

Adding Computational Resources to

(Version 2.0 - January 2007)

SURAGrid is a multi-institutional initiative to develop a multi-purpose grid infrastructure to support the sharing of resources within and between institutions. SURAGrid is both a development and operational environment, where flexibility in deployed processes and technologies supports a growing set of heterogeneous resources, provides interoperability between those resources and future compatibility with emerging national and international grid standards. The SURAGrid user community is also diverse and expected to increase in diversity as both new and traditional grid applications are supported. To view a current list of SURAGrid participants, see http://www.sura.org/programs/sura_grid.html.

This document outlines a “process-in-progress” for adding computational resources to SURAGrid and is targeted at system administrators of the resource that is being deployed. The SURAGrid infrastructure is currently implemented using Globus Toolkit (<http://www.globus.org/toolkit/>) technologies and the level of information provided assumes some familiarity with Globus, as well as basic understanding of the definition and intent of grid technology. If you are not sure if this is the latest version of this document, please check for the latest version on the SURAGrid Web site before proceeding: http://www.sura.org/programs/sura_grid_join.html. If you have comments or suggestions for this document, contact sg-implementation@sura.org.

Table of Contents

STEP 1: Joining SURAGrid	3
STEP 2: Agreement with SURAGrid Policies.....	3
STEP 3: Installing a SURAGrid-compatible Software Stack.....	3
SURAGrid Stack.....	3
Resource Manager.....	4
Installation.....	4
STEP 4: The SURAGrid Trust Fabric.....	5
Cross-certification	5
Cross-certify your site CA with the SURAGrid BCA	5
If your institution does not have a CA	5
Copy other certificate pairs to your resource.....	6
Establishing trust when cross-certification is not an option.....	7
STEP 5: Establishing User Accounts	7
SURAGrid accounts for local users	7
Users at other SURAGrid sites.....	8

Create Unix Accounts for SURA Grid Users	8
Create Home Directories for SURAgrid Users.....	8
Map SURAgrid Users to Unix Accounts.....	8
Synchronizing Passwords for SURAgrid Users.....	9
Update and Maintain Mappings of SURAgrid Users to Unix Accounts.....	9
STEP 6: Configuration and Verification of Basic Grid Services	9
Security Services	10
Firewall Settings.....	10
Configuration for SURAgrid.....	10
MyProxy services	11
Data Services	11
GridFTP	11
Job Management	11
Web Services	12
Webservice GRAM job submission	12
Pre-WS services.....	13
Pre-WS Gatekeeper	13
Pre-WS MDS	14
Pre-WS MDS configuration	14
Information Services	14
STEP 7: Configuring the SURAgrid Environment	14
STEP 8: Adding Your Resource to the SURAgrid Portal	15
SURAgrid Resource Data Collection Form	15
Install GPIR Resource Monitor Code	15
Enable automated resource verification via SURAgrid test scripts.....	15
Verify the Accessibility of Your Resource Across SURAgrid	16
STEP 9: Ongoing Verification and Support	16
Support of your SURAgrid resource	16
Support for resource owners	16
Acknowledgements	17
Appendix A: Example grid-info-search command results.....	18

STEP 1: Joining SURAgrid

Those institutions that contribute resources to SURAgrid are joining SURAgrid by default and will be added to the published list of SURAgrid participants. If your institution is not already listed as a participant at http://www.sura.org/programs/sura_grid.html, send an introductory email to the SURAgrid Project Manager, Mary Fran Yafchak <maryfran@sura.org>. If your institution is already participating, send an email to <sg-implementation@sura.org> and include an administrative contact (name, email, phone) for each resource to be added. All administrative contacts listed will subsequently be subscribed to the SURAgrid support listserv (suragrid-support@sura.org) for ongoing project coordination and peer support.

STEP 2: Agreement with SURAgrid Policies

The contribution and use of SURAgrid resources should be in alignment with the SURAgrid Acceptable Use Policy (AUP), available at http://www.sura.org/programs/sura_grid_aup.html. Note that SURAgrid users agree to observe the SURAgrid acceptable use policy, and also the acceptable use policies of the individual resources and connections that they are accessing through SURAgrid.

STEP 3: Installing a SURAgrid-compatible Software Stack

SURAgrid Stack

To accommodate heterogeneity, the SURAgrid software stack, grid services and application environment evolve based on setting a minimal set of requirements and recommendations that increase in specificity as needs dictate. Specifications are intended to be as loose possible while providing a basic level of interoperability. The current SURAgrid stack specification is:

- Required: Globus 4.x, with pre-WS for GRAM and gridFTP. Web services are also recommended for GRAM and RFT. GSI-OpenSSH is recommended for application staging
- Any version of operating system that supports the required services above, with Linux¹ 2.4 or higher recommended in order to provide a common platform for application development
- Addition of resource and relevant system detail to the resource monitor (GPIR²) of the SURAgrid portal
- A scheduler installed as part of your underlying resource configuration
- Cross-certification with SURAgrid Bridge CA - strongly recommended at this time and likely to be required in the future. (See <https://www.pki.virginia.edu/nmi-bridge>)
- Configuration of the required environment variables as defined in SURAgrid Environment Variables (http://www.sura.org/programs/SURAgrid_EnvVar.htm)
- Configuration of the optional environment variables also recommended.

¹ <http://www.linux.org/>

² <http://www.tacc.utexas.edu/projects/gpir.php>

Note 1: Different applications may dictate requirements beyond these, which will be treated as application-specific requirements until shown to be a more common need that should be moved to overall resource requirements.

Note 2: The SURAGrid stack is occasionally updated. To ensure you're working with the correct stack requirements, please see http://www.sura.org/programs/sura_grid_join.html for the current specification.

Resource Manager

Though optional, it is strongly recommended you install and configure a resource manager if your resource is a cluster. It is best to choose both a resource manager and a scheduler while choosing the software stack to install on the SURAGrid resource. The resource manager gathers information about resources (# of processors, memory, disk space, etc), submits jobs, manages and cleans up after jobs. A scheduler is a critical service and typically implements a good algorithm for resource allocation (best possible utilization). While in some packages (e.g, SGE³) the resource manager and scheduler are packaged together, for others (e.g., PBS⁴) you'll need to install a scheduler (e.g., MAUI⁵, MOAB⁶) separately.

There are many open source options available for installing resource managers and schedulers. For example:

- LSF⁷
- Condor⁸
- SGE
- PBS
- OpenPBS⁹ (Torque + MAUI)
- Altair® PBS Professional¹⁰™ 7.1 is also available as a free educational grant for academic, non-commercial use only
- Rocks¹¹ distributes SGE and OpenPBS rolls that can conveniently be deployed during or after cluster installation

Installation

SURAGrid does not offer a pre-packaged software installation bundle at this time. SURAGrid participants have used the following packages for installation of Globus:

Installing on an existing cluster or compute resource:

- NMI GRIDS Center Software Suite, <http://www.nsf-middleware.org/Lists/NMIR8/grids.aspx> (source code installation)
- Virtual Data Toolkit (VDT), <http://vdt.cs.wisc.edu> (pre-packaged installer)

³ <http://gridengine.sunsource.net/>

⁴ http://www.altair.com/software/pbs_abo.htm

⁵ <http://www.clusterresources.com/pages/products/maui-cluster-scheduler.php>

⁶ <http://www.clusterresources.com/pages/products.php>

⁷ <http://www.platform.com/Products/Platform.LSF.Family/Platform.LSF/>

⁸ <http://www.cs.wisc.edu/condor/>

⁹ <http://www.openpbs.org/about.html>

¹⁰ http://www.altair.com/software/pbs_edu1.htm

¹¹ <http://www.rocksclusters.org/>

Building a new cluster in conjunction with Globus installation:

- ROCKS, with Grid roll, http://www.rocksclusters.org/wordpress/?page_id=3, (installs full cluster, then grid services based on the NMI GRIDS software suite) (See the Note under the “Pre-WS MDS configuration” topic in Step 6’s “Pre-WS Services” for information regarding known issues.)

