



EGI-Engage

Production level gateway for Arts and Humanities

D6.17

Date	15 January 2017
Activity	WP6
Lead Partner	MTA SZTAKI
Document Status	FINAL
Document Link	https://documents.egi.eu/document/3021

Abstract

This deliverable describes the production-level science gateway created for the DARIAH community inside the DARIAH Competence Centre (CC) of the EGI-Engage project. The achievements behind this deliverable are the followings: (1) dedicated Virtual Organization (VO) inside the EGI-Engage infrastructure for the DARIAH CC, (2) production-level science gateway with portlets providing useful applications for the DARIAH community, (3) federated login for accessing the science gateway for users of eduGAIN member institutes.



This material by Parties of the EGI-Engage Consortium is licensed under a [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/).

The EGI-Engage project is co-funded by the European Union (EU) Horizon 2020 program under Grant number 654142 <http://go.egi.eu/eng>

COPYRIGHT NOTICE



This work by Parties of the EGI-Engage Consortium is licensed under a Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>). The EGI-Engage project is co-funded by the European Union Horizon 2020 programme under grant number 654142.

DELIVERY SLIP

	<i>Name</i>	<i>Partner/Activity</i>	<i>Date</i>
From:	Zoltan Farkas	MTA SZTAKI	15/01/2017
Moderated by:	Malgorzata Krakowian	EGI Foundation/WP1	
Reviewed by	Lars Wieneke	University of Lundt	20/02/2017
	Roberto Scopigno	CNR	18/02/2017
	David Hübner	DAASI/DARIAH	20/02/2017
	Martin Haase	DAASI/DARIAH	27/02/2017
Approved by:	AMB and PMB		7/03/2017

DOCUMENT LOG

<i>Issue</i>	<i>Date</i>	<i>Comment</i>	<i>Author/Partner</i>
v0.1	10/01/2017	First Draft	Z. Farkas - MTA SZTAKI
v1.0	01/03/2017	Updated based on reviews	Z. Farkas - MTA SZTAKI
FINAL	7/03/2017	FINAL version	

TERMINOLOGY

A complete project glossary and acronyms are provided at the following pages:

- <https://wiki.egi.eu/wiki/Glossary>
- <https://wiki.egi.eu/wiki/Acronyms>

Contents

1	Introduction.....	5
2	Service architecture	8
1.1	DARIAH CC Virtual Organisation	9
1.2	DARIAH CC Portal	10
1.1.1	eduGAIN login module.....	12
1.1.2	Application portlets	13
1.1.3	Workflow development portlets.....	13
1.1.4	ASM API.....	16
1.1.5	Cloud access portlet	16
1.1.6	File transfer portlet.....	19
3	Release notes.....	20
3.1	Requirements covered in the release	20
4	Feedback on satisfaction	20
5	Plan for Exploitation and Dissemination	21
6	Future plans.....	24
	Appendix I - shibboleth2.xml config file.....	25
	Appendix II. - httpd ssl configuration file	27
	Appendix III - ASMService methods	28

Executive summary

This deliverable details the work performed in order to achieve a production-level (continuously operated and maintained) gateway service for the DARIAH arts and humanities (A&H) community, and also presents the tools offered. The gateway is available at <https://dariah-gateway.lpds.sztaki.hu/>. The following main topics are discussed by the deliverable all of which are necessary to have a production-level science gateway: (1) Virtual Organization for the DARIAH Competence Center, (2) science gateway deployed with portlets, (3) federated login onto the portal.

The DARIAH CC VO allows the CC to collect compute and storage resources for the DARIAH community, taking available resources from partners of the EGI-Engage project, based on the technology provided by EGI-Engage. This base technology platform allows the CC to collect resources using different protocols to exploit their compute and storage capabilities (like gLite, OCCI, EC2, etc.), which makes it easy to integrate further resources in the future. Currently, the DARIAH CC VO contains cloud resources from two sites, RECAS-BARI and INFN-CATANIA-STACK, accessing them through the sites' OCCI interface.

The aim of the science gateway is to provide convenient web-based access to the resources available in the DARIAH CC VO. For this, it hides the details of accessing the compute and storage resources, and offers user-friendly portlets for running tasks related to the arts and humanities community. The portal currently offers three applications (Simple Semantic Search Engine, Parallel Semantic Search Engine, DBO@Cloud) and three services (Cloud Access, Workflow Development, File Transfer). The gateway provided by the DARIAH CC is based on WS-PGRADE/gUSE.

The federated login for the gateway enables researchers of eduGAIN member institutes to access the gateway by using their home institute credentials, thus there is no need to maintain a separate login for the gateway. The DARIAH community is also participating in eduGAIN with their own Virtual Organization (DARIAH VO - not to confuse with the DARIAH CC VO), thus the federated login implemented allows easy access for the DARIAH community members as well. This federated login is implemented based on Shibboleth, by using an additional login module for the gateway.

The services offered by the DARIAH CC in order to provide the production-level gateway for the DARIAH A&H community are continuously monitored and operated by MTA SZTAKI and INFN.

The outline of this deliverable is the following: first, we introduce the base technologies this deliverable relies on, and also show the high-level architecture which allows providing a production-level science gateway. Next, we present the setup and operation of the DARIAH CC VO. This is followed by describing the details of the gateway itself. Afterwards, we show the details of the federated login implementation. Finally, we conclude the work performed in order to achieve the results described in the deliverable.

1 Introduction

Scientific communities have access to a diverse set of infrastructures to perform computations related to their research. These infrastructures can be either local or global from the community's point of view. Local infrastructures usually represent some sort of small-scale computing cluster (for example using PBS or LSF) that a small-sized community (for example a university department) has direct and sometimes exclusive access to, whereas global infrastructures usually represent a large amount of geographically distributed compute resources based on some sort of grid middleware (like gLite, ARC, Globus or BOINC), with access offered to a large number of different communities. As it can be seen, both local and global infrastructures have their advantages and disadvantages. Unfortunately, using these kinds of infrastructures might be a difficult task for end-user scientists. In order to eliminate the requirement of in-depth knowledge to use different compute infrastructures, one may use science gateways as a user-friendly, easy-to-use web interface that enables end-user scientists to run their experiments quickly and without learning the particular features of the distributed infrastructure. The DARIAH CC's science gateway's base technology, WS-PGRADE/gUSE offers a complete workflow-based framework with different customization methods (all of them applying the core framework without any modification) to create easy-to-use science gateways.

The WS-PGRADE/gUSE gateway technology is based on the Liferay portlet container framework. This allows science gateway developers to create user-friendly portlets easily. Together with the customization methodologies of WS-PGRADE/gUSE, science gateway developers can create user-friendly portlets for different scientific communities which help to exploit the available compute and storage resources easily. MTA SZTAKI provides an API for creating portlets which are using the services of WS-PGRADE/gUSE, called the Application-Specific Module (ASM API). This API enables science gateway developers to call services of WS-PGRADE/gUSE for importing, modifying and running existing workflows.

A tool based on WS-PGRADE/gUSE and ASM API for easy execution of parameter-sweep application is called the Parameter-sweep job wizard. This wizard enables users to run their application processing large input data step by step following six simple steps: executable upload, static input upload, parameter-sweep input upload, command-line argument definition, resource selection, and definition of output files.

