

---

# AstroGrid-D

Deliverable 1.2



## AstroGrid-D Grid Infrastructure<sup>1</sup>

Deliverable	1.2 Integration of Astrogrid-D Resources
Authors	WG 1
Editors	H. Enke
Date	09/18/2006
Document Version	1.0.0
Current Version	1.0.0
Previous Versions	0.2.0

### **A: Status of this Document**

Working draft of Deliverable 1.2, describing the available grid resources and their integration status as of July 2006.

### **B: Reference to project plan**

For development and testing the implementation of components of the grid infrastructure, as described in other working drafts, compute resources with installed Globus Toolkit are necessary.

### **C: Abstract**

This document describes the status of Globus 4 installation and configuration on AstroGrid-D owned resources and their availability for the project as of July 2006.

---

<sup>1</sup>This work is part of the AstroGrid-D project and D-Grid. The project is funded by the German Federal Ministry of Education and Research (BMBF).

**D: Changes History**

<b>Version</b>	<b>Date</b>	<b>Name</b>	<b>Brief summary</b>
0.1.0	3.4.2006		Working Draft Creation
0.1.1	4.4.2006	M. Braun	Port and server specification
0.1.2	19.4.2006	M. Braun, H. Enke	Added sections 4,5,6 and minor edits
0.1.3	22.7.2006	M. Braun, H. Enke	Update of port specification, install procedures
0.1.4	23.7.2006	H. Enke	Added subsection to host certificates. Update of policy section, minor edits
0.2.0	15.9.2006	H. Enke, M. Braun	Revision, minor edits
1.0.0	18.9.2006	H. Enke	Minor edits

E:

## Contents

<b>1 Overview</b>	<b>4</b>
<b>2 Grid Policies</b>	<b>5</b>
<b>3 Port Requirements for Basic Services</b>	<b>6</b>
<b>4 Site Firewalls</b>	<b>6</b>
<b>5 Configuring grid users and workspaces</b>	<b>7</b>
<b>6 D-Grid requirements</b>	<b>9</b>
<b>7 Problems</b>	<b>9</b>
<b>8 Integration of resources</b>	<b>10</b>
8.1 Environment Settings . . . . .	10
8.2 Host Certification . . . . .	10
8.2.1 Required files from RootCA . . . . .	11
8.2.2 Revocation lists . . . . .	11
8.3 Setup of gsgatekeeper and gsiftp . . . . .	11
8.4 Configuration of RFT . . . . .	12
8.5 Setup of the Java WS Core Container . . . . .	13
8.6 Setup of gsissh . . . . .	14
8.7 Monitoring - Setup of Ganglia . . . . .	15
8.8 Monitoring - Enhance MDS4 . . . . .	16

# 1 Overview

This overview is based on the information, which was collected by Workgroup 1 based on an XML-Schema.

The purpose of this collection was twofold:

- to catalogue all resources, which were brought into the project as backing for cofunding of the proposal in as much detail as possible
- to catalogue the already available grid related installation of software and policies for on their usage, intended and actual

The first table gives an overview of all the hardware resources, which eventually will be incorporated into the grid infrastructure.

Institute	Resource-type	CE Count	Arch	N CPU	Memory (GB/node)
AEI	WS	1	IA32	1	0.25
	Cluster	1	IA32	129x2	2
	Cluster	1	IA64	48x2	4
AIP	WS	5	IA32/IA64	2(4)	2-16
	WS	7	Alpha	1	0.5-1.2
	Cluster	1	IA32	32x2	2
	Cluster	1	IA64	128x2	4-6
TUM	(Blade)	1	IA32	4x1	1
ZAH	WS	5	IA32/IA64?	2(4)	2-16
	Cluster	1	IA32	11x2	2
ZIB	Cluster	1	IA32	17x2	1(5)
	Cluster	1	CrayXD1	6x2	2
MPE	WS	3	IA32	2(4)	2-16
LRZ	Cluster	1	HitachiSR8k	168	8
	Cluster	1	SGI	128	4
RZG	Cluster	1	P4	32x1	2
UP	Cluster	1	IA32	17x1	1

Status described as of July 2006.

