



Enabling Grids for E-scienceE

FTS Administration for Tier-2s

Gavin McCance (gavin.mccance@cern.ch)

Tier-2 Tutorial 15 June 2006

www.eu-egee.org
www.glite.org



- **What transfers will I see and from where?**
- **What control do I have over my channels?**
- **How do I set up the client software?**
- **What errors/tickets will I see and how do I handle them?**
- **What else do I need to check for my site?**

- **gLite File Transfer Service is a data movement fabric service**
 - It is a multi-VO service, used to balance usage of site resources according to VO and site policies
- **Why is it needed ?**
 - For the user, the service it provides is the reliable point to point movement of Storage URLs (SURLs)
 - For the site manager, it provides a reliable and manageable way of serving file movement requests from their VOs
 - For the VO manager, it provides ability to control requests coming from his users
 - Re-ordering, prioritization,...
 - **The focus is on the “service”**
 - **It should make it easy to do these things well**

- **For management ease, the service supports splitting jobs into multiple “channels”**
 - Submitted jobs are assigned to a suitable channel for serving
- **A channel may be:**
 - A point to point network link
 - (e.g. we manage all the T0 to T1 links in LCG on a separate channel)
 - Various “catch-all” channels
 - (e.g. everything else coming to me, or everything to one of my tier-2 sites)
 - More flexible channel definitions are on the way (but not there yet)
- **Channels are uni-directional**

- **“Channel”**: It’s not a great name
 - Isn’t tied to a physical network path
 - It’s just a management concept
 - “Queue” might be better 😊
- **All file transfer jobs on the same channel are served as part of the same queue**
 - Intra-VO priorities for the queue
 - Internal-VO priorities within a VO
- **Each channel has its own set of transfer parameters**
 - Number of concurrent files running, number streams, etc
- **The WLCG model assigns each FTS sever responsibilities of what transfers it is supposed to manage**
 - Channels allow you to split up the management this responsibility as you see fit (*)

- **What use-cases does FTS support and how do we deploy the servers to do this?**
- **Primary use-cases from experiment computing models**
 - tier-0 export to tier-1
 - tier-1 import into tier-0
 - tier-1 to tier-1 data transfer
 - tier-2 upload of data to associated tier-1: MC upload
 - tier-1 push of data to associated tier-2: AOD
- **Secondary use-cases**
 - non-associated tier-2 to tier-1: backup MC upload
 - non-associated tier-1 to tier-2: AOD?
 - tier-2 to tier-2: ?

- **Deploy only at tier-0 CERN and tier-1 sites**
 - Ease of operations – put the service where the support is (*)
 - Simplifying the client job of “who do I submit to?”
 - Does lead to some “odd” channel definitions and introduces some restrictions on who can control what
- **Which FTS servers are responsible for which transfers?**
 - There are a few basic rules
- **Described in:**
 - <https://twiki.cern.ch/twiki/pub/LCG/FtsServerInstall15/SC4FTSsetupplan.doc>
 - <https://twiki.cern.ch/twiki/bin/view/LCG/FtsServerDeployExampleTier1>

- 1. Tier-0:** If it involves CERN, then CERN's FTS does the transfer
 - This covers tier-0 to tier-1 and tier-1 to tier-0.
- 2. Tier-1 sites:** if you are the destination, your FTS is responsible for running the transfer
 - (if you're the source, the other end is responsible)
- 3. Tier-2 sites:** if you are the destination, your associated tier-1's FTS is responsible for running the transfer
 - Regardless of who is the source
- 4. We prioritise control of writing over control of reading**

- **Primary use-cases from experiment computing models**
 - tier-0 export to tier-1: CERN
 - tier-1 import into tier-0: CERN
 - tier-1 to tier-1 data transfer: the destination tier-1
 - tier-2 upload of data to associated tier-1: the associated tier-1
 - tier-1 push of data to associated tier-2: the associated tier-1
- **Secondary use-cases**
 - non-associated tier-2 to tier-1: the destination tier-1
 - non-associated tier-1 to tier-2: the tier-1 associated to the destination tier-2
 - tier-2 to tier-2: the tier-1 associated to the destination tier-2

- We prioritise control of writing over control of reading
- All **inbound transfers** are controlled by your associated tier-1 site
 - You can control this
- The majority of your **outbound transfer** will be upload to your associated tier-1 site
 - You can control this – if your T1 set up an explicit channel for you 😊
- You should expect some **outbound traffic** transferring to other tier-1 sites or to other tier-2 sites
 - These will be controlled by the other tier-1 sites
 - We offer no easy way to shut this off because of the way the channels are defined

