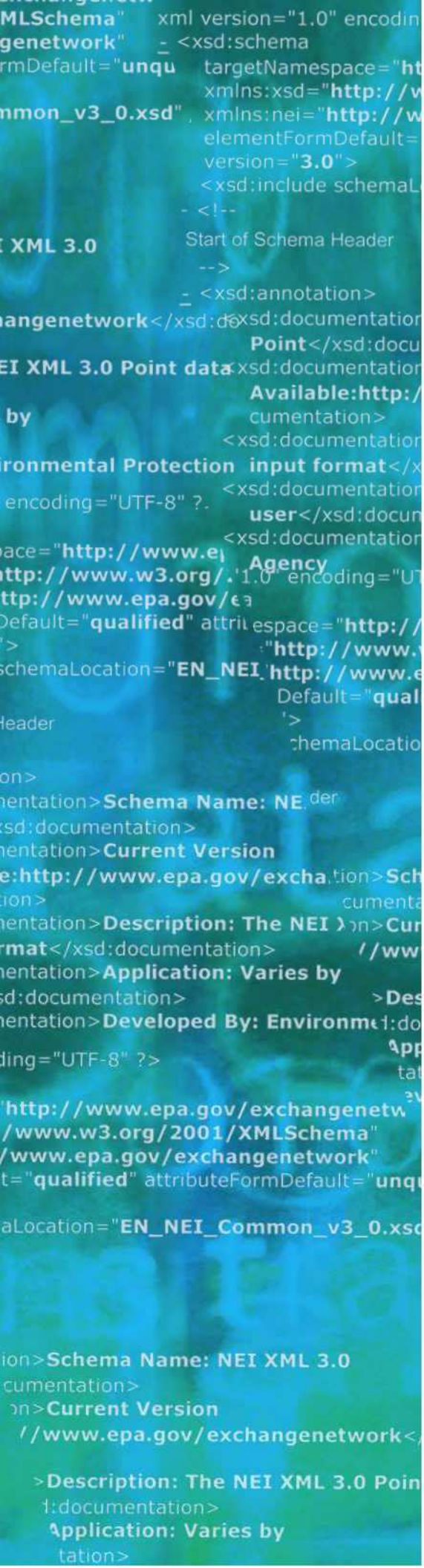


OpenNode2

AQS 3.0 Data Exchange Implementation Guide

Revision Date: 8/17/2016

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
9/12/2012	Windsor	Initial Version	1.0
10/9/2013	Windsor	Revised cover page	2.0a
6/24/2014	Windsor	Added staging table block diagram to Appendix B	2.1
7/9/2014	Windsor	Updated Install Plugin section to describe pre-bundled plugin process starting with OpenNode2 v2.6	2.2
8/17/2016	Windsor	Updates from v2.2 to v3.0. Notes added that plugin does not support new QA/QC modules in v3.0 at this time. Plugin still uses v2.2 staging tables	3.0

Table of Contents

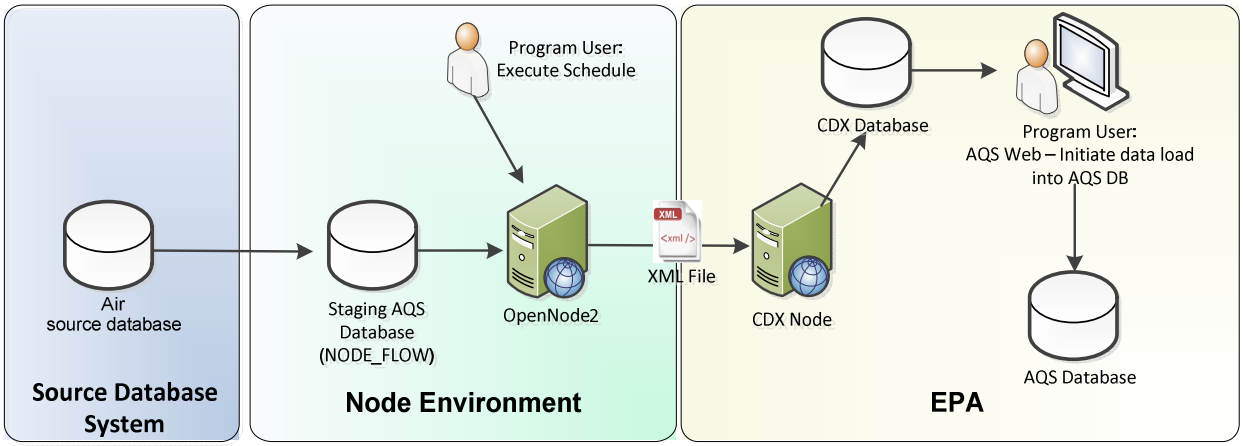
THIS PAGE INTENTIONALLY LEFT BLANK

Data Exchange Overview

The purpose of this document is to provide detailed instructions for the installation and configuration of the Air Quality System (AQS) data exchange on the Microsoft .NET implementation of the Exchange Network OpenNode2 (OpenNode2).

