
CS 696

Introduction to Grid Computing:
Lecture #19

Mary Thomas

San Diego State

Tuesday, 01-May-2007

Notes/comments

- Look for working examples in
~/mthomas/cs696
- Most are working fine

TeraGrid GRAM-WS Services

Resource Type	Address	Reference Properties
GRAM	https://cu12.ncsa.uiuc.edu:8443/wsrp/services/ManagedJobFactoryService	ResourceID=Loadleveler
GRAM	https://cu12.ncsa.uiuc.edu:8443/wsrp/services/ManagedJobFactoryService	ResourceID=Fork
GRAM	https://cu12.ncsa.uiuc.edu:8443/wsrp/services/ManagedJobFactoryService	ResourceID=Multi
GRAM	https://ds001.sdsc.edu:8443/wsrp/services/ManagedJobFactoryService	ResourceID=Fork
GRAM	https://ds001.sdsc.edu:8443/wsrp/services/ManagedJobFactoryService	ResourceID=Loadleveler
GRAM	https://ds001.sdsc.edu:8443/wsrp/services/ManagedJobFactoryService	ResourceID=Multi
GRAM	https://fred002.psc.teragrid.org:8443/wsrp/services/ManagedJobFactoryService	ResourceID=PBS
GRAM	https://fred002.psc.teragrid.org:8443/wsrp/services/ManagedJobFactoryService	ResourceID=Multi
GRAM	https://fred002.psc.teragrid.org:8443/wsrp/services/ManagedJobFactoryService	ResourceID=Fork
GRAM	https://h6.bigred.teragrid.iu.edu:8443/wsrp/services/ManagedJobFactoryService	ResourceID=Fork
GRAM	https://h6.bigred.teragrid.iu.edu:8443/wsrp/services/ManagedJobFactoryService	ResourceID=Loadleveler
GRAM	https://h6.bigred.teragrid.iu.edu:8443/wsrp/services/ManagedJobFactoryService	ResourceID=Multi
GRAM	https://maverick.tacc.utexas.edu:8443/wsrp/services/ManagedJobFactoryService	ResourceID=SGE
GRAM	https://maverick.tacc.utexas.edu:8443/wsrp/services/ManagedJobFactoryService	ResourceID=Fork
GRAM	https://maverick.tacc.utexas.edu:8443/wsrp/services/ManagedJobFactoryService	ResourceID=Multi
GRAM	https://repo.uc.teragrid.org:8443/wsrp/services/ManagedJobFactoryService	ResourceID=Multi
GRAM	https://repo.uc.teragrid.org:8443/wsrp/services/ManagedJobFactoryService	ResourceID=Fork
GRAM	https://tg-gatekeeper.purdue.teragrid.org:8443/wsrp/services/ManagedJobFactoryService	ResourceID=PBS
GRAM	https://tg-gatekeeper.purdue.teragrid.org:8443/wsrp/services/ManagedJobFactoryService	ResourceID=Condor
GRAM	https://tg-gatekeeper.purdue.teragrid.org:8443/wsrp/services/ManagedJobFactoryService	ResourceID=Fork
GRAM	https://tg-gatekeeper.purdue.teragrid.org:8443/wsrp/services/ManagedJobFactoryService	ResourceID=Multi
GRAM	https://tg-login.tacc.teragrid.org:8443/wsrp/services/ManagedJobFactoryService	ResourceID=Fork
GRAM	https://tg-login.tacc.teragrid.org:8443/wsrp/services/ManagedJobFactoryService	ResourceID=Condor
GRAM	https://tg-login.tacc.teragrid.org:8443/wsrp/services/ManagedJobFactoryService	ResourceID=Multi
GRAM	https://tg-login.tacc.teragrid.org:8443/wsrp/services/ManagedJobFactoryService	ResourceID=LSF
GRAM	https://tg-login1.sdsc.teragrid.org:8443/wsrp/services/ManagedJobFactoryService	ResourceID=Fork
GRAM	https://tg-login1.sdsc.teragrid.org:8443/wsrp/services/ManagedJobFactoryService	ResourceID=Multi
GRAM	https://tg-login1.sdsc.teragrid.org:8443/wsrp/services/ManagedJobFactoryService	ResourceID=PBS
GRAM	https://tg-login3.ncsa.teragrid.org:8443/wsrp/services/ManagedJobFactoryService	ResourceID=Multi
GRAM	https://tg-login3.ncsa.teragrid.org:8443/wsrp/services/ManagedJobFactoryService	ResourceID=PBS
GRAM	https://tg-login3.ncsa.teragrid.org:8443/wsrp/services/ManagedJobFactoryService	ResourceID=Fork
GRAM	https://tg-mayor1.uc.teragrid.org:8443/wsrp/services/ManagedJobFactoryService	ResourceID=Multi
GRAM	https://tg-mayor1.uc.teragrid.org:8443/wsrp/services/ManagedJobFactoryService	ResourceID=Fork
GRAM	https://tund.ncsa.uiuc.edu:8443/wsrp/services/ManagedJobFactoryService	ResourceID=LSF
GRAM	https://tund.ncsa.uiuc.edu:8443/wsrp/services/ManagedJobFactoryService	ResourceID=Multi
GRAM	https://tund.ncsa.uiuc.edu:8443/wsrp/services/ManagedJobFactoryService	ResourceID=Fork

