EN-NODE RESTFUL WEB SERVICE USER'S GUIDE (.NET VERSION)

Version: 1.0

October 30, 2013

enfoTech

1368 How Lane
North Brunswick, New Jersey 08902
www.enfotech.com

Revision History

Version	Date	Created By	Reviewed By	Description
1.0	10/30/2013	Charlie Tsai		

Tables of Contents

1   Introduction

1.1   Document Purpose

The EN-Node RESTful Web Service application is a new extension to the EN-Node application, which enables data providers to publish their query web services as REST services and data consumers to access the REST services from a dedicated webpage.

This document provides instructions on how to:

- Publish a query service as a REST service
- Configure contents of the REST webpage in the Node Admin Console
- Access the published REST service
- Request and obtain data via the REST service

1.2   Overview of EN-Node RESTful Web Service

The EN-Node is a web services based data exchange portal, which allows environmental state agencies to submit data to the EPA and exchange data with other environmental partner/Nodes.

The RESTful Web Service application is an Exchange Network compliant extension to the EN-Node software, and provides the data provider with the capability to publish their query services as REST services. In addition, the REST extension provides a dedicated interface (e.g. webpage) for data consumers to access and request data from these REST services.

For additional information on the concept and use of RESTful web services, please view the REST Guidance document on the Exchange Network website: http://www.exchangenetwork.net/rest-guidance/

1.3   The Structure of EN-Node SOAP and RESTful Web Services

The Node comes preconfigured with the 9 standard SOAP Web services defined in the Node 1.1/2.0 specifications. These 9 Web services include[1]:

- **Authenticate:** The Authenticate method authenticates a user using a supplied credential.
- **Submit:** The Submit method provides a generic way of sending one or more payloads to a service provider.
- **GetStatus:** GetStatus is a method for transaction tracking. Once submitted, a transaction enters into different processing stages. The GetStatus method offers the client a way of querying the current state of the transaction.
- **Query:** The Query method is a function in the Database interface. The method is intended to run a series of predefined information requests that return data in an XML instance document that conforms to a predefined standard schema.
- **Solicit:** The Solicit method performs the requested operation in the background or sometimes offline. It is designed especially for queries that may take a long time.
- **Notify:** The Notify method has three (3) intended uses: document notification, event notification, and status notification
- **Download:** The Download method is a function in the Retrieve Interface. After being notified by a submitter, a Node invokes the Download method to retrieve available documents
- **NodePing:** The NodePing method is a function in the Admin interface. It is a utility method for determining whether a Node is accessible.

---

[1] For more information on the Node Web Services, please refer to the Node Version 1.1 Specifications at: http://www.exchangenetwork.net

- **GetServices:** The GetServices method is a function in the Admin interface. It allows requesters to query services provided by a Network Node.

Unlike the 9 standard SOAP web services, REST services are represented as a URL. The data consumer may copy and paste a REST URL into Internet Explorer (IE) or any comparable Internet Browser to obtain data from the data provider.

An example REST service URL is provided below:

https:// localhost /RESTServices /Query?Dataflow=NCT&Request=Query_v1.0&Params=stringParameter|string&RowID=0&MaxRow s=100

The following is the list of standard parameters for calling a REST service:

Name	Description
Dataflow	The name of the dataflow associated with the service.
Request	The name of the service request.
Params	A concatenated listing of all parameters for the Query or Solicit service in the format of: *param1|value1;param2|value2*
	Where parameter name/value pairs are separated with a pipe (|) delimiter and parameters are separated with a semi-colon (;) delimiter.
	When passing multiple values for the same parameter, the values are expressed as distinct name/value pairs. For example, ZipCode|20635;ZipCode|20637
	Note this is a condensation of all Query parameters into one REST parameter.
RowId	An optional rowId parameter for the Query/Solicit method that identifies the starting record number to be retrieved when returning paged results.
MaxRows	An optional maxRows parameter for the Query/Solicit method that identifies the maximum number of records to return.
Format	Identifies the format in which data will be returned. Commonly used formats should use the following naming convention (xml, json, html, csv, xls). The returned format is xml if the parameter is omitted.

1.4 Overview of How EN-Node RESTful Web Service Works

EN-Node provides each of the Web services through the use of "Handlers". The handler determines the logic/approach by which a Web Service request will be handled. Within each Handler, multiple Operations can be defined that execute a specific sequence of logic. These Operations are defined by and organized by Domain Administrators.