The AQS Exchange involves a periodic submittal to EPA for the purposes of updating the EPA AQS database which manages air quality site locations and monitoring data. The frequency and content of the each submission can be configured to meet the agency's needs.

The diagram below illustrates the basic steps involved in the AQS data exchange.



The AQS data exchange processing workflow can be briefly summarized as follows:

1. Data for a given reporting period will be collected and validated by the State or Tribe air program.
2. In OpenNode2 the data manager will then initiate the data flow to the EPA by executing the extract and Xml generation data service. This will pull the data from the staging tables based on Site ID (optional), start date and end data parameters. The XML submission data service which will automatically generate the XML file based on data stored in the staging database and be submitted to the EPA CDX environment.
3. The EPA CDX archives a copy of the submission and performs basic file validity checks.
4. Upon successful validation, the data manager logs into the EPA AQS Web site to initiate the AQS load processing from within the AQS application
5. The data is then parsed and stored in the AQS database.
6. CDX sends an email to the submitter indicating success or failure.

Install and Configure AQS Data Flow

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

Create the AQS Staging Tables

The OpenNode2 software package contains a folder of database scripts to create the staging tables and objects used by the various plugins. The AQS staging table creation script can be found in **Sql\AQS.**

Run the script for your database platform. Scripts are available for Oracle and SQL Server.

Note: The plugin uses v2.2 staging tables although it produces an AQS v3.0 XML file. The new elements available in v3.0 relating to Quality Assurance data are not currently supported by the plugin.

Create AQS Data Exchange

The first step to implement the AQS data exchange on the OpenNode2 is to create the data exchange using the Node Admin Data Exchange Manager.

1. After logging into the Node Admin, click the **Exchange** tab on the top navigation bar.
2. Click the **Add Exchange** button. The Data Exchange Manager screen will be displayed:

3. Type *AQS* in the **Name** field.
4. Type a short description in the **Description** field, e.g., *Air Quality System data exchange*.
5. Select a user account name from the **Contact** drop-down menu. Contacts are populated with all accounts that have been set up on the OpenNode2. See the **Security** tab for a list of available accounts. This selection has no consequence on the exchange configuration.
6. Type any valid URL in the **Web Info** field. Ideally, this will be the page on the Exchange Network Web site that describes the AQS data exchange:

   http://www.exchangenetwork.net/data-exchange/air-quality-system/

7. Check the **Protected** box. This will limit external access to the AQS data services. External access is not required if the purpose of this flow is solely as a means of data submission to EPA's AQS System.
8. Click **Save** to save the data exchange.

Install AQS Plugin

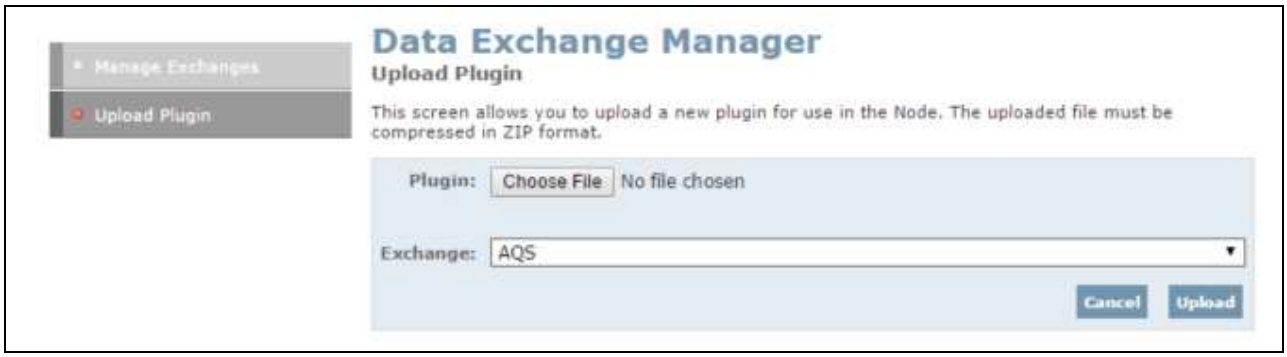
Once the AQS data exchange has been created, the next step is to upload the AQS plugin provided by Windsor into the OpenNode2 plugin repository.

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. From the **Exchange** tab, click the **Upload Plugin** button on the left side navigation block.