TeraGrid Reliable File Transfer Services

Resource	Address
RFT	https://cu12.ncsa.uiuc.edu:8443/wsrf/services/ReliableFileTransferFactoryService
RFT	https://ds001.sdsc.edu:8443/wsrf/services/ReliableFileTransferFactoryService
RFT	https://fred002.psc.teragrid.org:8443/wsrf/services/ReliableFileTransferFactoryService
RFT	https://h6.bigred.teragrid.iu.edu:8443/wsrf/services/ReliableFileTransferFactoryService
RFT	https://maverick.tacc.utexas.edu:8443/wsrf/services/ReliableFileTransferFactoryService
RFT	https://repo.uc.teragrid.org:8443/wsrf/services/ReliableFileTransferFactoryService
RFT	https://tg-gatekeeper.purdue.teragrid.org:8443/wsrf/services/ReliableFileTransferFactoryService
RFT	https://tg-login.tacc.teragrid.org:8443/wsrf/services/ReliableFileTransferFactoryService
RFT	https://tg-login1.sdsc.teragrid.org:8443/wsrf/services/ReliableFileTransferFactoryService
RFT	https://tg-login3.ncsa.teragrid.org:8443/wsrf/services/ReliableFileTransferFactoryService
RFT	https://tg-mayor1.uc.teragrid.org:8443/wsrf/services/ReliableFileTransferFactoryService
RFT	https://tund.ncsa.uiuc.edu:8443/wsrf/services/ReliableFileTransferFactoryService

TeraGrid WS-MDS Index Services

Resource Type	Address
WS-MDS Index	https://cu12.ncsa.uiuc.edu:8443/wsrp/services/DefaultIndexService
WS-MDS Index	https://ds001.sdsc.edu:8443/wsrp/services/DefaultIndexService
WS-MDS Index	https://fred002.psc.teragrid.org:8443/wsrp/services/DefaultIndexService
WS-MDS Index	https://h6.bigred.teragrid.iu.edu:8443/wsrp/services/DefaultIndexService
WS-MDS Index	https://maverick.tacc.utexas.edu:8443/wsrp/services/DefaultIndexService
WS-MDS Index	https://tg-gatekeeper.purdue.teragrid.org:8443/wsrp/services/DefaultIndexService
WS-MDS Index	https://tg-login.tacc.teragrid.org:8443/wsrp/services/DefaultIndexService
WS-MDS Index	https://tg-login1.sdsc.teragrid.org:8443/wsrp/services/DefaultIndexService
WS-MDS Index	https://tg-login3.ncsa.teragrid.org:8443/wsrp/services/DefaultIndexService
WS-MDS Index	https://tg-mayor1.uc.teragrid.org:8443/wsrp/services/DefaultIndexService
WS-MDS Index	https://tund.ncsa.uiuc.edu:8443/wsrp/services/DefaultIndexService