Note 1: An alternate approach used is to use ROCKS to install the cluster software, then VDT for grid services.

Note 2: SURAggrid resource owners should check SURAggrid support reference sources to view the information SURAggrid members have contributed regarding compatibility issues in SURAggrid stack and other grid software products. See Step 9 “Support for resource owners” for details on these information sources.

STEP 4: The SURAggrid Trust Fabric

In order for an application to run on any given SURAggrid resource, the user(s) that will initiate the application must be authenticated (identity verified) and authorized (accounts and permissions established) to use that resource. For authentication, the Globus Toolkit relies on PKI (public key infrastructure) and its related exchange of certificates, while shared authorization in Globus uses the grid-mapfile and associated system accounts.

Cross-certification

There are a variety of ways to obtain and exchange certificates to provide user authentication across grid resources. The SURAggrid authentication model is to leverage authoritative campus identity management for scalable exchange of trust information through a Bridge Certificate Authority (Bridge CA). Within SURAggrid, participating sites typically run their own Certificate Authority (CA) to provide both user and system certificates; these certificates are then cross-certified via the Bridge CA (BCA). Cross-certification between participating SURAggrid sites also supports the use of SURAggrid user accounts for accessing SURAggrid resources. If you need assistance with CA matters, please send an email message to [<sg-implementation@sura.org>](mailto:sg-implementation@sura.org).

Cross-certify your site CA with the SURAggrid BCA

Basic instructions for cross-certifying your site CA with the SURAggrid BCA are available from the University of Virginia, lead for development and maintenance of the SURAggrid Bridge CA, at <https://www.pki.virginia.edu/sura-bridge/>. The page explains how to download and sign the BCA's certificate request and how to generate a certificate request from your CA and send it to the BCA.

If your institution does not have a CA

There are instances when a new participating SURAggrid institution doesn't have a CA on their campus to use to cross-certify their site/resource. For instance:

- The institution has no CA in place.
- The institution has a CA, but its use is restricted and not available for use and cross-certification with the SURAggrid BCA.

In instances such as these, it is recommended that the SURAggrid representatives at the participating institution create a new CA that can be used to cross-certify their site/resource with the SURAggrid BCA. There is an automated tool available (the simpleCA Bundle) from SURAggrid that simplifies the creation of a new CA. Please see the instructions item 3 of “Download Software” at: <https://www.pki.virginia.edu/sura-bridge/scl/>.

Copy other certificate pairs to your resource

When a SURAggrid participant site cross-certifies their CA with the SURAggrid BCA, a certificate pair is created that enables the participant site to trust certificates from the other SURAggrid sites using only their own root as a trust anchor. The BCA stores the cross-certificate pair of each SURAggrid participant that completes the cross-certification process in the SURAggrid LDAP Directory Server. When a SURAggrid user logs into a SURAggrid resource using their single sign-on SURAggrid-enabled certificate, the Globus installations at all of the other participating sites are able to validate the user’s certificate leveraging the Bridged PKI infrastructure.

The SURAggrid BCA and the certificate pairs it stores create a trust fabric in SURAggrid that solves the N^2 problem of authenticating multiple users between multiple institutions. In the absence of the BCA, each participating SURAggrid site would need to establish mutual trust with every other site through the exchange of its certificate with all other SURAggrid sites. This would result in the N^2 problem (illustrated in Figure 1) where every institution of N institutions would exchange a certificate with every other of the N institutions, resulting in a total of $N \times N$ exchanges.

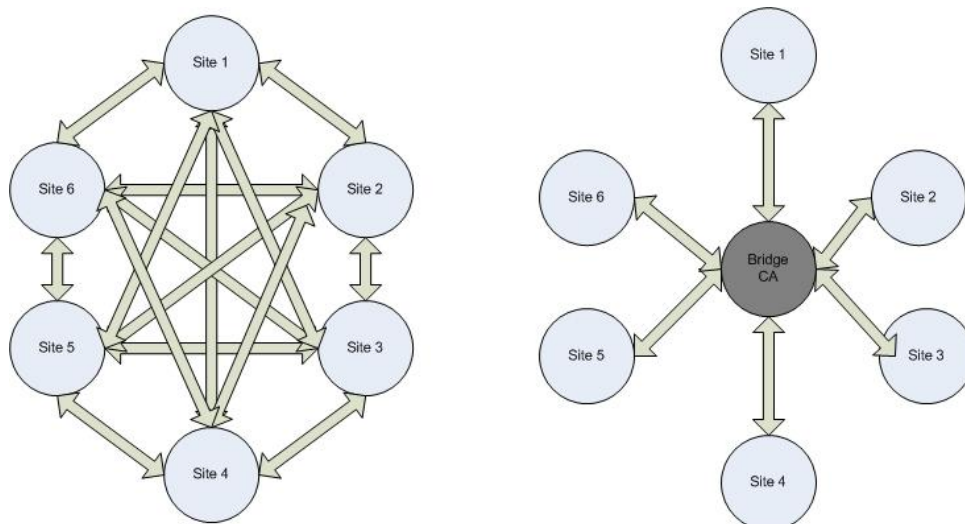


Figure 1. Illustration of the N^2 Problem

While the BCA solves the N^2 problem, the PKI path validation logic that Globus uses (based on OpenSSL¹²) is not bridge-aware which, in the standard implementation of Globus with a bridge, means cross-certificate pairs and signing policy files need to be manually pre-loaded on each bridged resource (see, for example, the list of stored certificates on the SURAggrid BCA at: <https://www.pki.virginia.edu/sura-bridge/certs/>.)

¹² <http://www.openssl.org/>

Fortunately, SURAGrid leverages the SURAGrid Directory server and associated tools (e.g., the `bridge-cert.pl` script) to eliminate the need for such hand configuration¹³. This script will install all the stored bridge certificates onto your resource and automatically generate the appropriate Globus Signing Policy files. This tool can be downloaded from: <https://www.pki.virginia.edu/sura-bridge/scl/#bridgecert>. As noted in the instructions on this page, you need to edit the script before running it and run the script periodically via a cron job so that your bridge certificates remain current.

