

Taverna Tutorial exercise 2: REST services from BioCatalogue

Stian Soiland-Reyes and Christian Brenninkmeijer
University of Manchester
materials by Katy Wolstencroft and Aleksandra Pawlik

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

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

<http://orcid.org/0000-0002-1279-5133>

<http://orcid.org/0000-0001-8418-6735>



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

Bonn University, 2014-09-01

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



REST services from BioCatalogue

- Start with the *Get Protein FASTA* workflow from Exercise 1.
- Now we will find out what functional motifs the protein contains, but first we have to tell Taverna about some new services

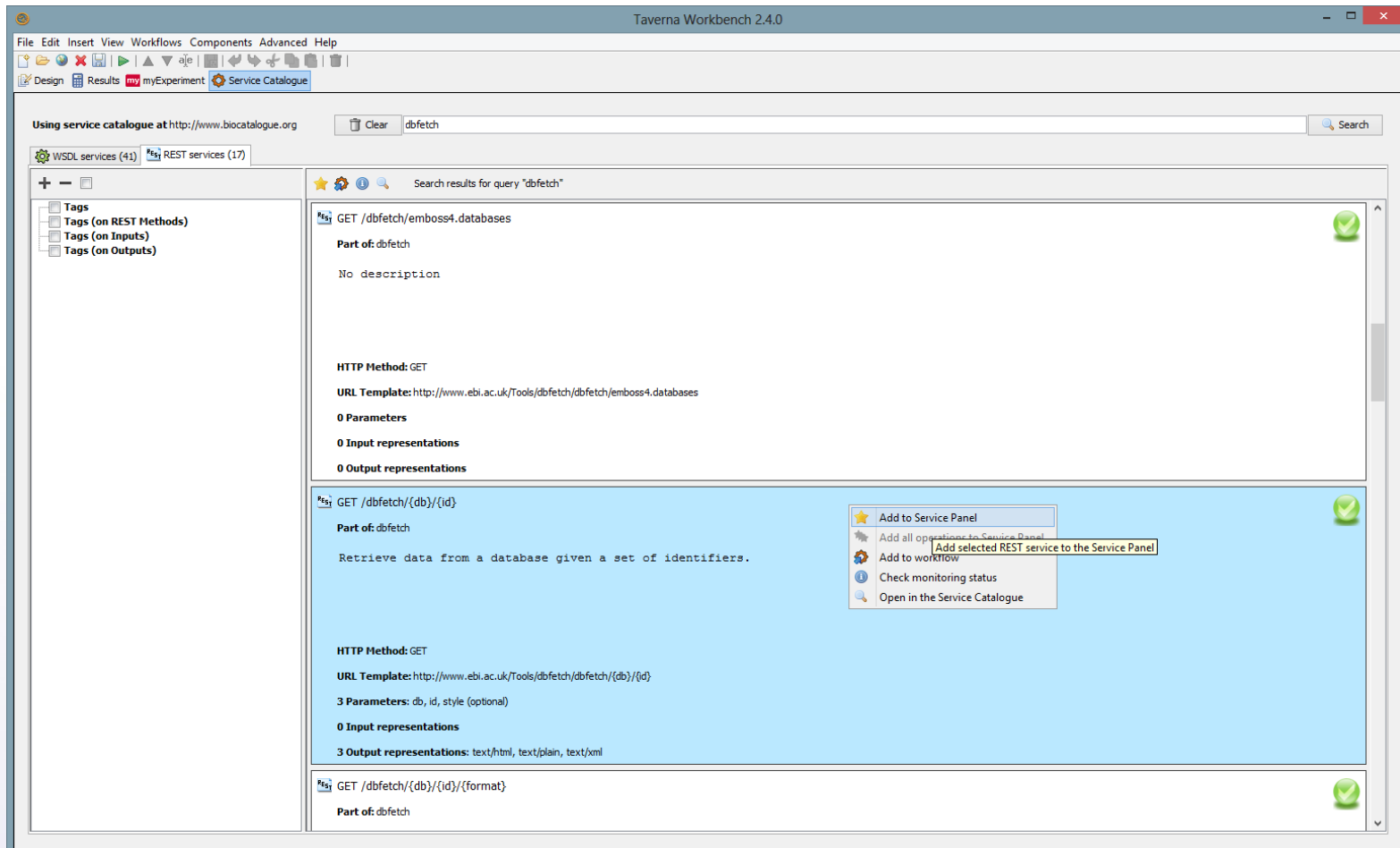


Service Catalogue tab

- Taverna can invoke any WSDL and REST web services
- The Service Catalogue <http://biocatalogue.org/> has registered over 1500 bioinformatics web services
- Go to the **Service Catalogue** perspective of Taverna
 - The Service Catalogue tab is included in the *Bioinformatics*, *Biodiversity* and *Enterprise* editions of Taverna Workbench.
 - For other editions, install Service Catalogue plugin from **Advanced -> Updates and plugins -> Find new plugins**
 - Note: *Biodiversity* edition is configured to use biodiversitycatalogue.org instead of biocatalogue.org and won't find dbfetch. See **File -> Preferences -> Service Catalogue**.
- Search for dbfetch

