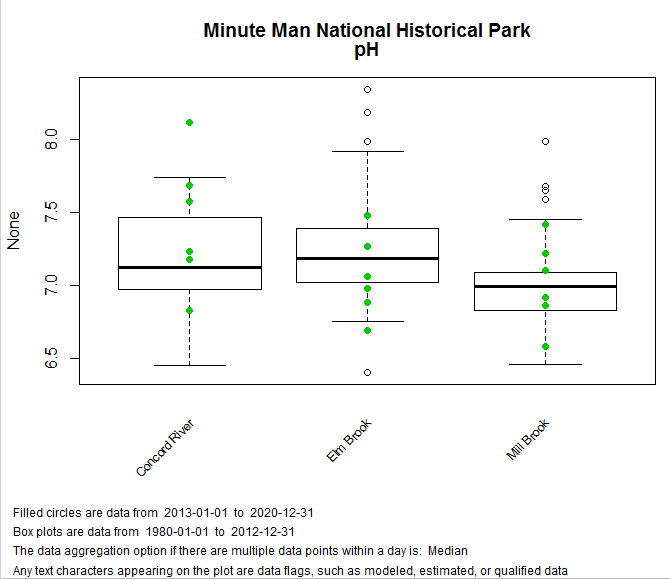
|  |  |  |
| --- | --- | --- |
|  | | |
| **National Park Service**  **U.S. Department of the Interior**  **Natural Resource Stewardship and Science** |  |  |

NOTE: There may be revised processes and documentation available.

NPS I&M Standard Operating Procedure: Data Visualization – Creating and Submitting Visualizer Packages



Suggested Citation: National Park Service. 2015. NPS I&M Standard Operating Procedure: Data Visualization – Creating and Submitting Visualizer Packages. Version 2015-06-29. National Park Service, Natural Resource Stewardship and Science. Fort Collins, Colorado.

Contents

[Overview 3](#_Toc423343074)

[Software Requirements 4](#_Toc423343075)

[Source Data Preparation 4](#_Toc423343076)

[Common View Data Template (CVDT) 4](#_Toc423343077)

[Quality Control 4](#_Toc423343078)

[Handling Provisional or Sensitive Data 5](#_Toc423343079)

[Posting CVDTs to Data Store 5](#_Toc423343080)

[Source Data Access 5](#_Toc423343081)

[Data Services 5](#_Toc423343082)

[Constructing a Data Request 6](#_Toc423343083)

[Connecting to Services 6](#_Toc423343084)

[Creating Data Frames 6](#_Toc423343085)

[Step-by-Step Example for Accessing Data 6](#_Toc423343086)

[Writing and Submitting Analysis Code 7](#_Toc423343087)

[Step 1: Write Graphics Scripts in R 8](#_Toc423343088)

[Step 2: Plot/Graphic Validation 8](#_Toc423343089)

[Step 3: Create a Visualization Package 8](#_Toc423343090)

[Step 4: Submit a Visualization Package 9](#_Toc423343091)

[Borrow an Existing Package 9](#_Toc423343092)

[Frequently Asked Questions 9](#_Toc423343093)

[What are data services? 9](#_Toc423343094)

[How do I Use OData Services? 9](#_Toc423343095)

[How does the Data Mart work? 10](#_Toc423343096)

[Where is the Visualization Collaboration Website? 10](#_Toc423343097)

[References 10](#_Toc423343098)

[Appendices 11](#_Toc423343099)

[Appendix 1: Known issues 11](#_Toc423343100)

[Appendix 2: Data service request examples 11](#_Toc423343101)

[Appendix 3: Example R Scripts 12](#_Toc423343102)

[Appendix 4: Recommended R Coding Practices 13](#_Toc423343103)

[Appendix 5: Data Service Entities (Tables) 15](#_Toc423343104)

