erwin®

Data Transformation

erwin DT 9.1

User Guide

Contents

About this guide

This guide explains how to configure and use the tool erwin Data Transformation, describing the various features and functionality available to the user.

It is recommended that you print this guide so that you can follow the instructions more easily. The guide is designed to be printed double-sided in booklet form.

Document audience

The User Guide is intended to be read by end users who are concerned with getting the most benefit from the erwin Data Transformation implementation.

Document conventions

DT erwin Data Transformation

CM CW Suite Desktop Modeling Suite (aka Corporate Modeler Suite)

CE erwin CW Exchange (aka Corporate Exchange)

Evolve erwin EA Web Platform

WF Workflow

Chapter 1

1 Tool overview

DT is aimed at configuring multiple, dynamic data flows, and managing those flows without relying on programming resources, with an easy to use interface to organize them in logically consistent job sequences, to be run on schedule/on demand:

- Extract and transform raw data *from third party tools* in order to create and update data into erwin CW Exchange repository or into erwin EA Agile workspaces
- Extract data *from erwin CW Exchange repository or from erwin EA Agile workspaces* in order to update third party tools.
- Configure operation to *run on a schedule*: after the initial configuration and test, the data flows can run automatically with no need for the user to intervene.

From now on, "*model*" refers to data logically organized either into CW Exchange repository or into erwin EA Agile workspaces.

DT can therefore be classified as a tool that enables users to:

- *Maintain model data aligned with external sources*
    - Get raw data from external data source via adapters
    - Transform data using XLST Transformation library
    - Map data to the model objects and load into the model
- *Export and publish model data*
    - Unload data from the model, using custom filters
    - Transform data using XLST Transformation library
    - Upload external databases, post data to cloud systems or send files to external folder data to destination

- *Manage operations*
    - Back up data
    - Schedule the data flows to be executed
    - Review audit logs
    - Configure email notification alerts

DT offers a number of ready-to-use standard adapters that cover a wide number of integration needs. This enables the application of a standard approach to integration requirements.

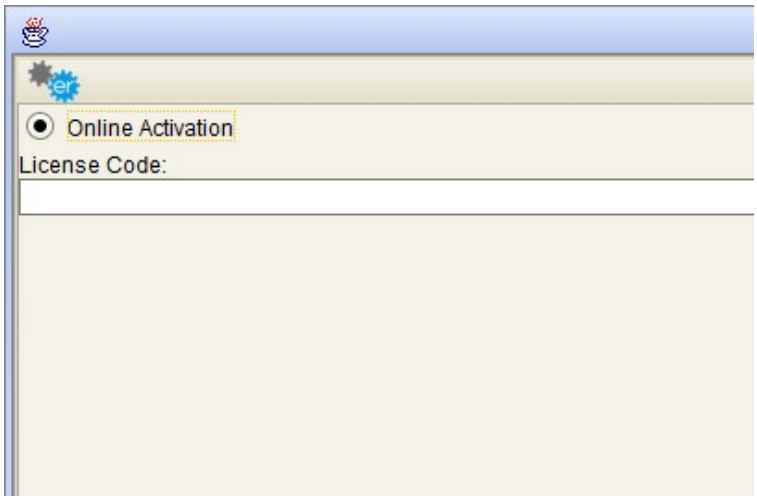
To ensure safe data operations, all updating activities are performed using standard APIs, and are preceded by data mapping and consistency checks. A user operation log, e-mail notification, scheduled operation log and system log are also provided (sever errors are notified via e-mail to system administrator).

DT stores all the necessary settings (environment parameters, access control and profiling, job scheduling, workflows, adapter parameters and rules for file handling) in an embedded database (users do not need to interact with this database directly, a tool administrator may do).

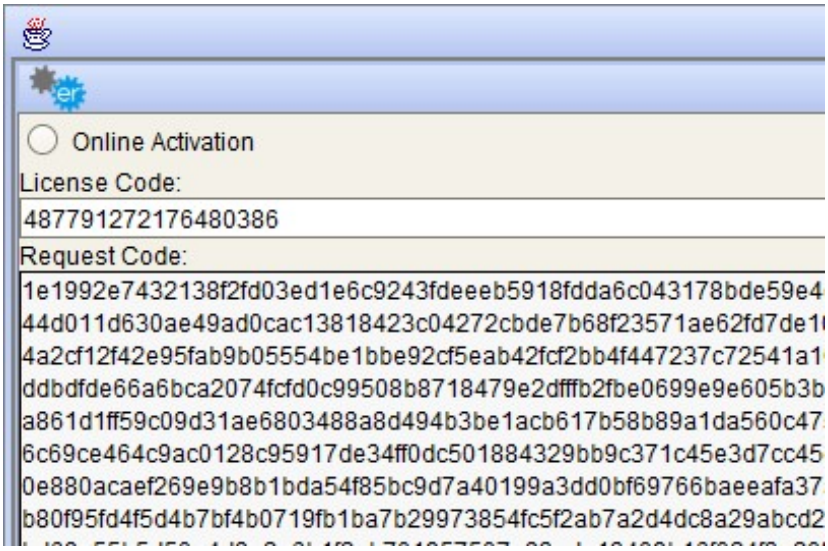
User may interact with the tool during workflow testing activities, checking data previews and if necessary, discarding the updates.

2 Licensing

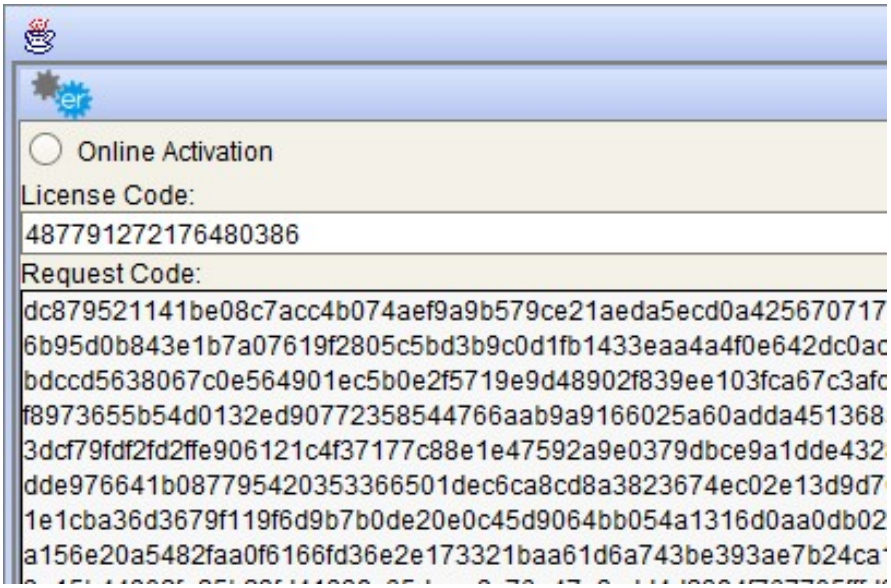
If no valid license is found on the system, the user is required to provide a license key when DT starts.



For offline licensing, a request code must be generated. This window should not be closed until the product activation is complete, as a new request would otherwise need to be generated.

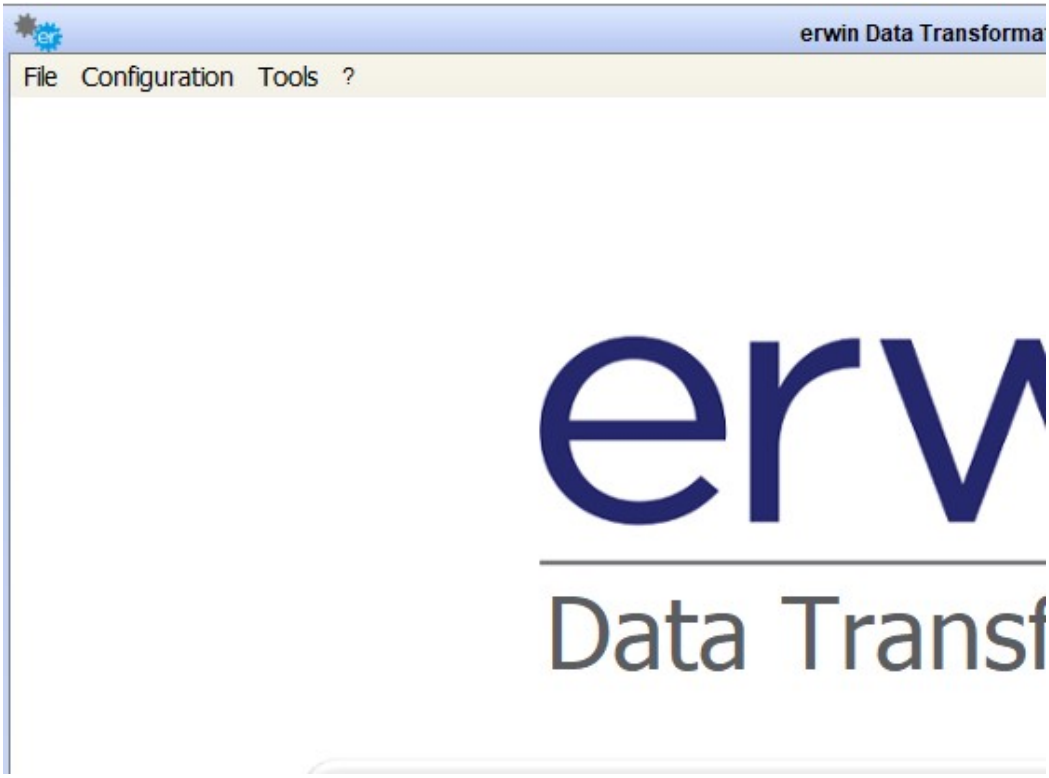


This code should be sent to erwin support, who will in turn provide an activation code specific to the request. This code should be used to activate the product. The activation code used must be specific to the request code provided to support; generating a new request code would necessitate a new activation code.



3 Home page

The home page allows the user to select the operation by clicking on the buttons or selecting from the menu at the top of the window.



Operations are logically divided into 2 groups:

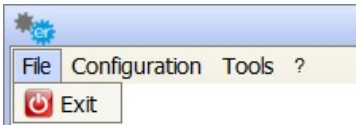
- *Setting adapters*:

- *Define model (or workspace) configuration*, *workflow* (data flows) *configuration* and *schedule* of the *jobs/job sequences*.

The same operations are available from the menu bar, as explained in following paragraphs.

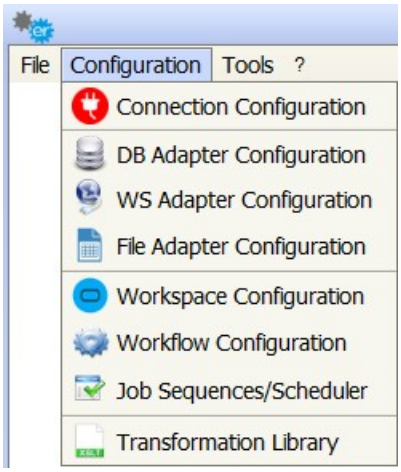
1.1.1. File Menu

This menu is used to close the application.



1.1.2. Configuration Menu

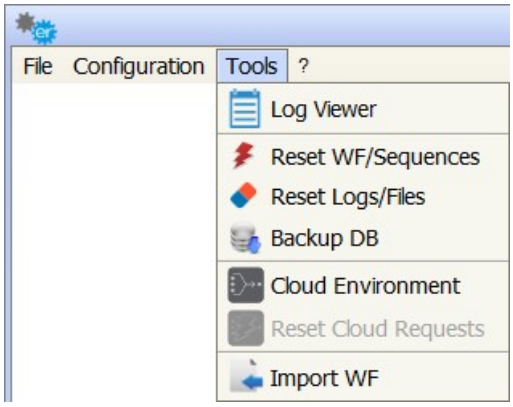
This menu provides access to add/delete/modify the settings for the following operations:



- Connection Configuration
- Adapter Configuration (DB, WebService or File Adapter)
- Model/Workspace Configuration
- Workflow Configuration
- Job Sequences/Scheduler
- Transformation Library

1.1.3. Tools Menu

This menu provides access to the application log viewer. Additionally the user can reset sequences that are running, reset logs, delete workflow files, backup CC database and import workflow from the CC workflow export file – (see Chapter 9 - Utilities)

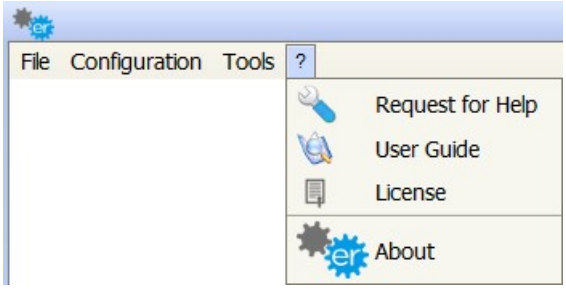


"Cloud Environment" and "Reset Cloud Requests" items refer to external environment configuration and operations – See "Chapter 7 - Running DT from Cloud Platform"

1.1.4. Help Menu

This menu provides information and tools:

- Request for Help on a given workflow – (see Chapter 9 - Utilities)
- DT User Guide (PDF)
- License
- About, DT version, build and type of license



Chapter 2

1 Adapters configuration

Adapters are tools used by DT to “GET” data from the external world (databases, web services, folders and files) or to “PUT”, or send data to the destination. “GET” adapters can be used as a data source when configuring a workflow; “PUT” adapters can be used as a workflow operation target (see later). Any time a “GET” adapter is created, a job is also created, in order to be scheduled and produce its dataset (triggering all the workflows which use that dataset as source). A “PUT” adapter cannot be scheduled, as its execution is triggered by the execution of workflows using it as target.

1.1 DB Adapter Configuration

Going through this configuration window, the user can set up an adapter to be used as data source or operation target in a workflow. It’s possible to:

- Execute a **query** on a specified relational database to use the result set as data source in a workflow
- Execute a **stored procedure** with the proper parameters to use the output file as a data source in a workflow
- Configure a **DB loader** to be used as the operation target in a workflow
- Execute a **query** on a MongoDB database to use the result set as data source in a workflow

The first group of fields in the window is the DB Adapter List, which lists all the DB Adapters available. The second group is the General Parameters, which include the Name, Description, Type and DB Adapter folder; all these fields are mandatory.

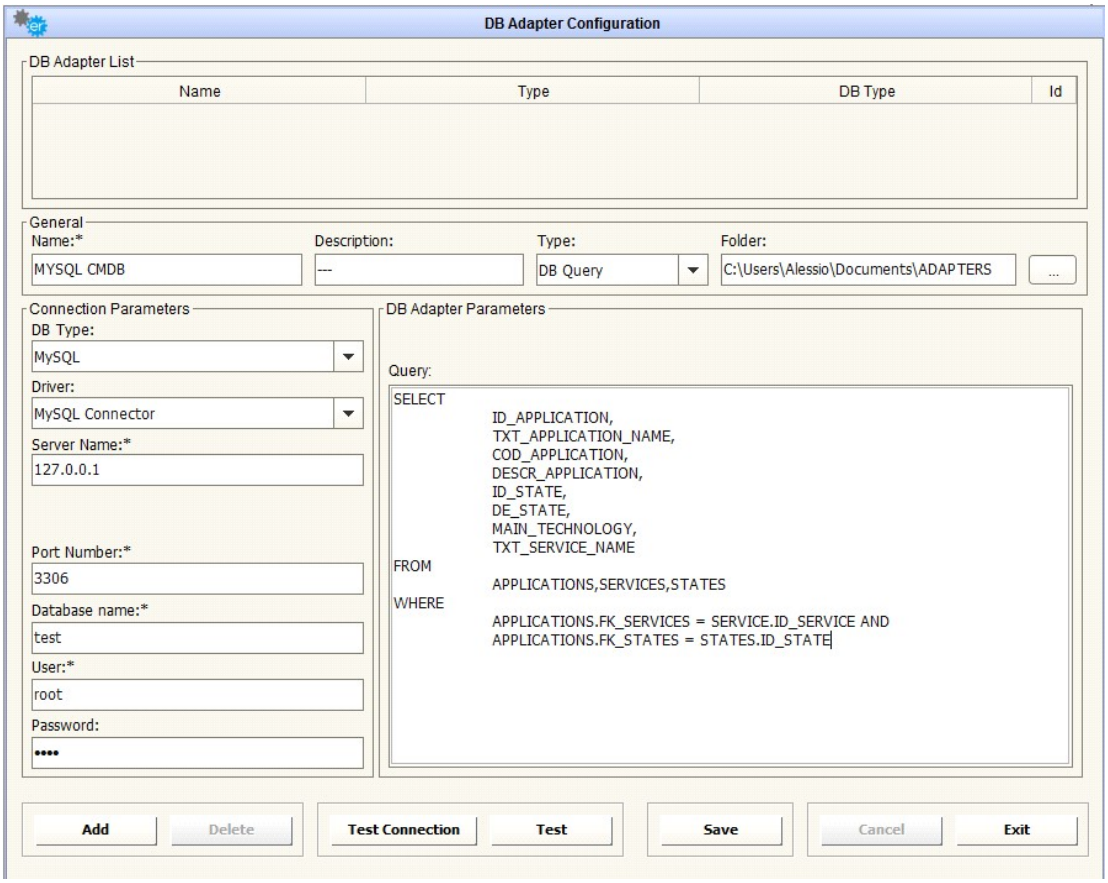
The third group is the Connection Parameters, which includes all the parameters needed to connect to the Data Base and finally the last group is the DB Adapter Parameters, which describes the query or the Stored Procedure and its parameters.

1.1.1 Add or Modify DB Adapter

To create a new DB Adapter just press the Add button and start editing the new adapter parameters.

Alternatively modify an existing adapter, simply selecting from the list – the parameters are always in edit mode.

The name, description, type and folder (used to save intermediate and result files for adapter execution) of the DB adapter are the first fields to enter in the form; then for any type of DB Adapter the user must provide connection parameters and adapter details as described below.



1.1.2 Connection Parameters

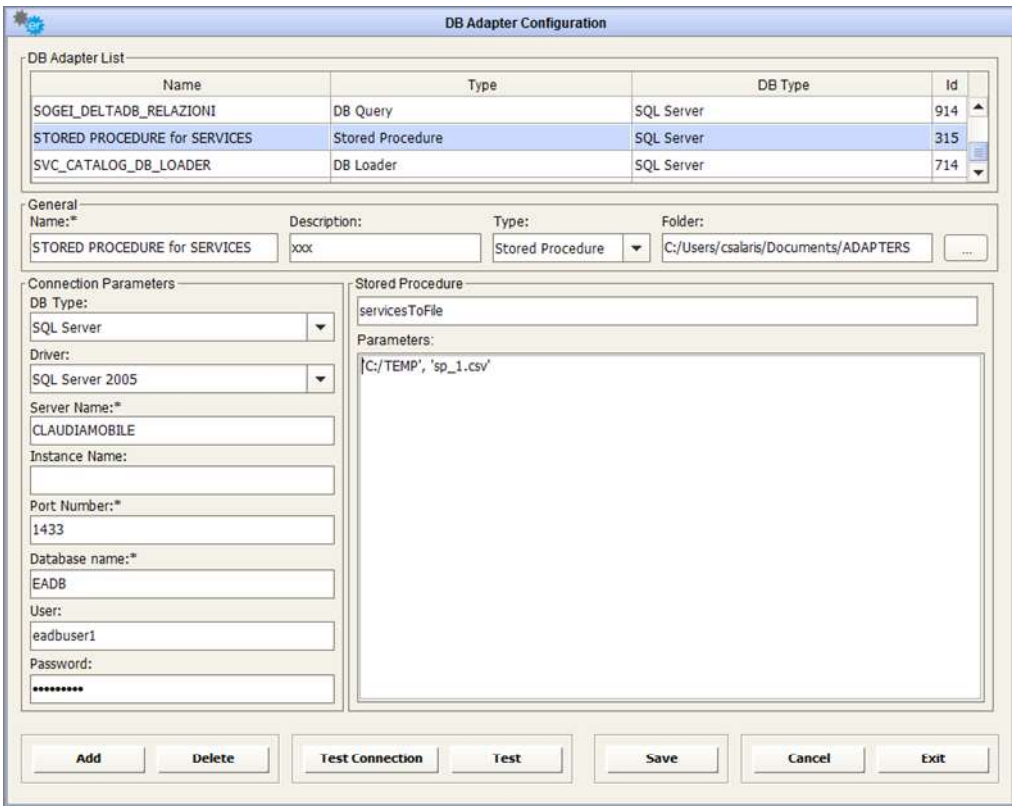
In the Connection Parameters pane the user has to provide all the parameters needed to set up the connection with the database. In order to check if a connection can run correctly, press the Test Connection button.

Then, depending on what has been selected in the drop-down list named Type (Stored Procedure, DB Query, DB Loader and NoSQL DB Query) the DB Adapter Parameters section shows the following.

1.1.3 Stored Procedure

When the field “Type” is set to “Stored Procedure”, then two other fields are highlighted:

- The field “Stored Procedure name” that has to be filled with the stored procedure name
- The field “Parameters” that has to be filled with the list of all the parameters separated by commas and optionally enclosed by quote (e.g. the same way the stored procedure could be called from outside DT, like a DB Client).

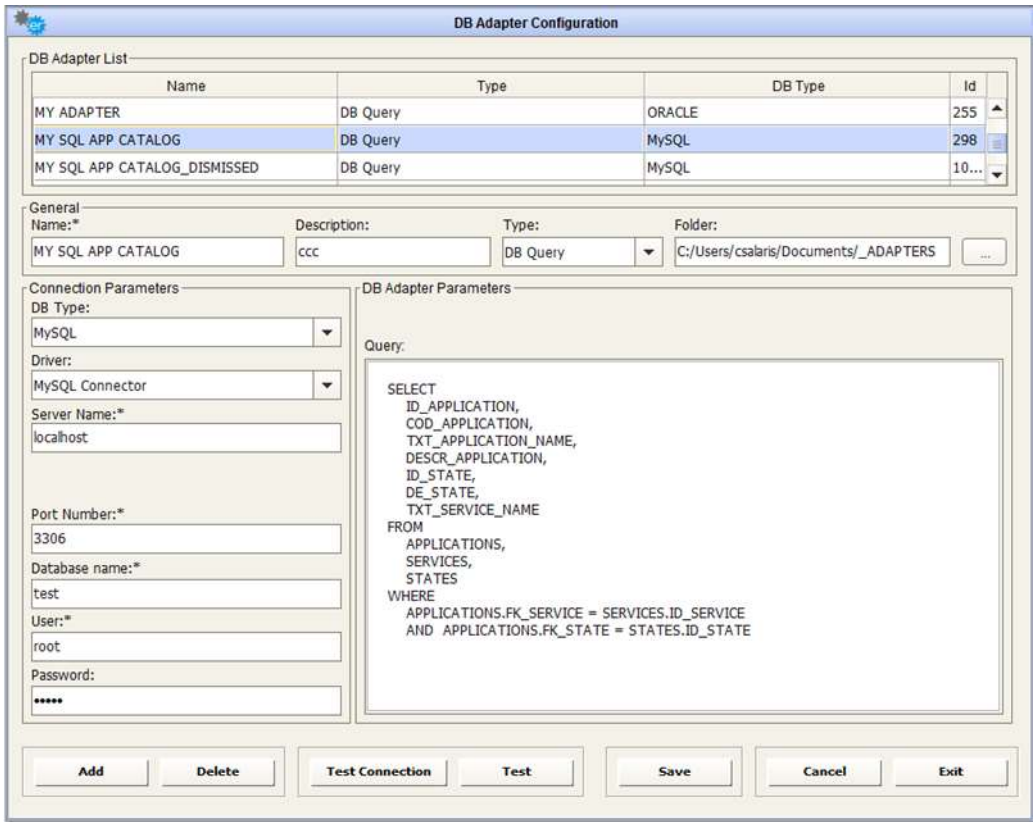


To be used as data source adapter in DT, stored procedures must produce a CSV file containing a data set; this means that at least two parameters are required (folder as the first paramater and file name as second), so that DT knows and accesses the resulting dataset.

For example, if an external database contains a stored procedure named “servicesToFile”, which accepts two parameters “folder” and “filename”, then the parameters text area must be filled with the folder path enclosed by quotes.

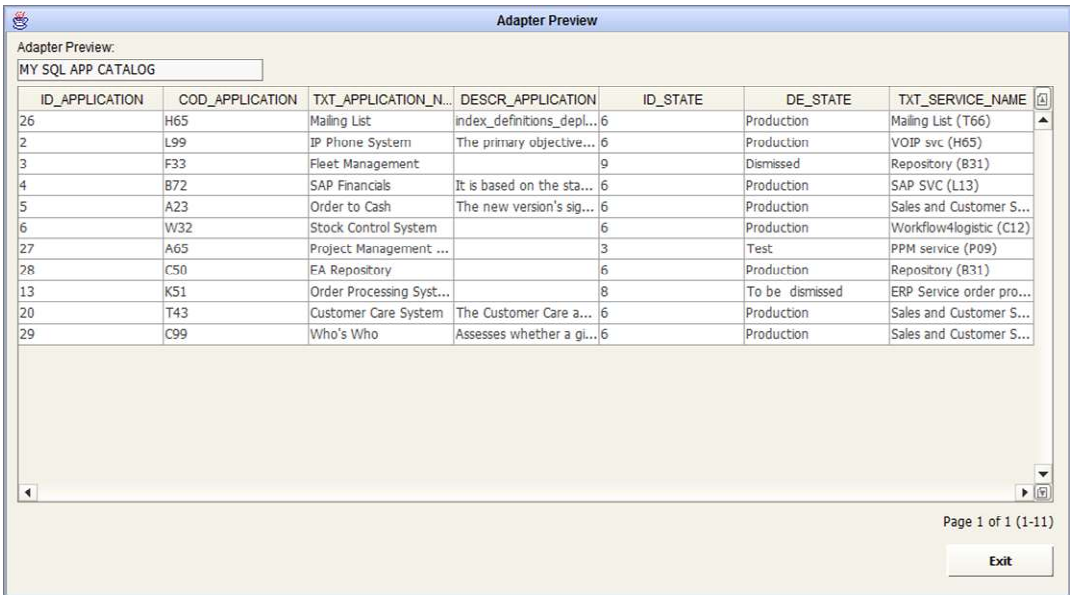
1.1.4 DB Query

When the field “Type” is set to “DB Query”, then the “Query” field is highlighted. In this field, the user has to write the query to be run by the Database.



In order to check if the previous DB query is written correctly, press the Test button. This opens a new window that contains the result set obtained from running the adapter.

The result set is paginated in groups of 50 records. Use the side window arrows to go up and down the list.



1.1.5 DB Loader

When the field “Type” is set to “DB Loader”, then the entire DB Adapter Parameters section is disabled. All the mapping parameters between source and target data structure, needed to upload a table in the externals database, will be provided in the workflow operation configuration (see later).

For every type, to save the new DB Adapter just created, press the Save button. The adapter will be saved and added to the DB Adapter List at the top of the window. Only if the Test Connection (and query execution, for that type) have valid results, the DB Adapter will be saved.

To cancel the adapter data just entered and to reset the window, press the Cancel button.

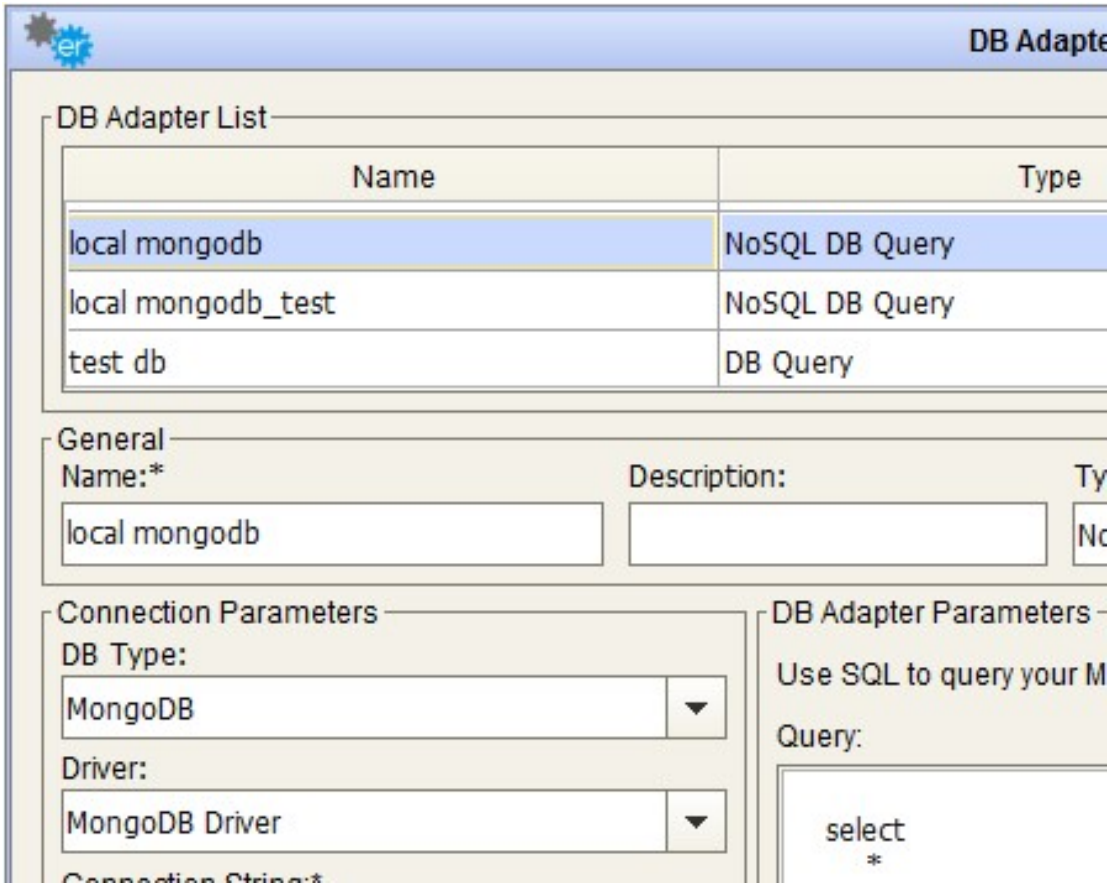
To delete an existing adapter, just select it and press the Delete button. It will be permanently removed from the DB Adapter List. Please note, all the items linked to the deleted DB adapter (job and workflows that use it as source) are also removed.

The Exit button closes the window.

1.1.6 NoSQL DB Query

When the field “Type” is set to “NoSQL DB Query”, then the “Query” field is highlighted. In this field, the user has to write the query to be run by the MongoDB Database and MongoDB item is selected in the “DB Type” dropdown list. The query must be written using SQL syntax; examples of supported SQL functions are:

- *select object.key1, object2.key3, object1.key4 from my_collection where object.key2 = 34 AND object2.key4 > 5*
- *select * from my_table where date(column,'YYY-MM-DD') >= '2016-12-12'*
- *select * from my_table where date(column,'natural') >= '5000 days ago'*
- *select * from my_table where regexMatch(column,'^[ae"gaf]+$') = true*
- *select distinct column1 from my_table where value IS NULL*
- *select * from my_table where value LIKE 'start%'*
- *select column1 from my_table where value IN ("theValue1","theValue2","theValue3")*
- *select column1 from my_table where value NOT IN ("theValue1","theValue2","theValue3")*
- *select column1 from my_table where column = true*
- *select borough, cuisine, count(*) from my_collection WHERE borough LIKE 'Queens%' GROUP BY borough, cuisine ORDER BY count(*) DESC;*
- *delete from my_table where value IN ("theValue1","theValue2","theValue3")*



In the Connection Parameters, user is asked for: Connection String, Database name and Password. For local connection, Connection string is usually *mongodb://127.0.0.1:27017*, while for cloud connections, Connection String can be automatically retrieved from the Connect button inside MongoDB web console. Connection String samples are:

mongodb://myusername:<PASSWORD>@cluster0-shard-00-00-pncrc.mongodb.net:27017,cluster0-shard-00-01-pncrc.mongodb.net:27017,cluster0-shard-00-02-pncrc.mongodb.net:27017/admin?replicaSet=Cluster0-shard-0&ssl=true

and

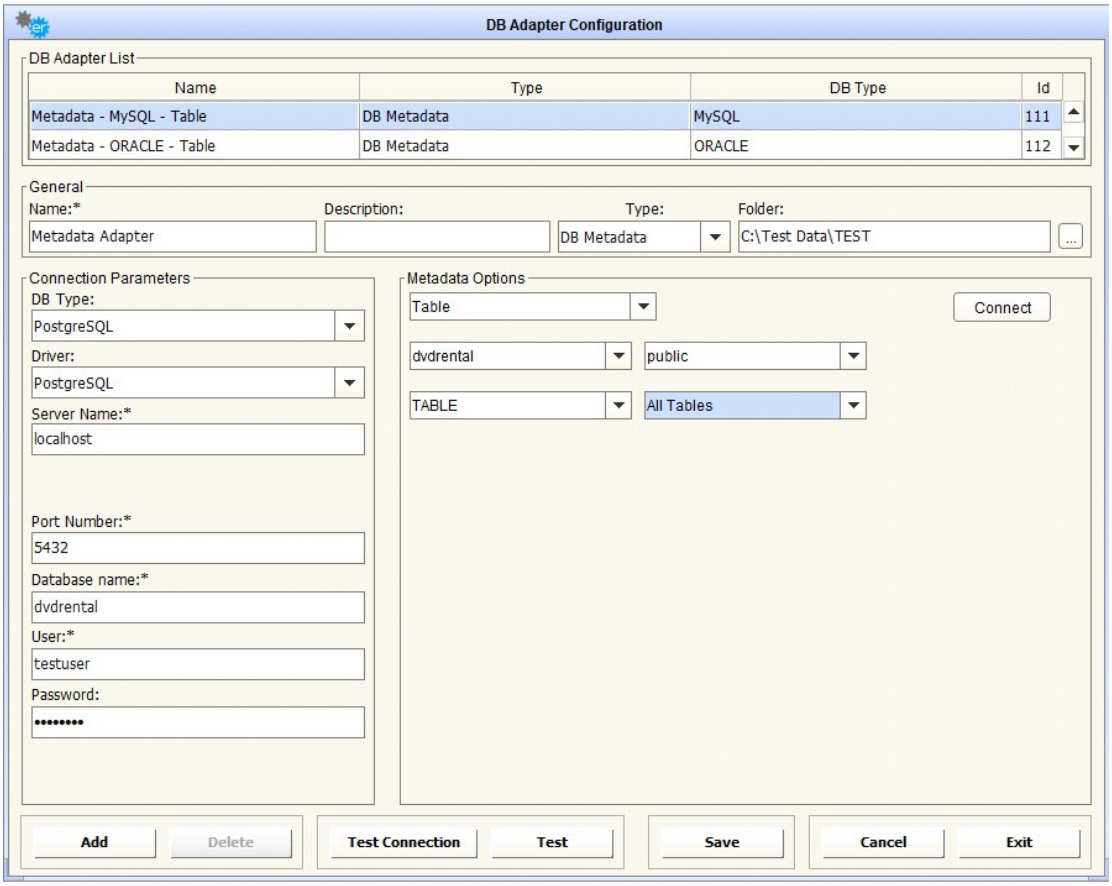
mongodb+srv://myusername:<PASSWORD>@cluster0-pncrc.mongodb.net/test

The user can replace <PASSWORD> wildcard with the real one or can keep <PASSWORD> wildcard in the Connection string and provide the encrypted password in the “Password” field.

1.1.7 DB Metadata

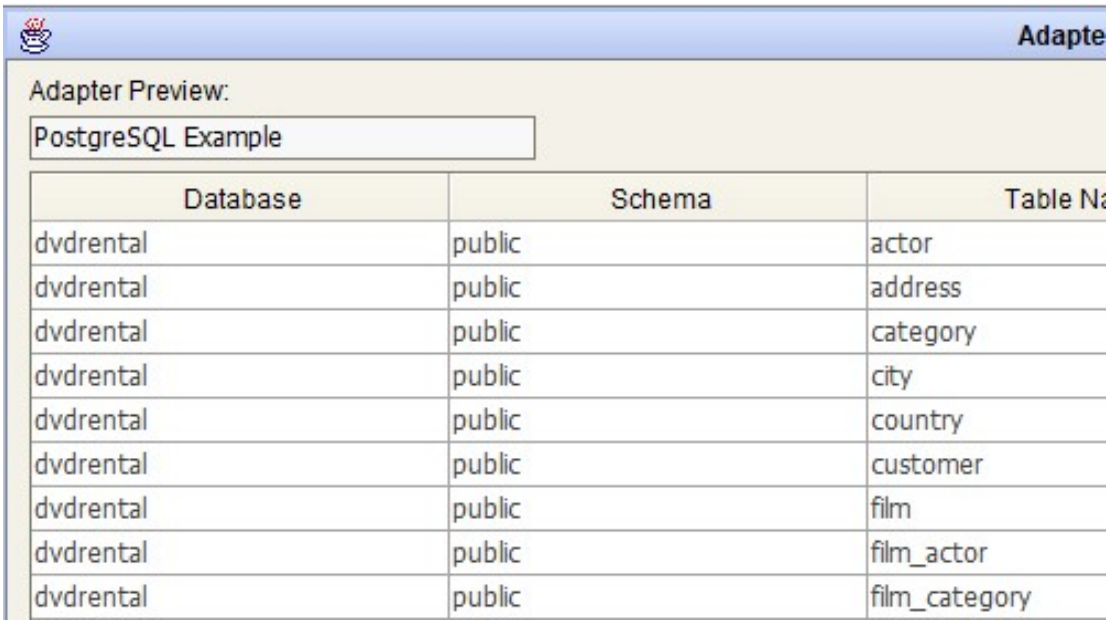
Setting the “Type” field to “DB Metadata” allows metadata to be retrieved from an SQL databases.

If the “Connection Parameters” are valid, clicking the Connect button will result in options being displayed for filtering the results by catalog name, schema, table name, and/or table type, as appropriate to the database type and requested metadata type.



If the a connection cannot be established, clicking the Connect button will result in a “Connection Refused” message being displayed.

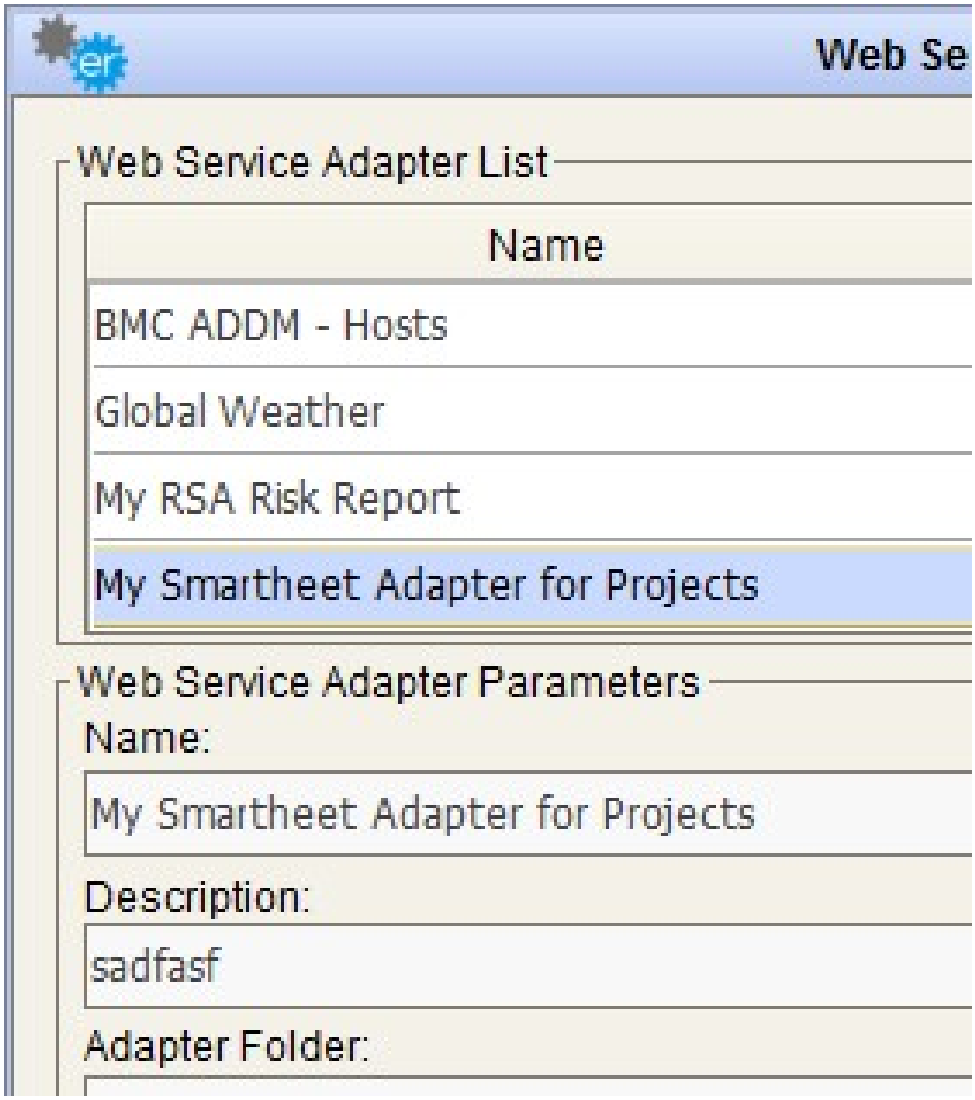
The first 50 results can be previewed by pressing the Test button.



Database	Schema	Table Na
dvdrental	public	actor
dvdrental	public	address
dvdrental	public	category
dvdrental	public	city
dvdrental	public	country
dvdrental	public	customer
dvdrental	public	film
dvdrental	public	film_actor
dvdrental	public	film_category

1.2 Web Service Adapter Configuration

Going through this configuration window, the user can set up an adapter to execute a connection with a known Web Service.



The first group of fields in the window relate to the Web Services Adapter List, which lists all the available adapters. For each adapter selected in the list, the Web Services Adapter Parameters fill the second group of fields, which describe the Web Service connection parameters and the third group of fields show the expected input parameters for the execution of the adapter.

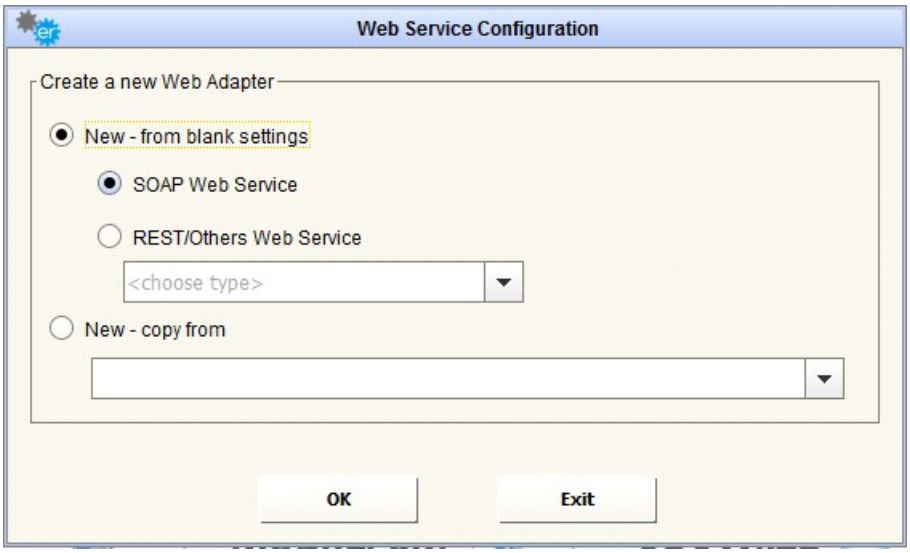
When adding a new one, it's possible to choose between:

- **SOAP** Web Service

- **REST**/others specific adapters, from supported third party systems

1.2.1 Add or Modify SOAP Web Service Adapter

To create a new adapter just press the Add, and select “SOAP Web Service” in the following popup:

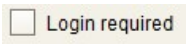


To modify an existing adapter selected from the list, press the Modify. An editing window pops up.

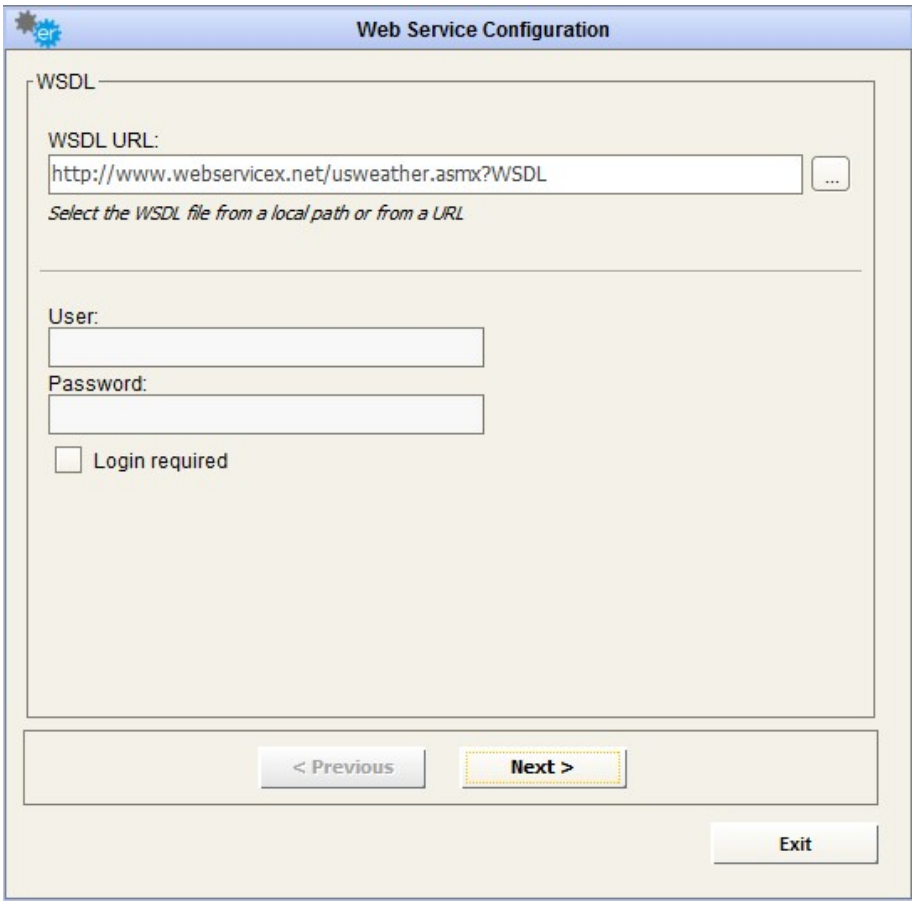
The same pop up window applies in both cases. The fields will be empty if the user is creating a new adapter. The same fields will contain data, if the user is going to modify an already existing adapter.

The first group of fields concern the WDSL Parameters while the second group allows the user to specify login parameters, if required.

The first field contains the URL of the WSDL file that should apply for the Web Service. The side button ... of the first field allows navigating into the file system and selecting the WSDL file from a local path.

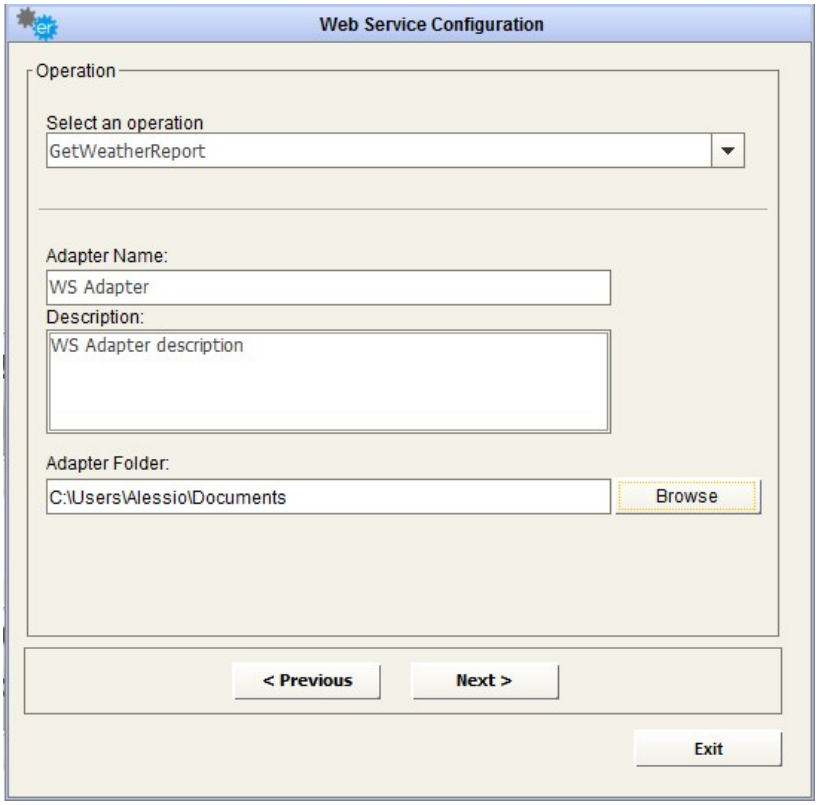
Should the Web Service require a login (username and password) to the WSDL server, a flag in the checkbox 

will enable the related fields allowing entry of the credentials. Please note that this authentication only applies to the WSDL server and not to the web service operation – operation authentication is not supported at the moment.



The 

button steps into the next window that shows all the available operations of that Web Service.



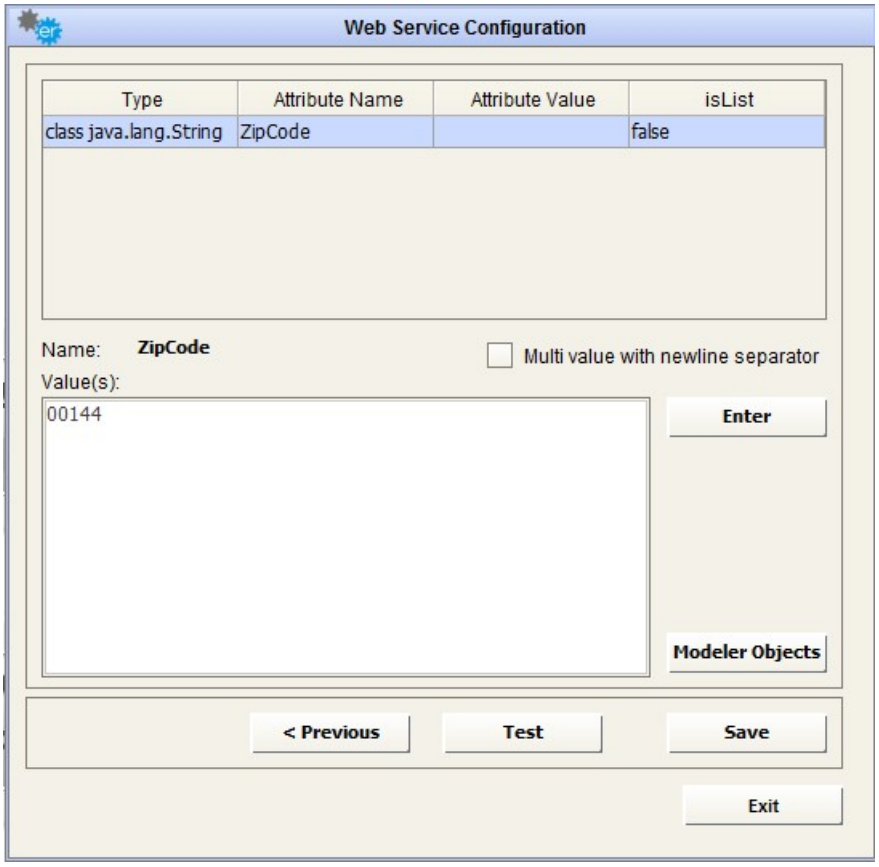
The first field of the window is a drop-down list that contains a list of the operations available for the selected Web Service.

After selecting the desired operation it is required to enter a name and a description for the adapter in the appropriate fields;the user must browse the file system for the adapter folder (used to save intermediate and result file for adapter execution).

The next button 

steps into the next window that allows the user to choose the values for the Web Service parameters.

If the Web Services has no parameters the user can leave the fields empty in the following wizard window.



The first field shows the list of the Web Service parameters, as read from the WSDL:

- Type of parameter (Type),
- Attribute Name (Class Name),
- Attribute Value (Value),
- Capability to accept list of values, as defined in WSDL (isList)

For each of the shown parameters, a value can be entered into the multiline text field, while the parameter is selected in the list above.

The Enter button sets the entered data.

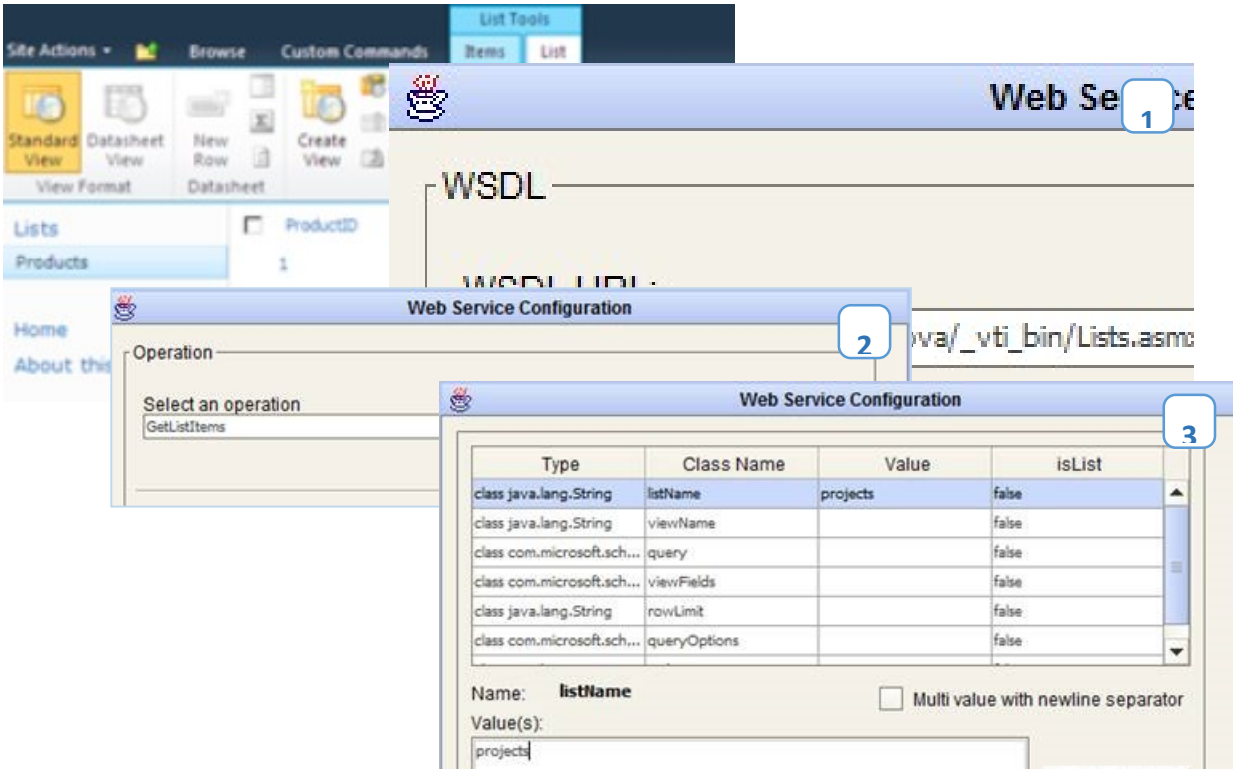
If a parameter is actually a list of values (column “isList” is true), then the user is required to optionally enter different values, separated by a new line (one per row). In this case, DT will call the Web Service only once, passing to it the multi-line string, containing the different values.

If a parameter natively does not allow a list of values (column “isList” is false), but the user needs to pass more than one value to the Web Service call, selecting the option Multi value with newline separator, the user is allowed to enter a list of values for the selected parameter, even if “isList” is false, with the new line as a separator.

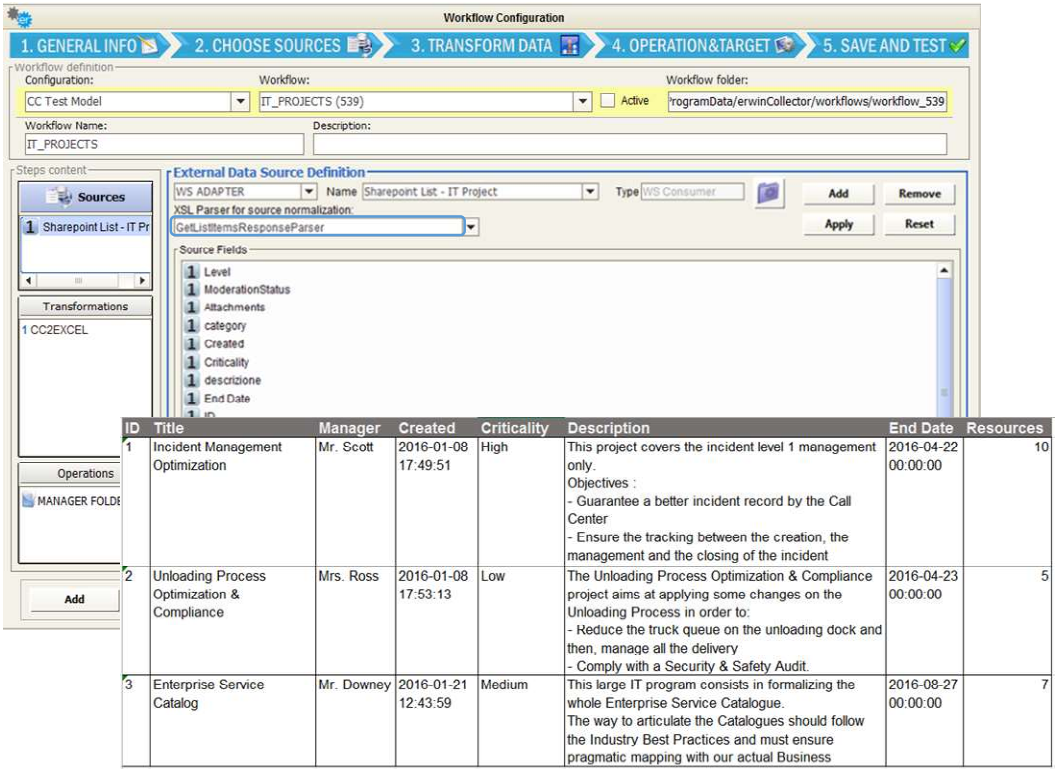
DT will then perform all the implied calls (as many times as the number of desired values in the list) to the Web Service at the scheduled time, by scheduling just one job.

Pressing the Save button will save all the changes made in the window.

One common application may be extracting information from ***MS SharePoint Lists, for on premise deployments***. In this case, the user could configure a web service adapter to read GetListItems operation from the Lists web service, specifying as a parameter the name of the list to get data from.



After configuring the adapter, users can use it in any workflow as a source, applying for source normalization the provided parser *GetListItemResponseParser* (which may be customized in terms of extracted properties, if needed), so that the list content can be used in an end to end operation.



1.2.2 Test Web Service Parameters

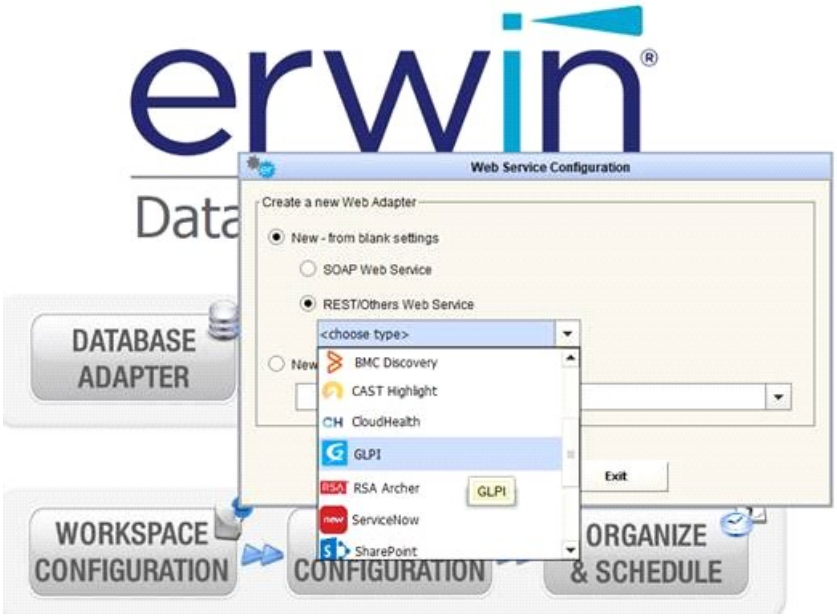
Before saving the Web Service adapter, the user can test it. Pressing the **Test** button will open a new pop-up window and the XML SOAP response resulting from the Web Service call is listed.

The Exit button closes the data preview window and goes back to Web Service configuration form.

If you have not saved before testing the Web Service Configuration, remember to press the Save button, before leaving the parent window and all the entered parameters will be stored into the DT databases.

1.2.3 Add or Modify REST/others Web Service Adapter

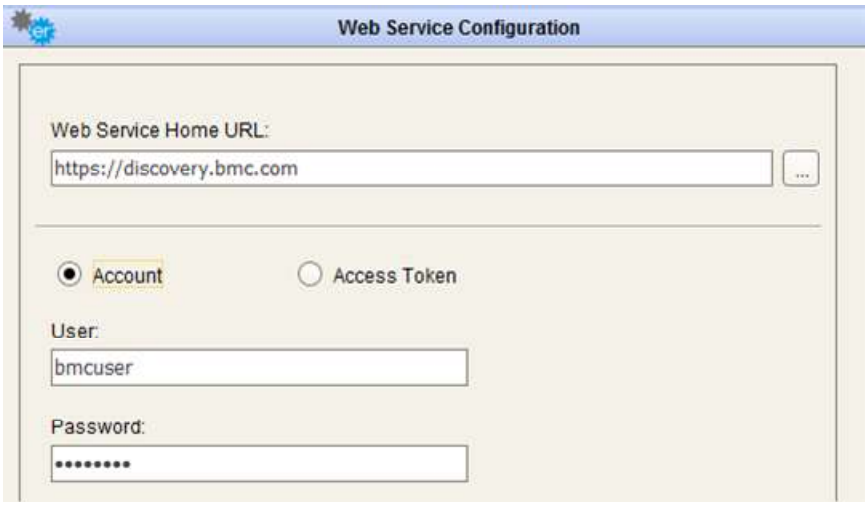
To create a new adapter just press the Add, and select “REST/Others Web Service” in the following popup:

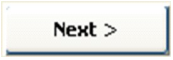


To modify an existing adapter selected from the list, press the Modify. An editing window pops up.

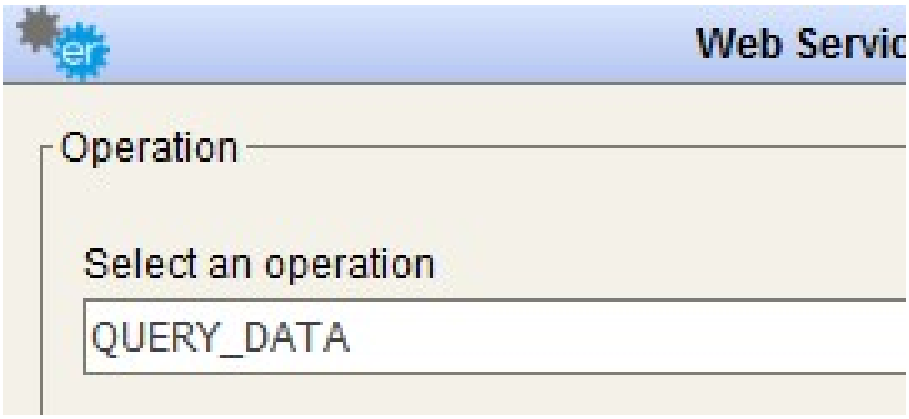
1.2.4 BMC Discovery

To use the adapter for **BMC Discovery**, fill the field “Web Service Home URL” with the SmartSheet home page URL you want to connect to. Then provide the **username** and **password** for a valid BMC Account or an **access token** to connect. To generate an access token, please refer to BMC Discovery System Administrator



The  button steps into the next window that shows the available operation of that Web Service, which is “**QUERY DATA**”, allowing users to query all data stored in the repository

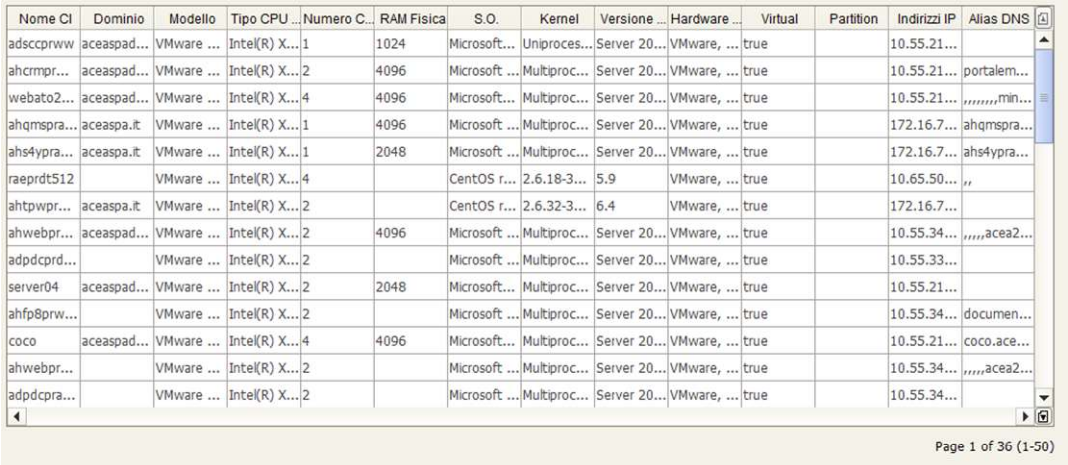
.



After selecting the desired operation, it is required to enter a name and a description for the adapter in the appropriate fields; the user must browse the file system for the adapter folder (used to save intermediate and result file for adapter execution).

The next button Next steps into the next window that allows the user to set values for the Web Service parameters.

Pressing the Test button will open a new pop-up window. The response resulting from the Web Service call is displayed.



Nome CI	Dominio	Modello	Tipo CPU ...	Numero C...	RAM Fisica	S.O.	Kernel	Versione ...	Hardware ...	Virtual	Partition	Indirizzi IP	Alias DNS
adsccprww	aceaspad...	VMware ...	Intel(R) X...	1	1024	Microsoft...	Uniproces...	Server 20...	VMware, ...	true		10.55.21...	
ahcrmpr...	aceaspad...	VMware ...	Intel(R) X...	2	4096	Microsoft ...	Multiproc...	Server 20...	VMware, ...	true		10.55.21...	portalem...
webato2...	aceaspad...	VMware ...	Intel(R) X...	4	4096	Microsoft...	Multiproc...	Server 20...	VMware, ...	true		10.55.21...	,,,,,,,,min...
ahqmspra...	aceaspa.it	VMware ...	Intel(R) X...	1	4096	Microsoft ...	Multiproc...	Server 20...	VMware, ...	true		172.16.7...	ahqmspra...
ahs4ypra...	aceaspa.it	VMware ...	Intel(R) X...	1	2048	Microsoft ...	Multiproc...	Server 20...	VMware, ...	true		172.16.7...	ahs4ypra...
raeprdt512		VMware ...	Intel(R) X...	4		CentOS r...	2.6.18-3...	5.9	VMware, ...	true		10.65.50...	,,
ahtpwpr...	aceaspa.it	VMware ...	Intel(R) X...	2		CentOS r...	2.6.32-3...	6.4	VMware, ...	true		172.16.7...	
ahwebpr...	aceaspad...	VMware ...	Intel(R) X...	2	4096	Microsoft ...	Multiproc...	Server 20...	VMware, ...	true		10.55.34...	,,,,,acea2...
adpdcprd...		VMware ...	Intel(R) X...	2		Microsoft ...	Multiproc...	Server 20...	VMware, ...	true		10.55.33...	
server04	aceaspad...	VMware ...	Intel(R) X...	2	2048	Microsoft...	Multiproc...	Server 20...	VMware, ...	true		10.55.21...	
ahfp8prw...		VMware ...	Intel(R) X...	2		Microsoft ...	Multiproc...	Server 20...	VMware, ...	true		10.55.34...	documen...
coco	aceaspad...	VMware ...	Intel(R) X...	4	4096	Microsoft...	Multiproc...	Server 20...	VMware, ...	true		10.55.21...	coco.ace...
ahwebpr...		VMware ...	Intel(R) X...	2		Microsoft ...	Multiproc...	Server 20...	VMware, ...	true		10.55.34...	,,,,,acea2...
adpdcpra...		VMware ...	Intel(R) X...	2		Microsoft ...	Multiproc...	Server 20...	VMware, ...	true		10.55.34...	

Page 1 of 36 (1-50)

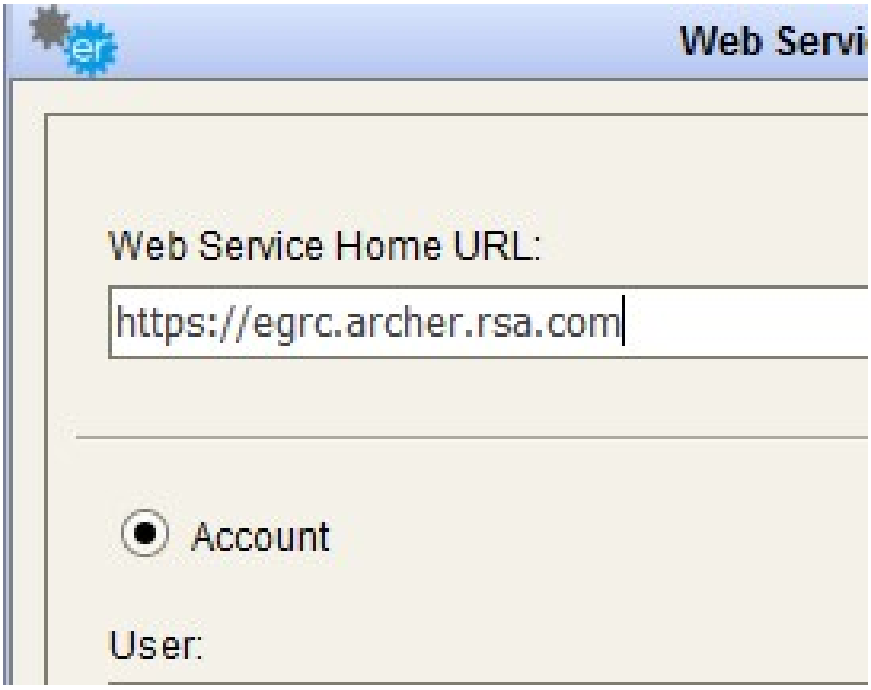
Pressing the SAVE button will store all the entered parameters for the web service adapter in the DT database.

The Exit button closes the data preview window and goes back to Web Service configuration form.

1.2.5 RSA Archer – VIEW REPORT

To use the adapter for **RSA Archer, to read report contents**, fill the field “Web Service Home URL” with the RSA URL you want to connect to.

Then, provide a valid instance URL and username/password to connect.



The Next > button steps into the next window that shows the available operations of this Web Service; choose "**VIEW REPORT**", to query data as per RSA report definition.

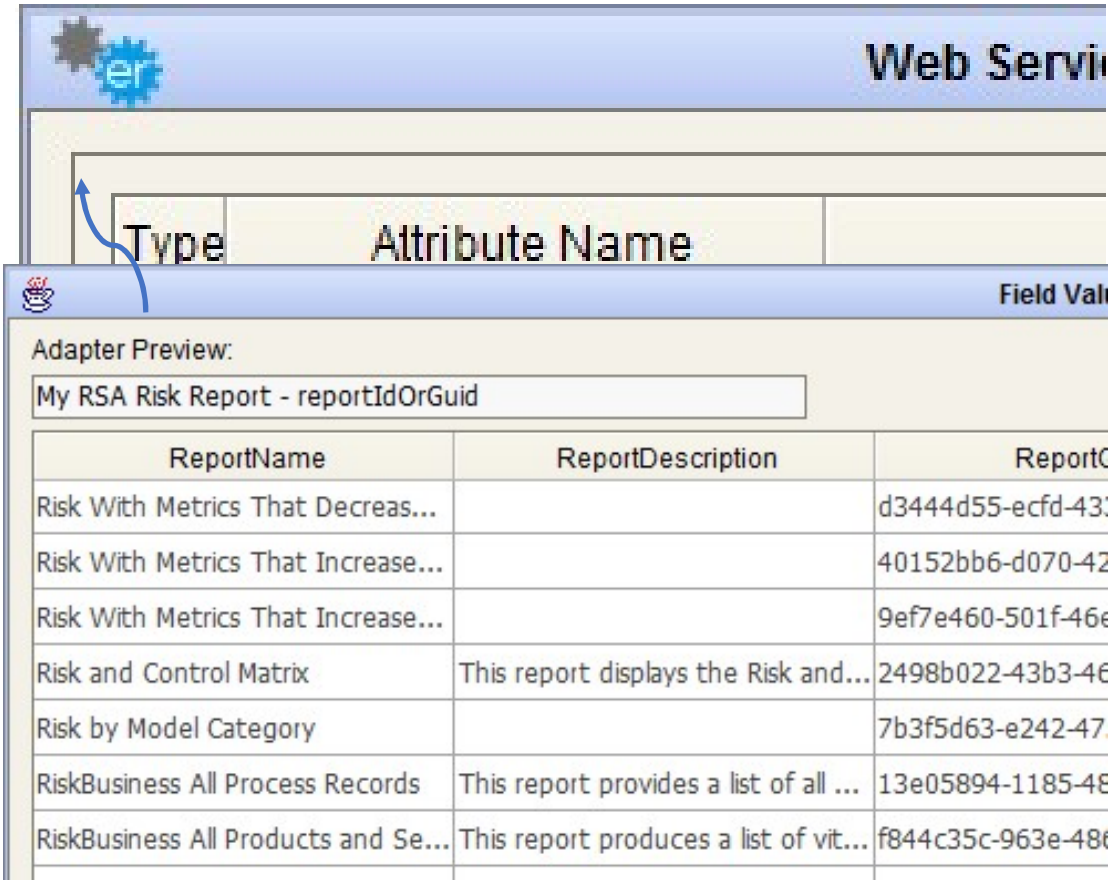
After selecting the desired operation, it is required to enter a name and a description for the adapter into the appropriate fields; the user must browse the file system for the adapter folder (used to save intermediate and result file for adapter execution).

