



# **Creating StationXML: Introducing Yasmine and the New Nominal Response Library Web Service**

SAGE/GAGE Community Science Workshop  
Short Course, June 13, 2022  
Pittsburgh, PA

# What is FDSN StationXML?

- StationXML is a standardized XML metadata exchange format for seismic data
- Successor of dataless SEED 2.4
- Developed by the International Federation of Digital Seismographic Networks (FDSN)
- <https://www.fdsn.org/xml/station/>

```
<?xml version="1.0" encoding="UTF-8"?>
<FDSNStationXML xmlns="http://www.fdsn.org/xml/station/1" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:iris="http://www.iris.edu/xml/station/1/iris" xsi:schemaLocation="http://www.fdsn.org/xml/station/1 http://www.fdsn.org/xml/station/fdsn-station-1.1.xsd" schemaVersion="1.1">
  <Source>IRIS-DMC</Source>
  <Sender>IRIS-DMC</Sender>
  <Module>IRIS WEB SERVICE: ph5ws-station | version: 1.1.7</Module>
  <ModuleURI>
    http://service.iris.edu/ph5ws/station/1/query?net=2A&sta=1012&loc=-&cha=DPZ&starttime=2016-04-11T20:31:48&endtime=2016-05-11T17:08:01&level=response&format=xml&nodata=404
  </ModuleURI>
  <Created>2022-05-24T20:04:39.5300</Created>
  <Network code="2A" startDate="2016-01-01T00:00:00.0000" endDate="2016-12-31T23:59:59.9999" iris:PH5ReportNum="17-008" restrictedStatus="open">
    <Description>Large-n Seismic Survey in Oklahoma (LASSO)</Description>
    <Identifier type="DOI">10.7914/SN/2A_2016</Identifier>
    <TotalNumberStations>1829</TotalNumberStations>
    <SelectedNumberStations>1</SelectedNumberStations>
    <Station code="1012" startDate="2016-04-11T20:31:48.0000" endDate="2016-05-11T17:08:01.0000" restrictedStatus="open" iris:alternateNetworkCodes="">
      <Latitude>36.952493</Latitude>
      <Longitude>-97.75067</Longitude>
      <Elevation>364.0</Elevation>
      <Site>
        <Name>1012</Name>
      </Site>
      <CreationDate>2016-04-11T20:31:48.0000</CreationDate>
      <TotalNumberChannels>1</TotalNumberChannels>
      <SelectedNumberChannels>1</SelectedNumberChannels>
      <Channel code="DPZ" locationCode="" startDate="2016-04-11T20:31:48.0000" endDate="2016-05-11T17:08:01.0000" restrictedStatus="open" iris:PH5Array="1001" iris:PH5Component="1" iris:PH5ReceiverId="1012">
        <Latitude>36.952493</Latitude>
        <Longitude>-97.75067</Longitude>
        <Elevation>364.0</Elevation>
        <Depth>0</Depth>
        <Azimuth>0</Azimuth>
        <Dip>90</Dip>
        <SampleRate>5E02</SampleRate>
      </Channel>
    </Station>
  </Network>
</FDSNStationXML>
```

# Levels of FDSN StationXML Detail

XML has a hierarchical structure

Network

→ Station

→→ Channel

→→→ Response

```
<?FDSNStationXML xmlns="http://www.fdsn.org/xml/station/1" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:iris="http://www.iris.edu/xml/station/1/iris" xsi:schemaLocation="http://www.fdsn.org/xml/station/1 http://www.fdsn.org/xml/station/fdsn-station-1.1.xsd" schemaVersion="1.1">
  <Source>IRIS-DMC</Source>
  <Sender>IRIS-DMC</Sender>
  <Module>IRIS WEB SERVICE: ph5ws-station | version: 1.1.7</Module>
  <ModuleURI>
    http://service.iris.edu/ph5ws/station/1/query?net=2A&sta=1012&loc=-&cha=DPZ&starttime=2016-04-11T20:31:48&endtime=2016-05-11T17:08:01&level=response&format=xml&nodata=404
  </ModuleURI>
  <Created>2022-05-24T20:04:39.5300</Created>
  <Network code="2A" startDate="2016-01-01T00:00:00.0000" endDate="2016-12-31T23:59:59.9999" iris:PH5ReportNum="17-008" restrictedStatus="open">
    <Description>Large-n Seismic Survey in Oklahoma (LASSO)</Description>
    <Identifier type="DOI">10.7914/SN/2A_2016</Identifier>
    <TotalNumberStations>1829</TotalNumberStations>
    <SelectedNumberStations>1</SelectedNumberStations>
    <Station code="1012" startDate="2016-04-11T20:31:48.0000" endDate="2016-05-11T17:08:01.0000" restrictedStatus="open" iris:alternateNetworkCodes="">
      <Latitude>36.952493</Latitude>
      <Longitude>-97.75067</Longitude>
      <Elevation>364.0</Elevation>
      <Site>
        <Name>1012</Name>
        <Site>
          <CreationDate>2016-04-11T20:31:48.0000</CreationDate>
          <TotalNumberChannels>1</TotalNumberChannels>
          <SelectedNumberChannels>1</SelectedNumberChannels>
        </Site>
        <Channel code="DPZ" locationCode="" startDate="2016-04-11T20:31:48.0000" endDate="2016-05-11T17:08:01.0000" restrictedStatus="open" iris:PH5Array="1001" iris:PH5Component="1" iris:PH5ReceiverId="1012">
          <Latitude>36.952493</Latitude>
          <Longitude>-97.75067</Longitude>
          <Elevation>364</Elevation>
          <Depth>0</Depth>
          <Azimuth>0</Azimuth>
          <Dip>90</Dip>
          <SampleRate>5E02</SampleRate>
        </Channel>
      </Site>
    </Station>
  </Network>
</FDSNStationXML>
```



# IRIS Nominal Response Library: The NRL

## What is the NRL?

- The NRL is a library of manufacturer's recommended nominal responses that is compiled, maintained, and distributed by IRIS
- Responses are generated from manufacturers documentation or direct communication and checked for
  - ◆ reasonable response curves
  - ◆ correct gain, FIR filter delays, and other attributes
  - ◆ format validity
- Currently contains over 23,000 responses from 35 sensor manufacturers and 29 datalogger manufacturers

## What is new for the NRL?

- Version 2 was released in April 2022!
- Access the Library through the web service:
  - ◆ <https://service.iris.edu/irisws/nrl>
- Access the Library through a new web page:
  - ◆ <https://ds.iris.edu/ds/nrl>
- Responses available as StationXML in addition to SEED RESP format
  - ◆ also available as a response-only subset of FDSN StationXML 1.1 that we are calling StationXML-Response
- New response types: integrated recorders, state-of-health channels, polynomial response descriptions

# More about the StationXML-Response format

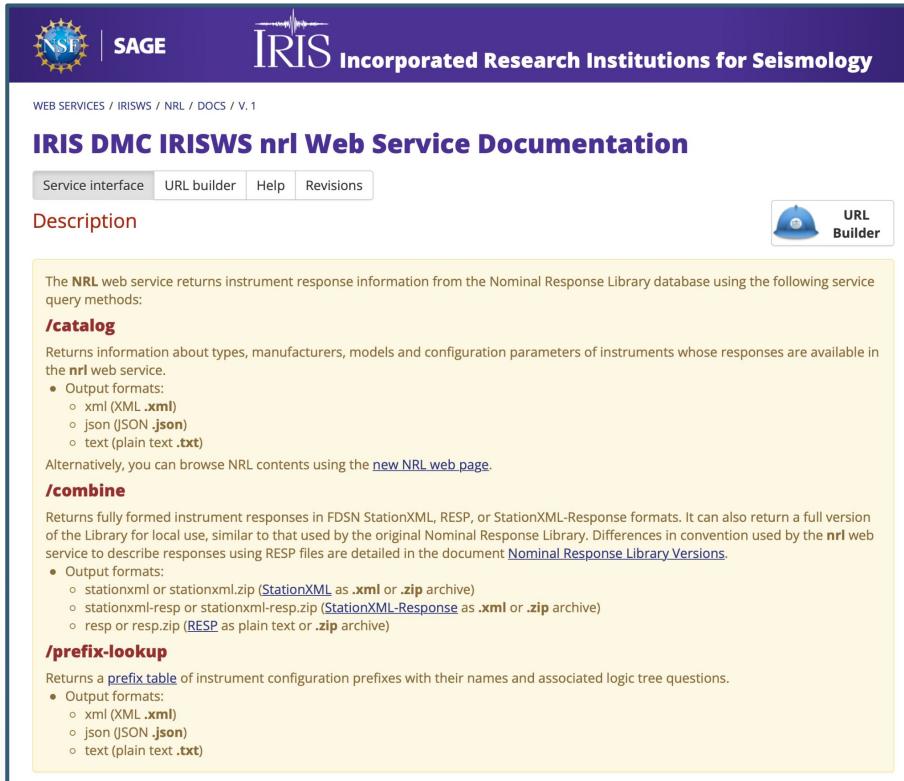
Schema available here:

