



GT4 Installation, Configuration and Tests.

Dipl.-Inf. Hamza Mehammed

mehammed@lrz.de

12.09.2006

Leibniz Computing Centre

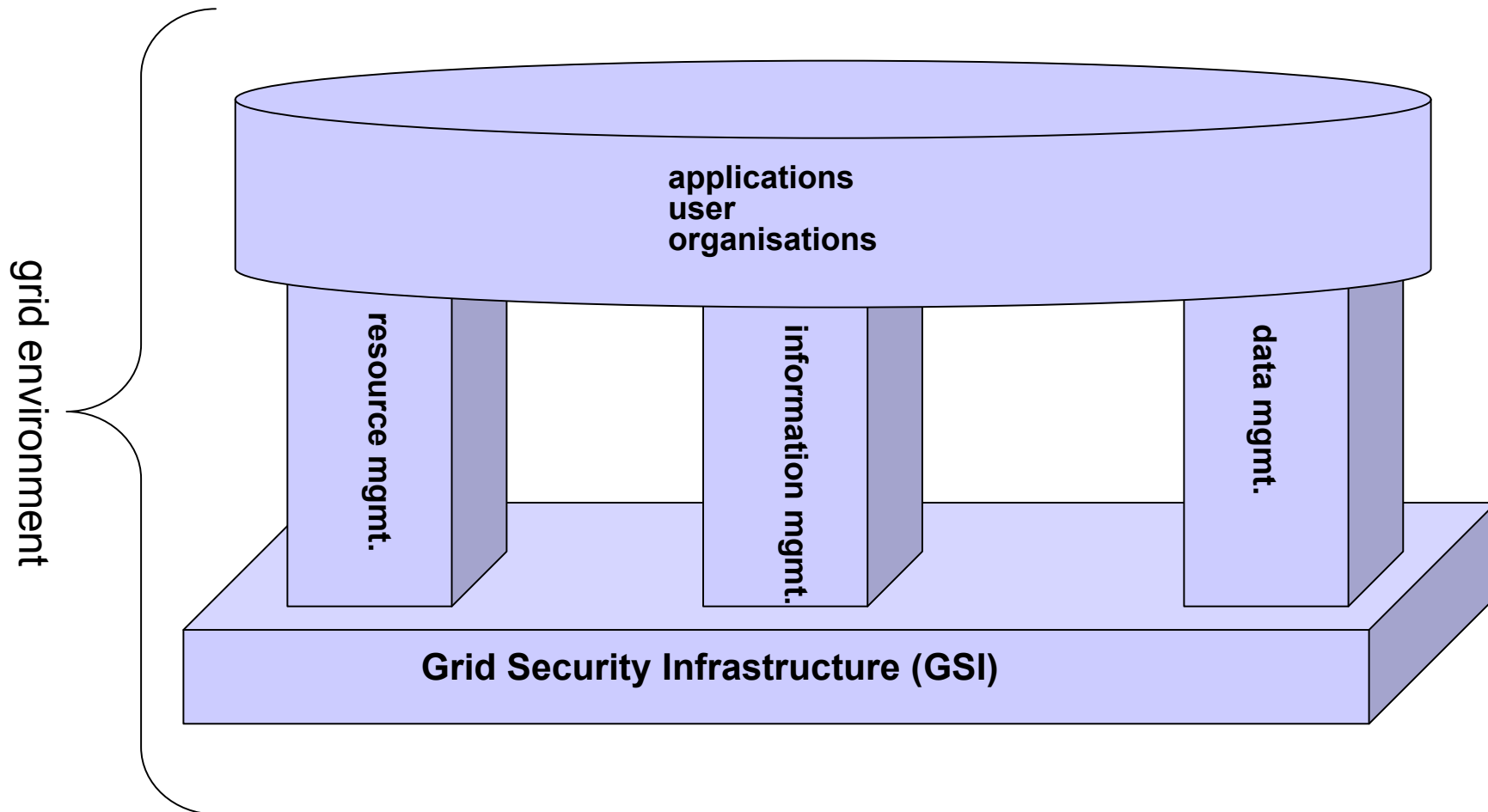




Contents

- roles and predefinitions**
- grid security**
 - **host, user und container certificate**
- installation**
 - **Grid Packaging Tool (GPT) & Globus Toolkit (GT4)**
- authentication and authorization**
- grid services**
 - **GRAM and WS-GRAM**
 - **GridFTP and RFT**
 - **MDS and MDS4**
- GT4 usage utatistics collection**