The next button steps into the next window that allows the user to set values for the Web Service parameters.

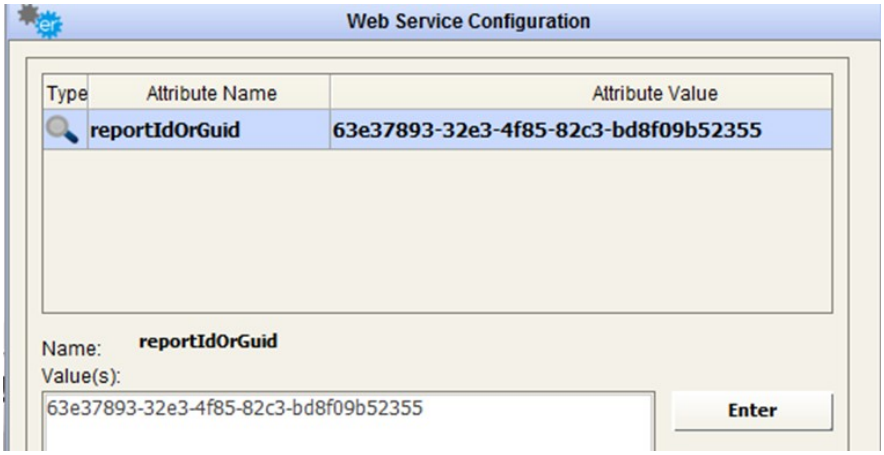
When a lens icon shows beside the parameter name:

this means that user may set the parameter value from a list of available items.

In the case of the RSA Archer “View Report” operation, double-click on the lens icon besides “reportIdOrGuid” parameter, a popup appears, with the available reports to query:

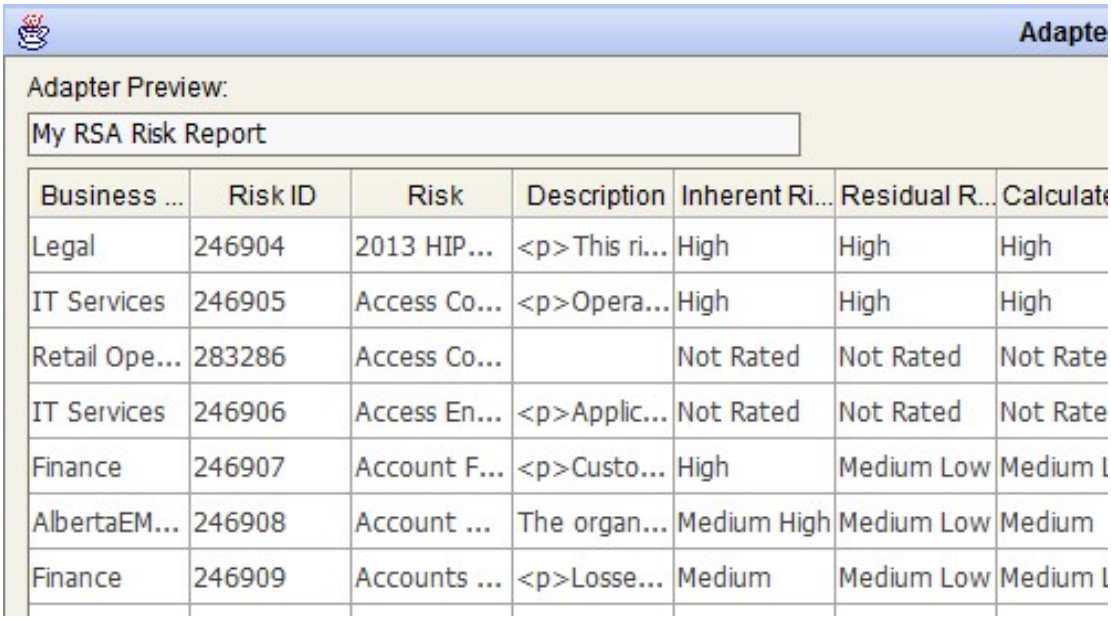


Select and apply the selection, and the desired report GUID fills the parameter value:

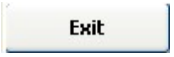


Should the provided account not be granted to read system tables (such as the one containing the list of user tables) the popup will be blank, but ***the parameters*** can always be ***manually filled***.

Pressing the button and a new pop-up window is opened and the response resulting from the Web Service call is displayed.



Pressing the button will store all the entered parameters for the web service adapter in the DT database.

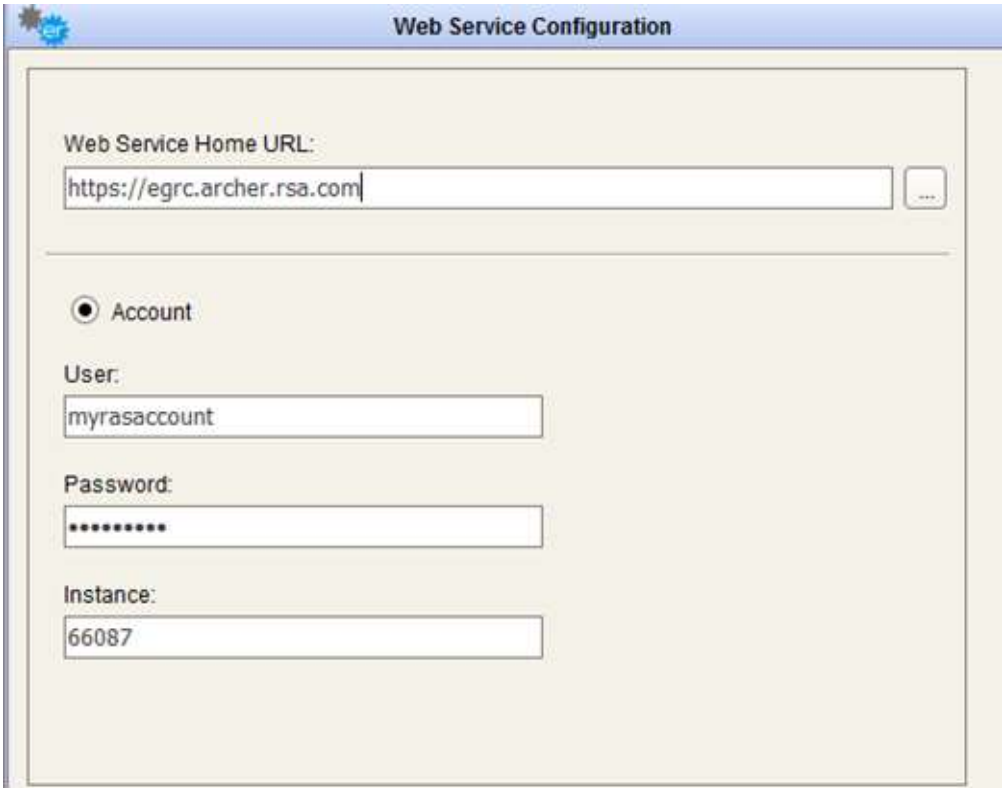
The 

button closes the data preview window and goes back to the Web Service configuration form.

1.2.6 RSA Archer – EXECUTE DATA FEED

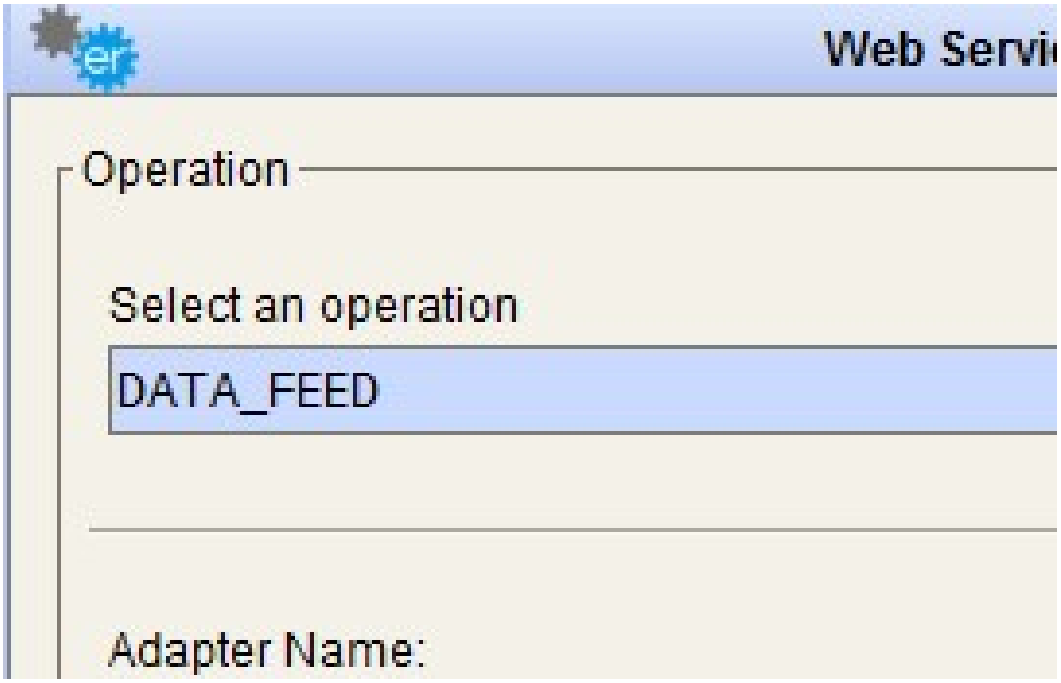
To use the adapter for **RSA Archer, to execute a data feed**, fill the field “Web Service Home URL” with the ServiceNow instance URL you want to connect to.

Then, provide a valid instance URL and ID, username/password to connect.



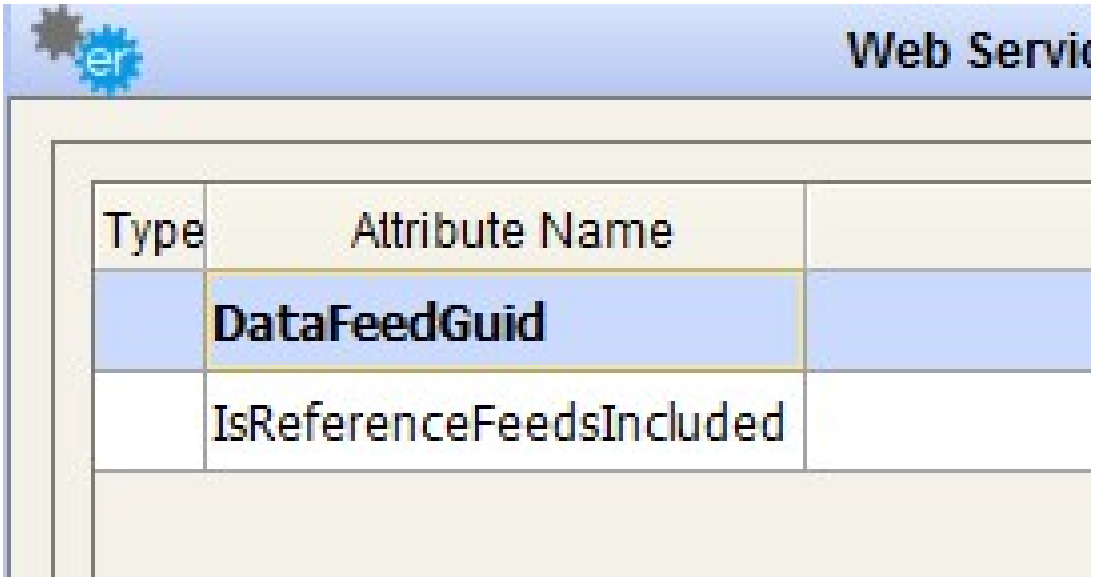
The 

button steps into the next window that shows the available operations of this Web Service; choose “**DATA FEED**”, to manage data feed executions.



After selecting the desired operation, it is required to enter a name and a description for the adapter in the appropriate fields; the user must browse the file system for the adapter folder (used to save intermediate and result file for adapter execution).

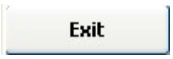
The next button Next steps into the next window that allows the user to set values for the Web Service parameters.



After that, user should complete configuration, filling the parameters (select the parameter, write value and press the button):

- **DataFeedGuid**: Data Feed GUID, it can be retrieved from RSA Archer application
- **IsReferenceFeedsIncluded**: must be “true” before any referenced data feeds can run. If you want to run a single data feed without any referenced feeds, you must ensure that the flag is set to “false”.

Pressing the button will store all the entered parameters for the web service adapter in the DT database.

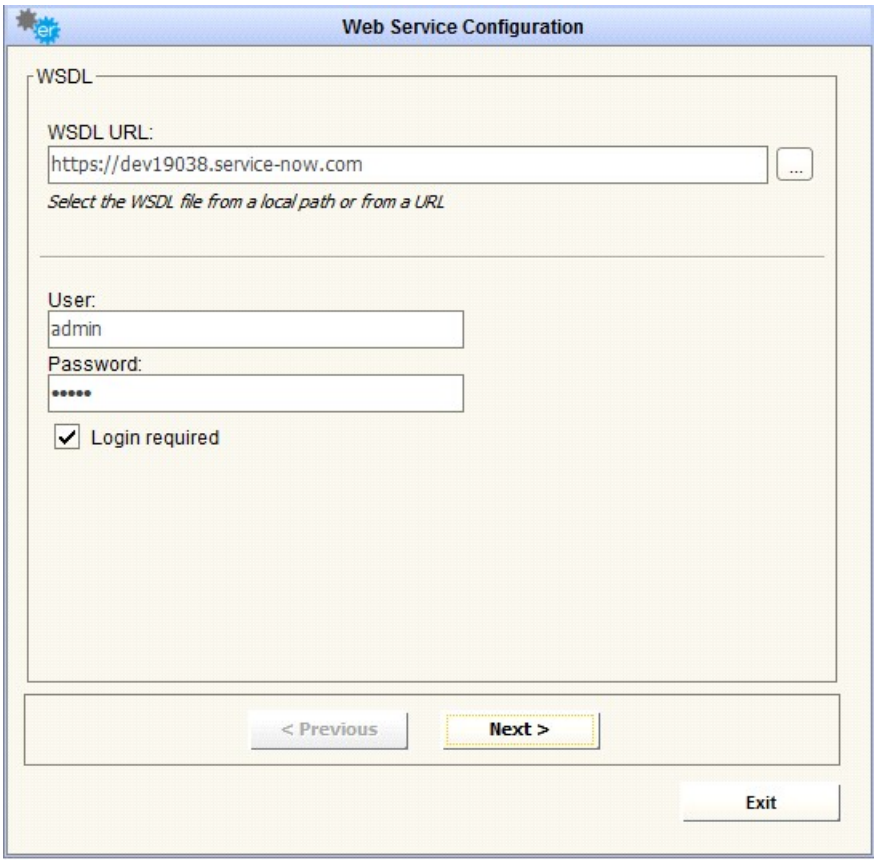
The 

button closes the data preview window and goes back to Web Service configuration form.

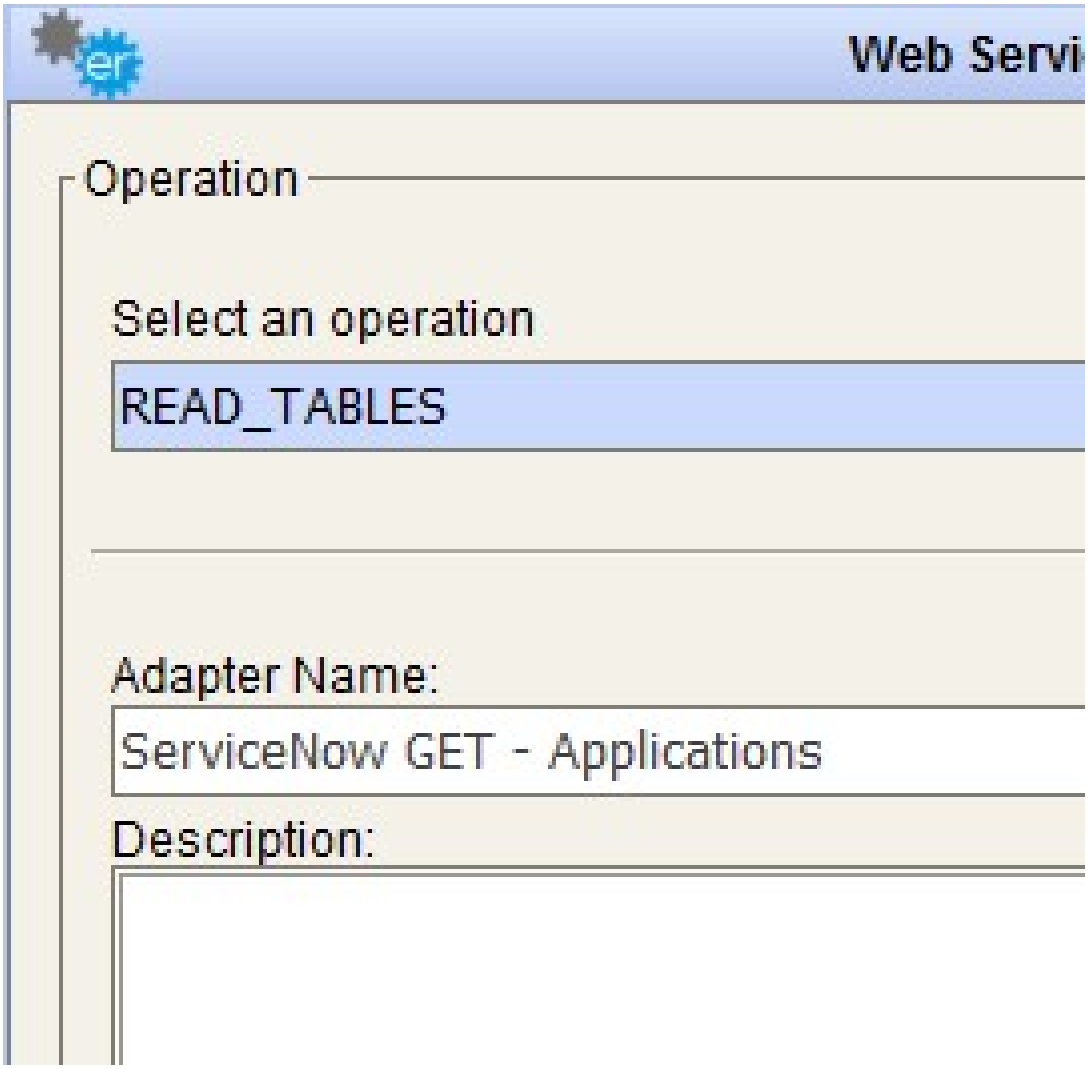
1.2.7 ServiceNow – READ TABLES

To use the adapter for **ServiceNow, to read table contents**, fill the field “Web Service Home URL” with the ServiceNow instance URL you want to connect to.

Then, provide a valid instance URL, username/password to connect.



The  button steps into the next window that shows the available operations of that Web Service; choose “**READ TABLES**”, to query data stored in ServiceNow tables.



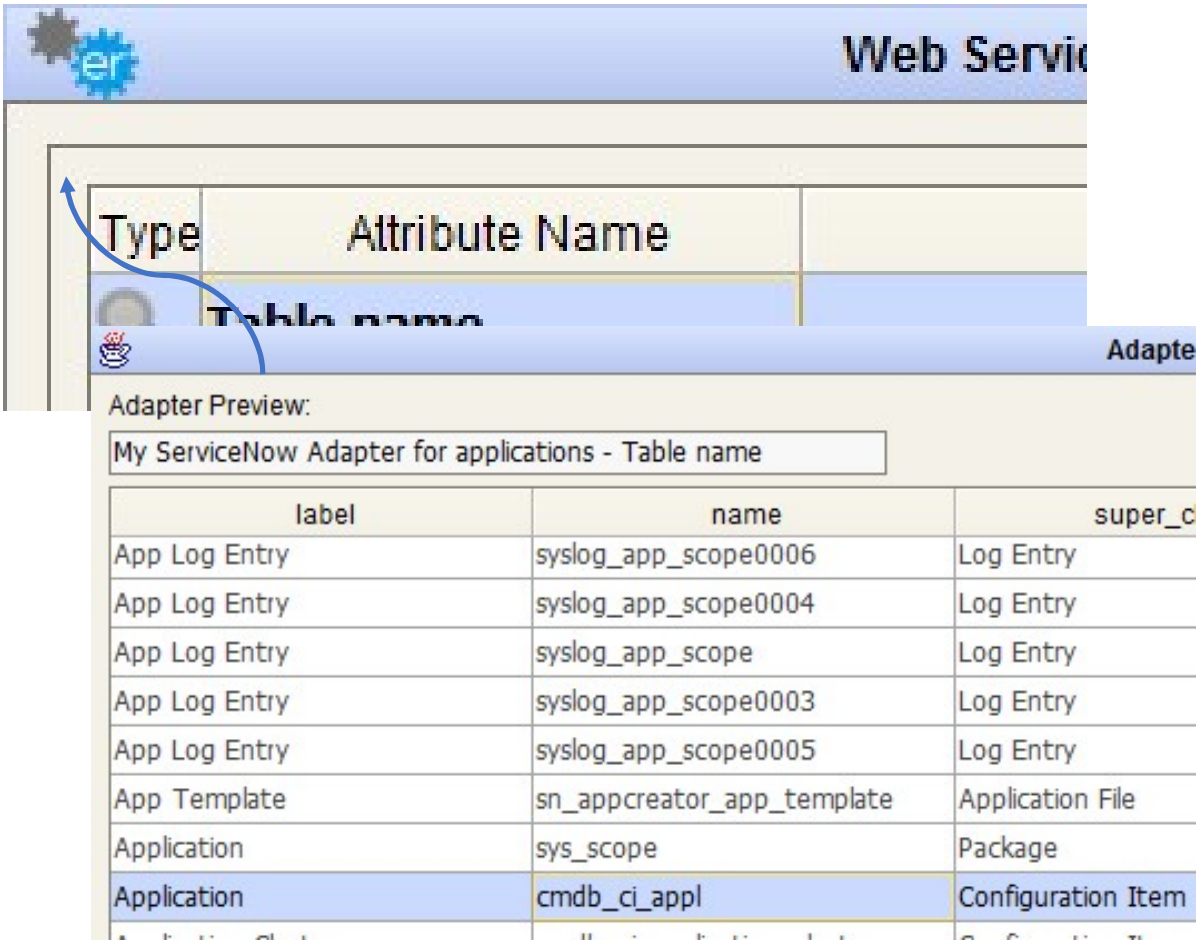
After selecting the desired operation, it is required to enter a name and a description for the adapter in the appropriate fields; the user must browse the file system for the adapter folder (used to save intermediate and result file for adapter execution).

The next button steps into the next window that allows the user to set values for the Web Service parameters.

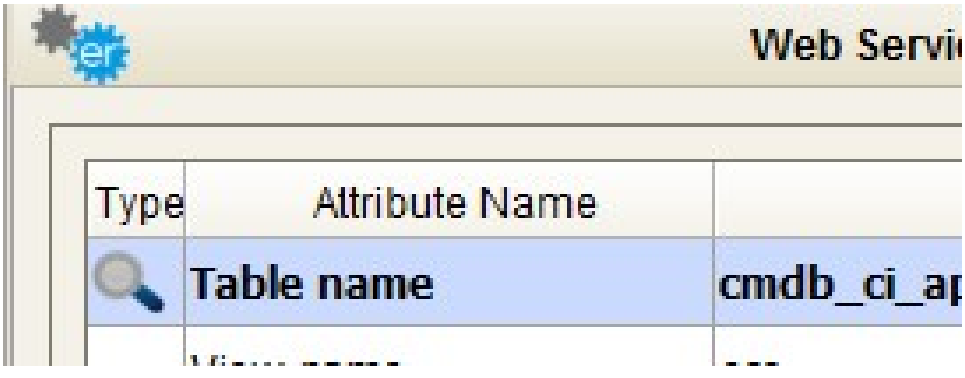
When a lens icon shows beside the parameter name:

this means that user may set the parameter value from a list of available items.

In case of ServiceNow “read tables” operation, with the double-click on the lens icon besides “Table name” parameter, a popup appears, with the available tables to query:



Select and apply the selection, and the desired table name (as expected by ServiceNow export service) fills the “Table name” parameter value:



Should the provided account not be granted to read system tables, as the one containing the list of user tables, the popup will be blank, but ***the parameters*** can always be ***manually filled***.

After that, the user may apply optional formatting/filter settings, filling the parameters (select the parameter, write value and press the button):

- **View Name**: name of the desired view. For example, to export fields visible from the Self Service view, parameter value is "**ess**"
- **Filters**: some additional parameters may be added to control for instance:
    - the sorting of results: "**ORDERBYname**"
    - the data filtering: **"category=Resource"**

  Concatenate different filters with ampersand; for instance:

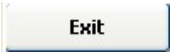
  category=Resource&ORDERBYname

Complete reference for filters can be found at the ServiceNow website:

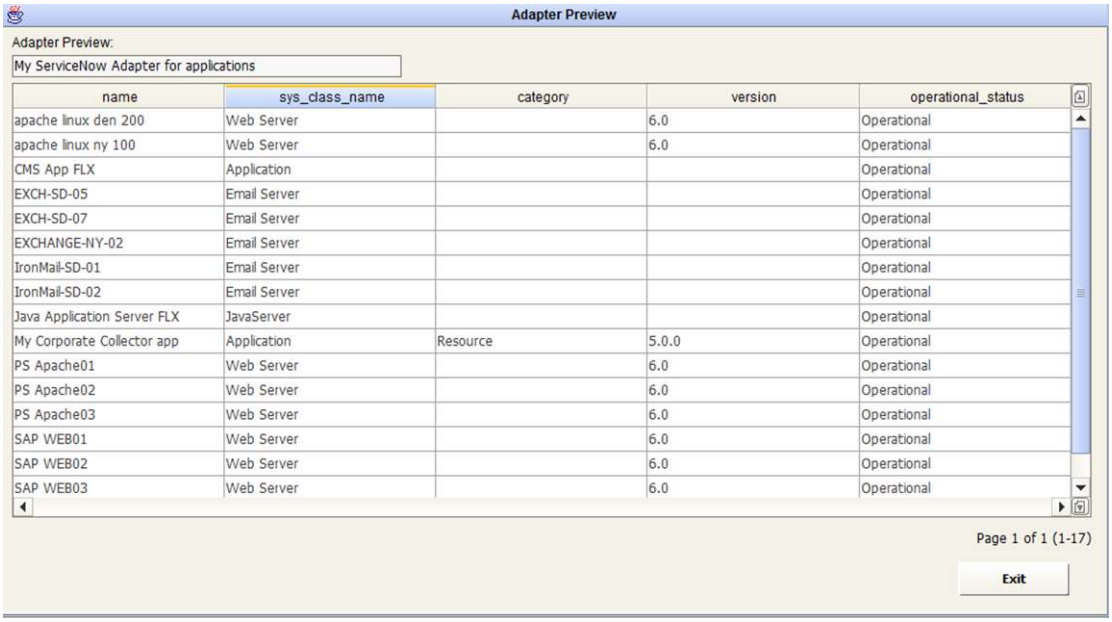
https://wiki.servicenow.com/index.php?title=Excel_Export_Threshold#Using_URL_Queries_to_Filter_List_Results

Pressing the  button will open a new pop-up window and the response resulting from the Web Service call is displayed.

Pressing the  button stores all the entered parameters for the web service adapter in the DT database.

The 

button closes the data preview window and goes back to Web Service configuration form.

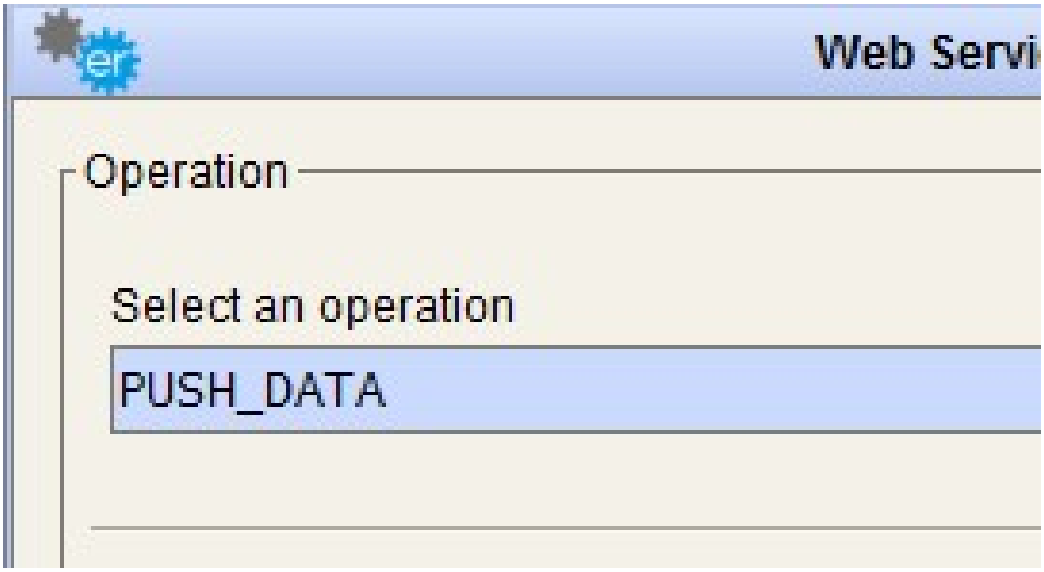


1.2.8 ServiceNow – PUSH DATA

To use the adapter for **ServiceNow, to push data to the system**, fill the field “Web Service Home URL” with the ServiceNow instance URL you want to connect to.

Then provide a valid instance URL, username/password to connect.

The Next button steps into the next window that shows the available operations of that Web Service; choose “**PUSH DATA**”, to query data stored in ServiceNow tables.



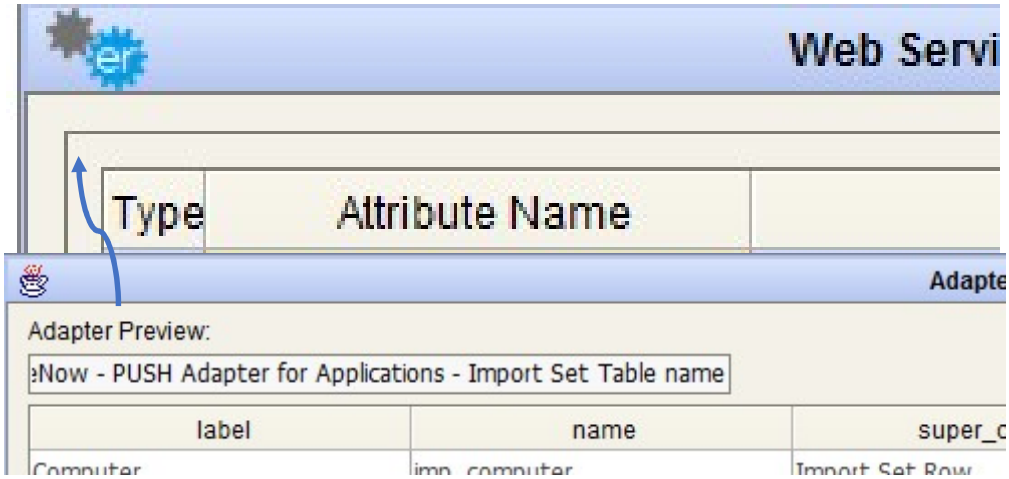
After selecting the desired operation, it is required to enter a name and a description for the adapter into the appropriate fields; the user must browse the file system for the adapter folder (used to save intermediate and result file for adapter execution).

The Next button steps into the next window that allows the user to set values for the Web Service parameters.

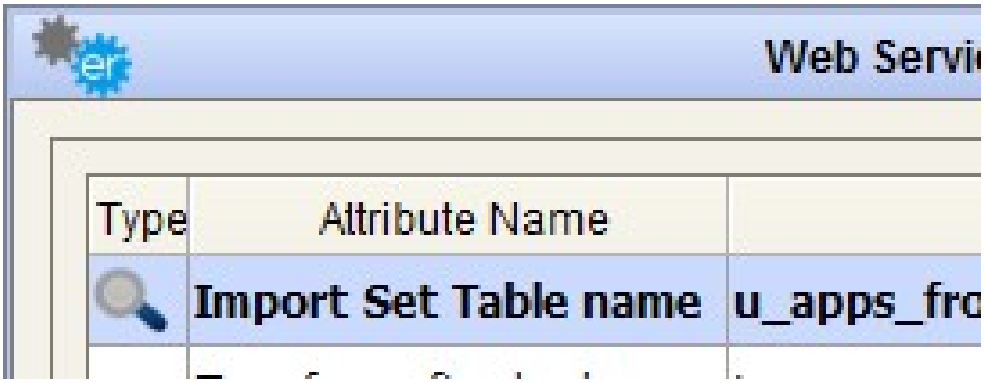
When a lens icon shows beside the parameter name, as for “**Import Set Table name**”:

this means that user may set the parameter value from a list of available items.

In case of ServiceNow “**push data**” operation, with the double-click on the lens icon, a table appears, with the available import set tables:



Select the proper one and apply the selection, and the desired import set table name fills the “Import Set Table name” parameter value:



Should the provided account not be granted to read system tables, as the one containing the list of user tables, the popup will be blank, but ***the parameters*** can always be ***manually filled***.

After that, user complete push operations configuration, filling the parameters (select the parameter, write value and press button):

- **Transform after load**: set to “true” if there is a Transformation Map set between the Import Set Table and ServiceNow target table to be applied after staging table load, “false” or blank otherwise
- **CSV separator**: comma as default

Documentation about import set tables and transformation map can be found at the ServiceNow website:

http://wiki.servicenow.com/index.php?title=Importing_Data_Using_Import_Sets#Posting_CSV_or_Excel_Files_Directly_to_an_Import_Set&gsc.tab=0

Pressing the SAVE button stores all the entered parameters for the web service adapter in the DT database.

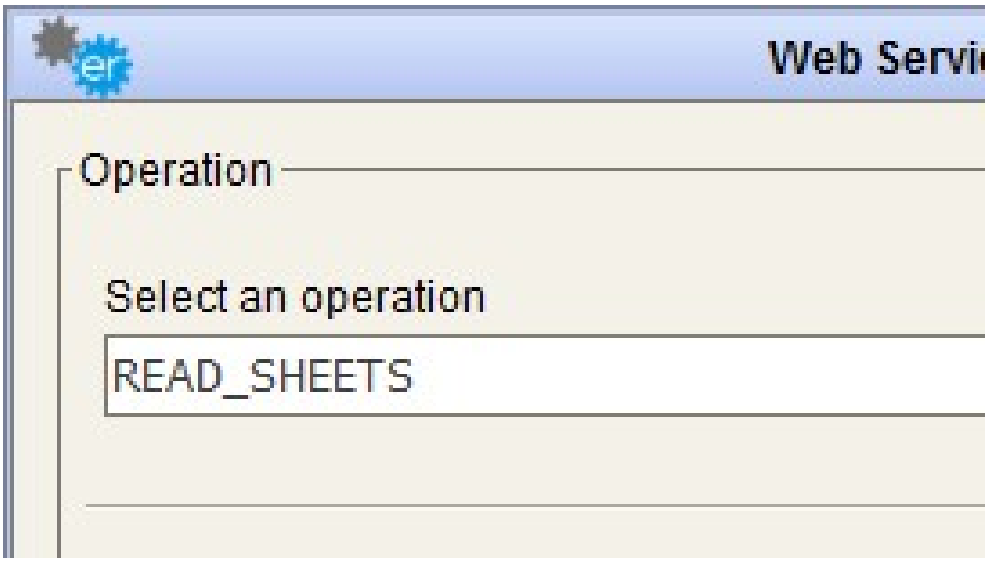
The Exit button closes the data preview window and goes back to Web Service configuration form.

1.2.9 SmartSheet

To use the adapter for **SmartSheet**, fill the field “Web Service Home URL” with the SmartSheet home page URL you want to connect to.

Then, provide a valid **access token** to connect. To generate an access token, requested to authenticate the SmartSheet service, follow instructions from https://smartsheet-platform.github.io/api-docs/?shell

The Next > button steps into the next window that shows the available operation of that Web Service, which is “**READ SHEETS**”, allowing user to query all data stored in user worksheets.



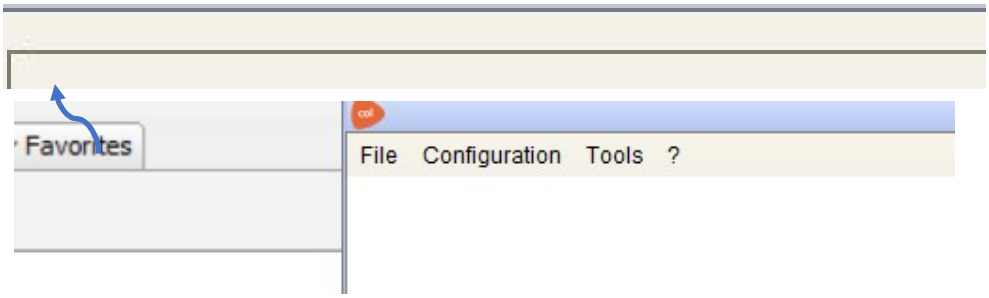
After selecting the desired operation, it is required to enter a name and a description for the adapter in the appropriate fields; the user must browse the file system for the adapter folder (used to save intermediate and result file for adapter execution).

The next button Next steps into the next window that allows the user to set values for the Web Service parameters.

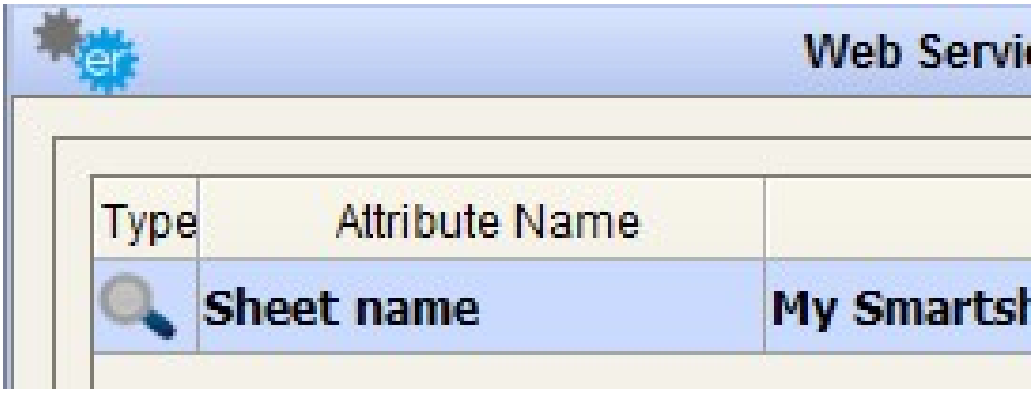
When a lens icon shows beside the parameter name:

this means that user may choose the parameter value from a list of available items.

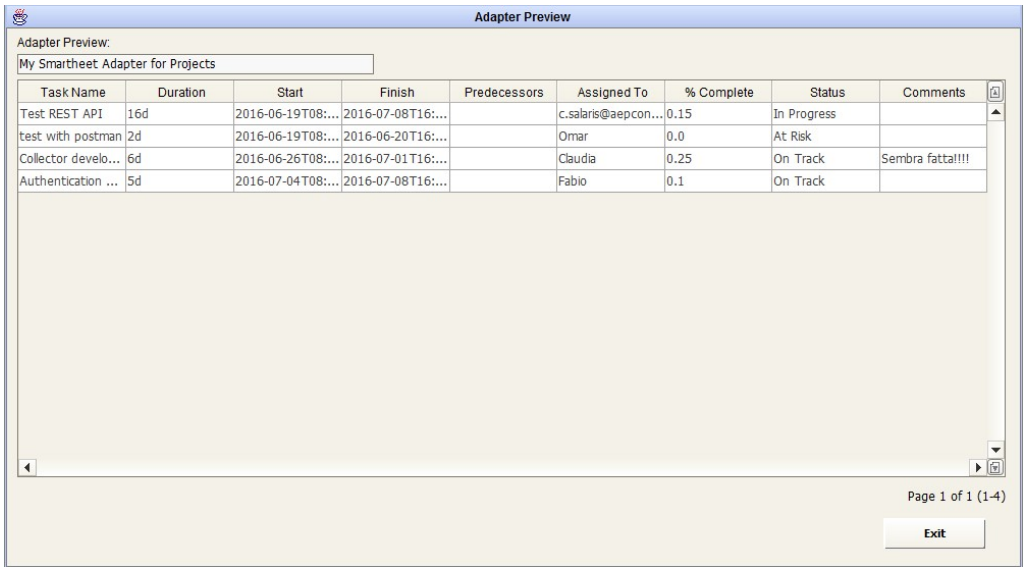
In case of SmartSheet “read sheets” operation, with the double-click on the lens icon, a table appears, with the available sheets to query:



Select and apply the selection, and the desired sheet name (as expected by Smartsheet export service) fills the “Sheet name” parameter value:



Pressing the button opens a new pop-up window and the response resulting from the Web Service call is displayed.

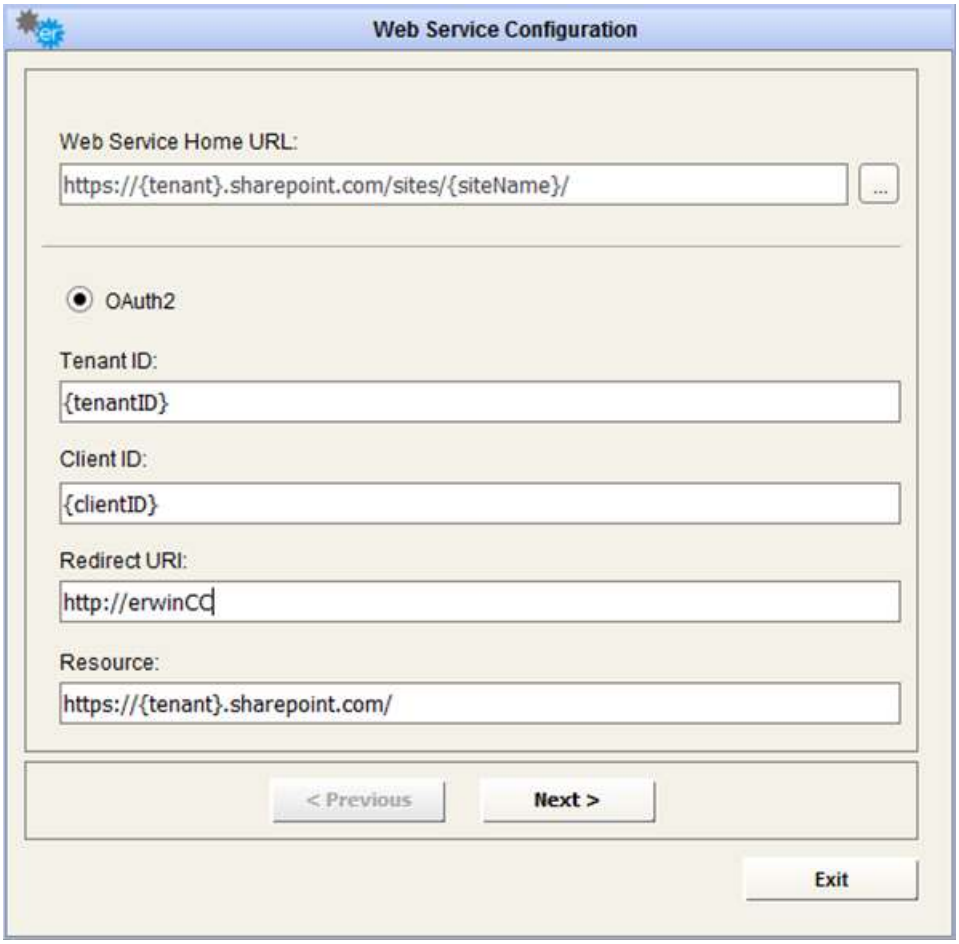


Pressing the button stores all the entered parameters for the web service adapter in the DT database.

The Exit button closes the data preview window and goes back to Web Service configuration form.

1.2.10 SharePoint Online

To use the adapter for **SharePoint Online**, fill the field “Web Service Home URL” with the SmartSheet home page URL you want to connect to.

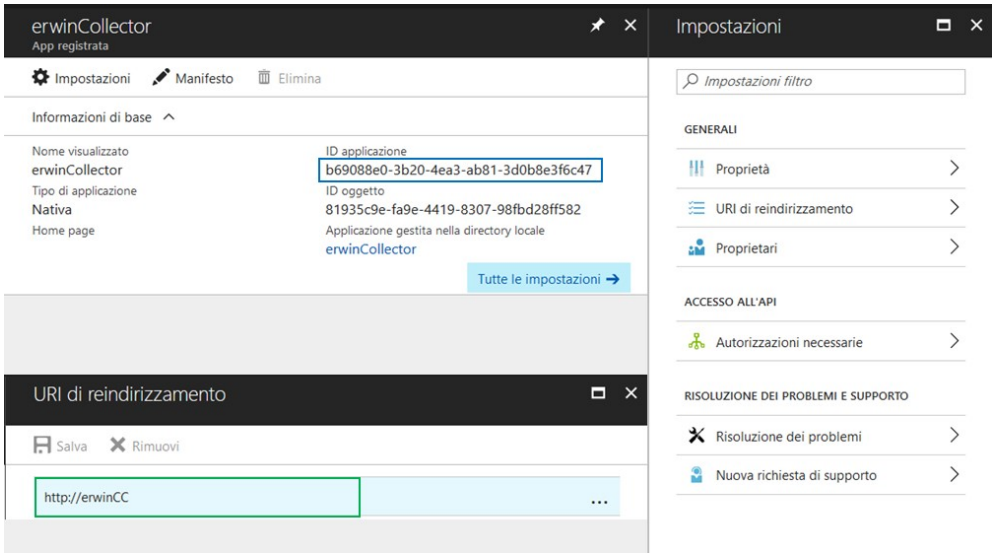


To provide required parameters, please ensure that erwin DT is registered on target tenant Azure Active Directory.

See: https://docs.microsoft.com/en-us/azure/app-service/app-service-mobile-how-to-configure-active-directory-authentication#optional-configure-a-native-client-application about registering a **native application**.

Register DT with:

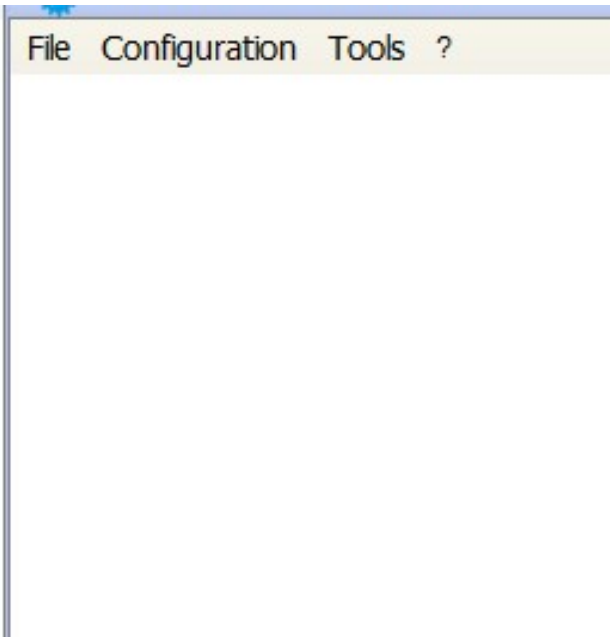
- **Name**: erwinCollector
- **Application Type**: Native
- **Redirect URI**: http://erwinCC



and provide all grants to read site list, for specific Microsoft API:

- Microsoft Graph:
    - Read items in all site collection
    - Access directory as the signed in user
- Microsoft Azure Active Directory:
    - Read all users' basic profiles
    - Sign in and read user profile
- Office 365 SharePoint Online (Microsoft.SharePoint)
    - Read and write items and lists in all site collections
    - Read and write items in all site collections
    - Read items in all site collections
    - Read managed metadata

The Next button steps into the next window, which requires user to browse to authorization URL and provide grant to the applica

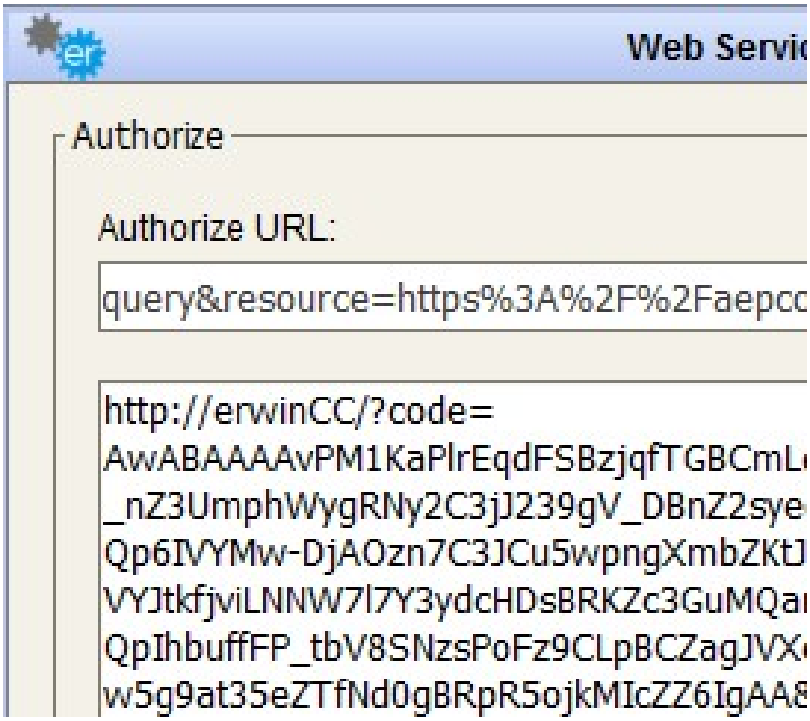


er

Wait for the browser to complete the request; a new page will be presented, with a URL like:

http://erwinCC/?code=
AwABAAAAvPM1KaPlrEqdFSBzjqfTGBCmLdgfSTLEMPGYuNHSUYB
rqqf_ZT_p5uEAEJJ_nZ3UmphWygRNy2C3jJ239gV_DBnZ2syeg95Ki-
374WHUP-i3yIhv5i-7KU2CEoPXwURQp6IVYMw-
DjAOzn7C3JCu5wpngXmbZKtJdWmiBzHpcO2aICJPu1KvJrDLDP20c
hJBXzVYJtkfjviLNNW7l7Y3ydcHDsBRKZc3GuMQanmcghXPyoDg41g
8XbwPudVh7uCmUponBQpIhbuffFP_tbV8SNzsPoFz9CLpBCZagJVX
eqWoYMPe2dSsPiLO9Alf_YIe5zpi-
zY4C3aLw5g9at35eZTfNd0gBRpR5ojkMIcZZ6IgAA&session_state=7
B29111D-C220-4263-99AB-6F6E135D75EF&state=D79E5777-702E-
4260-9A62-37F75FF22CCE &session_state=7B29111D-C220-4263-
99AB-6F6E135D75EF&state=D79E5777-702E-4260-9A62-
37F75FF22CCE

Copy the **entire URL** to DT text area:



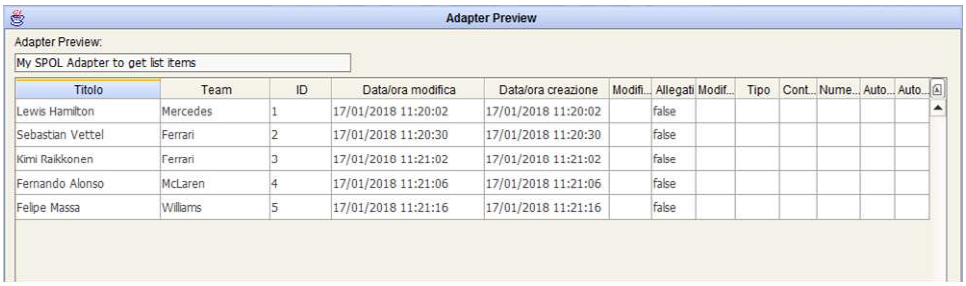
The Next button steps into the next window that shows the available operation of that Web Service, which is "**READ LISTS**.

After selecting the desired operation, it is required to enter a name and a description for the adapter in the appropriate fields; the user must browse the file system for the adapter folder (used to save intermediate and result file for adapter execution).

The next button Next steps into the next window that allows the user to set values for the Web Service parameters.

Select the list – double click on lens icon beside "List GUID" open a list of available lists – and apply the selection, and the desired list GUID (as expected by Sharepoint export API) fills the parameter value

Pressing the Test button opens a new pop-up window and the response resulting from the Web Service call is displayed.



Pressing the SAVE button stores all the entered parameters for the web service adapter in the DT database.

The Exit button closes the data preview window and goes back to Web Service configuration form.

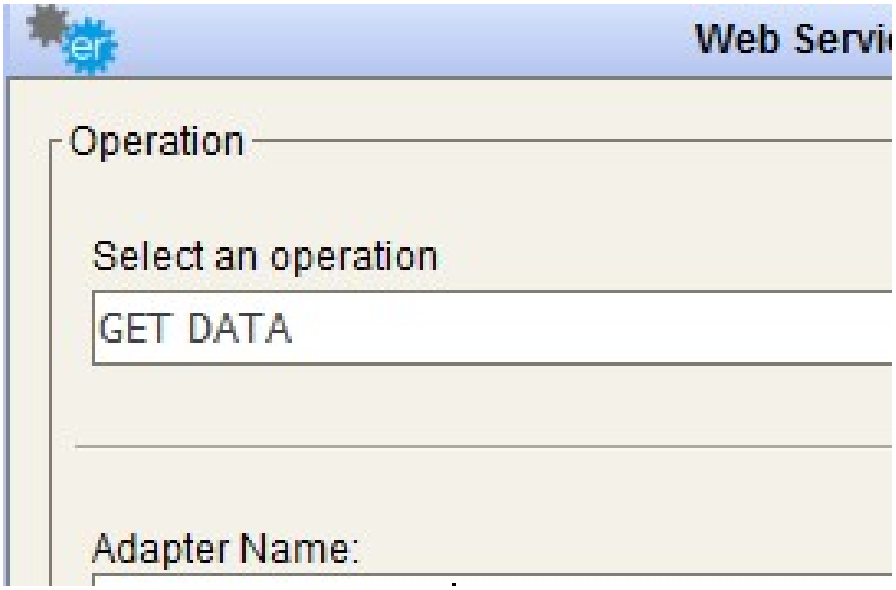
1.2.11 Generic data provider - CSV

To use the adapter for **a generic http data provider**, fill the field “Web Service Home URL” with the URL to the data provider service, ex:

https://pkgstore.datahub.io/core/country-list/data_csv/data/d7c9d7cfb42cb69f4422dec222dbbaa8/data_csv.csv

Then, provide a valid User/Password if service requires (basic) authentication, or select “Anonymous” access option

The Next button steps into the next window that shows the available operation of that Web Service, which is “**GET DATA**”, allowing user to query web data.

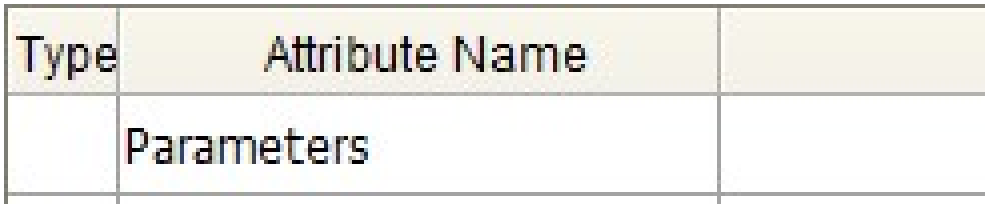


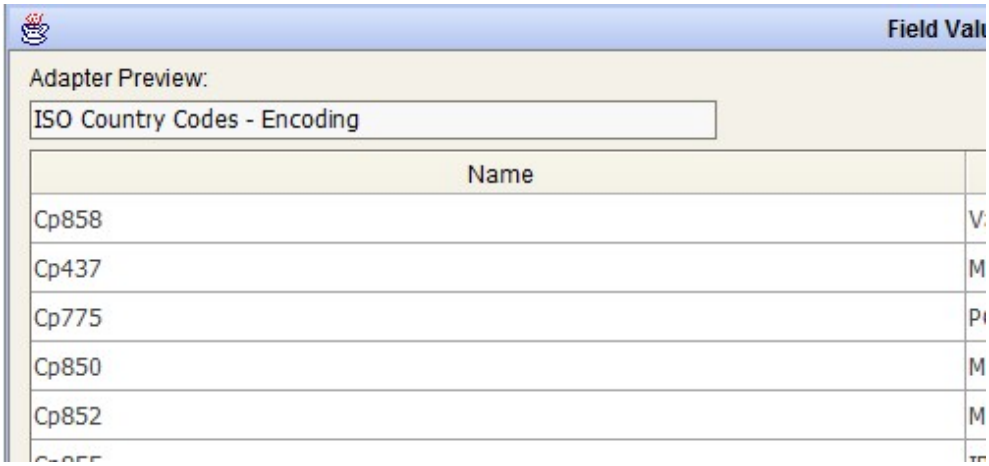
After selecting the desired operation, it is required to enter a name and a description for the adapter in the appropriate fields; the user must browse the file system for the adapter folder (used to save intermediate and result file for adapter execution).

The next button Next steps into the next window that allows the user to set values for the Web Service parameters.

When a lens icon shows beside the parameter name:

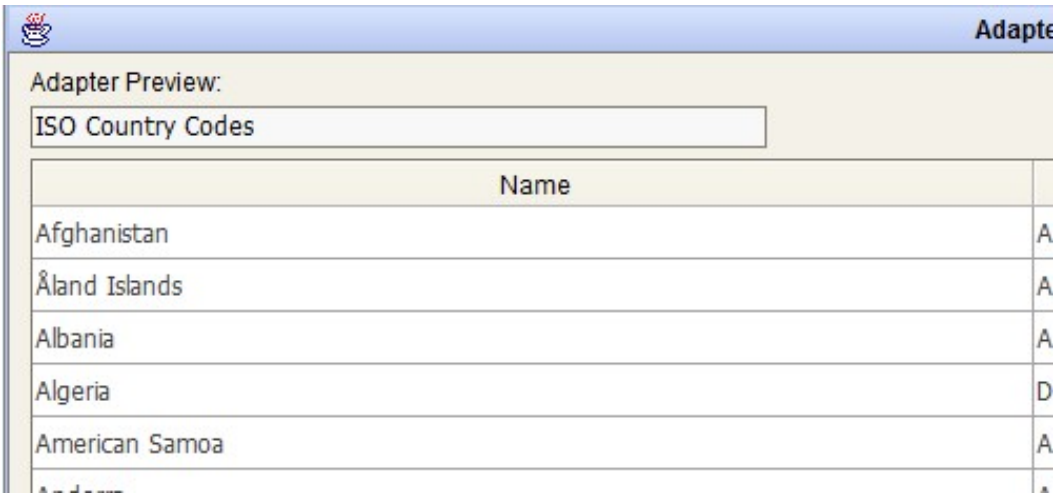
this means that user may choose the parameter value from a list of available items. In case of this example, with the double-click on the lens icon, a table appears, with the available encoding:



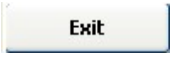


Select and apply the selection, and the desired values for other fields, like the CSV column separator and parameters should the service require them.

Pressing the button opens a new pop-up window and the response resulting from the Web Service call is displayed.



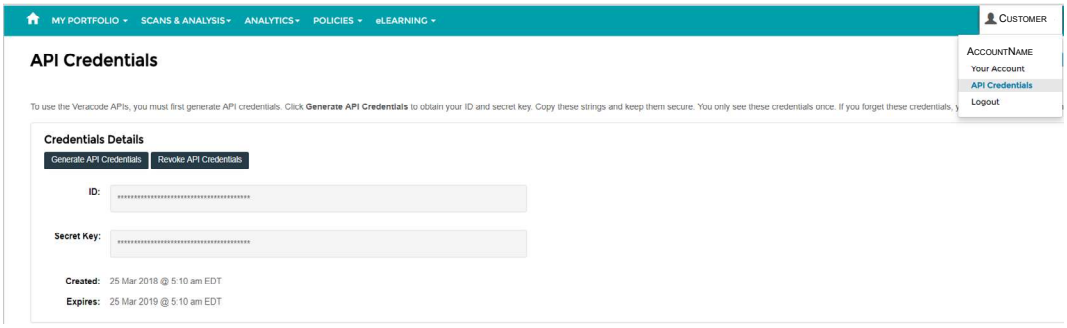
Pressing the button stores all the entered parameters for the web service adapter in the DT database.

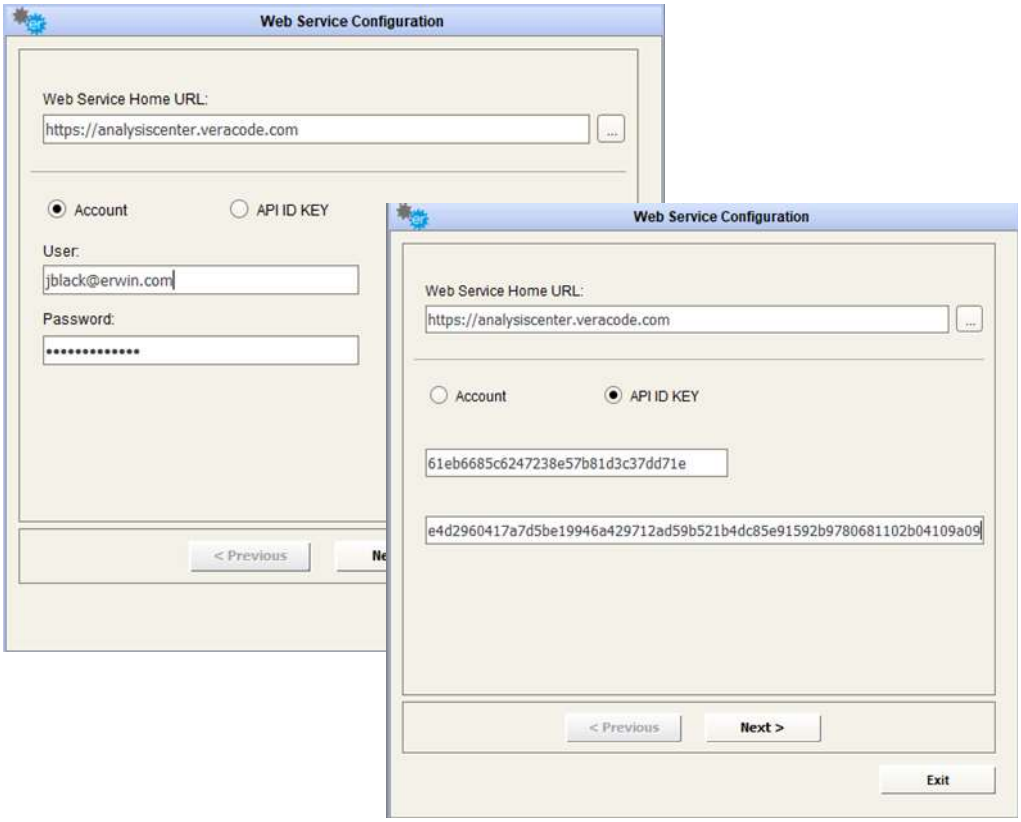
The 

button closes the data preview window and goes back to Web Service configuration form.

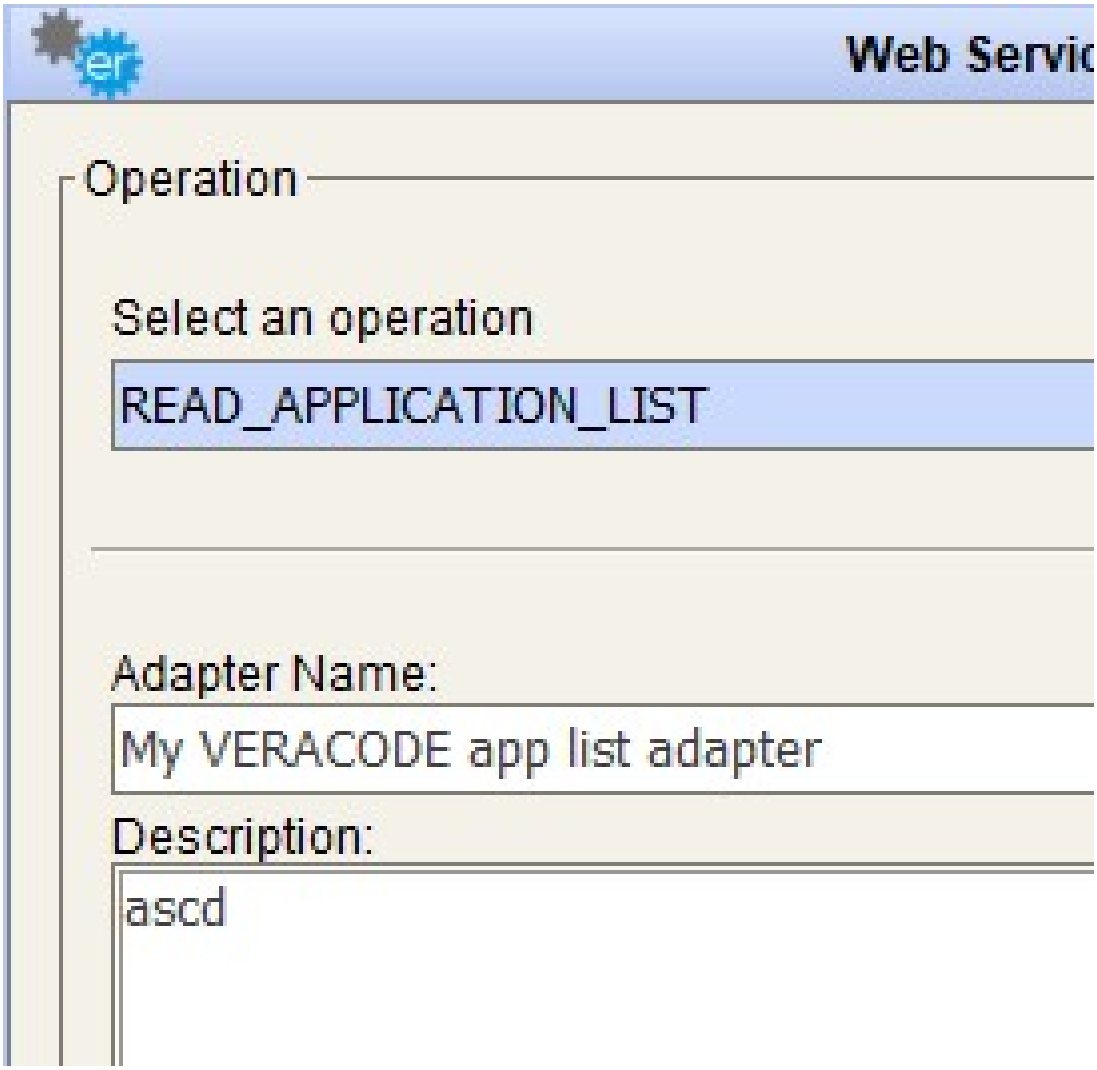
1.2.12 Veracode – READ APPLICATION LIST

To use the adapter for Veracode**, to read application list**, fill the field "Web Service Home URL" with the Veracode URL you want to connect to. Then, provide a valid username/password to connect, or the API credentials (ID/KEY), that can be generated by a valid account. Please, make sure that the account has the correct permissions to use Veracode API: https://help.veracode.com/reader/LMv_dtSHyb7iIxAQznC~9w/2nDnsqbBHfc2TPcl_Ps~KQ





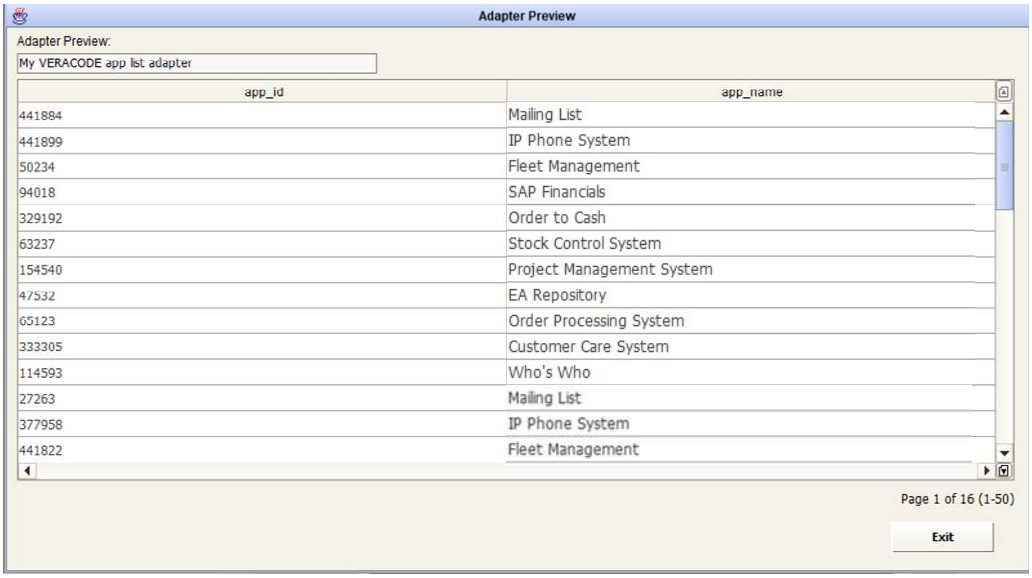
The  button steps into the next window that shows the available operations of this Web Service; choose "**READ APPLICATION LIST**", to get the list of application names and their Veracode ID.



After selecting the desired operation, it is required to enter a name and a description for the adapter into the appropriate fields; the user must browse the file system for the adapter folder (used to save intermediate and result file for adapter execution).

The next button steps into the next window that allows the user to set values for the Web Service parameters, not required for this operation.

Pressing the Test button and a pop-up window is opened and the response resulting from the Web Service call is displayed.

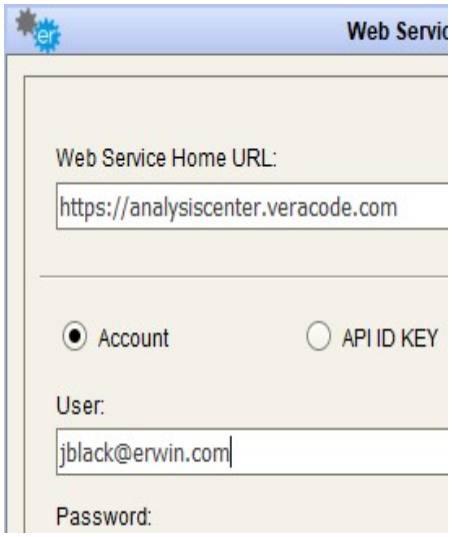


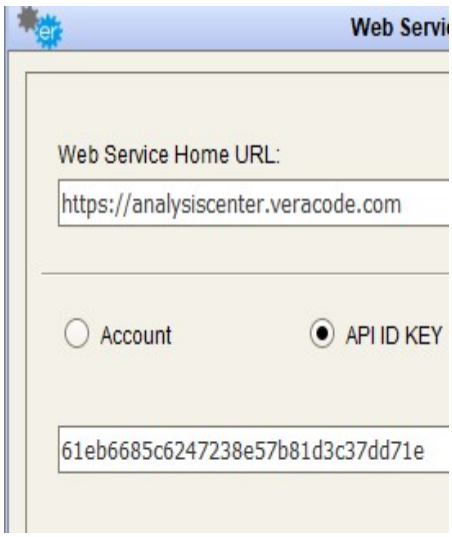
Pressing the SAVE button will store all the entered parameters for the web service adapter in the DT database.

The Exit button closes the data preview window and goes back to the Web Service configuration form.

1.2.13 Veracode – IMPORT APPLICATION DATA

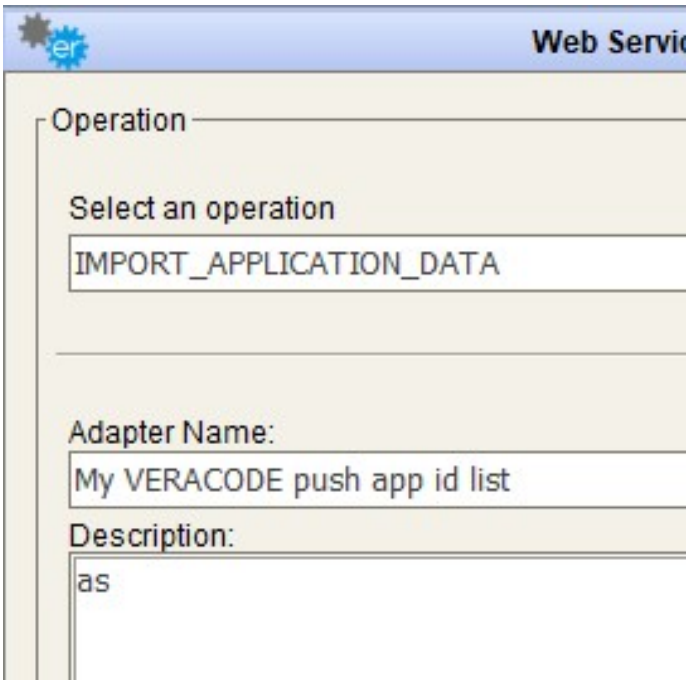
To use the adapter for Veracode**, to import application data**, fill the field “Web Service Home URL” with the Veracode URL you want to connect to. Then, provide a valid username/password to connect, or the API credentials (ID/KEY), that can be generated by a valid account. Please, make sure that the account has the correct permissions to use Veracode API: https://help.veracode.com/reader/LMv_dtSHyb7iIxAQznC~9w/2nDnsqbBHfc2TPcl_Ps~KQ





The 

button steps into the next window that shows the available operations of this Web Service; choose "**IMPORT APPLICATION DATA**", to configure an adapter that can be used in a workflow to push application metadata to Veracode (see later).