Finally, Data Avenue is a set of services enabling users to manage their data located on remote storage resources easily. The set of available operations includes browsing, directory creation/renaming/removal, file up- and download, file management (rename/remove/move) and file transfer between different types of storages. For developers, an easy-to-use API is available, which allows to perform these operations easily, but for end-users a two-panel, graphical interface is presented to perform these operations.

Thus, the core technology behind the DARIAH CC portal is based on the following components:

- WS-PGRADE/gUSE science gateway framework
- Application-Specific Module (ASM)
- Parameter-sweep job wizard
- Data Avenue.

The details of these tools are as follows:

Tool name	<i>WS-PGRADE/gUSE science gateway framework</i>
Tool url	Main webpage: http://guse.hu/ Source page: https://sourceforge.net/projects/guse/
Tool wiki page	http://guse.hu/documentation/how-to
Description	<i>WS-PGRADE/gUSE offers a customizable science gateway framework enabling the development and execution of scientific workflows, exploiting different compute (gLite, ARC, UNICORE, OCCl, AWS, ...) and storage (S3, LFC, SRM, ...) resources through a simple user interface.</i>
Value proposition	<i>The gateway eliminates the need of users to learn about the interfaces necessary to access the compute and storage resources. Also offers a way to easily create data-driven workflows.</i>
Customer of the tool	DARIAH CC
User of the service	Research groups, individual researchers
User Documentation	http://guse.hu/documentation/how-to
Technical Documentation	http://guse.hu/documentation/how-to
Product team	MTA SZTAKI
License	Apache License v2.0
Source code	https://sourceforge.net/p/guse/git/ci/master/tree/

Tool name	<i>Application-specific module API (ASM API)</i>
Tool url	https://sourceforge.net/projects/asmsp.guse.p/
Tool wiki page	https://sourceforge.net/p/guse/asmsp/wiki/Home/
Description	<i>The ASM API allows science gateway developers to create custom interfaces for workflows developed inside WS-PGRADE/gUSE.</i>
Value proposition	<i>The API helps to hide the details of WS-PGRADE/gUSE. Users of portlets developed with the help of the ASM API will only meet a user-friendly interface, without the need to learn the details of workflows.</i>
Customer of the tool	DARIAH CC
User of the service	Research groups, individual researchers
User Documentation	http://guse.hu/documentation/how-to
Technical Documentation	http://guse.hu/documentation/how-to
Product team	MTA SZTAKI
License	Apache License v2.0

Source code	https://sourceforge.net/p/guse/asmsp/code/ci/master/tree/
-------------	---

Tool name	Parameter-sweep job wizard
Tool url	Main page: https://sourceforge.net/projects/singlejobwizard.guse.p/
Tool wiki page	https://sourceforge.net/p/guse/singlejobwizard/wiki/Home/
Description	<i>The parameter-sweep job wizard is an easy-to-use job wizard tool based on WS-PGRADE/gUSE. Users of the wizard can run a parameter-sweep version of their application on different compute infrastructures by following six simple steps (executable upload, static input upload, parameter-sweep input upload, execution resource definition, command line argument definition, output files' definition).</i>
Value proposition	<i>The tool makes it easy for end-users to run their applications on compute infrastructures - they can also exploit parameter-sweep features without the need to learn the WS-PGRADE/gUSE workflow concept.</i>
Customer of the tool	DARIAH CC
User of the service	Research groups, individual researchers
User Documentation	https://ltos-gateway.lpds.sztaki.hu/web/wizard/help
Technical Documentation	https://sourceforge.net/p/guse/singlejobwizard/wiki/Home/
Product team	MTA SZTAKI
License	Apache License v2.0
Source code	https://sourceforge.net/p/guse/singlejobwizard/code/ci/master/tree/ /

Tool name	Data Avenue
Tool url	Main page: https://data-avenue.eu/
Tool wiki page	No Wiki page provided, set of manuals available through the main page
Description	<i>The Data Avenue provides a two-panel portlet view for managing data stored on different types of storage services. Users can access their data stored on these services easily, by clicking around in the portlet, like using a traditional file manager. The following functionalities are provided: browsing, data upload, data download, file transfer, file management (rename, remove), directory management (rename, remove). The Data Avenue is also integrated in WS-PGRADE/gUSE, so users have the possibility to manage remote data from inside scientific workflows as well.</i>

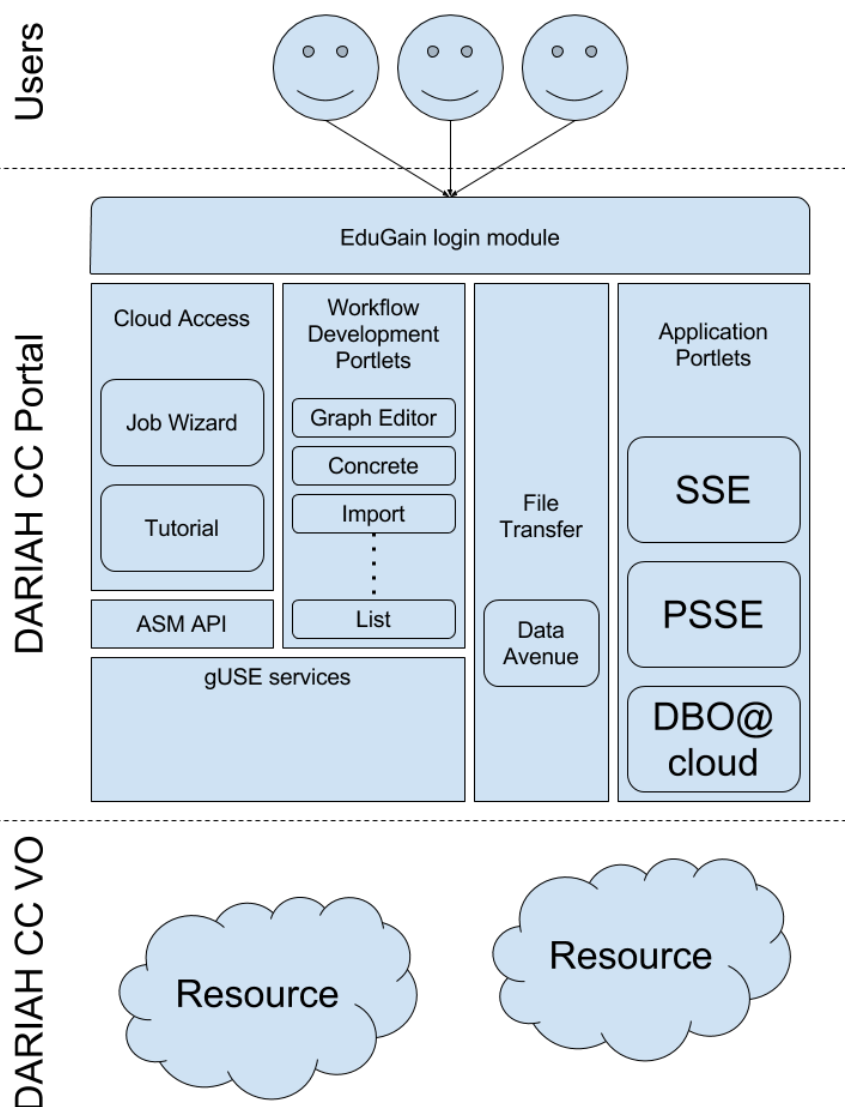
Value proposition	<i>Data Avenue eases the handling of data stored on remote storage resources, without the need to know anything about their interfaces.</i>
Customer of the tool	<i>DARIAH CC</i>
User of the service	<i>Research groups, individual researchers</i>
User Documentation	https://data-avenue.eu/data-transfer-manuals
Technical Documentation	https://data-avenue.eu/data-transfer-manuals
Product team	<i>MTA SZTAKI</i>
License	<i>Apache License v2.0</i>
Source code	https://gitlab.lpds.sztaki.hu/hajnal.akos/data-avenue-core-services

