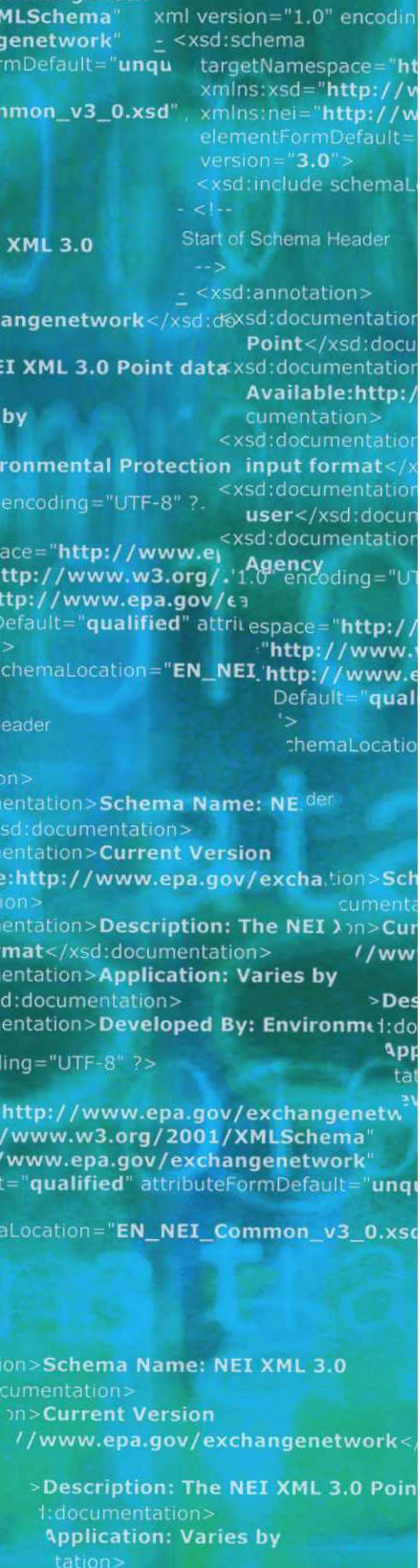


OpenNode2

TRI 6.0 Data Exchange Implementation Guide

Revision Date: 1/25/2017

Prepared By:

WINDSOR SOLUTIONS

4386 SW Macadam Ave, Suite 101
Portland, OR 97239
(503) 675-7833

Environmental Information exchange Network

Revision History

Date	Author	Changes	Version
11/26/2012	Windsor	Initial version for 5.0	1.0
10/9/2013	Windsor	Revised cover page	1.1
7/9/2014	Windsor	Updated Install Plugin section to describe pre-bundled plugin process starting with OpenNode2 v2.6	1.2
1/25/2017	Windsor	Updated for TRI v6.0. Clarified submission processor service configuration.	1.3

Table of Contents

THIS PAGE INTENTIONALLY LEFT BLANK

Data Exchange Overview

The purpose of this document is to provide detailed instructions for the installation and configuration of the Exchange Network Toxic Release Inventory (TRI) data exchange on the Microsoft .NET implementation of the Exchange Network OpenNode2 (OpenNode2). Further information about the TRI data exchange is available in the Flow Configuration Document (FCD) published at exchangenetwork.net.

The TRI plugin serves two main functions:

1. The TRISubmissionProcessor service receives TRI XML files from external partners (typically EPA CDX) and parses the data contained in those files into staging tables
2. The GetTRISubmitManifest service allows external systems and partners to retrieve a list of the received TRI submissions.

When a TRI XML file is received from an external partner, that file is stored in the OpenNode2 document repository. The TRISubmissionProcessor service can be run as a scheduled process to routinely look for newly received submissions. If a submission is located the service will parse the data to the TRI staging tables. These tables can then be used for direct read access by other applications.

In order to receive TRI submissions from US EPA, the OpenNode2 endpoint must be added to the TRI State Data Exchange (TRI SDX). For more information on how to join the TRI SDX, please visit http://epa.gov/tri/stakeholders/state/state_exchange/index.htm.

The second GetTRISubmitManifest service provided by the TRI plugin takes the form of a Network Query or Solicit service. This service enables an external partner to request a list of the TRI submissions received by the Node since a given date. This service is designed to be integrated into a downstream application, such as a tool that notifies staff that new TRI submissions have been received.