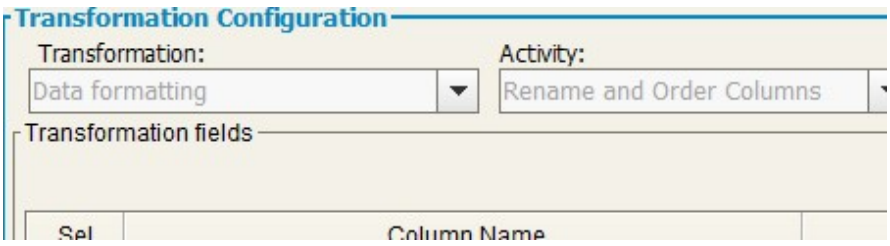
After selecting the desired operation, it is required to enter a name and a description for the adapter in the appropriate fields; the user must browse the file system for the adapter folder (used to save intermediate and result file for adapter execution).

The next button Next steps into the next window that allows the user to set values for the Web Service parameters, not required for this operation.

Pressing the SAVE button will store all the entered parameters for the web service adapter in the DT database.

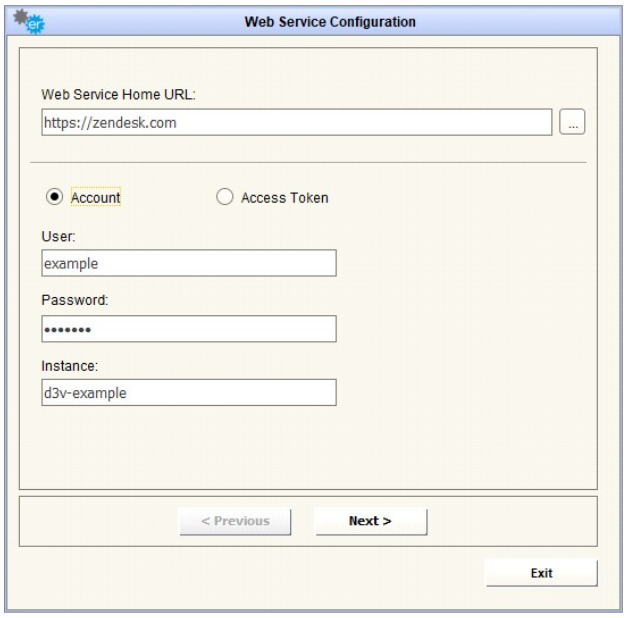
The Exit button closes the data preview window and goes back to Web Service configuration form.

The Adapter, as mentioned, can be used to push data to Veracode and fill applications metadata. For example, if the application object type in Veracode has two custom fields named “Casewise ID” and “Development Manager”, the alignment can be done configuring a workflow, with a source providing values for the fields, and with the target operation “Web Service POST”, using the configured adapter. To provide mappings between source and target field, use a “Data Formatting: Rename and Order Columns” transformation, with the required “**app_id**” and “**app_name**” fields, and all others desired fields. Where “**app_id**” (numeric) value is **null** or **0**, the application with provided name will be created, or updated, if an existing Veracode application with the same name already exists.

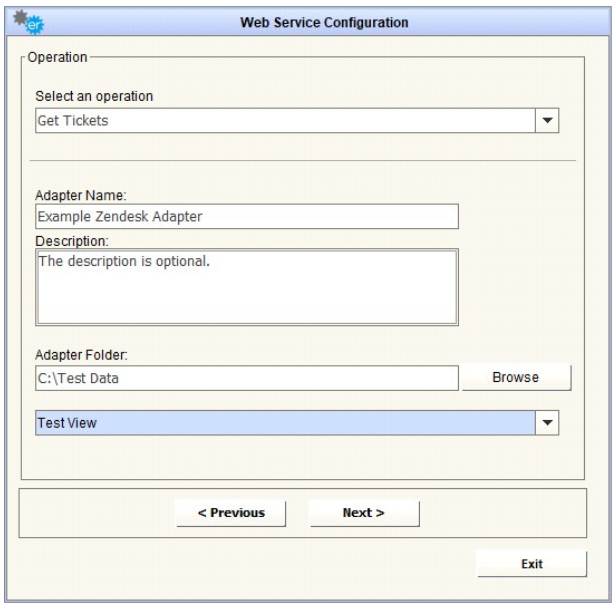


1.2.14 Zendesk – Get Tickets

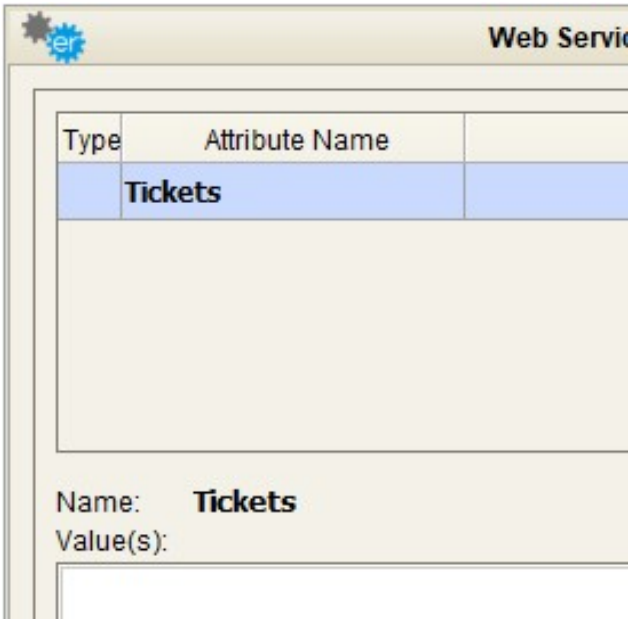
To use the adapter for Zendesk **to get tickets**, select the authorization method (Account / Access Token) and enter the relevant credentials.



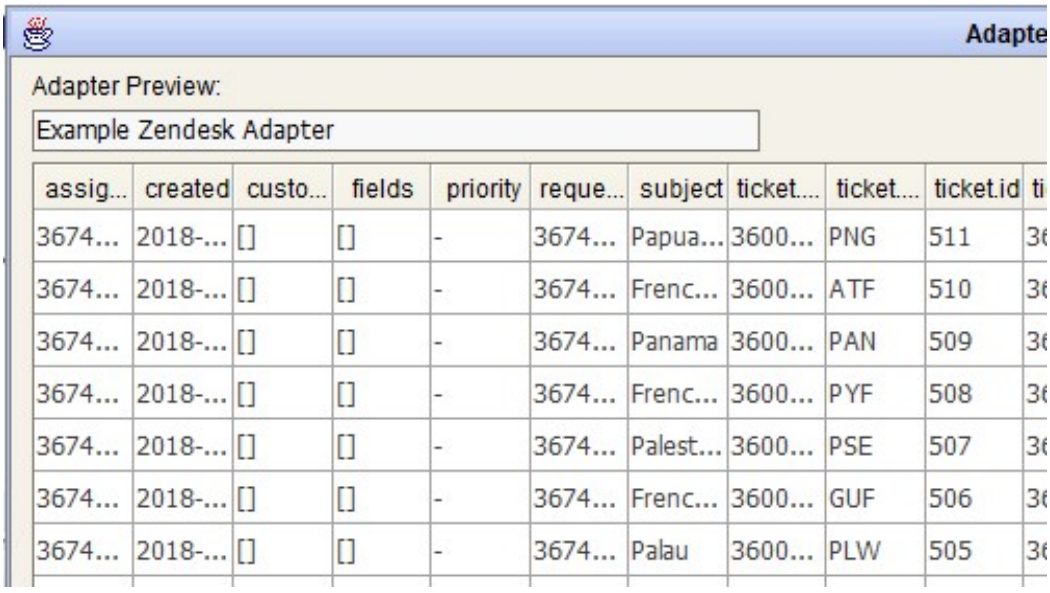
Click the  button to proceed to the next step, and select the operation “Get Tickets”. Enter a name and folder for the adapter, and select the view from which to retrieve tickets. A description can also optionally be entered for the adapter.



Click the Next > button to proceed to the next step, where the adapter can be previewed and/or saved.



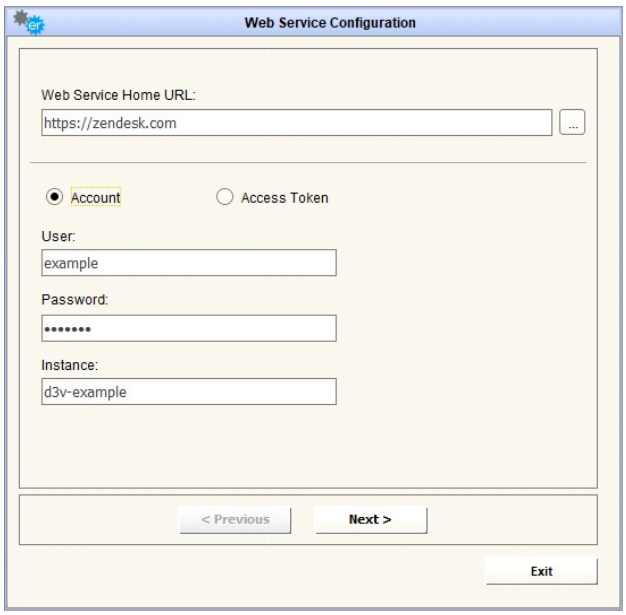
To preview a sample of the first 50 results, click the button.



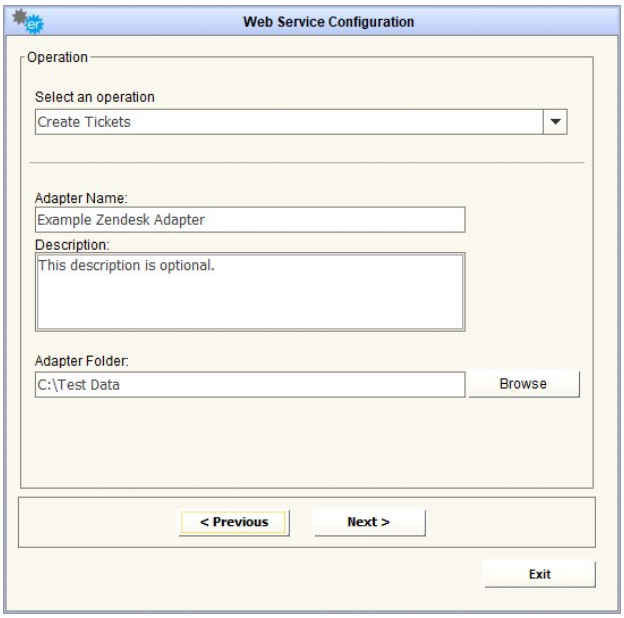
To save the adapter to the DT database, click the button.

1.2.15 Zendesk – Create Tickets

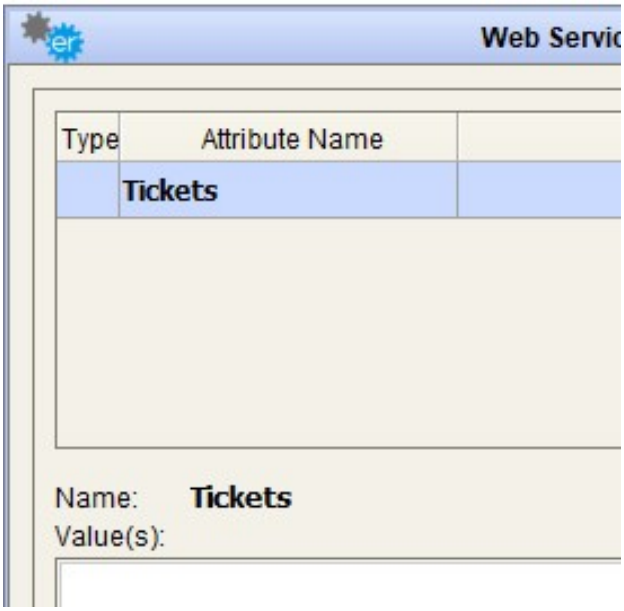
To use the adapter for Zendesk **to create new tickets**, select the authorization method (Account / Access Token) and enter the relevant credentials.



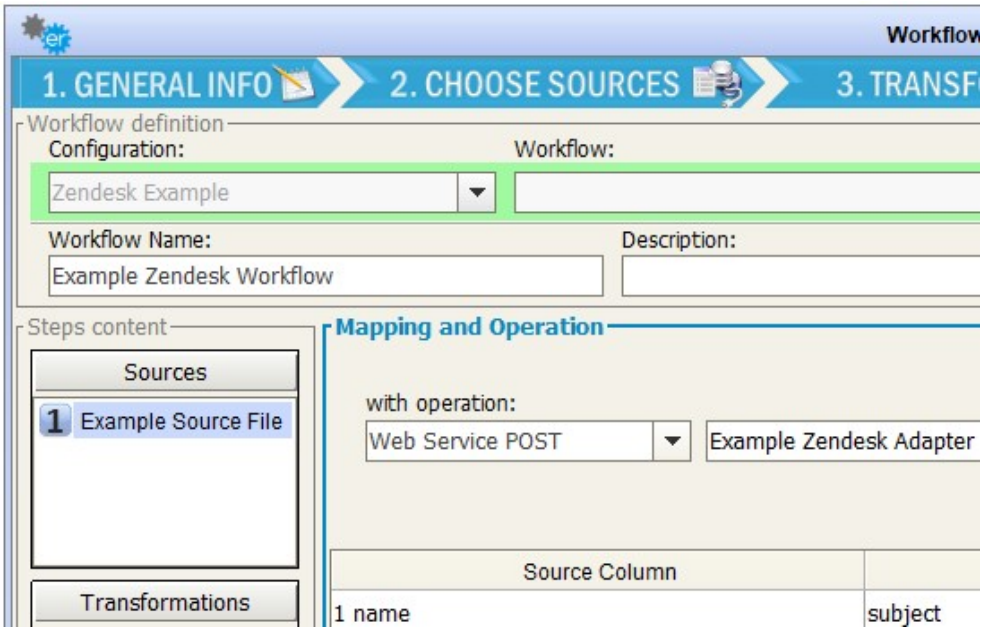
Click the  button to proceed to the next step, and select the operation “Create Tickets”. Enter a name and folder for the adapter. A description can also optionally be entered for the adapter.



Click the Next > button to proceed to the next step. To save the adapter to the DT database, click the button.

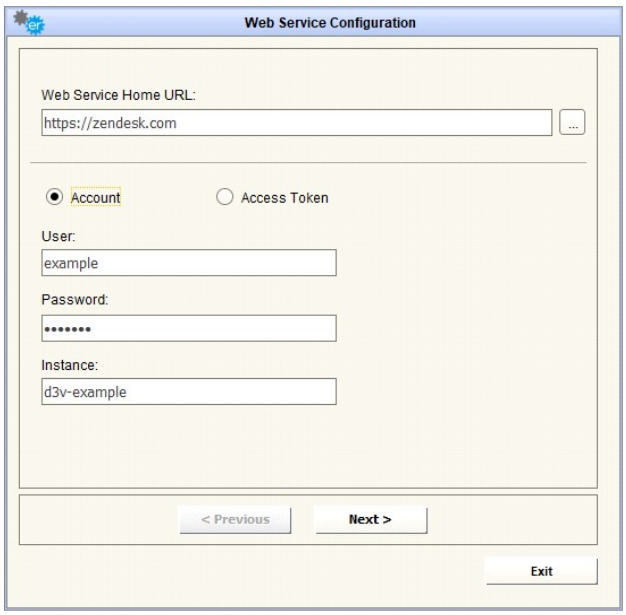


The adapter can now be used as a target endpoint when mapping data in a workflow. To select the adapter as a target, select “Web Service POST” as the operation and select your Zendesk adapter. A mapping table will appear, allowing the source fields to be mapped to relevant target fields for creating Zendesk tickets.

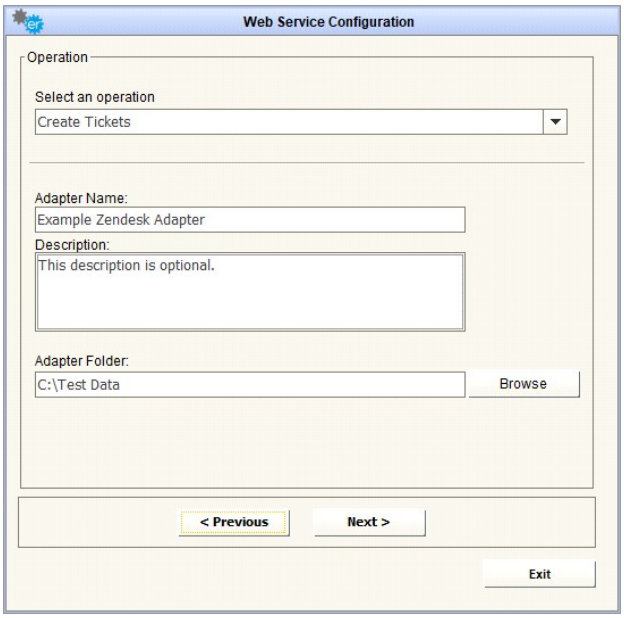


1.2.16 Zendesk – Update Tickets

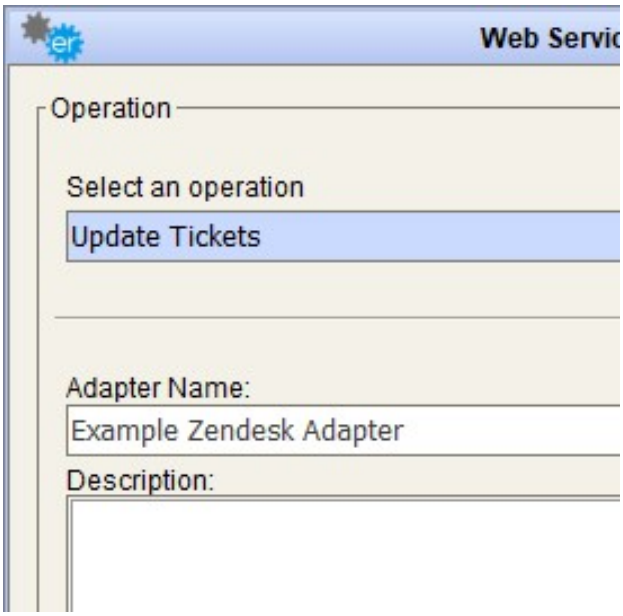
To use the adapter for Zendesk **to update existing tickets**, select the authorization method (Account / Access Token) and enter the relevant credentials.



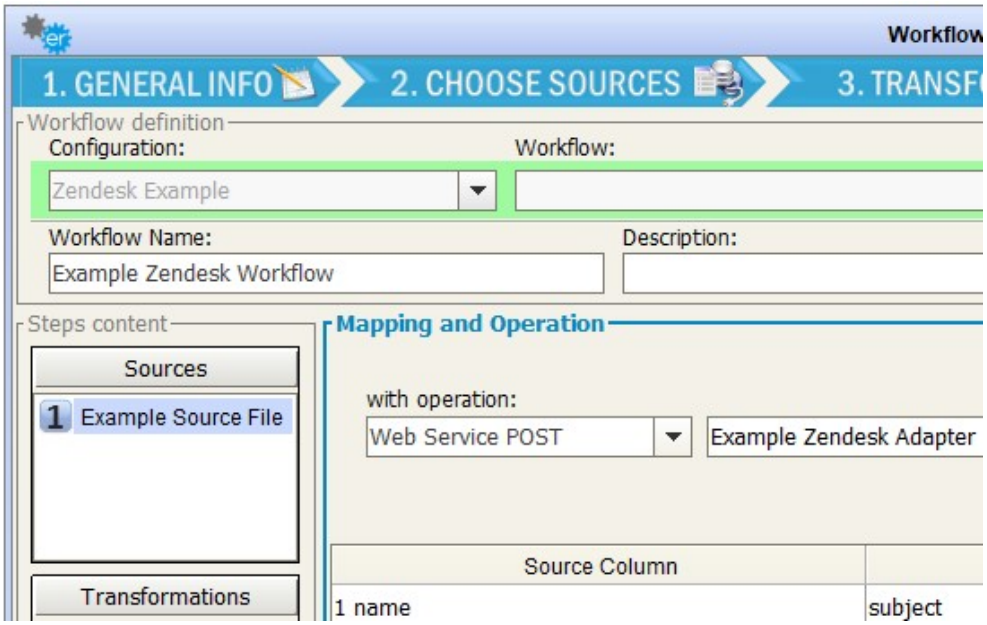
Click the  button to proceed to the next step, and select the operation “Update Tickets”. Enter a name and folder for the adapter. A description can also optionally be entered for the adapter.



Click the Next > button to proceed to the next step. To save the adapter to the DT database, click the button.

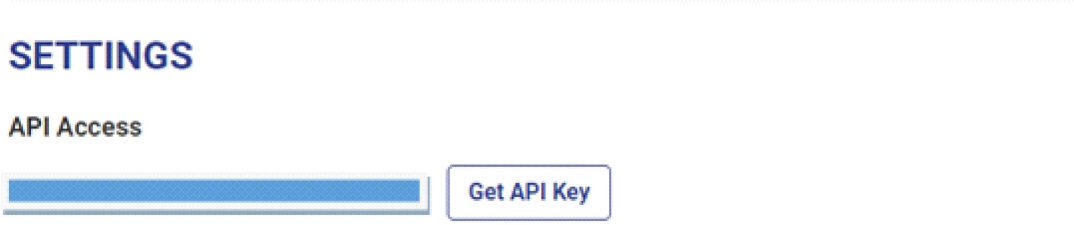


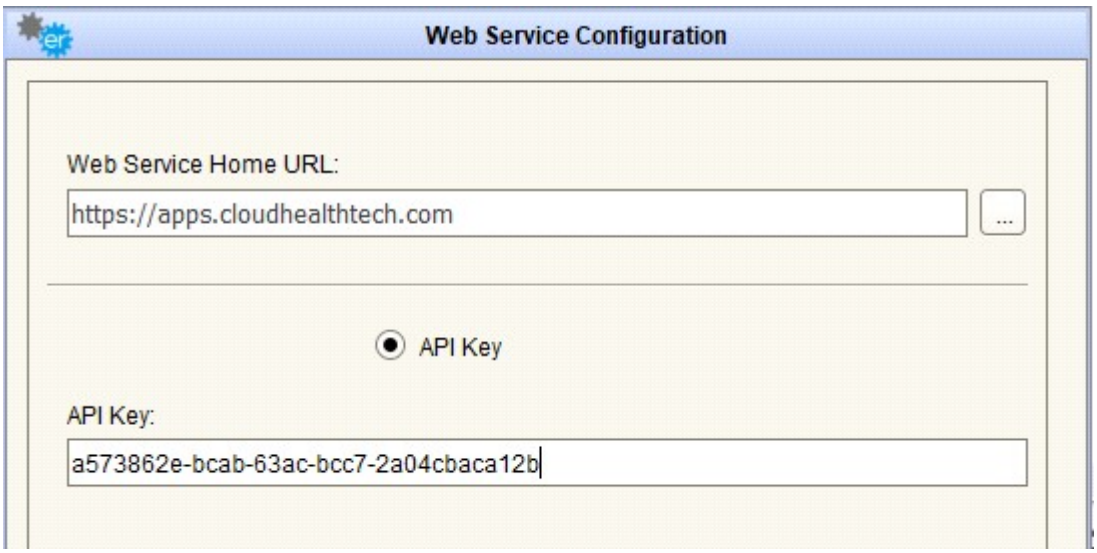
The adapter can now be used as a target endpoint when mapping data in a workflow. To select the adapter as a target, select “Web Service POST” as the operation and select your Zendesk adapter. A mapping table will appear, allowing the source fields to be mapped to relevant target fields for updating Zendesk tickets (using the id as a key).



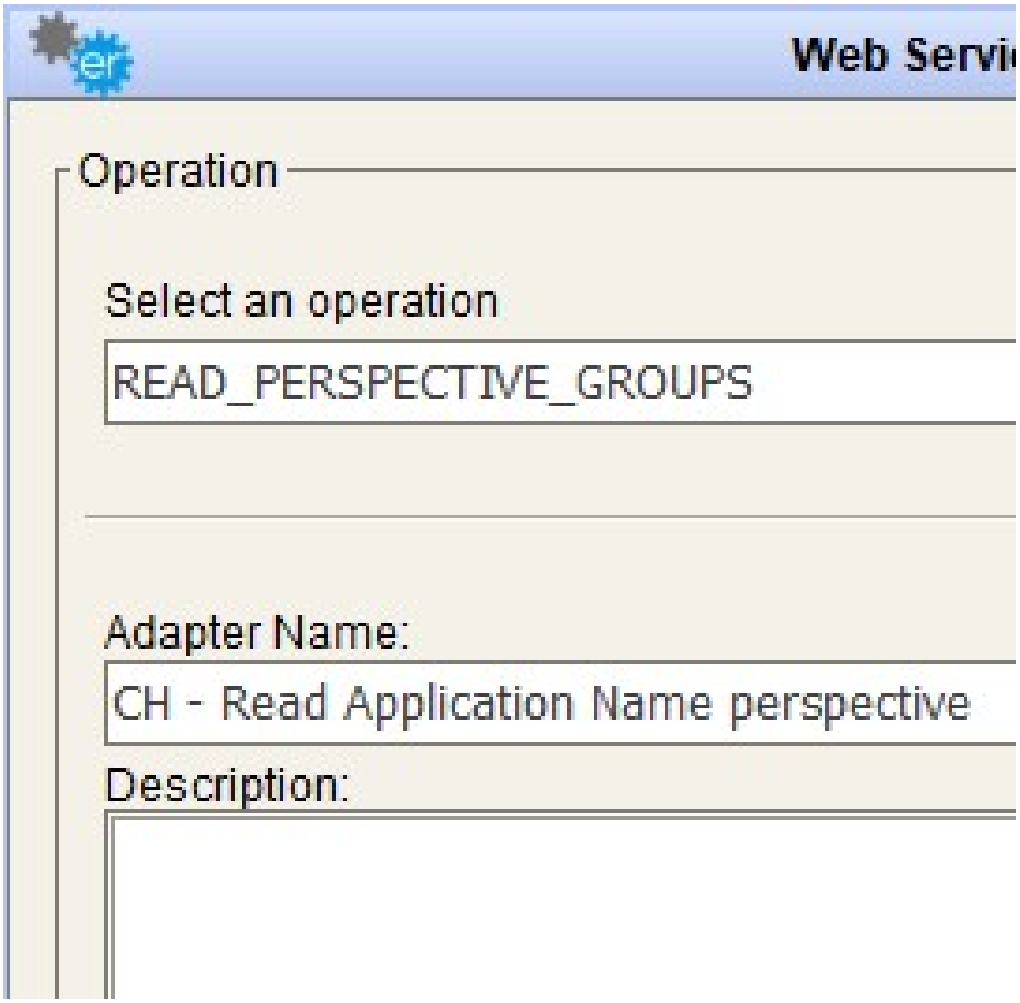
1.2.17 CloudHealth – READ PERSPECTIVE GROUPS

To use the adapter for CloudHealth**, to read perspective groups**, fill the field “Web Service Home URL” with the CloudHealth URL you want to connect to. Then, provide a valid username/password to connect, or the API credentials (API Key), that can be generated by a valid account. Please, make sure that the account has the correct permissions to use CloudHealth API: http://apidocs.cloudhealthtech.com/#documentation_getting-your-api-key





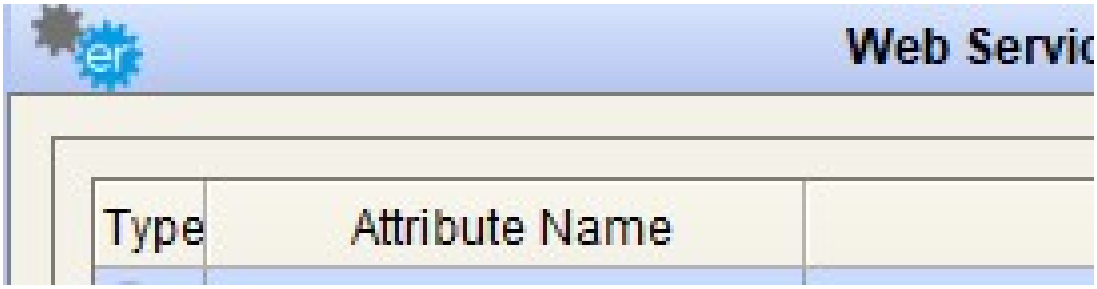
The  button steps into the next window that shows the available operations of this Web Service; choose “**READ PERSPECTIVE GROUPS**”, to get the list of groups in the desired perspective.



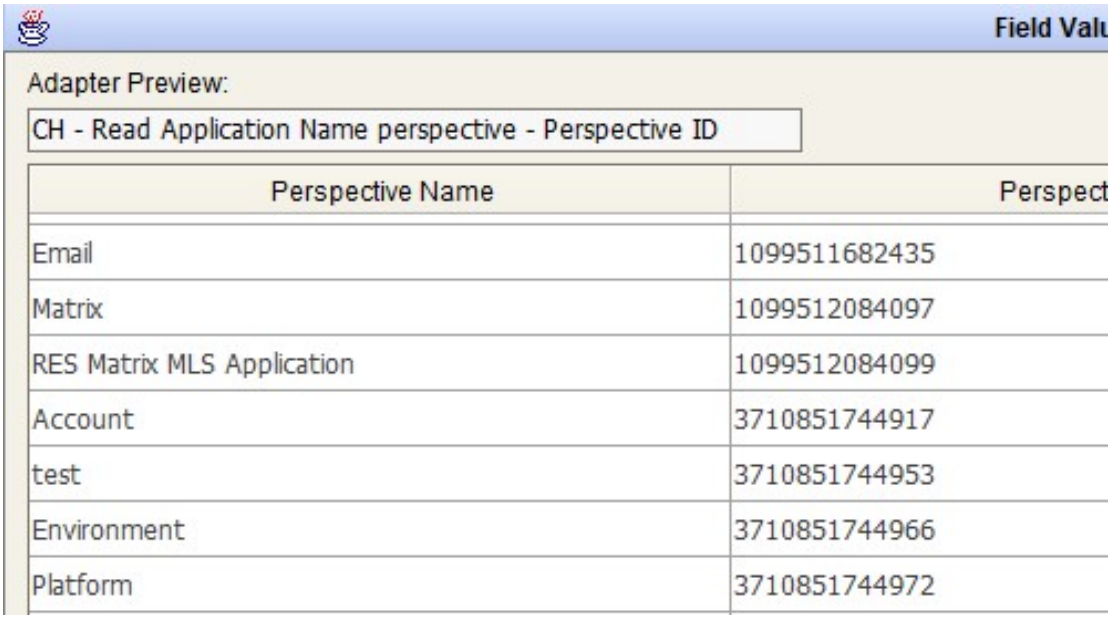
After selecting the desired operation, it is required to enter a name and a description for the adapter into the appropriate fields; the user must browse the file system for the adapter folder (used to save intermediate and result file for adapter execution).

The next button Next steps into the next window that allows the user to set values for the Web Service parameters.

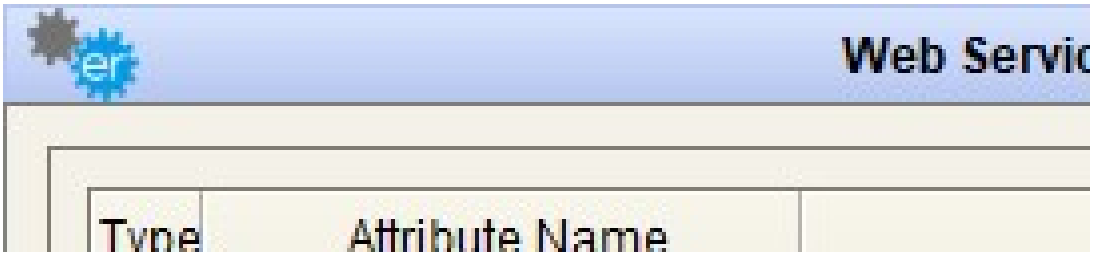
When a lens icon shows beside the parameter name:



this means that user may choose the parameter value from a list of available items. In case of this example, with the double-click on the lens icon, a table appears, with the available perspectives:



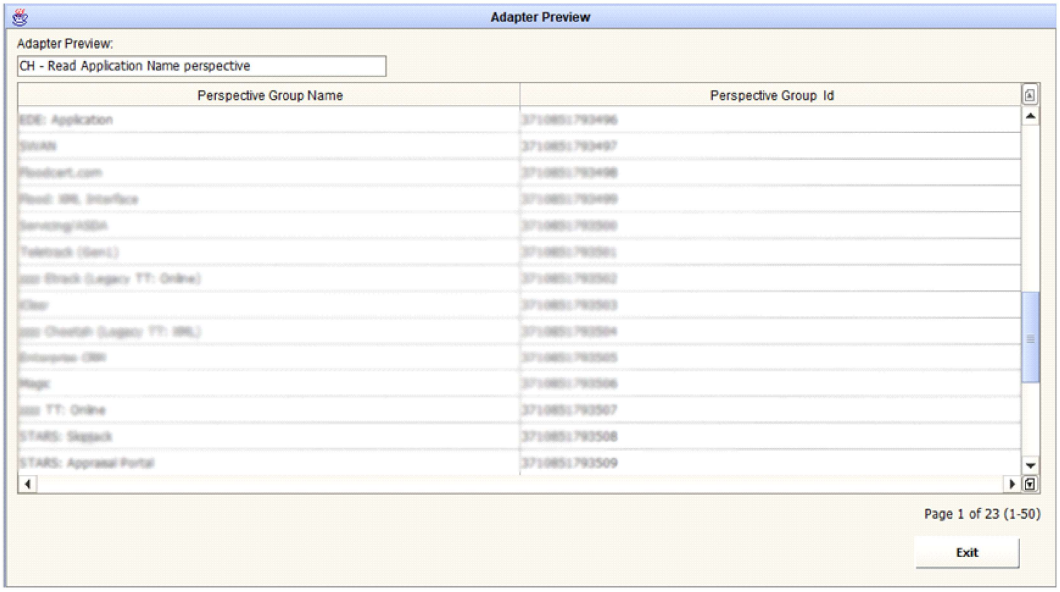
Select and apply the selection:



Pressing the button opens a new pop-up window and the response resulting from the Web Service call is displayed, with the name and ref_id of existing groups. These values can be stored in EA model object to manage groups creation and update, based on CH identifier (see after).

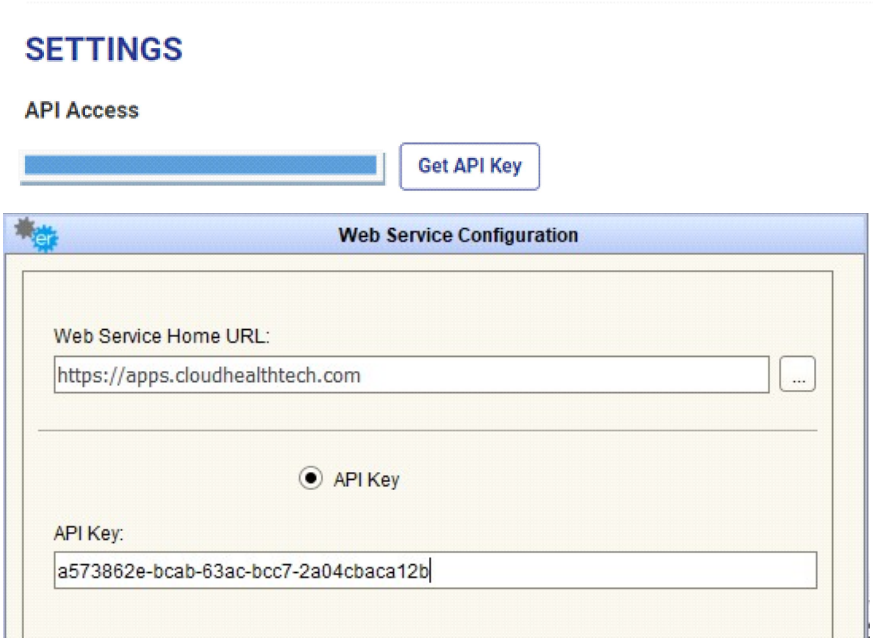
Pressing the button will store all the entered parameters for the web service adapter in the DT database.

The button closes the data preview window and goes back to the Web Service configuration form.

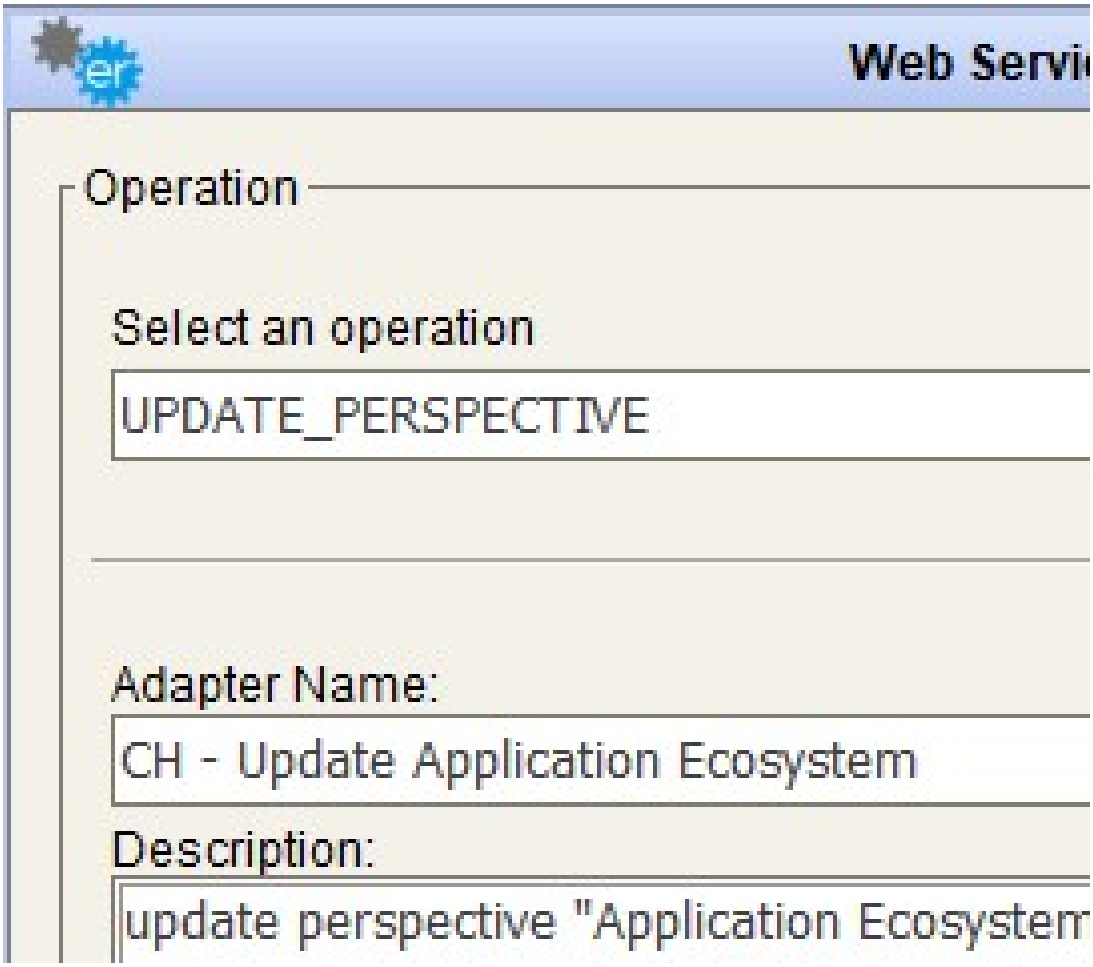


1.2.18 CloudHealth – UPDATE PERSPECTIVE

To use the adapter for CloudHealth**, to update perspective**, fill the field “Web Service Home URL” with the CloudHealth URL you want to connect to. Then, provide a valid username/password to connect, or the API credentials (API Key), that can be generated by a valid account. Please, make sure that the account has the correct permissions to use CloudHealth API: http://apidocs.cloudhealthtech.com/#documentation_getting-your-api-key

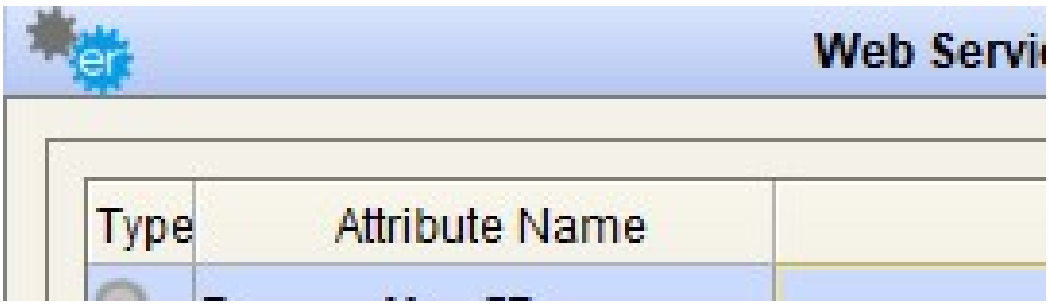


The “Next” button steps into the next window that shows the available operations of this Web Service; choose “UPDATE PERSPECTIVE”, to push groups to the desired perspective in CloudHealth.



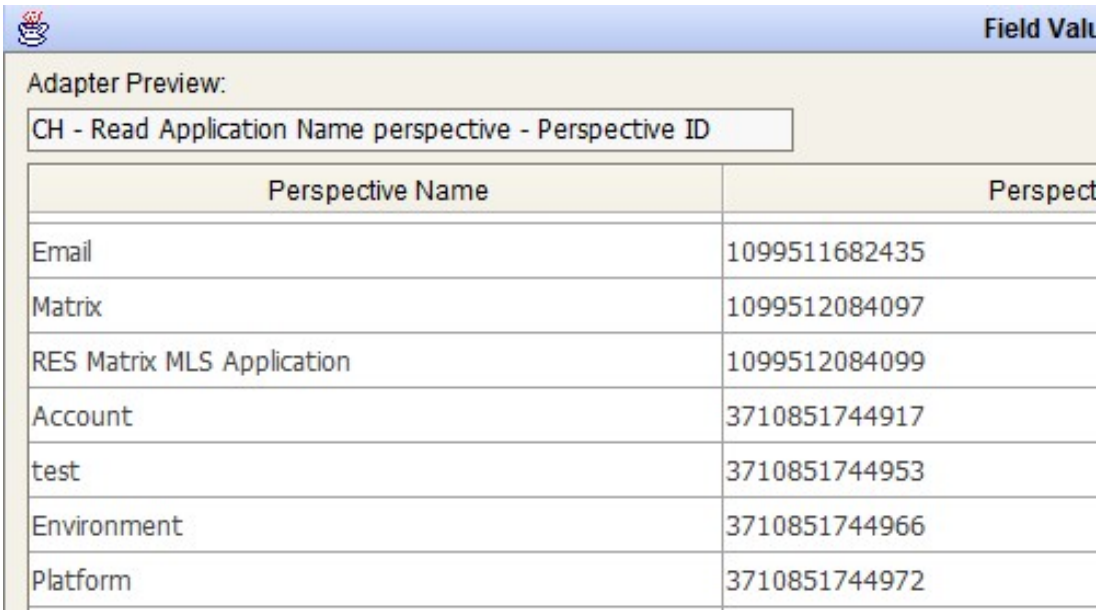
After selecting the desired operation, it is required to enter a name and a description for the adapter into the appropriate fields; the user must browse the file system for the adapter folder (used to save intermediate and result file for adapter execution).

The next button steps into the next window that allows the user to set values for the Web Service parameters.

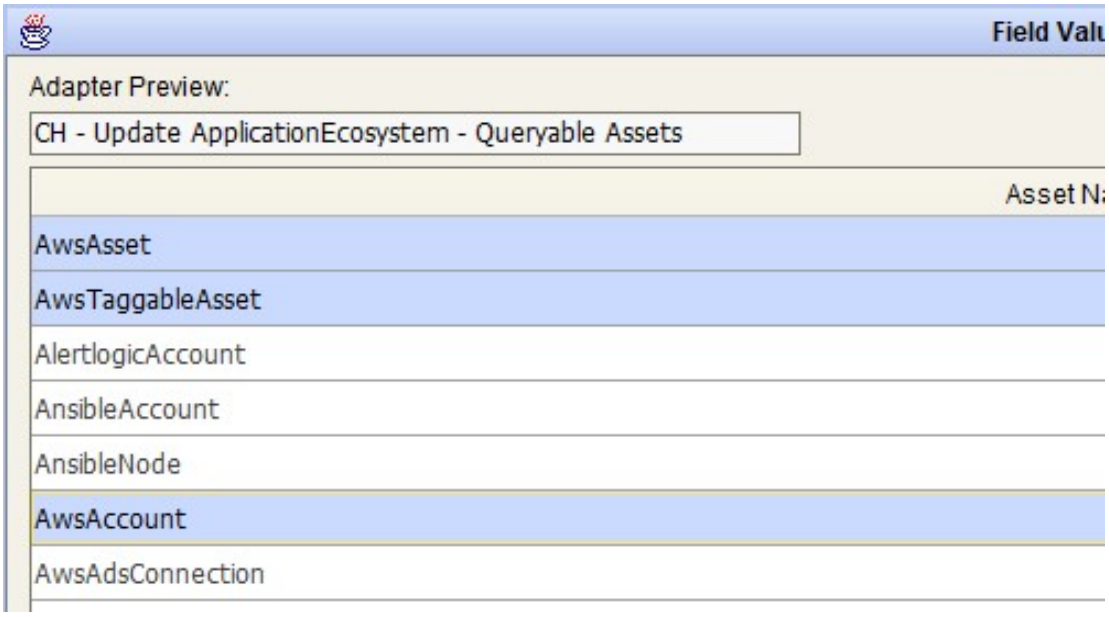


When a lens icon shows beside the parameter name, this means that user may choose the parameter value from a list of available items.

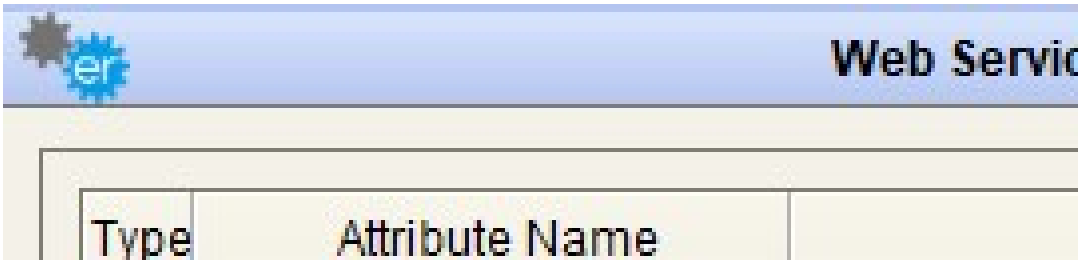
In case of this example, with the double-click on the lens icon, a table appears, with the available perspectives:



and Queryable assets:



Select and apply the selection, even using multiselection in case of queryable assets:

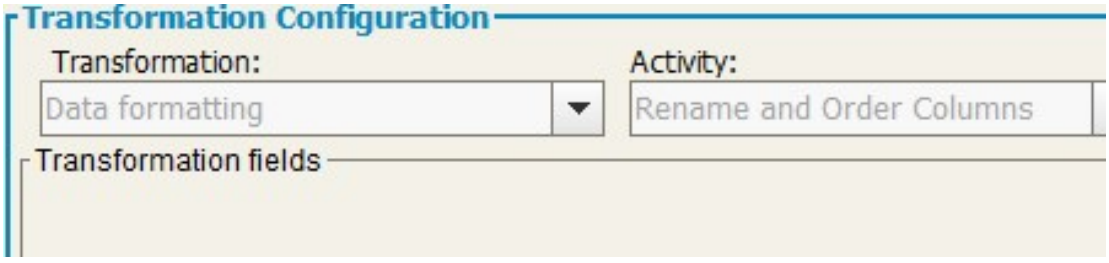


Pressing the “Save” button will store all the entered parameters for the web service adapter in the DT database.

The “Exit" button closes the data preview window and goes back to the Web Service configuration form.

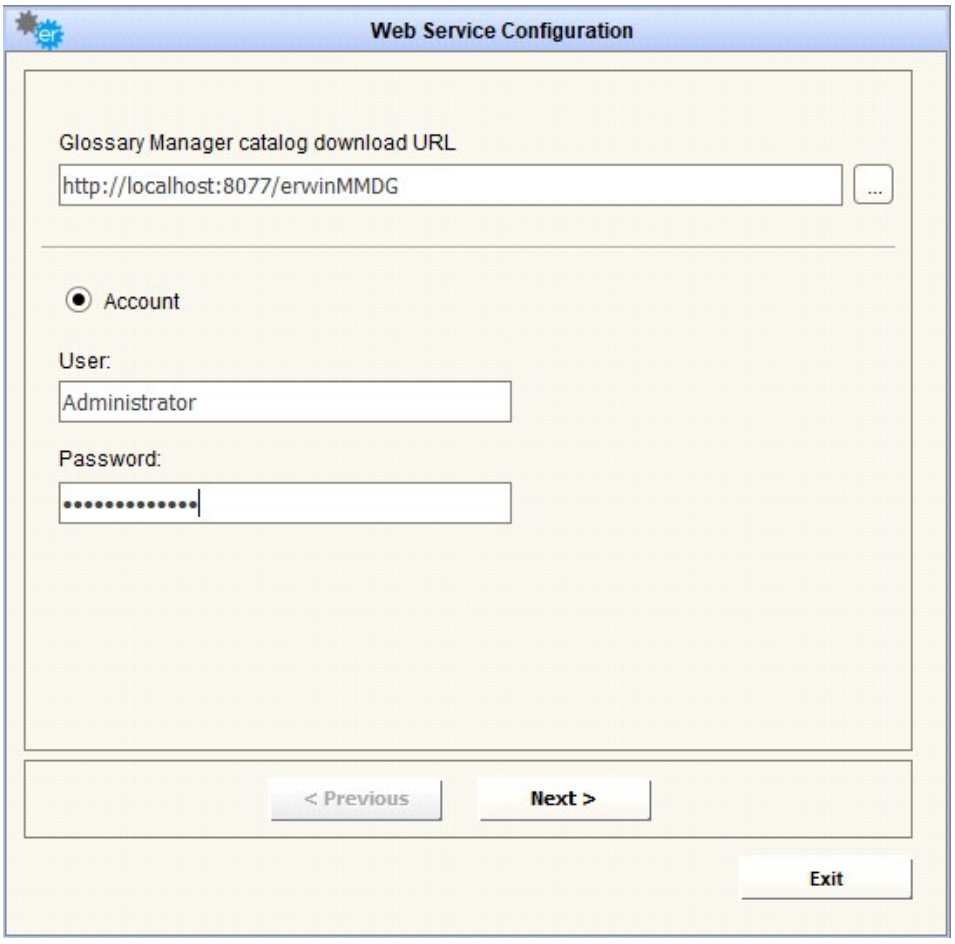
The Adapter, as mentioned, can be used to push data to CloudHealth and fill perspective groups. Based on an asset tag (for instance, application ID), corresponding metadata associated with the application ID will be pushed to CloudHealth in the form of perspectives and groups.

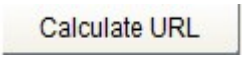
As an example, the perspective listing an Application Ecosystem and their associated applications can be aligned configuring a workflow, with a source providing values for application ecosystems name and associated application ID, and with the target operation “Web Service POST”, using the configured adapter. To provide mappings between source and target field, use a “Data Formatting: Rename and Order Columns” transformation, with the mandatory “**Name**” field for the perspective groups, the optional “**ref_id**” field to manage object renaming and additional fields name corresponding to tag fields to populate:



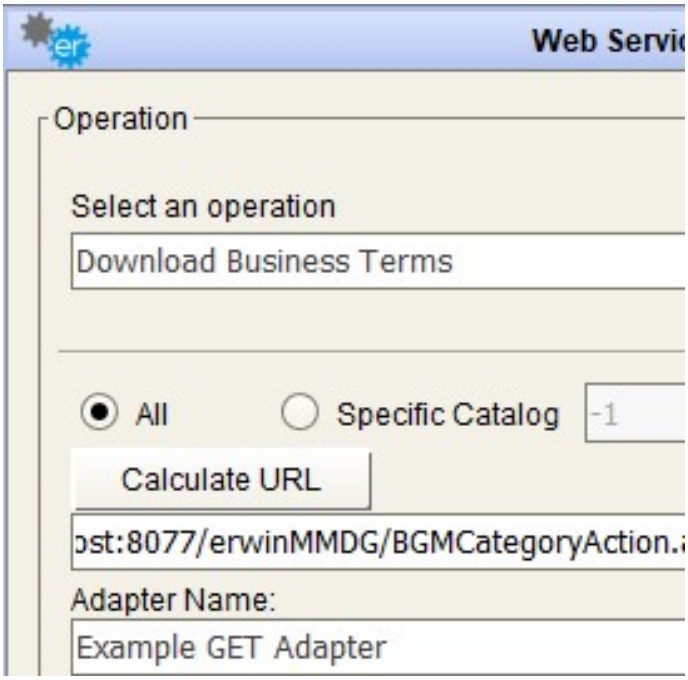
1.2.19 Business Terms from BGM – Download Business Terms

To use the adapter for BGM **to get Business Terms**, select the authorization method (Account / Access Token) and enter the relevant credentials.

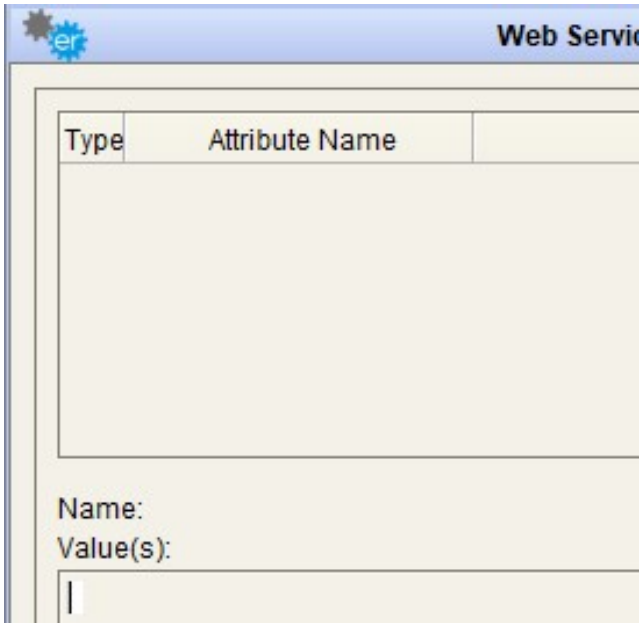


Click the  button to proceed to the next step, and select the operation “Download Business Terms”. Specify the specific Catalog ID or select “All” and click the  button. Enter a name and folder for the adapter. A description can also optionally be entered for the adapter.

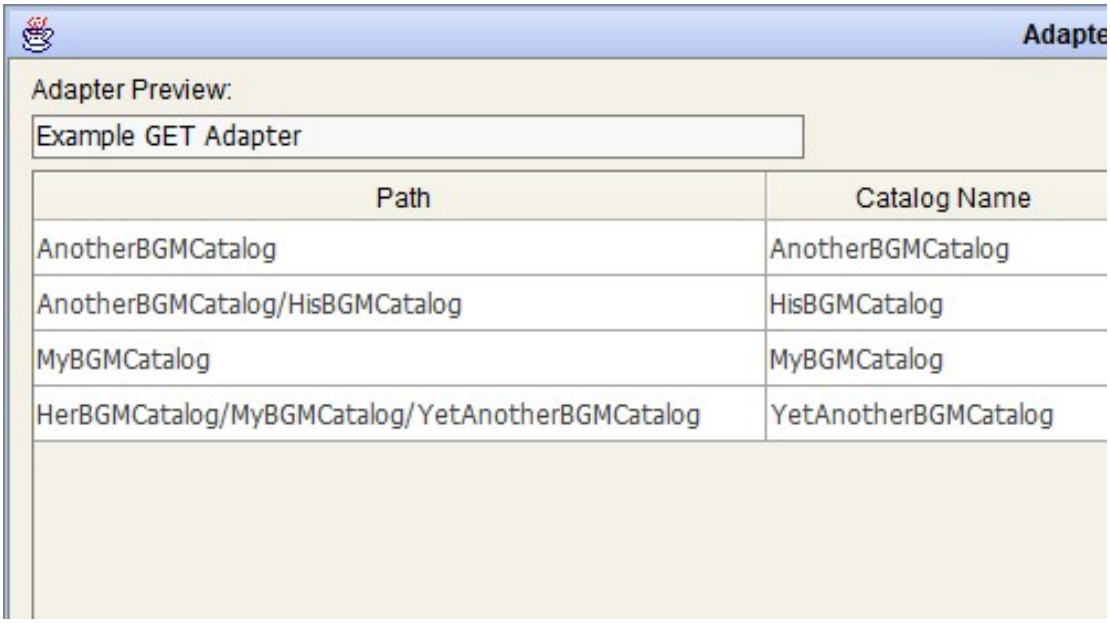
Click the Next > button to proceed to the next step. To save the adapter to the DT database, click the button.



Click the Next > button to proceed to the next step, where the adapter can be previewed and/or saved.



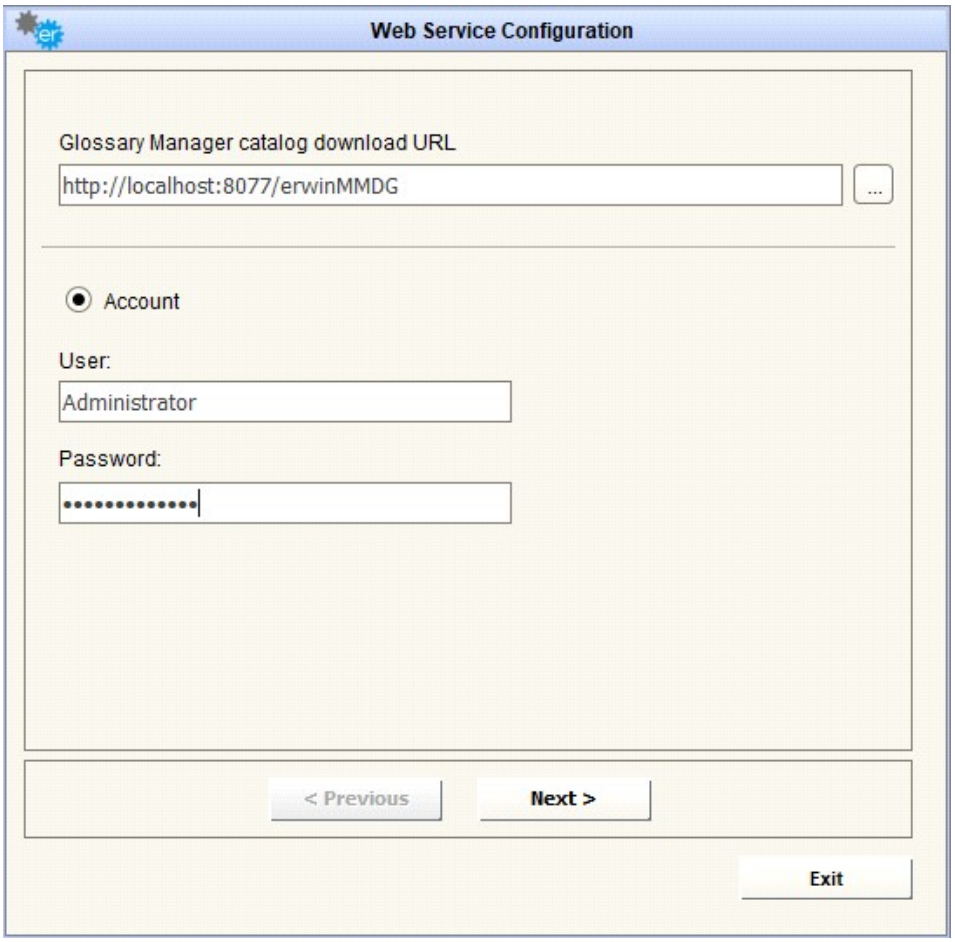
To preview a sample of the first 50 results, click the button.

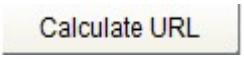


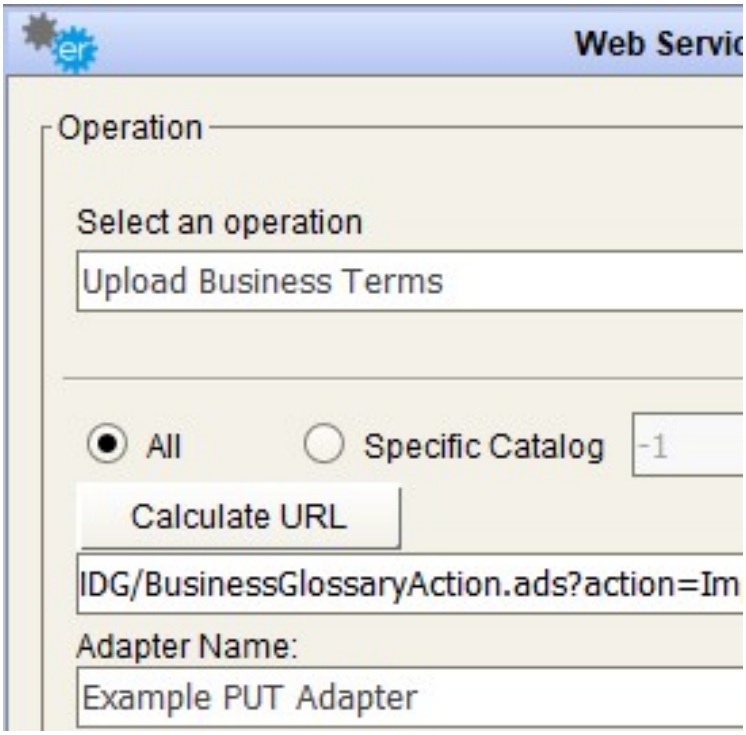
To save the adapter to the DT database, click the button.

1.2.20 Business Terms from BGM – Upload Business Terms

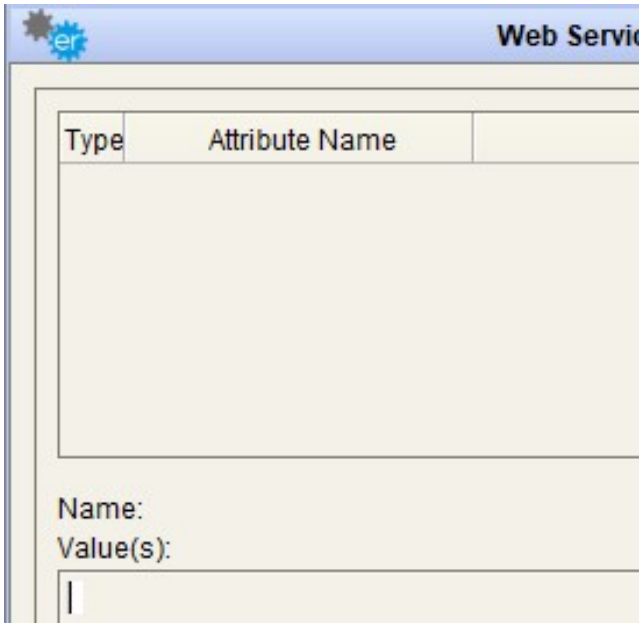
To use the adapter for BGM **to get Business Terms**, select the authorization method (Account / Access Token) and enter the relevant credentials.



Click the  button to proceed to the next step, and select the operation “Upload Business Terms”. Specify the a specific Catalog ID or select “All” and click the  button. Enter a name and folder for the adapter. A description can also optionally be entered for the adapter.

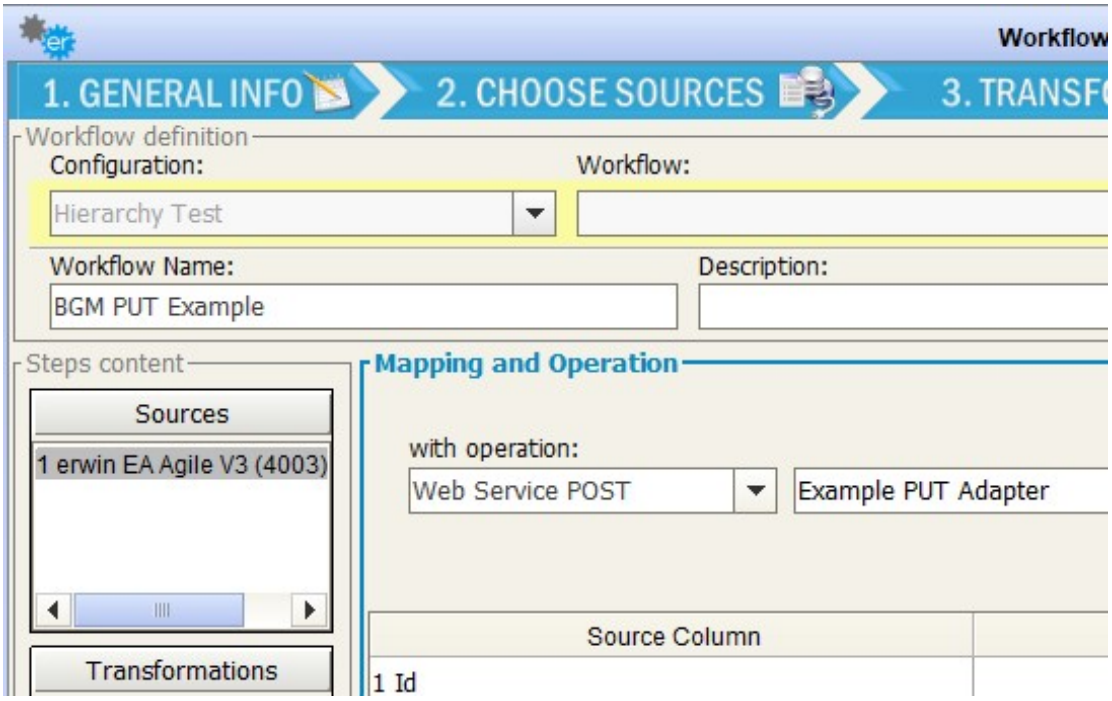


Click the Next > button to proceed to the next step. To save the adapter to the DT database, click the button.



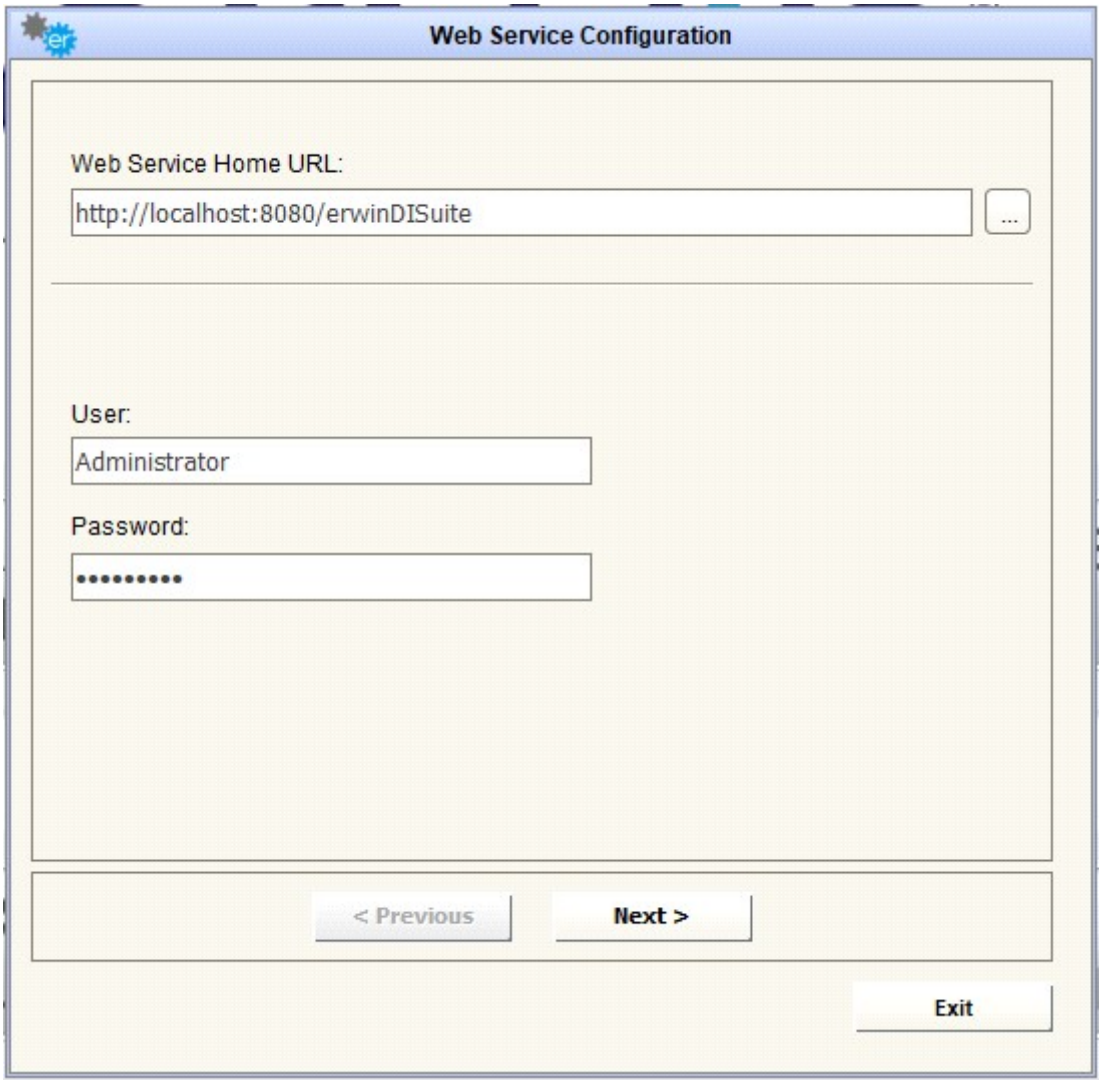
The adapter can now be used as a target endpoint when mapping data in a workflow. To select the adapter as a target, select “Web Service POST” as the operation and select your BGM adapter. A mapping

table will appear, allowing the source fields to be mapped to relevant target fields for creating BGM Business Terms.



1.2.21 erwin DISuite Metadata Manager

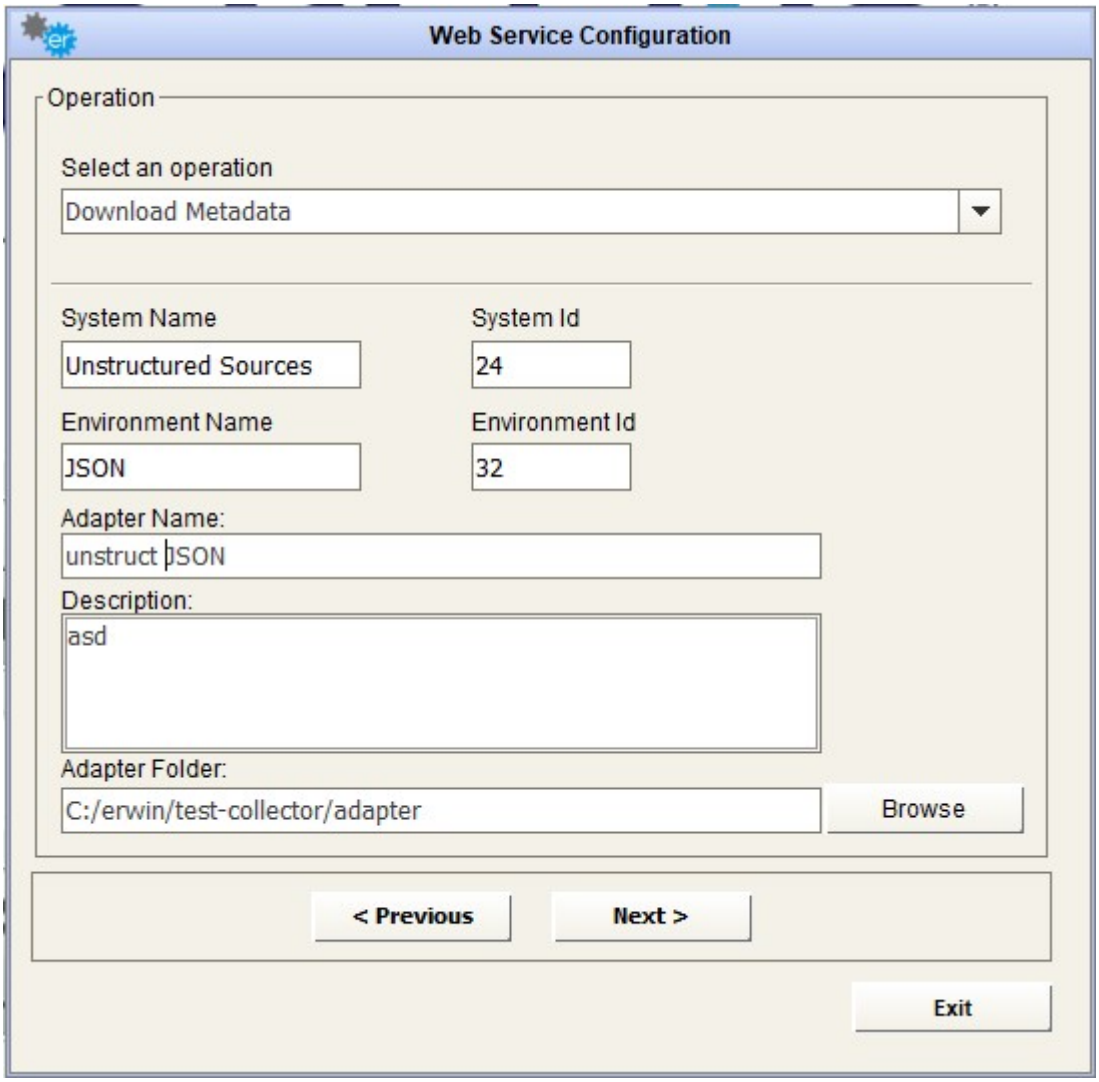
To use the adapter for erwin DISuite Metadata Manager **to download Data Dictionary**, provide the url of erwin DISuite and enter the relevant credentials



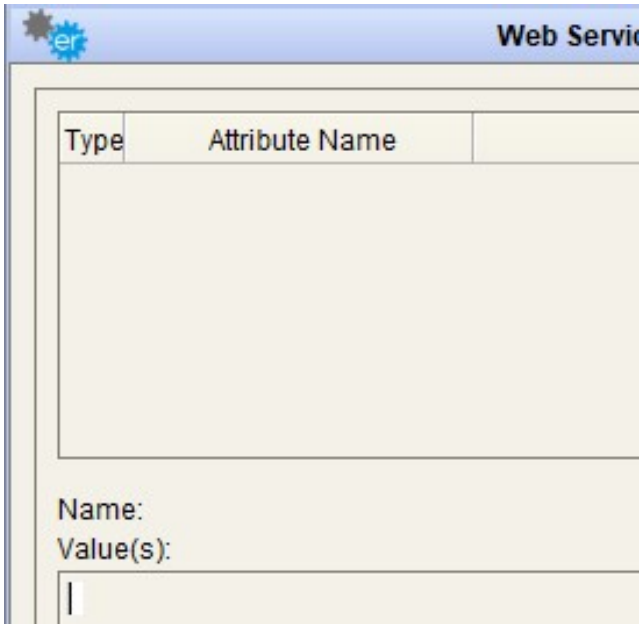
Click the Next > button to proceed to the next step, and select the operation “Download Metadata”.

