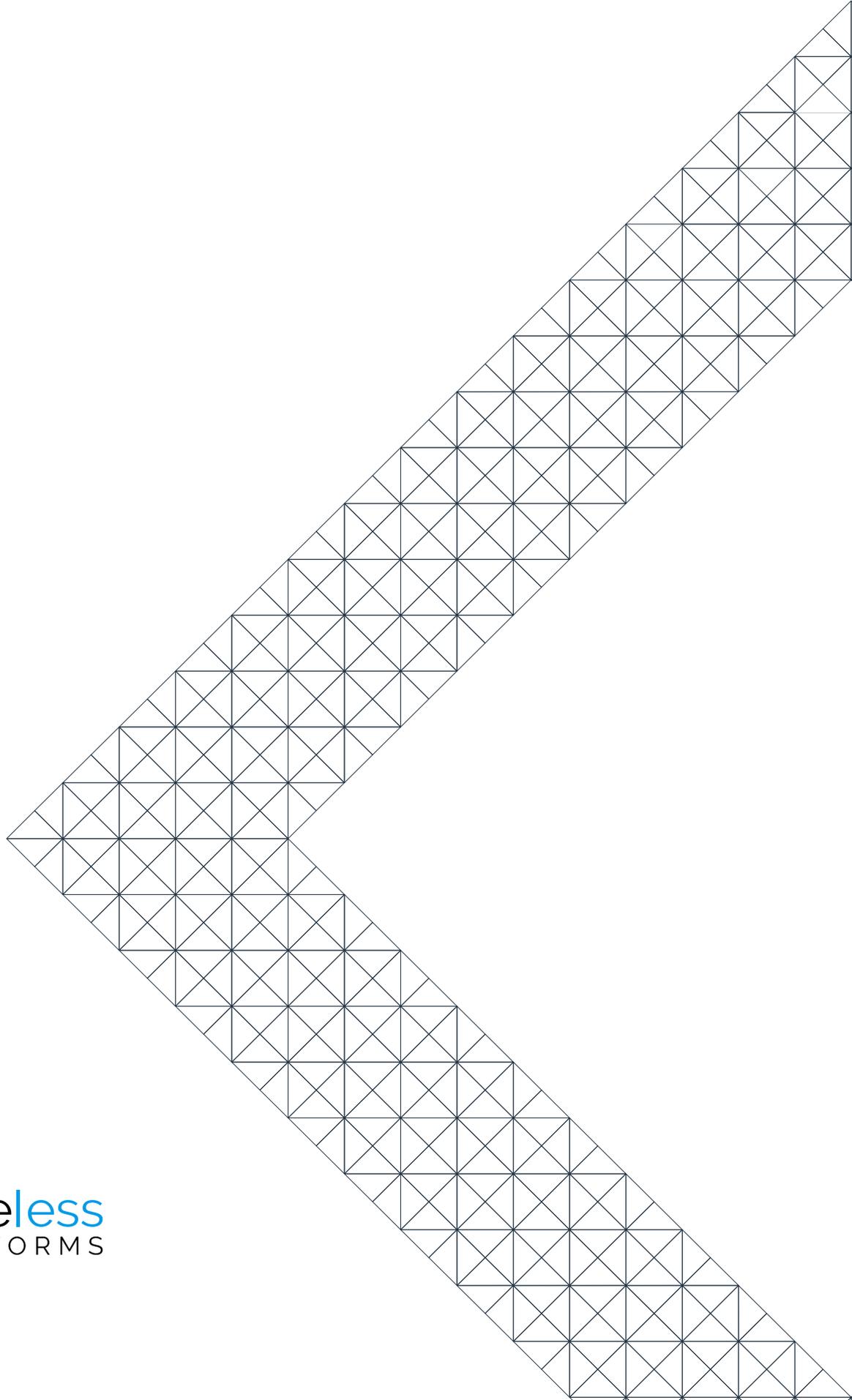


TaskCentre

Transform Data Tool

White Paper



Copyright

The copyright in this document is owned by Orbis Software T/A Codeless Platforms 2017. All rights reserved.

This publication may not, in whole or part, be reproduced, transmitted, transcribed, stored in a retrieval system or translated into any language in any form or by any means without the prior written consent of Orbis Software T/A Codeless Platforms.

