

Table of Contents

CREAM Test work plan	1
Unit tests.....	1
Deployment tests.....	1
Installation.....	1
Configuration.....	1
System tests.....	1
Basic functionality tests for legacy interface.....	1
Simple Submission Test.....	1
InputSandbox Test.....	2
Remote InputSandbox Test.....	2
InputSandboxBaseURI Test.....	2
OutputSandbox localhost and get-output Test.....	2
OutputSandboxDestURI Test.....	3
OutputSandboxBaseDestURI Test.....	3
OutputData test.....	3
Sandbox staging Test.....	4
Delegation tests.....	4
Environment test.....	4
Epilogue test.....	4
Prologue test.....	5
CPU Allocation Test 1.....	5
CPU Allocation Test 2.....	5
CPU Allocation Test 3.....	6
CPU Allocation Test 4.....	6
CPU Allocation Test 5.....	7
CPU Allocation Test 6.....	7
JobStatus test.....	8
List test.....	8
Cancel test.....	8
Manual cancel test.....	8
Suspend/Resume Test.....	8
Enable/disable job submission test.....	9
Purge test.....	9
Automatic job purger test.....	9
Proxy renewal test.....	10
Automatic proxy purger test.....	10
Authz test.....	10
Resource BDII test.....	10
Banning test.....	11
Admin manage job test.....	12
Service-info test.....	12
Limiter test.....	12
Bparser restart test.....	12
LB test.....	12
Basic functionality tests for EMI-ES interface.....	13
Delegation.....	13
Proxy renewal.....	13
Activity without data staging.....	14
Activity with failifExitCodeNotEqualTo.....	15
Activity with Input and Output: Client Data Push and Client Data Pull.....	15
Activity with Input and Output: Server Data Push and Server Data Pull.....	17
Activity with Input and Output: Server and Client Data Push, Server and Client Data Pull.....	18
Activity with Input and Output: Server and Client Data Push, Server and Client Data	

Table of Contents

CREAM Test work plan

Pull, and multiple Delegations.....	20
Activity with Input and Output with multiple Sources (one wrong, one correct).....	22
Activity with Input and Output with multiple Targets.....	23
Activity with Input and Output specifying a non existing delegationid.....	24
Proxy renewal of multiple delegations.....	25
Submission of multiple activities.....	27
Validation for unsupported capability.....	28
Pre and Post Executables.....	30
Activity with Environment.....	32
List.....	33
ActivityInfo.....	34
Pause and Resume.....	35
Cancel.....	36
Wipe.....	36
Regression tests.....	37
Performance and scalability tests.....	37
Submission and cancellation with polling.....	37
Submission with polling and sandbox transfer.....	37
Simple submission with polling, lease updated and proxy renewal.....	38
Standard compliance and conformance tests.....	38
Glue 1 compliance.....	38
Glue 2 compliance.....	38
Nagios probe.....	38
Robot based Testsuite.....	38
Old Testsuite.....	38
Installation and setup.....	38
Old Testsuite and ETICS.....	39
Releases and Changelog.....	39
Old Testsuite description.....	39
Proxy management.....	40
Delegation management.....	40
Isolation and interference tests.....	40
Hints for test tuning.....	40
Proposed tests.....	40
.....	42
.....	43
.....	44

CREAM Test work plan

Unit tests

TBD

Deployment tests

Installation

The following should be tested:

- Fresh installation of a CREAM-CE node, doing what is documented in the CREAM System administrator guide (Sections: "Installation of a CREAM CE node in no cluster mode" and "Installation of a CREAM CE node in cluster mode")
- Update from a previous version of the same EMI major release. This should be tested doing a simple yum update.
- Update from a previous EMI major release. TBD

The following installation and update scenarios should be tested:

- CREAM CE in no cluster mode
- CREAM CE in cluster mode with gLite-cluster deployed in the same CREAM-CE node

Configuration

The configuration via yaim should be tested.

The following scenarios should be tested (they are documented in CREAM System Administrator Guide):

- CREAM CE configured in no cluster mode
- CREAM configured in cluster mode with gLite-cluster deployed in the same CREAM-CE node
- CREAM configured in cluster mode with gLite-cluster deployed on another node
- CREAM configured using gJAF as authorization system
- CREAM configured using ARGUS as authorization system
- CREAM using the new BLAH blparser
- CREAM using the old BLAH blparser
- CREAM using Torque as batch system
- CREAM using LSF as batch system
- CREAM using GE as batch system

System tests

Basic functionality tests for legacy interface

Simple Submission Test

Submit a job which simply does `"/bin/uname -a"`.

Check the final job status which should be DONE-OK.

Status: Implemented in the Robot based testsuite

InputSandbox Test

Execute a bash shell script.

The script is stored on the UI.

The jdl attribute InputSandbox is used.

Firstly submit the jdl and then wait for final job state which should be DONE-OK.

Status: Implemented in the Robot based testsuite

Remote InputSandbox Test

Execute a bash shell script.

The script is stored in a gridftp server.

The jdl attribute InputSandbox is used.

First upload the executable script.

Then submit the jdl and check the the final state which should be DONE-OK.

Status: Implemented in the Robot based testsuite

InputSandboxBaseURI Test

Submit a job where the executable is a bash shell script.

The script is stored in a gridftp server.

The jdl attribute InputSandboxBaseURI is used.

First upload the executable script in the considered gridftp server.

Then submit the jdl.

Check the final state which should be DONE-OK.

Status: Implemented in the Robot based testsuite

OutputSandbox localhost and get-output Test

- Write a simple job which executes `#!/bin/uname =` and store std out and error streams on "gsiftp://localhost"
- Submit the jdl and wait for final job state which should be DONE-OK
- Download the produced files with `glite-ce-job-output`
- Get the output file list from the jdl file and compare the two file lists

Status: Implemented in the Robot based testsuite

OutputSandboxDestURI Test

- Write a simple job which executes `/bin/uname -a` and stores std out and error streams on a gridftp server. The jdl attribute `OutputSandboxDestURI` is used.
- Get output file list from jdl,base dest uri.
- `lcg-ls base dest uri` and files shouldn't exist
- Submit job and wait for its completion
- `lcg-ls base dest uri` and files should exist
- Delete files (cleanup)
- `lcg-ls base dest uri` and files shouldn't exist

Status: Implemented in the Robot based testsuite

OutputSandboxBaseDestURI Test

- Write a simple job which simply executes `/bin/uname -a .` and stores std out and error streams on a gridftp server. The jdl attribute `OutputSandboxBaseDestURI` is used.
- Get output file list from jdl base dest uri.
- `lcg-ls base dest uri` and files shouldn't exist
- Submit job and wait for its completion
- `lcg-ls base dest uri` and verify that files exist
- Delete files (cleanup)
- `lcg-ls base dest uri` and files shouldn't exist

Status: Implemented in the Robot based testsuite

OutputData test

Try OutputData JDL attributes considering all the possible combinations.

To do that, you can consider a JDL such as this one:

```
[
executable="/bin/sleep";
arguments="25";
Stdoutput = "env.out" ;
StdError = "env.err" ;
InputSandbox = {
"/etc/passwd",
"ssh1.sh",
"gsiftp://cream-18.pd.infn.it:2811/etc/hostname.txt",
"file:///etc/fstab"
};
outputsandbox={"out.out", "passwd", "ssh1.sh", "hostname.txt", "fstab"};
OutputSandboxBaseDestURI="gsiftp://localhost";
OutputData={
[
OutputFile="passwd";
LogicalFileName = "lfn:/sgaravat/passwd";
],
[
OutputFile="hostname.txt";
],
[
OutputFile="ssh1.sh";
StorageElement="prod-se-01.pd.infn.it";
],
[
OutputFile="fstab";
LogicalFileName="lfn:/sgaravat/fstab";
]
```

```
StorageElement="prod-se-01.pd.infn.it";  
]  
};  
]
```

Once the job is in DONE-OK, retrieve the OSB files using the `glite-ce-job-output` command. Check the result of the data upload operations in the file `DSUpload_.out`

Implemented in the Robot based testsuite

Sandbox staging Test

This test case tests the two different Sandbox staging operations available for the `/etc/glite-ce-cream/cream-config.xml`, LRMS and GSIFTP.

- Change the setting (if GSIFTP, set to LRMS, if LRMS, set to GSIFTP)
- Run the "OutputSandbox localhost and get-output" Test
- Reset the setting to its original and rerun the "OutputSandbox localhost and get-output" Test

Status: Implemented in the Robot based testsuite

Delegation tests

- Delegate a proxy with `glite-ce-delegaet-proxy` and submit a job referring to the previously delegate proxy. Job final state should be DONE-OK
- Try to delegate a a proxy using an already used delegationid. This should fail
- Try to submit a job referring to a non-existing delegationid. This should fail

Status: Implemented in the Robot based testsuite

Environment test

Consider a job where the `jdl` sets an environmental variable and the executable is a script which prints the value of that variable. Submit the `jdl`.

Check the the final state which should be DONE-OK.

Check the output file which should have the designated variable.

Status: Implemented in the Robot based testsuite

Epilogue test

Execute two bash shell scripts, one for the job and the other as an epilogue.

The epilogue script creates a file with a certain string, which is collected by the job as an output file

Submit the `jdl`.

Check the final state which should be DONE-OK.

Check the output file which should have the designated string.

Status: Implemented in the Robot based testsuite

Prologue test

Execute two bash shell scripts, one for the job and the other as a prologue.

The prologue script creates a file with a certain string, which is collected by the job as an output file

Submit the jdl.

Check the final state which should be DONE-OK.

Check the output file which should have the designated string.

Status: Implemented in the Robot based testsuite

CPU Allocation Test 1

Submit a job where in the JDL there is:

```
WholeNodes=true;  
SMPGranularity=G;  
Hostnumber=H;
```

with $H > 1$.

Verify that in the submission script there is

- for LSF:

```
BSUB -n S*H  
BSUB -R "span[ptile=S]  
BSUB -x
```

- for PBS:

```
PBS -l nodes=H:ppn=S  
PBS -W x=NACCESSPOLICY:SINGLEJOB
```

with S equal to the value published as `GlueHostArchitectureSMPSize`.

See:

http://wiki.italiangrid.it/twiki/bin/view/CREAM/TroubleshootingGuide#5_1_Saving_the_batch_job_submiss

to understand how to save and check the submission scripts.

Status: Implemented in the Robot based testsuite

CPU Allocation Test 2

Submit a job where in the JDL there is:

```
WholeNodes=true;  
SMPGranularity=G;
```

Verify that in the submission script there is

- for LSF:

```
BSUB -n S
BSUB -R "span[hosts=1]"
BSUB -x
```

- for PBS:

```
PBS -l nodes=1:ppn=S
PBS -W x=NACCESSPOLICY:SINGLEJOB
```

with S equal to the value published as GlueHostArchitectureSMPSize.

Status: Implemented in the Robot based testsuite

CPU Allocation Test 3

Submit a job where in the JDL there is:

```
WholeNodes=true;
HostNumber=H;
```

with $H > 1$.

Verify that in the submission script there is

- for LSF:

```
BSUB -n S*H
BSUB -R "span[ptile=S]"
BSUB -x
```

- for PBS:

```
PBS -l nodes=H:ppn=S
PBS -W x=NACCESSPOLICY:SINGLEJOB
```

with S equal to the value published as GlueHostArchitectureSMPSize.

Status: Implemented in the Robot based testsuite

CPU Allocation Test 4

Submit a job where in the JDL there is:

```
WholeNodes=false;
SMPGranularity=G;
CPUNumber=C;
```

Verify that in the submission script there is

- for LSF:

```
BSUB -n C
BSUB -R "span[ptile=G]"
```

- for PBS:


```
PBS -l nodes=N:ppn=G { [+1:ppn=R] if r>0 }
```

with:

```
N = C / G  
R = C % G
```

Status: Implemented in the Robot based testsuite

CPU Allocation Test 5

Submit a job where in the JDL there is:

```
WholeNodes=false;  
HostNumber=H;  
CPUNumber=C;
```

with $H > 1$.