Provide the “System Name”, “System Id”, “Environment Name”, “Environment Id” in their respective edit box.

Enter a name and folder for the adapter. A description can also optionally be entered for the adapter.



Click the Next > button to proceed to the next step, where the adapter can be previewed and/or saved.

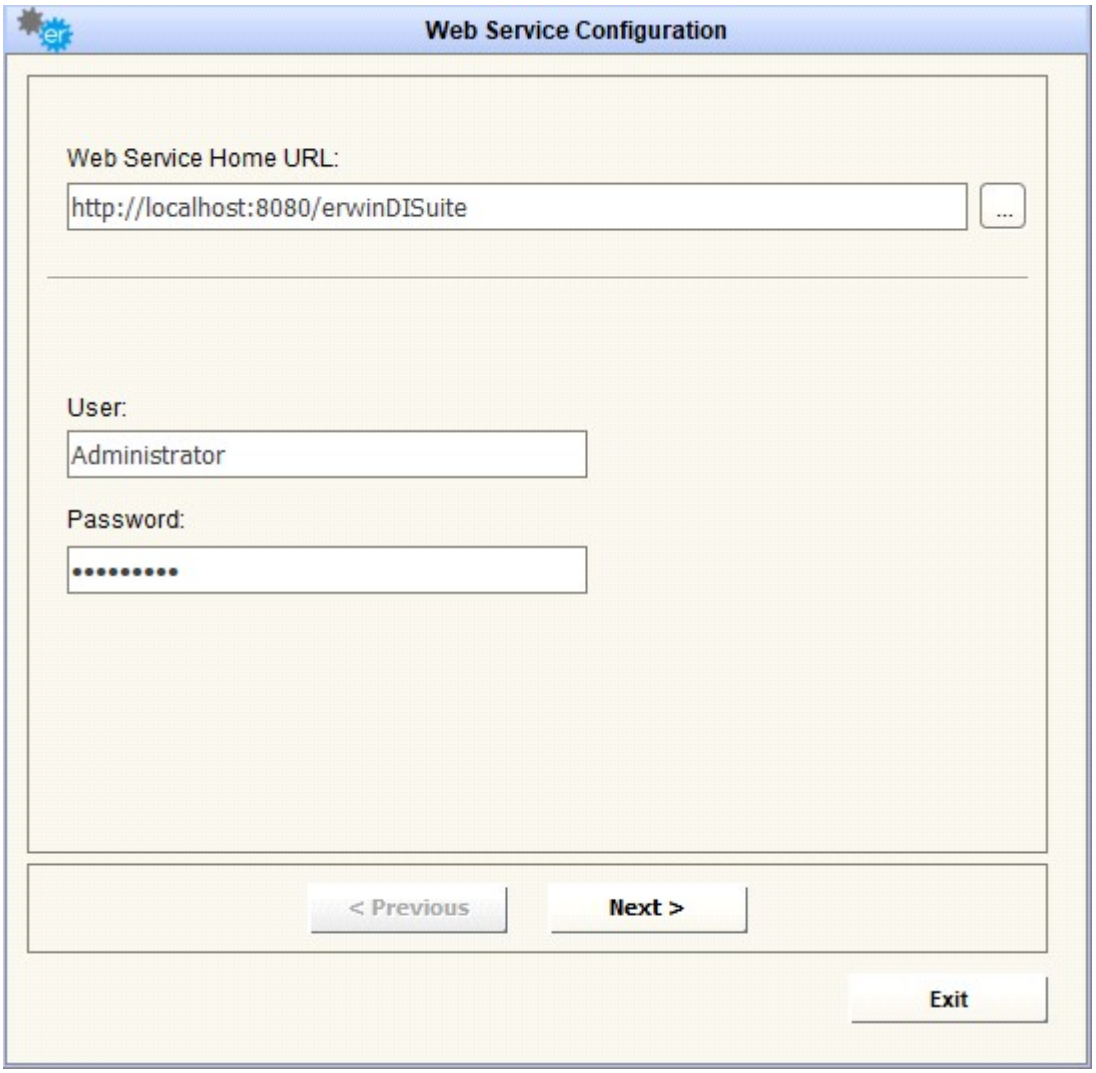


To preview a sample of the first 50 results, click the button

To save the adapter to the DT database, click the button.

1.2.22 erwin DISuite Mapping Manager

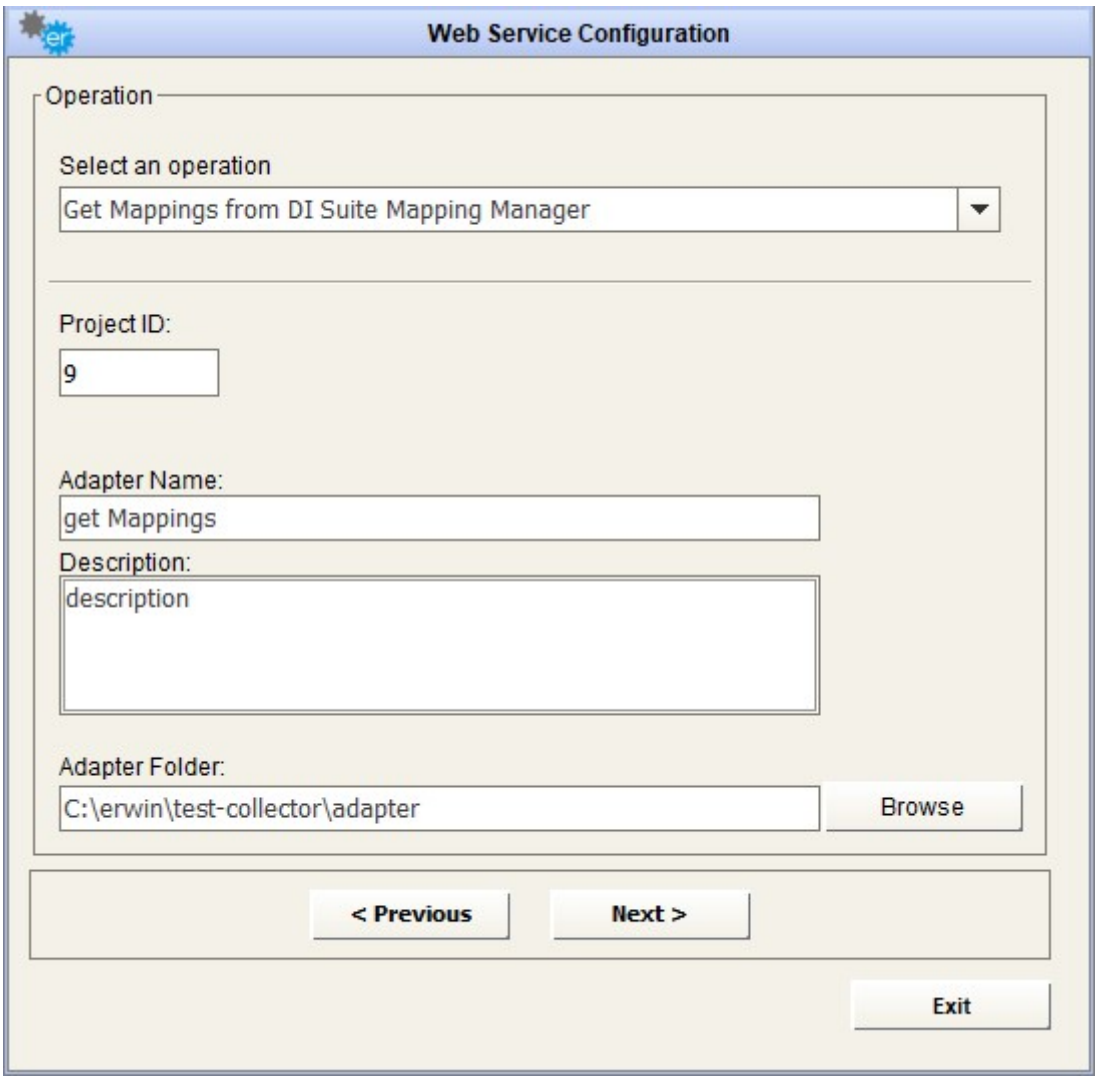
To use the adapter for erwin DISuite Mapping Manager **to download Mappings**, provide the url of erwin DISuite and enter the relevant credentials



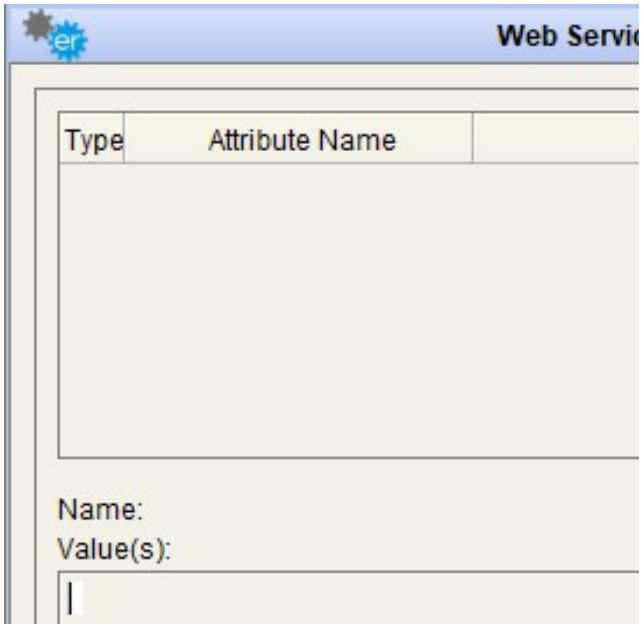
Click the Next > button to proceed to the next step, and select the operation “Get Mappings from DI Suite Mapping Manager”.

Provide the “Project Id” of the project in Mapping Manager.

Enter a name and folder for the adapter. A description can also optionally be entered for the adapter.



Click the Next > button to proceed to the next step, where the adapter can be previewed and/or saved.

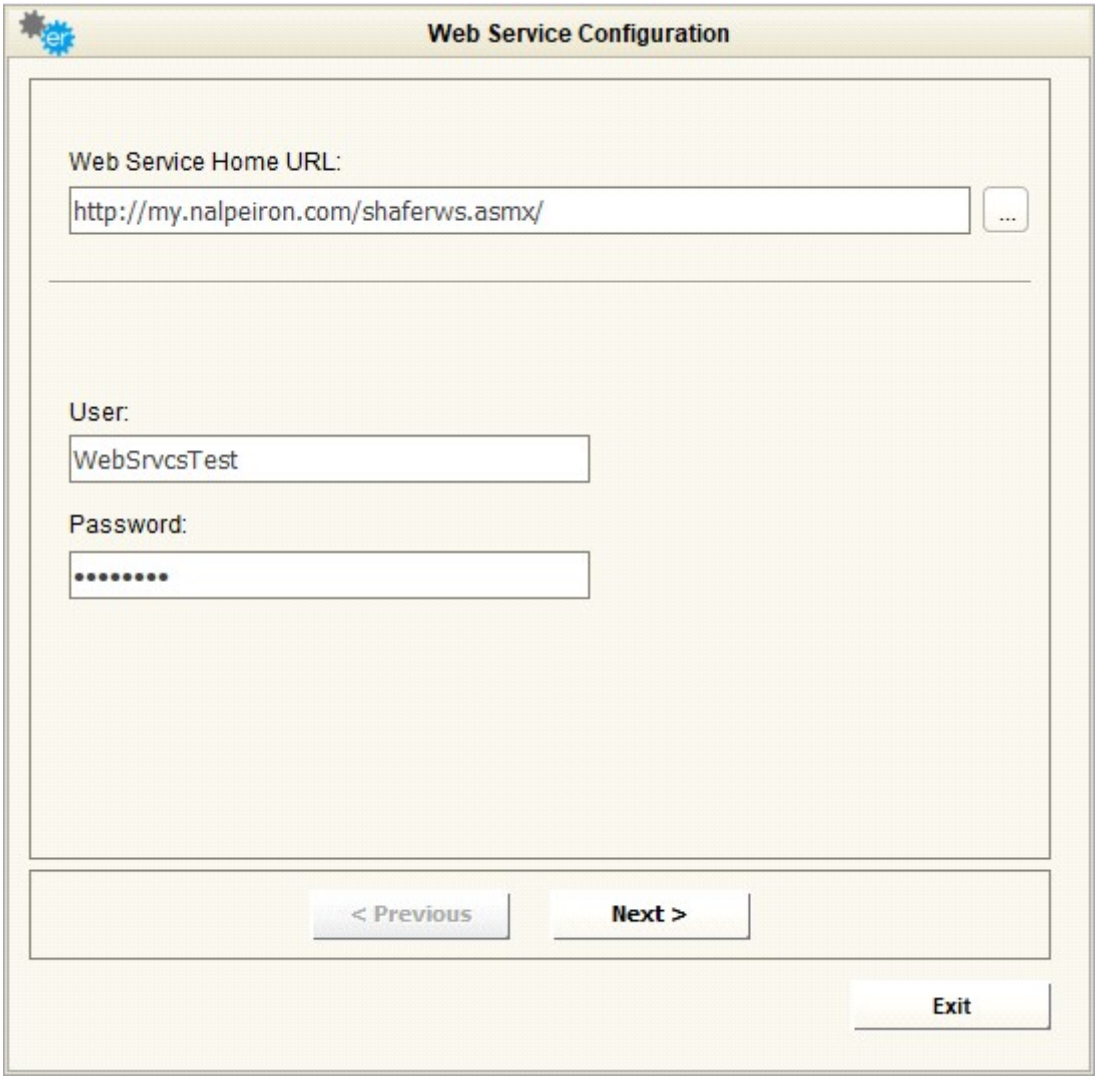


To preview a sample of the first 50 results, click the button

To save the adapter to the DT database, click the button.

1.2.23 Nalpeiron – get license details for company

To use the adapter for Nalpeiron **to download license details from Nalpeiron** , provide the url to access Nalpeiron webservice and enter the relevant credentials

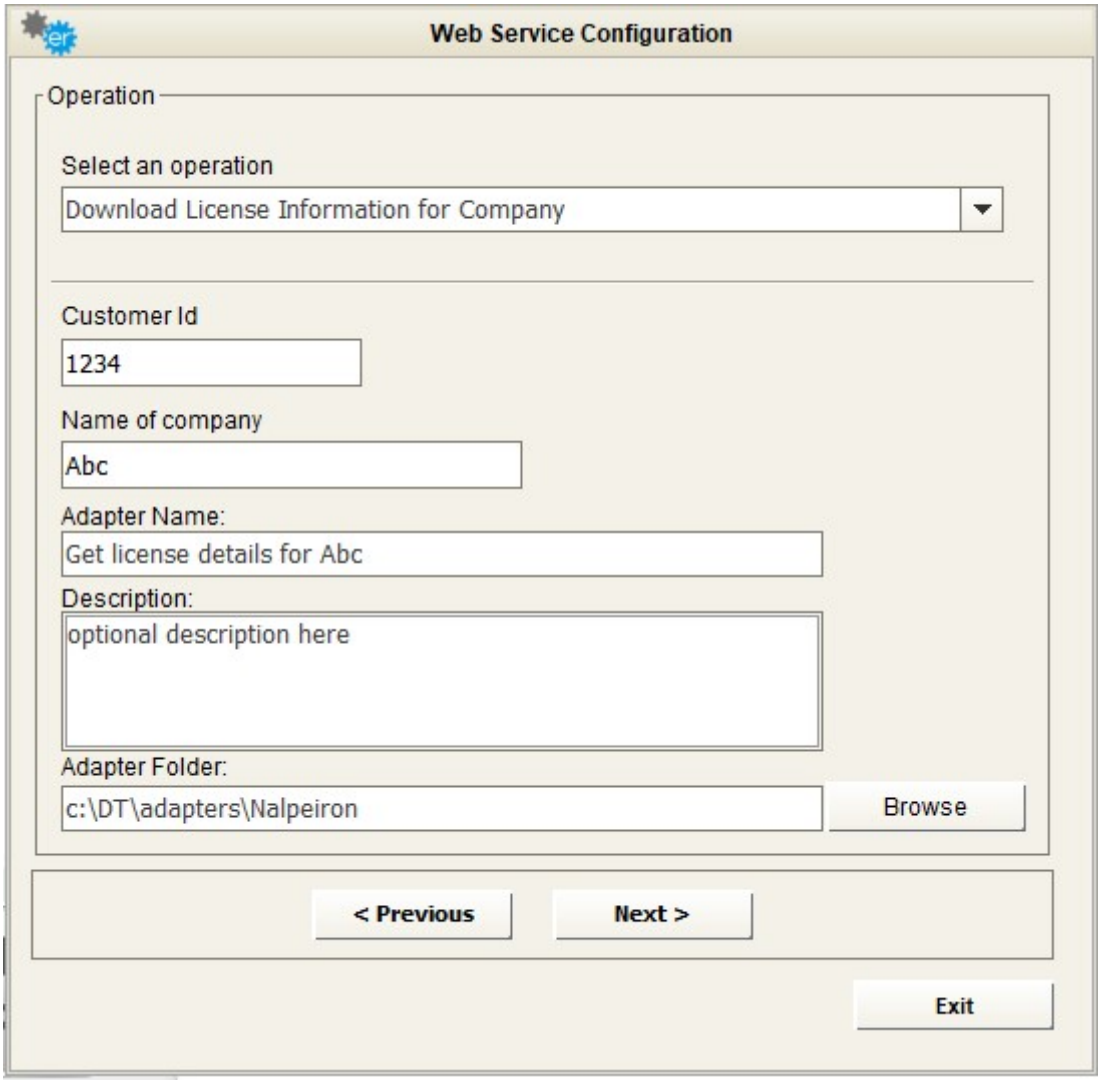


Click the Next > button to proceed to the next step, and select the operation “Download License information for Company”.

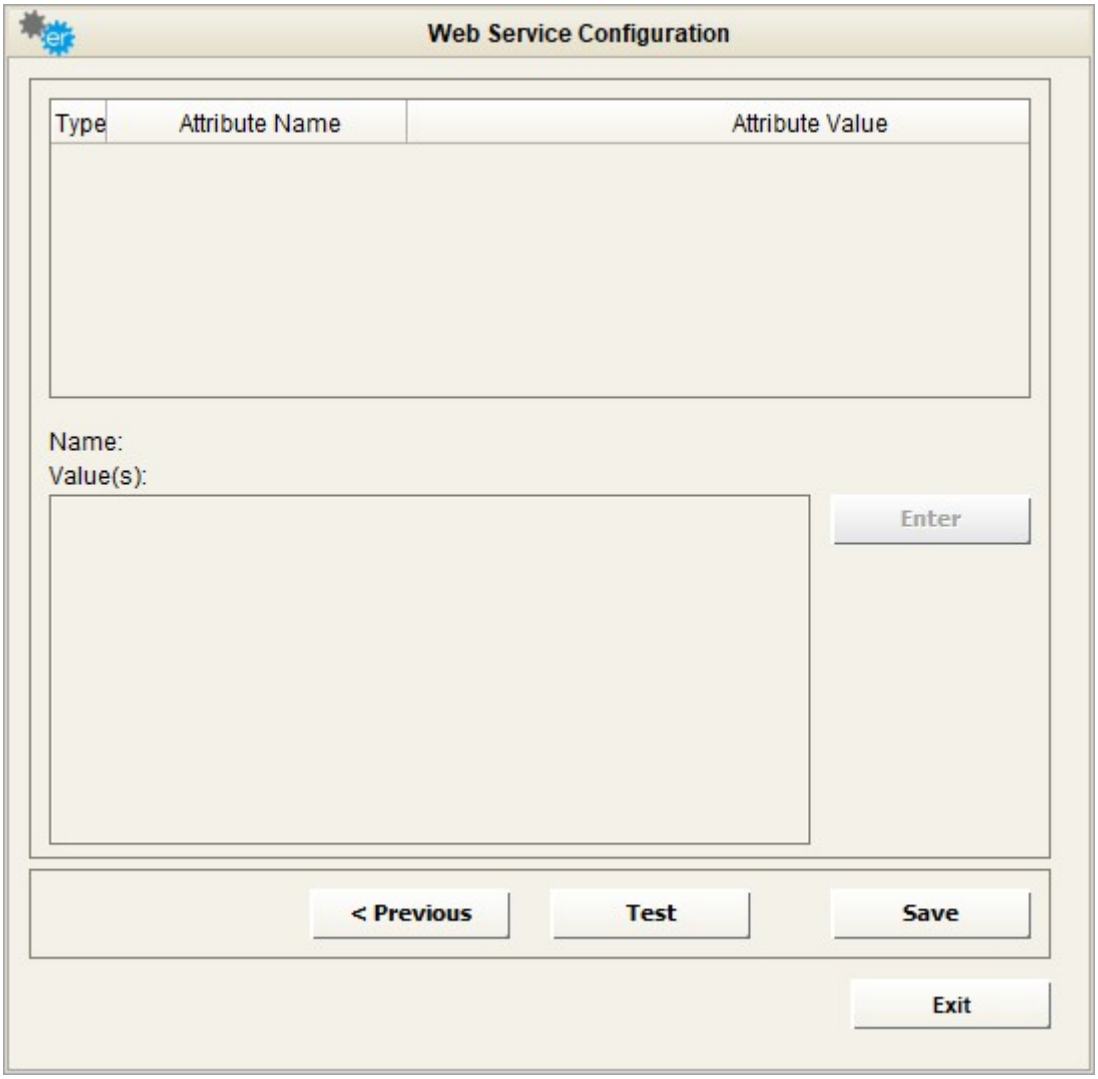
Provide the “Customer Id” that Nalpeiron has assigned to your company.

Provide the name of company (your customer) for whom you want to get license details from Nalpeiron

Enter a name and folder for the adapter. A description can also optionally be entered for the adapter.



Click the Next > button to proceed to the next step, where the adapter can be previewed and/or saved.

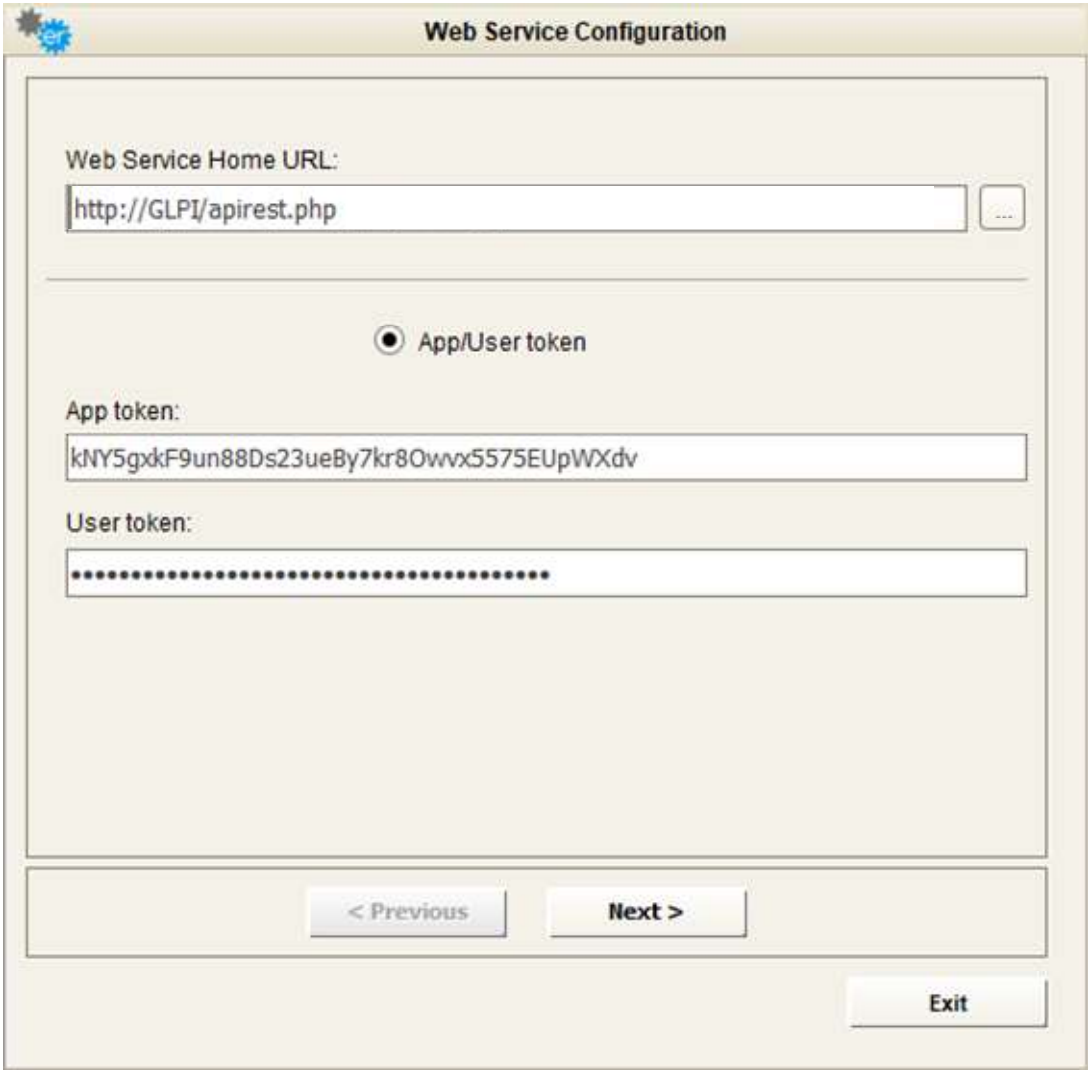


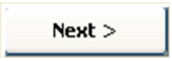
To preview a sample of the first 50 results, click the button

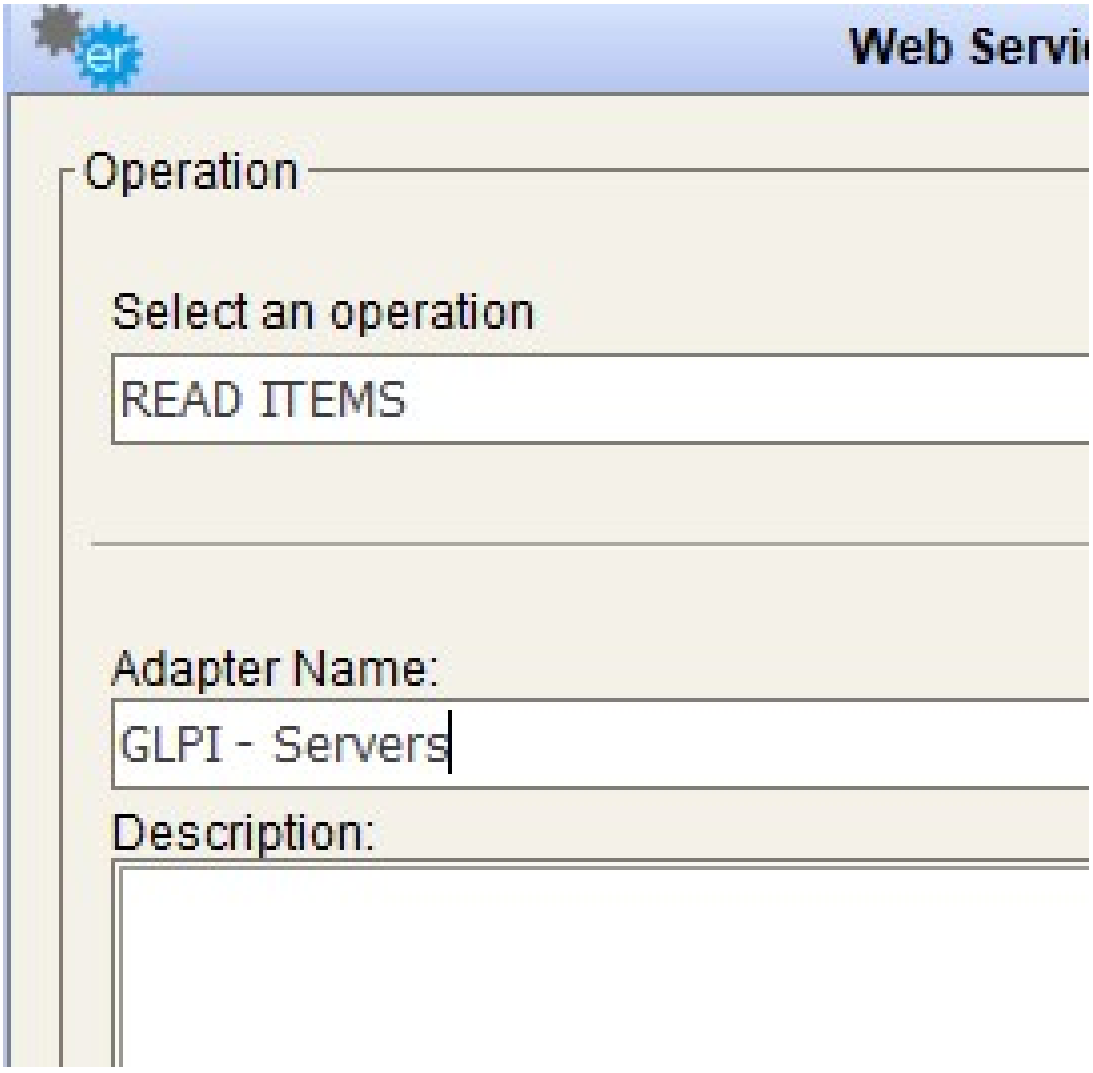
To save the adapter to the DT database, click the button.

1.2.24 GLPI – Read configuration items

To use the adapter for **GLPI**, fill the field “Web Service Home URL” with the GLPI home page URL you want to connect to. Then provide the **APP token** and **user token** to connect.



The  button steps into the next window that shows the available operation of that Web Service, which is “**READ ITEMS**”, allowing users to query all items in GLPI repository, being them computer, software or other item types.

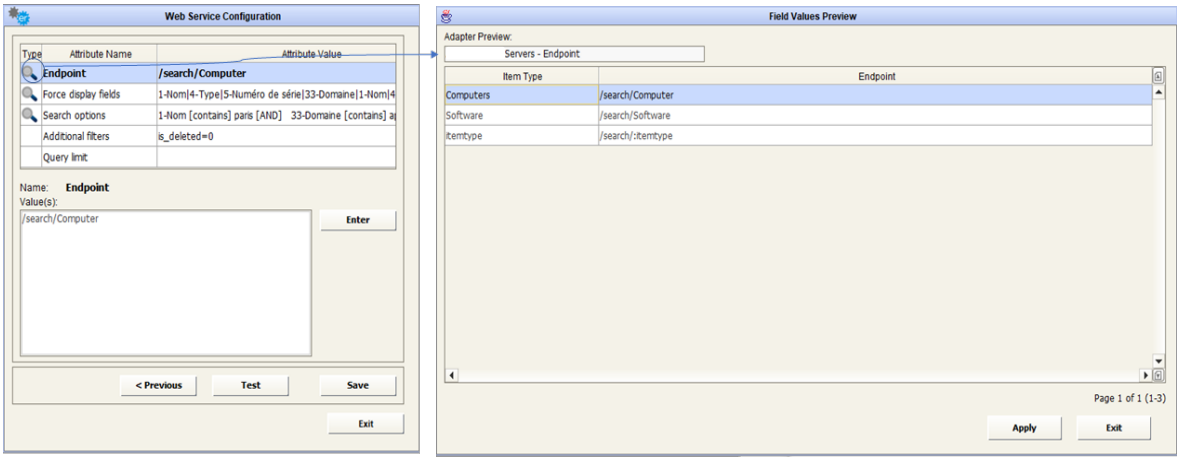


After selecting the desired operation, it is required to enter a name and a description for the adapter in the appropriate fields; the user must browse the file system for the adapter folder (used to save intermediate and result file for adapter execution).

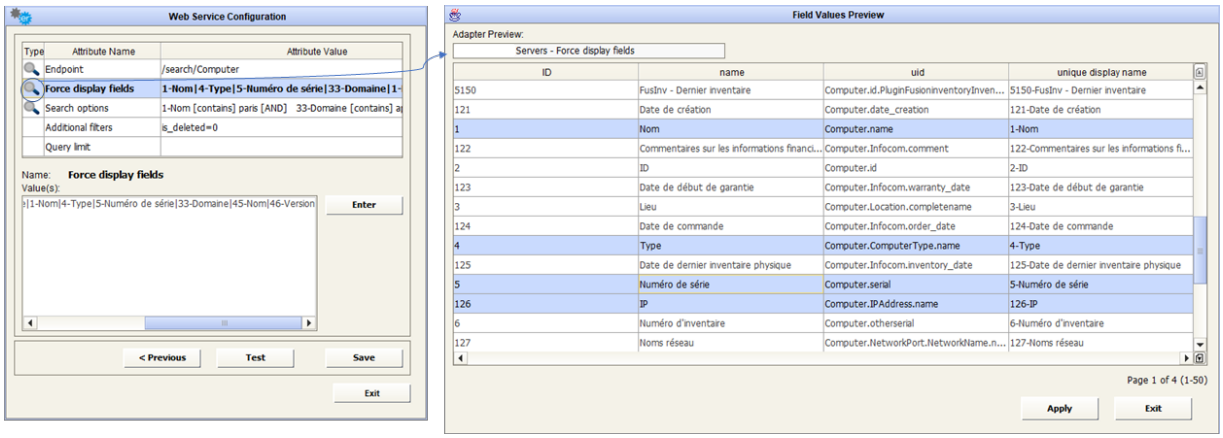
The next button Next steps into the next window that allows the user to set values for the Web Service parameters.

After that, the user is provided with a set of options, to select endpoint, display fields, search options and additional filters:

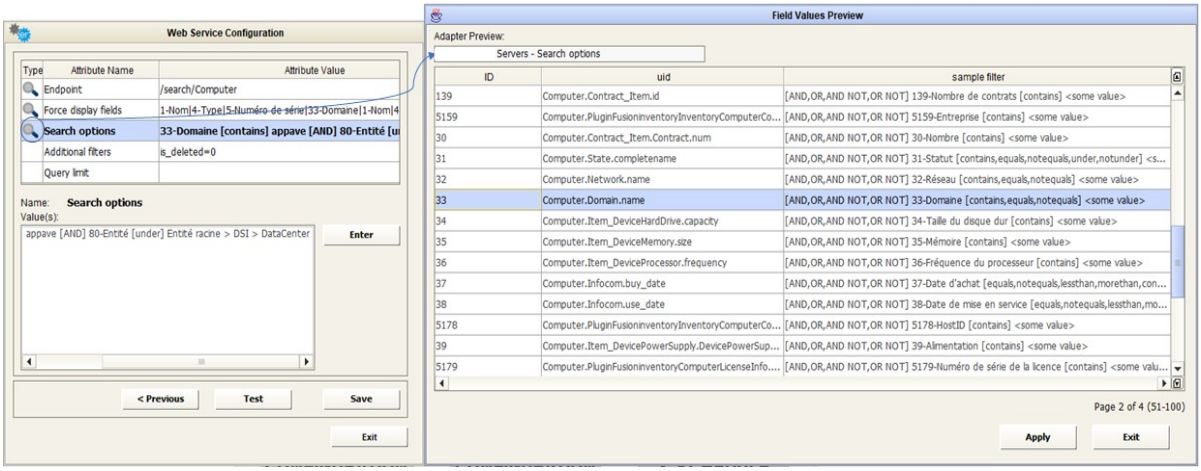
1) select between different endpoint, like Computer, Server, or any other item type:



2) select the fields of interest



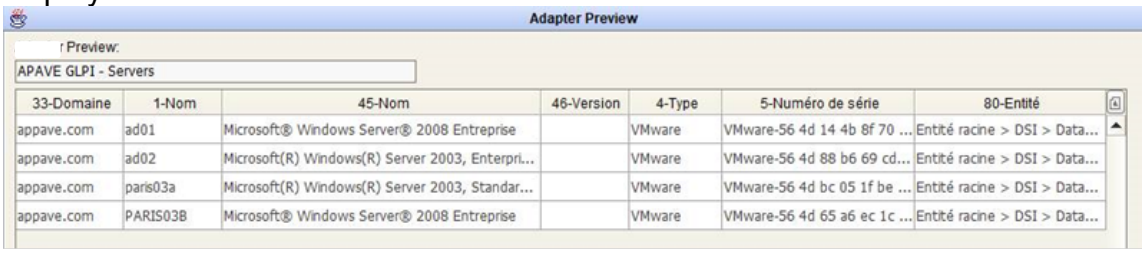
3) Get available search options (search criteria):



4) define additional filters, i.e:. “*is_deleted=0*” that will be appended to previous parameters. To add “metacriteria” to complete the search options, defined in point 3), append the metacriteria search options using the GLPI syntax here:
“is_deleted=0*&metacriteria[0][itemtype]=Computer&metacriteria[0][field]=2&metacriteria[0][searchtype]=equal&metacriteria[0][value]=>0*”

5) define a query limit (max returned range)

Pressing the button will open a new pop-up window. The response resulting from the Web Service call is displayed.

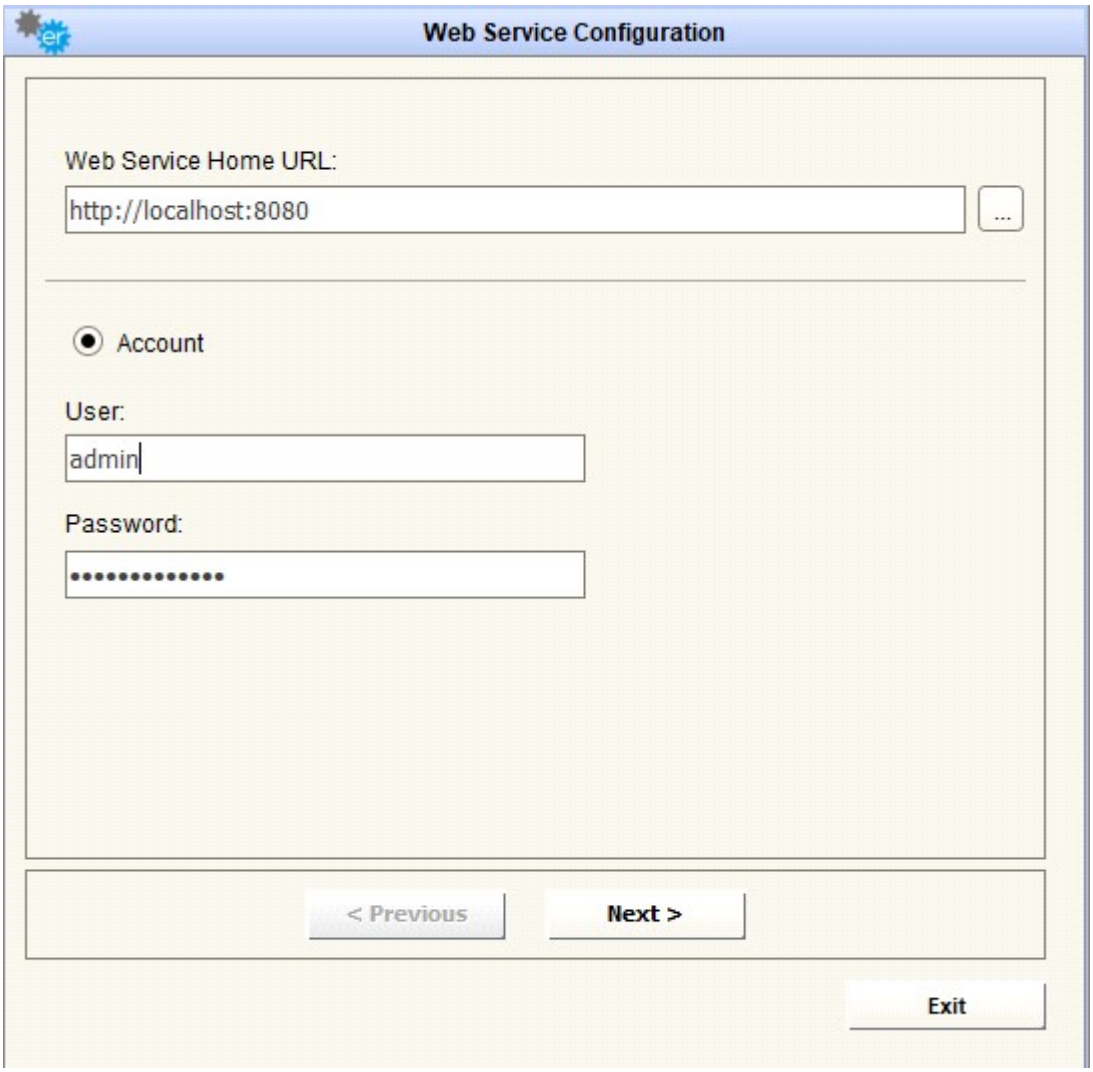


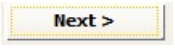
Pressing the button will store all the entered parameters for the web service adapter in the DT database.

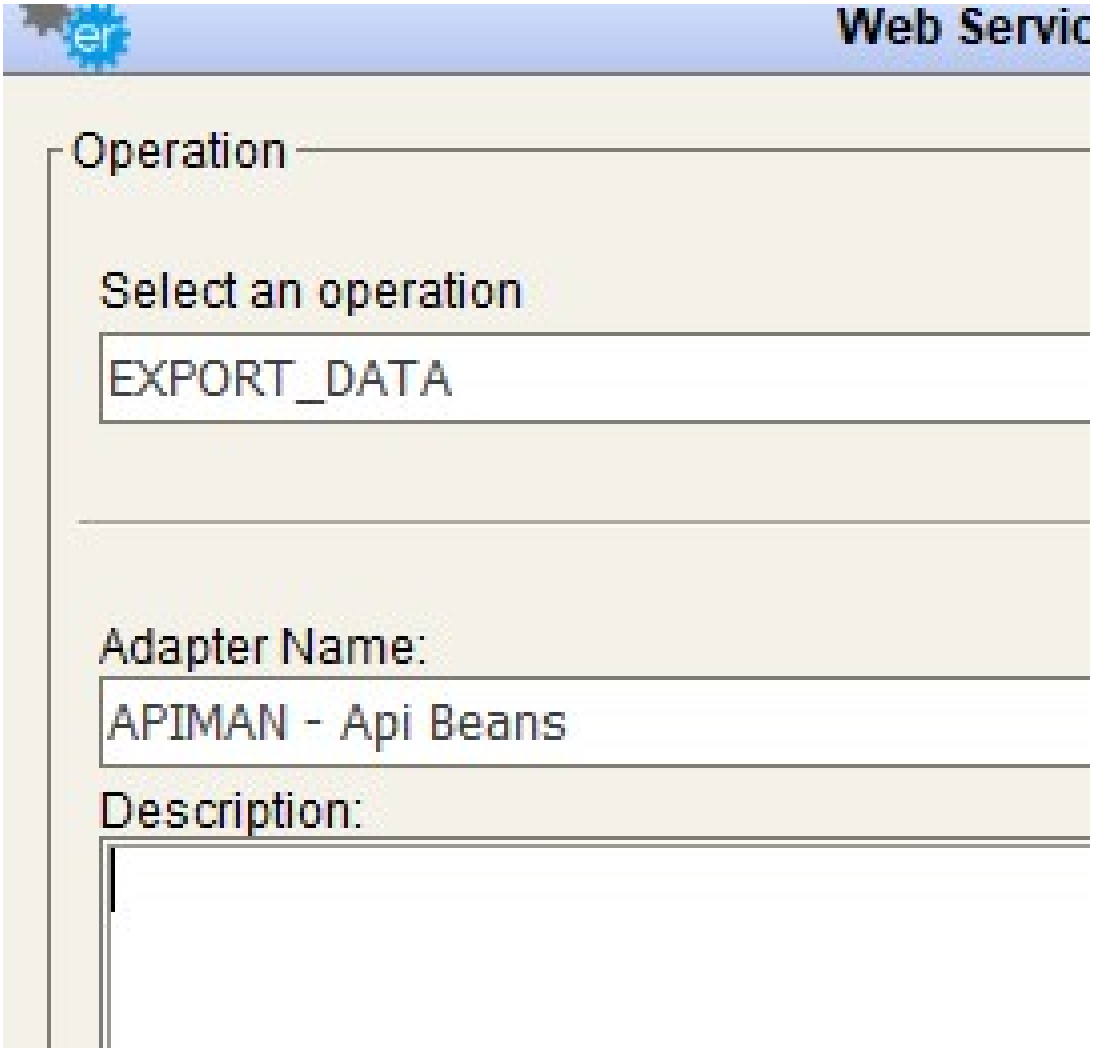
The Exit button closes the data preview window and goes back to Web Service configuration form.

1.2.25 APIMAN – Read APIs and contracts information

To use the adapter for APIMAN, fill the field “Web Service Home URL” with the APIMAN home page URL you want to connect to. Then provide valid user/password credentials to connect:



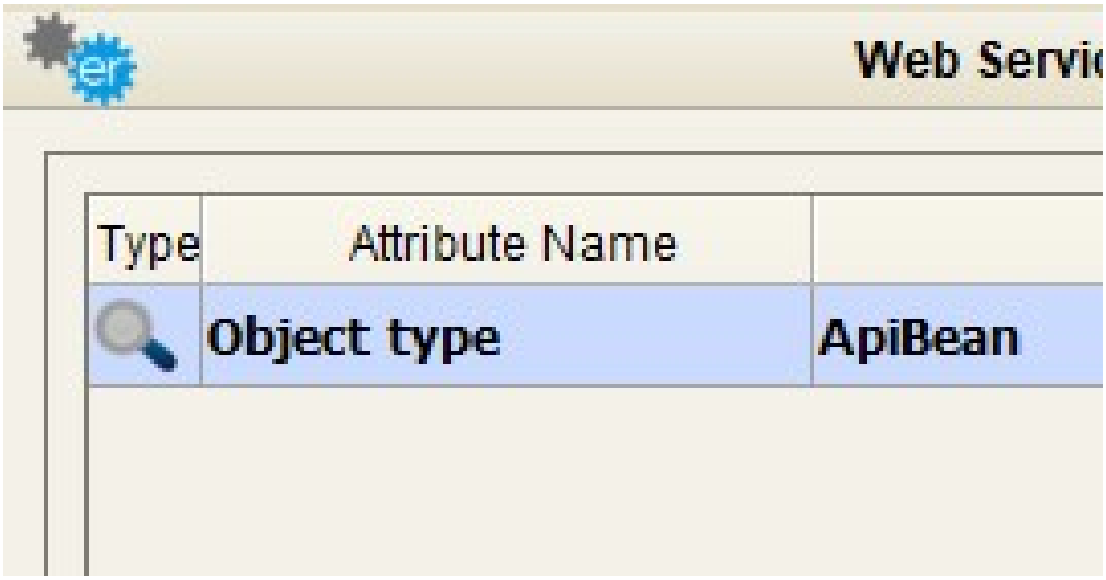
The  button steps into the next window that shows the available operations of that Web Service; choose “**EXPORT DATA**”, to query data stored in APIMAN.

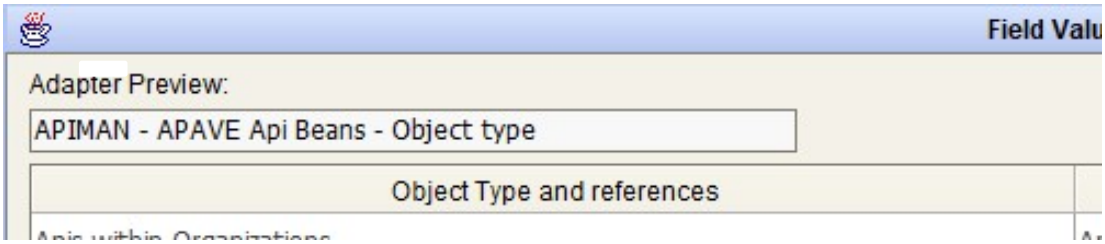


After selecting the desired operation, it is required to enter a name and a description for the adapter in the appropriate fields; the user must browse the file system for the adapter folder (used to save intermediate and result file for adapter execution).

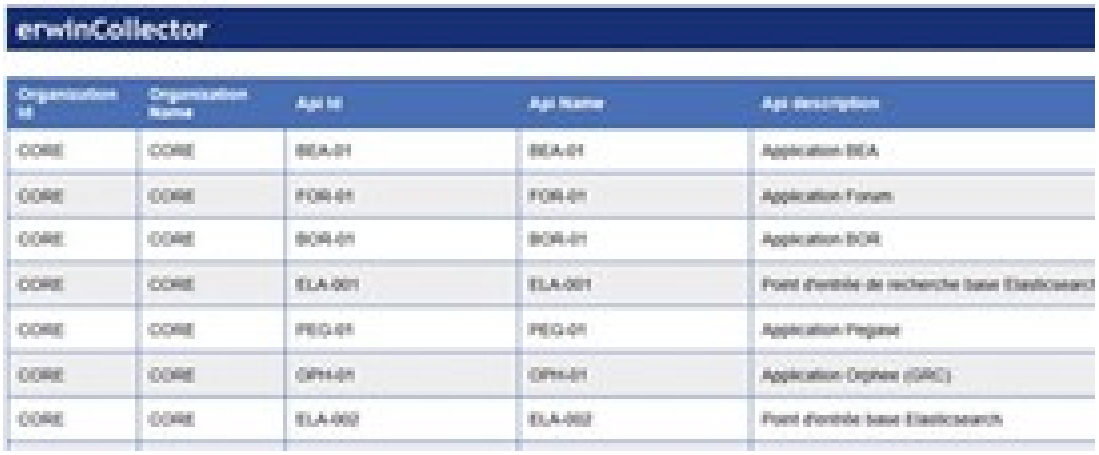
The next button Next steps into the next window that allows the user to set values for the Web Service parameters.

When a lens icon shows beside the parameter name, this means that user may choose the parameter value from a list of available items. In case of this example, with the double-click on the lens icon, a table appears, with the available endpoint for the Adapter:





Pressing the button will open a new pop-up window and the response resulting from the Web Service call is displayed.

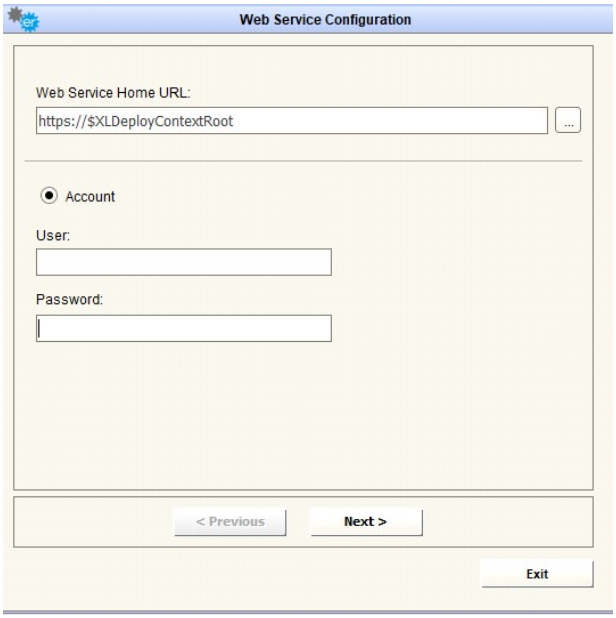


Pressing the button stores all the entered parameters for the web service adapter in the DT database.

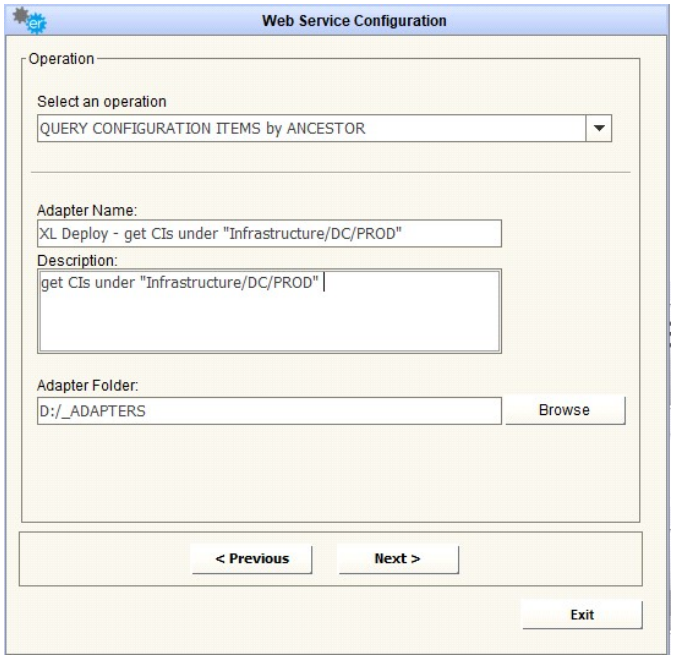
The Exit button closes the data preview window and goes back to Web Service configuration form.

1.2.26 XLDeploy – Read deployment information

To use the adapter for XLDeploy, fill the field “Web Service Home URL” with the APIMAN home page URL you want to connect to. Then provide valid user/password credentials to connect:

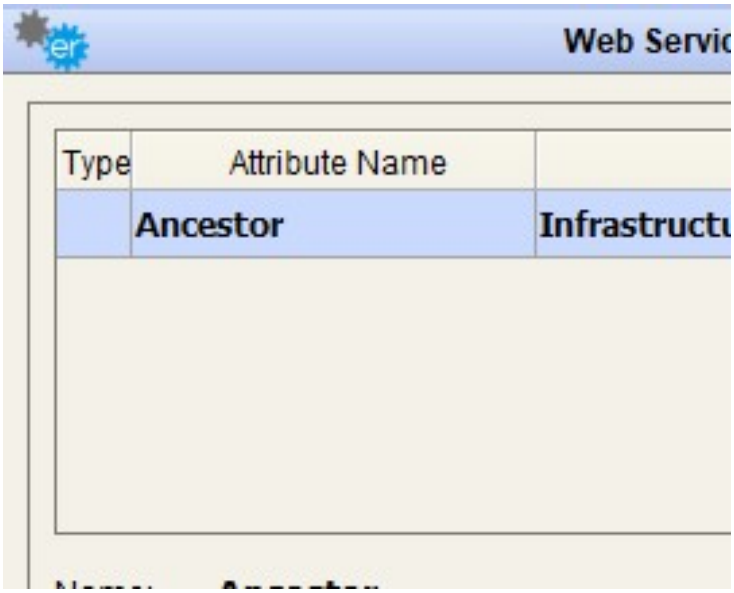


The  button steps into the next window that shows the available operations of that Web Service; choose “**QUERY CONFIGURATION ITEMS by ANCESTOR**”, to query data stored in XLDeploy:

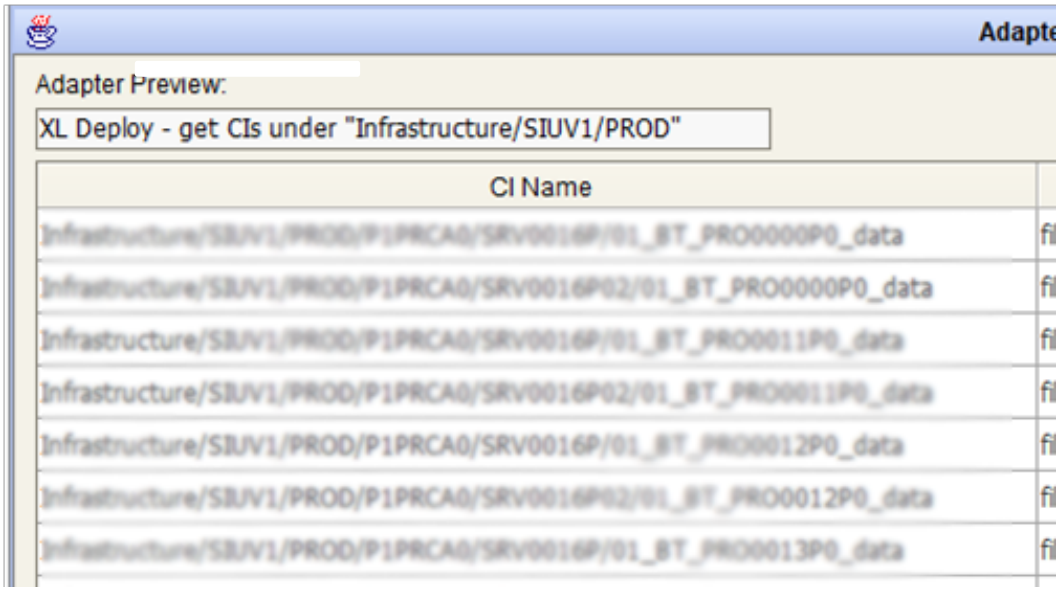


After selecting the desired operation, it is required to enter a name and a description for the adapter in the appropriate fields; the user must browse the file system for the adapter folder (used to save intermediate and result file for adapter execution).

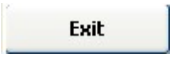
The next button Next steps into the next window that allows the user to set values for the Web Service parameters.



Pressing the Test button will open a new pop-up window and the response resulting from the Web Service call is displayed.



Pressing the  button stores all the entered parameters for the web service adapter in the DT database.

The 

button closes the data preview window and goes back to Web Service configuration form.

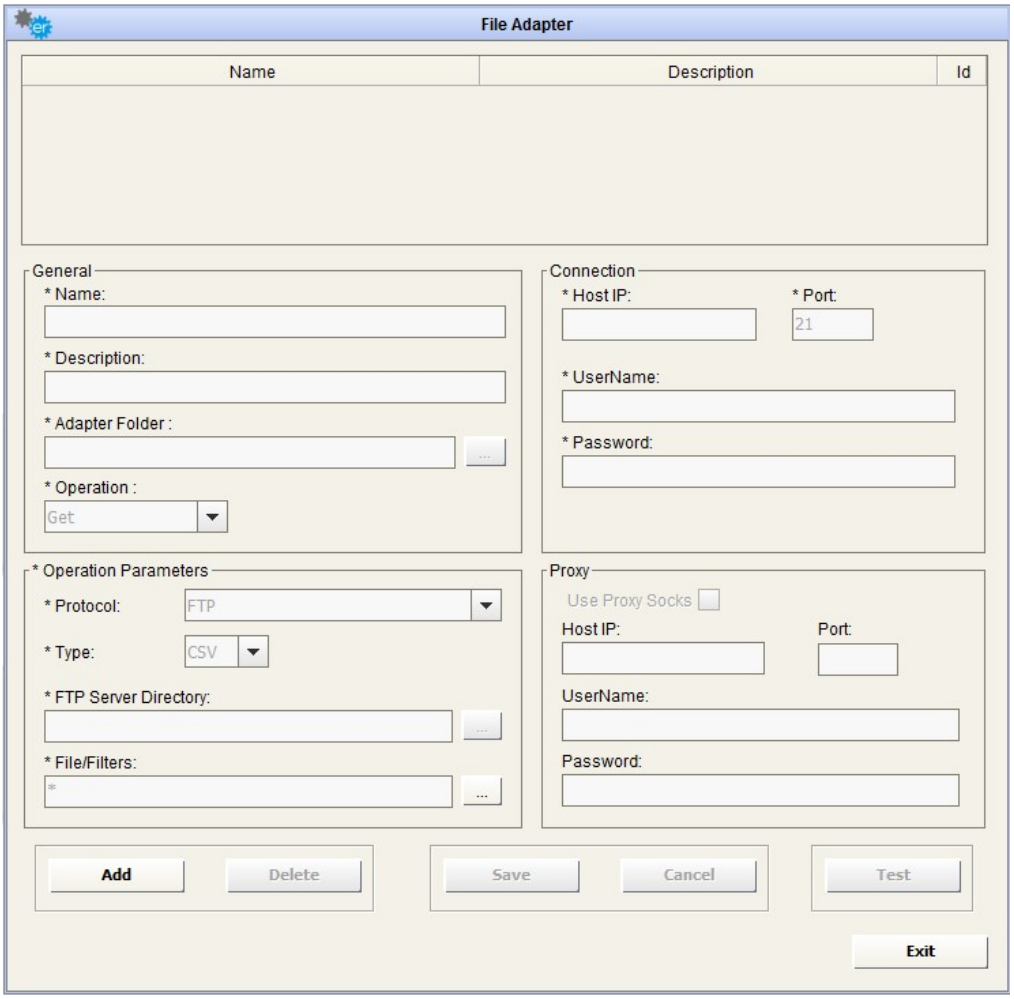
1.3 File Adapter Configuration

Going through this configuration window, the user can set up a file adapter to use as a data source or operation in a workflow.

It's possible to:

- Configure a File Adapter to use the file content as a data source in a workflow – **GET** type
- Configure a File Adapter to use as a workflow operation target – **PUT** type

See later for a detailed description of using a File Adapter the workflow configuration.



The first group of fields in the window relate to the File Adapter List, which lists all the File Adapters available.

The second group is the General Parameters, which include the Name, Description, File Adapter folder and Operation type (GET - PUT); all these fields are mandatory.

Please note that in this case the adapter folder is the one used as the destination folder for the GET adapter and the source folder for the PUT adapters, as explained later.

The third group is the Operation Parameters, including all the parameters needed to actually perform the requested operation, as a Protocol (NFS or FTP), file type, etc.

The fourth group is the Connection Parameters to be set up when configuring an FTP file adapter.

To create a new File Adapter just press the 

button and start editing the new adapter parameters.

The name, description and adapter folder for the File Adapter are the first fields to enter in the form; then, depending on the selection in the drop-down list named **Operation** (GET or PUT) the Operation Parameters section shows the following:

1.3.1 Operation “GET”

When the field “Operation” is set to “GET”, then the Operation Parameter section shows options for the file adapter execution. The first option is related to the protocol (NFS or FTP)

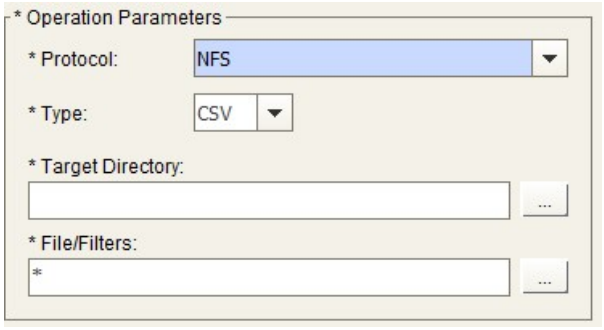
1.3.1.1 NFS Protocol

If the protocol is set to NFS, then the Operation Parameter section shows the following parameters:

- Type (CSV, Sheet, XML, MS Project, DM) – file type the adapter will manage
- Target (remote) Directory that will be used at adapter execution time to get files from; this may be anywhere in the Windows network file system, provided that it is accessible

and readable by the Windows user (“local system account”) performing batch operations.

- Filters – used to include more than one file in the adapter operation (using “*” as a wildcard for the beginning, in the middle, or at the end of file name). If filter is set as a single “*”, then the entire folder content will be used by the adapter during execution.



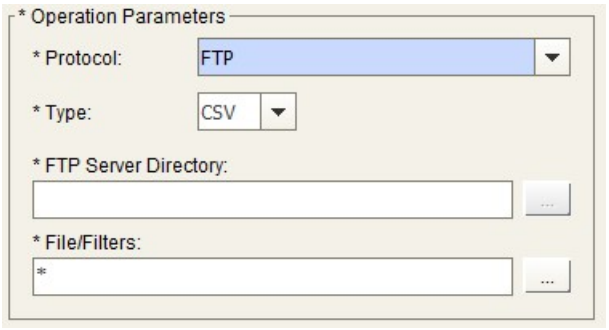
1.3.1.2 FTP Protocol

If the protocol is set as FTP then the Operation Parameter section shows the following parameters:

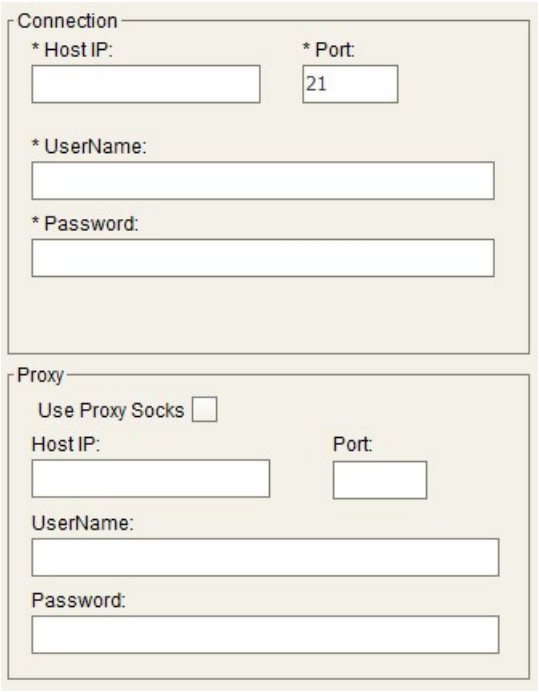
- Type (CSV, Sheet, XML, MS Project, DM) – type of file the adapter will manage

- FTP Server Directory that will be used at adapter execution time; this may be anywhere in the network file system, provided that it is accessible and readable by the adapter specified in the FTP account (see below)

- Filters – used to include more than one file in the adapter operation (using “*” as a wildcard for the beginning, in the middle, or at the end of file name). If filter is set as a single “*”, then the entire folder content will be used by the adapter during execution.



Moreover, the user has to define the Connection and Proxy sections with the parameters needed to access the FTP server.



All the parameters in the Connection section are mandatory, including the FTP account to be used to access the specified directory; if a proxy is used then the user must complete the Proxy section parameters.

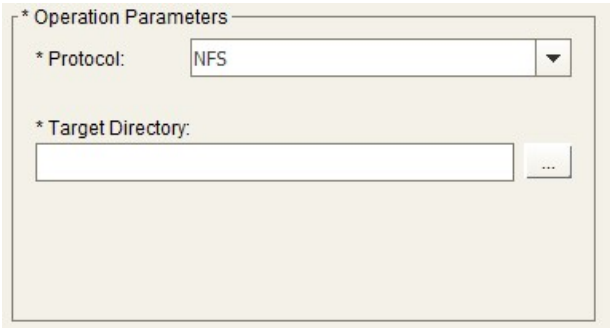
1.3.2 Operation “PUT”

When the field “Operation” is set to “PUT”, then the Operation Parameter section shows options for the file adapter execution. The first option is related to the protocol (NFS or FTP)

1.3.2.1 NFS Protocol

If the protocol is set to NFS, then the Operation Parameter section shows the following parameters:

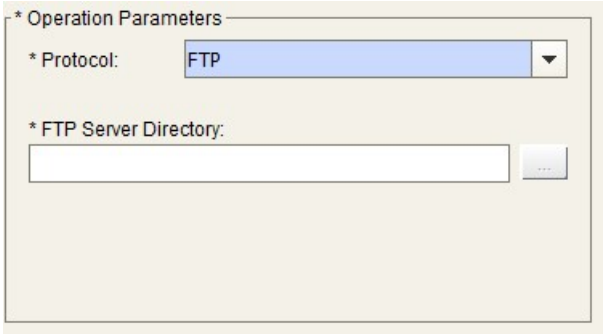
- Target (remote) Directory that will be used at adapter execution time, to drop files coming out from workflows using it as operation target; this may be anywhere in the Windows network file system, provided that it is accessible and writable by the Windows user (“local system account”) performing batch operations.



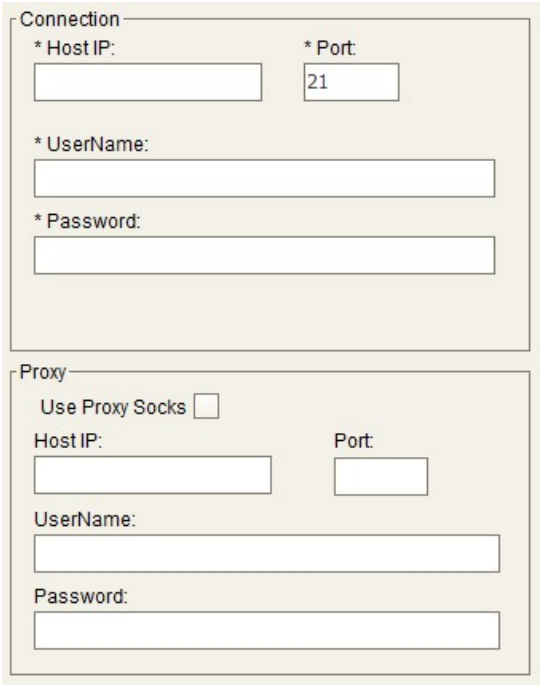
1.3.2.2 FTP Protocol

If the protocol is set to FTP, then the Operation Parameter section shows the following parameters:

- FTP Server Directory that will be used at adapter execution time, to drop files coming out from workflows using it as operation target, provided that it is accessible and writable by the adapter specified FTP account (see below)



Moreover, the user has to define the Connection and Proxy sections with the parameters needed to access the FTP server.



All the parameters in the Connection section are mandatory, including the FTP account to be used to access the specified directory; only if you have a proxy you must complete the Proxy section parameters.

To save the new File Adapter just created, press the Save button. The adapter will be saved and added to the File Adapter List at the top of the window.

To cancel the adapter data just entered and to reset the window, press the Cancel button.

To delete an existing adapter, just select it and press the Delete button. It will be permanently removed from the File Adapter List (all the items linked to the file adapter are removed: job, workflow using it as source).

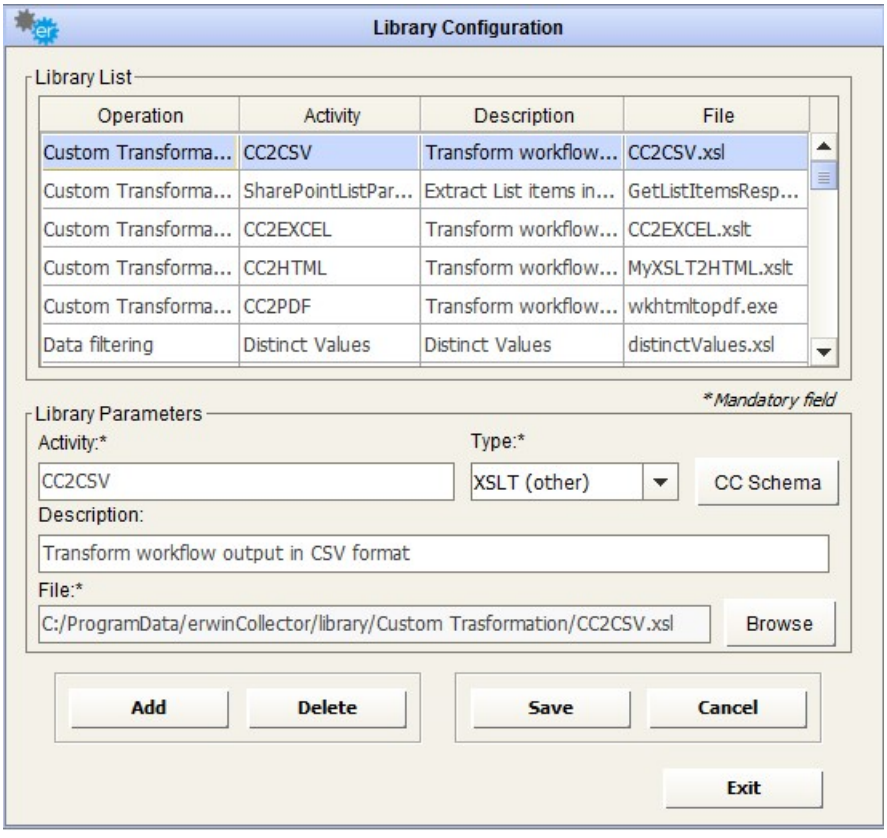
The Exit button closes the window.

2 Transformation Library Configuration

In the Transformation Library there are some pre-built parsers (based onto XSLT) to apply transformations to the data managed through the workflow (see after):

- Data cleanse (like Data Type Check,Not Allowed Text Check)
- Data format (like Format Date - Format Number - Capital/Small letter - Suffix/Prefix - Replace Text)
- Data structure change (like Combine Field - Split Field - Add Fixed value field)
- Data filter (like Fixed values, Interval Value)
- Data Derivation (like Sum values in rows/columns, counting values in rows/columns, Match keys and pick values)

These kinds of XSLT files can't be modified or deleted by the user.



The top field of the window shows the contents of the XSLT library, with a short description and the relative XSL file.

Selecting an activity from the list automatically completes the Library Parameters fields with the parameters of the library.

