CONTRACT NUMBER 508830

DEISA
DISTRIBUTED EUROPEAN INFRASTRUCTURE FOR SUPERCOMPUTING APPLICATIONS

European Community Sixth Framework Programme
RESEARCH INFRASTRUCTURES
Integrated Infrastructure Initiative

DESHL v2.0

Deliverable ID: DEISA-DJRA7-3.8
Due date: October 31st, 2005
Actual delivery date: November 25, 2005
Lead contractor for this deliverable: EPCC, UK

Project start date: May 1st, 2004
Duration: 5 years

Project co-funded by the European Commission within the Sixth Framework Programme (2002-2006)		
Dissemination Level		
PU	Public	**X**
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Table of Contents

1. Introduction

1.1 Executive Summary

The DESHL (**DE**ISA **S**ervices for the **H**eterogeneous management **L**ayer) has been developed by the DEISA Joint Research Activity JRA7 [1]. It provides standards-based access for users and their applications to manage jobs and transfer files in the DEISA heterogeneous supercomputing infrastructure.

This document, "DESHL v2.0", is deliverable DEISA-DJRA7-3.8 from Task T3.6 "Implementation of DESHL release 2.0" in Work Package 3 of the DEISA JRA7. This document reports on the release version 2.0 of the DESHL. This release contains the functionality for data staging and job management, designed using a layered stack with the UNICORE ARCON client as its base.

1.2 Document Structure

Section 1 of this document contains the executive summary, references, table of acronyms, etc. Section 2 indicates where background information on DESHL v2.0 can be obtained, with Sections 3 briefly describing the contents of DESHL v2.0. Section 4 indicates where the DESHL software can be obtained from and finally section 5 outlines the known issues with DESHL v2.0.

1.3 References and Applicable Documents

This document uses UML 1.0 notation for the UML [2] diagrams.

[1] DEISA Annex I – "Description of Work", November 5th 2003.

[2] UML, http://www.uml.org/

[3] "JRA7 Quality Plan v2.0", DEISA JRA7 Report, Deliverable ID DEISA-JRA7-1.2, May 3rd 2005.

[4] "Functional scope for DESHL v2.0", DEISA JRA7 Report, Deliverable ID DEISA-JRA7-3.6, June 28th 2005.

[5] "Final Design for DESHL v2.0", DEISA JRA7 Report, Deliverable ID DEISA-JRA7-3.7, November 2005.

[6] UNICORE, http://unicore.sourceforge.net

[7] SAGA GGF, https://forge.gridforum.org/projects/saga-rg/

1.4 Document Amendment Procedure

The document procedure is covered in the Quality Plan [3].

1.5 List of Acronyms and Abbreviations

API   Application Programming Interface

CLT	Command Line Tool
DEISA	Distributed European Infrastructure for Supercomputing Applications
DESHL	DEISA Services for the Heterogeneous management Layer
GGF	Global Grid Forum
HPC	High Performance Computing
JRA	DEISA Joint Research Activity
JRA7	Seventh Joint Research Activity
SAGA	Simple API for Grid Applications
UML	Unified Modelling Language
UNICORE	Uniform Interface to Computing Resources

2.   Background

The functional scope for DESHL v2.0 can be found in the DEISA JRA7 deliverable "Functional scope for DESHL v2.0" [4]. The design is presented in DEISA JRA7 deliverable "Final Design for DESHL v2.0" [5].

3.   DESHL v2.0 Overview

DESHL aims to provide standards-based access to DEISA resources. In the extended heterogeneous DEISA infrastructure, sites have different HPC resources but all sites provide access to those resources via UNICORE [6].

DESHL release v2.0 contains the DESHL client, that is implemented as a layered stack with a SAGA-inspired API [7] at its top and the UNICORE ARCON client at its base. The DESHL client can be used for data staging operations to, from and within the DEISA environment, and to submit, monitor and terminate jobs running on DEISA resources. Issuing a certificate for access to a DEISA site is the task of the certificate authority associated with the user's national location. These certificates are then held locally, and the DESHL client is configured via a single configuration file to allow seamless access to all such configured sites.

For more detailed information on the DESHL v2.0 architecture and the functionality available in this release please refer to the DESHL v2.0 Final Design document [5].

4.   Configuration

Note that that the deshl script needs to be configured for the user's environment. The release includes an example log4j configuration that the script variables can point to.

