

BPA
Platform

Technical Overview

Filter Data Tool

Copyright

The copyright in this document is owned by Orbis Software Ltd T/A Codeless Platforms 2020. 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 Ltd 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 99 88 700

Email: enquiries@codelessplatforms.com

Trademarks

Orbis Software Ltd T/A Codeless Platforms owns the registered trademark "TaskCentre®".

All other Trademarks used 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The information provided in this publication may contain errors, omissions, or typographical errors or may be out of date.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 Ltd 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 Ltd T/A Codeless Platforms in any way except to the extent that it is specifically indicated to be so.

Contents

Introduction	1
Features	1
Repository Filters	1
Visual Basic Filters	1
Working with other Tools	2
Consuming from Other Tools	2
Objects Consumed	2
Exposing to Other Tools	3
Objects Exposed	4
Step Configuration	4
About the General Tab	5
About the Main Tab	6
About the Repository Filter Dialog	7
Summary View	8
Detailed View	9
About the Visual Basic Filter	14
About the Options Tab	16

Introduction

The **Filter Data** tool extracts a subset of data from a recordset produced by a **Database Query (ODBC)**, **Database Query (OLEDB)**, or **Call Stored Procedure (OLEDB)** step and provides the data for use in further steps.

Features

- ▶ Single or multiple filters
- ▶ Multiple data purging options

Repository Filters

When a task containing an Output step which uses Memory is run, a Repository is automatically created for the task containing the memorised recordset. It contains no data until the task has been run, unless data is then manually entered, but may still be selected for use in the Repository Filter.

When a **Filter Data** step is added to the task, the Repository can be selected and a Repository Filter created within the step. The Repository Filter is used to compare the new recordset created when the task is run again against the existing memorised recordset in the Repository, and then set rules to filter the new subset of data produced.

Visual Basic Filters

By adding a **Filter Data** step to a task, a Visual Basic Filter may be created that applies a filter expression against the selected data source to filter the rows.





The script specified in the filter is executed once for each row in the record source. If the expression returns "true", the row is transferred to the output recordset.

Working with other Tools

The **Filter Data** tool can interact directly with the following tools, whether by exposing information to it or consuming information from it:

Consuming from Other Tools

The **Filter Data** tool can directly consume output from the following tools:

Icon	Tool Name	Tool Category
	Database Query (ODBC)	Input and Data Connectors
	Database Query (OLEDB)	Input and Data Connectors
	Import Flat File	Input
	Convert XML to Recordset	Format


Objects Consumed

The following objects, outputted by other task steps, can be consumed by the **Filter Data** step:

- ▶ **Recordset** — Tabular data from any BPA Platform tool capable of exposing recordsets (see above)
- ▶ **Variables (Text)** — Textual data is extracted from selected variables

Exposing to Other Tools

The following tools can consume output from the **Filter Data** tool:

Icon	Tool Name	Tool Category
	Call Stored Procedure (OLEDB)	Input, Data Connectors, Output, and Execute
	Convert Recordset to XML	Format
	Create Workflow Job	Format
	Format as Flat File	Format
	Format as HTML	Format
	Format as HTML Pro	Format
	Format as Text	Format
	Run Crystal Report	Format
	Run Microsoft Reporting Services	Format
	Run Microsoft Word (Merge)	Format
	Transform Data	Format
	Save File	Output
	Transfer File (FTP)	Output
	Call COM Object	Execute
	Run External Program	Execute

Objects Exposed

Filter Data exposes the following objects; these can be consumed by the above tools:

- ▶ **RecordSource** — Displays the columns included in the recordset being used as the data source
- ▶ **Recordset** — A subset of the consumed recordset information structured into columns and rows
- ▶ **FilterDefinitions** — Displays the Filters assigned to the step and the data being used.
- ▶ **Step Properties** — A number of properties for the step are exposed in the Environment tab of the Task Browser