Head Office:

Codeless Platforms
Suite 1 & 2 Bourne Gate
25 Bourne Valley Road
Poole
BH12 1DY
United Kingdom
Tel: +44 (0) 330 9988 700
Email: enquiries@codelessplatforms.com

Trademarks

Orbis Software T/A Codeless Platforms owns the registered Trademark 'TaskCentre'.

All other Trademarks used within this documentation are acknowledged as the property of their respective owners.

The information provided in this publication may contain errors, omissions, or typographical errors or may be out of date. Orbis Software T/A Codeless Platforms may change, delete, or update any published information at any time and without prior notice. The information published in this document is provided for informational purposes only and is not binding on Orbis Software T/A Codeless Platforms in any way except to the extent that it is specifically indicated to be so.

Contents

Introduction	2
Features.....	2
Working with other steps.....	2
Consuming from other steps	2
Exposing to other steps	3
Global Configuration.....	3
Step Configuration	3
General Tab	4
Main Tab.....	5
Mapping Tab.....	6
Function Types	7
Options Tab	9
Objects Consumed	9
Objects Exposed	10
User Security	11

Introduction

Transform Data is a **General** tool that enables a new structure to be created from an existing XML or recordset data source, or for an XML structure to be created from new. The output generated by this tool is ready for use by other XML consuming tools in TaskCentre; the XML may also be saved to file.

You may wish to use this tool where a requirement exists to manipulate data into a customised structure such as flattening or merging areas of content, or to change the datatypes or value formats in an XML structure.

Features

- Consume XML or Recordset data from existing steps as the input
- Replicate the schema of the input data source
- Build on existing XML structures for output to other tools
- Create new XML structures starting at the root
- Output to other tools in XML format

Working with other steps

The other steps that the **Transform Data** tool will directly interact with to consume XML or Recordset data from are listed below. This data will form the input structure to be manipulated.

Consuming from other steps

Step Icon	Name	Step Type
	XML Document Import	Input
	Flat File Import	Input
	Database Query (ODBC)	Input
	Database Query (OLEDB)	Input
	Call Procedure (OLEDB)	Input
	Recordset to XML	Format
	XML to Recordset	Format
	Data Filter	General

Step Icon	Name	Step Type
	Transform Data	General
	Web Service Connector	Execute

Exposing to other steps

A Transform Data step can directly expose its XML output for use by the following tools listed below:

Step Icon	Name	Step Type
	XML to Recordset	Format
	Save as File	Output
	Transform Data	General
	Web Service Connector	Execute

Global Configuration

No global configuration exists for this tool; it can be used in a task immediately.

Step Configuration

The Transform Data tool is dragged from the task browser under General tools and into the task design area. This will automatically open the step for configuration where you will be presented with the following four tabs:

- **GENERAL TAB**
- **MAIN TAB**
- **MAPPING TAB**
- **OPTIONS TAB**

General Tab

The **General** tab allows you to name and describe your **Transform Data** step and to define the input source to be used in the Mapping tab at runtime.