4.1   Client site configuration file

The client configuration file is a comma-separated-variable file, which contains a list of the DEISA sites that the user wishes to access and the associated security credentials.

ssl://host1.epcc.ed.ac.uk:4010/SITE_A	C:/deisa/cert_a.p12	<password>
ssl://host2.epcc.ed.ac.uk:4011/SITE_B	C:/deisa/cert_b.p12	<password>

Table 1: Example client configuration file

The above table shows an example configuration file for accessing two separate sites. Each line in the configuration file corresponds to access to a DESHL site (i.e. to a UNICORE VSite). Each line must therefore contain the full URL to the NJS, the location of the user's certificate for accessing that site and the password for the certificate.
In the first example in Table 1, the entry is composed of:
- Gateway host port: host1.epcc.ed.ac.uk:4011
- The name of the NJS: SITE_B

Note that the client also assumes that the CA certificate for the user certificate is in the same directory as the user certificate**.**

4.2   Typical configuration

A typical configuration file for a DESHL user will consist of an entry for a VSite considered as that user's home site within DEISA. Additional entries will be present for the sites that the user wishes to access for data staging functionality. Note that the level of logging is configurable by the user; however, this is intended for development and debugging by the DESHL developers.

5.   Examples of use

The DESHL client is invoked by executing a script file with appropriate arguments, which in turn invokes an instance of the Java virtual machine. Different versions of the script exist for Unix and Windows. Some examples of use of the DESHL client are shown below using the DESHL Windows script, deshl.bat

Note that for data staging commands, a convention for filenames must be followed. For each site configured in the DESHL configuration file, files and directories must be prefixed with the appropriate VSite URL and "/home". For example, consider a VSite configured as ssl://deisa.epcc.ed.ac.uk:4010/DESHL. To get a listing of the top level directory, the following command would be required:

deshl list ssl://deisa.epcc.ed.ac.uk:4010/DESHL/home

5.1   Listing commands and help

The list of supported commands can be displayed by running deshl –h

```
C:\libs>deshl -h

C:\libs>ECHO OFF
-h
[-h] command [command-options]
Where:
  -h,--help    Print this message.

With commands:
  fetch
  copy
  isDir
  move
  status
  jobs
  remove
  makeDir
  list
  submit
  sites
  terminate
  isFile
  exists

C:\libs>_
```

5.2   Submitting a job

Job scripts are submitted using the submit command:

```
C:\libs>
C:\libs>deshl submit sleeper.sh

C:\libs>ECHO OFF
submit
ajo id: org.unicore.AJOIdentifier@80878ab1
Your job: x2138600783ssl%3A%2F%2Fdeisa1.epcc.ed.ac.uk%3A4010%2FTEST_EPCC_NJS_A,
has been successfully submitted.
C:\libs>
```

5.3   Retrieving a job's status

The status of a submitted job can be obtained by using the status command, specifying the job identifier returned from the submit command:

```
C:\libs>
C:\libs>deshl status x2138600774ssl%3A%2F%2Fdeisa1.epcc.ed.ac.uk%3A4010%2FTEST_E
PCC_NJS_A

C:\libs>ECHO OFF
status
Job: x2138600774ssl%3A%2F%2Fdeisa1.epcc.ed.ac.uk%3A4010%2FTEST_EPCC_NJS_A, has s
tatus: DoneOk
C:\libs>
C:\libs>
```

5.4   Listing jobs

The list of current jobs can be displayed by using the jobs command:

```
C:\libs>
C:\libs>deshl jobs

C:\libs>ECHO OFF
jobs
x2138600774ssl%3A%2F%2Fdeisa1.epcc.ed.ac.uk%3A4010%2FTEST_EPCC_NJS_A
C:\libs>_
```

5.5   Fetching a job's outcome

The stdout and stderr for a job can be retrieved into the current directory by running the fetch command with the appropriate job identifier as before. This also frees up the resources for a job, and removes it from the list of jobs. (The output from the fetch command is verbose and not listed here.)

