1 SHIWA Workflow Repository Usage Policy

The SHIWA Workflow Repository has a role-based access policy. This policy defines four roles. Table 1 and Table 2 present these roles and operations enabled for these roles.

	create/modify/ delete users	manage user profile	create/modify /delete engines	create/modify/ delete groups
E-Scientist				
Workflow Developer		x		x
Validator		x		
Repository Administrator	x	x	x	x

Table 1: Group, engine and user management

	add/modify/ delete workflows	add/modify/ duplicate/delete implementations	browse/search & download workflows	browse/search & download implementations	Validate
E-Scientist			x	x	
Workflow Developer	x	x	x	x	
Validator			x	x	x
Repository Administrator	x	x	x	x	

Table 2: Workflow and implementation management

1.1 Managing users, groups and engines in the SHIWA Workflow Repository

The Repository Administrator is the only actor who can create modify and delete users and engines on the “Users/User” and “Engines/Engine” page. The Group Owners (Workflow Developers and Repository Administrator) can create, modify and delete groups and “Groups/Group” page.

SHIWA is supported by the FP7 Capacities Programme under contract nr RI-261556

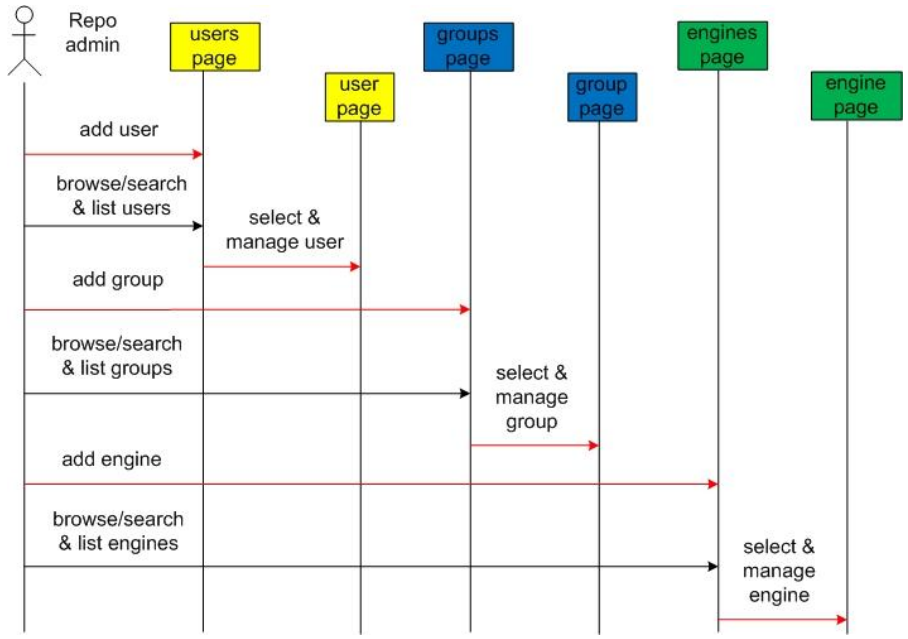

Figure 1: Managing users, groups and engines

1.1.1 Create a new user (Repository Administrator)

On the "Users" page clicking on the "Action menu's "New" tab, the "New User" page pops up. The Repository Administrator can

step 1: define the user's data (See Table 3)

step 2: specify the user's role (See Table 4)

login name	full name	organisation	password	e-mail

Table 3: user's data

	value
role	active
	validator
	administrator

Table 4: user's role

1.1.2 Create a new group (Workflow Developers)

On the "Groups" page clicking on the "Action's menu "New" tab, the "New Group" page pops up where Group Owners can

step 1: define the group's data (See Table 5)

name	owner

Table 5 : group's data

Remark: The actor who creates the group will be the initial group owner (or group leader).

SHIWA is supported by the FP7 Capacities Programme under contract nr RI-261556

1.1.3 Add a new engine (Repository Administrator)

On the “Engines” page clicking on the “Action menu “New” tab, the “New Engine” page pops up where Repository Administrator can.

step 1: define the engine’s data (See Table 6)

name	version	description

Table 6: engine’s data

1.2 Managing workflows and their implementations in the SHIWA Workflow Repository

Two actors, the Workflow Developers and the Repository Administrator can create, modify and delete workflows and their implementations using the “Workflows/Workflow” and “Implementations/Implementation” page.

