BPA
Platform

Magento Integration Tool v1.0.20

codeless PLATFORMS

Copyright

Head Office:

Codeless Platforms
Suite 2 Bourne Gate
25 Bourne Valley Road
Poole
BH12 1DY
United Kingdom
Tel: +44 (0) 330 99 88 700
Email: enquiries@codelessplatforms.com

Trademarks

Contents

Introduction

The **Magento Integration** tool allows communication between BPA Platform and Magento. All communication uses XML. You use the tool to map BPA Platform data to Magento objects and operations. Operations such as, `ADD`, `UPDATE`, `GET`, and `DELETE`, are supported for a variety of business objects, such as, Product, Customer and Order — for a detailed list, see **SUPPORTED OBJECTS ON PAGE 13**.

The XML responses received back from the Magento API can be saved to file or passed to another BPA Platform tool for further processing. The responses can include details of the object requested, any errors from the provider, or the values of any automatically generated ID fields. For example, you can use **Magento Integration** to provide an indirect link between systems that do not typically synchronise data with each other, such as synchronising contacts from a CRM platform as Magento customers, importing Magento sales orders into an ERP application, or even to provide reports containing data from both systems.

The Magento Integration Tool Pack

The tool pack consists of:

- **Magento Integration** — The tool communicates with the Magento API. It must be installed on the BPA Platform server, and on any BPA Platform client machines running **Magento Integration** tasks. It is this that provides the translation between BPA Platform data and Magento object and operation.

> **NOTE:** This tool pack has been developed by Orbis Software Benelux BV and has been certified for use with BPA Platform by Orbis Software Ltd T/A Codeless Platforms.

System Requirements

The **Magento Integration** tool requires BPA Platform (formerly "TaskCentre") 4.6 or above.

This tool is compatible with Magento versions 2.x.

Architecture

The diagram below provides a high-level system architecture overview of the **Magento Integration** tool pack, with BPA Platform and the Magento REST API.
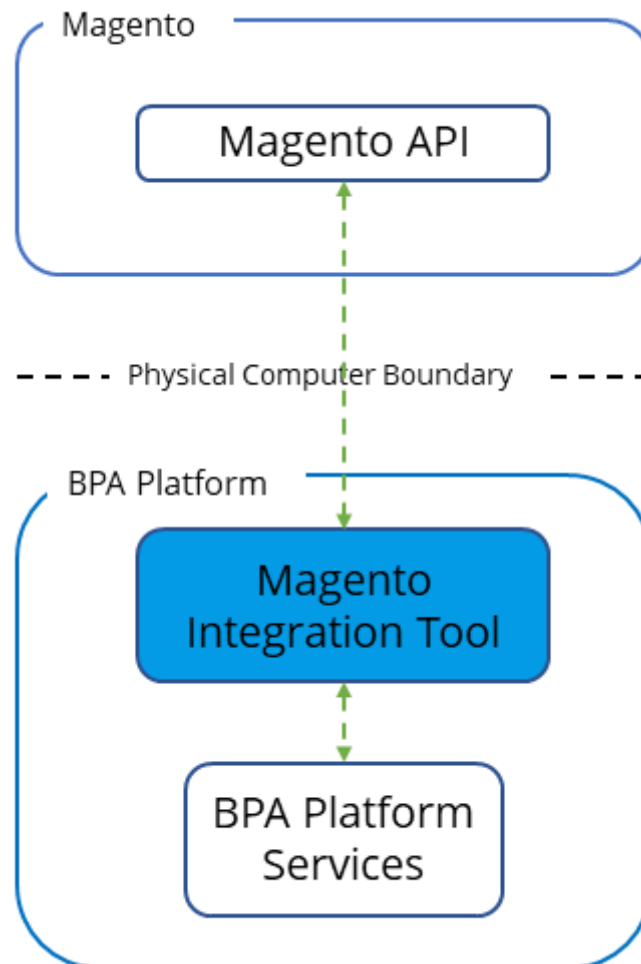

