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 D3.2
Extended Simulation Platform

Due date of deliverable: 30/06/2013
Actual submission date: 30/06/2013

Start date of project: 01/09/2012
Duration: 24 months

Lead Contractor: University of Westminster

Dissemination Level: PU

Version: Final

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

1 Table of Contents

2 List of Figures and Tables

2.1 Figures

2.2 Tables

3 Status and Change History

Status:	Name:	Date:	Signature:
Draft:	Gabor Terstyanszky	17/06/13	n.n. electronically
Reviewed:	Gergely Sipos	26/06/13	n.n. electronically
Approved:	Steve Winter	28/06/13	n.n. electronically

Table 4. Deliverable Status

Version	Date	Pages	Author	Modification
1.0	10/06	006	G Terstyanszky	Report template Section 5 Introduction
1.1	11/06	p07-09	G Terstyanszky	Section 6: CGI + ER-flow
1.2	12/06	p10-13	G Terstyanszky	Section 7: Communities and requirements
	12/06	p14	G Terstyanszky	Section 8: Enhancement requests
2.0	13/06		A Sasvari	Section 8: Domain management, community validation, concrete workflow URL
	13/06		B. Meihac	Section 8 new submission service, Taverna 2 CGI support, Taverna 2 workflow import from myExperiment
2.1	13/06		T. Glatard	Section 8: MOTEUR workflow export-import
	13/06		G Terstyanszky	Section 8: SHIWA Portal upgrade, workflow engine data management
2.2	14/06		G Terstyanszky	Section 8: access to gLite and UNICORE resources, repository content clean-up
	17/06		G Terstyanszky	Section 9: ER-flow execution environment
	19/06		T. Gottdank	Section 8: WS-PGRADE workflow export-import, robot certificate management, upgraded statistics service
	19/06		G Terstyanszky	Section 9: Conclusion
3.0	19/06		G Terstyanszky	Compiling the first complete version
3.1	26/06		G. Sipos	Internal review of the report
final	28/06		G Terstyanszky	Compiling the final version

Table 5. Deliverable Change History

4 Glossary

Axx	Major Enhancement Request
ASM	Application Specific Module
CGI	Coarse-Grained Interoperability
DCI	Distributed Computing Infrastructure
EGI	European Grid Infrastructure
FGI	Fine-Grained Interoperability
Ixx	Minor Enhancement Request
MTA-SZTAKI	Magyar Tudomanyos Akademia Szamitastechnikai Kutato Intezete
NGI	National Grid Infrastructure
SSP	SHIWA Simulation Platform
UoW	University of Westminster
VO	Virtual Organisation
WF	Workflow
WE	workflow engine
WP	Work package

Table 6. Glossary

5 Introduction

Workflows have become essential to integrate expertise of the application (user domain) and infrastructure domain (Distributed Computing Infrastructures - DCI) in order to support research communities. Workflows help e-scientists to formalize and structure complex scientific experiments to enable new scientific discoveries. Workflows represent, streamline and automate the analytical and computational steps that e-scientists need to go through from data selection and integration, computation and analysis to final data presentation and visualization. Research communities have developed different workflow systems and created large numbers of workflows to run experiments. These workflow systems have different workflow description languages, enactment strategies and middleware providing access to infrastructures. It takes a significant effort and time to learn how to use workflow systems, and it requires specific expertise and skills to develop and maintain workflows. As a result, creating, running and maintaining workflows need substantial efforts and expertise. E-scientists hesitate to learn new workflow systems to migrate their experiments to other workflow systems as this is a time-consuming and error prone process. They would prefer workflows sharing, i.e. automatic porting of workflows across workflow systems and DCIs to optimise their efforts. Currently, the major obstacle of workflow sharing is that workflow systems are not compatible and interoperable.

To address workflow interoperability the “Sharing Interoperable Workflows for Large-Scale Scientific Simulations on Available DCIs” (SHIWA) project [5.1] developed the Coarse-Grained (CGI) approach. SHIWA created and deployed a production-level CGI service, called the SHIWA Simulation Platform (SSP). The four research communities, involved in the ER-flow project use this production-level CGI service to create, integrate, share and run workflows. They and the technology provider project partners developed the ER-flow strategy to support e-scientists to create and execute workflows. The strategy defines two environments: development and execution environment. The development environment is the SHIWA Simulation Platform while the execution environment is the science gateways specialised for these four communities. These two environments are connected through two SHIWA services (the SHIWA Repository and SHIWA Submission Service), and through the gUSE/WS-PGRADE platform. The four research communities compiled a list of enhancement requests to upgrade both ER-flow development and execution environments based on their community specific requirements in October - November 2012. This list was presented in the D3.1 “Study of the adaptation options of the simulation platform” report. The research communities extended this enhancement list with further requests in March - April 2013.

WP3 analysed these requests and categorised them into major and minor enhancement requests. It created a schedule to implement these requests classifying them as short-, mid- and long-term requests. Major requests specify features which are essential for the research communities for example automatic workflow export-import while minor requests can significantly improve user experience for example domain name management. The work package started implementation of short-term requests. WP3 upgraded the SHIWA Simulation Platform with four major and eleven minor enhancements. WP3 participated in the upgrade of the science gateways of the research communities deploying some of these enhancements.

In Section 6 the report outlines the ER-flow strategy to support the research communities. It also gives a short overview of the ER-flow development and execution environment. In Section 7 describes the enhancement requests that the research communities and technology providers identified. In Section 8 the report presents completed and under development enhancement requests. Finally, in Section 9 we describe how the research communities upgraded their science gateways to be an ER-flow execution environment.

6 Using the Coarse-Grained Workflow Interoperability in the ER-flow Project

The SHIWA project targeted the challenges of (i) executing workflows of different workflow systems as non-native workflows (ii) combining workflows of different workflow systems into meta-workflows and (iii) running meta-workflows on different DCIs. It focused on workflow developers and the workflow system developers. The ER-flow project extended the target group to e-scientists. The technology providers (CNRS, MTA-SZTAKI and UoW) and research communities (Astrophysics, Computational Chemistry, Heliophysics and Life Sciences) involved in the ER-flow project evaluated the SHIWA Simulation Platform during the first three months of the project. They concluded that the simulation platform is an efficient and sophisticated workflow development environment, but it has some limitations as an execution workflow environment. Particularly, the SHIWA Portal offers two views: power and easy user view. Workflow developers use the power view to create abstract and concrete workflows, configure and execute concrete workflows through the graphical workflow editor and the relevant portlets. The portal generates the easy user view based on the configured concrete workflow. This view provides basic GUI to parameterise, execute and monitor workflows. To improve the e-scientists experience Application Specific Modules (ASM) can be developed to run workflows. ASMs can offer portlets customised according to requirements of e-scientists to parameterise, execute, monitor workflows and visualise results.

The four communities involved in the ER-flow project want to use the combination of easy user views and ASM based portlets to run applications. ER-flow will also support four additional external research communities. Considering the number of workflows they want to develop and execute it would be technically feasible to run all workflow through the SHIWA Portal but it would not have been reasonable. In this scenario having the SHIWA Portal as a single portal, it should manage easy user views of all workflows ported to the SHIWA Simulation Platform and ASM based portlets running of some selected workflows. The project consortium agreed to separate development and execution and to have two separate environments:

- ER-flow development environment with power user view and
- ER-flow execution environment with easy user view and/or ASM portlets.

6.1 ER-flow Development Environment

The development environment is based on the SHIWA Simulation Platform (SSP). It contains the SHIWA Science Gateway and provides access to VOs of different DCIs of the European Grid Infrastructure (EGI).

SHIWA Science Gateway contains a portal (SHIWA Portal), a submission service (SHIWA Submission Service or GEMLCA Service), a workflow repository (SHIWA Repository), and a proxy server (SHIWA Proxy Server) to support the Coarse-Grained Interoperability (CGI) concept. The **SHIWA Portal,** built on the gUSE/WS-PGRADE technology [6.1], provides the graphical user interface to create and execute workflow through gUSE web services and DCI Bridge. It is integrated with the WS-PGRADE Workflow System which is used as native workflow engine in the simulation platform. The **SHIWA Repository** manages workflow and workflow engine data and metadata. It offers a wide-range of browse and search features for e-scientists and workflow developers. The **SHIWA Submission Service** handles workflow engine data and metadata plus execution of workflows and workflow engines as legacy code applications. Workflow and workflow engine developers can describe, modify and delete workflows and workflow engines through repository and submission GUIs. To support non-native workflow execution the SHIWA Submission Service either invokes locally or remotely pre-deployed workflow engines or submits workflow engines with the workflow to local or remote resources to execute workflows. The **SHIWA Proxy Server** enables management of multiple certificates when the workflows are executed on different DCIs and VOs.