The Generic Handler contains the generic logic that EN-Node executes regardless of the Web Service that is invoked. The generic handler has the following behavior:

- Generates a transaction ID each time a new Web Service request is made on the e-Node
- Initiates logging of the Web Service request to track the status of the request
- Authorize the request to ensure if the web service is public and restful, if the web service is not public and restful, then the authorization request will fail.
- Once authorization is completed, the EN-Node searches for and executes in order any registered Pre-Processes for the Operation.
- EN-Node then searches for and executes the registered Process for the Operation and returns the invocation response value to the requestor as per Node specifications.
- Once the Web Service method has been processed, EN-Node searches for and executes in order any Post-Processes registered for the Operation.
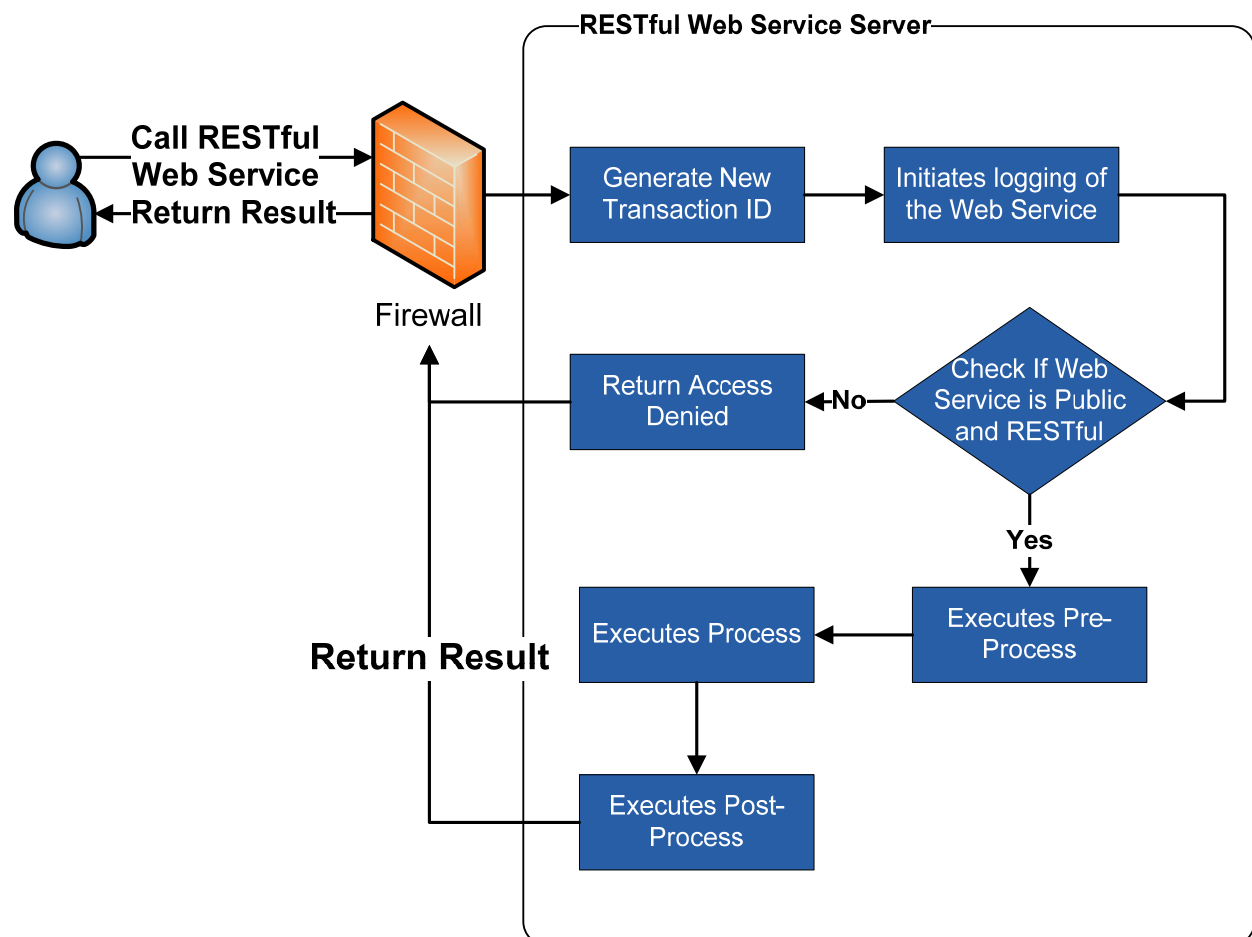

Figure 1-1: Node RESTful Web Service Process

2   Defining REST Web Services in the Node Admin Console

Data providers may create a "mirror" REST web service of an existing Query service in the Node Admin Console. When a data provider selects the option to set an existing Query as a REST service, the Node will automatically display the REST service in the REST Service Portal (please see section 3 for more details).

The new REST web service will not override or replace the existing Query service. Both services will coexist in the Node.