<https://ds.iris.edu/files/xml/station/fdsn-station-response-1.1.xsd>

Example:

```
<Response xmlns="http://www.fdsn.org/xml/station/1" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.fdsn.org/xml/station/1 http://ds.iris.edu/files/xml/station/fdsn-station-response-1.1.xsd">
  <InstrumentSensitivity>
    <Value>+1.5000E+03</Value>
    <Frequency>1</Frequency>
  <InputUnits>
    <Name>m/s</Name>
    <Description>Velocity in Meters Per Second</Description>
  </InputUnits>
  <OutputUnits>
    <Name>V</Name>
    <Description>Volts</Description>
  </OutputUnits>
  </InstrumentSensitivity>
  <Stage number="1" resourceId="NRL:AnaPAZ_IUMPs_N22_NP5_PHa4408a6_AN5.71508ep08_FN1::NRL:Gain_SG1500_FN1">
    <PolesZeros>...</PolesZeros>
    <StageGain>...</StageGain>
  </Stage>
</Response>
```

# NRL v2: Web Service service.iris.edu/irisws/nrl/1



The screenshot shows the "IRIS DMC IRISWS nrl Web Service Documentation" page. At the top, there are navigation links for "WEB SERVICES / IRISWS / NRL / DOCS / V. 1". Below the header, there are tabs for "Service interface", "URL builder", "Help", and "Revisions". A "Description" section is highlighted in red. To the right of the "Description" tab is a "URL Builder" button with a blue icon. The main content area contains three sections: "/catalog", "/combine", and "/prefix-lookup". Each section provides a brief description and a list of output formats (xml, json, text). The "/catalog" section also includes a link to the "new NRL web page".

**IRIS DMC IRISWS nrl Web Service Documentation**

WEB SERVICES / IRISWS / NRL / DOCS / V. 1

Service interface URL builder Help Revisions

Description

URL Builder

**/catalog**

The **NRL** web service returns instrument response information from the Nominal Response Library database using the following service query methods:

• Output formats:

- xml (XML [.xml](#))
- json (JSON [.json](#))
- text (plain text [.txt](#))

Alternatively, you can browse NRL contents using the [new NRL web page](#).

**/combine**

Returns fully formed instrument responses in FDSN StationXML, RESP, or StationXML-Response formats. It can also return a full version of the Library for local use, similar to that used by the original Nominal Response Library. Differences in convention used by the **nrl** web service to describe responses using RESP files are detailed in the document [Nominal Response Library Versions](#).

• Output formats:

- stationxml or stationxml.zip ([StationXML as .xml](#) or [.zip archive](#))
- stationxml-resp or stationxml-resp.zip ([StationXML-Response as .xml](#) or [.zip archive](#))
- resp or resp.zip ([RESP](#) as plain text or [.zip archive](#))

**/prefix-lookup**

Returns a [prefix table](#) of instrument configuration prefixes with their names and associated logic tree questions.

• Output formats:

- xml (XML [.xml](#))
- json (JSON [.json](#))
- text (plain text [.txt](#))

# Levels of Information

## → Element

- ◆ response element type: datalogger, sensor, integrated, soh

## → Manufacturer

- ◆ instrument manufacturer: e.g., Geotech, Guralp, Kinemetrics
- ◆ also available: GenericUnity; for retrieving generic unity responses

## → Model

- ◆ instrument model: e.g., KS-2000, CMG-5T, Obsidian
- ◆ also available: UnityDatalogger, UnityVelocity, UnityAccelerometer, UnityPolynomialPA; for retrieving generic unity responses

## → Configuration

- ◆ parameters that describe the instrument settings and other information identifying the response

# Configurations: Instconfig and Prefixes

- the **instconfig** is a string that uniquely identifies the response
  - ◆ comprised of element, manufacturer, model, and parameters
  - ◆ e.g., datalogger\_Agecodagis\_Kephren\_PG2\_FR1000
  
- configuration parameters are described by two character **prefixes** followed by values
  - ◆ e.g., PG2 refers to a preamp gain value of 2  
FR1000 refers to a final sample rate of 1000 Hz

# Configurations: Prefix-Lookup

- Table describing all the Prefixes used in the Instconfig strings are available through the web service **/prefix-lookup** endpoint

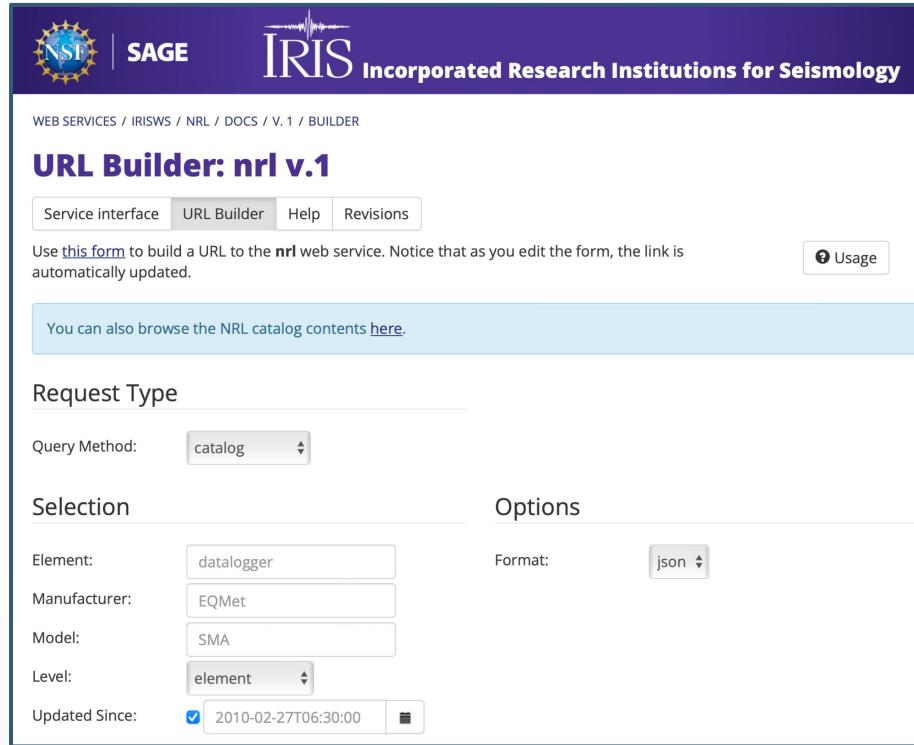
<http://service.iris.edu/irisws/nrl/1/prefix-lookup?format=text>

(other format options are JSON and XML)

# Exploring What is in the NRL: Catalog

- The **/catalog** endpoint displays information about what responses are available through the NRL web service
  - ◆ filter by element, manufacturer, model
  - ◆ display by level of information: element, manufacturer, model, configuration
    - the configuration level includes the instconfig and a human-readable text string describing the instconfig
  - ◆ output formats: JSON, XML, Text
    - JSON and XML include version date, parameter elements, and a “Detail” element that contains information helpful for response selection

# URL Builder Demo: Catalog Endpoint



The screenshot shows the URL Builder interface for the nrl web service. At the top, there are logos for NSF, SAGE, and IRIS, followed by the text "Incorporated Research Institutions for Seismology". Below this, a navigation bar includes links for WEB SERVICES / IRISWS / NRL / DOCS / V. 1 / BUILDER. The main title is "URL Builder: nrl v.1". A sub-header says "Service interface" is selected. There are tabs for URL Builder, Help, and Revisions. A note says "Use [this form](#) to build a URL to the nrl web service. Notice that as you edit the form, the link is automatically updated." To the right is a "Usage" button. A blue banner below the note says "You can also browse the NRL catalog contents [here](#)". The "Request Type" section has a dropdown for "Query Method" set to "catalog". The "Selection" section contains fields for "Element" (datalogger), "Manufacturer" (EQMet), "Model" (SMA), and "Level" (element). The "Options" section has a "Format" dropdown set to "json". At the bottom, there is a "Updated Since" field with a checked checkbox and the value "2010-02-27T06:30:00" and a calendar icon.