4. Click the **Choose File** button located to the right of the **Plugin** field.
5. Locate and select the compressed (zipped) file containing the code component for the AQS plugin you created in Step 2 above.
6. Select *AQS* from the **Exchange** drop-down menu. If *AQS* is not available, ensure that the previous step was completed (*Create AQS Data Exchange*).
7. Click the **Upload** button to install 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 AQS Data Services

Data services are distinct functions provided by a plugin to support a given data exchange. The AQS data exchange plugin provides the following services:

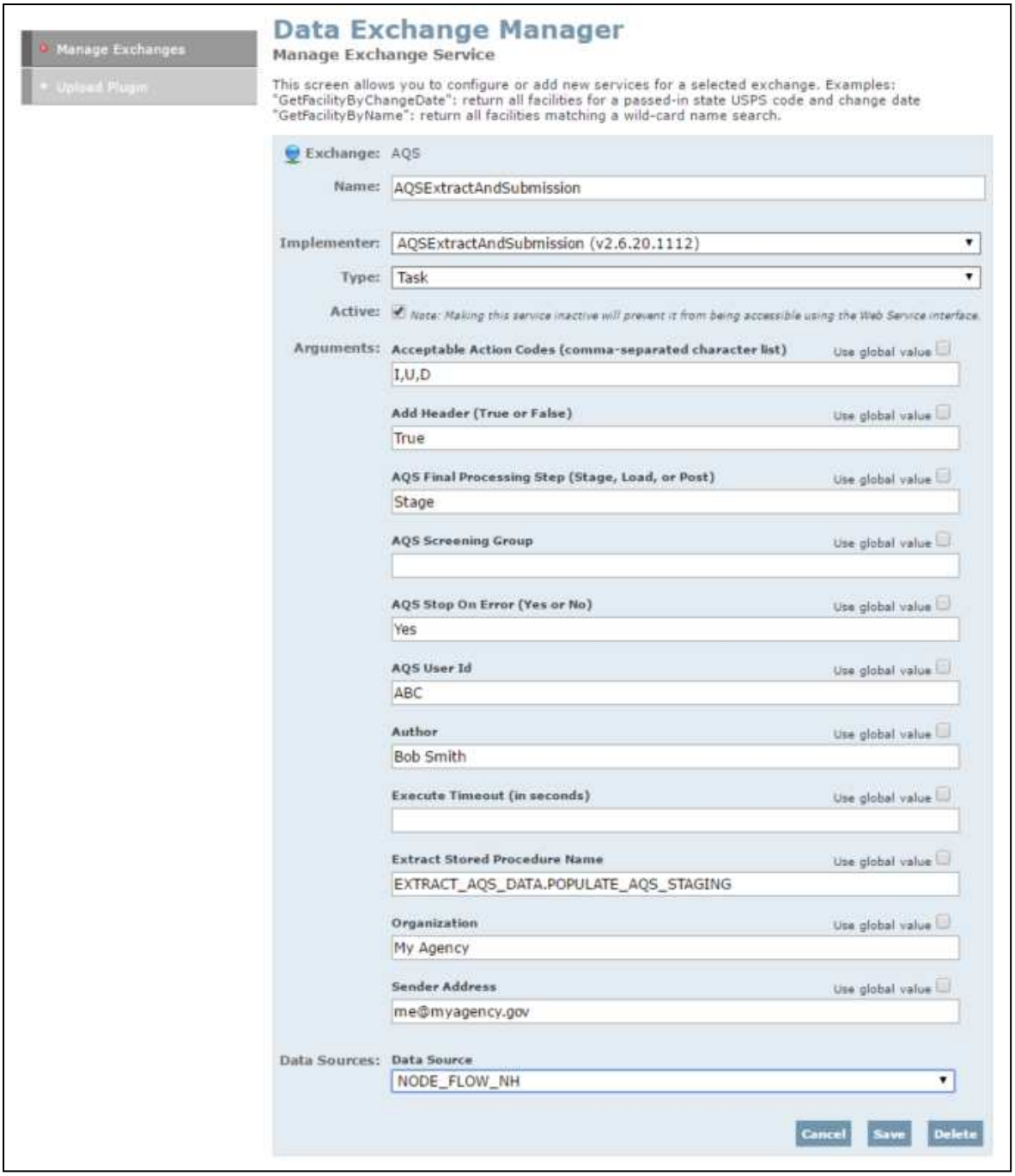
- The *AQSExtractAndSubmission* service extracts the data to populate the staging tables and submits the XML data to a specified endpoint.
- The *GetAQSStatusAndDownloadReports* service will access and update the status of all transactions in the OpenNode2.
- Several additional services are not supported and should not be used.

Only the *AQSExtractAndSubmission* and *GetAQSStatusAndDownloadReports* services are described in this document. The two other services were developed for a specific agency and are not usable outside of the agency's context.

