CREAM Computing Element Status and new developments

Massimo Sgaravatto - INFN Padova
On behalf of the CREAM CE Product Team

EMI is partially funded by the European Commission under Grant Agreement RI-261611

CREAM

- CREAM service: Computing Resource Execution And Management service
- It implements job management functionality at Computing Element (CE) level
  - Allows to submit, cancel, monitor, … jobs
- Can be used
  - Directly by the client
  - By some higher level services
    - *gLite-WMS and Condor support submission to CREAM*
- Part of gLite middleware stack
- Deployed in ~ 235 EGI sites (~ 390 CREAM CE hosts)
- Used by several user communities
  - Replaced LCG-CE in WLCG

Roadmap

Roadmap

- Integration with ARGUS Authorization service
- Support for gLite-CLUSTER
- Initial support for Glue 2.0 publication
- Support for execution on multi-core environments
- Support for JDL OutputData attribute
- Bug fixes

Roadmap

Roadmap

- Support for SL6 (and Debian6 ?)
- Initial EMI-ES support
- Finalization of Glue 2.0 publication
- Support for bulk submission in CREAM command Line
- Bug fixes

Roadmap

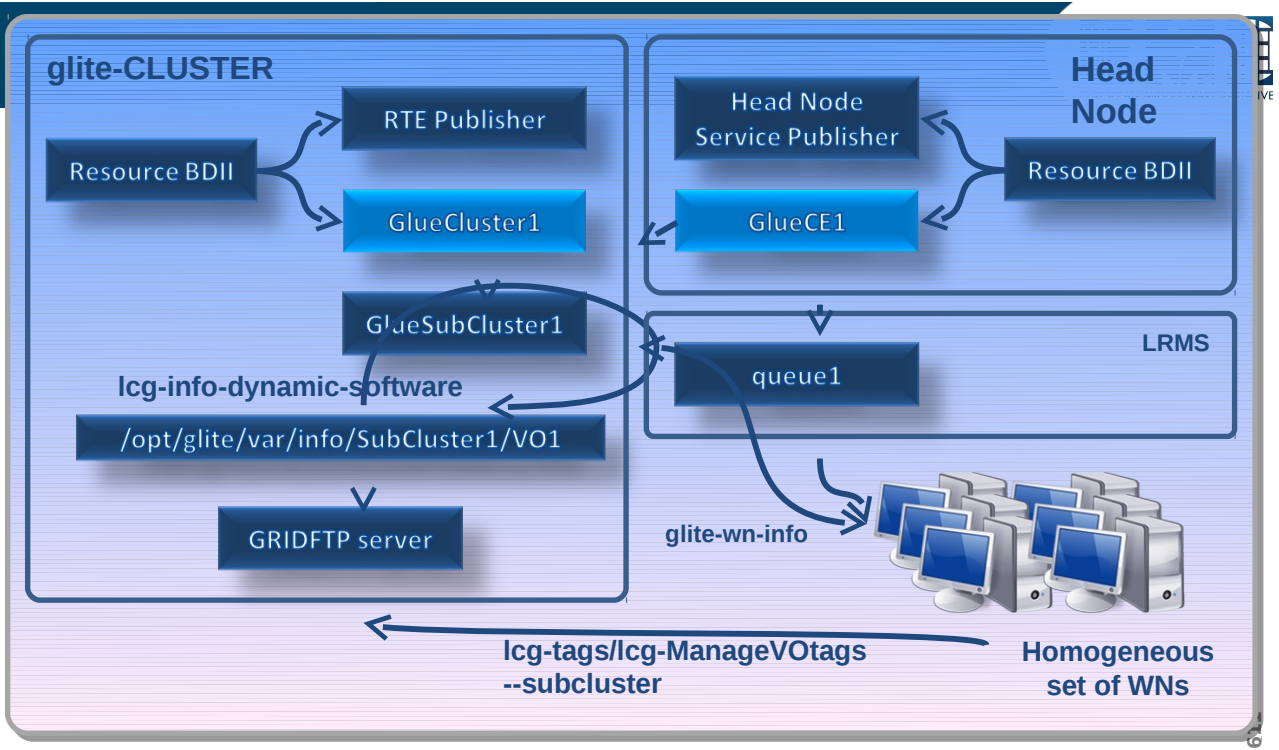
Timelines for release of new developments after EMI-2 not defined yet