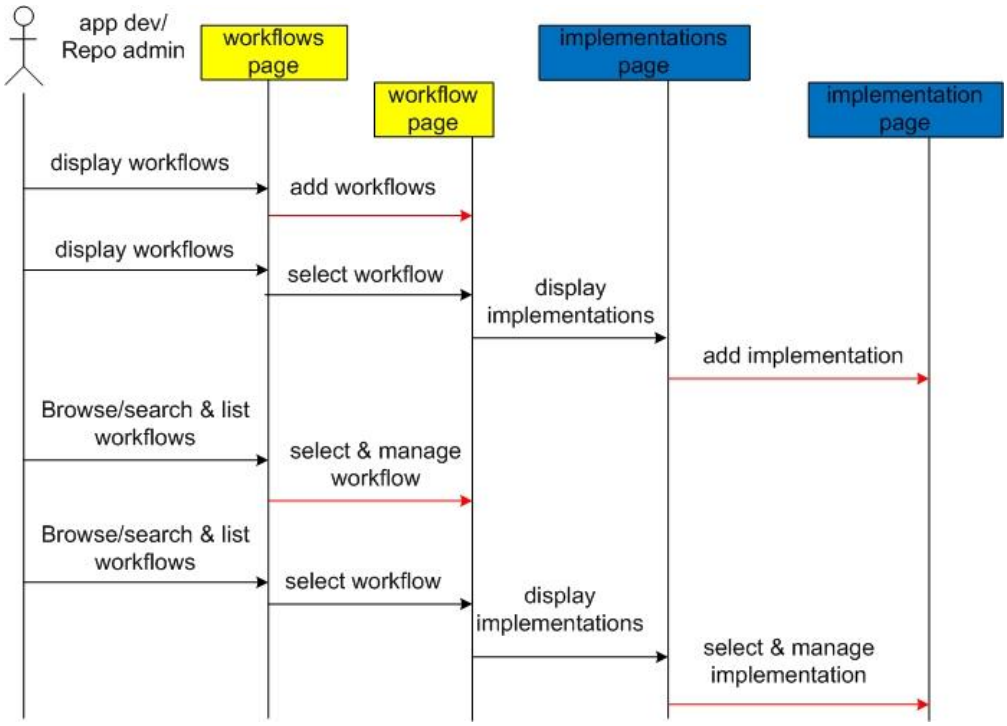

Figure 2: Managing workflows and their implementations

1.2.1 Add a new workflow

On the “Workflows” page clicking on the “Action menu’s “New” tab, the “New Workflow” page pops up where the Workflow Owners (Workflow Developers or Repository Administrators) can

step 1: define the workflow’s data and associate the new workflow with a group (See Table 7)

name	group	description

Table 7: workflow’s data

step 2: set access rights of the workflow (See Table 8)

	read	download	modify
group	True/False	True/False	True/False
others	True/False	True/False	N/A

Table 8: access rights

Remark: The initial workflow owner is the actor who added the workflow to the repository.

Remark: If the workflow owner wants to associate the new workflow with a new group, the owner should create a new group before adding a new workflow to the repository. (See 1.1.2)

Remark: If you enable others to read/download your workflow, it will be accessible to registered users only. A validator sees that the access rights of your workflow have been changed and will look at your workflow to make sure that the provided data is correct. Once the validator accepts your workflow, it's status will change to validated and it will be publicly available to guest users as well. Once a workflow has been validated, it cannot be modified.

Remark: See Figure A.1 (in the appendix) for a prototype workflow description.

1.2.2 Define workflow attributes

On the "Workflow" page clicking on the "Attributes" tab the Attributes page pops up where the Workflow Owners (Workflow Developers or Repository Administrators) can

step 1: define the workflow's interface (See Table 9)

Interfaces	Data			
Inports	Port ID	Title	Description	DataType
Outports	PortID	Title	Description	DataType

Table 9: workflow interface

Remark: It is always recommended to define all input and output files of the workflow. (Note that all port ids independently of their direction have to be unique.)

step 2: specify the workflow's dataset (See Table 10)

dataset ID	description	port reference	value

Table 10: workflow dataset

Remark: Dataset is a parameterization of the inports and outports of a workflow. It is recommended to define at least one dataset that provides a set of sample input files and the output files.

Remark: The Workflow Owners should upload all workflow files referred in the workflow datasets.

step 3: define tasktype, application, workflow domain and keywords that provide additional information about the workflow (See Table 11)

SHIWA is supported by the FP7 Capacities Programme under contract nr RI-261556

tasktype	application	domain	keywords

Table 11: workflow's tasktype, application, domain and keywords

Remark: See Figure A.3.simple and A.3.meta (in the Appendix), for typical workflow attribute specifications of the prototype simple and meta-workflows (respectively).

1.2.3 Add a new implementation to an workflow

Selecting the "Implementations" tab on the "Workflow" page displays the "Implementations" page. Clicking on the "Action menu's "New" tab, the "New Implementation" page pops up where the Workflow Developers and the Repository Administrators can

step 1: select the implementation's engine and define its version (See Table 12)

engine	Version	Gemlca ID

Table 12: implementation data

Remark: If the required engine does not exist the Workflow Developers should send an e-mail message to the Repository Administrator with the engine name and description to create the new engine.

