Project Number: **RI-312579**

Project Acronym: **ER-flow**

Project Full Title:

Building an European Research Community through Interoperable Workflows and Data

Theme: **Research Infrastructures**
Call Identifier: **FP7-Infrastructures-2012-1**
Funding Scheme: **Coordination and Support Action**

Deliverable D*5.4*
User evaluation of the simulation platform

Due date of deliverable: 31/05/2014
Start date of project: 01/09/2012

Actual submission date:
Duration: 12 months

Lead Contractor: University of Westminster
Dissemination Level: PU
Version: 0.1

ER-flow is supported by the FP7 Capacities Programme under contract no RI-312579

1 Table of Contents

2 List of Figures and Tables

3 Status and Change History

Status:	Name:	Date:	Signature:
Draft:	Giuliano Taffoni		n.n. electronically
Reviewed:			n.n. electronically
Approved:			n.n. electronically

Table 1. Deliverable Status

Version	Date	Pages	Author	Modification
0.1	15/04/14	??	GT	Created Skeleton
0.2	30/04/14	??	GT	Skel update
0.3	5/05/14	15	GT	Update Glossary and A&A section
0.4	15/05/14	25	VK, KV	LIfeScience Section + Feedback tools + New Com
0.5	16/05/14	28	KE, GT	Discussion
0.6	18/05/14	32	GabT	SHIWA platform description
0.7	19/05/14	41	BB&GP	Helio Section + Appendix A and B
0.8	20/05/14	42	RG, KV	Revision and comments
0.9	21/05/14	43	GabT, SDO	External community + SDO comments
1.0	22/05/14	41	ALL	Comments and general corrections

Table 2. Deliverable Change History

Glossary

ASM	Application Specific Module
CGI	Coarse-Grained Interoperability
DCI	Distributed Computing Infrastructure
EGI	European Grid Infrastructure
FGI	Fine-Grained Interoperability
gLite	Grid middleware used in EGI
HPC	High performance computing
HTC	High throughput computing
MOTEUR	Workflow management system
NGI	National Grid Infrastructure
SGW	Science Gateway
SSP	SHIWA Simulation Platform
UoW	University of Westminster
UNICORE	Grid middleware used in the German Grid
VO	Virtual Organisation
Y1	First year of the ER-flow project
Y2	Second year of the ER-flow project
WF	Workflow
WE	workflow engine
WP	Work package
WS-PGRADE	Workflow management system
WfMS	Workflow management system in general

Table 3. Glossary

4 Introduction

The FP7 "**Building a European Research Community through Interoperable Workflows and Data**" (ER-flow) project disseminates the achievements of the SHIWA project[1] and uses these achievements to build workflow user communities across Europe. ER-flow provides application support to research communities within and beyond the project consortium to develop, share and run workflows with the SHIWA Simulation Platform (SSP). In particular, the work package 5 (*WP5 Application Support)* deals with the creation and porting of applications to the SSP, as well as dissemination of the SHIWA platform, services and culture of workflow interoperability into diverse scientific communities. Initially four different research communities that participate as members in the project have been addressed, namely Heliophysics, Astrophysics, Computational Chemistry and Life Sciences. At the end of the first project year (Y1) and at the beginning of the second project year (Y2) two additional communities have been addressed, namely DRIHM (Distributed Research Infrastructure for Hydro-Meteorology) and SOMNO.Netz (network for sharing sleep research datasets).

This document summarises the experience of workflow developers during exploration of the SSP for the user communities. It is focused on the activities done during the second year of the ER-flow project (Y2), and contains feedback about the platform collected by the user communities. Note that, since the project kick-off meeting, a communication channel has been defined between WP5 and WP3 for feedback. WP3 is in charge of the infrastructure and technical support, and it has produced project deliverable D3.3 (*Expanded simulation platform report*). That deliverable has to be mentioned here because it describes the enhancement requests submitted by the ER-flow communities and how they were addressed by WP3. Those feedback and requirements contained in D3.3 are not duplicated here when not necessary; however, we briefly describe all the feedback tools and services setup during Y2 to collect information from the communities and to make users and developers interact.

From the user's perspective, the SHIWA platform offers three main services: the SHIWA Portal, the SHIWA Submission Service and the SHIWA Repository. For a more detailed and technical description of the complete SHIWA platform and services, please refer to the Deliverables D3.1 (*Study of the adaptation options of the simulation platform*[2]) and D3.2 (*Extended simulation platform*[3]), D3.3 (*Expanded simulation platform report*[4]*).*

During Y1, the communities conducted different plans for the exploration of these services; however, with the following similarities:

a) they use the SHIWA simulation platform;
b) they use the SHIWA Repository to find new workflows and/or to store their workflows and the associated documentation;
c) they run science gateways with customized interfaces for scientists;
d) the interfaces are optimized for (a set of) applications of relevance in that scientific field;
e) there were two main user profiles in each community: **workflow developer** and **scientists** (or **end-users).**

[1] http://www.shiwa-workflow.eu/project
[2] https://documents.egi.eu/document/1751
[3] https://documents.egi.eu/document/1859
[4] https://documents.egi.eu/document/2209

During Y2, on the contrary, the communities extended the use of the platform to a more complex approach. Sometimes their application reached the technological limits of the platform itself, giving the developer the opportunity to learn and increase the functionalities and the capabilities of the SSP. For example, thanks to the extended use of Taverna meta-workflows, the developers identify and reduce some of the redundancy of entries that was found during Y1 and highlighted in the D5.1 document by the Heliophysics community. Also the use of cloud computing resources allow to identify some limits in the platform as described in the following sections.

Moreover, since in these communities the scientists do not interact with the SHIWA services directly, in this deliverable we focus on the feedback provided by the workflow developers. Note that typically a small team of workflow developers supports a large number of scientists; therefore, the number of people who effectively provided feedback for this deliverable is small (typically 3-5 people for each community) and they are all only workflow developers.

This document is organized as follows. In Section 5 we present the development status of the SSP at the time the document has been written and we summarize all the feedback and collaboration tools that were setup during this project.

Then each community individually summarises its experiences with the SHIWA simulation platform and technology (Sections 6,7,8,9), following a common scheme:

a) First they present an update of the usage scenarios proposed in Y1.
b) Then they present new scenarios adopted in Y2. Here the communities describe the usage of the simulation platform in Y2, providing a short explanation about the development environment. Focus is given to the evolution between the approach used in Y1 and in Y2.

One section is dedicated to the new external communities that joined the project during Y2 (Section **Hiba! A hivatkozási forrás nem található.**~~10~~).

In Section 11, the communities discuss their experience when using the SHIWA services (workflow repository, submission service, portal) to carry out the envisioned usage scenarios during Y2. Both functionality and usability aspects are considered, such as user-friendliness and intuitiveness. The communities comment on the following points: development environment, services & functionality, user interface, scalability, and reliability. In the last part of Section 11 we present an overview of the feedback and recommendations for further developments, summarizing commonalities between the various communities.

5 SHIWA platform and Feedback tools

We present the development status of the SSP at the time the document has been written and we summarize all the feedback and collaboration tools we setup during this project. For more details please refer to Deliverable D3.3 (Expanded simulation platform report).

ER-flow research communities and technology developers defined in Y1 the following generic usage scenarios for the SSP:

- ER-flow development environment with power user view (See details in D3.2) and
- ER-flow execution environment with end user view and/or ASM portlets (See details in D3.3).

ER-flow Usage Scenarios

WP3 designed and implemented the SHIWA Submission Service to enable execution of non-native workflows through the community gateways. The work package defined three usage scenarios to execute non-native workflows:

- scenario 1 (Fig. 4): Researchers develop and run non-native workflows on the SHIWA Simulation Platform using its services as local services.
- scenario 2: (Fig 5) Research communities deploy and manage their own community gateway which is remotely connected to the SHIWA Repository and the SHIWA Submission Service. They use the SHIWA Submission Service of the simulation platform to run non-native workflows.
- scenario 3: (Fig. 6) Research communities deploy and run their own community gateway and a submission service, both connected to the SHIWA Repository. The SHIWA Repository is used as a remote service and the submission service as a local one.

WP3 deployed a prototype SHIWA Simulation Platform consisting of the extended SHIWA Repository and the new SHIWA Submission Service in March 2014. The work package ran the prototype platform in parallel with the production platform for one month. WP3 switched off the previous production platform and replaced it with the prototype platform in April 2014.

ER-flow usage scenario 1: running non-native workflows from the SHIWA Portal through the centrally deployed SHIWA Submission Service (Fig. 1)

This usage scenario uses only the SHIWA Simulation Platform. Researchers and workflow developers run non-native workflows accessing the SHIWA Repository and the SHIWA Submission Service of the simulation platform through the SHIWA Portal. Research communities without community gateways can use the simulation platform as both development and execution environment. Research communities with community gateways can use the simulation platform for developing and testing new features and services for example connecting a community gateway to new computing resources, creating the CGI support for a workflow system, etc.