Grid Environment with Globus



Roles



- ❑ system administrator
 - user **root**
- ❑ Globus administrator
 - user **globus**
- ❑ Postgresql administrator
 - user **postgres**
- ❑ Grid user
 - user **steffi**



Tasks of Roles (1)

- ❑ **system administrator (**root**)**
 - **server configuration**
 - **host certificate**
 - **authorisation**

- ❑ **Globus administrator (**globus**)**
 - **container certificate**
 - **Globus installation**
 - **configuration of the services**
 - **start/stop service container**



Tasks of Roles (2)

- ❑ **Postgresql administrator ([postgres](#))**
 - **database create, manage, ...**

- ❑ **Grid user (steffi)**
 - **resource usage**
 - **job submission, job status, ...**
 - **informations**
 - **monitoring, indexing**
 - **data transfer, data storage**
 - **security**
 - **user certificates, proxy certificates**

Predefinitions (1)



- ❑ installation and configuration
 - example: Suse Linux 10.1
 - with bash shell

- ❑ execution mode
 - root: # **command**
 - globus : > **command**
 - steffi: \$ **command**
 - postgres: % **command**

Predefinitions (2)



- ❑ installation directories for

- Globus Toolkit (GT4)
- Grid Packaging Tool (GPT)

- ❑ local for all readable files repository

```
# chmod -R a+rX /tmp/LrzGlobusWorkshop
```

- ❑ environment variables

```
# cp /tmp/LrzGlobusWorkshop/globus-env-setup.sh \  
/usr/local/bin/
```

and/or

```
# cp /tmp/LrzGlobusWorkshop/globus-env-setup.csh \  
/usr/local/bin/
```

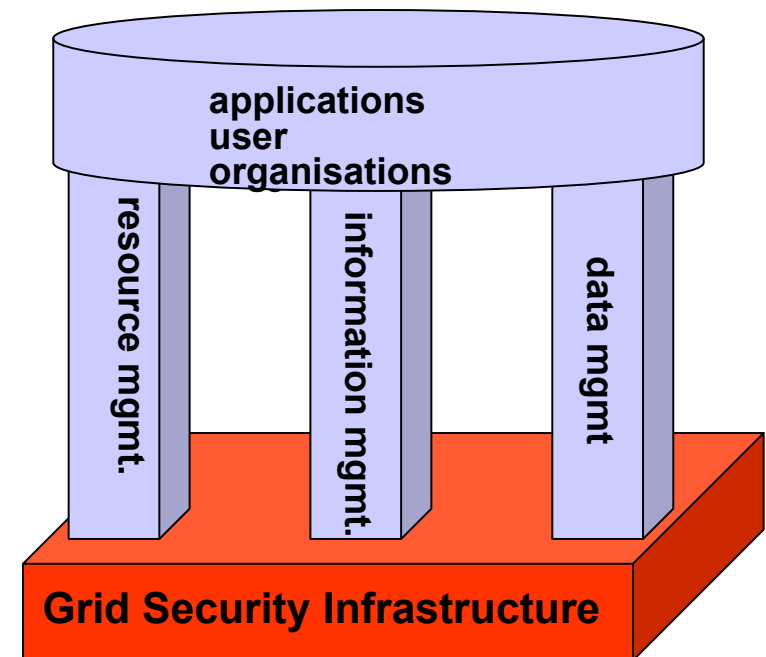


Preparation

- ❑ **create user globus**
 - # groupadd globus
 - # useradd -m -g globus -d /home/globus globus
 - # passwd globus
- ❑ **create grid user (steffi)**
 - # useradd -m -d /home/steffi steffi
 - # passwd steffi
- ❑ **create GPT installation directory**
 - # mkdir /usr/local/gpt-3.2
 - # chown globus.globus /usr/local/gpt-3.2
- ❑ **create GT4 installation directory**
 - # mkdir /usr/local/globus-4.0.1
 - # chown globus.globus /usr/local/globus-4.0.1