Looking at the same table for actual availability of resources with a Globus 4 installation, which is considered the basic requirement for incorporation into the grid infrastructure, the situation is as follows:

Institute	Resource-type	GT2.x	GT3.x	GT4.x	D-Grid CA
AEI	WS	-	-	X	yes
	Cluster	-	-	X	yes
	Cluster	-	-	X	yes
AIP	WS	-	-	X	yes
	WS	-	-	(X)	yes
	Cluster	-	-	(X)	yes
	Cluster	-	-	-	yes
TUM	(Blade)	-	X	-	?
ZAH	WS	X	-	X	?
	Cluster	X	-	X	?
ZIB	Cluster	X	-	X	yes
	Cluster	-	-	-	-
MPE	WS	X	-	(X)	yes
LRZ	Cluster	X	-	X	yes
	Cluster	X	-	-	yes
RZG	Cluster	-	-	X	?
UP	Cluster	-	X	-	?

Status described as of July 2006.

The resources of LRZ and RZG are generally only available through a special arrangement, which is granted based on asking for an individual allocation.

For a cluster, an installation of GTK on the head node is considered sufficient, but there may be changes necessary for MPI-jobs. The cluster resources are currently as well granted on an individual request base and in general only available from members of the institutes.

This leaves the workstations for building a testbed.

## 2 Grid Policies

Information on accepted Root CA is only available from AEI, AIP, MPE and LRZ. While the MPE workstations still run a private certification schema (GAVO), AEI and AIP accept certificates from RootCA FZK (GermanGrid) and from RootCA DFN (GridGerman). Certificates for AEI and AIP hosts are from Root CA FZK, LRZ obtains these from DFN.

For the basic Globus 4.x services, no additional service certificates are required, this has changed from GTK 2.4. Information on service certificates is only known from AIP, where service certificates are obtained from FZK. DFN does not issue service certificates.

Most of the clusters have restrictions (time limits) for grid access. The workstations currently have unrestricted access times.

### 3 Port Requirements for Basic Services

For joining the testbed a machine has to be accessible through the default basic globus connectivity. This requires, that the hosts are visible to the world, i.e have a FQDN and several ports on these hosts are accessible.

Pre-WS Globus services require the following ports[1]:

- GridFTP (port 2811)
- GlobusGateKeeper / GRAM (port 2119)
- GsiSsh, (port 2222)
- MyProxy, (port 7512) (on GridSphere-Servers)
- MDS2/GRIS, (port 2135) (preWS-monitoring)

WS based Globus services require the following ports:

- Globus Container (tomcat/globus, port 8443)
- Monitoring MDS4 (port 2135)
- MyProxy (port 7512)

### 4 Site Firewalls

Grid hosts are usually inside a firewall. The network administration may open all ports for grid hosts. Then the above port configuration allows an operational grid. If sites want to use a more restrictive policy and avoid possible conflicts with other site service configuration, they may choose to allow only a certain range of ports to be accessible for grid hosts. For this, Globus uses the GLOBUS\_TCP\_PORT\_RANGE variable. The rangewidth depends on which applications will run. For MPI jobs, based on mpichG2, a range of at least 500 ports should be available.

If a site restricts the port range, grid-clients and servers have to set this port range via this variable. A site needs to announce this port range to potential grid users. Otherwise a job may run, but is not retrievable from the grid host. The announcement of such restrictions has to be incorporated in monitoring services. The job submission services have to provide facilities to set a such a portrange, eg. on GridSphere portal.

The DGI suggests choosing portnumbers from 20000 to 25000, since these are not in use by any service[7]. This suggestion has been adopted by the D-Grid-StA as of August 2006. Each AstroGrid-D resource provider should configure their firewall settings accordingly.

## 5 Configuring grid users and workspaces

The common schema for a Globus installation is a mapping of local accounts of a user to the DN-Cert of the same user. This schema does in fact not allow for any abstraction as Virtual Organisations, but is compliant with wellknown administration procedures, auditing and access control can be achieved with usual tools.

But if the numbers of grid-users in a community grows and no longer are from only one administrative domain, this becomes not manageable in the long run. For Grid hosts, handling eg. 50 users according to the above schema places a heavy burden on the local administration of resources.