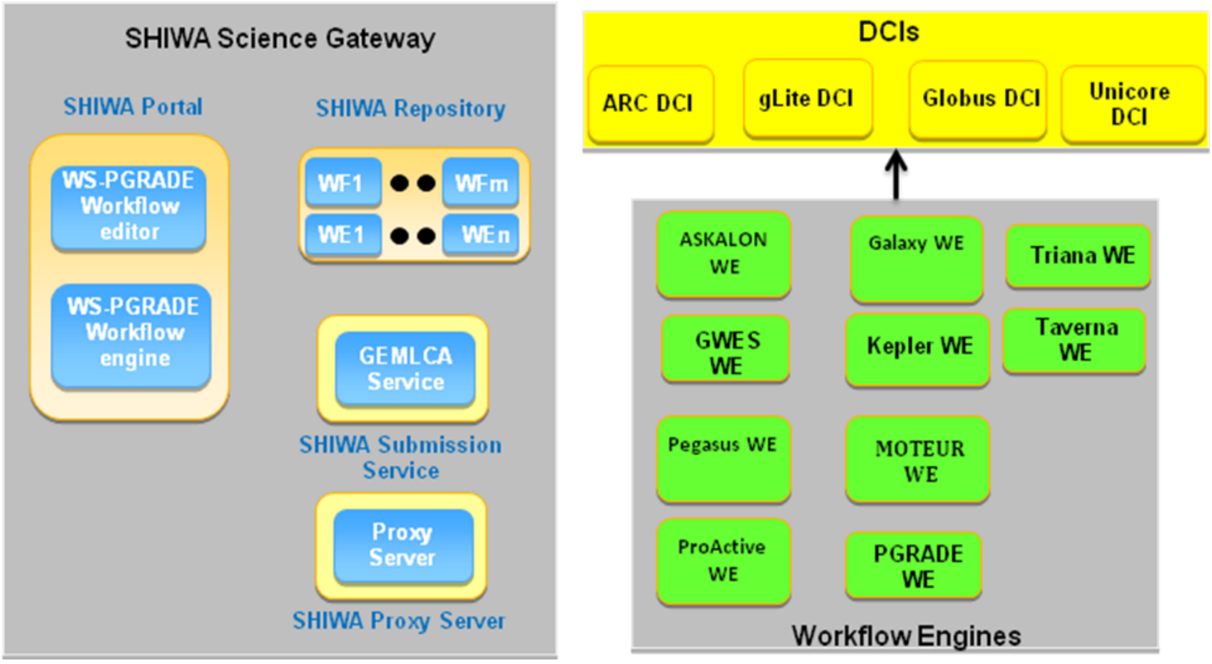


Figure 6.1: ER-flow development environment

where **WF_1,…WF_m** - workflow data and metadata,
WE_1,…WE_n - workflow engine data and metadata

Currently the simulation platform provides CGI support for the following **workflow systems**: ASKALON [6.2], Galaxy [6.3], GWES [6.4], Kepler [6.5], MOTEUR [6.6], WS-PGRADE [6.7], Pegasus [6.8], ProActive [6.9], Taverna [6.10] and Triana [6.11]. The SHIWA project consortium created the **SHIWA VO** to enable user authorization across the DCIs providing resources for the workflow execution. The project deployed the dedicated shiwa-workflow.eu VOMS server to allow access to the VO. The original SHIWA VO which consists of resources of the British, Dutch, French, Hungarian National Grid Infrastructures (NGI) was extended by resources of the German and Italian National Grid Infrastructures.

6.2 ER-flow Execution Environment

The Astrophysics, Computational Chemistry, Heliophysics and Life Science community deployed WS-PGRADE gateways within the framework of the SCI-BUS project. However these science gateways cannot be used as ER-flow execution environment. First, they are not able to support the CGI approach because they are not connected to the SHIWA Submission Service. Second, they do not provide the following services:

- SHIWA Repository export-import
- robot certificate handling and
- collecting workflow execution statistics.

Based on the Memorandum of Understanding signed by the ER-flow and SCI-BUS project three of the four communities upgraded their science gateways to WS-PGRADE 3.5.7 which offers the following new services:

- remote access to SHIWA Submission Service
- export-import workflows to/from SHIWA Repository
- robot certificate management and
- improved workflow execution statistics service.

The fourth science gateway (Computational Chemistry science gateway) will be upgraded in Year 2 because it has some specific MoSGrid features/services which cannot be transferred in a straightforward way to WS-PGRADE 3.5.7.

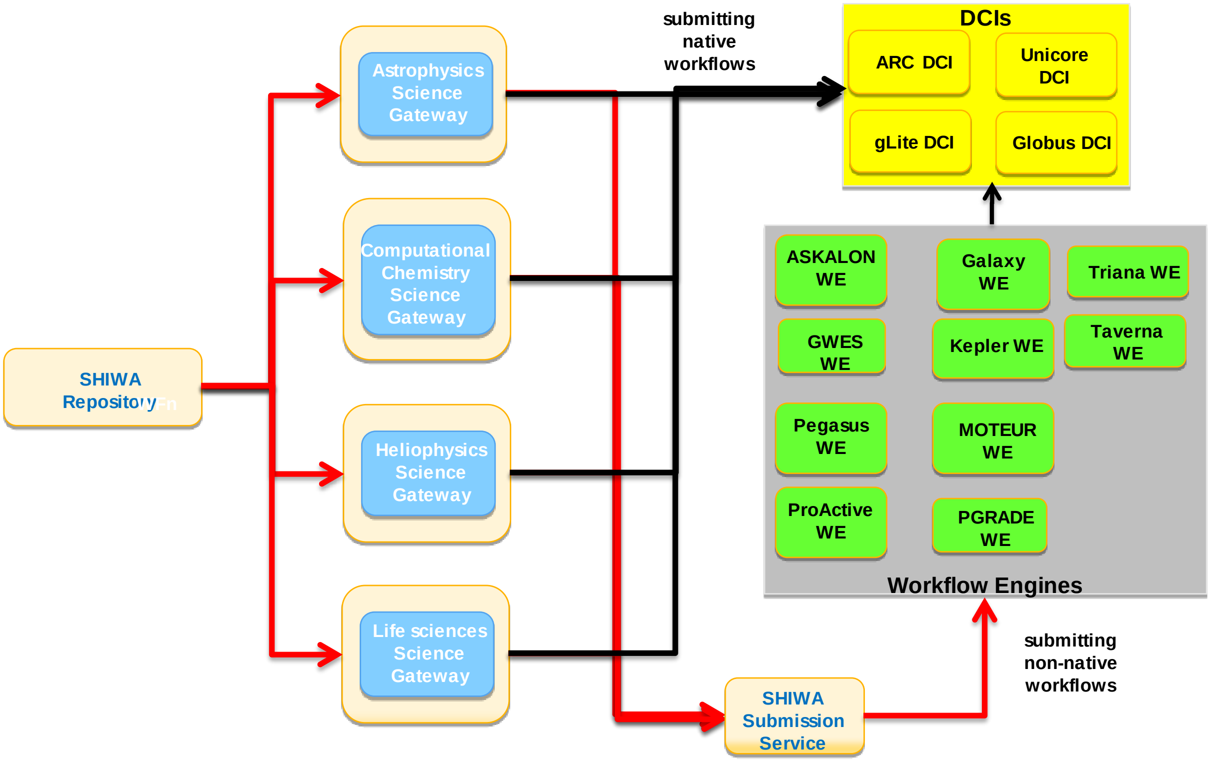


Figure 6.2: ER-flow execution environment

Remark: red arrow - remote SHIWA services
black arrow - community science gateway's services

The upgraded community gateways use two SHIWA services as remote services: the SHIWA Repository and the SHIWA Submission service. Researchers can download the selected workflows from the SHIWA Repository using the automatic workflow import function. The WS-PGRADE workflow engine submits the native (WS-PGRADE workflows) through the DCI Bridge to the selected Distributed Computing Infrastructure (It is depicted by black arrows on Fig. 6.2). To execute non-native workflows the workflow engine contacts the SHIWA Submission Service that manages the submission and execution of non-native workflows. (These operations are presented by red arrows on Fig. 6.2).

7 Community Requests towards the SHIWA Simulation Platform

ER-flow decided to have three phases of the requirements specification in the ER-flow project in Year 1:

- phase 1 October - November 2012
- phase 2 March - April 2013
- phase 3 May - July 2013

In **Phase 1** WP3 and WP5 analysed how and where Astrophysics, Computational Chemistry, Heliophysics and Life Science community previously run applications/workflows and compiled Table 7.1.

	workflow system	Portal	Repository	Infrastructure
Astrophysics	WS-PGRADE	WS-PGRADE	none	Italian NGI
Computational Chemistry	Unicore WS-PGRADE	WS-PGRADE	none	German NGI
Heliophysics	Taverna WS-PGRADE	WS-PGRADE	myExperiment	UK NGI
Life Science	MOTEUR WS-PGRADE	WS-PGRADE	SHIWA Repository	Dutch NGI

Table 7.1: Existing execution environments

Remark: This table is the updated version of the Table 7.1 in D3.1.

WP5 collected the workflows of the Astrophysics, Computational Chemistry, Heliophysics and Life Science community which they want to port to the SHIWA Simulation Platform. WP3 demonstrated the SHIWA Simulation Platform at the ER-flow kick-of meeting. The work package collected enhancement requests from researchers of these four communities in October – November 2012. WP3 defined two types of enhancement requests: major and minor and introduced three categories: short-, medium- and long-term requests. Table 7.2 and 7.3 presents the major and minor enhancement requests. The work package addressed some of these requests in the period of November 2012 - January 2013 and upgraded the simulation platform to SSP 4.1. See details in Section 8.

Phase 2 started at the ER-flow Application Porting Workshop (http://www.erflow.eu/19-22-march-2013-application-porting-workshop) held in London in 19 – 22 March 2013. WP3 upgraded the manuals of the SHIWA Portal and the SHIWA Repository and elaborated brand new portal and repository tutorials. The work package ran a portal and repository hands-on at the workshop for researchers of the four communities involved in the project plus for researchers of the hydrometeorology community (DRIHM project) and seismology community (VERCE project). The workshop participants started compiling the second set on enhancement requests (See Table 7.4 and Table 7.5) which was further extended in the period March – April 2013. WP3 addressed some of these requests in the period of April - June 2013 and upgraded the simulation platform to SSP 4.2. See details in Section 8.