WEB SERVICES / IRISWS / NRL / DOCS / V. 1 / BUILDER

## URL Builder: nrl v.1

Service interface URL Builder Help Revisions

Use [this form](#) to build a URL to the nrl web service. Notice that as you edit the form, the link is automatically updated.

You can also browse the NRL catalog contents [here](#).

### Request Type

Query Method: catalog

Selection	Options
Element: datalogger	Format: json
Manufacturer: EQMet	
Model: SMA	
Level: element	

Updated Since:  2010-02-27T06:30:00 

# Retrieving Responses: Combine

- The **/combine** endpoint is where you request response files
  - ◆ select responses by instconfig, element, manufacturer, model
  - ◆ output formats:
    - format=stationxml returns a single XML document for a single response
      - requests for multiple responses will not succeed
    - format=stationxml.zip returns a ZIP file of one or more XML documents
    - also: resp, resp.zip, stationxml-resp, stationxml-resp.zip
      - where stationxml-resp is abbreviated for stationxml-response

# Retrieving Responses: Combine

- The **/combine** endpoint can also combine responses of two instruments to create a single cascaded response
  - ◆ e.g., datalogger and sensor combination
  - ◆ for use with instconfig parameter only
    - join multiple instconfig strings with colon (:)
    - example:

```
instconfig=datalogger_Agecodagis_Kephren_PG2_FR1000:sensor_Streckeisen_STS-5A_LP120_SG1500_STgroundVel
```

- instconfig accepts comma-separated lists but not wildcards
- element, manufacturer, model accept both lists and wildcards

## Other Web Service Parameters

- epoch options (combine endpoint only):
  - ◆ add network, station, location, starttime, endtime information to the StationXML or RESP output
- updatedsince
  - ◆ return only responses that have been added or changed in the NRL since the specified date (00:00:00 UTC)
- nodata
  - ◆ specify HTTP status code when no information is returned
    - 204 No Content (default)
    - 404 Not Found

# Retrieving Responses: Examples

- request StationXML for the Agecodagis Kephren datalogger with a preamp gain value of 2 and a final sample rate of 1000 Hz

<http://service.iris.edu/irisws/nrl/1/combine?format=stationxml&instconfig=datalogger Agecodagis Kephren PG2 FR1000>

- request a ZIP file containing StationXML files for each configuration of Agecodagis Kephren datalogger

<http://service.iris.edu/irisws/nrl/1/combine?format=stationxml.zip&element=datalogger&manufacturer=Agecodagis&model=Kephren>

# Retrieving Responses: Examples

- request StationXML response for:

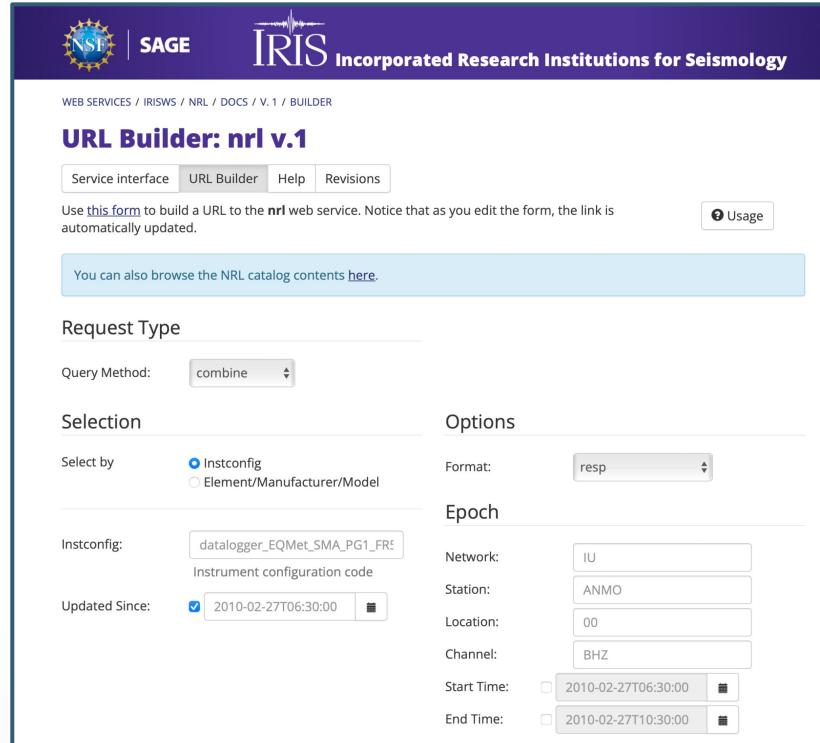
Quanterra Q330HR datalogger (preamp gain 1, final sample rate 1, ADC type HR, linear filter rates below 40 and no decimation filter)

combined with

Nanometrics Trillium 240 sensor (electronics generation 2, long period corner 244 seconds, sensitivity 1189, sensor type ground velocity)

[http://service.iris.edu/irisws/nrl/1/combine?instconfig=datalogger\\_Quanterra\\_Q3\\_30HR\\_PG1\\_FR1\\_ADHR\\_LRbelow40\\_DENone:sensor\\_Nanometrics\\_Trillium240\\_EG2\\_L\\_P244\\_SG1189\\_STgroundVel&format=stationxml](http://service.iris.edu/irisws/nrl/1/combine?instconfig=datalogger_Quanterra_Q3_30HR_PG1_FR1_ADHR_LRbelow40_DENone:sensor_Nanometrics_Trillium240_EG2_L_P244_SG1189_STgroundVel&format=stationxml)

# URL Builder Demo: Combine Endpoint



The screenshot shows the URL Builder: nrl v.1 interface. At the top, there are tabs for Service interface, URL Builder (which is selected), Help, and Revisions. A note says "Use this form to build a URL to the nrl web service. Notice that as you edit the form, the link is automatically updated." There is also a "Usage" button with a question mark icon.

Below this, a message says "You can also browse the NRL catalog contents [here](#)".

The main area is divided into Request Type, Selection, Options, and Epoch sections.

- Request Type:** Query Method: combine
- Selection:** Select by: Instconfig (radio button selected), Element/Manufacturer/Model (radio button unselected).
  - Instconfig: datalogger\_EQMet\_SMA\_PG1\_FR<sup>c</sup>
  - Instrument configuration code
- Options:** Format: resp
- Epoch:** Network: IU, Station: ANMO, Location: 00, Channel: BHZ
  - Start Time: 2010-02-27T06:30:00
  - End Time: 2010-02-27T10:30:00

# Full NRL ZIP Output from Web Service

→ The entire NRL is available in a single ZIP file

- ◆ use `instconfig=full_NRL_v2_zip`
- ◆ use `format=resp.zip` or `format=stationxml.zip`

[http://service.iris.edu/irisws/nrl/1/combine?instconfig=full\\_NRL\\_v2\\_zip&format=resp.zip](http://service.iris.edu/irisws/nrl/1/combine?instconfig=full_NRL_v2_zip&format=resp.zip)

[http://service.iris.edu/irisws/nrl/1/combine?instconfig=full\\_NRL\\_v2\\_zip&format=stationxml.zip](http://service.iris.edu/irisws/nrl/1/combine?instconfig=full_NRL_v2_zip&format=stationxml.zip)

