Globus workshop

Hands-on session with Globus 5

GridKa Summer School 2010

Florian Zrenner and Jarno Laitinen (<lastname>@lrz.de)
Slides also from: Siew Hoon Leong (LRZ)

Leibniz Supercomputing Centre (LRZ), Garching near Munich
Bavarian Academy of Sciences and Humanities

Overview of the hands-on session

Goal: to be able to act as a Globus admin and user

- Installation of GT5 (ssh ready?)

After general Globus introduction and lunch (12:30-13:20):

- Authentication and authorisation:
  - Certificates
  - Authorisation file
- Globus components: configuration and usage
  - Interactive access
  - Data transfer
  - Job submission
  - MyProxy proxy storage service

Das ist ein Hammer

- These are hands-on slides, but with much information
  - Download slides from:
    - http://tinyurl.com/GT5-handson (add -p to URL for pdf)
  - I will tell then when you need to do something
    - Often marked with **`bold courier text`**
- Information for administrators: A

  Information for users (client software): C

Questions: Who..

- might install Globus in future (not just use it)?
- is familiar with Globus, but expects to hear about GT5?

Installation: Overview

- Where to download Globus
- How to install it from the sources

Installation: where to find GT5?

- GT 5.0.2 download available at www.globus.org
  - Documentation, Downloads and Support
- Source available
  - "Builds on Apple OS X, RedHat, Fedora Core, Debian, SuSE, FreeBSD, and Solaris"
- "Third Party Releases"
  - repositories for Fedora, RHEL, Debian and Ubuntu.
  - Partial Windows support (client side).

Installation:
A login to your hands-on machine

- Ready to login?
  - Windows without SSH? Download PuTTY:

http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html (http://bit.ly/1kyS98)

- Then login to your personal hands-on host:

```
ssh root@<your host> -p 24
```

A Installation: screen

- Run command

`screen`

- With screen it does not matter if the network connection gets broken

- Later
  - To leave screen (running): **`Ctrl/Strg + A + D`**
  - To get back to screen session: **`screen -rd`**

Installation: globus user and installation directory

- Create user "`globus`":

```
groupadd globus
useradd -m globus -G globus
passwd globus  (you can freely choose it)
```

- Create an installation directory:

```
mkdir /opt/globus-5.0.2
chown globus:globus /opt/globus-5.0.2/
```

- Create directory for the certificates:

```
mkdir -p /etc/grid-security/certificates
```

- Copy host certificate (`hostkey.pem` and `hostcert.pem`):

```
cp /root/host* /etc/grid-security/
```

Installation: Download and compilation

- As user "globus" (**su - globus**)
  - Download the Globus 5.0.2 sources:

    Use **wget** to download it from www.globus.org

    directly to the hands-on machine.
  - Unpack it:

    tar xjvf gt5.0.2-all-source-installer.tar.bz2
  - Go to directory and run ./configure:

    cd gt5.0.2-all-source-installer

    ./configure --prefix=/opt/globus-5.0.2
  - Run: **make**

Installation: ./configure (1)

- There are good to know switches for `./configure`
- Batch scheduling system (BSS) support: PBS (Torque), Condor, LSF and SGE e.g.

```
--enable-wsgram-pbs
```

- TCP wrappers mechanism for gsisshd:

```
--with-gsiopensshargs="--with-tcp-wrappers"
```

Installation: ./configure (2)

- Batch Scheduling System support must be compiled separately:
  - `make gram5-pbs` (or `gram5-sge`/`gram5-lsf`/`gram5-condor`)
- Optional features for GridFTP:
  - `make udt`
  - `make globus-xio-extra-drivers`

Compilation: make install

- Login again to your hands-on machine, if connection is lost:
  - `ssh root@<your host> -p 24`
  - `screen -rd`
- Then run: `make install`

A Globus environment variables


```
export GLOBUS_LOCATION=/opt/globus-5.0.2
export PATH=$GLOBUS_LOCATION/bin:$PATH
export LD_LIBRARY_PATH=$GLOBUS_LOCATION/lib:$LD_LIBRARY_PATH
export GLOBUS_TCP_PORT_RANGE=20000,25000
export GLOBUS_USAGE_OPTOUT=1
export GLOBUS_HOSTNAME=<set hostname here>
```

(csh / tcsh: `setenv GLOBUS_LOCATION /opt/globus-5.0.2`)

- Those could be e.g. in a file `/etc/profile.d/grid-env.sh`
  - **wget http://tinyurl.com/gridenv** (as root)
  - **mv grid-env.sh /etc/profile.d/**
  - **source /etc/profile.d/grid-env.sh**

Initiative for Globus in Europe

AUTHENTICATION & AUTHORISATION

A&A: Overview

- Personal certificate
- Host certificate
- Certificate Authority (CA) certificates
- Conversion .pem ←→ .p12

- Authorisation file

- Additional information e.g. Simple CA

A&A: Client's certificates

C Personal certificate: .pem, .p12

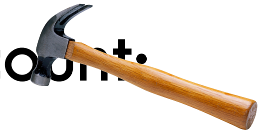
- PEM files:
  - `$HOME/.globus/usercert.pem` (public certificate part)
  - `$HOME/.globus/userkey.pem` (private key)
    - `chmod 400 userkey.pem`
  - Non-default place or name can be set with the environmental variables: `$X509_USER_CERT` and `$X509_USER_KEY`
- Instead of the `.pem` files a `.p12` file can be used:
  - `$HOME/.globus/usercred.p12`
    - `chmod 400 usercred.p12`
- In Windows put the files into: `%HOMEPATH%\.globus`
  - To create `.globus` start `cmd` program and run

```
mkdir %HOMEPATH%\.globus
```

A&A: Client’s certificates

C Personal certificate hands-on

- On **hands-on machine as root** create your user account:



```
useradd -m <your account>
passwd <your account>
```

- Login **to the login node**

```
ssh <your account>@<login host> -p 24
```

- Copy the files to the hands-on machine from the login node

```
scp -r .globus <your account>@<your hands-on host>:
```

A&A: CA certificates

A Where to find and put them?

- To authenticate certificates Certificate Authority (CA) files are needed Globus requires <hash>.0 and `<hash>.signing_policy` files.
  - The unique hash is a digest of subject name of the CA.
- CA files can be found e.g. via search-by-country functionality: http://www.eugridpma.org/
- SARA kindly provides a package for Globus

http://winnetou.sara.nl/deisa/certs/globuscerts.tar.gz

- Installation directory: `/etc/grid-security/certificates`
  - Non-default directory can be set with: `$X509_CERT_DIR`

A&A: CA certificates

A Certificate revocation list

- Each CA maintains a file of revoked certificates.
- `<hash>.crl_url` in certificates directory point to URL to download `<hash>.r0` files.
- There is a tool to update the files:

http://dist.eugridpma.info/distribution/util/fetch-crl/

- If not up-to-date an authentication failure may occur

  → fetch-crl to cron

- Globus command for CA check: `grid-cert-diagnostics`

A A&A: Installation of CA certificates

At hands-on machine **as root**

- Download and unpack CA certificates:

```
cd /etc/grid-security/certificates/
wget http://tinyurl.com/ca-packet
tar zxvf *
globus-update-certificate-dir
```

  - The last command required due openssl v. 1.0.0

    If interested see: http://www.cilogon.org/openssl1

A&A: Certificate conversion

- To create .pem files from .p12 file:

```
openssl pkcs12 -clcerts -nokeys -in usercert.p12 -out usercert.pem
openssl pkcs12 -nocerts -in usercert.p12 -out userkey.pem
chmod 0400 userkey.pem && chmod 0600 usercert.pem
```

- Browsers typically need a .p12 file. To create it from .pem files:

```
openssl pkcs12 -export -inkey userkey.pem -out \
usercert.p12 -name "Firstname Lastname" -in usercert.pem
```