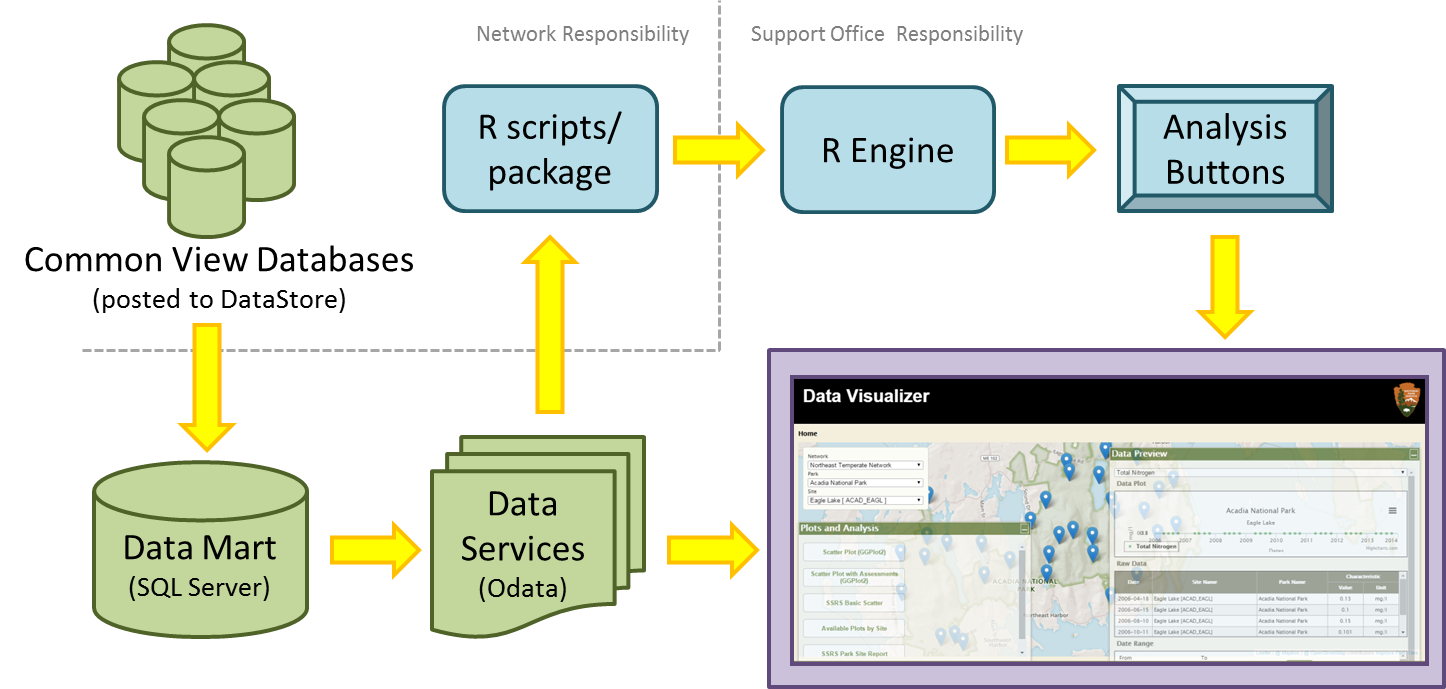
|  |  |  |
| --- | --- | --- |
| **Version History** | | |
| Version | Update Date | Changes |
| 20150629 | 20150629 | Revised draft based on comments from two networks |
| 20150603 | 20150603 | Revised draft; streamlined sections, updated code examples |
| 20150511 | 20150511 | First draft for review/test |
| 20150424 | 20150424 | Initial draft |

Overview

The I&M Data Visualizer was designed to provide an online, public-facing, graphical user-interface for natural resource data collected and managed by the National Park Service (NPS) Inventory and Monitoring Program (I&M). The Visualizer’s functionality is dependent on the interplay of three major components: source data, the analysis code, and the user interface.