- All inbound transfers are controlled by your associated tier-1 site
 - You can control this
- For each of its associated tier-2's, a tier-1 sets up a channel **STAR-TIER2**
 - This channel will match *any* source to your site
- Additionally, the associated tier-1 may wish to manage separately its transfers to the tier-2
 - This would be an explicit **TIER1-TIER2** channel
 - If this channel is not defined, traffic from your associated tier-1 will be matched on the general STAR-TIER2 channel

- **Upload to your tier-1. The tier-1 will define, either:**
 - **STAR-TIER1**. To match any traffic to the tier-1. In this case, you have limited control, since switching you off as a source switches off everyone else as well
 - Or... an explicit **TIER2-TIER1** channel. This allows you to control the reading from your site to your tier-1
- **Transfer to other tier-1s or tier-2s:**
 - This will be managed by the non-associated tier-1 server.
 - Although “possible”, it is not practical to expect that tier-1 to manage a distinct channel for upload from all potential tier-2 sites
 - So in practice, you have no control of these transfers



Enabling Grids for E-science

Setting up the clients

Setting up the client for admins

- **Site admins should use BDII to find the relevant FTS server (this will be your T1 site).**
 - Each is published with the GOCSDB site-name as the key.
- **export LCG_GFAL_INFOSYS=lcg-bdii.cern.ch:2170**
export GLITE_SD_PLUGIN=bdii
export GLITE_SD_SITE=GOCSITENAME
- **where GOCSITENAME is that of your tier-1**
- **Test (with verbose flag to tell you which endpoint it's using):**
 - **glite-transfer-channel-list -v**

- This should be done as part of the gLite 3.0 UI / WN install
- Generally it's the same as for admins:
 - LCG_GFAL_INFOSYS, GLITE_SD_PLUGIN
 - GLITE_SD_SITE for the server you want to submit to
- There is also a **glite-transfer-discovery** tool that will return the client the endpoint directly using the source and destination **SURLs** or sites
 - Not a perfect tool – still a bit too sensitive to bad information publication
 - GLITE_SD_SITE *should not* be set when using this tool (otherwise you restrict the query to just this site)

- **The initial channel setup at the tier-1**
- **Scheduled intervention on your SRM**
- **Unscheduled intervention on your SRM**
- **Changing the channel properties**

- Know who your tier-1 FTS service administrators are
- Know which channels have been setup for you
 - STAR-TIER2, TIER1-TIER2, TIER2-TIER1
- Make sure that that your local admin(s) have been set as channel managers for these channels

- Check that the source / destination for these channels is using the correct GOC-DB name for your site
 - glite-transfer-channel-list STAR-TIER2

- **If the channel commands don't work:**
 - “You are not authorised as admin for this channel”
- **Make sure you are pointing at the correct FTS server:**
 - `glite-transfer-channel-list -v`
- **Check your roles:**
 - `glite-transfer-getroles`
- **If you are a manager of a channel, you can check on the other managers:**
 - `glite-transfer-channel-listmanagers CHANNELNAME`
- **And you can add new people:**
 - `glite-transfer-channel-addmanagers CHANNELNAME
“/O=Grid/DN/of the new admin/”`

- In collaboration with your tier-1, decide what VO shares there should be on each channel and set them

- **Check the current shares:**

```
[root@fts102 CERN-RALcompleted]# glite-transfer-channel-list CERN-RAL
```

```
Channel: CERN-RAL
```

```
Between: CERN-PROD and RAL-LCG2
```

```
State: Active
```

```
Contact: lcg-support@gridpp.rl.ac.uk
```

```
Bandwidth: 0
```

```
Nominal throughput: 0
```

```
Number of files: 40, streams: 5
```

```
Number of VO shares: 5
```

```
VO 'dteam' share is: 10
```

```
VO 'alice' share is: 100
```

```
VO 'atlas' share is: 100
```

```
VO 'cms' share is: 100
```

```
VO 'lhcb' share is: 100
```

- **Change or add them:**

- `glite-transfer-channel-setvoshare CERN-RAL ops 100`

- The share is decided as:
- Find all the VOs with pending jobs. Total their shares. e.g.
- The share is calculated point-in-time w.r.t. current jobs. It does not use past data.
- **Example:**
 - LHCb and dteam have jobs
 - Atlas has no job
 - Total = $100 + 25 = 125$
 - Dteam gets $25/125 = 1/5$
 - LHCb get $100/125 = 4/5$