WS-GRAM: ManagedJobFactoryService Endpoint References

```
<ns6:MemberServiceEPR>  
  <ns21:Address>  
    https://tg-login1.sdsc.teragrid.org:8443/wsrf/services/ManagedJobFactoryService  
  </ns21:Address>  
  <ns22:ReferenceProperties>  
    <ns1:ResourceID>Fork</ns1:ResourceID>  
      OR: <ns1:ResourceID>PBS</ns1:ResourceID>  
      OR: <ns1:ResourceID>Multi</ns1:ResourceID>  
    </ns22:ReferenceProperties>  
  <ns23:ReferenceParameters/>  
</ns6:MemberServiceEPR>
```

Globus GRAM-WS: command line client

```
tg-login1 ac/thomasm> globusrun-ws -submit \  
-F https://tg-login3.ncsa.teragrid.org:8443/wsrp/services/ManagedJobFactoryService \  
-c /bin/touch touched-file  
Submitting job...Done.  
Job ID: uuid:0949586a-f752-11db-bf71-0007e9d81390  
Termination time: 05/01/2007 19:36 GMT  
Current job state: Active  
Current job state: CleanUp  
Current job state: Done  
Destroying job...Done.  
tg-login1 ac/thomasm>ls -al  
-rw-r--r--  1 thomasm mpk          0 2007-04-30 15:40 touched-file
```

<<TRYING TO RUN WITHOUT A VALID PROXY (rename proxy file)>>

```
/bin/touch ws-gram-file
```

```
Submitting job...Failed.
```

```
globusrun-ws: Error submitting job
```

```
globus_gsi_gssapi: Error with gss credential handle
```

```
globus_credential: Valid credentials could not be found in any of the possible locations  
specified by the credential search order.
```

```
Valid credentials could not be found in any of the possible locations specified by the  
credential search order.
```

GRAM-WS client: <single job schema>

```
tg-login1 thomasm/ws-gram> cat simple.job.sh
```

```
#!/bin/tcsh
echo "======"
echo "testing globusrun using input job description <xml> file"
set FAC="https://tg-login3.ncsa.teragrid.org:8443/wsrp/services/ManagedJobFactoryService"

#Tell globusrun-ws to read the job description from a file, using the -f argument:
set JOB="simple.job.xml"
echo "globusrun-ws -submit -F $FAC -f $JOB"
globusrun-ws -submit -F $FAC -f $JOB
```

```
tg-login1 thomasm/ws-gram> cat simple.xml
```

```
<job>
  <executable>/bin/echo</executable>
  <argument>CS696: Intro to Grid Computing</argument>
  <argument>Instructor: Mary Thomas</argument>
  <argument>Department: Computer Science</argument>
  <stdout>${GLOBUS_USER_HOME}/cs696/ws-gram/stdout</stdout>
  <stderr>${GLOBUS_USER_HOME}/cs696/ws-gram/stderr</stderr>
</job>
```

GRAM-WS client: <single job schema>

```
tg-login1 thomasm/ws-gram> ./simple.job.sh
testing globusrun using input job description <xml> file
globusrun-ws -submit
-F https://tg-login3.ncsa.teragrid.org:8443/wsrf/services/ManagedJobFactoryService
-f simple.xml
Submitting job...Done.
Job ID: uuid:18a29162-f768-11db-9ea9-0007e9d81390
Termination time: 05/01/2007 22:13 GMT
Current job state: Active
Current job state: CleanUp
Current job state: Done
Destroying job...Done.
tg-login1 thomasm/ws-gram> date
Mon Apr 30 17:14:08 CDT 2007
tg-login1 thomasm/ws-gram> ls
-rw-r-----  1 thomasm mpk      324 2007-04-30 17:12 simple.xml
-rwxr-xr-x   1 thomasm mpk      452 2007-04-30 16:25 simplest.sh
-rw-r--r--   1 thomasm mpk       0 2007-04-30 17:14 stderr
-rw-r--r--   1 thomasm mpk     84 2007-04-30 17:14 stdout
tg-login1 thomasm/ws-gram> cat stdout
CS696: Intro to Grid Computing Instructor: Mary Thomas Department: Computer Science
```

