

Nested workflows

<http://dev.mygrid.org.uk/wiki/display/tav250/Nested+workflows>

Stian Soiland-Reyes and Christian Brenninkmeijer
University of Manchester

<http://orcid.org/0000-0001-9842-9718>

<http://orcid.org/0000-0002-2937-7819>

Bonn University, 2014-09-01 / 2014-09-03

<http://www.taverna.org.uk/>



*This work is licensed under a
[Creative Commons Attribution 3.0 Unported License](http://creativecommons.org/licenses/by/3.0/)*

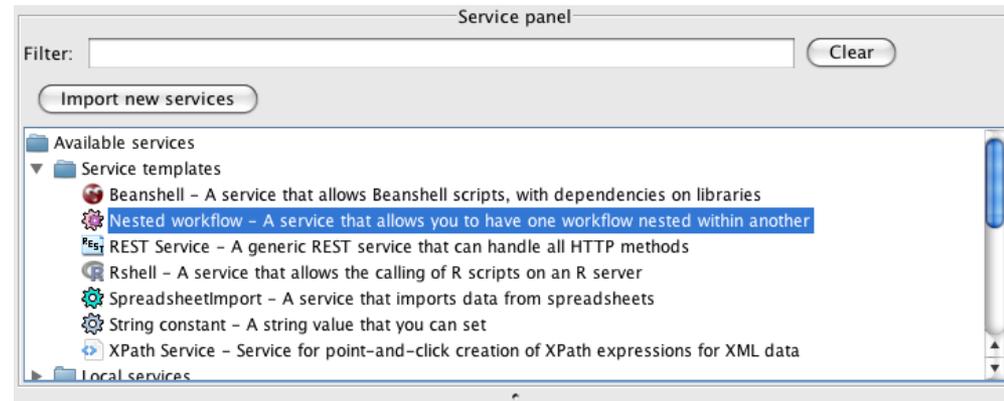


- A combination of services in a workflow can often be seen as playing a single functional role. For example:
 - A web service with *shims* that massage inputs and outputs
 - A series of services that perform a particular *scientific function* – e.g. filtering or merging
- **Nested workflows** is a way to group such workflows – represented as a delimited box
- Nested workflows can be **collapsed** to reduce clutter in the mother workflow
- Nested workflows can either be made from scratch, or by importing an **existing workflow**.
- A nested workflow can be **reused** in several mother workflows and are easier to replace/update.

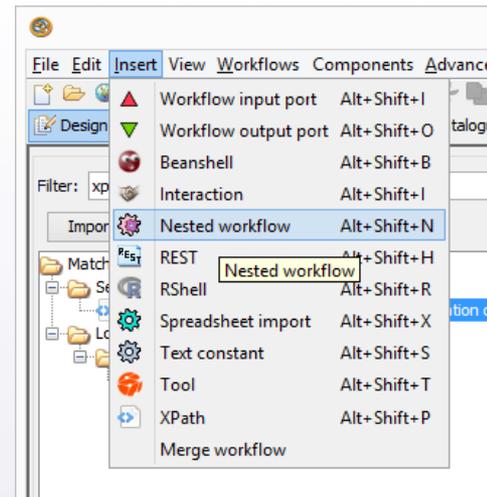


Adding a nested workflow

1. Start with File -> New workflow
2. Under **Available services**, expand **Service templates**
3. Select **Nested Workflow**.
4. Drag and drop the **Nested Workflow** to the **Workflow Diagram**.
5. In a dialog that pops up, you will have several options on how to *import* a nested workflow.



