L&B overview

(as seen by a tester)

O. Bouhali & S. Rugovac

IIHE-ULB

Brussels

EGEE and gLite are registered trademarks

Outline

- Introduction: job monitoring systems
- L&b: purpose/requirements/general concept
- L&B: current implementation
- J&P: introduction and current implementation

Introduction:

L&B

- initially designed, in EDG, as part of WMS
- In EGEE: it becomes an independent component of gLite

It should provide the missing features in most of job monitoring systems:

- **Security: authentication of all components through which the monitoring events are passed**
- **Deal with multiplicity of monitoring events sources**
- **User should be able to store specific monitoring events**
- **Robust and fault-tolerant logging**

L&B : purpose and requirements

Track WMS jobs as they are treated by individual Grid components:

- **Collect (raw) information from several Grid components**
- **Process these informations and make a global view**
- **Make it available at a single access point**
- **Keep tracks of job even after the execution has been finished**

L&B General concept: L&B events

- **A unique jobid for each job entering the grid**
- **Information gathered from Grid component: LB events**
- **LB events:**
  - LB events are structured into (attribute= value) pairs
  - Attribute: defined by the event type (e.g, timestamp, event origin,..)
  - LB events are strored in a DB
  - All events belonging to the same job must be sent to the same LB DataBase (address of the LB DB is within the jobid)
- **LB processes all LB events and provides a higher view of the job**

L&B General concept: Events processing

- **Event gathering based on the push model:**
  - Grid components produce and send events
- **Event delivery to the L&B server:**
  - Based on store-and-forward model
  - sent synchronously only to the nearest L&B component responsible for later delivery
  - Copied into a local disk file and a confirmation is sent back to the producing component
  - The L&B component responsible for delivery: treat the event, and deliver it to the destination server

L&B General concept: Events processing

- **Event processing:**
  - Raw L&B events originate from different grid components
  - may arrive in wrong order
  - complex processing is needed to give the correct view
  - A jobs state machine is defined: responsible for updating the job state whenever a new event (of particular type and from a particular source)is received
  - Highly fault-tolerant (for delayed, lost events,..)

L&B General concept: Events processing

Job state list

Submitted: Entered by the user to the User Interface or registered by Job Partitioner.

Waiting: Accepted by WMS, waiting for resource allocation.

Ready: Matching resources found.

Scheduled: Accepted by LRMS queue.

Running: Executable is running.

Done: Execution finished, output is available.

Cleared: Output transfered back to user and freed.

Aborted: Aborted by system (at any stage).

Canceled: Canceled by user.

Unknown: Status cannot be determined.

Purged: Job has been purged from bookkeeping server (for LB->RGMA interface).

Event type list

Transfer: Start, success, or failure of job transfer to another component.

Accepted: Accepting job (successful counterpart to Transfer).

Refused: Refusing job (unsuccessful counterpart to Transfer).

EnQueued: The job has been enqueued in an inter-component queue.

DeQueued: The job has been dequeued from an inter-component queue

HelperCall: Helper component is called.

HelperReturn: Helper component is returning the control.

Running: Job wrapper started.

Resubmission: Result of resubmission decision.

Done: Execution terminated (normally or abnormally).

Cancel: Cancel operation has been attempted on the job.

Abort: Job aborted by system.

Clear: Job cleared, output sandbox removed

Purge: Job is purged from bookkeeping server.

Match: Matching CE found.

Pending: No matching CE found yet.

RegJob: New job registration.

Chkpt: Application-specific checkpoint record.

Listener: Listening network port for interactive control.

CurDescr: Current state of job processing (optional event).

Event processing

Example: State= submitted

```
Submitting events to the job: https://pc900.iihe.ac.be:9000/JtyoxFtWvjstjqqqxAjmmg

event submitted.....................................[submitted]
UI=000000:NS=0000000000:WM=000000:BH=0000000000:JSS=000000:LM=000000:LRMS=000000:APP=000002
Events Sent.....
     1  UserTag
     2  RegJob
Events recorded in the LB server.....
     1  UserTag
     2  RegJob
```

Event processing

Example: State= Done

```
Submitting events to the job: https://pc900.iihe.ac.be:9000/ita2iJ5kZIpjllRKeUqKIQ

event submitted.....................................[done]
UI=000000:NS=0000000000:WM=000000:BH=0000000000:JSS=000000:LM=000000:LRMS=000000:APP=000002
Events Sent.....
     1  UserTag
     2  RegJob
     3  Accepted
     4  EnQueued
     5  DeQueued
     6  HelperCall
     7  Match
     8  HelperReturn
     9  EnQueued
    10  DeQueued
    11  Transfer
    12  Accepted
    13  Transfer
    14  Running
    15  Done
Events recorded in the LB server.....
     1  UserTag
     2  RegJob
     3  Accepted
     4  EnQueued
     5  DeQueued
     6  HelperCall
     7  Match
     8  HelperReturn
     9  EnQueued
    10  DeQueued
    11  Transfer
    12  Accepted
    13  Transfer
    14  Running
    15  Done
```

L&B General concept: User side

- **The user can retrieve the data in two modes:**

  Query and notification modes

- **Queries:**
  - user specifies the query conditions, connect to the querying infrastrcture endpoint, gets the result from the server
  - Two types of queries:
    - job queries: return job status,…
    - event queries: returns L&B raw events (for debugging)

- **Notifications:**
  - User can register to receive notifications about a specific job