Verify that in the submission script there is

- for LSF:

```
BSUB -n C  
BSUB -R "span[ptile={ N if R=0 ; N+1 if R>0 }]"
```

- for PBS:

```
PBS -l nodes=H-R:ppn=N { [+R:ppn=N+1] if R>0 }
```

with:

```
N = C / H  
R = C % H
```

Status: Implemented in the Robot based testsuite

CPU Allocation Test 6

Submit a job where in the JDL there is:

```
WholeNodes=false;  
CPUNumber=C;
```

Verify that in the submission script there is

- for LSF:

```
BSUB -n C
```

- for PBS:

```
PBS -l nodes=C
```

Status: Implemented in the Robot based testsuite

JobStatus test

Verify the proper functionality of the `glite-ce-job-status` command with different level of verbosity and different options.

Verify that with `-L 0` it is reported the "Status".

Verify that with `-L 1` it is reported the "Status", the "Job status changes" and the "Issued commands".

Verify that with `-L 1` it is reported the "Status", the "Job status changes", the "Issued commands", the "Worker Node", the "Deleg Proxy ID", the "DelegProxyInfo", "Local User", "JDL".

Verify the status of 2 jobs, with ids specified in the command line and with ids read from a file (option `-i`).

Verify the proper functionality of the `--from` and `-to` option. Specify a time range where a single job was submitted, and verify that only the status of this job is returned.

Status: Implemented in the Robot based testsuite

List test

Perform a `glite-ce-job-list`.

Perform a job submission

Perform again a `glite-ce-job-list`

The returned jobid shouldn't exist in the in the job list command output before the submission and should exist afterwards.

Status: Implemented in the Robot based testsuite

Cancel test

- Submit a job
- Cancel it using the `glite-ce-job-cancel` command
- Check job status: it should eventually be `CANCELLED`
- Try to cancel again the aforementioned job. A failure should be returned

Status: Implemented in the Robot based testsuite

Manual cancel test

- Submit a job and then manually `qdel/bkill` it on the CE host
- Check the final status which should be either `"DONE-FAILED"` or `"CANCELLED"`

Status: Implemented in the Robot based testsuite

Suspend/Resume Test

Submit a job.

Wait until it's state is `REALLY-RUNNING` and then suspend it.

Case LSF batch system: It's state should be HELD. Then resume the same job. It's status should be REALLY-RUNNING. The final status should be DONE-OK.

Case Torque batch system: It's state should still be REALLY-RUNNING. Resume the job and expect an error. It's status should be REALLY-RUNNING. The final status should be DONE-OK.

Status: Implemented in the Robot based testsuite

Enable/disable job submission test

This test case tests the proper functionality of the `glite-ce-allowed-submission` and `glite-ce-enable/disable-submission` commands, for admin and non-admin users. The test executes the following steps:

- Check with `glite-ce-allowed-submission` if submissions are enabled (they should)
- Executes a simple job (it should succeed)
- Try to Disable submission using `glite-ce-disable-submission` command considering that the current user is not a CREAM admin. This should fail
- Add the current user as a CREAM admin
- Try to disable submission using `glite-ce-disable-submission` command. This time it should succeed
- Check with `glite-ce-allowed-submission` if submission is allowed (it shouldn't)
- Try a simple job submission (it should fail)
- Enable submission with `glite-ce-enable-submission` command
- Try the submission of a simple job (it should succeed)
- Remove current user from CREAM admin
- Try to disable submission using `glite-ce-disable-submission=` command (should fail, since the user isn't a CREAM admin anymore).

Status: Implemented in the Robot based testsuite

Purge test

- Purges a job
- Then searches for its status and its sandbox dirs on the cream node. The job shouldn't be found by the status command and the sandbox dirs shouldn't exist.

Status: Implemented in the Robot based testsuite

Automatic job purger test

Modify the `JOB_PURGE_POLICY` attribute in the configuration file `/etc/glite-ce-cream/cream-config.xml` setting 0 days for CANCELLED. Then restart tomcat.

Submit a long job and then cancel it. Verify that the job status is cancelled.

After 24 hours recheck the status of the job with `glite-ce-job-status`. The job should not be found. Verify in the CREAM CE in the CREAM sandbox dir that a directory for this job doesn't exist anymore.

Status: Implemented in the Robot based testsuite

Proxy renewal test

Verify the functionality of the `glite-ce-proxy-renew` command.

Submit a job lasting 20 minutes using a proxy valid 10 minutes.

After 10 minutes create on the UI a long proxy, and renew the proxy with the `glite-ce-proxy-renew` command.

Verify the final job status which should eventually be `DONE-OK`

Status: Implemented in the Robot based testsuite

Automatic proxy purger test

Delegate a proxy valid 1 hour

Submit a job using that delegatioid. This should work.

Try again the submission using the same delegationid after 24 hours. This should not work (the error message should say that the delegation was not found)

Status: Implemented in the Robot based testsuite

Authz test

Try functionality of CREAM with both Argus based authorizations and gJAF based authorizations.

To do that simply perform the "Simple Submission Test", "Delegation test", "List test" and "Cancel test" first using a CREAM configured using gJAF, and then using a CREAM configured using ARGUS.

Resource BDII test

Check if the resource `bdii` publishes all the relevant objectclasses for both `glue1` and `glue2`

- Check if the resource `BDII` publishes glue 1 `GlueCE` objectclasses. There should be one `GlueCE` objectclass per each queue:

```
ldapsearch -h <CREAM CE hostname> -x -p 2170 -b "o=grid" objectclass=GlueCE
```

- Check if the resource `BDII` publishes glue 1 `GlueVOView` objectclasses. There should be one `GlueVOView` objectclass per each queue-VO:

```
ldapsearch -h <CREAM CE hostname> -x -p 2170 -b "o=grid" objectclass=GlueVOView
```

- Check if the resource `BDII` publishes glue 1 `GlueCluster` objectclasses. If the CE is configured in no cluster mode there should be one `GlueCluster` objectclass. If the CE is configured in cluster mode and the `gLite-CLUSTER` is deployed on a different node, there shouldn't be any `GlueCluster` objectclass.

```
ldapsearch -h <CREAM CE hostname> -x -p 2170 -b "o=grid" objectclass=GlueCluster
```

- Check if the resource `BDII` publishes glue 1 `GlueSubCluster` objectclasses. If the CE is configured in no cluster mode there should be one `GlueSubCluster` objectclass. If the CE is configured in cluster mode and the `gLite-CLUSTER` is deployed on a different node, there shouldn't be any `GlueSubCluster` objectclass.

CreamTestWorkPlan < CREAM < TWiki

```
ldapsearch -h <CREAM CE hostname> -x -p 2170 -b "o=grid" objectclass=GlueSubCluster
```

- Check if the resource BDII publishes glue 2 GLUE2ComputingService objectclasses. There should be one GLUE2ComputingService objectclass:

```
ldapsearch -h <CREAM CE hostname> -x -p 2170 -b "o=glue" objectclass=GLUE2ComputingService
```

- Check if the resource BDII publishes glue 2 GLUE2ComputingEndPoint objectclasses with GLUE2EndpointInterfaceName equal to org.glite.ce.CREAM . There should be one of such GLUE2ComputingService objectclasses:

```
ldapsearch -h <CREAM CE hostname> -x -p 2170 -b "o=glue" "(&(objectclass=GLUE2ComputingEndPoint))"
```

- Check if the resource BDII publishes glue 2 GLUE2Manager objectclasses. If the CE is configured in no cluster mode there should be one GLUE2Manager objectclass. If the CE is configured in cluster mode and the gLite-CLUSTER is deployed on a different node, there shouldn't be any GLUE2Manager objectclass.

```
ldapsearch -h <CREAM CE hostname> -x -p 2170 -b "o=glue" objectclass=GLUE2Manager
```

- Check if the resource BDII publishes glue 2 GLUE2Share objectclasses. If the CE is configured in no cluster mode there should be one GLUE2Share objectclass per each VOview. If the CE is configured in cluster mode and the gLite-CLUSTER is deployed on a different node, there shouldn't be any GLUE2Share objectclass.

```
ldapsearch -h <CREAM CE hostname> -x -p 2170 -b "o=glue" objectclass=GLUE2Share
```

- Check if the resource BDII publishes glue 2 GLUE2ExecutionEnvironment objectclasses. If the CE is configured in no cluster mode there should be one GLUE2ExecutionEnvironment objectclass. If the CE is configured in cluster mode and the gLite-CLUSTER is deployed on a different node, there shouldn't be any GLUE2ExecutionEnvironment objectclass.

```
ldapsearch -h <CREAM CE hostname> -x -p 2170 -b "o=glue" objectclass=GLUE2ExecutionEnvironment
```

- Check if the resource BDII publishes glue 2 GLUE2ComputingEndPoint objectclasses with GLUE2EndpointInterfaceName equal to org.glite.ce.ApplicationPublisher. If the CE is configured in no cluster mode there should be one of such objectclass. If the CE is configured in cluster mode and the gLite-CLUSTER is deployed on a different node, there shouldn't be any of such objectclasses.

```
ldapsearch -h <CREAM CE hostname> -x -p 2170 -b "o=glue" "(&(objectclass=GLUE2ComputingEndPoint))"
```

Status: Not Implemented

Banning test

For a CREAM-CE configured using gJAF:

- Add the user's DN to /etc/lcas/ban_users.db on the cream host
- Try to submit job: you should expect failure
- Remove DN from the aforementioned file
- Submit another job: you should expect a successful submission

The test should not be done using a CREAM-CE configured using ARGUS

Status: Implemented in the Robot based testsuite

Admin manage job test

Check that a user who is not admin can't check the status of jobs submitted by another user.

Check that a user who is admin can check the status of jobs submitted by another user.

Status: Implemented in the Robot based testsuite

Service-info test

Verify the output of glite-ce-service-info command for both verbosity levels.

Status: Implemented in the Robot based testsuite

Limiter test

Set a limiter threshold to a value which will trigger it.

Wait for the limiter script to be run automatically.

Submit and job and expect failure.

Reset the limiter to the original values.

Wait for the limiter script to be run automatically.

Submit a command and expect success

Status: Implemented in the Robot based testsuite

Blparser restart test

Configure a CREAM CE using the old blah blparser.

Stop the blparser and after a few seconds (e.g. 10) restart it.

Wait for 2 minute and try a job submission. The should run and complete properly.

Status: Not Implemented in the Robot based testsuite

LB test

Submit a job to CREAM through the gLite-WMS and wait till the job is Done status

Check the logging-info output of this job: among the logged events there should be 3 events logged as LRMS as source:

- Running
- Really-Running
- Done

Status: Not Implemented in the Robot based testsuite

Basic functionality tests for EMI-ES interface

Delegation

Delegate a proxy with the `glite-es-delegate-proxy` command.

It should return a delegationid.