Grid Security Infrastructure - GSI (1)

- public key infrastructure
 - private key
 - public key
 - digital certificate
- authentication
 - user
 - host
- authorisation
 - user
- delegation
 - proxy certificate





Grid Security Infrastructure - GSI (2)

- mutual authentication
 - Is it the one who it say it is
 - trust the CA of the other party
- uses X.509 certificate format (IETF)
- global name space (DN)
- information provided
 - subject
 - public key
 - identity of CA
 - digital signature of the CA
- grid-mapfile
 - mapping certificate to local user



Grid Security Infrastructure - GSI (3)

- proxy
 - avoid re-entering password
 - signed by owner
 - less secure
 - new certificate and private key
 - uses X.509 certificate format (IETF)
- Single sign-on
- myproxy
 - credential repository
 - global access
 - renewing proxy credentials by servers

Host Certificates



❑ host certificates

```
# mkdir /etc/grid-security
# cd /etc/grid-security
# cp yourhostkey.pem hostkey.pem
# cp yourhostcert.pem hostcert.pem
# cp hostkey.pem containerkey.pem
# cp hostcert.pem containercert.pem
# chown globus.globus containerkey.pem \
  containercert.pem
```

❑ access rights

```
# chmod 400 *key.pem
# chmod 644 *cert.pem
# ls -al /etc/grid-security
```

User Certificate



- ❑ **user certificate (steffi)**
 - \$ mkdir ~/.globus
 - \$ cd ~/.globus
 - \$ cp youruserkey.pem userkey.pem
 - \$ cp yourusercert.pem usercert.pem

- ❑ **access rights**
 - \$ chmod 400 *key.pem
 - \$ chmod 644 *cert.pem
 - \$ ls -al ~/.globus

CA certificates



- ❑ unpacking of the Certification Authority (CA) certificates
 - # `cd /etc/grid-security/`
 - # `tar xvf /tmp/LrzGlobusWorkshop/certificates.tar`

- ❑ contained CA Certificates
 - in D-Grid accepted certificates:
 - DFN-Verein (Root CA, Server CA, User CA)
 - GridKA
 - for this Summer School:
 - LRZ Simple CA

Adjust Setup Script



☐ Ant

- define path
which ant
- adjust ANT_HOME variable in setup file
vi /usr/local/bin/globus-env-setup.*

☐ Java

- define path
ls -al `which javac`
- follow links, until no link is present anymore
- adjust JAVA_HOME variable (without /bin/...) in setup file
vi /usr/local/bin/globus-env-setup.*



Installation GPT

- ❑ execute setup script as user globus

```
> . /usr/local/bin/globus-env-setup.sh
```

error message is OK here !

- ❑ GPT installation

```
> cd /tmp/LrzGlobusWorkshop
```

```
> tar zxvf gpt-3.2-src.tar.gz
```

```
> cd gpt-3.2
```

```
> ./build_gpt
```

Installation Globus Toolkit 4



- ❑ Suse binaries (provided by LRZ)

- Suse 10.x

```
> $GPT_LOCATION/sbin/gpt-install \  
    <Lrz_GT4_binary.tar.gz>
```

```
> $GPT_LOCATION/sbin/gpt-postinstall
```

Authorisation (1)



- ❑ execute setup script as grid user (steffi)
\$. /usr/local/bin/globus-env-setup.sh
- ❑ extraction of Distinguished Name (DN)
\$ grid-cert-info -subject
output, e.g.:
/C=DE/O=GridGermany/OU=Leibniz-Rechenzentrum/CN=steffi
- ❑ execute setup script as root
. /usr/local/bin/globus-env-setup.sh
- ❑ user authorisation in /etc/grid-security/grid-mapfile
grid-mapfile-add-entry -dn "<DN>" -ln <login>
grid-mapfile-check-consistency (without output)



Authorisation (2)

- ❑ insert user in /etc/sudoers

```
# cat /tmp/LrzGlobusWorkshop/sudoers >> \  
/etc/sudoers  
# vi /etc/sudoers  
globus ALL=(steffi,...) ... /...job-manager-script...  
globus ALL=(steffi,...) ... /...gram-local-proxy...
```

