Table of Contents

RB Notes

Introduction

The Resource Broker manages the job workflow on behalf of the user. Specifically,

- Accepts requests from a UI, authenticates the user and manages the input and outut sandboxes for the jobs. It may also register the user proxy for renewal with the MyProxy service.
- Performs matching to identify the best CE for the job based on the user's criteria and the current site usage
- Tracks the job status information and records it to a database for query by the user and other tools.

The RB participates in the following flows

- WmsFlows

Data

The RB data consists of

- A MySQL database to store the logging and bookkeeping data (LB)
- A file based storage of the currently running jobs managed by this RB
- The input and output sandboxes waiting for user retrieval

Configuration

The combination of MyProxy/ResourceBroker/VOMS server must be configured and consistent. The RB is selected on a per-site/per-VO basis. However, each RB only knows abouts the jobs which have been handled by it. Thus, the user must always go back to the same RB to determine the status of their jobs and retrieve the output.

It is planned that each VO will have at least 1 RB for general purpose users. Additional RBs may be made available to address load issues. However, since the users must go back to the same RB each time, care needs to be taken how these RBs would be allocated to avoid jobs being submitted and then 'lost' because the RB used was not remembered.

The Logging and Bookeeping service required a MySQL database. The database itself can be on a different box if required.

Performance

The resource broker is very performance sensitive, especially in bulk submission. An extended review of the RB performance can be found at https://twiki.cern.ch/twiki/bin/view/FIOgroup/TechPrbHwscanWrong.

High Availability

The RB service failure has the following impact

- New jobs cannot be submitted
- Status of existing jobs cannot be queried
- Jobs which complete will not be shown as completed until the RB service has been recovered

- Output data from jobs may be lost since they cannot copy the job results to the output sandbox
- User sandboxes will not be available for retrieval

Currently, the RB service does not support IP aliases. This is being worked on and should be fixed before SC4 implementation.

There is a drain function available to stop new submissions which allowing old submissions to complete.

Approach RB 1

An IP alias rb*vo*.cern.ch will be defined which allows the service to be switched between machines if required.

All state data will be stored 'off the box'. The state data consists of several directories (/var/edgwl,...) and the MySQL database server.

Thus, in the event of failure of the master, the slave would take over the external disks. The state data stored on file systems would be 'rolled back' using ext3 functions. The MySQL database would be restarted and would play its redo log to arrive at a consistent state.

Approach RB 2

The database for logging and bookkeeping is split off onto separate servers. The MySQL servers can then be shared between all of the resource brokers.

Using replication from the master to slave, the slave can take over the role of the master in the event of a failure. This also resolves the issue of hot online backups in MySQL since you just stop the slave, perform the backup and then start the slave again.

Approach RB 3

The RB configuration will be made in a standalone server with internal but hot-swappable disks. In the event of hardware failure of the machine, the disks can be moved to another machine. This operation would be manual and probably require the presence of an administrator.

Equipment required

Approach 1

Assuming n RBs and 2 spares, the hardware required is

Component	Number	Purpose
Midrange Server	n+2	RB masters and standby machines
FC HBA	n+2	Fibre channel connectivity
FC Switch Ports	2*n+2	Connectivity for the two servers
FC Disk space	20	Storage for credentials (2x10GB on different disk subsystems)

Approach 2

Assuming n RBs and 2 spares, the hardware required is

Component	Number	Purpose
Midrange Server	n+4+2	n RB masters and standby machines along with 2 MySQL clusters
FC HBA	n+4+2	Fibre channel connectivity
FC Switch Ports	2*n+8+2	Connectivity for the two servers
FC Disk space	100	Storage for database and logs. Sandbox disk space is covered below

Sandbox disk space requirements

Disk space required is based on the following data.

The 10MB limit on the input sandbox is already enforced. The output sandbox limit is not enforced but the aim will be similar as the input sandbox.

The number of jobs is based on the CERN LSF turnover assuming that in the LHC time frame, the majority of the jobs will be coming from the grid. With the current batch farm of 1400 machines, there are around 22000 jobs/day. For the LHC time frame, the total batch farm capacity will be around 4-5 times larger.

Parameter	Value (MB)
Size of input sandbox	10
Size of output sandbox	10
Jobs / Day currently	21000
Estimated Factor for LHC	3
Sandbox Purge Time (days)	14
Jobs in queue	35000
Disk Space Required	17640000

Thus, the total space required for all RBs is 17.6 TBytes.

Based on the estimated from the support team (IssueRbDiskSpace), 50% of this would be sufficient to allow for cases where users have retrieved their output earlier than the purge time.

Engineering required

Development	Purpose
Hot backup for MySQL	A Hot backup procedure needs to be developed. The MySQL database cannot be shutdown for extended periods of time while the backup is performed
Start/Stop/Status procedure	Scripts for RB operations
Replication procedure for MySQL	Enable MySQL master/slave setup
Lemon MySQL availability test	A lemon aware sensor which can be used for reporting availability.
Lemon RB availability test	A lemon aware sensor which can be used for reporting availability. Tests for GridFTP, LDAP, Condor-G, EDG-WL processes would be required.
Linux Heartbeat availability test	A Linux-HA aware sensor which would activate the procedure for automatic switch from master to slave
Switch procedure	Automatic switch from master to slave changing the DNS alias, disabling the master, enabling the slave in its new master role
Capacity Metric	Capacity metrics defined for Number of renewals / second

	Number of inits / second
Quattor configuration for Linux-HA	NCM component to configure Linux-HA/Heartbeat

Other Items to Consider

RBs are a per-VO configuration to avoid one VO causing problems for another one. The spare slave boxes though could be shared between the VOs until a problem occurs.

-- TimBell - 13 Sep 2005

This topic: LCG > RbNotes
Topic revision: r9 - 2006-09-12 - TimBell