LHCb	100
Atlas	100
dteam	25

- **I'm fixing / upgrading / etc my SRM (scheduled intervention). How to stop all transfers:**
 - Know in advance which of your tier-1 channels affects you (and which you are able to control)
 - 10 to 15 minutes before the intervention:
 - Inform your tier-1 that you are pausing the channel(s)
 - `glite-transfer-channel-set -S Inactive CHANNELNAME`
- **After the intervention:**
 - Inform your tier-1 that you are restarting the channel(s)
 - `glite-transfer-channel-set -S Active CHANNELNAME`

- **My SRM has just gone down (or otherwise become unavailable). Unscheduled intervention:**
 - The same procedure as a scheduled intervention. If the SRM is down, set the channels that involve you directly to **Inactive**.
 - Do this **as soon as you can** after the problem with the SRM is noticed to minimize the Failed jobs
 - you may find that your tier-1 site has noticed the problem first and switched you off already
 - Inform the tier-1 site that you are doing it.
 - Set the channels back **Active** once the problem is fixed, informing the tier-1.

- **Changing the transfer rate on the channel:**
 - If you need to lower or increase the transfer rate to your site
 - Again, know in advance which of your tier-1 channels affects you (and which you are able to control). This is the same as those you can switch Active/Inactive
 - Check the current number of concurrent files:
 - `glite-transfer-channel-list CHANNELNAME`
 - Set the number of concurrent files, as you need:
 - `glite-transfer-channel-set -f 20 CHANNELNAME`
 - You can also change the number of streams for URLCOPY transfers
 - `glite-transfer-channel-set -T 5 CHANNELNAME`
 - Usually you need to experiment a bit to find the optimum setting



Enabling Grids for E-science

GGUS tickets you will see

- **Transfer service tickets are generally of three categories:**
- **Information service tickets**
 - Publishing bad SRM information or not enough information
- **Explicit FTS problems or user problems**
 - Various possible configuration problems, failures, submitting to the wrong endpoint, etc
- **SRM problems**
 - (“FTS says my job failed with ...”)

- Any problem through GGUS will be assigned first to your ROC
 - And they will then involve the tier-2 admins if necessary
- Tier-2's should expect to process **information service problems** at their site and **SRM problems** at their site
- **FTS server problems and general user problems** should generally be taken by the tier-1 running the FTS server (or their ROC).

- **Make sure your SRMs are published in information system**
 - FTS only runs transfers to production SRMs published in BDII and GOCDB
 - In particular:
 - Make sure the SRM is registered in GOC-DB
 - make sure there is a Glue 1.2 **GlueService** entry, as well as the normal GlueSE entry
 - make sure the the **GlueServiceForeignKey GlueSiteUniqueID** is set to the correct GOC-DB name for your site (ideally upper-case)
 - make sure the VOs that you support are published in both the **GlueSAPath** *and* in the **GlueService** entry
- **Please check for your site now...**
 - SFT on the way...

- **A variety of service problems may be reported on the FTS server itself**
 - most of which are not the responsibility of the tier-2
 - **BUT: Your tier-1 site should tell you if there is a problem on their FTS which will affect the rate to your site**
- **Client setup problems also happen from time to time**
- **If you receive mails / complaints indicating the following problems(...), please ask the user to submit a ticket to GGUS**
 - These should be handled by your ROC
 - Feel free the short-cut if it's clear what the problem is and you can fix it 😊

- **Cannot submit to web-service**
- **USER USING WRONG ENDPOINT:**
Failed to determine the interface version of the service: getInterfaceVersion: SOAP fault: "http://xml.apache.org/axis/":Server.NoService - The AXIS engine could not find a target service to invoke! targetService is FileTransfr
- **USER USING WRONG HOST:**
Failed to determine the interface version of the service: getInterfaceVersion: SOAP fault: SOAP-ENV:Client - CGI-gSOAP: Could not open connection ! (TCP get host by name failed in tcp_connect())
- **BAD GRID CREDENTIALS / CA CERTS / CRL LISTS ON CLIENT OR SERVER:**
Failed to determine the interface version of the service: getInterfaceVersion: SOAP fault: SOAP-ENV:Client - CGI-gSOAP: GSS Major Status: Authentication Failed
GSS Minor Status Error Chain:
(null)
- **USER NOT IN FTS SUBMIT MAPFILE (generated from VOMS):**
submit: You are not authorised to submit jobs to this service
- **If your users report this to you (by email, or at coffee), and the solution isn't obvious, please ask them to submit to GGUS, and the request should be handled by your ROC**