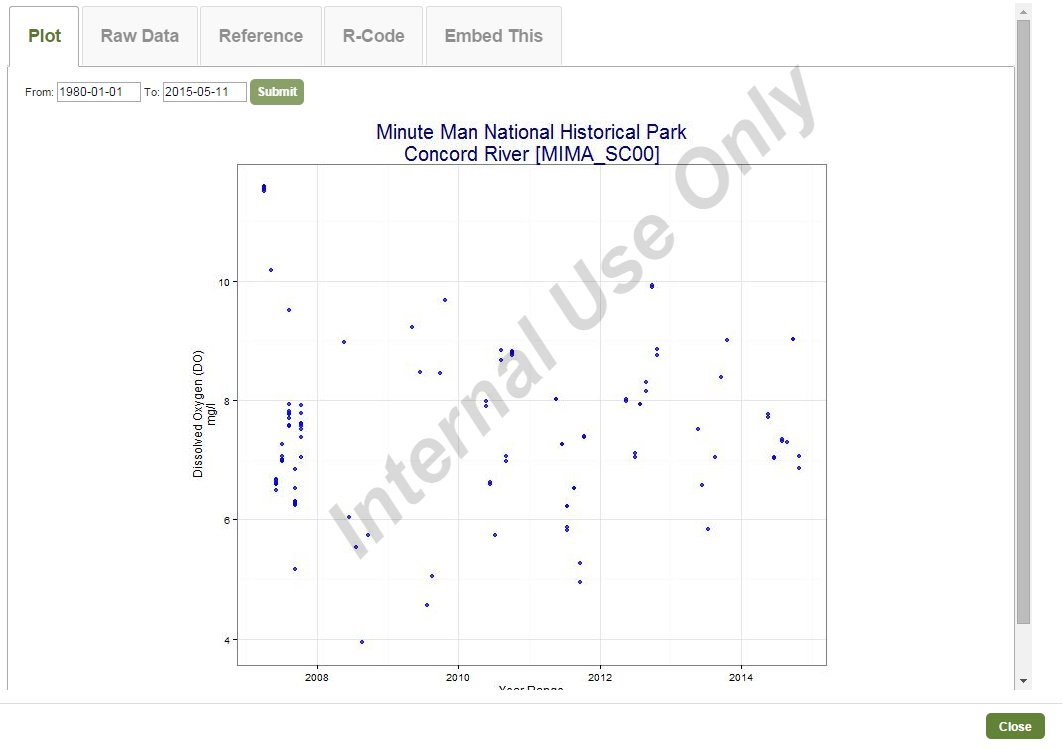
* Source data: Data used in the Data Visualizer are submitted using a standardized database format (common view) and then automatically converted to a data service. Quality control, quality assurance, and certification are the responsibility of the Network submitting the data.
* Analysis code (visualizer package): The code used to plot and/or conduct statistical interpretation of the data is authored by the Networks. “R” is the programming language of choice. These analysis plot scripts are documented then shared as analysis buttons in the NPS I&M Data Visualizer.
* User interface: A hybrid of HTML and Javascript, this online interface is the platform upon which data and their analyses are presented to the user. The user interface recognizes and adjusts output for users both internal and external to the DOI network.

This Standard Operating Procedure provides detailed instructions for the process steps highlighted in blue in Figure 1. More information, instructions, and code samples for the I&M Data Visualization Project can be found here: <http://imtest/im/reports/plots/Documentation/>



**Figure 1:** I&M Data Visualizer work flow - from source data steps (green) to analysis code steps (blue) to presentation in the user interface (violet).

Clicking an analysis button opens the Analysis Summary popup (Figure 2) that contains tabs for the analysis graphic, the raw data used in the graphic, detailed reference and citation information, the R code used to create the graphic, and HTML code to embed the popup in a separate web page. The Visualizer consumes Network-provided R scripts to make the graphic, and then adds the additional tab contents by consuming the data services and Network-provided descriptions. Analysis summary popups are created from Visualizer Packages. Note these are not R packages (hence the use of the term visualizer package.



**Figure 2:** Analysis summary popup example

Software Requirements

The Data Visualizer relies on R (<http://www.r-project.org/>) scripts to produce its analysis graphics. Networks are free to use their R integrated development environment (IDE) of choice (examples: [R Studio](http://www.rstudio.com/), [Tinn-R](http://sourceforge.net/projects/tinn-r/), etc.) to author and test R scripts that make use of data from the I&M data services.

Source Data Preparation

Common View Data Template (CVDT)

The Common View Database Template (CVDT) was developed to assist networks with converting their data to a consistent format for data visualization. The CVDT standardizes tables, data flags, and lookup values and applies best practices for optimizing visualization. Common View Database Templates will be available for the Water Quality Vital Sign handled by the Visualizer and was designed to accommodate the storage of as many local data schemas as possible. Data submitted for use in the Data Visualizer must adhere to the CVDT if it wishes to take advantage of existing analysis code and packages. The Guide to Implementing the Common View and template databases for available Vital Signs can be downloaded from <https://irma.nps.gov/App/Reference/Profile/2219159>.

Other Vital Signs may use a CVDT approach or will be based on existing data structures (e.g. the I&M Climate Station database).Quality Control

The accuracy and quality of the data presented by the Data Visualizer is only as good as the source data provided by the Networks. The CVDT encapsulates many (but not all) of the data preparation and quality checks necessary to ensures a level of consistency on which visualization depends. Quality checks prior to loading data into the CVDT help minimize nulls, incomplete measures, and errors. Before loading data into the CVDT, look at records or run queries to check for the following issues:

* Typos in network, park or site names
* Null location values (use 0 if a location needs to be obfuscated) or positive longitude values
* Unnecessary data replicates (continuous sensors)
* Correct values for assessment points (thresholds)

Handling Provisional or Sensitive Data

The CVDT is designed to accommodate both sensitive and non-sensitive data as well as both provisional and certified data. Flagging the data at the record level allows regulation of access to these data through permissions that recognize users as internal and external to the NPS network. It is the responsibility of the Network providing the data to make sure that sufficient attention is paid to identifying and marking records that are sensitive or provisional during the process of populating the CVDT. See the Guide to Implementing the Common View cited in the section above for more information.