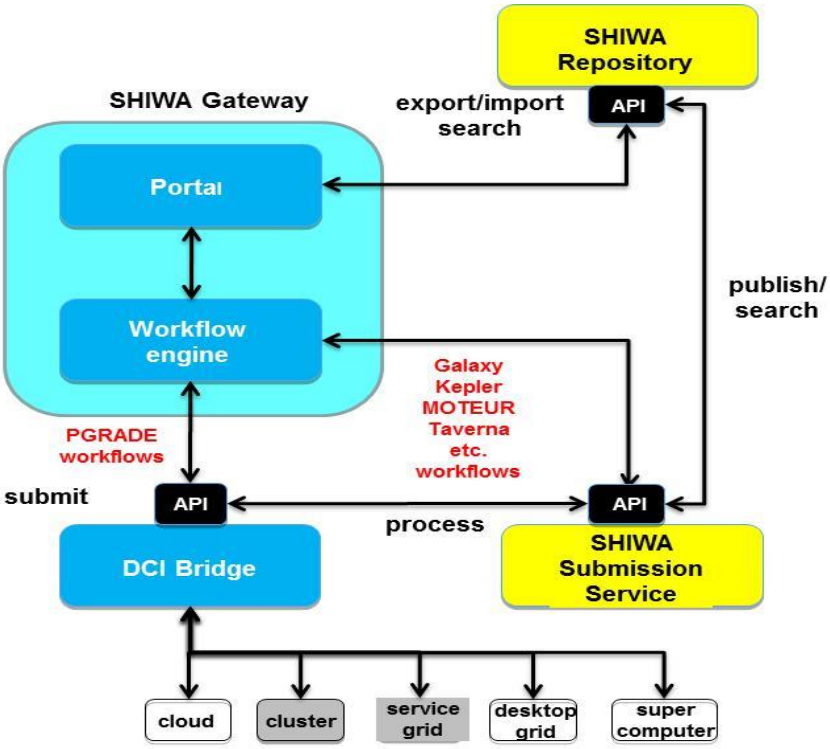

Figure 1 ER-flow usage scenario 1

ER-flow usage scenario 2: running non-native workflows from the community gateway through the centrally deployed SHIWA Submission Service (Fig. 2)

In this usage scenario researchers and workflow developers use the SHIWA Repository and the SHIWA Submission Service of the SHIWA Simulation Platform as remote services and their community gateway as a local service. They have to configure their gateway to enable remote access to the SHIWA Repository to perform workflows export and import operations, and access to the SHIWA Submission Service to submit non-native workflows. They can use the SHIWA Simulation Platform as the ER-flow Development Environment to create and test workflows and their community gateway as the ER-flow Execution Environment to run the workflows accessing the repository and submission service remotely.

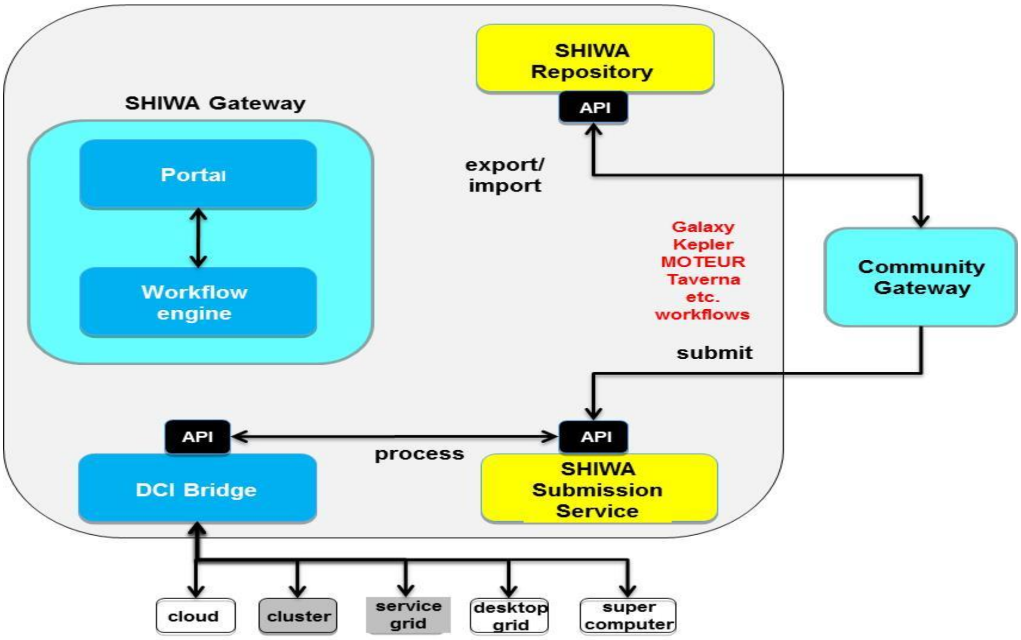

Figure 2 ER-flow usage scenario 2

ER-flow usage scenario 3: running non-native workflows from the community gateway through the locally deployed submission service (Fig. 3)

In this scenario the communities deploy and connect the SHIWA Submission Service to their community gateway. Researchers and workflow developers use the community gateway as both development and execution environment. They need access remotely only to the SHIWA Repository to download and upload workflows.

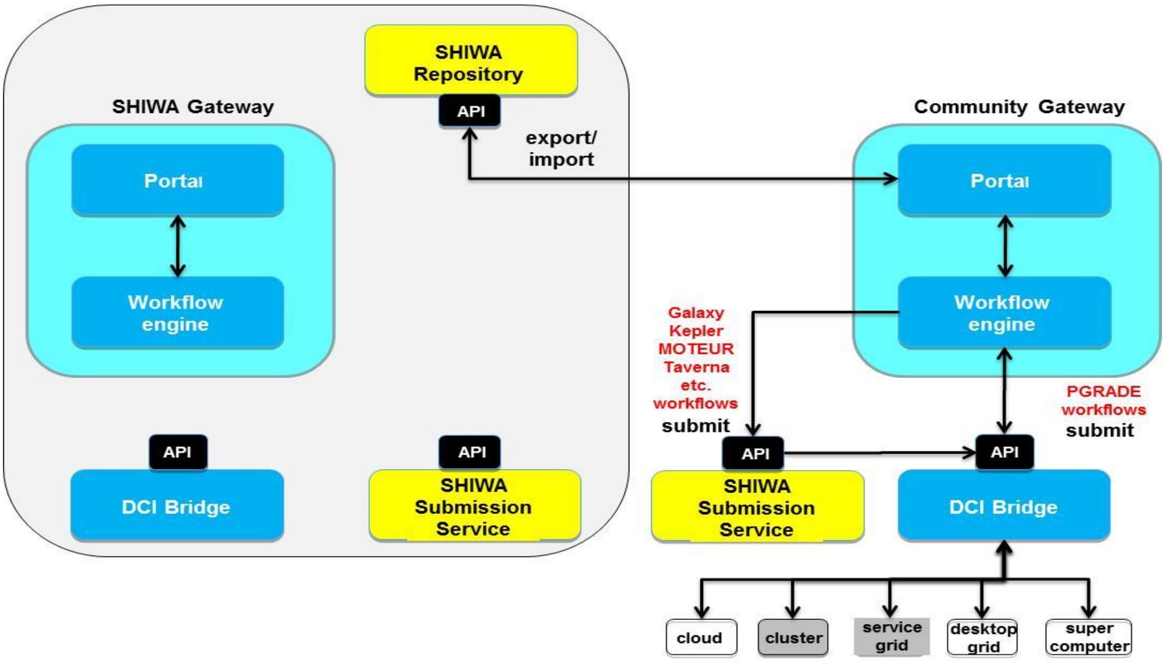

Figure 3 ER-flow usage scenario 3

5.1 *Feedback and collaboration tools*

Besides the regular communication channel between WP5 and WP3 to collect feedback and report enhancement requests, ER-flow established and manages the ER-flow Bugzilla, a set of tools to help the communities to exchange experiences among themselves and to provide feedback to the platform developers. See details in D3.3 (Expanded simulation platform report[5]*)*. Two of these tools are most relevant in the scope of this document: the SHIWA User Forum and User Steering Board.

The **SHIWA User Forum** was established during the SHIWA project to serve as the main communication channel between the users and the developers to build a strong user community and provide user feedback. At first it was a regular message board on the SHIWA website with different topics where users could post comments or start their own threads. After the start of the ER-flow project, the SHIWA User Forum was moved to the Google Groups platform to manage discussions. The main advantage is that this type of forum can work as a mailing list as well. The forum members are able to answer the posts or start a new discussion by email. Through the web interface anyone can easily join the User Forum.

The new User Forum's web interface (see Figure 4~~Figure 4~~) can be accessed from the SHIWA website[6], from the ER-flow website[7], from the SHIWA Portal[8] or from an independent URL[9]. There is also a possibility to embed it into any website (e.g. community websites, etc.). On the new Google Groups based User Forum no pre-defined categories and topics were established, as users found it difficult to search for information. Instead, topics are added dynamically when a new message is written. Two options are available to search between the available topics:

1. If the User Forum is visited through its web interface, then Google Groups provides a powerful search mechanism at the top of the page to find the topics of interest;
2. If the members use email, then they can use filtering or searching built into their email interface.

[5] https://documents.egi.eu/document/2209
[6] http://www.shiwa-workflow.eu/shiwa-user-forum
[7] http://www.erflow.eu/shiwa-user-forum
[8] https://shiwa-portal2.cpc.wmin.ac.uk/liferay-portal-6.1.0/web/guest/shiwa-user-forum
[9] https://groups.google.com/d/forum/shiwa-user-forum

Figure 4 The SHIWA User Forum

The users are heavily using the SHIWA User Forum and there are about 40 new posts per month. At the moment about 100 topics and 450 messages are present on the User Forum.

