



Enabling Grids for E-science

CRL distribution using L&B

Daniel Kouřil

CESNET, kouril@ics.muni.cz

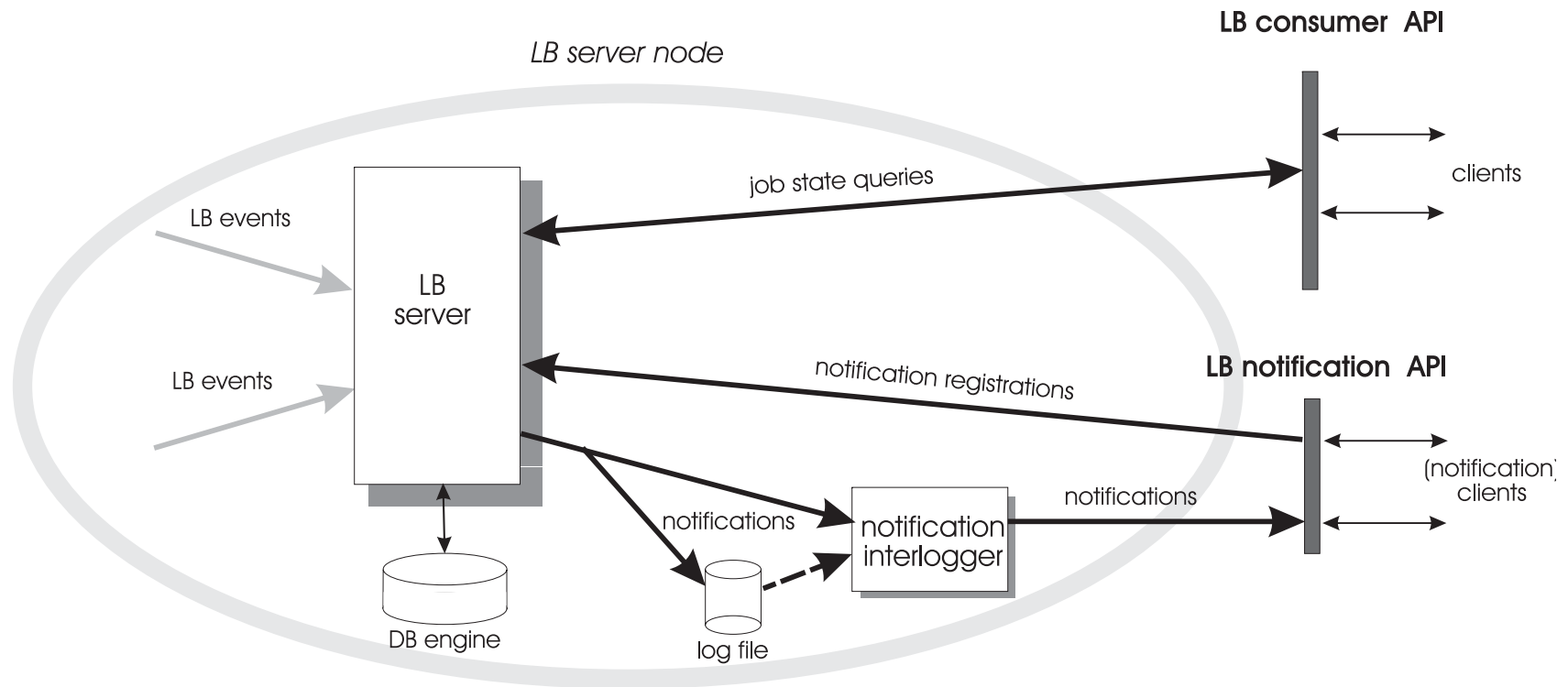
MWSG meeting, Jun 12-13 2007

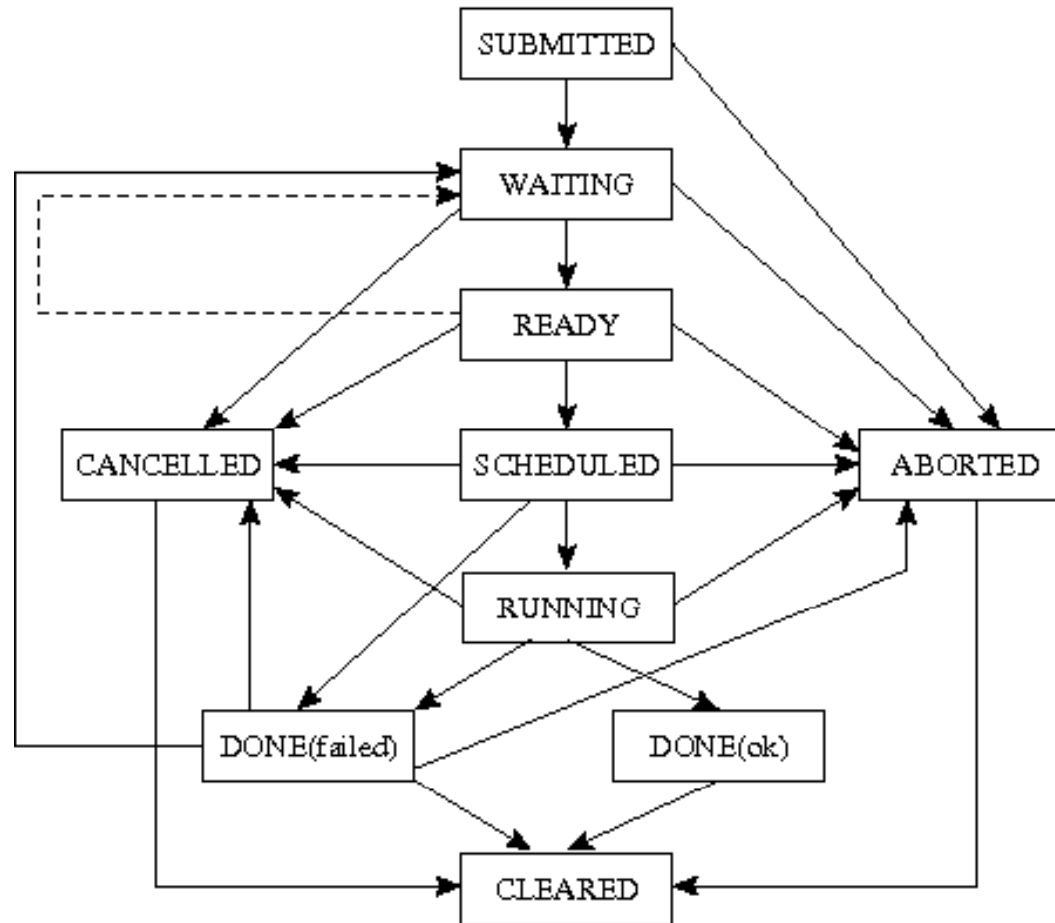
www.eu-egee.org



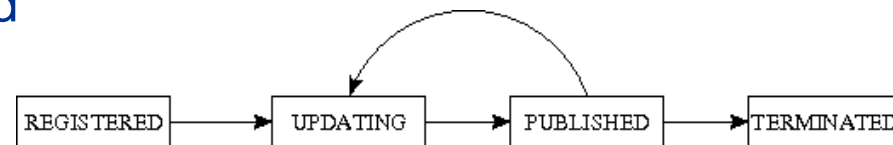
- **Monitoring system to track jobs in large grids**
 - in production for many years
- **Two basic layers**
 - LB server storing and processing job related data
 - LB messaging infrastructure
- **Currently for jobs passing via WMS**
 - the internal server part is separated
 - recently adapted for monitoring PBS and Condor jobs
- **Query interface**
 - complex queries on jobs and their status
- **Notifications**
 - sent by LB server on changes
 - a simple application needed on the client side

- **LB collects events from individual Grid components**
 - information about a important point in the job's lifetime
 - Events sent as messages to the LB server
 - The message delivery infrastructure is secure (encrypted, authenticated) and reliable (messages are eventually delivered even in case of temporal problems)
 - notifications use this messaging infrastructure too
- **Push model**
 - events are sent by the components (mostly WMS) upon changes
 - instrumented components or reading log files
- **Event are processed on the LB server**
 - LB defines Job state diagram
 - Each event could trigger a change in a job state (computed on the fly)





- **Current retrieval of CRLs has some issues**