Currently, alternative approaches to the problem are only available with a couple of packages, which are restricted to older versions of GT and fairly old OS versions (gLite/EDG VOMS[3] and LCMAPS[2]). Although VOMS (VO Membership Service) is recommended by D-Grid, this is not applicable until the packages are no longer tightly coupled to the complete gLite installation. Currently it is not possible to install the required packages on top of GT4.x.

One of the basic concepts of VOMS/LCMAPS is to use preconfigured local accounts for grid resources and a dynamic mapping of grid-users to these local accounts, substituting the globus gridmap-file, which normally provides the authorisation for using the local resources. Additionally VO based mapping to roles is introduced.

As already stated, an implementation is not available. But we can already use a preconfigured pool of local accounts now, preparing for later use with VOMS/LCMAPS. The local resource providers would benefit from this, since they could configure suitable accounts (and groups), while including a new grid user to access the resources would only require to add his Certificate-DN to the gridmap file.

We suggest the following schema as an intermediate setup:

Local accounts on grid-hosts:

A pool of x grid-accounts with a common workspace is set up:

*(gridpool[1-x]), HOME=/path/to/workspace/gridpool[1-x]*. Grid users of AstroGrid or D-Grid are mapped in the gridmapfile to a pool user in a one-to-one relation. This would allow even for setting up groups with different environments.

Access-policies:

On access hosts, gsissh/grid-ftp is enabled, otherwise only globus-gram (pre/ws and ws) are able to talk to the grid-hosts

On access-hosts, the grid users get a shell, otherwise nologin is configured (e.g: let mintaka.aip.de be the access-host, then submitting a job to another grid-host, e.g. gavo2.aip.de, is enabled, but the grid user cannot login to gavo2.aip.de). The submission of a job to one of these Compute Elements (CE) has to include the staging in/out process by default.

In order to integrate the pool of grid accounts into the WS-GRAM schema, the /etc/sudoers file has to be extended as follows (agdusr is the prefix chosen for this example of 16 users):

```
# Cmnd alias specification
Runas_Alias GP00L1=agdusr000,agdusr001,agdusr002,agdusr003,agdusr004,\
    agdusr005,agdusr006,agdusr007
```

```
Runas_Alias GPOOL2=agdusr008,agdusr009,agdusr010,agdusr011,agdusr012,\
  agdusr013,agdusr014,agdusr015
```

```
globus ALL=(GPOOL1, GPOOL2) NOPASSWD: \  
/work1/globus/gt402 /libexec/globus-gridmap-and-execute \  
-g /etc/grid-security/grid-mapfile \  
/work1/globus/gt402 /libexec/globus-job-manager-script.pl *  
globus ALL=(GPOOL1, GPOOL2) NOPASSWD: \  
/work1/globus/gt402 /libexec/globus-gridmap-and-execute -g \  
/etc/grid-security/grid-mapfile \  
/work1/globus/gt402 /libexec/globus-gram-local-proxy-tool *
```

## 6 D-Grid requirements

D-Grid has set up a website[6] monitoring of all available D-Grid resources with Globus-MDS at LRZ. AstroGrid-D resources are included into this overview from the beginning, using the VO of AstroGrid-D. The D-Grid-MDS-Server obtains all AstroGrid-D information through the central Globus-MDS server astrogrid-mds.aip.de[5].

## 7 Problems

The following problems remain to be solved in the next stages of resource integration:

The GridSphere portal is only able to process Globus MDS2 information. An update to MDS4 is required. Alternatively, the AstroGrid-D Information Service should be used.

Some resources like e.g. peyote.aei.mpg.de only provide Globus MDS2 information. MDS2 is not compatible to MDS4.

Information about restricted port ranges is not included in MDS information. MDS is using the GLUE schema version 1.1, which does not include a corresponding entry.

MDS information is not yet available in a suitable form for the information service. A translation mechanism to RDF as used by the AstroGrid-D Information Service has to be established.

## 8 Integration of resources

Since the testbed is based on Globus Toolkit 4.0.1 and 4.0.2, a corresponding installation is required on the resource to be integrated. This installation can be done with an appropriate binary distribution, or a compilation from the Globus sources could be made. The latest version of the detailed description of the source-based installation can be found on the web pages of AstroGrid-D[4]:

The user account `globus` should be created on the resource, and the software should be installed using this account. Root access is required for several configuration tasks.