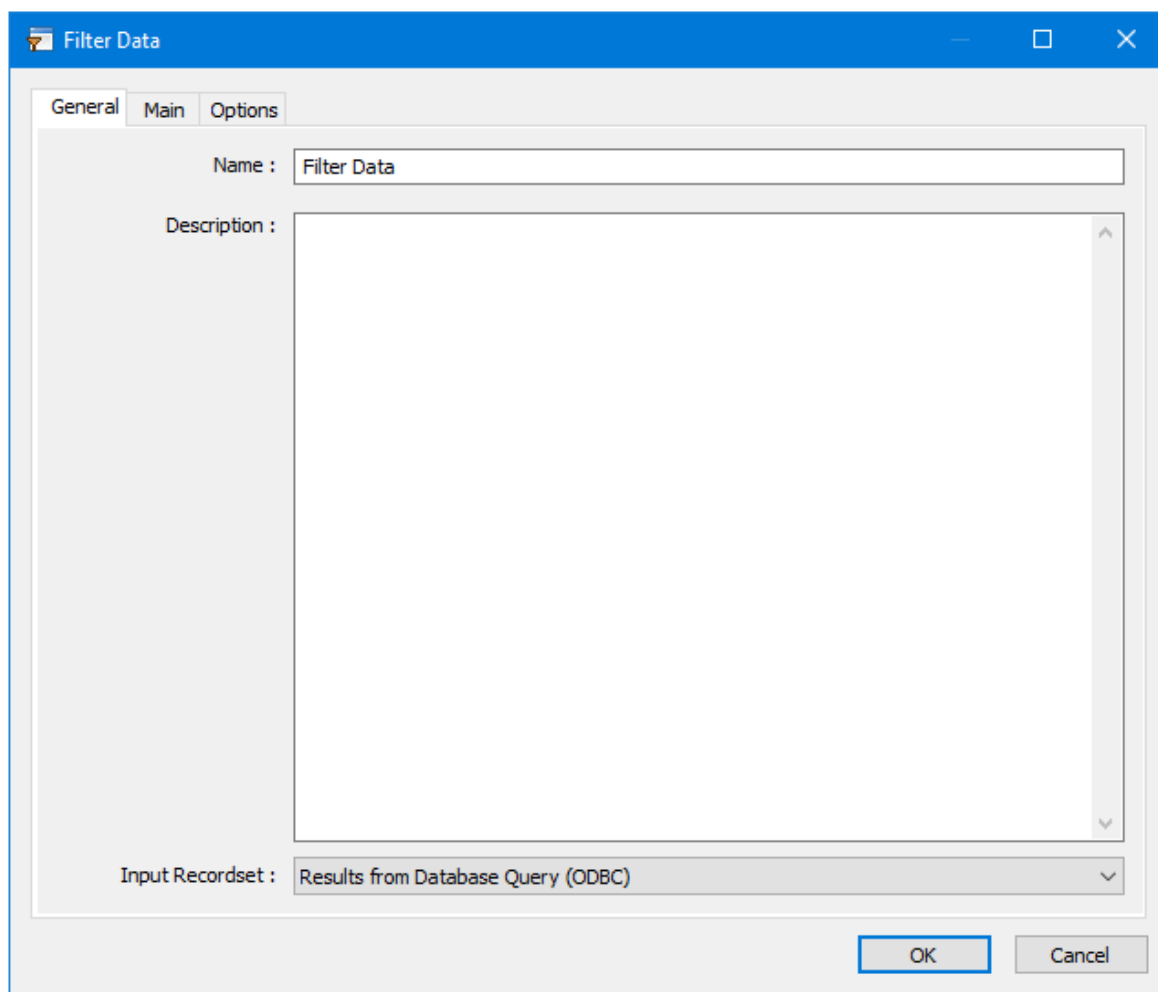
Step Configuration

To add a new **Filter Data** step to an existing task, you either:

- ▶ Click and drag the **Filter Data** icon from the **Task Browser** to the task **Design** area.
- ▶ From the task's **Design** tab, right-click on empty space and select **Add > General > Filter Data**.

For a detailed description of how to create new tasks, refer to the product help.

About the General Tab



The screenshot shows a dialog box titled "Filter Data" with a blue header bar. Inside, there are three tabs: "General", "Main", and "Options". The "General" tab is selected. It contains a "Name" field with the text "Filter Data", a "Description" field which is a large empty text area, and an "Input Recordset" dropdown menu showing "Results from Database Query (ODBC)". At the bottom right are "OK" and "Cancel" buttons.

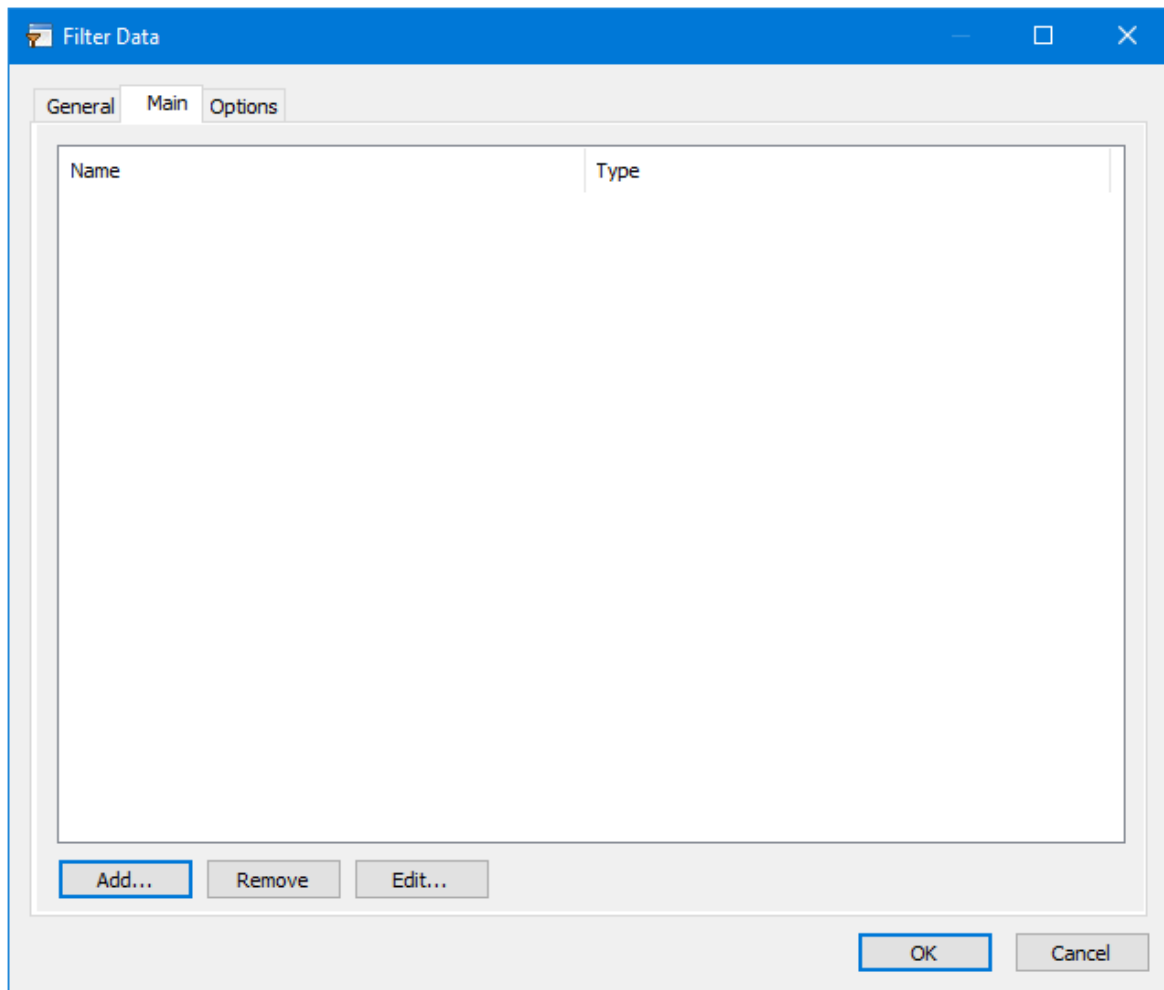
The **General** tab is used to enter the following details for the step:

- ▶ **Name** — Enter a meaningful name for the step

TIP: If this task instance makes use of two or more **Filter Data** steps, ensure the **Name** used is unique for each individual step.

- ▶ **Description** — If required, enter a description of this step
- ▶ **Input Recordset** — Displays a list of available recordsets from other task steps that the **Filter Data** step can use

About the Main Tab



The Main tab allows the creation and application of filters for filtering the recordset received from a **Database Query (ODBC)**, **Database Query (OLEDB)**, or **Call Stored Procedure (OLEDB)** Step.