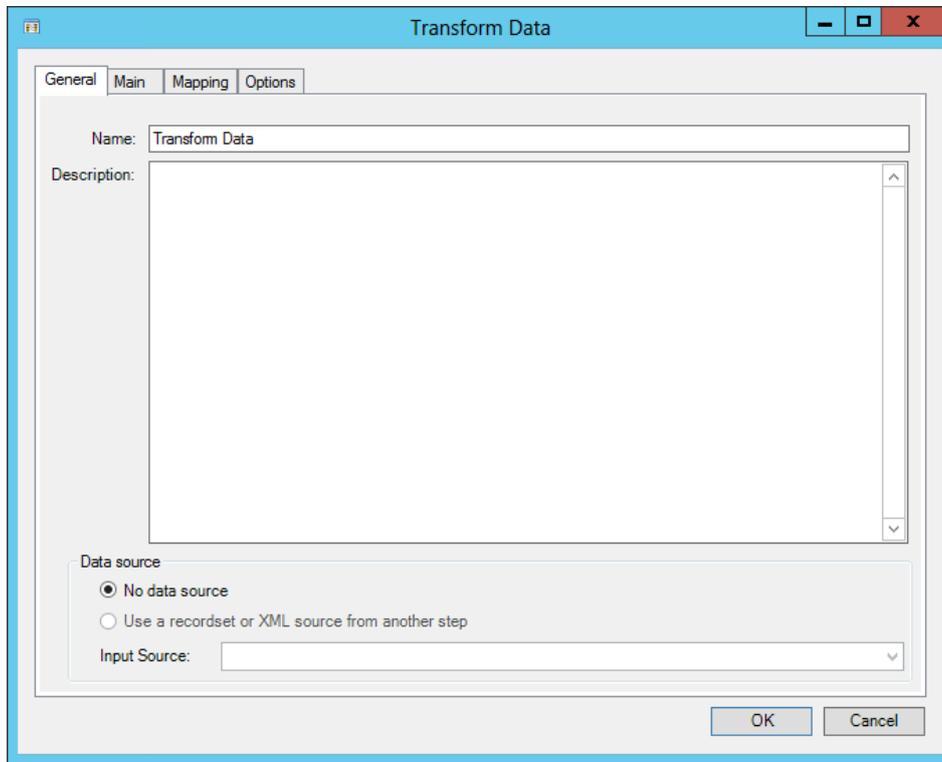


Figure 1. Transform Data tool — General tab

Data source is the schema that you wish to manipulate so a new XML structure can be formed and ultimately passed to your destination system. Alternatively, the content of an existing structure may be amended and built upon.

Two options are available:

- **No Data Source**

Select this option if you wish to dynamically create an XML document within the step. To achieve this go to the **Main** tab to create your required output structure. Using variables, you can map to the output within the **Mapping** tab to generate the XML output at runtime.

- **Use a recordset or XML source from another step**

Selecting this option will display all steps in the task that natively output XML or Recordset data via the **Input Source** drop down list. Select the step you require which will supply the schema for you within the **Mapping** tab.

The schema will appear in the left-hand pane labelled **Input Data**.

Main Tab

The **Main** tab is used to design your required XML structure for output by the tool, allowing the creation of nodes, elements and attributes.

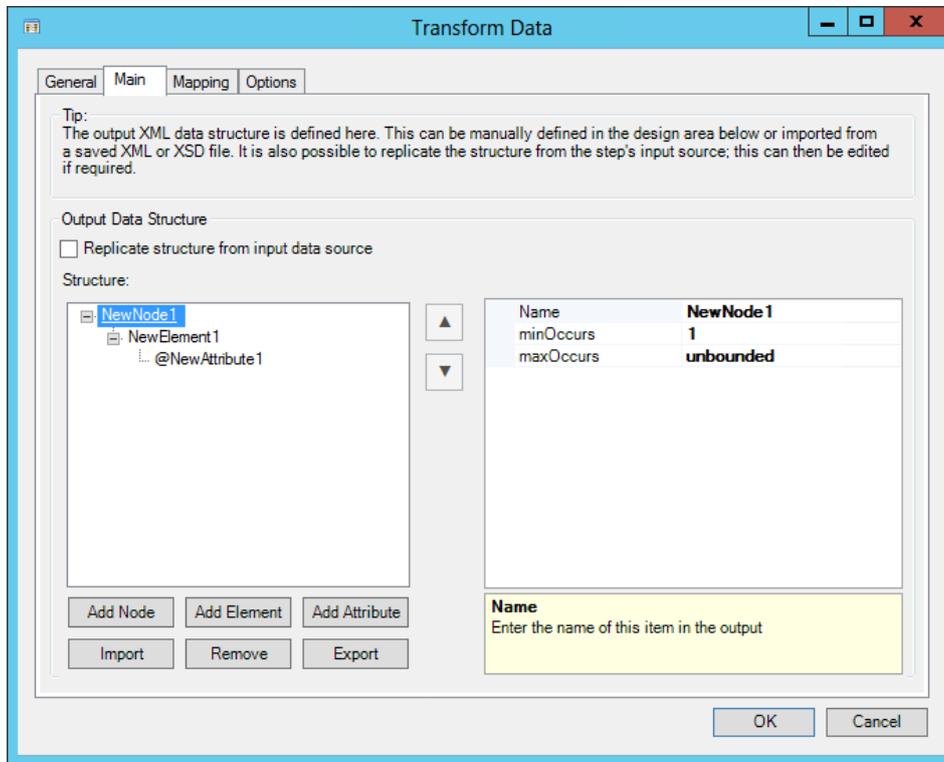


Figure 2. Transform Data tool — Main tab

The structure you create here will appear within the **Mapping** tab in the right-hand pane labelled **Output Data**, and will be made available to subsequent steps in a task.

