WQX Web Application Programming Interface (API)

- Introduction
- Access to the WQX Web API
- Formats

Introduction

EPA's Water Quality eXchange team, the Chesapeake Bay Program, and EPA Contractor Gold Systems have developed, tested, and utilized a new way to submit data to WQX and the Water Quality Portal. Until very recently the only two methods to publish your data via WQX were to set up an automated node to node communication via EPA's Exchange Network or to manually upload your data using the WQX Web user interface. The Chesapeake Bay Program recently began publishing their data using automated API services through WQX Web. The new WQX Web API services allow a data submitter to automate data submissions like a node but through WQX Web. The API is intended to provide programmatic access to WQX Web data submission functions and procedures. The intended audience is programmers who are familiar with the concepts and techniques of WQX data submissions and Web Services. This project is innovative because maintenance on the data mapping is managed within the WQX Web. The interface makes maintaining data translation rules easy and user friendly. WQX Web maintains compatibility with new business rules, elements and xml schema validation.

Access to the WQX Web API

The API is available to registered users within the WQX Web application. Before using the API, users must obtain a unique 88-character "Private Encryption Key": associated with a registered WQX Web user account.

All API access is through an authenticated Uniform Resource Identifier (URI). Data is submitted by sending an HTTP GET/POST to the URI with appropriate parameters supplied. The minimum parameters for every request include the UserID, Timestamp, and Signature, which includes the name of the method being invoked.

Program Language

The API may be used with any language capable of issuing HTTPS requests with an HMAC-SHA256 encryption algorithm (Java, PHP, Perl, Python, C, etc.

Data Return Format

The API returns data in one formats: JSON.

Documentation for the WQX Web API services can be referenced via WQX Web Help Menus. Example Code is also available via both WQX Web Help Menu and STORET FTP Site URL: ftp://ftp.epa.gov/storet/wqx/api/Example_code.txt

Where to find the "Private Encryption Key": (WQX Web)

WQX Web API or RESTful web services for WQX Web are designed and used to submit water quality data to WQX.

The basic process flow is as follows:

1. Upload a file
2. Request that WQX Web start importing the uploaded file
3. Get the status for that dataset, periodically, to confirm whether it imported without any errors
4. If the final status is "Imported", then ask WQX Web to submit that dataset to CDX. During that process, the data from your flat file or spreadsheet will be converted to XML (via an import configuration that you previously created) and submitted to CDX. Alternatively, you can request that WQX Web automatically export and submit your dataset to CDX upon completion of the import. It's important to understand that data is not final/permanent until it has been submitted to CDX and has received a final status of "Completed at CDX"
5. If a dataset receives a status of Failed, you can get a list of documents relating to the dataset, and download the "Import Log.xlsx" or "ProcessingReport.zip" (among other files) to review the issues with your import file or the submission to CDX.

RESTful Web Service Methods:

- **Upload:** upload a file to the web server (to be imported).
  - Action POST method
- **StartImport:** start importing a file that was previously uploaded
  - Action GET method
- **StartXmlExport:** start creating the XML submission file (for CDX)
  - Action GET method
- **SubmitDatasetToCdx:** submit a dataset to CDX
  - Action GET method
- **SubmitFileToCdx:** submit a previously uploaded WQX XML file to CDX
  - Action GET method
- **GetStatus:** get the status for a dataset. To avoid undue burden on the server, it is recommended that you not call this service more often than every 10 seconds. For large imports (longer than 20 minutes), calling this service periodically will guarantee that the server will not shutdown before the import completes.
  - Action GET method
- **GetDocumentList:** get the list of available documents for a dataset
  - Action GET method

Each web service call will include a header with the following items:

1. X-UserID
   - The caller's "User ID" to determine rights to private data
2. X-Stamp
   - Timestamp when the caller made the request. This must be UTC Time (i.e. Greenwich Mean Time), so you must convert from your local time zone.
   - Format is (mm/dd/yyyy hh:mi:ss AM)
3. X-Signature
   - Signature is made up of the following pieces of information (concatenated together as a single string value):
     - User ID
     - Timestamp
     - URI
     - Request Method
   - Signature is then encrypted using the HMAC-SHA256 encryption algorithm and placed in the header element.