Remark: The version can be given by any string but the preferred form is [0-9].[0-9], for example: '1.0'

Remark: To duplicate an implementation, use the *Actions->Duplicate*. This can be used to incrementally develop an implementation, leaving the existing version intact.

1.2.4 Define implementation attributes

On the "Implementation" page clicking on the "Attributes" tab the Attributes" page pops up where the Workflow Developers and the Repository Administrators can

step 1: define the implementation's dependencies (See Table 13)

dependency ID	title	description	Type

Table 13: implementation's dependencies

Remark: Define dependencies of the executable/bundle. They may have further executables, scripts, or libraries.

Remark: Upload all implementation files, most importantly the workflow definition file and graph previews, and the files required to resolve the dependencies.

Remark: If VO membership is required for execution, this is to be defined as DCI dependency.

step 2: specify the implementation's configuration (See Table 14)

configuration ID	dependency reference	value

Table 14: implementation's configuration

Remark: Define exactly one configuration for each implementation to resolve all its dependencies.

step 3: give the implementation's title, description, definition, language, rights, licence and keywords (See Table 15)

Title	Description	Definition	Graph	Language	Rights	Licence	keywords	uuid

Table 15: implementation's details

Remark: In the case of a service grid implementation please provide the filename of the implementation executable, while in the case of a desktop grid implementation please provide the filename of the implementation bundle.

Remark: See Figures A.7.simple and A.7.meta (in the appendix), for typical implementation attribute specifications of the prototype simple and meta-workflows (respectively).

1.3 Validation in the SHIWA Workflow Repository

After uploading a new implementation into the repository, Workflow Developers set the implementation's status to "submit for validation".

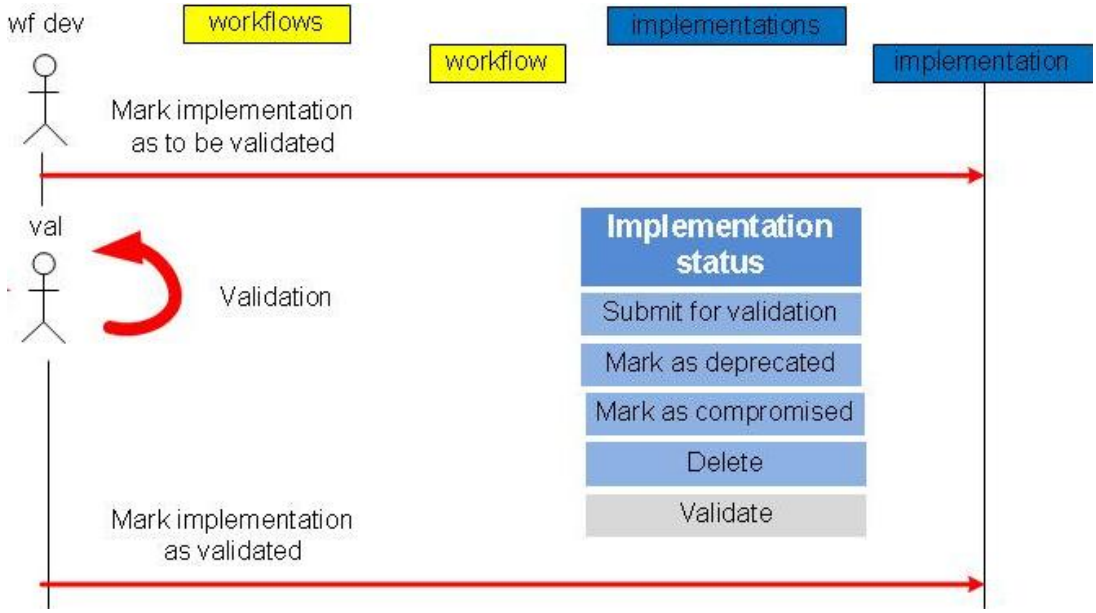

Figure 3: Validating implementations

In validator view, the list of "Implementations ready for validation" is provided.
Validators test the implementation and change its status to "validated" after successful tests.

To ensure that the abstract workflow data is valid, workflows can be validated as well by validators or administrators after they checked whether the provided workflow data and description is sound and correct.
Once a workflow is made accessible to others, it will be accessible to registered users only and will be automatically submitted for validation. In validator view, the list

of "workflows to be validated"
is provided. Only validated workflows and their validated implementations are available to non-registered, guest users. However, a workflow can be validated even if it has no validated implementations, since the fact that a workflow is validated only means that the abstract workflow description is correct and sound, but does not imply that its implementations are valid as well. Note that validated workflows and implementations cannot be modified. If a workflow developer has to change a validated workflow or an implementation, an administrator or validator has to be contacted.

