



IST-2001-32133

GridLab - A Grid Application Toolkit and Testbed

Testbed Prototype

Author(s):	Luděk Matyska and Miroslav Ruda
Document Filename:	D5-2
Work package:	Testbed
Partner(s):	Masaryk University
Lead Partner:	Masaryk University
Config ID:	GridLab-5-TPR-0001-1-0
Document classification:	INTERNAL

Abstract: This document defines GridLab testbed prototype environment, its basic properties and requirements, that must be fulfilled by all the participating nodes.





Contents

1 Overview	2
1.1 GridLab Testbed Purpose	2
2 Security and Accounts	2
2.1 User Accounts	3
2.2 Account Generation	3
3 Information Service	4
4 Required Software on Nodes	5
5 Local Settings	6
5.1 Local Batch Systems	6
6 Accounting	6
7 Testbed Management	7
7.1 Testbed versions	7
8 Open Issues	7

1 Overview

The main purpose of this document is to define GridLab testbed environment, its basic properties and requirements, that must be fulfilled by all the participating nodes. The documents defines the principal properties of individual nodes together with a proposal how to control and check their correct functions. This document describes the GridLab testbed prototype and will undergo further evolution as the testbed itself will evolve. Parts of this document will be incorporated into the technical paper *Testbed Policy* (Deliverable D5.1).

1.1 GridLab Testbed Purpose

The GridLab testbed is built to support the GridLab project providing a functioning environment, where the *development* of the different GATs can be carried on and tested, and where the *proof of concept* test production runs can be performed from time to time. While the testbed is constructed to achieve a production-level stability, this does not mean that the testbed could or will be used for a long lasting production runs. The resources and the testbed behavior necessary for the long term production runs are out of scope of the GridLab testbed.

The GridLab testbed will be available primary for people participating directly on the GridLab project. The development nature of the testbed will require rather often changes in the underlying fabrics, i.e., all or parts of the installed software, the ways the testbed is run, the interfaces to its basic functions etc. These changes will be done with close coordination with the development teams of individual GridLab workpackages, with the emphasis to support testing of new services and features. Also user oriented access to the GridLab testbed through the portals and similar high level approaches will change in time, respecting the development of new portal features.

This document describes the functionality and requirements posed on the “core” testbed, i.e., the nodes committed to comply with the general setup. These nodes will be primary provided by the GridLab project active partners and subcontractors. The interoperability with other testbeds or collaboration with nodes that will not completely comply with this document is out of scope of this document and will be discussed elsewhere.

2 Security and Accounts

The first GridLab testbed will use GSI (Grid Security Infrastructure) as its security infrastructure. The GSI is based on public key cryptography, X.509 certificates and the SSL (Secure Sockets Layer) communication protocol, and is currently widely accepted in many Grids¹. We expect all the services (`ftp`, `ssh`, job submit, `cvs` etc.) to be GSI enabled and no other security mechanism will be supported in the testbed version 1. This may change in the following version based on the recommendations and requirements of the security working group.

The key concept in the GSI is the certificate, as every entity on the Grid—user, host, in general every service— authenticates itself via a certificate, that was issued and signed by some Certification Authority (CA). For the GridLab testbed purpose we recommend to establish a list of individual CAs accepted by the nodes. These CAs will issue certificates that should be accepted by all (or at least majority of) the computing nodes— similar procedure is currently under way within, e.g., the DataGrid project, whose trusted CAs can be probably used for the GridLab testbed, too. Each individual node will have right to add a particular CA to its list

¹More detailed description of the security infrastructure can be found in I. Foster, C. Kesselman, G. Tsudik and S. Tuecke: *A Security Architecture for Computational Grids*, ACM Conference on Computers and Security, pp. 83–91 (1998) or on the web <http://www.globus.org/research/papers.html#security-arch>.

of trusted CAs on the base of local acceptance policy for a CA. However, we plan to maintain a list of accepted CAs, its actual version can be found on the GridLab web pages (<http://www.gridlab.org/WorkPackages/wp-5/testbed.html>). The initial set of accepted CAs will be build around the EU DataGrid CAs and several US ones (primary Alliance and DOE CAs). We expect existence of multiple user certificates signed by different CAs. All certificates of one person will be mapped to the same local account. This will allow to overcome the potential problem with nodes accepting only subset of all the CAs otherwise accepted by the GridLab testbed nodes (we expect such limitation with, e.g., some US nodes with strict access policies).