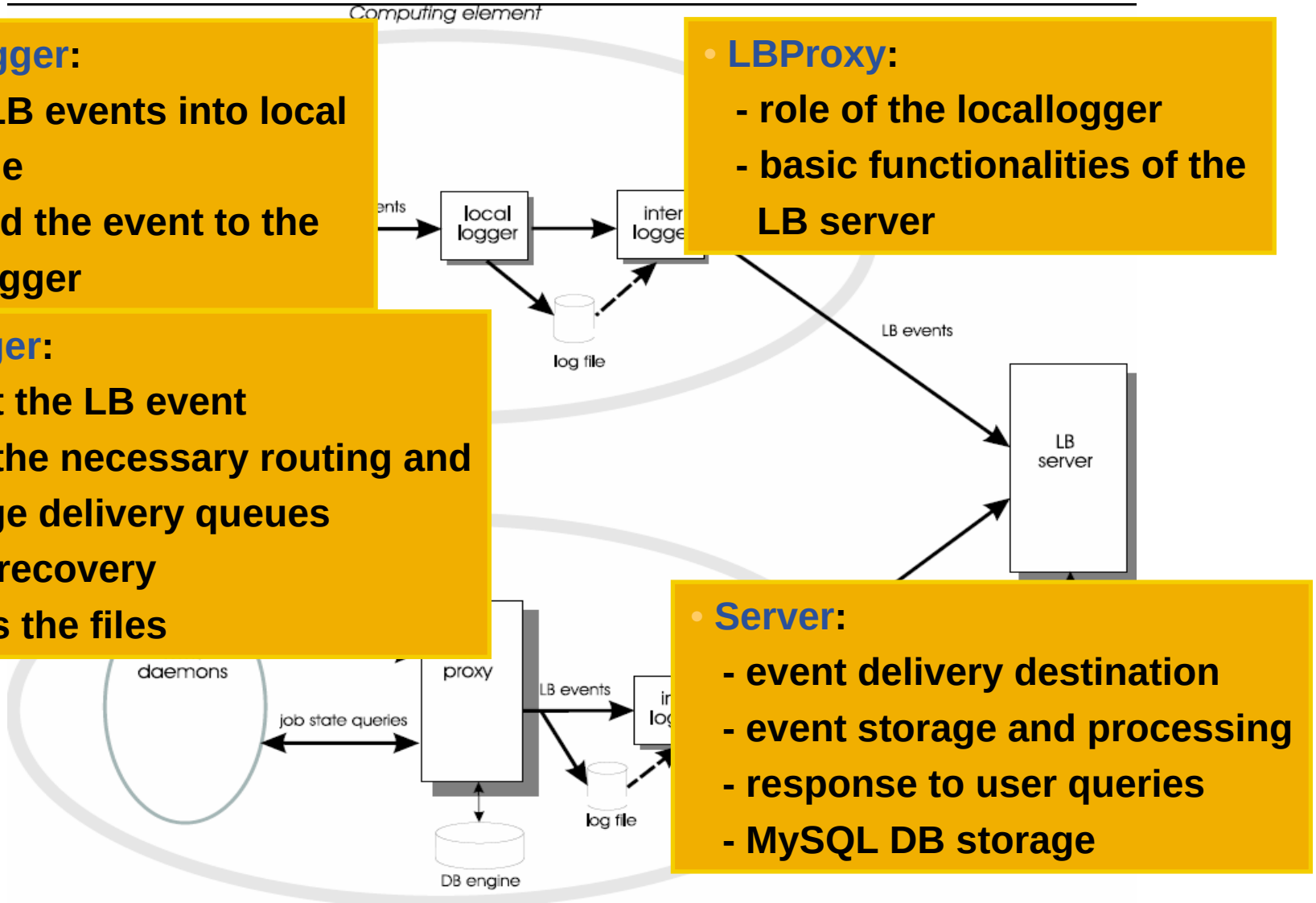
L&B: current version

- **Locallogger:**
  - copy LB events into local disk file
  - forward the event to the interlogger

- **Interlogger:**
  - accept the LB event
  - make the necessary routing and
  - manage delivery queues
  - crash recovery
  - purges the files

- **LBProxy:**
  - role of the locallogger
  - basic functionalities of the LB server

- **Server:**
  - event delivery destination
  - event storage and processing
  - response to user queries
  - MySQL DB storage



L&B: current version

- **Logging events and querying the service is accomplished using calls to a public L&B API set.**

- **Query is also available as a web interface**
- **Security:**
  - All L&B componenets communicate over authenticated channels (TLS protocal)
  - ACL are implemented by the L&B server: job information is only allowed to the job owner
  - Jobs owner can assign an access control list

Job Provenance (JP): puropose/requirements

- Provides permanent storage system
- Provides a querying interface to the data about the jobs and the conditions they were run in (be able to re-run the job)
- Keep detailed information/size as small as possible
- Enable the user to add annotations
- Deal with old and new data format

Job Provenance (JP): Data gathering

- **JP data are organized in a per job basis**
- **Three kinds of data:**
  - Job input: JDL + input files provided by the user (not remote storage files)
  - Job execution: L&B data, CE info (environnement, OS, version,..)
  - User annotations: relationships between the job and other external entities (remote storage files,..)
- **Once a piece of data is recorded it can never be updated/replaced. New values can be appended**

Job Provenance (JP): Architecture

JP: two classes of services:

- **The Primary Storage (JPPS):**
  - Responsible for gathering the job data and their long-term archival
  - One JPPS per VO is foreseen (within EGEE): is this enough?
- **Index Server (JPIS):**
  - Responsible for processing and re-arranging JPPS data into a form suitable for user queries.

Summary

- **LB and JP: lack of information (being solved)**

  exploring the services is taking time

- **Useful links are:**

(wiki page for testing with JRA1)

http://egee.cesnet.cz/mediawiki/index.php?title=LB_and_JP_Performance_Testing

LB and JP user's Guides

http://egee.cesnet.cz/en/JRA1/