Figure 1 — High-level system architecture of the Magento Integration tool

About the Magento Integration Tool

Global Configuration

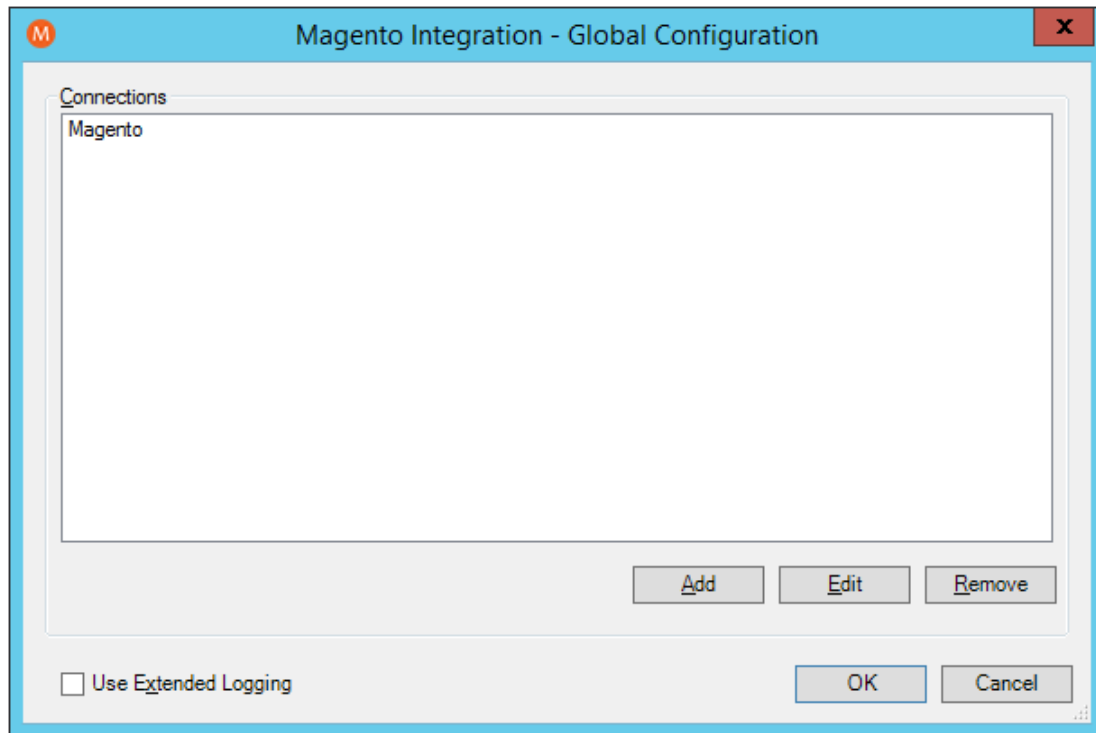The global configuration for this tool is used to create connections to Magento.

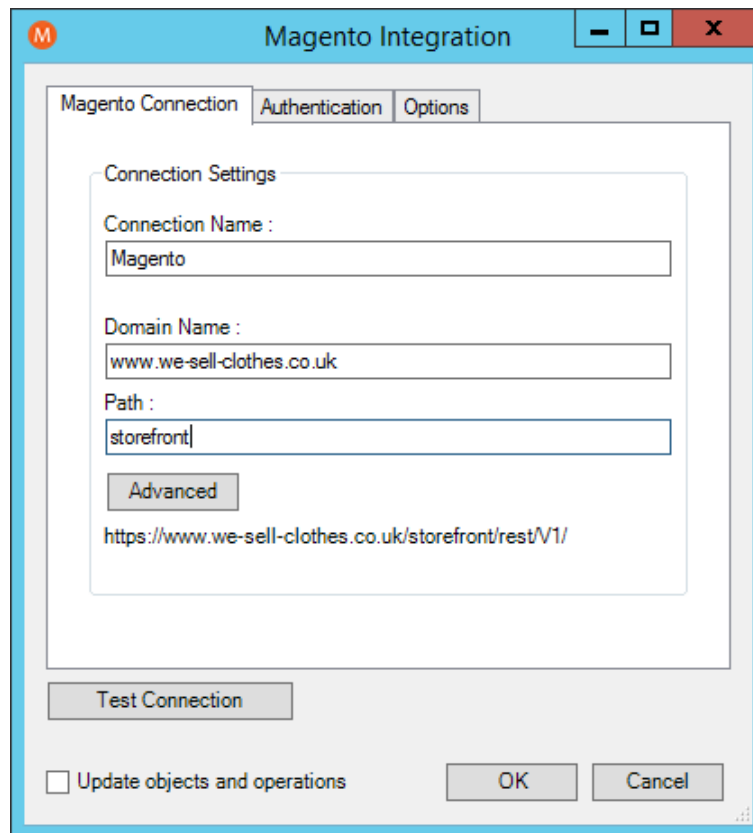*Figure 2 — Magento Integration - Global Configuration window*

Click **Add** to create a connection to Magento.

Use Extended Logging

Enabling this option exposes the XML parsed between the **Magento Integration** tool and the Magento API. You can view the log from the BPA Platform Event Log (*Manage > Event Log*).

Magento Connection Tab

You configure the connection to Magento here.

Figure 3 — Magento Integration – Magento Connection Tab

Configure a meaningful **Connection Name** for this connection; this is the name displayed when adding tasks for Magento.

Within **Domain Name**, specify the domain name of the URL to Magento. For example, if the Magento webshop is running on https://www.we-sell-clothes.co.uk/storefront, the corresponding domain name is `www.we-sell-clothes.co.uk`.

Within **Path**, specify the path of the URL to Magento. With the previous example, this is `storefront`.