Support for SLURM
Finalization of EMI-ES support
CREAM High Availability
Support for job streams
Integration of common authentication library
Bug fixes

gLite-CLUSTER

- gLite-CLUSTER: node type responsible to publish
  - Information concerning physical resources in the site
  - Information about software tags
- “Used” by one or more CREAM CEs available in the site
- It helps the deployment of sites where:
  - There are two or more CREAM CE front-end machines
    - *In this case, without gLite-CLUSTER, each CREAM CE publishes everything in its resource BDII*
      - Same information published multiple times
      - To avoid overcounting the installed capacity, you need to publish zero values for the installed capacity from all but one of the nodes

  and/or
  - There are two or more subclusters, i.e. sets of Worker Nodes each one having homogeneous characteristics
    - *Without using gLite-CLUSTER, yaim configuration tool allows to configure a single cluster referring to a single subcluster*


glite-CLUSTER
RTE Publisher
Resource BDII
GlueCluster1
GlueSubCluster1
lcg-info-dynamic-software
/opt/glite/var/info/SubCluster1/VO1
GRIDFTP server
Head Node
Head Node Service Publisher
Resource BDII
GlueCE1
LRMS
queue1
glite-wn-info
lcg-tags/lcg-ManageVOtags --subcluster
Homogeneous set of WNs

Integration with ARGUS

- Single authorization system

  Responsible to decide if a certain operation for a certain Grid user is authorized

  Also responsible to give the local id mapped to that Grid user

- → Avoid inconsistent authorization decisions

  Scenario which is instead possible without using ARGUS, since in this case several components are involved in authorization in the CREAM CE

  gJAF, LCAS, LCMAPS, gLexec

- → Simplification of overall management
- Use of ARGUS is optional in the CREAM CE

Glue 2 publication

- First support in EMI-1, finalized with EMI-2
- Support in CREAM CE and gLite-CLUSTER
- Static and batch system dynamic information provided
- All the information published in Glue1 is also published in Glue2
- What is published

  ComputingService, ComputingEndpoints and relevant policies, ComputingShares (map to Glue1 VOViews), ComputingManager, Benchmarks, ExecutionEnvironments (map to Glue1 Subclusters) and relevant mapping policies, ApplicationEnvironments, ToStorageServices
- What is not published

  ComputingActivities (jobs), ApplicationHandle

EMI Execution Service (EMI-ES)

- Common interface between ARC, gLite and UNICORE computing services defined in the context of the EMI project
- Common language (Activity Description Language) to describe activities (jobs)
- Architecture:

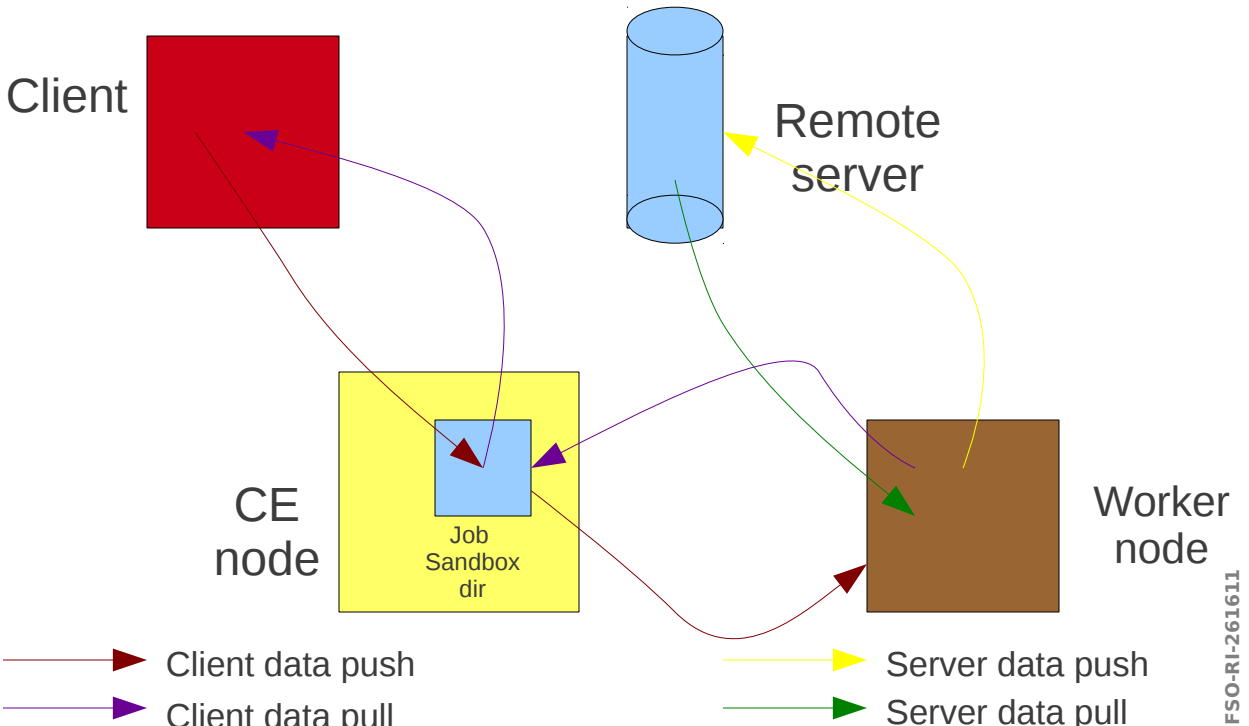
  ActivityFactory, responsible to create activities and manage resource (CE) information