1.4 Browsing and searching workflows and implementations in the SHIWA Workflow Repository

E-scientists are the end users of the SHIWA Workflow Repository. They can access the repository without registration. They can browse and search workflows and implementations, and download implementations.

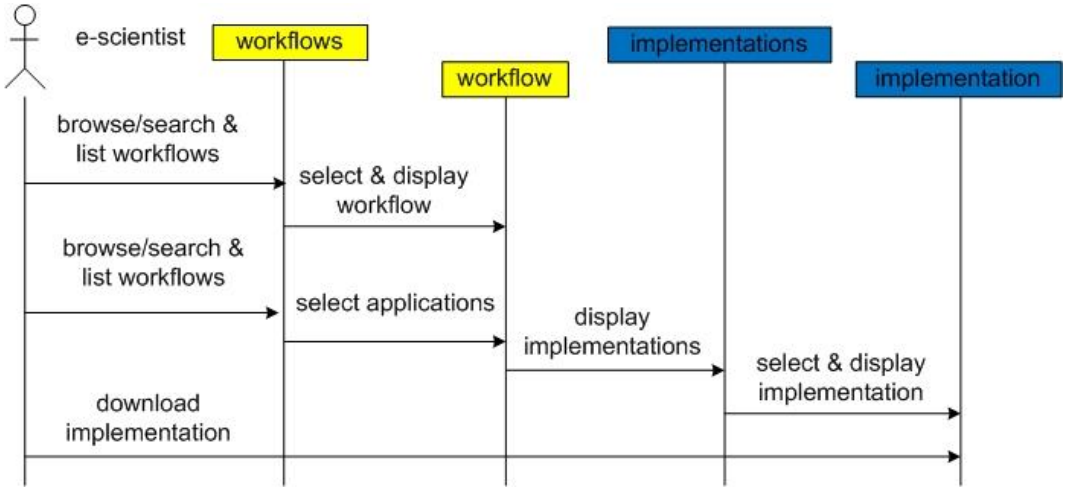

Figure 4: Browsing and searching workflows and implementations

1.5 Managing user profile

Workflow developers and Administrators can access the ”My Workflow”, “My Groups”, “My Owned Groups”, “My Details” and “My Password” tabs.

On the “Home” page clicking on the “My details“ tab the My Details” page pops up where the actors can change their profile data.

step 1: modify their own data (See Table 16)

name	organisation	e-mail

Table 16: user’s data

step 2: change their password (See Table 17)

password

Table 17: user’s password

Appendix – prototype workflow in SHIWA repository

Foreword:

This appendix contains screen shots of a prototype SHIWA repository representation of the simple workflow SimpleWF_IntegerSubtractor; Figures A.1, A.2, A.3.simple, A.4, A.5, A.6, A.7.simple & A.8).

This appendix also includes screen shots of a prototype SHIWA respository representation of the meta-workflow MetaWF_ImageManipulation, where there are substantial differences between the respesentation of simple and meta-workflows; Figures A.3.meta and A.7.meta.