Some of the main topics that started vivid conversations:

- Proxy transformations inside SHIWA portal
- Moteur workflows on SHIWA portal
- Can't export a workflow

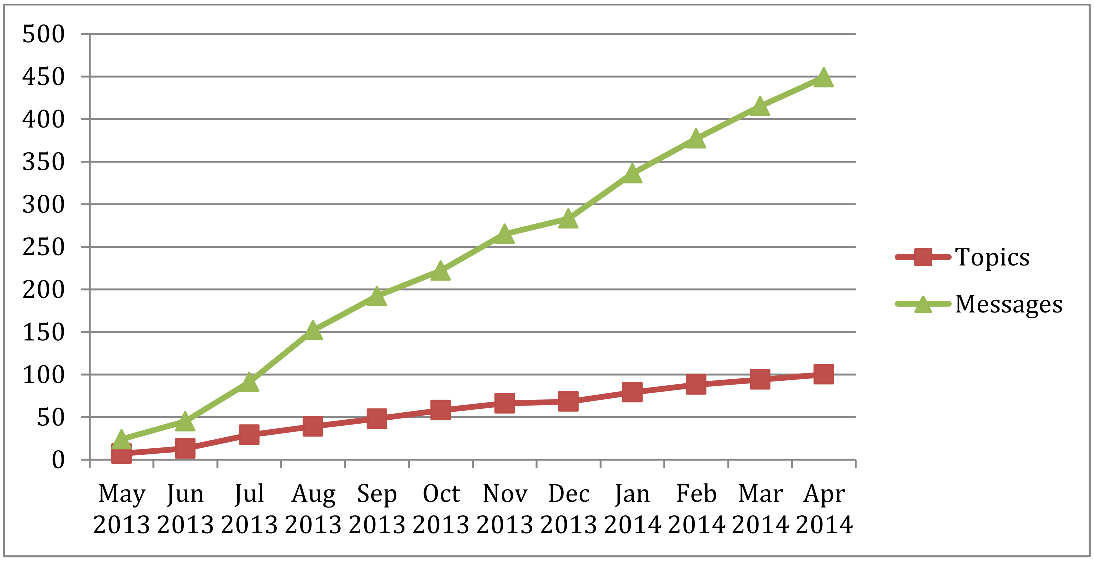

Figure 5 SHIWA Forum usage

The ER-flow **User Steering Board** gathered in Y1 information about issues that are affecting the ability to accomplish the goals of ER-Flow. Some issues are associated with SHIWA platform but others may be related to community-related capabilities. The User Steering Board was also interested in any suggestions that anyone had about how to improve both the platform and its usage. The Board prepared and circulated an issue report form that allows user to suggest improvements to the platform in terms of science production and usability. Only a few forms were completed by the main four user communities (<10). The collected feedback was not different from information available through other channels, and referred mostly to platform malfunction. Most of the problems have been addressed by WP3.

6 Astrophysics

Astronomy is a natural science that deals with the study of celestial objects, and Astrophysics is the branch of astronomy that deals with the physics of the Universe, including the physical properties of celestial objects, as well as their interactions and behaviour. A&A (Astronomy and Astrophysics) compose a worldwide, geographically distributed community that operates large experiments and observing facilities, and that uses large computing and data resources. Actually, A&A includes a constellation of sub-communities and related projects that have different scientific approaches and computing/data requirements, but who share the use of common tools and services:

a) a common data and computing infrastructures;
b) a common data format (FITS);
c) a common set of libraries and software (e.g. cfitsio);
d) a standard for data access and exploitation (given by the Virtual Observatory project).

Within the ER-flow project the A&A community is represented by INAF (Istituto Nazionale di Astrofisica, Italy). It coordinates the activity of other European partners that mainly contribute with applications and pre-existing workflows to the project and also with technological knowledge related to DCIs and workflow systems

6.1 *SHIWA Platform usage*

During Y1, A&A setup the infrastructure composed by 6 Science Gateways (SGW):

- INAF - Osservatorio Astrofisico, Catania (OACT) Italy[10]
- INAF - Osservatorio Astronomico, Teramo (OATE) Italy[11]
- INAF - Osservatorio Astronomico, Trieste (OATS) Italy[12]
- University of Portsmouth, UK (UP)[13]
- Astronomical Institute of Slovak Academy of Sciences (AISAS)[14]
- SHIWA Portal, University of Westminster, London, UK[15]

During Y2 the A&A community works on the consolidation of this infrastructure:

a) a common Single Sign-On system has been designed: it allows users to use the same credential to access different SGW
b) a distributed file system is under evaluation: data will be shared among SGWs;
c) SGWs improve their functionalities:
    a. INAF- Osservatorio Astronomico Trieste SGW is operative and it is able to submit tasks on a local cluster (92 Cores with 2GB Ram per core) and on EGI DCI using gUSE/WS-PGRADE.

Five A&A Gateways formalize their infrastructure and collaboration building the STARnet Gateway Federation[16]. namely: INAF-OACT, INAF-OATE, INAF-OATS, UP, IASAS.

[10] http://visivo.oact.inaf.it:8080/
[11] http://193.204.1.135:8081/
[12] http://guse-fe.oats.inaf.it:8080/
[13] http://148.197.12.1:8081/
[14] http://sg-mph.ta3.sk:8081/
[15] https://shiwa-portal2.cpc.wmin.ac.uk/liferay-portal-6.1.0/
[16] http://www.oact.inaf.it/STARnet/

Y1 was focused on design and development of SSP native WFs (they were coded in WS-PGRADE/gUSE). The workflows were stored in the SHIWA Repository, and tested during Y2.

During Y1 we identified *workflow developers* who:

a) implement the scientific applications as WF;
b) modify the SGW incorporating the WF and implement a customised web interface,.

and *end-users* (Astronomers and Astrophysicists) who

a) benefit from the customized SGW to run the applications.

All astronomical applications ported on SSP during Y1 use gLite as Grid middleware, except the VisIVO services. During the Y2 we integrated also local resources in order to have full control on the running tasks. Due to their nature, distributed infrastructures are more complex to monitor and jobs may failure for different reasons. SGWs are designed to hide the complexity of these infrastructures, but they make it difficult to debug any problem. This is however an intrinsic problem that affects any WF service and any SGW tool. It is not related to the SSP nor to A&A.

The A&A WFs ported on SSP during the first year do not need access to astrophysical data repositories: data are locally transferred where the application runs. On the other hand during Y2 we focused on WFs that require the access to data based on Virtual Observatory standards, as discussed below.

6.1.1 *SHIWA Platform usage in Y2*

In Y2 A&A adopts a different approach that envisage a new usage scenario. The following considerations motivate this approach:

a) The great amount of legacy software produced by the community in the last decades is a heritage that the community should not waste and that can be reused by new projects and research activities;
b) The Virtual Observatory has developed tools and services to access and share data;
c) In the framework of the WF4ever EU funded project[17], A&A has developed more than 50 workflows using Taverna and AstroTaverna plugin that allow to access and manipulate data using Virtual Observatory tools and services.
d) Most of the WFs developed in Taverna are the result of a successful cooperation among Astronomers and Computer Scientists. They are simple WFs that can be considered as building blocks of more complex operations.

AstroTaverna is a plugin for the Taverna 2.4 workflow management system. It integrates existing Virtual Observatory web services as first-class building blocks in Taverna workflows (e.g. search a registry, add found services to the workflow, Manipulate VOTables, convert coordinates, and so on). AstroTaverna has been developed in the framework of the WF4ever project.

Thanks to the experience done by the other communities involved in ER-flow, A&A became familiar with the idea of **meta-workflows**: Taverna developed workflows are used as compound services that can be combined together in workflows of greater complexity.

Following this approach the most important goals for Y2 are:

- Storing AstroTaverna WFs in the SHIWA Repository;
- Testing AstroTaverna WFs in the SSP;

[17] http://www.wf4ever.eu/

- Combing them into a complex and complete WF;

Moreover, it is important to share the WFs within the A&A community at large. Once the WFs have been properly documented and uploaded to the SHIWA Repository, it is possible to refer to these workflows and share them within the community and with other community projects. Providing a large set of building blocks facilitates this approach: new users are tempted to use the SSP because there they are free to combine the building blocks into the WFs they need.

In this case the following steps are taken:

1. Workflows are developed and executed in their native workflow systems.
2. Workflows are uploaded to the SHIWA Repository with executable, metadata, documentation and sample and test data.
3. The direct link (URL) is used to refer to these workflows externally.
4. Some other person clicks on the link (URL) and is taken directly to the page about that workflow in the SHIWA Repository.
5. This new user can import the WFs it into his own local execution environment available: in particular the StarNET federation offers a set of a community portals the user can access to test and run the WF.

6.2 ***References***

1. U. Becciani et al. Science gateway technologies for the astrophysics community. Concurrency and Comp: Practice and Experience, 2014, DOI: 10.1002/cpe.3255
2. U. Becciani et al. Creating gateway alliances using WS-PGRADE/gUse. In Science Gateways for distributed computing infrastructures. Chapter accepted to SCI-BUS book, Springer.

7 Computational Chemistry

As already described in D5.1, the Computational Chemistry community, MoSGrid, consists of three application domains, namely molecular dynamics, quantum chemistry, and docking, which cover the diverse disciplines of computational chemistry in this community. During the first year, we developed and ported workflows from the three domains. In the second year, we continue this strategy but expand it to more complex workflows that even combine two domains.

After requirement analyses with our community, we designed more workflows, meta-workflows and even meta-meta-workflows of high utility. In Y1, we focused on WS-PGRADE workflows, in year 2 this is extended towards UNICORE workflows and the implementation of non-native workflows.

7.1 *SHIWA Platform Usage*

The end user uses the MoSGrid Science Gateway as an execution environment, which can be found at www.mosgrid.de. In general, MoSGrid standard users use pre-defined workflows within the specific portlets of their domain provided by the MoSGrid developers. In parallel, advanced users develop new workflows for their special needs. These users are also Chemistry experts who additionally have acquired distributed computing skills, and are the main users who reported experiences in Section 10.

Most workflows are based on WS-PGRADE; they can be stored on the MoSGrid repository and also exported to other repositories such as the SHIWA Repository. The MoSGrid repository is part of the MoSGrid platform. The advanced users can exchange their workflows via this repository.

In year 1 of the ER-flow project, the MoSGrid community became familiar with the idea of meta-workflows and learned to combine workflows. SHIWA usage by the MoSGrid community was mainly the advanced development of new more efficient workflows in all three domains and the porting to the SHIWA platform for future exchange and better interoperability with other chemical communities.

