



Enabling Grids for
E-science in Europe

www.eu-egee.org

IT/CZ cluster meeting, Milano, 3–4 May 2004

Future of the L&B (will it be bright?)

Aleš Křenek



Notification (HOLUB) service

- basic implementation working, not stable yet
- conditions specifying when the notification should be sent have to contain JobId (or a list of them)
- extending notification validity and change of destination not yet implemented
- C++ API implemented, not tested thoroughly

- prototype working, “single job state” functionality
- standalone server (calling original L&B server core code)
- native clients (not using original L&B client library) in C and Java
- using GSOAP for C and AXIS for Java
- general outcome: transformation to plain WS is not hard
- security (vs. interoperability) is the major issue

Possible L&B contributions to the EGEE prototypes

some light into the “L&B role unclear” statements in Cork presentations:

- stable and proven service, available now
- lightweight, specialized and optimized for the purpose of job monitoring
- security
 - all communication is mutually authenticated and encrypted
 - authorization applied whenever user credentials are involved
 - authorization information controlled dynamically by the users on a per-job basis
- high flexibility and scalability due to many-to-many L&B server to Resource Broker mapping (unlike AliEn job monitoring tightly coupled with Task Queue)
- fault tolerance:
 - event delivery: events accepted by local service; if final destination is not available, delivery is retried repeatedly
 - job state machine: even losing some events is not fatal, at least partial job state is still available
- supported advanced features: interactive jobs, job dependencies (DAGs) and resubmissions

Job life within ARDA architecture

- L&B implements the requirements on Job Monitoring service besides sandbox management and direct RB monitoring (the current LM task, e. g. triggering resubmissions)
- retain the L&B design of events being gathered to form job state
- the only new feature (wrt. EDG) is CE pulling job from TQ but the CE's request to pull is out of scope of L&B (not yet related to particular JobId)
- i. e. pull and push submission to CE are virtually the same from L&B point of view
- adapting L&B to modified flow of events is easy
 - both event and job state types and their fields are defined in a single place, language bindings and protocols are auto-generated
 - processing events to compute job state is concentrated in one function

Implementing Job Provenance service

- why is it so separated from Job Monitoring?
- L&B events contain most of the necessary information (and can be extended)
- “aged” events can be dumped on per-job basis for archival instead of simply purged (already implemented)
- searchable catalogue (or index) of dumped data can be provided
- we are ready to take care of this service

- both R-GMA and L&B are grid monitoring infrastructures, despite having different focus
- it is worth to identify common features precisely and achieve feasible level of integration
- profitable for both sides
 - we get the functionality of registry & mediator
 - we can provide secure & persistent transport layer
- webservice framework could be the right way
 - interoperability at the protocol level
 - independent implementations with purpose-specific properties
- planned actions
 - review of L&B architecture, precise component mapping to the GMA concept
 - review of current and proposed WS-enabled R-GMA
 - starting negotiations with the R-GMA group to achieve common and stable service description (WSDL)
 - independent but interoperable implementation of the functionality required by L&B