

Mapping Mode

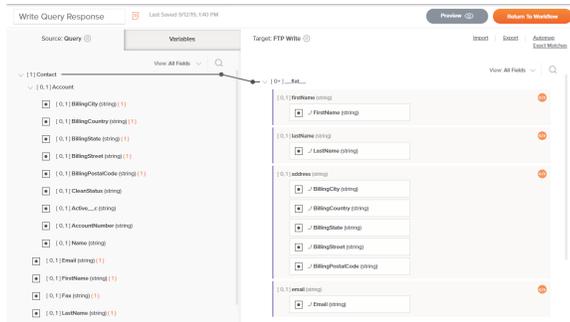
Introduction

Mapping mode provides an overview of a transformation mapping and the basic tools to perform the mapping. Also see [Common Mode Elements](#) for elements of the user interface that are shared by multiple display modes.

Accessing Mapping Mode

Mapping mode is the default display mode when opening a transformation for configuration.

In mapping mode, the left side contains the source structure and variables available to be used within the mapping of the target structure on the right side:



To return to mapping mode while in [script mode](#), click the arrow on the left side of the screen:

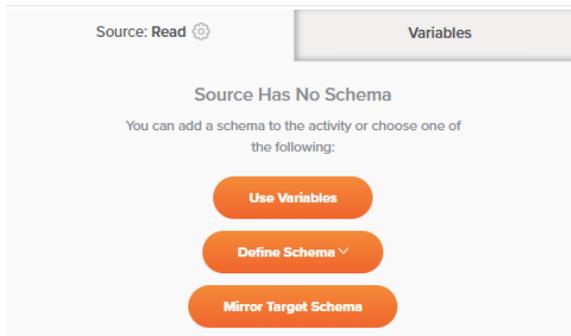


Source and Target Schemas

In mapping mode, the left side of the transformation displays the source schema, while the right side displays the target schema. These schemas can be defined either [within an activity](#) that is located adjacent to the transformation, or directly [within a transformation](#). If a schema has not yet been defined, you will be presented with the options below for the source or target side, respectively.

NOTE: A target schema must be specified prior to being able to configure a transformation mapping.

Source Has No Schema



It is not required to specify a source schema (see [When to Use a Schema](#)). If you don't want a source schema to be used, you don't need to do anything with the source side of the transformation.

If you have not specified a source schema and want to specify one, you can [return to the workflow](#) to provide a schema [within an activity](#) located adjacent to the transformation, or select one of these options:

- **Use Variables:** This toggles to the **Variables** tab, where you can add variables as described in [Mapping Variables](#).

On This Page

- [Introduction](#)
- [Accessing Mapping Mode](#)
- [Source and Target Schemas](#)
 - [Source Has No Schema](#)
 - [Target Has No Schema](#)
- [Target Nodes](#)
- [Target Fields](#)

Related Articles

- [Automapping](#)
- [Common Mode Elements](#)
- [Conditional Mapping](#)
- [Custom Flat Schema](#)
- [Custom Hierarchical Schema](#)
- [Mapping Custom Values](#)
- [Mapping Source Objects](#)
- [Mapping Variables](#)
- [Mapping with Scripts](#)
- [Mirrored Schemas](#)
- [Nodes and Fields](#)
- [Sample File Schema](#)
- [Script Mode](#)
- [Transformation Mapping Validity](#)

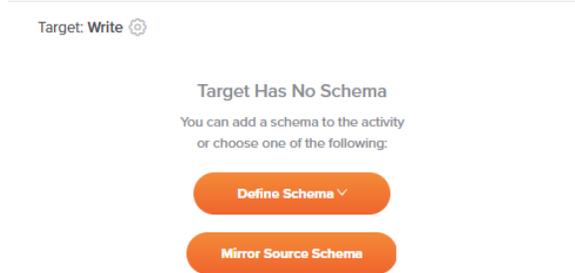
Related Topics

- [Cloud Studio](#)
- [Schemas](#)
- [Schemas Defined in a Transformation](#)
- [Schemas Defined in an Activity](#)
- [Scripts](#)
- [Transformation Basics](#)
- [Transformation Display Modes](#)
- [Transformations](#)
- [Variables](#)

Last updated: Sep 18, 2019

- **Define Schema:** This provides an additional menu of options including **Create Flat**, **Create Hierarchical**, and **Use Sample File**. When an option is selected, this opens an interface for creating a **flat** or **hierarchical** file schema, or providing a **sample file** as described under **Schemas**.
- **Mirror Target Schema:** **Copies the target schema** as described under **Schemas**. This option is present only if (1) there is a defined target schema *and* (2) the source of the transformation is *not* an adjacent server-based activity whose schema is generated directly from the endpoint, such as is the case with **SOAP**, **database**, and **Salesforce** endpoints, as well as some **custom endpoints**.

Target Has No Schema



A target schema must be specified prior to being able to configure a transformation mapping (see [When to Use a Schema](#)).

If you have not specified a target schema, you can [return to the workflow](#) to provide a schema *within an activity* located adjacent to the transformation, or select one of these options:

- **Define Schema:** This provides an additional menu of options including **Create Flat**, **Create Hierarchical**, and **Use Sample File**. When an option is selected, this opens an interface for creating a **flat** or **hierarchical** file schema, or providing a **sample file** as described under **Schemas**.
- **Mirror Source Schema:** **Copies the source schema** as described under **Schemas**. This option is present only if (1) there is a defined source schema *and* (2) the target of the transformation is *not* an adjacent server-based activity whose schema is generated directly from the endpoint, such as is the case with **SOAP**, **database**, and **Salesforce** endpoints, as well as some **custom endpoints**.

Target Nodes

A defined target schema is made up of nodes that can contain fields, as well as other, child nodes that can contain fields. Source data cannot be mapped to the target nodes themselves, but can be mapped only to the fields they contain. Once you have defined a target schema, to access options available on each node, hover over the target node, then click the actions menu icon  to show the actions menu. Each action is described below.



- **Clear mappings beneath this node:** This will delete all mappings on fields contained within the node, as well as all mappings on fields contained within child nodes of that node. Upon selecting this option, a message will ask you to confirm that you want to delete mappings. Note that using this option on a root node will remove all mappings from the transformation.
- **Clear invalid mappings only:** This will delete all invalid mappings on fields contained within the node, as well as all invalid mappings on fields contained within child nodes of that node. For more information about invalid mappings, see [Transformation Mapping Validity](#).
- **Expand all nodes beneath this node:** This will expand all child nodes, making it easier to work with structures that have many nodes. By default, nodes are expanded up to 6 levels deep.
- **Duplicate mapping node:** Available only on nodes whose **cardinality** indicates they can occur more than once (with the exception of the root node), this will duplicate the node and its fields, as well as any child nodes and their fields. This option is typically used for the purposes of creating additional nodes to map a denormalized source to a hierarchical target.
- **Remove duplicated mapping node:** Available only on nodes that were created through the **Duplicate mapping node** functionality, this will remove the duplicated node and its fields, as well as any duplicated child nodes and their fields.

- **Add condition to node:** Available only on nodes that do not already have a condition applied, this is used to conditionally apply a mapping to the fields contained within a node. Upon selecting this option, you will be brought into script mode to create the condition. For details about creating the condition, see [Conditional Mapping](#).
- **Edit condition:** Available only on nodes that already have a condition applied, this opens script mode for editing the existing condition in script mode. For more details, see [Conditional Mapping](#).
- **Remove condition:** Available only on nodes that already have a condition applied, this removes the condition.

Target Fields

For a transformation mapping to result in output data from the transformation, you must map to at least one field in the target schema. Once you have defined a target schema, you can map to the target fields in any of the ways covered below, including by mapping source objects, variables, scripts, or custom values to the target fields or adding other script logic:

- **Automapping**
Automapping is used to automatically map source and target fields whose names are an exact match within identical data structures. You can perform automapping using the entire data structure or within selected source and target nodes.
- **Mapping Source Objects**
In addition to automapping, source fields can be manually mapped to a target field. When you map a source field to a target field, the reference path of the source object is inserted into a script on the target field.
- **Mapping Variables**
When you map a variable, the variable reference is inserted into a script on the target field. Variable references are constructed with a dollar sign (\$) preceding the variable name.
- **Mapping Custom Values**
Custom values are simply hard-coded values that can be mapped to a target field.
- **Mapping with Scripts**
All transformation mappings are made up of scripts on individual nodes and fields within a target schema. This page covers how to open transformation scripts and reference project components and functions within them, although any valid script logic can be used.