2.1 User Accounts

Access to the individual computing nodes will require use of local accounts. The Grid-wide user identity is provided via certificates, whose subjects are mapped to local login names via a `grid-mapfile` on each local resource. This provides the simplest authorization scheme that will be used in the first testbed version. The information necessary to generate localized `grid-mapfiles` will be stored in an LDAP tree (see below, together with support for this info retrieval).

The user account system should provide a secure and flexible way how to map grid-wide identities to the local accounts. The system should support at least the following features:

- The user is always mapped to a local account that is different (for the whole existence of this mapping) from accounts used by other users of the same local resource.
- The mapping information (not necessarily the mapping itself) must be persistent, i.e., it should be possible to identify a user even after her job finished and the mapping may no longer be valid (the local account could be re-used).
- If the local account is re-used by different users, some mechanism to protect local data must be available (e.g., in case of job crash the data must be removed or, better, transported somewhere where only the original owner can retrieve them).

It will be also beneficial, although not mandatory, to guarantee that there will always be a local account available.

While in longer term the GridLab testbed may use some virtual user account system, for the first testbed release we propose a schema where on each node an account named `gridlabXXX` (i.e., `gridlab001`, `gridlab002` and so on) will be created whenever a new user is accepted on the GridLab testbed. A common `grid-mapfile` can thus be provided, valid on all the testbed sites. As a benefit, we will provide the same local login names on all the computing resources.

The common `grid-mapfile` will be provided testbed-wide and we expect that individual nodes will append this GridLab specific `grid-mapfile` to their own mapfiles. Where a particular user already has an account different from the `gridlabXXX` account generated for her for the GridLab testbed, she will thus use this (older) account instead of the `gridlabXXX` account. We expect that this original account already has an item in the local node mapfile (mapping her certificate subject to it) and as the mapfile is scanned linearly, this will be used before the GridLab related account. However, the `gridlabXXX` account will remain reserved and will not be used for other user.

2.2 Account Generation

We assume that it will be the sole responsibility of each individual node to grant access to a user creating the specified `gridlabXXX` account. However, in order to simplify the whole process we expect to collect the information each individual node requires to grant access in a central

node. Each user will have to fill only one document (form), available on the web pages of the GridLab testbed, and sign it with his certificate. This information together with the expected login name `gridlabXXX` generated centrally will be sent to all GridLab testbed nodes, that at this moment will start their own admission procedure. This may include direct contact with the user, bypassing the GridLab Testbed Operation Center (GTOC) in Brno. At the end of this procedure the user is either granted the access to the resources—the `gridlabXXX` account is created—or the request is refused (no local account is created).

If at least one of the nodes accepts the user, the appropriate line is added to the common `grid-mapfile`, otherwise the user is not accepted and the account `gridlabXXX` originally reserved for him can be reused.

This model expects that the GridLab testbed serves just one virtual organization (VO), namely the VO of the GridLab project. Extensions to more VOs are possible, but are out of scope of this version of the paper.

3 Information Service

The information service for the first GridLab testbed version will use the LDAP based information infrastructure. Due to the heterogeneity of the information about the GridLab testbed, several *logically* independent LDAP trees will be operated by the GTOC. The logical independence does not always subsume physical independence, but in some cases, e.g., the accounting server, we expect to actually run different LDAP servers.

The information provided could be divided into the following major components:

1. Information about machines (hostname, number of CPUs, available memory, actual state etc.). Information will be provided via local Grid Resource Information Service (GRIS) and could be retrieved via the Grid Index Information Service. The Globus MDS-2.0 will be used here, with its schema. The address of the primary server is `ldap://ldap.gridlab.org`.

While the actual Globus/MDS-2 schema will be used, not all the information “providers” (i.e., the scripts) are already available or are correct. Patches and extensions will be provided in order to have up to date and correct information about all the nodes.

2. The information about the installed basic software (name, version, installation details etc.) will be also provided, in the way analogous to the machine info. However, precise schema must be developed and the corresponding information “providers” should be implemented.

An open question is whether the “basic software” information covers also the GridLab specific software (i.e., software developed as part of the GridLab project) or whether we should deploy a different LDAP subtree to store this information. It can be assumed that the GridLab developed software will change more often than the basic (middleware) software on individual nodes. It may be also possible that the developer would like to add some specific information, which may not be included in the basic software schema (annotations, specific comments, expected time of the next release or a patch etc.).