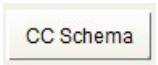
Optionally, it's also possible here to define custom XSLT library elements to be used in the workflow configuration for input or output data formatting (see later). The custom parsers must have been developed in advance, with a proper XML/XSLT editor.

To add a new custom XSLT to the library, press the 

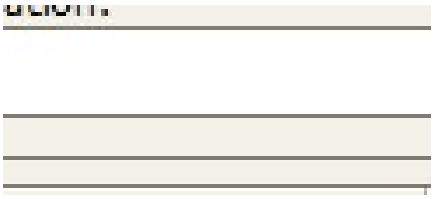
button; once the name and description are provided, the browse button pops up a new window allowing the user to select an .XSLT file from the file system. The selected file will be copied into the subfolder "library\Custom Transformation" of your DT data installation folder.

A custom transformation could be alternatively defined as:

- CC compliant – output is XML, compliant with the CC internal XML schema
- CC uncompliant – output is XML, with a different schema, or HTML, or TXT, or MS Excel
- CM compliant – output is XML, compliant with CM operation schema (only to be used to load in CM models multiple intersection object type at once)

It's possible to have a preview of the CC internal Schema / CM operation Schema pressing the 

button, with the appropriate dropdown type selection:



The  button saves the XSL library once created or modified by the user.

The  button deletes the XSL library that user selected from the list.

The  button resets all the fields just edited by the user.

The  button closes the window.

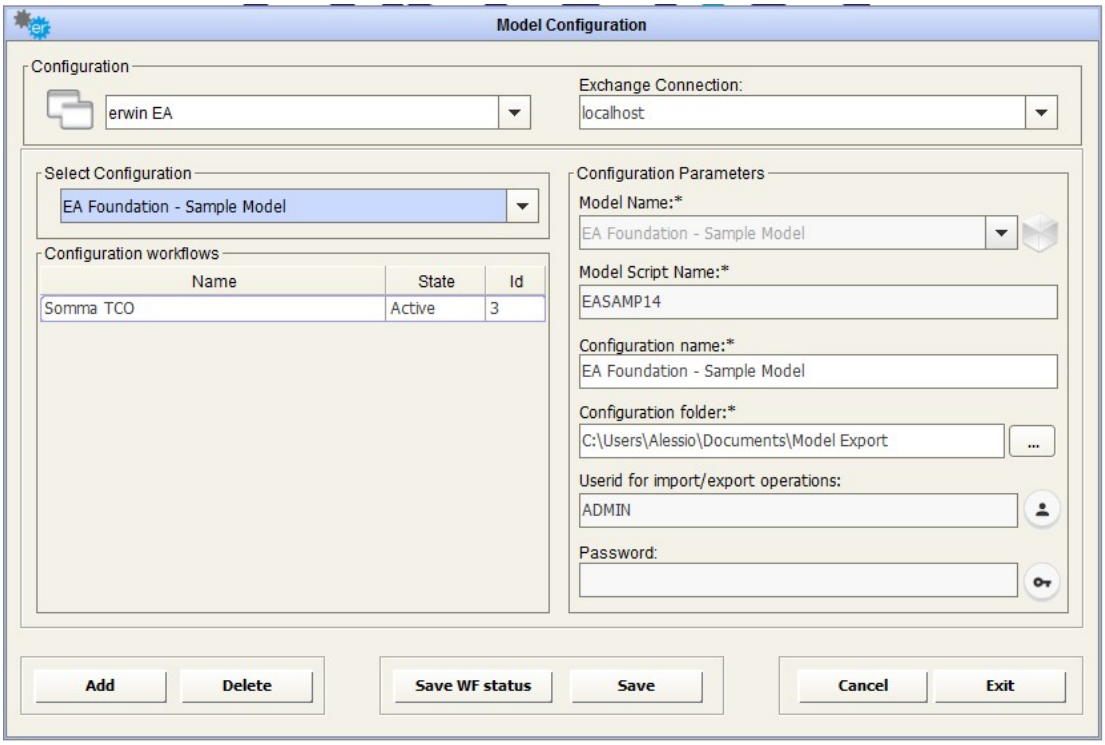
Chapter 3

1 Model Configuration

If logged to a CE Repository, this window can add, delete or modify a *model configuration*. A *model configuration* is a group of parameters that:

- Allows DT to connect to models in CE repository
- Allows the user to configure preferences (publication sets, export folder, etc.)

In order to prevent inconsistency that may arise if two or more configurations address the same data, one model can be associated to only one configuration.



The first step is to select a configuration from the pick list, or add a new one by clicking the  button.

Selecting an existing configuration fills all fields in the form automatically.

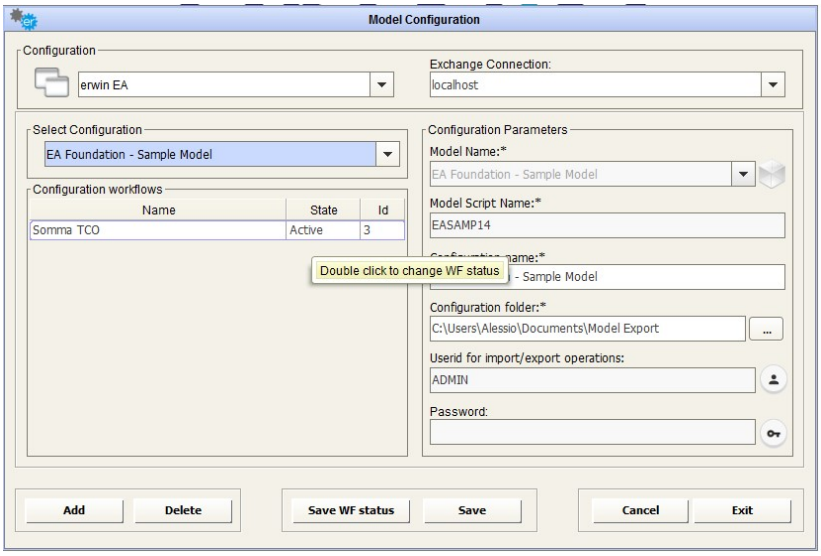
On pressing the “Add” button, the user is required to set the configuration name, to select a model name and an export folder, which will be also *the working folder for automatically created adapters to import/export data from workspace*.

In particular, the user should select model name and preferences in the section “Model parameters”. All fields are mandatory.

In detail, the following parameters have to be defined:

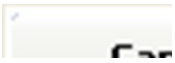
- ***Model name*** – select the name of the model you want to work with.
- ***Configuration Folder*** – browse to a path where you wish the model backups and exports to be saved

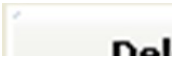
At the left side of model configuration window, is a list of the existing workflows created to manage information flows in and out of that model. In particular, you can see the name, the activation status and ID of each (useful for troubleshooting purposes, see later).



You can change “Active/Not Active” status of the single workflow by double-clicking on status cell, and then simultaneously saving the current set with the button.

To save the Model Configuration use the  button .

Pressing the  button clears all fields on this form.

To delete an existing configuration just select it from the configurations list and press the  button. The configuration will be removed from the list. Please note that if a configuration is deleted, all the adapters, jobs and workflows associated to the configuration will be deleted.

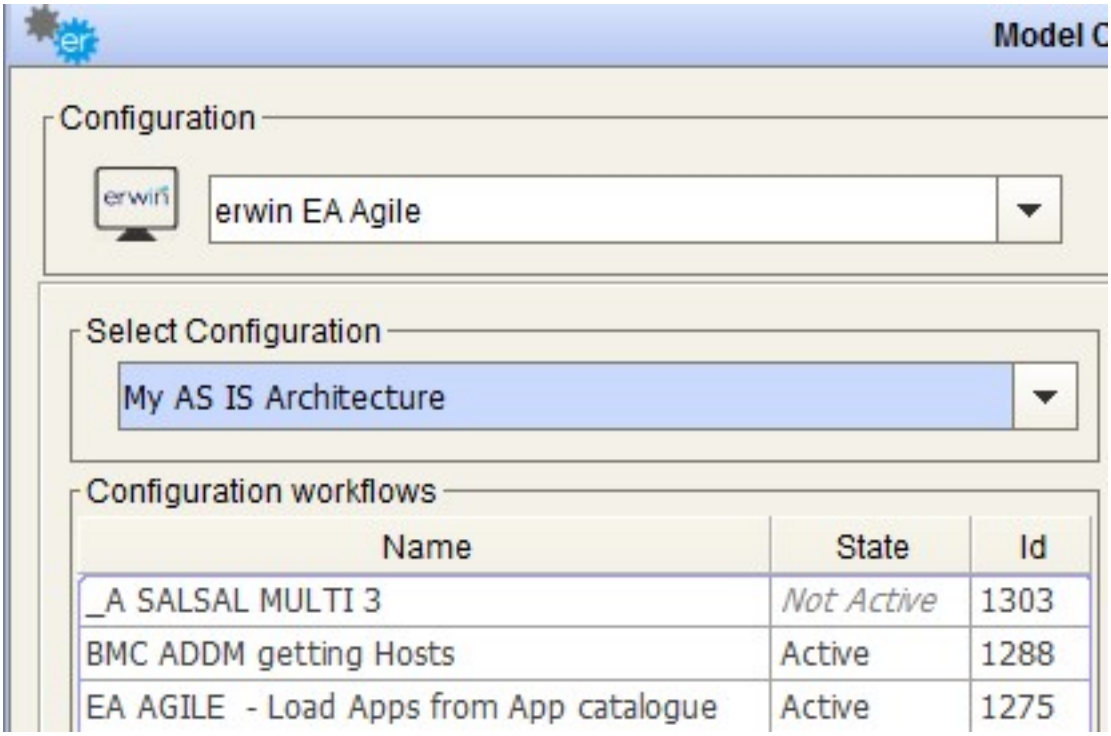
The  button closes the window.

2 Workspace Configuration

If logged in to EA Agile or EA Agile V3, this window is to add, delete or modify a *workspace configuration*. A *workspace configuration*, in a similar way to CM *model configuration*, from the tool perspective, is a group of parameters that:

- Allows DT to connect to an EA Agile workspace
- Allows the user to configure preferences for export folder

In order to prevent inconsistency that can arise if two or more configurations address the same data, one workspace can be associated to only one configuration.



The first step is to select a configuration from the pick list, or add a new one by clicking the 

button.

Selecting an existing configuration fills in all fields on the form automatically.

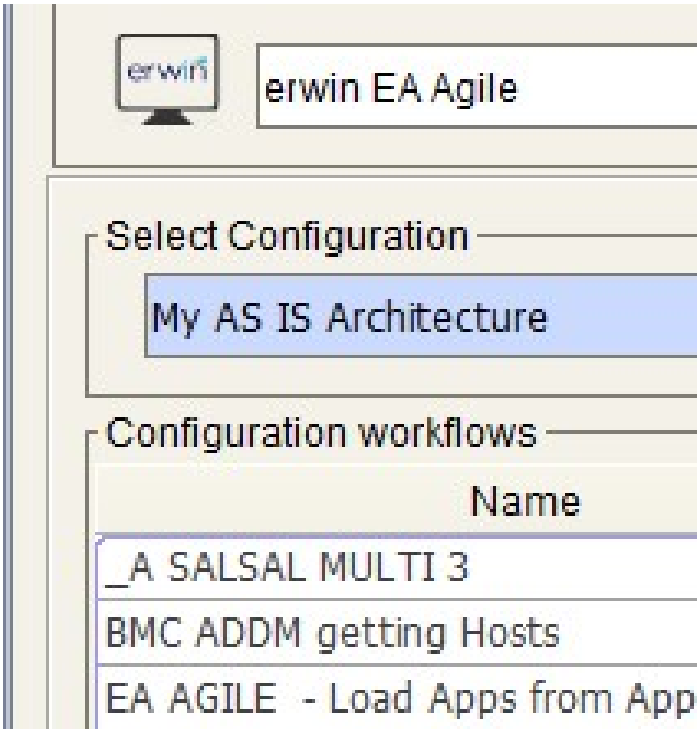
On pressing the “Add” button, the user is required to set the configuration name, to select a workspace name and the export folder, which will be also *the working folder for automatically created adapters to import/export data from workspace*.

In particular, the user should select model name and preferences in the section “Model parameters”. All fields are mandatory.

In detail, the following parameters have to be defined:

- ***Workspace name*** – select the name of the workspace you want to work with.
- ***Configuration Folder*** – browse to a path where you wish the exports to be saved

At the left side of model configuration window, is a list of the existing workflows created to manage information flows in and out of that model. In particular, you can see the name, the activation status and ID of each (useful for troubleshooting purposes, see later).



You can change “Active/Not Active” status of the single workflow by double-clicking on status cell, and then simultaneously saving the current set with the button.

To save all the Model Configuration use the button.

Pressing the button clears all fields on this form.

To delete an existing configuration just select it from the configurations list and press the button. The configuration will be removed from the list. Please note that if a configuration is deleted, all the adapters, jobs and workflows associated to the configuration will be deleted.

The button closes the window.

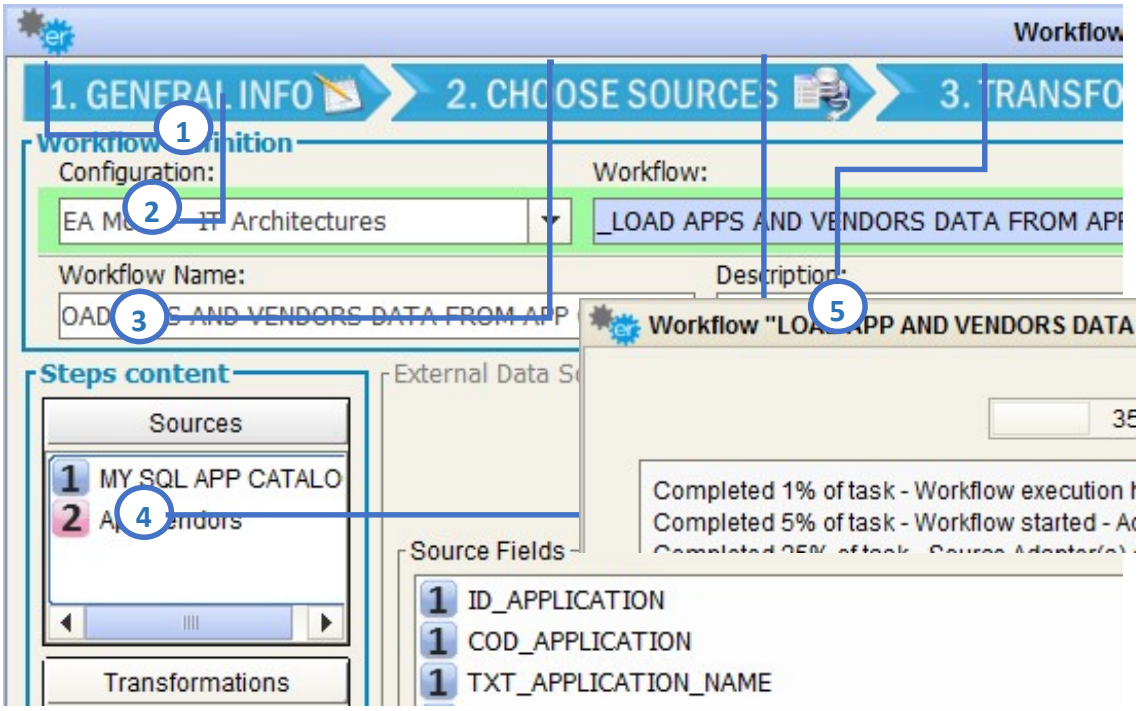
Chapter 4

1 WorkFlow Configuration

This window goes through the configuration and management of the Workflow. For each CW model, for which a configuration has been set up (see

Chapter 3), the user can define more than one workflow.

Each workflow is described through three main sections:
- Data Source, containing information for getting the input data
- Transformation, containing activities to be applied to transform the input data
- Operation, which details the target of the workflow, i.e. where to send the output data



Configuring a workflow means:

1. Providing **general information**, like name, descriptions and "active" status
2. Choosing applicable **data sources**, and providing normalization parameters (like name of the sheet for Excel files, or the object type to extract for an EA/EA Agile Adapter)
3. Optional data **transformation** – filters on source data and other needed data manipulations settings
4. **Operation and target** – what system/human resource is the target for the desired operation

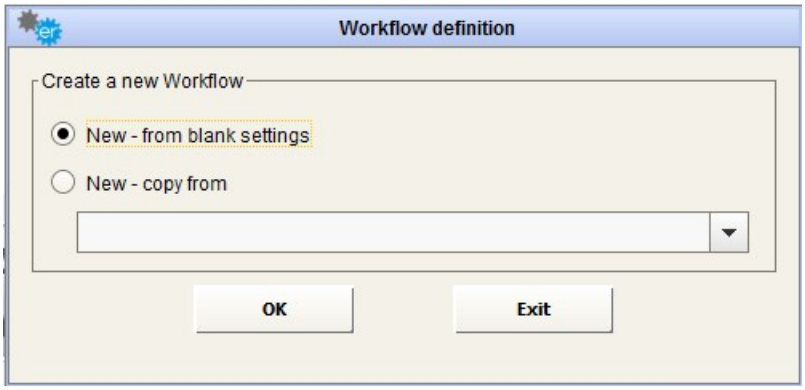
5. **Test** and **save** the workflow**:** workflow is ready to be tested (whatever the setting for “Active” checkbox is) and saved for further scheduled executions (if set in “Active” mode)

First of all the user is required to select a model/workspace configuration in the **Configuration** drop-down list.

In the “Workflow” drop down list are displayed all the existing workflows for that model configuration.

To create a new Workflow, press the  button. This pops up a new window with two options:

- Create the new Workflow from blank settings (option “New – from blank settings”), or
- Copy it from an existing Workflow, that the user can choose from the displayed drop-down list (option “New – copy from”).



When the user presses the “Add” button the workflow name and description are set and then the three sections of a workflow are defined.

1.1 Data Source

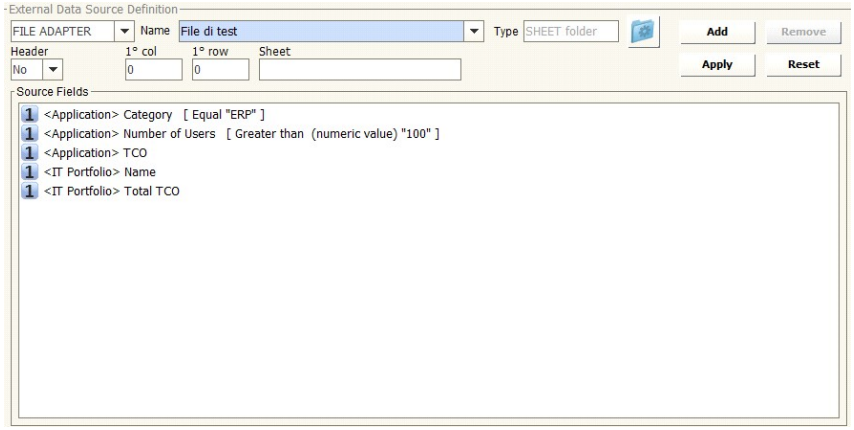
To define the source of a Workflow follow the steps in the section “External Data Source Definition”:

- Select ADD button
- Select the external data source type

According to the type of the data source, the External Data Source Definition section shows the requested data normalization parameters as described below.

User may also create File Adapter, DB Adapter and Web Service Adapter using the <Create New> item in the available adapters list, and pressing “Apply” button.

1.1.1 File Adapter



Select the name of the File Adapter from the drop-down list, Name (as mentioned in Chapter “File Adapter Configuration” only “GET” File Adapters will appear in the list).

- If the type of file is “***CSV***” it’s necessary to complete the following information:
  - Header (yes/no)
  - Separator (between columns)

- If the type of file is “***Sheet***” it’s necessary to complete the following information:
  - Header (yes/no)
  - 1° column
  - 1° row
  - Sheet name

- If the type of file is “***XML***” it’s necessary to complete the following:
  - XSL Parser
  This parser can be selected in a list containing all “non CC compliant” custom transformations, previously defined in “XSLT

Library". Any XSLT parser developed to manage XML data has to provide a CSV structure, with a "semicolon" as a separator.

- If the type of file is "***Project***" it's necessary to complete the following:
  - Project Information (TASK, RESOURCE)
  This allows the user to extract information from the project file, related to the work breakdown structure, task start and end dates, task duration, task cost; or the resource assignments in task, with earned value information, budget and actual cost and work.

- If the type of file is "***DM***" it's necessary to complete the following:
  - DM Information (ENTITY_ATTRIBUTES, RELATIONSHIPS)
  This allows the user to extract information from DM export file, about Entities, their attributes and Primary Key/Foreign Key roles and Relationships:

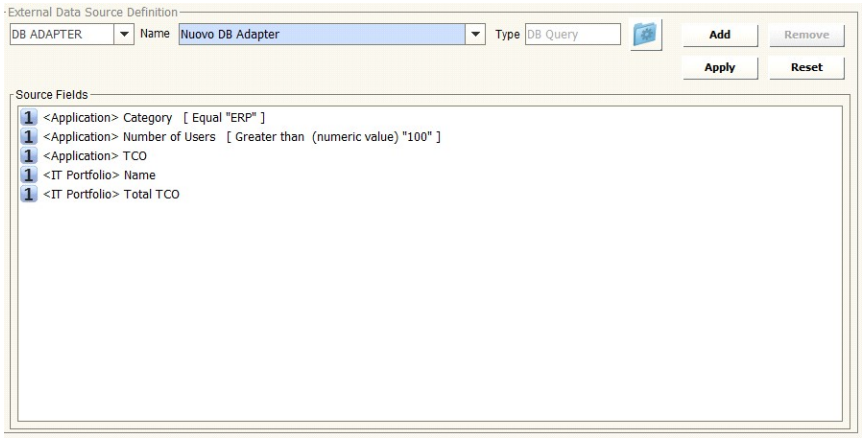
ENTITY_NAME	ATTRIBUTE_NAME	ATTRIBUTE_WTABLE_NAME	ATTRIBUTE_TYPE	ATTRIBUTE_IS_PK
Person	firstName	Person.firstName	char(18)	
Person	surName	Person.surName	char(20)	
Person	ssn	Person.ssn	char(18)	Person
Address	address_line_1	Address.address_line_1	char(18)	
Address	address_line_2	Address.address_line_2	char(18)	
Address	city	Address.city	char(18)	
Address	zip_code	Address.zip_code	char(18)	
Address	address_Identifier	Address.address_Identifier	char(18)	Address
Address	ssn	Address.ssn	char(18)	

REL_NAME	TABLE2TABLE_NAME	PAR
may have	Person may have Address	Pers

  - Click the "Apply" button to save the data source

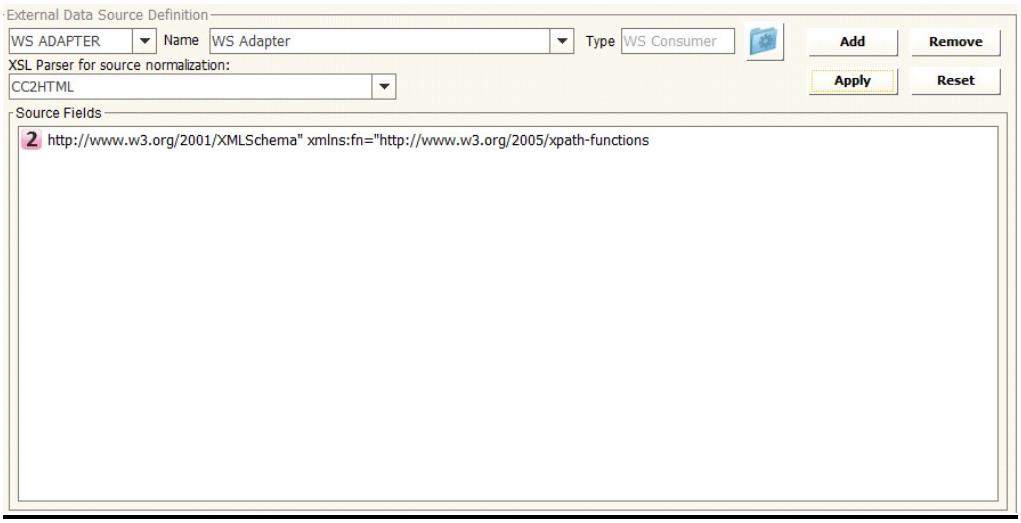
  - A preview of the source fields will be shown

1.1.2 DB Adapter



- Select the name of the DB Adapter from the drop-down list, Name (only DB Query and Stored Procedure Adapters will appear in the list)
- Click the “Apply” button to save the data source
- A preview of the source fields will be shown

1.1.3 WS Adapter

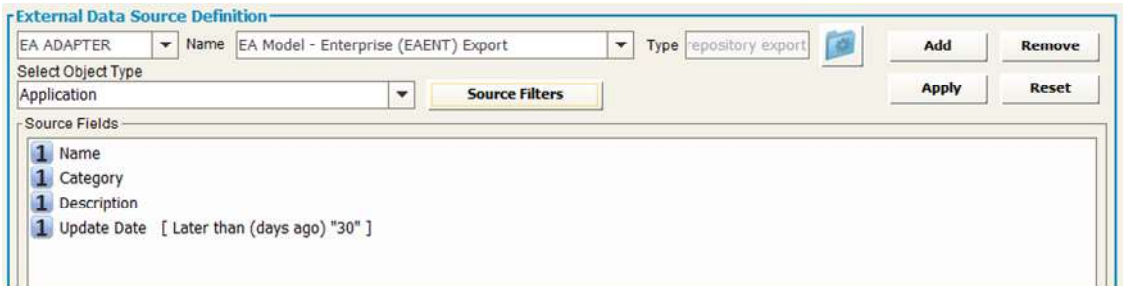


- Select the name of the Web Services Adapter from the drop-down list
- Select XSL Parser for source normalization from the list

This parser can be chosen in a list containing all “non CC compliant” custom transformations previously defined in “XSLT Library”; in particular, any XSLT parser developed to manage outside XML data has to provide a CSV structure, with “semicolon” as separator, to be effective in this step.

- Click on the “Apply” button to save the data source
- A preview of the source fields will be shown

1.1.4 EA Adapter (formerly CM Adapter)

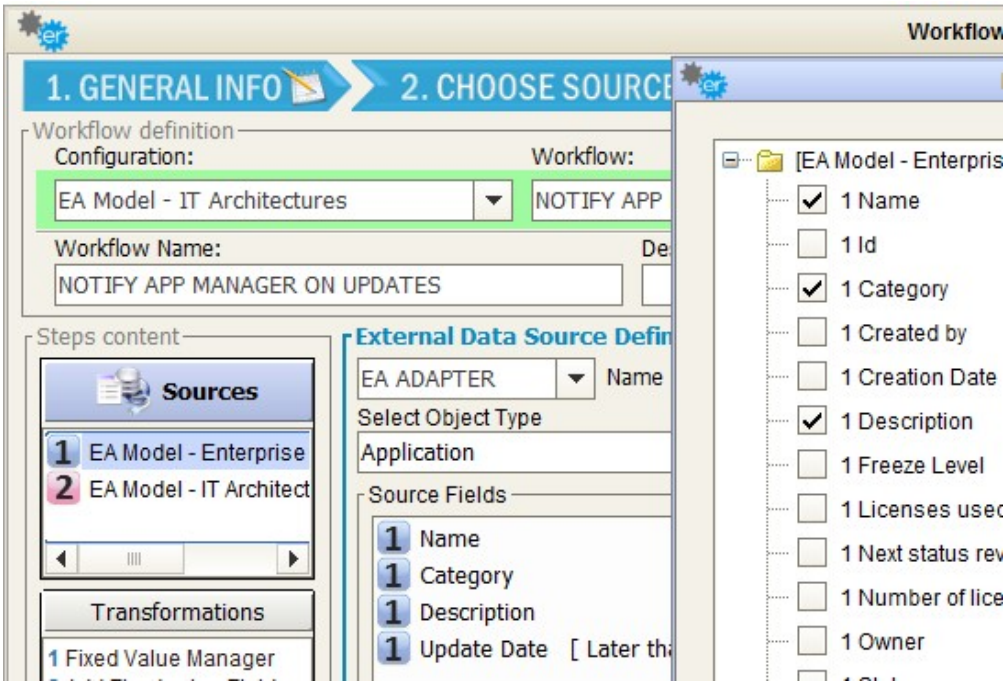


When creating a model configuration, adapters to import / export data from model are automatically created (the working folder is the one set as “Export folder”):

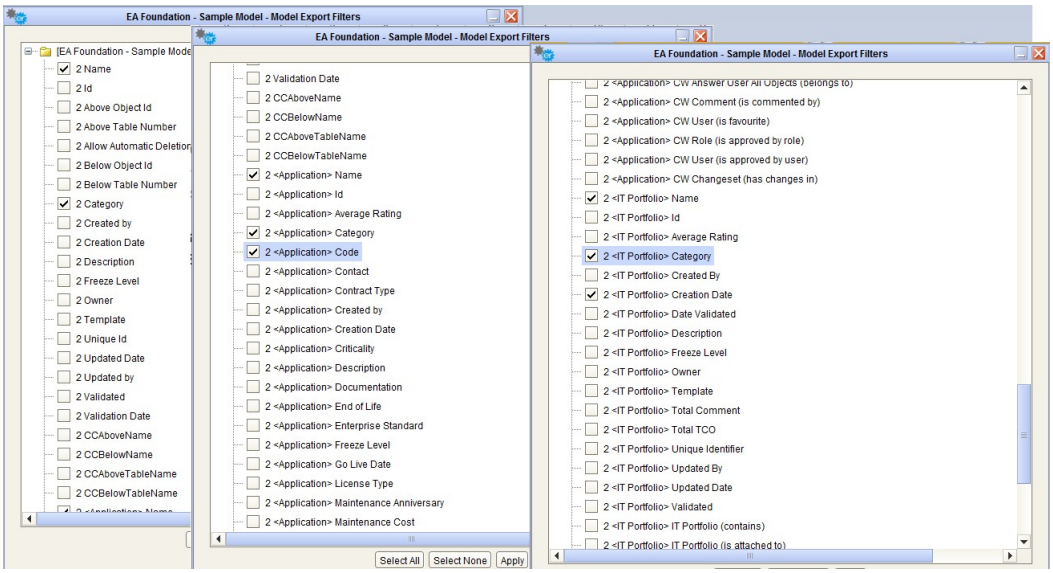
Select the name of the EA Adapter from the drop-down list Name (only the “EA Adapters for import” will be shown in the list)

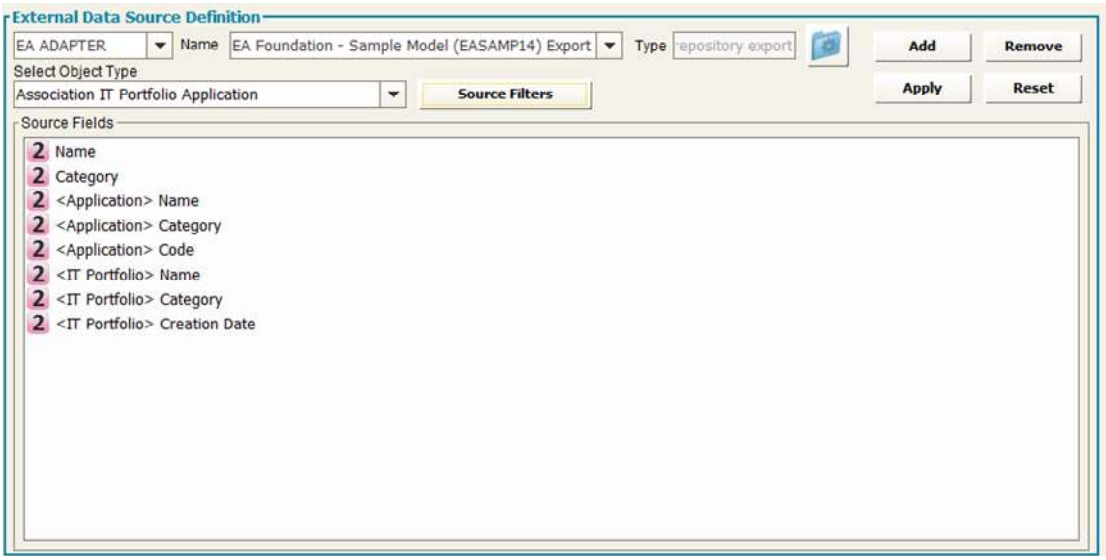
- Select the object type from the drop-down list “Select Object Type”
- Click on the “Apply” button to save the data source
- A preview of the source fields will be shown

It’s also possible to choose which Properties/Associations extract through an EA Adapter for the selected object type. The “Source Filters” button allow user to pick only the desired information from a model, while default operation includes all of them.



When selecting an intersection object type, the list of exportable fields includes not only the ones defined for the intersection object itself, but the ones defined for associated objects. This way, it's easy to have with a single data source a combined set of source object and target object of the association:

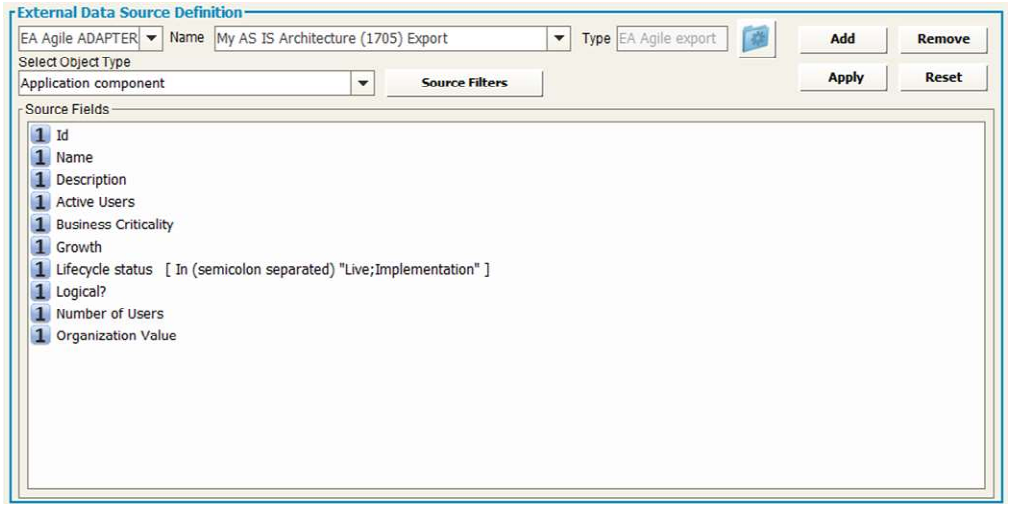




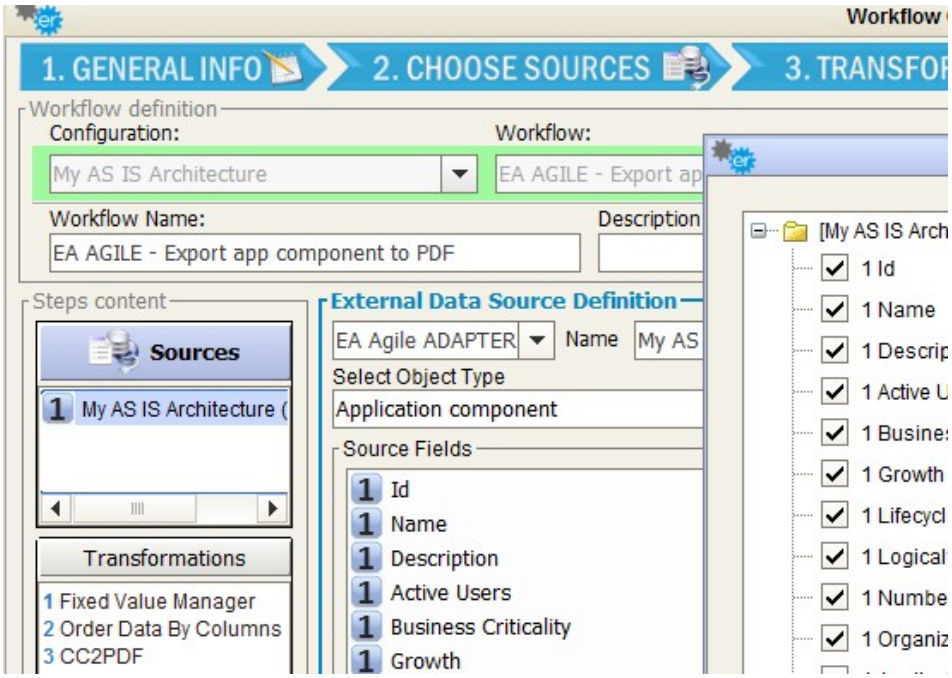
1.1.5 EA Agile Adapter

When creating a workspace configuration, adapters to import / export data from workspace are automatically created (the working folder is the one set as “Export folder”):

- Select the name of the EA Agile Adapter from the drop-down list Name (only the “EA Agile Adapters for export” will be shown in the list)
- Select the object type from the drop-down list “Select Object Type”
- Click the “Apply” button to save the data source
- A preview of the source fields will be shown



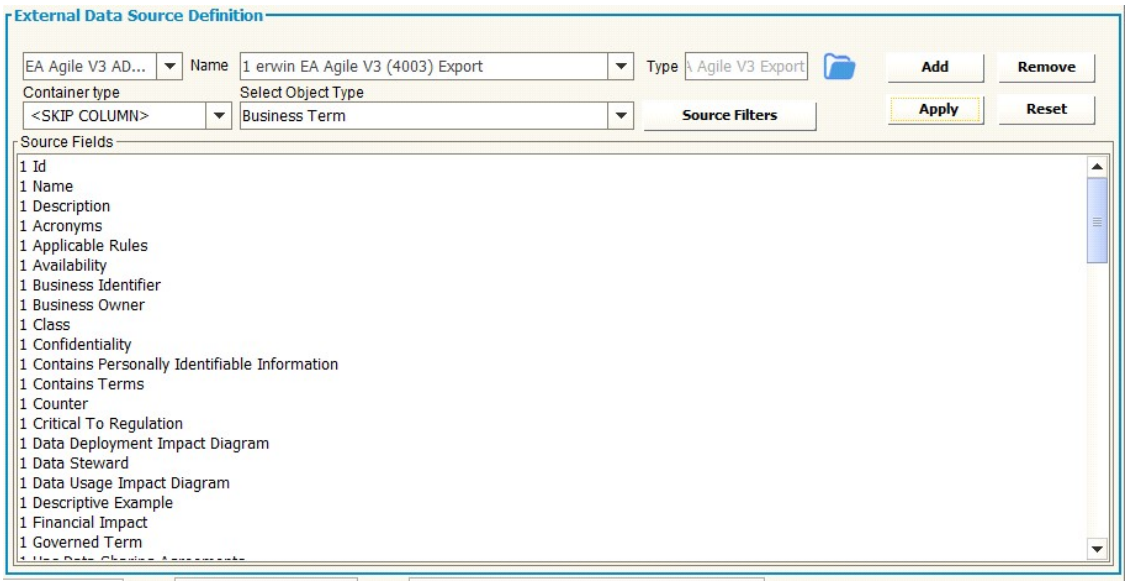
It’s also possible to choose which Properties/Associations extract through an EA Agile Adapter for the selected object type. The “Source Filters” button allow user to pick only the desired information, while default operation includes all of them.



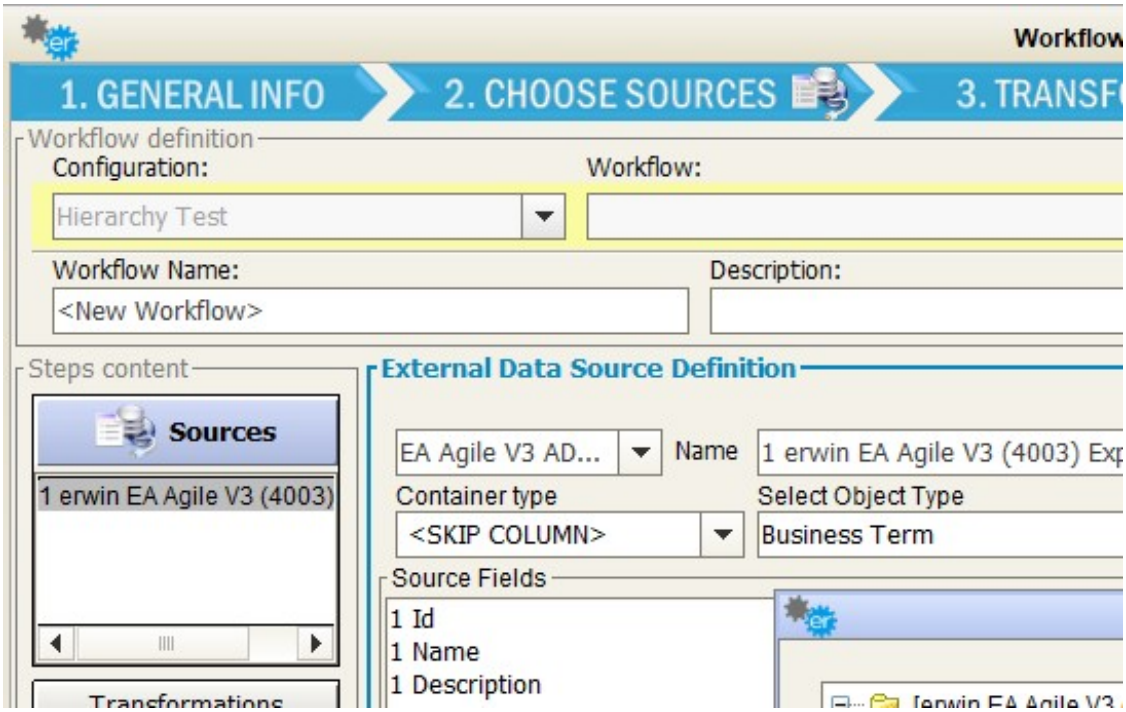
1.1.6 EA Agile V3 Adapter

When creating a workspace configuration for EA Agile V3, adapters to import / export data from the platform are automatically created (the working folder is the one set as “Export folder”):

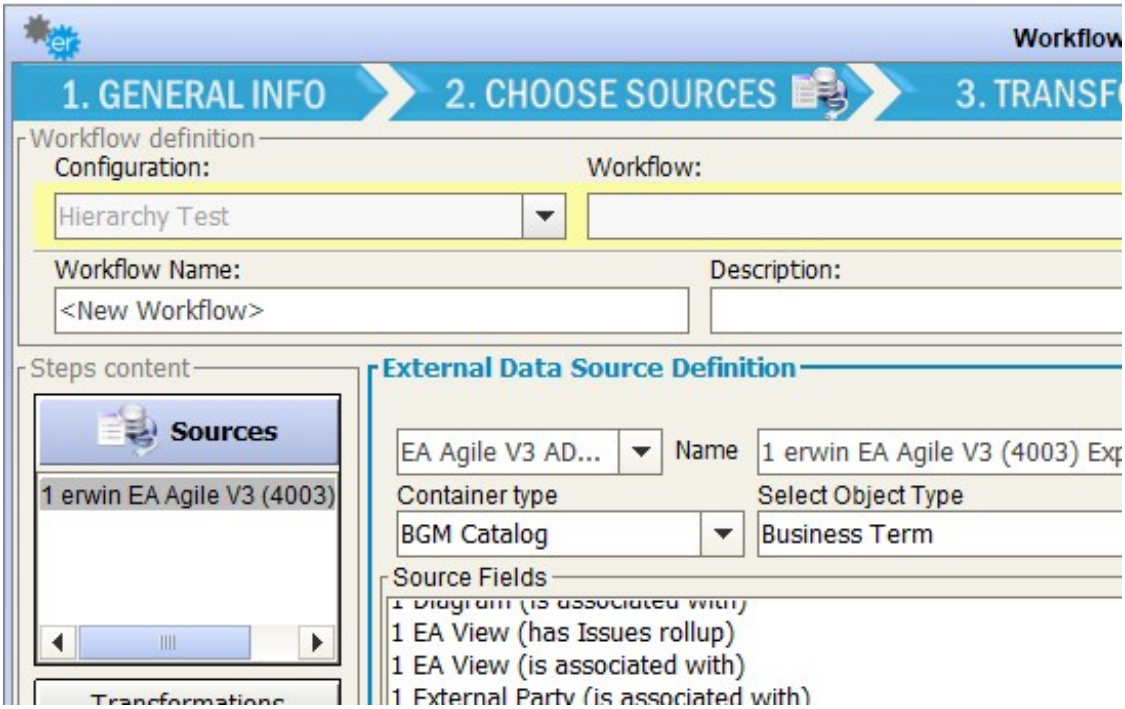
- Select the name of the EA Agile V3 Adapter from the drop-down list Name (only the “EA Agile V3 Adapters for export” will be shown in the list)
- Select the object type from the drop-down list “Select Object Type”
- Click the “Apply” button to save the data source
- A preview of the source fields will be shown
-



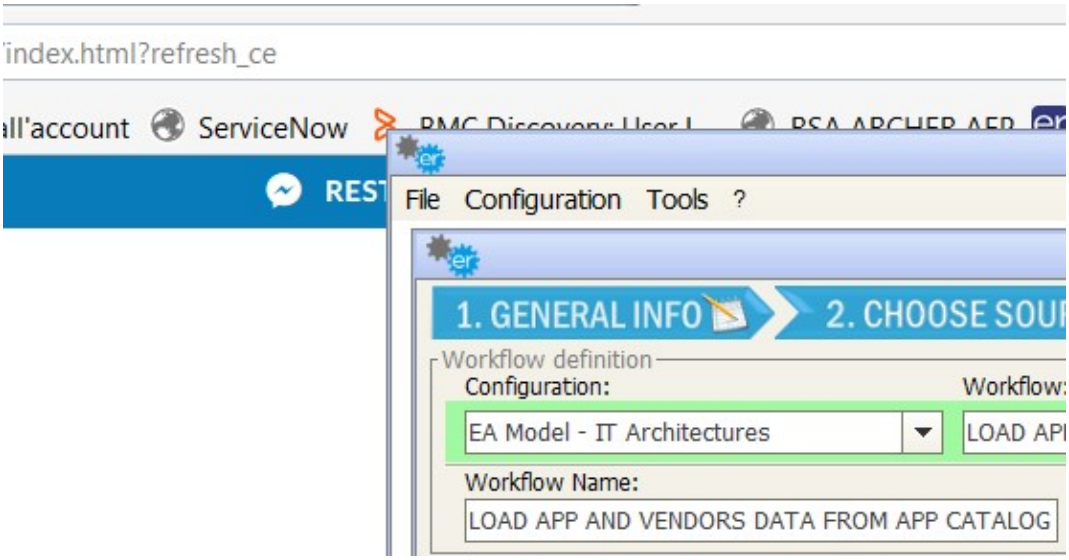
It’s also possible to choose which Properties/Associations
extract through a EA Agile V3 Adapter for the selected object type. The “Source Filters” button allow user to pick only the desired information, while default operation includes all of them.



It is also possible to optionally select a “Container type” for the selected; for an object type that contains either an object of the same type or the selected object type as a terminal node, the hierarchy of the source object type will be resolved in generated fields. This cannot be combined reliably with the source filters.



Regardless of the data source type, it is possible to insert more than one data source in the same workflow of the same type or of different types); each data source is then marked with a sequence number.
In this case you have a “multi source” workflow, and a specific “**multisource job**” is created to be scheduled and produce all the different data sets involved at the same time (thus ensuring the time consistency of the whole set of related information). The data sources defined in a workflow will be displayed in the section Source Definition in the left upper area of the Workflow window:



It’s possible to select one of these data sources to modify it. Please be careful, as when a user modifies the source of a workflow, all fields will be deleted for the transformation and mapping rules, where they are not required by the new source definition.

The Remove button deletes the selected source from the workflow; the Reset button clears all the settings for the selected source.

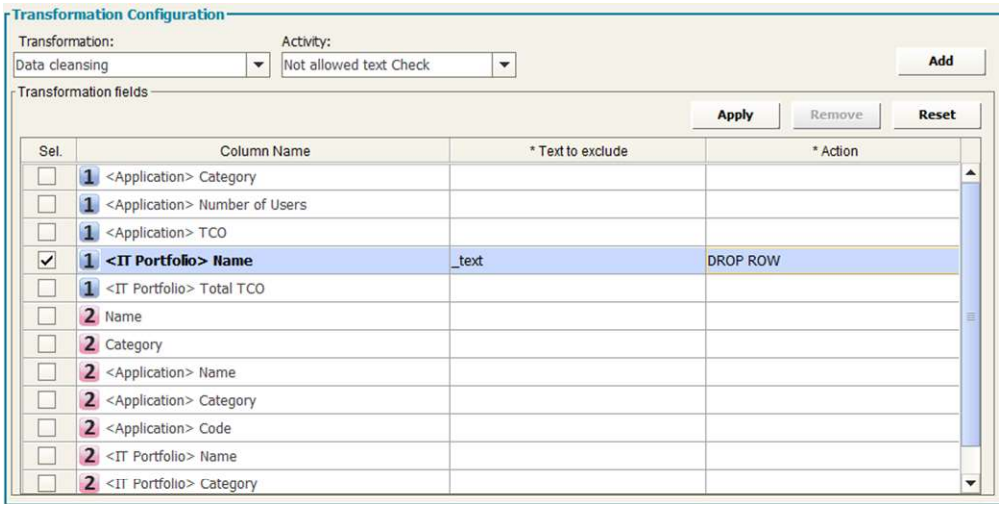
Please note that changing the source will result in a transformation and mappings reset, unless the new source provides exactly the same source columns as the previous.

1.2 Transformation

To define the transformations in the intermediate step of the new Workflow follow the following steps in the section Transformation Configuration:

- Click on the ADD button to insert a transformation. It's possible to add more than one transformation for each workflow and apply a specific transformation to the data source fields or to the new columns coming from a previous transformation of the same workflow. - Select the transformation from the "Transformation" and "Activity" drop-down lists

- Click the "Apply" button to apply the selected transformation rule



The available Transformations and Activities are:

1.2.1 Data Cleansing

1.2.1.1 Data Type Check

The "*Data Type Check*" Transformation applies in the case the user needs to check the *format* of the columns and perform an *action* in case the data format is not compliant with the specified format. The actions are *drop text* (exclude from the following workflow steps only the invalid format values in the specified column), *drop row* (exclude from the following workflow steps the whole record containing an invalid value for any specified column) or *drop file* (discard the entire file – no following operations will be performed). All these configurations may be done separately for each source field.

1.2.1.2 Not Allowed Text Check

The "*Not Allowed Text Check*" Transformation applies when it's required to check if in the selected column there is a specific *string value* and perform an *action* in the case of invalid data.

If a cell contains a value containing that text (i.e. is not valid) the user can choose the action to perform. The possible choices of action are *drop text* (remove specified text from the cells in the specified column), *drop row* (exclude from the following workflow steps the whole record containing an invalid value for any specified column) or *drop file* (discard the entire file – no following operations will be performed). All these configurations may be done separately for each source field.

1.2.1.3 Normalize Blank Spaces

The "*Normalize Blank Spaces*" Transformation applies when it's required to delete some redundant blank spaces in the cell value. It's possible to choose between "Left Trim", "Right Trim", "Trim (both sides)", "Normalize Blank Spaces" (this will trim blank spaces from both sides and delete internal spaces between words if more than one. All these configurations may be done separately for each source field.

1.2.2 Data Formatting

1.2.2.1 Format Date

The "*Format Date*" Transformation applies if it is required to *transform* a column with internal data type DATE from a specific *source format* (for example EUR date format) into a different *target format* (for example USA date format). If the column value can't be transformed into the specific format, the user can choose the action to perform. The possible actions are; *drop text* (exclude from the following workflow steps only the values in the specified column where transformation can't be performed), *drop row* (exclude from the following workflow steps the whole record for a specified column where the transformation can't be performed in any specified column) or *drop file* (discard the entire file – no following operations will be performed). All these configurations may be done separately for each source field.

1.2.2.2 Format Number

The "*Format Number*" Transformation applies if it is required to transform a column with internal data type NUMBER (integer or decimal) from a specific source format (for example EUR number format) into a different target format (for example USA number format). If the column value can't be transformed into the specified format, the user can choose the action to perform. The possible actions are *drop text* (exclude only the values in the specified column where the transformation can't be performed), *drop row* (exclude the whole record for a specified column where the transformation can't be performed in any specified column) or *drop file* (discard the entire file and stop). All these configurations may be done separately for each source field.

1.2.2.3 Capital/Small letter

The "*Capital/Small Letter*" Transformation applies if it is required to change the string format of a specific column; the user can choose from Capitalized, To Upper and To Lower. The transformation will be applied to the specified column value. All these configurations may be done separately for each source field.

1.2.2.4 Suffix/Prefix

The "*Suffix/Prefix*" Transformation applies if it is required to add a specific string to a specific column value; the user can choose between Suffix and Prefix and insert the string to add as prefix/suffix to the column value. All these configurations may be done separately for each source field.

1.2.2.5 Replace Text

The "*Replace Text*" Transformation applies if it is required to replace a specific string value with another string value into a specific column values; the user can edit the text to be replaced and the text to replace it with. Some special replace can be obtained: in a cell containing multi-values separated by commas, for instance, using "**\n**" to replace "," will result in a cell containing a list of those values separated by a newline. Also, wildcards can be used: for instance, when asked to replace "**PROC*_**" with "", DT will delete this sub word from the cell value regardless of the

dynamic part of it. All these configurations may be done separately for each source field.
Special parameter can be used for particular replace requirements:

- ***<%BLANK%>*** as text to be replaced*:* to replace empty text
- ***<%SYSDATE%>*** as text to replace with: to obtain datetime (at execution time) in *UTC format or* ***<%SYSDATE_EUR%>*** to obtain datetime in *European format* or ***<%SYSDATE_USA%>*** to obtain datetime in *United States format* (see here for date format descriptions).

1.2.2.6 Replace Value on Condition

The "*Replace Value on Condition*" Transformation applies if it is required to set a specific value if a condition is matched and another if it's not. The user can edit the condition on the source column, choosing between "Equal", "Not Equal", "Less than (numeric value)", "Greater than (numeric value)", "Like" or "Not Like" options, and then editing the value to be compared with the source column value. Then the user may pick between the available columns to set the column target to be used in case the condition is matched or not. Parameters can be applied as previous transformation.

1.2.2.7 Rename and Order Columns

The "*Rename and Order Columns*" Transformation applies if it is required to change the name of a specific column, and or its order, in the output; the user can choose to rename a column, to give it a different order from the natural one (as coming from the source), or both. The order is not mandatory, while the given orders must not be conflicting. The transformation will be applied to the specified column value. All these configurations may be done separately for each source field.

1.2.2.8 Selected Value in List

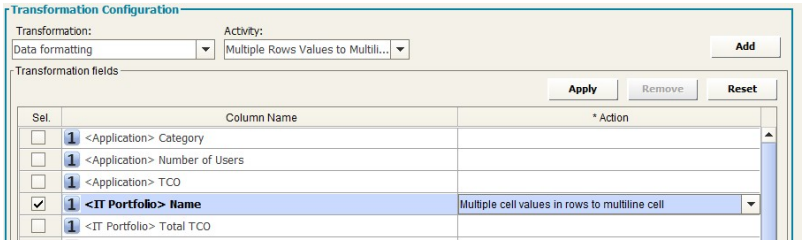
The "*Selected Value in List*" Transformation applies if it there is a column containing a multiline text, as when exporting from a CM model an object and a list of associated objects, and it's needed to have a single item of that list, either the first or the last. The user can set the option on the source column, choosing between

"First value in List", "Last value in List". All these configurations may be done separately for each source field.

1.2.2.9 *Multiple Rows Values to Multiline Cell*

The "*Multiple Rows Values to Multiline Cell*" Transformation applies to get a list of items from a column containing different values, fixed other columns, grouping them in a multiline cell.

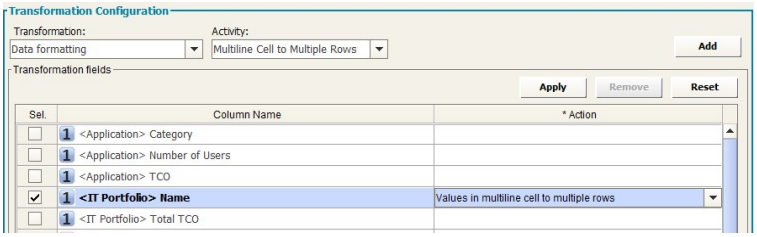
Typical use case: when exporting an intersection object type, with couples of associated objects, from a model, for a given couple obj1-obj2, obtain one single row with obj1 and a list of all the obj2 for that obj1 value.



1.2.2.10 *Multiline Cell to Multiple Rows*

The "*Multiline Cell to Multiple Rows*" Transformation applies to get, from a single multi-value (multiline) cell, one different row for one single different value in multiline value

Typical use case: getting an object from CM, with properties and associations and obtain one row per associated object, for selected association type

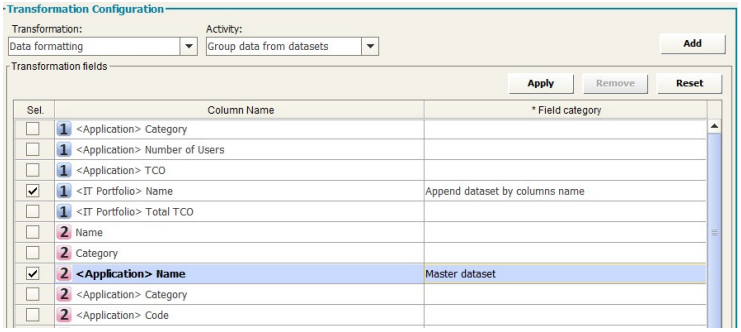


1.2.2.11 *Group data from datasets*

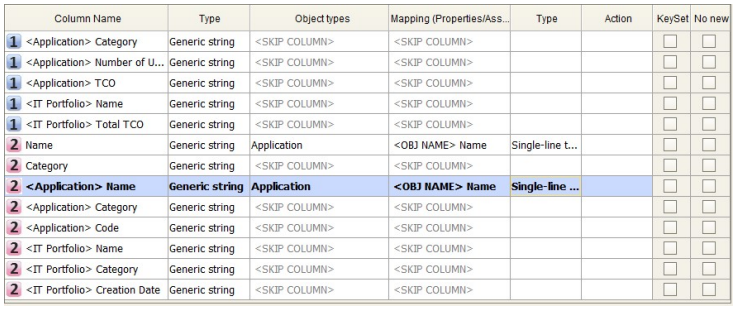
The "*Group data from datasets*" Transformation applies when data belonging to different dataset must be grouped into one.

Typical use case: different lists of the same object type are provided by different data sources, and must be loaded into model, in a single workflow operation. This can be obtained defining:

- one dataset to be the **master** (this will contain all merged data): set this option on whatever column in the dataset
- one or more dataset providing data to **append by column name** (data will be appended to the master one by its columns name): set this option on whatever column in the dataset
- one or more dataset providing data to **append by column position** (data will be appended to the master one by its columns position): set this option on whatever column in the dataset



After the transformation, the master dataset will contain all data as per configured merging options, and can be used to send data to model:



1.2.2.12 Sort Data by Columns

The "*Sort Data by Columns*" Transformation applies if it is required to sort data by column contents. Use columns "Order

Index” (1..4) and “Order Type” (“A to Z”, “Z to A”, “Numerical Ascending”, “Numerical Descending”) to have data sorted by the corresponding content and settings.

1.2.2.13 Compare and Replace

The “*Compare and Replace*” Transformation applies if it is required to compare two column values, and replace a third or fourth column value, if the condition is matched or not. Check the source column to be compared, set the condition, and the value for the comparison; select the true value (column to be used to replace value of the source, when condition is matched) and false value (column to be used to replace value of the source, when condition is not matched). After the transformation, the source column value will be replaced with the “true” value or “false” value, depending form the result of the condition evaluation.

1.2.2.14 Value Substring

The “*Value Substring*” Transformation applies if it is required extract part of a text from the value of a given field. Check the source column, the *start from* index, and the *length* of the text to extract. After the transformation, the source column value will be replaced with the extracted text. For instance, applying this transformation to “Application” with a *start from* equal to 1 and a *length* of 3, the output will be “App”. Providing a negative number for the *start from* parameter, the start will be considered from the end of the original string. For instance, applying the transformation to “New York” with a start from equal to -4 and a length of 4, the output will be “York”.

1.2.3 Data Structure Changing

1.2.3.1 Combine Field

The “*Combine Field*” Transformation applies if it is required to create a new column as result of the combination of two other column values. The user can insert a name of the new column, the *first field* of the combine operation, a *separator* and the *second field*.

The user must indicate which is **the data source to assign to** the new column, using the “Adapter to assign column” drop-down list. This new column can be used by another transformation or by the mapping operation. Special chars like “**\n**” (combine with newline) can be used for specific requirements.

1.2.3.2 Split Field

The “*Split Field*” Transformation applies if it is required to create two new columns as result of the splitting an existing column value.
The user can insert the name of the *two new columns* and the *separator* that the system must use to perform the split operation. If you have more than one data source, you must indicate which is the data source to assign the new column to, using the “Adapter to assign column” drop-down list. This new column can be used by another transformation or by the mapping operation. Special chars like “**\n**” (split by newline) can be used for specific requirements.

1.2.3.3 Add Fixed value field

The “*Add Fixed Value Field*” Transformation applies if it is required to create a *new column* with the same *value* for all the entries. The user can insert a name for the *new column* and the *value* to assign to it. When you have more than one data source you must indicate which is the data source to assign the new column to, using the “Adapter to assign column” drop-down list. This *new column* can be used by another transformation or by the mapping operation. A parameter can be used to populate the new column with the system date and time: ***<%SYSDATE%>*** for *UTC format*, ***<%SYSDATE_EUR%>*** for *EUR format*, ***<%SYSDATE_USA%>*** for *USA format* (see paragraph 1.3.1 for date format descriptions).

1.2.3.4 Copy Column

The “*Copy Column*” Transformation applies if it is required to create a *new column* with the *same values of another*. The user can insert the name of the new column and the column value to assign to it, picking it from a list of available columns. The user must indicate which is the data source to assign the new column to, using the “Adapter to assign column” drop-down list.

This new column can be used by another transformation or by the mapping operation.

1.2.3.5 Drop Column

The "*Drop Column*" Transformation applies if it is required to delete one or more *columns* from the output. The user must check the column to be deleted and set the action to "DROP COLUMN". The selected columns will not be included in the resulting output. A second option, "DROP DATASET", will remove from the output the entire dataset that the column is in.

1.2.4 Data Filtering

1.2.4.1 Fixed value manager

The "*Fixed Value Manager*" Transformation applies if it is required to *filter* the workflow data by some *fixed value* specified for a given column. The user can insert the format of the column, the *filter operator* between "*Equal", "Not Equal", "Greater than", "Less than", "Like"* or *"Not Like*", *"Equal sysdate", "Later than (days ago)", "Later than (hours ago)", "Later than (minutes ago)", "Earlier than (days ago)", "Earlier than (hours ago)"* , *"Earlier than (minutes ago)", "In (semicolon separated)", "Not in (semicolon separated)"* and the value to use as filter. For example, the condition "Not Like" and the value "a" will remove from the resulting dataset all the rows in which that column value does not contain "a" (case sensitive). All the configurations may be done separately for each source field.

When using "Like" and "Not Like", wildcard "**%**" can be used to filter content containing text ("%CRM%": only cells containing "CRM" match filter), starting with text ("CRM%": only cells starting with "CRM" match filter), ending with text ("%CRM": only cells ending with "CRM" match filter).

When possible, **filters defined as first transformation in the list, for columns coming from a CM Object Type export**, are executed at export time, allowing for more compact and less consuming workflow execution.

1.2.4.2 Interval value manager

The "*Interval Value Manager*" Transformation applies if it is required to *filter* the workflow data by a *list of possible values* specified for a given column. Users have to define the *value list*, and the *separator* between the list items. All these configurations may be done separately for each source field.

1.2.4.3 Distinct values

The "*Distinct Value*" Transformation applies if it is required to *obtain distinct occurrences of data* by a *list of columns* specified as the keyset. Users have to define the columns to be the keyset setting the *action "DISTINCT"*. The result will contain only one occurrence for any combination of values of the selected "DISTINCT" columns list.

1.2.4.4 Not All Null values

The "*Not All Null values*" Transformation applies if it is required to *check that at least one value is not null* in a *list of columns* specified by user. Users have to define the columns to be verified setting the *action "Check value in Column"*. The result will contain only rows containing at least one non null value in the selected columns.

1.2.4.5 In/Not In Dataset

The "*In/Not In Dataset*" Transformation applies if it is required to *filter one dataset by the condition that a key column value is (not) in the allowed values list, provided by a slave key column in a second dataset*. Users have to define for the master dataset a column as the "*Key (master)*" and a column as the *"Key (Slave) – IN"* or *"Key (Slave) – NOT IN*". The result will contain rows in the master dataset, only if its master key column value (does not) exist in the slave dataset, in the slave key column.

1.2.5 Data Derivation

1.2.5.1 Match Keys and Pick Values

The "*Match Keys and Pick Values*" Transformation (former "*Format Values On Conditions"*) applies if it is required to join values from different data sets, apply conditions to different column values, and then specify a result column value if the conditions (evaluated all together) are matched or not. For first, given two datasets, the user must choose a column for the first and a column for the second, setting one as "*Key (master)*" and the other as "*Key (slave)"* for the join operation. Then, for the remaining columns, the user may optionally define a "*condition*" as described for the "Replace value on condition" transformation: they have to all be matched to make the transformation condition be true. Lastly, the user can choose one or more columns as "*result*" column, thus choosing which column value to set for it if the condition is true and which if it's false, picking them from the joined dataset columns. User may also define a single column as "*Key (master)/Result"*, to be used both as a key for the join, and to host the result when matching occurs. Please be sure that column that is identified as "Key (slave)" contains only one occurrence per key value, while this is not mandatory for the master.

1.2.5.2 Count Items In Column

The "*Count Items In Column*" Transformation applies if it is required to evaluate the number of item in a list, with an item per line in a multiline *column*. The user must give a name for the new column that will contain the number of items (one per line of the multiline cell) and select from a dropdown list the column containing the multiline text. The output column will contain the number of items (lines) in the selected multiline column.

1.2.5.3 Count Items In Rows

The "*Count Items In Rows*" Transformation applies if it is required to count the *occurrence of a given keyset for several rows*. The user must choose one or more columns to be the "*Key (for data aggregation)*", a column to be the "*Source column (to count items for keyset)*" and a third column to be the "*Result*". The output will contain a single row for any combination of keys; the "*Result*"

column will contain the count of selected "*Source*" column in different rows for the same keyset. It can be used for instance to count associated objects for a master object.

1.2.5.4 Sum Values In Rows

The "*Sum Values In Rows*" Transformation applies if it is required to sum the *numerical* values in a selected *column for several rows*. The user must choose one or more columns to be the "*Key (for data aggregation)*", a column to be the "*Source column (to sum values from)*" and a third column to be the "*Result*". The output will contain a single row for any combination of keys; the "*Result*" column will contain the sum of values in the selected "*Source*" column in different rows (non numerical values will be skipped).

1.2.5.5 Sum Values In Columns

The "*Sum Values In Column*" Transformation applies if it is required to sum the *numerical* values in selected *columns for one single rows*. The user must choose one or more columns to be the "*Source column (to sum values from)*" and another column to be the "*Result*". The "*Result*" column will contain the sum of values in the selected "*Source*" columns in same row (non numerical values will be skipped).

1.2.5.6 Divide Values In Two Columns

The "*Divide Values In Columns*" Transformation applies if it is required to get the division between values in two columns of a dataset . The user must choose one column to be the "*Numerator*", a column to be the "*Denominator*" and a third column to be the "*Result (number)*" or "*Result (percent)*". The output will contain in the "*Result*" column the division between values in "*Numerator*" and "*Denominator*" columns in the selected format (non numerical values will produced a blank result).

1.2.5.7 Two Levels Aggregation

The "*Two Levels Aggregation*" Transformation applies if it is required to sum the *numerical* values in a selected *column for several rows, with two different levels of aggregation*.

The user must choose one column to be the "*Key (for first level aggregation)*", a column to be the "*Result (for first level aggregation)",* a column to be the "*Key (for second level aggregation)",* a column to be the "*Result (for second level aggregation)",* and a column to be the *"Source column (to sum values from)*". The output will contain a single row for any combination of keys; the "*Result (for first level aggregation)*" column will contain the sum of values in the selected "*Source column (to sum values from)*" column in different rows, given a single "*Key (for first level aggregation)"* value, and the "*Result (for second level aggregation)*" column will contain the sum of values in the selected "*Source column (to sum values from)*" column in different rows, given a single "*Key (for second level aggregation)"* value (non numerical values will be skipped).

Typical use case: A model contains a technical chain from server with number of CPU, associated to IT services, associated with application, aggregated in platforms: exporting the two intersection object types, and using the "Sum Values in Rows" for association between servers (with CPU) and IT Services, collecting the CPU for IT Service in the first dataset (association between platform and application), using the IT service as key in "Match keys and pick values" transformation, this output can be obtained:

Platform	CPU per Platform	Application
Billing&Credit Management	6730	CELLOM 2.0
Billing&Credit Management	6730	ADM Credit C
Billing&Credit Management	6730	SAP IS-U QW
Billing&Credit Management	6730	SAP IS-U IB
Billing&Credit Management	6730	SAP IS-U EC
Billing&Credit Management	6730	SAP IS-U EB
Billing&Credit Management	6730	SAP IS-U EG
Billing&Credit Management	6730	CREDIT CAR

1.2.6 Custom Transformation

If it is required to apply a custom transformation the user must define a specific XSLT Library using the XSLT Library Configuration Tool (see before).

The user may define a "**CC Uncompliant**" or "**CC Compliant**" (compliant to CC schema) XSLT Library item.

Please note that the "**CC uncompliant**" custom transformation can't be followed by any other transformation and the workflow operation can only be set to "File Adapter".

"**CC compliant**" transformations are instead not subject to any constraint (besides adhering to CC internal XML schema) when used in workflow transformations.

A number of "CC uncompliant" transformations are included, producing a formatted workflow:

- CC2CSV: produces a csv output file
- CC2HTML
- CC2EXCEL: produces a MS Excel compatible file via XSLT, so that it can be slightly customized
- CC2MSEXCEL: produces an XLSX file through Excel API, not configurable
- CC2OGFF: Produces an Open Group File Format XML output file. Note that this additionally requires the application of a fixed value field (*OGFF_Type*) specifying the target object type.

A special third type is "**CM Compliant**" transformations, which means that the output file is an XML ready to be consumed by the import module CM4Collector (i.e. describing target model, operation, objects metamodel, identity constraints, maps and object instances to be uploaded). Using that one, all the user interface setting for operation will be skipped at workflow execution time.

Please note that saving the entire workflow using the “Save” button will also save the selected transformations.

It’s possible to insert more than one transformation in the same workflow of the same type or of different types. The transformation defined in a workflow will be displayed in the section “Transformation” in the left middle section of the Workflow window.

It’s possible to select one of these transformations to delete it; if the user deletes a transformation in a workflow, the mapping rules applying to the fields that are no longer required (i.e. created by that transformation) will also be deleted.

1.3 Operation

To define the operation of a Workflow follow the steps in the section “Mapping and Operation”:

- Select the operation in the “with operation” drop-down list.
- Optionally, provide the email configuration parameters (see specific chapter for details)

According to the rules of the operation type, the Mapping and Operation section shows the requested parameters as described below.

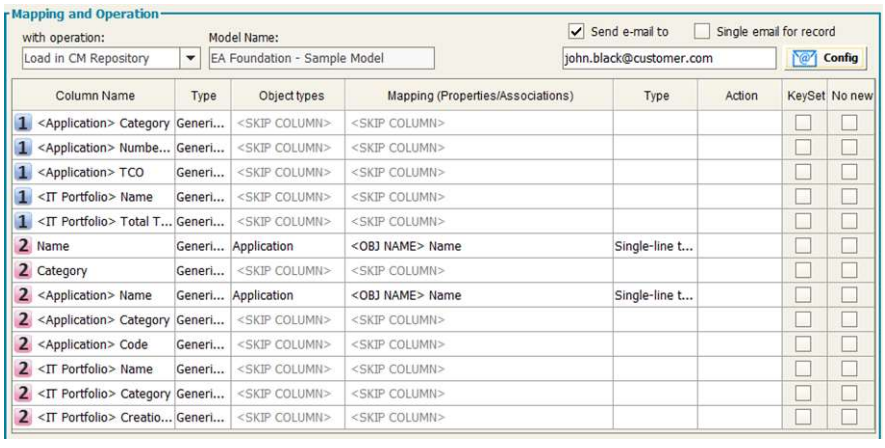
1.3.1 Load in CM Repository

In the “Model Name” field you can see the model linked to the selected configuration.

The user may proceed with the mapping operation in the mapping table, having the requested parameters described in the following list:

Column Name: shows the columns coming from the data source and from the transformation steps; a sequence number that the system assigns to each data source can be seen as prefix to the column name, so the user can easily recognize the source of the data, in the case of multiple data sources.

Column type and format: choose the type and format of the column of the source (String, numeric, boolean, date). This is important if user wants DT to perform a re-format operation on the target column data type.



In particular, to properly import *date fields* with DT, it's important to follow the following rules.

First, it's helpful to configure the query/file in order to have preformatted source data.

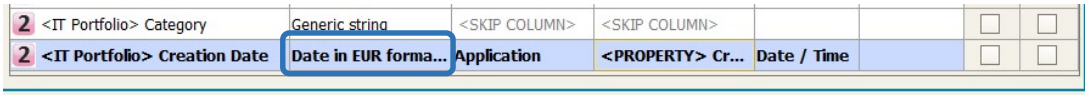
In particular, dates coming from data sources that are mapped onto *CM datetime* property types have to be one of the following:

- *DD/MM/YYYY i.e. 31/08/2016 (EUR FORMAT)*
- *MM/DD/YYYY i.e. 08/31/2016 (USA FORMAT)*
- *YYYY-MM-DD i.e. 2016-08-31 (UTC FORMAT)*

while time, when provided, has to be mandatory in the format:

- *HH24:mm:ss*

This input date format has to be set in workflow mapping step:



Model object type: choose the model object type to load the data; it's possible to select different object types in the same operation

Mapping (prop./assoc.): once you have selected the object type you can map the property or the association between the ones defined in CM for that object type. Please be sure that the "Name" property of an object type is always mapped, in order to allow DT to identify the object instance to work with (except for Association Types imports, see later). Please take into consideration that you can map Unique ID to keep the values when moving objects from a model to another.