The dialog box contains the items listed below.

- ▶ **Filter** — Displays a list of filters created for the Step
- ▶ **Add** — Initially opens the Choose Filter Provider dialog box which provides a list of Filter Providers for selection:
 - ☐ **Repository Filter** — Opens the [Repository Filter](#) dialog box to enable a new Repository Filter to be created
 - ☐ **Visual Basic Filter** — Opens the [Visual Basic Filter](#) dialog box to enable a new Visual Basic Filter to be created
- ▶ **Remove** — Removes the selected filter from the table
- ▶ **Edit** — Opens the applicable filter dialog box to enable details for the filter selected in the table to be edited

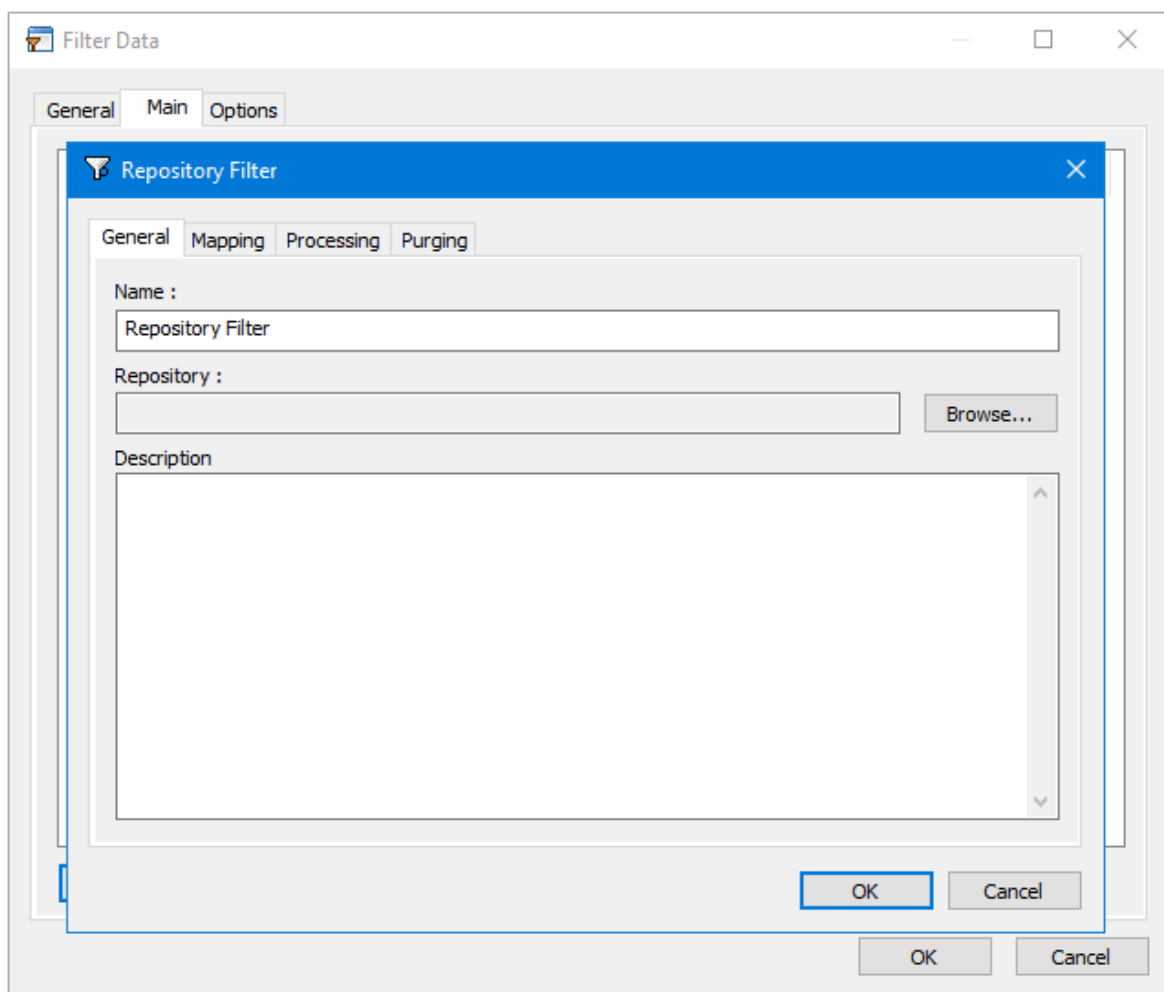
About the Repository Filter Dialog

The Repository Filter dialog box is displayed when the **Repository Filter** option is selected in the **Choose Filter Provider** dialog box. It is used to create new filters or edit details for existing ones.

When a task containing an Output step which uses Memory is run, a Repository is automatically created for the task containing the memorised recordset. It contains no data until the task has been run, unless data is then manually entered, but may still be selected for use in the Repository Filter.

When a **Filter Data** step is added to the task, the Repository can be selected and a Repository Filter created within the step. The Repository Filter is used to compare the new recordset created when the task is run again against the existing memorised recordset in the Repository, and then set rules to filter the new subset of data produced.

About the General Tab



The General tab is used to enter the following details for a Repository Filter:

- ▶ **Name** — Enter meaningful name for this filter
- ▶ **Description** — If required, enter a description of this filter

► **Repository** — Displays the Repository for this filter. Use **Browse** to [select a new Repository](#).

You create new Repositories in the Output step of this task.