Establishing trust when cross-certification is not an option

If cross-certifying a new or existing institutional CA with the SURAGrid Bridge CA is simply not practical, but your institution does have a CA and can issue user and server certificates for use with your SURAGrid work, mutual trust can still be established for access to SURAGrid resources. This can be done through the direct exchange of certificates and signing policies between the new SURAGrid resource site and all other participating SURAGrid sites. To enable access to your resource via the SURAGrid Portal, exchange of certificates with SURAGrid Portal's MyProxy¹⁴ server(s) is needed as well. See the "Using the Portal" subtab on the Documentation tab of the [SURAGrid portal](#) to view the current list of SURAGrid's MyProxy servers.

STEP 5: Establishing User Accounts

There are two types of SURAGrid user accounts¹⁵ which should be established on your resource:

- SURAGrid accounts for local user(s) – While your site may not have users that need access to SURAGrid resources, you will at least want to have one user from your site setup as a SURAGrid user so that you can initially configure, test and properly maintain your SURAGrid resource in the future.
- Accounts for application users from other SURAGrid sites – The resources contributed by SURAGrid participants are considered available to all SURAGrid application users when the application's system requirements can be accommodated. (Finer-grained policy may be required as traffic and contention increase, as application-specific scenarios arise or as local policy requires.)

SURAGrid accounts for local users

SURAGrid accounts are created and maintained via the SURAGrid Central User Authentication and Authorization System tool at: <https://www.pki.virginia.edu/suragrid/>, which creates and maintains user entries in the SURAGrid LDAP directory (see the example user entry at: <http://www.people.virginia.edu/~7Ejaj/sura/20060922/ldap-person.gif>). Each participating SURAGrid institution assigns a SURAGrid User Administrator that is authorized to use this SURAGrid Central User Authentication and Authorization System tool. Typically, the SURAGrid Resource Administrator is also the User Administrator, but SURAGrid does not require this arrangement. To request your site's SURAGrid User Administrator account, send an email to sg-implementation@sura.org.

¹³ Sites may still opt to manually configure certificates on their resource. See: <https://www.pki.virginia.edu/sura-bridge/#GlobusConfig>

¹⁴ <http://grid.ncsa.uiuc.edu/myproxy/>

¹⁵ SURAGrid users have Unix loginids of the form *sitename-username*, such as *uva-jaj*. The Unix uid numbers start at 1,000,000 to avoid conflicts with local uid numbers.

Users at other SURAgrid sites

Globus assumes that standard Unix user accounts exist on all systems including account password file entries (e.g., Name, UID, GID, shell) and home directories. Globus maps PKI Subject Distinguished Names (DNs) to local Unix IDs. The web page:

<https://www.pki.virginia.edu/sura-bridge/scl/> explains the steps necessary to manually create Unix accounts for all SURAgrid users and map SURAgrid user accounts to Unix accounts.

However, these mappings are maintained in the grid-mapfile on each Globus gatekeeper and the complexity of the DN naming convention makes manual maintenance of the grid-mapfile cumbersome (at best) and problematic (e.g., likeliness of typo's, scalability issues with dozens of sites and large numbers of researchers and students). Therefore, this web page also explains several automation tools provided by SURAgrid that simplify the creation of Unix accounts and the required home directories for all SURAgrid users. In addition, it provides several levels of automation that a site can implement to keep their grid-mapfile mappings of SURAgrid user accounts to Unix accounts current.

The steps involved in SURAgrid user account management and the related automation tools are summarized below.

Create Unix Accounts for SURA Grid Users

You need to add all the SURA grid users to your local Unix user database. The SURAgrid LDAP server contains posixAccount schema data. To view the discussion of how you can automate replacement of Unix password files via nsswitch.conf, see: <https://www.pki.virginia.edu/sura-bridge/scl/#adduser>.

Create Home Directories for SURAgrid Users

The bulleted section entitled “Create Home Directories¹⁶ for SURAgrid Users” at <https://www.pki.virginia.edu/sura-bridge/scl/#homedir> discusses how to use the homedir.pl Perl script to create home directories for SURAgrid users (note that you must edit this script before running it; see the comments in the script for instructions.)

You will need to issue certificates to each of *your site's users* created by homedir.pl and create a .globus subdirectory under the directory created by homedir.pl (since all globus commands expect the directory created by homedir.pl to have the user's certificate.) *Users from other sites* will use the certificate credentials issued by their home site. For example, “/home/username/.globus” is the original directory set up with user certificate and “home/xxx-username (where xxx = the acronym for the user's institution)” is the directory the script homedir.pl creates. You will need to copy the .globus directory where the user certificate is stored to the “home/xxx-username” directory. Note that the institutional acronyms in SURAgrid range between three to five characters, depending on the institution.

Map SURAgrid Users to Unix Accounts

SURAgrid provides a Globus module called “ldap_authz_callout” (commonly referred to as simply the “LDAP Callout”) that augments your grid-mapfile by looking up a SURAgrid user by

¹⁶ Home directories have the form /home/loginid.

subject DN on the SURA grid LDAP server to obtain the corresponding Unix login id. The bulleted section entitled “Map SURA Grid Users to Unix Accounts” at <https://www.pki.virginia.edu/sura-bridge/scl/#gridmapfile> explains how to download the callout source and provides related installation instructions. The scripts require the Open::LDAP module; use the commands “cpan>install Net::SSLeay”, then “cpan>install Net::LDAP”. You must compile this NMI-based source code in your Globus environment.

If you would prefer not to use ldap_authz_callout, you can use the grid-mapfile.pl script to add the user mappings in the SURAggrid LDAP server to your grid-mapfile. You must edit the script before running it. See the comments in the script for instructions.

Synchronizing Passwords for SURAggrid Users

The SURAggrid stack recommendation for GSI-SSH on all SURAggrid resources and the SURAggrid LDAP Directory Server schema allow SURAggrid user to access all SURAggrid resources¹⁷ with just their SURAggrid account. To configure your Linux system to support SURAggrid account/password access to your resource, configure your LDAP to augment your local /etc/passwd file by editing /etc/nsswitch.conf file and replacing the line that begins with “passwd:” with this line “passwd: files ldap”.

Grid users authenticate to Globus using certificates, so they do not need Unix passwords. Currently the LDAP server has no Unix passwords for SURA grid users, so there is no need to augment /etc/shadow (which stores Unix passwords) with LDAP.” (see <https://www.pki.virginia.edu/sura-bridge/scl/#adduser>)

Update and Maintain Mappings of SURAggrid Users to Unix Accounts

Whether a site uses the LDAP Callout or chooses the grid-mapfile.pl script, the mappings stored in the grid-mapfile on the SURAggrid directory server need to be kept synchronized with the mappings on all SURAggrid resources. While use of the LDAP Callout should mean resource sites do not need to create or manage the grid-mapfile for true Globus applications, unfortunately at present GSISSH (GSI-OpenSSH)¹⁸ only uses the grid-mapfile. Therefore, in order to allow SURAggrid users to run GSISSH on SURAggrid resources, both the site administrators that use the LDAP Callout and those that choose the grid-mapfile.pl script need to install and run this script periodically with a cron job.

STEP 6: Configuration and Verification of Basic Grid Services

Before you add your resource to the SURAggrid Portal, you need to configure and verify the functionality of its basic grid services using the information below.