  ActivityManager, responsible to manage activities and activity related information

EMI-ES in CREAM-CE

- In CREAM-CE EMI-ES optionally deployed along with legacy interface starting with EMI-2
- What will be available in EMI-2
  - Job submission of single or multiple activities (jobs)
  - Data staging
    - *Support for client data push and pull, and for server data push and pull*
    - *Support for multiple sources and targets*
    - *Support for multiple delegations*
  - Delegations: creation, renewal, info
  - Job cancel, job pause, job restart, job wipe, notify
  - Job status, job info, list

Data staging



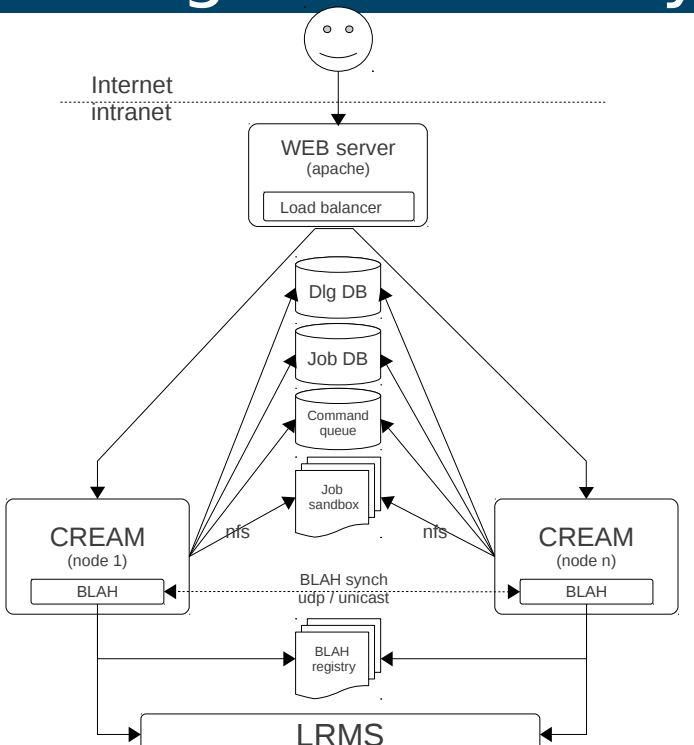
EMI-ES in CREAM-CE (cont.ed)

- What will be available in EMI-2 (cont.ed)
  - Command Line Interface (glite-es-* commands) to use this functionality
  - Deployed in UI along with CREAM CE legacy CLI
  - Being tested with CREAM-ES and also with UNICORE CE ES
- What will be available in an EMI-2 Update
  - ResourceInfo Port-Type
    - *Information about the Computing Element in Glue 2*
  - But this information is already available via the BDII

