



SHIWA Portal tutorial 4 -

Embedding the meta-workflow from Tutorial 3 within a job of a meta-meta-workflow.

Submission Service version, 03/06/2014

Noam Weingarten (weingan@wmin.ac.uk)

Contents

Goals of the tutorials and definitions	2
Native workflows	2
Non-native workflows.....	2
Meta-workflow	2
Meta-meta-workflow.....	2
Usage of the SHIWA 2 Portal	3
Tutorial 4: Embedding the meta-workflow from Tutorial 3 within a job of a meta-meta-workflow.	4
4.1 Preparing the pre-existing meta-workflow for embedding.....	4
4.2 Creating Templated (or Genuine Embedded) workflow	5
4.3 Preparing graph of the meta-meta-workflow.....	5
4.4 Creating a concrete for the meta-meta-workflow.....	6
4.5 Configuring job of the meta-meta-workflow.....	6
4.6 Verifying and submitting meta-meta-workflow	7



Goals of the tutorials and definitions

The aim of this tutorial is to introduce the basic features provided by the SHIWA Simulation Platform.

The following features will be covered by this tutorial:

- Creating, configuring and submitting a meta-meta-workflow, including a job, which is configured to contain an existing meta-workflow as an embedded workflow – this embedded meta-workflow may consist of both native and non-native workflows.

Native workflows

For the purpose of this tutorial, the term *native workflows* will refer to a standard job in a WS-PGRADE workflow which executes an executable uploaded to the portal during configuration of the WS-PGRADE workflow.

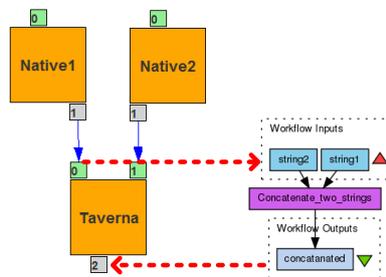
Non-native workflows

For the purpose of this tutorial, the term *non-native workflow* will refer to a job in a WS-PGRADE workflow, which executes a workflow of some workflow engine, is selected from a SHIWA Repository and submitted using the Submission Service.

Meta-workflow

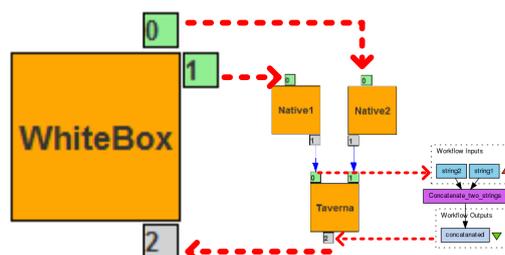
For the purpose of this tutorial, the term *meta-workflow* will refer to a WS-PGRADE workflow which consists of more than one workflow embedded within. These embedded workflows may be native, non-native or a mixture.

The graphic below is of the meta-workflow from the second tutorial. The two native jobs are the job from Tutorial 1 (although the job from Tutorial 2 would work just the same) . The job *Taverna* is a non-native job, which executes the Taverna 2.4 workflow in the graphic.



Meta-meta-workflow

For the purposes of this tutorial, the term *meta-meta-workflow* will be used to describe a WS-PGRADE workflow which includes a job, which is configured to be interpreted as an existing meta-workflow in the form of an embedded workflow. The graphic below depicts this tutorial, where the above *meta-workflow* from tutorial 3 is embedded in a job in a WS-PGRADE workflow.





Usage of the SHIWA 2 Portal

The Prototype SHIWA 2 Portal is available at this address:

<http://ssp.shiwa-workflow.eu/>

If you own an account, then you can use it to log in. Otherwise, the next subsection will explain how you can create one.

This tutorial assumes that:

- you have a SHIWA Portal account,
- you have a grid certificate,
- you are a member of the required VO(s) (**V**oluntary **O**rganisations) for execution of the desired workflows – the workflow in this tutorial will work with VO shiwa.workflow.eu or Westfocus.

It is strongly recommended to complete tutorials 1, 2 and 3 before attempting this tutorial, as the above pre-requisites are covered.