WS-Gram: Multiple Jobs

```
tg-login2 thomasm/ws-gram> cat multi.job.xml
<?xml version="1.0" encoding="UTF-8"?>
<multiJob xmlns:gram="http://www.globus.org/namespaces/2004/10/gram/job"
  xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/03/addressing">
  <factoryEndpoint>
    <wsa:Address>https://tg-login3.ncsa.teragrid.org:8443/wsrf/services/ManagedJobFactoryService</wsa:Address>
    <wsa:ReferenceProperties>
      <gram:ResourceID>Multi</gram:ResourceID>
    </wsa:ReferenceProperties>
  </factoryEndpoint>
  <directory>${GLOBUS_LOCATION}</directory>
  <count>1</count>

  <job>
    ...
  </job>

  <job>
    ...
  </job>

  <job>
    ...
  </job>

</multiJob>
```

WS-Gram: Multiple Jobs

```
<job>
  <factoryEndpoint>
    <wsa:Address>https://tg-login3.ncsa.teragrid.org:8443/wsrf/services/ManagedJobFactoryService</wsa:Address>
    <wsa:ReferenceProperties> <gram:ResourceID>Fork</gram:ResourceID> </wsa:ReferenceProperties>
  </factoryEndpoint>
  <executable>/bin/date</executable>
  <stdout>${GLOBUS_USER_HOME}/ws-gram/stdout.p1</stdout>
  <stderr>${GLOBUS_USER_HOME}/ws-gram/stderr.p1</stderr>
  <count>2</count> <!-- run this twice -->
</job>
<job>
  <factoryEndpoint>
    <wsa:Address>https://tg-login3.ncsa.teragrid.org:8443/wsrf/services/ManagedJobFactoryService</wsa:Address>
    <wsa:ReferenceProperties> <gram:ResourceID>Fork</gram:ResourceID> </wsa:ReferenceProperties>
  </factoryEndpoint>
  <executable>/bin/hostname</executable><argument>-f</argument>
  <stdout>${GLOBUS_USER_HOME}/ws-gram/stdout.p1</stdout>
  <stderr>${GLOBUS_USER_HOME}/ws-gram/stderr.p1</stderr>
  <count>3</count> <!-- run this three times and append to stdout.p1 -->
</job>
<job>
  <factoryEndpoint>
    <wsa:Address>https://tg-login3.ncsa.teragrid.org:8443/wsrf/services/ManagedJobFactoryService</wsa:Address>
    <wsa:ReferenceProperties> <gram:ResourceID>Fork</gram:ResourceID> </wsa:ReferenceProperties>
  </factoryEndpoint>
  <executable>/bin/echo</executable>
  <argument>Hello World!</argument>
  <stdout>${GLOBUS_USER_HOME}/ws-gram/stdout.p2</stdout>
  <stderr>${GLOBUS_USER_HOME}/ws-gram/stderr.p2</stderr>
  <count>6</count>
</job>
```

WS-GRAM: MultiJob Output

```
tg-login2 thomasm/ws-gram> cat stdout.p1
Mon Apr 30 18:17:15 CDT 2007
Mon Apr 30 18:17:15 CDT 2007
tg-login3.ncsa.teragrid.org
tg-login3.ncsa.teragrid.org
tg-login3.ncsa.teragrid.org
tg-login2 thomasm/ws-gram> cat stdout.p2
Hello World!
Hello World!
Hello World!
Hello World!
Hello World!
Hello World!

tg-login2 thomasm/ws-gram>
```