Searching for REST services

- In the **REST Service** tab, select *GET /dbfetch/{db}/{id}*
Retrieve data from a database given a set of identifiers
- **Right-click** on the service and **Add to Service Panel**



The screenshot shows the Taverna Workbench 2.4.0 interface. The main window displays a search for REST services at <http://www.biocatalogue.org> with the search term "dbfetch". The search results are displayed in a list view. The second service, "GET /dbfetch/{db}/{id}", is highlighted in blue. A context menu is open over this service, with the option "Add to Service Panel" selected. The menu also includes "Add all operations to Service Panel", "Add selected REST service to the Service Panel", "Check monitoring status", and "Open in the Service Catalogue".

Using service catalogue at <http://www.biocatalogue.org>

WSDL services (41) REST services (17)

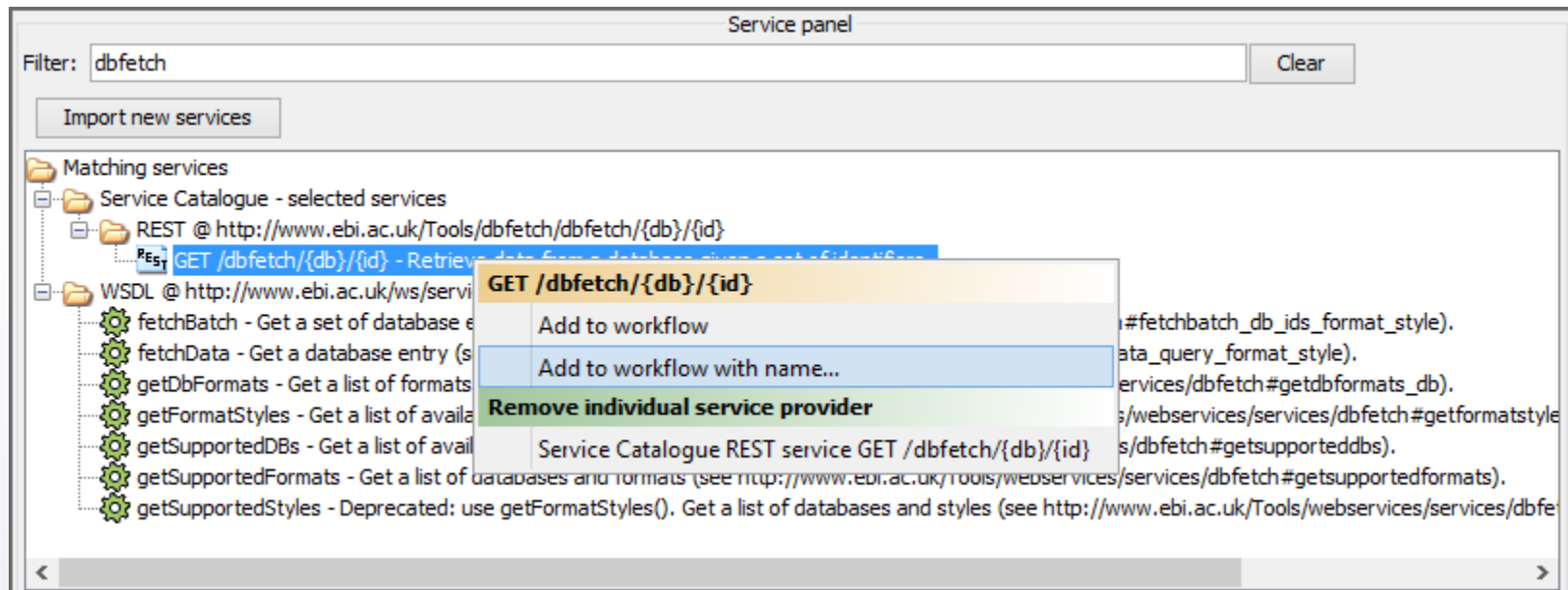
Search results for query "dbfetch"