The creation of a symbolic link to the actual toolkit version is very useful in view of possible version upgrades:

```
mkdir /usr/local/globus
ln -s /work1/globus/gt402 /usr/local/globus/gtk
```

### 8.1 Environment Settings

The environment of each grid user has to be extended by several definitions. This example could be included in `/etc/profile` or `~/.bashrc` (users of `csh` or `tcsh` have to use adequate settings):

```
# export globus-env by default
GLOBUS_LOCATION=/usr/local/globus/gtk
GLOBUS_TCP_PORT_RANGE=20000,25000
GLOBUS_PATH=$GLOBUS_LOCATION/sbin
. $GLOBUS_LOCATION/etc/globus-user-env.sh
PATH=$GLOBUS_PATH:$PATH
export PATH GLOBUS_LOCATION GLOBUS_TCP_PORT_RANGE
```

`GLOBUS_LOCATION` has to be set to the local location of the toolkit - in this case using the symbolic link, while `GLOBUS_TCP_PORT_RANGE` defines the range of ports as used by the Globus gatekeeper.

### 8.2 Host Certification

A detailed description of the certification process can be found on the intranet pages. Depending on the Root CA, the basic configuration for requesting certificates in `/etc/grid-security/certificates` has to be edited. This is only used for cert-requests and does not have any impact on accepting certificates from other Root CA later. For each institution there should be only one Root CA, from which all (host, user, service) certificates are obtained. The host certificates (`host-cert.pem`), which are received from Root CA, have to be installed into `/etc/grid-security` and for Globus 4.x copied to `container-cert.pem`.

### 8.2.1 Required files from RootCA

D-Grid policy is to accept Root CA FZK and Root CA DFN. Therefore, the root-certificates of both CA have to be installed into `/etc/grid-security/certificates`: These are at least: `1149214e.(0, crl_url)`, `34f8e29c.(0, crl_url)` for DFN and `dd4b34ea.(0,crl_url)`. The EuroGridPMA issues a release of all accredited RootCA files every two month, and AstroGrid-D resource providers should check the releases, especially, if they accept additional certificates from other than D-Grid RootCA.

### 8.2.2 Revocation lists

From each RootCA a list of revoked certificates is available. It is obtained by using the `<hash>.crl_url`. With a simple cron-job and a script from IGTF, regular updates of this lists should be made, resulting in additional `<hash>.r0` files. Globus checks these files and refuses to accept certificates listed in the revocation lists. And it refuses valid certificates if revocation lists are expired. This is a very basic security measure, that should be established on each AstroGrid-D resource. Additional information is available on the intranet-pages.

### 8.3 Setup of gsgatekeeper and gsiftp

Root access is required to configure these services. The file `/etc/services` needs to be extended by the following lines:

```
gsgatekeeper 2119/tcp
gsiftp 2811/tcp
```

And two files have to be created in the directory `/etc/xinetd.d/`, firstly `gsgatekeeper`:

```
service gsgatekeeper
{
    env = LD_LIBRARY_PATH=/usr/local/globus/gtk/lib
    env += GLOBUS_TCP_PORT_RANGE=20000,25000
    env += GLOBUS_TCP_SOURCE_PORT_RANGE=20000,25000
    socket_type = stream
    protocol = tcp
    user = root
    server = /usr/local/globus/gtk/sbin/globus-gatekeeper
    server_args = -conf /usr/local/globus/gtk/etc/globus-gatekeeper.conf
    wait = no
}
```

and `gsiftp`:

```
service gsiftp
{
    env = GLOBUS_LOCATION=/usr/local/globus/gtk/
```

```
env += LD_LIBRARY_PATH=/usr/local/globus/gtk/lib
env += GLOBUS_TCP_PORT_RANGE=20000,25000
log_on_success += DURATION USERID
log_on_failure += USERID
socket_type = stream
protocol = tcp
user = root
server = /usr/local/globus/gtk/sbin/globus-gridftp-server
server_args = -i
wait = no
instances = 100
nice = 10
}
```

The port ranges and the pathes should be adjusted to the local values. Finally, xinetd can be reconfigured:

```
/etc/init.d/xinetd reload
```