Submit to another gatekeeper

```
tg-login2 thomasm/ws-gram> hostname -f  
tg-login2.ncsa.teragrid.org
```

Where are the files?

```
tg-login2 thomasm/ws-gram> ./simple.job.tst  
=====  
testing globusrun using input job description <xml> file  
globusrun-ws -submit  
-F https://tg-login1.sdsc.teragrid.org:8443/wsrf/services/ManagedJobFactoryService  
-f simple.job.xml  
Submitting job...Done.  
Job ID: uuid:2f7ac074-f837-11db-ac2f-0007e9d81328  
Termination time: 05/02/2007 22:56 GMT  
Current job state: Active  
Current job state: CleanUp  
Current job state: Done  
Destroying job...Done.  
=====
```

```
tg-login2 thomasm/ws-gram> ls std*  
ls: No match.
```

```
tg-login2 thomasm/ws-gram> gsissh tg-login.sdsc.teragrid.org "ls -al ws-gram"  
total 12  
drwxr-sr-x  2 mthomas sdu233    4096 2007-04-30 15:38 .  
drwxr-sr-x 18 mthomas sdu233    4096 2007-05-01 15:58 ..  
-rw-r--r--  1 mthomas sdu233    168 2007-05-01 15:56 stderr  
-rw-r--r--  1 mthomas sdu233    168 2007-05-01 15:56 stdout
```

Running parallel jobs

Parallel jobs

- Can be run in ‘interactive’ mode but limited to a small number (say 4)
- Most are submitted to a queuing system
 - From command line, or using globusrun, globusrun-ws, condor
 - Use a shell script

Simple MPI Example: hello.f

```
program hello
  include 'mpif.h'
  integer n, myid, numprocs, i, rc

  call MPI_INIT( ierr )
  call MPI_COMM_RANK( MPI_COMM_WORLD, myid, ierr )
  call MPI_COMM_SIZE( MPI_COMM_WORLD, numprocs, ierr )
  print *, 'Process ', myid, ' of ', numprocs, ' is alive'

10  if ( myid .eq. 0 ) then
      write(6,98)
98   format(5x,'I am the parent node:',2i)
      else
          write(6,99)
99   format(5x,'I am a child node:',2i)
      endif

30  call MPI_FINALIZE(rc)
      stop
      end
```

MPI Example: ring-around.f

```
tg-login1 cs696/pbs_ex> cat ring-around.f
! ring-around.f
include 'mpif.h'
integer                :: i, j, ierr, nprocs, iam, outunit, nans
double precision :: dnans
character      :: char_id*8, outfile*21, teststring*20
call mpi_init(ierr)
call mpi_comm_size(mpi_comm_world, nprocs, ierr)
call mpi_comm_rank(mpi_comm_world, iam, ierr)
if( iam.eq.0 ) print *, ' Welcome to the MPI example. # CPUs: ', nprocs
write(char_id,'(i3)') iam
outfile = "out_"//adjustl(char_id)
outunit = 11
open(outunit, file=outfile,status='unknown')
print *, ' Node ', iam, ' outfile:', outfile
call mpi_barrier(mpi_comm_world,ierr)
nans = 1
call mpi_bcast(nans,1,mpi_integer,0,mpi_comm_world,ierr)
call dotted(nans,outunit)
call mpi_finalize(ierr)
close(11)
end program example
```