2 Service architecture

In this section we describe the service architecture behind the DARIAH CC gateway. This architecture includes the following main components:

- DARIAH CC VO
- DARIAH CC portal hosting:
 - eduGAIN login module
 - Application portlets
 - Workflow development portlets
 - ASM API
 - Cloud access portlet
 - File transfer portlet

The following figure shows the outline architecture:



In the followings we will describe the different components in detail.

1.1 DARIAH CC Virtual Organisation

The goal of the DARIAH CC Virtual Organisation (VO) is to foster arts and humanities research and collaborations, taking advantage of the EGI Infrastructure. The VO operates as part of the EGI infrastructure and is open for researchers, research projects and research groups/communities from the arts and humanities disciplines. The VO has been set up by the DARIAH Competence Centre of the EGI-Engage project. Further information about the Competence Centre is available at <https://wiki.egi.eu/wiki/CC-DARIAH>.

The VO is a closed VO - registration for users is not open, as access to the resources provided by the VO is implemented by using robot certificates (X.509 certificates allowing to access resources

of a VO on behalf of the user, without the need to possess an X.509 certificate by the user) through the DARIAH CC Portal.

Currently, the VO includes the following resources:

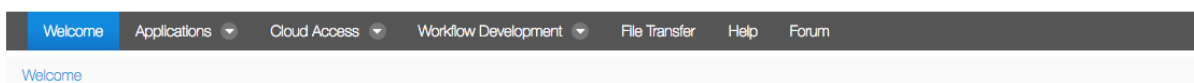
- INFN-CATANIA-STACK site:
 - Exposed API: OCCi
 - OCCi Endpoint: <https://stack-server-01.ct.infn.it:8787/>
 - URL: <https://appdb.egi.eu/store/site/inf-n-catania-stack>
 - Status: Certified
 - Number of Virtual CPU cores: 20
 - Memory: 50 GB
 - Scratch/ephemeral storage: 1 TB
 - Public IP addresses: 10
 - Access type: Opportunistic
 - Middleware: Openstack
- RECAS-BARI site:
 - Exposed API: OCCi
 - OCCi Endpoint: <http://cloud.recas.ba.infn.it:8787/occi/>
 - URL: <https://appdb.egi.eu/store/site/recas-bari>
 - Status: Certified
 - Number of Virtual CPU cores: 10
 - Memory: 20 GB
 - Scratch/ephemeral storage: 1 TB
 - Max number of instances: 7 VM
 - Public IP address: 7
 - Access type: Opportunistic
 - Middleware: Openstack

Details on the VO and the necessary steps to be performed in order to join are described on the Wiki page of the VO¹.

1.2 DARIAH CC Portal

The DARIAH CC Portal is accessible through the following URL: <https://dariah-gateway.lpds.sztaki.hu/>. The welcome screen of the gateway is shown in the following screenshot:

¹ https://wiki.egi.eu/wiki/EGI_Virtual_Organisation_for_arts_and_humanities:vo.dariah.eu



Welcome to the EGI DARIAH Gateway!

This gateway was developed within the [EGI-Engage H2020 project](#) and operated by the [EGI DARIAH Competence Centre](#). This gateway displays state-of-the-art capabilities of what's available in digital science with the use of EGI. The gateway provides various web-based applications and services for the Digital Humanities researchers, institutes and communities.

The gateway includes a [Help](#) page for giving more information on the applications and services it provides.

The gateway includes the following **applications**:

- **Simple Semantic Search Engine (SSE):**
Allows users to search in the [e-Infrastructure Knowledge Base](#) (Open Access Document Repositories and Data Repositories).
No authentication is required to access the service.
- **Parallel Semantic Search Engine (PSSE):**
PSSE is a parallelised version of SSE enabling simultaneously search across the [e-Infrastructure Knowledge Base](#), [Europeana](#), [Cultura Italia](#), [Isidore](#), [OpenAgris](#), [PubMed](#) and [DBpedia](#) platforms.
No authentication is required to access the service.
- **DBO@Cloud:**
A Cloud-based repository presenting 100+ years old collection of Bavarian dialects.
No authentication is required to access the service.

The gateway includes the following **services**:

- **Cloud Access:**
Single-job applications and parameter-sweep applications can be run on the DARIAH VO clouds without porting efforts.
[Authentication](#) is required to access the gateway service and then robot certificate provides access to the cloud.
- **Workflow Development:**
Complex workflow applications can be developed and run on all the resources of the DARIAH VO.
[Authentication](#) is required to access the gateway service and then robot certificate provides access to the cloud.
- **File transfer:**
Enables transferring data from, to and between storage services providing HTTP, HTTPS, SFTP, GSIFTP, SRM, iRODS and S3 protocols.
[Authentication](#) is required to access the gateway service and then additional authentication is needed to access the selected storage service.

Powered By Liferay

As it can be seen, the welcome page describes the available applications and services in a nutshell. The applications of the gateway can be accessed without logging in to the gateway, whereas services require authenticated access. Users can log in by clicking on the 'Sign In' button in the top-right corner. This will invoke the eduGAIN login module, and users will be able to log in by entering their home institution credentials.

The top navigation bar on the welcome page allows one to access the portal's applications and services:

- Applications: offers access to semantic search and DBO@cloud applications
 - Simple Semantic Search Engine (SSE)
 - Parallel Semantic Search Engine (PSSE)
 - DBO@cloud
- Cloud Access: provides access to the Parameter-Sweep Job Wizard
 - Parameter-Sweep Job Wizard
 - Tutorial
- Workflow Development: provides access to the core workflow development portlets of the WS-PGRADE/gUSE science gateway framework
 - Web-based graph editor portlet

- Graph portlet
- Template portlet
- Create concrete workflow portlet
- Concrete workflow portlet
- Import workflow
- Upload workflow
- List workflows
- File Transfer: includes the data avenue portlet for managing data stored on remote storages
- Help: a simple Help page
- Forum: a discussion forum.

The DARIAH CC Portal is hosted in MTA SZTAKI's OpenNebula cloud infrastructure, on a VM using 4 vCPUs, 8 GB of RAM and 142 GB of storage. The OS is CentOS 6.8, extended with Shibboleth, EGI and UMD repositories, continuously updated.

The Portal itself is based on the following software stack:

- MySQL server as the database backend
- Apache Tomcat 7 running as the servlet container
- Liferay 6.2 CE running on top of Apache Tomcat
- WS-PGRADE/gUSE version 3.7.5.