3. We also plan to store information about individual applications that should be supported on the GridLab testbed. The schema for this information should be developed in close collaboration with individual application providers and could include, among other, the name, version, installation directory or directories, the environmental variables and their default values etc.

4. Information about accounts (i.e., information necessary to generate the common `grid-mapfile`), pairing users' certificate subjects with the appropriate `gridlabXXX` accounts will be also stored on an LDAP based information server. The primary information producer will be the GridLab Testbed Operation Center, with information about users' acceptance or refusal will be filled in by individual nodes. All nodes will have read access to the tree and simple scripts to generate the appropriate `grid-mapfile` will be provided.

The accounting related information will be stored in a LDAP tree with rather restricted access rights to provide secure access to this potentially sensible information.

5. A specific LDAP server for the middleware control may be also provided, storing and providing a kind of "bookkeeping" information, e.g., information that may be valid only during a lifetime of a particular application or information used to synchronize some of the GridLab middleware components.

The access to the information servers will be authenticated and both read and write permissions will be granted only to authorized entities (users or services). All the master servers will run on machines in Brno, but replicas may be available at other places (volunteers?) to decrease the risk of the single point of failure. The master replica administration will be the sole responsibility of the GTOC. The only exception to this rule may be the application specific LDAP (or LDAPs) that may be started and operated independently from the GTOC.

4 Required Software on Nodes

We expect a UNIX compatible operating system available on all nodes, with IPv4 connectivity (possibly through the firewall). We also expect that a "standard" development environment, including some Fortran, C and C++ compilers, shells, Perl and/or other scripting languages and probably some support for Java (not mandatory) is also available at all sites.

Common shell environment setup will be arranged to guarantee that all required programs will be accessible via the same procedure ² on all sites.

The GridLab specific software belongs to the "middleware" layer and consists primary of the following:

- Globus 1.1.4. The full installation is expected to be available on all nodes. Scripts to generate the GridLab testbed `grid-mapfile` will be provided and the sites will be responsible to run them regularly (or to use other ways how to keep their `grid-mapfiles` updated.

We plan to upgrade to Globus 2.0 in the later stages (second testbed version?), in the meantime we expect to run Globus 2.0 (currently beta 2) on several nodes in parallel to the testbed "standard" Globus 1.1.4 (using non-standard port numbers to access individual services). Based on the experience with this second-level testbed, total upgrade to Globus 2.0 will be scheduled.

- MDS-2.0 based GRIS must run on each site. This GRIS must be registered in the master GIIS run in Brno and each site is responsible to fill the information server with up to date and correct information about the agreed upon local resources and their state.
- `gsi-ftp` from the Globus 2 distribution. Site refusing to install `gsi-ftp` should provide `gsi-wuftpd`, but this means no GridFTP extensions will be available on these sites.

²e.g., the directories with the required software in PATH and common commands to access different versions of required software.

- `gsi-ssh`, preferably the `ssh2` version from `openssh`. The open question is the port number allocation: if the `gsi-ssh` is not the local `ssh`, then the local `ssh` occupies the standard `ssh` port. Also, some local `gsi-ssh` modifications (e.g., Kerberos support) may collide with the standard port number³.
- MPI should be available on all sites. WP1 through WP3 must specify the required MPI features (like MPICH or MPICH-G version etc.).
- Other software may be also required by WP1 to WP3, like the HDF5.

The Globus, `ssh` and `ftp` must be installed on all sites, the application specific software may not be always available. The actual information about all the installed software must be always exported via the information service.

5 Local Settings

The GridLab testbed must not distort local operation of individual sites. However, different local services (e.g., local batch systems) has to be configured to support common environment of GridLab testbed.

5.1 Local Batch Systems

Several different local batch systems are expected to be used on testbed sites. Unified access to these systems is guaranteed by use of Globus GRAM service, but there are still small divergences in required job specification. For this purpose we expect that each individual site will create one batch queue called `gridlab` where all the GridLab authorized access will be locally scheduled. This queue will usually operate only limited local resources, not allowing thus their abuse by the GridLab collaborators.

In the later stages, more queues with specific properties may be established, but for the first GridLab testbed version only one queue is currently assumed.

Also, all sites should provide a uniform environment setting for all the `gridlabXXX` accounts. A set of local scripts for environment settings will be provided to fulfill this task.