# Differences between Full ZIP file RESP output

## NRL Version 1

1. top level subdirectories: sensors, dataloggers
2. manufacturer directories could have more than one subdirectory
3. sensor RESP files contain a Stage 2 generic unity datalogger
4. datalogger RESP files contain a Stage 1 generic unity sensor, Stage 2 preamp, Stage 3 ADC, Stage 4-N FIR and other filter/gain
5. channel response always contains a sensor (Stage 1) and a datalogger (Stage 2-N)
6. Units are upper case

## NRL Version 2

1. top level subdirectories: sensor, datalogger, integrated, soh
2. manufacturer directories have no subdirectories
3. sensor RESP files only contain a Stage 1 sensor and no generic unity datalogger
4. datalogger RESP files do not contain a generic unity sensor. Stages 1-N depend on the datalogger
5. channel response contains one or more response element types
6. Units are lower case

full accounting at: <https://ds.iris.edu/files/nrl/NominalResponseLibraryVersions.pdf>

## Plans to Retire Version 1

- NRL v1 will be maintained in parallel with v2 until at least Spring 2023
  - ◆ to allow the community time to adapt their metadata software and applications to use the new version
  - ◆ the final retirement date will be set when there is no longer a compelling need for v1
    - this will take into consideration the success of community adoption



# NRL v2: Web Page [ds.iris.edu/ds/nrl](http://ds.iris.edu/ds/nrl)

contact us | sign in

 DATA INSTRUMENTATION EDUCATION ABOUT

DATA SERVICES / NOMINAL RESPONSE LIBRARY

## Nominal Response Library

Questions? [Contact us](#)

In 2006, the IRIS DMC began to collect an "authoritative" set of manufacturers' recommended nominal instrument responses in SEED RESP format and publish these on the web. The goal behind the Library is to make it easier for the seismological community to both share and create metadata for common instrumentation, and to improve response accuracy for users of the data. All links to responses are queries to the [NRL web service](#).

To cite the the IRIS Nominal Response Library, please use - Mary E. Templeton (2017): IRIS Library of Nominal Response for Seismic Instruments. Incorporated Research Institutions for Seismology. Dataset. <https://doi.org/10.17611/S7159Q>

Download the full Library as [RESP](#) or [StationXML](#) files.

---

Search for Manufacturer or Model

**datalogger** Manufacturer help >

[Agecodagis](#)  
Kephren

[CNSN](#)  
GD2

[DAQSystems](#) Model help >  
DAQ24USB4 DAQ24USB4CH DAQ24USB5V DAQ24USBXR

[DiGOSOmnirecs](#)  
DataCube

# NRL Exercises

1. Go to the new NRL web page ([ds.iris.edu/ds/nrl](https://ds.iris.edu/ds/nrl))
  - a.
  - b.
  
2. Go to the NRL Web Service URL Builder ([service.iris.washington.edu/irisws/nrl](https://service.iris.washington.edu/irisws/nrl))
  - a.



# Yasmine

Yet Another Station Metadata Information Editor

IRIS/RESIF

# Yasmine

**Yasmine** is a general purpose tool for creating, editing, and writing StationXML

- Jointly developed and maintained by the Incorporated Research Institutions for Seismology ([IRIS](#)) and the French Seismological and Geodetic Network ([RESIF](#))
- Intended audiences include network operators, principal investigators, and anyone needing to create StationXML metadata for their seismic instrumentation
- Code is in two GitHub Repositories, in beta:
  - ◆ Yasmine Editor: <https://github.com/iris-edu/yasmine-stationxml-editor>
  - ◆ Yasmine CLI: <https://github.com/iris-edu/yasmine-stationxml-cli>

## Response Sources Used By Yasmine: NRL vs. AROL

- The Nominal Response Library (NRL)
  - ◆ Yasmine uses NRL v1, bundled as a .zip file
  - ◆ Has not been updated to use latest version yet (no access to integrated, SOH, or polynomial instruments)
- The Atomic Response Objects Library (AROL)
  - ◆ A new instrument response library under development by Résif
  - ◆ Easier, faster selection of instrument configurations
  - ◆ Includes a smaller set of instruments

# Running Yasmine Editor

1. Install [Docker Compose](#), if not already on your machine
  - to determine if it is installed, try running on the command line:  
> docker-compose version
2. Clone the Yasmine Editor Git repository:  
> git clone <https://github.com/iris-edu/yasmine-stationxml-editor>
3. **Note:** If you are running on an Apple M1 machine, uncomment the lines indicating the target platform in the `docker-compose.yml` file.
4. Run the application:  
> cd yasmine-stationxml-editor && docker-compose up

This step may take 10-15 minutes to start up the first time you run the Yasmine Editor while it loads the NRL

# Running Yasmine Editor

4a. The first time, look for the following command line output:

```
> yasmine-frontend | [INF] Waiting for changes...
> yasmine-backend | INFO:yasmine.app.helpers.nrl.nrl_helper:Unzipping NRL
> yasmine-backend | INFO:yasmine.app.helpers.nrl.nrl_helper:NRL has been unzipped
> yasmine-backend | INFO:yasmine.app.helpers.nrl.nrl_helper:Creating an NRL key files
> yasmine-backend | INFO:yasmine.app.helpers.nrl.nrl_helper:NRL key files have been
                           created
```

4b. Otherwise, Look for the following command line output:

```
> yasmine-frontend | [INF] Waiting for changes...
```

5. Visit <http://localhost:1841> in browser window

(to stop Yasmine Editor, run `docker-compose down` on the command line)