 - long window between the actual publication and retrieval
 - unnecessary polling CA servers on the other hand
- **OCSP introduces overhead to communication**
- **LB to maintain and distribute CRLs**
 - push model
 - using the notification mechanism
 - sends only what's needed



• CRL state diagram

- query interface
 - „give me current CRLs of CA A, B, C“
 - „give me CRLs of A, B, C that changes since last Monday“

- **Each CRL treated as a single job**
 - A global unique identifier
 - LB identification (protocol, hostname, port)
 - Common prefix labeling the CRL
 - Hash computed from the CA public key
 - The Job identifier can be constructed by a client in a deterministic way
- **Update of CRL sent as an event**
 - Simple agent that queries CA frequently
 - Each event changes the state (usually UPDATING -> PUBLISHED)
 - Each state change triggers a notification

- **Notification interface**
 - Each client must subscribe
 - List of CAs whose CRLs it is interested in
 - Subscription time limited (must be renewed)
- **Robust delivery**
 - undelivered notifications remain active and are re-tried
 - Client can re-register under a new address (mobile clients)
- **A simple client available**
 - Capable of use both notification and querying
 - Always store the up to date CRLs on the local disk
 - The user does not see any difference from the standard solutions with downloaded CRLs
 - firewall problems
 - users can still query the LB

- **An alternative way to distribute CRLs**
- **Demonstration of the LB potential**