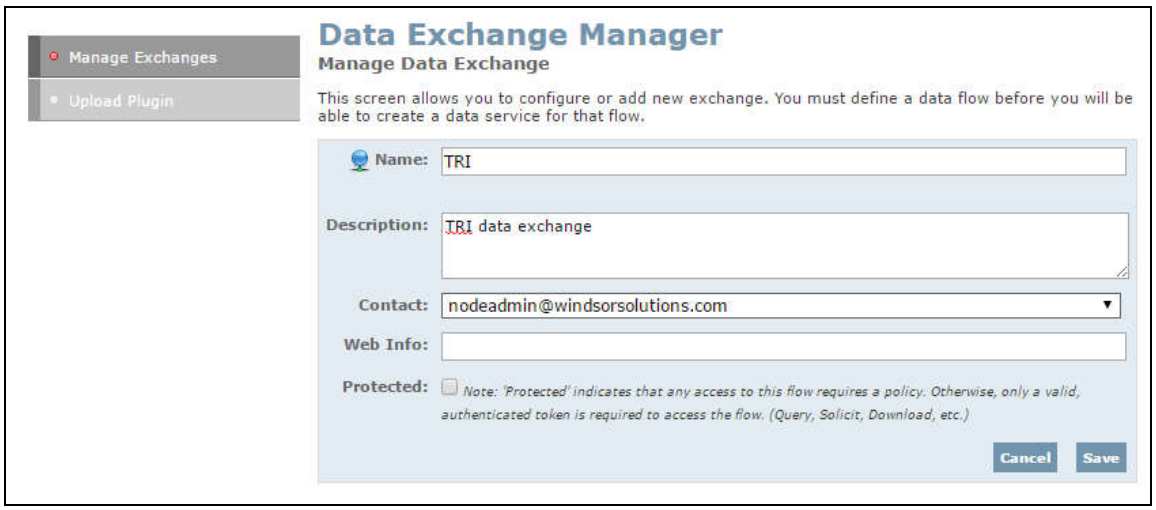
Install and Configure the TRI Data Flow

This section describes the steps required to install and configure the TRI data exchange on the Microsoft .NET implementations of the OpenNode2 using the Node Administration Web application (Node Admin).

Create the TRI Data Exchange

The first step is to create the TRI exchange using the OpenNode2 Node Admin Web application.

1. Click the **Exchange** tab on the top navigation bar.
2. Click the **Add Exchange** button. The Manage Data Exchange screen will be displayed as follows:



3. Type "TRI" in the **Name** field.
4. Type a short description in the **Description** field.
5. Select a user account name from the **Contact** drop down box. Contacts are populated with all accounts that have been set up on the Node 2008. See the **Security** tab for a list of available accounts. The contact information is not used by the node.
6. In the **Web Info** field, enter a URL where more information can be found about the exchange. It is recommended that the following URL be used for this purpose: http://www.exchangenetwork.net/exchanges/cross/tri.htm.
7. Checking the **Protected** box will require that users are granted authorization on this node for this flow before they will be able to query or download TRI submissions.
8. Click the **Save** button to save the data exchange to the OpenNode2 repository.

NOTE:

The TRI Data Flow does not require a plugin to be installed. If the TRI exchange is created with no plugin, submissions can still be received but no processing will occur. If you do not wish to configure any special processing of received files, no additional configuration is needed.

Create the Staging Database and Set Up Data Source

The TRI plugin package includes a SQL script to create the staging tables that can be used to store TRI report data parsed from received XML files. Follow the steps below to create the staging tables and make them available to OpenNode2.

Create the Staging Database

1. Create the staging database or schema if one has not already been established to support another data exchange (typically named NODE_FLOW). Alternatively, a stand-alone database can be created to store only TRI data.
2. Run the DDL script included with the plugin to create the staging database. The staging tables closely reflect the structure and naming of the TRI XML schema.

Create the OpenNode2 Staging Table Data Source

NOTE:

This step is only necessary if the TRI staging tables were placed in a new database or schema in the previous step. If the TRI staging tables were placed in an existing database or schema, the data source has likely already been configured for a previous flow.

OpenNode2 must be made aware of the TRI staging database. Follow the steps below to create the new data source within OpenNode2.

1. Log into OpenNode2 Admin and click the **Configuration** tab.
2. Click the **Data Sources** link on the left navigation bar. The Data Sources screen will be displayed.
3. Click **Add Data Source**.
4. Type a name for the new connection such as "TRI_STAGING_DB".
5. Select the data provider to be used (SqlClient is for SQL Server).
6. Type the connection string to be used to connect to the staging database.
7. Click **Check Connection** to verify to connection information.
8. Click **Save** when done.

Create the OpenNode2 Data Source for the GetTRISubmitManifest service (optional)

If implementing the GetTRISubmitManifest service, a data source must be created for the plugin to access the OpenNode2 metadata database. The node's metadata database stores the transaction information needed by the plugin to produce the TRI manifest.