By default, the connection to Magento is over HTTPS, on TCP port 443. Use the **Advanced** button to either **Set a Custom Port Number**, to not **Use HTTPS** to communicate to the server or specify a different **Connection Timeout time** in seconds.
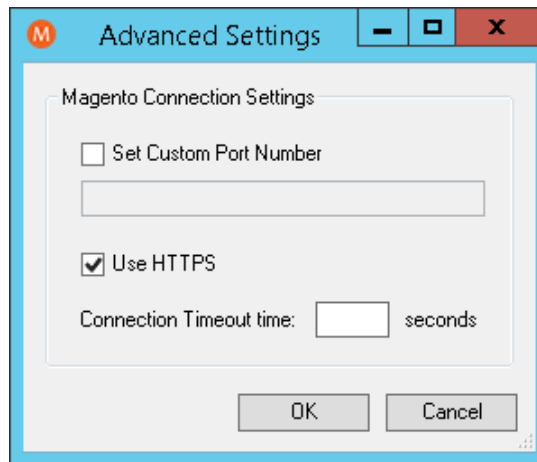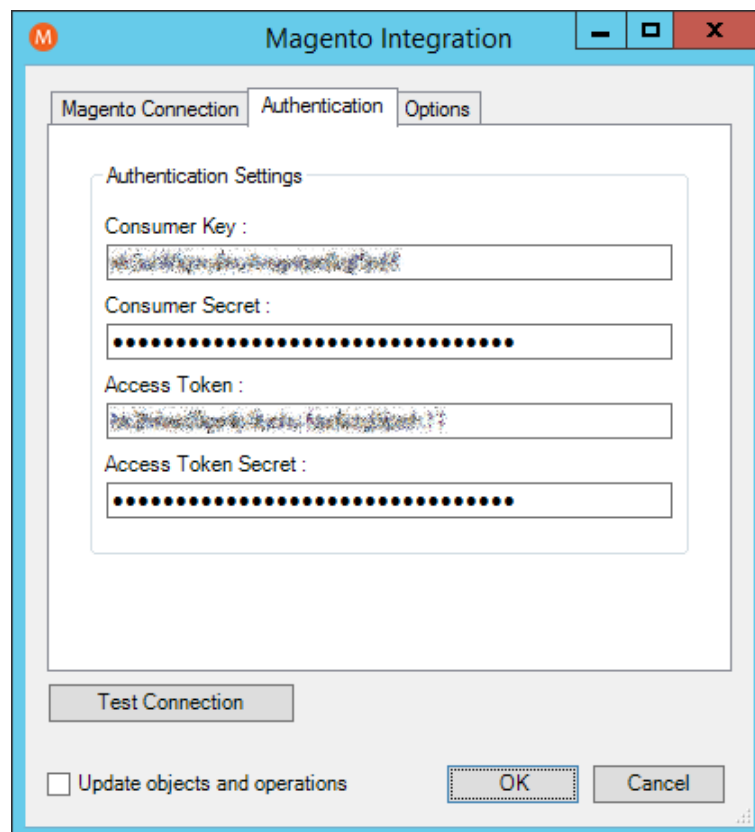

Figure 4 — Advanced Settings window

Authentication Tab

You can configure the Magento credentials here that allow access to Magento.

Specify the **Consumer Key**, **Consumer Secret**, **Access Token** and **Access Token Secret** created in Magento.

Figure 6 — Magento Integration - Options tab

If required, you can limit the objects available for mapping to BPA Platform data — choose a **Module** to show only objects from that module, see **MAPPING TAB ON PAGE 9**.

The `Product` and `Category` objects have an optional property, `storeview_code`, that can be mapped in the Mapping tab. This property determines which Magento store view is used for the operation. When the property isn't mapped, the value specified under **Default Storeview Code** is used.

The list of user-defined **Multiselect attributes** is refreshed when using clicking **Test Connection** or **OK**. Attributes of type `multiselect` are treated differently than other attributes. When new `multiselect` attributes are added in Magento, you must refresh this list here to ensure proper conversion of these attribute values (click **Test Connection**).

Using the **Test Connection** button allows you to test the connection to the Magento API.

Update Objects and Operations

If you have installed an updated version of the **Magento Integration** tool which comes with new objects and operations, you must refresh the connection's schema for these updates to be made available within an existing connection.