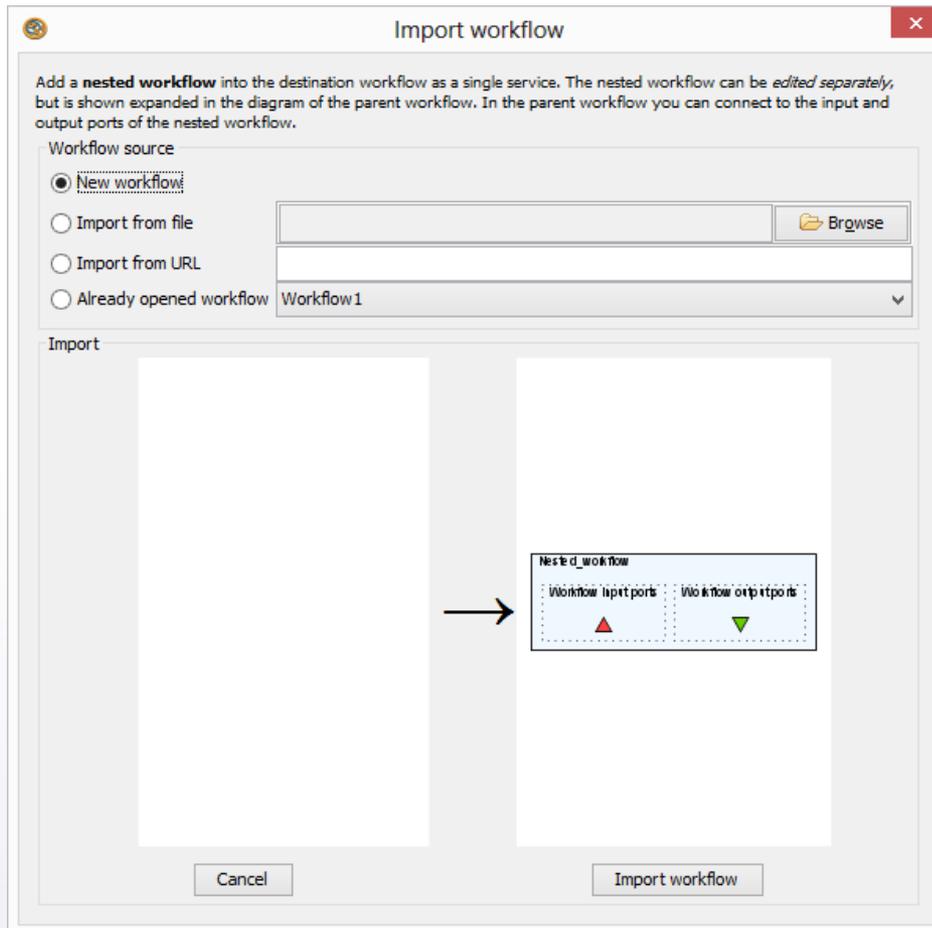
- Alternatively, in the menu, click **Insert -> Nested workflow**