Phase 3 covers May - July 2013. The key event of this period was the second ER-flow project meeting. WP3 demonstrated the latest version of the SHIWA Simulation Platform presenting the new or upgraded features and services. The four research communities started creating the third set of enhancement requests which will be included in D5.1 “User evaluation of the simulation platform” deliverable. Work package will address the requests raised in D5.1 in July - August 2013 and in the second project year.

7.1.1 Phase 1: enhancement requests

The research communities and technology providers identified 17 major enhancement requests in this phase. They classified four of them as short-term, seven of them as mid-term and six of them as long-term requests considering how they can improve user experience of the simulation platform.

No.	major enhancement requests	partners to deliver	schedule
A01	To upgrade the **SHIWA Portal** to offer better performance and services needed by the research communities.	UoW	ST
A02	To create an **export-import service** to improve workflow management between the SHIWA Portal and SHIWA Repository	SZTAKI + UoW	ST
A03	To implement **single-sign-on** for the SHIWA Portal and the SHIWA Repository and use the SHIWA Portal as a single point of entry to the simulation platform	SZTAKI + Uow	ST
A04	To support **robot certificate** to offer simple access for both workflow developers and e-scientists not using users' certificates.	SZTAKI	ST
A05	To provide either the **end-user interface** or **ASM based interface** for e-scientist considering their requirements	UoW	MT
A06	To re-engineer the **SHIWA Submission (or GEMLCA) Service** replacing GT4 with web services and upgrading its functions	UoW	MT
A07	To **monitor the components of the simulation platform** (portal + repository + submission service + workflow systems)	SZTAKI + UoW	MT
A08	To create **personalized access to the SHIWA Repository**, i.e. access to the public or the private view of the repository based on the user profile	UoW	MT
A09	To collect **information about workflow execution** (domain, user, etc.) including both meta and sub-workflows	SZTAKI	MT
A10	To create a **personalised view** of the SHIWA Repository (my own workflows, my favourite workflows, etc.)	UoW	MT
A11	To support **monitoring of non-native sub-workflow execution** and retrieve proper fault information	SZTAKI	MT
A12	To define and publish the **SHIWA Repository API** to support upload, listing, searching and downloading workflows	UoW	LT
A13	To define and publish the **SHIWA API** to expose the simulation platform services which manage workflow execution	SZTAKI + UoW	LT
A14	To **integrate** login with **EGI SSO, facebook, gmail**, etc.	SZTAKI + UoW	LT
A15	To enable **automatic workflow execution environment deployment** on cloud and workflow execution on cloud	SZTAKI + UoW	LT
A16	To support **workflow downloading** in CGI bundle format without the SHIWA desktop concept	SZTAKI + UoW	LT
A17	To extend **ASM with SHIWA API** to allow users to manage workflow data and execution	SZTAKI	LT

Table 7.2: Major enhancement requests in phase 1

Legends: ST – short-term MT – medium-term LT – long-term
green colour – completed upgrade blue colour – upgrade under development

Remark: This table is the updated version of the Table 7.2 in D3.1.

The research communities and technology providers identified 14 minor enhancement requests in phase 1. They specified five of them as short-term and nine of them as mid-term requests. WP3 implemented all short-term enhancement requests and three of the mid-term requests.

No.	minor enhancement requests	partners to deliver	schedule
I01	To introduce **pre-defined domains** and **sub-domains** and re-allocate workflows to relevant domains	UoW	ST
I02	To support proper **workflow versioning** in the SHIWA Repository	UoW	ST
I03	To modify the **welcome page of the SHIWA Repository** and provide direct access to the browse page of abstract workflows	UoW	ST
I04	**Controlling characters** entered into the SHIWA Repository (lowercase, uppercase, white spaces should not matter in the repository)	UoW	ST
I05	To enable **automatic re-submission of failed non-native sub-workflows**	SZTAKI	ST
I06	To assign **URL to each workflow implementation** uploaded to the repository (the URL should guide to workflow implementation stored in the repository)	UoW	MT
I07	To create and **display lists of** the latest 5-10 workflows and the most frequently used 5-10 workflows	UoW	MT
I08	To enable **access to the simulation platform via Facebook**	SZTAKI	MT
I09	To enable **downloading Taverna 2 workflows** from the myExperiment repository	UoW	MT
I10	To enable specifying input file type (for example GIF file for ImageMerger wf)	UoW	MT
I11	To replace the existing validation strategy with **community based validation** and implement it	UoW	MT
I12	To **transfer workflows** uploaded to the SHIWA Repository to the EGI App database	UoW	MT
I13	To allow users to move from the **easy user view** to the **power user view** by a single click assuming that user will access the end user view after login	SZTAKI	MT
I14	To **test workflows** uploaded to the repository on cloud or on local resources by a single click		MT

Table 7.3: Minor enhancement requests in phase 1

Remark: This table is the updated version of the Table 7.3 in D3.1.

7.1.2 *Phase 2: enhancement requests*

The research communities and technology providers defined two more mid-term major enhancement requests in phase 2.

No.	major enhancement requests	partners to deliver	schedule
A18	To enable collection and processing workflow execution statistics	UoW	MT
A19	To support **interactive workflow nodes**	UoW	MT

Table 7.4: Major enhancement requests in phase 2

The research communities and technology providers identified further five short-term and two mid-term enhancement requests in this phase. WP3 implemented three short-term requests.

No.	minor enhancement requests	partners to deliver	schedule
I15	To provide access to **gLite resources**	UoW	ST
I16	To upgrade **export-import service** for MOTEUR workflow system	UoW	ST
I17	To offer **remote access** to the SHIWA Repository and the SHIWA Submission Service	SZTAKI UOW	ST
I18	To provide access to **UNICORE resources**	UoW	ST
I19	To implement **CGI support** for the **UNICORE workflow system**	UoW	ST
I20	To restrict **number of users** who can user the portal at the same time	UoW	MT
I21	To extend **CGI support** for the **DISPEL workflow system**	UoW	MT

Table 7.5: Minor enhancement requests in phase 2

8 Upgrading the ER-flow Development Environment

8.1 Major Upgrades

8.1.1 Workflow export-import service enhancement requests: A02

Problem: The SHIWA Repository provides the Developer (or Table) View as GUI to specify abstract and concrete workflows. Workflow developers can enter workflow data (metadata, configuration and dependencies) through this GUI. Uploading workflow data requires knowledge of the abstract and concrete workflow data structure plus workflow developers should be familiar with the Developer View.

Implementation: Workflow upload and download is implemented by two operations: export and import operation. Both the SHIWA Portal and the SHIWA Repository support the CGI (or WS-PGRADE) bundle. The bundle is an archive file that stores data of abstract and concrete workflows. The export operation (Fig. 8.1) first retrieves the workflow from the gateway's local storage and creates the workflow bundle. Next, it uploads the bundle into the SHIWA Repository, which converts the workflow data into the repository data format. The import operation (Fig. 8.2) first gets workflow data from the SHIWA Repository, generates workflow bundle and forwards it to the SHIWA Portal. The portal unzips the bundle and stores it in the local storage of the gateway. The new export/import services are integrated into the Concrete/Export and Workflow/Import menus of WS-PGRADE portal.

Usage: The workflow developers can export workflows after creating them in the SHIWA Portal. First, they have to select the "Export" tab and the "Remote SHIWA Repository" option. Next, they have to login to the SHIWA Repository and select a group to which they want to assign the workflow. Finally, they have to click on the "Export in WS-PGRADE format" button.

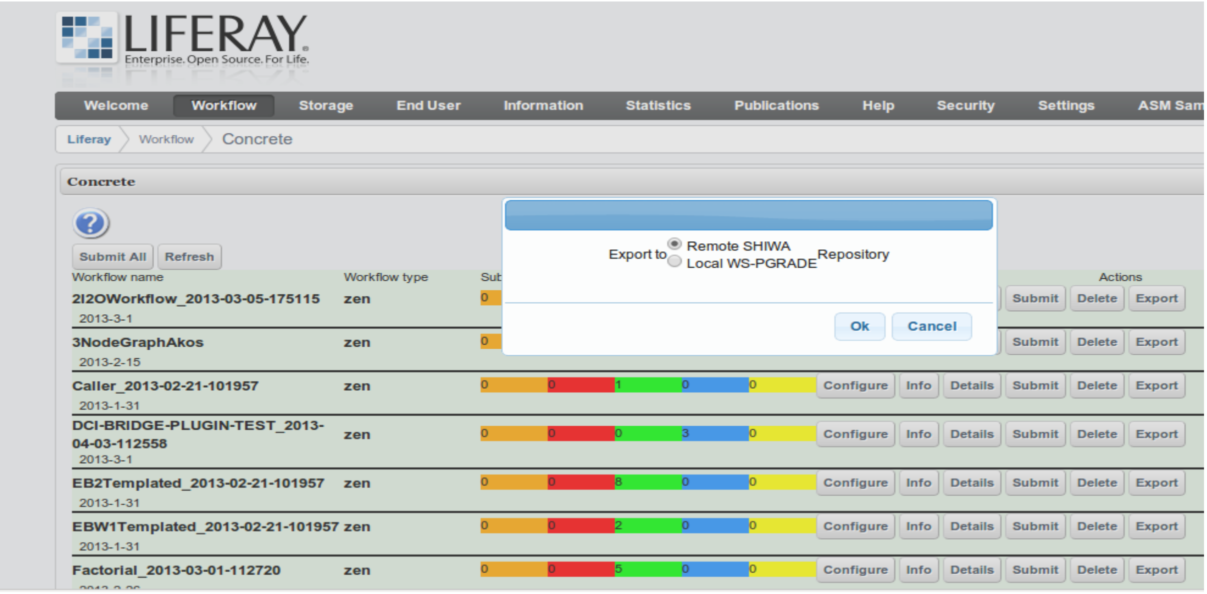


Figure 8.1: Exporting workflows to the SHIWA Repository

The import operation is even simpler than the export operation. Workflow developers have to click on the "Import" tab in the "Workflow" tab and select the SHIWA Repository as source of the import operation. Next, the portal will display the list of the workflows available in the repository. The developers have to select one of them and repository forwards it workflow bundle to the portal.