# Yasmine User Interface

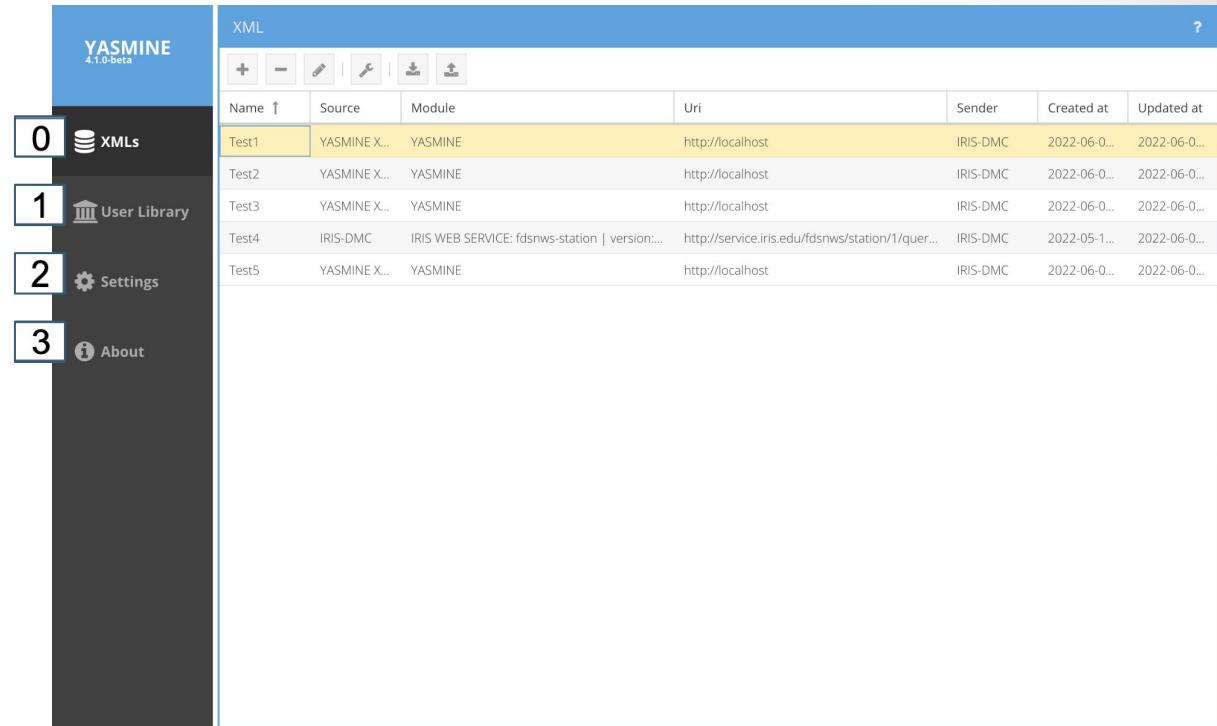
## Main Left Side Tabs

0: XMLs

1: User Library

2: Settings

3: About

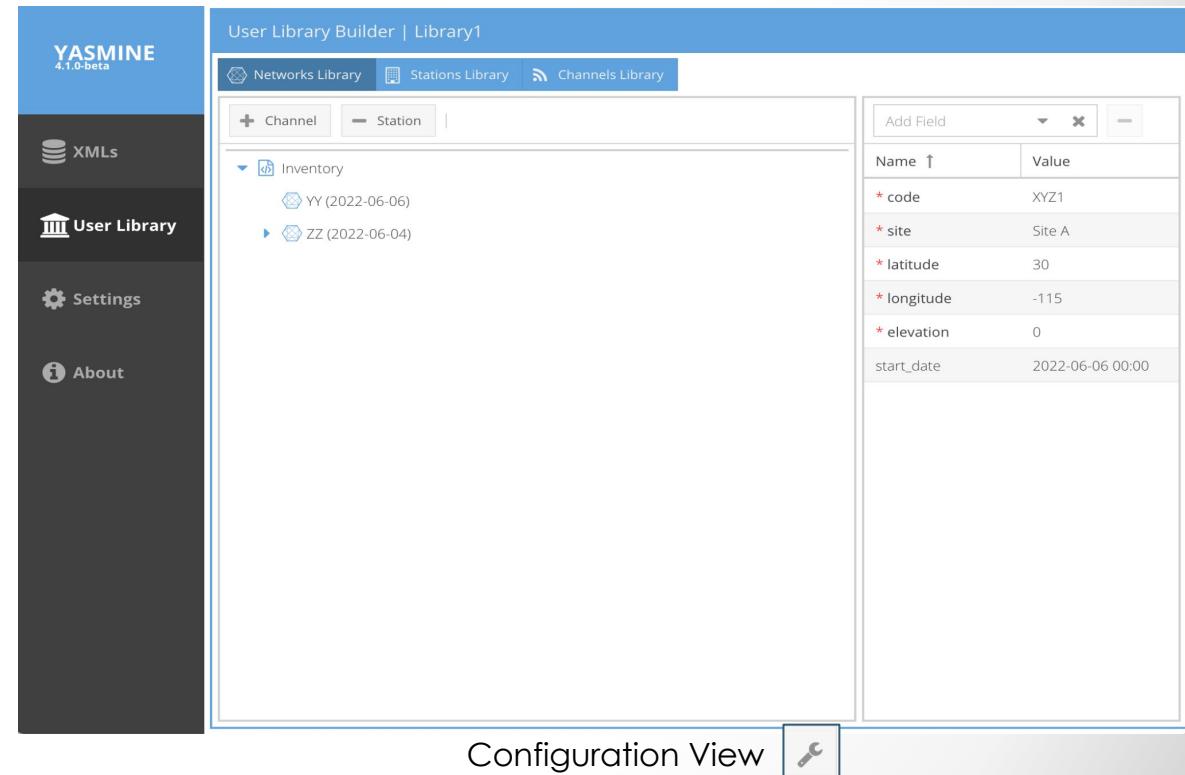


The screenshot shows the Yasmine User Interface. On the left, there is a vertical sidebar with four tabs labeled 0 through 3. Tab 0 is selected and highlighted in blue, showing the 'XMLs' icon. The other tabs are labeled 1 (User Library), 2 (Settings), and 3 (About). The main content area is titled 'XML' and displays a table of XML entries. The table has columns for Name, Source, Module, Uri, Sender, Created at, and Updated at. The first entry, 'Test1', is highlighted in yellow. The data in the table is as follows:

Name ↑	Source	Module	Uri	Sender	Created at	Updated at
Test1	YASMINE X...	YASMINE	http://localhost	IRIS-DMC	2022-06-0...	2022-06-0...
Test2	YASMINE X...	YASMINE	http://localhost	IRIS-DMC	2022-06-0...	2022-06-0...
Test3	YASMINE X...	YASMINE	http://localhost	IRIS-DMC	2022-06-0...	2022-06-0...
Test4	IRIS-DMC	IRIS WEB SERVICE: fdsnws-station   version:...	http://service.iris.edu/fdsnws/station/1/quer...	IRIS-DMC	2022-05-1...	2022-06-0...
Test5	YASMINE X...	YASMINE	http://localhost	IRIS-DMC	2022-06-0...	2022-06-0...

# Yasmine User Library

- StationXML elements may be saved in the User Library for later reuse and fast prototyping
  - ◆ custom templated or derived from imported StationXML files
- Contains three sub-libraries:
  - ◆ Network Library
  - ◆ Station Library
  - ◆ Channel Library



The screenshot shows the YASMINE 4.1.0-beta software interface. On the left is a sidebar with the following options:

- XMLS
- User Library** (highlighted)
- Settings
- About

The main area is titled "User Library Builder | Library1". It has tabs for Networks Library, Stations Library, and Channels Library, with Stations Library selected. Below the tabs are buttons for "Channel" and "Station".

The central pane displays a tree view under "Inventory":

- YY (2022-06-06)
- ZZ (2022-06-04)

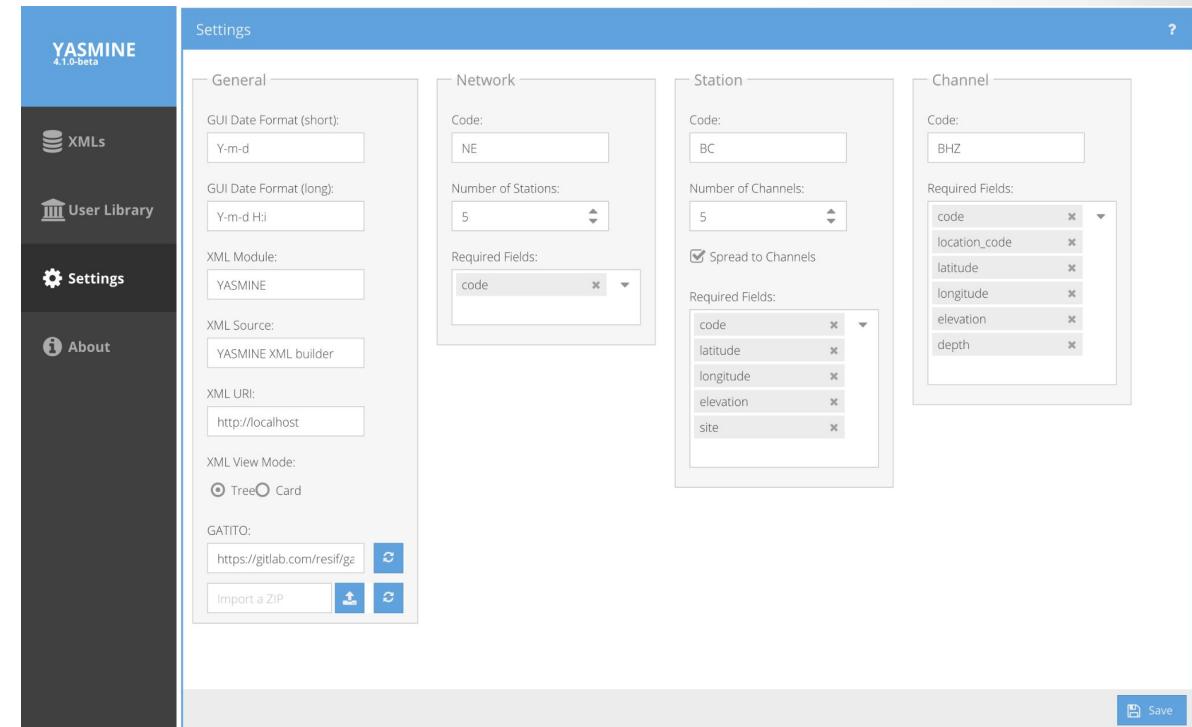
To the right is a "Configuration View" panel containing a table of fields:

Name ↑	Value
* code	XYZ1
* site	Site A
* latitude	30
* longitude	-115
* elevation	0
Start_date	2022-06-06 00:00

A "Configuration View" button is located at the bottom right of the panel.