E.g.:

```
glite-es-delegate-proxy -e cream-38.pd.infn.it
DelegationID = 5655008298193933
```

Check with the `glite-es-delegation-info` command about this delegationid. In particular check the `lifetime` field which should match the expiration time of the proxy in the UI:

```
$ glite-es-delegation-info -e cream-38.pd.infn.it 5655008298193933
Lifetime = Tue Feb 28 21:14:46 2012
Issuer   = CN=proxy,CN=proxy,CN=Massimo Sgaravatto,L=Padova,OU=Personal Certificate,O=INFN,C=IT
Subject  = CN=Massimo Sgaravatto,L=Padova,OU=Personal Certificate,O=INFN,C=IT
```

Status: Not Implemented

Proxy renewal

Create a 1 hour proxy and delegate it:

```
[sgaravat@cream-50 ADLEExamples]$ voms-proxy-init -voms dteam -valid 01:00
Enter GRID pass phrase:
Your identity: /C=IT/O=INFN/OU=Personal Certificate/L=Padova/CN=Massimo Sgaravatto
Creating temporary proxy ..... Done
Contacting voms2.hellasgrid.gr:15004 [/C=GR/O=HellasGrid/OU=hellasgrid.gr/CN=voms2.hellasgrid.gr]
Creating proxy ..... Done
```

Your proxy is valid until Sun Mar 18 11:59:28 2012

```
[sgaravat@cream-50 ADLEExamples]$ glite-es-delegate-proxy -e cream-47.pd.infn.it
DelegationID = 6694176412145996
```

Verify with `glite-es-delegation-info` that the proxy lasts 1 hour:

```
[sgaravat@cream-50 ADLEExamples]$ glite-es-delegation-info -e cream-47.pd.infn.it 6694176412145996
Lifetime = Sun Mar 18 11:58:35 2012
Issuer   = CN=proxy,CN=proxy,CN=Massimo Sgaravatto,L=Padova,OU=Personal Certificate,O=INFN,C=IT
Subject  = CN=Massimo Sgaravatto,L=Padova,OU=Personal Certificate,O=INFN,C=IT
```

Create a 2 hours proxy and renew the previously delegated proxy:

```
[sgaravat@cream-50 ADLEExamples]$ voms-proxy-init -voms dteam -valid 02:00
Enter GRID pass phrase:
Your identity: /C=IT/O=INFN/OU=Personal Certificate/L=Padova/CN=Massimo Sgaravatto
Creating temporary proxy ..... Done
Contacting voms2.hellasgrid.gr:15004 [/C=GR/O=HellasGrid/OU=hellasgrid.gr/CN=voms2.hellasgrid.gr]
Creating proxy ..... Done
```

Your proxy is valid until Sun Mar 18 13:00:27 2012

```
[sgaravat@cream-50 ADLEExamples]$ glite-es-delegation-renew -e cream-47.pd.infn.it 6694176412145996
Delegation with identifier [6694176412145996] successfully renewed
```

Verify with `glite-es-delegation-info` that the proxy lasts 2 hours:

```
[sgaravat@cream-50 ADLEexamples]$ glite-es-delegation-info -e cream-47.pd.infn.it 669417641214599
Lifetime = Sun Mar 18 12:59:36 2012
Issuer   = CN=proxy,CN=proxy,CN=Massimo Sgaravatto,L=Padova,OU=Personal Certificate,O=INFN,C=IT
Subject  = CN=Massimo Sgaravatto,L=Padova,OU=Personal Certificate,O=INFN,C=IT
```

Status: Not Implemented

Activity without data staging

Submit an activity where in the Application part there is just the Executable without `failifExitCodeNotEqualTo`.

```
$ cat activity_sleep60.adl
<ActivityDescription>
  <ActivityIdentification>
    <Name>sleep60</Name>
    <Description>sleep 60</Description>
    <Type>single</Type>
  </ActivityIdentification>
  <Application>
    <Executable>
      <Path>/bin/sleep</Path>
      <Argument>60</Argument>
    </Executable>
  </Application>
  <Resources>
    <QueueName>creamtest2</QueueName>
  </Resources>
</ActivityDescription>
```

```
$ glite-es-activity-create -e cream-38.pd.infn.it activity_sleep60.adl
```

```
$ glite-es-activity-create -e cream-47.pd.infn.it activity_sleep60.adl
*****
ActivityID      = CR_ES156078531
ActivityMgrURI  = https://cream-47.pd.infn.it:8443/ce-cream-es/services/ActivityManagementService
Status         = PREPROCESSING
Status Attrs   = {}
Timestamp      = Mon Mar 12 12:16:33 2012
Description    =
ETNSC         =
STAGEIN Dir    = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
SESSION Dir    = {}
STAGEOUT Dir   = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
```

Eventually check with `glite-es-activity-status` the state which should be `TERMINAL` without attributes:

```
$ glite-es-activity-status -e cream-47.pd.infn.it CR_ES156078531
*****
JobID          = CR_ES156078531
Status         = TERMINAL
Attributes     = {}
Timestamp      = Mon Mar 12 12:17:42 2012
Description    = reason=0
```

Status: Not Implemented

Activity with failIfExitCodeNotEqualTo

Submit an activity where in the Application part there is just the Executable with failIfExitCodeNotEqualTo.

```
$ cat activity_sleep60_failif.adl
<ActivityDescription>
  <ActivityIdentification>
    <Name>sleep60</Name>
    <Description>sleep 60</Description>
    <Type>single</Type>
  </ActivityIdentification>
  <Application>
    <Executable>
      <Path>/bin/sleep</Path>
      <Argument>60</Argument>
      <FailIfExitCodeNotEqualTo>1</FailIfExitCodeNotEqualTo>
    </Executable>
  </Application>
  <Resources>
    <QueueName>creamtest2</QueueName>
  </Resources>
</ActivityDescription>
```

```
$ glite-es-activity-create -e cream-47.pd.infn.it activity_sleep60_failif.adl
```

```
*****
```

```
ActivityID      = CR_ES908177323
ActivityMgrURI  = https://cream-47.pd.infn.it:8443/ce-cream-es/services/ActivityManagementService
Status         = PREPROCESSING
Status Attrs   = {}
Timestamp      = Mon Mar 12 12:18:57 2012
Description    =
ETNSC         =
STAGEIN Dir    = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
SESSION Dir    = {}
STAGEOUT Dir   = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
```

Eventually check with `glite-es-activity-status` the state which should be `TERMINAL` with attribute `APP-FAILURE`:

```
$ glite-es-activity-status -e cream-47.pd.infn.it CR_ES908177323
```

```
*****
```

```
JobID          = CR_ES908177323
Status         = TERMINAL
Attributes     = {APP_FAILURE}
Timestamp      = Mon Mar 12 12:20:01 2012
Description    = Executable exitCode 0 instead of 1
```

Status: Not Implemented

Activity with Input and Output: Client Data Push and Client Data Pull

Consider this ADL:

```
<ActivityDescription>
  <ActivityIdentification>
    <Name>cat</Name>
    <Description>cat</Description>
    <Type>single</Type>
  </ActivityIdentification>
  <Application>
    <Executable>
      <Path>/bin/cat</Path>
```

CreamTestWorkPlan < CREAM < TWiki

```
</Executable>
<Input>file.inp</Input>
<Output>file.out</Output>
</Application>
<Resources>
  <QueueName>creamtest2</QueueName>
</Resources>
<DataStaging>
  <ClientDataPush>>true</ClientDataPush>
  <OutputFile>
    <Name>file.out</Name>
  </OutputFile>
</DataStaging>
</ActivityDescription>
```

Submit it:

```
$ glite-es-activity-create -e cream-47.pd.infn.it datastag0.adl
*****
```

```
ActivityID      = CR_ES853730468
ActivityMgrURI  = https://cream-47.pd.infn.it:8443/ce-cream-es/services/ActivityManagementService
Status         = PREPROCESSING
Status Attrs   = {CLIENT_STAGEIN_POSSIBLE}
Timestamp      = Tue Mar 13 09:20:33 2012
Description    =
ETNSC         =
STAGEIN Dir    = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
SESSION Dir    = {}
STAGEOUT Dir   = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
```

Stage the input files in the STAGEIN Dir:

```
[sgaravat@cream-50 ADLExamples]$ uberftp cream-47.pd.infn.it
220 cream-47.pd.infn.it GridFTP Server 6.5 (gcc64, 1323378368-83) [unknown] ready.
230 User dteam042 logged in.
UberFTP> cd /var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_Personal_Certificate_O_INFN_C_IT_d
UberFTP> put file.inp
file.inp: 18 bytes in 0.015343 Seconds (1.146 KB/s)
UberFTP> bye
221 Goodbye.
```

Notify the service that the stagein was done:

```
[sgaravat@cream-50 ADLExamples]$ glite-es-notify-service -e cream-47.pd.infn.it CR_ES853730468:CL
```

Check the status that eventually should be TERMINAL with CLIENT_STAGEOUT_POSSIBLE as attribute:

```
[sgaravat@cream-50 ADLExamples]$ glite-es-activity-status -e cream-47.pd.infn.it CR_ES853730468
*****
JobID          = CR_ES853730468
Status        = TERMINAL
Attributes    = {CLIENT_STAGEOUT_POSSIBLE}
Timestamp     = Tue Mar 13 09:21:22 2012
Description   = reason=0
```

Retrieve the output files from the STAGEOUT Dir:

```
[sgaravat@cream-50 ADLExamples]$ uberftp cream-47.pd.infn.it220 cream-47.pd.infn.it GridFTP Serve
230 User dteam042 logged in.
UberFTP> cd /var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_Personal_Certificate_O_INFN_C_IT_d
UberFTP> ls
drwx-----  3 dteam042    dteam          4096 Mar 13 09:21 .
drwx-----  4 dteam042    dteam          4096 Mar 13 09:21 ..
-rw-r--r--  1 dteam042    dteam           18 Mar 13 09:21 file.out
```

CreamTestWorkPlan < CREAM < TWiki

```
UberFTP> get file.out
file.out: 18 bytes in 0.010151 Seconds (1.732 KB/s)
UberFTP> bye
221 Goodbye.
```

Notify the service that the stageout was done:

```
[sgaravat@cream-50 ADLEExamples]$ glite-es-notify-service -e cream-47.pd.infn.it CR_ES853730468:CL
```

Check the job status which should be TERMINAL without attributes:

```
[sgaravat@cream-50 ADLEExamples]$ glite-es-activity-status -e cream-47.pd.infn.it CR_ES853730468
*****
JobID      = CR_ES853730468
Status     = TERMINAL
Attributes = {}
Timestamp  = Tue Mar 13 09:21:23 2012
Description = reason=0
```

Status: Not Implemented

Activity with Input and Output: Server Data Push and Server Data Pull

Delegate a proxy:

```
[sgaravat@cream-50 ADLEExamples]$ glite-es-delegate-proxy -e cream-47.pd.infn.it
DelegationID = 32141661326088133
```

Consider this JDL (the DelegationID should be the one returned by the glite-es-delegate-proxy operation):

```
[sgaravat@cream-50 ADLEExamples]$ cat datastag1.adl
<ActivityDescription>
  <ActivityIdentification>
    <Name>cat</Name>
    <Description>cat</Description>
    <Type>single</Type>
  </ActivityIdentification>
  <Application>
    <Executable>
      <Path>/bin/cat</Path>
    </Executable>
    <Input>file.inp</Input>
    <Output>file.out</Output>
  </Application>
  <Resources>
    <QueueName>creamtest2</QueueName>
  </Resources>
  <DataStaging>
    <InputFile>
      <Name>file.inp</Name>
      <Source>
        <URI>gsiftp://cream-35.pd.infn.it/tmp/file.inp</URI>
        <DelegationID>32141661326088133</DelegationID>
      </Source>
    </InputFile>
    <OutputFile>
      <Name>file.out</Name>
      <Target>
        <URI>gsiftp://cream-35.pd.infn.it/tmp/file.out</URI>
        <DelegationID>32141661326088133</DelegationID>
      </Target>
    </OutputFile>
  </DataStaging>
```

CreamTestWorkPlan < CREAM < TWiki

</ActivityDescription>

Create the input file `gsiftp://cream-35.pd.infn.it/tmp/file.inp`.

Submit the activity:

```
[sgaravat@cream-50 ADLExamples]$ glite-es-activity-create -e cream-47.pd.infn.it datastag1.adl
*****
ActivityID      = CR_ES691768495
ActivityMgrURI  = https://cream-47.pd.infn.it:8443/ce-cream-es/services/ActivityManagementService
Status         = PREPROCESSING
Status Attrs   = {}
Timestamp      = Tue Mar 13 09:39:34 2012
Description     =
ETNSC          =
STAGEIN Dir    = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
SESSION Dir    = {}
STAGEOUT Dir   = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
```

Check the job status which should eventually be `TERMINAL`:

```
[sgaravat@cream-50 ADLExamples]$ glite-es-activity-status -e cream-47.pd.infn.it CR_ES691768495
*****
JobID          = CR_ES691768495
Status         = TERMINAL
Attributes     = {}
Timestamp      = Tue Mar 13 09:39:45 2012
Description    = reason=0
```

Verify that the specified output file (`gsiftp://cream-35.pd.infn.it/tmp/file.out`) has been produced.