¹⁷ There are two situations in which SURAggrid users have to use a local account to access a site’s SURAggrid resource: 1) To access resources at SURAggrid sites that are cross-certified (allowing them to take advantage of the BCA for certificate exchange) but has local policies that require local accounts be used for resource access, 2) 1) To access resources at SURAggrid sites that are not cross-certified.

¹⁸ GSI-SSH is currently a strongly recommended component of the SURAggrid stack (see Step 3) and is likely to become required in future stack updates

Security Services

Firewall Settings

If your resource is behind a firewall, make sure that the Globus ports (globus-gatekeeper and gsiftp) can be accessed through the firewall¹⁹. Only a specific range of ports needs to be opened.

A separate document that more fully addresses this important topic is being developed by SURAGrid participants. Please check the SURAGrid website: http://www.sura.org/programs/sura_grid_join.html to determine if the supplemental firewall document is available for download. If the document is unavailable or you need more information, send an email to sg-implementation@sura.org.

Configuration for SURAGrid

Pre-WS GRAM – this item should be configured.

Pre-WS gatekeeper (default TCP port 2119) – this port needs to be open for job submission and so that the SURAGrid Portal can get information about your resource. For the latter, a job manager uses a port in a specified range to query your resource. For this purpose, a port range can be specified using the Globus environment GLOBUS_TCP_PORT_RANGE (see Section 4 <http://www.globus.org/toolkit/security/firewalls/>). If your site's local security policies prevent this, the next best thing might be to open the port to a selected set of machines (i.e., all SURAGrid resources)

An additional set of ports – needs to be opened so that Globus can use them for things such as the GRAM job manager and gridFTP callbacks.

The GSI-OpenSSH service - is like regular SSH (similarly, GSI-SCP is like the regular SCP service). However, there is one difference –the GSI-version of these services can use the grid security infrastructure to provide authentication. By enabling GSI-SSH and providing SURAGrid with the port number you're using, you ensure that an authenticated SURAGrid user will not be prompted for a password when using GSI-SSH to access your resource (see “Step 5: User Accounts” sections “Synchronizing Passwords for SURAGrid Users” and “Update and Maintain Mappings of SURAGrid Users to Unix Accounts” for more detail.)

You will need to start GSI-SSH on a separate port²⁰. SURAGrid default ports selections are 22 and 2222. However, you may choose to open another port for GSI-SSH if your site has local reasons that make ports 22 or 2222 unworkable for you. To make GSI-SSH listen on a different port, add the following line to <SXXsshd command>:

(copied \$GLOBUS_LOCATION/sbin/SXXsshd as /etc/rc.d/init.d/SXXsshd)

Start the server with `$/etc/rc.d/init.d/SXXsshd start`

#

SSHD arguments can be added here within the following

¹⁹ Globus provides reference information at: <http://www.globus.org/toolkit/security/firewalls/>

²⁰ When you submit your resource's information for publication in the SURAGrid portal, there will be a field in which you will enter the port number you have configured GSI-SSH on.

```
# set of double quotes.  
#  
SSHD_ARGS="-p XXXX" (where XXXX=port #)
```

MyProxy services

The SURAggrid Portal uses MyProxy²¹ services to manage SURAggrid user credentials and enable them to be authenticated on SURAggrid resources. The SURAggrid portal has a current list of SURAggrid MyProxy servers²². Refer to the “Using the Portal” page of the Documentation tab in the SURAggrid Portal for information on how to upload and manage MyProxy credentials, submit and manage jobs, and transfer files using the SURAggrid Portal.

Note: The machine that you run the myproxy-init from should first be cross-certified with the SURAggrid BCA and should have all the necessary files copied into the /etc/grid-security/certificates directory (see “Cross-certify your site with the SURAggrid BCA” in Step 4). If your site does not have SURAggrid Bridge cross-certification, send the signing_policy and .0 files for the CA that issued your host certificate to the MyProxy server’s resource administrator and ask them to install the files on their MyProxy server (see “Establishing trust when cross-certification is not an option” in Step 4).

Data Services

GridFTP

To verify that gridFTP is setup correctly, test your resource’s ability to move files on SURAggrid by using a utility such as globus-url-copy against SURAggrid’s Bandera resource (or any operational SURAggrid host listed on the portal).

Example:

```
globus-url-copy file://bandera.tacc.utexas.edu/tmp/file1  
gsiftp://bandera.tacc.utexas.edu/tmp/file2
```

Note: Ensure the user running the above commands has their .globus directory properly configured (see “Create Home Directories for SURAggrid Users” in Step 5.)

RFT

Configuration of this service is optional.

Replica Location Services

Configuration of this service is optional.

Job Management

It is strongly recommend that you set up a local resource manager and a scheduler, especially if your resource is a compute cluster or a symmetric multiprocessor. The section below includes a discussion of how to configure your resource to accept job requests from SURAggrid users. While there are many ways of submitting jobs, from basic to advanced, we discuss some fairly common means of job submission.

²¹ More information about MyProxy can be found at <http://grid.ncsa.uiuc.edu/myproxy/> and www.globus.org/alliance/publications/papers/myproxy.pdf

²² <https://gridportal.sura.org/gridsphere/gridsphere?cid=documentation&JavaScript=enabled>

Web Services

Before proceeding with the instructions below, ensure RFT has been configured for job staging.

Webservice GRAM job submission

Please refer to the WS GRAM Admin Guide for details on configuring WS GRAM services:

<http://www.globus.org/toolkit/docs/4.0/admin/docbook/ch11.html>

You can test if WS GRAM was correctly setup with the following commands:

```
choate % globusrun-ws -submit -c /bin/true
Submitting job...Done.
Job ID: uuid:3304e3f2-55f2-11da-8b8f-00d0b7b7c0bc
Termination time: 11/16/2005 16:09 GMT
Current job state: Active
Current job state: CleanUp
Current job state: Done
Destroying job...Done.
choate % echo $?
0
choate % globusrun-ws -submit -c /bin/false
Submitting job...Done.
Job ID: uuid:456b7c9a-55f2-11da-9b0d-00d0b7b7c0bc
Termination time: 11/16/2005 16:09 GMT
Current job state: Active
Current job state: CleanUp
Current job state: Done
Destroying job...Done.
choate % echo $?
1
```

Please refer to section 3 of the Globus document “GT 4.0 WS GRAM: User's Guide”^{*} for more details about various usage scenarios and to section 3.6 specifically for additional information about configuring your local resource manager with WS GRAM.

^{*} Available from <http://www.globus.org/toolkit/docs/4.0/execution/wsgram/user-index.html#s-wsgram-user-usagescenarios>

See section 3.5.2. “Local resource manager configuration” and section 3.9. “WS GRAM auto-registration with default WS MDS Index Service” in the Globus document “GT 4.0 WS GRAM : System Administrator's Guide”^{**}

^{**} Available from <http://www.globus.org/toolkit/docs/4.0/execution/wsgram/admin-index.html#s-wsgram-admin>

Information about registering with mds-servicegroup-add and the Index Service is also available from Globus.org and can be found at:

<http://www.globus.org/toolkit/docs/4.0/info/aggregator/re01.html#mds-servicegroup-add-registering>

Note that you will need to edit the file:

```
$GLOBUS_LOCATION/etc/globus_wsrf_mds_aggregator/example-aggregator-  
registration.xml
```

to represent your local resource (in particular, edit the hostname and resource manager name.)