-

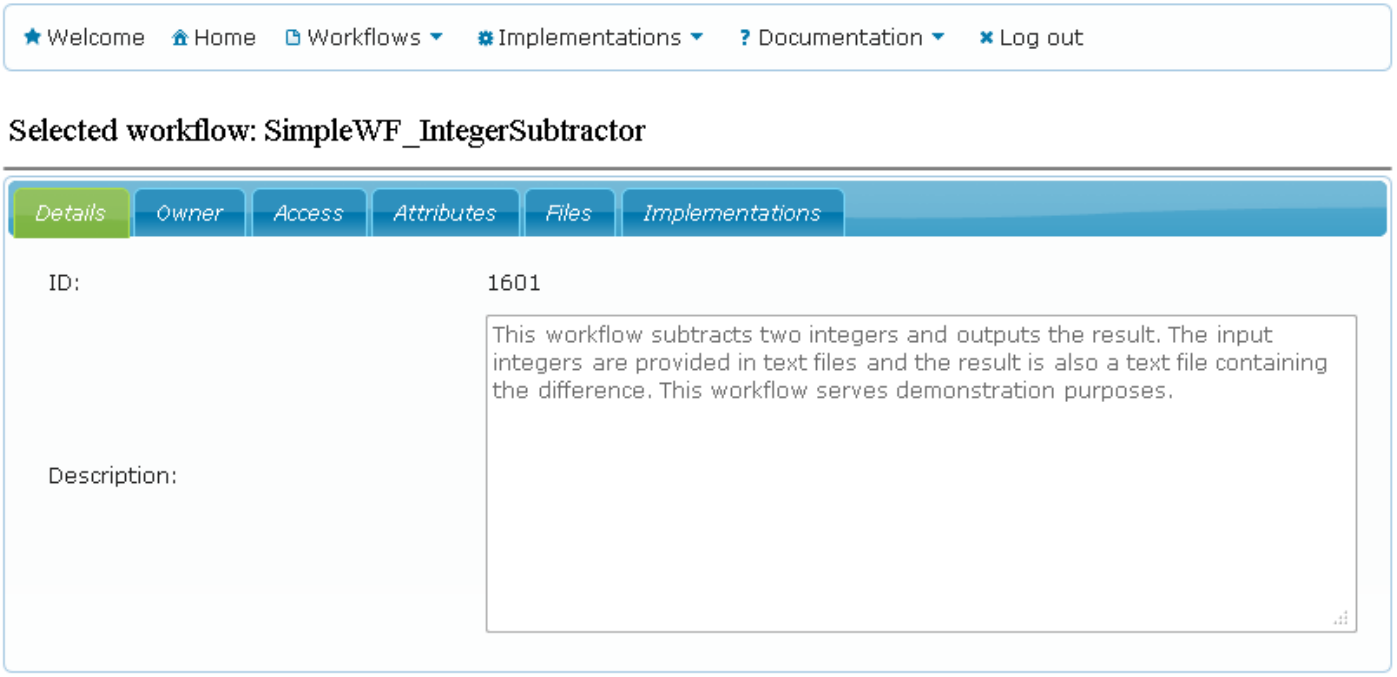

Figure A.1

Figure A.2

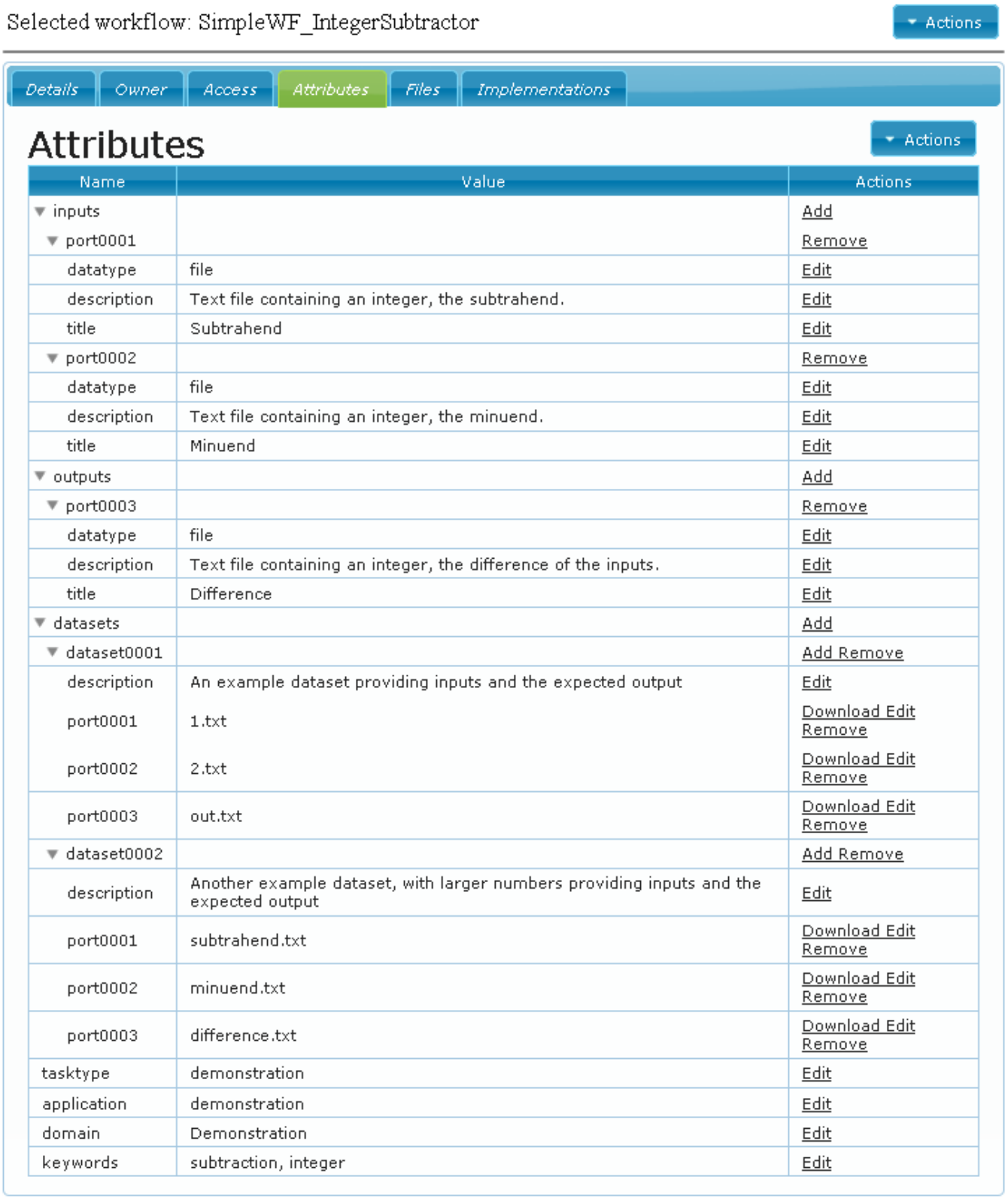

Figure A.3.simple

Selected workflow: MetaWF_ImageManipulationDemo

Actions

Details | Owner | Access | Attributes | Files | Implementations

Attributes

Actions

Name	Value	Actions
▼ inputs		Add
▼ port0001		Remove
datatype	file	Edit
description	This file contains the data request for fetching images from a given web service.	Edit
title	Input zip	Edit
▼ outputs		Add
▼ port0002		Remove
datatype	file	Edit
description	This file contains the manipulated images.	Edit
title	Output zip	Edit
▼ datasets		Add
▼ dataset0001		Add Remove
description	An example dataset providing an input	Edit
port0001	example_input.zip	Download Edit Remove
tasktype	demonstration	Edit
application	Shiwa Image Manipulation Demo	Edit
domain	Demonstration	Edit
keywords	web service, images, edge highlighting, RGB	Edit

Figure A.3.meta

Selected workflow: SimpleWF_IntegerSubtractor

Actions

Details | Owner | Access | Attributes | Files | Implementations

Files

Actions

25 (1 of 1)