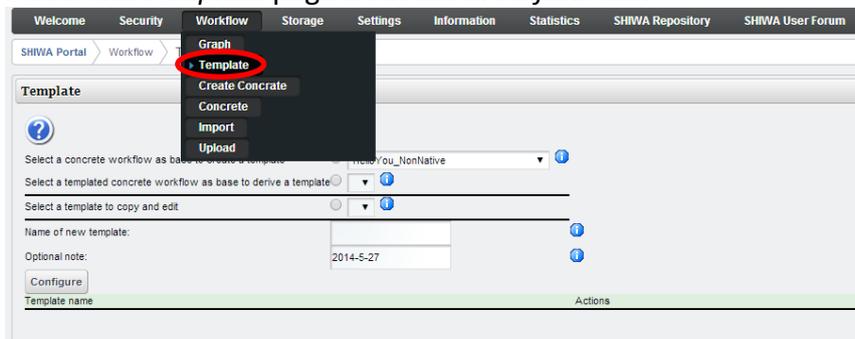
Tutorial 4: Embedding the meta-workflow from Tutorial 3 within a job of a meta-meta-workflow.

This tutorial will cover WS-PGRADE meta-workflows by the White-box approach. These are composite WS-PGRADE workflows, which consist of one or more pre-existing workflows. The pre-existing workflows will be embedded within jobs of the meta-workflows using the white-box approach. The meta-workflow from tutorial 3 will be used as the pre-existing workflow for embedding. The resulting workflow will be 3-layered meta-workflow, or meta-meta-workflow.

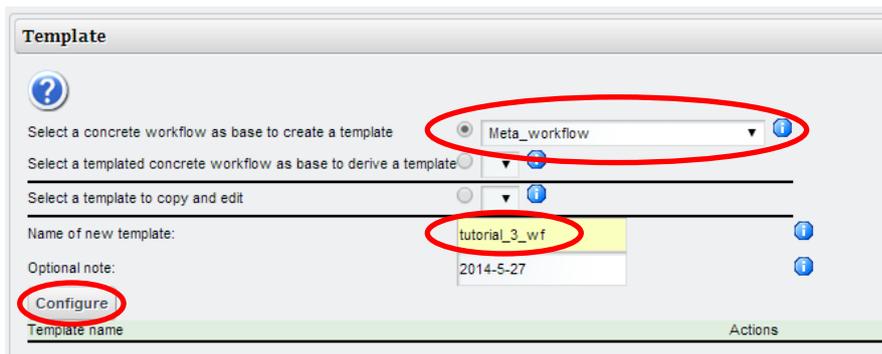
4.1 Preparing the pre-existing meta-workflow for embedding

Once tutorial 3 has been completed, and the meta-workflow is submitting and completing correctly, the pre-existing meta-workflow is ready for embedding within a meta-meta-workflow.

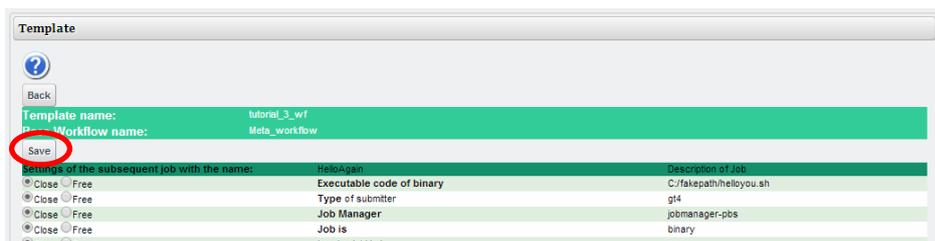
Select the *Template* page from the *Workflow* tab.



Specify the meta-workflow from tutorial 3 as the concrete workflow as a base to create a template from and give the template a name.



Click **configure**, leave all the details as closed, and click **save** to save the template.





4.2 Creating Templated (or Genuine Embedded) workflow

Create a concrete workflow from the template, to produce a Genuine Embedded workflow.