Here are some quick links to the commands:

globusrun-ws - Official job submission client for WS GRAM

<http://www.globus.org/toolkit/docs/4.0/execution/wsgram/rn01re01.html>

managed-job-globusrun - (DEPRECATED) Java-based job submission client for GRAM

<http://www.globus.org/toolkit/docs/4.0/execution/wsgram/rn01re02.html>

globus-job-run-ws - Interactive job submission script for WS GRAM

<http://www.globus.org/toolkit/docs/4.0/execution/wsgram/rn01re03.html> (requires globus_wsrf_gram_client_tools update package from <http://www.globus.org/toolkit/downloads/development/>)

globus-job-submit-ws - Batch job submission script for WS GRAM

<http://www.globus.org/toolkit/docs/4.0/execution/wsgram/rn01re04.html> (requires globus_wsrf_gram_client_tools update package from <http://www.globus.org/toolkit/downloads/development/>)

globus-job-get-output-ws - Job output fetch script for WS GRAM

<http://www.globus.org/toolkit/docs/4.0/execution/wsgram/rn01re05.html> (requires globus_wsrf_gram_client_tools update package from <http://www.globus.org/toolkit/downloads/development/>)

globus-job-clean-ws - Destroy and clean up a batch job for WS GRAM

<http://www.globus.org/toolkit/docs/4.0/execution/wsgram/rn01re06.html> (requires globus_wsrf_gram_client_tools update package from <http://www.globus.org/toolkit/downloads/development/>)

Pre-WS services

Pre-WS Gatekeeper

Install and configure Globus Gatekeeper (port 2119)

Test if the gatekeeper is running with the following command:

```
globusrun -a -r <suragrid-hostname>
```

For example:

```
[16:55] mileva:~]globusrun -a -r mileva.hpc.odu.edu
```

```
GRAM Authentication test successful
```

```
[16:55] mileva:~]
```

Next, test job submissions via the job manager by submitting a job request to your resource using the command:

```
globus-job-submit <suragrid-hostname> -np 1 /bin/date
```

For example:

```
[17:39] mileva:~]globusrun -o -r mileva.hpc.odu.edu/jobmanager-pbs  
'&(executable="/bin/echo") (arguments="Grid Status Test")'  
Grid Status Test  
[17:40] mileva:~]
```

In order to determine which Job manager is in use, one can use `grid-info-search` command. See Appendix A for example results from a SURAgrid resource.

Pre-WS MDS

Note on MDS query: Although you can query a different MDS server than the host of the GRAM and GridFTP servers, it is typically the MDS running on the same node as the services that is queried. An example query is:

```
grid-info-search -anonymous -L -h myresource.myinstitution.org --nowrap
```

Pre-WS MDS configuration

There are two methods that can be used to configure pre-WS MDS. One is to provide anonymous (and perhaps less secure) queries, while the second is to provide LDAP-based (and perhaps more secure) queries. When using the LDAP option, the `-pre-ws-mds` configure switch must be used during Globus installation in order to generate some LDAP certificates. However, there is a security warning from Globus Alliance noting that pre-WS MDS uses port 2135 as the default port. Some SURAgrid sites (e.g., ODU) have therefore restricted this service (port) to certain SURAgrid users' machines (e.g., SCOOP/RENCI for the ADCIRC application²³); this method allows the application to be supported but should not compromise your system's security.

Note: There is a known issue in **Rocks 4.1 Grid Roll** that causes `grid-info-search` hangs. This problem is limited to using the pre-built binaries contained in the Grid Roll 4.0.1 (it is unclear whether the problem extended to 4.2 as well). One workaround is of course to not use the Grid Roll and to instead build Globus from source or use the `globus.org` binary download. However since the Grid Roll makes for a quick initial code load, another option is to still use the Grid Roll and to then go back and set each service separately. For more current/peer advice, you can send an email to the SURAgrid support listserv (SURAgrid-support@sura.org).

Information Services

- Pre-WS (MDS) or WS monitoring services (GPIR, etc)
- Integration of pre-WS MDS and/or GPIR with Resource Manager (PBS, SGE, etc.)

To integrate pre-WS MDS with PBS, see the reference material at:

<http://calclab.math.tamu.edu/grid/ADCIRC.xhtml>

Also, if your resource is firewalled, make sure you read the Security Services section above.

STEP 7: Configuring the SURAgrid Environment

Part of the SURAgrid stack specification is the configuration of at least the required environment variables as defined in SURAgrid Environment Variables document. Version 1 of the document lists both the required and recommended variables to be set on all SURAgrid resources. The specification is available at http://www.sura.org/programs/SURAgrid_EnvVar.htm.

²³ For more information about ADCIRC, see http://www.sura.org/programs/docs/sura_grid_adcirc_sep_05.pdf

STEP 8: Adding Your Resource to the SURAGrid Portal

SURAGrid's resources can be monitored and accessed through the SURAGrid Portal (<http://gridportal.sura.org>). Please follow the instructions below to add your resource to the SURAGrid Portal and to allow the SURAGrid to run test scripts periodically to verify the operability of your resource's required grid services. If you need assistance while working through the steps below, send an email to PortalAdmin@sura.org.

SURAGrid Resource Data Collection Form

Fill out and submit the "SURAGrid Resource Data Collection Form" form at: <http://sura.org/programs/RDCform.html>. Your resource's details are published in various places in the SURAGrid portal and are also used for SURAGrid technical and administrative support purposes.

Install GPIR Resource Monitor Code

Install the remote GPIR code for use with SURAGrid's GPIR resource monitor (this code gathers data about your resource (e.g., job, load) that is returned to and published in the SURAGrid Portal):

- Download the GPIR Service containing the remote provider code from <http://gridport.net/services/gpir/gpir-download.html>
- Install and configure the remote provider code using the instructions from <http://gridport.net/services/gpir/providers.html>

Note 1: It is not necessary to install the full GPIR service contained in the downloaded package. Only a subset (the GPIR provider) needs to be installed.

Note 2: For step six in the configuration instructions, use the following values:

- `hostname=yourHostname`
- `gpir.contact=cuero.tacc.utexas.edu:12080/gpir/webservices`
- `admin.email=yourAdminEmailAddress`

Enable automated resource verification via SURAGrid test scripts

SURAGrid's GPIR scripts run periodically during each hour under the SGmonitor account. This script tests the following on your SURAGrid resource:

- Gatekeeper authentication
- GSIFTP Server availability
- GSISSH Server operation
- Grid Programs in PATH
- Simple Test of jobmanager
- GSIFTP Job Test of jobmanager

The test script runs using a certificate issued by TACC and under a user account of SGmonitor, which must be established on each resource. To enable the test script to run:

- Refer to Step 4 for instructions on use of certificates between SURAGrid sites. If you are ready to verify your resource but cross-certification is still in planning or progress for your site, you may enable access through exchange of your CA's .0 and signing_policy files directly with TACC by sending them to PortalAdmin@sura.org.