- **The transfer fails with error:**

"No site found for host srm.random.site"

"No Channel found or VO not authorized for transferring between TIER2SITE and TIER1SITE"

- **These are likely to be information service problems**

- SRM not published
- VO not published for a given SRM
- Site name incorrect in GlueService

- **..or FTS configuration problems:**

- VO not given a share on the channel
- Channel not defined for this transfer

- **Same again – ask users to submit to GGUS if the problem isn't obvious**

- Feel free to shortcut with your tier-1 admin if you know what's wrong

- **SRM problems: These will be reported (faithfully) by FTS if the retries all fail**
 - These are the most common tickets you will likely see

- **If the failure was on write, you will see an error with “SRM put”:**
 - Failed on SRM put: SRM getRequestStatus timed out on put" and "Failed on SRM put: Failed SRM put on httpg://srm.cern.ch:8443/srm/managerv1 ; id=770376972 call. Error is CastorStagerInterface.c:2457 Device or resource busy (errno=0, serrno=0)

- **If the failure was on read, you will see an error with “SRM get”:**
 - FINAL:SRM_SOURCE: Failed on SRM get: Failed SRM get on httpg://srm.cern.ch:8443/srm/managerv1 ; id=759760078 call. Error is specified file(s) does not exist%

- **Sometimes the failure is on the gridFTP transfer:**
 - FINAL:TRANSPORT: Transfer failed. ERROR the server sent an error response: 425 425 Cannot open port: java.lang.Exception: Pool manager error: No write pool available for <sc_tape:SC@osm>

- **The message should contain the error string that was returned by the SRM or gridFTP server to the FTS - and hopefully this is sufficient to start investigating the problem**

- **Otherwise...**
 - Detailed logging. If it's not obvious from the error message, the tier-1 FTS admin can give you the FTS transfer log to correlate with your SRM logs
- **The successful call pattern to your SRM is:**
 - SRM get on source with source SURL
 - SRM getStatus to get the source TURL
 - SRM put on destination with dest SURL
 - SRM getStatus to get the dest TURL
 - globus_gass_copy_url_to_url (VDT gridFTP client) using TURLs
 - SRM setStatus("Done") on source SRM
 - SRM setStatus("Done") on dest SRM