Additional services running on the portal VM are the following:

- Shibboleth SP: helps the portal to appear as a Service Provider inside eduGAIN
- httpd: web server used to provide http and https entry points to the portal, implementing connection to Apache Tomcat.

1.1.1 eduGAIN login module

The eduGAIN login module is using the following Liferay module: <https://github.com/mheder/liferay-shibboleth-plugin>. This module enables users authenticated through Shibboleth to log in automatically to the portal. No special configuration is necessary for the module, but extensive configuration was necessary in the httpd web server and the Shibboleth SP. Installation and configuration steps did include the following:

- add the shibboleth package repository to the OS
- install the 'shibboleth' package
- install necessary certificates for the service provider
- update the configuration files of shibboleth following the Shibboleth documentations^{2 3} (see Appendix I for the configuration file)
- install the 'httpd' package

² <https://wiki.shibboleth.net/confluence/display/SHIB2/NativeSPConfiguration>

³ <https://www.switch.ch/aai/guides/sp/configuration/>

- configure httpd to connect to Apache Tomcat using AJP (see Appendix II for the configuration file)
- configure httpd for shibboleth.

With this configuration, if the users click on the 'Sign In' link, they are redirected to a WAYF (Where Are You From) service, where they can select their home institute. Once selected, they are redirected to their home institute's Identity Provider (IdP), where they can enter their credentials. If the credentials are correct, the users are redirected back to the portal, which received their data, so a new user is registered in the gateway (or if the user already exists, the user is logged in).

1.1.2 Application portlets

The Application portlets menu contains the following items on the portal: Semantic Search (including simple Semantic Search Engine - SSE - and Parallel Semantic Search Engine - PSSE) and DBO@cloud.

The Semantic Search applications are simply including portlets hosted on the Catania Science Gateway⁴, whereas the DBO@cloud application is a locally deployed portlet.

The simple Semantic Search Engine (SSE), customized for the DARIAH Competence Centre project allows users to search in the e-Infrastructure Knowledge Base in more than 100 languages across more than 30 million resources contained in the thousands of semantically enriched Open Access Document Repositories and Data Repositories. Search results are ranked according to the Ranking Web of Repositories.

The parallelised SSE (PSSE) is configured to simultaneously search across the above e-Infrastructure Knowledge Base, Europeana, Cultura Italia, Isidore, OpenAgris, PubMed and DBpedia platforms.

DBO@Cloud is a Cloud-based repository presenting the work of a 100+ years old collection of Bavarian dialects within the Austrian-Hungarian monarchy from the beginning of German language to nowadays. The datasets are provided by the Austrian Academy of Science.

1.1.3 Workflow development portlets

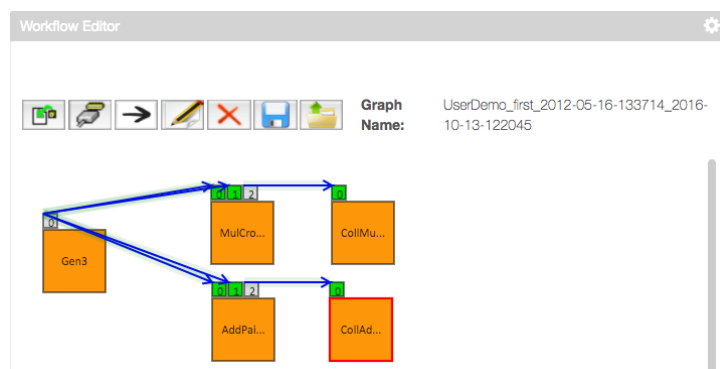
The set of Workflow development portlets exposes the main workflow creation, editing and management capabilities of WS-PGRADE/gUSE. It includes the following portlets:

- Graph portlet: this portlet presents the list of available graphs, which can be edited and allows creating new ones with a Java Web Start-based application. With the Graph Editor the user can create, modify and save a graph in an animated, graphic way. The Editor can be handled by the menu items or by the pop up menu commands, appearing after a right click on the graphic icons of jobs, ports (input and output data elements of the job) or edges (channels). The taskbar containing the icons "Job", "Port" and "Delete" gives an alternative to create jobs, ports (of a selected job) or to delete a selected job, port, or

⁴ <https://csgf.egi.eu/>

channel. With the slider, the user can zoom in/out the image of the created workflow. The recently touched object (created or identified by left click) becomes "selected". The selected state is distinguished by a red frame around the icon's graphic image. A special - third - editing mode is required for the creation of edges (channels).

- Web-based graph editor portlet: this graph editor portlet provides the same functionalities as the Java Web Start application inside the Graph Portlet, but the user doesn't need to run Java Web Start to use it.



- Template portlet: this portlet allows users to create template workflows - workflows which have some properties fixed (for example executable files, command line arguments, name and contents of input files, etc.). Templates allow for the creation of base workflows.
- Create concrete workflow portlet: this portlet allows users to create concrete workflows - workflows which can be submitted once all properties are set.
- Concrete workflow portlet: this is the main workflow portlet of WS-PGRADE/gUSE. Users can find a complete list of all of their workflows here, they can use this portlet to modify their properties, and this is where they can submit their created workflows, monitor their

execution or download their results.

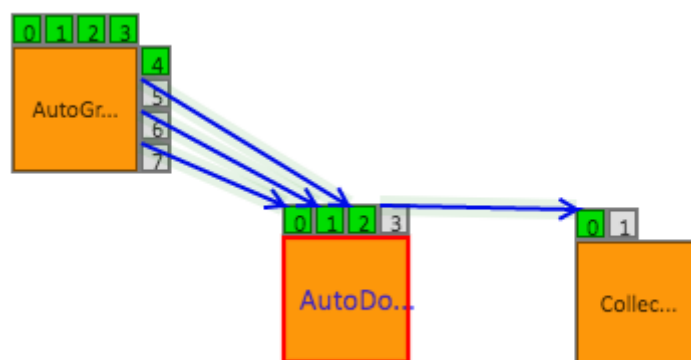
The screenshot shows the 'Concrete' workflow management interface. It features a 'Refresh' button and a table of workflows. The table has columns for 'Workflow name', 'Running', 'Finished', 'Error', 'Suspended', and 'Actions'. The workflows listed are:

Workflow name	Running	Finished	Error	Suspended	Actions
SinglePSJobByMTASZTAKI_local_2016-03-17-101029_2016-10-10-111401_2016-10-10-111449 2014-9-8	0	0	0	0	Configure Info Details Submit Delete Export
SinglePSJobByMTASZTAKI_local_2016-03-17-101029_2016-10-10-111401_2016-10-17-111123 2014-9-8	0	0	0	0	Configure Info Details Submit Delete Export
SinglePSJobByMTASZTAKI_local_2016-03-17-101029_2016-10-10-111401_2016-11-17-062019 2014-9-8	0	0	0	0	Configure Info Details Submit Delete Export
UserDemo_first_local_2016-10-13-122045 2012-5-16	0	0	1	0	Configure Info Details Submit Delete Export

Powered By gUSE

- Import workflow: this portlet can be used to import workflows either from the portals internal, or from the global workflow repository.
- Upload workflow: users can use this portlet to upload workflows from their own desktop machines or laptops.
- List workflows: this portlet shows a list of the user's portlets, with some actions related to them (download, remove, etc.).