There are a number of methods that you can use to define the structure:

- **Replicate structure from input source** — This will automatically generate a structure for you which will be identical to the one of the XML or Recordset source selected within the **General** tab.

When a structure has been replicated this provides the base for you to work with, it can be left as it is or you can amend the structure by using the appropriate 'add' or 'remove' buttons within the UI.

- Create and edit the structure manually using the **Add Node**, **Add Element**, and **Add Attribute** buttons.
- Use the **Import** button to import a predefined XML or XSD file to automatically generate the structure.

Using the **Export** button, the structure may be saved if you wish to quickly recreate it again within other Transform Data steps, or if you wish to simply keep the custom structure as part of a backup. An .XSD file will be created on export.

NOTE: You can set the properties of each node, element, and attribute using the settings `Type`, `minOccurs`, and `maxOccurs`. These settings only apply to the nodes, elements and attributes of the output structure to validate the data being mapped to them. Therefore, during design you must consider the data types and how many times a value is expected to be seen otherwise task failure may occur.

Mapping Tab

The **Mapping** tab is used to define links between the input XML or Recordset data and the new XML output structure to be generated by the tool at runtime. Variables may also be dragged from the browser as an additional source for mapping.

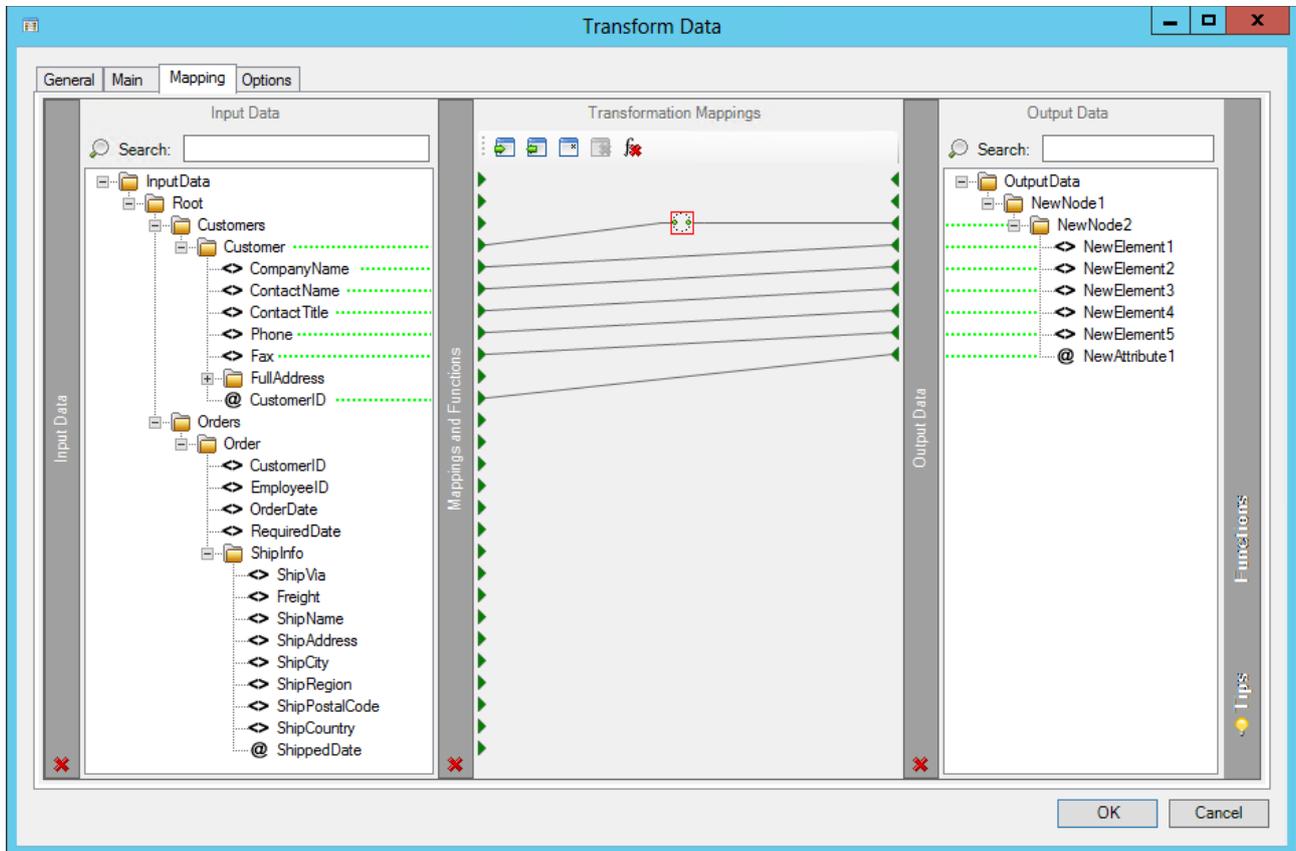


Figure 3. Transform Data tool — Mapping tab

If you have previously selected **No Data Source** in the **General** tab then by using variables alone you can create an input source “on-the-fly”.

The **Input Data** pane on the left reflects the structure of the data source if one was selected from the **General** tab. The **Output Data** pane on the right reflects the designed XML structure from the **Main** tab.

To create a mapping you drag an element from the left pane on to an element in the right pane. The **Transformation Mappings** pane in the middle will then display a visual link indicating that a mapping has been completed. At task runtime, data will be passed as per the mappings.

Function Types

A number of functions exist within the tool which can help with the transformation of your data.