5.6   Terminating a job

Jobs may be terminated via the terminate command. Once a job has been terminated, it is removed from the job list and its output may no longer be obtained.

```
C:\libs>
C:\libs>deshl terminate 1008844443ssl%3A%2F%2Fdeisa1.epcc.ed.ac.uk%3A4010%2FTEST
_EPCC_NJS_A

C:\libs>ECHO OFF
terminate
Job: 1008844443ssl%3A%2F%2Fdeisa1.epcc.ed.ac.uk%3A4010%2FTEST_EPCC_NJS_A, termin
ated.
C:\libs>_
```

5.7   Listing a DESHL directory

DESHL directories may be listed via the list command:

```
C:\libs>
C:\libs>deshl list ssl://deisa1.epcc.ed.ac.uk:4010/TEST_EPCC_NJS_A/home/unicore-
quick

C:\libs>ECHO OFF
list
/unicore-quick/readme
/unicore-quick/quick-1.1.0-A/
/unicore-quick/unicore_demo_1_1.tar.gz
/unicore-quick/quick-1.1.0-B/
C:\libs>_
```

5.8   Copying files

Files may be imported to or exported from DESHL, and between sites in DESHL.

```
C:\libs>
C:\libs>deshl copy C:/testdata.dat ssl://deisa1.epcc.ed.ac.uk:4010/TEST_EPCC_NJS
_A/home/importdata.dat

C:\libs>ECHO OFF
copy
ajo id: org.unicore.AJOIdentifier@a44c92d7
C:\libs>
C:\libs>_
```

The existence of the imported file can be verified by running the exists command.

```
C:\libs>
C:\libs>deshl exists ssl://deisa1.epcc.ed.ac.uk:4010/TEST_EPCC_NJS_A/home/import
data.dat

C:\libs>ECHO OFF
exists
exists: target ssl://deisa1.epcc.ed.ac.uk:4010/TEST_EPCC_NJS_A/home/importdata.d
at exists
C:\libs>_
```

5.9   Example log file

The following shows the contents of a log file for a jobs command:

```
2005-11-21 15:21:37,262
org.deisa.jra7.deshl.ui.clt.command.DESHLCommandFactory#initCommands
INFO: No config file set using
org.deisa.jra7.deshl.ui.clt.command.DESHLCommandFactory.config.file
2005-11-21 15:21:37,324 de.fzj.roctopus.unicore.UnicoreSite#getAllJobs
INFO: Creating new GetJobs request...
2005-11-21 15:21:37,902 de.fzj.roctopus.unicore.UnicoreSite#getAllJobs
INFO: Getting jobs from TEST_EPCC_NJS_A
2005-11-21 15:21:38,527 de.fzj.roctopus.unicore.UnicoreSite#getAllJobs
INFO: Creating new GetJobs request...
2005-11-21 15:21:38,574 de.fzj.roctopus.unicore.UnicoreSite#getAllJobs
INFO: Getting jobs from TEST_EPCC_NJS_B
```

6.   Availability of DESHL v2.0

DESHL v2.0 is a public release and will be available from the DEISA JRA7 NeSCForge development site at http://forge.nesc.ac.uk/projects/deisa-jra7/.

The release is contained in a zip archive that contains the following files:
- `deshl-intf-1.0.jar` – the SAGA interface definitions
- `deshl-impl-1.0.jar` – the implementation of the SAGA interfaces to the grid interface library
- `DESHL-CLT-1.2.jar` – the DESHL command line tool
- `roctopus.jar` – the grid access library
- `deshl.bat` – a Windows script for running the DESHL command line tool
- `deshl` – a Unix script for running the DESHL command line tool
- `config.csv` – an example configuration file for the DESHL client
- `sleeper.sh` – a simple example of a job script
- a set of 3rd-party libraries required for the DESHL client

7. Known Issues

At the time of writing, the following issues are known.

7.1   User Documentation
User and installation documentation is currently very sparse. Users are directed towards the design documentation for more information on use of functionality. Further documentation will be produced in due course.

7.2   File Export
Export of directories from DESHL to local storage is not currently supported. However, export of individual files is currently supported.

7.3   Copying directories between UNICORE Sites

There is currently believed to be an issue copying directories between UNICORE sites. This has been successfully demonstrated but some further testing is required to validate it.

7.4   Configuration Security

Currently the DESHL client is configured locally via a comma-separated-variable text file, where an entry is required for each site that the user wishes to access. Each entry defines the name of the site, a path to a valid certificate for that site and a password for the certificate. As passwords are stored unencrypted, this is potentially a security risk in the case of the user's account being compromised or the configuration file being copied to an insecure location. It is recommended that users encrypt the configuration file when the DESHL client is not being used. A more secure solution to this issue will be implemented in a subsequent release of the DESHL.