Selecting a Repository

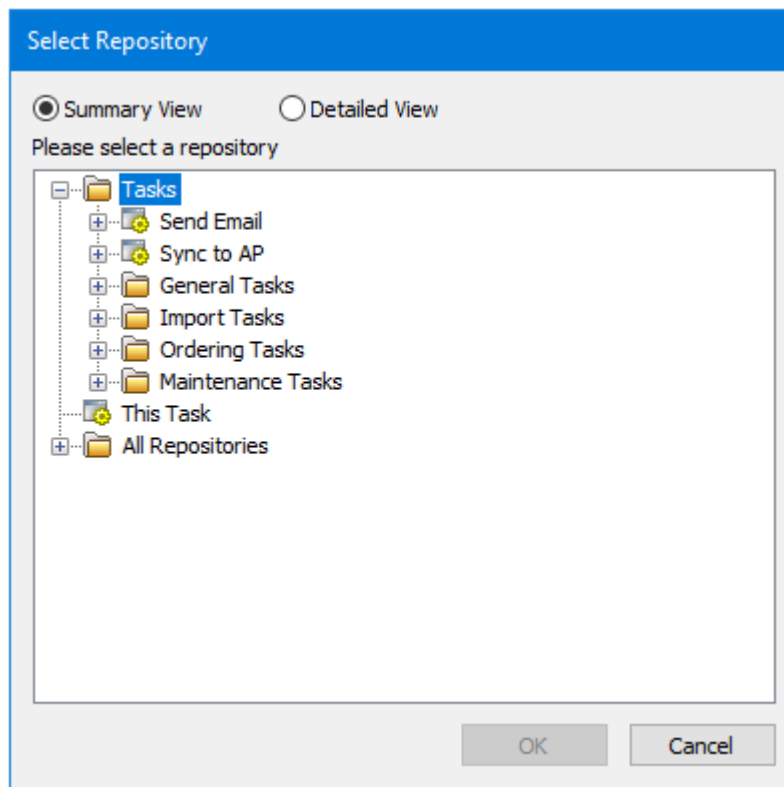
The Select Repository dialog is used to select a repository for either the Repository Filter in the **Filter Data** tool or for use in Internal Memory.

When a task containing an Output step which uses Memory is run, a Repository is automatically created for the task containing the memorised recordset. It contains no data until the task has been run, unless data is then manually entered, but may still be selected for use in the Repository Filter.

When a **Filter Data** step is added to the task, the Repository can be selected and a Repository Filter created within the step. The Repository Filter is used to compare the new recordset created when the task is run again against the existing memorised recordset in the Repository, and then set rules to filter the new subset of data produced.

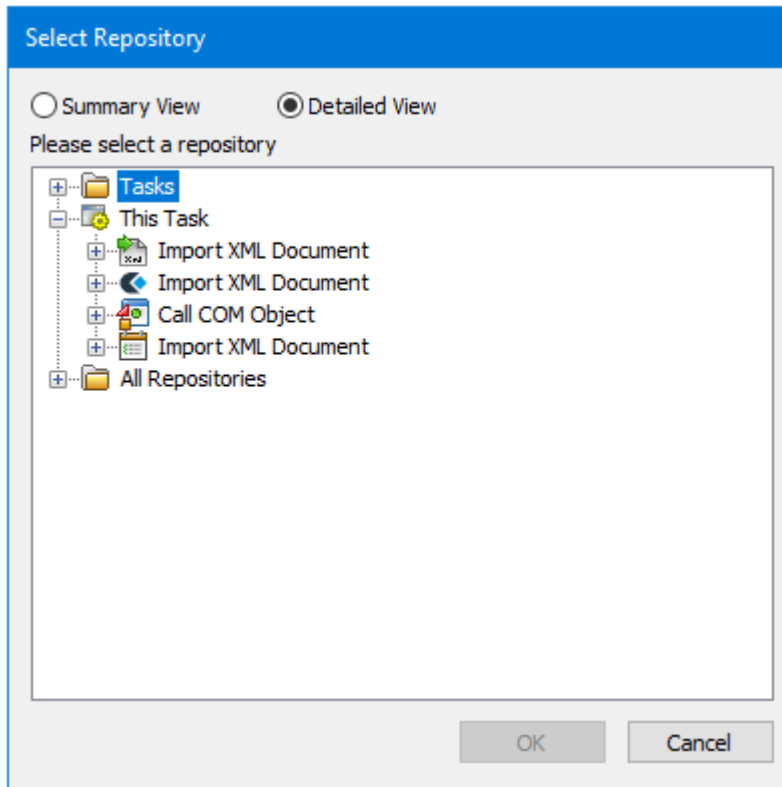
Summary View

The **Summary View** option lists the Repositories directly under each task in a hierarchical tree structure.





Detailed View


In addition to the above, the **Detailed View** option also allows you to see the task / step / Memory Provider / Repository in the hierarchy.




The tree behaves in a similar manner to a Windows explorer to displays or hide items nested below it as explained below.

 **Folder** — Contain tasks and sub-folders

 **This Task** — Contain steps. These are the Output steps through which a Repository has been created. Double-clicking on the task displays the items below it.