## 8.4 Configuration of RFT

RFT requires a database. Here, postgres is used. Check for its status:

```
/etc/init.d/postgresql status
```

If it is not running, try to start it (as root):

```
/etc/init.d/postgresql start
```

Postgres should be configured to allow local access via tcpip sockets, so in `/var/lib/pgsql/data/postgresql.conf` the corresponding value needs to be set to true:

```
tcpip_socket = true
```

The local host has to be added to the list of trusted hosts, as well. Just edit `/var/lib/pgsql/data/pg_hba.conf` and append:

```
host all all 127.0.0.1 255.255.255.255 trust
host all all 141.33.4.98 255.255.255.255 trust
```

Instead of 141.33.4.98, use your local IP number, please. A restart of postgres is required now:

```
/etc/init.d/postgresql restart
```

As user postgres, a database for RFT has to be created and filled with the RFT schema, while globus will be established as database user:

```
su - postgres
createdb rftDatabase
psql -d rftDatabase -f /usr/local/globus/gtk/share/globus_wsrf_rft/rft_schema.sql
createuser globus
```

When asked to allow globus to create databases and new users, just answer with y. Then, exit since the user postgres has done his job. Postgres is configured for the use of RFT.

## 8.5 Setup of the Java WS Core Container

The container needs a certificate as well, but only a copy of the host certificate is required, so:

```
cd /etc/grid-security
cp hostcert.pem containercert.pem
cp hostkey.pem containerkey.pem
chown globus.globus containercert.pem containerkey.pem
```

A start-stop script will be very useful, so create it as user globus:

```
vi $GLOBUS_LOCATION/start-stop
```

The contents of this script shall follow this outline (pathes need to be adapted):

```
#!/bin/sh
set -e
export GLOBUS_LOCATION=/usr/local/globus/gtk/
export JAVA_HOME=/usr/local/jdk/jsdk
export ANT_HOME=/usr/local/jdk/ant
export GLOBUS_OPTIONS="-Xms256M -Xmx512M"
. $GLOBUS_LOCATION/etc/globus-user-env.sh
cd $GLOBUS_LOCATION
case "$1" in
  start)
    $GLOBUS_LOCATION/sbin/globus-start-container-detached -p 8443
    ;;
  stop)
    $GLOBUS_LOCATION/sbin/globus-stop-container-detached
    ;;
  *)
    echo "Usage: globus {start|stop}" >&2
    exit 1
;;
esac
exit 0
```

Just make this file executable:

```
chmod +x $GLOBUS_LOCATION/start-stop
```

As user root, another useful script has to be created (/etc/init.d/globus):

```
#!/bin/sh -e
export GLOBUS_LOCATION=/usr/local/globus/gtk/
case "$1" in
start)
    su - globus $GLOBUS_LOCATION/start-stop start
    ;;
stop)
    su - globus $GLOBUS_LOCATION/start-stop stop
    ;;
restart)
    $0 stop
    sleep 1
    $0 start
    ;;
*)
    printf "Usage: $0 {start|stop|restart}\n" >&2
    exit 1
    ;;
esac
exit 0
```

This script should also be executable:

```
chmod +x /etc/init.d/globus
```

Just start the Java WS core container with:

```
/etc/init.d/globus start
```

## 8.6 Setup of gsissh

If you want to enable grid users to login on your resource using gsissh, we suggest port 2222. Port 22 is kept this way for the use of the standard sshd. As user root, establish the service:

```
cd /etc/init.d
ln -s /usr/local/globus/gtk/sbin/SXXsshd gsisshd
/sbin/chkconfig --add gsisshd
```

chkconfig might be a system specific task (e.g. Redhat or Suse). Edit the configuration file:

```
vi $GLOBUS_LOCATION/etc/ssh/sshd_config
```

and uncomment the Port line and change 22 to 2222 (Port 2222). Enter the similar line (Port 2222) in the client's configuration:

```
vi $GLOBUS_LOCATION/etc/ssh/ssh_config
```

Define gsissh as a service, i.e. edit /etc/services and include this line:

```
gsissh 2222/tcp
```