Type: shows the type of the mapped column of the target (String, numeric, boolean, date)

Action: user can now choose what to do for a single attribute/association mapping. This means that:

- For multiline property: the user can choose between APPEND, REPLACE or EMPTY model values

- For other data type properties: the user can choose EMPTY model values (not for Name or ID)
- For association type: user can choose between MERGE, REPLACE or EMPTY model values
- For property types:
    - UUID
    - Created By
    - Creation Date
    - Updated by
    - Updated Date

  user is requested to choose between two Actions: KEEP TARGET (default) and KEEP SOURCE.

  KEEP TARGET: target object property is maintained regardless of what is mapped from the user (the property is skipped)

  KEEP SOURCE: target object property is overridden with source property

  UUID, Creation Date and Created By properties cannot be updated.

  KEEP SOURCE action should be used in a federated models environment and is also supported by Synch in CM Repository

Unique Key: users must check, for all the mapped object types, which set of columns must be considered as the unique keyset when performing the upload of the data. Keep in mind that this setting will work according to the usual Corporate Modeler behaviour, where “Name” has to be unique in the object type instances list, while mapping ID or Unique ID allows object name to be updated. For example, the user may check “Name” as key, and all other properties/associations of an existing object with that name will be updated, or a new object with

that name will be created by DT if it is not already in use, or will concatenate it with a sequence number.

Any other mapped column can be included in the keyset, without the “name” – in this case, if an object is identified by the configured keyset, DT will try to update the name with the uniqueness rules described above.

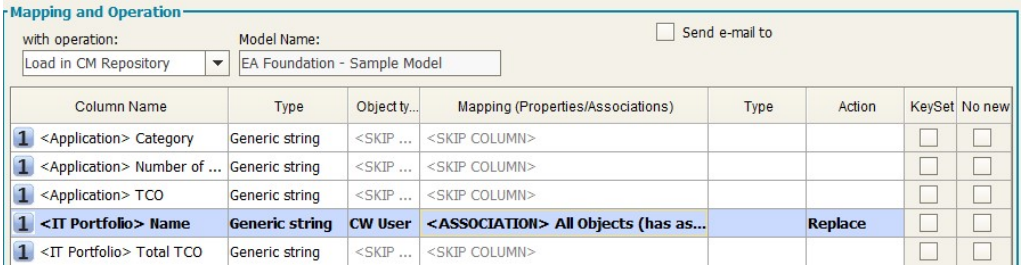
No New: when importing data from an external source, it’s possible that the master list of objects involved in the operation is the one contained in the model. In such a case, the user would want existing objects to be updated (only for the properties used in mapping), but no new objects to be created. If so, user may choose to check “No New” option, on the keyset of the master object type, and this will prevent new objects to be created, while existing will be updated within bounds of mapped properties. The same applies when the object list which has not to be extended is the one related to an object type associated with the master, involved in the operation. In such a case, the user may choose to check the “No New” option, on the record related to the association, and this will prevent new associated objects from being created.

As a result for this operation, data coming from the source adapter will be transformed as defined in “Transformation” step and uploaded into specified model according to the defined mapping rules.

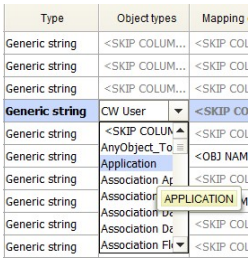
Some additional requirements apply to specific kind of operations, as follows.

1. In case of “*All Objects*” type of associations, differently from other type of associations, the target object type has to be specified, allowing a proper execution of data import. To do this, after mapping a source column to an association to all object, it is then mandatory to map one more specific column:

    - **TARGET_OBJTYPE:** scriptname of the **object type** at the other side of the “all object“ association with the same name, which user wants the objects to belong



To help this mapping, the tooltip of Model Object type, when selecting one, is equal to the SCRIPTNAME of the selected.



This can be added in the source with a fixed value field, with scriptname value, to be then used in mapping.

2. In case of *multiple data source*, it is implied that the different dataset, alternatively:

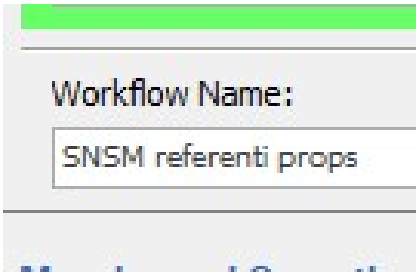
   - Have to be joined, if each of them contains a subset of properties/associations information for the same Object Type. If so, the “Name” property of the object type must be mapped onto each dataset resulting in multiple “Name” mappings – this will be the field used to perform the join operation at runtime on the actual data.
   - Have to be used to map on different object types; if so, “Name” property must mapped only once for each Object Type.

3. In case of *association type*, when the user wants, for instance, to import the association’s properties, it is mandatory to map four specific columns:

   - **CCAboveName**: name of the object **instance** at one side of the association (the one defined as “Source Object Type” at design time)

- **CCBelowName**: name of the object **instance** at one side of the association (the one defined as “Target Object Type” at design time)
- **CCAboveTableName**: scriptname of the **object type** at one side of the association (the one defined as “Source Object Type” at design time)
- **CCBelowTableName**: scriptname of the **object type** at one side of the association (the one defined as “Target Object Type” at design time)

To help the last two mappings, the tooltip of Model Object type, when selecting one, is equal to the SCRIPTNAME of the selected.



4. In case of *Users or User Group* import operation, which is allowed using the Admin Model configuration, the operation, will be executed within the following conditions:

    - New **users** can be created, if:
        - **User Name** and **Logon Name** are both **unique** - records which are uncompliant to this Corporate Modeler requirement are skipped at import time, and logged into the operation log file
    - **Password** cannot be provided
    - **Power Level** has to be **provided**, in a numeric form, and is documented by a tooltip:

      Users - 1 (System Manager), 2 (Project Manager), 3 (Normal User), 4 (Read Only User)
    - No new **user groups** are going to be created; user can only associate **users** to existing user groups
    - The operation **key** has to be one and only one of the following fields:

- **User Name** – in that case, for existing objects, **fields** can be updated, **except for Logon Name, Power Level and Password**
- **Logon Name** – in that case, for existing objects, **name and other fields** can be updated, except for **Power Level and Password**
- **CW ID** - in that case, for existing objects, **name and other fields** can be updated, except for **Logon Name, Power Level and Password**

1.3.2 Delete in CM Repository

When selecting this operation, the only mandatory mapping is the NAME of the object type that the user wants to manage: as a result of this operation, data coming from the source adapter will be *physically deleted* in the specified model accord to the defined mapping rules, for the ones that are not diagrammed (while diagrammed ones will be listed in the operation log file).

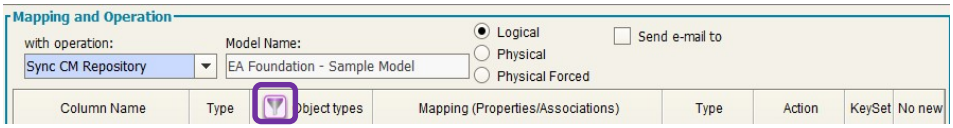
1.3.3 Synch in CM Repository

When selecting this operation, it is mandatory that user choose a *keyset* that's *unique in the source dataset* to avoid undesired results. In particular:

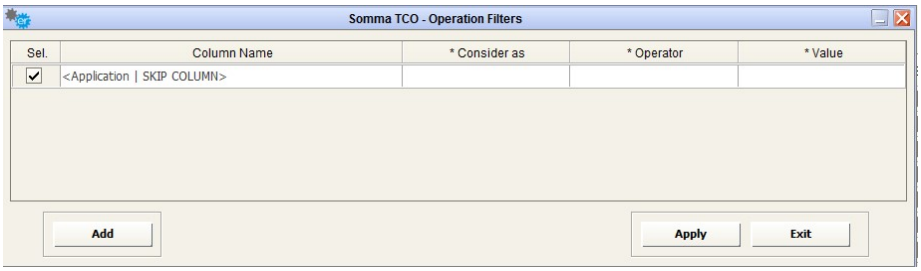
- *Objects coming from the source dataset* will be *inserted/updated* in the specified model accord to the defined mapping rules
- *Objects* already in the specified model object type, that are *not listed in the source dataset*, will be:
    - When "*logical*" option is set, *logically deleted*, and renamed with the prefix "_TO_BE_DELETED_"
    - When "*physical*" option is set, *physically deleted*, if not diagrammed, or renamed with the prefix "_TO_BE_DELETED_" if diagrammed
    - When "*physical forced*" option is set, *physically deleted*, even if diagrammed

If the synchronization operation must be done against a subset of objects, and not the whole set of instances (for example, when contributing external source are more than one for a given object type, each one managing a set of instances, like a CMDB providing data for “Applications in Production environment”), this can be managed using the **sync filters** interface.

Double clicking on the “filter” icon on object type column name

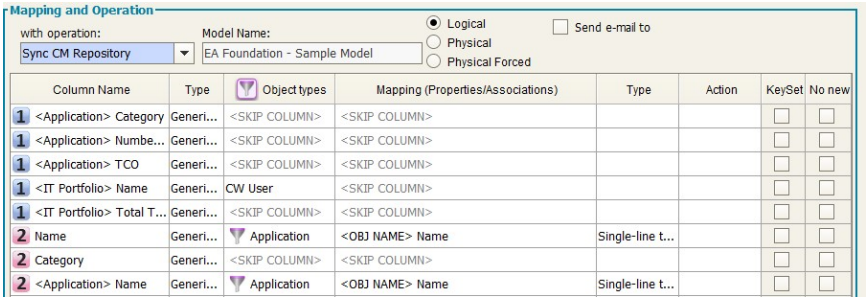


Opens a popup window allowing the creation of sync filters and showing previously defined filters:



User can add, delete (uncheck the filters to be deleted) and apply desired filters.

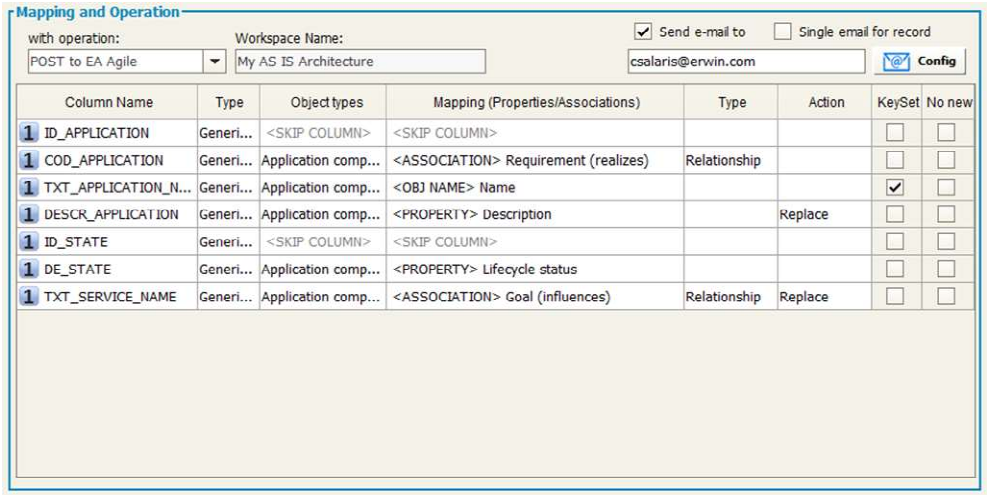
When the sync operation is defined to be executed according to filters, the involved object type name will be marked with a specific filter icon in the map table, the name will be plain otherwise:



For filtered sync option, the load step will follow the usual behavior, while the deletion of redundant object instances will be done within the filtered subset on objects compliant with the filters.

1.3.4 POST to EA Agile

In the “Workspace Name” the name of target workspace will be shown.



User may proceed with the mapping operation in the mapping table, having the requested parameters described in the following list:

Column Name: shows the columns coming from the data source and from the transformation steps; a sequence number that the system assigns to each data source can be seen as prefix to the column name, so the user can easily recognize the source of the data, in the case of multiple data sources.

Column type and format: choose the type and format of the column of the source (String, numeric, boolean, date). This is important if user wants DT to perform a re-format operation on the target column data type.

Date in UTC format

Date / Datetime

The following ISO-8601 formats are supported:

- YYYY-MM-DD (e.g. 2019-01.28)
- YYYY-MM-DDThh:mm (e.g. 2019-01-28T01:02)
- YYYY-MM-DDThh:mm:ss (e.g. 2019-01-28T01:02:03)
- YYYY-MM-DDThh:mm:ssTZD (e.g. 2019-01-28T01:02:03+04:05)

When positing to EA Agile, if a time and offset are not provided, these will default to 00:00Z (i.e. 00:00+00)

Date Ranges

For the supported UTC date formats outlined above, DT supports the following separators:

- $UTCDATE/$UTCDATE
- $UTCDATE,$UTCDATE
- $UTCDATE;$UTCDATE
- start:$UTCDATE end:$UTCDATE
- start: $UTCDATE end: $UTCDATE

Date in EUR format

Date / Datetime

The following formats are supported:

- dd/MM/yyyy (e.g. 28/01/2019)
- dd/MM/yyyy HH24:mm (e.g. 28/01/2019 13:01)
- dd/MM/yyyy HH24:mm:ss (e.g. 28/01/2019 13:01:02)
- dd/MM/yyyy HH:mm AM/PM (e.g. 28/01/2019 01:01 PM)
- dd/MM/yyyy HH:mm:ss AM/PM (e.g. 28/01/2019 01:01:02 PM)
- dd/MM/yyyy HH:mmAM/PM (e.g. 28/01/2019 01:01PM)
- dd/MM/yyyy HH:mm:ssAM/PM (e.g. 28/01/2019 01:01:02PM)

Date Ranges

For the supported EUR date formats outlined above, DT supports the following date range formats:

- start:$EURDATE end:$EURDATE
- start: $EURDATE end: $EURDATE
- $EURDATE;$EURDATE
- $EURDATE,$EURDATE

Date in USA format

Date / Datetime

The following formats are supported:

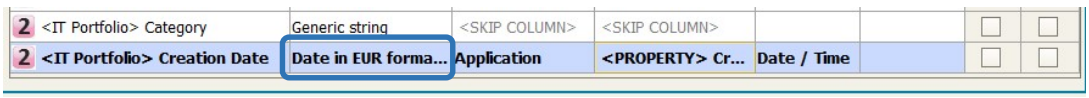
- MM/dd/yyyy (e.g. 01/28/2019)
- MM/dd/yyyy HH24:mm (e.g. 01/28/2019 13:01)
- MM/dd/yyyy HH24:mm:ss mm (e.g. 01/28/2019 13:01:02)

Date Ranges

Dates ranges may be formatted as:

- start:$USADATE end:$USADATE
- start: $USADATE end: $USADATE
- $USADATE;$USADATE
- $USADATE,$USADATE

This input date format has to be set in workflow mapping step:

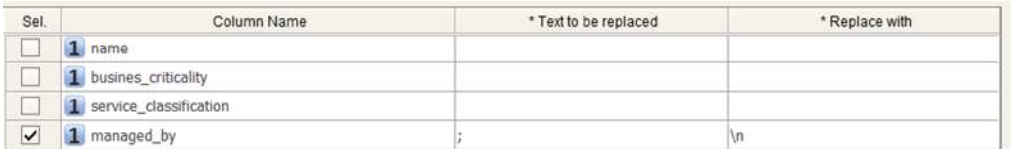


Object type: choose the model object type to load the data; it's possible to select different object types in the same operation

Mapping (prop./assoc.): once you have selected the object type you can map the property or the association between the ones defined in EA Agile for that object type. Please be sure that the "**Name**" property of an

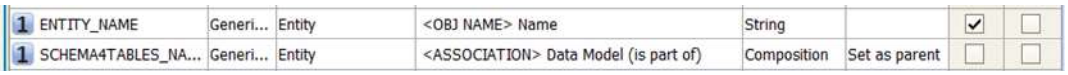
object type is always mapped, in order to allow DT to identify the object instance to work with (except for Association Types imports, see later).

About **associations** mappings, multiple values can be associated putting the different values in a single cell, **newline** separator. Other separators can be replaced with “Data Formatting: Replace Text”:



Sel.	Column Name	* Text to be replaced	* Replace with
☐	name		
☐	busines_criticality		
☐	service_classification		
☑	managed_by	;	\n

If an association is of “composition” type, like “Entity is part of Data Model”, the action “set as parent” must be selected from drop down list in order to properly identify objects parent in the platform.



ENTITY_NAME	Generi...	Entity	<OBJ NAME> Name	String		☑	☐
SCHEMA4TABLES_NA...	Generi...	Entity	<ASSOCIATION> Data Model (is part of)	Composition	Set as parent	☐	☐

Type: shows the type of the mapped column of the target (String, numeric, boolean, date). To help the user in mapping, in case of list or key values allowed values are shown in the “type” cell tooltip:

No New: when importing data from an external source, it’s possible that the master list of objects involved in the operation is the one contained in the workspace. In such a case, the user would want existing objects to be updated (only for the properties used in mapping), but no objects from being created. If so, the user may choose to check the “No New” option, on the keyset of the **master object type**, and this will prevent new objects from being created, while existing will be updated within bounds of mapped properties.

The same applies when the object list which has not to be extended is the one related to an object type **associated with the master**, involved in the operation. In such a case, user may choose to check “No New” option, on the record related to the association, and this will prevent new associated objects from being created.

1.3.5 Delete in EA Agile

When selecting this operation, user should map only the “**Name**” of the object type that he wants to manage: as a result of this operation, data

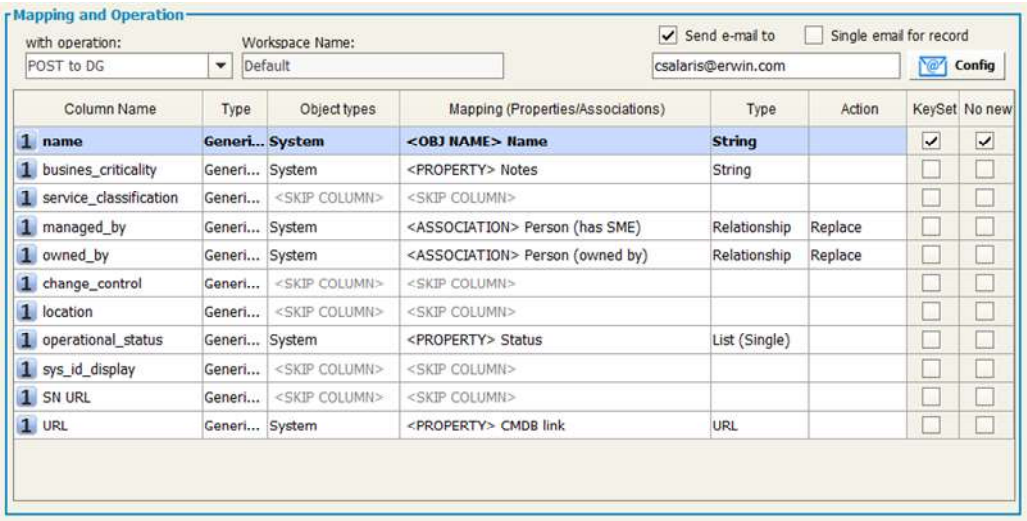
coming from the source adapter will be deleted in the specified workspace according to the defined mapping rules.

1.3.6 POST to EA Agile V3

In the “Workspace Name” the name of target workspace will be shown (“Default”).

User may proceed with the mapping operation in the mapping table, having the requested parameters described in the following list:

Column Name: shows the columns coming from the data source and from the transformation steps; a sequence number that the system assigns to each data source can be seen as prefix to the column name, so the user can easily recognize the source of the data, in the case of multiple data sources.



Column Name	Type	Object types	Mapping (Properties/Associations)	Type	Action	KeySet	No new
1 name	Generi...	System	<OBJ NAME> Name	String		☑	☑
1 busines_criticality	Generi...	System	<PROPERTY> Notes	String		☐	☐
1 service_classification	Generi...	<SKIP COLUMN>	<SKIP COLUMN>			☐	☐
1 managed_by	Generi...	System	<ASSOCIATION> Person (has SME)	Relationship	Replace	☐	☐
1 owned_by	Generi...	System	<ASSOCIATION> Person (owned by)	Relationship	Replace	☐	☐
1 change_control	Generi...	<SKIP COLUMN>	<SKIP COLUMN>			☐	☐
1 location	Generi...	<SKIP COLUMN>	<SKIP COLUMN>			☐	☐
1 operational_status	Generi...	System	<PROPERTY> Status	List (Single)		☐	☐
1 sys_id_display	Generi...	<SKIP COLUMN>	<SKIP COLUMN>			☐	☐
1 SN URL	Generi...	<SKIP COLUMN>	<SKIP COLUMN>			☐	☐
1 URL	Generi...	System	<PROPERTY> CMDB link	URL		☐	☐

Column type and format: choose the type and format of the column of the source (String, numeric, boolean, date). This is important if user wants DT to perform a re-format operation on the target column data type.

Date in UTC format

Date / Datetime

The following ISO-8601 formats are supported:

- YYYY-MM-DD (e.g. 2019-01.28)
- YYYY-MM-DDThh:mm (e.g. 2019-01-28T01:02)

- YYYY-MM-DDThh:mm:ss (e.g. 2019-01-28T01:02:03)
- YYYY-MM-DDThh:mm:ssTZD (e.g. 2019-01-28T01:02:03+04:05)

When positing to EA Agile, if a time and offset are not provided, these will default to 00:00Z (i.e. 00:00+00)

Date Ranges

For the supported UTC date formats outlined above, DT supports the following separators:

- $UTCDATE/$UTCDATE
- $UTCDATE,$UTCDATE
- $UTCDATE;$UTCDATE
- start:$UTCDATE end:$UTCDATE
- start: $UTCDATE end: $UTCDATE

Date in EUR format

Date / Datetime

The following formats are supported:

- dd/MM/yyyy (e.g. 28/01/2019)
- dd/MM/yyyy HH24:mm (e.g. 28/01/2019 13:01)
- dd/MM/yyyy HH24:mm:ss (e.g. 28/01/2019 13:01:02)
- dd/MM/yyyy HH:mm AM/PM (e.g. 28/01/2019 01:01 PM)
- dd/MM/yyyy HH:mm:ss AM/PM (e.g. 28/01/2019 01:01:02 PM)
- dd/MM/yyyy HH:mmAM/PM (e.g. 28/01/2019 01:01PM)
- dd/MM/yyyy HH:mm:ssAM/PM (e.g. 28/01/2019 01:01:02PM)

Date Ranges

For the supported EUR date formats outlined above, DT supports the following date range formats:

- start:$EURDATE end:$EURDATE
- start: $EURDATE end: $EURDATE
- $EURDATE;$EURDATE
- $EURDATE,$EURDATE

Date in USA format

Date / Datetime

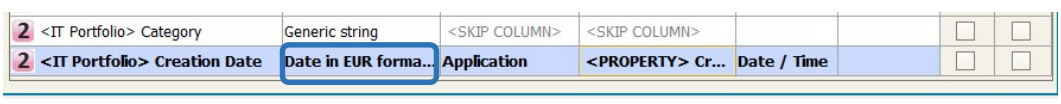
The following formats are supported:

- MM/dd/yyyy (e.g. 01/28/2019)
- MM/dd/yyyy HH24:mm (e.g. 01/28/2019 13:01)
- MM/dd/yyyy HH24:mm:ss mm (e.g. 01/28/2019 13:01:02)

Date Ranges

Dates ranges may be formatted as:

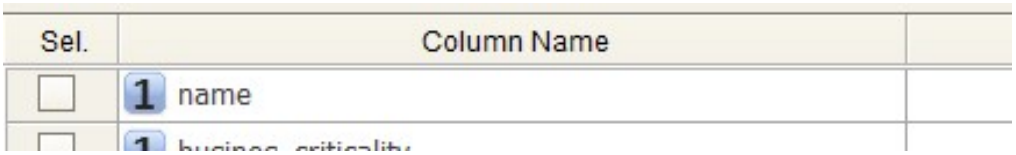
- start:$USADATE end:$USADATE
- start: $USADATE end: $USADATE
- $USADATE;$USADATE
- $USADATE,$USADATE



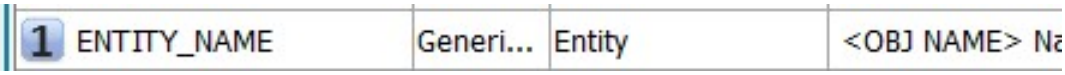
Object type: choose the model object type to load the data; it's possible to select different object types in the same operation

Mapping (prop./assoc.): once you have selected the object type you can map the property or the association between the ones defined in EA Agile V3 for that object type. Please be sure that the "**Name**" property of an object type is always mapped, in order to allow DT to identify the object instance to work with (except for Association Types imports, see later).

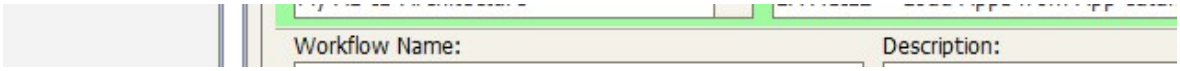
About **associations** mappings, multiple values can be associated putting the different values in a single cell, **newline** separator. Other separators can be replaced with "Data Formatting: Replace Text":



If an association is of "composition" type, like "Entity is part of Data Model", the action "set as parent" must be selected from drop down list in order to properly identify objects parent in the platform.



Type: shows the type of the mapped column of the target (String, numeric, boolean, date). To help the user in mapping, in case of list or key values allowed values are shown in the "type" cell tooltip:

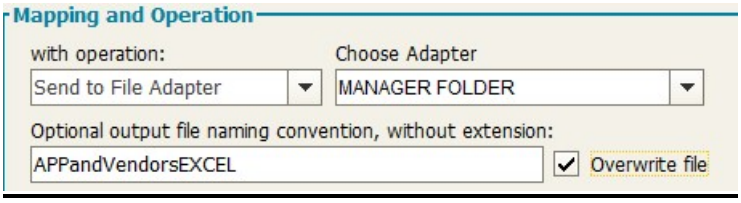


No New: when importing data from an external source, it's possible that the master list of objects involved in the operation is the one contained in EA Agile V3 workspace. In such a case, the user would want existing object to be updated (for the only properties used in mapping), but no objects to be created. If so, user may choose to check "No New" option, on the keyset of the **master object type**, and this will prevent new objects to be created, while existing will be updated within bounds of mapped properties. The same applies when the object list which has not to be extended is the one related to an object type **associated with the master**, involved in the operation. In such a case, user may choose to check "No New" option, on the record related to the association, and this will prevent new associated objects to be created.

1.3.7 Delete in EA Agile V3

When selecting this operation, user should map only the "**Name**" of the object type that he wants to manage: as a result of this operation, data coming from the source adapter will be deleted in the specified workspace according to the defined mapping rules.

1.3.8 Send to File Adapter

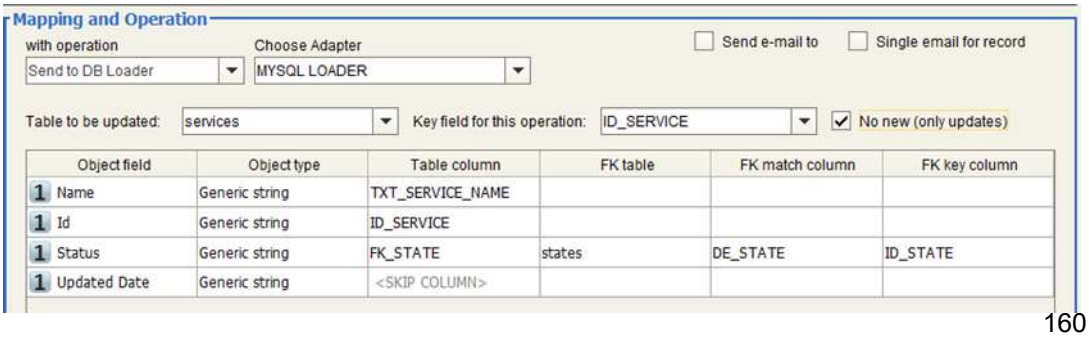


- Choose the *File Adapter* in the "Choose Adapter" drop-down list (only the "PUT" File Adapter will be available in the list)

- Optionally, define a customized name for the ouput file, without the extension (it will be *.xml*, if no final custom transformation is applied; it will be *.csv*, if *Custom transformation (CC uncompliant):CC2CSV* transformation is applied, or *.xls*, for *Custom transformation (CC uncompliant):CC2EXCEL*, or *.html*, for *Custom transformation (CC uncompliant):CC2HTML*, etc). Moreover, the user can choose between overwriting file if existing, or have a different file for different executions (in this case, the name will be post fixed with a timestamp)

As a result of this choice, data coming from the source adapter will be transformed as defined in "Transformation" step and sent to the File Adapter folder. When completed it will be delivered to the destination.

1.3.9 Send to DB Loader



- Select the *Table* to be uploaded with data, coming from the source and transformations of the workflow, among the ones owned by the user specified at DB Loader Adapter configuration time.
- Select the *Key column* among the ones describing the Table, to be used to allow DT to properly perform an “INSERT” or “UPDATE” SQL command at runtime, for any given field value occurrence in input data.
- Then, for each input column name the user may alternatively:
  1. Map onto a specified column of the previously chosen table, when the input value is natively hosted by that
  2. Map onto a specified column of the previously chosen table, when this contains the foreign key of another table in the same database, actually containing the input values, providing:
     - The name (“FK table”)
     - The column hosting the input values (“FK match column”)
     - The column hosting the foreign keys for given table (“FK key column”)
- The option “*No New (only updates)*” allows user to set import operation to be limited to updates on selected table

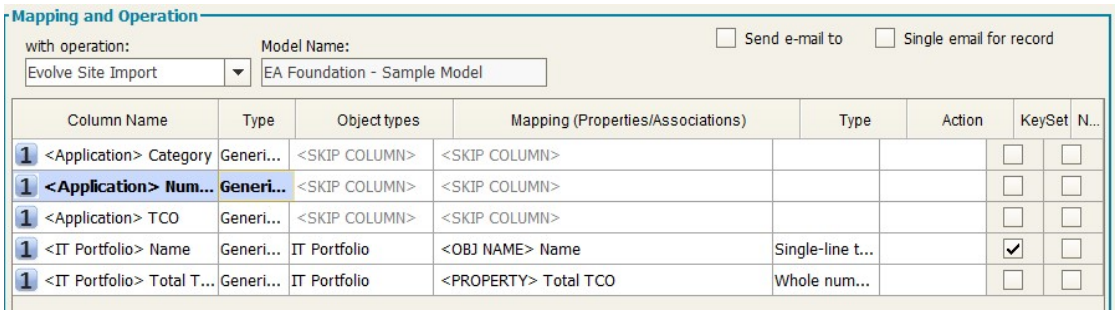
As a result of this choice, data coming from the source adapter will be transformed as defined in “Transformation” step and sent to the DB Loader Adapter for the upload operation previously defined.

1.3.10 Web Service POST

- Choose the *Web Service POST Adapter* in the “Choose Adapter” drop-down list (only the Web Service “POST” Adapters will be available in the list)

As a result of this choice, data coming from the source adapter will be transformed as defined in “Transformation” step and sent to the Web Service, as defined in its configuration.

1.3.7 Evolve Site Import



In order to copy Evolve sites from model to model, you should consider that:

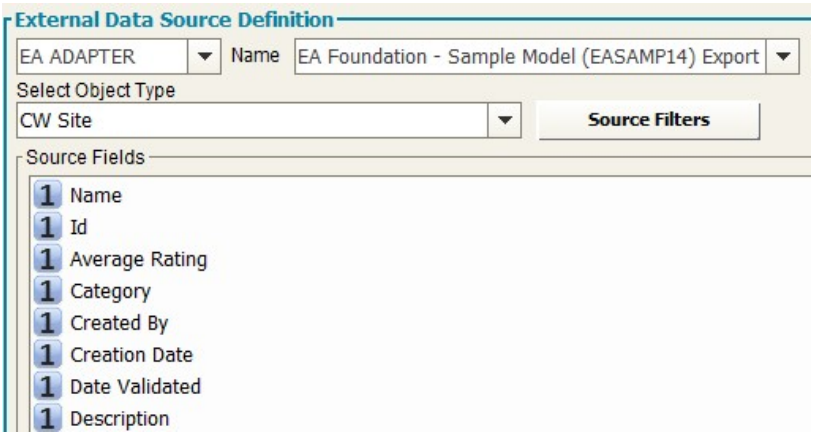
- A target site has to be created before executing the operation
- Only the following components are copied from source model to target model: Menu, Index Pages, Object Pages, Diagram Popouts, Diagram Designer Layouts
- If you want to manage Diagram Designer Layouts you should enable target model with Diagram Designer
- If any objects is already defined in the target site it will be overwritten
- It is recommended that all the pages linked to source Evolve site, should not be associated to other Evolve sites

The following instructions have to be followed in order to copy an Evolve Site. It is recommended to back up your target model before the operation, as a bad configuration may result in a loss of information.

1 Create a new workflow configuration on your target model configuration

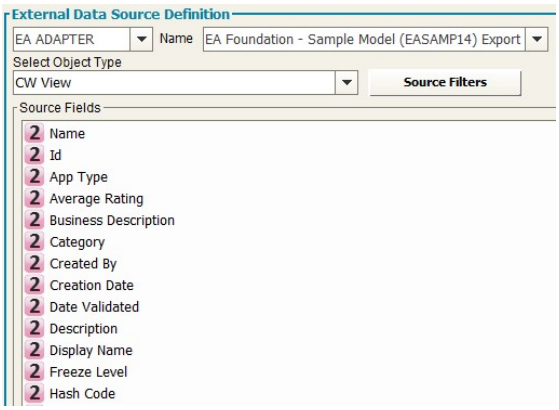
2 Select as source EA Adapter and select the export adapter related to your source model

3 Select **CW Site** as object type and filter only Name and Description properties



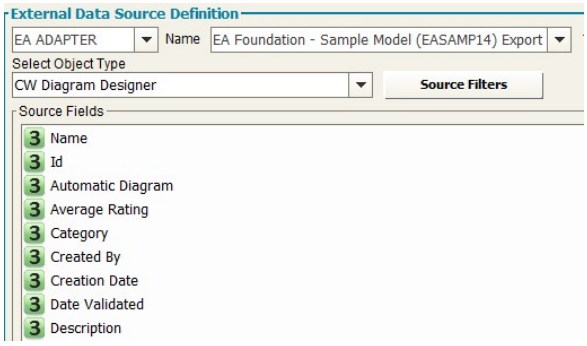
4 Add a second source EA Adapter and select the export adapter related to your source model: select **CW View** as object type and filter the following properties/associations:

a. Name
b. App Type
c. Business Description
d. Category
e. Description
f. Display Name
g. Hash Code
h. Root Object Type
i. CW Site (belongs to (index))
j. CW Site (belongs to (single))
k. CW Site (belongs to (diagram pop out))



5 (Optionally, only if you want to copy **Diagram Designer Layouts**) Add a third source EA Adapter, select the export adapter related to your source model and filter the following properties:

a. Name
b. Automatic Diagram
c. Category
d. Description
e. Enabled Version
f. Root Object Type



6 Add a Fixed Value Manager transformation on the following columns:

Src. Number	Column Name	Consider as	Operator	Value
1	Name	String	Equal	source_site_name

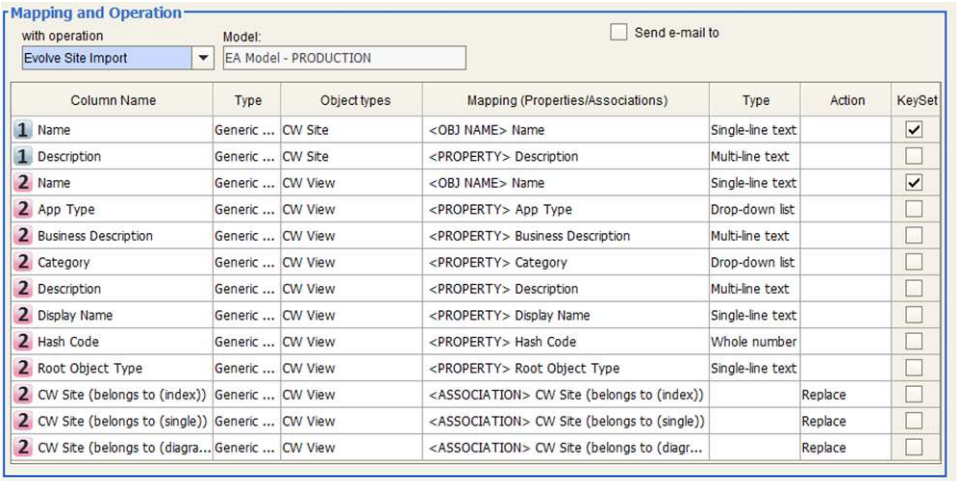
2	CW Site (belongs to (index))	String	In (semicolon separated)	;source_site_name*
2	CW Site (belongs to (single))	String	In (semicolon separated)	;source_site_name*
2	CW Site (belongs to (diagram popout)	String	In (semicolon separated)	;source_site_name*

*: use semicolon in value cell as specified

7 Add a Replace Text transformation on the following columns:

Src Number	Column Name	Text to be replaced	Replace with
1	Name	source_site_name	target_site_name
2	CW Site (belongs to (index))	source_site_name	target_site_name
2	CW Site (belongs to (single))	source_site_name	target_site_name
2	CW Site (belongs to (diagram popout)	source_site_name	target_site_name

8 Choose the operation **Evolve Site Import** that will automatically map all needed fields



1.3.11 Email Configurations

The email notification allows the user to be easily aware of what's going on with DT operations, particularly for the owners of model information managed by the tool.

Notification email for the workflow operation results can be configured with a custom setting for recipients, CC recipients, email subject and message text.

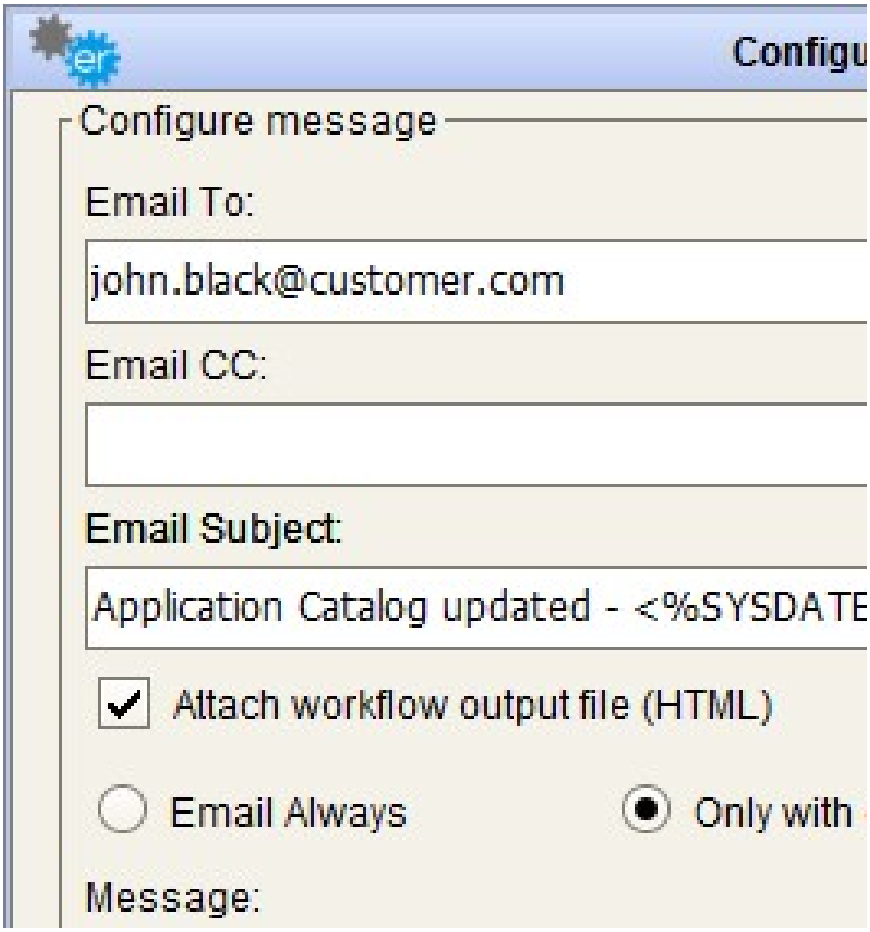
The workflow output, in HTML format, will be attached to email. If it is not needed – for example, the workflow is configured to produce a file that will be also attached to email – the option “Attach workflow output file” should be deselected:

another option is provided to send email always (per default), only with data, or only on errors:

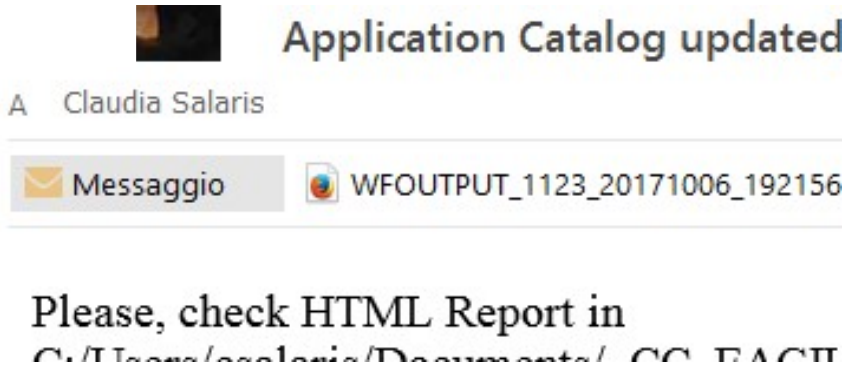
Some parameters can be used in subject or message text:

- *<%SYSDATE%>,<%SYSDATE_EUR%>,<%SYSDATE_USA%>, <%SYSDATE_UTC%>:* replaced at execution time with date time, using requested format.
- *<%OPERATION_FOLDER%>:* replaced at execution time workflow operation folder path.
- *<%WORKFLOW_FOLDER%>:* replaced at execution time workflow folder path.

The email can contain links to specified URL: just use “Add URL” button to have a sample link text and replace your own site address and name:



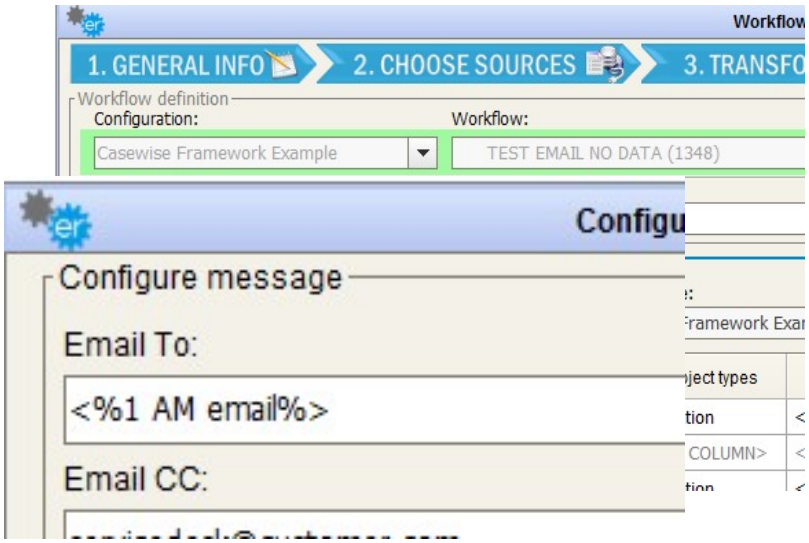
The resulting email looks like this:



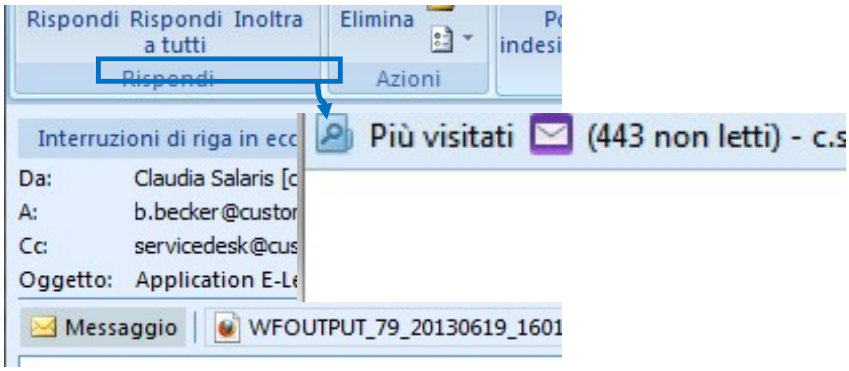
It is possible to use the option "*Single email for record"* to send a notification for each input dataset record.

Users may put parameters in the email configuration using column name to give specific information regarding the recipient: to be recognized by DT as a parameter, a column name must be enclosed between a prefix "<%" and a suffix "%>".

For instance if the first dataset of a workflow has a column named "**1 AM email**", containing the application manager email related to the column "1 Application name", a proper email recipient could be configured in the email as ***<%1 AM email %>*** and the subject of the message could be:
*"Application **<%1 Application name%>** has been updated in EA model"*



This setting, when the "*Single email for record"* option is selected, produces a single notification per source dataset record, replacing parameters with their value as read from the input.



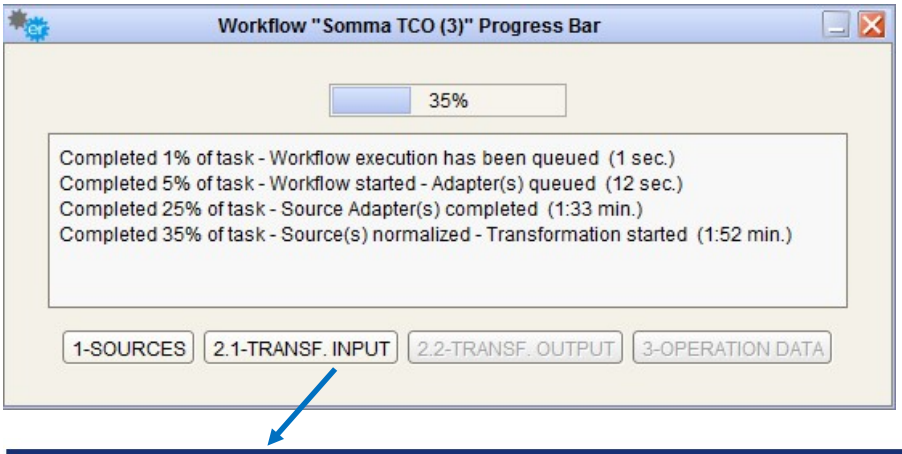
2 Workflow Test

Once configured, the workflow can be tested using the button "Test", even if is not yet active.

When tested, a progress bar will make the user aware of the major workflow steps:

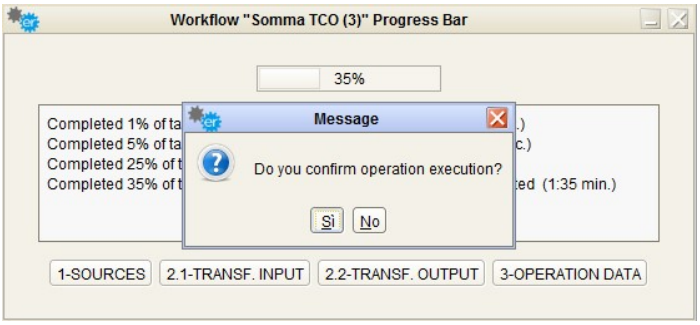
1. Adapter execution
2. Source data normalization – to provide transformation input data
3. Transformation execution
4. Workflow output normalization – to provide data for operation

As soon as the step is performed and corresponding data is produced, a specific button makes it available to the user.

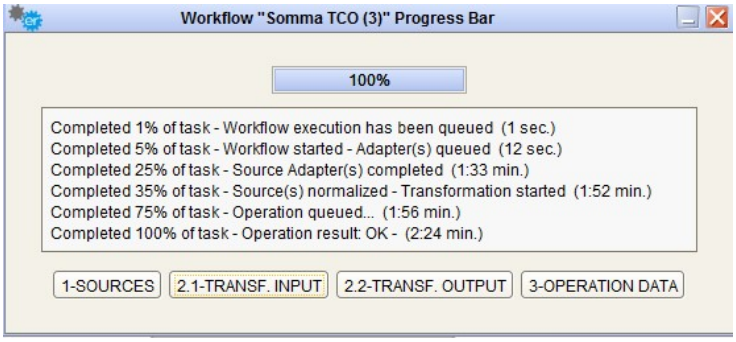


Corporate Collector

<Application> Category	<Application> Number of Users	<Application> TCO
Legacy	40	7500
ERP	402	100000
Legacy	200	78000
Middleware	150	220000
Software	150	30000
ERP	110	50000
ERP	150	7500
ERP	20	80000
Legacy	152	7500
ERP	370	200000
Software	300	65000



The user is then informed about the result of the operation execution, if requested.



Chapter 5

1 Administrator Tools

1.1 Execute and Schedule Jobs

The administrator tools include the scheduler, which is the engine that runs the jobs mainly related to the adapters set up by the user. The scheduler allows the user to schedule a single job; the user may define and schedule job sequences that run the desired jobs with the assigned priority.

According to the type of jobs available and the timing facilities, the user may schedule each single job (or a job sequence) with a certain frequency.

In this window the user can choose to manage the jobs available for all the configurations or to select a specific model configuration to work on the model administration jobs. In particular, selecting from the upper dropdown list:

- "*All Configurations*", the user can schedule the Custom Job (see related section) and the Adapters (CM, WS, File and DB Adapters) which may be used by different models within their specific workflows. In this case, when an adapter is executed by the scheduler engine, an embedded workflow manager will trigger all the workflows using that adapter as its data source to execute

  For each of the available Adapters, the user can set up the definition of single job to be scheduled (section "Job Scheduler")

- *A specific model configuration name*, the user manages the scheduling of the given model administration jobs, as

  - Model backup

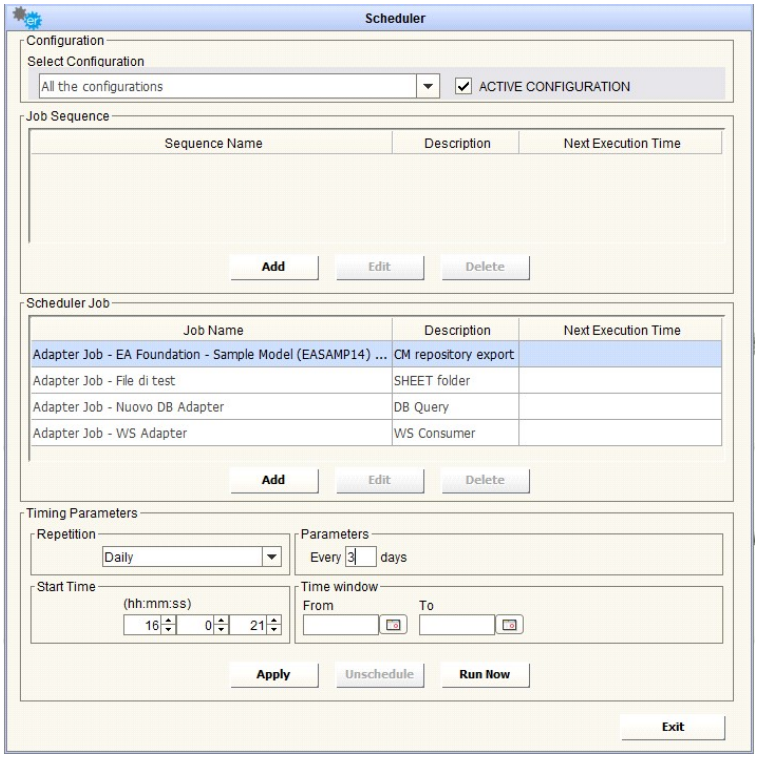
- Multisource jobs defined within that configuration workflow (thus needed only for those and not available for any other)

For each of the available model configuration, the user can set up:

- If it is active or not (checkbox “Active configuration”) – a configuration could exist even if it is not active
- The definition of single job to be scheduled (section “Job Scheduler”)
- The definition of each job sequencer (“Job Sequencer”)

1.1.1 Job Scheduler

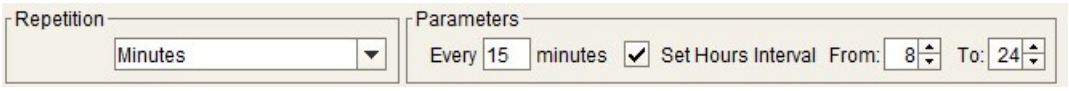
For any kind of job, the user can set the scheduling time for each job after selecting it from the list.



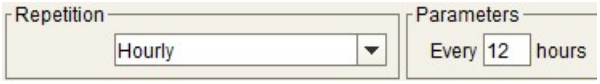
Here are the timing parameters available:

- *Time Window* - gives the opportunity to set a date interval for job execution. The jobs are executed according to the following repetition options and set-up time field.

- *Repetition by "Minutes"* - set the repetition of a job every "xx" minutes (optionally within an hour interval).



- *Repetition "Hourly"* - set the repetition of a job every "xx" hours (12 hours in the example).

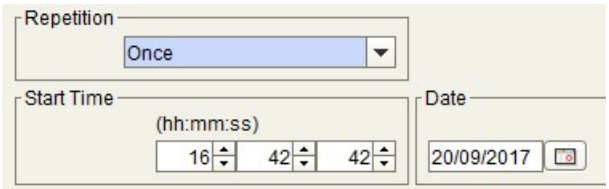


- *Repetition "Daily"* - set the repetition of a job every "xx" days (2 days in the example).

- *Repetition "Weekly"* - set the repetition of a job each "day" of the week (Sunday and Monday in the example).

- *Repetition "Monthly"* - the repetition of the job every "xx" (ordinal number) of the month (first day in the example).

- *Repetition "Once"* - schedule the job to run just once, in the specified date and accordingly to the "Start Time" field.



When the user has completed the scheduling configuration, the user can store the scheduling parameters in the DT database by pressing the Apply button.

When the user presses the “Apply” button, either if they selected a job sequence or a single job, the field “Next Execution Time” (in the Job Sequencer or Job Scheduler respectively) is filled with the next expected time the job will run.

If it is required to unschedule a job, the user has to select it in the list and then press the Unschedule button.

If it is required to execute a job or a job sequencer immediately, the user has to select it in the list and then press the Run Now button.

The Exit button closes the window.

1.1.2 Active Configurations

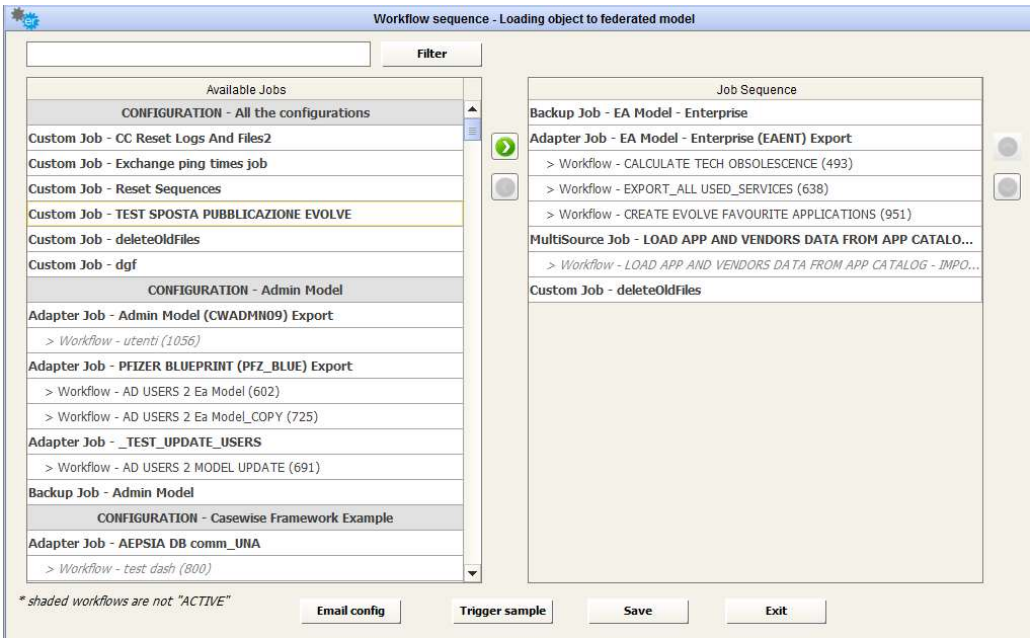
If you want the scheduler to execute the jobs at the scheduled time, activate the configuration by checking the ACTIVE CONFIGURATION check box. Once activated the check-box changes colour from yellow to green.

Please note that a configuration does not necessarily need to be activated in order to schedule its jobs. The user is allowed to schedule any of the jobs, but the jobs will not be executed if at the execution time the configuration is not checked as “active”.

The “All the configurations” configuration is always active:

1.1.3 Define Jobs Sequences

The user may define job sequences, configuring *adapters and workflow* in a real *logical flow*; the user may choose between available adapters and related workflows which ones, and which order they belong in the Job Sequence:



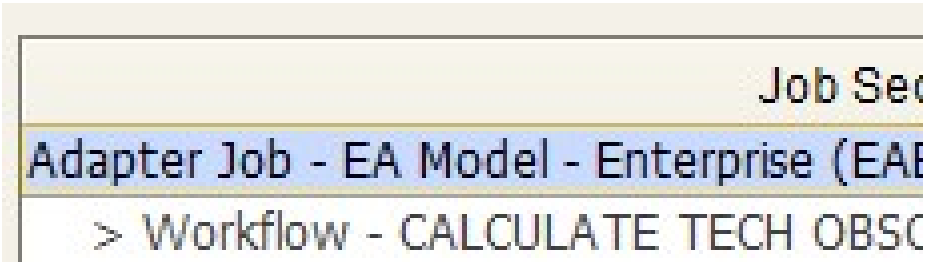
Using right/left arrows to include/ exclude an item, and up/down arrow to change their order.

User may include more than one occurrence of adapter, with different workflows, backup and custom jobs.

You can schedule a job using the standard timing parameters: the job will result in the execution of all the adapters belonging to the job sequencer definition and all of the selected workflow, in the configured order.

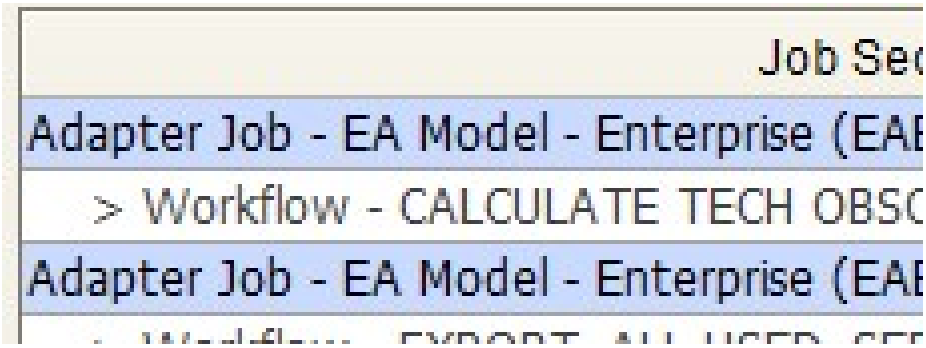
Please take into consideration that if a workflow included in a sequence is set *inactive* in the workflow configuration, then it will still belong to the sequence, but it will appear "*grayed out*" in the sequence definition window, and its execution will be skipped when running the sequence itself.

Take into account the following internal behavior: when different workflows use source objects exported from the same model:



All the different requests will be aggregated, in terms of **different object types**, and in terms of **filters** (which attributes/associations are to be exported; the definition of filters on data).

So, if the objects to be exported, all at once, are typically more than a few thousand, or the filters are conflicting, it's recommended to replicate the adapter execution, moving workflows in order to make the adapter request more efficient and avoid conflicts:

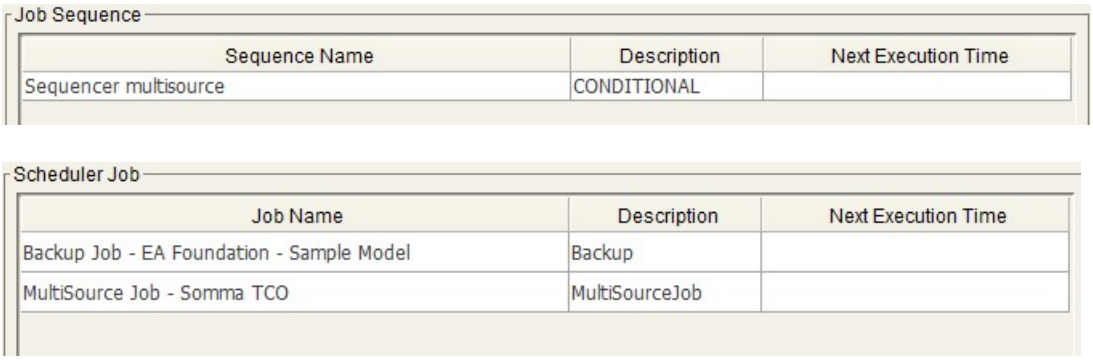


This is not mandatory and a single rule of workflow aggregation cannot be provided, as it hugely depends on the size/type of required export.

Sometimes, if DT Windows Services are forced to stop, or in case of severe internal error, sequences may enter an inconsistent state, that could stop other jobs from being executed. In that case the administrator may use the "**Reset sequences**" item in the "Tools" menu (view Chapter 9 – Utilities).

1.1.4 Multi Source Jobs

When the user in a workflow configuration defines more than one source (workflow with multi source) the system creates a “special” job named “MultiSource Job” that will be shown in the Job Sequencer and Job Scheduler sections of the specific workflow configuration:

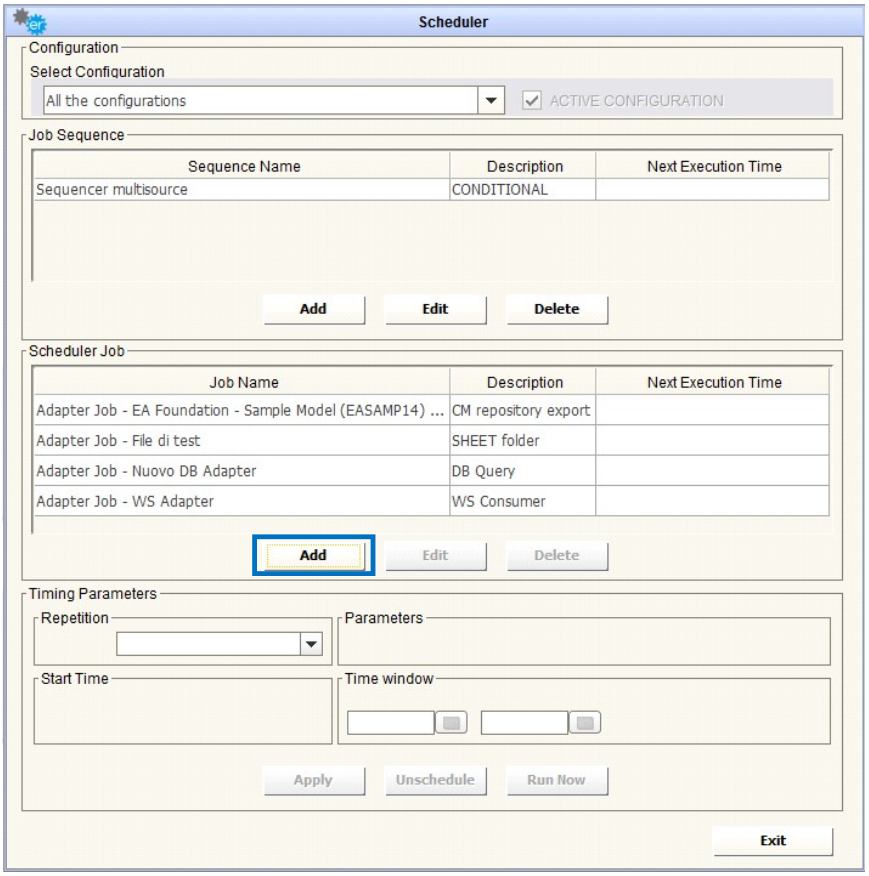


You can schedule it using the standard timing parameters: the job will result in the execution of all the adapters belonging to the source definition of the workflow that will the trigger the workflow execution itself.

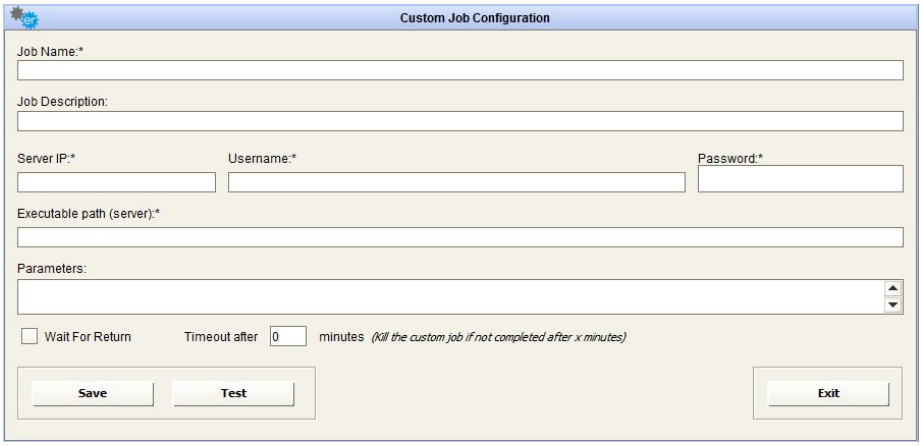
Workflows that have any of those adapters as single source or in a different multisource definition will not be triggered.

1.1.5 Custom jobs

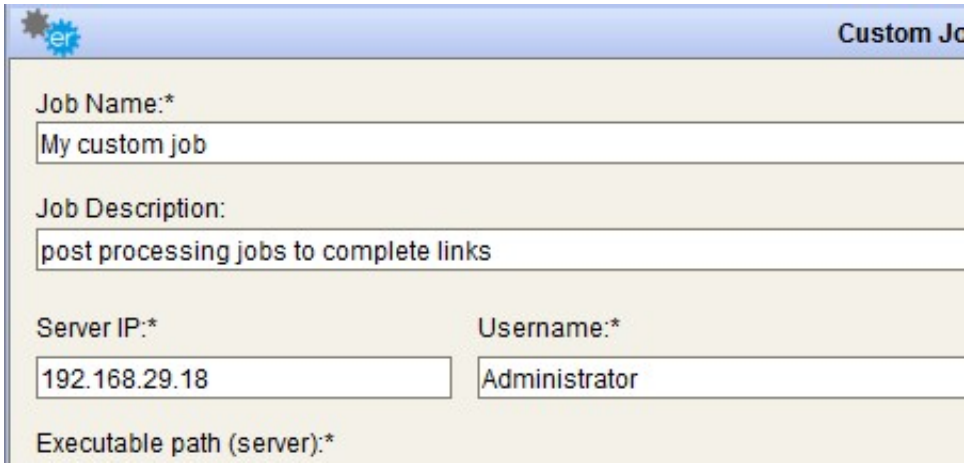
If needed, users can configure a custom job, aimed to run any external executable that can be launched through a command line, using the “Add” button at the bottom of the job list, available with the “All the configurations”.



To configure a Custom Job, beside the Job Name and Description, the user must provide the environment information, such as the IP of the server in which the execution has to be launched, the username and password of a Windows account (as specified in the Windows domain) which has the grant to login to the server and run the execution.



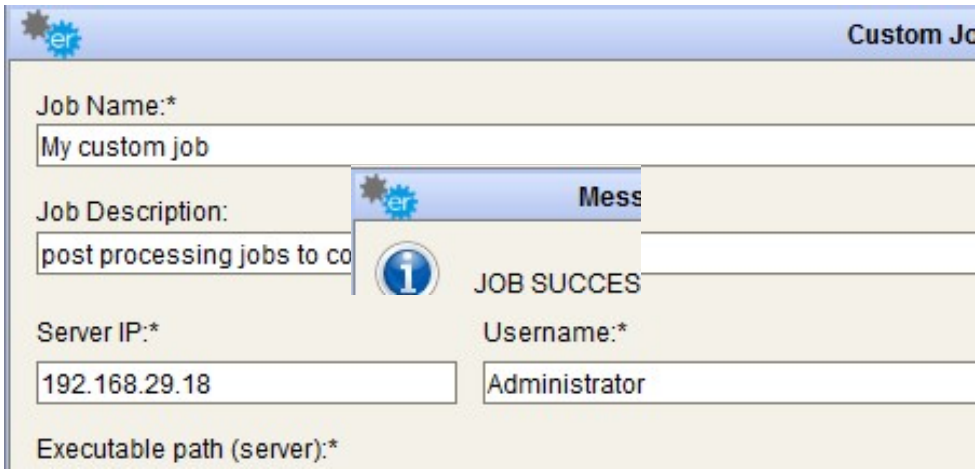
The user must provide the physical path, including the file name, of the given execution in the server, and the optional parameters required by the execution itself, enclosed by quotes if needed, as they would be written in a command line.



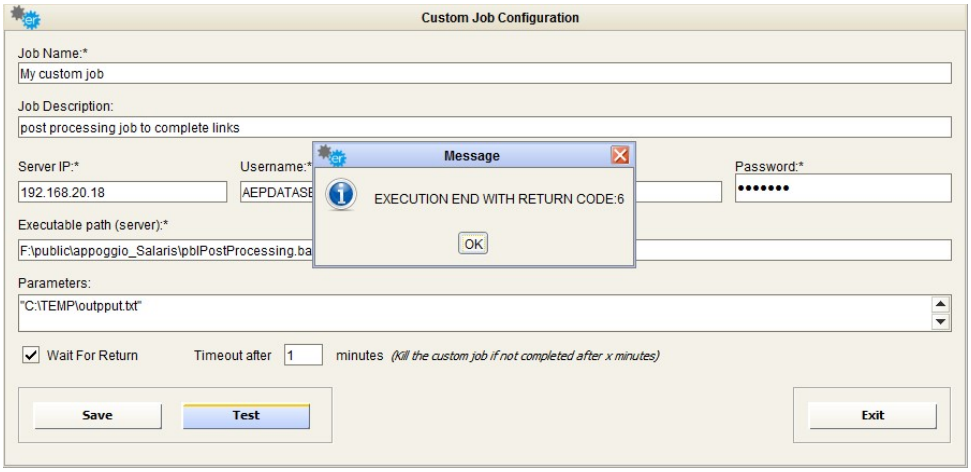
The option “Wait For Return” forces the process that launches the execution to be listening for the external process to end and wait for the exit code. In this case it’s mandatory to define the timeout in minutes, to avoid DT hanging, if the executable does not exit in the given elapsed.

DTWhen “Wait For Return” is not checked, user may define a number of minutes that DT will wait before scheduling the next job, if the custom job is included in a job sequence.

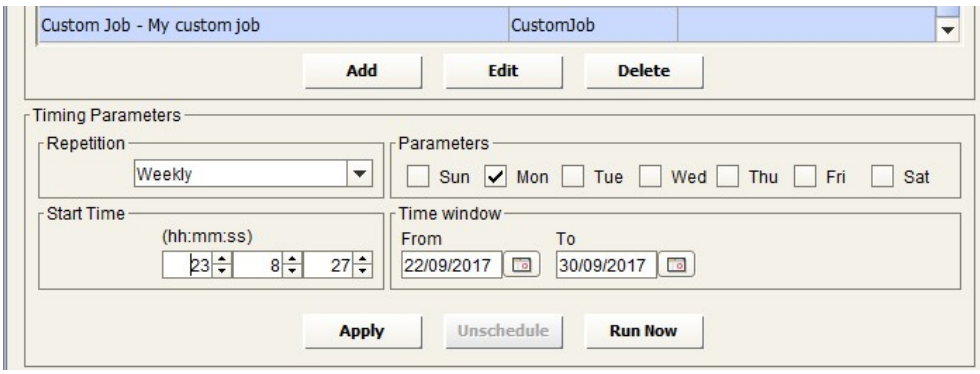
After saving the configuration, user can test the execution of the job:



In case of error, DT will provide the return code of the process execution.



You can schedule it using the standard timing parameters: the job will result in the launch of the external execution.



Chapter 6

1 Running DT on Event

1.1 Running an Adapter on Event

In a real life environment, it could be important to acquire external data on a scheduled basis, or in an "*on event*" mode; in particular, when a DB Adapter is used for reading the information from a database. It's quite easy to implement, for example, a stored procedure that drops a text file, containing the trigger information for DT to start a given adapter.