- GET /dbfetch/emboss4.databases
Part of: dbfetch
No description
HTTP Method: GET
URL Template: <http://www.ebi.ac.uk/Tools/dbfetch/dbfetch/emboss4.databases>
0 Parameters
0 Input representations
0 Output representations
- GET /dbfetch/{db}/{id}**
Part of: dbfetch
Retrieve data from a database given a set of identifiers.
HTTP Method: GET
URL Template: <http://www.ebi.ac.uk/Tools/dbfetch/dbfetch/{db}/{id}>
3 Parameters: db, id, style (optional)
0 Input representations
3 Output representations: text/html, text/plain, text/xml
- GET /dbfetch/{db}/{id}/{format}



Adding service to workflow

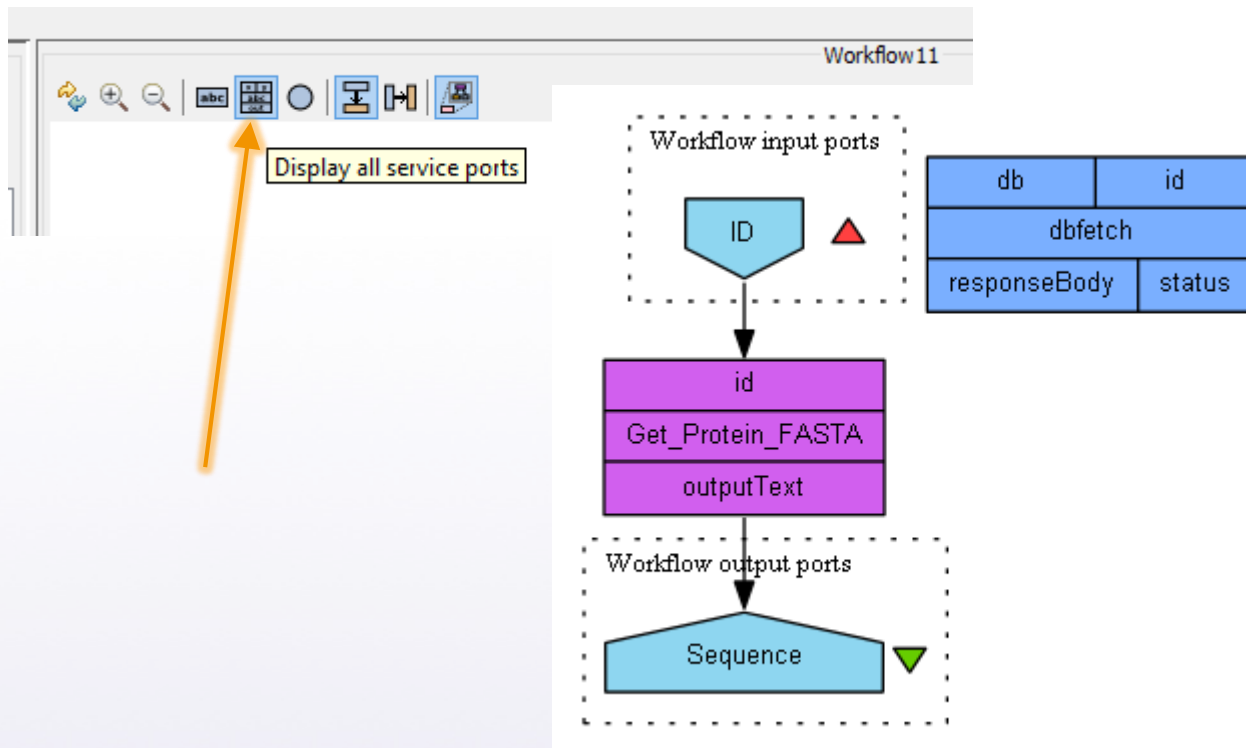
- Back in the **Design** perspective, in the **Available Services** search panel:
 - Search for dbfetch
 - Right-click on the *GET* service and choose “**Add to workflow with name...**”
 - Enter a name such as dbfetch and click **OK**





E2: REST service in workflow

- The workflow now has a new REST service *dbfetch*
- Click the **Display all service ports** button to see service inputs and outputs