New nested workflow

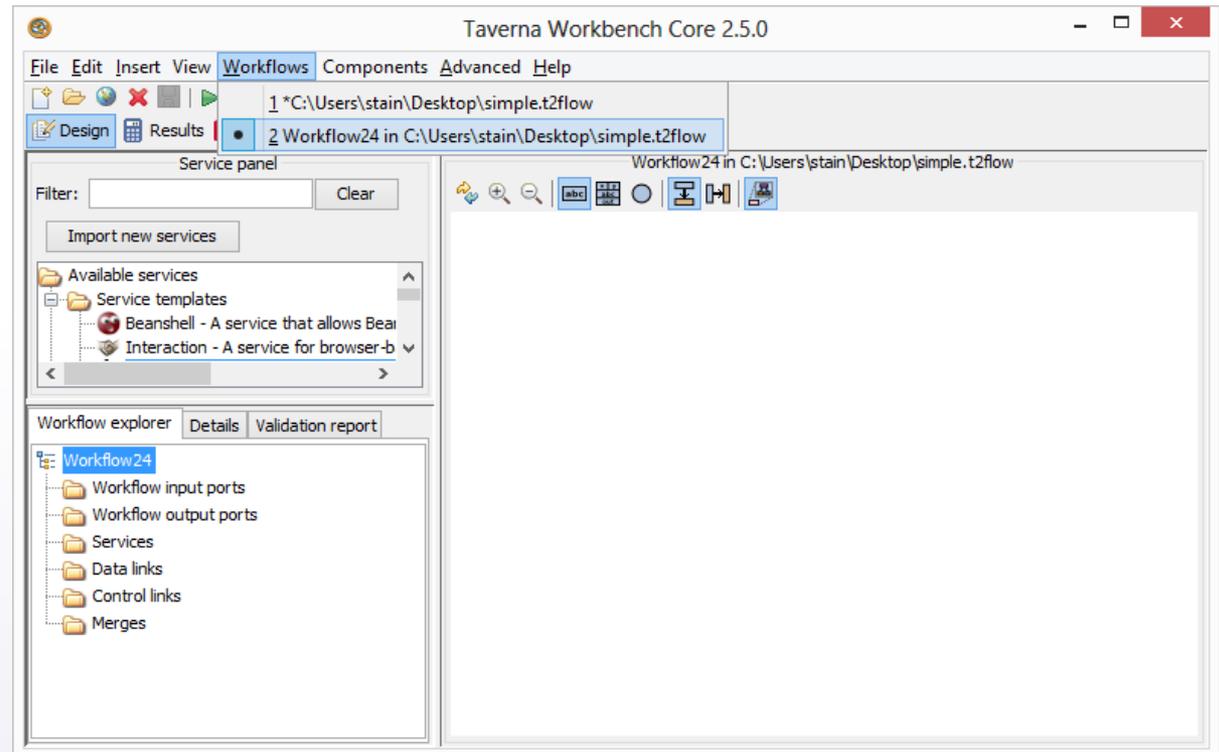
- Select **New workflow** and click **Import workflow**





Workflows menu

- What happened? Why is the *Design* perspective empty?
- Look in the **Workflows** menu
- Are you able to switch back and forth between the nested and mother workflow?