Just start the service:

```
/etc/init.d/gsisshd start
```

## 8.7 Monitoring - Setup of Ganglia

Firstly, ganglia needs to be installed. Just get the ganglia sources from <http://ganglia.info/downloads.php> (ganglia-3.0.2.tar.gz) and save it as /tmp/ganglia-3.0.2.tar.gz. Please, take the following commands as examples and change the path concerning your local requirements:

```
cd /work1/globus/  
tar xvfz /tmp/ganglia-3.0.2.tar.gz  
cd ganglia-3.0.2/  
./configure  
make  
make install
```

This should have compiled and installed ganglia. Just make a test:

```
/usr/sbin/gmond  
telnet localhost 8649
```

At its start, gmond should produce the following text: Configuration file '/etc/gmond.conf' not found. Anyhow, gmond should run, and the given telnet command should provide an xml output containing many system parameters. Just try the following queue then:

```
telnet localhost 8649 | grep "HOST NAME=" | wc -l
```

The output should be 1 (one). If you get a bigger number, ganglia is already collecting information about neighbouring machines. This can cause problems with the interface to MDS4, later. Please, kill the gmond process after testing. Then, create the config file:

```
/usr/sbin/gmond -t > /etc/gmond.conf
```

You can edit `/etc/gmond.conf` now. If you recognize problems with MDS4 later which might be caused by the listing of more than one machine by gmond, you might consider to change the values of `mcast_join` and the bind address in the `udp_send_channel` and `udp_rcv_channel` sections. For now, install gmond as a service on your resource:

```
cp /work1/globus/ganglia-3.0.2/gmond/gmond.init /etc/rc.d/init.d/gmond
/sbin/chkconfig --add gmond
/sbin/chkconfig --list gmond
/etc/rc.d/init.d/gmond start
```

## 8.8 Monitoring - Enhance MDS4

MDS4 is known to work with ganglia, so you should try to configure MDS4 correspondingly:

```
vi $GLOBUS_LOCATION/etc/globus_wsrf_mds_usefulrp/gluerp.xml
```

Just exchange the defaultProvider definitions, you need to enable this one:

```
<defaultProvider>java org.globus.mds.usefulrp.glue.GangliaElementProducer</defaultProvider>
```

while the other one should (none) be commented.

Edit the hierarchy file:

```
vi $GLOBUS_LOCATION/etc/globus_wsrf_mds_index/hierarchy.xml
```

and define the following:

```
<upstream>https://astrogrid-mds.aip.de:8443/wsrf/services/DefaultIndexService</upstream>
```

This will register your MDS4 at astrogrid-mds (currently mintaka.aip.de), thus enabling our VO MDS4 monitoring.

In order to provide a fully qualified domain name instead of the IP number to MDS, change `$GLOBUS_LOCATION/etc/globus_wsrf_core/server-config.wsdd` and insert the following two lines:

```
<parameter name="logicalHost" value="Your.Resource.Name"/>
<parameter name="publishHostName" value="true"/>
```

just behind `<globalConfiguration>`.

Restart the WS core container:

```
/etc/init.d/globus restart
wsrf-query -a -z none -s https://127.0.0.1:8443/wsrf/services/DefaultIndexService
```

You will get an xml output which should also contain the transformed ganglia output like processor load, file system and memory information etc.

## References

- [1] Globus Firewall Requirements <http://www.globus.org/toolkit/security/firewalls/Globus\%20Firewall\%20Requirements-7.pdf>
- [2] Guide to LCMAPS [http://www.dutchgrid.nl/DataGrid/wp4/lcmaps/edg-lcmaps\\_gcc3\\_2\\_2-0.0.23/lcmaps.pdf](http://www.dutchgrid.nl/DataGrid/wp4/lcmaps/edg-lcmaps_gcc3_2_2-0.0.23/lcmaps.pdf)
- [3] VOMS <http://edg-wp2.web.cern.ch/edg-wp2/security/voms/voms.html>
- [4] AstroGrid-D Grid support <http://www.gac-grid.org/project-products/grid-support/grid-installation.html>
- [5] AstroGrid-D MDS <http://www.gac-grid.org/project-products/grid-status/astrogrid-mds.html>
- [6] D-Grid MDS <http://webmds.lrz-muenchen.de:8080/webmds/>
- [7] D-Grid: Firewall Requirements <http://www.gac-grid.org/project-products/grid-support/grid-installation.html>