The real simulation will take place within the MoSGrid environment, but for application porting, the SHIWA Portal and the SHIWA Repository are needed. The workflow combination was done in the SHIWA Portal.

The most important usage types are:

- Storing MoSGrid workflows in the SHIWA Repository
- Exchanging MoSGrid workflows with other chemical communities
- Publishing MoSGrid workflows and making them available via SHIWA
- Combining small workflows into larger meta-workflows in the SHIWA Portal

7.2 *References*

1. S. Herres-Pawlis, A. Hoffmann, R. Grunzke, Lars Packschies, Orbital analysis of Oxo and Peroxo Dicopper Complexes via Quantum Chemical Workflows in MoSGrid, *CEUR Workshop Proceedings* **2013**, *993*, Paper 3.

2. S. Herres-Pawlis, A. Hoffmann, S. Gesing, J. Krüger, A. Balasko, P. Kacsuk, R. Grunzke, G. Birkenheuer, L. Packschies, User-Friendly Workflows in Quantum Chemistry, *CEUR Workshop Proceedings* **2013**, *993*, Paper 14.
3. R. Grunzke, S. Breuers, S. Gesing, S. Herres-Pawlis, M. Kruse, D. Blunk, L. de la Garza, L. Packschies, P. Schäfer, C. Schärfe, T. Schlemmer, T. Steinke, B. Schuller, R. Müller-Pfefferkorn, R. Jäkel, W. E. Nagel, M. Atkinson, J. Krüger, Standards-based metadata management for molecular simulations, *Concurrency Computat.: Pract. Exper.* **2014,** DOI: 10.1002/cpe.3116, in print.
4. S. Herres-Pawlis, A. Hoffmann, A. Balasko, P. Kacsuk, G. Birkenheuer, A. Brinkmann, L. de la Garza, J. Krüger, S. Gesing , R. Grunzke, G. Terstyansky, N. Weingarten, Quantum chemical metaworkflows in MoSGrid, *Concurrency Computat.: Pract. Exper.* **2014,** in print.
5. A. Hoffmann, U. Flörke, S. Herres-Pawlis, Insights into Different Donor Abilities Within Bis(pyrazolyl)pyridinylmethane Transition Metal Complexes, *Eur. J. Inorg. Chem.* **2014**, 2296 - 2306.
6. R. Grunzke, J. Hesser, J. Starek, N. Kepper, S. Gesing, M. Hardt, V. Hartmann, S. Kindermann, J. Potthoff, M. Hausmann, R. Müller-Pfefferkorn, R. Jäkel, Device-driven Metadata Management Solutions for Scientific Big Data Use Cases, PDP14, 2014, http://doi.ieeecomputersociety.org/10.1109/PDP.2014.119
7. R. Grunzke, J. Krüger, S. Gesing, S. Herres-Pawlis, A. Hoffmann, L. de la Garza, Improved Resilience and Usability for Science Gateway Infrastructures via Integrated Virtual Organizations, EGI Community Forum 2014, presented, https://indico.egi.eu/indico/getFile.py/access?contribId=93&sessionId=27&resId=0&materialId=slides&confId=1994.
8. Krüger, J., Grunzke, R., Herres-Pawlis, S., Hoffmann, A., de la Garza, L., Kohlbacher, O., Nagel, W.E., and Gesing, S., Performance Studies on Distributed Virtual Screening, BioMed Research International 2014, accepted.
9. Grunzke, R., Gesing, S., Jäkel, R., and Nagel, W.E. Towards Generic Metadata Management in Distributed Science Gateway Infrastructures, CCGrid 2014 (14th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing), May 2014, Chicago, IL, US, accepted.
10. Herres-Pawlis, S., Hoffmann, A., de la Garza, L., Krüger, J., Grunzke, R., Gesing, S., Weingarten, N., and Terstyansky, G., Meta-metaworkflows for Combining Quantum Chemistry and Molecular Dynamics in the MoSGrid Science Gateway, IWSG 2014 (6th International Workshop on Science Gateways), June 2014, Dublin, Ireland. Accepted.
11. Hoffmann, A., Herres-Pawlis, S., de la Garza, L., Krüger, J., Grunzke, Expansion of Quantum Chemical Metadata for Workflows in the MoSGrid Science Gateway, IWSG 2014 (6th International Workshop on Science Gateways), June 2014, Dublin, Ireland. Accepted.
12. S. Gesing, J. Krüger, R. Grunzke, L. de la Garza, S. Herres-Pawlis, A. Hoffmann, “The MoSGrid science gateway – tailored to the molecular simulation community”. Book chapter accepted to SCI-BUS book, Springer
13. J. Krüger, R. Grunzke, S. Gesing, S. Breuers, A. Brinkmann, L. de la Garza, O. Kohlbacher, M. Kruse, W. Nagel, L. Pachschies, R. Müller-Pfefferkorn, P. Schäfer,

C. Schärfe, T. Steinke, T. Schlemmer, K. Warecha, A. Zink, S. Herres-Pawlis, The MoSGrid Science Gateway – A Complete Solution for Molecular Simulations, *J. Chem. Theory Comput.* **2014**, in print, http://pubs.acs.org/doi/abs/10.1021/ct500159h.

8 Heliophysics

Heliophysics is a relatively new branch of physics that investigates the interactions between the Sun and the other bodies of the Solar System. Satellites, telescopes and other instrument collect data scattered around the Solar System, and this information is used to investigate the various physical events that originate on the Sun and propagate in space.

The most relevant outstanding issue for the Heliophysics community is how to clearly show in the SHIWA repository the relationships between the meta-workflows and its components. Discussion on how to best address this issue have started between the Heliophysics community and University of Westminster and are under way.

8.1 *SHIWA Platform usage*

In the second year of the project, the Heliophysics community has re-thought radically its approach to development and porting of workflows based on the experience gathered in the first year. The major problem that the community faced was that, being Heliophysics an event-driven discipline, the workflows may have a very limited lifespan and repurposing and flexibility of use become key features for success.

According to this new approach, the Heliophysics community has developed a multi-layered approach based on the concepts of meta-workflows and that of workflow interoperability. Small workflows have been developed in both Taverna and WS-PGRADE that address well-defined use cases that serve as building blocks for larger workflows. Larger workflows are composed with these building blocks to address science cases that describe well-defined scientific challenges. Finally, WS-PGRADE extensions of these meta-workflows are developed to execute meta-workflows as parameter-sweep jobs to investigate multiple phenomena. This approach has allowed the community to intensively test and assess functional and non- functional characteristics of both the SHIWA Repository and the SHIWA Submission Platform.

The infrastructure used by the community in Y2 and the areas where the most important testing has been brought on is summarized in Figure 6.

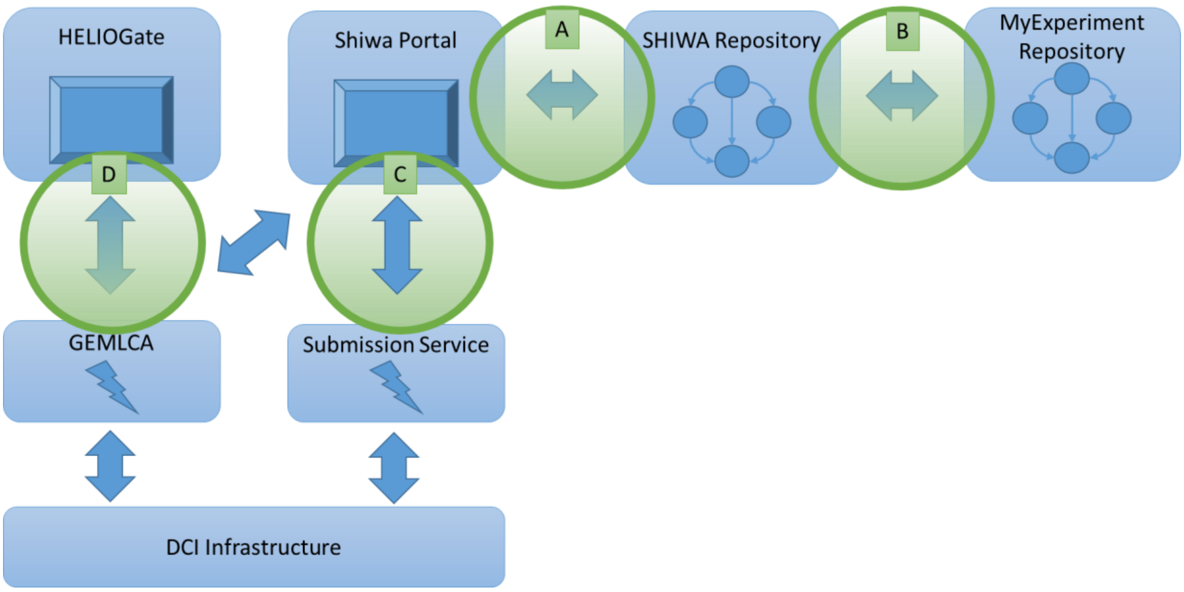

Figure 6 Heliophysics infrastructure used in Y2

Those areas are:

- **Porting and configuring workflows and meta-workflows to the SHIWA Repository**. This part of the system (described in Figure 6 by circle B) has been tested by porting and configuring 12 Taverna workflows to the SHIWA repository.
- **Configuring and executing workflows and meta-workflows using the SHIWA Portal**. This part of the system (described in Figure 6 by the circles A and C) has been tested by executing the ported Taverna workflows first using GEMLCA-based CGI implementation platform and, later to the new SHIWA Submission Service. Regarding the interface, a certain degree of redundancy was found in the interfaces and reduced after a feedback cycle between TCD and University of Westminster. This redundancy was particularly severe for workflows with many input and output ports. Minor bugs and usability issues have been also reported and fixed.
- **Installing and Configuring a GEMLCA Submission System**. TCD has installed a GEMLCA system to allow workflow interoperability on its HELIOGate portal[18]. The installation and configuration of the system has proven quite complicated and prevented the GEMLCA installation at TCD to reach production. To overcome these difficulties it has been decided to substitute GEMLCA with the new SHIWA Submission Service.