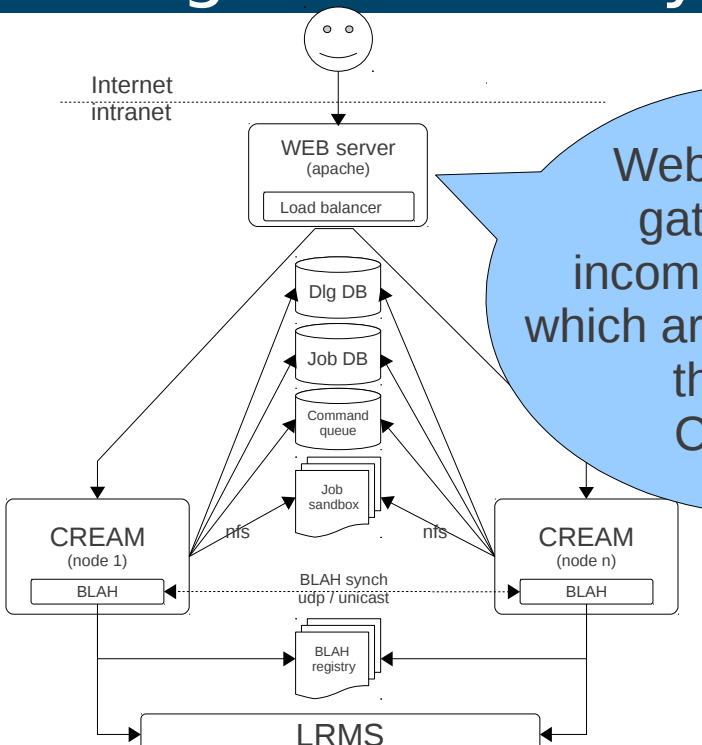
CREAM High Availability

- Pool of CREAM CE machines seen as a unique logical CREAM CE
- To provide high availability and to improve scalability
- Something is already possible now with DNS alias, but there are too many limitations