6 Accounting

The first GridLab testbed version will not deploy any GridLab related accounting system. It is assumed that each site runs its own local accounting system, which is capable to report basic accounting information (the CPU and probably also wall clock time) about completed jobs. We propose that all the accounting information will be stored in a specific LDAP tree. Individual nodes will be granted authorization to add new accounting information (via the LDAP access protocol). This information will be stored and made available, usually in aggregate form, for statistical purposes and also to evaluate use of sites by individual users. Later on, more up to date information may be required by the resource broker to optimize job placement also with respect to the already consumed resources. The precise schema for the accounting information to be provided by the individual sites and made available via the LDAP will be therefore developed with close cooperation of providers (the individual nodes) and the potential consumers.

³If the port number of the `grid-ssh` is stored in the LDAP server, it may be a good candidate to stress test the GAT capability to discover a particular service or its local modifications. However, the different port numbers may be a nightmare for end users' direct access.

The proposed static mapping between certificate subject (i.e., individual user) and local account makes the accounting information storage and retrieval rather simple. It can be always stored under the local account name and the global database (used to generate the common `grid-mapfile`) can be used to query the “real” identity of a particular GridLab account (the authorization to run this query may be severely limited, is necessary).

7 Testbed Management

While the GTOC does not have any managerial power over local sites, the reliable testbed operation is impossible without defined control procedures. Each site should delegate one person to the Testbed Management Board, that will be chaired by the representative of the WP6 (one of the primary authors of this document). This TMB will decide on all issues where an approval of all sites is necessary (like the launch of the Globus 2.0 upgrade or the acceptance of a CA). The autonomy of individual testbed sites will allow them to withdraw when a majority decision goes against their interest. In such cases, another round of negotiations will be initiated by the TMB chair to solve the dispute.

We require each site to provide also a local contact point — this means e-mail and telephone number of a person or a helpdesk — which is prepared to take care of bug reports and complaints about local resources. The bug system will be based on bugzilla, again run by GTOC in Brno, with all the GridLab testbed local contacts subscribed. The list of contact points will be maintained on the WP5 (Testbed) web pages.

In the later stages, the GridLab testbed monitoring system will be launched, using both the standard software (like NetSaint) and the software developed within the GridLab project itself. This system is expected to actively monitor both the basic functionality as well as more advanced services that should be available on the GridLab testbed, and to notify (either directly or via the bugzilla system) the testbed administrators (both global and the appropriate local ones).

7.1 Testbed versions

We plan to number each consecutive testbed version. The first testbed, launched at the end of March 2002, will have version number 1.0. The major number will change with a substantial testbed middleware and/or management software change, while the changes in minor number(s) are reserved for smaller upgrades and/or bug or functionality fixes. We do not expect to have a predefined schedule for new testbed versions, new version will be deployed as a follow up of some middleware change.

A mechanism for new testbed version deployment will be defined, based on common agreement between individual nodes and GridLab software developers. All changes must be coordinated to have a compatible middleware installed on all sites. The change of minor testbed version number may be announced no more than few days beforehand, while the major number upgrade will be prepared in much longer time scale (in order of several weeks at least). The agreed upgrades will be mandatory for all the GridLab nodes.

8 Open Issues

The following open issues were already identified:

1. State of the GridLab developed software. Are there plans for a staged deployment of the GridLab software (they may require changes in the underlying testbed infrastructure) or is it more expected (due to the more experimental nature of the GridLab project) that

changes are to be made available on the testbed as soon as possible, with no strict a-priori schedule?

And are there specific requirements with respect to the information service or will the GridLab developed software fit into the same schema with the basic software?

2. Port number allocation for the `gsi-ssh`. The initial suggestion is to use port 22 for local `ssh` and port 2222 for the `gsi-ssh`.
3. List of required software that must be available on all nodes. This may include specific parts of the development environment (compilers, shells etc.), application specific programs (e.g., HDF5) and programs usually understood as part of the middleware (e.g., `ftp`).
4. Firewalls, i.e., requirements on the connectivity (what kind of connectivity is expected by the applications) and how to fulfill them in on nodes behind the firewall. This may also include general rules how to setup a firewall (individual port numbers and ranges, functionality restrictions etc.).
5. LDAP schema for accounting information and the periodicity of the information provided by the nodes must be agreed on (when a job finishes, once a day, a week, ...).