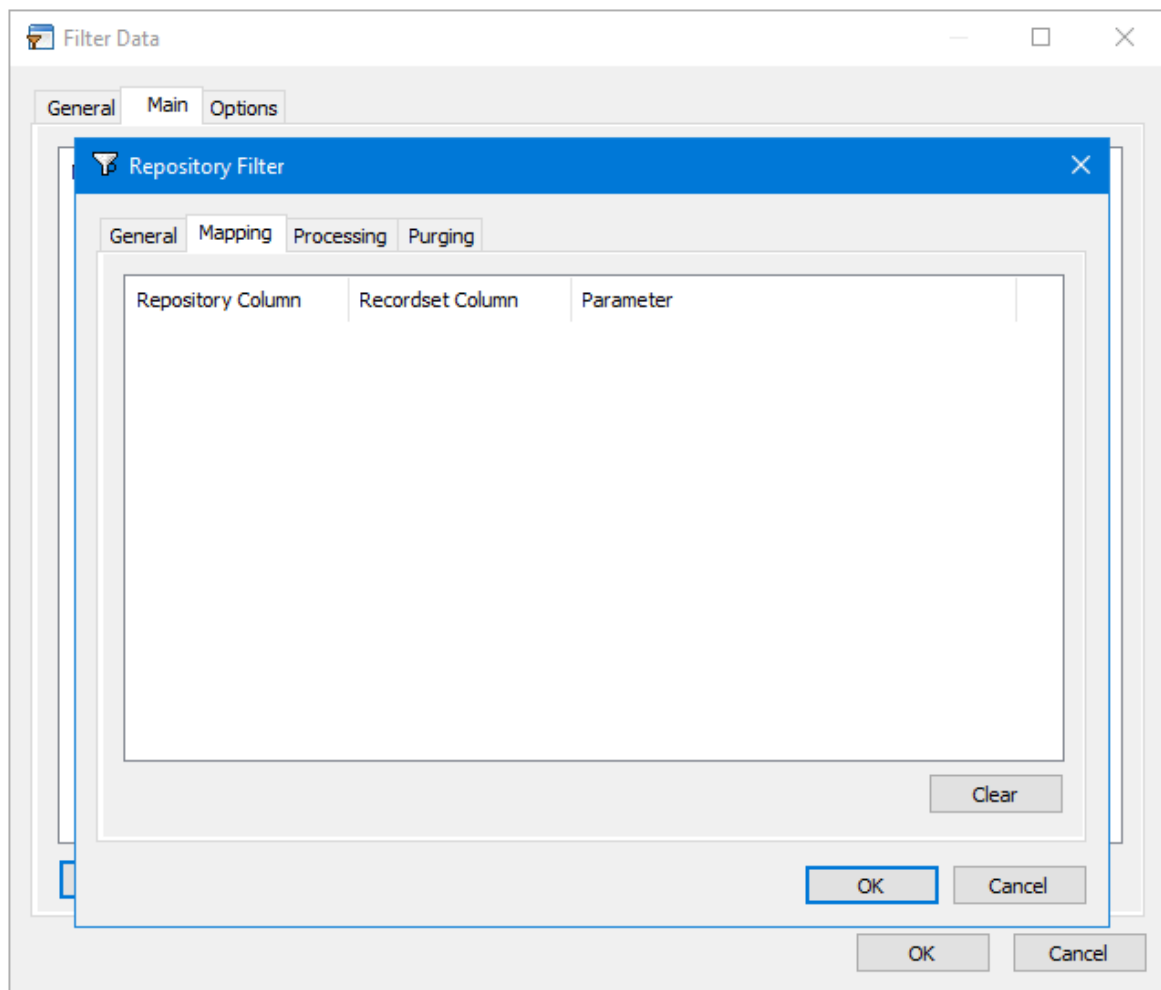
 **Step (Example)** — Contains Memories created for the step. Double-clicking on the step displays the items below it.

 **Memory** — Contains Repositories. Double-clicking on the Memory displays the Repositories below it.

 **Repository** — Displays the Repository name. Double clicking on the name then selects it for inclusion in the Repository Filter.

About the Mapping Tab

The Mapping tab is used to select the recordset columns used by the filter which are then mapped to the Repository columns.



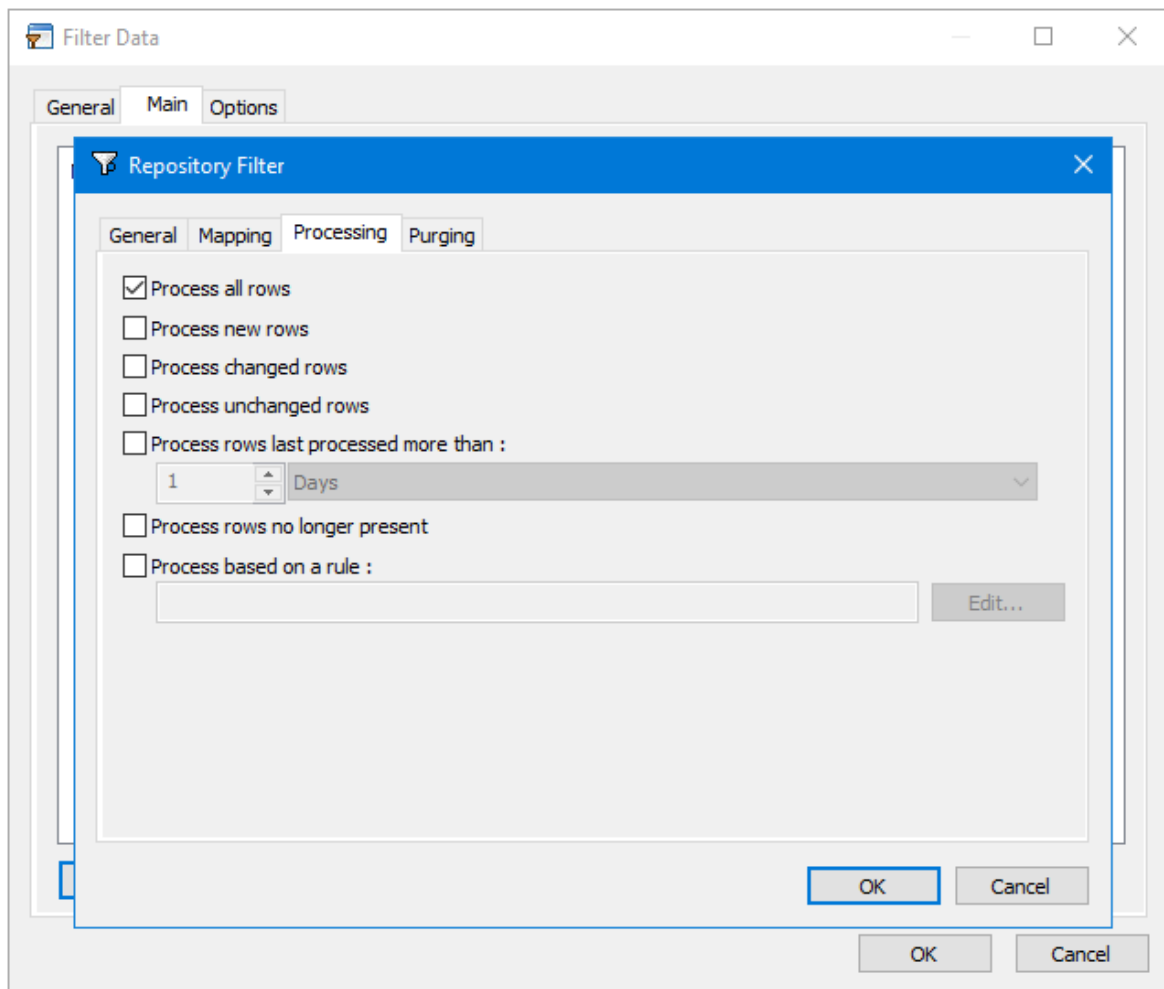
It contains a table which is automatically populated when a Repository is selected.