# Yasmine Settings

- Configures the global behavior of the application
- Options include:
  - ◆ General properties
    - Datetime format
    - XML Module/Source/URI
    - XML navigation view
    - Gitlab repository for AROL responses (GATITO)
  - ◆ Network, Station, and Channel element properties
    - Required fields
    - Default codes



The screenshot shows the YASMINE 4.1.0-beta settings interface. The left sidebar has icons for XMLs, User Library, Settings (selected), and About. The main area has four tabs: General, Network, Station, and Channel.

- General:** GUI Date Format (short: Y-m-d), GUI Date Format (long: Y-m-d H:i), XML Module: YASMINE, XML Source: YASMINE XML builder, XML URI: http://localhost, XML View Mode: Tree (selected), GATITO: https://gitlab.com/resif/ge, Import a ZIP.
- Network:** Code: NE, Number of Stations: 5, Required Fields: code.
- Station:** Code: BC, Number of Channels: 5, Required Fields: code, latitude, longitude, elevation, site. A checked checkbox "Spread to Channels" is present.
- Channel:** Code: BHZ, Required Fields: code, location\_code, latitude, longitude, elevation, depth.

A "Save" button is at the bottom right.

# Yasmine XMLs

- Main view for manipulating metadata
  - ◆ Import existing StationXML for editing
  - ◆ Create StationXML using Wizard Tool
  - ◆ Create StationXML using a default template
  - ◆ Create StationXML using metadata stored as templates in the User Library

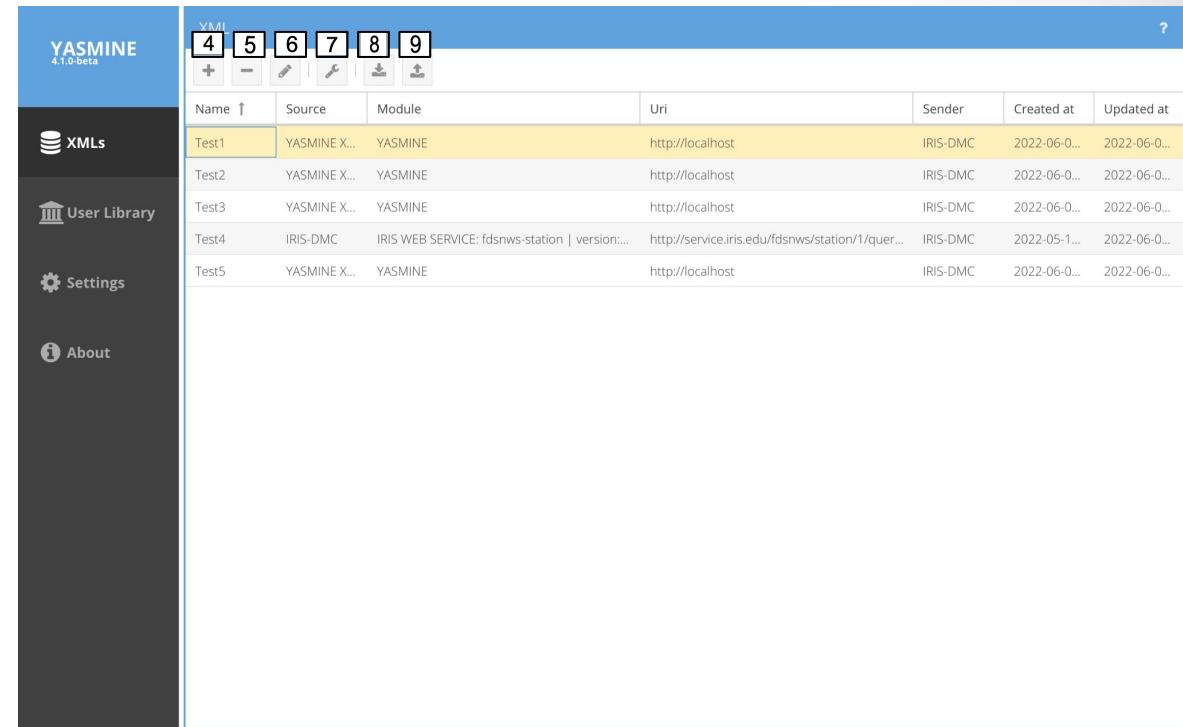
The screenshot shows the Yasmine XMLs application interface. On the left is a dark sidebar with the YASMINE 4.1.0-beta logo at the top, followed by four menu items: XMLs, User Library, Settings, and About. The main area is titled "XML" and contains a table with the following data:

Name ↑	Source	Module	Uri	Sender	Created at	Updated at
Test1	YASMINE X...	YASMINE	http://localhost	IRIS-DMC	2022-06-0...	2022-06-0...
Test2	YASMINE X...	YASMINE	http://localhost	IRIS-DMC	2022-06-0...	2022-06-0...
Test3	YASMINE X...	YASMINE	http://localhost	IRIS-DMC	2022-06-0...	2022-06-0...
Test4	IRIS-DMC	IRIS WEB SERVICE: fdsnws-station   version:...	http://service.iris.edu/fdsnws/station/1/quer...	IRIS-DMC	2022-05-1...	2022-06-0...
Test5	YASMINE X...	YASMINE	http://localhost	IRIS-DMC	2022-06-0...	2022-06-0...

# Yasmine XMLs

## Top Icons

- 4: Create XML
- 5: Delete XML
- 6: Edit XML
- 7: XML Builder
- 8: Import XML
- 9: Export XML

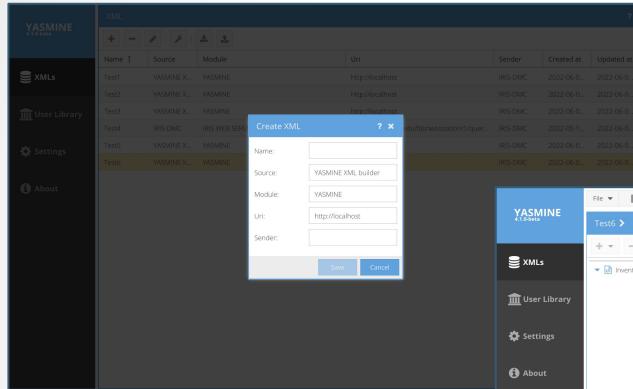


The screenshot shows the Yasmine XMLs application interface. On the left is a dark sidebar with the Yasmine logo and version (4.1.0-beta), followed by navigation icons for XMLs, User Library, Settings, and About.

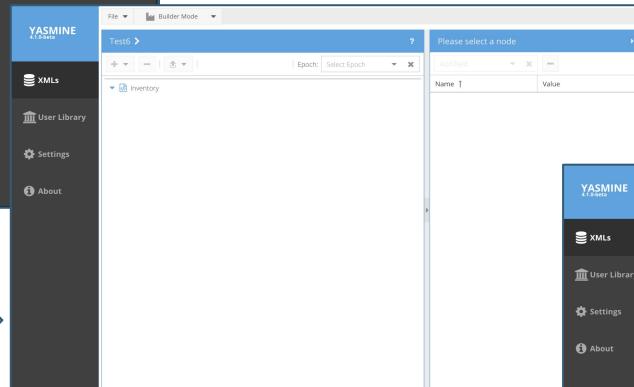
The main area has a header with a toolbar containing icons for XML creation, deletion, edit, builder, import, and export. Below the toolbar is a table listing XML entries:

Name ↑	Source	Module	Uri	Sender	Created at	Updated at
Test1	YASMINE X...	YASMINE	http://localhost	IRIS-DMC	2022-06-0...	2022-06-0...
Test2	YASMINE X...	YASMINE	http://localhost	IRIS-DMC	2022-06-0...	2022-06-0...
Test3	YASMINE X...	YASMINE	http://localhost	IRIS-DMC	2022-06-0...	2022-06-0...
Test4	IRIS-DMC	IRIS WEB SERVICE: fdsnws-station   version:...	http://service.iris.edu/fdsnws/station/1/quer...	IRIS-DMC	2022-05-1...	2022-06-0...
Test5	YASMINE X...	YASMINE	http://localhost	IRIS-DMC	2022-06-0...	2022-06-0...

