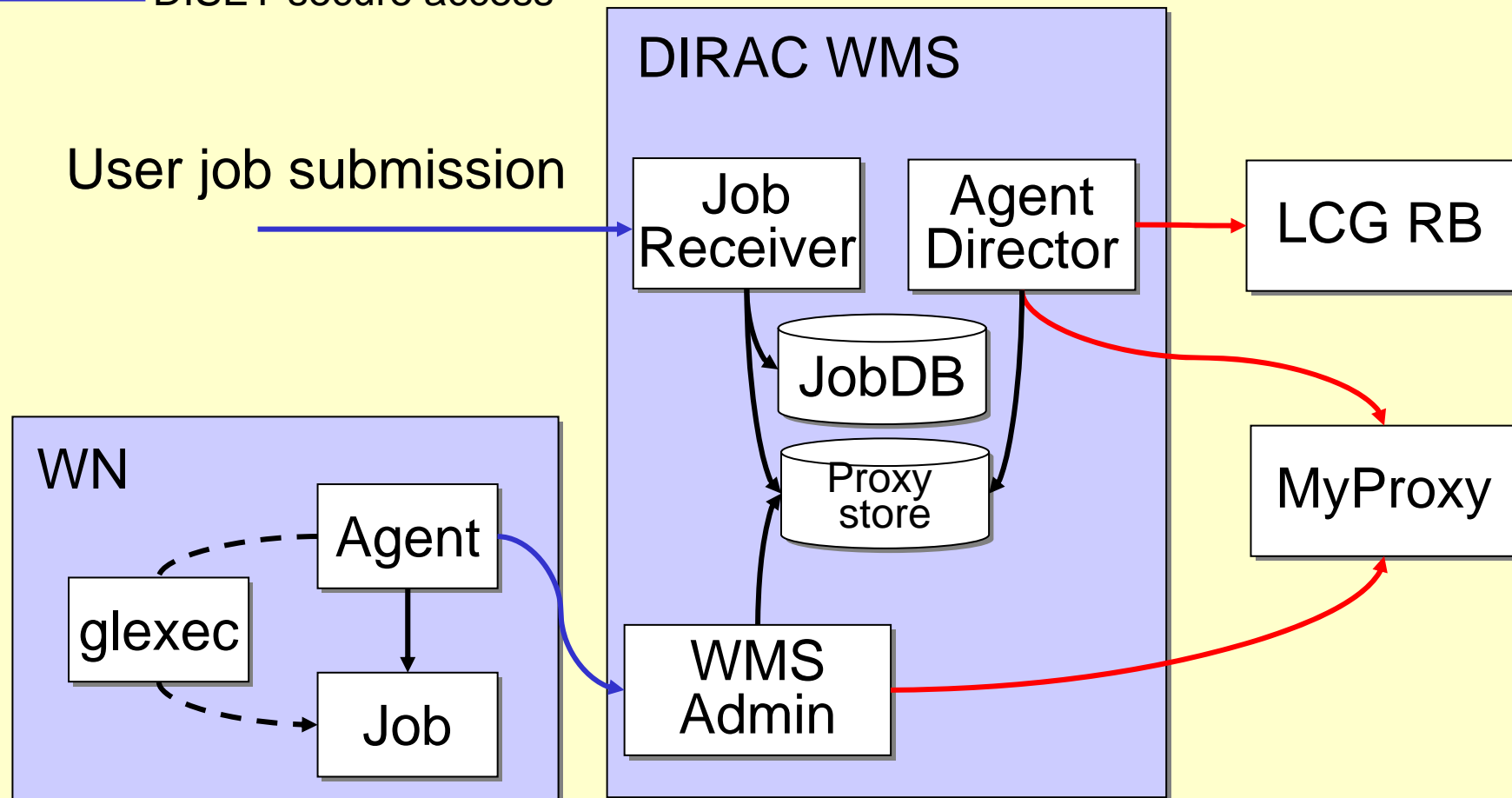
User proxy delegation

- ◆ User grid job performs operations needing user credentials
 - ✦ Access to data, to catalogs, etc
- ◆ DIRAC WMSAdministrator service can serve user proxies to the requests of the LHCb administrator users
 - ✦ DISET secure service
 - ✦ Uses MyProxy server to serve only proxies with a minimally required life time
- ◆ Pilot Agent should acquire the credentials (proxy) of the actual owner of a job
 - ✦ For the moment using the DIRAC tools
 - ✦ Eventually LCG/gLite provided proxy delegation mechanism
 - We would be happy to try it out
- ◆ The user proxy is passed to the **glexec** together with the user workload

User proxy delegation in DIRAC

— LCG secure access

— DISET secure access



Job traceability

- ◆ Job traceability – how to know which workload is being executed on a WN at each moment
- ◆ Possible solutions
 - ✦ Log file on the WN, for example:

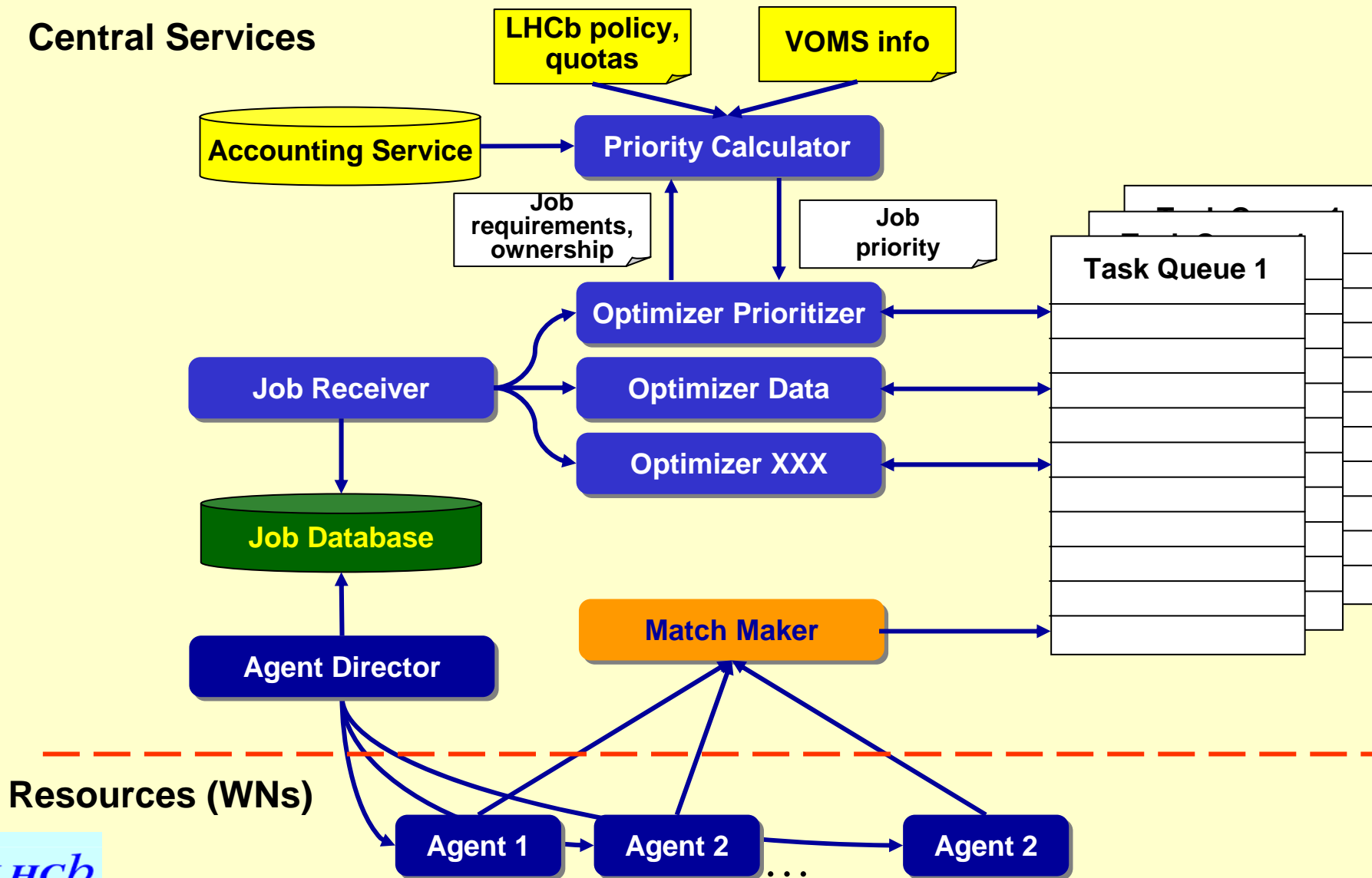
```
2006-10-11 03:05:03: Starting job 00001455_00000744
2006-10-11 03:05:03: Job owner /O=GRID-FR/C=FR/O=CNRS/OU=CPPM/CN=Andrei Tsaregorodtsev
2006-10-11 03:05:03: Job top process id 12345
2006-10-11 08:04:34: Finished job 00001455_00000744
2006-10-11 08:04:34: CPU consumption for 00001455_00000744: 34875.4 s
```

- ✦ Providing a VO service to query this same information
 - The information is kept in the DIRAC Job Monitoring service anyway
- ✦ System auditing tracing problems to the *id* set by **glexec**

Where we are

- ◆ We are practically ready to proceed with the testing of the overall schema and **glexec** in particular
 - ✦ Pilot Agent is generating dynamically job wrapper scripts
 - Wrappers are passed to the LRMS on the DIRAC sites
 - Can be passed to the **glexec** as the executable workload
 - ✦ The LHCb/DIRAC fine grained status and accounting reports are done within the wrapper scripts
 - This will not be deteriorated by the glexec generating a new process group
 - ✦ The user proxy delegation are done by DIRAC tools
- ◆ Job priorities evaluation is in place
 - ✦ Rather simple ones for the moment
 - ✦ More elaborated mechanisms are in the works

DIRAC workload management



Where we are

- ◆ We discussed with CC/IN2P3, Lyon, about the possible tests
 - ✦ Everybody agrees, need to sort out technical details of the *glexec* deployment
 - ✦ More details to be clarified for the sites that do workload optimization based on the job properties
 - For example, CPU bound vs I/O bound jobs

Conclusions

- ◆ LHCb intends to employ “central strategy” for the job prioritization problem
- ◆ This solution necessitates the use of generic Pilot Agents running on LCG nodes and executing arbitrary user jobs
- ◆ **glexec** is the crucial part of it which ensures the job traceability and application of the site policies
- ◆ LHCb is ready to do the tests of the **glexec** functionality in the full job prioritization procedure

Conclusions

- ◆ LHCb intends to employ “central strategy” for the job prioritization problem
- ◆ This solution necessitates the use of generic Pilot Agents running on LCG nodes and executing arbitrary user jobs
- ◆ **glexec** is the crucial part of it which ensures the job traceability and application of the site policies
- ◆ LHCb is ready to do the tests of the **glexec** functionality in the full job prioritization procedure