The following figure shows the workflow of the Autodock molecular docking simulation:



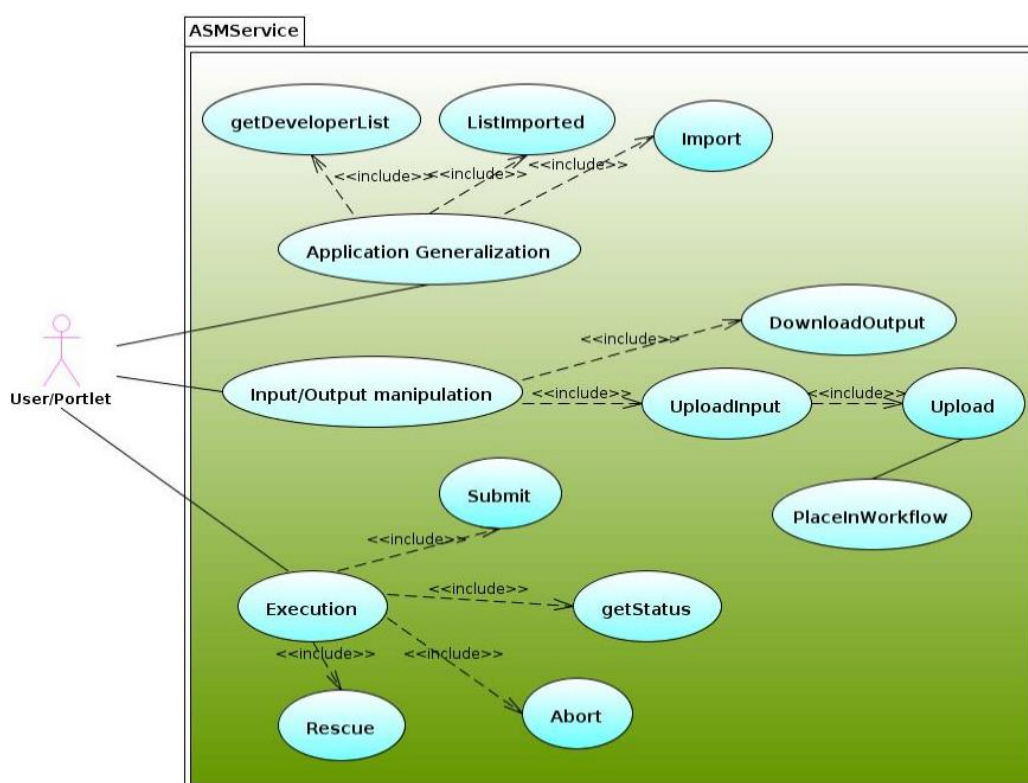
In purpose of this workflow is to perform the molecular docking of multiple ligands against a receptor molecule. For this, the user has to provide the set of ligands and the receptor. The ligand set is split into parts by the first job (AutoGrid), allowing to run multiple docking simulations in

parallel by the second job (AutoDock). Finally, the results of the different docking simulations are collected by the last job (Collector) in order to get the best docking result.

1.1.4 ASM API

Users of WS-PGRADE define their applications as workflows. They can share their applications using its internal repository solution. Hence users can import, parameterize and execute workflows developed by others. Using this idea, the Application Specific Module (ASM) offers a Java API for all of these functionalities to develop Application Specific Portlets fitting to the end-users' needs and hiding all the complex mechanisms of the underlying system.

All the functionalities, which are detailed in the following figure, can be accessed via methods of the ASMService singleton class:



ASMService class diagram shows the public (and private) methods provided by ASMService. Many of them can be easily associated to the different use cases presented by the Appendix III.

1.1.5 Cloud access portlet

The Cloud access portlet simplifies workflow creation and execution in the simplest case, if the workflow contains one job only. What looks like a significant limitation for workflow development, provides more advantages to create and execute these kinds of workflows: no Workflow Editor required, several creation and configuration interfaces are not needed in this case anymore. The job will be executed on resources that are prepared by the portal administrator, so if the

mechanism of Robot Certificates is used, users do not have to take care about authentication neither.

We were focusing on simplicity, therefore the users are able to drop their applications, the inputs (including parameter sweep) required, and then specify the names of the output files going to be generated by the application. Afterwards, the job is going to be submitted to the resource.

The following figure shows the main screen of the Cloud access portlet:

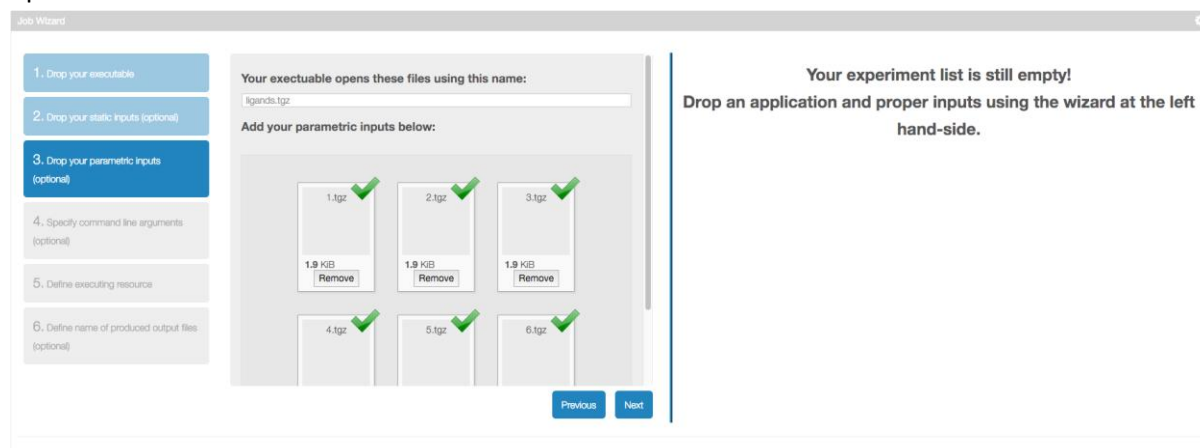
Select	Application Name	Start time	Progress	Status
<input type="radio"/>	test.sh	10 okt., H, 11:20:35 DE	100%	FINISHED
<input type="radio"/>	vina_package.sh	1 dec., Cs, 01:21:39 DU	100%	FINISHED

The left part of the screen can be used to define new parameter sweep execution, the right part of the screen shows already created executions.