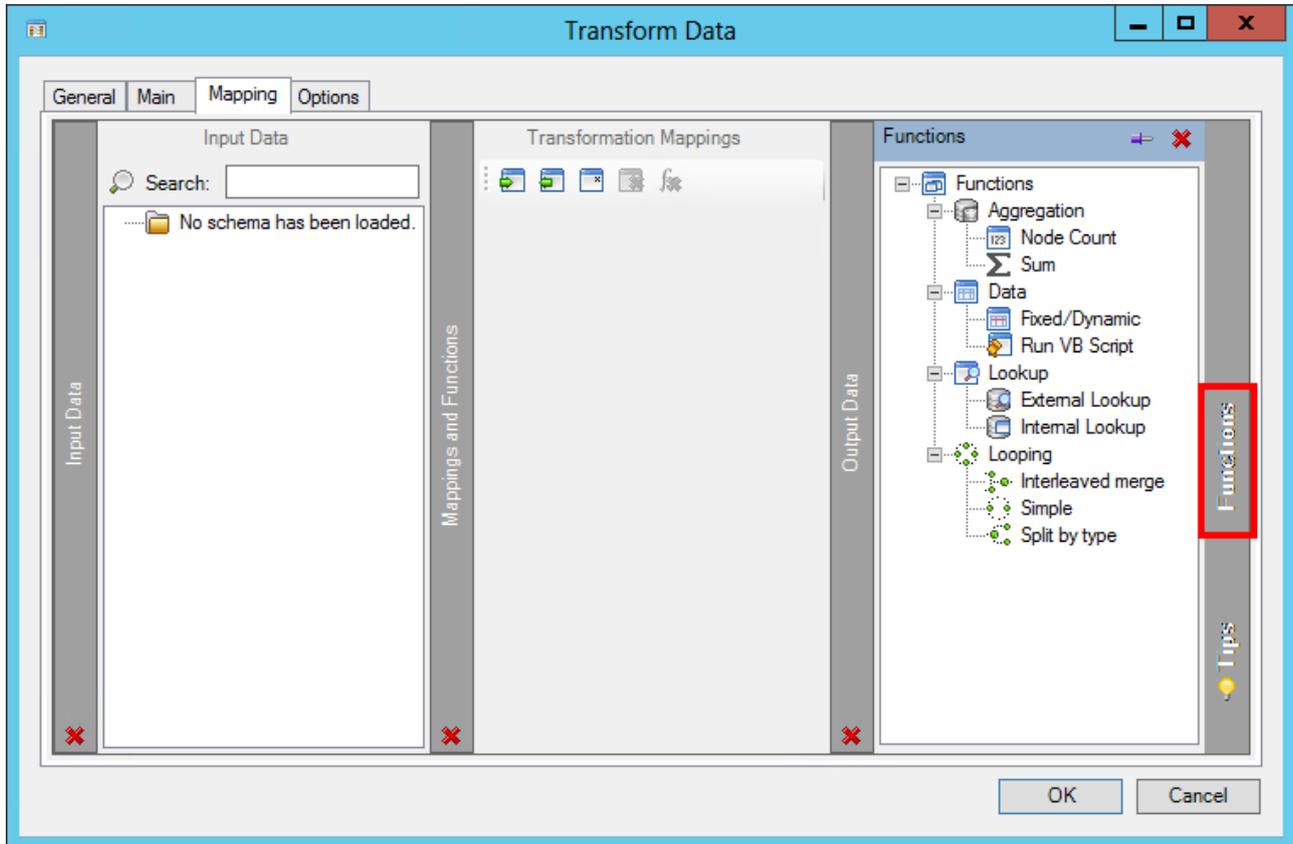


Figure 4. Transform Data tool — Mapping tab — Functions

The following functions are available:

- **Node Count**  — Requires input and output mappings. This function allows the number of iterations of a selected element in the input structure to be counted. The numeric value that is generated is then passed to an element within the output structure.
- **Sum**  — Requires input and output mappings. This function allows a numeric value to be produced based on the sum of several records contained in an element of the input structure. For example, the source XML or recordset may contain multiple sales orders, the SUM function can calculate the total net worth of the combined orders. Where string values exist, this will be a concatenation of the text.
- **Fixed/Dynamic**  — Requires only output mappings. This function has two types of functionality; the ability to supply a fixed value that never changes at runtime, or a dynamic value such as a TaskCentre variable or recordset object from the task browser where the value may change each time the task is run.
- **Run VB Script**  — Requires input and output mappings. This function allows complex data transformations, for example, to combine various input fields to produce a single output field, i.e. multiple address lines into one line. Alternatively, you may require a simple reformat of a date value which can be achieved via this function.

- **External Lookup**  — Requires input and output mappings. Uses an ODBC or OLEDB global connection to retrieve a replacement value from a database and insert it into the mapped element of the output structure. This is based on criteria set in the function using a field from the input structure.
- **Internal Lookup**  — Requires input and output mappings. Allows you to define a table of data within the function for use as a lookup. When criteria is met using a field from the input structure, a lookup is carried out and the relevant replacement value is passed from the table to the mapped element of the output structure.
- **Interleaved Merge**  — Requires input and output mappings. This function takes data from two or more elements and merge them to create one occurrence in the mapped element of the output structure. For a detailed description of how to use this function, refer to our knowledge base article [Transform functions – Interleaved merge looping](#).
- **Simple Loop**  — Requires input and output mappings. This simple loop function allows the tool to iterate through a set of records within an XML document or a TaskCentre recordset. For example, this would be beneficial for processing XML or recordset data where multiple sales orders exist. Without a loop, only the first sales order would be processed. This function is automatically placed for you when mappings are created between parent nodes of the input and output structures.
- **Split by type**  — Requires input and output mappings. This function takes data from an element of the input structure and splits it into two or more destination elements of the output structure. This is based on the criteria allocating it to each particular element. For a detailed description of how to use this function, refer to our knowledge base article [Transforms functions – Split-by-type](#).

Options Tab

The **Options** tab allows you to define how errors in the step are handled at runtime.