29164

GSIFTP

```
srm://srm.cern.ch/castor/cern.ch/grid/dteam/sc4/055/file-086.dat
srm://dcache.gridpp.rl.ac.uk/pnfs/gridpp.rl.ac.uk/data/dteam/sc3/2006-06-13/file-5485-cbeddd91-1227-4a86-849b-a198fb9cedec.dat
2006-06-13 13:09:33,027 [INFO ] - STATUS:START
2006-06-13 13:09:33,028 [WARN ] - Getting source from SRM: ; srm://srm.cern.ch/castor/cern.ch/grid/dteam/sc4/055/file-086.dat
2006-06-13 13:09:33,028 [DEBUG ] - contacting srm for get at ; http://srm.cern.ch:8443/srm/managerv1
2006-06-13 13:09:33,042 [DEBUG ] - Entered SrmUtil::constructor
2006-06-13 13:09:33,043 [INFO ] - STATUS:BEGIN:SRM_GET
2006-06-13 13:09:33,043 [DEBUG ] - Now calling requestTurlFromSurl ; srm://srm.cern.ch/castor/cern.ch/grid/dteam/sc4/055/file-086.dat
2006-06-13 13:09:33,043 [DEBUG ] - Entered SrmUtil::requestTurlFromSurl
2006-06-13 13:09:33,043 [DEBUG ] - Entered SrmUtil::get
2006-06-13 13:09:33,043 [DEBUG ] - Entered SrmUtil::checkService - srm ping
2006-06-13 13:09:33,326 [DEBUG ] - Exiting SrmUtil::checkService - srm__ping to 0.0.0.0
2006-06-13 13:09:34,083 [DEBUG ] - Exiting SrmUtil::get - srm__get to 0.0.0.0
2006-06-13 13:09:34,083 [DEBUG ] - Request Status Ready
2006-06-13 13:09:34,083 [INFO ] - SRM get request ID: 679085254
2006-06-13 13:09:34,083 [DEBUG ] - Exiting SrmUtil::prepareRequest
2006-06-13 13:09:34,083 [DEBUG ] - got file status state Ready
2006-06-13 13:09:34,083 [DEBUG ] - Exiting SrmUtil::requestTurlFromSurl - finishing
2006-06-13 13:09:34,083 [DEBUG ] - Entered SrmUtil::finishRequest
2006-06-13 13:09:34,083 [DEBUG ] - TURL is for SRM get gsiftp://lxsrc6208.cern.ch/castor/cern.ch/grid/dteam/sc4/055/file-086.dat
2006-06-13 13:09:34,083 [INFO ] - STATUS:END:SRM_GET
2006-06-13 13:09:34,083 [DEBUG ] - TURL = ; gsiftp://lxsrc6208.cern.ch/castor/cern.ch/grid/dteam/sc4/055/file-086.dat
2006-06-13 13:09:34,083 [DEBUG ] - Entered SrmUtil::setFileStatus
2006-06-13 13:09:34,083 [DEBUG ] - setting file status to Running
2006-06-13 13:09:34,355 [DEBUG ] - Exiting SrmUtil::srm__setFileStatus to 0.0.0.0
2006-06-13 13:09:34,355 [DEBUG ] - SIZE = 1048576000
2006-06-13 13:09:34,355 [WARN ] - Prepare put to SRM: ; srm://dcache.gridpp.rl.ac.uk/pnfs/gridpp.rl.ac.uk/data/dteam/sc3/2006-06-13/file-5485-cbeddd91-1227-4a86-849b-a198fb9cedec.dat
2006-06-13 13:09:34,355 [DEBUG ] - contacting srm for put at ; http://dcache.gridpp.rl.ac.uk:8443/srm/managerv1
2006-06-13 13:09:34,355 [DEBUG ] - Entered SrmUtil::constructor
2006-06-13 13:09:34,356 [INFO ] - STATUS:BEGIN:SRM_PUT
2006-06-13 13:09:34,356 [DEBUG ] - Now calling requestTurlFromSurl ; srm://dcache.gridpp.rl.ac.uk/pnfs/gridpp.rl.ac.uk/data/dteam/sc3/2006-06-13/file-5485-cbeddd91-1227-4a86-849b-a198fb9cedec.dat
2006-06-13 13:09:34,356 [DEBUG ] - Entered SrmUtil::requestTurlFromSurl
2006-06-13 13:09:34,356 [DEBUG ] - Entered SrmUtil::put
2006-06-13 13:09:34,356 [DEBUG ] - Entered SrmUtil::checkService - srm ping
2006-06-13 13:09:34,939 [DEBUG ] - Exiting SrmUtil::checkService - srm__ping to 0.0.0.0
2006-06-13 13:09:35,799 [DEBUG ] - Exiting SrmUtil::put - srm__put to 0.0.0.0
2006-06-13 13:09:35,799 [DEBUG ] - Request Status Pending
2006-06-13 13:09:35,799 [INFO ] - SRM put request ID: -2147439870
2006-06-13 13:09:35,799 [DEBUG ] - Exiting SrmUtil::prepareRequest
2006-06-13 13:09:35,799 [DEBUG ] - got file status state Pending
2006-06-13 13:09:35,799 [DEBUG ] - Exiting SrmUtil::requestTurlFromSurl - not finished yet
2006-06-13 13:09:35,799 [DEBUG ] - Entered SrmUtil::updateTurlFromSurlRequest
2006-06-13 13:09:35,799 [DEBUG ] - Entered SrmUtil::getRequestStatus
2006-06-13 13:09:36,533 [DEBUG ] - Exiting SrmUtil::srm__getRequestStatus to 0.0.0.0
2006-06-13 13:09:36,533 [DEBUG ] - Request Status Active