8.1.2 Robot certificate management enhancement requests: A04

Problem: Many e-scientists have no user certificate and are reluctant to acquire it. They are discouraged from using workflows if they cannot execute them without user certificates. Robot certificates make user certificates unnecessary for this type of users. They provide a convenient authentication solution with the resources that does not require any involvement from users. This authentication method identifies the person who runs a particular workflow from the gateway on a particular resource through the workflow, but the jobs are executed using a community certificate that is pre-installed in the gateway.

Implementation: The portal administrator allocates robot certificates to workflows or workflow jobs. Having access to portal accounts they can identify who runs the workflows. The robot certificate management is implemented according to the EGI VO Portal Policy and EGI mode classification. Other necessary requirements were: the portal must be capable of limiting the job submission rate and must keep audit logs for all interactions with infrastructures. The Figure 8.2 shows the gUSE services and their roles in the job submission using robot certificate. The main characteristics of robot job submission are on the one hand the user level creation of a robot permission association (RPA that binds the executable to a robot certification) in the job configuration page in WS-PGRADE. On the other hand the DCI Bridge stores the executable instead of the Portal Storage.

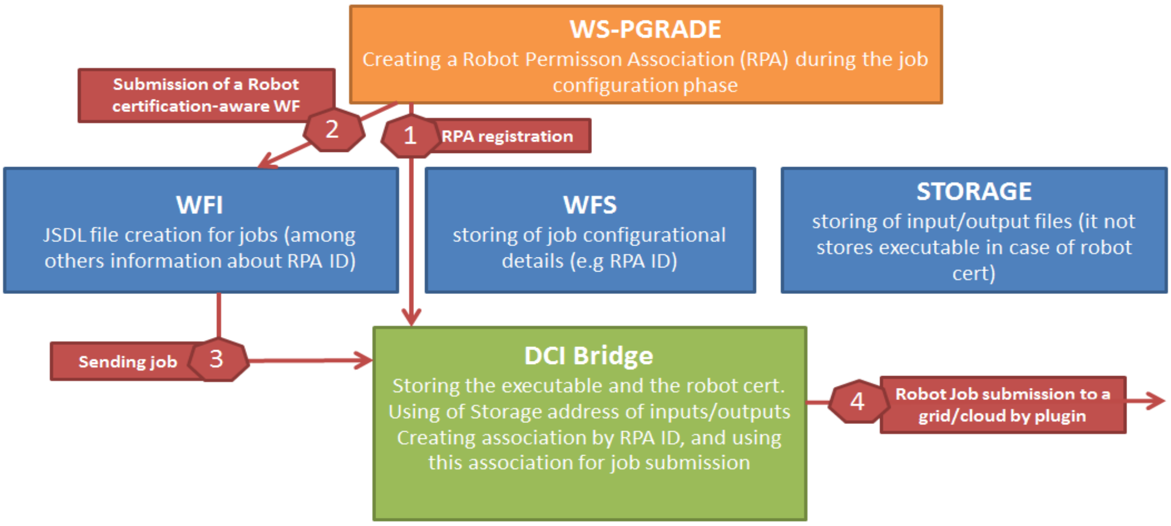


Figure 8.2: Job submission with robot certificate

The most important implementation considerations are:

1. **The WS-GRADE extension to support robot certificate**: The robot certificates should be hidden from end users, but they should be available for the portal administrator. The workflow with robot certificate is stored in the local gUSE repository.
2. **The process to assign robot certificates to workflow nodes:** To execute a workflow robot certificates should be assigned to workflow nodes. Some workflow nodes can be run with robot certificates, while other nodes might require user certificates.
3. **Storing the robot certificate for a given node**: When the portal developer saves the workflow that contains a job with robot certificate the following happens:
   - The portal stores the job with the required robot certificate within the DCI Bridge. This job bundle (RPA) gets a job bundle identifier (RPA ID).
   - In the job configuration this identifier is added to the original configuration information
   - When the workflow is saved, this new extended configuration field (containing the RPA ID) is be saved as well.
4. **Workflow execution with robot certificate:** When the workflow engine interprets a job with a robot certificate, the RPA ID is placed into the JSDL too. After getting a job with a

RPA ID, the DCI Bridge submits the job together with the executable and the robot certificate according to the given RPA ID.

Portal GUI level: The WS-PGRADE GUI was extended by adding of robot permission association (RPA) in the job configuration phase in the *Configuration/Job Executable* window of the portal. The RPA makes relation between the previously configured infrastructures and he workflow that user wants to run.

Usage: With robot certificates, users can run workflows without any individual authentication data for the resource. Thus, they need to import/upload the previously created robot certificate-aware workflows for submissions. The certificate is registered in a VO and the workflow has access to resources that are available registered members of the specified VO. WS-PGRADE supports robot certificate for every resource type that is accessible from the portal and requires authentication including clusters, service grids (ARC, gLite, Globus, UNICORE) and clouds (via CloudBroker Platform and Google App Engine).

8.1.3 Workflow Execution Statistics service enhancement requests: A18

Problem: The gUSE/WS-PGRADE statistics service is responsible for collecting and storing usage metrics in a database and for displaying these metrics. The previous statistics service introduced a significant overload on the gateway.

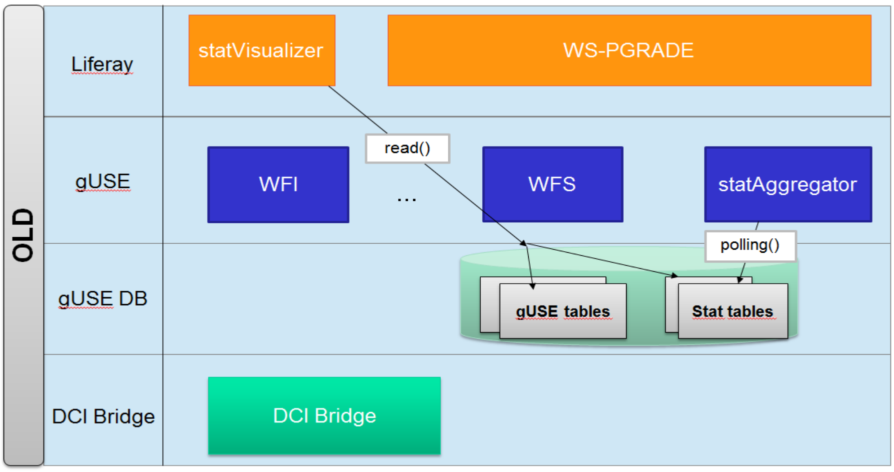


Figure 8.3:Previous statistics service in gUSE

The main bootleneck was the Stataggregator and the Statvisualizer services. The Stataggregator service, which is responsible for collection, organizing and aggregating data sent by the Workflow Storage (WFS), polls the stat tables in the database in an infinite loop. This operation leads significant overheads. This service also reads gUSE data tables (it can lead to transaction errors) and most of the statistical data processing was written as triggers (it resulted bad efficiency). The Statvisualizer service reads directly from both gUSE and statistics tables, which even further increases the overload. To improve the performance a faster statistics service is needed that manages the gUSE database at a lower level and exploits the portal resources more effectively than in the previous gUSE versions.

Implementation: MTA-SZTAKI developed a new event-driven data managing and processing solution to handle statistics data. They implemented the statistics service as background standalone service with its own database to improve the performance. As it is seen on Fig. 8.4, the gUSE and statistics data was separated and put in the gUSE and statistics database. The new implementation modified both the Stataggregator and the Statvisualizer services. The Statvisualizer does not read data directly from the statistics database. It gets data through WFS web service. WFS collects every status changes of every job and sends them to the Stataggregator through the web service instead of inserting them into the database one by one. As a further improvement, the data aggregation runs in Java instead of using triggers to decrease the number of I/O operations. The performance can be further boosted if the statistics service is deployed on a separate server.

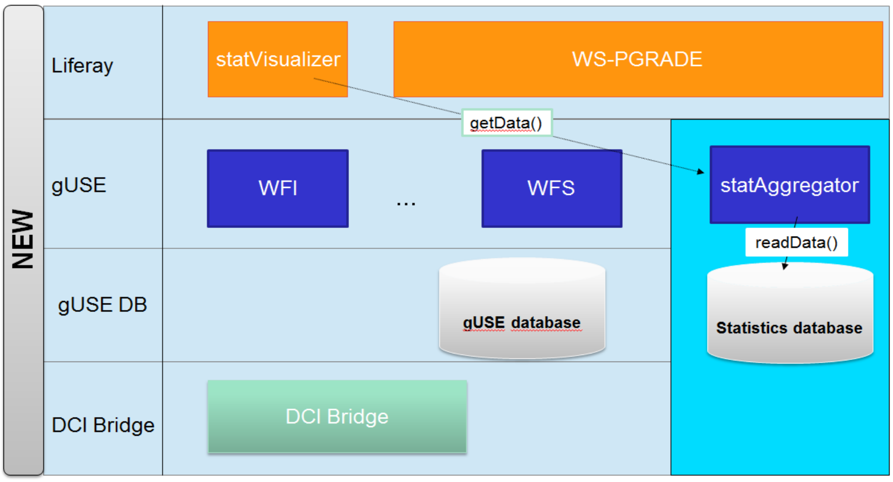


Figure 8.4: New statistics service in gUSE

Usage: The statistics function allows viewing metrics in four tabs – portal, DCI, user, and workflow –, and on seven levels – portal, user, DCI, resource, concrete workflow, workflow instance and abstract job. This is accomplished by allowing users to navigate to different pages to see the level of statistics they want. This solution (via less system loading than earlier) is helpful for administrator in resource settings (mainly from portal and DCI statistics) and for users interested in middleware, resource selecting and in job execution time estimation (mainly from user and workflow statistics).