Status: Not Implemented

Activity with Input and Output: Server and Client Data Push, Server and Client Data Pull

Delegate a proxy:

```
[sgaravat@cream-50 ADLExamples]$ glite-es-delegate-proxy -e cream-47.pd.infn.it
DelegationID = 3368110324080861
```

Consider this ADL (the `DelegationID`) must be the one returned by the `glite-es-delegate-proxy` command:

```
[sgaravat@cream-50 ADLExamples]$ cat datastag2.adl
<ActivityDescription>
  <ActivityIdentification>
    <Name>cat</Name>
    <Description>cat</Description>
    <Type>single</Type>
  </ActivityIdentification>
  <Application>
    <Executable>
      <Path>myscript.sh</Path>
    </Executable>
    <Output>myscript.out</Output>
    <Error>myscript.err</Error>
  </Application>
  <Resources>
    <QueueName>creamtest2</QueueName>
  </Resources>
  <DataStaging>
    <ClientDataPush>true</ClientDataPush>
```

CreamTestWorkPlan < CREAM < TWiki

```
<InputFile>
  <Name>file.inp</Name>
  <Source>
    <URI>gsiftp://cream-35.pd.infn.it/tmp/file.inp</URI>
    <DelegationID>3368110324080861</DelegationID>
  </Source>
</InputFile>
<OutputFile>
  <Name>myscript.out</Name>
  <Target>
    <URI>gsiftp://cream-35.pd.infn.it/tmp/myscript.out</URI>
    <DelegationID>3368110324080861</DelegationID>
  </Target>
</OutputFile>
<OutputFile>
  <Name>myscript.err</Name>
</OutputFile>
</DataStaging>
</ActivityDescription>
```

Create the input file `gsiftp://cream-35.pd.infn.it/tmp/file.inp` and submit the activity:

```
[sgaravat@cream-50 ADLExamples]$ glite-es-activity-create -e cream-47.pd.infn.it datastag2.adl
*****
ActivityID      = CR_ES526044183
ActivityMgrURI  = https://cream-47.pd.infn.it:8443/ce-cream-es/services/ActivityManagementService
Status         = PREPROCESSING
Status Attrs   = {CLIENT_STAGEIN_POSSIBLE}
Timestamp      = Tue Mar 13 10:01:29 2012
Description    =
ETNSC         =
STAGEIN Dir    = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
SESSION Dir    = {}
STAGEOUT Dir   = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
```

Upload the input file in the STAGEIN Dir and notify the service that the upload was done:

```
[sgaravat@cream-50 ADLExamples]$ uberftp cream-47.pd.infn.it220 cream-47.pd.infn.it GridFTP Serve
230 User dteam042 logged in.
UberFTP> cd var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_Personal_Certificate_O_INFN_C_IT_dt
[sgaravat@cream-50 ADLExamples]$ uberftp cream-47.pd.infn.it
220 cream-47.pd.infn.it GridFTP Server 6.5 (gcc64, 1323378368-83) [unknown] ready.
230 User dteam042 logged in.
UberFTP> cd /var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_Personal_Certificate_O_INFN_C_IT_d
UberFTP> put myscript.sh
myscript.sh: 50 bytes in 0.015672 Seconds (3.116 KB/s)
UberFTP> bye
221 Goodbye.
[sgaravat@cream-50 ADLExamples]$ glite-es-notify-service -e cream-47.pd.infn.it CR_ES526044183:CL
```

Check the status which eventually should be **TERMINAL** with **CLIENT_STAGEOUT_POSSIBLE** as attribute:

```
[sgaravat@cream-50 ADLExamples]$ glite-es-activity-status -e cream-47.pd.infn.it CR_ES526044183
*****
JobID          = CR_ES526044183
Status        = TERMINAL
Attributes    = {CLIENT_STAGEOUT_POSSIBLE}
Timestamp     = Tue Mar 13 10:02:25 2012
Description   = reason=0
```

Download the output files from the StageOUT Dir and notify the service:

```
[sgaravat@cream-50 ADLExamples]$ uberftp cream-47.pd.infn.it220 cream-47.pd.infn.it GridFTP Serve
```

CreamTestWorkPlan < CREAM < TWiki

```
230 User dteam042 logged in.
UberFTP> cd /var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_Personal_Certificate_O_INFN_C_IT_d
UberFTP> ls
drwx----- 3 dteam042    dteam          4096 Mar 13 10:02 .
drwx----- 4 dteam042    dteam          4096 Mar 13 10:02 ..
-rw-r--r--  1 dteam042    dteam           55 Mar 13 10:02 myscript.err
UberFTP> get myscript.err
myscript.err: 55 bytes in 0.012255 Seconds (4.383 KB/s)
UberFTP> bye
221 Goodbye.
[sgaravat@cream-50 ADLExamples]$ glite-es-notify-service -e cream-47.pd.infn.it CR_ES526044183:CL
```

Check the status which should be TERMINAL without attributes:

```
[sgaravat@cream-50 ADLExamples]$ glite-es-activity-status -e cream-47.pd.infn.it CR_ES526044183
*****
JobID      = CR_ES526044183
Status     = TERMINAL
Attributes = {}
Timestamp  = Tue Mar 13 10:02:26 2012
Description = reason=0
```

Verify that the output file `gsiftp://cream-35.pd.infn.it/tmp/myscript.out` has been produced.

Status: Not Implemented

Activity with Input and Output: Server and Client Data Push, Server and Client Data Pull, and multiple Delegations

Perform two delegations:

```
[sgaravat@cream-50 ADLExamples]$ glite-es-delegate-proxy -e cream-47.pd.infn.it
DelegationID = 059683951543786695
[sgaravat@cream-50 ADLExamples]$
[sgaravat@cream-50 ADLExamples]$ glite-es-delegate-proxy -e cream-47.pd.infn.it
DelegationID = 5352938890620005
```

Consider this ADL (the 2 delegationids must be the ones returned by the `glite-es-delegate-proxy` commands):

```
[sgaravat@cream-50 ADLExamples]$ cat datastag3.adl
<ActivityDescription>
  <ActivityIdentification>
    <Name>cat</Name>
    <Description>cat</Description>
    <Type>single</Type>
  </ActivityIdentification>
  <Application>
    <Executable>
      <Path>myscript.sh</Path>
    </Executable>
    <Output>myscript.out</Output>
    <Error>myscript.err</Error>
  </Application>
  <Resources>
    <QueueName>creamtest2</QueueName>
  </Resources>
  <DataStaging>
    <ClientDataPush>true</ClientDataPush>
    <InputFile>
      <Name>file.inp</Name>
      <Source>
        <URI>gsiftp://cream-35.pd.infn.it/tmp/file.inp</URI>
```

Activity with Input and Output: Server and Client Data Push, Server and Client Data Pull, and multiple Delegation

CreamTestWorkPlan < CREAM < TWiki

```
<DelegationID>059683951543786695</DelegationID>
</Source>
</InputFile>
<OutputFile>
  <Name>myscript.out</Name>
  <Target>
    <URI>gsiftp://cream-35.pd.infn.it/tmp/myscript.out</URI>
    <DelegationID>5352938890620005</DelegationID>
  </Target>
</OutputFile>
<OutputFile>
  <Name>myscript.err</Name>
</OutputFile>
</DataStaging>
</ActivityDescription>
```

Create the input file `gsiftp://cream-35.pd.infn.it/tmp/file.inp` and submit the activity:

```
[sgaravat@cream-50 ADLEExamples]$ glite-es-activity-create -e cream-47.pd.infn.it datastag3.adl
*****
ActivityID          = CR_ES034507131
ActivityMgrURI     = https://cream-47.pd.infn.it:8443/ce-cream-es/services/ActivityManagementService
Status             = PREPROCESSING
Status Attrs      = {CLIENT_STAGEIN_POSSIBLE}
Timestamp          = Tue Mar 13 10:15:11 2012
Description        =
ETNSC              =
STAGEIN Dir       = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
SESSION Dir       = {}
STAGEOUT Dir      = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
```

Upload the input files in the STAGEIN Dir and notify the service:

```
[sgaravat@cream-50 ADLEExamples]$ uberftp cream-47.pd.infn.it
220 cream-47.pd.infn.it GridFTP Server 6.5 (gcc64, 1323378368-83) [unknown] ready.
230 User dteam042 logged in.
UberFTP> cd /var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_Personal_Certificate_O_INFN_C_IT_d
UberFTP> put myscript.sh
myscript.sh: 50 bytes in 0.015260 Seconds (3.200 KB/s)
UberFTP> bye
221 Goodbye.
[sgaravat@cream-50 ADLEExamples]$ glite-es-notify-service -e cream-47.pd.infn.it CR_ES034507131:CL
```

Check the activity status which eventually should be `TERMINAL` with `CLIENT_STAGEOUT_POSSIBLE` as attribute:

```
[sgaravat@cream-50 ADLEExamples]$ glite-es-activity-status -e cream-47.pd.infn.it CR_ES034507131
*****
JobID              = CR_ES034507131
Status             = TERMINAL
Attributes        = {CLIENT_STAGEOUT_POSSIBLE}
Timestamp          = Tue Mar 13 10:16:05 2012
Description        = reason=0
```

Download the output files from the StageOUT Dir and notify the service:

```
[sgaravat@cream-50 ADLEExamples]$ uberftp cream-47.pd.infn.it220 cream-47.pd.infn.it GridFTP Serve
230 User dteam042 logged in.
UberFTP> cd /var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_Personal_Certificate_O_INFN_C_IT_d
UberFTP> ls
drwx-----  3 dteam042    dteam          4096 Mar 13 10:16 .
drwx-----  4 dteam042    dteam          4096 Mar 13 10:16 ..
-rw-r--r--   1 dteam042    dteam           55 Mar 13 10:16 myscript.err
UberFTP> get myscript.err
```

Activity with Input and Output: Server and Client Data Push, Server and Client Data Pull, and multiple Delegation

CreamTestWorkPlan < CREAM < TWiki

```
myscript.err: 55 bytes in 0.011562 Seconds (4.645 KB/s)
UberFTP> bye
221 Goodbye.
[sgaravat@cream-50 ADLEExamples]$ glite-es-notify-service -e cream-47.pd.infn.it CR_ES034507131:CL
```

Check the status which should be **TERMINAL** without attributes:

```
[sgaravat@cream-50 ADLEExamples]$ glite-es-activity-status -e cream-47.pd.infn.it CR_ES034507131
*****
JobID      = CR_ES034507131
Status    = TERMINAL
Attributes = {}
Timestamp = Tue Mar 13 10:16:05 2012
Description = reason=0
```

Verify that the output file `gsiftp://cream-35.pd.infn.it/tmp/myscript.out` has been created.

Status: Not Implemented

Activity with Input and Output with multiple Sources (one wrong, one correct)

Delegate a proxy:

```
[sgaravat@cream-50 ADLEExamples]$ glite-es-delegate-proxy -e cream-47.pd.infn.it
DelegationID = 7658329806304154
```

Consider the following ADL (the delegationid is the one returned by the previous command):

```
[sgaravat@cream-50 ADLEExamples]$ cat datastag8.adl
<ActivityDescription>
  <ActivityIdentification>
    <Name>cat</Name>
    <Description>cat</Description>
    <Type>single</Type>
  </ActivityIdentification>
  <Application>
    <Executable>
      <Path>/bin/cat</Path>
    </Executable>
    <Input>file.inp</Input>
    <Output>file.out</Output>
  </Application>
  <Resources>
    <QueueName>creamtest2</QueueName>
  </Resources>
  <DataStaging>
    <InputFile>
      <Name>file.inp</Name>
      <Source>
        <URI>gsiftp://host1.domain1/tmp/file.inp</URI>
        <DelegationID>7658329806304154</DelegationID>
      </Source>
      <Source>
        <URI>gsiftp://cream-35.pd.infn.it/tmp/file.inp</URI>
        <DelegationID>7658329806304154</DelegationID>
      </Source>
    </InputFile>
    <OutputFile>
      <Name>file.out</Name>
      <Target>
        <URI>gsiftp://cream-35.pd.infn.it/tmp/file.out</URI>
        <DelegationID>7658329806304154</DelegationID>
      </Target>
    </OutputFile>
  </DataStaging>
</ActivityDescription>
```


CreamTestWorkPlan < CREAM < TWiki

```
</Target>
</OutputFile>
</DataStaging>
</ActivityDescription>
```

Create the input file `gsiftp://cream-35.pd.infn.it/tmp/file.inp` and submit the activity:

```
[sgaravat@cream-50 ADLExamples]$ glite-es-activity-create -e cream-47.pd.infn.it datastag8.adl
*****
ActivityID      = CR_ES475196853
ActivityMgrURI  = https://cream-47.pd.infn.it:8443/ce-cream-es/services/ActivityManagementService
Status         = PREPROCESSING
Status Attrs   = {}
Timestamp      = Thu Mar 15 11:39:07 2012
Description    =
ETNSC         =
STAGEIN Dir   = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
SESSION Dir   = {}
STAGEOUT Dir  = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
```

Check the status which should eventually be `TERMINAL`:

```
[sgaravat@cream-50 ADLExamples]$ glite-es-activity-status -e cream-47.pd.infn.it CR_ES475196853
*****
JobID          = CR_ES475196853
Status        = TERMINAL
Attributes     = {}
Timestamp     = Thu Mar 15 11:39:18 2012
Description   = reason=0
```

Verify that the output file `gsiftp://cream-35.pd.infn.it/tmp/file.out` has been created.