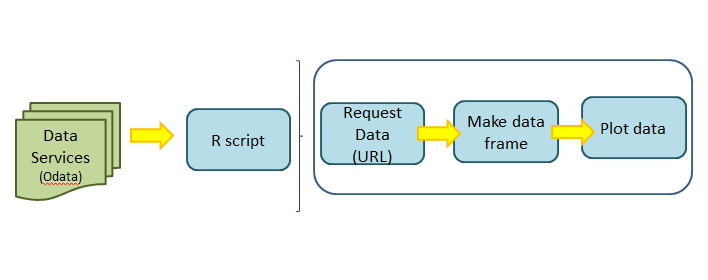
Posting CVDTs to Data Store

Refer to the SOP for uploading a CVDT database on the Data Visualization SharePoint site: <http://inpniscvsp05:39904//imdev/projects/Pages/NDV.aspx>.

Once the CVDT databases are loaded to the DataStore and the Visualization team is notified, they are automatically harvested into the visualization Data Mart. The Data Mart is the database on top of which data services are created. All data access for visualization occurs via these services that query the Data Mart database.

Source Data Access

The Visualizer application relies on data services that vend data from a data mart database (Figure 1 above). The basic workflow is shown in Figure 3. See a step-by-step example at the end of this section.



**Figure 3:** Accessing and using data services

Data Services

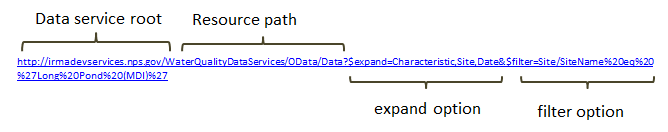
Data services provide data on demand via a URL web request (see the FAQ section for more details). In the case of the Visualizer, data services are query-able URLs that vend data from the Data Mart to any web-enabled client (web apps, R scripts, Python, Excel, etc.). Analysis (e.g. local QA/QC and data exploration) and visualization (e.g. graphing and mapping) tasks are accomplished by processing data retrieved through these services. This SOP emphasizes using R to submit, and process the results of structured data queries. See Appendix 2 for detailed examples of data service requests.

Constructing a Data Request

Requests to the Data Mart are constructed using the OData syntax. OData requests allow users great flexibility in how they customize/subset their data extractions. OData request URLs start with a data service root path that identifies the server, followed by a resource path that indicates what service to use (Figure 4). After the data service root and resource path, the request contains 3 options that control which data entities to query and return:

* expand: related entities (‘tables’) to return
* filter: query to subset data to return
* select (*optional*): properties (‘fields’) to return (simplifies and minimizes the response)

Once you understand the syntax, writing a good request URL only requires knowledge of what tables and fields are available in the source data you are querying.



**Figure 4:** Simple request URL – does not include the ‘select’ option, so all data specified by the filter will be returned.

See the ‘How do I use data services’ frequently asked question below for details on the content of the data service. Examples of OData service requests specific to the CVDT can be found at the end of this section and in Appendix 2 and in the example script files.

Connecting to Services

Data service requests can be made internally (from within the NPS network) or externally (from the public internet). Requests originating from within the NPS network have access to all data in the Data Mart. Requests originating from outside the NPS network will only return records in the Data Mart flagged as being non-sensitive and non-provisional.

The example R scripts included with this SOP have a data access section that creates and submits query URLs to data services. Additionally, the Data\_Access\_Example.R script (Appendix 3) is an example of data service access. It formats a URL, then submits it and grabs the response, putting it into a data frame.

If working offline is desired, make a data request to a service, then export the response data locally to a CSV file or spreadsheet. In this scenario, please test your R scripts against the live (online) data service URL(s) prior to submitting the visualization package.

Creating Data Frames

A data service request results in a JSON (formatted text) response. These responses can be converted to R data frames using the [jsonlite](http://cran.r-project.org/web/packages/jsonlite/index.html) R library. Then, these data frames may be manipulated further in R. All of the example scripts use this technique.