A&A: Certificate conversion:

C Create a p12 from pems

- Open an ssh connection to login node with your account

```
ssh <your account>@<login node> -p 24
```

- Go to `$HOME/.globus` directory and create the p12 file:

```
cd $HOME/.globus

openssl pkcs12 -export -inkey userkey.pem \
-out usercert.p12 -name "Firstname Lastname" \
-in usercert.pem

chmod 600 usercert.p12
```

A&A: Get Certificate information

- To view e.g. validity of `usercert.pem` file:
  - `openssl x509 -in $HOME/.globus/usercert.pem \`
    `-text -noout`
  - OR just run **`grid-cert-info`** Globus command as `gs0xx`.
- For p12:
  - Again, you can use a Globus command: `grid-cert-info`
  - OR with `openssl` even temporary pem file is needed
    - `openssl pkcs12 -in cert.p12 -out temp.pem`
      (asks passwords)
    - `openssl x509 -in temp.pem -noout -enddate`
    - `rm temp.pem`

C
A&A: Proxy certificate

- To create proxy: **`grid-proxy-init`**
  - More information with `-debug`
- The default location: `/tmp/x509up_${UID}`
- By default valid for 12 hours (`-valid <h:m>`).
- Some Globus commands require that proxy is valid e.g. 3 h
- To view information: **`grid-proxy-info`**

A&A: Certificate security issues

- For security reasons you can delete proxy on the machine when you do not need it anymore:

  grid-proxy-destroy

- The proxy file is readable only by your account.

AUTHORISATION

Who can use the service?

A&A: grid-mapfile

- **As normal user** check your certificate's DN:

```
grid-cert-info -subject
```

- **As root** at hands-on machine

```
$GLOBUS_LOCATION/sbin/grid-mapfile-add-entry \
    -dn "<Distinguished Name>" -ln <username>
```

(verify with `cat /etc/grid-security/grid-mapfile`)

- To delete an entry:

```
grid-mapfile-delete-entry
-dn "<Distinguished Name>" -ln <username>
```

- To check if any duplicate DNs and the accounts exists:

```
$GLOBUS_LOCATION/sbin/grid-mapfile-check-consistency
```

A&A: Additional information (1): A SimpleCA

- For testing and internal purpose Globus provides SimpleCA to act as a Certificate Authority.
- `$GLOBUS_LOCATION/setup/globus/setup-simple-ca` script can install CA files to any directory with

  `-dir` option. See more options with `-usage`.
- The script will create a tar.gz packet of the CA files to be distributed on the machines where needed.
- To sign the certificate request:

```
grid-ca-sign -in usercert_request.pem
 -out usercert.pem
```

- See also SimpleCA Admin Guide: http://bit.ly/cDdC8q

A&A: Additional information (2)

- **Time** settings of client and server must be within 5 minutes tolerance (otherwise the authentication can fail).

- Host certificate DN must have the **fully qualified host name.**
- If the host certificate does not match FQHN the client needs to specify the DN in Globus command parameter.

INTERACTIVE ACCCESS via GSI-OpenSSH

GSI-SSH: Overview

Administration

- Configuration of Globus gsisshd service

Client

- Globus `gsissh` command-line client
- Java GSISSH-Term usage

GSI-SSH: sshd configuration

A init.d script and logging

- Globus provides init.d script for gsissh daemon.
- **As root** at hands-on machine:

```
cp $GLOBUS_LOCATION/sbin/SXXsshd \
   /etc/init.d/gsisshd

chmod 744 /etc/init.d/gsisshd
```

GSI-SSH: sshd configuration

A init.d script and logging

- Optional: To differentiate from normal `sshd` in log file `/var/log/messages`:
  - `mv $GLOBUS_LOCATION/sbin/sshd \ $GLOBUS_LOCATION/sbin/gsisshd`
- In `/etc/init.d/gsisshd` **correct**

  `# Provides: gsisshd` (there is already sshd)

  `SSHD=${sbindir}/gsisshd` (if your renamed sshd)

  `PID_FILE` to `gsisshd.pid`
- Disable the usage statistic collection by adding: `GLOBUS_USAGE_OPTOUT=1`

GSI-SSH: configuration

A sshd_config and ssh_config

- `sshd_config` (server) and `ssh_config` (client) in

  `cd $GLOBUS_LOCATION/etc/ssh/`

- **Edit `sshd_config`** and change port from 22:

  `Port 2222` (no comment mark #!)

- You can disable protocols which you do not need e.g.

```
Protocol 2 # no 1 available
RSAAuthentication no
PubkeyAuthentication no
PasswordAuthentication no
ChallengeResponseAuthentication no
```

GSI-SSH: configuration
A additional information

- "If compiled with **PAM support** (`--with-pam`) set `"UsePAM yes"` in `$GLOBUS_LOCATION/etc/ssh/sshd_config` after installation. "

- If compiled with **TCP wrapper** edit `/etc/hosts.allow` e.g. `2222:ALL:ALLOW`

- Privilege separation method: See the required steps: http://grid.ncsa.illinois.edu/ssh/admin.html#privsep

GSI-SSH:
A /etc/services and start-up

- Edit `/etc/services` e.g. for `netstat -tap`:

  `gsisshd 2222/tcp`

- To start it now: **`/etc/init.d/gsisshd start`**

- To start gsissh during the boot:

  `/sbin/chkconfig -a gsisshd`

GSI-SSH

Client Part

GSI-SSH: gsissh client

- Usage of command line client:
  - Syntax: `gsissh [-p <port>] [account@]host`
    - Use a **full host name**
    - Debug: `-v` or `-vv`
    - By default it uses the **port** set in

      `$GLOBUS_LOCATION/etc/ssh/ssh_config`

- As your normal user account on your hands-on host:
  - **`grid-proxy-init`** (if not yet done)
  - **`gsissh <your hostname> -p 2222`**

GSISSH-Term: Introduction

- Java terminal client.
- 3rd party software (not from Globus)
- Supports .pem, .p12, browser certificates and can fetch a proxy stored at a MyProxy service.

- Java 5 or 6 needed.
- Java Cryptography Extension libraries might be needed

  You can find it in the end of the list

http://www.oracle.com/technetwork/java/javase/downloads/index.html (http://bit.ly/bMkbpo)

GSISSH-Term: configuration

C

Java Cryptography Extension

- To find where java is installed

  readlink `which java` *(Linux)*
  ($(ls -la `which java` | sed 's/.-> \(.*\)/\1_home/'))* *(Mac)*
  *C:\Program Files\Java\jre*** *(Windows)*

- Backup your already existing files (optional)

  cd $JAVA_HOME/lib/security

  mv US_export_policy.jar US_export_policy.jar_orig
  mv local_policy.jar local_policy.jar_orig

- Replace the files US_export_policy.jar and local_policy.jar

  cd <back to unpacked zip file directory>
  cp US_export_policy.jar $JAVA_HOME/lib/security
  cp local_policy.jar $JAVA_HOME/lib/security

GSISSH-Term:
Java Webstart: start

- At your laptop / PC start Java Webstart version

  Surf to http://tinyurl.com/gsissh

- It is LRZ's version which takes care of CA certificates and supports using Safari browser certificate.
- There appear two "digital signature cannot verified" windows which you have to accept.