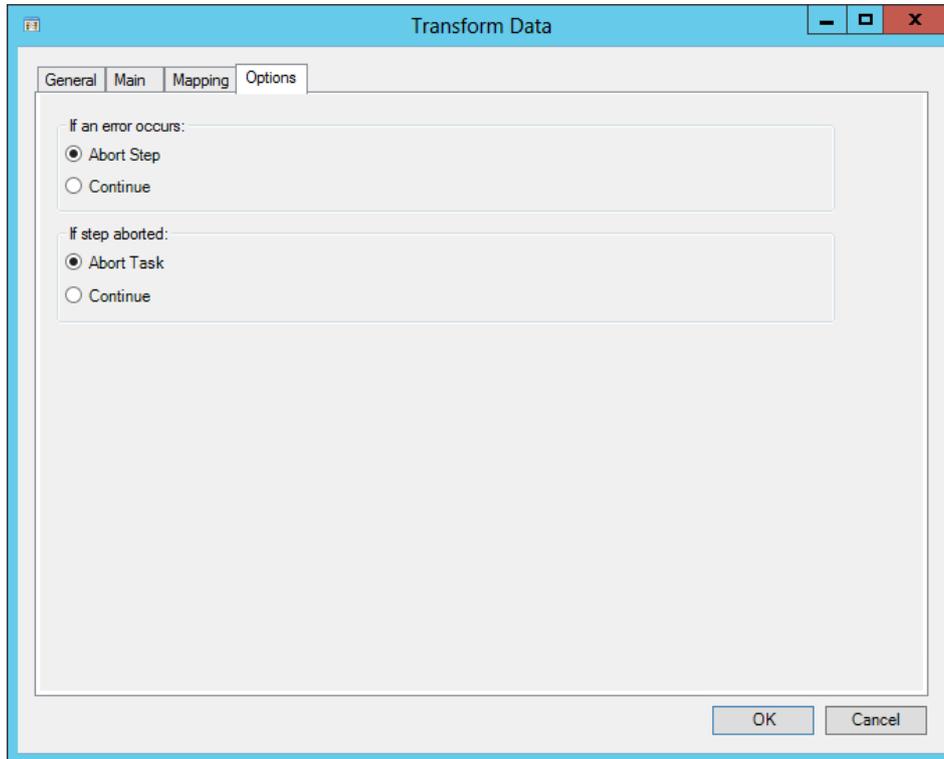


Figure 5. Transform Data tool — Options tab

For example, in the event of an error you can decide if you want to continue running and process the next step along, or abort and terminate the running of the whole task.

Any errors are recorded to the TaskCentre event logs for investigation.

Objects Consumed

Objects that the Transform Data step consumes that are exposed by other steps are listed below:

- **XML** — XML data from any TaskCentre tool capable of exposing XML, such as the Recordset to XML or XML Document Import tool.
- **Recordset** — Information structured into columns and rows produced by a tool, such as the **Database Query (ODBC)** tool.

Objects Exposed

Objects that a Transform Data step exposes that can be utilised by other tools are listed below.

- **OutputXML** — The `XmlString` object is the XML document produced by the tool. The XML structure will be the one designed within the Main tab and will contain values governed by the mappings and any transformations performed on the source data. The `XmlSchema` object contains the schema structure of the output XML as provided by the Transform Data tool.
- **ErrorXML** — The `XmlString` object contains the error data reported by the tool at runtime, this object will only become populated in the event of a step failure. Error messages posted to the TaskCentre event log will appear in the XML document. The `XMLSchema` object contains the schema structure of the error XML.
- **Step Properties** — Standard step properties are also available allowing you to utilise statistical data of the Transform Data step.