AQSExtractAndSubmission Service

This data service combines two primary operations, including extracting the data to populate the staging tables and submitting the XML data to a specified endpoint.

1. From the **Exchange** tab, scroll down the list of installed data exchanges until the AQS exchange is located.
2. Click the **Add Service** button located just beneath the AQS data exchange record. The following page will be displayed to allow a new data service to be added.



3. In the **Service Name** field, type “AQSExtractAndSubmission”.
4. From the **Implementer** drop down box, select the value “AQSExtractAndSubmission”

   Note: When the implementer is selected, several arguments and data sources will appear. The OpenNode2 application will obtain these properties directly from the AQS plugin loaded previously.

5. From the **Type** drop down box, select how you wish to make the services available. The options available will be obtained from the plugin. Select “Task”.

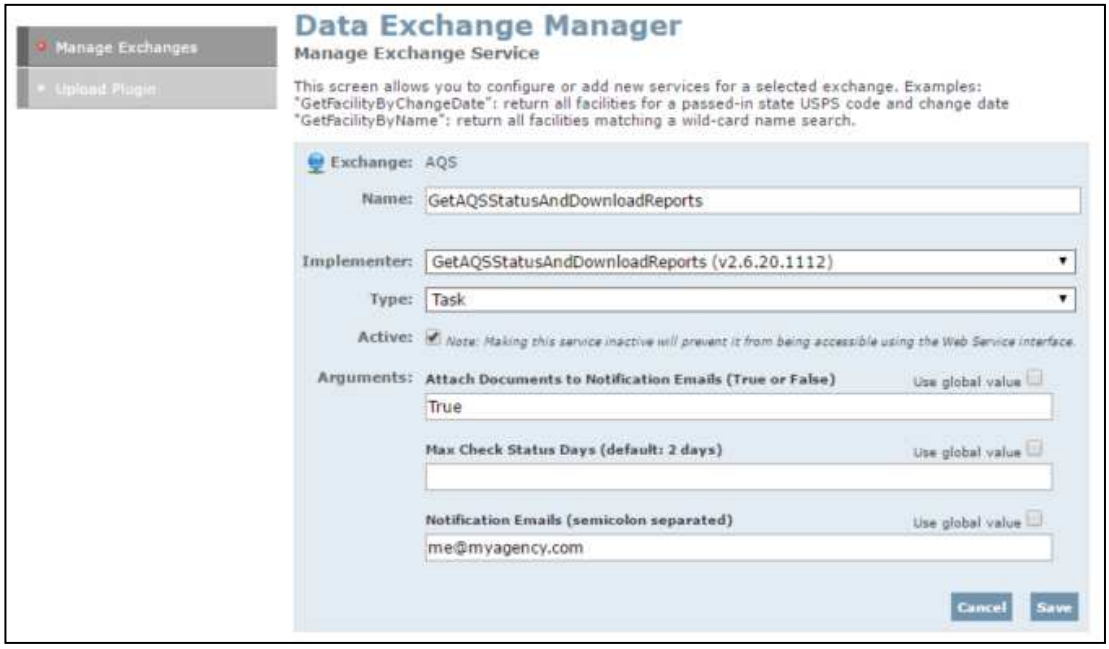
6. Based on the selection made from the implementer dropdown box, OpenNode2 will determine what argument and data source requirements the plugin has and will refresh the page to display the relevant data entry fields as follows:
7. Arguments:
    a. **Acceptable Action Codes**: I, U, D
    b. **Add Header**: type "*True*". The Header document is required by EPA CDX for AQS submissions.
    c. **AQS Final Processing Step:** type either "Stage", "Load", or "Post" as desired to indicate to the EPA AQS load processor the last step of the process that should be performed on the submitted data.
    d. **AQS Screening Group**: type the name of the AQS screening group that owns the monitors included in the data submission.
    e. **AQS Stop On Error:** type either "Yes" or "No" as desired to indicate to the AQS load processor whether or not to stop processing of valid input transactions if any one of the transactions submitted contains an error
    f. **AQS User Id**: type the AQS user id for the individual at the agency having authority to submit data to AQS over the Exchange Network. This user id must be granted access to perform submissions by EPA
    g. **Author**, type the name of the developer of the data service.
    h. **Execute Timeout**: leave blank if a stored procedure name is not supplied in the following argument.
    i. **Execute Timeout (in seconds)**: The number of seconds to wait for the Extract Stored Procedure to finish before cancelling the database transaction. Used in conjunction with the Extract Stored Procedure Name parameter below.
    j. **Extract Stored Procedure Name**: set to the stored procedure name to execute prior to reading the staging table data into Xml. Leave blank if you do not want to execute a stored procedure as part of the service execution.
    k. **Organization**: type the name of the organization that is providing submissions created from the data service.
    l. **Sender Address**: type the email address(s) that will receive a submit notification email. Note that this notification will be in addition to emails automatically sent to the AQS user id specified in step 13 above, as obtained from the address defined in the AQS Profile record for that user.
8. Data Sources:
    a. Data Source: Select the data source that is configured for the database connection to the AQS staging tables. See the OpenNode2 Administration guide for more information on creating and testing Data Sources.
9. Click the **Save** button to save the service.