Copy personal certificate to your personal computer

- With **Linux** / **Mac** very easy:

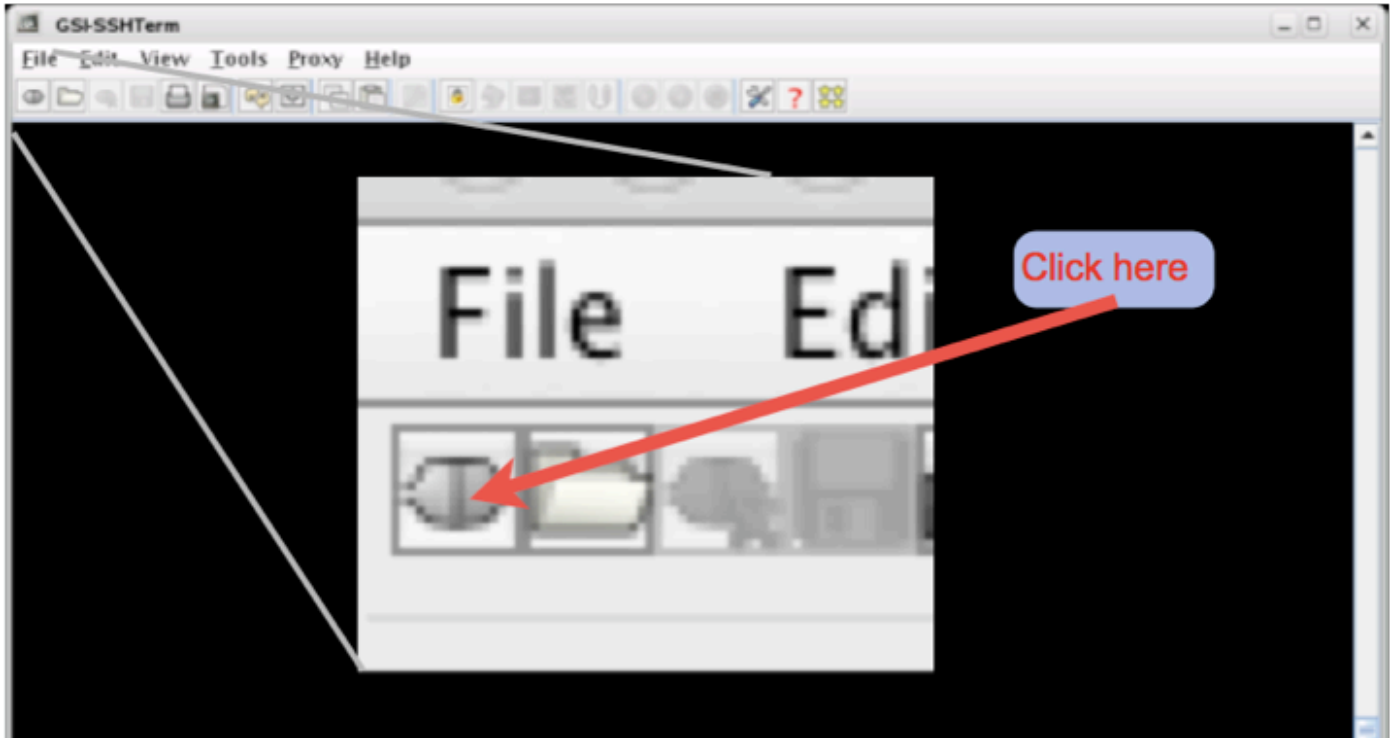
```
scp -r <account>@<login host>:.globus $HOME
```

- With **Windows**
  - Download file from login host with your account and GSISSH-Term:
    - Click *New Connection - Advanced options*
    - Set *host, username, port* and *select passwords*. Connect to the host.
    - Select *Tools - SFTP Session*
      - Select *File - Download*
      - Double-click `usercert.pem` and `userkey.pem` in `.globus` to download.

GSISSH-Term: start-up

Login to hands-on machine

- Login to your hands-on machine
  - Open a login window (see picture).
  - Set the hostname and press OK.



GSISSH-Term: start-up

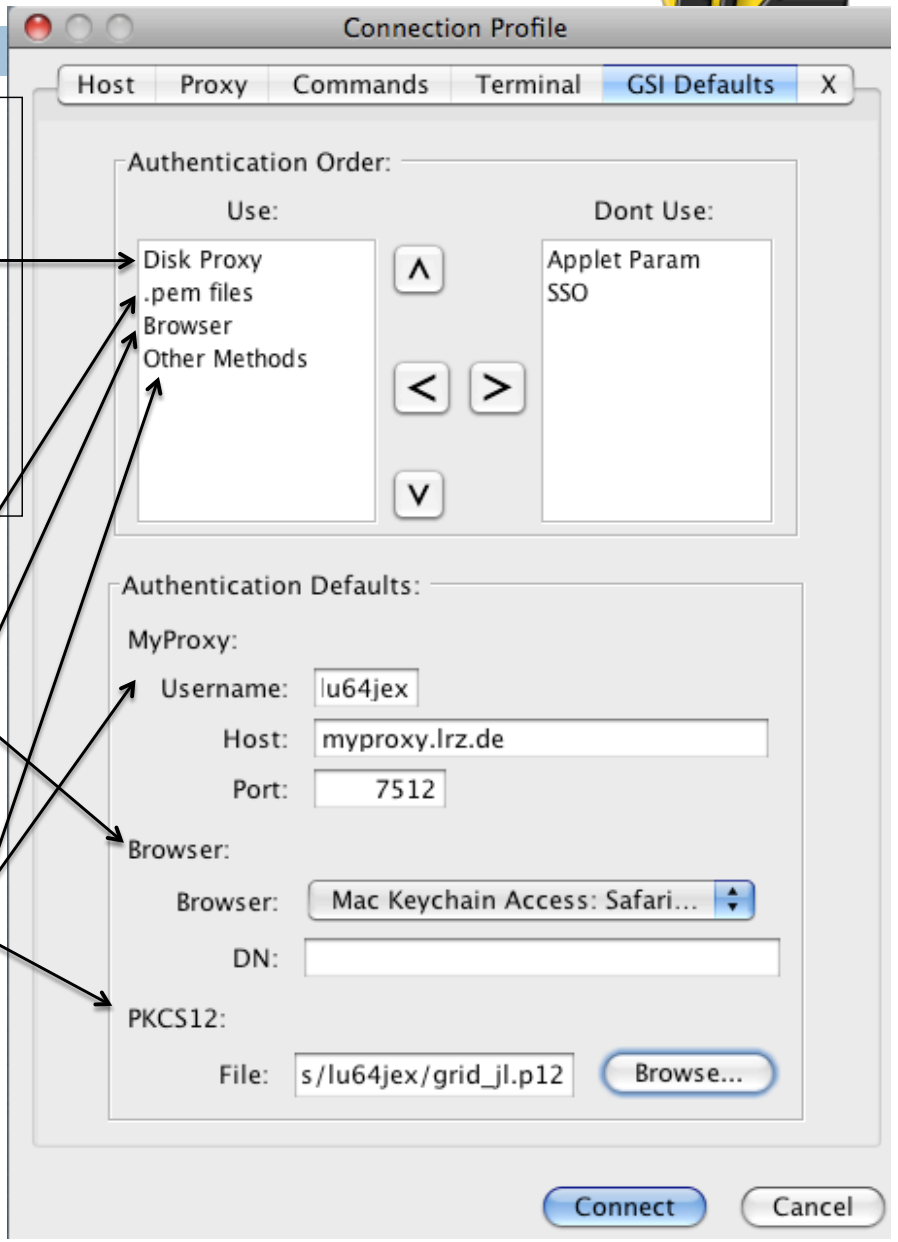
Authentication methods

A proxy can exist already due to:
- previous GSISSH-Term session
- Globus grid-proxy-init
- Short Lived Credential Service

- .pem files in .globus directory

- A web browser certificate

Other Methods:
- MyProxy service
- PKCS12 (.p12) file



DATA TRANSFER with GridFTP

GridFTP: Overview

- Administration
  - Start-up script (xinet.d)
  - Firewall issues
- Client
  - Globus `globus-url-copy`
  - UberFTP (3rd party)
  - Graphical user interface clients (Globus and 3rd party)

GridFTP: xinet.d script (1)

- xinet.d template:

```
service gsiftp
{
    instances     = 100
    socket_type   = stream
    protocol      = tcp
    wait          = no
    user          = root
    bind          = <your host IP>
    env     += LD_LIBRARY_PATH=/opt/globus-5.0.2/lib
```

continues on the next slide...