Ring-around.f

```
subroutine dotted(nans,outunit)
include 'mpif.h'
integer, intent(in)      :: outunit,nand,nprocs,ierr,l,iam
call mpi_comm_size(mpi_comm_world, nprocs, ierr)
call mpi_comm_rank(mpi_comm_world, iam, ierr)
! Send own ID to the next process - nans times
do i=1,nans
if( iam.lt.nprocs-1 ) then
  print *, ' Node ', iam, ' sending to node',iam+1
  write(outunit,*) ' Node ', iam, ' sending to node',iam+1
  call mpi_send(iam,1,mpi_integer,iam+1,99,mpi_comm_world,ierr)
elseif( iam.eq.nprocs-1 ) then
  print *, ' Node ', iam, ' sending to node 0'
  write(outunit,*) ' Node ', iam, ' sending to node 0'
  call mpi_send(iam,1,mpi_integer,0,99,mpi_comm_world,ierr)
endif
if( iam.gt.0 ) then
  call mpi_recv(heis,1,mpi_integer,iam-1,99, mpi_comm_world,status,ierr)
elseif( iam.eq.0 ) then
  call mpi_recv(heis,1,mpi_integer,nprocs-1,99, mpi_comm_world,status,ierr)
endif

!   if( modulo(i,100).eq.0 ) print *,!'
enddo
end subroutine dotted
```

PBS Batch Script Example

```
tg-login1 cs696/pbs_ex> cat run.pbs.pi3.sh
#!/bin/sh
#
#PBS -q dque
#PBS -N example
#PBS -l nodes=2:ppn=2
#PBS -l walltime=0:05:00
#PBS -o ring.out
#PBS -e ring.err
#
## Export all my environment variables to the job
#PBS -V
#
## Change to my working directory
cd /users/mthomas/cs696/pbs_ex

### Run my parallel job (the PBS shell knows PBS_NODEFILE)
mpirun -machinefile $PBS_NODEFILE -np 4 ./pi3
```

Running Job using Interactive Nodes

```
tg-login1 cs696/pbs_ex> qsub -l -V -l walltime=00:30:00 -l nodes=4:ppn=2  
./run.pbs.pi3.sh
```

```
Found valid account 'SDU233' for queue 'dque'
```

```
Found a default account in ACL 'sdsc_teragrid:mthomas:sdu233:tgroam'  
on SDSC TeraGrid system roaming nodes: all queues
```

```
Using default project SDU233
```

```
This job will be billed against the project SDU233
```

```
qsub: waiting for job 362040.dtf-mgmt1.sdsc.teragrid.org to start
```

Using a Queuing System: PBS

```
tg-login1 mthomas/mpirun_ex> qsub run.pbs.ring.sh
Found valid account 'SDU233' for queue 'dque'
  Found a default account in ACL 'sdsc_teragrid:mthomas:sdu233:tgroam'
  on SDSC TeraGrid system roaming nodes: all queues
  Using default project SDU233
This job will be billed against the project SDU233

362039.dtf-mgmt1.sdsc.teragrid.org
tg-login1 mthomas/mpirun_ex> qstat -a | grep mthomas
362039.dtf-mgmt1.sds mthomas dque  example  --  2 --  -- 00:05 Q  --
```

```
<<<<<JOB MIGHT TAKE A WHILE>>>>>
tg-login2 cs696/pbs_ex> qstat -a | wc
  1116  12248  98061
tg-login2 cs696/pbs_ex> qstat -a | grep " R " | wc
   143   1573  12584
tg-login2 cs696/pbs_ex> qstat -a | grep " Q " | wc
   897   9867  78936
```

Submitting job to PBS using WS-GRAM

```
tg-login2 cs696/ws-gram> cat pbs.job.xml
<job>
  <executable>/usr/local/pbs/ia64/bin/qsub</executable>
  <argument>${GLOBUS_USER_HOME}/cs696/pbs_ex/run.pbs.hello.sh
</argument>
  <stdout>${GLOBUS_USER_HOME}/cs696/ws-gram/pbs.stdout</stdout>
  <stderr>${GLOBUS_USER_HOME}/cs696/ws-gram/pbs.stderr</stderr>
</job>
```

```
tg-login2 cs696/ws-gram> cat pbs.job.sh
#!/bin/tcsh
echo "====="
echo "testing globusrun using input job description <xml> file"
set FAC="https://tg-
login3.ncsa.teragrid.org:8443/wsrp/services/ManagedJobFactoryService"
#Tell globusrun-ws to read the job description from a file, using the -f argument:
set JOB="pbs.job.xml"
echo "globusrun-ws -submit -F $FAC -f $JOB"
globusrun-ws -submit -F $FAC -f $JOB
```