1. Log into OpenNode2 Admin and click the **Configuration** tab.
2. Click the **Data Sources** link on the left navigation bar. The Data Sources screen will be displayed.
3. Click **Add Data Source**.
4. Type a name for the new connection such as "OPENNODE2_METADATA".
5. Select the data provider to be used (SqlClient is for SQL Server).
6. Type the connection string to be used to connect to the OpenNode2 metadata database. The connection string can be found in the **deployment.config** configuration file.

7. Click **Check Connection** to verify to connection information.
8. Click **Save** when done.

Install the TRI Plugin

The TRI Plugin provides the capability for OpenNode2 to process received TRI files by parsing the file into the TRI staging database. The TRI plugin can be downloaded from the OpenNode2 SourceForge open source Web site.

Follow the steps below to install the TRI plugin on OpenNode2.

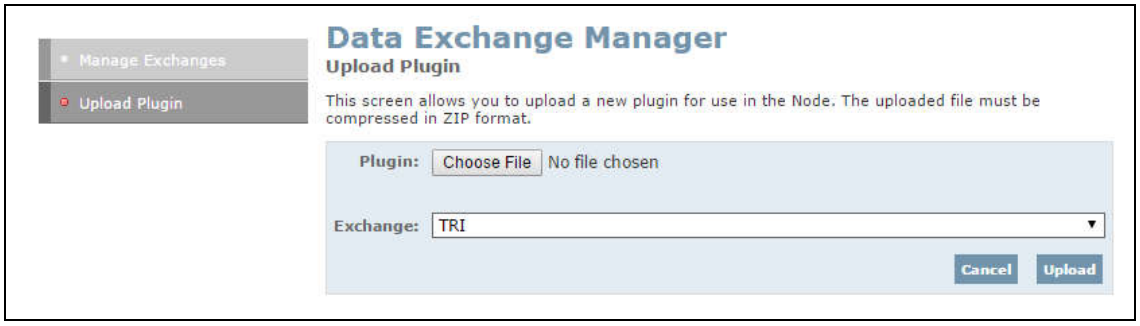
NOTE:

If you are using OpenNode2 v2.6 or higher, this step is not necessary. Starting with v2.6, all plugins are pre-installed with the OpenNode2 software installation package. By creating the exchange above, the plugin will automatically be loaded and associated with the exchange. To validate that the plugin was installed automatically, follow the steps below:

1. From the **Exchange** tab, scroll down the list of installed data exchanges until the WQX exchange is located.
2. Click the **Add Service** button located just beneath the WQX data exchange record. If the Implementer drop down box is not empty, then the plugin has been installed successfully.

If the steps above reveal that the plugin is not installed, perform the following steps to install it.

1. Navigate to the plugin directory in the **Plugins\[Flow Name]\[version number]** directory included with the OpenNode2 installation files.
2. Create a new zip file containing the two Windsor.Node2008.WNOSPlugin.[Flow name].dll and .pdb files.
3. Click the **Exchange** tab on the top navigation bar.
4. Click the **Upload Plugin** link on the left navigation bar. The Upload Plugin screen will be displayed as follows:



5. Click the **Browse** button which is located to the right of the **Plugin** field.
6. Locate and select the plugin (ZIP format) that you created in step 2 above.
7. Select the data exchange name "TRI" from the **Exchange** dropdown box.
8. Click the **Upload** button to upload the plugin.

The newly uploaded plugin code will be placed in the OpenNode2 plugin repository. Any previous plugin versions will be retained in the repository but won't be accessible through the Node Admin. Only the latest version of any one plugin is made available during the next step to establish data services.