2006-06-13 13:09:36,533 [DEBUG ] - got file status state Ready
2006-06-13 13:09:36,533 [DEBUG ] - Exiting SrmUtil::updateTurlFromSurlRequest - finishing
2006-06-13 13:09:36,533 [DEBUG ] - Entered SrmUtil::finishRequest
2006-06-13 13:09:36,533 [INFO ] - SRM put on http://dcache.gridpp.rl.ac.uk:8443/srm/managerv1 ; id=-2147439870 call returned an error:
2006-06-13 13:09:36,533 [INFO ] - This error is ignored since we did receive a TURL:gsiftp://gftp0444.gridpp.rl.ac.uk:2811/pnfs/gridpp.rl.ac.uk/data/dteam/sc3/2006-06-13/file-5485-cbeddd91-1227-4a86-849b-a198fb9cedec.dat
2006-06-13 13:09:36,533 [DEBUG ] - TURL is for SRM put gsiftp://gftp0444.gridpp.rl.ac.uk:2811/pnfs/gridpp.rl.ac.uk/data/dteam/sc3/2006-06-13/file-5485-cbeddd91-1227-4a86-849b-a198fb9cedec.dat
2006-06-13 13:09:36,533 [INFO ] - STATUS:END:SRM_PUT
2006-06-13 13:09:36,533 [DEBUG ] - TURL = ; gsiftp://gftp0444.gridpp.rl.ac.uk:2811/pnfs/gridpp.rl.ac.uk/data/dteam/sc3/2006-06-13/file-5485-cbeddd91-1227-4a86-849b-a198fb9cedec.dat
2006-06-13 13:09:36,533 [DEBUG ] - Entered SrmUtil::setFileStatus
2006-06-13 13:09:36,533 [DEBUG ] - setting file status to Running
2006-06-13 13:09:37,246 [DEBUG ] - Exiting SrmUtil::srm__setFileStatus to 0.0.0.0
2006-06-13 13:09:37,246 [INFO ] - STATUS:BEGIN:TRANSFER
2006-06-13 13:09:37,246 [WARN ] - Starting gsiftp transfer
TURL source = gsiftp://lxsrc6208.cern.ch/castor/cern.ch/grid/dteam/sc4/055/file-086.dat
TURL dest = gsiftp://gftp0444.gridpp.rl.ac.uk:2811/pnfs/gridpp.rl.ac.uk/data/dteam/sc3/2006-06-13/file-5485-cbeddd91-1227-4a86-849b-a198fb9cedec.dat
2006-06-13 13:09:37,246 [DEBUG ] - Calling gsiftp getfilesize
FILE SIZE = 1048576000
2006-06-13 13:09:38,436 [DEBUG ] - Calling gsiftp transfer
2006-06-13 13:10:12,313 [INFO ] - Transferred 253165568 out of 1048576000 bytes. Inst: 8241.07 KB/s. Avg: 8241.07 KB/s
2006-06-13 13:10:42,372 [INFO ] - Transferred 662110208 out of 1048576000 bytes. Inst: 13312.00 KB/s. Avg: 10776.53 KB/s
2006-06-13 13:11:12,430 [INFO ] - Transferred 1019609088 out of 1048576000 bytes. Inst: 11598.67 KB/s. Avg: 11051.19 KB/s
2006-06-13 13:11:14,601 [INFO ] - GSIFTP: source: set up FTP mode. DCAU disabled. Streams = 5, TCP buffersize = 0
GSIFTP: dest: set up FTP mode. DCAU disabled. Streams = 5, TCP buffersize = 0
2006-06-13 13:11:17,440 [DEBUG ] - Gsiftp transfer done.
2006-06-13 13:11:18,956 [DEBUG ] - filesize at destination = 1048576000
2006-06-13 13:11:18,956 [INFO ] - STATUS:END:TRANSFER
2006-06-13 13:11:18,956 [WARN ] - Done with gsiftp transfer
```

Times are UTC

SRM request IDs are provided

All SRM calls and returned statuses are logged

gridFTP transfer markers are reported

- The unsuccessful call pattern is (for example):
 - SRM get on source with source SURL
 - SRM getStatus to get the source TURL
 - SRM put on destination with dest SURL
 - SRM getStatus to get the dest TURL
 - globus_gass_copy_url_to_url FAILS
 - SRM setStatus(“Done”) on source SRM
 - SRM setStatus(“Done”) on dest SRM
 - SRM advisoryDelete on dest SURL
- It always attempts to cleanup and delete the destination SURL upon failure (although when things are going wrong it doesn't always succeed).

- **You can control your inbound transfers and (most) outbound transfers**
- **The majority of your transfers should be run by your tier-1's FTS server**
- **The client tools should be setup using BDII**
- **Reviewed common service operations**
- **You should expect to be assigned tickets for:**
 - Information publishing problems
 - SRM problems for investigation
- **Bump other problems to your ROC via GGUS**
 - Encourage users to make use of GGUS to report problems!
- **Check you're publishing your SRMs into BDII!**