Status: Not Implemented

Activity with Input and Output with multiple Targets

Create a delegation:

```
[sgaravat@cream-50 ADLExamples]$ glite-es-delegate-proxy -e cream-47.pd.infn.it
DelegationID = 6749966998262459
```

Consider this ADL, where the delegationid is the one returned by the `glite-es-delegate-proxy` command:

```
[sgaravat@cream-50 ADLExamples]$ cat datastag5.adl
<ActivityDescription>
  <ActivityIdentification>
    <Name>cat</Name>
    <Description>cat</Description>
    <Type>single</Type>
  </ActivityIdentification>
  <Application>
    <Executable>
      <Path>/bin/cat</Path>
    </Executable>
    <Input>file.inp</Input>
    <Output>file.out</Output>
  </Application>
  <Resources>
    <QueueName>creamtest2</QueueName>
  </Resources>
  <DataStaging>
    <InputFile>
```

CreamTestWorkPlan < CREAM < TWiki

```
<Name>file.inp</Name>
<Source>
  <URI>gsiftp://cream-35.pd.infn.it/tmp/file.inp</URI>
  <DelegationID>6749966998262459</DelegationID>
</Source>
</InputFile>
<OutputFile>
  <Name>file.out</Name>
  <Target>
    <URI>gsiftp://cream-47.pd.infn.it/tmp/file.out</URI>
    <DelegationID>6749966998262459</DelegationID>
    <Mandatory>>true</Mandatory>
  </Target>
  <Target>
    <URI>gsiftp://cream-35.pd.infn.it/tmp/file.out</URI>
    <DelegationID>6749966998262459</DelegationID>
    <Mandatory>>true</Mandatory>
  </Target>
</OutputFile>
</DataStaging>
</ActivityDescription>
```

Create the input file `gsiftp://cream-35.pd.infn.it/tmp/file.inp` and submit the activity:

```
[sgaravat@cream-50 ADLExamples]$ glite-es-activity-create -e cream-47.pd.infn.it datastag5.adl
*****
ActivityID      = CR_ES417521144
ActivityMgrURI  = https://cream-47.pd.infn.it:8443/ce-cream-es/services/ActivityManagementService
Status         = PREPROCESSING
Status Attrs   = {}
Timestamp      = Tue Mar 13 10:33:14 2012
Description     =
ETNSC          =
STAGEIN Dir    = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
SESSION Dir    = {}
STAGEOUT Dir   = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
```

Check the job status which eventually should be `TERMINAL` without attributes:

```
[sgaravat@cream-50 ADLExamples]$ glite-es-activity-status -e cream-47.pd.infn.it CR_ES417521144
*****
JobID          = CR_ES417521144
Status         = TERMINAL
Attributes     = {}
Timestamp      = Tue Mar 13 10:33:26 2012
Description    = reason=0
```

Verify that the output files `gsiftp://cream-47.pd.infn.it/tmp/file.out` and `gsiftp://cream-35.pd.infn.it/tmp/file.out` have been both created.

Status: Not Implemented

Activity with Input and Output specifying a non existing delegationid

Consider this ADL (where the specified delegationid doesn't exist):

```
[sgaravat@cream-50 ADLExamples]$ cat datastagnodel.adl
<ActivityDescription>
  <ActivityIdentification>
    <Name>cat</Name>
    <Description>cat</Description>
    <Type>single</Type>
  </ActivityIdentification>
```

CreamTestWorkPlan < CREAM < TWiki

```
<Application>
  <Executable>
    <Path>myscript.sh</Path>
  </Executable>
  <Output>myscript.out</Output>
  <Error>myscript.err</Error>
</Application>
<Resources>
  <QueueName>creamtest2</QueueName>
</Resources>
<DataStaging>
  <ClientDataPush>>true</ClientDataPush>
  <InputFile>
    <Name>file.inp</Name>
    <Source>
      <URI>gsiftp://cream-35.pd.infn.it/tmp/file.inp</URI>
      <DelegationID>xyz</DelegationID>
    </Source>
  </InputFile>
  <OutputFile>
    <Name>myscript.out</Name>
    <Target>
      <URI>gsiftp://cream-35.pd.infn.it/tmp/myscript.out</URI>
      <DelegationID>xyz</DelegationID>
    </Target>
  </OutputFile>
  <OutputFile>
    <Name>myscript.err</Name>
  </OutputFile>
</DataStaging>
</ActivityDescription>
```

and try to submit it. An error should be returned:

```
[sgaravat@cream-50 ADLEexamples]$ glite-es-activity-create -e cream-47.pd.infn.it datastagnodel.ad
*****
ActivityID          =
FAULT: Message=[delegation [delegId=xyz; dn=CN_Massimo_Sgaravatto_L_Padova_OU_Personal_Certifica
ActivityMgrURI     =
Status             = N/A
ETNSC              =
STAGEIN Dir        = {}
SESSION Dir        = {}
STAGEOUT Dir       = {}
```

Status: Not Implemented

Proxy renewal of multiple delegations

Create a short proxy:

```
[sgaravat@cream-50 ADLEexamples]$ voms-proxy-init -voms dteam -valid 00:30
Enter GRID pass phrase:
Your identity: /C=IT/O=INFN/OU=Personal Certificate/L=Padova/CN=Massimo Sgaravatto
Creating temporary proxy ..... Done
Contacting voms2.hellasgrid.gr:15004 [/C=GR/O=HellasGrid/OU=hellasgrid.gr/CN=voms2.hellasgrid.gr]
Creating proxy ..... Done

Your proxy is valid until Sun Mar 18 10:12:19 2012
```

Delegate it twice, i.e. create two delegations:

```
[sgaravat@cream-50 ADLEexamples]$ glite-es-delegate-proxy -e cream-47.pd.infn.it
DelegationID = 5670349996819731
```

CreamTestWorkPlan < CREAM < TWiki

```
[sgaravat@cream-50 ADLExamples]$ glite-es-delegate-proxy -e cream-47.pd.infn.it
DelegationID = 6261618400431184
```

Consider this ADL (the delegationids are the ones returned by the glite-es-delegate-proxy commands):

```
[sgaravat@cream-50 ADLExamples]$ cat proxyrenewal.adl
<ActivityDescription>
  <ActivityIdentification>
    <Name>Proxy Renewal</Name>
    <Description>Proxy Renewal</Description>
    <Type>single</Type>
  </ActivityIdentification>
  <Application>
    <Executable>
      <Path>proxyrenewal.sh</Path>
    </Executable>
    <Input>file.inp</Input>
    <Output>file.out</Output>
  </Application>
  <Resources>
    <QueueName>creamtest2</QueueName>
  </Resources>
  <DataStaging>
    <ClientDataPush>>true</ClientDataPush>
    <InputFile>
      <Name>file.inp</Name>
      <Source>
        <URI>gsiftp://cream-35.pd.infn.it/tmp/file.inp</URI>
        <DelegationID>5670349996819731</DelegationID>
      </Source>
    </InputFile>
    <OutputFile>
      <Name>file.out</Name>
      <Target>
        <URI>gsiftp://cream-35.pd.infn.it/tmp/file.out</URI>
        <DelegationID>6261618400431184</DelegationID>
      </Target>
    </OutputFile>
  </DataStaging>
</ActivityDescription>
```

```
[sgaravat@cream-50 ADLExamples]$ cat proxyrenewal.sh
#!/bin/sh
sleep 3600
/usr/bin/tee
```

Create the input file `gsiftp://cream-35.pd.infn.it/tmp/file.inp` and submit this activity:

```
sgaravat@cream-50 ADLExamples]$ glite-es-activity-create -e cream-47.pd.infn.it proxyrenewal.adl
*****
ActivityID      = CR_ES898737831
ActivityMgrURI  = https://cream-47.pd.infn.it:8443/ce-cream-es/services/ActivityManagementService
Status         = PREPROCESSING
Status Attrs   = {CLIENT_STAGEIN_POSSIBLE}
Timestamp      = Sun Mar 18 09:47:58 2012
Description     =
ETNSC         =
STAGEIN Dir    = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
SESSION Dir    = {}
STAGEOUT Dir   = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
```

Upload the proxyrenewal.sh input file and notify the service:

```
[sgaravat@cream-50 ADLExamples]$ uberftp cream-47.pd.infn.it220 cream-47.pd.infn.it GridFTP Serve
```

CreamTestWorkPlan < CREAM < TWiki

```
230 User dteam004 logged in.
UberFTP> cd /var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_Personal_Certificate_O_INFN_C_IT_d
UberFTP> put proxyrenewal.sh
proxyrenewal.sh: 34 bytes in 0.011920 Seconds (2.785 KB/s)
UberFTP> bye
221 Goodbye.
[sgaravat@cream-50 ADLEExamples]$ glite-es-notify-service -e cream-47.pd.infn.it CR_ES898737831:CL
```

After 10 minutes create a long proxy and renew the 2 delegations:

```
[sgaravat@cream-50 ADLEExamples]$ voms-proxy-init -voms dteam
Enter GRID pass phrase:
Your identity: /C=IT/O=INFN/OU=Personal Certificate/L=Padova/CN=Massimo Sgaravatto
Creating temporary proxy ..... Done
Contacting voms2.hellasgrid.gr:15004 [/C=GR/O=HellasGrid/OU=hellasgrid.gr/CN=voms2.hellasgrid.gr]
Creating proxy ..... Done

Your proxy is valid until Sun Mar 18 21:51:31 2012

[sgaravat@cream-50 ADLEExamples]$ glite-es-delegation-renew -e cream-47.pd.infn.it 567034999681973
Delegation with identifier [5670349996819731] successfully renewed
[sgaravat@cream-50 ADLEExamples]$ glite-es-delegation-renew -e cream-47.pd.infn.it 626161840043118
Delegation with identifier [6261618400431184] successfully renewed
```

After 30 minutes check the job status which should be TERMINAL without attributes:

```
[sgaravat@cream-50 ADLEExamples]$ glite-es-activity-status -e cream-47.pd.infn.it CR_ES898737831
*****
JobID          = CR_ES898737831
Status         = TERMINAL
Attributes     = {}
Timestamp      = Sun Mar 18 10:49:42 2012
Description    = reason=0
```

Verify also that the output file `gsiftp://cream-35.pd.infn.it/tmp/file.out` has been created.