Enabling the **Update objects and operations** option, allows the **Magento Integration** schema within BPA Platform to be refreshed.
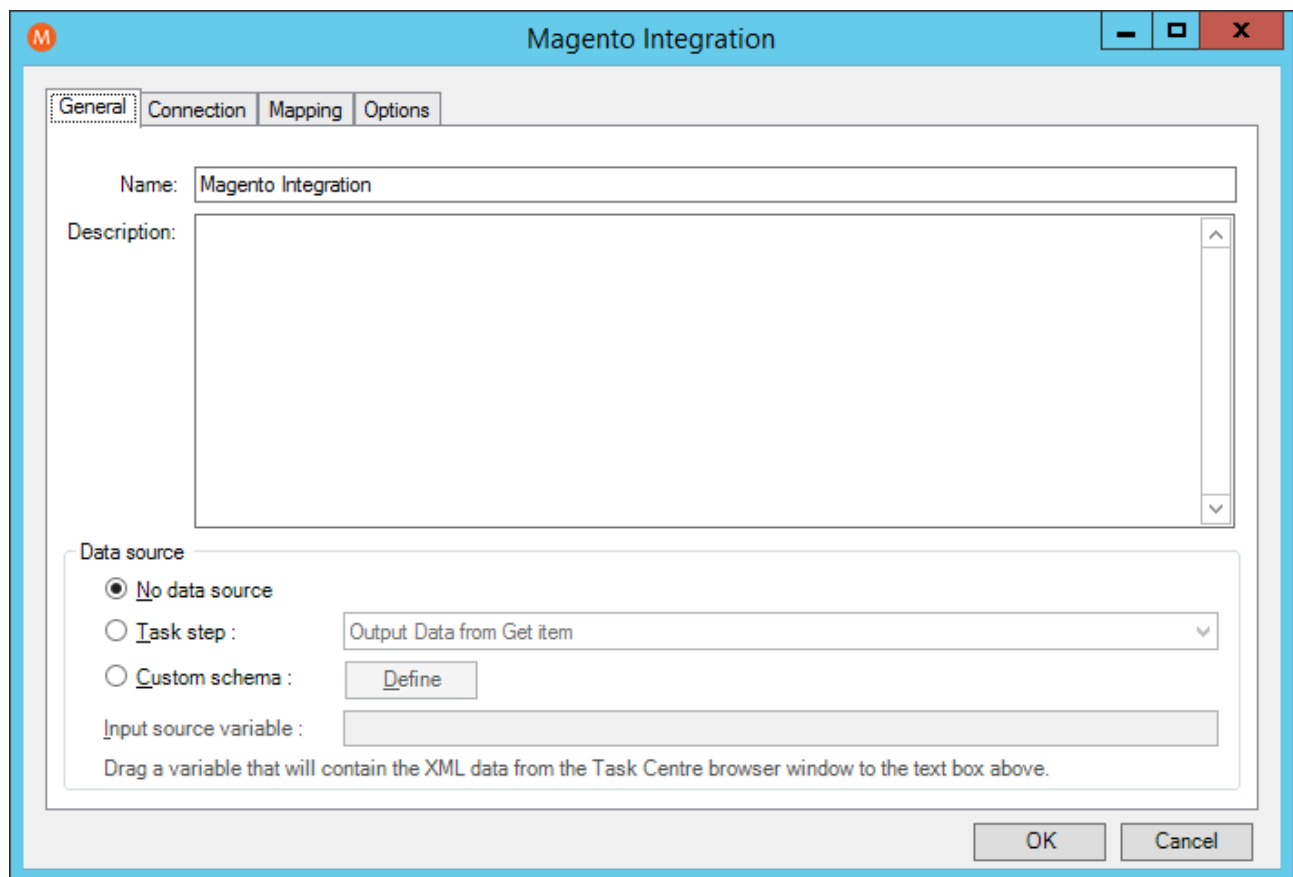
> **NOTE**: Existing task steps using the Magento connection must be refreshed in order to use the latest schema.

Step Configuration

When creating new tasks, the **Magento Integration** tool is located under **Data Connectors** of the Task Browser.

> **NOTE:** It is not possible to add the connector tool to the task until at least one connection to Magento has been defined in the global configuration

General Tab

Figure 7 — Magento Integration - General tab

Provide a meaningful **Name** and **Description** for this step.

Choose your **Data source**. This can either be:

- **No data source** — If you don't make use of a dedicated XML input source, select this option to use BPA Platform variables in place of the XML objects' fields. These can then be mapped to operational fields — see **MAPPING TAB ON PAGE 9**.

- **Task step** — The data source can be set to an available BPA Platform XML data source. Only those steps that are capable of natively exposing an XML document at runtime are listed. These may be another **Magento Integration** step, or a tool, such as **Convert Recordset to XML**.