- ❑ insert complete host name (FQDN) in /etc/hosts

- extract FQDN

```
# cd /etc/grid-security  
# grid-cert-info -file hostcert.pem -subject
```

- extract IP address

```
# host <FQDN>
```

```
# vi /etc/hosts
```

```
entry: <IP-Address> <FQDN>
```



Testing the Globus-Installation (1)

- ❑ test of the service container without security

```
> globus-start-container -nosec
```

```
Result: 51 Grid Web Services on port 8080
```

- ❑ test example

```
$ counter-client -s \
```

```
http://<FQDN>:8080/wsrp/services/CounterService
```

- ❑ stop the started container

```
> globus-start-container -nosec
```



Testing the Globus-Installation (2)

- ❑ test of the service container with security

```
> globus-start-container
```

```
Result: 51 Grid Web Services on port 8443
```

- ❑ test of the Grid Security Infrastructure (GSI)

```
$ grid-proxy-init -verify -debug
```

- ❑ test Grid Service example

```
$ counter-client -s \
```

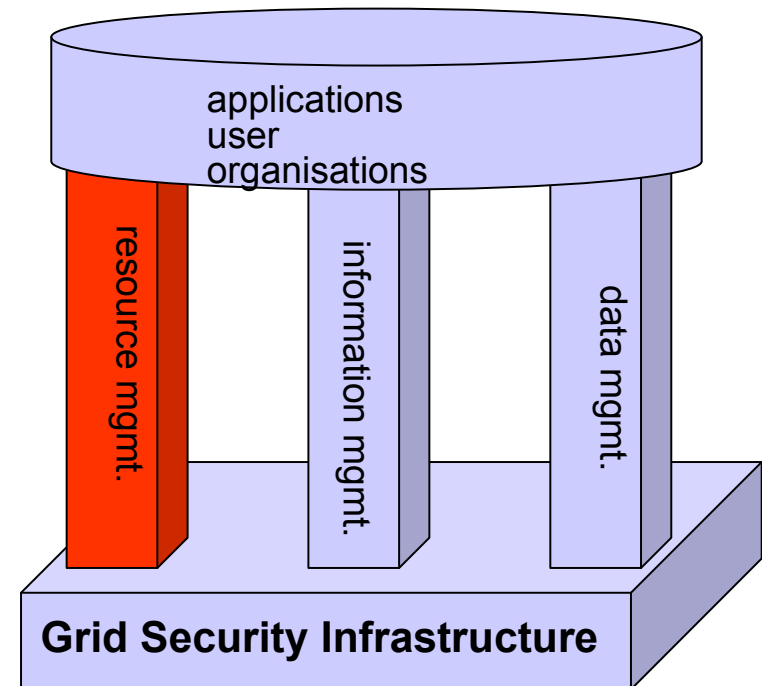
```
https://<FQDN>:8443/wsrp/services/CounterService \
```

```
-z none
```



Grid Resource Allocation Manager (GRAM)

- job submission
 - submit remote jobs
 - query status
 - fetch results
- gatekeeper
- jobmanager
- scheduler
 - Fork (default)
 - PBS
 - Condor
 - LSF





GRAM: Configuration

- ❑ assigning port 2119

```
# echo gsigatekeeper 2119/tcp >> /etc/services
```

- ❑ copy configuration file

```
# cp /tmp/workshop/gsigatekeeper \  
/etc/xinetd.d/gsigatekeeper
```

- ❑ restart super daemon

```
# /etc/init.d/xinetd restart
```

- ❑ please note: machine time must be up to date!