A parameter sweep execution can be performed in 6 simple steps:

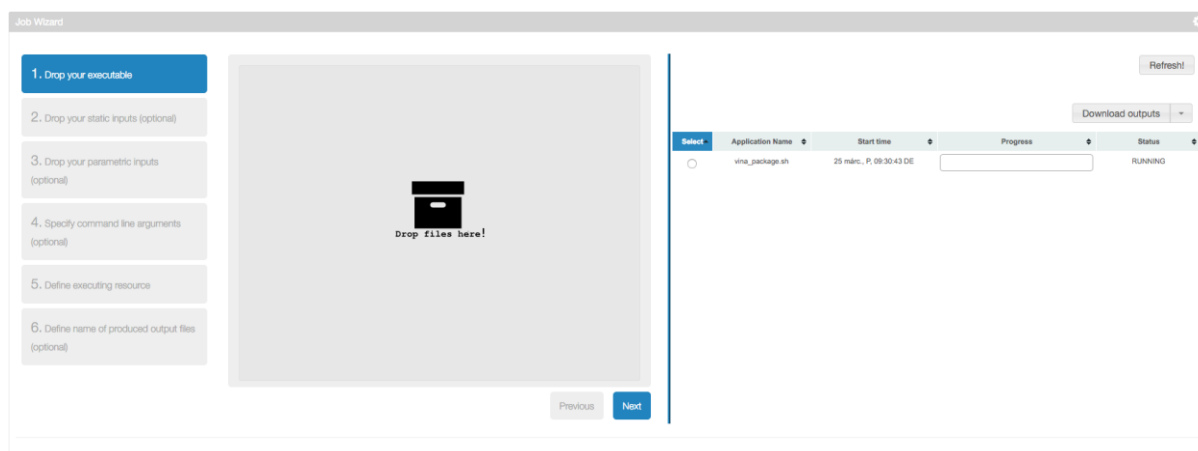
1. Upload the executable (usually a Linux binary or a shell script): the file uploaded here will be the executable for every job instance in the user's experiment. One can click the grey part in the above figure to open a file browsing dialog, or drag'n'drop the file onto the grey box.
2. Define static inputs for the executable: here the users can define those input files, which every instance of the jobs will receive. These can be for example configuration files, or additional files needed by the executable.
3. Define parametric input files: here the users can upload those files which form the parameter space. As many jobs will be executed as files were uploaded here. Each job will receive the same executable and static inputs, but all of them will receive exactly one of the files uploaded in this step. The following figure shows an example where six files is

uploaded:



4. Define command line arguments: users can define the command line arguments for the executable here.
5. Define execution resource: users can select which resource they'd like to use.
6. Define output files: finally, the users can define the set of output files their jobs will produce.

If all these steps are done, the new experiment is submitted, and its progress can be observed in the right part of the portlet:



After selecting an experiment with the radio button, the user has the possibility to perform multiple operations with the experiment:

The screenshot shows a dropdown menu with the following options: "Download all the workflow", "Abort experiment", "Remove experiment" (highlighted in blue), and "Remove All experiments". Below the menu is a table with the following columns: "Select", "Application Name", "Start time", "Progress", and "Status". The table contains one row with the following data: "vina_package.sh", "25 márc., P, 09:30:43 DE", "83%", and "RUNNING".

Select	Application Name	Start time	Progress	Status
<input type="radio"/>	vina_package.sh	25 márc., P, 09:30:43 DE	83%	RUNNING

By selecting the experiment, and clicking on the triangle next to the "Download outputs" button to get the list of operations:

- Download outputs: clicking this will download the outputs of your experiment,
- Download all the workflow: clicking this will download the whole WS-PGRADE/gUSE workflow belonging to your experiment. The downloaded workflow can later be uploaded to any WS-PGRADE/gUSE-based portal. This contains internal log files as well, so might help if debugging the reasons of an application failure,
- Abort experiment: can be used to abort a running experiment,
- Remove experiment: if the user has an experiment in the list not useful anymore, it can be removed. This will also remove all the files (both input and output) used and produced by the experiment,
- Remove all experiments: all of the experiments will be removed.

1.1.6 File transfer portlet

The File transfer portlet, based on Data Avenue enables users to manage their data stored on remote storage resources.

Use of the File transfer portlet is very similar to a file commander (like Norton Commander or Midnight Commander) with a simple two-panel (source and destination) form for data transfer.

A typical scenario for using the portlet is:

- Select a protocol to use (see the list of supported protocols at <https://data-avenue.eu/data-transfer-protocols>)
- Specify the URL (host) related to the remote storage (you can do it from Favourites set earlier), and then click Go.
- Select an authentication method and give the necessary credentials in the authentication window. The above link explains which protocols can be used for the different authentication methods. (In case of S3 protocol the access key/secret key pair should be entered instead of username/password for authentication.)
- At this point one can use the one-panel operations: users can create, delete, or rename their directories, as well as they can upload/download/refresh their files.

- If users want to use a two-panel operation (copy or move), then they need to perform the steps 1-3 in the second panel as well. Then they need to choose the source and destination panels for operation.

At this point users can use all available operations.

The following 8 operations can be executed in the File transfer portlet:

- Refresh (current directory contents)
- Mkdir (new subdirectory in the current directory)
- Rename (the selected file or directory)
- Delete (the selected file or directory)
- Copy (the selected source file or directory to the destination directory)
- Move (the selected source file or directory to the destination directory)
- Download (the selected file to your local hard drive)
- Upload (a file from your local hard drive to the current remote directory).

Upload, copy and move operations are executed asynchronously. These operations may require a Refresh operation to update the directory contents in the corresponding side.

3 Release notes

3.1 Requirements covered in the release

The work described in this deliverable shows that the competence centres have set up a research infrastructure inside the EGI-Engage infrastructure for the DARIAH community:

- the DARIAH CC VO helps resource providers offer their compute and storage resources for the arts and humanities community through a well-established technology, with already two sites participating and providing resources for the VO
- the DARIAH portal offers wide application and service stack for the DARIAH community: three A&H-specific portlets are already available, the Cloud Access helps researchers to access the DARIAH CC VO through a simple job wizard interface, finally the workflow development tools enable the exploitation of workflow features for advanced users
- The eduGAIN login module allows DARIAH members to access the portal's interface by using their home institute credentials; therefore it is not necessary to register separately for using the portal.

4 Feedback on satisfaction

The portal service did receive continuous updates based on testing feedbacks throughout its installation. The following areas have been tested by project members:

- Functionality of the portlets: basically every project partner has provided feedback on the functionality of the portlets. The different *applications* offered have been tested by INFN, as they did provide the portlets for this group. The other *services* (Cloud Access, Workflow development, File Transfer) have been tested by MTA SZTAKI members, by Davor Davidovic, and by Giuseppe La Rocca.
- Information presented on the gateway: the contents of the different pages (Welcome, Tutorial, Help) were reviewed by SA2 leader Gergely Sipos, by Giuseppe La Rocca, and other project participants as well.
- Usability of the DARIAH VO: the functionality of portlets relying on the resources offered by the DARIAH VO, more specifically the *Cloud Access* portlet was heavily tested by Giuseppe La Rocca. Based on his comments the portlet now offers full access to the DARIAH VO resources.

Further improvements will be performed based on future feedback from the DARIAH community.