results

```
<<<submitted job>>>>
```

```
tg-login2 cs696/ws-gram> cat pbs.stdout
```

```
990146.tg-master.ncsa.teragrid.org
```

```
tg-login2 cs696/ws-gram> cat pbs.stderr
```

```
tg-login2 cs696/ws-gram> qstat -a | grep 990146
```

```
990146.tg-master.ncs thomasm dque hello -- 2 1 -- 00:05 R --
```

```
tg-login2 cs696/ws-gram>
```

Submitting job to PBS using WS-GRAM

```
=====
testing globusrun using input job description <xml> file
globusrun-ws -submit
-F https://tg-login3.ncsa.teragrid.org:8443/wsrf/services/ManagedJobFactoryService
-f pbs.job.xml
Submitting job...Done.
Job ID: uuid:520fd9f4-f84f-11db-b419-0007e9d81328
Termination time: 05/03/2007 01:49 GMT
Current job state: Active
Current job state: CleanUp
Current job state: Done
Destroying job...Done.
```

Specifying file staging

- must add specific elements to the job description and delegate credentials appropriately (see Delegating credentials).
- The file transfer directives follow the RFT syntax, which allows only for third-party transfers.
 - Each transfer specify source & destination GridFTP URL (for remote files) or as file URLs for local files
 - For staging a file in, the source URL would be a GridFTP URL (for instance `gsiftp://job.submitting.host:2811/tmp/mySourceFile`)
 - This resolves to a source document accessible on the *file system of the job submission machine* (for instance `/tmp/mySourceFile`).
 - At run-time the Reliable File Transfer service used by the MEJS on the remote machine would reliably fetch the remote file using the GridFTP protocol and write it to the specified local file (for instance `file:///${GLOBUS_USER_HOME}/my_transferred_file`, which resolves to `~/my_transferred_file`).

```
<fileStageIn>  
  <transfer>  
    <sourceUrl>gsiftp://job.submitting.host:2811/tmp/mySourceFile</sourceUrl>  
    <destinationUrl>file:///${GLOBUS_USER_HOME}/my_transferred_file</destinationUrl>  
  </transfer>  
</fileStageIn>
```

File staging in/out

```
<<<< note: this is not working yet >>>>
<job>
  <executable>my_echo</executable>
  <directory>${GLOBUS_USER_HOME}</directory>
  <argument>Hello</argument>
  <argument>World!</argument>
  <stdout>${GLOBUS_USER_HOME}/stdout</stdout>
  <stderr>${GLOBUS_USER_HOME}/stderr</stderr>
  <fileStageIn>
    <transfer>
      <sourceUrl>gsiftp://gridftp-hg.ncsa.teragrid.org:2811/bin/echo</sourceUrl>
      <destinationUrl>file:///${GLOBUS_USER_HOME}/my_echo</destinationUrl>
    </transfer>
  </fileStageIn>
  <fileStageOut>
    <transfer>
      <sourceUrl>file:///${GLOBUS_USER_HOME}/stdout</sourceUrl>
      <destinationUrl>gsiftp://tg-gridftp.sdsc.teragrid.org:2811/tmp/stdout</destinationUrl>
    </transfer>
  </fileStageOut>
  <fileCleanUp>
    <deletion>
      <file>file:///${GLOBUS_USER_HOME}/my_echo</file>
    </deletion>
  </fileCleanUp>
</job>
```

Where are the schemas?

On the Teragrid:

Is `/usr/local/globus-4.0.1-r3/share/schema/gram/`

Also, look in `pyGridWare/share/schemas`