GetAQSStatusAndDownloadReports Service

The *GetAQSStatusAndDownloadReports* service will access the EPA CDX Node and will update the status of all transactions in the OpenNode2 metadata database associated with the *AQSExtractAndSubmission* service that do not have a status of either "Completed" or "Failed".

The service may be configured to send notification emails for each transaction that changes status from an incomplete state (any status other than "Completed" or "Failed") to a completed state ("Completed", "Failed").

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



3. In the **Name** field, enter *GetAQSStatusAndDownloadReports*.
4. Select the implementer "GetAQSStatusAndDownloadReports" from the **Implementer** drop-down menu.
5. From the **Type** drop-down menu, select how you wish to make the services available. The only option available will be obtained by the Node Admin from the plugin itself and will be *Task*.
6. Enable the service by checking the **Active** checkbox.
7. When the implementer is selected, several service parameter arguments will appear. The Node Admin will obtain these properties directly from the loaded AQS plugin. These parameters are used to customize the operation of the data service.
8. Arguments:
    a. **Attach Documents to Notification Emails**, type the desired value ("True" or "False"). This will cause the service to attach any downloaded processing reports from EPA CDX to the emails that are submitted to the designated recipients. Note that setting this value to "True" may cause email payloads to become very large depending on the recent transaction activity.

b. **Max Check Status Days:** type the number of days in arrears for which the service should attempt to determine the status of submitted transactions. The default value if not specified is two days.

c. **Notification Emails:** type a semi-colon delimited string of valid email addresses to which the downloaded processing status and associated documents should be sent.