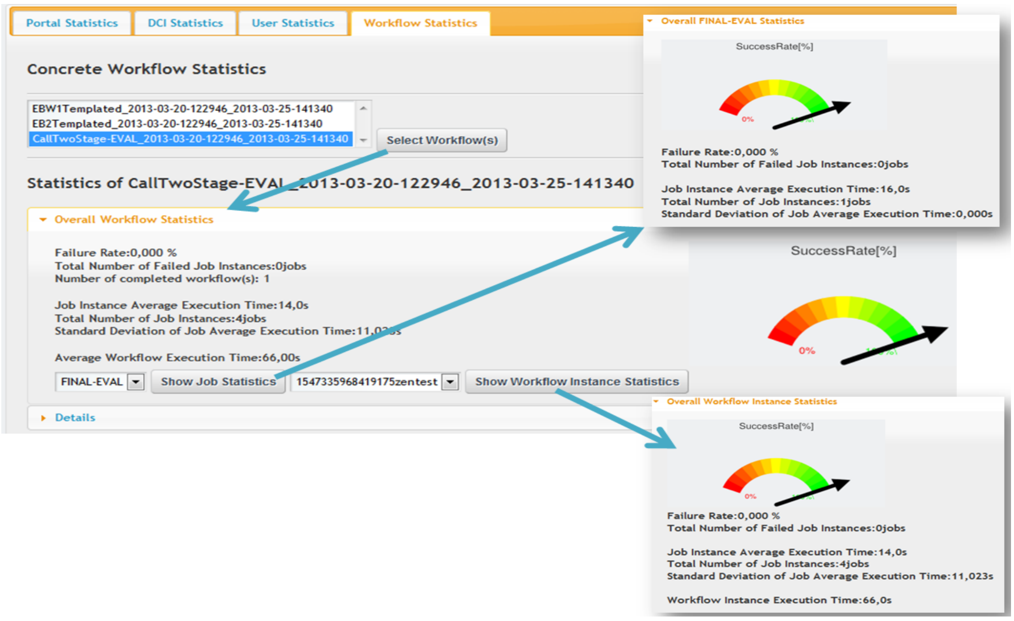


Figure 8.5: Statistics service - Workflow Statistics tab

Based on the Memorandum of Understanding between the ER-flow and SCI-BUS project, SCI-BUS developed the export-import service and the robot certificate management, and upgraded the statistics service. The ER-flow project deployed these services on both the ER-flow development and execution environment.

8.1.4 SHIWA Portal upgrade enhancement requests: A01

Problem: The SHIWA project re-deployed the SHIWA Portal four times to offer more advanced and more efficient services for research communities. The portal was last time upgraded in May 2012 using WS-PGRADE 3.4.5. Analysing requirements of the ER-flow research communities WP3 identified that this portal either did not offer some of the services needed and requested by the research communities:

- automatic export-import workflow service between the SHIWA Portal and SHIWA Repository
- robot certificate management service

or provided services which were not good enough for the ER-flow project:

- execution statistics service

Implementation: WP3 re-deployed the SHIWA Portal three times in Year 1 to add or improve available services and provide better performance. We outline the new or upgraded features and services which the SHIWA Portal provides after each re-deployment:

1. re-deployment No. 1 v3.5.2 November 2012

- The work package deployed this sub-version skipping a few subversions. This sub-version offers the following improvements and upgrades:
- **visual feedback service** – It provides visual feedback about processes with long execution times.
- **improved DCI Bridge functionality** - v3.5.0 fixed the GEMLCA GT4 job submission issue.

2. re-deployment No. 2 v3.5.5 May 2013

- **robot certificate service** - Considering that ER-flow targets both workflow developers and e-scientists it is essential to provide seamless access DCIs through the SHIWA Simulation Platform. Workflow developers use these infrastructures on a regular basis and they have user certificates to access them. In contrast getting and managing certificates could be a blocking issue for e-scientists. Supporting robot certificates the SHIWA Portal is able to provide access to Dutch and Italian gLite resources. In Year 2 the robot certificate based access will be extended to UNICORE resources.
- **improved DCI Bride performance** - The bridge performance was optimised to offer significantly faster job submission.

3. re-deployment No. 3 v3.5.7 June 2013

- **workflow export-import** service - This service (available from v3.5.6) adds to the previous manual workflow upload-download operation the automatic workflow export into the SHIWA Repository and workflow import to the SHIWA Portal. The export-import service is based on the CGI (or WS-PGRADE) bundles. The uploaded and downloaded workflows can be native WS-PGRADE workflows, non-native and meta-workflows embedded in WS-PGRADE workflows.
- **statistics service** - The previous statistics service significantly decreased the portal performance when it was switched on. As a result, we were not able to use this service 24/7. The new service enables fast statistical data collection and processing without significantly affecting the portal performance.

8.1.5 SHIWA Submission Service enhancement requests: A06

Problem: There are four major limitations of the SHIWA Submission Service (or GEMLCA Service). First, it is implemented as a GT4 based service, which requires GT4 deployment to run it. According to the ER-flow strategy, science gateways of research communities are execution environments that run workflows. These gateways need the submission service to execute non-native workflows which requires GT4 deployment. Second, the GEMLCA Service supports submission only to GT2 and GT4 based resources, while communities also want to submit to gLite and UNICORE resources. As a temporary solution the GEMLCA Service uses the DCI Bridge to submit workflow to gLITE and UNICORE resources. Third, currently there are two repositories in the simulation platform: GEMLCA Repository and SHIWA Repository. The first one stores execution-enabled workflows plus workflow engine data and executable. As a result, there are two data formats to describe workflow and workflow engine data, plus two GUIs to manage this data. Fourth, the GEMLCA Service has some performance issues during workflow execution because of the implemented caching solution.

Implementation: The Figure 8.6 gives an overview of the new submission service. There are two major development tasks. First, we have to extend the SHIWA Repository to allow management of workflow engine data and enable workflow engine execution. This development incorporates the integration of workflow engines management and the communication between the repository and the submission service. Second, the existing submission service has to be re-implemented. The submission service will enable communication among the DCI Bridge, the SHIWA Portal and the SHIWA Repository. We started the development of the **Search** and **Process** API. The **Search** API provides search operations for the SHIWA Portal and the SHIWA Repository to find workflows and workflow engines. The **Process** API allows workflow submission through the DCI Bridge.

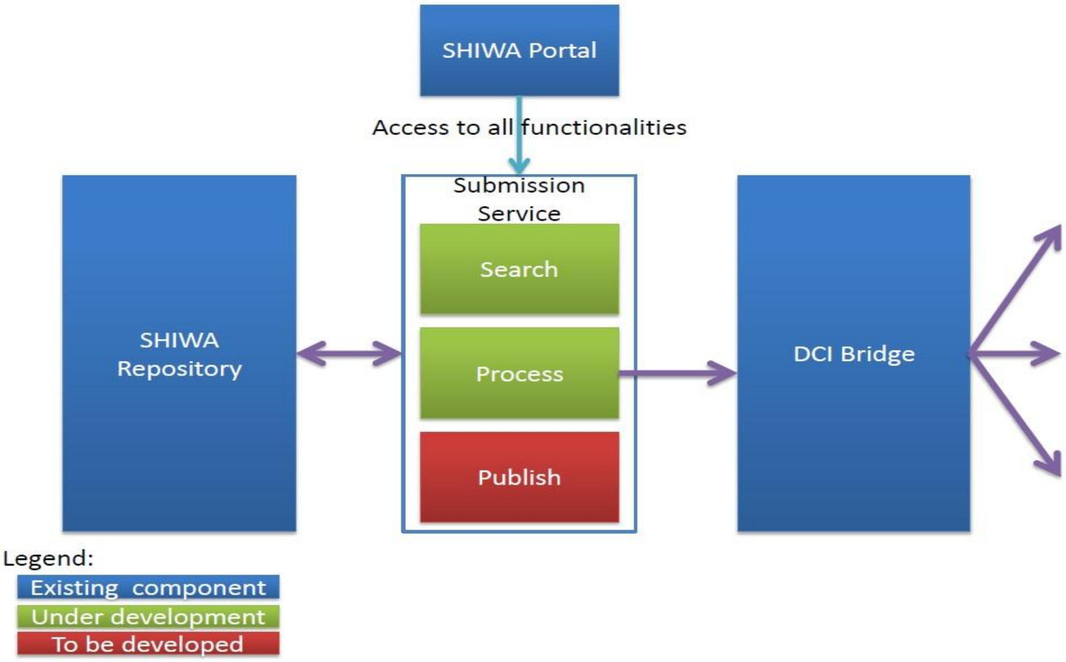


Figure 8.6: Current development state of the new submission service

Usage: The new submission service should be able to provide a simplified access to the SHIWA Repository with different APIs as shown on Figure 8.6. It will allow to:

- Request data stored on the SHIWA Repository (workflows, workflow engines, etc.);
- Manage workflows with basic set of operations (i.e. add, delete, modify);
- Execute workflows on the middleware needed using the DCI Bridge.