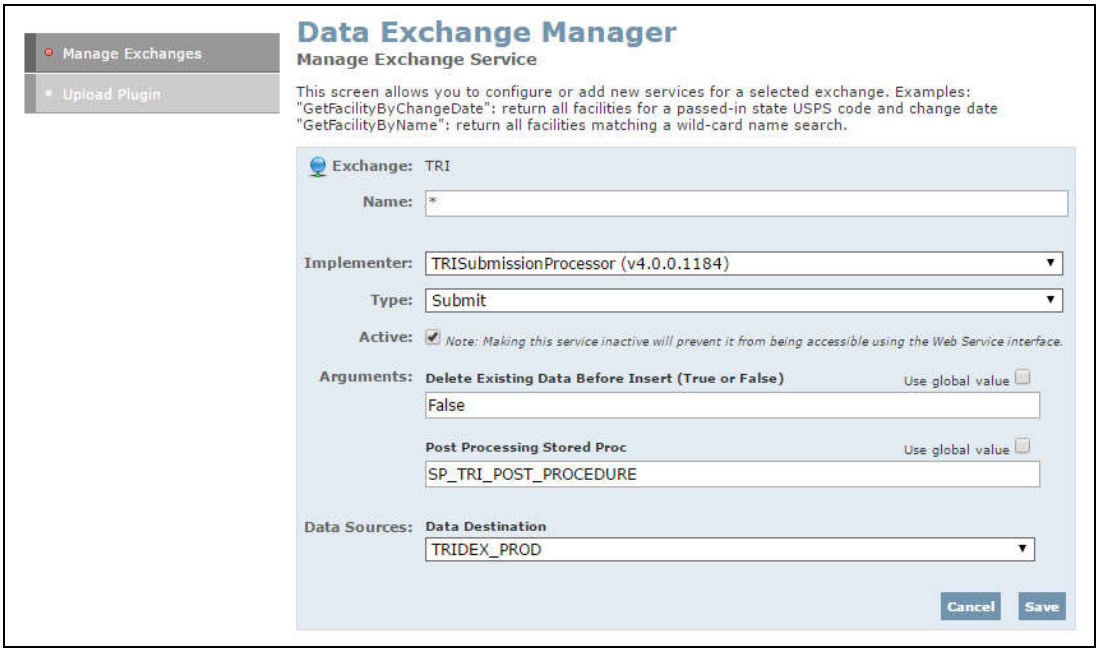
Create the TRI Submission Processor Service

The Submission Processor service will make the TRI submission parsing services available to the Node. Once this service has been established, it will be possible to create a Schedule in the Node Admin which will regularly check for newly received XML files and will parse the data from those files into the staging tables established earlier.

NOTE:

This service can also be configured to optionally execute a custom database procedure which, if desired can be used to further move data that has been parsed into the staging tables on to an additional data repository. If the provided staging tables are to be the final destination for the received TRI data, then this step can be omitted.

1. From the **Exchange** tab, locate the TRI data exchange in the list of available exchanges.
2. Click the **Add Service** button located just beneath the TRI exchange entry. The following page will be displayed to allow a new data service to be added.



3. In the **Name** field, type "TRISubmissionProcessor"
4. In the Implementer drop-down menu, select the TRISubmissionProcessor service

   Note: When the implementer is selected, several arguments will appear. The Node Admin will obtain these properties directly from the TRI plugin.

5. From the Type drop-down menu, choose **Submit**. This is the only option available.
6. Enable the service by checking the **Active** checkbox.
7. Set the field **Delete Existing Data Before Insert (True or False)** to False.

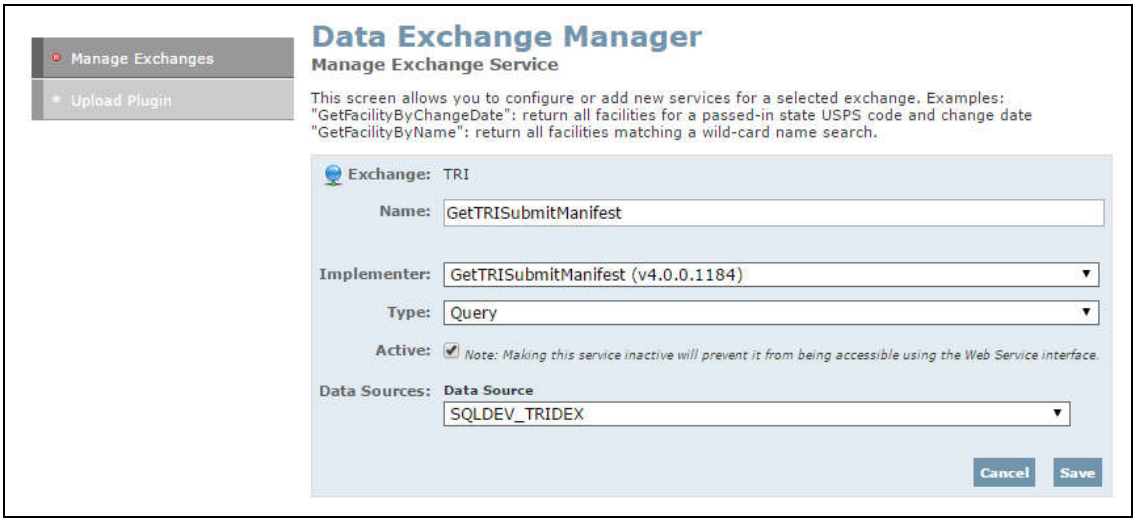
8. Optionally set the field **Post Processing Stored Proc** to the name of a stored procedure that should be executed by the service to further process the parsed XML data from the staging tables.
9. In the Data Sources area, choose the TRI data source created earlier as the **Data Destination**.
10. Click the Save button to save the Service.