Status: Not Implemented

Submission of multiple activities

Consider the follow ADL:

```
[sgaravat@cream-50 ADLEExamples]$ cat multi.adl
<CreateActivities>
  <ActivityDescription>
    <ActivityIdentification>
      <Name>sleep60</Name>
      <Description>sleep 60</Description>
      <Type>single</Type>
    </ActivityIdentification>
    <Application>
      <Executable>
        <Path>/bin/sleep</Path>
        <Argument>60</Argument>
      </Executable>
    </Application>
    <Resources>
      <QueueName>creamtest2</QueueName>
    </Resources>
  </ActivityDescription>
  <ActivityDescription>
    <ActivityIdentification>
      <Name>sleep60</Name>
      <Description>sleep 60</Description>
```

Submission of multiple activities

CreamTestWorkPlan < CREAM < TWiki

```
<Type>single</Type>
</ActivityIdentification>
<Application>
  <Executable>
    <Path>/bin/sleep</Path>
    <Argument>600</Argument>
  </Executable>
</Application>
<Resources>
  <QueueName>creamtest2</QueueName>
</Resources>
</ActivityDescription>
</CreateActivities>
```

Submit it (two jobs should be submitted):

```
[sgaravat@cream-50 ADLExamples]$ glite-es-activity-create -e cream-47.pd.infn.it multi.adl
*****
ActivityID      = CR_ES063075266
ActivityMgrURI  = https://cream-47.pd.infn.it:8443/ce-cream-es/services/ActivityManagementService
Status         = PREPROCESSING
Status Attrs   = {}
Timestamp      = Tue Mar 13 16:12:02 2012
Description     =
ETNSC          =
STAGEIN Dir    = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
SESSION Dir    = {}
STAGEOUT Dir   = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
*****
ActivityID      = CR_ES016651242
ActivityMgrURI  = https://cream-47.pd.infn.it:8443/ce-cream-es/services/ActivityManagementService
Status         = PREPROCESSING
Status Attrs   = {}
Timestamp      = Tue Mar 13 16:12:03 2012
Description     =
ETNSC          =
STAGEIN Dir    = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
SESSION Dir    = {}
STAGEOUT Dir   = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
```

Check their status using a single command:

```
[sgaravat@cream-50 ADLExamples]$ glite-es-activity-status -e cream-47.pd.infn.it CR_ES016651242 C
*****
JobID          = CR_ES016651242
Status         = PROCESSING_RUNNING
Attributes     = {}
Timestamp      = Tue Mar 13 16:12:15 2012
Description    =
*****
JobID          = CR_ES063075266
Status         = TERMINAL
Attributes     = {}
Timestamp      = Tue Mar 13 16:13:10 2012
Description    = reason=0
[sgaravat@cream-50 ADLExamples]$
```

Status: Not Implemented

Validation for unsupported capability

Consider a ADL specifying a not supported attribute (e.g. RemoteLogging):

```
[sgaravat@cream-50 ADLExamples]$ cat remotelogging.adl
```

Validation for unsupported capability

```
<ActivityDescription>
  <ActivityIdentification>
    <Name>sleep60</Name>
    <Description>sleep 60</Description>
    <Type>single</Type>
  </ActivityIdentification>
  <Application>
    <Executable>
      <Path>/bin/sleep</Path>
      <Argument>60</Argument>
    </Executable>
    <RemoteLogging>
      <ServiceType>LB</ServiceType>
      <URL>https://locallogger.com</URL>
    </RemoteLogging>
  </Application>
  <Resources>
    <QueueName>creamtest2</QueueName>
  </Resources>
  <DataStaging>
  </DataStaging>
</ActivityDescription>
```

Submit it: a "capability not supported" message should be returned:

```
[sgaravat@cream-50 ADLExamples]$ glite-es-activity-create -e cream-47.pd.infn.it remotelogging.adl
*****
ActivityID      =
FAULT: Message=[remote logging capability not supported!] Timestamp=[1332346982]
ActivityMgrURI =
Status         = N/A
ETNSC         =
STAGEIN Dir   = {}
SESSION Dir   = {}
STAGEOUT Dir  = {}
```

Repeat the test specifying optional=false. The same error message should be returned:

```
[sgaravat@cream-50 ADLExamples]$ cat remotelogging.adl
<ActivityDescription>
  <ActivityIdentification>
    <Name>sleep60</Name>
    <Description>sleep 60</Description>
    <Type>single</Type>
  </ActivityIdentification>
  <Application>
    <Executable>
      <Path>/bin/sleep</Path>
      <Argument>60</Argument>
    </Executable>
    <RemoteLogging optional="false">
      <ServiceType>LB</ServiceType>
      <URL>https://locallogger.com</URL>
    </RemoteLogging>
  </Application>
  <Resources>
    <QueueName>creamtest2</QueueName>
  </Resources>
  <DataStaging>
  </DataStaging>
</ActivityDescription>
```

```
[sgaravat@cream-50 ADLExamples]$ glite-es-activity-create -e cream-47.pd.infn.it remotelogging.adl
*****
ActivityID      =
```

CreamTestWorkPlan < CREAM < TWiki

```
FAULT: Message=[remote logging capability not supported!] Timestamp=[1332347088]
ActivityMgrURI =
Status        = N/A
ETNSC        =
STAGEIN Dir   = {}
SESSION Dir   = {}
STAGEOUT Dir  = {}
```

Repeat the test specifying optional=true. In this case the activity should be submitted:

```
[sgaravat@cream-50 ADLExamples]$ cat remotelogging.adl
<ActivityDescription>
  <ActivityIdentification>
    <Name>sleep60</Name>
    <Description>sleep 60</Description>
    <Type>single</Type>
  </ActivityIdentification>
  <Application>
    <Executable>
      <Path>/bin/sleep</Path>
      <Argument>60</Argument>
    </Executable>
    <RemoteLogging optional="true">
      <ServiceType>LB</ServiceType>
      <URL>https://locallogger.com</URL>
    </RemoteLogging>
  </Application>
  <Resources>
    <QueueName>creamtest2</QueueName>
  </Resources>
  <DataStaging>
  </DataStaging>
</ActivityDescription>
```

```
[sgaravat@cream-50 ADLExamples]$ glite-es-activity-create -e cream-47.pd.infn.it remotelogging.adl
*****
ActivityID      = CR_ES663995663
ActivityMgrURI  = https://cream-47.pd.infn.it:8443/ce-cream-es/services/ActivityManagementService
Status         = PREPROCESSING
Status Attrs   = {}
Timestamp      = Wed Mar 21 17:25:09 2012
Description     =
ETNSC         =
STAGEIN Dir    = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
SESSION Dir    = {}
STAGEOUT Dir   = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
```

Status: Not Implemented

Pre and Post Executables

Consider this ADL:

```
[sgaravat@cream-50 ADLExamples]$ cat prepost.adl
<ActivityDescription>
  <ActivityIdentification>
    <Name>prepost</Name>
    <Description>prepost</Description>
    <Type>single</Type>
  </ActivityIdentification>
  <Application>
    <Executable>
      <Path>/bin/hostname</Path>
      <Argument>-f</Argument>
```


CreamTestWorkPlan < CREAM < TWiki

```
</Executable>
<Output>prepost.out</Output>
<PreExecutable>
  <Path>/bin/touch</Path>
  <Argument>/tmp/file1</Argument>
</PreExecutable>
<PreExecutable>
  <Path>/bin/touch</Path>
  <Argument>/tmp/file2</Argument>
</PreExecutable>
<PostExecutable>
  <Path>/bin/touch</Path>
  <Argument>/tmp/file3</Argument>
</PostExecutable>
<PostExecutable>
  <Path>/bin/touch</Path>
  <Argument>/tmp/file4</Argument>
</PostExecutable>
</Application>
<Resources>
  <QueueName>creamtest2</QueueName>
</Resources>
<DataStaging>
  <OutputFile>
    <Name>prepost.out</Name>
  </OutputFile>
</DataStaging>
</ActivityDescription>
```

Submit it:

```
[sgaravat@cream-50 ADLEexamples]$ glite-es-activity-create -e cream-47.pd.infn.it prepost.adl
*****
ActivityID      = CR_ES121359049
ActivityMgrURI  = https://cream-47.pd.infn.it:8443/ce-cream-es/services/ActivityManagementService
Status         = PREPROCESSING
Status Attrs   = {}
Timestamp      = Mon Mar 19 09:36:28 2012
Description    =
ETNSC         =
STAGEIN Dir    = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
SESSION Dir    = {}
STAGEOUT Dir   = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
```

Wait the termination of the job and retrieve the output file:

```
[sgaravat@cream-50 ADLEexamples]$ glite-es-activity-status -e cream-47.pd.infn.it CR_ES121359049
*****
JobID          = CR_ES121359049
Status         = TERMINAL
Attributes     = {CLIENT_STAGEOUT_POSSIBLE}
Timestamp      = Mon Mar 19 09:36:38 2012
Description    = reason=0

[sgaravat@cream-50 ADLEexamples]$ uberftp cream-47.pd.infn.it
220 cream-47.pd.infn.it GridFTP Server 6.5 (gcc64, 1323378368-83) [Globus Toolkit 5.2.0] ready.
230 User dteam004 logged in.
UberFTP> cd /var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_Personal_Certificate_O_INFN_C_IT_d
UberFTP> ls
drwx-----  3 dteam004    dteam          4096 Mar 19 09:36 .
drwx-----  4 dteam004    dteam          4096 Mar 19 09:36 ..
-rw-r--r--   1 dteam004    dteam           26 Mar 19 09:36 prepost.out
UberFTP> get prepost.out
prepost.out: 26 bytes in 0.012021 Seconds (2.112 KB/s)
UberFTP> bye
221 Goodbye.
```

Check where the job was executed:

```
[sgaravat@cream-50 ADLExamples]$ cat prepost.out
prod-wn-002.pn.pd.infn.it
```

Verify that in these WNs the files created by the pre and post executables have been created:

```
[sgaravat@cream-50 ADLExamples]$ ssh root@prod-wn-002.pn.pd.infn.it
The authenticity of host 'prod-wn-002.pn.pd.infn.it (192.168.132.102)' can't be established.
RSA key fingerprint is ac:fb:ee:57:ec:16:90:82:e7:de:36:ef:cb:0e:f6:95.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'prod-wn-002.pn.pd.infn.it,192.168.132.102' (RSA) to the list of known
root@prod-wn-002.pn.pd.infn.it's password:
Last login: Sun Mar 18 09:50:57 2012 from cream-47.pd.infn.it
[root@prod-wn-002 ~]# ls -al /tmp/file*
-rw-r--r-- 1 dteam004 dteam 0 Mar 19 09:36 /tmp/file1
-rw-r--r-- 1 dteam004 dteam 0 Mar 19 09:36 /tmp/file2
-rw-r--r-- 1 dteam004 dteam 0 Mar 19 09:36 /tmp/file3
-rw-r--r-- 1 dteam004 dteam 0 Mar 19 09:36 /tmp/file4
```

Status: Not Implemented

Activity with Environment

Consider this activity:

```
[sgaravat@cream-50 ADLExamples]$ cat environment.adl
<ActivityDescription>
  <ActivityIdentification>
    <Name>env</Name>
    <Description>env</Description>
    <Type>single</Type>
  </ActivityIdentification>
  <Application>
    <Executable>
      <Path>/bin/env</Path>
    </Executable>
    <Output>env.out</Output>
    <Environment>
      <Name>colore</Name>
      <Value>rosso</Value>
    </Environment>
    <Environment>
      <Name>forma</Name>
      <Value>rombo</Value>
    </Environment>
  </Application>
  <Resources>
    <QueueName>creamtest2</QueueName>
  </Resources>
  <DataStaging>
    <OutputFile>
      <Name>env.out</Name>
    </OutputFile>
  </DataStaging>
</ActivityDescription>
```

Submit it:

```
[sgaravat@cream-50 ADLExamples]$ glite-es-activity-create -e cream-47.pd.infn.it environment.adl
*****
ActivityID      = CR_ES953523652
ActivityMgrURI  = https://cream-47.pd.infn.it:8443/ce-cream-es/services/ActivityManagementService
Status         = PREPROCESSING
```

CreamTestWorkPlan < CREAM < TWiki

```
Status Attrs      = {}
Timestamp         = Sun Mar 18 11:22:47 2012
Description       =
ETNSC            =
STAGEIN Dir      = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
SESSION Dir      = {}
STAGEOUT Dir     = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
```