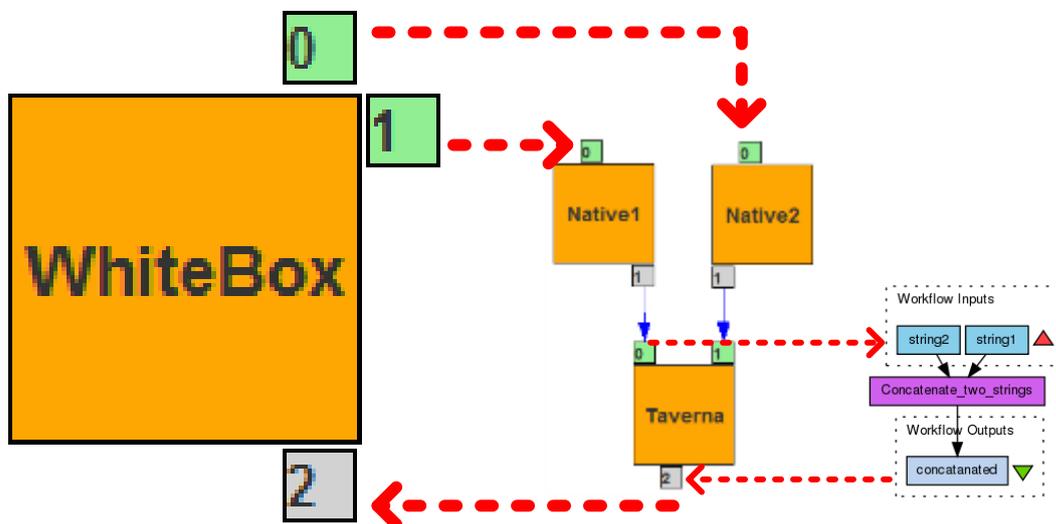


Select the template just created, and give the new Genuine Embedded workflow a name.

At this stage, the Genuine Embedded workflow should be verified, and submitted, to ensure it is correctly configured.

4.3 Preparing graph of the meta-meta-workflow

The meta-meta-workflow contains a single job, which has input and output ports to relate to the input and output ports of the workflow being embedded.



This graphic displays that the job in the meta-meta-workflow would require two inputs and one output port. These relate to the single inputs of job Native1 and Native 2, and the single output port of the job Taverna.

Prepare a graph with a single job, with two input ports and one output port. (Details on how to complete this stage are in the tutorials 1,2 & 3)



4.4 Creating a concrete for the meta-meta-workflow

Create a concrete based on the meta-meta-workflow graph that has just been created.
(Details on how to complete this stage are in the tutorials 1,2 & 3)

4.5 Configuring job of the meta-meta-workflow

The job of the meta-meta-workflow, within which the meta-workflow will be embedded must be configured for interpretation of the job as a **workflow**.
Then select the templated meta-workflow that was created in stage 4.2 .

Configure

Job's name: WhiteBox

Optional note: Description of Job

[Job Executable] [Job I/O] [IDL/RSL] [History]

Workflow Binary

The list box opens a selectable list of Workflows constrained by a Template. The selected Workflow will be embedded in this Job. In an eventual later configuration phase selected by the *Job Inputs and Outputs* tab the entries "In embedded Workflow" will enumerate the Ports of the embedded Workflow which may be associated with the Ports of the caller (or envelop) Job.

for embedding select a workflow created from a Template: Tutorial_3_meta_wf (tutorial_3...)

Replicate settings in all Jobs:

Assigned breakpoint:

- No breakpoint
- Before submission
- After termination

Blocker timeout of user interaction: 60

✓



4.6 Configuring the ports of the job containing the embedded workflow

The graphic of the meta-meta-workflow (in part 4.3) illustrates which ports to connect to which.

The Job I/O tab of this job will look different to what you have seen in jobs interpreted as binaries, in the previous tutorials.

Select **yes** to connect the port to Job/Input port of the embedded workflow.

Configure

Job's name: WhiteBox
Optional note: Description of Job

[Job Executable] [Job I/O] [JDL/RSL] [History]

Port Number:0 Port Name: in1 Description of Port

Input Port's Internal File Name: in1

Connect input port to a Job/Input port of the embedded WF: Yes No

Job/Input port: Native1/input
Native2/input

Source of input directed to this port: View Hide

Parametric Input details: View Hide

Port Number:1 Port Name: in2 Description of Port

Port Number:2 Port Name: out Description of Port

Select the proper input port of the embedded Workflow to be associated with the actual input port.

To help the user to identify the proper port, the list of possible ports is encountered by hierarchical names: concatenating relating Job and Port names separated by "/".

For input ports, you will also have to upload the input file in the manner covered in tutorial 1, unless the input file is channelled from another job in the meta-meta-workflow.

For output ports, you only have to configure the port connection.

Don't forget to save the configurations by clicking the green ticks at the bottom, and save the workflow.

4.6 Verifying and submitting meta-meta-workflow

The meta-meta-workflow is now ready to verify and submit in the method explained in previous tutorials.