Step-by-Step Example for Accessing Data

The example URL requests data for one site and all characteristics:

<http://irmadevservices.nps.gov/WaterQualityDataServices/OData/Data?$expand=Characteristic,Site,Date&$filter=Site/SiteName%20eq%20%27Long%20Pond%20(MDI)%27>

Note: A URL request in R does not recognize spaces and certain characters. Therefore, %20 and %27 values are escape sequences used to indicate a space and a single apostrophe, respectively.

The steps for incorporating the above URL request into your R code are:

1. Create an R script and add jsonlite library to it:

library (jsonlite)

1. Construct URL (*targetURL*) and replace spaces with escape sequences:

targetURL <- "http://irmadevservices.nps.gov/WaterQualityDataServices/OData/Data?$expand=Characteristic,Site,Date&$filter=Site/SiteName eq 'Long Pond (MDI)'"

targetURL <- gsub(" ", "%20", targetURL)

1. Make data request using the jsonlite *fromJSON* method and create data frame (*plotdf*):

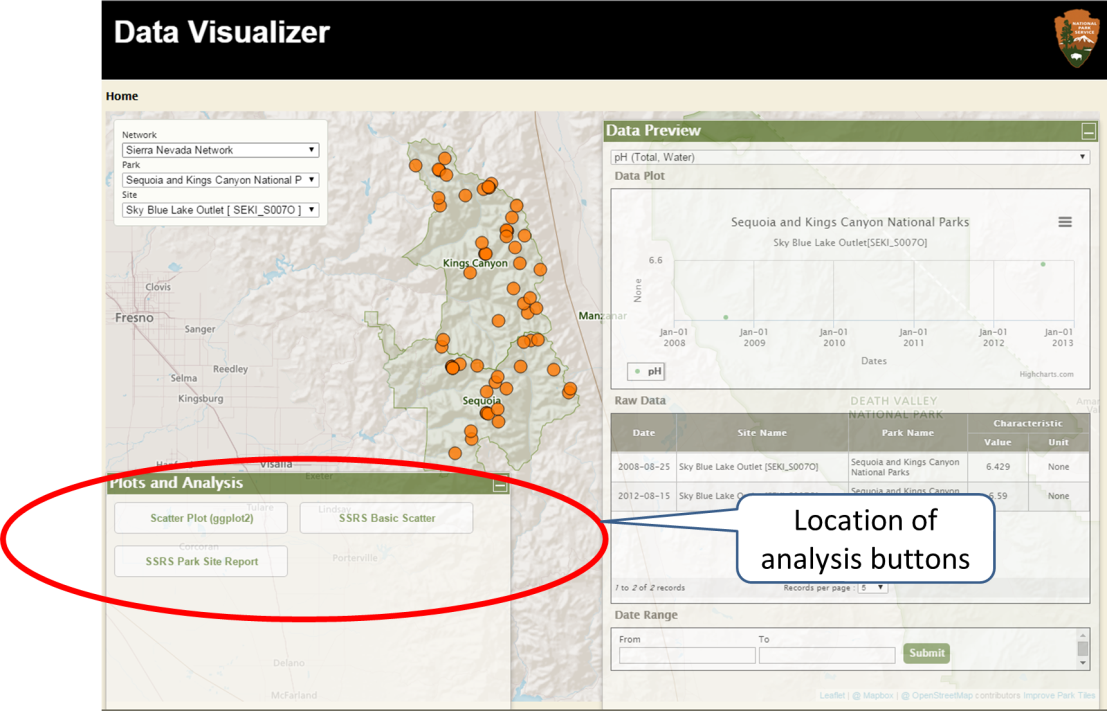
plotdf <- fromJSON(targetURL, , flatten = TRUE)[[ 2 ]]

Note: flatten causes the responses’ nested data frames to be flattened into a 2 dimensional format. [[2]] gets the second item of the response (the data). The first item is the URL used to make the request.

1. Manipulate the data frame as needed to create the desired graph.

The code examples included with this SOP use these same steps. See the code comments for more detail, particularly for data frame creation and use. Writing and Submitting Analysis Code

Once you have mastered the ability to retrieve data from the data services using URLs, you will want to start processing and analyzing those data. This processing step can be run either external to the visualizer (for your own use/consumption), or written to be run when an analysis button is clicked in the Visualizer (Figure 3). Analysis code (R scripts) and a visualization package (not to be confused with an R package) are the fundamental components of an analysis button in the visualization application. When a visualization package is submitted, its script(s) and documentation are translated into an analysis summary popup which is linked to a new or existing analysis button.