8.1.6 *Workflow engine data management enhancement requests: A06*

Problem: In the Coarse-Grained Interoperability approach each workflow is assigned to a workflow engine in the SHIWA Repository. Workflow developers have to specify the workflow engine and its version as a parameter of the concrete workflow (or workflow implementation). In the current SHIWA Simulation Platform the SHIWA Submission Service (or GEMLCA Service) manages non-native workflows as legacy code applications. This service combines non-native workflows with non-native workflow engines which execute them. The GEMLCA Service has the built-in GEMLCA Repository which manages non-native workflow engines data (data, metadata, binaries, configuration, dependencies, etc.) Workflow engine developers use the GEMLCA Repository Portlet (or GEMLCA Admin Portlet) to specify, upload, edit and delete workflow engine data. As a result, they have to use two GUIs in the CGI approach: GEMLCA Repository to manage workflow engine data and SHIWA Repository to handle workflow data.

Implementation: The aim is to extend the SHIWA Repository to take over the workflow engine data management from the GEMLCA Repository. This upgrade will make the GEMLCA Repository obsolete. University of Westminster extended the data structure of the SHIWA Repository to manage data, meta-data, binaries, configuration and dependencies of workflow engines. Currently, WP3 is extending the repository code to enable workflow engine data management in the SHIWA Repository.

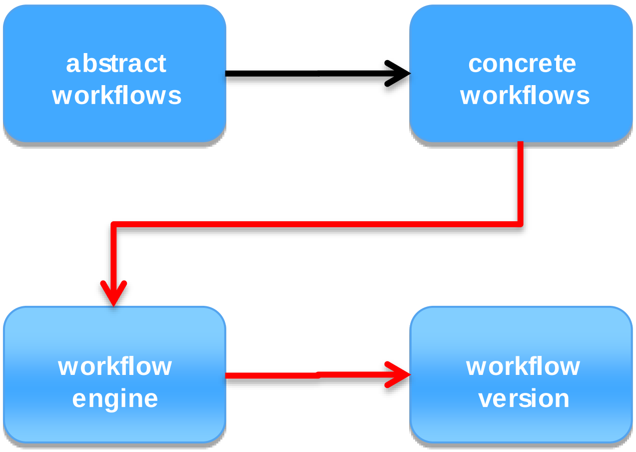


Figure 8.7: Managing workflows and workflow engines

To execute non-native workflows the new submission service (see section 8.1.5) will combine the workflow and the workflow engine after retrieving the data of the workflow and the workflow engine from the SHIWA Repository.

Usage: Workflow engine developers have to define data of workflow engines using the SHIWA Repository GUI. They should also upload binaries, configuration and dependency data of workflow engines into the repository through the same GUI. Workflow developers have to select first the workflow engine, and then its version from a drop-down list of workflow engines and to associate a workflow engine with a concrete workflow.

8.2 Minor Upgrades

8.2.1 Workflow domain and sub-domain management

enhancement requests: I01

Problem: Currently, workflow developers can assign any kind of domain to workflows without any restriction in the SHIWA Repository. As a result, there are many different workflow domains and some domains appear multiple times with different names. When users try to browse workflows and open the drop-down list, they have to select from more than 30 domains.

Implementation: WP3 created a predefined list of domains and sub-domains which enables users to select domain and sub-domain names from a drop-down list when they add new workflows to the repository. The ER-flow research communities defined the following domains and sub-domains:

domain	sub-domain
Astrophysics	Cosmology, Computational cosmology, Meteors dynamics, Comet dynamics, Stellar evolution, Stellar Astrophysics, Astrophysical visualizatio
Computational Chemistry	Docking, Molecular Dynamic, Quantum Chemistry
Heliophysics	Solar Physics, Propagation Models, Solar Cycle
Life Sciences	Neuroimaging, Next Generation Sequencing, Mass Spectrometry

WP3 also introduced two more domains: demonstration and test domain. Having these domains and sub-domains the work package cleaned up the SHIWA repository. WP3 re-assigned workflows to the new domain and sub-domains. See details in 8.2.2

Usage: When a user creates a new workflow, he/she can use a drop-down list containing all the predefined domain names. See below:

Figure 8.8: Domain drop-down list

To manage sub-domains users can use a second drop-down list whose content depends on the domain drop-down list. Either a new attribute has to be introduced for the workflow or they can be considered as keywords.

8.2.2 SHIWA Repository clean-up ***enhancement requests: I01***

Problem: In the SHIWA project workflow developers uploaded abstract and concrete workflows into the SHIWA Repository. The repository enabled developers to specify domain and sub-domain names and assign workflows to them. As a result, there are multiple domain names which cover the same science area. For example workflow developers introduced the bio-science, life science and neuro-science domain names for Life Sciences. The situation with the sub-domain names is even more complicated several sub-domain names identify he same area.

Implementation: WP3 introduced a pre-defined list of domain and sub-domain names. The abstract and concrete workflows uploaded in the SHIWA project belong to four main domains: to Computational Chemistry and Life Sciences at one side and to demonstration and test at the other side. The researchers of the Computational Chemistry and Life Sciences community analysed the workflows in the repository. The work package in cooperation with these communities re-allocated the workflows to the new domains and sub-domains. We categorised all the other workflows available in the SHIWA Repository. The work package identified further three domains: Data Mining, Meteorology and Multimedia. WP3 introduced the relevant domain names and re-allocated workflows belonging to these domains plus the demonstration and test workflows to the relevant domains.

Usage: Currently, there are the following domains in the SHIWA Repository:

- Astrophysics, Computational Chemistry, Heliophysics, Life Sciences
- Data Mining, Meteorology, Multimedia and
- Demonstration, Test.

Workflow developers are expected to use these domain names when they upload new workflows. When new research communities want to use the repository, WP3 will extend the list of domain and sub-domain names based on their requests.

8.2.3 Taverna CGI support upgrade enhancement requests: I09

Problem: New version of the Taverna workflow engine was released (Taverna 2) which is not fully compatible with Taverna 1 from the CGI approach's point of view. This version is used by one of the ER-flow communities (Heliophysics) and needed to be supported by the Coarse-Grained Interoperability approach in the same way as Taverna version 1 is supported.

Implementation: The Taverna community provides a command line version of the Taverna 2 workflow engine. This version was deployed on the Westminster cluster and linked with the SHIWA Submission Service in order to create and submit workflows using the SHIWA Portal. The Fig. 8.9 presents the modifications implemented for this integration. WP3 has created a shell script to invoke Taverna 2 workflow engine through a command line client. This shell script submits a Taverna 2 workflow to the Taverna 2 workflow engine using the command line tool. To support execution of Taverna 2 workflows we created a Taverna 2 GIB (*Generic Backend Interpreter*). This GIB combines the Taverna 2 workflows with the Taverna 2 workflow engine and manages its submission.

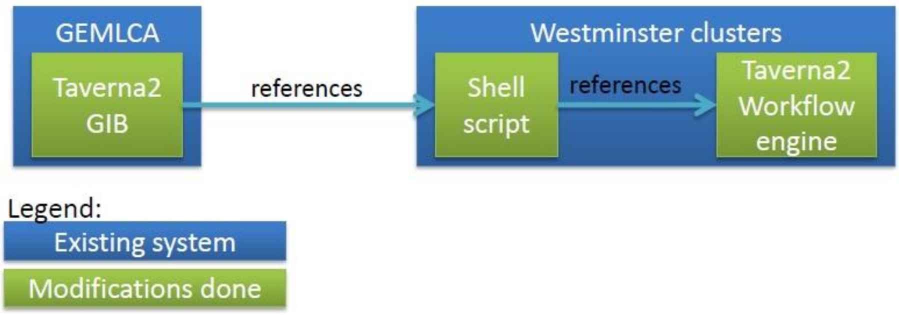


Figure 8.9: CGI support for the Taverna 2 workflow engine

Usage: The following steps have to be done to submit a workflow:

- Create the workflow and its implementation in the SHIWA Repository;
- Enable the execution of this the implementation using the new Taverna 2 GIB;
- Deploy the implementation on GEMLCA;
- Create and submit a workflow via the SHIWA Portal using the deployed implementation.

8.2.4 Exporting Taverna2 workflows from myExperiment enhancement requests: I09

Problem: The Taverna community uses the myExperiment repository to store Taverna workflows and to share them. The SHIWA Repository was able to automatically import Taverna 1 workflows from the myExperiment repository but not Taverna 2 workflows because the syntax between a Taverna 1 and Taverna 2 workflows is slightly different.

Implementation: The SHIWA Repository supported automatic import of Taverna 1 workflows from the myExperiment via the myExperiment module. We have extended this module to support automatic import of Taverna 2 workflows. (See Figure 8.10) The extended module imports Taverna 2 workflows from the myExperiment repository and adds them to the SHIWA Repository.