- ❑ insert in /etc/hosts.allow (tcp wrapper):

- **ALL:ALL:rfc931:ALLOW !!!**



WS-GRAM: Test (1)

- ❑ test of the gatekeeper

```
$ globus-personal-gatekeeper -start
```

output: GRAM contact: <FQDN>:<PORT>:<DN>

```
$ globus-job-run "<contact>" /bin/hostname
```

```
$ globus-personal-gatekeeper -killall
```

- ❑ pre-WS-GRAM command (GT2):

```
$ globus-job-run localhost /bin/hostname
```



WS-GRAM: Test (2)

❑ WS-GRAM command (interactive):

- `globusrun-ws -submit -F JobFactoryURL \`
`-Ft FactoryType -s -c command`

Test:

```
$ globusrun-ws -submit -c /bin/hostname
```

❑ WS-GRAM command (batch):

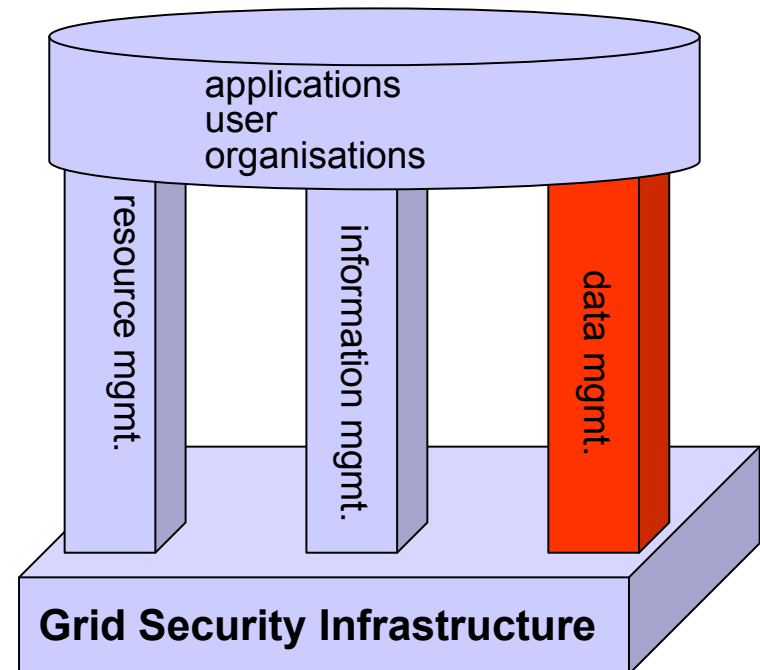
- `globusrun-ws -submit -batch -F JobFactoryURL \`
`-Ft FactoryType -o EPRfile -c command`

❑ Job status

```
$ globusrun-ws -status -job-epr-file EPRfile
```

Grid File Transfer Protocol - GridFTP (1)

- ❑ support GSI security
- ❑ based on File Transfer Protocol (FTP)
- ❑ A base for RFT
 - TCP buffer sizes
- ❑ transfer efficiency
 - Multiple TCP streams
 - TCP buffer sizes
- ❑ Striping functionality





Grid File Transfer Protocol - GridFTP (2)

- ❑ **Command line tool**

 - \$ globus-url-copy**

 - **parameter:**

 - [-tcp-bs buffersize] [-p parallelism] source dest**

- ❑ **source - dest format**

 - **protocol://host:port/path**

- ❑ **Supported protocols**

 - **https, http, gsiftp, ftp, and file**



GridFTP: Configuration

- ❑ assigning port 2811

```
# echo gsiftp 2811/tcp >> /etc/services
```

- ❑ copy configuration file

```
# cp /tmp/LrzGlobusWorkshop/gsiftp  
/etc/xinetd.d/gsiftp
```

- ❑ restart super daemon

```
# /etc/init.d/xinetd restart
```



GridFTP: Test

❑ copy: local → local

```
$ globus-url-copy gsiftp://localhost/etc/hosts \  
file:///tmp/hosts_copy
```

❑ copy: remote → local

```
$ globus-url-copy \  
gsiftp://<Hostname>/etc/hosts \  
file:///tmp/hosts_copy_<YourLogin>
```

❑ copy: local → remote

```
$ globus-url-copy \  
file:///tmp/hosts_copy_<YourName> \  
gsiftp://<Hostname>/tmp
```



Reliable File Transfer (RFT)

- using database information
 - postgresql as a default database
 - check pointing transfer state
 - to recover from failures
 - control and supervision
- retrying transfers
 - using checkpoints
- recursive directory transfer
- transfer all or none



Postgresql: Configuration (1)