GridFTP: xinet.d script (2)


```
    env             += GLOBUS_TCP_PORT_RANGE=20000,25000
    env            += GLOBUS_HOSTNAME=<YOUR HOSTNAME>
    server    = /opt/globus-5.0.2/sbin/globus-gridftp-server
    server_args  = -i -l /opt/globus-5.0.2/var/gridftp.log
  -d error,warn,info -log-filemode 0600 -disable-usage-
  stats
    nice            = 10
    disable         = no
    port            = 2811
}
```

- To improve disk performance a block size option can be set e.g. `-bs 16777216`

GridFTP: xinet.d script (3)

- As root on hands-on machine:

```
cd /etc/xinet.d
wget http://tinyurl.com/gsiftp-xinetd
mv gsiftp_xinet.txt gsiftp
```

- Correct:
  - The IP address for `bind` (`/sbin/ifconfig`)
  - The hostname for `GLOBUS_HOST` (`hostname -f`)
- These settings help in multiple interface cases.

GridFTP: firewall (1)

- **Control process** port is by default: 2811

- **Data port range**
  - Varies often from a hundred to some thousands e.g. 20000,25000 or 50000,50100
  - Port range should be set to the same values as used by your grid partner sites.
  - The needed amount depends on the estimated number of the clients.

GridFTP: firewall (2)

- Client can set port range for **outgoing** firewall

```
export GLOBUS_SOURCE_PORT_RANGE=20000,25000
```

- Data port range is also used by the Globus job submission service for file transfer.

A GridFTP: Service start-up

- start-up:

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

GridFTP: O/S settings (1)

- Bottleneck can be too low TCP buffer settings in operating system.
- New Linux kernel versions (>=2.6.17) should tune itself.
- E.g. in Linux root can adjust them with `sysctl -w` command or put them permanently into

  `/etc/sysctl.conf`
- Optimal values are hard to find.
- See more information from http://kb.pert.switch.ch

GridFTP: O/S settings (2)

- Example settings from `/etc/sysctl.conf`:

```
net.ipv4.tcp_rmem = 4096 2097152 8388608
net.ipv4.tcp_wmem = 4096 2097152 8388608
net.core.rmem_default = 2097152
net.core.wmem_default = 2097152
net.core.rmem_max = 8388608
net.core.wmem_max = 8388608
net.core.netdev_max_backlog = 2000
```

GridFTP: extra I/O: UDT

- Instead of TCP transfer also UDP based UDT protocol is available.
- Might be useful in case of high latencies.

- Needs to be compiled (`make udt`) and configured (`-dc-whitelist udt,gsi,tcp` in xinet.d script).
- `globus-url-copy` has `-udt` parameter

GridFTP: extra I/O: Netlogger (1)

- Netlogger can provide some useful information of bottlenecks.
- Must be enabled in globus compilation step:
  - `--enable-netlogger`
- See more from :

http://www.cedps.net/index.php/Gridftp-netlogger

GridFTP: extra I/O: Netlogger (2)

- Example output of `globus-url-copy` with '-nlb' switch:

```
Total instantaneous throughput:
  disk read    = 2278.8 Mbits/s
  disk write   = 1381.0 Mbits/s
  net read     = 664.3 Mbits/s
  net write    = 288.8 Mbits/s
Bottleneck: Unknown
```

GridFTP

Client Part

GridFTP: globus-url-copy syntax

- Example:

```
globus-url-copy file:///$PWD/source.txt \
gsiftp://gks-1-101.fzk.de/~/target.txt
```

  - Source: **local** machine (no gridftp server): **`file:///`**`path/file`
  - Target: **GridFTP** server: **`gsiftp://`**`host<:port>/path/file`

- ~ can be used to refer to home directory.
- Paths must be **absolute**.

GridFTP: globus-url-copy switches

- More verbose output: -vb
- Copy files from subdirectories: -r
- Create destination directories if needed: -cd
- http://www.globus.org/toolkit/docs/5.0/5.0.2/data/gridftp/user/#gridftpUser (http://bit.ly/cNpSBk)
- Try at hands-on machine with your normal account:

```
globus-url-copy -vb \
file:///etc/grid-security/grid-mapfile  \
gsiftp://<your host>/~/
```

GridFTP: globus-url-copy performance options

- Optimal value depends on TCP settings of kernel, latency, bottlenecks. Just try now with e.g.
  - **Parallel streams:** `-p 4`
  - **TCP buffer** size: `-tcp-bs 4m`
  - **Concurrent FTP connections:** `-cc 2`
- If multiple data nodes are available following might help:
  - `-stripe`
  - `-sbs 0` (so called partitioned block size)

GridFTP: Mode E

- In `gsiftp://` → `gsiftp://` and with

  `-p <number>` transfer so called mode E is used.

- Data sending server establishes data channel
  - Data port range must be open on target server (firewall!)

- Can be more efficient than normal stream mode.

GridFTP: reliability options

- Client can save status to a file to recover from some failures

```
#!/bin/sh
STATEF=statusfile.txt;
while [ ! -e $STATEF -o -s $STATEF ]; do
globus-url-copy -restart -rst-timeout 10 -vb -dumpfile $STATEF \
gsiftp://srchost/srcdirpath/ gsiftp://dsthost/dstdirpath/;
done;
```

  - Dumpfile contains untransferred URLs during the transfer
  - File will be emptied if transfer succeeds
  - If empty file exists no transfer is done

GridFTP: gsiscp client

- Globus provides also simple `gsiscp` client, which syntax is familiar from normal scp command:
  - `gsiscp -P 2222 source.txt <host>:`
  - `gsiscp -P 2222 <host>:source.txt target.txt`
  - 3rd party transfer did not seem to work

GridFTP: UberFTP client (1)

- UberFTP client can be installed from

  http://dims.ncsa.illinois.edu/set/uberftp/

- It provides normal file transfer (gsiftp, ftp and file)
- Why to use it?
  - Interactive (login) to GridFTP server.
  - Provides commands (-cmd): `cat`, `chgrp`, `chmod`, `dir (ls)`, `mkdir`, `rm`, `rmdir` and `size`.
    - `help` lists available commands

GridFTP: UberFTP client

C

Example: getting size of a file

- In interactive session:

```
UberFTP> size test.rsl
25
```

- ...or execute the command directly from the shell:

```
$ ./uberftp lxgt2.lrz-muenchen.de "size test.rsl"
220 lxgt2.lrz-muenchen.de GridFTP Server 2.8
   (gcc64dbg, 1217607445-63) [Globus Toolkit 4.0.8]
   ready.
230 User lu64jex6 logged in.
25
```

GridFTP: GUIs

- Following are available, but perhaps not very robust and perfect.
- Pre-alpha version of Globus Java Webstart client:

http://www-unix.globus.org/cog/demo/ogce/ftp.jnlp

(http://tinyurl.com/ftpgui)