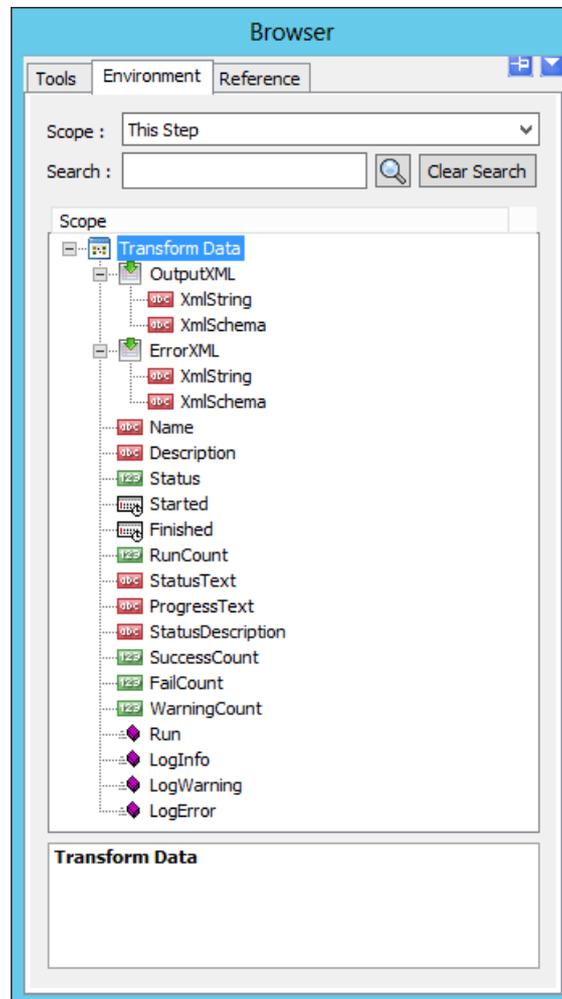


Figure 6. Transform Data tool — Task Browser objects

User Security

The Transform Data tool will appear within the **User Properties** dialog under the **Tool Access** tab. Only users with the tool checked will have permission to use the tool when building a task.

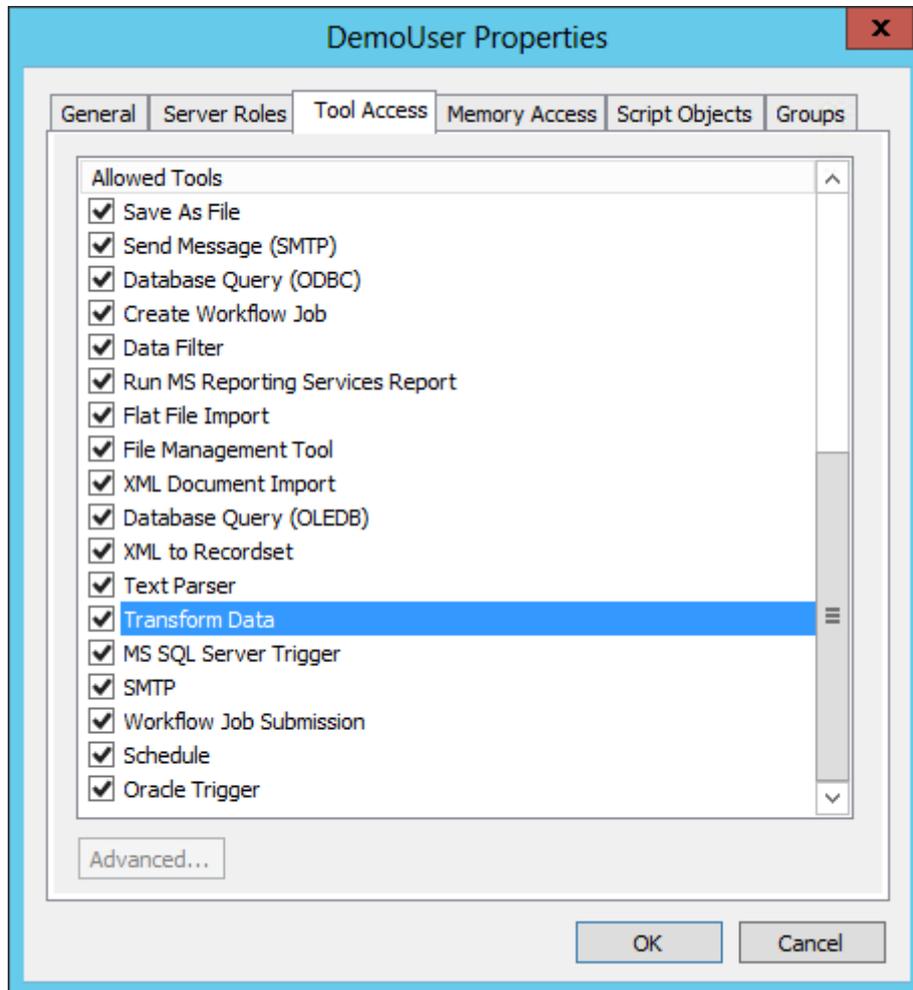


Figure 7. User Properties — Allowing access to use the tool

For further information about this tool, or about TaskCentre in general, please do not hesitate to contact us:

Codeless Platforms

Suite 1 & 2 Bourne Gate

25 Bourne Valley Road

Poole

BH12 1DY

United Kingdom

Tel: +44 (0) 330 99 88 700

Email: enquiries@codelessplatforms.com