- ❑ initialise database (DB)

```
% initdb -D /var/lib/pgsql/data
```

- ❑ access restriction to Globus user

```
% vi /var/lib/pgsql/data/pg_hba.conf
```

insert at end of file:

```
host [TAB] rftDatabase [TAB] globus [TAB] <Ihre-IP> \  
[TAB] 255.255.255.255 [TAB] trust
```

- ❑ enable TCP/IP connections

```
# vi /etc/sysconfig/postgresql
```

```
addition: POSTGRES_OPTIONS="-i"
```

- ❑ start Postgresql database

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

Postgresql: Configuration (2)



- ❑ **create DB account for user globus**
 - `% createuser globus`
 - answer following question with "yes"

- ❑ **execute setup script as user postgres**
 - `% . /usr/local/bin/globus-env-setup.sh`

- ❑ **create DB and initialise DB schema**
 - `% createdb rftDatabase`
 - `% psql -d rftDatabase -f $GLOBUS_LOCATION/\share/globus_wsrf_rft/rft_schema.sql`

Postgresql: Configuration (3)



- ❑ **configure Postgresql for Globus**

- > `vi $GLOBUS_LOCATION/etc/globus_wsrft/\`
`jndi-config.xml`

- use empty string as password

- ❑ **stop container**

- > `globus-stop-container`

- ❑ **start container**

- > `globus-start-container`

Here no error message must appear!



Reliable File Transfer (RFT): Test

- ❑ copy test file

```
$ cp /tmp/LrzGlobusWorkshop/transfer.xfr /tmp
```

- ❑ in /tmp/transfer.xfr replace "localhost" by FQDN

- ❑ create test file

```
$ touch /tmp/rftTest.tmp
```

- ❑ RFT test

```
$ rft -h <FQDN> -f /tmp/transfer.xfr
```

result: [...]

All transfers are completed



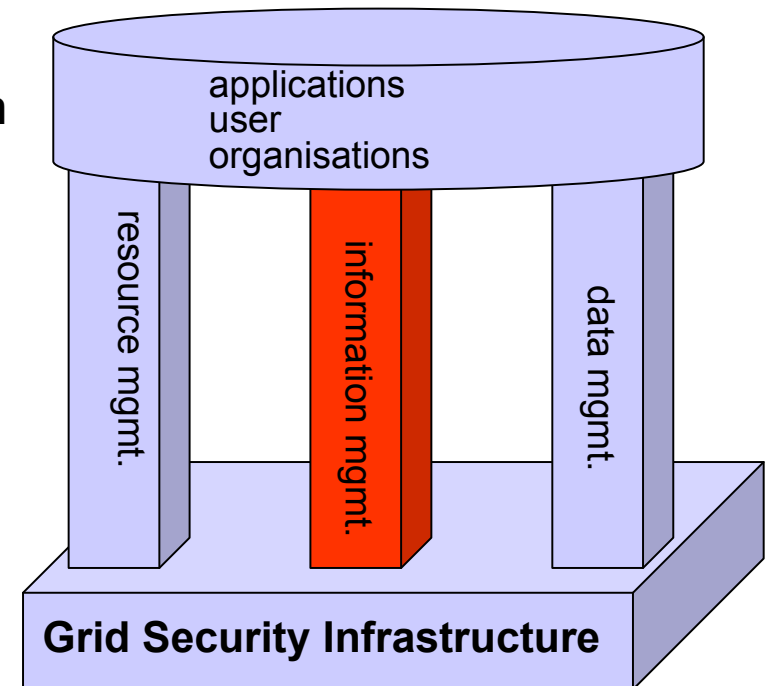
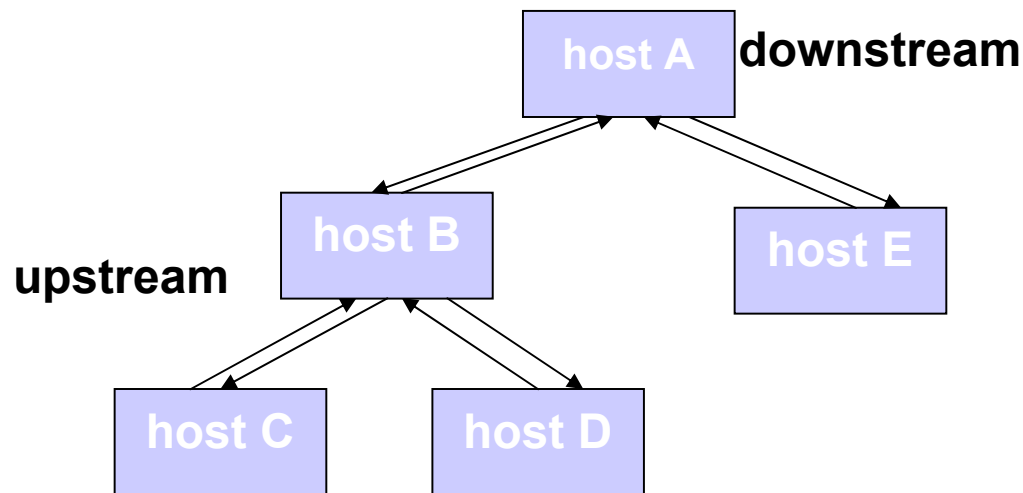
Monitoring and Discovery Service - MDS (1)

