Sara Grid Service User Guide

Author	: Bart Heupers
Date	: October 13, 2005
SaraGridService version	: 0.2 alpha

Introduction

The purpose of the **SARA Grid SOAP Service** is to make the LCG Grid accessible from any computing environment which is able to do SOAP calls. These include programming languages such as C, Java and Perl, Python on UNIX and Windows platforms, Workflow systems such as Kepler.

With the SARA Grid SOAP server it is no longer required to log in to a LCG User Interface machine to submit jobs.

The **SARA Grid SOAP Service** maps the LCG commands which are normally required to start and manage jobs available as a SOAP service.

For the following major LCG commands for managing jobs a corresponding SOAP operation is defined:

LCG command	**SOAP Service**
grid-proxy-init	proxy_init
grid-proxy_info	proxy_info
grid-proxy-destroy	proxy_destroy
edg-job-submit	job_submit
edg-job-list-match	job_list_match
edg-job-status	job_status
edg-job-get-output	job_get_output
edg-job-cancel	job_cancel
	job_list
	get_version

The SARA Grid SOAP Service is available via:

[https://mu7.matrix.sara.nl:443/SaraGridServiceLCG]

(Old version was [https://mu7.matrix.sara.nl:443/SaraGridServiceLCG])

A simple Python program to retrieve the version is:

```
from SOAPpy import SOAPProxy
server = SOAPProxy('https://mu7.matrix.sara.nl:443/SaraGridServiceLCG',
                    namespace='urn:SaraGrid')
print server.get_version()
```

A WSDL description of this service, which can be used to access these services in a generic way, is available in:

[http://mu7.matrix.sara.nl/SaraGridService/SaraGridServiceLCG.wsdl]

Example

A typical session involves the following steps:

1) Locally generate a proxy from your certificate. This can usually be done with the Globus command grid-proxy-init or same command of the Java COG kit.
[http://www.cogkit.org]
proxy can also be generated with the **GenProxy.pl** script.

2) Transfer the proxy to the SOAP Server. This is being done with the **proxy_init** service. The return value of the **proxy_init** service is an identifier which is required for any subsequent steps which need the proxy certificate.

```
result = server.proxy_init(proxy)
if result['status'] == 'OK':
    identity = result['value']
```

3) Submit a job. A job is specified as a JDL-file (Job Description Language, see [http://server11.infn.it/workload-grid/docs/DataGrid-01-TEN-0102-0_2-Document.pdf])
A job also requires a VO (Virtual Organization) and it is optionally possible to specify on which CE (Compute Element) the job must be submitted.

If there are files which are required for starting the job they need to be transmitted. They must be specified in the InputSandbox in the JDL-file, and they should also be base-64 encoded as strings and added to the job which is submitted to the server. The job_submit service returns a job_id which is required to retrieve the status or output of the job.

```
job['files'] = files
job['jdl'] = jdl
job['vo'] = vo
job['ce'] = ce

result = server.job_submit(identity, job)
if result['status'] == 'OK':
    job_id = result['value']
```

4) Get the status of the job. The status can be Waiting, Scheduled, Running, Done or Aborted etc

If the job is Aborted we also would like to know why the job was aborted on the client. Therefore the job_status service returns the entire output of the edg-job-status command.

The actual status itself needs to be extracted out of this text like :

```
result = server.job_status(identity, job_id)
if result['status'] == 'OK':
    output = result['value']
    p = re.compile('Current Status: *([^ ]*) ')
    m = p.search(output)
    if m:
        status = m.group(1)
    else:
       status ='Unknown'
```

Repeat this until the job status is ‘Done’ or ‘Abort’

5) Retrieve the output of the job. Any output files which were specified in the OutputSandbox in the JDL-file are being base-64 encode transferred.

```
result = server.job_get_output(identity, job_id)
if result['status'] == 'OK':
    files = result['files']
    for file in files:
        data = file['data'].decode('base64')
        print file['name'], '\n', data, '\n\n'
```

After the output for a job is retrieved any information about that job is deleted.

6) Destroy the proxy. This will delete the proxy on the server and all information about jobs that were started with this proxy.

```
server.proxy_destroy(identity)
```

Reference

Detailed description of the various operations for this service are below.

get_version

Returns a version for the SARA Grid Service. Use to test communication or to test version. The version described here is **0.2 alpha**

No arguments

Returns

version : string

proxy_init

This operation takes a proxy certificate and copies it to the Grid SOAP Service so that it can be used for subsequent operations. For every user (identity in certificate) the proxy is stored in a different place.

Arguments

proxy : string containing the entire proxy certificate

Returns

status : OK or FAILED value : identity of the certificate of OK otherwise error message

identity : random string that needs to be given in subsequent operations to use the proxy certificate

proxy_info

Returns information about the proxy that is currently stored for the identity of the user.

Arguments

identity: string with the identity of the user (return value of proxy_init)

Returns

status : OK or FAILED value : string containing all the proxy details if OK or errormessage if FAILED.

proxy_destroy

Removes the certificate proxy for the given identity from the server. Also remove any remaining files for jobs for that identity.

Arguments

identity : string with the identity of the user (return value of proxy_init)

Returns

status : OK or FAILED value : empty or errormessage if FAILED.

job_submit

Submit a job on the server.

Arguments

identity : string with the identity of the user (return value of proxy_init) job : string with complete JDL description, VO and optionally a CE. files : list of filename, filedata tuples where the filedata is the base64 encoded contents of the file

Returns

status : OK or FAILED value : job_id or errormessage if FAILED.

job_list_match

identity : string with the identity of the user (return value of proxy_init) job : string with complete JDL description

Returns

status : OK or FAILED value : list of CE's where this job for selected VO can be executed or errormessage if FAILED.

job_status

Retrieve the status of a job which was earlier submitted to the grid.

Arguments

identity : string with the identity of the user (return value of proxy_init) job_id : string with job_id (return value of job_submit)

Returns

status : OK or FAILED value : The output of the edg-job-status command. This contains the string **Current Status: STATUS** where status can be 'Scheduled', 'Running', 'Done', 'Cleared' etc. Or errormessage if FAILED.

job_cancel

Cancel the job which was earlier submitted to the grid.

Arguments

identity : string with the identity of the user (return value of proxy_init) job_id : string with job_id (return value of job_submit)

Returns

status : OK or FAILED value : The output of the edg-job-cancel command. Or errormessage if FAILED.

job_list

List all the jobs that are curently busy for the selected proxy.

Arguments

identity : string with the identity of the user (return value of proxy_init)

Returns

status : OK or FAILED value : List of all job identifiers that are busy. Or errormessage if FAILED.

job_get_output

Retrieve output from a job that is finished

Arguments

identity : string with the identity of the user (return value of proxy_init) job_id : string with job_id (return value of job_submit)

Returns

status : OK or FAILED value : empty or errormessage if FAILED. files : list of filename, filedata tuples where the filedata is the base64 encoded contents of the file

FAQ

- To disable certificate checking from Java (Taverna, Axis) SSL clients add to the mygrid.properties file : axis.socketSecureFactory = org.apache.axis.components.net.SunFakeTrustSocketFactory

- For LCG job submission a 'full legacy globus proxy' is required. To generate such a proxy with the Java COG kit (1.2 or later ??) : grid-proxy-init -old

- On Windows don't use ActiveState's Python for the client because SSL is not enable for ActiveState's Python