8.2 *References*

1. Dr. Gabriele Pierantoni, Dr. Eoin Carley, Dr. Jason Byrne, Dr. David Perez-Suarez, Prof. Peter T. Gallagher, "***A Workflow-Oriented approach to Propagation Models in Heliophysics"***. Accepted to Journal of Computer Science.
2. Dr. Gabriele Pierantoni, Dr. Eoin Carley, "***HELIOGate, a portal for Heliophysics"***. Book chapter accepted to SCI-BUS book, Springer.
3. Dr. Gabriele Pierantoni and Dr. Eoin Carley, "***Metaworkflows and Workflow Interoperability for Heliophysics***" accepted to IWSG14

[18] http://portal03.cs.tcd.ie:8080/liferay-portal-6.1.0

9 Life Sciences

Life sciences (LS) comprise the fields of science that involve the scientific study of living organisms, such as microorganisms, plants, animals, and human beings. The Life Sciences community in the ER-flow project is represented by the Academic Medical Center (AMC) of the University of Amsterdam. AMC has a large community of biomedical scientists who carry out research mostly based on the analysis of large data collections. This community focuses on biomedical research, which is a subfield of life sciences with the aim of better understanding the mechanisms of disease, how they manifest themselves in detectable ways, and how they can be influenced to treat the patient. Since 2005 the AMC has conducted e-science activities to exploit public infrastructures in the Netherlands for large scale data analysis. In particular scientists from three sub-domains have been involved in e-science activities: neuroimaging, biochemistry and genomics (next generation sequencing).

9.1 *SHIWA platform usage*

During the Year 1 of the ER-flow project the LS community has enabled a number of applications from neuroimaging and bioinformatics domains on the SHIWA Simulation Platform. The neuroimaging applications are executed from the customized interface of the neuroscience gateway, and the others can be executed from the generic interface of the AMC WS-PGRADE gateway.

The SHIWA services used in Y1 are

- Storing workflows in the SHIWA Repository
- Exchanging workflows with other communities
- Publishing workflows and making them available via the SHIWA Portal (limited)

Note that the SHIWA services are used mostly for sharing purposes. For actual data analysis, this community gives preference to customized interfaces at a higher conceptual level that uses terminology from the scientific area. Moreover, these customized interfaces (science gateways) handle data transfers and security, which are important factors due to privacy or intellectual property constraints

9.1.1 *Y2 approach and usage scenarios*

In the Year 2 the same strategy has been followed, with a few additions:

- Workflows are revised to improve robustness
- First experiments with cloud infrastructure are performed
- A few meta-workflows are developed in the SHIWA Portal with WS-PGRADE and Taverna components
- Additional customized interfaces are developed (additional portal for virtual screening)

Increasing Robustness

After the experiences in the first year, where workflow execution was not sufficiently robust to enable production level in the AMC gateway, we decided to invest on a new software installation platform that was developed by CERN and advertised by EGI via the user community board. The CernVM-FS (Cern Virtual Machine File System) is a network file system based on HTTP and optimized to deliver experiment software in a fast, scalable, and reliable way. Files and file metadata are aggressively cached and downloaded on demand. Thereby the CernVM-FS decouples the life cycle management of the application software releases from the operating system. The VLEMED VO, operated by the AMC, installed its own software repository, and also developed a test suite to verify the software installation conditions on the various clusters of the Dutch Grid. Many of the workflows developed during Y1 were modified to support distribution of required software by CernVM-FS. This can give more guarantees that execution environment is known, even on heterogeneous grid resources that normally can carry different versions of software libraries that might lead to incompatibilities and failure of workflow components.

Cloud Workflows

In collaboration with the SOMNO.Netz project, workflows were ported to run on OpenStack cloud infrastructure (see details in Section 11.1010.10)

Meta-workflows

During Y2 the new SHIWA Submission Service was introduced which helped solving long-standing problem of reaching MOTEUR workflow services (this issue arose after CNRS stopped supporting passwordless MyProxy service which was used by earlier submission subsystem in SHIWA platform). Currently local services running MOTEUR workflows (in particular, the service at the AMC) are reachable not only from the local gateway but also from the central SHIWA Simulation Platform, which makes it possible to create heterogeneous meta-workflows.

A new meta-workflow (Tracula) that combines three other applications ported in year 1 ((DTI Pre-processing, Freesurfer, and FSL BedpostX) is planned to be developed during Y2. We will also develop meta-workflows combining TAVERNA and WS-PGRADE workflows for analysis of DNA sequencing data. This work has not finished yet, therefore no feedback can be provided at this point.

Applications

In Year 2 new applications are introduced with new usage scenarios and technologies: cloud-based workflows; workflows using the network file system CVMFS; meta-workflows.

In essence, usage of SHIWA services by LS community is the following:

- SHIWA Repository is used extensively to store, re-use and share applications. One of the main benefits is that applications can be shared not only within a single SHIWA Simulation Platform but between all the connected local gateways.
- SHIWA Portal is used for development and execution of proof-of-concept workflows to be used on local gateways for production, as well as for development and execution of meta-workflows running over a set of workflow engines and/or infrastructures.

9.2 *References*

1. M. Jaghoori, A. van Altena, B. Bleijlevens, and S.D. Olabarriaga. A grid-enabled virtual screening gateway. In *Proceedings of the 6th International Workshop on Science Gateways*, June 2014.
2. S. Olabarriaga, M. Jaghoori, V. Korkhov, B.D.C. Van Schaik and A. van Kampen. Understanding workflows for distributed computing: nitty-gritty details. Proceedings of the 8th Workshop on Workflows in Support of Large-Scale Science (WORKS'13), pp 68-76, ISBN: 978-1-4503-2502-8
3. G. Terstyanszky, T. Kiss, V. Korkhov, S.D. Olabarriaga. Analyzing and Modeling of Medical Data on Distributed Computing Infrastructures. HAWAII INTERNATIONAL CONFERENCE ON SYSTEM SCIENCES (HICSS-47), January 6-9, 2014, Hilton Waikoloa, Big Island

4. J.L. Font, H.C. Lee, M.M. Jaghouri, S.D. Olabarriaga. Compatibility Check Framework for Software on Grid Computing Elements. EGI Community Forum 2014 at the Helsinki University, Finland. 19-23 May 2014
5. S. Shahand, M. M. Jaghoori, A. Benabdelkader, J. L. Font-Calvo, J. Huguet, M. W. A. Caan, A. H. C. van Kampen and S. D. Olabarriaga. Computational Neuroscience Gateway: a Science Gateway Based on the WS-PGRADE/gUSE Book chapter accepted to SCI-BUS book, Springer
6. S. Shahand, A. Benabdelkader, M. M. Jaghoori, M. al Mourabit, J. Huguet, M. W. A. Caan, A. H. C. van Kampen, and S. D. Olabarriaga. A Data-Centric Neuroscience Gateway: Design, Implementation, and Experiences. Concurrency and Computation: Practice and Experience, Published online, 2014.

10 ER-flow external communities

ER-flow approached several FP7 projects and research communities to support them in developing, publishing and sharing workflows and enable them to run non-native workflows. These communities are:

- Earth Science — Earth Server project
- Hydro-meteorology — DRIHM project
- Life Sciences — N4Y, Somno.netz project
- Seismology — VERCE project

ER-flow signed an MoU with the DRIHM and Somno.netz projects. All these communities, but the Earth Science and the Hydrometeorology communities, have been using workflow technology for a long time. The VERCE project develops Dispel4Py and WS-PGRADE workflows. All other communities and projects use only WS-PGRADE workflows. As a result, only VERCE is interested in the workflow interoperability.

The cooperation between the ER-flow and SCI-BUS projects is special because SCI-BUS partners and sub-contractors represent multiple research disciplines from Material Science to Meteorology, from Life Sciences to Physics. They have different backgrounds in using computing infrastructures and workflow technology.

10.3 *ER-flow external communities*

The Earth Science, the Hydrometeorology and the Seismology communities were at different phases workflow technology usage before starting the cooperation with ER-flow. The Earth Science and the Hydrometeorology communities were new to the workflow technology, while the Seismology community had a vast experience with it. To support these communities there are two major tasks:

1. To connect community gateways to the SHIWA Repository and test the workflow export-import operation.
2. To enable execution of non-native workflows based on the Coarse-Grained Interoperability (CGI) concept.

To support the first task, WP3 created a manual about how to connect a community gateway to the SHIWA Repository. The manual explains how to configure the community gateway to access the remote SHIWA Repository. The work package also extended SHIWA Tutorial #1, Creating and running native workflows, to explain how to use the workflow export and import operation. To help the second task, WP3 upgraded SHIWA Tutorial #2, Creating and running non-native workflows, and Tutorial #3, Creating and running meta-workflows. These tutorials explain step by step how define and execute non-native and meta-workflows. WP3 is working on extending the CGI support for the Dispel4Py workflow system used by the Seismology community in order to enable execution of Dispel4Py workflows as non-native workflows and combining them with WS-PGRADE workflows to build meta-workflows.

The **Earth Science community** selected sample applications in order to create workflows and test them on the SHIWA Simulation Platform. Currently, this community and WP5 are creating the first Earth Science workflows and testing them on the simulation platform. It's too early to get any feedback from this community.

The **Hydrometeorology community** deployed the DRIHM community gateway having support from the SCI-BUS project in 2013. The community developed WS-PGRADE workflows with WP5 support (MTA-SZTAI and UoW). These workflows are stored in the

Workflow Repository of their community gateway. DRIHM and WP3 connected the DRIHM community gateway to the SHIWA Repository and they are testing the workflow export-import operation.