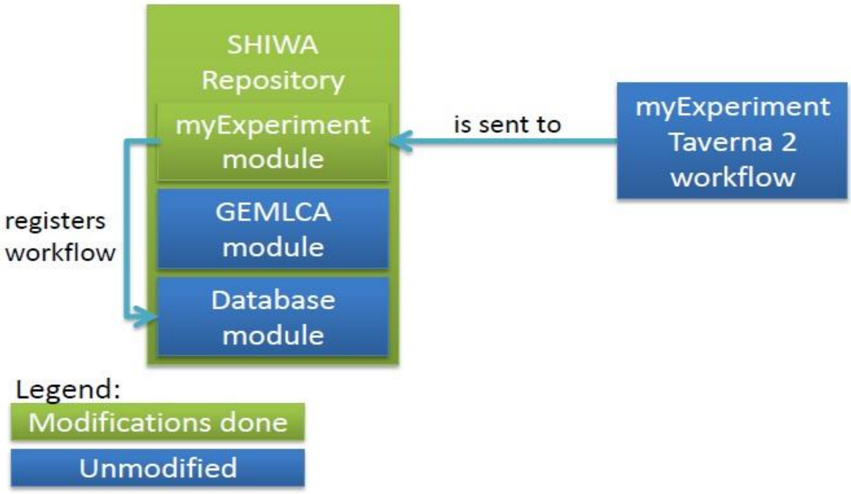


Figure 8.10: Exporting Taverna 2 workflows in the SHIWA Repository

Usage: User communities are able to export Taverna 2 workflows from the myExperiment repository into the SHIWA Repository and execute them on the SHIWA Simulation Platform. First, users have to browse the myExperiment repository to find the workflow they need. Next, they have to obtain the ID of the selected workflow, which is the number of the workflow in the browsing bar. Further, having this ID they should enter it in the myExperiment tab in the SHIWA Repository. Finally, they have to assign name for the workflow and the abstract workflow and its implementation are created automatically.

8.2.5 Access to gLite and UNICORE resources enhancement requests: I15

Problem: In the SHIWA project the SHIWA Portal provided access to the SHIWA Virtual Organisation. The technology providers offered access to gLite, GT2 and GT4 based Virtual Organisations. The ER-flow research communities want to access further gLite based resources and UNICORE based resources. The Astrophysics community wants to use the Astro, INAF, Planck and VOCE Virtual Organisations, while the Life Sciences uses the VLEMED Virtual Organisation to run workflows. The Computational Chemistry community needs access to the UNICORE based German NGI resources.

Implementation: WP3 configured the DCI Bridge of the SHIWA Portal to access gLite based resources. Currently, Planck and VOCE resources are available for the Astrophysics community, and the Life Sciences community can use VLMED resources. The work package is configuring the DCI Bridge to access the Astro and INAF Virtual Organisations. Providing access to UNICORE resources is a more demanding task than for gLite resources because both the DCI Bridge and the WS-PGRADE portal have been modified in the MoSGrid portal to deliver some UNICORE specific services. WP3 and the Computational Chemistry community is investigating how to deliver the UNICORE specific services through the SHIWA Portal.

Usage: Workflow developers identify the resource where the workflow can be executed. The concrete workflow (or workflow implementation) stores this information in the SHIWA Repository. The SHIWA Portal retrieves this information from the repository. Next, either the SHIWA Submission Service (if the workflow is non-native one) or the WS-PGRADE workflow engine (if he workflow is native one) forwards this information towards the DCI Bridge which submits the workflow to the specified resource.

8.2.6 Workflow access through URL enhancement requests: I05

Problem: Currently, it is a quite complicated to In order to refer workflows and/or their implementations from external pages (such as the EGI Application Database and SCI-BUS Portlet Repository). An URL to each workflow and each implementation uploaded to the repository has to be assigned.

Implementation: Each workflow has an URL as parameter which contains the workflow ID. Workflows can be referenced through these URLs in papers, publications, etc. When a workflow with this URL is selected in a paper the URL navigates to the workflow stored in the SHIWA Repository. See on Fig. 8.11.

Figure 8.11: Accessing implementation via URL

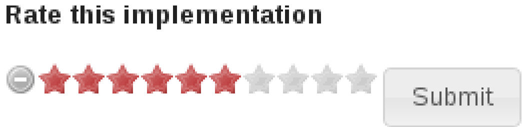
When users click on the workflow link, the repository gets the workflow (abstract or concrete) ID as a GET parameter in the request. The repository code handles this parameter as a "managed property" and redirects user to the relevant repository page.
Remark: A prototype has been implemented but it has not been deployed on the SHIWA Repository.

Usage: The URL guides users to the abstract or concrete workflow stored in the SHIWA Repository. For example: to access a workflow in the browse view users should enter the following URL to the browser:
http://shiwa-repo.cpc.wmin.ac.uk/shiwa-repo/public/details-view.xhtml?appid
where appiad = [ID OF THE ABSTRACT WORKFLOW]
To access an implementation in the table view:
http://shiwa-repo.cpc.wmin.ac.uk/shiwa-repo/public/edit-implementation.xhtml?impid
where impid = [ID OF THE IMPLEMENTATION]

8.2.7 Workflow validation enhancement requests: I11

Problem: The SHIWA Repository has the Validator actor who is responsible for testing both the abstract and concrete workflows uploaded to the repository. In SHIWA the project partners validated all the abstract workflows but there was not enough manpower to test all concrete workflows. According to the repository usage policy only validated workflows can be published. As a result of the manpower shortage many concrete workflows are available only in the private domain and hidden from the research communities. This type of "centralized" validation has not worked in the SHIWA project. The ER-flow project has even less resources than the SHIWA project. Thus, the “centralised” validation will not work either.

Implementation: Workflow developers, who crate the abstract and concrete workflows, can decide which workflows to make publicly available (public domain) and which ones to hide from non-registered users (private domain). WP3 proposed to replace the current validation approach with community (or social) validation**.** This type of validation enables users to execute and verify uploaded implementations. Registered users can rate public workflows using a simple, star-based rating component (ten scale system). Users can review and change the score number they gave.



An average value is calculated from all reviews and shown when displaying the particular concrete workflow.

Usage: Both e-scientists and workflow developers can evaluate workflows and give feedback using the community validation service.

8.2.8 Export-import of MOTEUR workflows enhancement requests: I16

Problem: During the SHIWA project, the MOTEUR workflow system was interfaced with the SHIWA Repository through the SHIWA Desktop version 1.5.1. This integration made it possible to:

- Upload workflows created with MOTEUR workflow designer directly into the repository (after filling in a form to describe workflows metadata), See Figure 8.12.
- Browse repository workflows through the MOTEUR interface and download selected workflows for display in the MOTEUR workflow designer. See Figure 8.13 below.

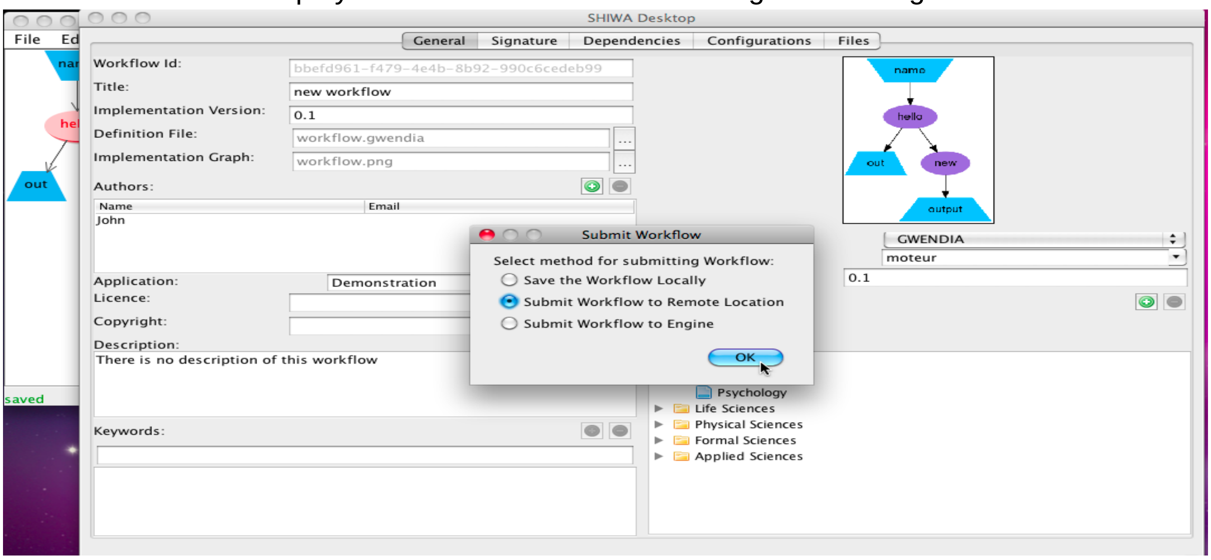


Figure 8.12: MOTEUR workflow upload interface to the SHIWA Repository. Metadata concerning the workflow is edited before sending the workflow bundle to the remote repository.

A new release (1.6) of the API has been made available since, and the workflow repository was migrated to it. Recent tests show that the former API version is not backward compatible, causing MOTEUR to fail when interfacing with the repository. MOTEUR code has to be updated to the latest API version.

Implementation: This update required a limited work on MOTEUR code which restored the complete export-import functionality previously available.

Usage:

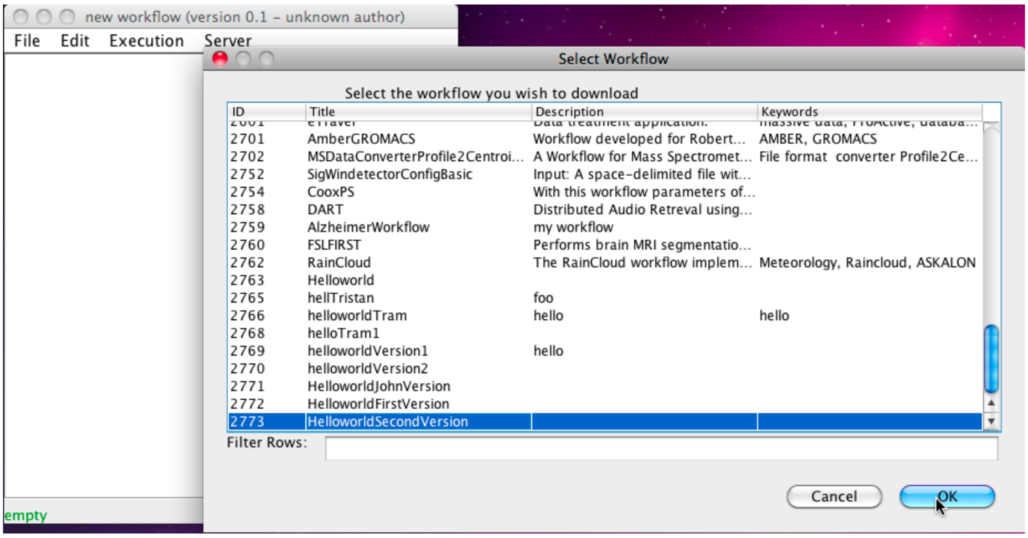


Figure 8.13: MOTEUR workflow download interface from the SHIWA Repository. The repository content is listed and workflows listed for the end user to select a workflow to download.