Filename
1.txt
2.txt
difference.txt
minuend.txt
out.txt
subtrahend.txt

Figure A.4

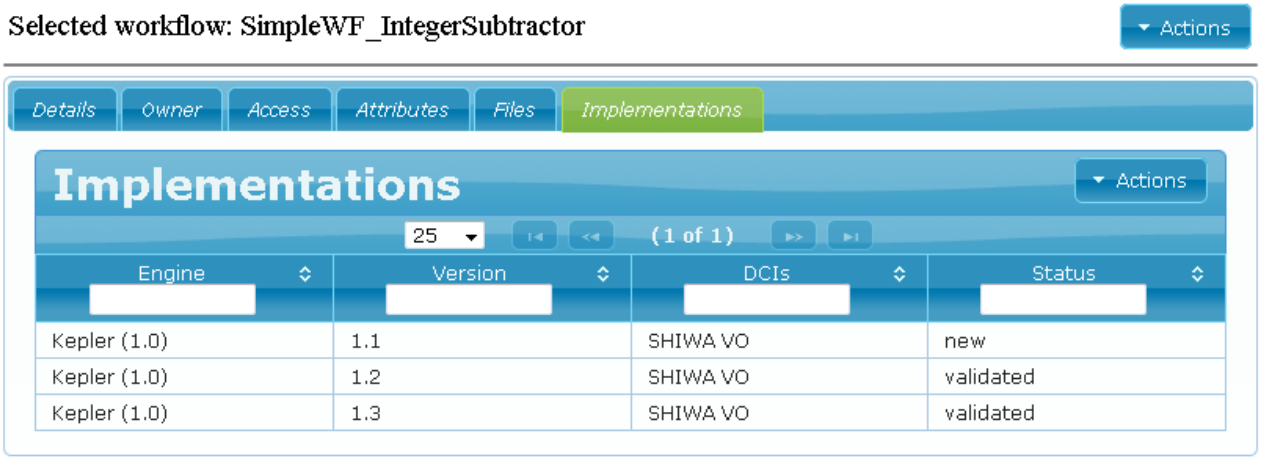

Figure A.5

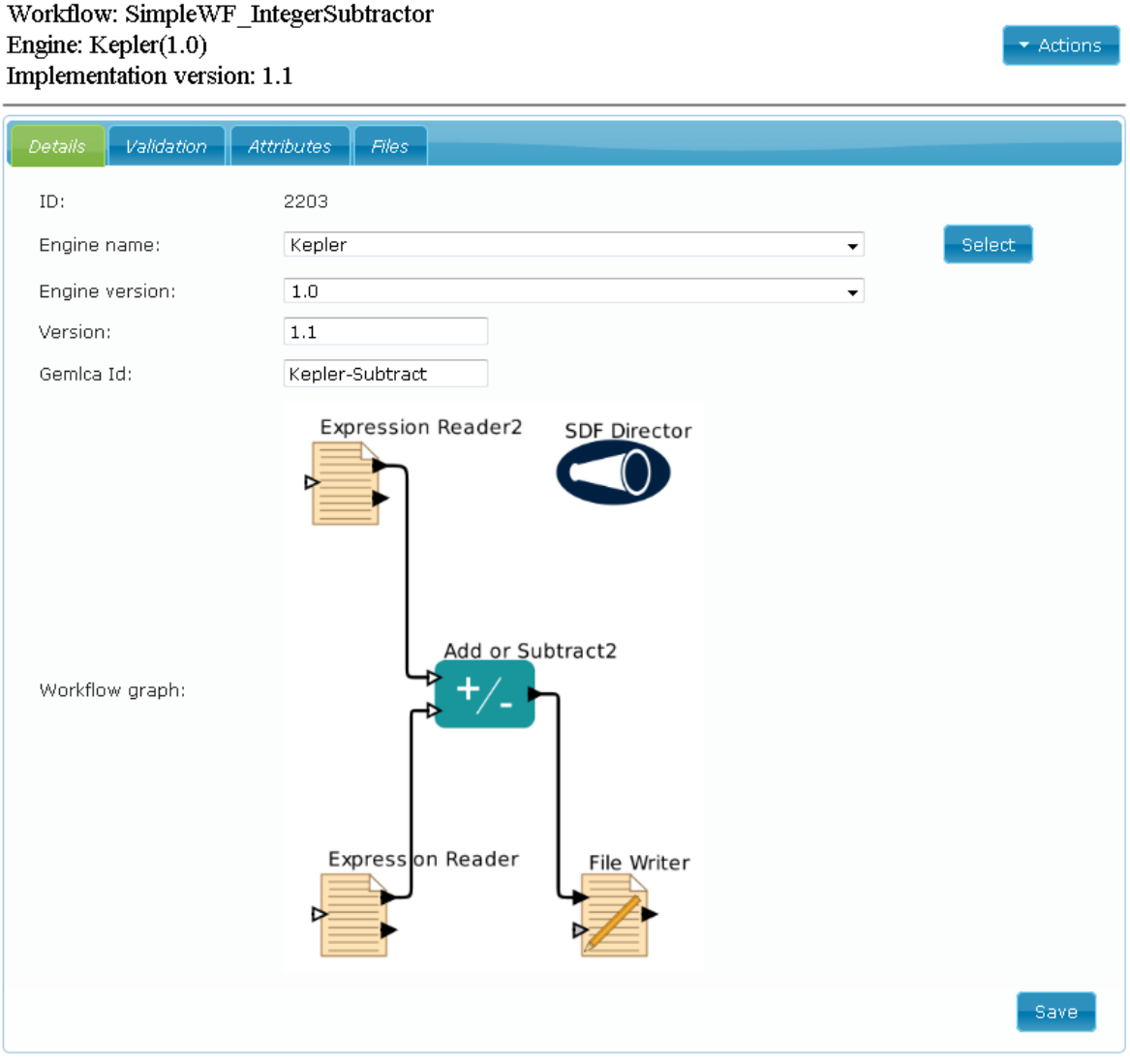

Figure A.6

Workflow: SimpleWF_IntegerSubtractor
Engine: Kepler(1.0)
Implementation version: 1.1

Actions

Details | Validation | Attributes | Files

Attributes

Actions

Name	Value	Actions
▼ dependencies		Add
▼ dep0001		Remove
type	DCI	Edit
description	The VO the user has to be member of to execute this workflow in the SSP.	Edit
title	VO for execution in SSP	Edit
▼ configurations		Add
▼ conf0001		Add Remove
dep0001	SHIWA VO	Edit Remove
title	Kepler Subtract Demo	Edit
description	This workflow is executed locally to the Kepler engine.	Edit
definition	Kepler_wf1.xml	Download Edit
graph	Kepler_wf1.png	Download Edit
language	MOML	Edit
rights	SHIWA project	Edit
licence	Apache License, Version 2.0	Edit
keywords	Kepler, local, subtract, integer	Edit
uuid		Edit

Figure A.7.simple

Workflow: SimpleWF_IntegerSubtractor
Engine: Kepler(1.0)
Implementation version: 1.1