The **Seismology community** deployed the VERCE community gateway having support from the SCI-BUS project in late 2013. The community developed WS-PGRADE workflows with WP5 support (MTA-SZTAI and UoW). These workflows pre-process seismology data. They are stored in the Workflow Repository of the VERCE gateway. The seismology community wants to publish and share these workflows. To allow it, VERCE and WP3 are connecting the VERCE community gateway to the SHIWA Repository and they are exporting-importing test workflows.

10.4 *SCI-BUS partners and sub-contractors*

ER-flow and SCI-BUS signed a Memorandum of Understanding in Year 1 of the ER-flow project to support cooperation between these two projects. The SCI-BUS partners and sub-contractors deploy and run community gateways based on the WS-PGRADE/gUSE gateway technology. They connected these gateways to computing infrastructures used by the research communities. There are two major user groups: workflows developers and domain researchers. Workflow developers create and run WS-PGRADE workflows on these gateways using the power user view. These communities do not need workflow interoperability support. They want to publish workflows and make them available for their own communities. To enable workflow publishing and sharing, MTA-SZTAKI and UoW provided support to connect these gateways to the SHIWA Repository. The SCI-BUS workflow developers uploaded 32 workflows to the SHIWA Repository using the workflow export operation. See details in Table 4 and Table 5. Domain researchers use either the end user view or ASM based customised portlets. Both user groups use the workflow import operation to download workflows to their gateways.

gateway	partner	developer	workflows
Statistical gateway	METU	Celebi Kocair	SSSG-AU SS1 SS2 SS3 SS4 SS5 SS6 SS7
Swiss biology gateway	ETH	Bela Hullar	LFQ LibraryCreation OpenSwath Rosetta TPP
Citizen gateway	EGroup	Aron Szabo Joaquin van Schoren	EGROUP ChartExMiner

Table 4 List of workflows uploaded by the SCI-BUS partners

gateway	partner	developer	workflows
Meteorology gateway	The Meteorology Group of Universidad de Cantabria	Jose Carlos Blanco	preprocessorECMWF
Meteorology gateway	Ruđer Bošković Institute Croatia	Davor Davidov	get_geog_cro-ng wrf_cro_ngi wrf_complete
Bioscience gateway	NVG Scientific Sdn Bhd, Malaysia	Elisabeth Chia Muhammad Farhan	AMBER-MD Blast Blender CLUSTALW GROMACS MUSCLE NWCHEM FASTDNAML OpenFoam PHASE
Physics gateway	Institute of Physics Belgrade (IPB)	Dusan Vudragic	AEGIS_CMP
Material Science gateway	G.V. Kurdyumov Institute for Metal Physics, Ukraine	Yuri Gordienko	CFG-MD R_LOCAL

Table 5 List of workflows uploaded by the SCI-BUS sub-contractors

The SCI-BUS users were asked to evaluate the export-import operations between the community gateways and the SHIWA Repository, as well as the features and services of the SHIWA Repository. Workflow developers prefer the automatic workflow export/import mode to other modes. The export operation sends a workflow bundle, which contains the abstract and concrete workflow, its configuration and dependencies. To specify further information about the workflow, such as workflow description, etc., they have to use the Table View. As a result, they are familiar with both the Browse View and the Table View of the SHIWA Repository. They consider the Browse View good, but find the Table View too complicated. The workflow developers liked the four basic operations: export, search, delete and import workflow, but they raised some issues with the workflow export and delete operation. They submitted a few enhancement requests:

- To concentrate command options because they are often scattered among several buttons and tabs.
- To make "basic attributes" more easily accessible and hide advanced attributes.
- To improve importing data, especially ports-related information because currently it is too difficult. Menu closes every time when I added a port or a dataset and it has to be re-opened every time. That makes it easy to re-edit an existing port by mistake. It would be nice if uses would have an open list with all the ports, and examples data sets as a matrix and that the system would save all the data automatically
- To extend page with bundles with search capabilities.
- To announce the gUSE/WS-PGRADE version that supports automatic workflow export to the SHIWA Repository.
- To submit requests for new workflow category/domain via a web form to the repository administrator, instead of writing and sending email to the repository administrator.
- To upgrade the workflow manual as soon as there is a new portal and repository version with new repository specific functions and operations.
- To provide the similar buttons to download the whole workflow in ZIP-ed file with all available dependencies and input data.

10.5 *SOMNO.Netz project*

In sleep research large amounts of patient data are collected during sleep studies, also called polysomnography data. In the course of the SOMNO.Netz project a data archiving and processing infrastructure is provided, to manage the data of 300 sleep laboratories in Germany. The infrastructure is based on the XNAT[19] as an archiving tool for medical data and on an OpenStack private cloud for scalable data processing. The processing itself is done using MATLAB algorithms executed on the cloud environment.

As a result of dissemination activities carried out by the Life Science community represented by the AMC, the SOMNO.Netz user community signed an MoU with the ER-flow project during Y2. This new community is interested in executing workflows on a cloud infrastructure, and has been supported by the AMC in the first exploratory path using the WS-PGRADE system.

One of the major roadblocks was related to connecting their own private cloud located in Berlin, Germany, to the SSP. In the SSP, the access to cloud resources is possible in two different ways: either using CloudBroker services or using direct cloud access implemented in gUSE.

The standard CloudBroker services turned out not be an option, as there is no connectivity between these services and the private cloud resources. Although CloudBroker proposes a commercial solution to deploy their services locally in Berlin, in the range of direct connection to the private cloud, this alternative has not been fully investigated yet.

The second option of direct cloud access from gUse required installing local community tailored portal (supporting SHIWA technologies) on the border of the private and public networks. A local WS-PGRADE portal has been deployed in Berlin, with the possibility to connect to central SHIWA services (e.g. SHIWA repository, to download and upload workflows), contact cloud resources in the private network and to be reachable externally. Several workflows have been developed utilizing different patterns of data management to implement SOMNO.Netz applications.

By introducing gUse and SHIWA technologies to the SOMNO.Netz architecture, some new useful functionality became available. The most obvious is the introduction of workflows, which handle the inputs and outputs of multiple dependent jobs. Although the sample application used throughout the experiments does not need long processing times, it is important to keep a good scalability for more sophisticated workflows in mind. Having a workflow system, which can resume failed jobs without the need of running all previous jobs again, can increase the productivity.

The second important feature is the handling of cloud resources, which is done by gUSE. With the functionality of the DCI-Bridge cloud plugin, cloud resources can be handled efficiently by reusing running VM instances for multiple jobs. It reduces the execution times caused by booting up new VMs.

Some feedback on the platform based on this work is provided in Section 11.10~~10.10~~, together with feedback from the Life Science community. In essence, more flexible error handling mechanisms would be desired on the platform, and a number of possible cloud management enhancements could be useful for better data handling and VM management..

[19] Open source imaging informatics platform, developed by the Neuroinformatics Research Group at Washington University; http://www.xnat.org

10.6 *StarNET gateway federation*

The STARnet Gateway Federation is a federation of A&A oriented science gateways designed and implemented to support the A&A community and its peculiar needs. STARnet envisages sharing a set of services for authentication, a common and distributed computing infrastructure, data archives and workflows repositories. Each gateway provides access to specialized applications via customized workflows, the first implementation of STARnet provides workflows for cosmological simulations, data post-processing and scientific visualization. These applications run on local or shared computing infrastructures, thus guaranteeing resources availability, and they can be shared between the different communities of the federation, either being published worldwide for dissemination purposes or kept locally for privacy issues. Users can then execute these workflows in an interactive and user-friendly way by means of the supplied web graphical user interfaces (portlets) endowed with gUSE Application Specific Modules API to manage them.

The current STARnet architecture is developed by means of virtual machines containing the WS-PGRADE/gUSE gateway installation and proper configuration for each STARnet site. The use of a virtual machine and the modularity of the WS-PGRADE/gUSE framework ensure easy set up and maintenance of the overall infrastructure, while preserving reliable computational performance because all the heavy simulations and computational tasks are performed on the linked DCIs.

The flexibility of the SSP allows implementing easily the federation and its integration with common services.

The constitution of the federation allows a great opportunity for scientists to share their scientific expertise among different fields and to train the user community under the same technological principles. This reduces the training time to get used to the web framework and all the gateway facilities to run their applications to use diverse computing facilities without dealing with any technical details.

11 Discussion and Recommendations

In this section we present the experience when using the SHIWA services (workflow repository, submission service, portal) to carry out the envisioned usage scenarios during Y2. Both functionality and usability aspects are considered, such as user-friendliness and intuitiveness. The communities comment on the following points: development environment, services & functionality, user interface, scalability, and reliability.

11.7 *Astrophysics*

During Y1 of ER-flow the porting of workflows was mainly limited to importing and exporting them to/from the SHIWA Repository. In Y2 A&A tests the WFs and focusses on meta-WFs. A&A also identifies 3 different types of users:

1. *Workflow developers*;
2. *Expert users:* Astronomer and/or Astrophysicists with a special interest in workflow technology that are willing to learn the use of the SHIWA platform, run single WFs, combine WFs into a meta-workflow.
3. *Users* that will execute the workflows through SGW web interfaces.

One of the main difficulties A&A faced concerns the interaction with gLite resources. The setup of domain specific Virtual Organization required a systematic interaction with portal experts from WP3, and it involved also the support of the gLite Italian Regional Centre. After a long debugging activity A&A managed to support all the Virtual Organizations.

Regarding the meta-workflows, all the AstroTaverna WFs are stored in the myExperiment repository. The porting process in the SHIWA Repository from myExperiment is straightforward, even if A&A needed the help of the Heliophysics community to complete the import procedure. In fact, after importing the WFs it is necessary to set all the execution parameters to enable the AstroTaverna workflows for execution. This task is a non-trivial operation and it needs a bit of time to get used to.