5 Plan for Exploitation and Dissemination

Name of the result	DARIAH Science Gateway
DEFINITION	
Category of result	Software & service innovation
Description of the result	<p>The DARIAH Science Gateway aims to become a central hub and the access point to the EGI FedCloud infrastructure and services for the researchers coming from the digital Arts and Humanities domains, with the special emphasis on DARIAH community.</p> <p>The gateway provides a web interface, based on the WS-PGRADE/gUSE technology, for accessing the DARIAH-CC and EGI services, such as cloud sites. The gateway is an eduGAIN service provider and connected with the DARIAH AAI. The gateways are based on the robot-certificate solutions.</p> <p>Currently, the gateway supports (via portlets) several user-oriented services:</p> <ul style="list-style-type: none"> • Simple Cloud Access service • Workflow development portlet • File transfer service • Parallel Semantic Search engine • Semantic Search engine • DBO@Cloud – Bavarian dialects repository
EXPLOITATION	
Target group(s)	All Arts and Humanities researchers, which could be both institutional (with facilities like authentication) and non-institutional, that wants to use a service

	<p>that is provided via EGI infrastructure. Individual researcher, research groups, service providers coming from DARIAH-EU. In addition, DARIAH Working Groups and DARIAH (DH) related projects (international collaborations).</p>
<p>Needs</p>	<p>The main DARIAH (DH) requirements that the Gateway tries to fulfil:</p> <ul style="list-style-type: none"> • researchers need access to sustainable storage and compute resources • researchers cannot access and/or authenticate to use EGI FedCloud services • researchers do not have financial resources nor technical expertise to initiate their own services on EGI infrastructures • researcher do not have an overview of what EGI (Cloud) can provide and how to consume these offers • researchers want to use their existing institutional and/or DARIAH account to access the resources and services
<p>How the target groups will use the result?</p>	<p>The DARIAH Science Gateway is a web-based service available at https://dariah-gateway.lpds.sztaki.hu/ to everyone. However, apart from the three services (Parallel semantic search, Semantic search and DBO@Cloud), all other offered services are protected and users need to registered before in order to use them. Users can access these services via DARIAH Science Gateway by login using their valid institutional accounts (if their institution is a member of eduGAIN) or by using their DARIAH account (if have one). Users without these credentials cannot use the Gateway services. Upon successful log-in, users can access all the services.</p> <p>New services can be added to the portal if required by users. However, the process of adding a new service is not automatic and requires a technical support that will be provided via DARIAH working group called “Cloud infrastructure”.</p> <p>For accessing the Gateway, the users only need laptop/desktop computer and internet connection.</p>
<p>Benefits</p>	<p>The DARIAH Gateway provides an environment to interface between the DARIAH (DH) researchers and EGI services.</p> <p>DARIAH community will benefit from an easy access to EGI FedCloud resources by using their own authentication (provided by their home institution or DARIAH AAI) which excludes the additional overhead of authentication and authorization within EGI ecosystem. The access to FedCloud is possible without any additional technical expertise, possibility to prepare and run their use-case specific</p>

	workflows and share it with other researchers.
How will you protect the results?	DARIAH Science Gateway as well as WS-PGRADE/gUSE framework are fully open source components licensed under Apache 2.0 license.
Actions for exploitation	<p>The DARIAH Science Gateway supports the DARIAH Virtual Organization. The detailed user documentation on how to use the gateway itself and the specific services offered can be found directly on the gateway web-site and on the DARIAH Competence Centre wiki pages: wiki.egi.eu/wiki/Competence_centre_DARIAH. The technical support for the gateway will be provided by the DARIAH Competence Centre and after the end of the EGI-Engage project, the operational activities will be transferred to the DARIAH working group "Cloud infrastructure".</p> <p>The gateway itself continuously receives security upgrades, and also will be upgraded to future versions of WS-PGRADE/gUSE. Users may contact the administrators through the forum provided on the gateway itself: https://dariah-gateway.lpds.sztaki.hu/forum. This forum also provides a platform for discussion about the different portlets offered by the gateway.</p>
URL to project result	https://dariah-gateway.lpds.sztaki.hu/
Success criteria	<p>Number of users who accessed and login to the gateway</p> <p>Number of job submitted via the gateway (to EGI FedCloud and other infrastructures)</p> <p>Number of cloud sites (e.g. EGI FedCloud, DARIAH-related cloud sites)supported by the gateway (via DARIAH VO)</p> <p>Number of offered services and application integrated/connected with the gateway</p>
DISSEMINATION	
Key messages	With DARIAH Science Gateway you can easily access and use various cloud resources using your existing home-institutional account. Furthermore, your existing services and application can also benefit from using the cloud infrastructure if integrated into the gateway which also provide secure and transparent authentication of users of you services.
Channels	<p>DARIAH newsletter and DARIAH web-pages, announcements to DARIAH mailing list. Posting news on Twitter.</p> <p>Submitting technical notes, posters and presentations to DH-related scientific events and workshops.</p> <p>Giving lectures and presentations on DARIAH Annual meetings.</p> <p>Communication (direct contact, skype calls) with DARIAH Working Group</p>

	leaders, research groups and projects.
<i>Actions for dissemination</i>	<p>Past dissemination actions:</p> <ul style="list-style-type: none"> • Presentation at EGI Community Forum, Bari, 2015 • Presentation at EGI User Forum, Amsterdam, 2016 • Presentation at DI4R Conference, Krakow, 2016 • Full paper published in conference proceedings of the XVII Euralex conference, Tbilisi, Georgia, 2016 • Workshop at Euralex Conference, Tbilisi, Georgia, 2016 • Lightning talk at DARIAH Annual meeting, Ghent, 2016 • Workshop, DARIAH Annual meeting, Ghent, 2016 • Service demonstration, DARIAH Annual meeting, Ghent, 2016 <p>Future event (tentative list):</p> <ul style="list-style-type: none"> • Presentation/workshop/demonstration at DARIAH Annual meeting, Berlin, 2017 • Presentation at DPASSH2017, Brighton (UK), 2017 • ENel COST meeting and workshop, Budapest, 2017 <p>We have foreseen to prepare a short video presenting the functionality of the gateway.</p>
<i>Cost</i>	5,000 €
<i>Evaluation</i>	<p>The impact of the dissemination actions will be evaluated by monitoring the following success criteria:</p> <ul style="list-style-type: none"> • number of given presentations and workshops • number of accepted scientific and technical papers • number of visitors/users of the gateway • number of new applications deployment request • an increased number of the portal users (registered) • an increased number of different countries from which users are coming from

6 Future plans

The DARIAH CC portal in its present status offers most features to have DARIAH computation started on the EGI infrastructure. As such, in the near future the DARIAH CC plans to perform the following tasks related to the portal:

- perform intense promotion and advertising within digital humanities communities

- operating system security updates
- monitoring implementation
- if a new version of WS-PGRADE/gUSE is released, upgrade the portal to use the new version
- if new portlets for the arts and humanities community are developed and released, they should be deployed onto the portal.

To sum up, within the CC we plan to operate the portal further, with new portlet deployments on request. DARIAH CC plans to further provide technical support on the integration and portlet deployment onto the portal in two directions. The first direction is the integration and adoption of new applications and service that could be done through new project proposals. The second direction is the deployment of the services and applications with already developed and released portlets (in-kind contribution).