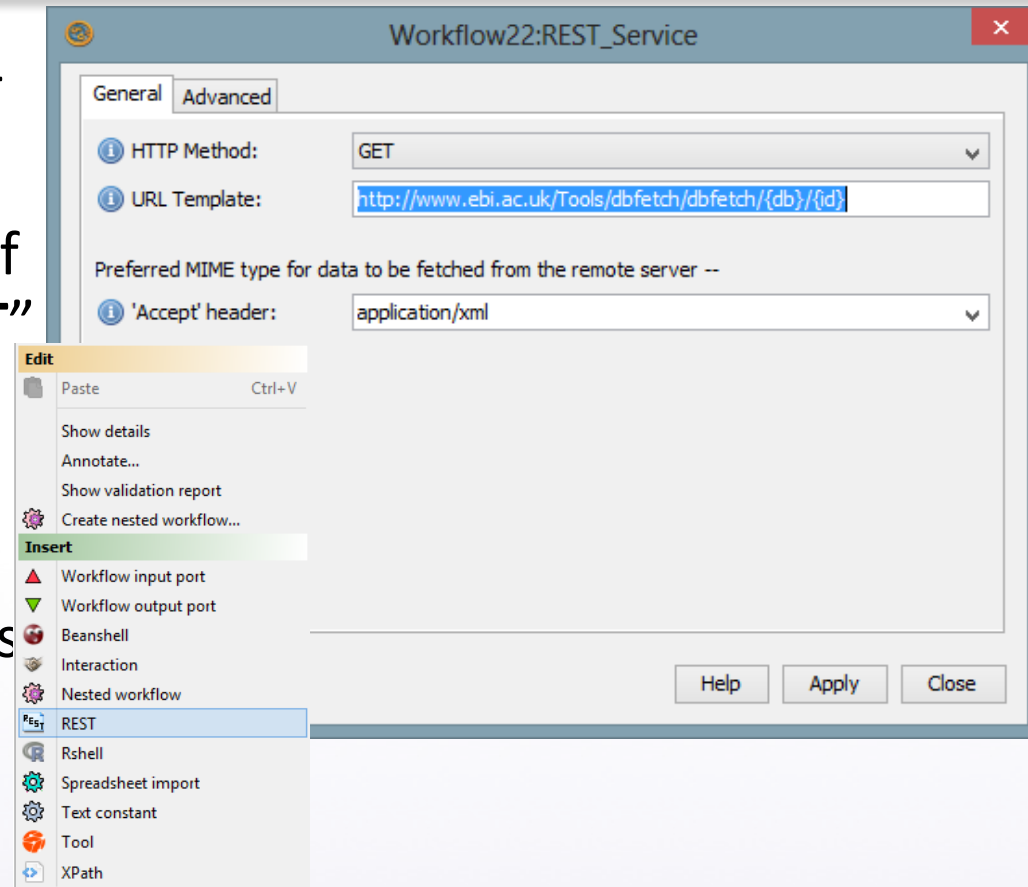




REST service using URI templates

Alternatively, you can add a REST service manually:

- Right-click on an empty area of the workflow and select “**REST**” from the “**Insert**” section
- Enter the **URL Template**, (see below), click **Close**
- A new REST service with inputs `db` and `id` is added



`http://www.ebi.ac.uk/Tools/dbfetch/dbfetch/{db}/{id}`



Details about REST service

- For this service, we need to supply the input ports *db* and a *protein id*.
- We are not sure what database names to use, so let's go back to **Service Catalogue** perspective
- Right-click on the `dbfetch/{db}/{id}` entry and **Open in the Service Catalogue**

GET /uniref400fetch
GET /uniref100fetch
GET /uniref50fetch
GET /uniref90fetch

Input Parameters:

URL Parameters:

db

Description(s):

Database to retrieve data from.

by **Hamish McWilliam** (almost .

[Login to](#)

Default Value:

No default value specified.

[Login to update this paramet](#)

Constrained Values:

This parameter is not constrained to any set of values.

[Login to update this parameter's cc](#)

Example Data:



embl

by **Hamish McWilliam** (almost .



uniprotkb

by **Hamish McWilliam** (almost .



uniref100

by **Hamish McWilliam** (almost .

- This service has extensive documentation in BioCatalogue
- Input parameter db is described with example values
- We will pick uniprotkb