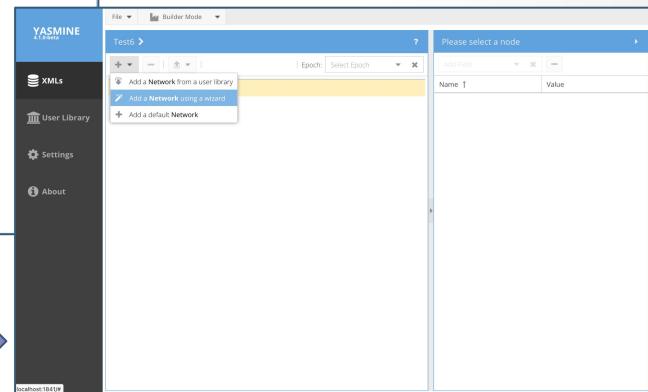
# Creating Metadata



The screenshot shows the YASMINE XML builder interface. On the left, there's a sidebar with 'YASMINE ELEMENTS' and icons for 'XMLs', 'User Library', 'Settings', and 'About'. The main area has a table with columns: Name, Source, Module, URL, Sender, Created At, and Uploaded At. There are six rows labeled Test1 through Test6. A modal window titled 'Create XML' is open in the center, containing fields for Name, Source (set to 'YASMINE XML builder'), Module ('YASMINE'), URL ('http://localhost'), and Sender ('IRIS DMC'). Buttons for 'Save' and 'Cancel' are at the bottom.



The screenshot shows the YASMINE XML builder interface with 'Builder Mode' selected. The left sidebar includes 'File', 'Builder Mode' (selected), 'Help', 'XMLs', 'User Library' (highlighted), 'Inventory', 'Settings', and 'About'. In the center, a table titled 'Test6' lists nodes with columns: Name, Value, and Epoch. A modal window titled 'Please select a node' is open, showing a table with 'Name' and 'Value' columns. A purple arrow points from the 'User Library' icon in the sidebar to this modal window.

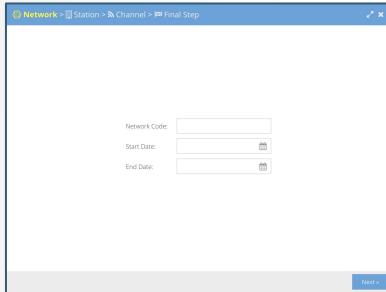


The screenshot shows the YASMINE XML builder interface with 'File' selected in the top menu. The 'File' menu contains three items: 'Add a Network from a user library' (unchecked), 'Add a Network using a wizard' (checked), and 'Add a default Network'. A purple arrow points from the 'File' menu to the checked option.

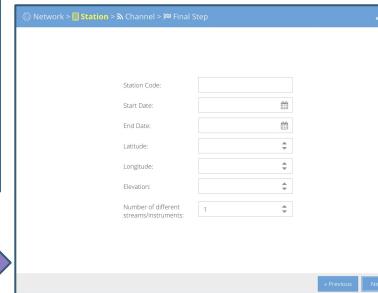
Add from Library  
Add from Wizard  
Add from Default

# Walking Through the Wizard

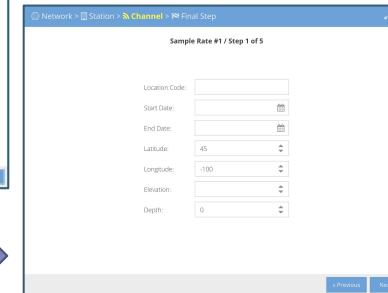
## Network Level



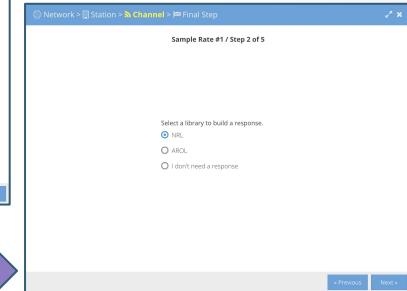
## Station Level



## Channel Level



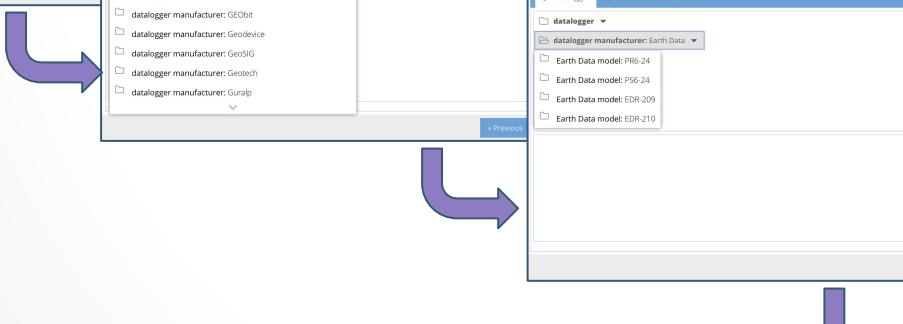
## Response Level



Choice of:  
NRL  
AROL  
No Response

# Walking Through the Wizard - NRL Response

Choose datalogger parameters



The screenshots illustrate the configuration process:

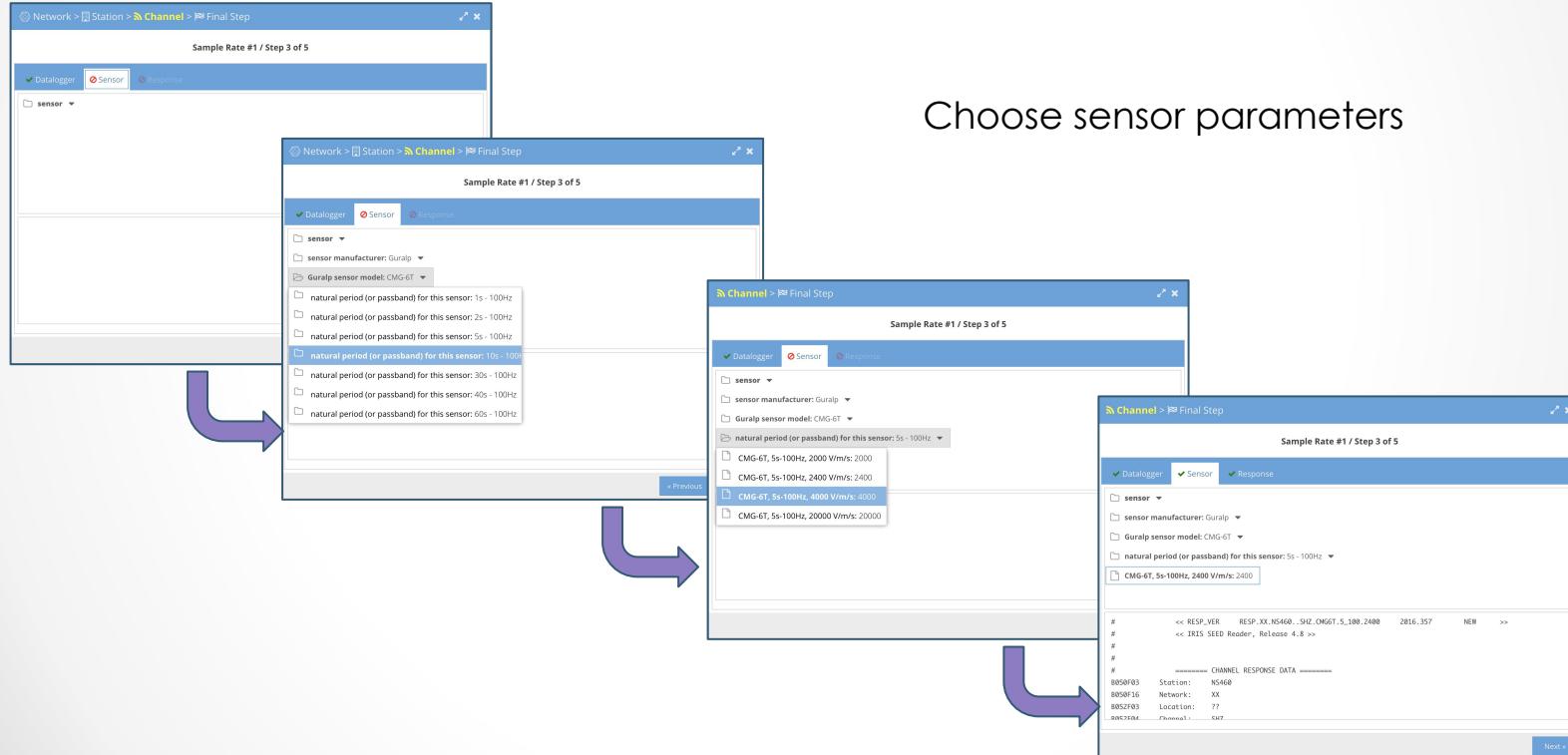
- Step 1:** Shows the initial configuration screen for a datalogger.
- Step 2:** A dropdown menu lists various datalogger manufacturers, with "datalogger manufacturer: Earth Data" selected.
- Step 3:** The configuration screen now shows "datalogger manufacturer: Earth Data" and "Earth Data model: PS6-24" selected.
- Step 4:** The final configuration screen displays the selected parameters and the generated RESP file content.