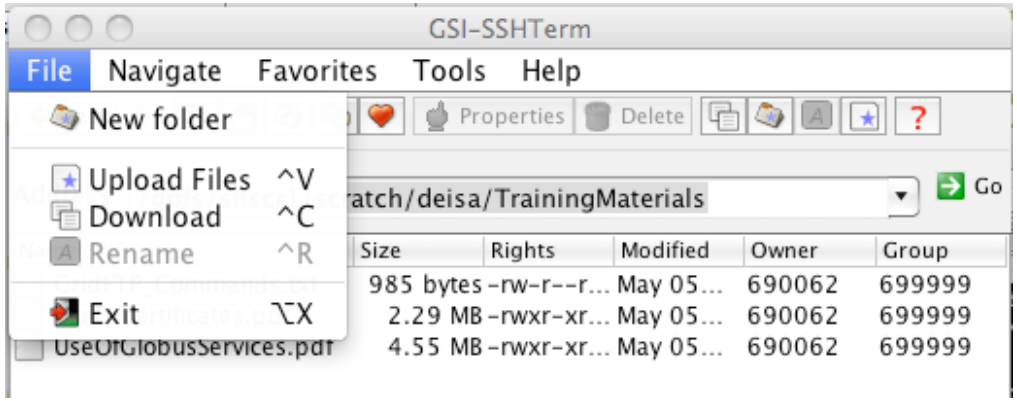
- SGGC is a Java based client. LRZ's usage instructions:

http://www.grid.lrz-muenchen.de/en/mware/globus/client/sggc.html

- A standalone or Eclipse plug-in based Java client: http://bi.offis.de/gridftp/downloads.html

GridFTP: GSISSH-Term

- The Java Webstart tool has a simple file transfer GUI
- Allows to upload and download files from/to your PC
  - Connect first to your hands-on machine via gsissh-term
  - Select: Tools - SFTP Session



JOB SUBMISSION via GRAM5

GRAM5: Overview

- Administration:
  - Start-up script
  - Configuration
- Client:
  - `globus-job-run`
  - `globusrun`
  - a batch job (non-blocking)
  - a batch scheduling system jobs
  - GRAM5 job scripts (RSL)

GRAM5: configuration

A xinet.d script example


```
service gsigatekeeper
{
  socket_type  = stream
  protocol     = tcp
  wait         = no
  user         = root
  env          += GLOBUS_TCP_PORT_RANGE=20000,25000
  env          += LD_LIBRARY_PATH=/opt/globus-5.0.2/lib
  server       = /opt/globus-5.0.2/sbin/globus-gatekeeper
  server_args  = -conf /opt/globus-5.0.2/etc/globus-gatekeeper.conf
  disable      = no
  # bind       =  <optional if one interface. Otherwise set here IP address>
  port         = 2119
}
```

GRAM5: configuration

A xinet.d for hands-on machine

- **As root:**

```
cd /etc/xinet.d/

wget http://tinyurl.com/gsigatekeeper

chmod 744 gsigatekeeper
```

GRAM5: configuration

A globus-gatekeeper.conf

- `globus-gatekeeper.conf` was referred in the start-up script:

```
-x509_cert_dir /etc/grid-security/certificates
-x509_user_cert /etc/grid-security/hostcert.pem
-x509_user_key /etc/grid-security/hostkey.pem
-gridmap /etc/grid-security/grid-mapfile
-home /opt/globus-5.0.2
-e libexec
-logfile var/globus-gatekeeper.log
-port 2119
-grid_services etc/grid-services
-inetd
```

GRAM5: configuration

A Default job manager

- `$GLOBUS_LOCATION/etc/grid-services`

  contains symbolic link (`ln -s`) for the default job manager

```
lrwxrwxrwx 1 globus globus-g   15 2010-07-19 10:58 jobmanager
  -> jobmanager-fork
-rw-r--r-- 1 globus globus-g  188 2010-07-19 15:30
  jobmanager-fork
-rw-r--r-- 1 globus globus-g  187 2010-07-19 15:58
  jobmanager-sge
```

GRAM5: configuration

A etc/globus-fork.conf et al.

- The `$GLOBUS_LOCATION/etc` directory contains LRMS configuration files e.g. `globus-fork.conf, globus-sge.conf` referring to respective log files.
  - You should check that "`make install`" has found the log files.
- Fork's log file is in `$GLOBUS_LOCATION/var/` with following permissions (622).
- There is also `globus-gatekeeper.log` (600).

GRAM5: configuration

globus-job-manager.conf (1)

- $GLOBUS_LOCATION/etc/globus-job-manager.conf:

```
-home "/opt/globus-5.0.2"
-globus-gatekeeper-host gks-1-101.fzk.de
-globus-gatekeeper-port 2119
-globus-gatekeeper-subject "/C=DE/O=GermanGrid/
OU=dech-school/CN=gks-1-101.fzk.de"
-globus-host-cputype x86_64
-globus-host-manufacturer unknown
-globus-host-osname Linux
-globus-host-osversion 2.6.34-12-desktop
```

GRAM5: configuration

globus-job-manager.conf (2)


```
-globus-toolkit-version 5.0.2
-stdio-log "$(HOME)"
-log-levels 'FATAL|ERROR'
-state-file-dir /opt/globus-5.0.2/tmp/gram_job_state
-globus-tcp-port-range 20000,25000
-stdio-log "$(HOME)"
-disable-usagestats
-log-levels 'ALL|FATAL|ERROR'
-state-file-dir /opt/globus-5.0.2/tmp/gram_job_state
```

- It is not recommend state-file-dir to be on a shared file system.

GRAM5: Configuration files for the LRMS

- Batch scheduling system might require some options which are not by default in Globus BSS adaptors.
- The scripts, which generate the BSS job script, are in Perl.

  E.g. to set into PBS script `nodes` from `host_count` of a Globus job:

```
print JOB '#PBS -l nodes=', $description->host_count(), "\n";
```

- The files (e.g. `pbs.pm`) are located in

`$GLOBUS_LOCATION/lib/perl/Globus/GRAM/JobManager`

- Update the files if path for BSS commands changes.

GRAM5: system configuration

A firewall and /etc/services

- In `globus-job-manager.conf` can be set **gridftp data port range** (e.g. 20000,25000).
- In `globus-gatekeeper.conf` and in xinet.d script is set the **gatekeeper port,** which is by default 2119.
- In `/etc/services` can be set:

```
gsigatekeeper 2119/tcp
```

GRAM5: Start

- Start gatekeeper:

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

GRAM5

Client Part

GRAM5: general hints for client

- If your job seems to get stuck try to kill your job-manager processes:

```
killall globus-job-manager
```

- For logs see your home directory (`ls -lart gram*`)
- See also in `$HOME/.globus/job/`

- Gatekeeper log
  - `$GLOBUS_LOCATION/var/gatekeeper.log`
  - It might be visible for administrator only.

GRAM5: globus-job-run

C blocking submission

- With `globus-job-run` it is simple to submit a job
- It is a blocking command (i.e. it does not release the shell until the job finishes)

- Example: **`globus-job-run <your hostname> /bin/date`**
- It is possible to give various parameters e.g. directing standard output or error. See `-help` or User Guide http://bit.ly/c8FYK0

GRAM5: globus-job-submit

C non-blocking command submission

- `globus-job-submit` returns command to shell right after the submission and gives **job contact string**

- `globus-job-status <job_contact_string>`
- `globus-job-get-output <job_contact_string>`
- `globus-job-clean <job_contact_string>`
  - needed after job status is DONE

```
gs002@gks-1-101:~> globus-job-submit gks-1-101.fzk.de /bin/date
https://gks-1-101.fzk.de:24384/16073723895661987071/15700714982003976859/
gs002@gks-1-101:~> globus-job-status https://gks-1-101.fzk.de:24384/16073723895661987071/15700714982003976859/
DONE
```

GRAM5: globus-job-submit non-blocking command submission (2)

- gs002@gks-1-101:~> globus-job-get-output https://gks-1-101.fzk.de:24384/16073723895661987071/15700714982003976859/

```
Sat Sep  4 21:02:43 CEST 2010
```

- gs002@gks-1-101:~> globus-job-clean https://gks-1-101.fzk.de:24384/16073723895661987071/15700714982003976859/

```
    WARNING: Cleaning a job means:
        - Kill the job if it still running, and
        - Remove the cached output on the remote resource

    Are you sure you want to cleanup the job now (Y/N) ?
Y

Cleanup successful.
```

GRAM5: globusrun and RSL (1)

- `globusrun` command is the most suitable for real "production" jobs
- It takes as a parameter a script written in Globus Resource Specification Language (RSL).
- GRAM5 uses different syntax than Globus version 4.