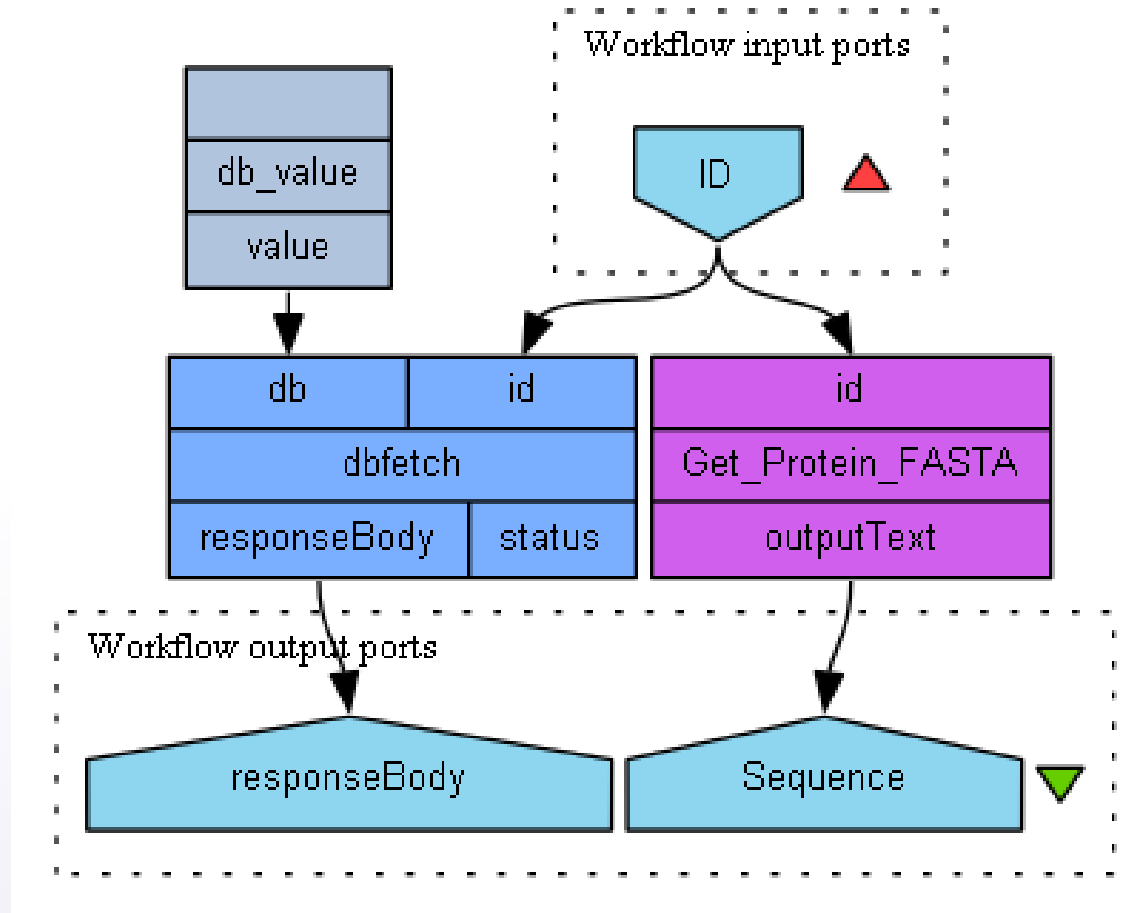


Connecting REST service

- Right-click on the REST service input `id` to **Connect with output from** the Workflow Input Port `id`
- Right-click on the `db` input port on the REST service and select **'Constant value'**.
- Add the constant value `'uniprotkb'` and click **"OK"**
 - A new *Text Constant* service is added and connected
- Add a Workflow Output Port and connect it to the REST Output Port `responseBody`



Finished workflow





Workflow results

- Save and run your workflow with the input value from exercise 1, P15409
- Now your Results will include the Uniprot entry for your protein on the **responseBody** output port.
 - Tip: Slide up the separator line above *Workflow Results* to see more of the values

Workflow results

Try also Value type *Text*

▲ ID ▼ Sequence ▼ responseBody

Click in tree to view values

Value type Seq Vista Refresh

OPSD_MOUSE

- General Inform
- Reference
- Reference
- Reference
- Reference
- Reference
- Reference
- Reference
- ID/identific
- AC/accessi
- DT/date
- DT/date
- DT/date
- DE/descript
- PP/feat

	OPSD_MOUSE			
1	MNGTEGPNFY	VFFSNVTGVV	RSPFEQPOYY	LAEPWQFSML AAYMFLLIVL
51	GFPINFLTLY	VIVQHKKLRT	PLNYILLNLA	VADLFMVFGG FTITLYTSLH
101	GYFVFGPTGC	NLEGFFATLG	GEIALWLSLV	LAIERYVVVC KPMSNFRFGE
151	NHAIMGVVFT	WIMALACAAP	PLVGWSRYIP	EGMQCSCGID YYTLKPEVNN
201	ESFVIYMFVV	HFTIPMIVIF	FCYQQLVFTV	KEAAAQQQES ATTQRAEKEV
251	TRMVIIMVIF	FLICWLPYAS	VAFYIFTHOG	SNFGPIFMTL PAFFAKSSSI
301	YNPVIYIMLN	KOFRNCMLTT	LCCGKNPLGD	DDASATASKT ETSOVAPA