- ❑ **monitor and discover information about**
 - **application**
 - **resources**
 - **services**

- ❑ **WSRF-based services**
 - **index service**
 - **collects data**
 - **query/subscription**
 - **trigger service**
 - **triggering actions**

Monitoring and Discovery Service - MDS (2)

- hierarchy based structures



- archive service
- web browser
 - WebMDS



MDS4: configuration

- ❑ **construct monitoring hierarchy**

```
> vi $GLOBUS_LOCATION/etc/globus_wsrp_mds_index/\
  hierarchy.xml
```

```
<upstream>
```

```
  https://<parent-host>:8443/wsrp/services/DefaultIndexService
```

```
</upstream>
```

```
<downstream>
```

```
  https://<child-host>:8443/wsrp/services/DefaultIndexService
```

```
</downstream>
```

- ❑ **restart container**

```
> globus-stop-container; globus-start-container
```

Summary



- ❑ **Globus installation**
 - **Grid Security Infrastructure (GSI)**
 - **Grid Resource Allocation Management (GRAM)**
 - **Data Management (GridFTP, RFT)**
 - **Monitoring and Discovery Service (MDS4)**

GT4 Usage Statistics Collection



- ❑ **GT4 components send data for usage statistics collected by the Globus Alliance**

- ❑ **Transmission**
 - UDP packets to usage-stats.globus.org:4810
 - Data is sent by server, not by client

- ❑ **Affected Components:**
 - Java/C WS Core, GRAM, GridFTP, RFT, RLS

- ❑ **Transferred data (ia)**
 - all: component/data format identifiers, source IP/hostname, time stamps
 - GRAM: scheduler type, job type, some flags, success/failure, etc.
 - GridFTP: transfer type/size, transfer start/end, block/TCP buffer size, etc.

- ❑ **Further Information**
 - http://www.globus.org/toolkit/docs/4.0/Usage_Stats.html

Usage Stats: Motivation and Difficulty



❑ Motivation

- Globus Alliance
 - receives support from US government funding agencies
 - has to demonstrate that the scientific community is benefiting from their investment

❑ Difficulty

- Data privacy protection law in Germany / Europe
 - Transmission of person related data to the US is not permitted without explicit allowance by the user
 - IP addresses are regarded as person related data
 - Providing software which contains such functionality is only permitted after demonstrably informing the user

Usage Stats: Opt Out



□ How to disable the data transfer

- for Java Components:
in `$GLOBUS_LOCATION/etc/globus_wsrp_core/`
change in file `server-config.wsdd`:
`<globalConfiguration>`
 `<parameter name="usageStatisticsTargets"`
 `value=" [delete] "/>`
`</globalConfiguration>`
or remove the parameter

- for C Components:
change environment variable:
`export GLOBUS_USAGE_OPTOUT=1` (for sh/bash)
`setenv GLOBUS_USAGE_OPTOUT 1` (for csh/tcsh)