The trigger file can be detected by DT if dropped in a given folder:

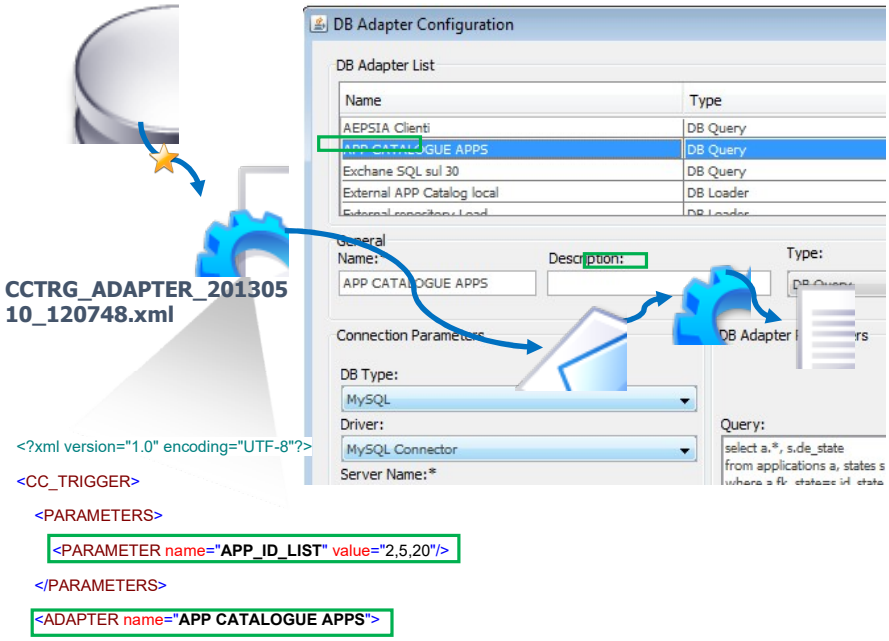
- any of the DT *adapter folders* (including the specific adapter, but not mandatory)
- the folder "*C:\ProgramData\erwin\Data Transformation\workflows*"

The described *triggering feature* applies to any other adapter type.

The trigger files must be compliant with the naming convention:

CCTRG_ADAPTER_YYYYMMDD_HHMMSS.xml

and the sample content describing the schema is detailed below (see next paragraph for <*PARAMETERS*> tag):



Please note that you could even choose whether to execute all the (active) workflows defined against that adapter as source, or a sub list of your choice:

```
<?xml version="1.0" encoding="UTF-8"?>
<CC_TRIGGER>
   <PARAMETERS>
        <PARAMETER name="APP_ID_LIST" value="2,5,20"/>
  </PARAMETERS>
  <ADAPTER name="APP CATALOGUE APPS">
        <DELAY HH="0" MIN="0" SEC="30" />
        <WORKFLOWS all="false">
              <WORKFLOW name="CCWORKFLOW_1">
                    <DELAY SEC="0" MIN="0" HH="0"/>
              </WORKFLOW>
        </WORKFLOWS>
  </ADAPTER>
</CC_TRIGGER>
```

1.1.1 Running a DB Adapter on Event with Parameters

When configuring and using a trigger file for executing a DB adapter on event, it is possible to add parameters to drive the query resultset through them.

Let's suppose that you want to configure a DB Query Adapter, which is named "MY ADAPTER" and extract detail of just the Applications that have been updated / inserted in a table named "MY_TABLE". The Applications are selected, based on their ID.

The SQL query to do this would look like:

*SELECT * from MY_TABLE where APPLICATION_ID in (131, 156, 653)*

When triggering DT on event, you may want this adapter to just process applications that were updated or inserted in the DB table. The IDs of the updated / inserted Applications can be obtained and passed to the adapter using parameters in the SQL query. e.g.

*SELECT * from MY_TABLE where APPLICATION_ID in (<%APP_ID_LIST%>)*

<%APP_ID_LIST%> in the SQL query is the parameter name to obtain the name of the parameter in the XML trigger file.

Please note the required syntax for DT: parameter names must be provided enclosed *by the prefix "**<%**" and suffix "**%>**",* like: *<%PARAM_NAME%>.*

The values of the parameter must be written in the XML trigger file each time that the SQL Trigger executes on the Table where the Application detail will be updated / inserted.

The XML trigger file must contain the following:

<PARAMETERS>

<PARAMETER name="APP_ID_LIST" value="345, 436"/>

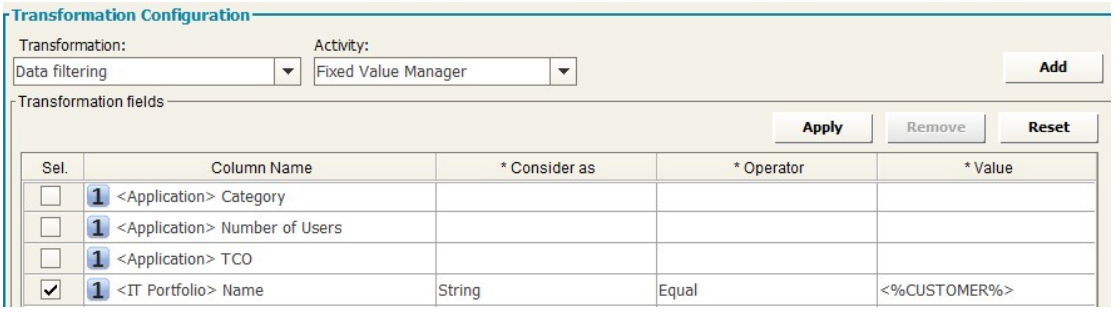
</PARAMETERS>

The parameter name is hard coded into the SQL Trigger, and the values are the IDs of the changed Applications which is written directly into the xml trigger file by the SQL Trigger. These values can be different, each time that the database trigger is executed.

When the DT Database query is executed, the values will be put in place of the parameter name, as follows:

SELECT * from MY_TABLE where APPLICATION_ID in (345, 436)

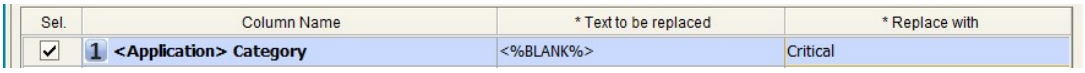
Parameters included in the trigger file can be used even in workflow transformations, when they accept fixed values, like:



Special parameters include:

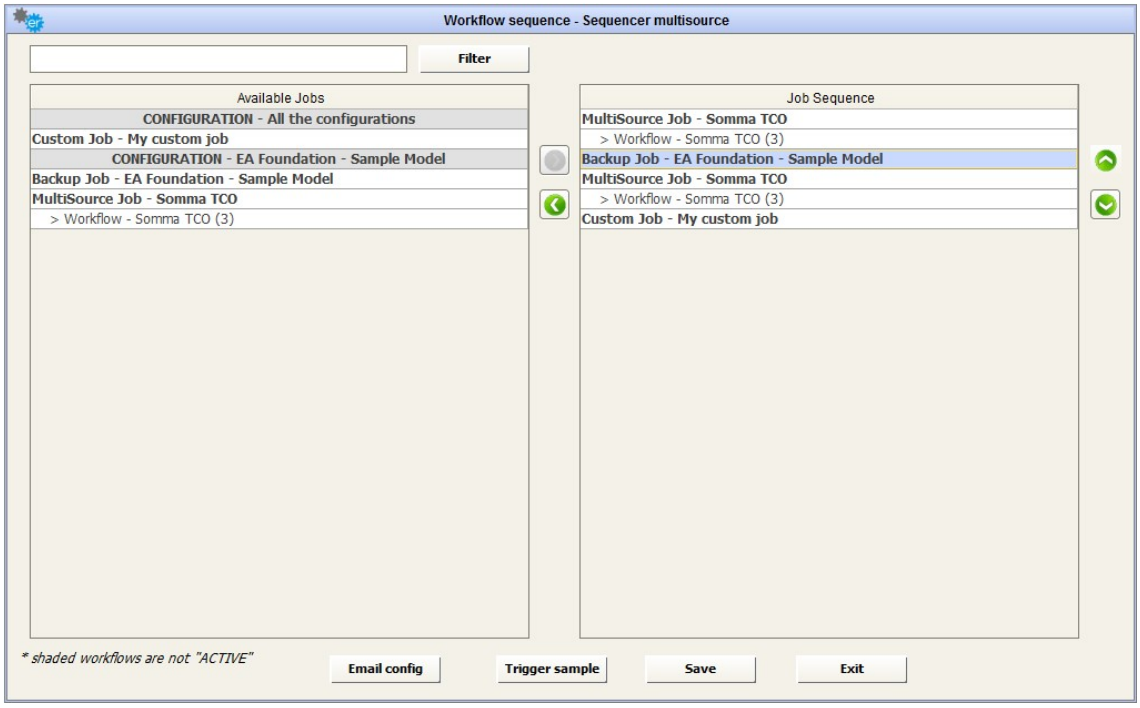
***<%SYSDATE%>**:* a parameter can be used to populate a fixed value column, or a filter value, with the system date and time (*UTC format*); if the value <%SYSDATE%> is entered, DT will replace the parameter with the datetime value, at execution time.

***<%BLANK%>**:* in Replace Text transformation, when you want it to be applied to an empty value.



1.2 Running a Sequence on event

In this version, you can define job sequences, not only on schedule time basis, but in a real logical flow; that is, the user may choose between available adapters and related workflows; which ones, and in which order they belong in the Job Sequence:



The job sequences can be launched on demand, on schedule, or on event in a similar way to adapters.

A sample of the needed trigger file could be obtained pushing the “Trigger sample” button:

```
<?xml version="1.0" encoding="UTF-8"?>
<CC_TRIGGER sequence="MySequenceJob">
    <PARAMETERS>
      <PARAMETER name="MY_CATEGORY_PARAM" value="sample_value" />
    </PARAMETERS>
</CC_TRIGGER>
```

And it will contain all the parameters included in the included adapter/workflow definitions (where “*sample_value*” must be replaced by the actual "*MY_CATEGORY_PARAM*” value when producing the real trigger file).

The trigger files must be compliant to the naming convention:

CCTRG_SEQUENCE_YYYYMMDD_HHMMSS.xml

Chapter 7

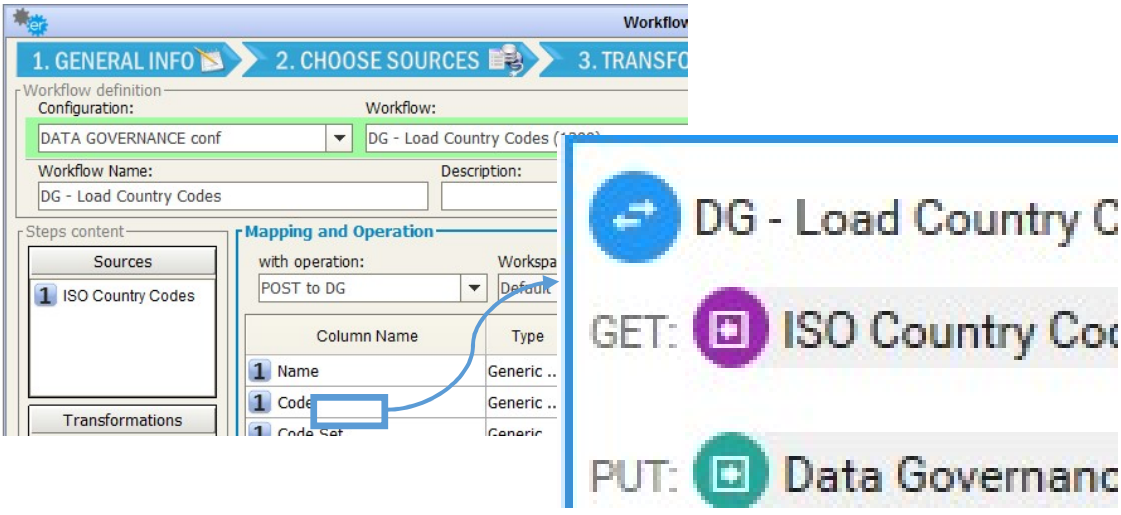
1 Running DT from Cloud Platform

Collector4Cloud provides EA Agile/EA Agile V3 users the capability to interact with onPrem DT:

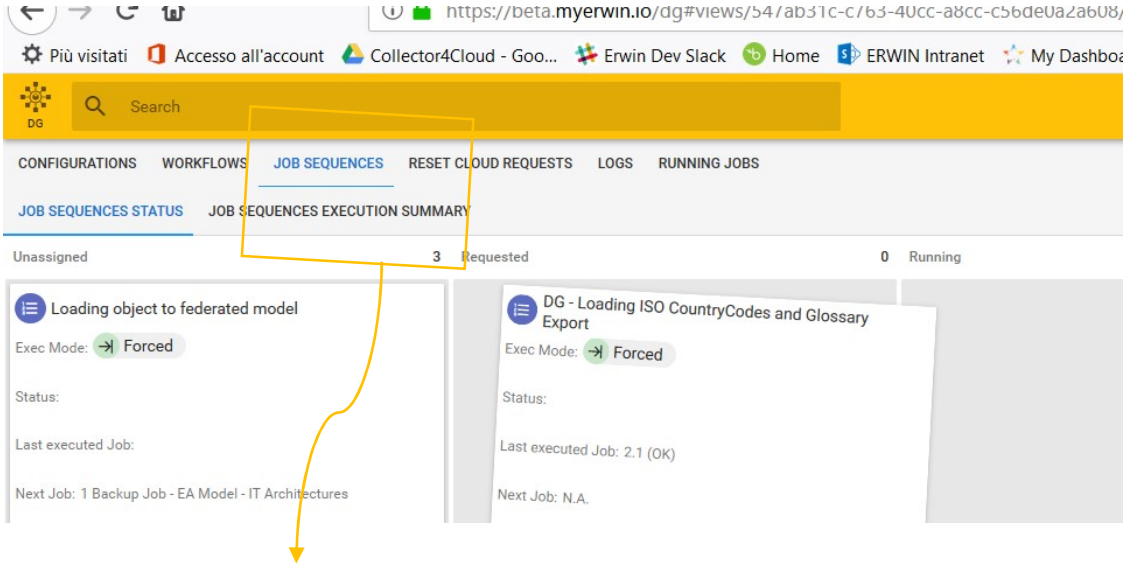
- View available **DT flows / job scheduler**
- View the **workflow definition**, in terms of sources/target Adapters and configuration
- View the **Job Sequence definition**, in terms of included workflows and custom jobs
- **Run** a specific **flow or job sequence on demand** or **by schedule**, following the execution status
- view the **data managed by the flow** to the target system / people
- **check** if the flow exited with **no data**
- **check** if there were **errors**, and view the **logs** in case
- Have a complete updated view of the **daily operations log**

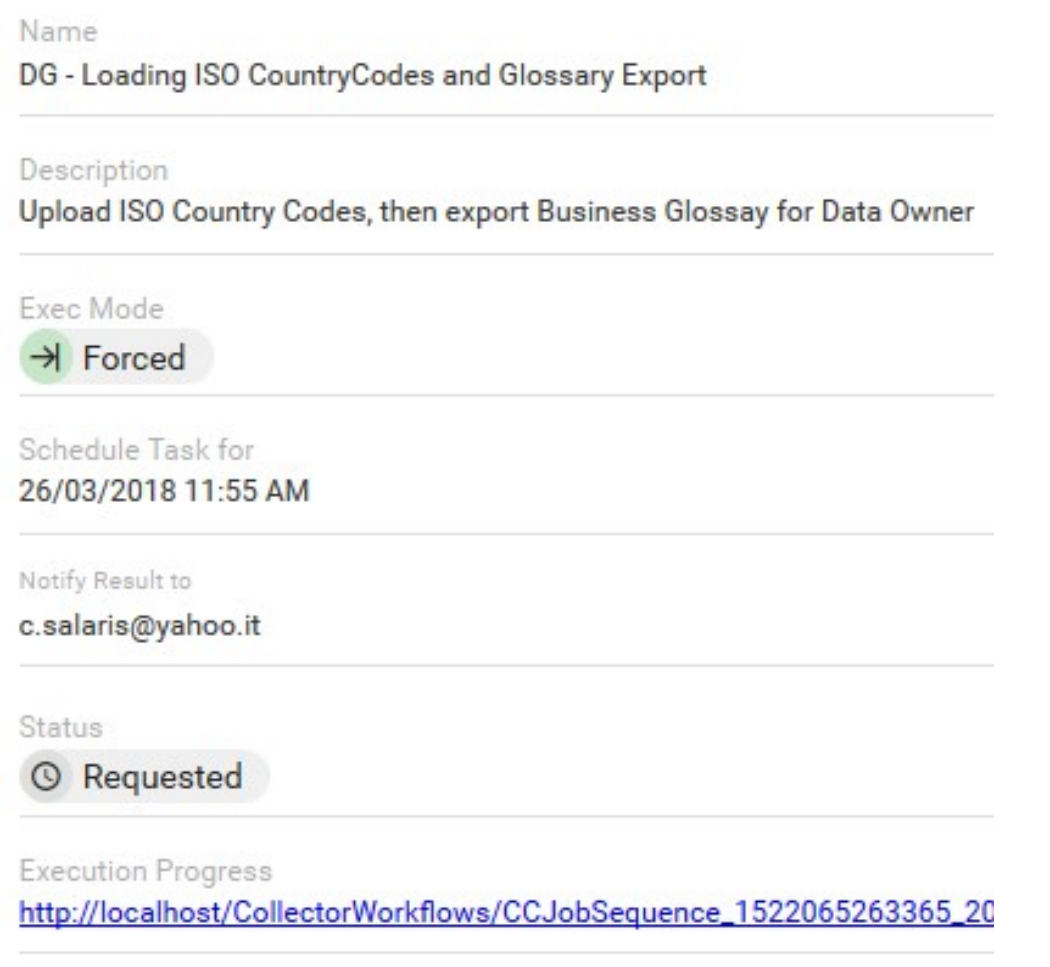
Both **erwin EA Agile** and **erwin EA Agile V3** are supported, depending on the specific configuration (see Cloud Environment). This enables Listener Windows service to support on premise application to work with cloud:

- *Job (workflows and sequences) definitions* are pushed to cloud

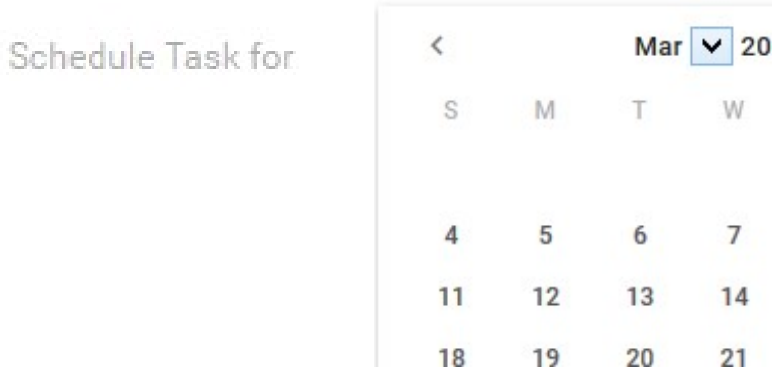


- *Job requests* are pulled from cloud
    - *one time requests*– drag and drop to “Requested” status kanban



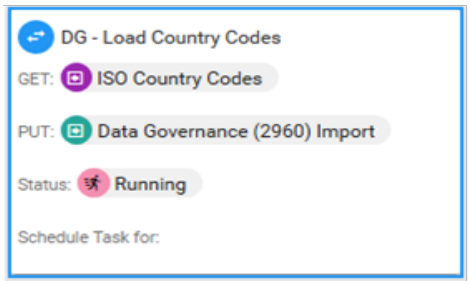


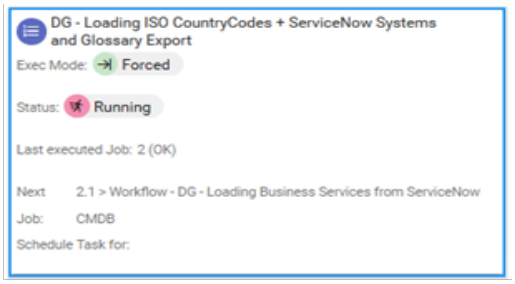
- *on schedule* – setting the datetime field “Schedule Job for”



Please, note that **repetitive schedules** are only configurable by **on premise application**.

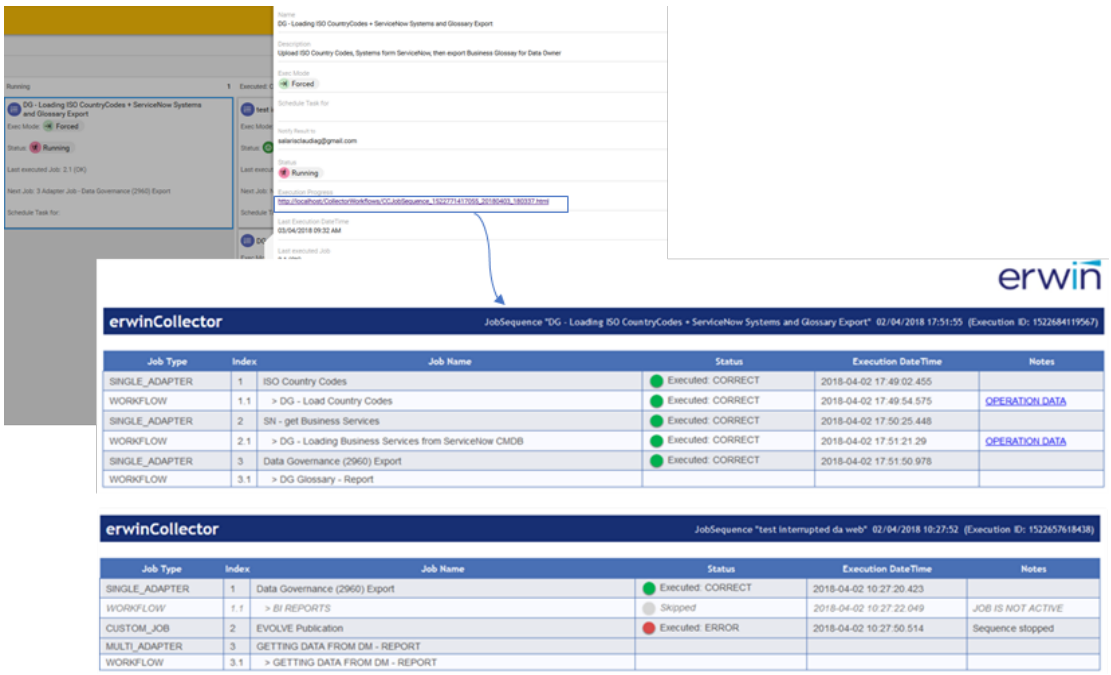
User can **monitor** the **execution of jobs** by different views and fields and linked reports:



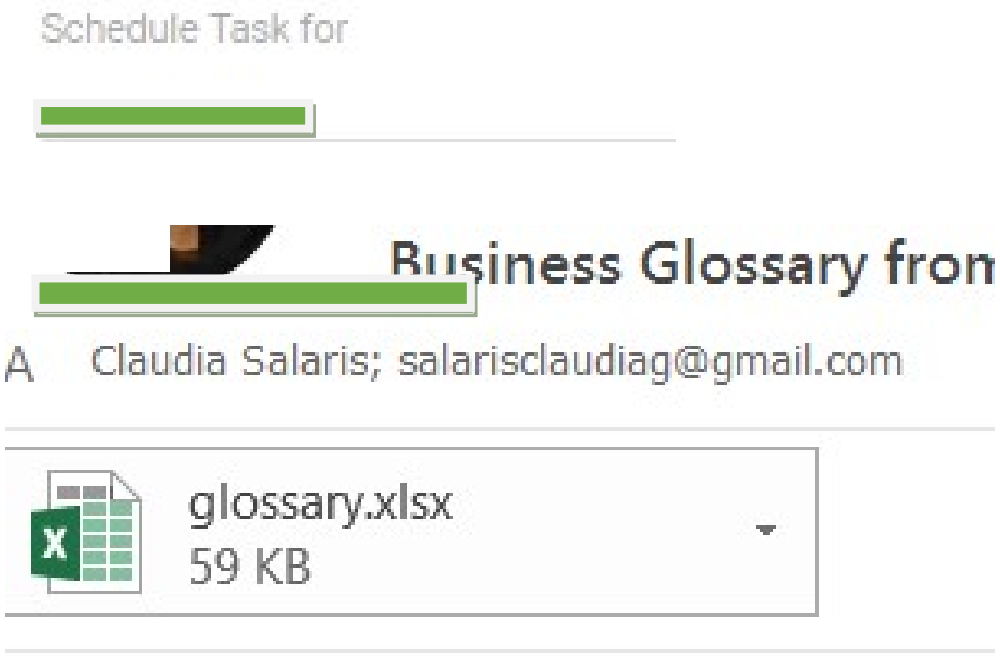


- *“Last execution Dataset”* for workflows:

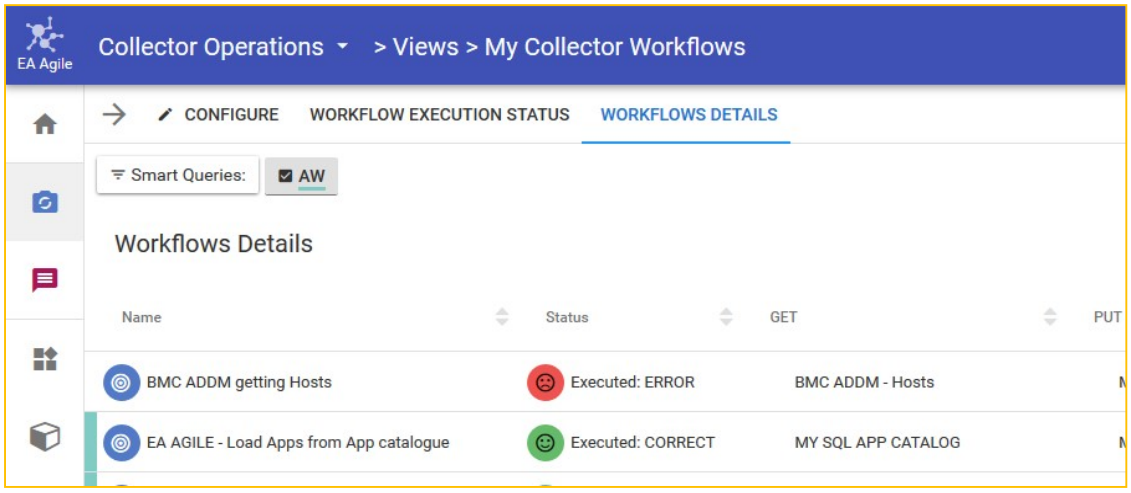
- *“Execution Progress”* for Job Sequences (with links to workflows dataset), continuously updated:



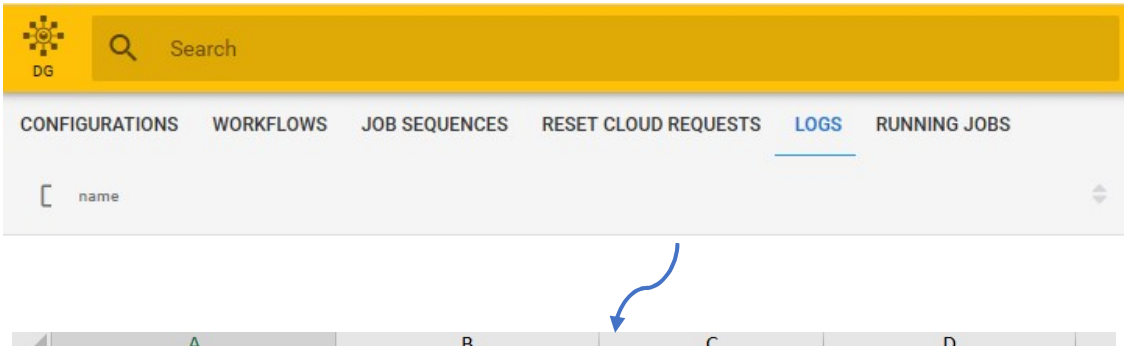
- *Notification emails* – can be requested by cloud setting the "Notify Result to" field on workflow/job sequences



Other **summary views** show more detailed information about last run, including last execution datetime and link to output file (if any) or error file (in case of errors):



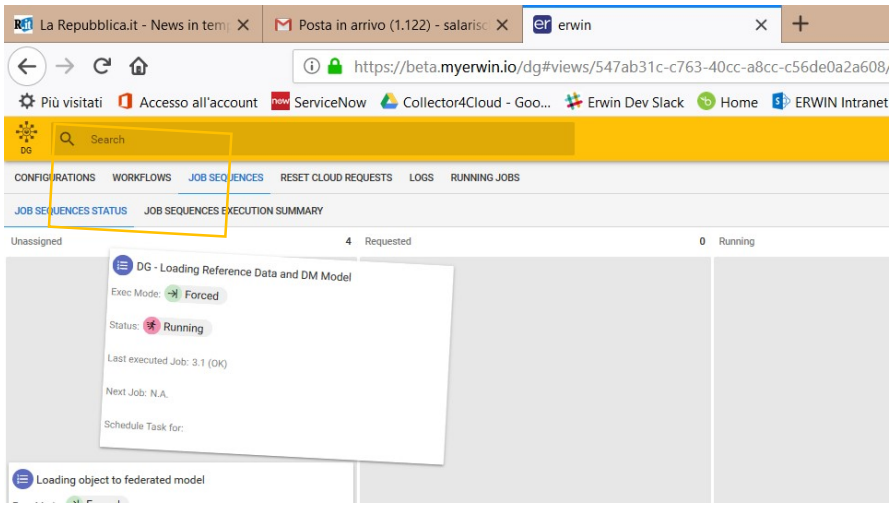
A dedicated **logs view** enables cloud users to be aware of DT operations, from the request through intermediate steps to execution, with a XLSX report, updated in near real time:



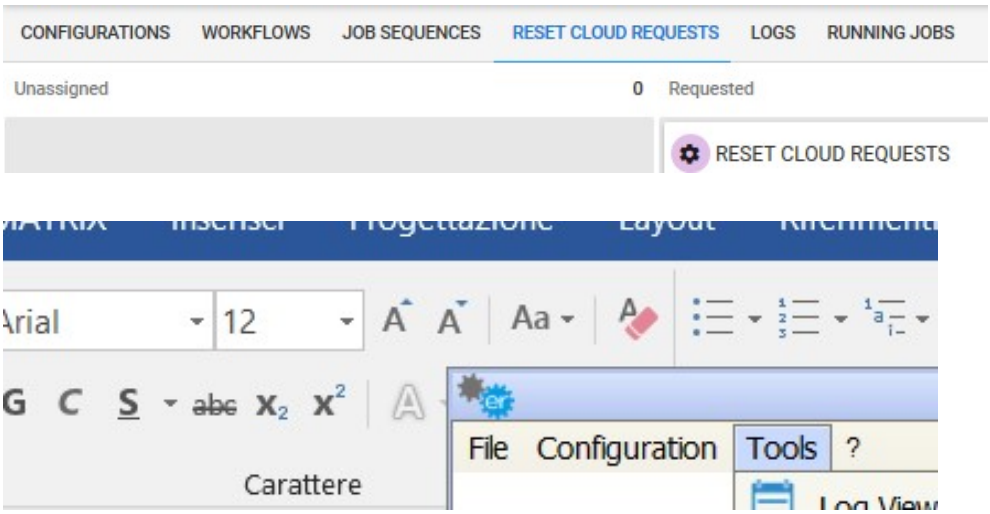
	A	B	C	D	
1	**CONFIGURATION**	**Operation Date**	**USER**	**OPERATION**	**NO**
2	All the configurations	2018-03-26 16:17:31.862	SequenceManager	**Sequence completed**	Seq
3	DATA GOVERNANCE conf	2018-03-26 16:17:20.665	File Adapter	**workflow completed**	Wo
4	DATA GOVERNANCE conf	2018-03-26 16:17:08.924	Workflow manager	waiting for operation...	Wo
5	DATA GOVERNANCE conf	2018-03-26 16:17:04.656	Workflow manager	**workflow starting...**	Wo
6	All the configurations	2018-03-26 16:16:18.616	SequenceManager	**Sequence completed**	Seq
7	All the configurations	2018-03-26 16:16:10.728	Data Governance	DG_endExport	Ada
8	DATA GOVERNANCE conf	2018-03-26 16:16:04.559	File Adapter	**workflow completed**	Wo
9	DATA GOVERNANCE conf	2018-03-26 16:15:51.55	DG Adapter	**workflow completed**	Wo
10	DATA GOVERNANCE conf	2018-03-26 16:15:33.331	Workflow manager	waiting for operation...	Wo
11	DATA GOVERNANCE conf	2018-03-26 16:15:28.033	Workflow manager	waiting for operation...	Wo

User can also **reset** requests:

- *one time requests* – drag and drop on “unassigned” status kanban
- *on schedule* – clearing the datetime field “Schedule Job for”, if not started; with drag and drop on “unassigned” status kanban, if running



- *all requests* can be deleted through “**RESET CLOUD REQUESTS**” job from cloud, or “**Tools→Reset Cloud Requests**” command from onprem UI.



Chapter 8

1 Log Viewer

The log window lists all the most important actions performed by DT.

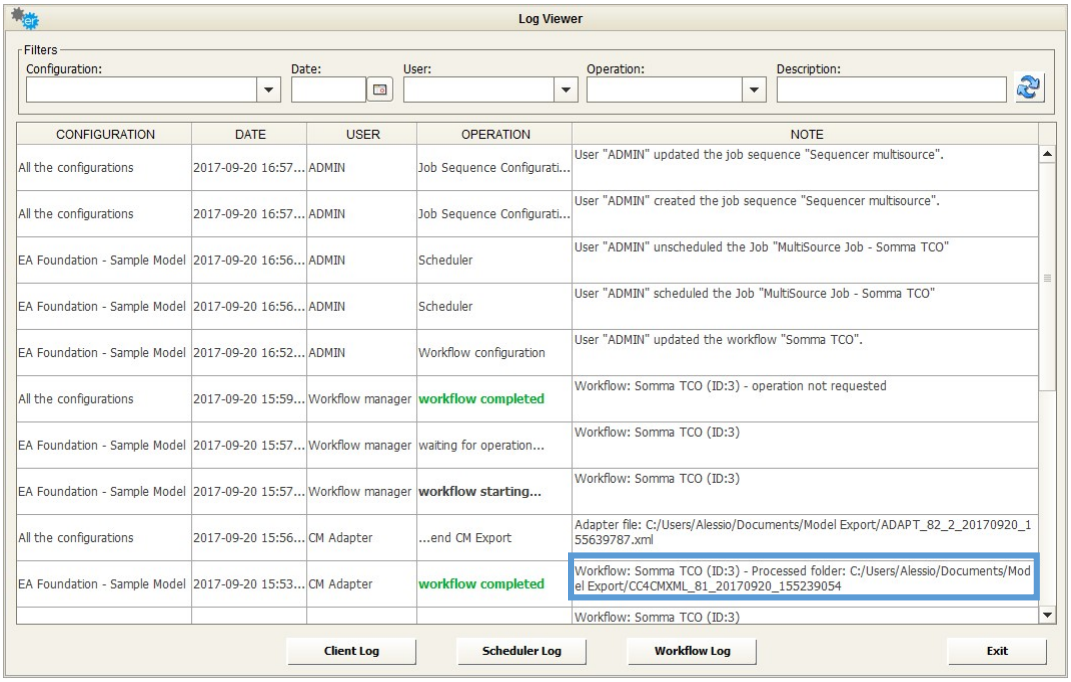
For each row, you can see the name of the configuration (column name *CONFIGURATION* – available only if linked to the action), the date (column name *DATE* – timestamp of the action), the user (column name *USER* – the user who owns the action), the operation (column name *OPERATION* – what has been done) and a short description of the action (column name *DESCRIPTION*).

Filters (one for each column) can be used to see a subset of the whole Log.

By pressing the  button the selected filters are applied.

Clicking on the head of each column, the rows will be ordered alphabetically.

If “Note” refers to a file or folder, double click on the note text will open it.



1.1 Client, Scheduler and Workflow log

The buttons open different kinds of logs showing a detailed view of the system events. Very detailed information can be found there, such as query messages, exceptions, errors etc.

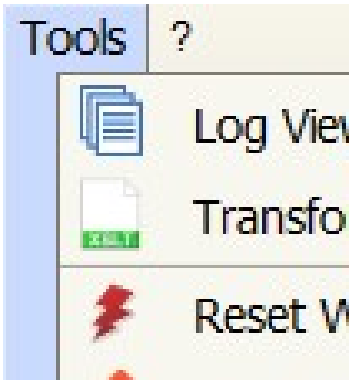
The log will be shown in your predefined text editor:

- *Client Log* refers to the operations executed through the user interface
- *Scheduler Log* refers to the batch operations related to Adapter execution
- *Workflow Log* refers to the operations required by workflow transformation and operation

Chapter 9

1 Utilities

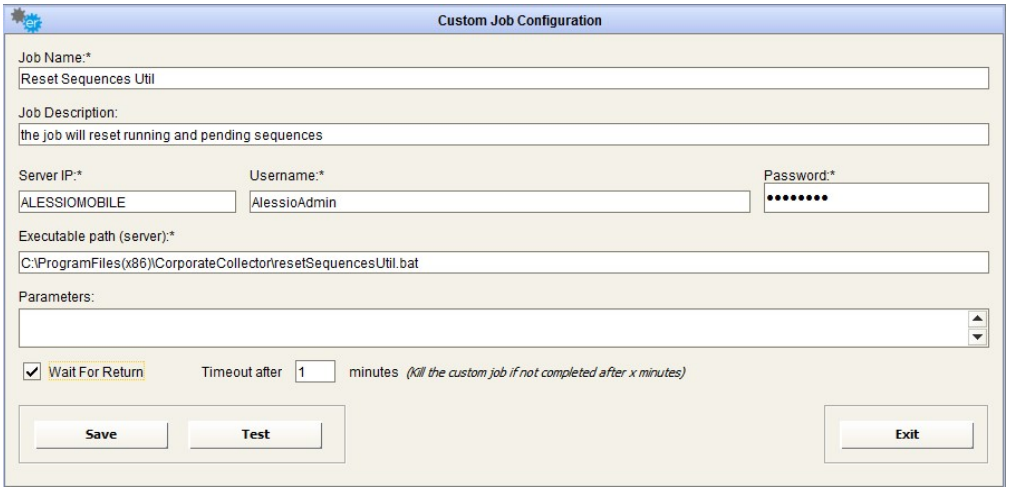
Additional administration features are available from the “Tools” Menu.



1.1 Reset Workflows/Sequences

If DT Windows Services are forced to stop, or in the case of a severe internal error, sequences may enter an inconsistent state which could stop other jobs from being executed. In such a case, the administrator may use the “Reset WF/Sequences” item in the “Tools” menu.

This operation can be done manually, as described, or by scheduling it in a custom job, using the sample execution: *resetSequencesUtil.bat* that can be found in the DT installation folder:



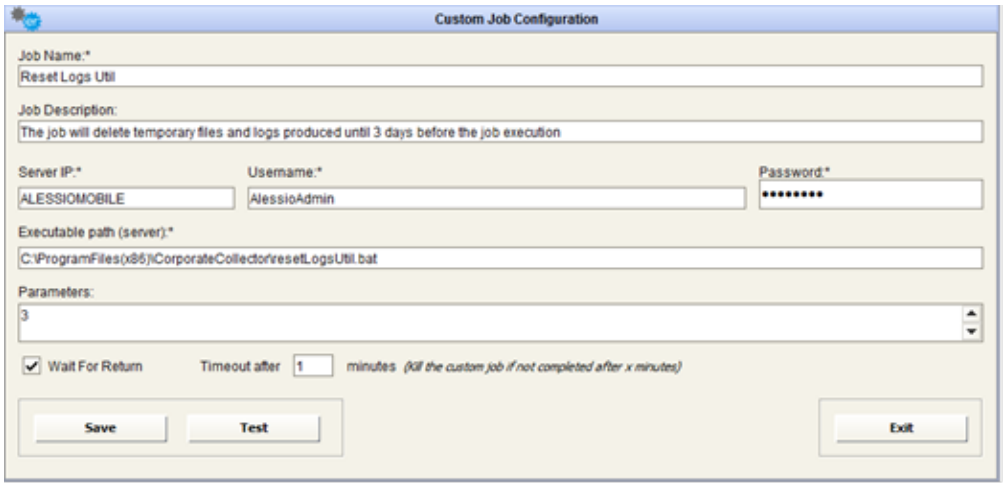
1.2 Reset Logs/Files

When needed, it is possible to reset the logs table just by pressing the “Tools” menu item “Reset Logs/Files”.

This action will clean the database of all the application logs, thus preventing the DB from being overloaded by historical tracking data.

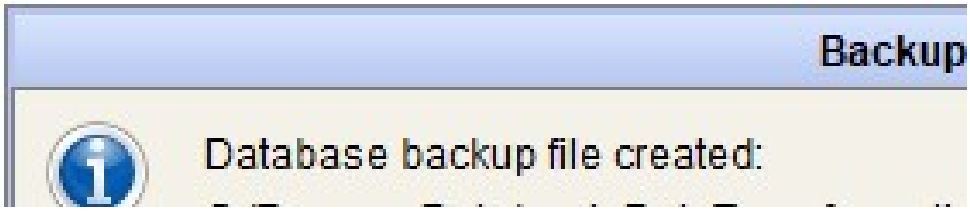
All the physical files mentioned in the deleted logs will also be deleted in the internal workflow folders; please be sure that no active workflows are running when using this feature.

This operation can be done manually, as described, or by scheduling it as a custom job, using the sample execution: *resetLogsUtil.bat* that can be found into the DT installation folder. The executable optionally accepts a numerical integer parameter, which is “days to be maintained”. For example, if called with “3”, past logs until 3 days before the job execution will be deleted.



1.3 Backup DB

Regular DT database backups are recommended as a best practice. This can be obtained with “Tools→Backup DB”: it produces a zip file with a dump of CC DB into “*C:\ProgramData\erwinDataTransformation\data\H2\BAK*” folder, with backup datetime:

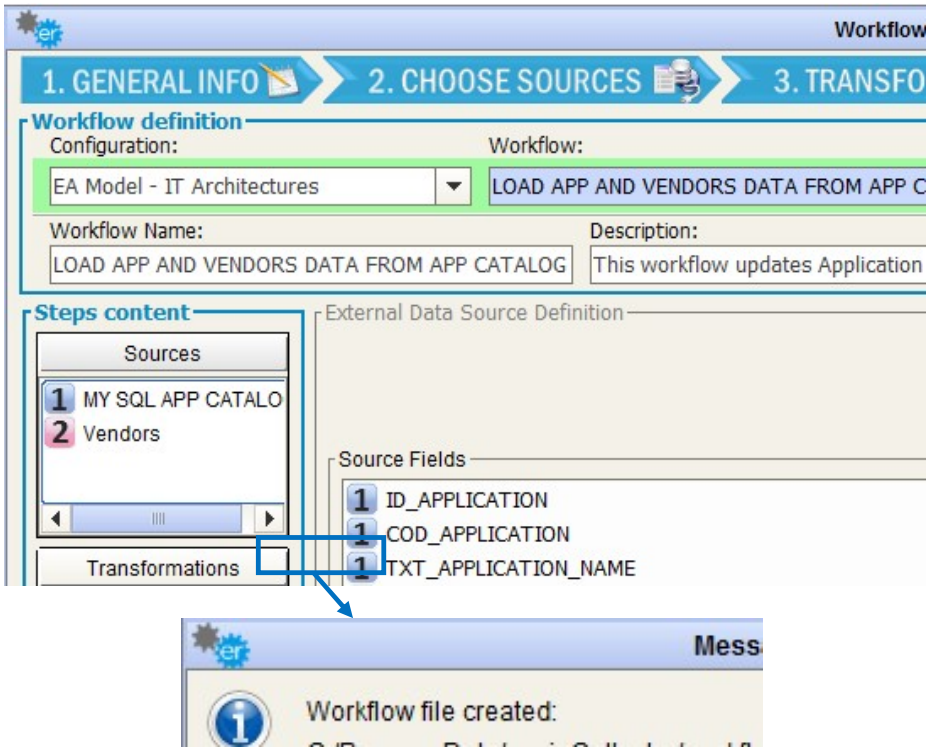


1.4 Export/Import Workflow

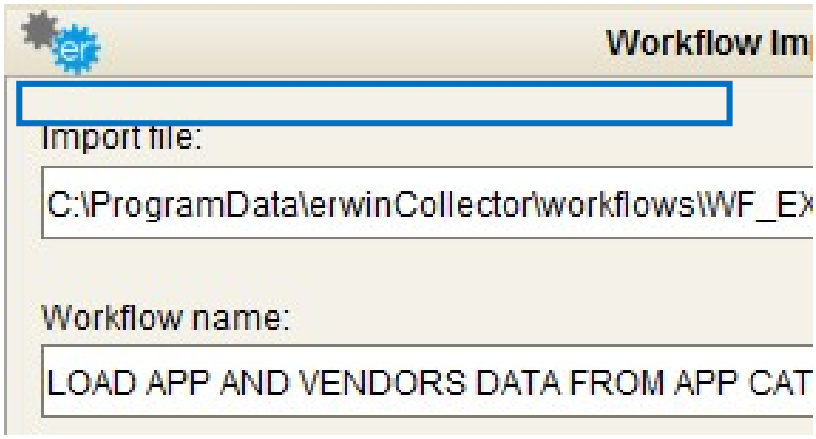
It can sometimes be necessary to configure and test workflows in a working DT instance before deploying them in a production site.

Workflows can be exported and imported in a different DT instance.

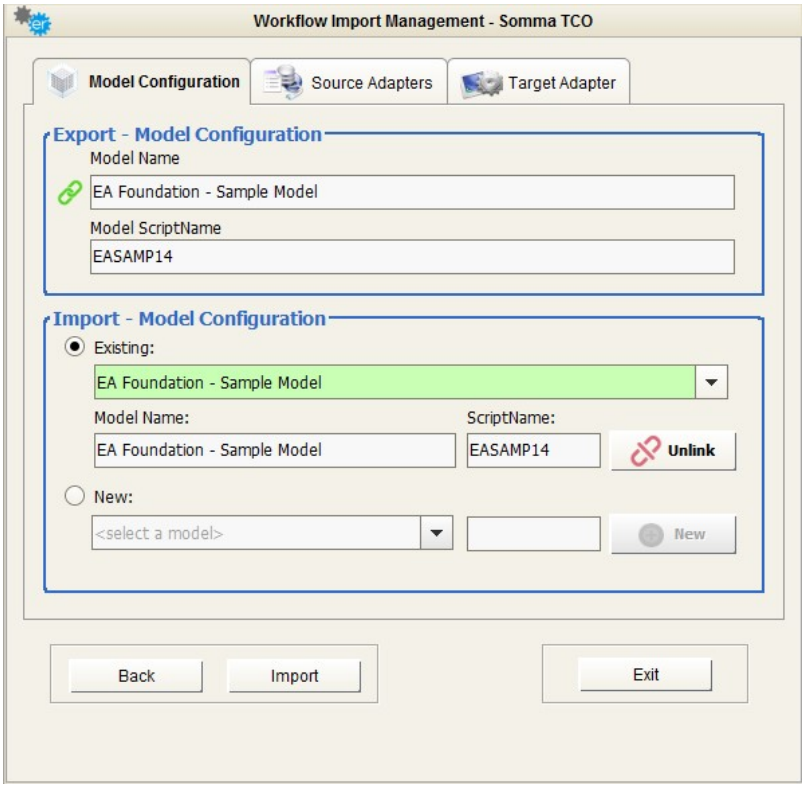
- To **Export workflows**, just open the workflow of interest, and create an archive (zip file) containing the workflow and its adapters/activity information:

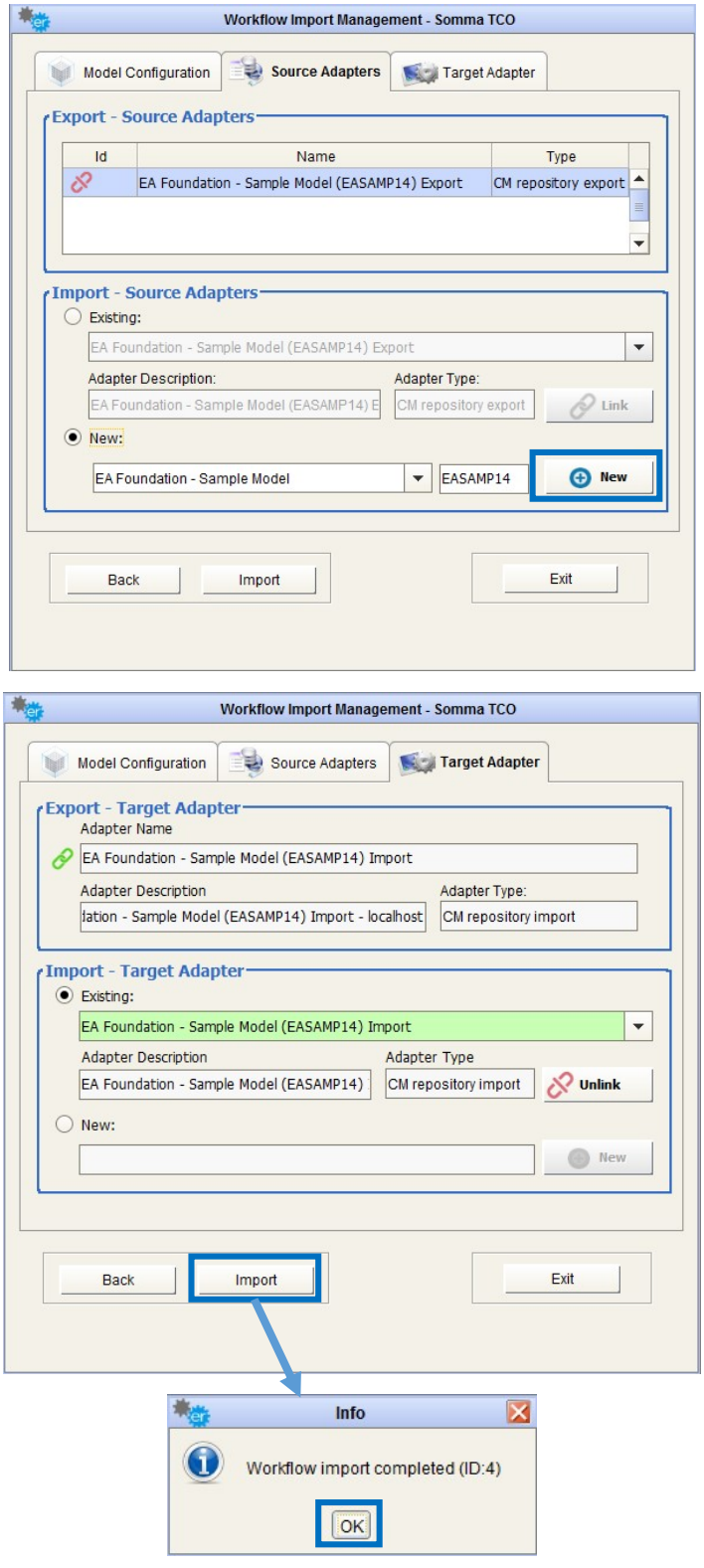


- To **Import workflows**, use the “Tools”→”Import WF” - or the button “Import” in the workflow configuration interface, after selecting a model configuration - and browse to the exported zip file previously created (or drag the file to the text field):

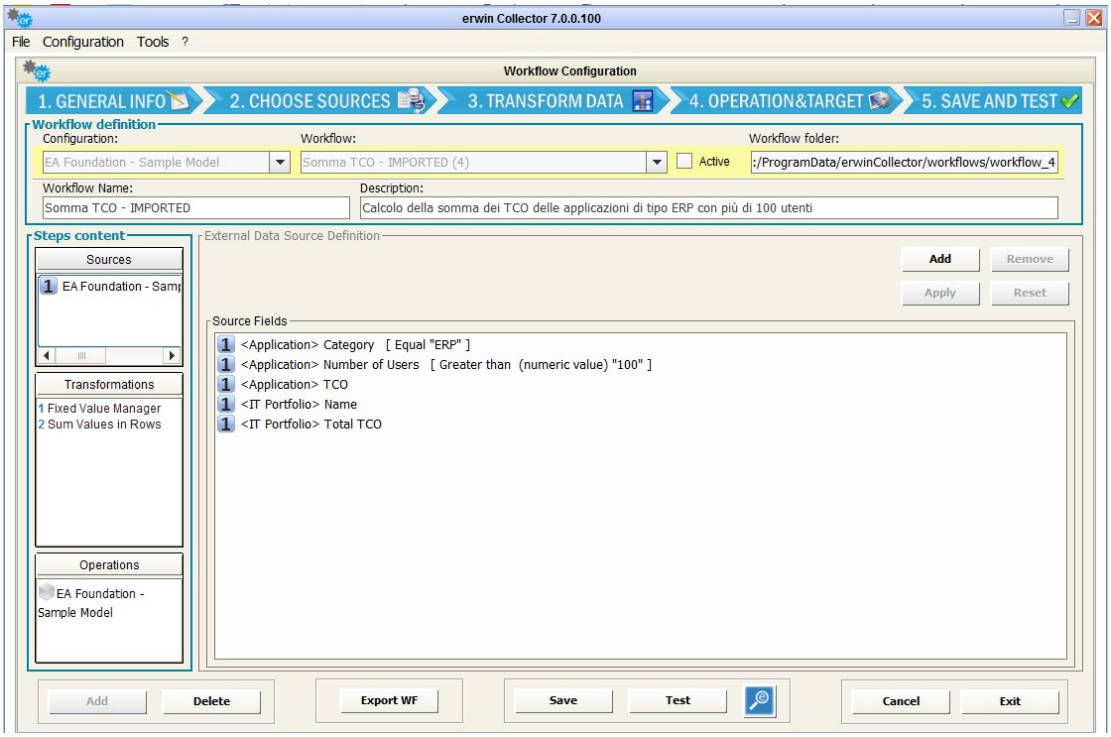


Then map the **model configuration**, **source(s) adapter(s),** and **target adapter**. In every step, should the needed configuration or adapter not exist in the target DT instance, **it can be contextually created**, allowing for smart import operations:



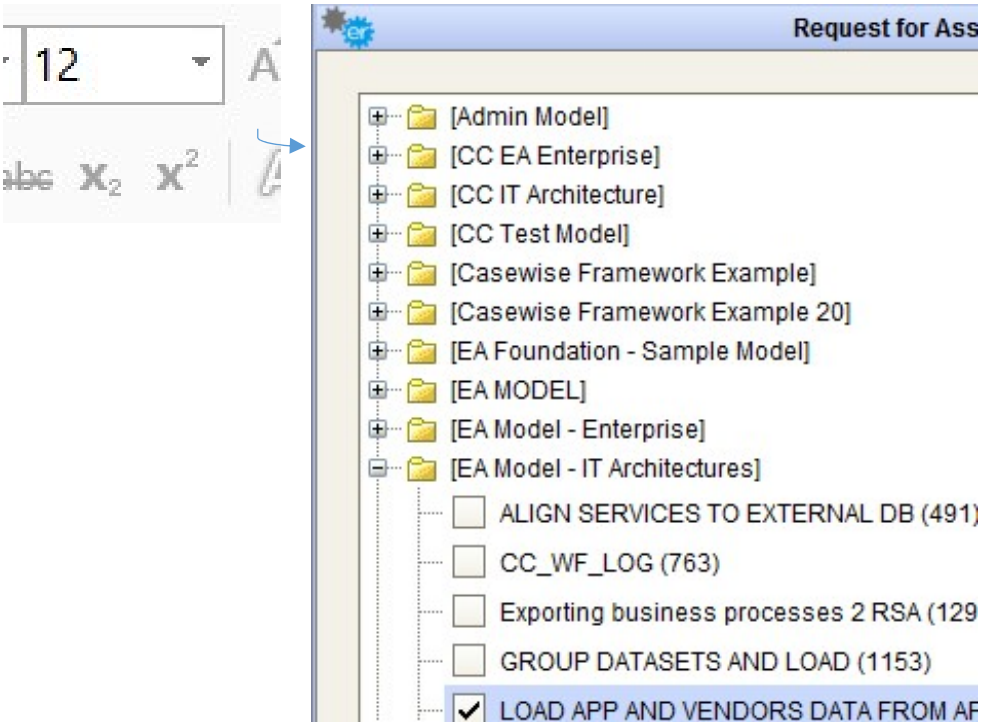


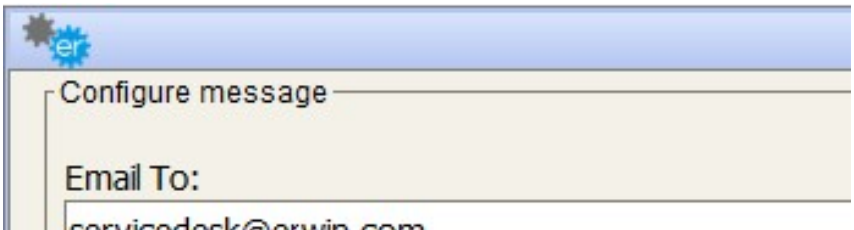
The result of the import operation is then shown to the user, which can edit/test/schedule workflow as usual:



1.5 Request for Help

Users can ask for assistance for workflows through a dedicated features in main toolbar:

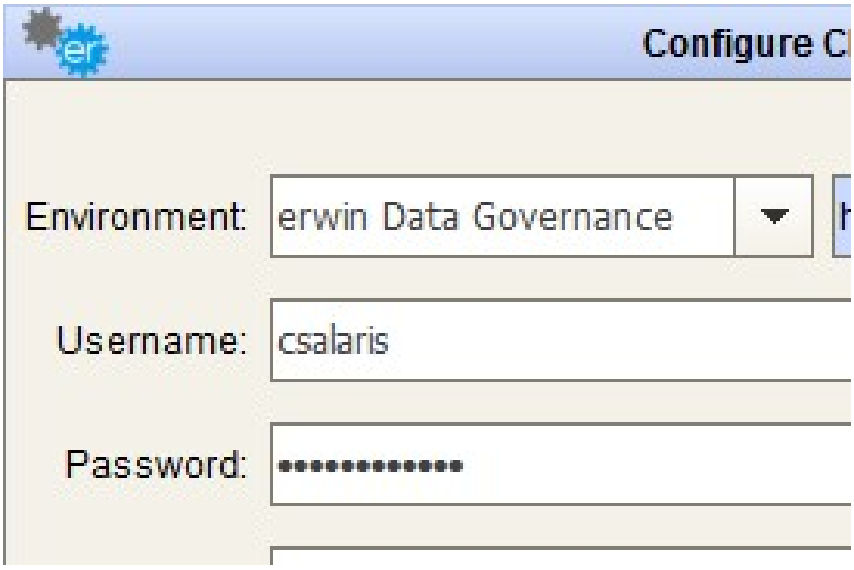




A message with a workflow definition and log file attached (more files can be added) will be prepared to be sent to ServiceDesk if DT email parameters have been provided during tool configuration.

1.6 Cloud Environment

Use this menu item to configure Cloud Environment parameters, to connect DT to erwin Cloud platform (see Running DT from Cloud Platform)



1. Select environment parameters – erwin EA Agile or erwin EA Agile V3 – and instance
2. Provide credentials for the selected instance
3. Test the connection
4. Select workspace containing the DT package (see before)
5. Set the communications to be active or not. In last case, DT will not take charge of requests from cloud. This may be applied as a temporary setting for upgrade/maintenance purposes etc.
6. Select a physical folder. This will be used as a publishing folder for web reports (HTML, pdf, csv, Excel), like web logs, workflows operation data, sequence progress reports. It has to be:
    a. Readable/writable by Windows user running Windows services and the DT UI on premise
    b. Published in LAN or public web, making DT reports visible and accessible by web.
7. Save
8. Restart Windows service “erwinDataTransformationEventManager”
9. Reset WF/Sequences from Tools menu item.

1.7 Reset Cloud Requests

Use this menu item to reset requests sent by erwin Cloud platform (see Running DT from Cloud Platform).

If DT Windows Services are forced to stop, or in the case of a severe internal error, sequences may enter an inconsistent state which could stop other jobs from being executed. In such a case, the administrator may use the “Reset Cloud Requests” item in the “Tools” menu, or request for “RESET CLOUD REQUESTS” job by cloud.

Chapter 10

1 Users, Roles and Security

DT interacts with CM contents – models, objects and so on – through the same logic and application components that Corporate Modeler and Evolve rely on.

Normal authentication *is the only* ***fully supported*** *option*.

Windows Authentication *is* ***not supported*** in DT, as it is not suitable for batch job execution.

General rules about security, authentication and authorization are:

- *Users' logon* is performed using their CM userid/password, provided that a valid DT license has been installed.
- Once logged on, users can define *model configurations*, and associated *workflows*, only for *models* which have been granted through Corporate Modeler Administration.
- When a user defines a model configuration, *users' credentials of related CM Adapters* (created automatically for data export/import on that model) are encrypted and stored in the CC internal database. This is needed in order to be available when workflows using those adapters are executed in a batch mode (which "Test workflow" simulates).

  This means that CM Users actually executing the operation are the person that configured and saved the associated model configuration. *If the user has to be changed*, the new user must log on to DT, and save the configuration, in order to bind their CM credentials to it; the workflows will be automatically updated.
- *Read/write/delete operations* on object type instances will be executed within the same user's rights scope existing in model. This means that they are impacted by general and specific grants on model and object type, by the status of the object (for

instance, Frozen level) and property type configuration (Read Only properties will be not updated).

When requested, these operations will be executed, if possible, or a log will be produced, containing details about what prevented the operation to be finalized.

Chapter 11

1 Troubleshooting Information

DT creates and uses some specific paths/folder during the execution of client configurations or scheduled operations. Main folders and files are explained in the following paragraphs.

1.1 Installation Path

The **installation path** is by default:

C:\Program Files (x86)\erwin\Data Transformation

Contents:

- *Configuration files*:
  - *<inst_path>\config\erwinDataTransformation.properties*
  - *<inst_path>\config\quartz.properties*
  - *<inst_path>\config\workflows\config\jdbc\CollectorDS.properties*

  containing the DT database connection string
- *Binaries*
- *External libraries*
- *CM4Collector.exe*

  performs import/export CM models operations

1.2 Data Path

The **data path** is by default:

C:\ProgramData\erwin\Data Transformation

Subfolders:

- *data*

  DT database files

- *libraries*

  XSLT parsers used for library/custom transformations

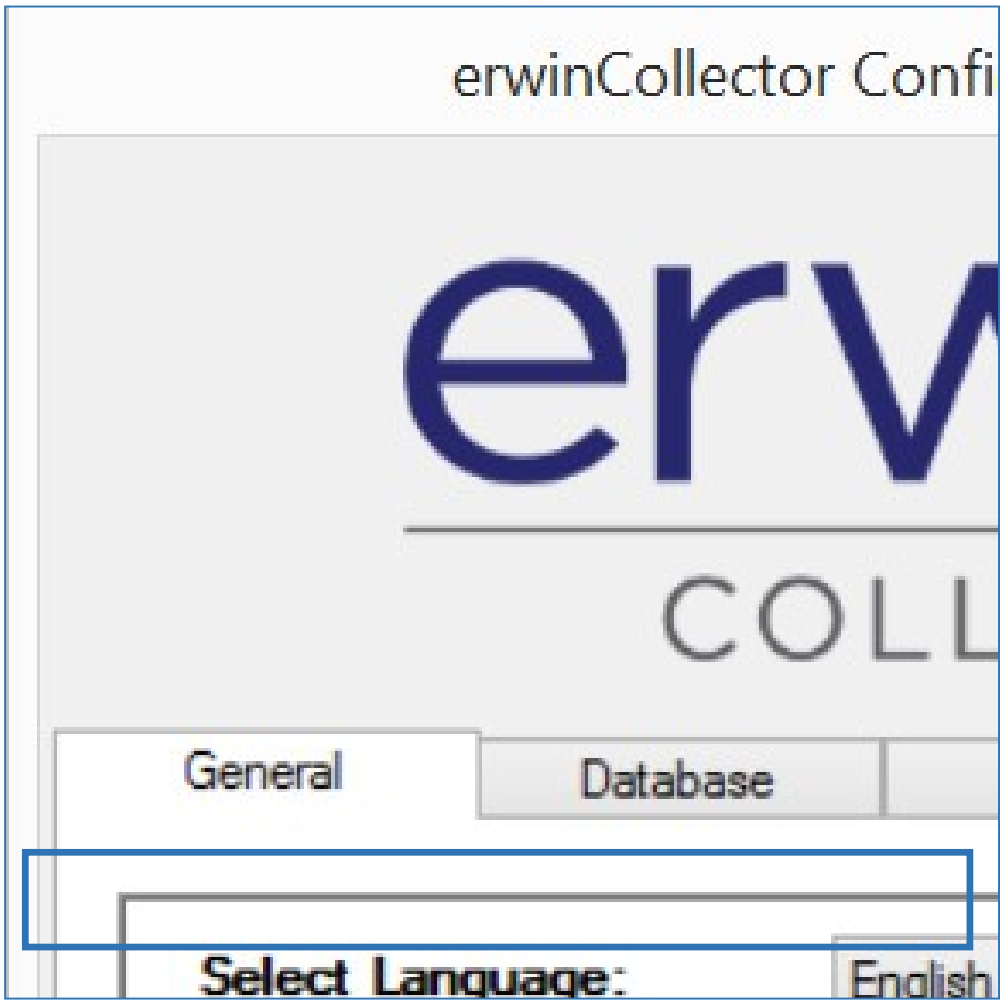
- *log*

  Client.log, event.log, server.log, listener.log

- *workflows*

  Workflows operation data, workflow testing triggers

The data path can be different from the default; configure the tool providing the desired path:



1.3 Setup / Windows Services

DT installs four Windows Services, which have to be launched by a Windows user belonging to the Administrator group:

- *erwinDataTransformationH2DB:*

  Runs the application database

- *erwinDataTransformationListenerManager*

  Detects newly created files into DT folders (see later)

- *erwinDataTransformationEventManager*

  Manage the event queue, as workflows to be run for adapter files detected by Listener manager

- *erwinDataTransformationServerManager*

  Performs scheduled jobs (adapters or sequences execution)

1.4 Adapter Folders

When configuring an adapter, DT requires to assign a folder.

This will contain:

- A file, created when saving the adapter, which is used to inherit the information about the structure of datasets coming from the adapter, named:

  ADAPT_<adapt_id>_struct.<adapt_ext>

  For instance, ADAPT_79_struct.csv will be produced when saving a DB adapter with internal id equal 79, containing the resultset columns and a sample of data.

- The following adapter files coming from its execution, that will be named as:

  ADAPT_<adapt_id>_<operation_timestamp>.<adapt_ext>

1.5 Model Configuration Folder

When setting up a model configuration, DT requires assignment of a folder for the backup. This will be used also as *adapter folder* for the automatically created CM import-export adapter.

This will contain:

- The backup of the model, when required by scheduling:

  <model scriptname>_<operation_timestamp>.mdb

- The structure file for export adapter, containing the metamodel: *ADAPT_<adapt_id>.xml*
- The CM export adapter files, when requested:

  ADAPT_<adapt_id >_<operation_timestamp>.xml

- Operation folders for the given model (see later)

1.6 Workflow Folder

When configuring a workflow, DT will automatically create a subfolder in the <datafolder>/workflows, assigned for the specific workflow, named:

workflow_<wf_id>

This will contain:

- the source adapter normalized file
- the multisource folder when using more than a source
- Support file containing trigger and parameter information
- the xslt dynamically created to perform requested transformation
- the intermediate and final result of the transformation chain

1.7 Operation Folder

When configuring a workflow operation, this will be in charge of an adapter (*CM Import, DB Loader, Folder*).

The workflow manager, at the end of the transformation chain, prepares the files which allow it to perform the requested operation, and puts them into a subfolder of the adapter folder.

In particular, for:

- *Load/Synch/Delete CM Repository*

  The subfolder will be created into target model folder

- *Send to DB Loader*

  The subfolder will be created into adapter folder

- *Send to File Adapter*

The subfolder will be created in the target folder

Chapter 12

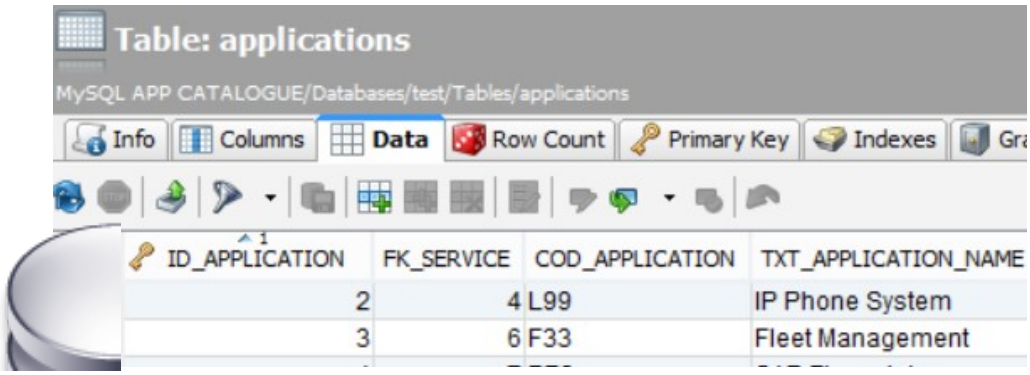
1 Examples of Workflow Configurations

1.1 Loading data from external sources into a model

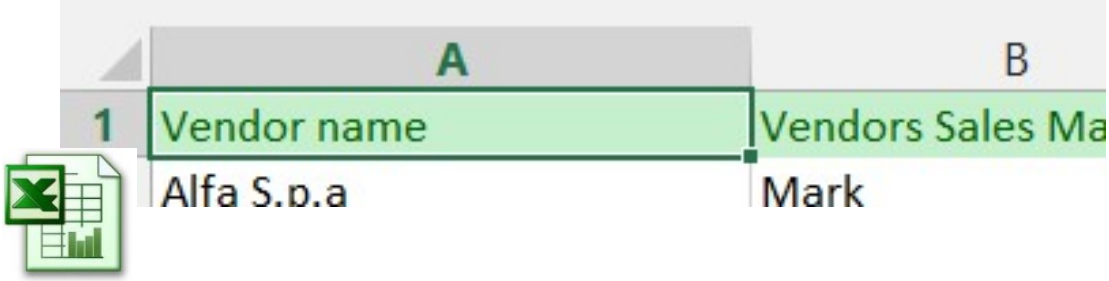
This example describes the following use case: application and vendor data stored in different external tools must be loaded consistently in a CE repository.

Let us suppose that:

- a **CMDB** manages the **application data**, stored in a MySQL **database table**. The application table identifies the item with a "COD_APPLICATION" field, which has a different erwin ID for the model for the same item:



- **Vendor data** for any application are provided by another APM tool, via an **Excel** report. The file relates the Vendor and the Sales Contact with an Application through its CMDB code:



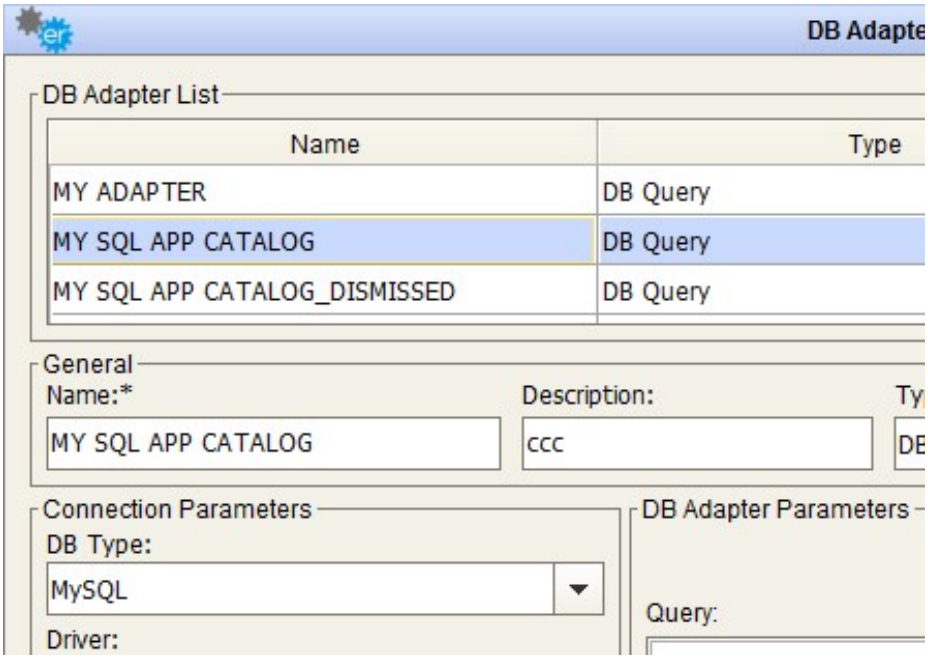
A DT workflow importing and refreshing application and vendor data can be configured with the following steps:

- CREATE MODEL CONFIGURATION

  Log in to DT and create a model configuration, for instance on model “Framework Example”. To load data into a model choose the specific model configuration.