9. Click the **Save** button to save the service.

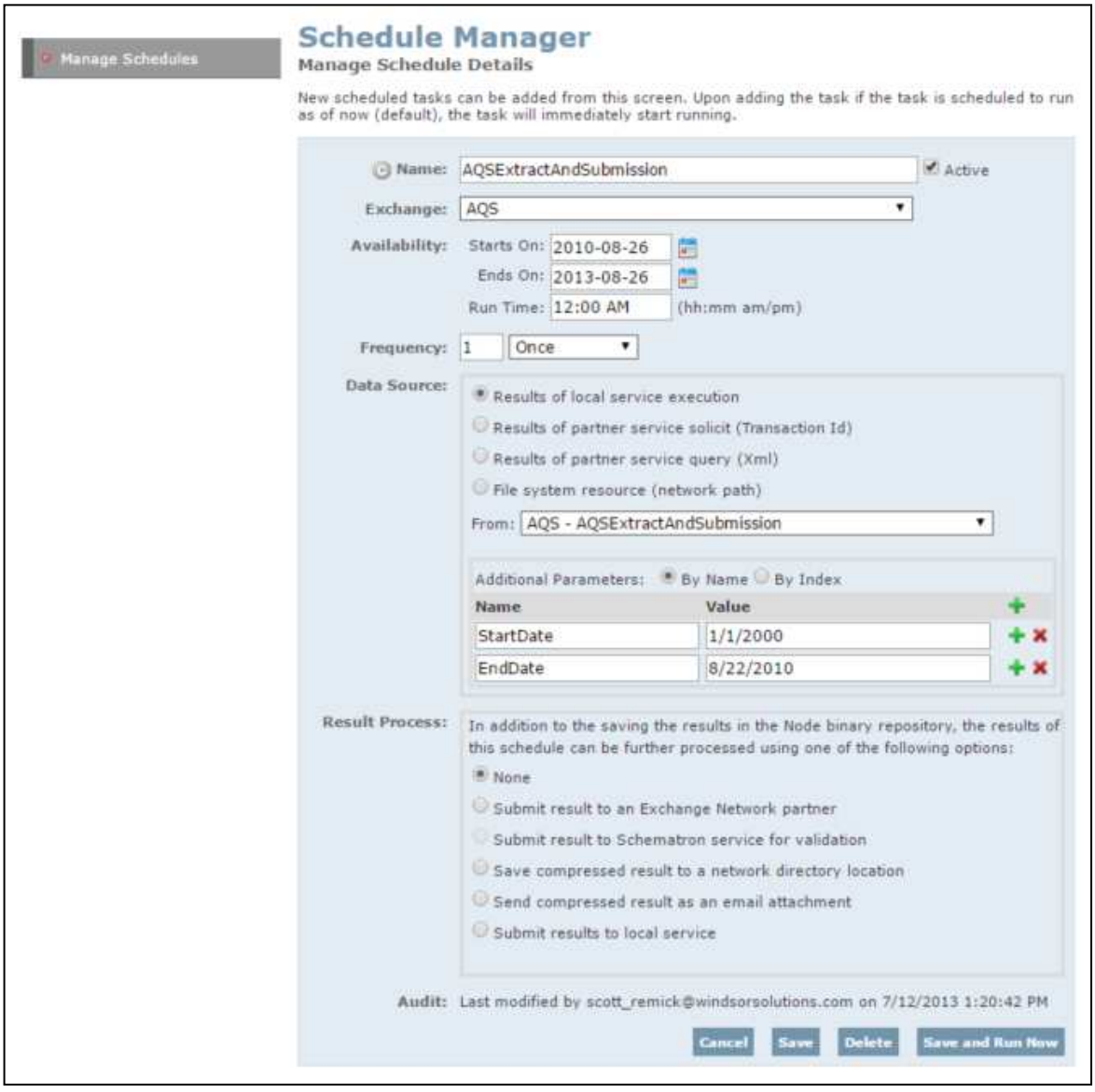
Define Data Exchange Schedules

Scheduled jobs can be configured to perform automated and manual tasks, such as submitting data to external partners or processing received files.

The AQS data exchange contains two schedules; one required to initiate the import of raw data files into the source AQS database, and one to perform the extract of data and submission of data from the staging tables to the EPA.

Create AQSExtractAndSubmission Schedule

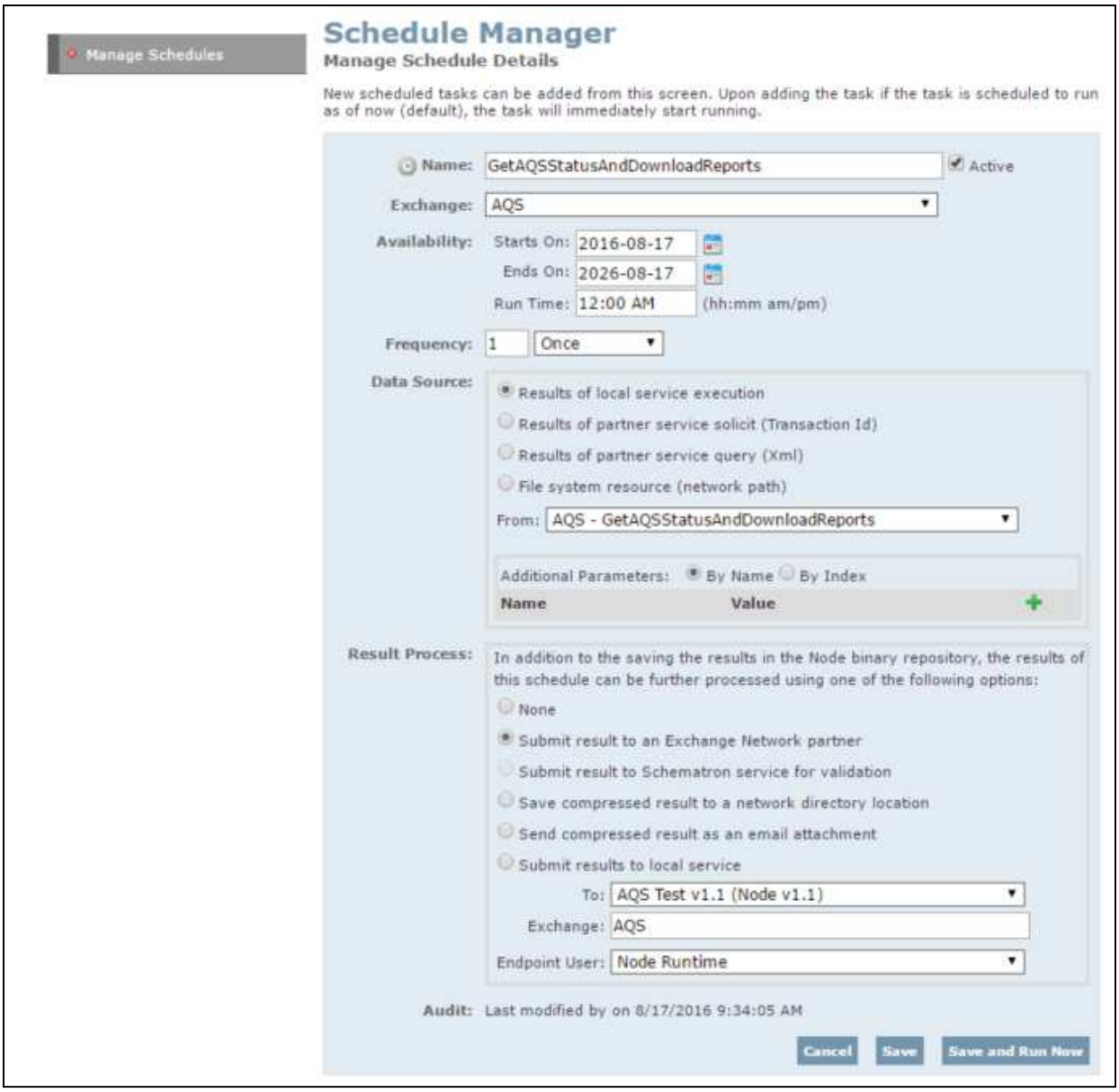
1. From the **Schedules** tab, click the **Add Schedule** button. The Schedule Manager screen will display as follows:



2. Type "AQSExtractAndSubmission" in the **Name** field.
3. Enable the schedule by clicking the **Active** checkbox.
4. Select "AQS" from the **Exchange** dropdown list.
5. This is a manual flow, and so the **Availability** and **Frequency** can be ignored. Although to "save and now", the frequency "once" should be selected.
6. In the **Data Source** area, check the radio button labeled **Results of local service execution**.
7. In the **Data Source** "from" dropdown box, select the value "AQS - AQSExtractAndSubmission."
8. In the **Additional Parameters** area, select "By Name" radio button.
    a. Enter "StartDate" in the first **Name** column, and the start date in which the data flow should extract for submission. In the **Value** field, for example, "4/1/2010". This value can be changed by the submitter based on the date range of submission desired.
    b. Enter "EndDate" in the first **Name** column, and the end date in which the data flow should extract for submission. In the **Value** field, for example, "5/31/2010". This value can be changed by the submitter based on the date range of submission desired.
9. In the **Result Process** area, check the "Submit result to an Exchange Network partner." A new **To**: dropdown will appear with a list of valid parent end points. In this case select "AQS EPA Endpoint Test (or Prod)." In the **Exchange** field, type in **"**AQS". The endpoint can be changed by the user depending during each submission.
10. Click the **Save** button to save the schedule. (Note: to run immediately, click the **Save and Run Now** button)

Create GetAQSStatusAndDownloadReports Schedule

1. From the **Schedules** tab, click the **Add Schedule** button. The Schedule Manager screen will display as follows:



2. Type "GetAQSStatusAndDownloadReports" in the **Name** field.
3. Enable the schedule by clicking the **Active** checkbox.
4. Select "AQS" from the **Exchange** dropdown list.
5. This is a manual flow, and so the **Availability** and **Frequency** can be ignored. Although to "save and now", the frequency "once" should be selected.

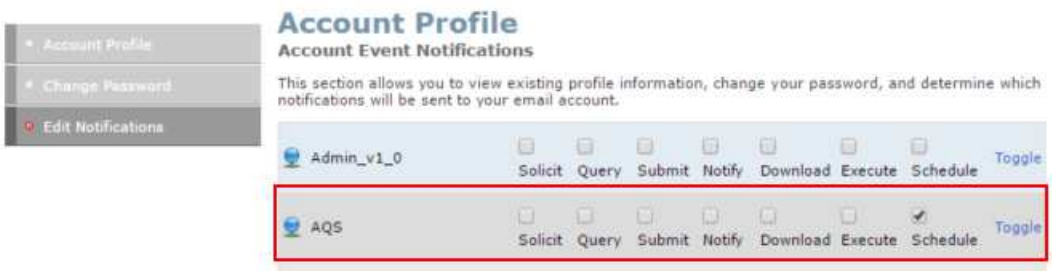
6. In the **Data Source** area, check the radio button labeled **Results of local service execution**.
7. In the **Data Source** "from" dropdown box, select the value "AQS - GetAQSStatusAndDownloadReports."
8. In the **Result Process** area, check the "Submit result to an Exchange Network partner." A new **To**: dropdown will appear with a list of valid parent end points. In this case select "AQS EPA Endpoint Test (or Prod)." In the **Exchange** field, type in **"**AQS".
9. Click the **Save** button to save the schedule. (Note: to run immediately, click the **Save and Run Now** button)

Set up Email Notifications

Email notifications can be sent to a NAAS user account email address when certain Data Services are executed by the Node. You may request that you are notified in the event of a particular data service being queried, solicited, downloaded, submitted, executed (for schedules), or notified. This is a not a necessary step when creating the AQS data flow, but it is highly recommended as a way to ensure successfully transmissions and aid in the troubleshooting process.