**Figure 5:** Location of the buttons on the NPS Data Visualizer that trigger the running of custom data analysis code.

Step 1: Write Graphics Scripts in R

Plots (graphics) are produced using R scripts that query data services. The examples provided (in zip file on Data Store) are simple and do not contain R user-defined functions. Eventually, it is expected that a set of data functions (i.e. a vital sign R package) will evolve and be included.

Scripts have three operational sections: data query, data frame manipulation, and plot generation.

* Data query: create request URLs to get data from service(s)
* Data frame manipulation: reformat the data response as needed for to generate the plot
* Plot generation: use the data frame(s) to create plot desired

The data query determines which sites and characteristics are used by the plot. Data frame reformatting, summarization, and column re-naming may be needed to drive the desired plot.

Plots should include a title, axes, and descriptive legend that indicate the park(s), site(s), characteristics, date ranges, and where applicable, data flags. One of the example scripts provided shows how this flag is used to annotate the plot graphic. If applicable, a descriptive caption should be included. Another example script includes a caption example.

Externally vended (i.e. public) data must be filtered appropriately. This can happen when data are requested (i.e. in the request URL filter) or by sub-setting the response data frame. Step 2: Plot/Graphic Validation

R plot scripts should be tested directly against the service response before submitting the visualization package. Use your R IDE of choice to test.

Step 3: Create a Visualization Package

A visualization package is a zip file containing the R code for plot generation, plot description, specifications, and citations. Template files are posted along with this SOP.

This zip file gets submitted and the contents are used to create the content for the analysis button’s summary popup.

Submit the following items:

\*.R: R script file used to produce analysis plot (see Appendix 4 - Recommended R Coding Practices)

Plot Specifications (text file – see template in zip file):

List of R libraries used

List of Variables defined in R script

Networks/Parks targeted

List of characteristic(s) used

Plot caption/legend: note if provisional data are included

Optional: Shiny interactive options needed

Plot Citations/References (text file – see template in zip file):

This content appears in the Reference tab of the Analysis Button.

Plot Description: text description of plot content with summary detail on scientific and

statistical attributes (enough to enable a user to understand the concept being portrayed)

Link to Data Store collection or profiles:

links to a Collection containing the SOPs for data collection, processing, and

QA/QC, and other relevant citations

or,

links to individual reference profiles for the same

Other/external links (as needed)

Step 4: Submit a Visualization Package

Once the zipped visualization package file is complete, create a package upload request in the Visualization Issue Tracker site (NPS intranet only).

1. If needed, connect via VPN to get on the NPS network.
2. Open the Issue Tracker: <http://inpniscvsp05:39904/imdev/projects/Pages/NDV.aspx>
3. Create a new issue (scroll to bottom of page and click ‘Add new item’). Use these formats for the Plot Spec and Description fields:

Plot Specs: Visualization Package Submission: <*NetworkCode*>, <*ParkCode(s)*>

Description: <*Analysis Button Label*> <*package zip file name*>

(Example: Scatter Plot with Assessment Points NETN\_ScatterPlot\_20150410.zip)

1. Select a desired Due Date and use the Comments field for special instructions.
2. Click Attach File from the popup ribbon and navigate to the visualization package zip file. Upload it.

Click Save to submit the package. You will receive notification when your visualization package has been deployed as an analysis button to the testing server.

Borrow an Existing Package

Sample packages will be posted on the Visualization Collaboration Website (<http://imtest/im/reports/plots/Documentation/>). Download a package, unzip it, then modify the data service URLs and R parameters in the R script(s). Also, change the plot description and citations are needed. Create a new visualization package and submit it.

Frequently Asked Questions

What are data services?

Data service is a generic term describing the vending of data via a URL. Many technologies can be used to create data services. In most cases, the technique involves connecting to a source database or databases, then creating data entities (representations of data tables and/or queries). Then, these data entities are exposed as components of the overall data service. The service allows simple or detailed querying of the data in the entities via URLs (HTTP or HTTPS).

For I&M visualization, database views (queries) are the sources of data tables. Data tables vary by vital sign. For water quality, the tables are: Networks, Parks, Sites, Characteristics, Data (Datum), Date, Time, AssessmentPointsBySiteAndChar, and GeoJSONStations. See Appendix 5 for a relationship diagram of these tables and their fields. Note that the relationship diagram calls the Data table ‘Datum’ – this is because the utility used to create OData services forces names to be plural.

How do I Use OData Services?