- RSL script can be passed:
  - from a command-line (in " ")

```
gs002@gks-1-101:~> globusrun -s -r gks-1-101.fzk.de "&(executable=/bin/date)"
Sat Sep  4 21:10:40 CEST 2010
```

  - in an RSL file

GRAM5: globusrun and RSL (2)

- The simplest RSL script is specifying the executable:

```
&(executable=/bin/date)
```

  - Please store this line to a file `job.rsl`

- The `&` is needed only on the first row.

- All rows are surrounded in ().

GRAM5: globusrun command line parameters

- Submission which streams (`-s`) standard output and error to the display

```
globusrun -s -r <your host> -f job.rsl
Thu Aug 12 17:04:13 CEST 2010
```

- For complete list of possible attributes see

  http://bit.ly/d6cQbL

GRAM5: globusrun and RSL (3)

- Some useful RSL attributes:

```
& (rsl_substitution = (DIR "/tmp/my_dir") )
(environment = (MSG 'Hello'))
(stderr = $(DIR)/stderr.txt)
(stdout = $(DIR)/stdout.txt)
(executable=/usr/bin/env)
(* (arguments="Hello ") *)
```

- A variable set in `environment` is not possible to use in RSL script.

GRAM5: globusrun non-blocking operation (1)

- With **-b** option non-blocking command is sent and a contact string is then returned.
- Edit `job.rsl`:

```
&(executable=/bin/sleep)
(arguments=1000)
```

- Run:

```
globusrun -b -r <your host> -f job.rsl
```

GRAM5: globusrun non-blocking operation (2)

- Status query:

```
globusrun -status <job_contact_string>
```

  - Possible job statuses: ACTIVE, FAILED, SUSPENDED, DONE, UNSUBMITTED, STAGE_IN, STAGE_OUT and UNKNOW JOB STATE

- Cancelling the job:

```
globusrun -k <job_contact_string>
```

GRAM5: File staging (1)

- The possible steps in a job are:
  - File **stage in**: files from the client to the GRAM5 server
  - File **stage out**: files from the GRAM5 server to the client
  - File **clean-up**: remove the files from the GRAM5 server

- Internal or external GridFTP can be used.
- To use internal file transfer mechanism (GASS) uses predefined variable:

```
$(GLOBUSRUN_GASS_URL)/$(HOME)/input.txt
```

- Used on the client side

GRAM5: File staging (2)

GASS example (1)


```
& (executable=$(HOME)/compile.sh)
(stdout=stdout.txt)
(stderr=stderr.txt)
(file_stage_in =
($(GLOBUSRUN_GASS_URL)/$(HOME)/compile.sh $(HOME)/
  compile.sh))
(file_stage_out =
(stderr.txt $(GLOBUSRUN_GASS_URL)/$(HOME)/stderr.txt)
(stdout.txt $(GLOBUSRUN_GASS_URL)/$(HOME)/
  stdout.txt))
(file_clean_up=stdout.txt)
(file_clean_up=stderr.txt)
```

GRAM5: File staging (3)

C GASS example (2)

- In previous job script the executable script (`compile.sh`) could have been for example:

```
#!/bin/bash -l
mpicc mpi_test.c -o mpi_test
chmod 755 mpi_test
```

- The command to submit the job:

```
globusrun -s -r <HOSTNAME> -f compile.rsl
```

GRAM5: File staging (4)

C GridFTP example


```
& (rsl_substitution = (GRIDFTP_SERVER gsiftp://<GFTPHOST>))
  (executable=/bin/cat)
  (arguments=$(HOME)/input_file)
  (stdout=stdout.txt)
  (stderr=stderr.txt)
  (file_stage_in = ($(GRIDFTP_SERVER)/$(HOME)/input_file $
  (HOME)/input_file))
  (file_stage_out = (stderr.txt $(GRIDFTP_SERVER)/$(HOME)/
  stderr.txt)
                    (stdout.txt $(GRIDFTP_SERVER)/$(HOME)/
  stdout.txt))
  (file_clean_up = $(HOME)/input_file)
```

Example how to use optional common variable `GRIDFTP_SERVER`

GRAM5: File staging (5)

GridFTP example (2)

- Try script from previous slide:
  - Download script: **wget http://tinyurl.com/filestage-rsl**
  - Fix the hostname.
- Create file **`input.txt`** to `$HOME` and put there some text.
- Run:

```
globusrun -s -r <HOSTNAME> -f filestage.rsl
```

- See output:

```
cat $HOME/std*
```

GRAM5: MPI job example (1)

- MPI job should be submitted to batch scheduling systems. Since it will take time so `-b` option is used:

```
globusrun -s -b -r gram5.lrz.de/jobmanager-pbs
-f mpigt5.rsl
```

- In the next slide is an example of MPI RSL job script.
  - The number of the MPI processes is set with `count`.
  - `job_type` must be set to `mpi`.
- You can set needed memory (in MBs) and wall-clock time (in minutes).

GRAM5: MPI job example (2)

C RSL script gt5mpi.rsl (1)


```
&(executable=$(HOME)/mpi_test)
(job_type=mpi)
(count=2)
(max_wall_time=20)
(max_cpu_time=10)
(max_memory=10)
(stdout=stdout.txt)
(stderr=stderr.txt)
```

GRAM5: MPI job example (3)

RSL script gt5mpi.rsl (2)


```
(file_stage_out =
(stderr.txt $(GLOBUSRUN_GASS_URL)/$(HOME)/
 stderr.txt)
(stdout.txt $(GLOBUSRUN_GASS_URL)/$(HOME)/
 stdout.txt))
(file_clean_up=stdout.txt)
(file_clean_up=stderr.txt)
```

GRAM5: Advanced RSL:
C Proxy renewal operation & dbg

- By default proxy certificate lives 12 hours
- If proxy expires and need to get results of the job:
  - `grid-proxy-init`
  - `globusrun -r <host> \`
    `"&(restart=<job_contact_string>)"`
- New debug feature in v. 5.0.2:

  to save Globus internal job descriptions add:

  `(save_job_description=yes)`

GRAM5: Auditing & accounting

- Globus does not support **accounting**.
  - Batch scheduling system can provide accounting information.
  - Tip: add a specific string to job name attribute in BSS Perl file to identify Globus jobs

- Globus provides some **auditing** database functionality.
  - You can look at http://bit.ly/cCVCpK

PROXY STORAGE SERVICE

MyProxy

MyProxy: proxy storage service

- Administration
  - Security remarks
  - Configuration of the service (MyProxy and system)
  - Logging and start-up

- Client
  - Storing proxy with Globus commands
  - Fetching the stored proxy with Globus command

3rd party tools:

  - Firefox plug-in
  - Java Webstart application

MyProxy: security remarks

- Root can access the proxies on the server.
- A dedicated machine for this service only with restricted access and strict firewall
- Port 7512 (by default) must be open for Internet
- Subscribe yourself to the security alert mailing lists:
  http://grid.ncsa.illinois.edu/myproxy/security/

MyProxy: configuration A compilation, init.d script

- If only MyProxy needs to be installed:

```
make gsi-myproxy
```

- You find **init.d** and xinet.d start-up scripts from

```
cd $GLOBUS_LOCATION/share/myproxy/
cp etc.init.d.myproxy /etc/init.d/myproxy
chmod u+x /etc/init.d/myproxy
```

  - **Set** (at least) **GLOBUS_LOCATION**
  - Activate the init.d script for the boot:
    - **chkconfig -a myproxy**

MyProxy: configuration

A removing expired proxies

- To delete invalid credentials:

```
cp $GLOBUS_LOCATION/share/myproxy/myproxy.cron \
/etc/cron.hourly
chown globus /etc/cron.hourly/myproxy.cron
chmod u+x /etc/cron.hourly/myproxy.cron
```

- Correct **$GLOBUS_LOCATION**

- Put it to run on `globus` user's crontab (**crontab -e**):

```
59 * * * * /etc/cron.hourly/myproxy.cron > /dev/null
```

MyProxy configuration: myproxy-server.config file (1)

- Template file is located in

```
cp $GLOBUS_LOCATION/share/myproxy/myproxy-server.config \
   $GLOBUS_LOCATION/etc/
```

- It is a long file with many commented settings.