Steps on how to set an existing Query service as a REST service are provided below:
1. Log into the Node Admin Console
2. Navigate to your desired query service in the Node Domain web part.

Figure 2-1: Domain WebPart Screen

3.   Click the Edit link of the Query.

Figure 2-2: Operation Management Screen

4. Click the checkbox for the "Enable RESTful Service" to enable the query service as a REST service in the REST Portal.

Figure 2-3: Define Query as REST Service

Below are some constraints that should be taken into consideration when setting up the REST Service:

- As stipulated in the EN REST_Guidance_v1.1 document: "The Exchange Network recommends using REST services only for data that are considered public. Exchange Network partners should use the EN's SOAP-based web services in conjunction with the Network Authentication and Authorization Service (NAAS) for any data that require restricted access."

- In addition, the EN REST_Guidance_v1.1 document also states that: "The HTTP GET method is well suited for supporting simple queries that do not require multi-step processing. Business processes that require behind the scenes status checks or complex data extractions before making results available are typically better served by the asynchronous behavior of EN Nodes using the SOAP-based Solicit method. Similarly, data submission services that include status checking and downloading of results may also be poor candidates for the REST approach."

The EN-Node REST Portal has been developed in accordance with the recommendations stipulated in the EN REST_Guidance_v1.1 document, and is intended to be used for public query services.

3   Using the EN-Node RESTful Web Service Portal

The EN-Node RESTful Web Service Portal provides an interface for user to discovery available REST services, and request data from the data provider.

The portal supports the following functions:
- Display available REST services
- Display available parameters for the selected REST Service
- Construct REST URL String based on filter criteria selected by the user
- Request data from the data provider's Node through the REST URL

An example REST portal URL is shown here: "http://<<InstallLocation>>/RESTServices/default.aspx".

3.1   REST Portal Header

Text displayed in the REST Portal Header can be configured by the Node admin in the Node Admin Console.

An example screenshot of the REST Portal Header is provided below:

Instructions on how to configure the REST Portal Header text are provided below:
1. Log into the Node Admin Console.
2. Under the Node Configuration web part, click the "RESTful Configuration" link.
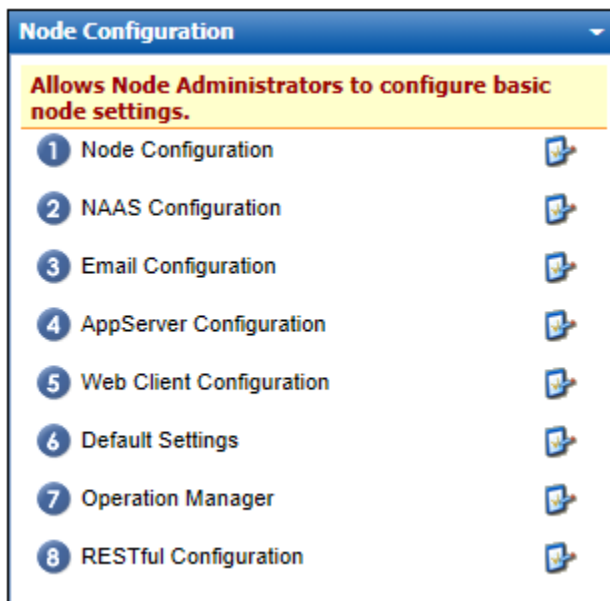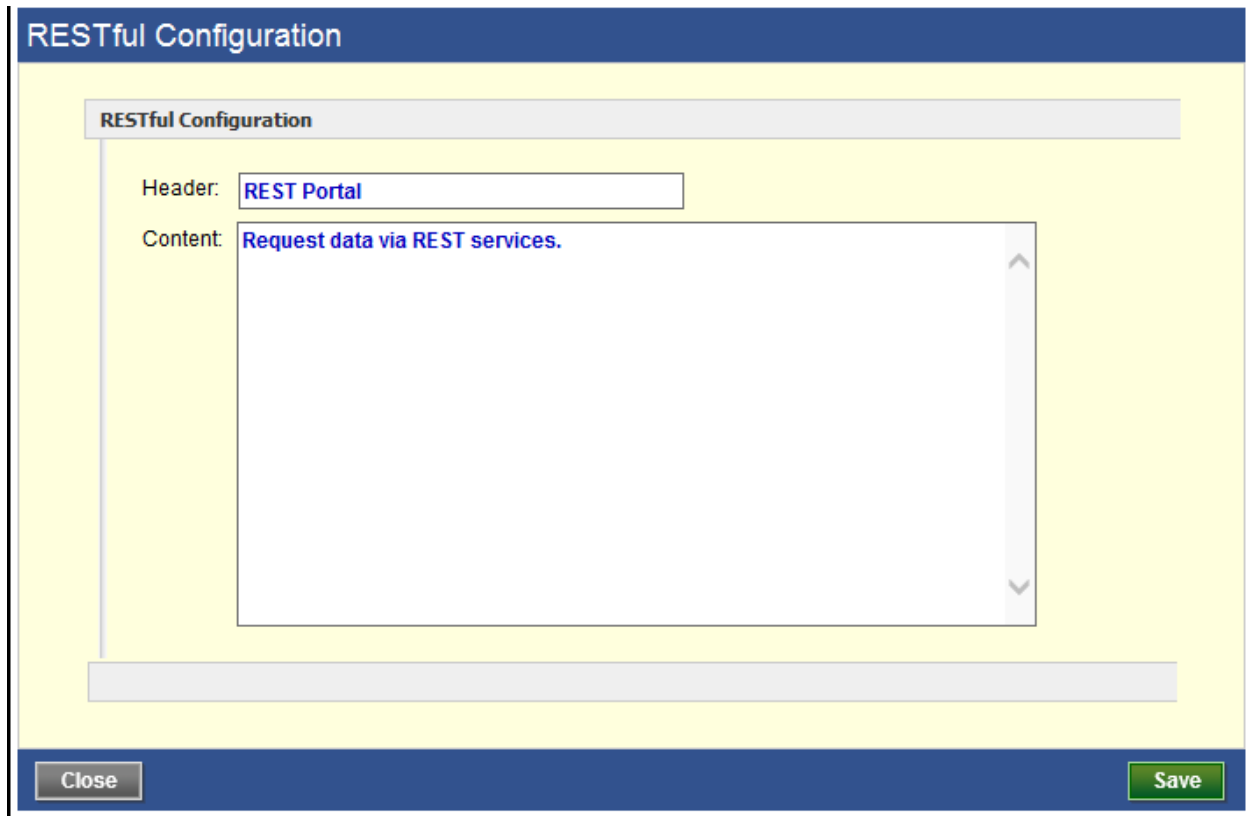