When the status is `TERMINAL` with `CLIENT_STAGEOUT_POSSIBLE` retrieve the output file:

```
[sgaravat@cream-50 ADLEExamples]$ glite-es-activity-status -e cream-47.pd.infn.it CR_ES953523652
*****
JobID             = CR_ES953523652
Status           = TERMINAL
Attributes       = {CLIENT_STAGEOUT_POSSIBLE}
Timestamp        = Sun Mar 18 11:22:51 2012
Description      = reason=0

[sgaravat@cream-50 ADLEExamples]$ uberftp cream-47.pd.infn.it
220 cream-47.pd.infn.it GridFTP Server 6.5 (gcc64, 1323378368-83) [Globus Toolkit 5.2.0] ready.
230 User dteam004 logged in.
UberFTP> cd /var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_Personal_Certificate_O_INFN_C_IT_d
UberFTP> ls
drwx-----   3 dteam004    dteam           4096 Mar 18 11:22 .
drwx-----   4 dteam004    dteam           4096 Mar 18 11:22 ..
-rw-r--r--    1 dteam004    dteam           3179 Mar 18 11:22 env.out
UberFTP> get env.out
env.out: 3179 bytes in 0.012435 Seconds (249.658 KB/s)
UberFTP> bye
221 Goodbye.
```

Verify that in output file the two variables are defined:

```
[sgaravat@cream-50 ADLEExamples]$ grep -i colore env.out
colore=rosso
[sgaravat@cream-50 ADLEExamples]$ grep -i forma env.out
forma=rombo
```

Status: Not Implemented

List

Use the list operation and check if all your jobs are returned:

```
[sgaravat@cream-50 ADLEExamples]$ glite-es-activity-list -e cream-47.pd.infn.it
CR_ES121359049
CR_ES271831319
CR_ES355205210
CR_ES393049605
CR_ES699482608
CR_ES745555371
CR_ES799188399
CR_ES810395877
CR_ES898737831
CR_ES931295038
CR_ES953523652
CR_ES975108994
```

Test the limit option. Only at most the specified number of entries should be returned:

```
[sgaravat@cream-50 ADLEExamples]$ glite-es-activity-list -e cream-47.pd.infn.it -l 1
CR_ES121359049
```

Submit a job, and while it is running issue a list with the filter on the PROCESSING-RUNNING status:

```
[sgaravat@cream-50 ADLExamples]$ glite-es-activity-create -e cream-47.pd.infn.it activity_sleep60
*****
ActivityID      = CR_ES747877778
ActivityMgrURI  = https://cream-47.pd.infn.it:8443/ce-cream-es/services/ActivityManagementService
Status         = PREPROCESSING
Status Attrs   = {}
Timestamp      = Tue Mar 20 14:45:24 2012
Description     =
ETNSC         =
STAGEIN Dir    = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_Person
SESSION Dir    = {}
STAGEOUT Dir   = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_Person

[sgaravat@cream-50 ADLExamples]$ glite-es-activity-list -e cream-47.pd.infn.it -S PROCESSING_RUNNING
CR_ES747877778
```

Test the filter on the submission time, e.g.:

```
[sgaravat@cream-50 ADLExamples]$ glite-es-activity-list -e cream-47.pd.infn.it -F '2012-03-20 10:00:00'
CR_ES699482608
CR_ES747877778
CR_ES799188399

[sgaravat@cream-50 ADLExamples]$ glite-es-activity-list -e cream-47.pd.infn.it -F '2012-03-01 10:00:00'
CR_ES121359049
CR_ES271831319
CR_ES393049605
CR_ES745555371
CR_ES810395877
CR_ES898737831
CR_ES931295038
CR_ES953523652
CR_ES975108994
```

Status: Not Implemented

ActivityInfo

Submit an activity and then check it with the activityinfo operation, e.g.:

```
[sgaravat@cream-50 ADLExamples]$ glite-es-activity-info -e cream-47.pd.infn.it CR_ES799188399

**** ActivityID = CR_ES799188399

ID=https://cream-47.pd.infn.it:8443/ce-cream-es/services/ActivityManagementService?CR_ES799188399
Name=sleep60
OtherInfo={}
CreationTime=1332250577 (Tue Mar 20 14:36:17 2012)
Validity=N/A
**** Extension #0:
    LocalID=CR_ES799188399
    Key=ACTIVITY_HISTORY
    Value=ACTIVITY_HISTORY
**** Extension #1:
    LocalID=CR_ES799188399
    Key=STAGE_IN_URI
    Value=gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_Person
**** Extension #2:
    LocalID=CR_ES799188399
    Key=STAGE_OUT_URI
    Value=gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_Person
BaseType=
```

CreamTestWorkPlan < CREAM < TWiki

```
Type=SINGLE
IDFromEndpoint=https://cream-47.pd.infn.it:8443/ce-cream-es/services/ActivityManagementService?CR
LocalIDFromManager=lsf/20120320/607375
JobDescription=emi:adl
State={ACCEPTED, PREPROCESSING, PROCESSING-ACCEPTING, PROCESSING-QUEUED, PROCESSING-RUNNING, TERM
RestartState={}
ExitCode=0
ComputingManagerExitCode=N/A
Error={}
WaitingPosition=N/A
UserDomain=dteam
Owner=CN_Massimo_Sgaravatto_L_Padova_OU_Personal_Certificate_O_INFN_C_IT_dteam_Role_NULL_Capabili
LocalOwner=dteam004
RequestedTotalWallTime=N/A
RequestedTotalCPUTime=N/A
RequestedSlots=N/A
RequestedApplicationEnvironment={}
StdIn=N/A
StdOut=N/A
StdErr=N/A
LogDir=N/A
ExecutionNode=prod-wn-003.pn.pd.infn.it
Queue=creamtest2
UsedTotalWallTime=N/A
UsedMainMemory=N/A
SubmissionTime=1332250577 (Tue Mar 20 14:36:17 2012)
ComputingManagerSubmissionTime=1332250577 (Tue Mar 20 14:36:17 2012)
StartTime=1332250588 (Tue Mar 20 14:36:28 2012)
ComputingManagerEndTime=1332250648 (Tue Mar 20 14:37:28 2012)
EndTime=1332250648 (Tue Mar 20 14:37:28 2012)
WorkingAreaEraseTime=N/A
ProxyExpirationTime=N/A
SubmissionHost=N/A
SubmissionClientName=N/A
OtherMessages=
[sgaravat@cream-50 ADLExamples]$
```

Status: Not Implemented

Pause and Resume

Create a long activity:

```
[sgaravat@cream-50 ADLExamples]$ glite-es-activity-create -e cream-47.pd.infn.it activity_sleep60
*****
ActivityID      = CR_ES679500221
ActivityMgrURI  = https://cream-47.pd.infn.it:8443/ce-cream-es/services/ActivityManagementService
Status         = PREPROCESSING
Status Attrs   = {}
Timestamp      = Tue Mar 13 16:25:08 2012
Description    =
ETNSC         =
STAGEIN Dir    = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
SESSION Dir    = {}
STAGEOUT Dir   = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
```

Wait some minutes and then pause it:

```
[sgaravat@cream-50 ADLExamples]$ glite-es-activity-pause -e cream-47.pd.infn.it CR_ES679500221
```

Wait some minutes and the check its status which should be PROCESSING_RUNNING with CLIENT_PAUSED as attribute:

```
[sgaravat@cream-50 ADLExamples]$ glite-es-activity-status -e cream-47.pd.infn.it CR_ES679500221
```

CreamTestWorkPlan < CREAM < TWiki

```
*****
JobID      = CR_ES679500221
Status     = PROCESSING_RUNNING
Attributes = {CLIENT_PAUSED}
Timestamp  = Tue Mar 13 16:25:33 2012
Description =
```

Resume it:

```
[sgaravat@cream-50 ADLExamples]$ glite-es-activity-resume -e cream-47.pd.infn.it CR_ES679500221
```

Wait some minutes and then check its status again which should be **PROCESSING_RUNNING** without attributes:

```
[sgaravat@cream-50 ADLExamples]$ glite-es-activity-status -e cream-47.pd.infn.it CR_ES679500221
*****
JobID      = CR_ES679500221
Status     = PROCESSING_RUNNING
Attributes = {}
Timestamp  = Tue Mar 13 16:26:16 2012
Description =
```

Status: Not Implemented

Cancel

Submit a long activity:

```
[sgaravat@cream-50 ADLExamples]$ glite-es-activity-create -e cream-47.pd.infn.it activity_sleep60
*****
ActivityID   = CR_ES619765347
ActivityMgrURI = https://cream-47.pd.infn.it:8443/ce-cream-es/services/ActivityManagementService
Status       = PREPROCESSING
Status Attrs = {}
Timestamp    = Tue Mar 13 15:59:25 2012
Description   =
ETNSC        =
STAGEIN Dir  = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
SESSION Dir  = {}
STAGEOUT Dir = {gsiftp://cream-47.pd.infn.it/var/cs-es/dteam/CN_Massimo_Sgaravatto_L_Padova_OU_
```

Wait a while and then cancel it:

```
[sgaravat@cream-50 ADLExamples]$ glite-es-activity-cancel -e cream-47.pd.infn.it CR_ES619765347
```

Check the status that eventually should be **TERMINAL** with **xxx_CANCEL** as attribute:

```
[sgaravat@cream-50 ADLExamples]$ glite-es-activity-status -e cream-47.pd.infn.it CR_ES619765347
*****
JobID      = CR_ES619765347
Status     = TERMINAL
Attributes = {PROCESSING_CANCEL}
Timestamp  = Tue Mar 13 16:00:04 2012
Description = cancelled by the user
```

Status: Not Implemented

Wipe

Submit an activity and wait that the status is **TERMINAL**:

Cancel

CreamTestWorkPlan < CREAM < TWiki

```
[sgaravat@cream-50 ADLExamples]$ glite-es-activity-status -e cream-47.pd.infn.it CR_ES668400340
*****
JobID      = CR_ES668400340
Status     = TERMINAL
Attributes = {}
Timestamp  = Tue Mar 13 15:51:24 2012
Description = reason=0
```

Issue the wipe:

```
[sgaravat@cream-50 ADLExamples]$ glite-es-activity-wipe -e cream-47.pd.infn.it CR_ES668400340
```

Retry a status, which should report that the activity is not found:

```
[sgaravat@cream-50 ADLExamples]$ glite-es-activity-status -e cream-47.pd.infn.it CR_ES668400340
*****
JobID      = CR_ES668400340
Message    = Activity not found!
Timestamp  = Tue Mar 13 15:51:57 2012
Description = N/A
FailCode   = N/A
```

Verify that the job sandbox directory has been removed

Status: Not Implemented

Regression tests

See: <http://wiki.italiangrid.it/twiki/bin/view/CREAM/RegressionTestWorkPlan>

Performance and scalability tests

Submission and cancellation with polling

Submit 1000 simple jobs (without sandbox transfers) in bunch of 50 jobs and then cancel them.

Check the status of these jobs using the queryevent operation.

All jobs should eventually have CANCELLED as final state.

Status: Implemented in the old testsuite

```
cream-test-monitored-cancel -r 30 -n 1000 -m 50 -C 50 -l log4py.conf -j simple.jdl -R cream-35.pd
```

Submission with polling and sandbox transfer

Submit 1000 jobs (with sandbox transfers) in bunch of 50 jobs.

Check the status of these jobs using the queryevent operation.

All jobs should eventually have DONE-OK as final state.

Status: Implemented in the old testsuite

```
cream-test-monitored-submit -r 30 -n 1000 -m 50 -C 50 -l log4py.conf -j testsandbox.jdl -R cream-
```

Simple submission with polling, lease updated and proxy renewal

Submit 100 jobs lasting 6000 seconds in bunch of 10 jobs, with an initial lease of 1200 seconds and with an initial proxy valid 20 minutes.

Renew the lease and the proxy.

Check the status of these jobs using the queryevent operation.

All jobs should eventually have DONE-OK as final state.

Status: Implemented in the old testsuite

```
cream-test-monitored-lease-updated -r 60 -n 100 -m 10 -C 10 -l log4py.conf -W 1200 --vo dteam --v
```

Standard compliance and conformance tests

Glue 1 compliance

Glue 1 compliance of the information published by the CREAM-CE resource bdii should be tested. This should be done using GlueValidator .

Status: Implemented in the Robot based testsuite

Glue 2 compliance

Glue 2 compliance of the information published by the CREAM-CE resource bdii should be tested. This should be done using GlueValidator .