- You can add there:

```
disable_usage_stats   "true"
```

MyProxy configuration: A myproxy-server.config file (2)

Uncomment the following settings from the template:

```
accepted_credentials          "*"
authorized_retrievers         "*"
default_retrievers            "*"
authorized_renewers           "*"
default_renewers              "none"
authorized_key_retrievers     "*"
default_key_retrievers        "none"
trusted_retrievers "*"
default_trusted_retrievers "none"
```

MyProxy: start-up and logging

- Add into `/etc/services` row

```
myproxy-server   7512/tcp     # Myproxy server
```

- To start either init.d

```
/etc/init.d/myproxy start
```

- The logs are in `/var/log/messages`
- More verbose messages with `-d` and `-v` parameters

MyProxy

Client part

MyProxy: Storing the proxy with myproxy-init Globus command

`myproxy-init -a -s myproxy.lrz.de`

- It will ask password for your private key. A proxy is not enough.
- Then it asks twice new passphare to secure your credential at MyProxy server

- Username (`-l <un>`) (by default: user account name)
- Credential lifetime (`-c <hours>`) (one week= 168 h)
- Proxy lifetime (-t <hours>) (12 h)
- For other options see `myproxy-init -help`

MyProxy: Server name to the client environment variable

- Default MyProxy server can with: `MYPROXY_SERVER`

bash:

```
export MYPROXY_SERVER=<set myproxy host here>
```

csh:

```
setenv MYPROXY_SERVER <set myproxy host here>
```

- or via command line parameter: `-s <myproxy host>`

MyProxy: Deletion and info: myproxy-destroy, myproxy-info

- To **remove** the proxy:

```
myproxy-destroy -l <username> -s <server>
```

- To view **status** of the proxy **at MyProxy server:**

```
myproxy-info -l <username> -s <server>
```

MyProxy: retrieve proxy certificate with myproxy-logon

- To retrieve the proxy:

```
myproxy-logon -l <username> -s <server>
```

  - `-t <lifetime>` of proxy in hours (by default 12 h)
    - This cannot be greater that what was set with `-t` in `myproxy-init`

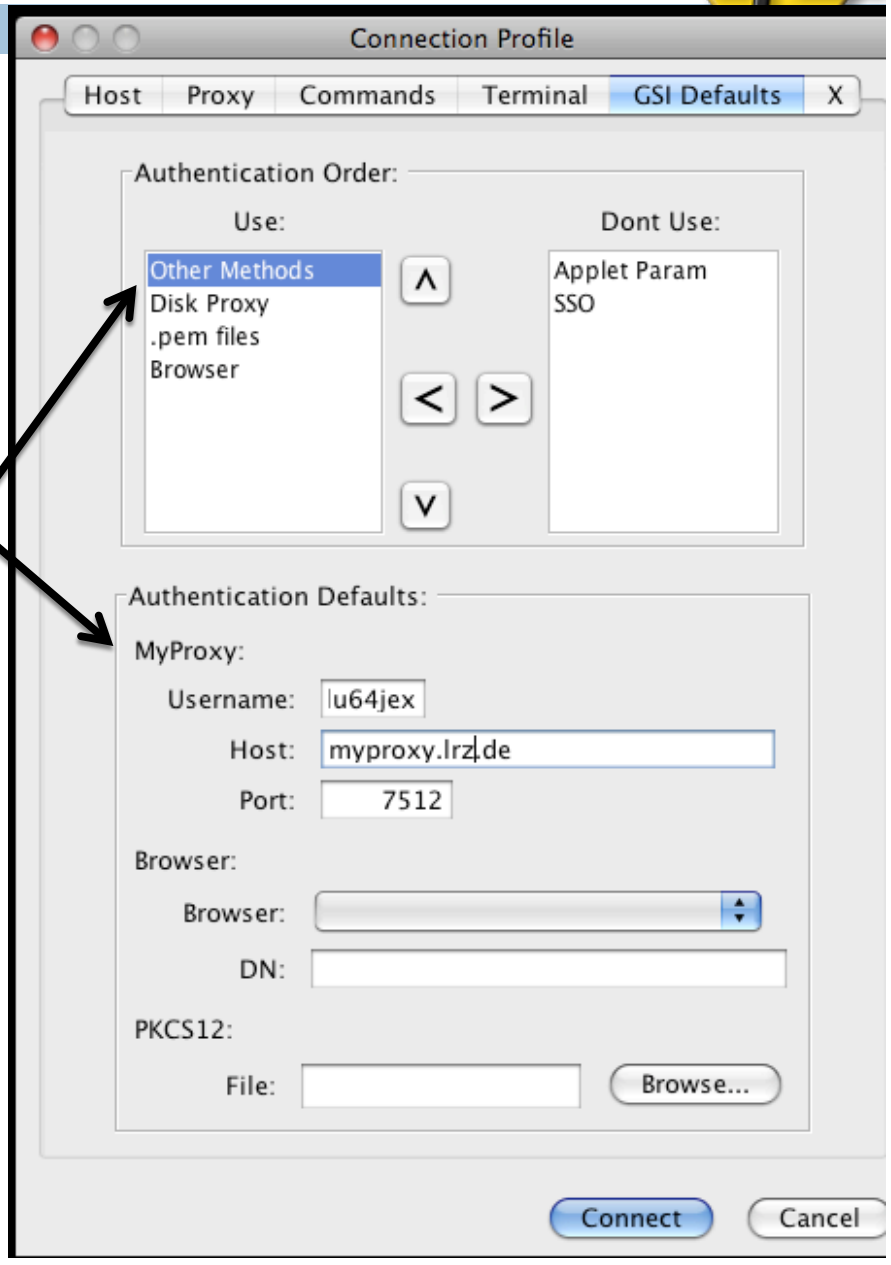
- To view your proxy status at the client machine:

```
grid-proxy-info
```

MyProxy: using the proxy with C Java based GSI-SSH TERM

- The Java based gsissh client can use MyProxy service.