- **Custom schema** — An XML schema defines the structure of the parsed XML: what tags are present, and the nesting of the tags. You **Define** the schema of the XML that will be used as the input data source for this step. The industry standard XSD format is used by the **Magento Integration** tool. Any XML processed by this step must conform to this schema else an error will be reported.
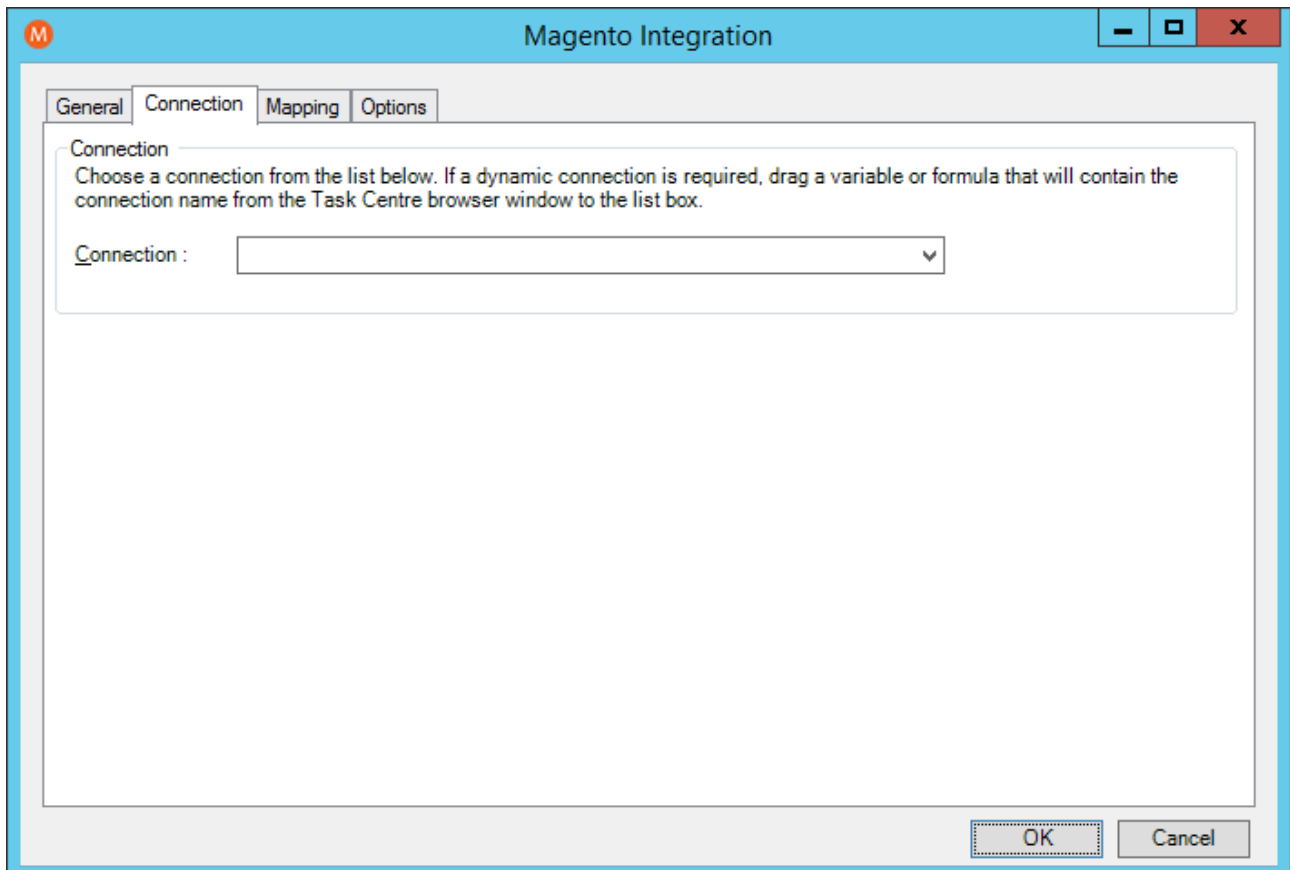
  If the XSD schema is available, either import it into the **Custom Schema Configuration** (use **the Import XSD/XML File** button) or copy and paste it into the configuration box.

  If the XSD schema is not available, you can import an example of the runtime XML (**Import XSD/XML File**) or copy and paste it into the configuration box. Use the **Parse** button to create the schema.

  o   **Input source variable** — As well as defining the schema, you must specify the BPA Platform variable that contains the XML data at runtime.

Connection Tab

You specify the Magento **Connection** this step must use.

Figure 8 — Magento Integration - Connection tab

All connections created in **GLOBAL CONFIGURATION ON PAGE 3** are presented here.

Alternatively, you can use a BPA Platform variable to create a dynamic connection, where the connection used is determined by runtime circumstances. At runtime, the contents of the variable must match the name of one of the GLOBAL CONFIGURATION connections — this is case-sensitive.

Mapping Tab

Here you define links between the incoming XML and the outgoing data that is sent to the Magento API. This defines how, at runtime, the incoming XML is to be translated into the XML required for the relevant object and operation.
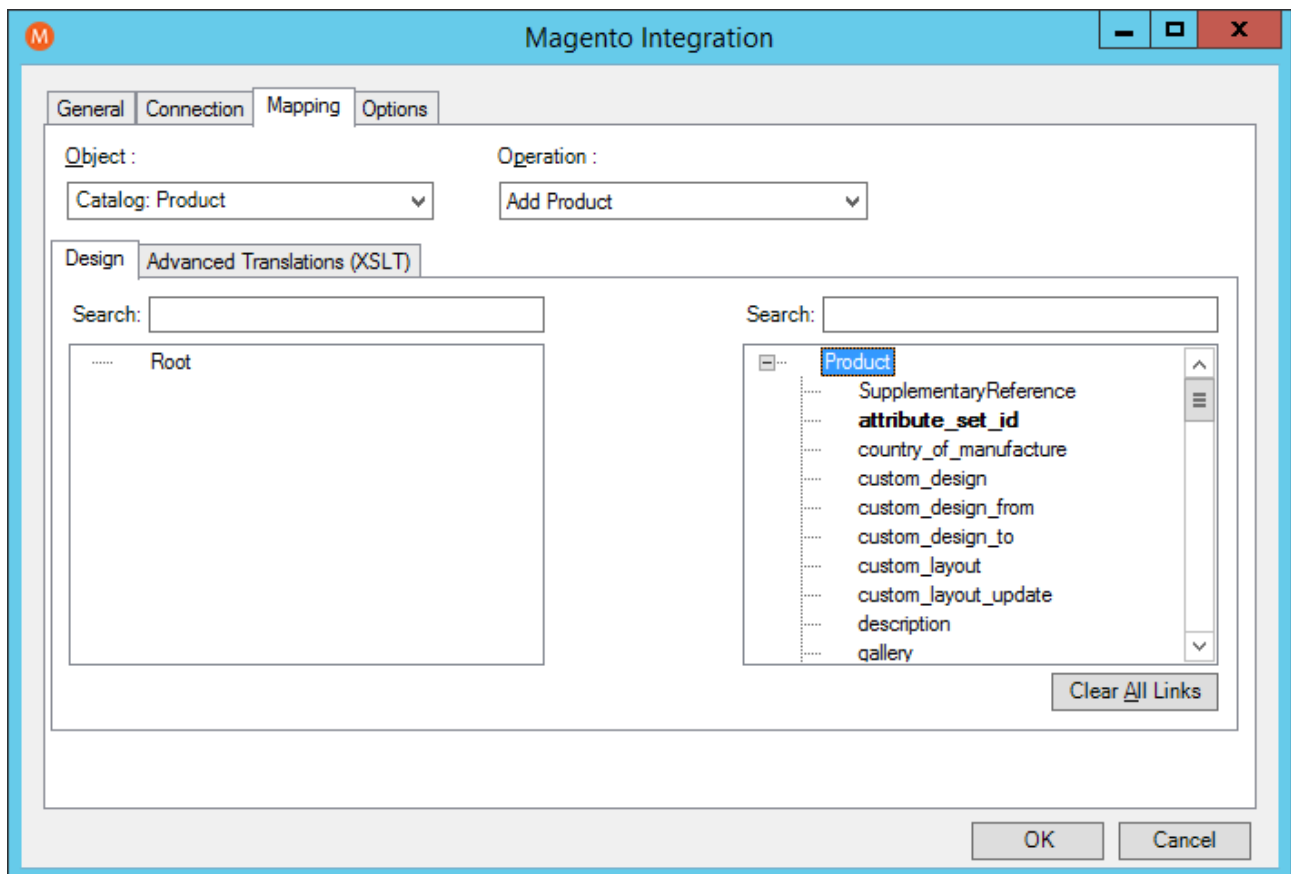
Figure 9 — Magento Integration - Mapping tab

The **Object** drop-down shows the available objects. The **Operation** drop-down shows the operations available for the selected **Object**. For more information about which operation is available for an object, see **SUPPORTED OBJECTS ON PAGE 13**.

You create the links by clicking and dragging an object field onto its corresponding operation field. Note that only linked fields are used in the output XML. BPA Platform formulas and variables can be included in the source data — drag them from the Task Browser to the object's XML tree control. These can then be linked to operation fields. Note that BPA Platform recordsets must first be converted to XML using either the **Convert Recordset to XML** or **Transform Data** tool before they can be used here.

Each operation has an additional field, `SupplementaryReference`, which allows for traceability when transferring data from one place to another. When mapped, the data resides locally at runtime. It is added to the output and creates a record for reference purposes only — you can choose to map any field to `SupplementaryReference` to assist with checking where the data originated from or at what time the data transfer occurred, for example.

The **Magento Integration** tool uses eXtensible Stylesheet Language Transformations (XSLT) to translate the received XML. The **Advanced Translations (XSLT)** tab shows the XSLT generated for the links created for the object and operation. Use **Enable Free Type Mode** to directly edit the XSLT — this is particularly useful when translating a nonstandard requirement.

Options Tab

The **Options** tab allows you to define how errors in this step are handled at task runtime.
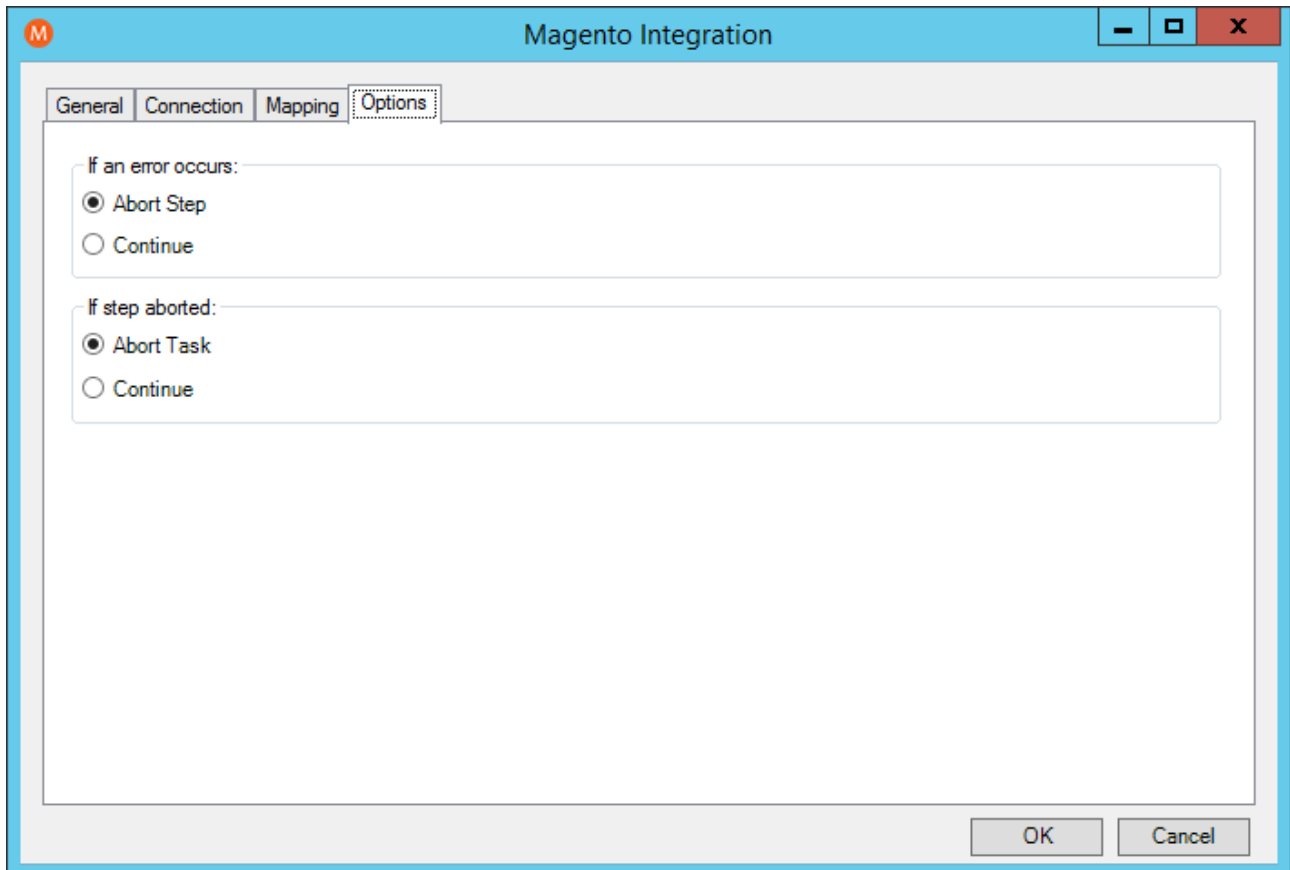
Figure 10 — Magento Integration - Options tab

If an error occurs, you can decide whether the step should **Continue** processing, or terminate the step immediately (**Abort Step**).

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**).

Allowing the task to continue allows you to use the error XML received back from **Magento Integration** in a **Save File** step for investigation purposes, for example.

Working with Other Steps

Magento Integration can directly interact with the following steps:

Consuming XML from Other Tools

The **Magento Integration** tool can consume and process XML outputted by the following tools:

Step Icon	Name	Step Type
	Import Flat File	Input
	Import XML Document	Input
	Convert Recordset to XML	Format
	Transform Data	Format
	Call Task Tool	Execute
	Magento Integration	Data Connectors

Exposing XML to Other Tools

The following tools can consume and process the XML outputted by the **Magento Integration** tool:

Step Icon	Name	Step Type
	Convert XML to Recordset	Format
	Run Microsoft Reporting Services	Format
	Transform Data	Format
	Save File	Output
	Call Task Tool	Execute
	Web Service Connector Tool	Data Connectors
	Magento Integration	Data Connectors

The **Magento Integration** tool outputs two documents:
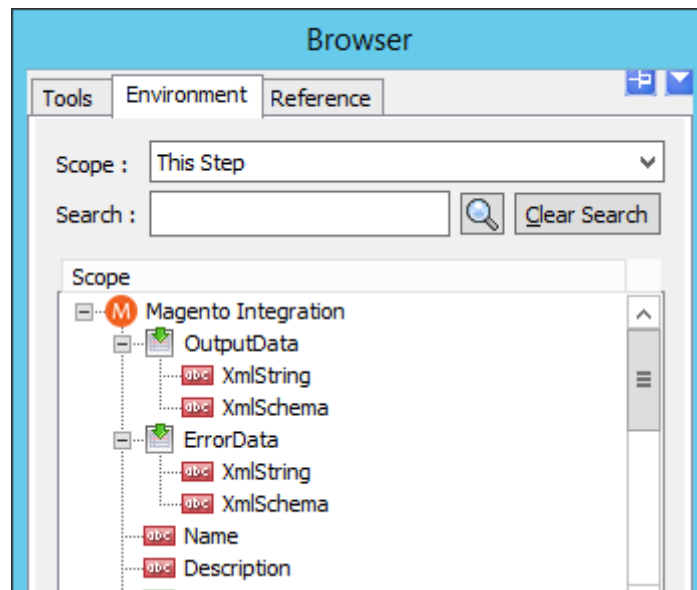
Figure 11 — Magento Integration tool outputs

- `OutputData` XML document contains:

    o   Data returned from **Magento Integration**, for instance the created, updated or retrieved entity instances

    o   `SupplementaryReference`

- `ErrorData` XML document is structured as follows:

    `<Error>` — All errors are created as an `<Error>` node, with the following sub-nodes:

    `<Object>` — The name of the requested object, such as, `Product`

    `<CODE>` — The error code returned by Magento

    `<MESSAGE>` — The corresponding error message

    `<EXTENDEDINFO>` — A string containing additional information about the error

    `<INPUTDATA>` — The input data mapped for the object, plus all data contained in the `SupplementaryReference` field

Both outputs can be used:

- Directly by subsequent tool steps which consume XML data inputs

    o   If required, use the **Convert XML to Recordset** tool to convert the XML to a BPA Platform recordset

- Accessed using the output or error objects for response data (`XmlString`), or the output or error schema (`XmlSchema`) present in the Task Browser

Supported Objects

At the time of writing, the following objects and operations are supported:

Magento Object	Standard Operations						Comment
	Get	**Add**	**Update**	**Delete**	**Email**	**Other**	
Catalog: Category	✓	✓	✓	✓		✓	Additional operation: • Move
Catalog: Category Attribute	✓						
Catalog: Category Link	✓	✓		✓			
Catalog: Product	✓	✓	✓	✓			
Catalog: Product Attribute	✓	✓	✓				
Catalog: Product Attribute Option	✓	✓		✓			
Catalog: Product Link	✓	✓	✓	✓			Operations: • Get Product Links • Add or Update Product Link • Delete Product Link
Catalog: Product Media Gallery Entry	✓	✓	✓	✓			Operations: • Get Product Media Gallery Entries • Get Product Media Gallery Entry by ID • Add Product Media Gallery Entry • Add or Update Product Media Gallery Entry • Update Product Media Gallery Entry • Delete Product Media Gallery Entry
Catalog: Product Stock Item	✓						Operation: • Get Product Stock Item by ID
Catalog: Product Website Link		✓		✓			
Customer: Customer	✓	✓	✓	✓			

Magento Object	Standard Operations						Comment
	Get	Add	Update	Delete	Email	Other	
Customer: Customer Address	✓	✓	✓	✓			
Customer: Customer Group	✓	✓	✓	✓			
Directory: Country	✓						Operation: • Get Countries and Regions
Directory: Currency	✓						
Inventory: Source Item	✓						Additional operations: • Assign Products to Source • Unassign Products from Source
Sales: Credit Memo	✓				✓		Operations: • Get Credit Memo • Get Credit Memo by ID • Email Credit Memo
Sales: Credit Memo Comment	✓	✓					Operations: • Get Credit Memo Comment • Add Credit Memo Comment
Sales: Invoice	✓				✓		
Sales: Invoice Order		✓					
Sales: Order	✓				✓	✓	Additional operations: • Add comment • Cancel order • Hold order • Release order
Sales: Order Address	✓						
Sales: Order Comment	✓	✓					
Sales: Order Item	✓	✓		✓			

Magento Object	Standard Operations						Comment
	Get	Add	Update	Delete	Email	Other	
Sales: Refund Invoice		✓					Additional operation: • Create Refund for Invoice Use this service if the invoice was created using an online payment method.
Sales: Refund Order		✓					Operation: • Create Refund for Order Use this service if the invoice was created using an offline payment method.
Sales: Shipment	✓				✓		
Sales: Ship Order		✓					

Runtime

The incoming XML is translated into the XML format for the object and operation selected in the configuration. The data for the linked fields is brought across into the output XML — only those fields that were linked are brought across. The XML is passed to the integration tool, which then:

- Processes the data

- Performs the operation requested

- Sends back an XML document containing the response

The XML can then be used by other BPA Platform steps

Error Handling

Errors are written to the BPA Platform Event Log (*Manage > Event Log*). You define how errors are handled in the **Options** tab of the **Magento Integration** tool — see **OPTIONS TAB ON PAGE 10**.

Reasons for the errors could include:

- Web service connection errors

- User privilege errors

- Errors from the Magento API

- Warnings and messages from the Magento API

Magento Account Security

The Magento API is the only method used to connect to Magento and perform read / write tasks — all security present in the API is used.

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

codeless PLATFORMS