- ▶ **Repository Column** — Displays the Repository column name.
- ▶ **Recordset Column** — Displays the recordset column name which the Repository column is mapped to. This is initially selected automatically if the column names are the same or displays **None** if it is not. However clicking on the name displays a drop down list of all available columns in the recordset for selection. Each row for the mapped column value created at task run-time is then processed by the filter and stored in the Repository against the selected Repository column name.

NOTE: Repository "key" columns should not be mapped to non-unique columns to ensure that duplicate rows are not memorised.

- ▶ **Parameter** — Displays the parameter for the selected column. The parameter is resolved for each row in the record source and is used to look up existing entries in the Repository when the task is run. This is the mechanism used to detect new rows, if rows have changed, and so on.

About the Processing Tab



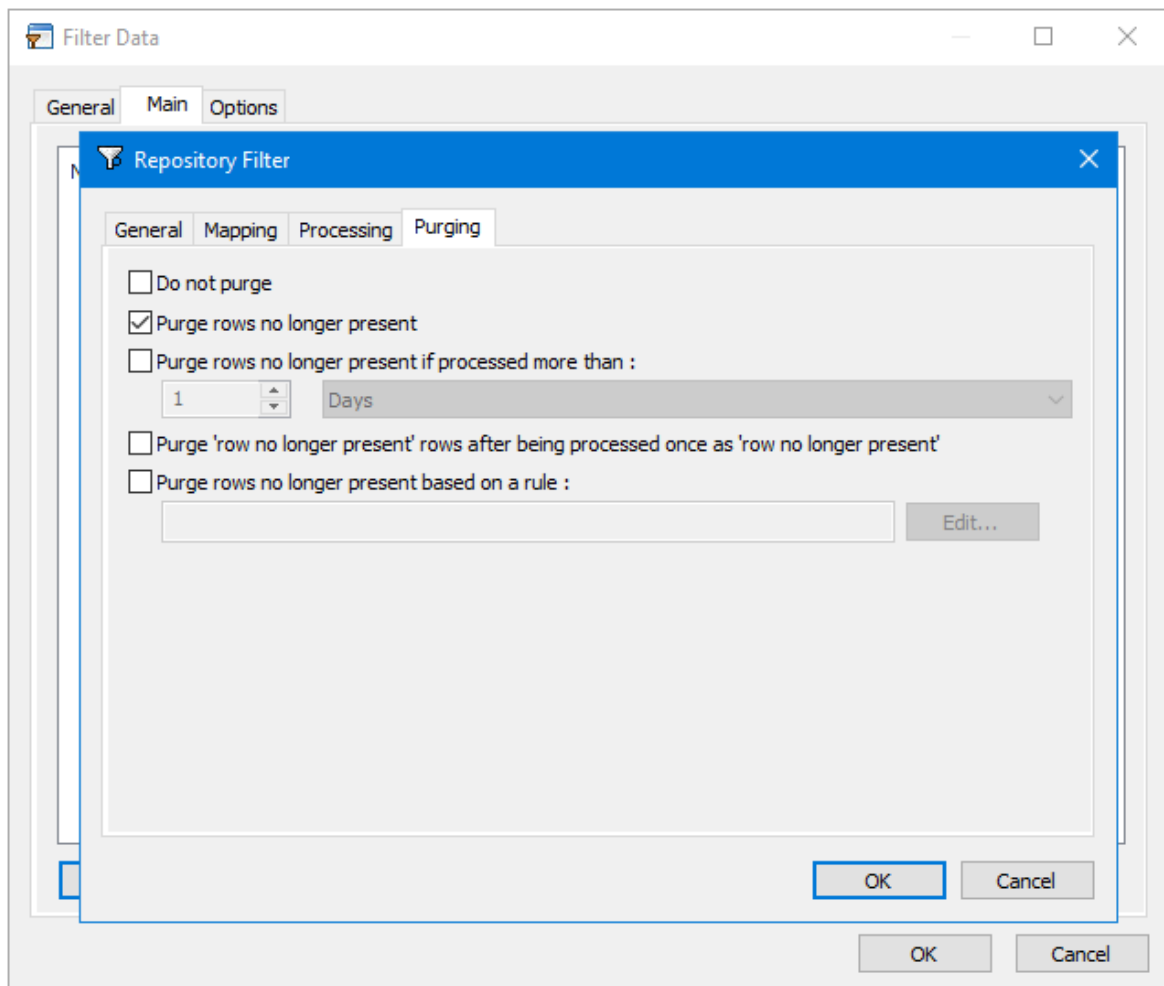
The Processing tab is used to configure which rows are processed by the filter through a number of selectable option check boxes as follows:

- ▶ **Process all rows** — All rows in the recordset are included in the subset of data provided by the step.
- ▶ **Process new rows** — Only new rows are included in the subset of data provided by the step.
- ▶ **Process changed rows** — Only changed rows are included in the subset of data provided by the step.
- ▶ **Process unchanged rows** — Only unchanged rows are included in the subset of data provided by the step.

- ▶ **Process rows last processed more than** — This option is used in conjunction with two fields below it which enable a time period to be selected. Any rows that have been previously processed more than the selected time period are included in the subset of data provided by the step.
- ▶ **Process rows no longer present** — Any rows that are no longer present are included in the subset of data provided by the step.
- ▶ **Process based on a rule** — All rows in the recordset are compared against those in the Repository. An **Edit** button is provided to open the a Rule Script dialog box to enable a rule to be manually scripted. The actual rule logical expression itself is then displayed in the field provided and is applied when the recordset is processed.

About the Purging Tab

The Purging tab is used to manage the number of rows contained in the Repository and is carried out after all processing has been completed.