CREAM High Availability

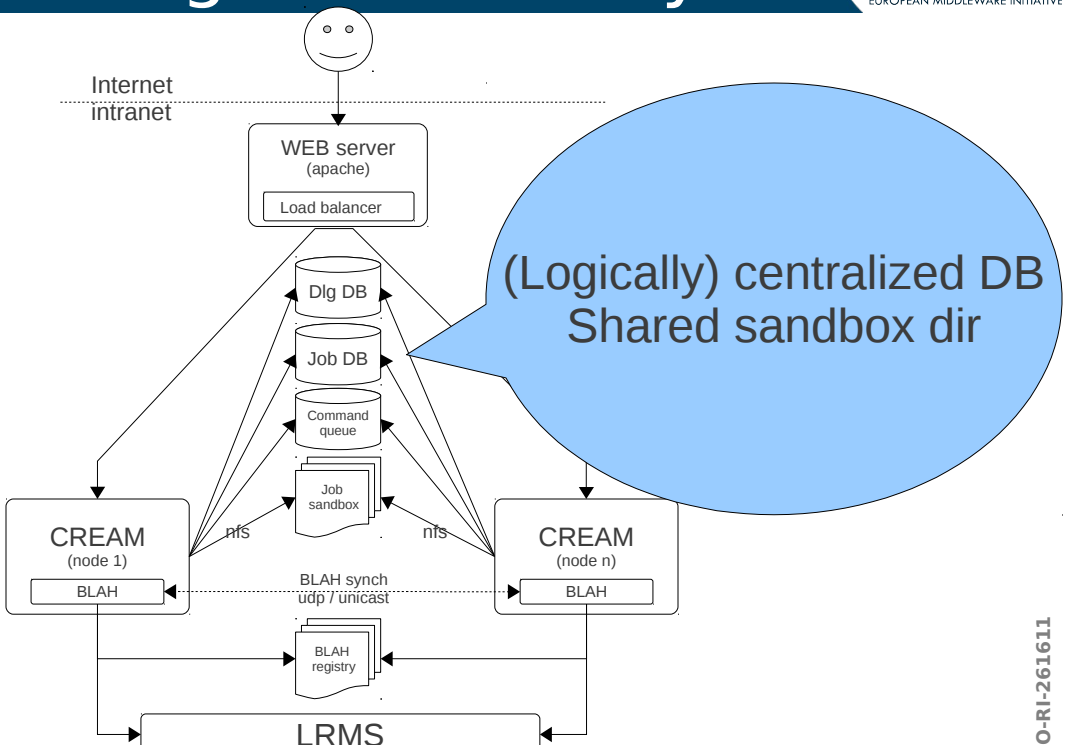


CREAM High Availability

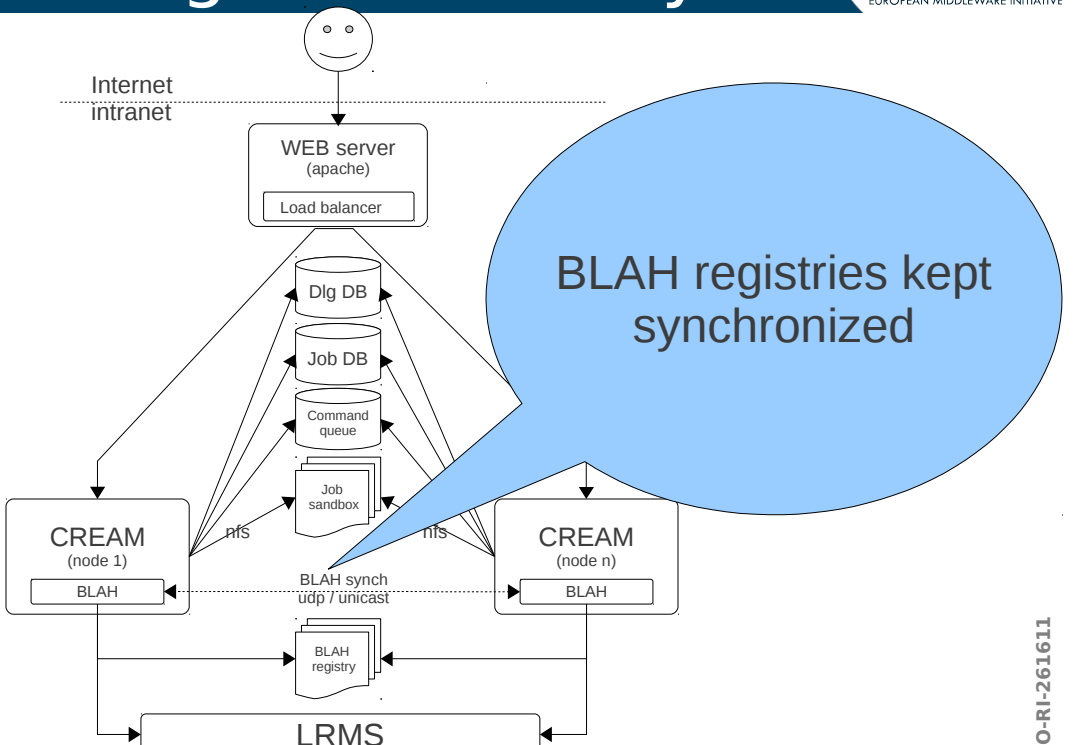


Web server as gateways for incoming requests, which are forwarded to the “best” CREAM

CREAM High Availability



CREAM High Availability



Support for job streams

- Support for constantly keep N jobs queued at a given queue until a given condition is satisfied (e.g. no work to be done)
- Recent request coming from the WLCG Workload Management TEG to optimize submission of pilots
- To discuss with relevant users to better understand what is really needed

Other info

- CREAM site

  https://wiki.italiangrid.it/CREAM

- EMI Execution Service

  https://twiki.cern.ch/twiki/bin/view/EMI/EmiExecutionService

Thank you

EMI is partially funded by the European Commission under Grant Agreement INFSO-RI-261611

Backup slides

Execution on multicore envs

- Possible to specify how many cores are needed, how many nodes, how many cores per host, if all the cores of a node should be allocated
- Support for new JDL attributes
  - SMPGranularity
  - WholeNodes
  - HostNumber
- Initially required by the MPI users but now also other communities are interested
  - E.g. some LHC experiments

Bulk submission

- Possibility to submit multiple independent jobs with a single glite-ce-job-submit command
- Bulk submission functionality in the server side was already implemented