- Refer to Step 5 for instructions on establishing SURAggrid user accounts in general. Specifically, ensure that your grid-mapfile²⁴ allows SGmonitor access by including the following authorized account information:
 User: SuraGrid Monitor
 DN: /C=US/O=UTAustin/OU=TACC/CN=SuraGrid Monitor/UID=sgmonitor

Verify the Accessibility of Your Resource Across SURAggrid

In order to verify your resource has been properly added to the SURAggrid portal, that it's fully accessible to SGmonitor for testing and that SURAggrid users can run jobs on it:

- Find your machine in the Resource Monitor subtab of the SURAggrid portal's Welcome tab (<https://gridportal.sura.org/gridsphere/gridsphere?cid=resource-monitor&JavaScript=enabled>)
- If your resource shows an "up arrow" indicator under the Current Status field on the Resource Monitor page, send an email to PortalAdmin@sura.org so that your SURA can formally thank you and announce the availability of your resource to SURAggrid users and other resource administrators.
- If your resource does *not* show an "up arrow" indicator under the Current Status field on the Resource Monitor page, click on your resource's (or any other resource's) arrow to see the results of SURAggrid's test scripts. Contact PortalAdmin@sura.org for assistance in troubleshooting test failures.
- In addition, refer to the "Using the Portal" page of the Documentation tab in the SURAggrid Portal for instructions on using the portal (e.g., portal user authorization and login; working with files and jobs.)

STEP 9: Ongoing Verification and Support

Support of your SURAggrid resource

It is understood in SURAggrid that contributed resources are supported at varying levels, in keeping with what each resource owner is able to provide. Your site's support levels and methods for each of your SURAggrid resources should be documented as fully as possible and are published in the SURAggrid portal. As described in Step 8 above, all SURAggrid resources are verified regularly through a script used by the SGmonitor user and the results of these tests are also published for SURAggrid users in the SURAggrid Portal.

Support for resource owners

Support as well as troubleshooting assistance for installation, configuration and integration of SURAggrid resources is available from the SURAggrid implementation team (sg-implementation@sura.org) team and via peer support through the SURAggrid support listserv: SURAggrid-support@sura.org. The list includes threaded and searchable archives for access by subscribers (<http://www.sura.org/archives/suraggrid-support>) and can support subject line queries. Your questions and comments are always welcome and will be used by the SURAggrid team to

²⁴ See Step 5 in this document and <https://www.pki.virginia.edu/sura-bridge/scl/#gridmapfile> to learn how you can configure your grid-mapfile to be automatically updated with the SGmonitor and other SURAggrid user accounts.

modify this document as needed as well as in the analysis of SURAGrid's infrastructure and policies.

Acknowledgements

SURA thanks the SURAGrid implementation team and others that contributed to this document. Of particular note are contributions from Mahantesh Halappanavar of Old Dominion University and Brian Brooks of Kennesaw State University.

Appendix A: Example grid-info-search command results

```
[17:38] mileva:~]grid-info-search -h mileva.hpc.odu.edu -x
version: 2

#
# filter: (objectclass=*)
# requesting: ALL
#

# mileva.hpc.odu.edu, local, grid
dn: Mds-Host-hn=mileva.hpc.odu.edu,Mds-Vo-name=local,o=grid
objectClass: MdsComputer
objectClass: MdsComputerTotal
objectClass: MdsFsTotal
objectClass: MdsHost
objectClass: MdsMemoryRamTotal
objectClass: MdsMemoryVmTotal
objectClass: MdsNet
objectClass: MdsNetTotal
objectClass: MdsOs
Mds-Computer-isa: x86_64
Mds-Computer-platform: x86_64
Mds-Computer-Total-nodeCount: 1
Mds-Fs-freeMB: 1005
Mds-Fs-freeMB: 18979
Mds-Fs-freeMB: 4644
Mds-Fs-freeMB: 485
Mds-Fs-freeMB: 6925
Mds-Fs-freeMB: 8991
Mds-Fs-freeMB: 9281
Mds-Fs-sizeMB: 1005
Mds-Fs-sizeMB: 14762
Mds-Fs-sizeMB: 24861
Mds-Fs-sizeMB: 487
Mds-Fs-sizeMB: 9844
Mds-Fs-Total-count: 11
Mds-Fs-Total-freeMB: 126226
Mds-Fs-Total-sizeMB: 170091
Mds-Host-hn: mileva.hpc.odu.edu
Mds-keepsto: XXXX
Mds-Memory-Ram-freeMB: 384
Mds-Memory-Ram-sizeMB: 2010
Mds-Memory-Ram-Total-freeMB: 384
Mds-Memory-Ram-Total-sizeMB: 2010
Mds-Memory-Vm-freeMB: 4093
Mds-Memory-Vm-sizeMB: 6000
Mds-Memory-Vm-Total-freeMB: 4093
Mds-Memory-Vm-Total-sizeMB: 6000
Mds-Net-addr: XXX.XXX.XXX.XXX
Mds-Net-addr: XXX.XXX.XXX.XXX
Mds-Net-addr: XXX.XXX.XXX.XXX
Mds-Net-addr: XXX.XXX.XXX.XXX
Mds-Net-addr: XXX.XXX.XXX.XXX
Mds-Net-name: eth0
Mds-Net-name: eth1
Mds-Net-name: lo
Mds-Net-name: vmnet1
Mds-Net-name: vmnet8
Mds-Net-netaddr: XXX.XXX.XXX.XXX/X
Mds-Net-netaddr: XXX.XXX.XXX.XXX/24
Mds-Net-netaddr: XXX.XXX.XXX.XXX/24
Mds-Net-netaddr: XXX.XXX.XXX.XXX/24
Mds-Net-netaddr: XXX.XXX.XXX.XXX/24
Mds-Net-Total-count: 5
Mds-Os-name: Linux
Mds-Os-release: 2.6.9-22.ELsmp
Mds-Os-version: 1 SMP Sat Oct 8 21:32:36 BST 2005
Mds-validfrom: 20070109223843Z
```

Mds-validto: 20070109223844Z

memory, mileva.hpc.odu.edu, local, grid

dn: Mds-Device-Group-name=memory, Mds-Host-hn=mileva.hpc.odu.edu, Mds-Vo-name=local, o=grid

objectClass: MdsMemoryRamTotal

objectClass: MdsMemoryVmTotal

objectClass: MdsDeviceGroup

Mds-Device-Group-name: memory

Mds-validfrom: 20070109223844Z

Mds-validto: 20070109223844Z

Mds-keepsto: 20070112173204Z

Mds-Memory-Ram-Total-sizeMB: 2010

Mds-Memory-Ram-Total-freeMB: 384

Mds-Memory-Vm-Total-sizeMB: 6000

Mds-Memory-Vm-Total-freeMB: 4093

Mds-Memory-Ram-sizeMB: 2010

Mds-Memory-Ram-freeMB: 384

Mds-Memory-Vm-sizeMB: 6000

Mds-Memory-Vm-freeMB: 4093

physical memory, memory, mileva.hpc.odu.edu, local, grid

dn: Mds-Device-name=physical memory, Mds-Device-Group-name=memory, Mds-Host-hn=mileva.hpc.odu.edu, Mds-Vo-name=local, o=grid