9 Upgrading the Community Science Gateways

MTA-SZTAKI and UoW as technology providers are organising a two day-long science gateway upgrade workshop for the ER-flow research communities in parallel with the 'Summer School on Grid and Cloud Workflows and Gateways' event[1].

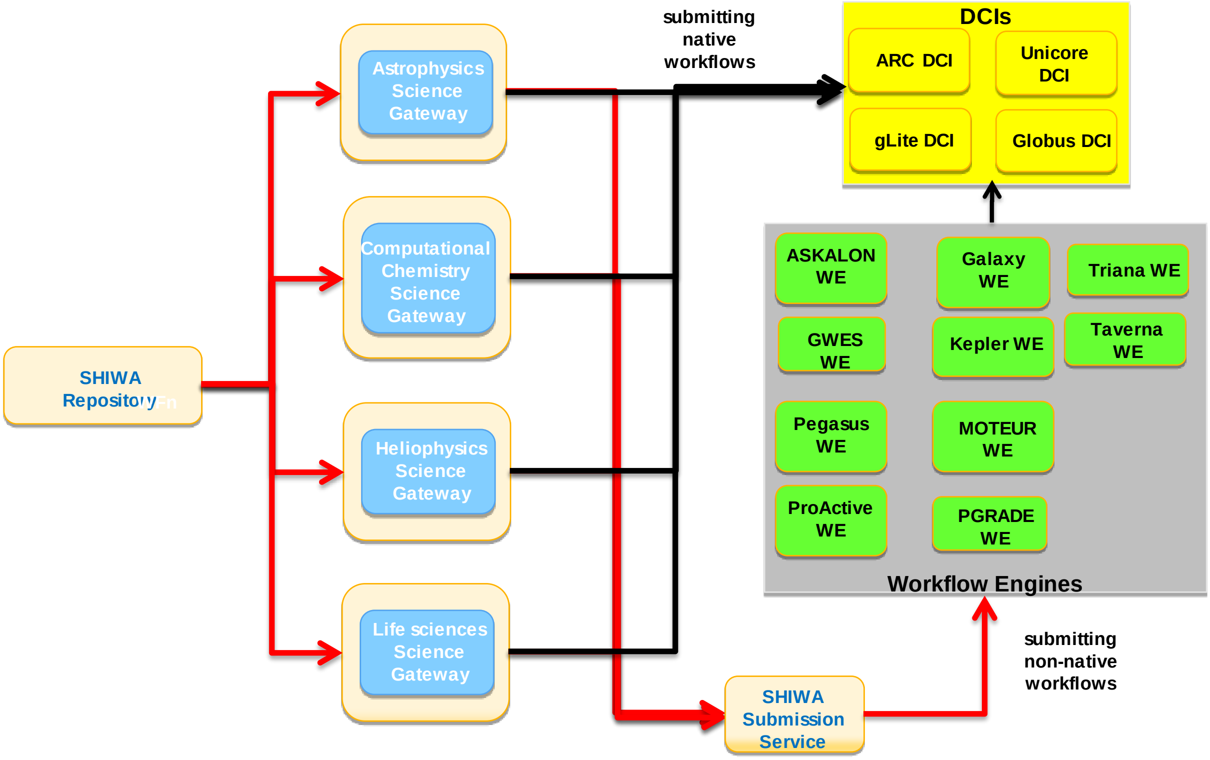


Figure 9.1: ER-flow execution environment

The ER-flow community gateways have the following versions before the workshop:

- Astrophyics v 3.5.2
- Computational Chemistry v 3.4.5
- Heliophysics v 3.5.2
- Life Sciences v 3.5.6

Considering the status of the community gateways the workshop will focus on two major tasks:

1. upgrading the SCI-BUS community science gateway to v.3.5.7
2. providing remote access to the SHIWA Repository and the SHIWA Submission Service.

Remark: The Computational Chemistry gateway will not be updated because of the UNICORE specific issues mentioned in 8.2.5.

[1] Summer School on Grid and Cloud Workflows and Gateways: http://www.lpds.sztaki.hu/summerschool2013/

10 Conclusion

The ER-flow research communities selected the workflows which they want to execute on the SHIWA Simulation Platform. WP3 demonstrated the simulation platform and trained the researchers of the Astrophysics, Computational Chemistry, Heliophysics and Life Science community to use the platform. These four research communities and the ER-flow technology providers analysed the features and services of the simulation platform considering the requirements of these communities and the selected workflows.

The research communities and the technology providers elaborated the ER-flow application support strategy. It incorporates two environments: the ER-flow development environment and the ER-flow execution environment. The development environment is the SHIWA Simulation Platform, while the execution environment contains four community science gateways developed within the framework of the SCI-BUS project. The ER-flow project partners defined enhancement requests considering the community requirements at one side and the current status of the simulation platform and the community gateways at the other side.

WP3 implemented five major and eleven minor enhancement requests defined in phase 1 and phase 2 of community feedback. As a result of these enhancements both the ER-flow development and the ER-flow execution environment provides more advanced and better features and services than previously.

Important new features and services of the ER-flow development environment

- automatic export-import of workflows
- robot certificate management
- upgraded statistic data management
- improved domain and sub-domain management
- community-based workflow validation

Selected new features and services of the ER-flow execution environment

- automatic export-import of workflows
- robot certificate management
- upgraded statistic data management
- remote access to the SHIWA Submission Service to support execution on non-native workflows
- direct URL to workflows

WP3 is working on one major enhancement request (upgrade the SHIWA Submission Service including workflow engine data management) and on several minor enhancement requests to further improve both the development and the execution environment.

References

[5.1] SHIWA: www.shiwa-workflow.eu

[6.1] Peter Kacsuk: P-GRADE portal family for grid infrastructures, Concurrency and Computation: Practice and Experience, Special Issue: IWPLS 2009, Volume 23, Issue 3, pages 235–245, 10 March 2011

[6.2] Thomas Fahringer, Radu Prodan, Rubing Duan, Jüurgen Hofer, Farrukh Nadeem, Francesco Nerieri, Stefan Podlipnig, Jun Qin, Mumtaz Siddiqui and Hong-Linh Truong, et al: ASKALON: A Development and Grid Computing Environment for Scientific Workflows, Workflows for e-Science, 2007, Part III, 450-471, DOI: 10.1007/978-1-84628-757-2_27

[6.3] Goecks, J.; Nekrutenko, A.; Taylor, J.; Galaxy Team, T: "Galaxy: A comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences". *Genome Biology* **11** (8): R86. doi:10.1186/gb-2010-11-8-r86. PMC 2945788. PMID 20738864. http://www.pubmedcentral.nih.gov/articlerender.fcgi?tool=pmcentrez&artid=2945788.

[6.4] GWES - http://www.gridworkflow.org/kwfgrid/gwes-web

[6.5] Bertram Ludäscher, Ilkay Altintas, Chad Berkley, Dan Higgins, Efrat Jaeger, Matthew Jones, Edward A. Lee, Jing Tao, Yang Zhao: Scientific workflow management and the Kepler system, Concurrency and Computation: Practice and Experience, Special Issue: Workflow in Grid Systems, Volume 18, Issue 10, pages 1039–1065, 25 August 2006

[6.6] T. Glatard, J. Montagnat, D. Lingrand, X. Pennec: "Flexible and efficient workflow deployement of data-intensive applications on grids with MOTEUR" in International Journal of High Performance Computing Applications (IJHPCA), 22(3):347-360, SAGE, August 2008.

[6.7] Z. Farkas and P. Kacsuk: P-grade portal: A generic workflow system to support user communities. Future Generation Computer Systems, 27(5):454 - 465, 2011.

[6.8] Ewa Deelman , Gurmeet Singh , Mei-hui Su , James Blythe , Yolanda Gil , Carl Kesselman , Gaurang Mehta , Karan Vahi , G. Bruce Berriman , John Good , Anastasia Laity , Joseph C. Jacob , Daniel S. Katz: Pegasus: a framework for mapping complex scientific workflows onto distributed systems, Scientific Programming Journal, 2005, Vol. 13, pp 219-237

[6.9] BR Odgers, JW Shepherdson, SG Thompson: Distributed workflow co-ordination by proactive software agents, In Proceedings of Intelligent Workflow and Process Management, IJCAI-99 Workshop, 1999

[6.10] D. Hull, K. Wolstencroft, R. Stevens, C. Goble, M. Pocock, P. Li, and T. Oinn: Taverna: a tool for building and running workflows of services. In Nucleic Acids Research, vol. 34, Web Server issue, pp. 729-732, 2006.

[6.11] I. Taylor, M. Shields, I. Wang, A. Harrison: The Triana Workflow Environment: Architecture and Applications. In Workflows for e-Science, pages 320-339. Springer, New York, Secaucus, NJ, USA, 2007.