Moreover, the execution process in the simulation platform poses certain difficulties to the users that are not familiar with the system. For example, although it has solid theoretical background, the overall concept of abstract and concrete workflows is not intuitive to a final user and takes some time to get acquainted to. Moreover, after importing non-native workflows from the repository, the user must set up the parameters again and this implies a good knowledge of the imported workflow and application.

Those difficulties arise in particular when *expert users* (that do not request the support of WF developers) want to use the infrastructure. Also new *users* that are willing to test the SSP will encounter this problem.

More generally, a SGW is designed to hide the complexity of the infrastructure, however sometimes some more information may be useful, for example:

- there is no information on the progress of the execution of the imported WFs: this is annoying in case of long running non native WFs;
- there is no debugging information of the execution of the WFs: it's maybe a problem when using gLite or more in general DCIs.

11.7.1 *Lesson Learned*

The experiences collected during the project allow focusing some general considerations on the use of science gateways and SSP.

- Due to the complexity of a grid infrastructure, and more in general of DCIs, it is difficult to support more than one virtual organization in the same science gateway.
- Extended debugging information should be available for WFs developers for troubleshooting. Maybe it could be useful to provide a different view (interface) on the system for workflow developers;
- To enhance WFs sharing it is useful to provide some simple WFs examples and associate them a demo video. This approach could be useful also to support new users.

11.8 *Computational Chemistry*

As overall experience, the SHIWA portal is visually fine and working. However, the basic feature that one needs two sets of logins for Repository and Portal is complicated to understand by chemistry-oriented workflow developers. The two sets of logins and passwords pose still numerous small problems that can be solved, but should be facilitated for the user. The authentication in principle is fine, since it is similar to the certificate authentication used in the MoSGrid gateway.

After a long learning curve to port workflows, the chemistry-oriented workflow developers became familiar with the porting process of standard workflows. The direct upload of the MoSGrid-workflows into the SHIWA Repository (which was wished in D5.1) is now possible and works very fine. Moreover, the workflow export from the SHIWA Repository is quite easy and re-import to MoSGrid works fine.

Unfortunately, the development of meta-workflows was hampered by multi-levelled problems that needed a longer time period to be solved by the chemistry-oriented workflow developers together with the Westminster support team. The strategy is the so-called *white-box approach*, which consists of several steps:

- workflow definition as graph
- implementation of a concrete workflow with the defined graph (including successful runs)
- building a template from the correctly running workflow
- implementation of a concrete workflow out of this template
- combination of the concrete workflow (from step d) into the larger meta-workflow

One should always check if the regarded sub-workflows are functional before combining them, because after portal upgrades some workflows give strange errors (they “forget” files, applications etc.). One should never assume that a functional workflow always stays so.

First, the sub-workflows must be prepared and tested. Here it is really important that the port names are absolutely correctly defined with regard to their final role in the meta-workflow. When an error occurs here, we found that this might only be recognized at the end of the workflow building process when the output port names of sub-workflow and of meta-workflow do not fit together. Then one has to restart with correction of the graph, new definition of the concrete, definition of template, definition of concrete from template and combination to meta-workflow. In the next step, the meta-workflow needs to be defined. Here it is crucial that the input-port 0 of the Job0 of the sub-workflow must receive an input file from the input-port 0 of the sub-workflow of the meta-workflow. The output-port 1 of the Job1 of the sub-workflow passes its output file to the output-port 1 of the sub-workflow of the meta-workflow.

During the implementation, we encountered several problems:

- the workflows in MoSGrid “forgot” their application (gUSE upgrade problem);

- the correct template definition out of the functional workflows failed. Here the chemistry-oriented workflow developers needed some time to get acquainted with this concept.
- the meta-workflows completed, but produced non-reproducible output which was rather confusing.
- the meta-workflows completed, but produced no real output. Here it is really difficult to find the error message because the sub-workflows were successful. In the end, we found that it was a port definition problem in the template definition.

All problems together gave a strange mixture that was extremely difficult to resolve since multiple errors in different levels of meta-workflow definition occurred. This situation could only be clarified by joint efforts of the portal administrators, the SHIWA support team and workflow developers. We could solve these problems via a MoSGrid portal upgrade performed by J. Krüger and Luis de la Garza at Universität Tübingen together with R. Grunzke. The meta-workflow definition problems could be solved by N. Weingarten and S. Herres-Pawlis. N. Weingarten wrote a comprehensive tutorial for the white-box approach for metaworkflow definition which has been tested and amended by S. Herres-Pawlis. One highly critical and error-prone point was the definition of templates of sub-workflows and creating concrete workflows out of these templates. With the tutorial, this step is now much clearer.

11.8.1 *Enhancing the Platform*

The most important developments needed for efficient SHIWA usage are the following:

- Introduce a facilitated access to SHIWA Portal and Repository: the easiest would be a single-sign-on.
- Implement the automatic graph creation feature. This feature was really helpful, but has disappeared since March 2013.
- Implement a more user-friendly workflow creation
- Implement a workflow configuration that:
    - is more guided;
    - has a more clean and straight forward user interface
    - makes it more difficult to make errors
    - does not forget properties
- Provide UNICORE support to enable integrated development and testing of MoSGrid workflows from within SHIWA Support and integrate European federated AAI infrastructures to enable scientists to login via their already existing university logins
- Increase the resilience of the platform.

11.9 *Heliophysics*

During Y2, the Heliophysics community developed and tested several workflows:

- **Porting and configuring workflows and meta-workflows to the SHIWA Repository**. The main feature of the system that the Heliophysics community asked to ameliorate is the redundancy of information request during the porting process. This redundancy made it difficult and cumbersome to port workflows, especially those that had multiple input and output ports. A more detailed feedback on this part of the system was provided by the HELIO community and is attached to this deliverable as Appendix A
- **Configuring and executing workflows and meta-workflows using the SHIWA Submission Service.** The reliability of the new submission service has been tested with multiple executions of small and larger workflows; this service was reliable enough to be put into production after a few final changes. The submission service superseded the previous GEMLCA submission service. A more detailed feedback on

this part of the system was provided by the HELIO community and is attached to this deliverable as Appendix B

11.10 *Life Science*

During the Y2 the LS community used SHIWA services mostly for sharing and re-using workflows. Normally LS users prefer to use dedicated gateways with customized interfaces specific to the scientific domain. Moreover, these customized interfaces (science gateways) handle data transfers and security, which are important factors due to privacy or intellectual property constraints. However, the need to share and re-use is high, so the services provided SHIWA Repository are highly required.

SHIWA Repository provides a wide range of capabilities and has evolved a lot during the last year. In the latest release of the SHIWA platform the connection between SHIWA Repository and portal became much stronger. This is a major achievement that allows to browse the repository right from the portal and select workflows to be imported in much simpler way. However some drawbacks still persist:

- One of the major drawbacks is missing functionality to export new workflow implementation from WS-PGRADE gateway as a part of already exiting workflow in the SHIWA Repository. Currently new workflow entry is created every time a workflow implementation is created.
- The WF hyperlinks are big and a bit cumbersome. A simpler way to share a stored WF, like http://shiwa-related-domain/WF-ID would be helpful. For example, MyExperiment registered WF's are a lot easier to be referenced, in the form: http://www.myexperiment.org/workflows/3355

Large part of Y2 new experience in using WS-PGRADE/gUSE, which forms the core of the SHIWA platform, was gained during experiments in collaboration with the SOMNO.Netz project, which enabled applications on the cloud.

11.10.1 *Enhancing the Platform*

Error handling and monitoring in the SHIWA Portal still needs improvement. This is known for gLite resources, but the same also applies to cloud support:

- Lack of monitoring and error handling information: WS-PGRADE/gUSE lacks of some general and some cloud specific functionality, which could improve the efficiency of working with this system. In general gUSE does not provide enough information about errors that occurred. In addition, to debug and control the cloud integration it is necessary to get as much real time information as possible. However, application error logs can be downloaded In the web interface and the exit code of the application is given.

Cloud functionality possible enhancements:

- Data transfers: instead of transferring the data back to the centralized gUSE server, it would be more suitable to transfer the data directly between the Virtual Machines. Also dependent follow-up jobs should be executed in the same Virtual Machine if possible, so that no data transfer is necessary. We hope that the recently released Data Avenue software can help solving issues of this kind with data management in the future.
- Monitoring features: visualization of running and used Virtual Machines; number of jobs waiting in line to be executed together with VM they are assigned to;

- VM lifecycle management: gUSE always shuts down Virtual Machines after a certain time, even if there has been a problem with the application or gUSE. An option to keep VMs alive in case an error occurred could be useful.

11.11 *Common Issues and Common Recommendations*

During Y1 and Y2 the communities involved in ER-flow tested and used extensively the SHIWA platform and the repository. The communities that participate to ER-flow project have specific and unique characteristics, as clearly illustrated in this document.

Regarding the workflow systems used, both Astrophysics and Heliophysics use a combination of Taverna and WS-PGRADE workflows to access a varied computing and data infrastructures. The Computational Chemistry is using MosGrid science gateway as execution environment and the meta-workflows approach on the SSP. The set-up of the Life Sciences community is based on WS-PGRADE and the Dutch grid infrastructure. New communities are willing to use Cloud Computing infrastructures.

Also the types of workflows differ within and among the communities: some are data oriented, others are compute-oriented; some perform long computations (e.g. parameter sweeps) and others perform short data manipulations. The duration, patterns and usage scenarios of the workflows also vary a lot. Some workflows are executed only once (e.g. in Heliophysics experiments), whereas others are repeated for numerous input data (e.g. biomedical data analysis experiments).