Contents for the exercises

- grid security infrastructure (GSI)**
 - **certificates**
 - **proxy**
 - **Certificate Authority**
- using available grid services**
- resource management**
 - **counter example**
- GridFTP and Reliable File Transfer (RFT)**
- Monitoring and Discovery Service (MDS)**

Grid Security Infrastructure - GSI (1)



- ❑ creating proxy for 200 hours and 30 minutes

```
$ grid-proxy-init -valid 200:30
```

```
Result: /tmp/x509up_u<UID>
```

```
Note: read and write permission only for the owner
```

- ❑ getting certificate information

```
$ grid-cert-info
```

```
Result: information including issuer, validity, ...
```

- ❑ getting proxy information

```
$ grid-proxy-info
```

```
Result: information including time left
```

Grid Resource Allocation and Management - GRAM (1)



- ❑ **WS oriented job submission command (GT4):**

```
$ globusrun-ws -submit -c /bin/date
```

```
$ globusrun-ws -submit -F <host> -c /bin/date
```

- ❑ **streaming the output**

```
$ globusrun-ws -submit -s -c /bin/date
```

- ❑ **output go to a file**

```
$ globusrun-ws -submit -s -so job.out -c \  
/bin/date
```

Using Resource Specification Language



❑ create RSL file

```
$ export GLOBUS_USER_HOME=$HOME ; vi job.rsl
<job>
  <executable>/bin/echo</executable>
  <argument>this is an example_string </argument>
  <argument>CoreGrid Globus Workshop</argument>
  <stdout>${GLOBUS_USER_HOME}/stdout</stdout>
  <stderr>${GLOBUS_USER_HOME}/stderr</stderr>
</job>
```

❑ rsl job submission

```
$ globusrun-ws -submit -job-description-file
  job.rsl
```

Using End Point References (EPR)



- ❑ create a batch job

```
$ vi batchJob.sh
#!/bin/sh
sleep 10
$ chmod +x batchJob.sh
```

- ❑ create an end point reference file

```
$ globusrun-ws -submit -b -o epr1.file \
-c $HOME/batchJob.sh
```

- ❑ request the status of the job

```
$ globusrun-ws -status -j epr1.file
```

- ❑ cancel the job

```
$ globusrun-ws -kill -j epr1.file
```



Using Available Grid Service (1)

- ❑ creating a counter resource

```
counter-create -s \  
https://<FQDN>:8443/wsrp/services/CounterService \  
-z none > epr1.file
```

- ❑ using the created counter resource

```
$ counter-add -e epr1.file 3  
Result: 3  
$ counter-add -e epr1.file 5  
Result: 8
```



Using Available Grid Services (2)

- ❑ creating another counter resource

```
counter-create -s \  
https://<FQDN>:8443/wsrp/services/CounterService \  
-z none > epr2.file
```

- ❑ using the newly created counter resource

```
$ counter-add -e epr2.file 4  
Result:4  
$ counter-add -e epr2.file 2  
Result:6
```

- ❑ using the previously created counter resource

```
$ counter-add -e epr1.file 7  
Result: 15
```

GridFTP



❑ copy: local → local

```
$ globus-url-copy gsiftp://localhost/etc/hosts \  
file:///tmp/hosts_copy
```

❑ copy: local → remote

```
$ globus-url-copy \  
file:///tmp/hosts_copy_<YourName> \  
gsiftp://<remote host name>/tmp
```

❑ copy: remote → remote (third party transfer)

```
$ globus-url-copy gsiftp://<remote host name1>\  
/hosts \  
gsiftp://<remote host name2>\  
/tmp/hosts_<login>
```



Reliable File Transfer

- ❑ **vi /tmp/transfer.xfr**
 - change some parameter to test

- ❑ **Test the changed transfer parameters**
 - `$ rft -h <FQDN> -f /tmp/transfer.xfr`

Monitoring and Discovery Service (MDS4)



❑ Test the registration

```
> wsrquery -s https://<FQDN>:8443/wsr/\
  services/DefaultIndexService '/*' |wc -l
Result: 6
```



Thank you for your attention!!!
Thank you for your attention!!!

**ENDLICH
GESCHAFFT !**