objectClass: Mds

objectClass: MdsDevice

objectClass: MdsMemoryRam

Mds-Device-name: physical memory

Mds-Memory-Ram-sizeMB: 2010

Mds-Memory-Ram-freeMB: 384

Mds-validfrom: 20070109223844Z

Mds-validto: 20070109223844Z

Mds-keepsto: 20070112173204Z

virtual memory, memory, mileva.hpc.odu.edu, local, grid

dn: Mds-Device-name=virtual memory, Mds-Device-Group-name=memory, Mds-Host-hn=mileva.hpc.odu.edu, Mds-Vo-name=local, o=grid

objectClass: Mds

objectClass: MdsDevice

objectClass: MdsMemoryVm

Mds-Device-name: virtual memory

Mds-Memory-Vm-sizeMB: 6000

Mds-Memory-Vm-freeMB: 4093

Mds-validfrom: 20070109223844Z

Mds-validto: 20070109223844Z

Mds-keepsto: 20070112173204Z

filesystems, mileva.hpc.odu.edu, local, grid

dn: Mds-Device-Group-name=filesystems, Mds-Host-hn=mileva.hpc.odu.edu, Mds-Vo-name=local, o=grid

objectClass: MdsFsTotal

objectClass: MdsDeviceGroup

Mds-Device-Group-name: filesystems

Mds-validfrom: 20070109223844Z

Mds-validto: 20070109223844Z

Mds-keepsto: 20070110043844Z

Mds-Fs-freeMB: 1005

Mds-Fs-freeMB: 18979

Mds-Fs-freeMB: 4644

Mds-Fs-freeMB: 485

Mds-Fs-freeMB: 6925

Mds-Fs-freeMB: 8991

Mds-Fs-freeMB: 9281

Mds-Fs-sizeMB: 1005

Mds-Fs-sizeMB: 14762

Mds-Fs-sizeMB: 24861

Mds-Fs-sizeMB: 487

Mds-Fs-sizeMB: 9844

Mds-Fs-Total-count: 11

Mds-Fs-Total-freeMB: 126226

Mds-Fs-Total-sizeMB: 170091

```
# /export, filesystems, mileva.hpc.odu.edu, local, grid
dn: Mds-Device-name=/export, Mds-Device-Group-name=filesystems, Mds-Host-hn=mileva.hpc.odu.edu, Mds-Vo-name=local, o=grid
objectClass: MdsDevice
objectClass: MdsFs
Mds-Device-name: /export
Mds-Fs-sizeMB: 24861
Mds-Fs-freeMB: 18979
Mds-Fs-mount: /export
Mds-validfrom: 20070109223844Z
Mds-validto: 20070109233844Z
Mds-keepsto: 20070110043844Z
```

```
# /opt, filesystems, mileva.hpc.odu.edu, local, grid
dn: Mds-Device-name=/opt, Mds-Device-Group-name=filesystems, Mds-Host-hn=mileva.hpc.odu.edu, Mds-Vo-name=local, o=grid
objectClass: MdsDevice
objectClass: MdsFs
Mds-Device-name: /opt
Mds-Fs-sizeMB: 14762
Mds-Fs-freeMB: 6925
Mds-Fs-mount: /opt
Mds-validfrom: 20070109223844Z
Mds-validto: 20070109233844Z
Mds-keepsto: 20070110043844Z
```

```
# /scratch, filesystems, mileva.hpc.odu.edu, local, grid
dn: Mds-Device-name=/scratch, Mds-Device-Group-name=filesystems, Mds-Host-hn=mileva.hpc.odu.edu, Mds-Vo-name=local, o=grid
objectClass: MdsDevice
objectClass: MdsFs
Mds-Device-name: /scratch
Mds-Fs-sizeMB: 9844
Mds-Fs-freeMB: 9281
Mds-Fs-mount: /scratch
Mds-validfrom: 20070109223844Z
Mds-validto: 20070109233844Z
Mds-keepsto: 20070110043844Z
```

```
# /tmp, filesystems, mileva.hpc.odu.edu, local, grid
dn: Mds-Device-name=/tmp, Mds-Device-Group-name=filesystems, Mds-Host-hn=mileva.hpc.odu.edu, Mds-Vo-name=local, o=grid
objectClass: MdsDevice
objectClass: MdsFs
Mds-Device-name: /tmp
Mds-Fs-sizeMB: 9844
Mds-Fs-freeMB: 8991
Mds-Fs-mount: /tmp
Mds-validfrom: 20070109223844Z
Mds-validto: 20070109233844Z
Mds-keepsto: 20070110043844Z
```

```
# /home/thorpe, filesystems, mileva.hpc.odu.edu, local, grid
dn: Mds-Device-name=/home/thorpe, Mds-Device-Group-name=filesystems, Mds-Host-hn=mileva.hpc.odu.edu, Mds-Vo-name=local, o=grid
objectClass: MdsDevice
objectClass: MdsFs
Mds-Device-name: /home/thorpe
Mds-Fs-sizeMB: 24861
Mds-Fs-freeMB: 18979
Mds-Fs-mount: /home/thorpe
Mds-validfrom: 20070109223844Z
Mds-validto: 20070109233844Z
Mds-keepsto: 20070110043844Z
```

```
# /home/uab-afgane, filesystems, mileva.hpc.odu.edu, local, grid
dn: Mds-Device-name=/home/uab-afgane, Mds-Device-Group-name=filesystems, Mds-Host-hn=mileva.hpc.odu.edu, Mds-Vo-name=local, o=grid
objectClass: MdsDevice
```

```

objectClass: MdsFs
Mds-Device-name: /home/uab-afgane
Mds-Fs-sizeMB: 24861
Mds-Fs-freeMB: 18979
Mds-Fs-mount: /home/uab-afgane
Mds-validfrom: 20070109223844Z
Mds-validto: 20070109233844Z
Mds-keepsto: 20070110043844Z

# networks, mileva.hpc.odu.edu, local, grid
dn: Mds-Device-Group-name=networks, Mds-Host-hn=mileva.hpc.odu.edu, Mds-Vo-name
=local, o=grid
objectClass: MdsNetTotal
objectClass: MdsNet
objectClass: MdsDeviceGroup
Mds-Device-Group-name: networks
Mds-validfrom: 20070109223844Z
Mds-validto: 20070110043844Z
Mds-keepsto: 20070110043844Z
Mds-Net-addr: XXX.XXX.XXX.XXX
Mds-Net-addr: XXX.XXX.XXX.XXX
Mds-Net-addr: XXX.XXX.XXX.XXX
Mds-Net-addr: XXX.XXX.XXX.XXX
Mds-Net-addr: XXX.XXX.XXX.XXX
Mds-Net-name: eth0
Mds-Net-name: eth1
Mds-Net-name: lo
Mds-Net-name: vmmnet1
Mds-Net-name: vmmnet8
Mds-Net-netaddr: XXX.XXX.XXX.XXX/8
Mds-Net-netaddr: XXX.XXX.XXX.XXX/24
Mds-Net-netaddr: XXX.XXX.XXX.XXX/24
Mds-Net-netaddr: XXX.XXX.XXX.XXX/24
Mds-Net-netaddr: XXX.XXX.XXX.XXX/24
Mds-Net-Total-count: 5

# eth0, networks, mileva.hpc.odu.edu, local, grid
dn: Mds-Device-name=eth0, Mds-Device-Group-name=networks, Mds-Host-hn=mileva.h
pc.odu.edu, Mds-Vo-name=local, o=grid
objectClass: MdsDevice
objectClass: MdsNet
Mds-Device-name: eth0
Mds-Net-name: eth0
Mds-Net-netaddr: XXX.XXX.XXX.XXX/24
Mds-Net-addr: XXX.XXX.XXX.XXX
Mds-validfrom: 20070109223844Z
Mds-validto: 20070110043844Z
Mds-keepsto: 20070110043844Z

# eth1, networks, mileva.hpc.odu.edu, local, grid
dn: Mds-Device-name=eth1, Mds-Device-Group-name=networks, Mds-Host-hn=mileva.h
pc.odu.edu, Mds-Vo-name=local, o=grid
objectClass: MdsDevice
objectClass: MdsNet
Mds-Device-name: eth1
Mds-Net-name: eth1
Mds-Net-netaddr: XXX.XXX.XXX.XXX/24
Mds-Net-addr: XXX.XXX.XXX.XXX
Mds-validfrom: 20070109223844Z
Mds-validto: 20070110043844Z
Mds-keepsto: 20070110043844Z

# lo, networks, mileva.hpc.odu.edu, local, grid
dn: Mds-Device-name=lo, Mds-Device-Group-name=networks, Mds-Host-hn=mileva.hpc
.odu.edu, Mds-Vo-name=local, o=grid
objectClass: MdsDevice
objectClass: MdsNet
Mds-Device-name: lo
Mds-Net-name: lo
Mds-Net-netaddr: 127.0.0.0/8