Status: Implemented in the Robot based testsuite

Nagios probe

For testing nagios probes for worker nodes see here , tests description for nagios probes for direct cream job submission is here

Robot based Testsuite

Information about the Robot Based Testsuite is available at:

<https://twiki.cern.ch/twiki/bin/view/EMI/CREAMRobotFuncTests>

Old Testsuite

The old CREAM testsuite is a set of python scripts that interacts with the CREAM command line tools in order to perform several functional tests.

Installation and setup

The easiest way to deploy the old testsuite is to have a UI node already available.

The old testsuite can then be installed using yum creating a file called testsuites.repo in the yum.repos.d directory with the following definition:


```
[ETICS-name-CREAM-service-testsuite]
name=ETICS Repository of CREAM-service-testsuite
baseurl=http://etics-repository.cern.ch:8080/repository/pm/registered/repomd/name/CREAM-service-t
protect=0
enabled=1
gpgcheck=0
```

and then running

```
yum update
yum install glite-testsuites-cream
```

After the installation it is suggested to check if the environment variable **GLITE_LOCATION** points to the correct gLite installation directory, i.e. the one containing the bin folder with the CREAM clients.

Old Testsuite and ETICS

The old testsuite can be downloaded and build directly from the ETICS framework. For any details concerning the ETICS client installation refer to the User Guide or to the twiki site .

The following list of commands is required in order to retrieve and build the latest stable release:

```
etics-workspace-setup
etics-get-project org.glite
etics-checkout --config glite-testsuites-cream_R_1_0_7_0 --project-config glite_branch_3_1_0 org.
etics-build org.glite.testsuites.cream
```

Releases and Changelog

- 1.0.3.0 (2009-02-24):
 - ◆ Fixed bug: bad termination condition in cream-monitored-* tests with failures during the submission
- 1.0.4.0 (2009-03-05):
 - ◆ Changed command line of the CE monitor client, this version can be used only with the clients whose version is 1.11 or higher
- 1.0.5.0 (2009-04-17):
 - ◆ Fixed minor issues
 - ◆ New options "nopurge" and "sotimeout" are available
- 1.0.6.0 (2009-05-26):
 - ◆ Fixed man pages in rpm
 - ◆ Fixed minor issues
- 1.0.7.0 (2009-03-02):
 - ◆ Support for event query

Old Testsuite description

The set of test can be divided into two main categories, according to the type of status detection mechanism for a job:

- with polling (synchronous):
 - ◆ cream-test-monitored-cancel
 - ◆ cream-test-monitored-lease-expired
 - ◆ cream-test-monitored-lease-updated
 - ◆ cream-test-monitored-proxy-expired
 - ◆ cream-test-monitored-submit
- with notifications (asynchronous):
 - ◆ cream-test-notified-cancel

- ◆ cream-test-notified-lease-expired
- ◆ cream-test-notified-lease-updated
- ◆ cream-test-notified-submit

Proxy management

Each test can be run using either an external user voms-proxy or with an internal management of the voms-proxy.

The default mechanism is using an external voms-proxy. The external voms-proxy is not renewed by the test, the user is responsible for updating that credential if required in such a way that it does not interfere with tests, for example "renaming" the new proxy into the old one.

The external voms-proxy path is defined with the environment variable **X509_USER_PROXY**, the main default is /tmp/x509_uuid.

With the internal management of the voms-proxy the testsuite keeps track of the renewal of that credential, the user has just to provide a valid personal certificate and private key. The environment variables required for enabling this mode are: **X509_USER_CERT** and **X509_USER_KEY**; when this mode is selected each test requires the definition of the VO to be used (option **--vo**) and the voms-proxy-init client must be properly configured.

Delegation management

Each test can be run either using a single delegated proxy on the CE for all the submissions or with one delegated proxy per submitted job, see **--delegationType** option.

The tester must be aware of using the second mode can overload the entire system, both the service and the testsuite, so the tuning of the test parameters must consider that issue.

Isolation and interference tests

One of the basic requirements for a test is to be isolated; each test handles only its own jobs and resources in general (delegated proxies, lease tokens, etc.) with no interferences from other tests.

Multiple tests can be executed in parallel in order to detect interferences on the service side; for example the *cream-test-monitored-lease-expired* test can be run together with *cream-test-monitored-submit* for discovering a wrong lease management for the jobs submitted by the latter.

Hints for test tuning

- Due to unavoidable delays in the command scheduling, the life-time of the job submitted with the *cream-test-notified-cancel* has to be at least 3 times longer than the rate parameter. With shorter life-time the job can terminate before the testsuite is able to detect the change of its status.
- In order to test the internal proxy renewal mechanism of the CREAM service it's strongly suggested to run at least one test with notifications and proxy renewal all alone, with no other tests running on the same CE.
- It is strongly suggested to configure the socket timeout according to the latency of the network using the **--sockettimeout** option (available since version 1.0.5)

Proposed tests

The set of tests described in this section is used for the pre-certification of the latest version of the CREAM CE and can be considered a template for external testers. In the following the CREAM CE under test is *cream-12.pd.infn.it* (port 8443) with Torque as batch system and *cream_A* as assigned queue, the hostname

CreamTestWorkPlan < CREAM < TWiki

and port must be replaced with the correct values according to the arrangement of the testbed; the VO dteam is used.

The simple.jdl contains:

```
[ executable="/bin/sleep"; arguments="180";]
```

The long.jdl contains:

```
[ executable="/bin/sleep"; arguments="1800";]
```

The onehoursleep contains:<verbatim>[executable="/bin/sleep"; arguments="3600";]</verbatim>

The testsandbox.jdl contains:

```
[
executable="/bin/cp";
arguments="grid-mapfile out-`date +%s%N`-`whoami`.txt";
inputsandbox={
  "gsiftp://cream-12.pd.infn.it/etc/grid-security/grid-mapfile"
};
outputsandbox={
  "out*.txt"
};
outputsandboxbasedesturi="gsiftp://cream-12.pd.infn.it/tmp";
epilogue="/bin/sleep";
epiloguearguments="60";
]
```

The log4py.conf contains:

```
[Default]
Format: %T %L %C [%F] - %M
Target: cream-test-cream-12.log
Ansi: False
LogLevel: Debug
```

Simple submission with polling
cream-test-monitored-submit -r 30 -n 1000 -m 50 -C 50 -l log4py.conf -j simple.jdl -R cream-12.pd.infn.it
The renewal process is disabled, the variable <i>X509_USER_PROXY</i> must point to a 12-hours long proxy
Simple submission with notification
cream-test-notified-submit -r 30 -n 1000 -m 50 -C 50 -p 9000 -l log4py.conf -j simple.jdl -R cream-12.pd.infn.it
The renewal process is disabled, the variable <i>X509_USER_PROXY</i> must point to a 12-hours long proxy
Submission and cancellation with polling
cream-test-monitored-cancel -r 30 -n 1000 -m 50 -C 50 -l log4py.conf -j simple.jdl -R cream-12.pd.infn.it
The renewal process is disabled, the variable <i>X509_USER_PROXY</i> must point to a 12-hours long proxy.
Submission and cancellation with notification
cream-test-notified-cancel -r 30 -n 1000 -m 50 -C 50 -p 9000 -l log4py.conf -j simple.jdl -R cream-12.pd.infn.it
The renewal process is disabled, the variable <i>X509_USER_PROXY</i> must point to a 12-hours long proxy.
Submission with polling and sandbox
cream-test-monitored-submit -r 30 -n 1000 -m 50 -C 50 -l log4py.conf -j testsandbox.jdl -R cream-12.pd.infn.it
The renewal process is disabled, the variable <i>X509_USER_PROXY</i> must point to a 12-hours long proxy. The output sandbox is used
Submission with notifications and sandbox
cream-test-notified-submit -r 30 -n 1000 -m 50 -C 50 -p 9000 -l log4py.conf -j testsandbox.jdl -R cream-12.pd.infn.it
The renewal process is disabled, the variable <i>X509_USER_PROXY</i> must point to a 12-hours long proxy. The output sandbox is used
Simple submission with polling and proxy
cream-test-monitored-submit -r 60 -n 50 -m 10 -C 10 -l log4py.conf -j long.jdl --vo dteam --valid 00:20
The variables <i>X509_USER_CERT</i> and <i>X509_USER_KEY</i> must point to the location of the user certificate and private key
Simple submission with notifications and proxy
cream-test-notified-submit -r 60 -n 50 -m 10 -C 10 -p 9000 -l log4py.conf -j long.jdl --vo dteam --valid 00:20
The variables <i>X509_USER_CERT</i> and <i>X509_USER_KEY</i> must point to the location of the user certificate and private key
Simple submission with polling and proxy
cream-test-monitored-proxy-expired -r 60 -n 50 -m 10 -C 10 -l log4py.conf --vo dteam --valid 00:20
The variables <i>X509_USER_CERT</i> and <i>X509_USER_KEY</i> must point to the location of the user certificate and private key
Simple submission with polling and lease
cream-test-monitored-lease-expired -r 60 -n 50 -m 10 -C 10 -l log4py.conf -W 1200 -R cream-12.pd.infn.it
The renewal process is disabled, the variable <i>X509_USER_PROXY</i> must point to a 12-hours long proxy. One lease token is used
Simple submission with notifications and lease
cream-test-notified-lease-expired -r 60 -n 50 -m 10 -C 10 -l log4py.conf -W 1200 -p 9000 -R cream-12.pd.infn.it
The renewal process is disabled, the variable <i>X509_USER_PROXY</i> must point to a 12-hours long proxy. One lease token is used

Simple submission with polling, lease expired a
<code>cream-test-monitored-lease-expired -r 60 -n 50 -m 10 -C 10 -l log4py.conf -W 1800 --vo dteam --va</code>
The variables <i>X509_USER_CERT</i> and <i>X509_USER_KEY</i> must point to the location of the user certificate and private k
Simple submission with notifications, lease expired
<code>cream-test-notified-lease-expired -r 60 -n 50 -m 10 -C 10 -l log4py.conf -W 1800 -p 9000 --vo dte</code>
The variables <i>X509_USER_CERT</i> and <i>X509_USER_KEY</i> must point to the location of the user certificate and private k
Simple submission with polling and lease
<code>cream-test-monitored-lease-updated -r 60 -n 50 -m 10 -C 10 -l log4py.conf -W 1200 -R cream-12.pd.</code>
The renewal process is disabled, the variable <i>X509_USER_PROXY</i> must point to a 12-hours long proxy. One lease toke
Simple submission with notifications and le
<code>cream-test-notified-lease-updated -r 60 -n 50 -m 10 -C 10 -l log4py.conf -W 1200 -p 9000 -R cream</code>
The renewal process is disabled, the variable <i>X509_USER_PROXY</i> must point to a 12-hours long proxy. One lease toke
Simple submission with polling, lease updated a
<code>cream-test-monitored-lease-updated -r 60 -n 50 -m 10 -C 10 -l log4py.conf -W 1800 --vo dteam --va</code>
The variables <i>X509_USER_CERT</i> and <i>X509_USER_KEY</i> must point to the location of the user certificate and private k
Simple submission with notifications, lease updated
<code>cream-test-notified-lease-updated -r 60 -n 50 -m 10 -C 10 -l log4py.conf -W 1800 -p 9000 --vo dte</code>
The variables <i>X509_USER_CERT</i> and <i>X509_USER_KEY</i> must point to the location of the user certificate and private k
Proxy renewal for job in IDLE
<code>cream-test-monitored-submit -r 60 -n \$TOTAL -m \$PART -C \$PART -l log4py.conf -j onehoursleep.jdl</code>
The variables <i>X509_USER_CERT</i> and <i>X509_USER_KEY</i> must point to the location of the user certificate and private k of jobs which can be run concurrently in the batch system, the variable \$TOTAL is a multiple of \$PART.

This topic: CREAM > CreamTestWorkPlan

Topic revision: r31 - 2013-10-09 - TWikiAdminUser



Copyright © 2008-2023 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.

Ideas, requests, problems regarding TWiki? Send feedback