Appendix I - shibboleth2.xml config file

```
<?xml version="1.0"?>
<SPConfig xmlns="urn:mace:shibboleth:2.0:native:sp:config"
xmlns:conf="urn:mace:shibboleth:2.0:native:sp:config"
xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
xmlns:md="urn:oasis:names:tc:SAML:2.0:metadata" clockSkew="180">
  <ApplicationDefaults entityID="https://dariah-gateway.lpds.sztaki.hu/shibboleth"
REMOTE_USER="uid eppn persistent-id targeted-id" attributePrefix="AJP_">
    <Sessions lifetime="28800" timeout="3600" relayState="ss:mem" checkAddress="false"
handlerSSL="true" cookieProps="https">
      <SSO discoveryProtocol="SAMLDS" discoveryURL="https://wayf.aai.dfn.de/DFN-AAI-
eduGAIN/wayf">
        SAML2
      </SSO>
    <Logout>SAML2 Local</Logout>
    <Handler type="MetadataGenerator" Location="/Metadata" signing="false"/>
    <Handler type="Status" Location="/Status" acl="127.0.0.1 ::1"/>
    <Handler type="Session" Location="/Session" showAttributeValues="false"/>
    <Handler type="DiscoveryFeed" Location="/DiscoFeed"/>
  </Sessions>
  <Errors supportContact="root@localhost" helpLocation="/about.html"
stylesheet="/shibboleth-sp/main.css"/>
  <MetadataProvider type="Chaining">
    <MetadataProvider type="XML" uri="http://metadata.eduid.hu/current/niifi.xml"
backingFilePath="niifi.xml" legacyOrgNames="true" reloadInterval="7200">
      <MetadataFilter type="Signature" certificate="href-metadata-signer-2011.crt"/>
    </MetadataProvider>
  </MetadataProvider>
</SPConfig>
```

```

    <MetadataProvider      type="XML"      uri="http://metadata.eduid.hu/current/hexaa.xml"
    backingFilePath="hexaa.xml" legacyOrgNames="true" reloadInterval="7200">
        <MetadataFilter type="Signature" certificate="href-metadata-signer-2011.crt"/>
    </MetadataProvider>

    <MetadataProvider      type="XML"      uri="http://metadata.eduid.hu/current/sztaki.xml"
    backingFilePath="sztaki.xml" legacyOrgNames="true" reloadInterval="7200">
        <MetadataFilter type="Signature" certificate="href-metadata-signer-2011.crt"/>
    </MetadataProvider>

    <MetadataProvider      type="XML"      uri="https://ldap-dariah.esc.rzg.mpg.de/idp/shibboleth"
    backingFilePath="dariah-prod-idp-metadata.xml" reloadInterval="7200">
        </MetadataProvider>

    <MetadataProvider      type="XML"      uri="https://ldap-dariah-
    clone.esc.rzg.mpg.de/idp/shibboleth"      backingFilePath="dariah-test-idp-metadata.xml"
    reloadInterval="7200">
        </MetadataProvider>

    <MetadataProvider      type="XML"      uri="https://gridp.garr.it/metadata/gridp.xml"
    backingFilePath="gridp.xml" reloadInterval="7200">
        </MetadataProvider>

    <MetadataProvider      type="XML"      uri="https://gridp.garr.it/metadata/gridp-test.xml"
    backingFilePath="gridp-test.xml" reloadInterval="7200">
        </MetadataProvider>

    <MetadataProvider      type="XML"      uri="https://www.aai.dfn.de/fileadmin/metadata/DFN-AAI-
    eduGAIN+idp-metadata.xml" backingFilePath="wayf.xml" reloadInterval="7200">
        </MetadataProvider>
    </MetadataProvider>

    <AttributeExtractor type="XML" validate="true" reloadChanges="false" path="attribute-
    map.xml"/>

    <AttributeResolver type="Query" subjectMatch="true"/>

    <AttributeFilter type="XML" validate="true" path="attribute-policy.xml"/>

    <CredentialResolver type="File" key="sp-key.pem" certificate="sp-cert.pem"/>
</ApplicationDefaults>

<SecurityPolicyProvider type="XML" validate="true" path="security-policy.xml"/>

<ProtocolProvider type="XML" validate="true" reloadChanges="false" path="protocols.xml"/>
</SPConfig>

```

Appendix II. - httpd ssl configuration file

```
LoadModule ssl_module modules/mod_ssl.so

Listen 443

<VirtualHost *:443>
    ServerName      dariah-gateway.lpds.sztaki.hu
    ServerAlias     dariah-gateway.lpds.sztaki.hu
    ServerAdmin     root@localhost
    LogLevel warn
    SSLEngine on
    SSLCertificateFile /etc/ssl/certs/dariah-gateway_lpds_sztaki_hu.crt
    SSLCertificateKeyFile /etc/ssl/certs/dariah-gateway_lpds_sztaki_hu.key
    SSLCertificateChainFile /etc/ssl/certs/DigiCertCA.crt
    SSLProtocol all -SSLv2
    <Location /c/portal/login>
        AuthType shibboleth
        ShibRequireSession On
        require valid-user
    </Location>
    BrowserMatch ".*MSIE.*" \
        nokeepalive ssl-unclean-shutdown \
        downgrade-1.0 force-response-1.0
    ProxyPass      /shibboleth      !
    ProxyPass      /Shibboleth.sso  !
    ProxyPass      /                  ajp://localhost:8009/
    ProxyPass      /wspgrade         ajp://localhost:8009/wspgrade
    ProxyPass      /wfs              ajp://localhost:8009/wfs
    ProxyPass      /submitter        ajp://localhost:8009/submitter
    ProxyPass      /wfi              ajp://localhost:8009/wfi
    ProxyPass      /dci_bridge_service ajp://localhost:8009/dci_bridge_service
    ProxyPass      /statvisualizer   ajp://localhost:8009/statvisualizer
    ProxyPass      /singlejobwizard  ajp://localhost:8009/singlejobwizard
    <Proxy *>
        Order allow,deny
        Allow from all
    </Proxy>
</VirtualHost>
```

Appendix III - ASMService methods

Functionality	Method	Return Type	Description
getDeveloperList	getWorkflowDevelopers	String[]	Get a stringlist of application Developer IDs that have exported at least one application to the local repository according to the necessary requirements (see ApplicationType object)
ListImported	getASMWorkflows	List<ASMWorkflow>	Returns a list of ASMWorkflow objects that represent the available applications for the user
Import	ImportWorkflow	Void	Imports an application from the local shared repository to the user space
DownloadOutput	getFiletoPortalServer	Void	Downloads the needed file to the portal server
DownloadOutput	getFileStream	Stream	Returns the file in a stream
Upload	uploadFiletoPortalServer	Void	Uploads a file to the portal server
PlaceInWorkflow	placeUploadedFile	Void	Place a file that has been already uploaded to portal server, in a workflow as an input file
Get Remote Input	getRemoteInputPath	String	Returns the actual remote input path associated to a specified job/port
Set Remote Input	setRemoteInputPath	Void	Sets the remote input path according to the string added as parameter
Get Remote Output	getRemoteOutputPath	String	Returns the actual remote input path associated to a specified job/port
Set Remote Output	setRemoteOutputPath	Void	Sets the remote input path according to the string added as parameter
Get Command line argument	getCommandLineArg	String	Returns the actual command line argument that is set for a specified job
Set Command line argument	setCommandLineArg	Void	Sets the command line argument for a specified job
Get resource	getResource	ASMRResourceBean	Returns the actual resource where the job will be submitted to.
Set resource	setResource	Void	Sets a resource
Submit	submit	Void	Submits an imported application
getStatus	getWorkflows -> Overall status statistics in JobStatistics bean	List<ASMWorkflow>	Gets general status statistics about the workflow that is being executed
getStatus	getDetails	WorkflowInstanceBean	Returns detailed status codes about every job that is being submitted according to the application (it returns)
Rescue	rescue	ASMService	Rescues an application
Abort	abort	ASMService	Aborts an application