

Table of Contents

EMI WMS best practices.....	1
1 Objective.....	2
2 Hardware.....	3
2.1 Suggested hardware for WMS service.....	3
2.2 Hardware (WMSMonitor).....	3
2.3 Hardware (DNS if used only for this purpose).....	3
3 Physical vs Virtual Machines.....	4
3.1 DNS round robin load balacing.....	4
3.2 WMSMON metrics.....	4
4 WMS maintenance.....	5

EMI WMS best practices

1 Objective

When using more WMS instances at one site, it is possible to coordinate them to obtain high-availability and load balancing. This is done via a specific service, called WmsMon, maintained by EGI. This document is mainly to provide guidelines on how to set up a WMS pool through the deployment of the WmsMonitor service. Information on suggested hardware and software configurations for a single WMS instance are also provided.

Topics:

- Requirements to deploy a wmsmonitor service
- Best practices from a client perspective
- Best practices to implement a High Availability WMS service
- WMS maintenance

2 Hardware

2.1 Suggested hardware for WMS service

Each WMS should be installed in a dedicated HW. There are reported deployments on virtual machines, and no specific constraints are found in this case, yet performance might be severely affected.

Number of cores, RAM, disk space and number of machine should be proportional to the number of supported VOs and the number of job submitted to the WMS. Minimum requirements are:

- quad core CPU
- 150 GB storage
- 4 GB RAM on 32-bit architectures, 8 GB RAM on 64-bit

Peak write rate for the storage should be about 100Mb/s at least, which is a relaxed requirement with what provided by the market nowadays.

With this configuration, each WMS node should be able to process about 30.000 jobs/day (repeated for several days in a single month), with peaks at 50.000 jobs/day. The directories that are more most populated/written by the system are:

```
/var/lib/mysql  
/var/glite/SandboxDir
```

mounting these two directories in two different physical disks might significantly improve the performance.

2.2 Hardware (WMSMonitor)

- dual core CPU
- 20GB of hard disk space
- 4 GB RAM

2.3 Hardware (DNS if used only for this purpose)

- dual core CPU (it can also be installed as a virtual machine)
- 10GB of hard disk space
- 1 GB RAM

3 Physical vs Virtual Machines

Given the minimum hardware requirements there should not be any difference in using a Physical or Virtual Machine for the WMSMonitor and the DNS. However in the case of WMSMonitor the use of a database and the frequent disk access could be a limiting factor in using a Virtual Machine. For a small number of clients this should not represent an issue. The use of virtio can improve performances.

3.1 DNS round robin load balacing

Load balancing is a technique to distribute workload evenly across two or more resources. A load balancing method, which does not necessarily require a dedicated software or hardware node, is called round robin DNS.

We can not assume that all the job submitted to the WMS will require the same amount of resources and thus generate the same resource load (this depends on the job request, if there are errors in the submission and the job needs to be resubmitted, how many times it must be resubmitted, etc.). The load depends also on the type of hardware in which the WMS is installed. For an effective load balancing, a pool of available WMS should be regularly updated and the WMSs that have a higher load should be removed from that pool. All the WMSs that are in the pool should be used in a round robin fashion based on the DNS name resolution. With the help of the sensors installed in each WMS, the loadbalancing takes care to add and remove the WMSs from the pool on the DNS by updating records mapping on the same hostname. This results in a hostname that maps to multiple IP addresses under the configured DNS zone. As an example: in dns.top.domain, add multiple A records mapping to the same hostname with multiple IP addresses

```
Zone wms.zone.domain
name.wms.zone.domain IN A x.x.x.x
name.wms.zone.domain IN A y.y.y.y
name.wms.zone.domain IN A z.z.z.z
```

The 3 records are always served as answer but the order of the records will rotate in each DNS query. If the metrics on the test performed on one of those WMS report problems, the WMS is removed from the pool by removing the corresponding entry on the DNS. This mechanism provides fault tolerance of the WMSs.

In a similar way, a configurable number of WMSs that is under highest utilization is kept out of the pool to allow the submission of new jobs only to the WMSs that have less load.

3.2 WMSMON metrics

The metric measured by the wmsmonitor rely on sensors installed on each WMS. The detailed procedures for the WMS sensors and WMSMonitor server installation are available at this address:

https://wiki.italiangrid.it/twiki/bin/view/WMSMonitor/InstallationProcedureV2_1

Documentation for WMSMON is available at this address:

<https://wiki.italiangrid.it/twiki/bin/view/WMSMonitor/WebDownload>

and the packages are distributed on request.

4 WMS maintenance

During normal operation, there is regular maintenance to perform. The MySQL database usually grows for each new job that is submitted and does not shrink when jobs are removed. This leads to a higher disk utilization. For this reason one of the two following operations (one which requires that the WMS is drained and one that does not require to put offline the WMS) can and should be performed if the free space is less than 20% on the storage used by MySQL:

- (WMS does not need to be drained) before the installation of the WMS and the creation of the databases this configuration should be added, if not already present, in the MySQL configuration file "/etc/my.cnf":

```
innodb_file_per_table
default-storage-engine=InnoDB
```

in the "[mysqld]" section. When the available space is low it is now possible to use the following command:

```
mysqlcheck --optimize lbproxy -u root -p
```

that optimize and reduce the size of the MySQL tables and files on the disk.

- (WMS needs to be drained) after the WMS is drained the services can be stopped:

```
service gLite stop
service mysqld stop
```

It is then possible to remove the MySQL files

```
rm -rf /var/lib/mysql
```

at this point a reconfiguration of the WMS is necessary. This will recreate the database and tables structure.

Another operation that might be required is to add a line in the configuration file of the WMS to limit the size of the OutputSandbox. WMS should not be used to transfer big files from the WN to the user. A suitable SE should be used instead. This can be enforced by adding a line similar to:

```
MaxOutputSandboxSize = 55000000;
```

in "WorkloadManager" section of "/opt/glite/etc/glite_wms.conf" configuration file which limits, in this case, the OutputSandbox to about 55Mb. The service should be restarted after the change of configuration:

```
/opt/glite/etc/init.d/glite-wms-wm restart
```

-- MarcoCecchi - 2013-04-05

This topic: WMS > WMSBestPractices

Topic revision: r2 - 2013-04-15 - MarcoCecchi



Copyright © 2008-2021 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.

Ideas, requests, problems regarding TWiki? Send feedback