From a workflow management perspective, these communities have diverse requirements, however they identify at least two main user profiles:

1. *Users*. They are mainly scientists that are most interested in the scientific results that a workflow generates, and may have just marginal roles in the development of workflows, applications and gateways.
2. *Workflow developers*. They design and develop workflows that can be later used by themselves or by *users.*

Moreover, the communities identified common issues in using the SSP. The common issues described in this document and extensively discussed in the communities' related subsections. We focus here on four main aspects:

1. The usability and friendliness of some aspect of user interface. This is particularly relevant for users and workflow developers.
2. Workflows manipulation. The communities identify the complexity of some operations and procedure that are related to the export, import, creation, modification and setting parameters on workflows.
3. Transparency of the SSP Portal. In particular the communities highlight the lack of a more informative error messages (useful for users and workflow developers) and a more verbose debugging output (useful for workflow developers and for science gateway operators).
4. Documentation of the SHIWA technology

The main community specific issues have been summarized in Table 6..

Community	**SHIWA Portal**	**SHIWA Repository**
Astrophysics	Setup of grid enabled SGWs requires systematic interaction with technology experts.	After importing non-native WFs the configuration and parameters setup requires a good knowledge of the workflow and the application itself (redundancy of entries).

Computational Chemistry	Meta-workflow creation procedure is complex and not transparent. Workflow configuration is not very well guided and does not present with a clear and straightforward user interface.	Single Sign-On for SSP Portal and Repository would be useful.
Heliophysics	Ameliorate the redundancy of information request during the porting process of meta-workflows	
Life Sciences	Lack of monitoring and error handling information: level and details of error messages are often not enough to track the source of problems.	It is not possible to export a new workflow implementation as a part of existing workflow in the Repository. Better integration of Portal into Repository giving the possibility to browse the workflows straightforward and import them. Shorter workflow urls.
DRIHM		
SOMNO.Netz	Cloud functionality: multiple issues in gUSE/WS-PGRADE frameworks.	

Table 6 Summary of identified issues of the SHIWA platform by communities

The experiences gathered so far by the scientific communities described in the previous sections is telling about the challenges and solutions that are met in the daily management and development of their applications and infrastructures based on the SSP.

It is very important to observe how communities that different wildly in size, technology and usage patterns had identified a set of common recommendations to improve the SSP.

Besides the known requirements for supporting scientist *users* that were successfully addressed in Y1, during Y2 additional recommendations emerge to support the development and operation of science gateways that are based on the workflow system.

It is necessary to implement new interfaces for workflow developers to simplify the debugging and troubleshooting activities. While workflow developers and science gateway developers provide comfortable front-end for *users,* in the same way the SSP should provide front-end specific for *workflow developers* that allows exploring a detailed error messaging system and a user-friendlier workflow manipulation and creation interface.

A more efficient debugging front-end will positively affect also the work of the science gateway operators that configure and maintain the portal. On the back-end, the gateway may connect to different computing and storage services (e.g. grid, local cluster, cloud infrastructure). The operator takes care of monitoring the services and troubleshooting. For this reason implementing a more user-friendly error messaging system may simplify the activity of operators when problems regarding access to computing resource are encountered. This is particularly important when using DCIs and Cloud Computing infrastructures.

As Cloud Computing is becoming more and more common for scientific computing, also the Cloud functionality support on SSP should increase and become more “workflow developer-friendly”. In particular it is necessary to identify open source solutions to access private clouds and federated clouds.

This is particular important as workflows and science gateway are valuable tools to explore and analyse large data sets. The combination of Big Data (driven in part by mobility and social media) and cloud computing is exponentially expanding the amount and types of available data; the SSP has the potentiality to become a unique tool to approach this kind of problems.

12 Conclusions

The SHIWA platform has evolved and improved enormously since the beginning of ER-flow project. This was possible because of feedback provided by users, because of the tools and services that allow the users and the developers to communicate efficiently (e.g. SHIWA user forum) and, most importantly, due to a technology provider team committed to support and improve the platform.

The SHIWA platform is a complex system that is difficult to develop, test and maintain. This is natural, giving the number of different services that are necessary under the hood to implement the vision of the CGI workflow interoperability. Many problems are detected by combinations of situations that are difficult to test and sometimes even to reproduce when developing and testing the platform. Moreover, during the Y2 communities develop more complex workflows and science gateway that involve Grids and Cloud Computing infrastructures, and they explore more complex usage scenarios.

When the project was started, the idea of using the customized gateways to interface with the scientists, instead of directly using the SHIWA Portal, was not very welcome because it deviated from the original usage scenarios considered during the SHIWA project and the ER-Flow project proposal. At the end of Y2 we see that this new approach opened up many other interesting possibilities, for example, for the construction of federations of gateways as proposed by the Astrophysics community implemented in the StarNET project.

Moreover, science gateways are now able to interface to new resources giving more possibilities to user communities. This of course opens new patterns in the use of SSP and implies new issues and requirements. However, understanding patterns and requirements of these cases is important to indicate future directions of further improvements of SSP as we did during Y1.

Appendix A Importing and executing a TAVERNA workflow for test.

Introduction

The user wants to execute a TAVERNA workflow as a test. If the workflow in myExperiment has already the example data set, in theory he should be able to execute without giving any other information but she/he has to enter information on input and output 4 times.

1. Enter/validate the input/output as parameters in the workflow. The consequences of selecting file instead of string or other value are not clear to the user at that stage (As an example the type of SEN seen from the portal will change, no input ports for values, input ports for files)
2. Enter/validate the datasets in the workflow.
3. Enter/validate the datasets in the implementation
4. Enter/validate the input/output in the execution portal.

Some steps of this process are not clear to a user, more specifically:

1. The difference between the data sets in the abstract workflow and in the implementation.
2. The consequences of selecting file or value (string, integer, etc) as inputs. This changes how to select the command line parameters in the implementation, forces to upload test files, and modifies the number of input ports of the SEN from the portal.

Detailed description

Step	Where	Information Required	Comments
Import from myExperiment	Repository	myExperiment ID	Easy and intuitive
Set Parameter	Details	--	Easy and intuitive
	Owner	--	Easy and intuitive
	Access	Group/privileges/visibility	To a new user, it is not clear how to set the group name
	Attributes	Input / Output and Datasets	To a new user, it is not clear that by selecting files, it is then necessary to provide example files. To the new user it is not clear that by not selecting file, the number of input ports will change when selecting the sen from the portal. The dialog box of output for dataset is really confusing. How can I select an output file of a workflow that has not been produced yet?
Implementation	Parameters	Input / Output	This information has already been provided with the datasets. It would be nice if there was the possibility to directly use it. For a user that has not read tutorials it is impossible to understand how to fill in the switch name

			fields. This information is already included in the taverna bundle. Edit command line and Edit fixed is not really clear.
	Toggle executable	--	To a new user it is not clear the importance of this step. A big button like "Validate and make executable" would be better.

Submission Service

Once the problems with the firewall on the execution system have been solved, the submission were 100% successful (on 78 executions). 3 different workflows were used for these tests (CME Propagation, HEC Query and SimpleTest).

Appendix B SHIWA Repository and Submission Tests

Version	Date	Author	Notes
1.0	7/2/14	Pierantoni	First draft

Introduction
This is the first draft of the results of the tests performed by the HELIO community of the SHIWA Repository (3.1) and Submission Service

Test details

SHIWA Repository	http://161.74.26.14:8080/shiwa-repo/
Liferay Portal	http://161.74.26.58:8080/liferay-portal-6.1.0/
Execution Service	PreDeployWestFocus
Tester	Gabriele Pierantoni (pierang@cs.tcd.ie)
Client environment (OS)	Ubuntu 11 and Windows7
Client Browser	Chrome
Time Frame	From 28.1.14 to 05.2.14

Tests Performed
We tested two main functionalities of the system:

- Porting of workflows from myExperiment
- Execution of the ported workflows on the pre-deployed west focus computation facility.

Results
The tests are still being executed these are only preliminary and incomplete results.

Functionality of the porting process
The tests on the porting process highlighted two bugs (36 and 37)

Usability of the porting process
From the limited perspective of a user that wants to port and execute TAVERNA workflows (thus unaware of design constraints and the intricacies of a multi-workflow environment), the process has improved but still requires entering the same information multiple times. This can be difficult and confusing.
Ideally, the TAVERNA file in myExperiment contains most of the information needed to configure the ported workflows on the repository and to create the wrapping WS-PGrade node and execute it. We stress again: this is a user view and does not take into account design and implementation restrictions that may be the cause of duplication of information. The following table recapitulates the user view of the information flow of the porting and execution of a simple TAVERNA workflow.

Service	Step	Information	Notes
Repository	Import from myExperiment	NA	Simple and intuitive. OK
Repository	Set Attributes of wf	Input and Output ports	Ported from myExperiment. OK
Repository	Set Attributes of wf	Data sets	The data sets values (and not just the names) can be inferred from the example value of myExperiment.

			The use of the same interface to define input and output is confusing.
Repository	Set Files of wf	NA	Not really clear what files are these.
Repository	Implementation	SEN parameters	There should be a sintax suggestion on how to pass params as command line.
Portal	WS-PGRADE wrapper	Wrapper Workflow	The WS-PGrade workflow can be automatically defined as it is determined by the number of IO ports of the SEN
Portal	WS-PGRADE wrapper	IO ports	The default values can be determined from the SEN

Reliability of the submission service.
Only a limited set of tests have been performed. To isolate the functionalities of the submission service tests have been performed on a simple, one node TAVERNA workflows with a simple IO composed by a single string.
The workflow has been submitted through the SHIWA Submission Service 50 times and succeeded 35 times with a success rate of 70%. Errors seems to have originated at the WestFocusPreDeployed execution layer and may not be linked to the submission service. Not all errors have been analysed.