Actions

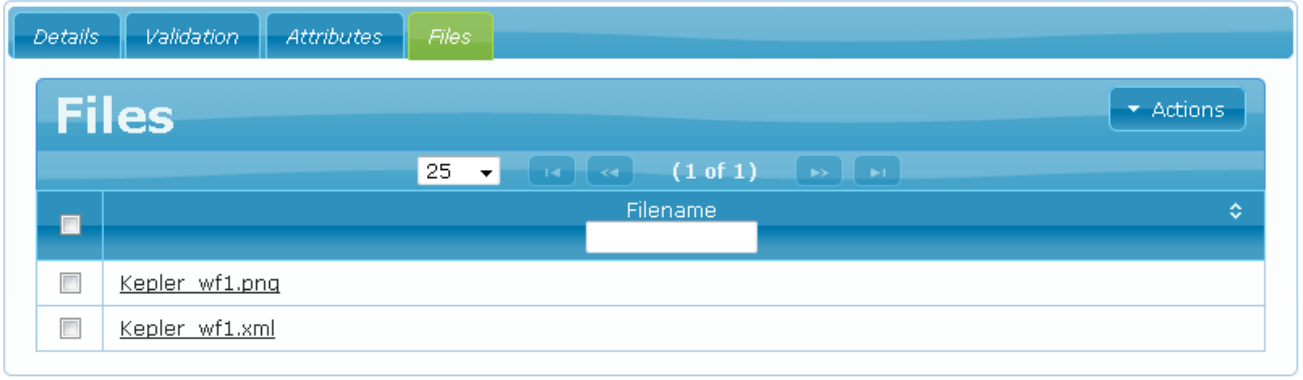

Figure A.7

Workflow: SimpleWF_IntegerSubtractor
Engine: Kepler(1.0)
Implementation version: 1.1

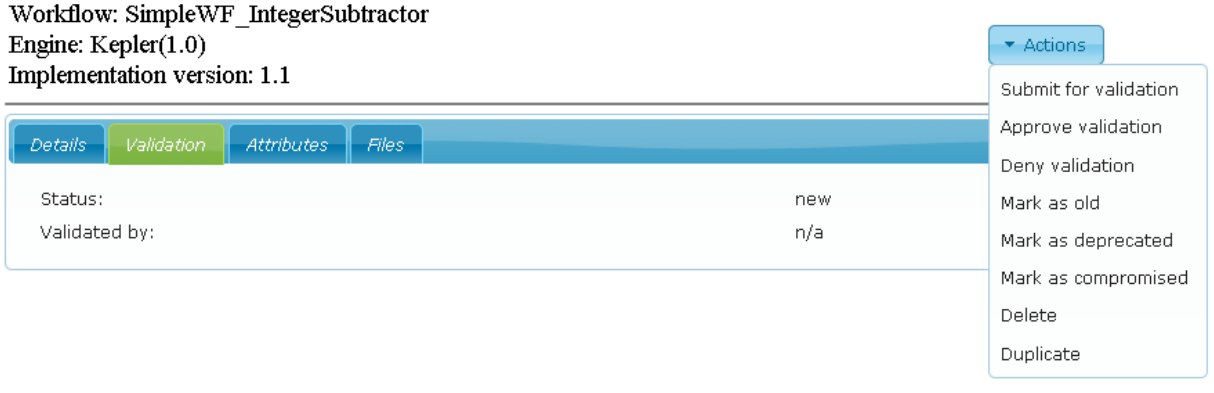

Figure A.8

Workflow: MetaWF_ImageManipulationDemo
Engine: P-GRADE(2.4.1)
Implementation version: 1.0

Actions

Details | Validation | Attributes | Files

Attributes

Actions

Name	Value	Actions
▼ dependencies		Add
▼ dep0001		Remove
type	Sub Workflow	Edit
description	An implementation of the FetchImages workflow.	Edit
title	SubWorkflow1	Edit
▼ dep0002		Remove
type	Sub Workflow	Edit
description	An implementation of the EdgeHighlighting workflow.	Edit
title	SubWorkflow2	Edit
▼ dep0003		Remove
type	Sub Workflow	Edit
description	An implementation of the RGBColorComponents workflow.	Edit
title	SubWorkflow3	Edit
▼ dep0004		Remove
type	Sub Workflow	Edit
description	An implementation of the DirectoryList workflow.	Edit
title	SubWorkflow4	Edit
▼ dep0005		Remove
type	DCI	Edit
description	The VO the user has to be member of to execute this workflow in the SSP.	Edit
title	VO for execution in SSP	Edit
▼ configurations		Add
▼ conf0001		Add Remove
dep0001	FetchImages/Taverna(1.7)/1.0	Edit Remove
dep0002	EdgeHighlighting/Kepler(1.0)/1.0	Edit Remove
dep0003	RGBColorComponents/Triana(3.2.3)/1.0	Edit Remove
dep0004	DirectoryList/MOTEUR(0.1)/Moteur-2.0.1-GIB	Edit Remove
dep0005	SHIWA VO	Edit Remove
title	ImageManipulationDemoMetaWF	Edit
description	This P-GRADE metaworkflow embeds workflows of different kinds for demonstration purposes.	Edit
definition	ImagesDemoMetaWF.zip	Download Edit
graph	pgrade-multiImage.png	Download Edit
language	Condor DAG	Edit
rights	SHIWA project	Edit
licence	Apache License, Version 2.0	Edit
keywords	P-GRADE, meta workflow, image manipulation, demo	Edit
uuid		Edit

Figure A.7.meta