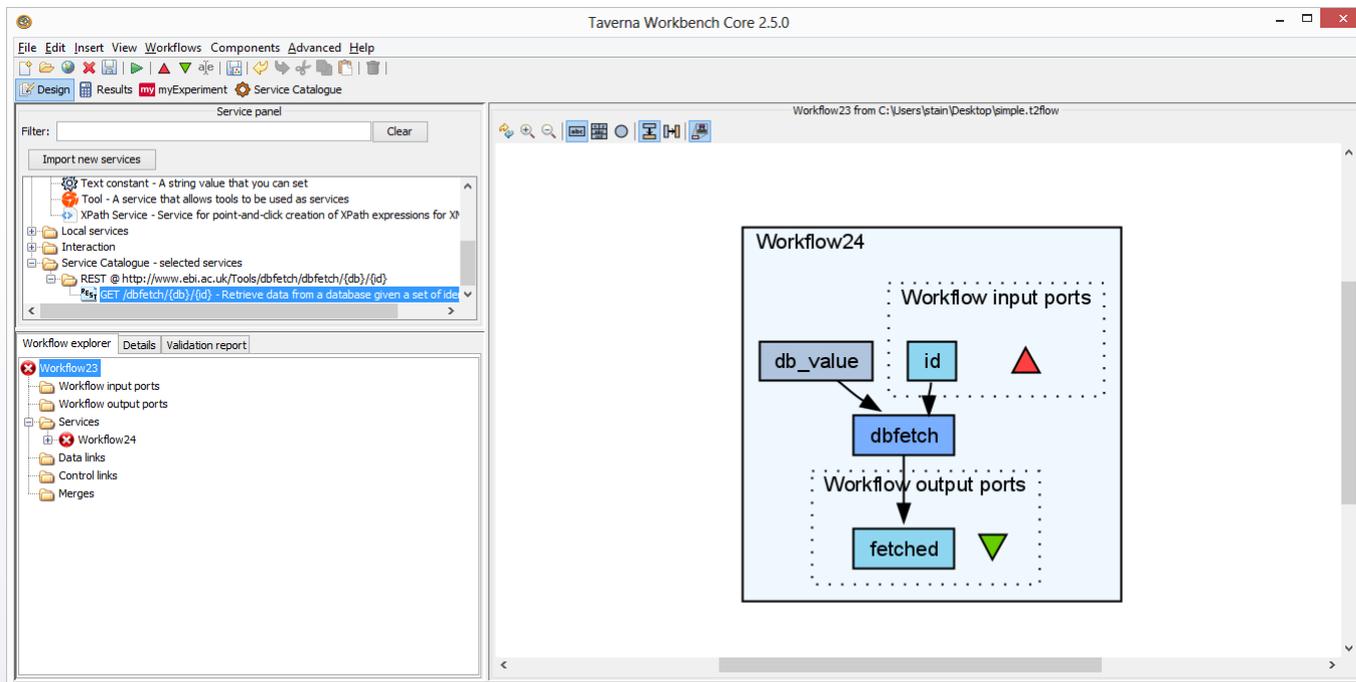
Building dbfetch workflow

- Editing the nested workflow, and add:
 - REST service *dbfetch*
 - As of tutorial 2, from Service Catalogue / Available Services
 - Or using the *URL template*
<http://www.ebi.ac.uk/Tools/dbfetch/dbfetch/{db}/{id}>
 - Workflow Input Port *id*
 - Workflow Output Port *fetched* (nicer name than responseBody)
 - Text Constant *db* – value uniprotkb
- Connect dbfetch to workflow ports and text constant
- To test, run the nested workflow separately, using the input P15409



Saving nested workflow

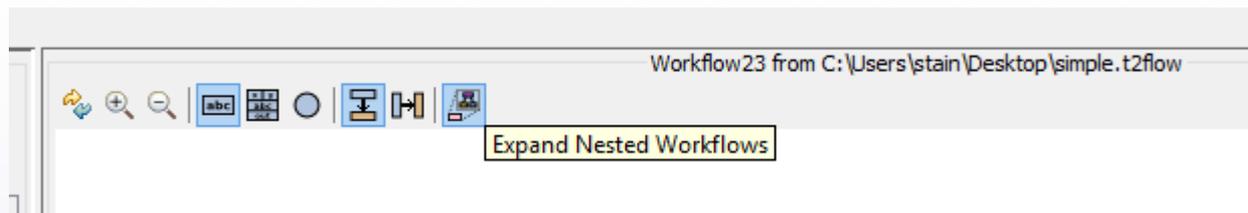
- Go back to the Design perspective.
- To “save” the nested workflow into its parent, do **File -> Save**
- Go to the parent using the **Windows** menu





Collapse the nested workflow

- Nested workflows can be **collapsed** to hide the implementation details
- In the **diagram** toolbar, click to disable **Expand Nested Workflows**
- Also click to **Display all service ports**





Nested workflow as a service

- The nested workflow now appears as a regular service in the master workflow
- Rename from *Workflow4* to *dbfetch*
- Right-click on *id* to set the **Constant Value P15409**
- Connect *fetches* to a new Workflow Output port.
- **File -> Save** the master workflow
- Run the master workflow



Further exercises

- Right-click on Workflow24 and Edit Nested Workflow
 - Replace db_value with another workflow input port
 - Use **File -> Save as...** to a file, **then File -> Close**
 - In the master workflow, **Expand nested workflow**. What happened?
 - Right click and use **Replace nested workflow**
- Create a new nested workflow
 - Use “Import from file” or “Already opened workflow” to import the InterproScan workflow from the Asynchronous Service tutorial
 - Check how your master workflow now contains a nested, nested workflow.
 - (Advanced): Do you see any advantage of this approach with regards to the Control Link?
- (Advanced): Try **Insert -> Merge workflow**
 - What do you see as advantages and problems with this approach?