Repeat the steps above to create a second submit service, substituting an asterisk (*) for the Name in step 3. This is needed to tie the TRI submit processor to the Node v1.1 endpoint which does not support the operation parameter (added in Node 2.0).

Create the Get TRI Submit Manifest Service (Optional)

Data services are distinct functions provided by a plugin to support a given data exchange. The TRI exchange contains only one service, GetTRISubmitManifest that allows external partners or applications to retrieve a list of received TRI files.

1. From the **Exchange** tab, locate the TRI data exchange in the list of available exchanges.



2. Click the **Add Service** button located just beneath the TRI exchange entry. The following page will be displayed to allow a new data service to be added.
3. In the **Name** field, type "GetTRISubmitManifest."
4. From the **Implementer** drop down box, select "GetTRISubmitManifest".
5. From the **Type** drop-down list, select the "Query" option.
6. Enable the service by checking the **Active** checkbox.
7. In the Data Sources area, choose the OpenNode2 metadata data source created earlier as the **Data Source** (e.g. OPENNODE2_METADATA).
8. Click the **Save** button to save the service.

Set Up Email Notifications

If desired, the Node administrator may create NAAS accounts for one or more staff members and create notifications for the any OpenNode2 events related to the TRI data exchange. Please see the Node Administration Guide for more information on setting up notifications.

Monitor Flow Activity

The OpenNode2 will track all TRI data exchange activity and can be accessed to monitor and debug related flow activities. Please see the OpenNode2 Administration User Guide for more information on accessing and searching the available OpenNode2 activity reports.

Appendix A – TRI Manifest XML Format

The GetTRISubmitManifest returns data in an XML format. The schema and a sample XML file are listed below. Please note that this schema is custom to the OpenNode2 plugin and is not defined in the TRI FCD.

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:tns="http://www.windsorsolutions.biz/xsd/TRI_v5.0/SubmittedDocuments.xsd"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.windsorsolutions.biz/xsd/TRI_v5.0/SubmittedDocuments.xsd"
elementFormDefault="qualified">
	<xs:element name="SubmittedDocumentList" type="tns:SubmittedDocumentList"/>
	<xs:complexType name="SubmittedDocumentList">
		<xs:sequence>
			<xs:element name="SubmittedDocuments" type="tns:ArrayOfSubmittedDocument"
minOccurs="0"/>
		</xs:sequence>
	</xs:complexType>
	<xs:complexType name="ArrayOfSubmittedDocument">
		<xs:sequence>
			<xs:element name="SubmittedDocument" type="tns:SubmittedDocument"
nillable="true" minOccurs="0" maxOccurs="unbounded"/>
		</xs:sequence>
	</xs:complexType>
	<xs:complexType name="SubmittedDocument">
		<xs:sequence>
			<xs:element name="TransactionID" type="xs:string" minOccurs="0"/>
			<xs:element name="ReceivedDate" type="xs:dateTime"/>
		</xs:sequence>
	</xs:complexType>
</xs:schema>
```

TRI Manifest XML Schema

```
<?xml version="1.0" encoding="UTF-8"?>
<tns:SubmittedDocumentList
xsi:schemaLocation="http://www.windsorsolutions.biz/xsd/TRI v5.0/SubmittedDocuments.xsd
TRIManifest.xsd" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:tns="http://www.windsorsolutions.biz/xsd/TRI_v5.0/SubmittedDocuments.xsd">
	<tns:SubmittedDocuments>
		<tns:SubmittedDocument>
			<tns:TransactionID>111</tns:TransactionID>
			<tns:ReceivedDate>2009-12-17T09:30:47.0Z</tns:ReceivedDate>
		</tns:SubmittedDocument>
		<tns:SubmittedDocument>
			<tns:TransactionID>222</tns:TransactionID>
			<tns:ReceivedDate>2009-12-17T09:30:47.0Z</tns:ReceivedDate>
		</tns:SubmittedDocument>
		<tns:SubmittedDocument>
			<tns:TransactionID>333</tns:TransactionID>
			<tns:ReceivedDate>2009-12-17T09:30:47.0Z</tns:ReceivedDate>
		</tns:SubmittedDocument>
	</tns:SubmittedDocuments>
</tns:SubmittedDocumentList>
```

Sample TRI Manifest XML File