- CREATE DATABASE ADAPTER FOR SAMPLE APPLICATION CATALOG DB

  Press the “DB Adapter” button on the Home Page and configure the following adapter:



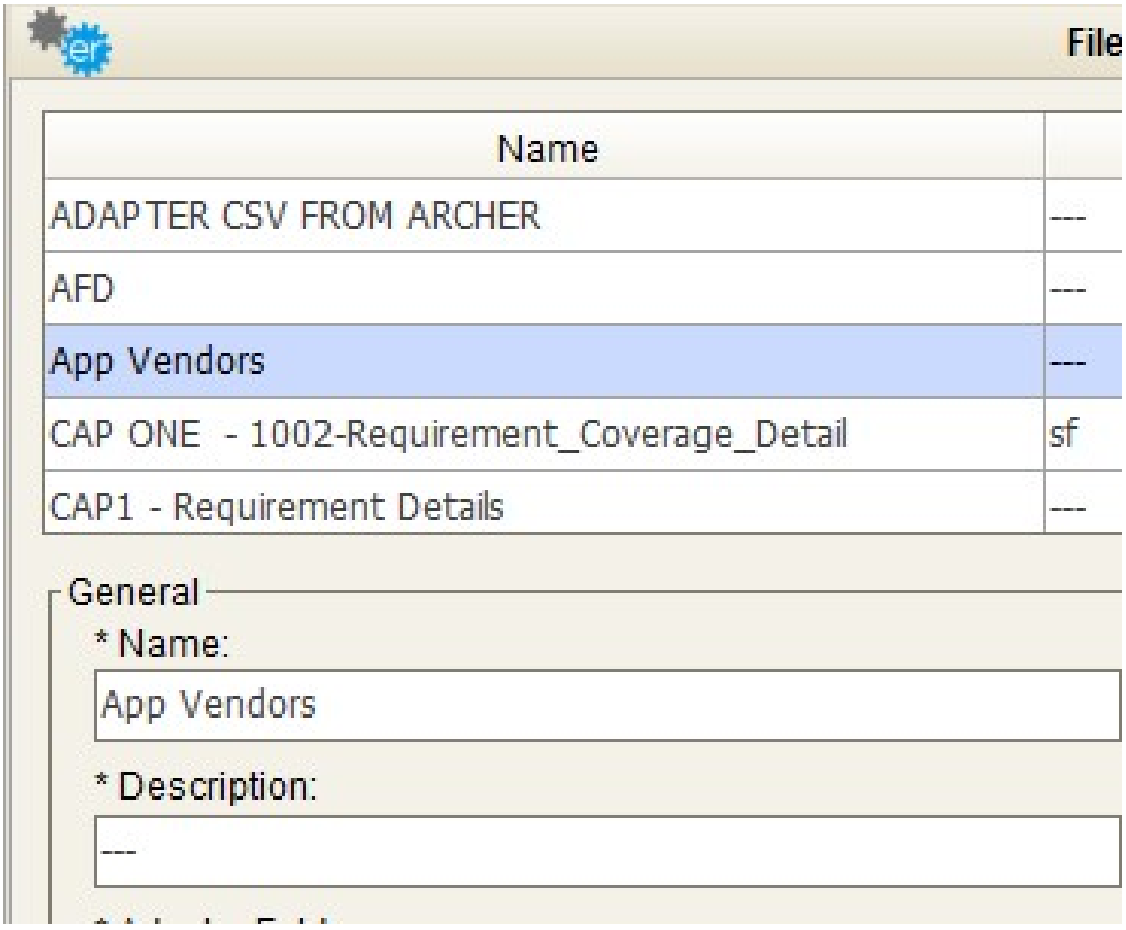
having the database type equal to “MySQL”, server, port, database name, user, password and query producing the desired data (preview with the “Test” button):


Adapter Preview:
MY SQL APP CATALOG

ID_APPLICATION	COD_APPLICATION	TXT_APPLICATION_N...	DESCR_AP
26	H65	Mailing List	index_definitior
2	L99	IP Phone System	The primary ob
3	F33	Fleet Management	
4	B72	SAP Financials	It is based on tl

- CREATE FILE ADAPTER FOR VENDORS INFORMATIONS FILE

Press the “File Adapter” button on the Home Page and configure the following adapter:

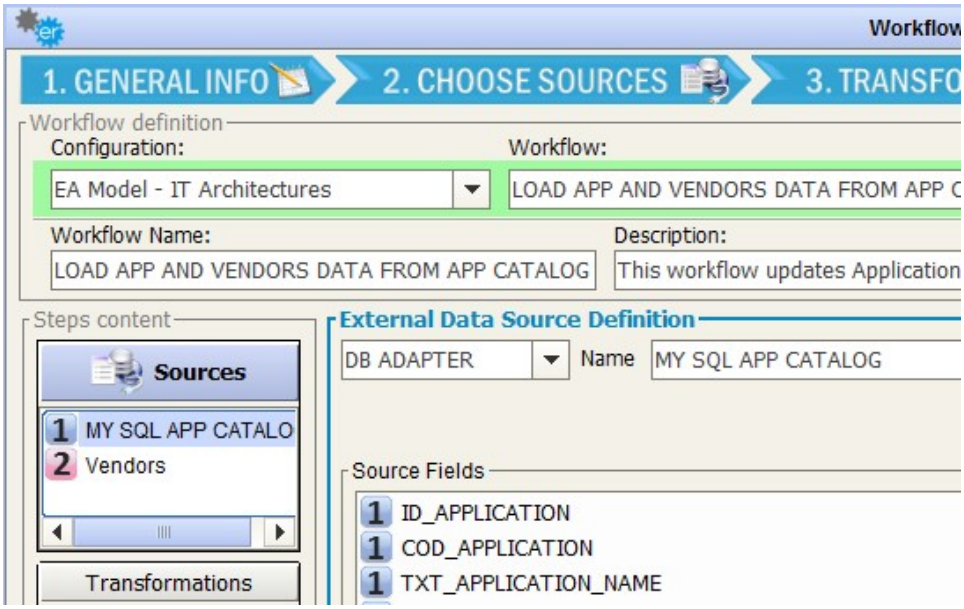


- CREATE WORKFLOW TO LOAD APPLICATIONS AND VENDOR DATA

  Press the “Workflow configuration” button on the Home Page and configure the following workflow choosing the model configuration set in the previous steps:

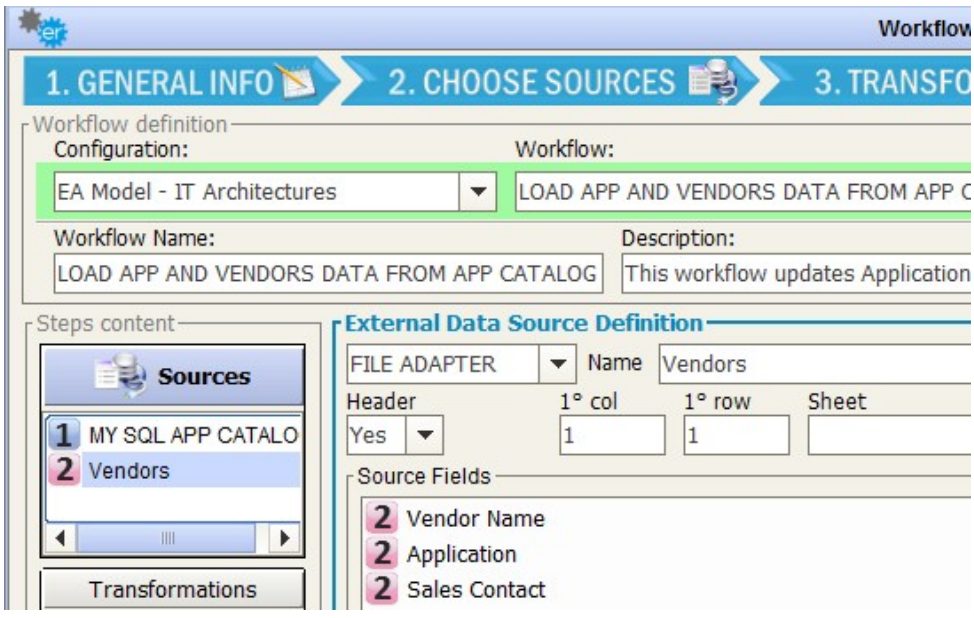
  SOURCE DEFINITION – Source 1

  Choose previously configured DB adapter on MYSQL:

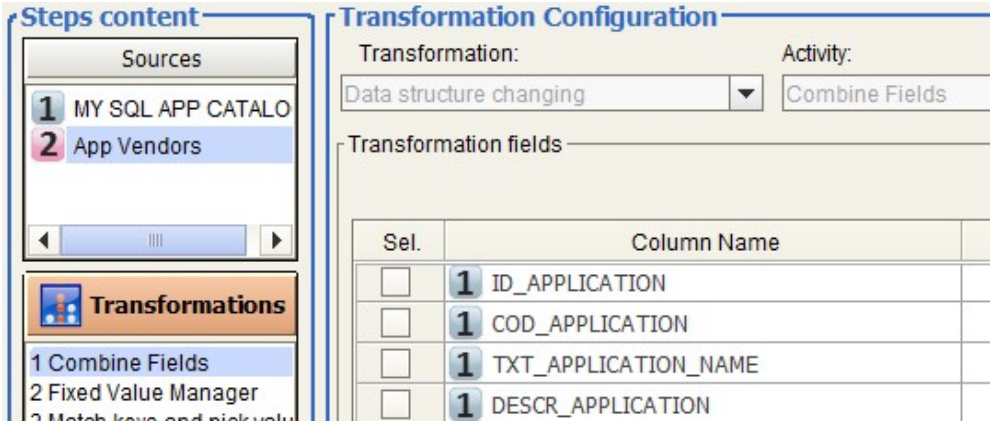


  SOURCE DEFINITION – Source 2

  Choose previously configured file adapter for vendor info:

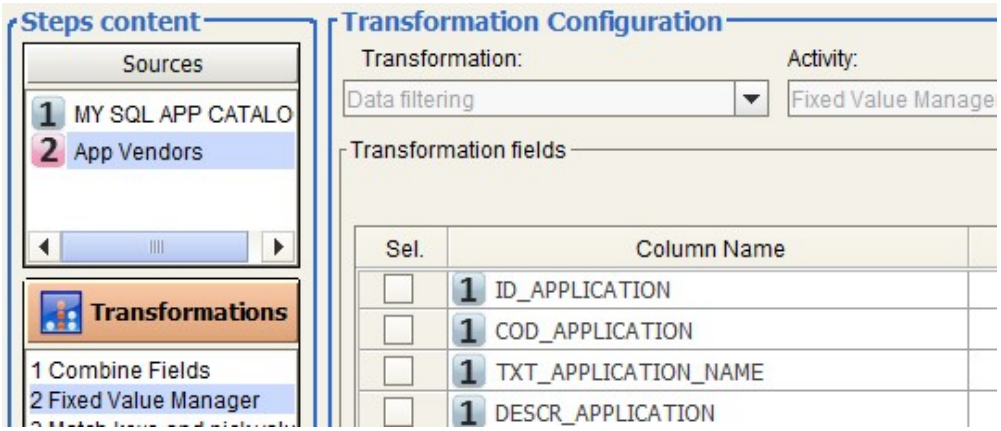


Let us suppose that “Status” field in CM model is coded, combining the status id with the status description (for example “6 - Production”). Combine the application status code and application status description (columns “1 ID_STATE” and “1 DE_STATE”) in a new column named “1 Coded state” with a given separator according to target model conventions:



TRANSFORMATION 2 – Filter Status Values

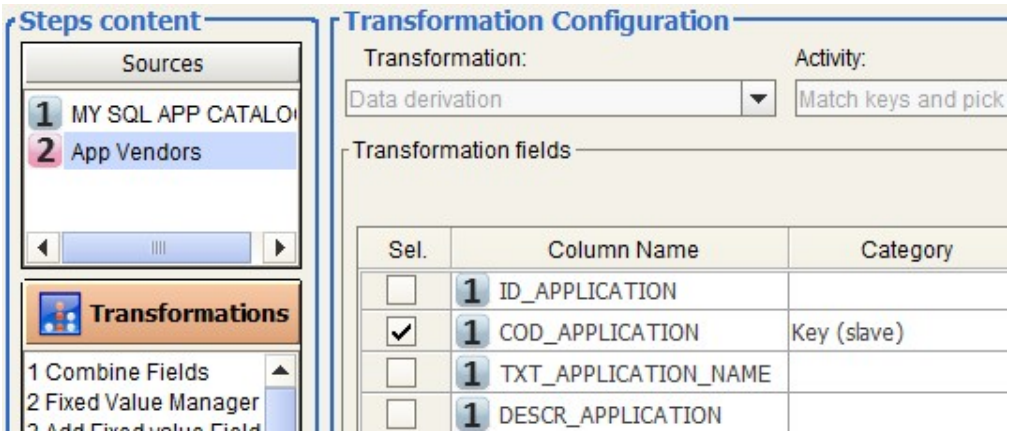
CM model manages only active applications: choose “DATA FILTER”, “Expected fixed Value Manager” to manage only application with a status different from “Dismissed”:



TRANSFORMATION 3 – Pick the application name from the App catalogue dataset, given its id

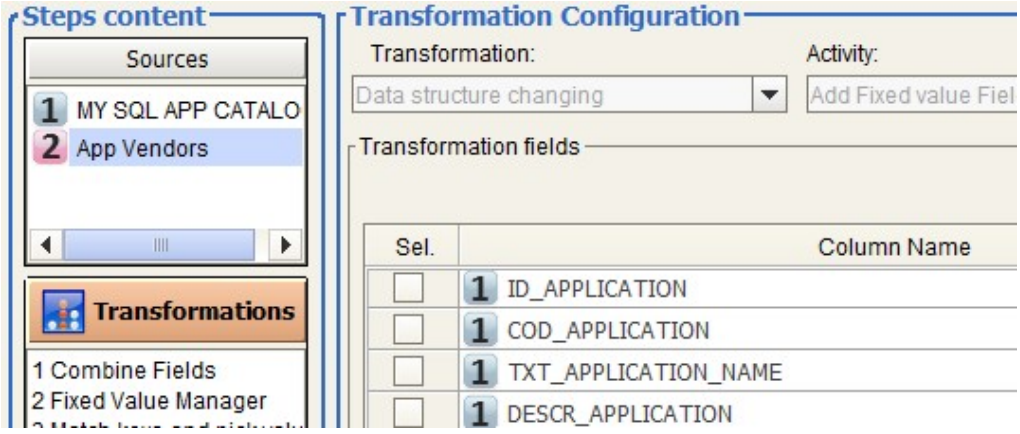
The “Data derivation” activity “Match keys and pick values” applies if it is required to join values from different data sets and have a different result column value if the keys are matched or not. For the purposes of this exercise:

- Set the ‘Category’ field to “Key (slave)” for “COD_APPLICATION” and “Key (master/result)” for “ApplicationID” in the Vendor dataset: pick “TXT_APPLICATION_NAME” column from app catalogue dataset to get the application name, when the keys match (“Value If True”) and the column itself, otherwise (“Value If False”).



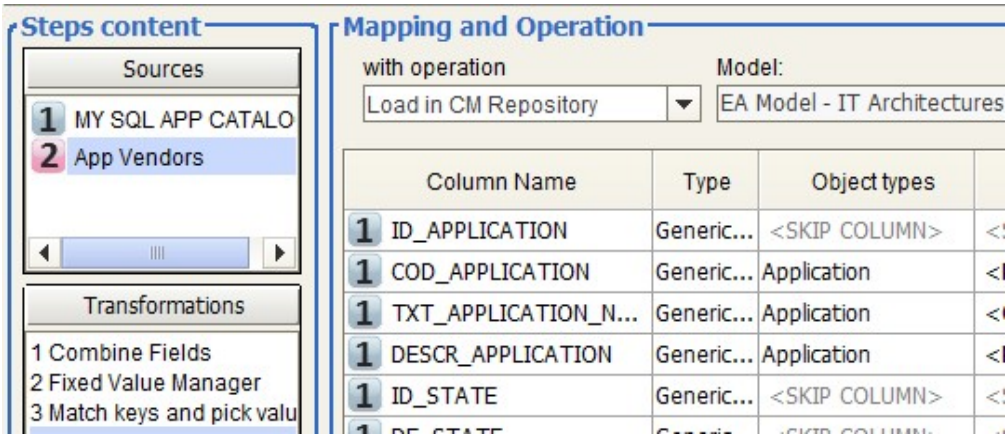
TRANSFORMATION 4 – Set Vendor Category.

Choose “DATA STRUCTURE CHANGING” and “Add fixed value Field”, with name “Vendor Category” and value “VENDOR” and assign to Adapter “2 App Vendors”:



OPERATION – Load in CM Repository

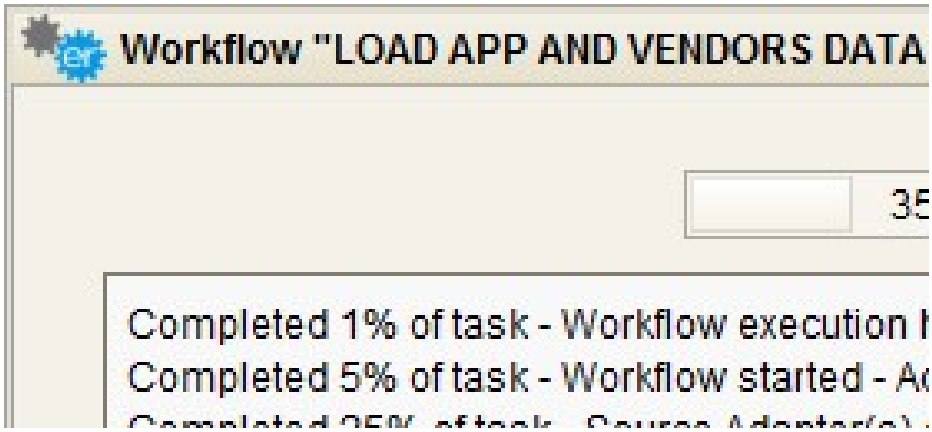
Load application and vendor data in to model, as in the next picture.



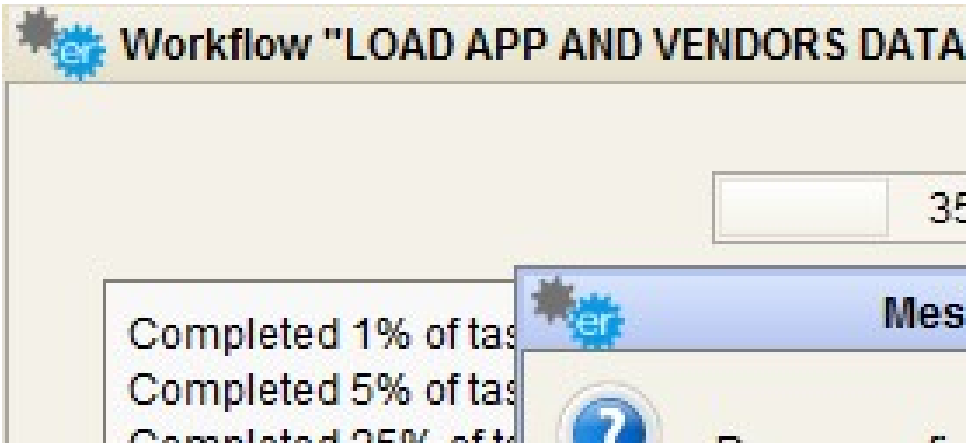
Note that the keyset for operation is set to “<Property> CIID” to manage the renaming of applications. The “Forced replace” option under ‘Action’ for “TXT_SERVICE_NAME” will make the association between “applications” and “IT services” be replaced even when associations are diagrammed.

- TEST THE WORKFLOW

  Test the configured workflow with the “Test” button, and follow its execution through the progress bar or Logs and email notifications.



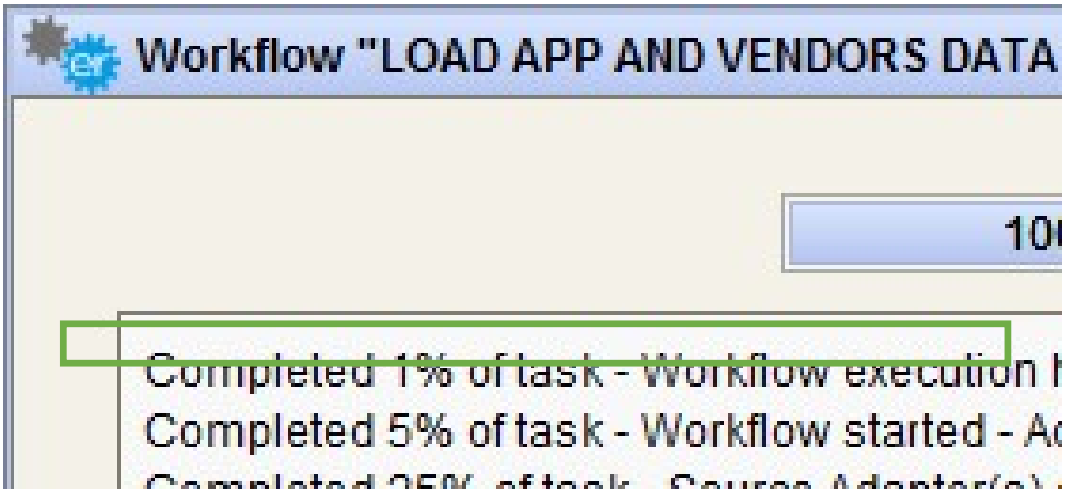
At the end of workflow transformation if operation data is available and ready, the user may choose whether to execute the operation or not:



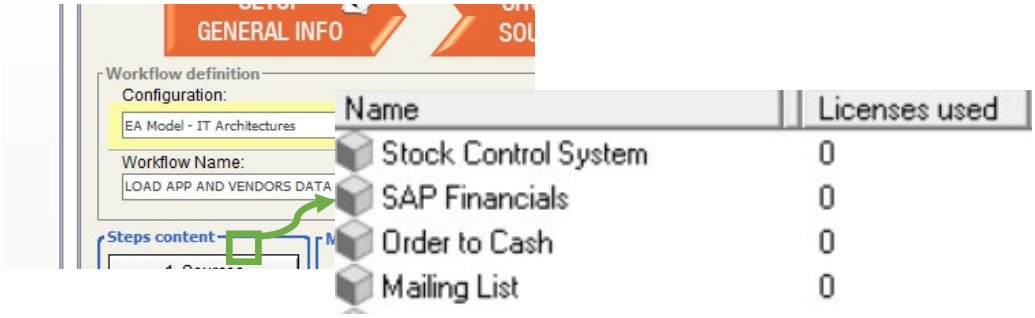
- CHECK RESULT

  Moreover, you can check the result:

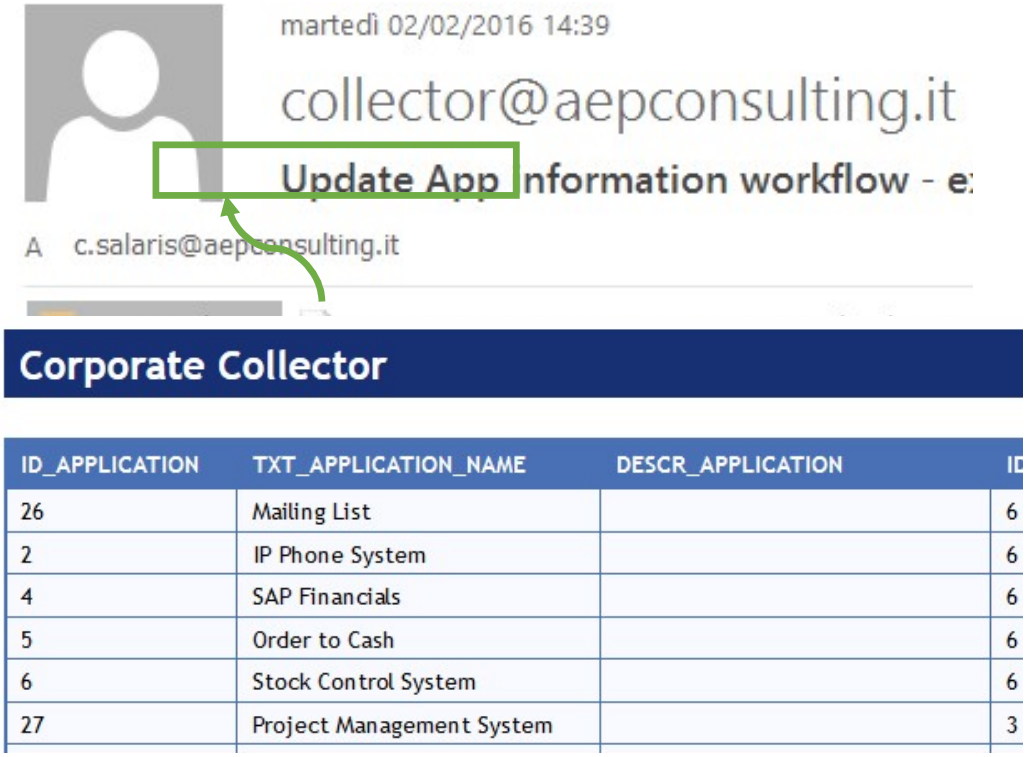
  - from the test workflow **progress bar**:



  - in **CM**, accessing the tool through the CM icon provided in the button bar.



- with the **email notification**, if configured for the single workflow:


martedì 02/02/2016 14:39

collector@aepconsulting.it

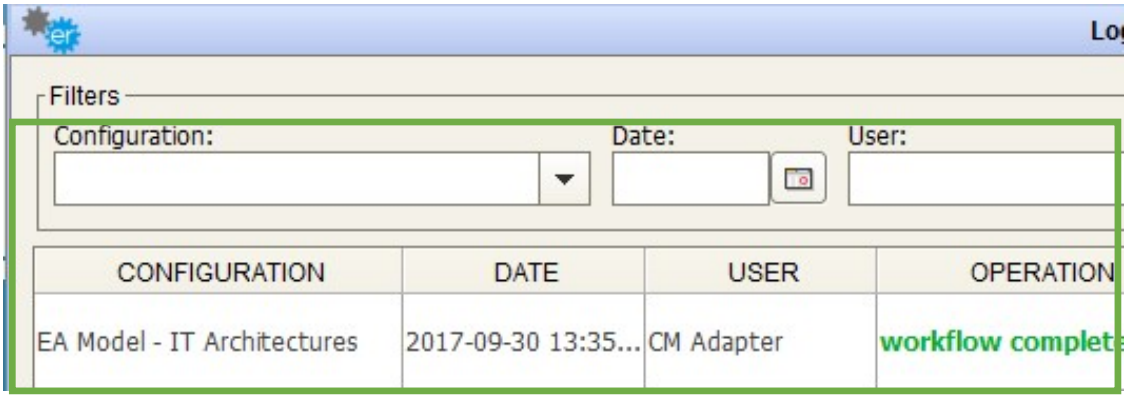
Update App nformation workflow - e:

A c.salaris@aepconsulting.it

Corporate Collector

ID_APPLICATION	TXT_APPLICATION_NAME	DESCR_APPLICATION	ID
26	Mailing List		6
2	IP Phone System		6
4	SAP Financials		6
5	Order to Cash		6
6	Stock Control System		6
27	Project Management System		3

- from the **DT Log Viewer**:


Filters

Configuration: Date: User:

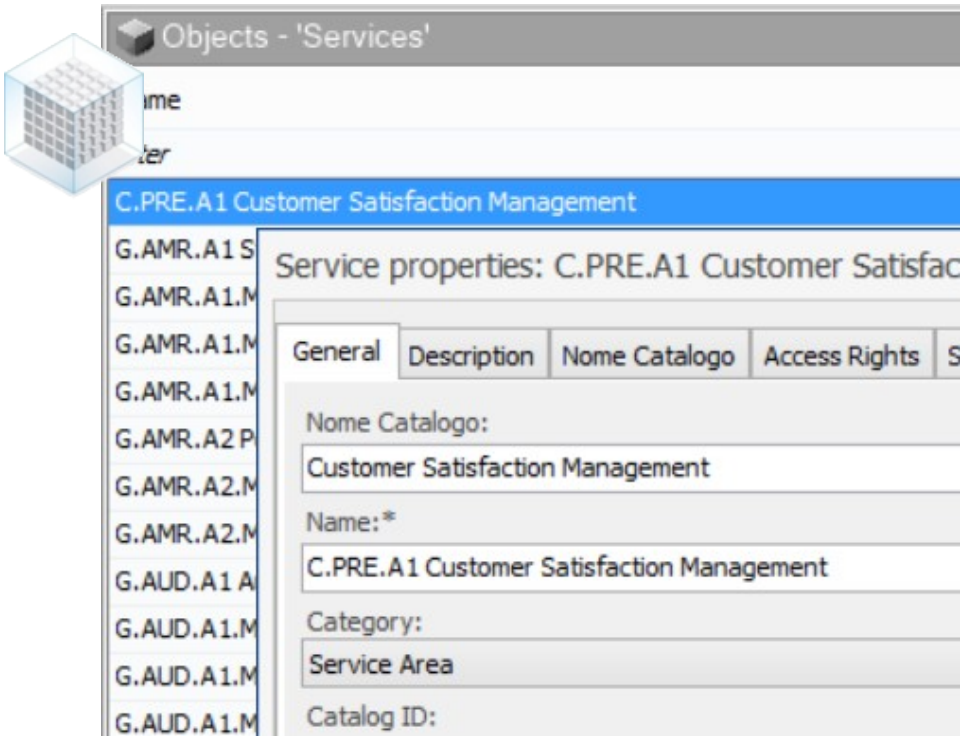
CONFIGURATION	DATE	USER	OPERATION
EA Model - IT Architectures	2017-09-30 13:35...	CM Adapter	workflow complet

1.2 Align External DB with Model Contents

The example describes the following use case: business service data, managed natively into a CE repository, must be loaded consistently in a MS SQL database, hosting a third party Service Catalogue.

Let us suppose that:

- **Business service data** are into the **CM model**, in a dedicated **object type**



- a **Service Catalogue** manages **enterprise services data**, stored in a MS SQL **database table**. The service table identifies the item with a “COD_APPLICATION” field, which is differentiated by the Erwin ID of the model for the same item:

	SVC_ID	NAME	DESCRIPTION
	C.PRE.A1	Prestazioni sanitarie	
	G.AMR.A1	Statistica	
	G.AMR.A1.M01	Consulenza statistica-attuariale	
	G.AMR.A1.M01.S01	Raccolta dati statistici esterni all'Istituto	Acquisizione dei dati e dell...
5	G.AMR.A1.M01.S02	Elaborazione e analisi statistico attuariale	Studio, analisi ed elaborazi...
6	G.AMR.A2	Politiche di indirizzo	
7	G.AMR.A2.M01	Politiche di indirizzo	

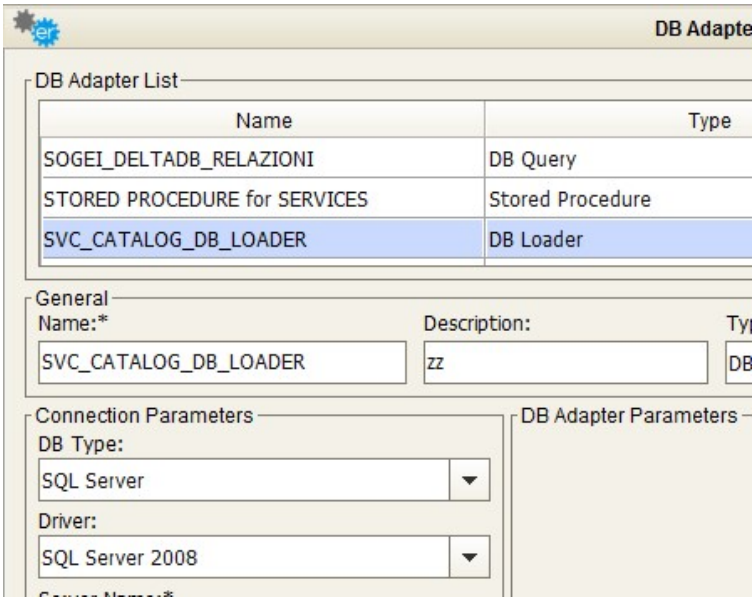
A DT workflow aligning Business Services data from Erwin to the Service Catalogue database can be configured with the following steps:

- CREATE MODEL CONFIGURATION

  Log in to DT and create a model configuration for the model involved in the use case

- CREATE DATABASE ADAPTER TO LOAD SERVICE CATALOG DB

  Press “DB Adapter” button in Home Page and configure the DB Loader Adapter:



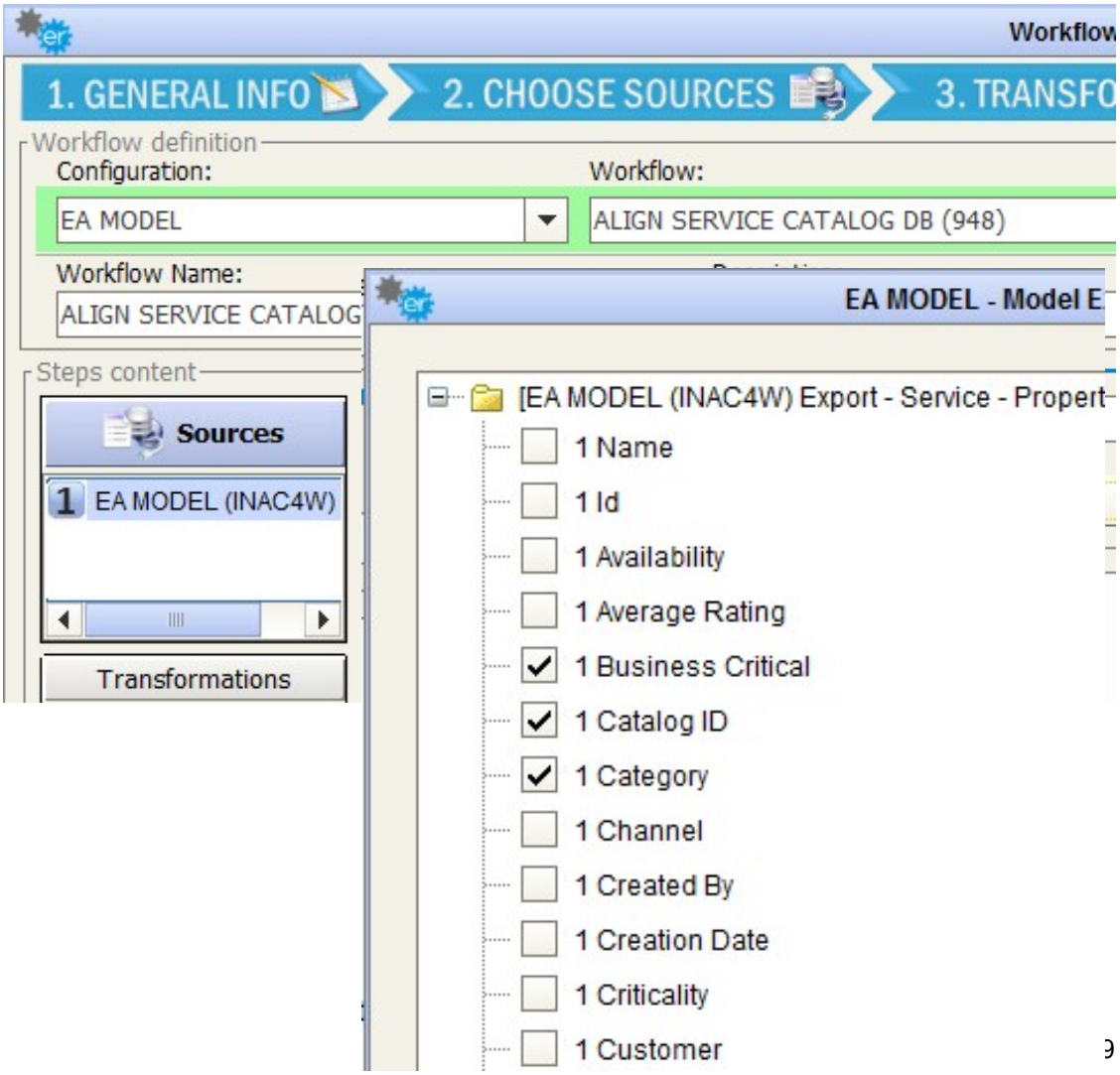
with the proper DB type, server, port, database name, user, password for the target

- CREATE WORKFLOW TO ALIGN SERVICE CATALOG DATABASE

  Press the “Workflow configuration” button on the Home Page and configure the following workflow choosing the model configuration according to requirements (please note that the model configuration is mandatory for a given model, only for “Load, Sync or Delete” operation against that model)

  SOURCE DEFINITION – Source 1

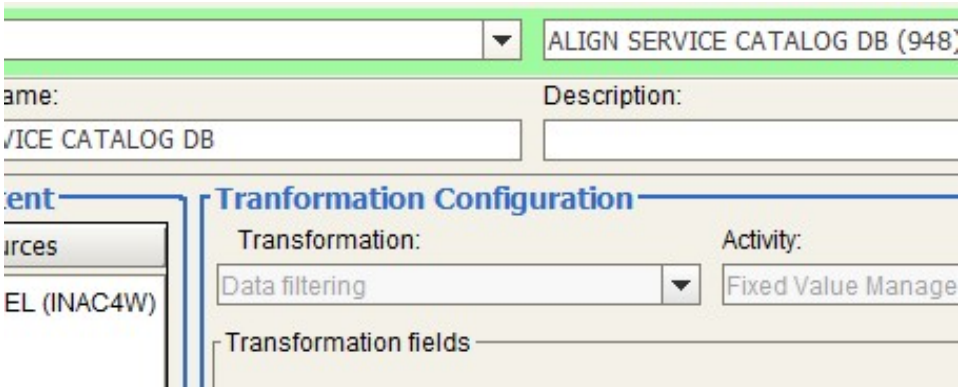
  Choose CM Adapter and the specific CM Model Export adapter from the dropdown list; then pick the “Service” object type from the list and check the desired properties / associations from the “CM Filters” interface:



TRANSFORMATION 1 – Filter services at export time

To optimize the data flow, filter source data in order to export only the more recently updated. Use then a “Data filtering”, “Fixed Value Manager” to manage only services with “Updated Date” up to 3 days before the operation (i.e. running the workflow every 3 days, they will be only needed services, if updated in the last 3 days).

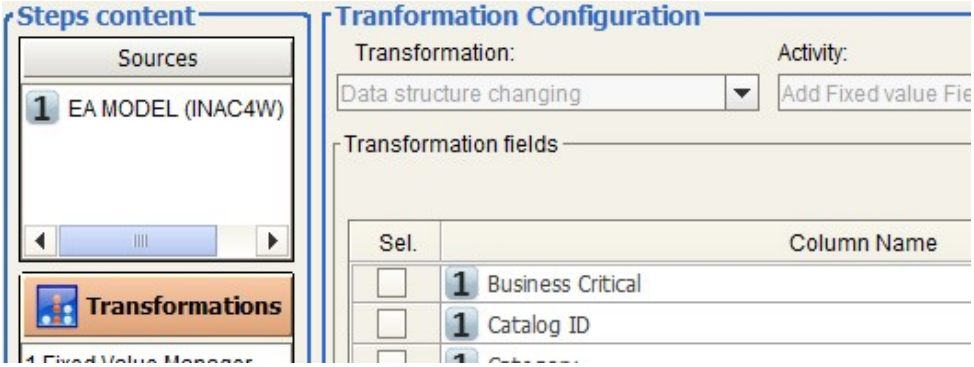
Please note that filters defined as the first transformation will be shown in the source field list (see picture above) and executed at export time (if a CM Adapter is the source for filter fields):

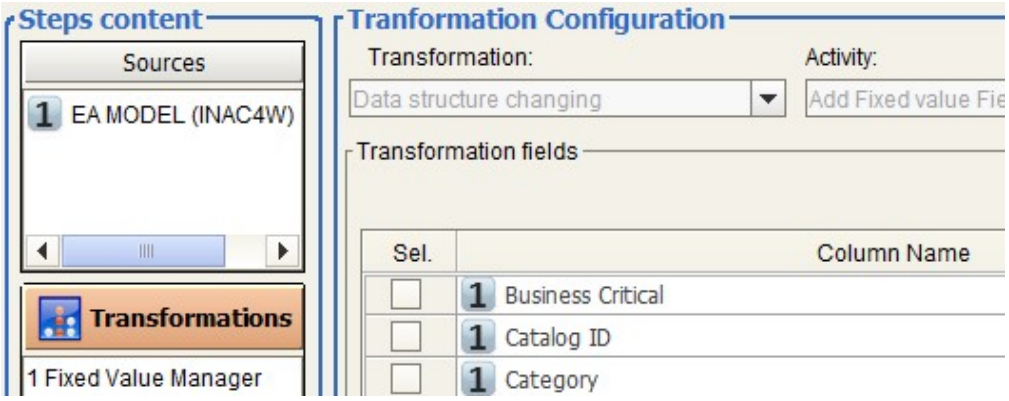


TRANSFORMATION 2/3 – Add Fixed Value fields for target Business Critical fields

The target database stores the Business Criticality information in a bit field, with “0” or ”1” values. In CM there is a checkbox field, exported by CM Adapter as “True” / “False” values.

A text conditional replace is then needed so that two different fixed value fields, with values 0 and 1 respectively, replace “False” and “True” in the following replace transformation (see later):

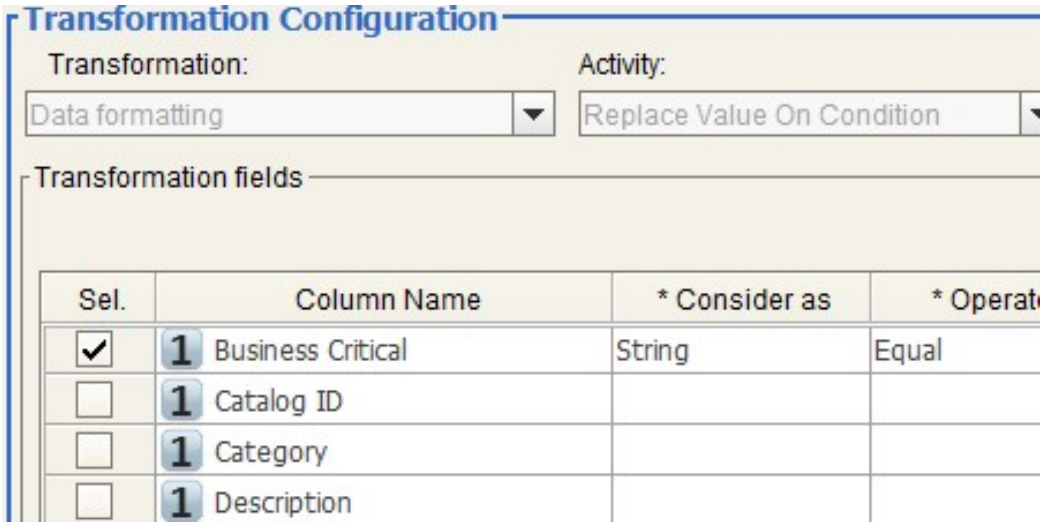




TRANSFORMATION 4 – Prepare Application name column into Vendor dataset

Vendors are to be imported as Application associated objects, but the Vendor file does not contain the application name, which is needed to execute the operation.

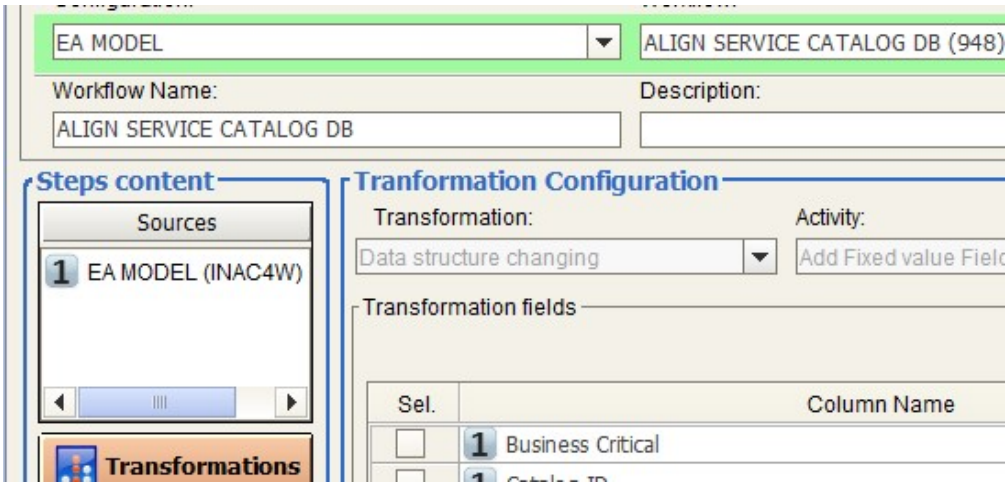
A “Replace Value on Condition” is needed to decode the “True”/”False” value for the source field hosting Business Criticality information in Modeler to the target 1/0 format:



TRANSFORMATION 5 – Add Fixed Value to fill a datetime target field

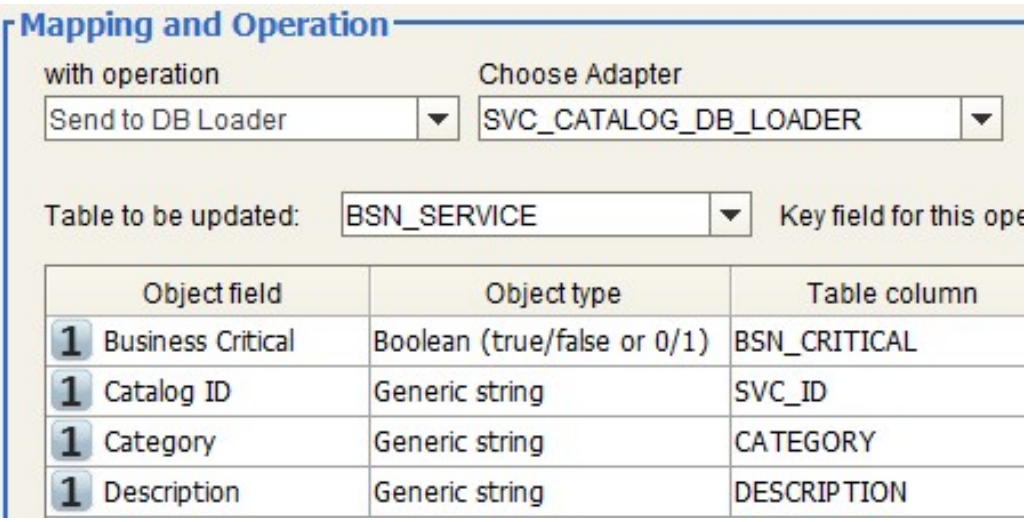
Let’s suppose that a datetime field in the target table is uploaded with the DT execution datetime, for any created/updated record.

In such a case, a fixed value field, with value equal to parameter “<%SYSDATE_EUR%>” (to get a European “DD/MM/YYYY” format, or “<%SYSDATE_USA%>” for a “MM/DD/YYYY” format or “<%SYSDATE%>” for a standard “DDDD/MM/YY” UTC format):

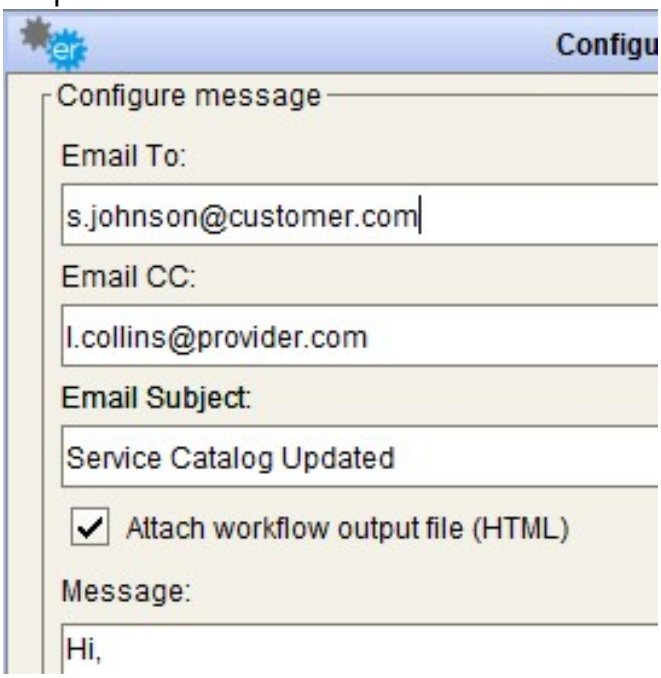


OPERATION – Send to DB Loader

In the “Mapping and Operation” step, choose the previously configured DB Loader, the target table, and the target field to be used as a key for the operation. Then map all the relevant source fields to target fields. In case of datetime, boolean or numeric target fields, the source datatype must be specified to ensure a correct data transfer:

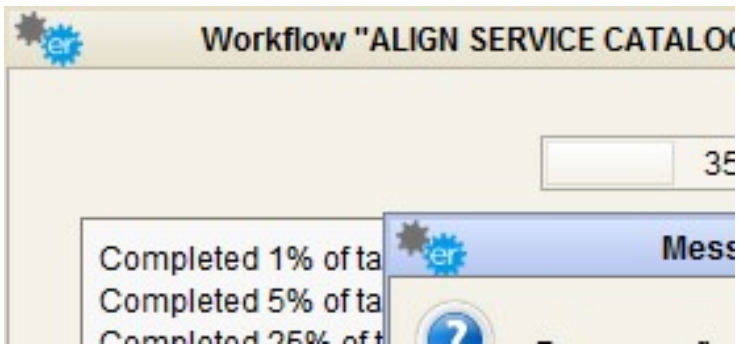


Email can be configured with default text (simply filling the “email to” field) or with custom properties (“Conf”). Please refer to the “Email configurations” chapter for a detailed description on the email configuration options.



- TEST THE WORKFLOW AND CHECK RESULT

  Test the configured workflow with the “Test” button, and follow its execution through the progress bar or Logs and email notifications. At the end of the workflow execution, check the result in the target database:



Results | Messages

	SVC_ID	NAME	DESCRIPTION	CATEGORY	SERVICE_MANAG
1	C.PRE.A1	Prestazioni sanitarie		Service Area	Graham Bering
2	S.ICT.A4	Information Services		Service Area	Alex Williams
3	S.ICT.A4.M01	Analytics Services		MacroService	
4	S.ICT.A4.M01.S01	Servizi Analitici e di Reportistica	Messa a disposizi...	Service	
5	S.ICT.A4.M02	Data Services		MacroService	

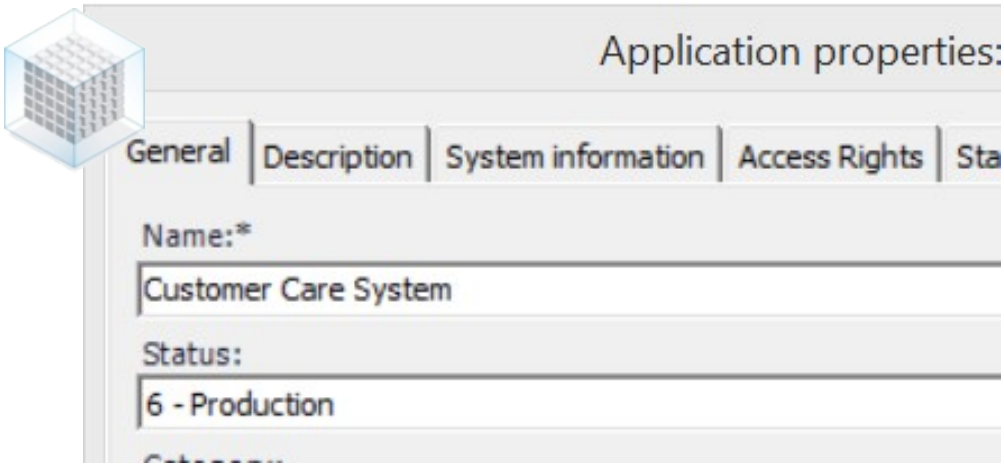
1.3 Align Federated Models

This example describes the following use case: an enterprise organized its EA information in two federated models, one specifically aimed to support IT and the second for more general business analysis purposes. The two models partially share the same metamodel, given to different IT and business teams requirements.

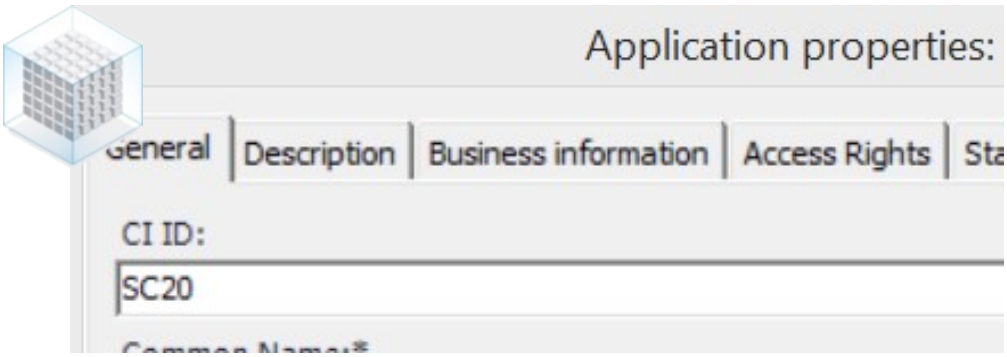
The first model is used to synchronize the second about application data, but only for a specific subset of applications, i.e. applications with a given lifecycle status (“production”) and only for those coming from the enterprise application catalogue (the source of application information for IT models, while the second hosts applications from other sources, relevant for the business).

Let us suppose that:

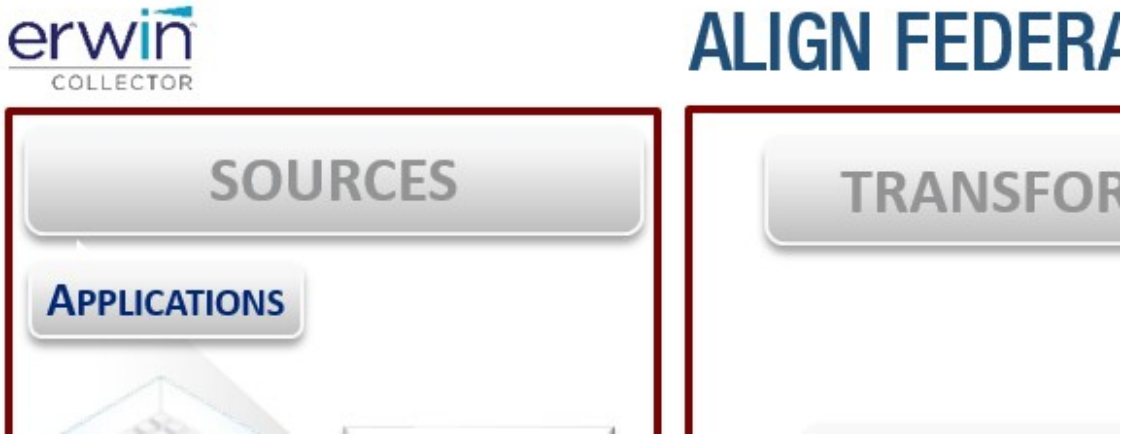
- **Application data** are in the source **CM model** in a dedicated **object type:**



- a **second CM Model** hosts **application data**, which is needed to execute Business Analysis in a dedicated object type, and a slightly different metamodel from the IT Model:



A DT workflow aligning the database of two CM models can be configured with the following steps:

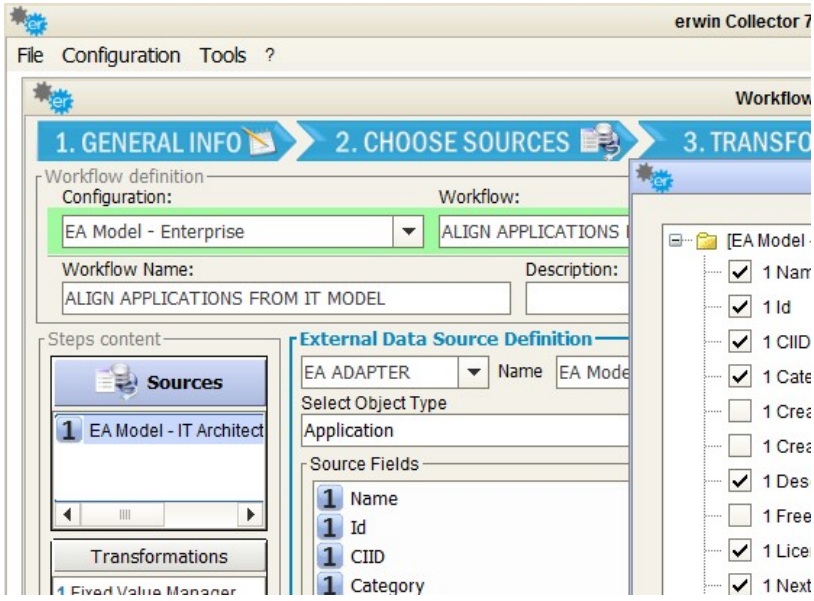


- CREATE WORKFLOW TO ALIGN FEDERATED MODELS

  Press the “Workflow configuration” button on the Home Page and configure the following workflow choosing the model configuration according to requirements (please note that the model configuration is mandatory for a given model, when the “Load, Sync or Delete” operation must be executed on that model)

  SOURCE DEFINITION – Source 1

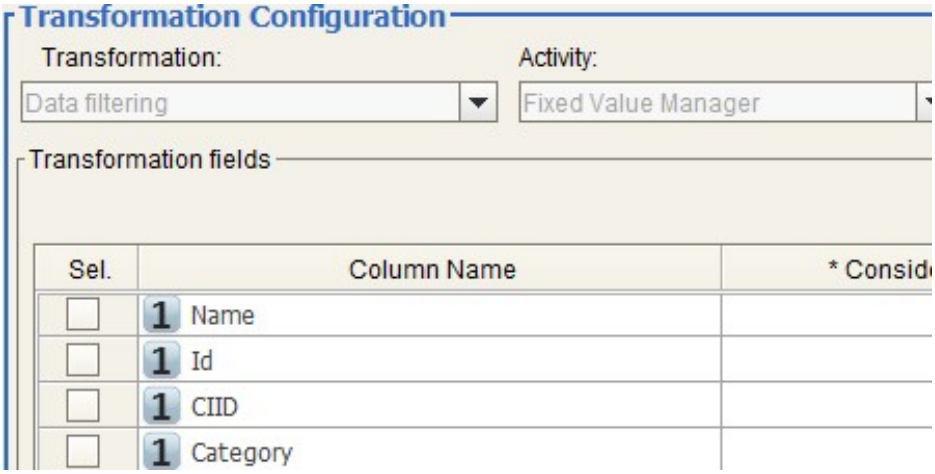
  Choose the CM Adapter and specific CM Model Export adapter (i.e. the IT Model one) from the dropdown list; then pick the “Application” object type from the list and check the desired properties/associations from the “Source Filters” interface:



TRANSFORMATION 1 – Filter Application in "Production"

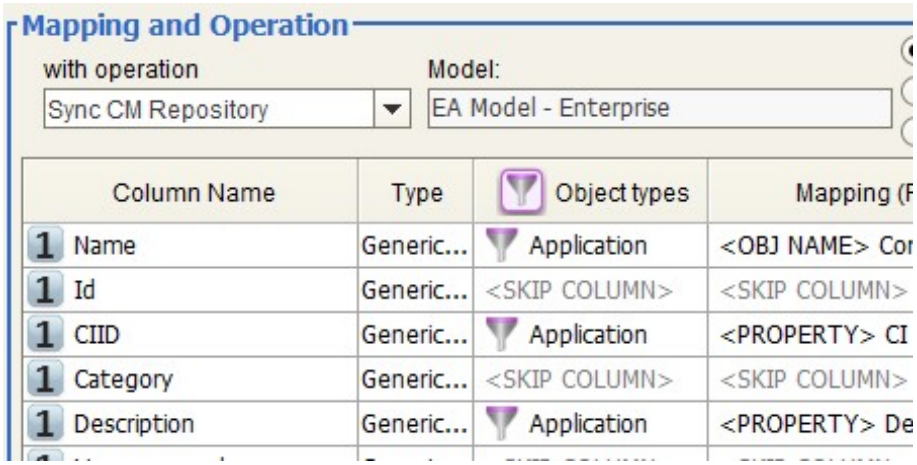
To optimize the data flow, filter source data in order to export only the more recently updated. Use then a "Data filtering", "Fixed Value Manager" to manage only services with "Status" equal "6 – Production".

Please note that filters defined as the first transformation will be shown in source field list (see picture above) and executed at export time (if an EA Adapter is the source for filter fields):



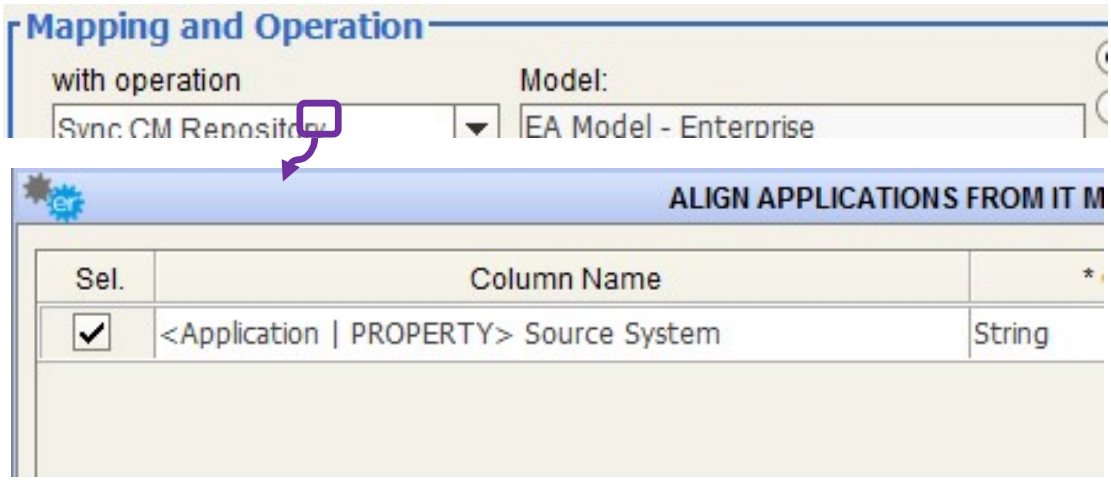
OPERATION – Sync CM Repository

In the "Mapping and Operation" step, choose "Sync CM Repository". The target model is the one related to the Model Configuration the workflow has been assigned to. Then map all the relevant source fields to the target fields for "Application" object type. Define the desired synchronization action ("Logical", "Physical" or "Physical Forced" deletion, which will affect redundant objects in the target model in comparison with the exported list from the source model):



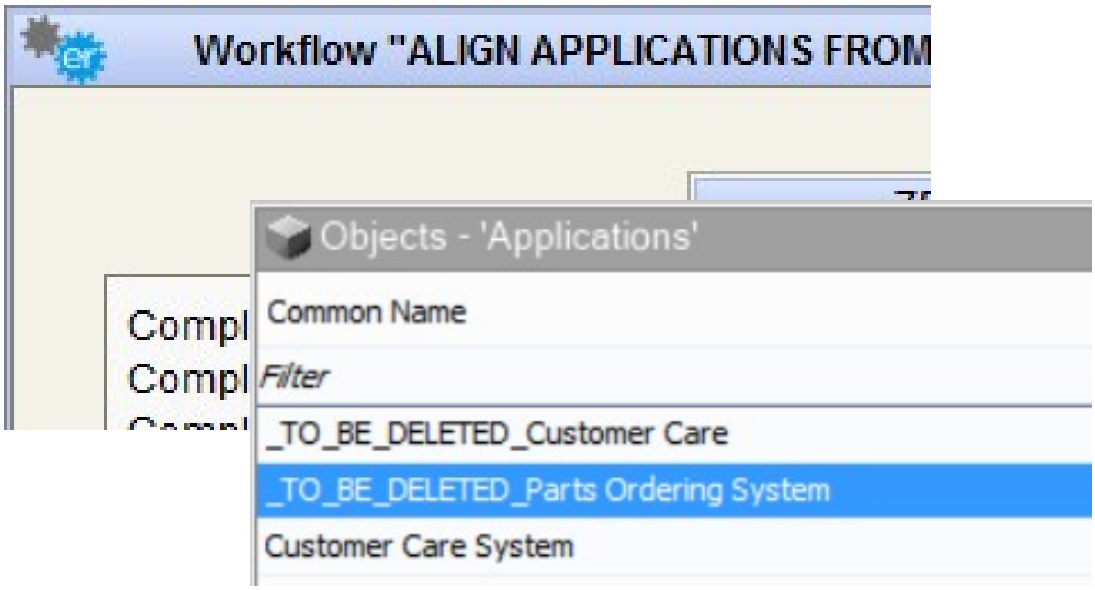
Please note the keyset on the operation defined to be the CI ID of the applications is a reasonable setting because the second model receives application data from various sources. For object types with instances coming only from the source model, it can be also used the Erwin Id.

To apply the synchronization within requirements (it must occur only for applications with a specific source system - the Application Catalogue) configure a filter for the synch operation, by double clicking on the filter icon on the “Object Type” column:



- TEST THE WORKFLOW AND CHECK RESULT

  Test the configured workflow with the “Test” button and follow its execution through the progress bar, Logs and email notifications. At the end of workflow execution, check the result in the target CM Model:

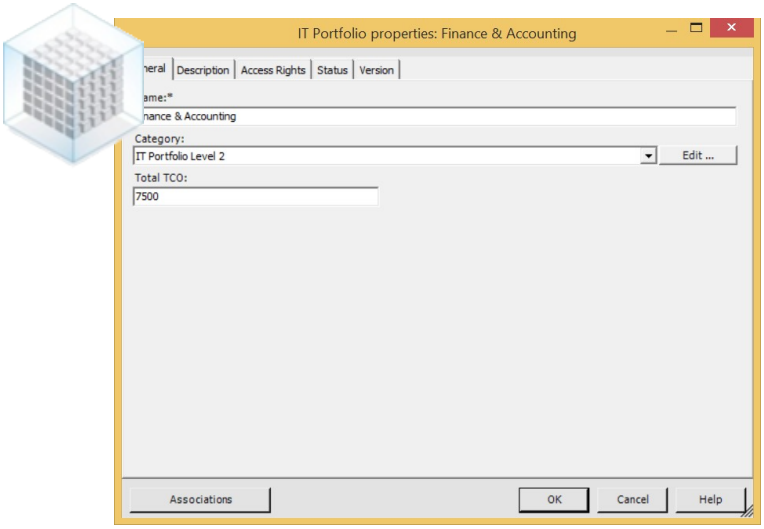


1.4 Deriving values and update consistency data

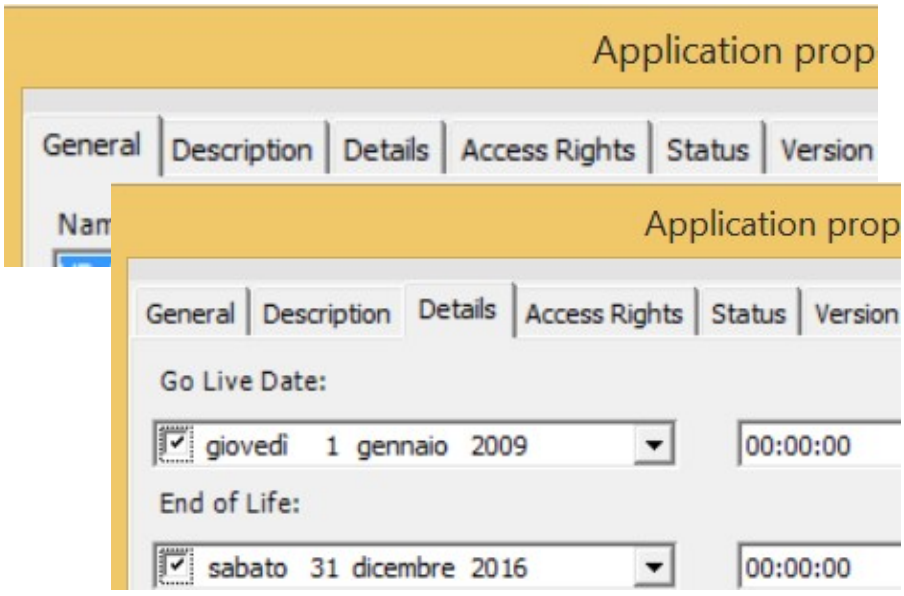
The example describes the following use case: an enterprise organized its EA information in a model which supports general business analysis purposes with reports and dashboards. One of them is about IT Portfolios, and shows summary information to managers.

Let us suppose that:

- **IT Portfolio data** are in the source **CM model** in a dedicated **object type:**



and the "**Total TCO**" value must be calculated as the sum of associated **application TCO**, for the ones of "**ERP**" **category** and a **number of users greater than 100**.



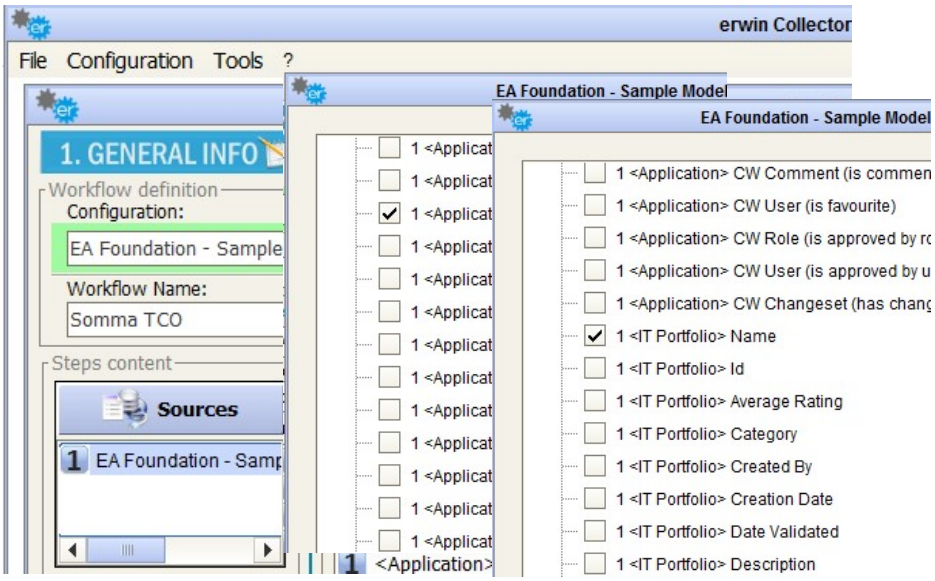
A DT workflow calculating Total TCO for IT Portfolios can be configured with the following steps:

- CREATE WORKFLOW TO DERIVE VALUES AND UPDATE CONSISTENCY DATA

  Press the “Workflow configuration” button on the Home Page and configure the following workflow choosing the model configuration according to requirements (please note that the model configuration is mandatory for a given model, when the “Load, Sync or Delete” operation must be executed towards that model)

  SOURCE DEFINITION – Source 1

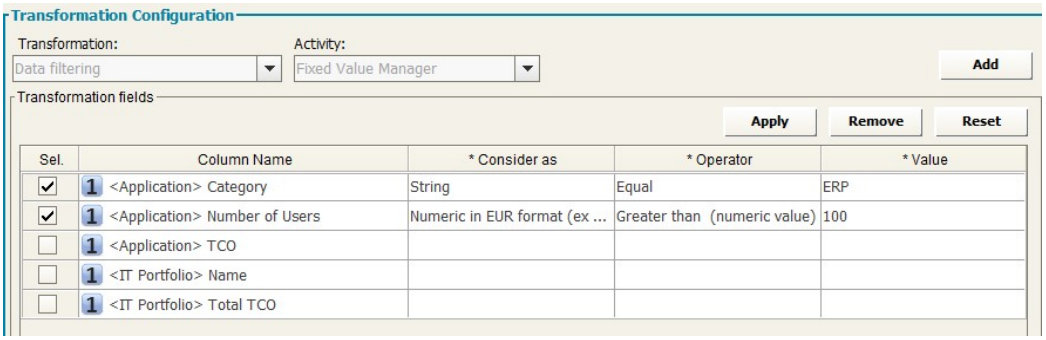
  Choose CM Adapter, and specific CM Model Export adapter (i.e. the IT Model one) from the dropdown list; the select “Association IT Portfolio-Application” from the Object Type list, and select <Application> Category, <Application> TCO, <Application> Number of Users, <IT Portfolio> Name, <IT Portfolio> Total TCO fields:



TRANSFORMATION 1 – Filter application by "category" and "number of users"

To export the relevant application, set a filter on "category" and "number of users" with "DATA FILTERING"-"Fixed Value Manager" to manage applications having:

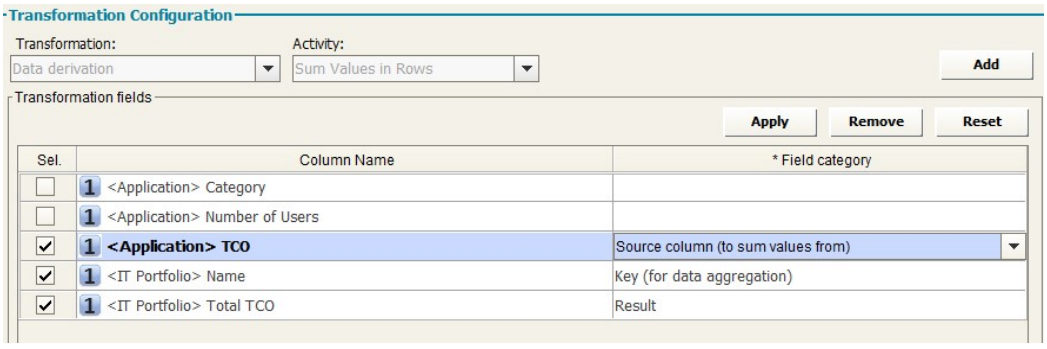
- "Category" equal to "ERP"
- "Number of users" greater than "100".



TRANSFORMATION 2 – Derive IT Portfolio "Total TCO" from associated applications

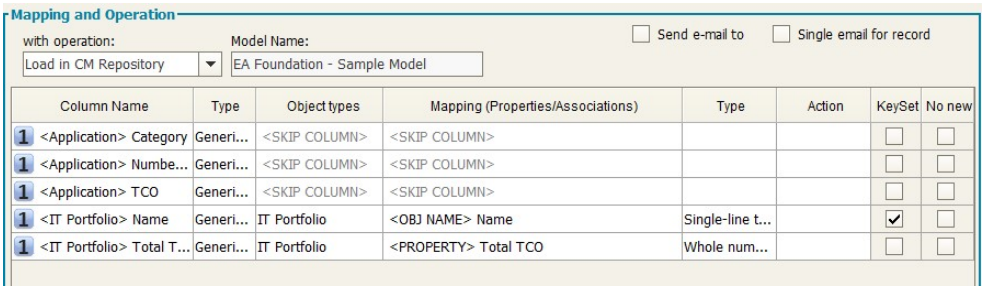
Choose a "DATA DERIVATION", "Sum Values in Column" transformation and set:

- The "<Application> TCO" column as "Source Column (to sum values from)"
- The "<IT Portfolio> Name" column as "Key (for data aggregation)"
- The "<IT Portfolio> Total TCO" as "Result" for the transformation



OPERATION – Sync CM Repository

In the “Mapping and Operation” step, choose the “Load CM Repository”. The target model is the one related to the Model Configuration which the workflow has been assigned. Then map all the relevant source fields to target fields:



- TEST THE WORKFLOW AND CHECK RESULT

  Test the configured workflow with the “Test” button and follow its execution through the progress bar, Logs and email notifications. At the end of the workflow execution check the result in CM Model IT Portfolios: