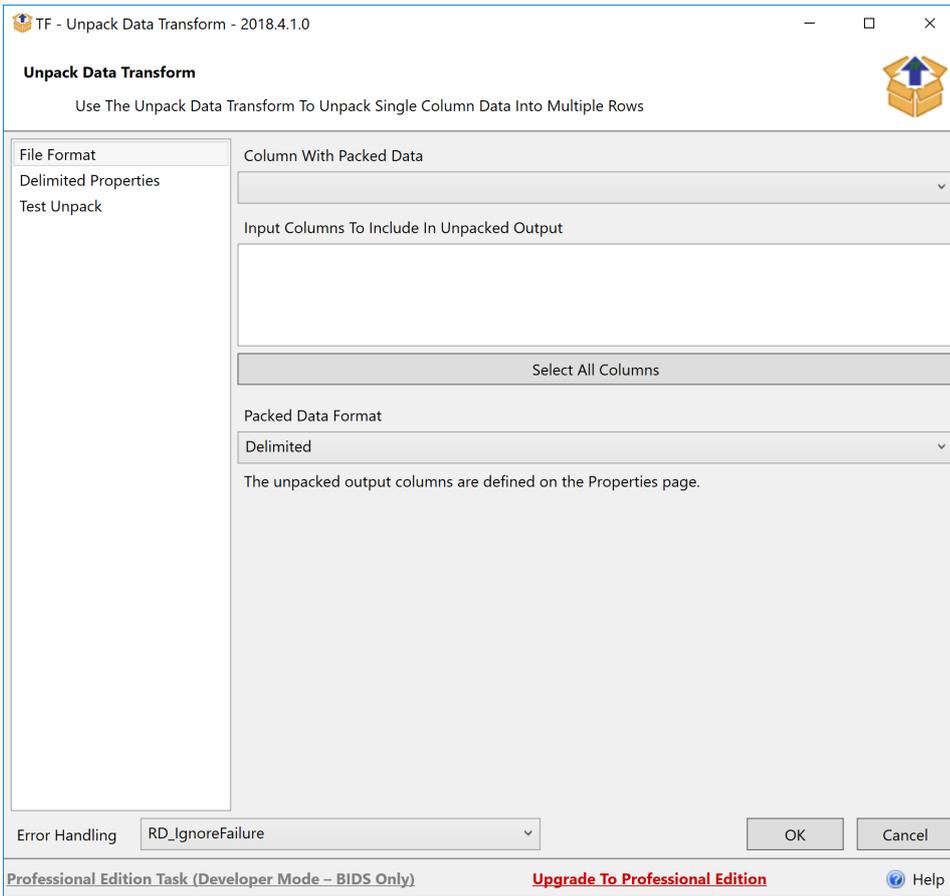# Task Factory Unpack Data Transform

📢 Task Factory users running version 2020.1.4 or older (released prior to May 27, 2020): **There's an important Task Factory update.** Please visit here for more details.

Unpack Data Transform

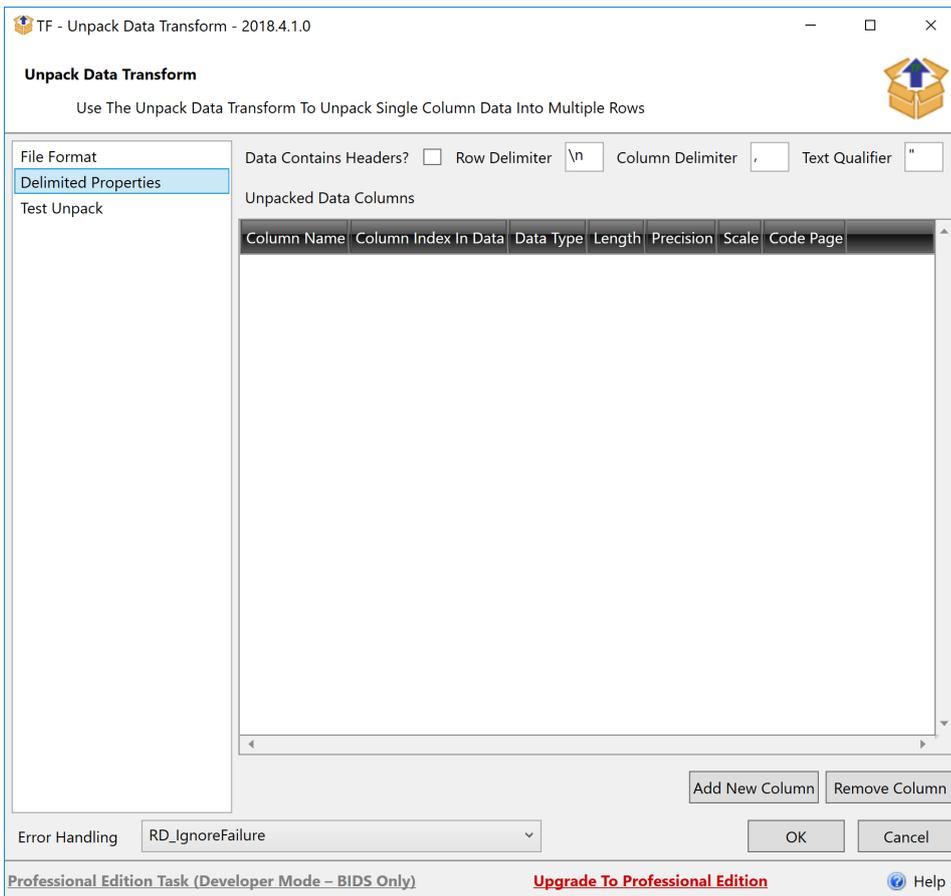⚠**Important:** Unpack Data is available for SQL versions 2012 and higher.

As of Feb 2, 2018, DT_EMPTY, DT_NULL and all datatypes that included BYREF have been removed. Any package configured to use these datatypes will error and have to be reconfigured.

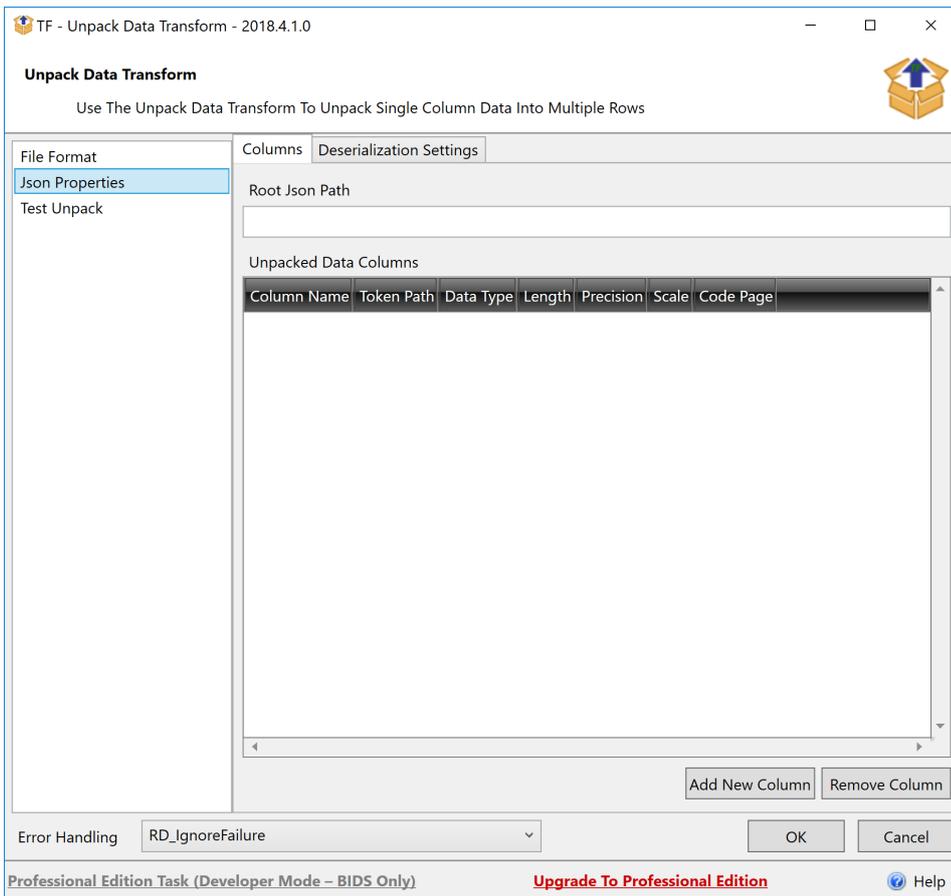| Transform Icon	Transform Description
	The **Unpack Data Transform** is used to output delimited, XML, or Json data contained within a single column source.

Option	Description
Column With Packed Data	Selects the column that stores the delimited or XML data to be unpacked.
Input Columns To Include In Unpacked Output	Selects other columns to be included in the Unpacked Output (non-selected columns continue down the standard output.)
Packed Data Format	Selects the format of the packed data . Data must be one of the following: • Delimited • JSON • XML

Delimited Configuration

Option	Description
Data Contains Headers?	Similar to the native Flat File Source, this selection identifies the first row as containing column headers.
Row Delimiter	Identifies a character or carriage return **(\n)** to signify a new row.
Column Delimiter	Identifies the character used to separate values for the different columns such as a comma.
Text Qualifier	Identifies the character used to wrap values such as quotation marks.
Unpacked Data Columns	Users can create, remove, and configure the name, index (zero-based), data type, length, precision, and scale of the columns being extracted or unpacked.

Json Configuration

Option	Description
Root Json Path	Json queries can return multiple levels, therefore, this field Identifies the root to be used.
Unpacked Data Columns	Users can create, remove, and configure the name, index (zero-based), data type, length, precision, and scale of the columns being extracted or unpacked.

Using Wildcards (*)

The Unpack Data Transform allows for wildcards when arrays are returned.

Example:
Using the Json Results below, the Root Json Path would be **results[0].appInventory[*]**. This information tells the component to use the array found at position **0**. Because some child arrays do not have an object name, a **wildcard** or ***** can be used to return all child array objects.

```
{
    "results": [
        {
            "appInventory": [
                {
                    "name": "Sample name 1",
                    "identifier": "123",
                },
                {
                    "name": "Sample name 2",
                    "identifier": "456",
                },
                {
                    "name": "Sample name 3",
                    "identifier": "789",
                }
                                ]
            }
        ]
}
```
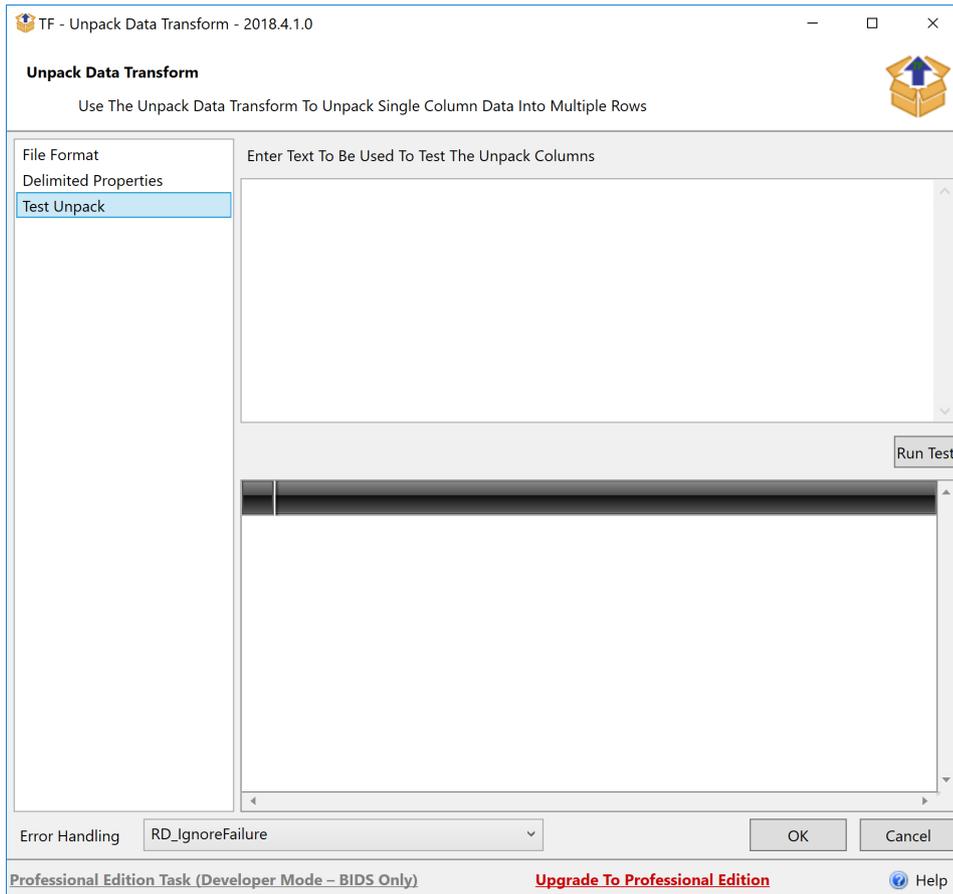
XML Configuration

Option	Description
Root XPath Query	Xml queries can return multiple levels, therefore, this field Identifies the root to be used.
	• **Prefix** - Identifies the prefix used in an xml namespace.

Option	Description
Namespaces:	○ Example: In the element the prefix is url. ○ **Url** - The Url or address identified in the namespace. In the example above, http://www.site.com is the uri that should be configured.
Unpacked Data Columns	Users can create, remove, and configure the name, index (zero-based), data type, length, precision, and scale of the columns being extracted or unpacked.

Test Unpack Columns

Option	Description
Enter Text To Be Used To Test The Unpack Columns	Users can copy and paste a row of packed data into this field to test the component's configuration and output.
Run Test	Executes a test based on the component's configuration and data pasted into the window above. Results display in the window below.

⧉Additional Information: See the Task Factory Error Row Handling article for more information about this functionality.