This is achieved through a number of selectable option check boxes as follows:

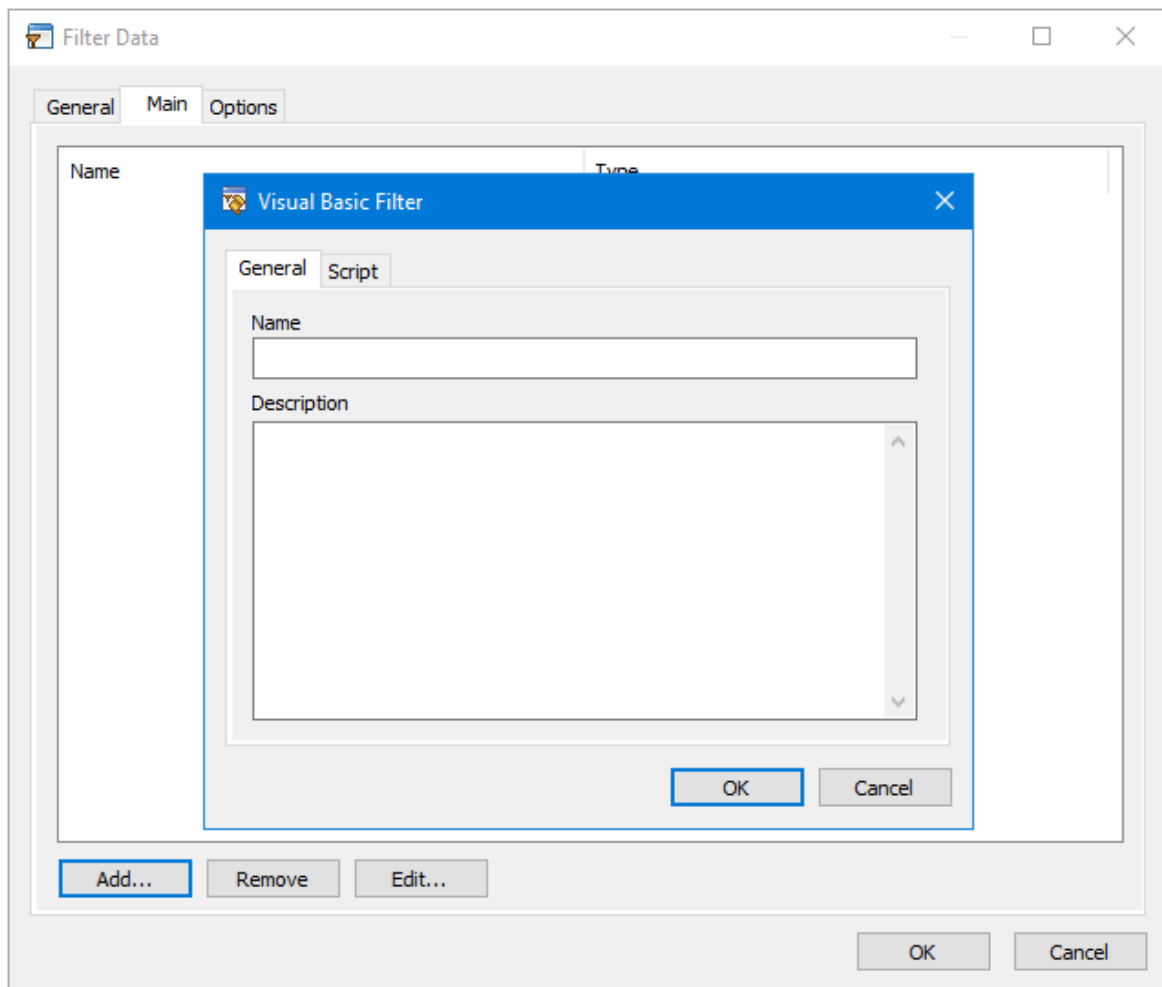
- ▶ **Do not purge** — No rows in the Repository are deleted.
- ▶ **Purge rows no longer present** — All rows in the recordset are compared against those in the Repository. Rows that appear in the Repository but not in the recordset are removed from the Repository.
- ▶ **Purge rows no longer present if processed more than** — All rows in the recordset are compared against those in the Repository. Rows that are no longer present which were processed more than the selected time period ago are removed from the Repository. For example, if a time period of **1 Minute** is selected then rows that are no longer present that were last processed more than 1 minute previously are removed.
- ▶ **Purge row no longer present rows after being processed as 'row no longer present'** — All rows in the recordset are compared against those in the Repository. Rows that appear in the Repository for a second time with a status of **row no longer present** are removed from the Repository.
- ▶ **Purge rows no longer present based on a rule** — All rows in the recordset are compared against those in the Repository. Any matching the selected rule are removed.

About the Visual Basic Filter

By adding a **Filter Data** step to a task, a Visual Basic Filter may be created that applies a filter expression against the selected data source to filter the rows.

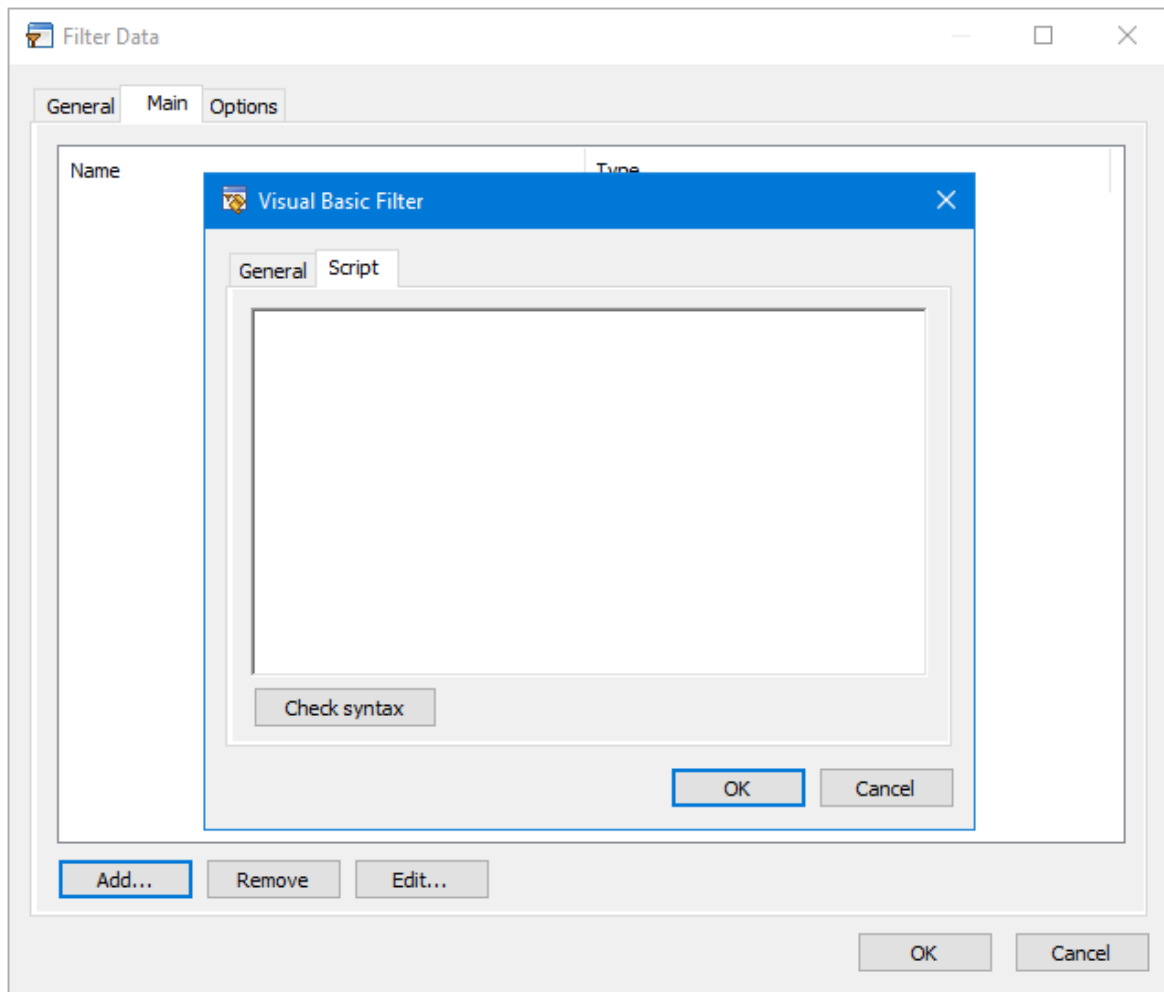
The script specified in the filter is executed once for each row in the record source. If the expression returns "true", the row is transferred to the output recordset.