```

```

Mds-Net-addr: 127.0.0.1
Mds-validfrom: 20070109223844Z
Mds-validto: 20070110043844Z
Mds-keepsto: 20070110043844Z

# vmnet1, networks, mileva.hpc.odu.edu, local, grid
dn: Mds-Device-name=vmnet1, Mds-Device-Group-name=networks, Mds-Host-hn=mileva
.hpc.odu.edu, Mds-Vo-name=local, o=grid
objectClass: MdsDevice
objectClass: MdsNet
Mds-Device-name: vmnet1
Mds-Net-name: vmnet1
Mds-Net-netaddr: xxx.xxx.xxx.xxx/24
Mds-Net-addr: 192.168.92.1
Mds-validfrom: 20070109223844Z
Mds-validto: 20070110043844Z
Mds-keepsto: 20070110043844Z

# operating system, mileva.hpc.odu.edu, local, grid
dn: Mds-Software-deployment=operating system, Mds-Host-hn=mileva.hpc.odu.edu, M
ds-Vo-name=local, o=grid
objectClass: MdsSoftware
objectClass: MdsOs
Mds-Software-deployment: operating system
Mds-Os-name: Linux
Mds-Os-release: 2.6.9-22.ELsmp
Mds-Os-version: 1 SMP Sat Oct 8 21:32:36 BST 2005

# MDS, mileva.hpc.odu.edu, local, grid
dn: Mds-Software-deployment=MDS, Mds-Host-hn=mileva.hpc.odu.edu, Mds-Vo-name=lo
cal, o=grid
objectClass: MdsSoftware
objectClass: MdsService
objectClass: MdsServiceLdap
Mds-Software-deployment: MDS
Mds-Service-type: ldap
Mds-Service-hn: mileva.hpc.odu.edu
Mds-Service-port: 2135
Mds-Service-Ldap-timeout: 30
Mds-Service-admin-contact: unspecified
Mds-Service-Executable-PID: 6318
Mds-Service-Path: /opt/globus-4.0.2
Mds-Service-admin-comment: This is an MDS 2.4 deployment.
Mds-validfrom: 20070109223844Z
Mds-validto: 20070110043844Z
Mds-keepsto: 20070112173204Z

# jobmanager-pbs, mileva.hpc.odu.edu, local, grid
dn: Mds-Software-deployment=jobmanager-pbs, Mds-Host-hn=mileva.hpc.odu.edu, Mds
-Vo-name=local, o=grid
objectClass: Mds
objectClass: MdsSoftware
objectClass: MdsService
objectClass: MdsServiceGram
objectClass: MdsComputer
objectClass: MdsOs
Mds-Software-deployment: jobmanager-pbs
Mds-Service-type: x-gram
Mds-Service-hn: mileva.hpc.odu.edu
Mds-Service-port: 2119
Mds-Service-url: x-gram://mileva.hpc.odu.edu:2119/jobmanager-pbs:/O=ODU/OU=ODU
grid/OU=ODUgrid-sly.hpc.odu.edu/CN=host/mileva.hpc.odu.edu
Mds-Service-protocol: 0.1
Mds-Computer-isa: x86_64
Mds-Os-release: 2.6.9-22.ELsmp
Mds-Os-name: Linux
Mds-Computer-manufacturer: unknown
Mds-Service-Gram-schedulertype: pbs
Mds-validfrom: 200701092238.44Z
Mds-validto: 200701092239.14Z

```

```
Mds-keepsto: 200701092239.14Z
```

```
# batch, jobmanager-pbs, mileva.hpc.odu.edu, local, grid
```

```
dn: Mds-Job-Queue-name=batch, Mds-Software-deployment=jobmanager-pbs, Mds-Host  
-hn=mileva.hpc.odu.edu, Mds-Vo-name=local, o=grid
```

```
objectClass: Mds
```

```
objectClass: MdsSoftware
```

```
objectClass: MdsJobQueue
```

```
objectClass: MdsComputerTotal
```

```
objectClass: MdsComputerTotalFree
```

```
objectClass: MdsGramJobQueue
```

```
Mds-Job-Queue-name: batch
```

```
Mds-Computer-Total-nodeCount: 4
```

```
Mds-Computer-Total-Free-nodeCount: 4
```

```
Mds-Memory-Ram-Total-sizeMB: 0
```

```
Mds-Memory-Ram-sizeMB: 0
```

```
Mds-Gram-Job-Queue-maxtime: 0
```

```
Mds-Gram-Job-Queue-maxcputime: 0
```

```
Mds-Gram-Job-Queue-maxcount: 4
```

```
Mds-Gram-Job-Queue-maxrunningjobs: 0
```

```
Mds-Gram-Job-Queue-maxjobsinqueue: 0
```

```
Mds-Gram-Job-Queue-whenactive: 0
```

```
Mds-Gram-Job-Queue-status: enabled
```

```
Mds-Gram-Job-Queue-dispatchtype: batch
```

```
Mds-Gram-Job-Queue-priority: NULL
```

```
Mds-Gram-Job-Queue-jobwait: NULL
```

```
Mds-Gram-Job-Queue-schedulerSpecific: NULL
```

```
Mds-validfrom: 200701092238.44Z
```

```
Mds-validto: 200701092239.14Z
```

```
Mds-keepsto: 200701092239.14Z
```

```
# local, Grid
```

```
dn: Mds-Vo-name=local, o=Grid
```

```
objectClass: GlobusStub
```

```
# search result
```

```
search: 2
```

```
result: 0 Success
```

```
# numResponses: 28
```

```
# numEntries: 27
```

```
[17:38] mileva:~]
```