- Click Advanced button in "Connect to host.." dialog.
- "Other Methods" has to be in the "Use:" list of GSI Defaults tab



MyProxy: Java WS client (1)

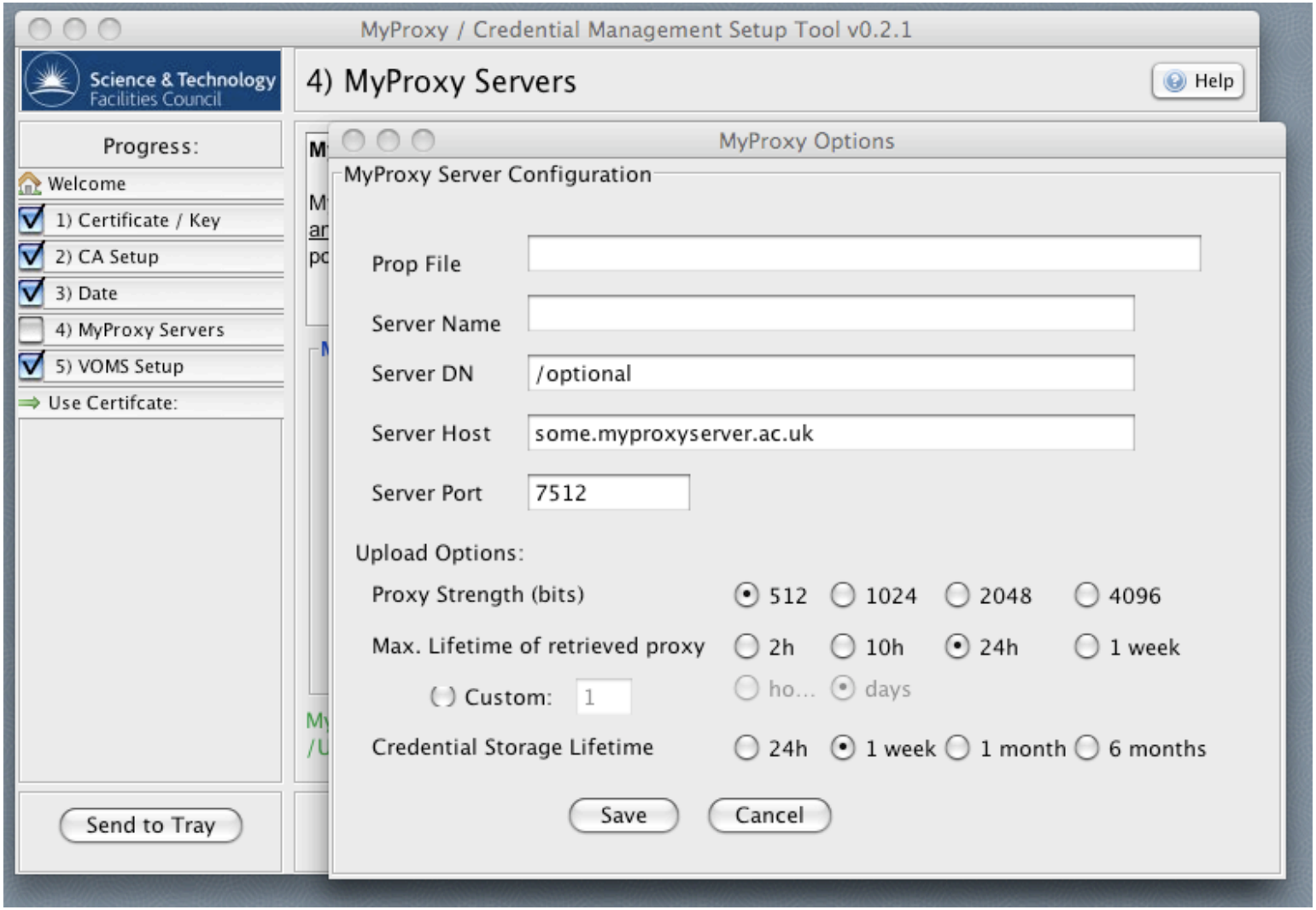
- http://www.ngs.ac.uk/tools/certwizard
- There you can find a Java Webstart based client "Certificate Management Wizard", which can upload a proxy without Globus installation.
- Also jar version is available (no browser then)
- It provides also proxy download feature
  - to use Java GSI-SSH Term is not needed

MyProxy: Java WS client (2) Configuration (1)

- Configure in "4) MyProxy Servers" your MyProxy hosts.
- There you can select the default one.
- To add a new MyProxy server
  - Server Name (description) and Server Host (address) are required.

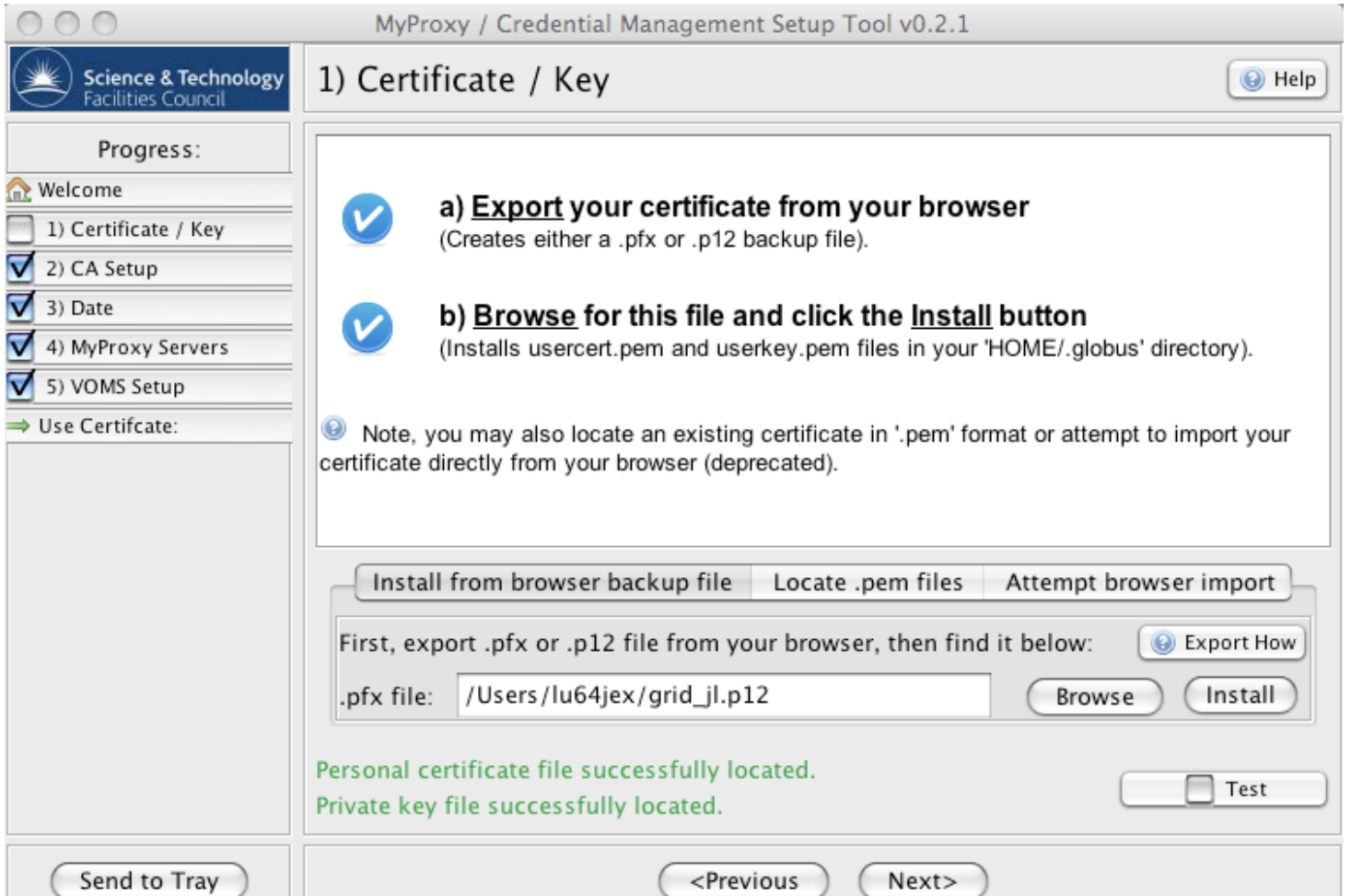
    (a screenshot is on the following slide)
  - Server DN you can leave as "`/optional`".
  - In that dialog you can set the lifetime values.

MyProxy: Java WS client (3) Configuration (2)



MyProxy: Java WS client (4) Configuration (3)

- After configuration of the MyProxy server, set your .pem or .p12 in "1) Certificate / Key" menu.



MyProxy: Java WS client (5) Configuration (4)

- Then click "Use Certificate" and in MyProxy tab press Upload.
- It asks the Passwords and the Username.