About the General Tab



The General tab is used to enter the following details for a Visual Basic Filter:

- ▶ **Name** — Enter a meaningful name for this filter
- ▶ **Description** — If required, enter a description of the filter

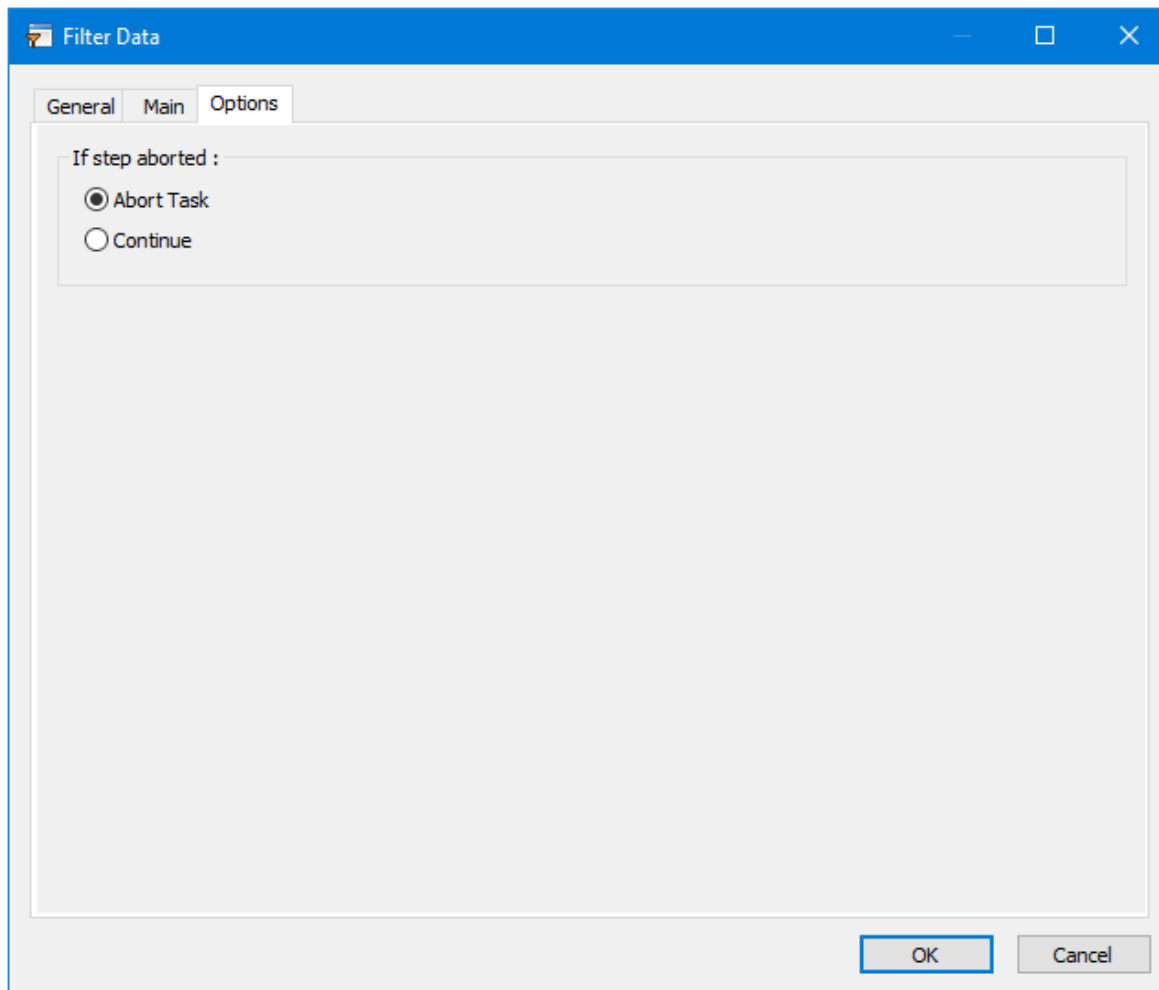


The Script tab is used to enter the Script for the filter through the following:

- ▶ **Script** — Variables, properties, columns, formulas, functions and scripting elements may be dragged and dropped into the created script from the Task Browser dialog
- ▶ **Check Syntax** — Checks the script syntax and displays the result in a pop-up dialog

About the Options Tab

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



If the step is aborted, you can choose to **Continue** processing onto the next step in the task, or terminate the whole task immediately (**Abort Task**).

Want to learn more?

Discover how Codeless Platforms can help your business by improving performance, boosting efficiency and cutting costs.



+44 (0) 330 99 88 700



enquiries@codelessplatforms.com



www.codelessplatforms.com