To set up an email notification:

1. Click **Profile** tab on the top navigation bar
2. Click **Edit Notifications** on the left navigation bar
3. A screen will appear with all data flows and the account event notifications as follows:



4. Selecting the *Schedule* box will provide email notifications for all of the AQS data services.
5. Click the **Save** button to save the changes to the notification settings and return to the Account Profile

Please see the Node Administration Guide for more information on setting up notifications.

Monitor Flow Activity

The OpenNode2 will track all AQS 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 – Header Document Settings

Element Name	Required	AQS Use	OpenNode2 Implementation
AuthorName: Originator of the document. This should be the name of a person or a network node ID if the document is automatically generated.	No	Not Used by AQS.	Populate from Service Parameter. Optional.
OrganizationName: The organization to which the author belongs. It may be a state name, an organization name or a company name. For submissions to the CDX node, this should be the name of the organization.	No	Not Used by AQS.	Populate from Service Parameter. Optional.
DocumentTitle: Title of the document.	No	Not Used by AQS.	Set to "AQS Data" by the plugin.
CreationDateTime: This is a timestamp that marks when the document, including payloads and header part, was created.	No	Not Used by AQS.	Insert current timestamp from node server.
Keywords: Words that best describe the payload. Multiple keywords should be separated by commas. This is for transaction categorization and searching.	No	Not Used by AQS.	(null)
Comment: Additional comments for processors.	No	Not Used by AQS.	(null)
DataFlowName: The name of the data flow associated with the payload. It could be the name of the data source for Query results.	Yes	AQS	Set to "AQS" by the plugin.
DataServiceName: Name of a data service that generated the document. This is the name of the procedure that was used to initiate the creation of the payload. This would apply only for Query and Solicit and would not be applicable for Download and Submit.	No	N/A	(null)
SenderContact: The sender's additional contact	No	Not Used by AQS.	(null)

Element Name	Required	AQS Use	OpenNode2 Implementation
information. It could contain sender's electronic address and/or telephone numbers where the author can be reached.			
ApplicationUserIdentifier: The user ID for the backend system if it is different from the NAAS user ID.	Yes	AQS User-ID	Populate from Service Parameter. Required.
SenderAddress: A well-formed URI where result/report can be sent. Currently the Network will make use of the Notification mechanism at the Document Level as described in the Protocol and Specification. Note that this could contain multiple addresses, including that of the submitter and/or other technical people related to contents of the payload.	No	Email address, in addition to that stored for the AQS user in the AQS Profile, that will receive the submit notification email.	Populate from Service Parameter. Optional.
Property Begin Tag: Other properties of the document (use name value pairs). This is an extension mechanism to cover any other elements that are not defined in the specification.	Yes	<hdr:Property>	
Property: AQS.Screening Group	Yes	<hdr:PropertyName> AQS.ScreeningGroup </hdr:PropertyName> <hdr:PropertyValue> The AQS screening group that owns the monitors related to the submission. </hdr:PropertyValue>	Populate from Service Parameter. Required.
Property: AQS.FinalProcessingStep Last step of the AQS Load process that will be performed on the user's data.	No	<hdr:PropertyName> AQS.FinalProcessingStep </hdr:PropertyName> <hdr:PropertyValue>	Populate from Service Parameter. Optional.

Element Name	Required	AQS Use	OpenNode2 Implementation
		Member of the set: {“Stage”,” Load”, “Post”}. </hdr:PropertyValue>	
Property: AQS.StopOnError Option to tell AQS whether or not to stop processing of valid input transactions if any one of the transactions submitted contains an error.	No	<hdr:PropertyName> AQS.StopOnError </hdr:PropertyName> <hdr:PropertyValue> Member of the set: {“Yes”, “No”}. </hdr:PropertyValue>	Populate from Service Parameter. Optional.
Property: AQS. PayloadType	Yes	<hdr:PropertyName> AQS.PayloadType </hdr:PropertyName> <hdr:PropertyValue> Member of the set: {“XML”, “FLAT”,”CARD”}. </hdr:PropertyValue>	Set to “XML” by the plugin.
Property: AQS.SchemaVersion	Yes if AQS.FileType = “XML”	<hdr:PropertyName> AQS.SchemaVersion </hdr:PropertyName> <hdr:PropertyValue> The schema version	Populate from Service Parameter. Required.
Property End Tag	Yes	</hdr:Property>	
Signature: An XML signature associated with the document (Use http://www.w3.org/2000/09/xmldsig#)	No	Not Used by AQS.	(null)
id: A unique identifier for the document. This is an	Yes	User name for input file –	GUID generated by OpenNode2 and set

Element Name	Required	AQS Use	OpenNode2 Implementation
attribute of ExchangeNetworkDocument and provides a unique Id for each document in the payload.		This is the name that will show up on the AQS Batch form for the submitted document.	by the plugin.

Appendix B: Staging Table Block Diagram