Figure 3-1: Node RESTful Web Service Navigation Panel

3. Specify the page title in the "Header:" section. Specify the contents of the header in the "Contents" section and click "Save".

Figure 3-2: Node RESTful Portal Configuration

3.2   Discovering Services from the REST Portal

The REST portal displays all available REST services in the left-hand pane tree view. The objects shown in the tree view are structured in the following hierarchy:

- Dataflow 1
    - o   REST Service 1
    - o   REST Service 2
    - o   …

By default, all dataflows will be expanded and all REST services will be shown in the left panel. The user may click on the dataflow to minimize or expand the REST services listed under it, or click on the REST service link to display the REST service operation panel.

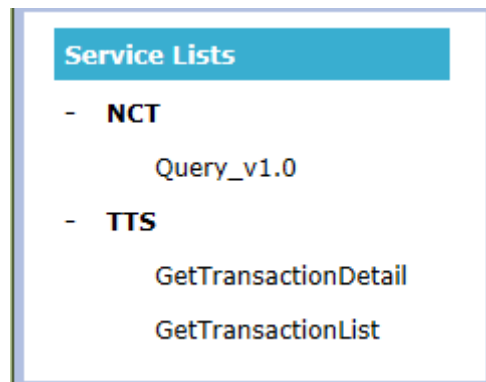An example screenshot of the REST navigation panel is provided below.

Figure 3-3: Node RESTful Web Service Navigation Panel

3.3   EN-Node RESTful Web Service Operation Window

Users may request data via the RESTful Web Service Operation Window. Instructions on how to build and submit a request are outlined below.

1. Click on a REST service in the left-hand REST navigation panel to display the REST service window. You may minimize, expand, or close the window by clicking on the "-". "+", "x" buttons respectively as depicted in the screen below.

Figure 3-4: Node RESTful Web Service Window close / toggle function

2. Fill in your desired filter parameter values. All REST parameters are displayed as "free text". The REST URL will automatically be adjusted depending on the parameters values inputted by the user.

Figure 3-5: The Example of RESTful Service Detail

3.   Click the Invoke button to request data.

RESTful URL:

```
http://vs08concord/ENNode/RESTServices/Query?
Request=Query_v1.0&RowId=5&MaxRows=10&Format=XML
```

Invoke

Figure 3-7: Invoke RESTful Service

4.   The REST service will return a zipped data file to the user when the request has been completed by the Node. In the event that the request fails (e.g. due to an internal Node error or incorrect parameters), the Node will return a zipped XML file to the end user which contains the following message: "Error: error occurred while processing your RESTful request. Please contact the data provider for further assistance."