The Open Data Protocol (OData) allows data access via Uniform Resource Indicators (URIs) to data using URLs. Background documentation and example syntax can be found here:

<http://www.odata.org/documentation/odata-version-2-0/uri-conventions/>

The NPS I&M data services are vended internally (all data) and externally (non-sensitive and non-provisional data). These services are created using the Microsoft WebAPI framework and run under the IIS (Internet Information Services) web server.

|  |  |
| --- | --- |
| *Availability* | *OData Metadata URL* |
| Internal (all data) | <http://irmadevservices.nps.gov/WaterQualityDataServices/Odata/$metadata> |
| External (non-sensitive and non-proprietary data) | TDB |

Examples of OData service requests specific to the CVDT can be found in Appendix 2.

How does the Data Mart work?

For details, see the Water Quality Data Mart documentation located here: <URL TBD>

Where is the Visualization Collaboration Website?

The I&M data visualization collaboration site is here: <http://imtest/im/reports/plots/Documentation/>

References

Google Code. 2015. Google’s R Style Guide. Available at: <https://google-styleguide.googlecode.com/svn/trunk/Rguide.xml> (Accessed 20150324).

OData.org. 2015. OData – the best way to REST. Available at: <http://www.odata.org/>. (Accessed 20150324).

Wickham, Hadley. 2015. R Style Guide. Available at: <http://adv-r.had.co.nz/Style.html>. (Accessed 20150324).

Appendices

Appendix 1: Known issues

Data availability

Data services pull directly from the data mart database which is populated with the data contained in the common view databases provided by each network.

Additional issues

Details here.

Appendix 2: Data service request examples

These example data service URLs use one network (NETN). For details on refining OData service URLs, see the OData Reference section under Frequently Asked Questions.

All sites and all characteristics for a network:

<http://irmadevservices/WaterQualityDataServices/OData/Data?$expand=Characteristic,Site,Date&$filter=Site/NetworkCode%20eq%20%27NETN%27>

All data for a single site (SiteName): <http://irmadevservices.nps.gov/WaterQualityDataServices/OData/Data?$expand=Characteristic,Site,Date&$filter=Site/SiteName%20eq%20%27Long%20Pond%20(MDI)%27>

All data for a single site (SITE\_ID): <http://irmadevservices.nps.gov/WaterQualityDataServices/OData/Data?$expand=Characteristic,Site,Date&$filter=Site/SITE_ID%20eq%2026>

All data for multiple (specified) sites: [http://irmadevservices.nps.gov/WaterQualityDataServices/OData/Data?$expand=Characteristic,Site,Date&$filter=Site/SiteName%20eq%20%27Long%20Pond%20(MDI)%27 or Site/SiteName%20eq%20%27Duck%20Pond%20Brook%27 or Site/SiteName%20eq%20%27Cadillac%20Stream%27](http://irmadevservices.nps.gov/WaterQualityDataServices/OData/Data?$expand=Characteristic,Site,Date&$filter=Site/SiteName%20eq%20%27Long%20Pond%20(MDI)%27%20or%20Site/SiteName%20eq%20%27Duck%20Pond%20Brook%27%20or%20Site/SiteName%20eq%20%27Cadillac%20Stream%27)

All data for a network and a single characteristic: <http://irmadevservices.nps.gov/WaterQualityDataServices/OData/Data?$expand=Characteristic,Site,Date&$filter=Characteristic/CharacteristicName%20eq%20%27Dissolved%20oxygen%20(DO)%27%20and%20Site/NetworkCode%20eq%20%27NETN%27>

All data for a site for multiple (specified) characteristics:

<http://irmadevservices.nps.gov/WaterQualityDataServices/OData/Data?$expand=Characteristic,Site,Date&$filter=Site/SiteName%20eq%20%27Long%20Pond%20%28MDI%29%27%20and%20%28Characteristic/CharacteristicName%20eq%20%27Dissolved%20oxygen%20%28DO%29%27%20or%20Characteristic/CharacteristicName%20eq%20%27Acid%20Neutralizing%20Capacity%20%28ANC%29%27%29>

Data for a specific site, characteristic, and date range:

<http://irmadevservices.nps.gov/WaterQualityDataServices/OData/Data?$expand=Characteristic,Site,Date&$filter=Characteristic/CharacteristicName%20eq%20%27Dissolved%20oxygen%20%28DO%29%27%20and%20Site/SiteName%20eq%20%27Long%20Pond%20%28MDI%29%27%20and%20%28Date/DATE_ID%20gt%2020100101%20and%20Date/DATE_ID%20lt%2020121231%29>

Assessment Points by Site, Characteristic, and Level:

<http://irmadevservices.nps.gov/WaterQualityDataServices/OData/AssessmentPointsBySiteAndChars?&$filter=SiteCode%20eq%20%27MORR_SC00%27%20and%20CharacteristicKey%20eq%2043%20and%20Level%20eq%201>

Appendix 3: Example R Scripts

Example scripts that produce plots are provided. Please see the zipped file (Visualization\_SOP\_Scripts.zip) containing two standalone example scripts (and templates for plot specifications and description):

* a scatter plot with assessment points (one site and one characteristic):

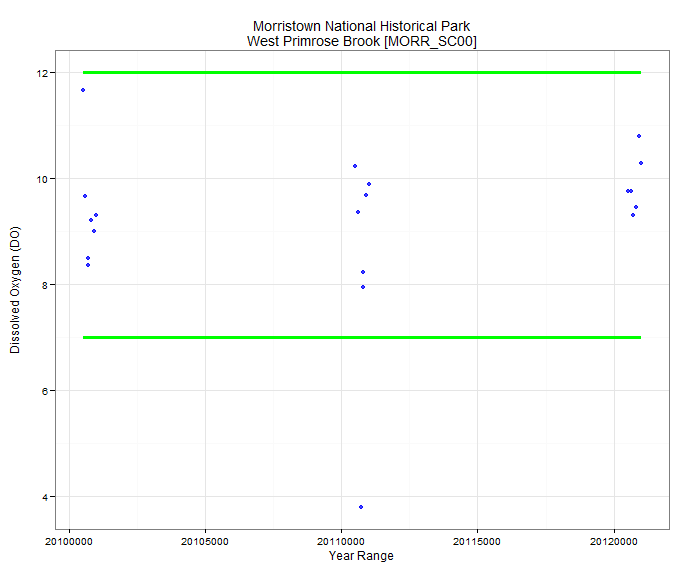
Scatter\_Plot\_with\_AssessmentPoints.R (Figure 4)

* a box plot with scatter plot overlay and caption (multiple sites and one characteristic):

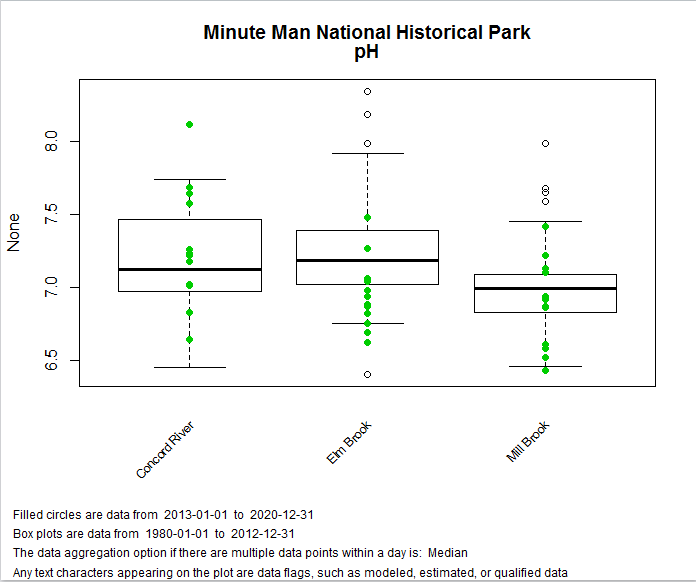
Box\_Plot\_with\_Scatter\_Plot\_Overlay.R (Figure 5)

* example data service query request (one site and all characteristics):

Data\_Access\_Example.R



**Figure 6:** Example graphic produced by Scatter\_Plot\_with\_AssessmentPoints.R



**Figure 7:** Example graphic produced by Box\_Plot\_with\_Scatter\_Plot\_Overlay.R

The scripts generally follow the recommended R coding practices (Appendix 5) and are sequential. With documented parameters, the scripts are written to be consumed directly by the R engine noted in the visualization workflow (Figure 1).

Appendix 4: Recommended R Coding Practices

Standalone R scripts should have three sections: data query, data frame manipulation, and plot generation.

* Data query: create request URLs to get data from service(s)
* Data frame manipulation: reformat the data response as needed for to generate the plot
* Plot generation: use the data frame(s) to create plot desired

Use URL options to specify data to include; this minimizes the content sent over the wire and improves performance.

Make sure scripts include comments.

Do not include:

* Graphic output format (this is hard-coded in the analysis summary popup)
* Nested functions
* S4 code

For code formatting recommendations, see the R Style Guide (Wickam, 2015) and Google’s R Style Guide (Google Code, 2015).

Appendix 5: Data Service Entities (Tables)