**Generated RESP File Content:**

```
# << RESP_VER    RESP_XX_NEON08..M42.PS6_24.L_4 2089.154 NEW >>
#
# ***** CHANNEL RESPONSE DATA *****
B089E03 Station: NEON8
B089E16 Network: XX
B08CF83 Location: ?
B087604 Channel: M42
```

# Walking Through the Wizard - NRL Response

Choose sensor parameters



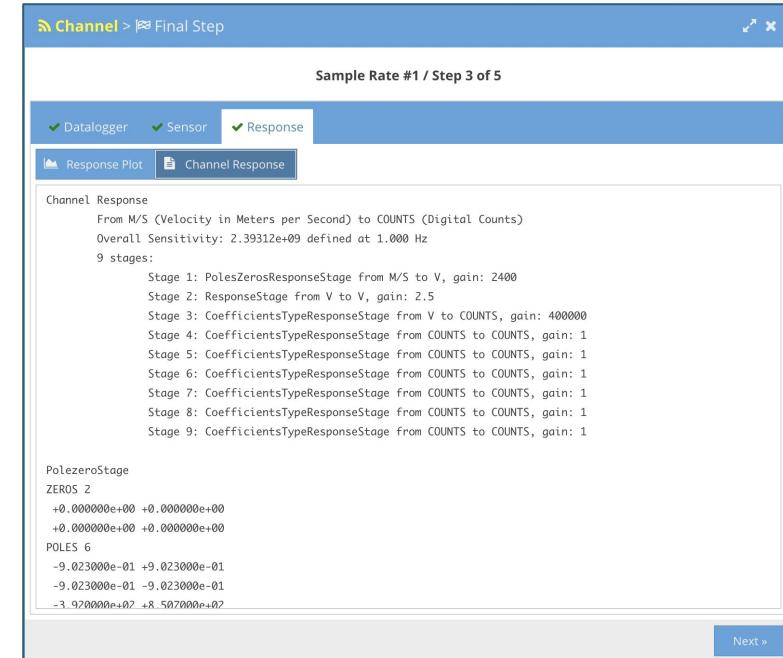
The diagram illustrates the process of setting up a sensor response through a series of four windows:

- Network > Station > Channel > Final Step**: The first window shows the path "Sample Rate #1 / Step 3 of 5". It has tabs for Datalogger, Sensor, and Response. Under the Sensor tab, it lists "sensor" and "Guralp sensor model: CMG-6T". A purple arrow points from this window to the second window.
- Network > Station > Channel > Final Step**: The second window also shows "Sample Rate #1 / Step 3 of 5". It has tabs for Datalogger, Sensor, and Response. Under the Sensor tab, it lists "sensor", "sensor manufacturer: Guralp", and "Guralp sensor model: CMG-6T". It also lists "natural period (or passband) for this sensor: 1s - 100Hz", "natural period (or passband) for this sensor: 2s - 100Hz", "natural period (or passband) for this sensor: 5s - 100Hz", "natural period (or passband) for this sensor: 10s - 100Hz", "natural period (or passband) for this sensor: 30s - 100Hz", "natural period (or passband) for this sensor: 40s - 100Hz", and "natural period (or passband) for this sensor: 60s - 100Hz". A purple arrow points from the first window to this one.
- Channel > Final Step**: The third window shows "Sample Rate #1 / Step 3 of 5". It has tabs for Datalogger, Sensor, and Response. Under the Sensor tab, it lists "sensor", "sensor manufacturer: Guralp", and "Guralp sensor model: CMG-6T". It also lists "natural period (or passband) for this sensor: 5s - 100Hz", "CMG-6T, 5s-100Hz, 2000 V/m/s: 2000", "CMG-6T, 5s-100Hz, 2400 V/m/s: 2400", "CMG-6T, 5s-100Hz, 4000 V/m/s: 4000", and "CMG-6T, 5s-100Hz, 20000 V/m/s: 20000". A purple arrow points from the second window to this one.
- Channel > Final Step**: The fourth window shows "Sample Rate #1 / Step 3 of 5". It has tabs for Datalogger, Sensor, and Response. Under the Sensor tab, it lists "sensor", "sensor manufacturer: Guralp", and "Guralp sensor model: CMG-6T". It also lists "natural period (or passband) for this sensor: 5s - 100Hz" and "CMG-6T, 5s-100Hz, 2400 V/m/s: 2400". Below the tabs, there is a text area:

```
#      << RESP_VER      RESP_XX_NS460..SHZ_CMG6T_5_180_2400     2016.357    NEW >>
#
#      ===== CHANNEL RESPONSE DATA =====
#B000103  Station: NS46B
#B000116  Network: XX
#B021703  Location: ?
#B027741  Channel: C47
```

A purple arrow points from the third window to this one.

# Walking Through the Wizard - NRL Response



Resulting response plot and channel response

# Walking Through the Wizard - NRL Response

Channel > Final Step

Sample Rate #1 / Step 4 of 5

Channel Prefix: SH

Channel Orientation:

- ZNE (3 channels)
- Z12 (3 channels)
- Z (1 channel)

Next »

Channel > Final Step

Sample Rate #1 / Step 5 of 5

Channel:

SHZ	SHN	SHE
-----	-----	-----

Dip:

-90	▲	0	▼	0	▼
-----	---	---	---	---	---

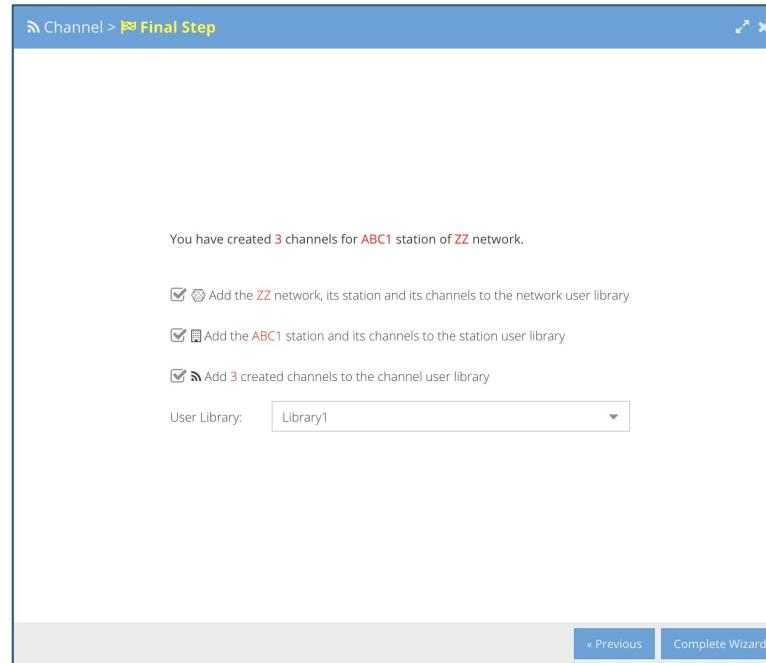
Azimuth:

0	▼	0	▼	90	▼
---	---	---	---	----	---

Next »

Assigning Channels and Orientations

# Walking Through the Wizard - NRL Response



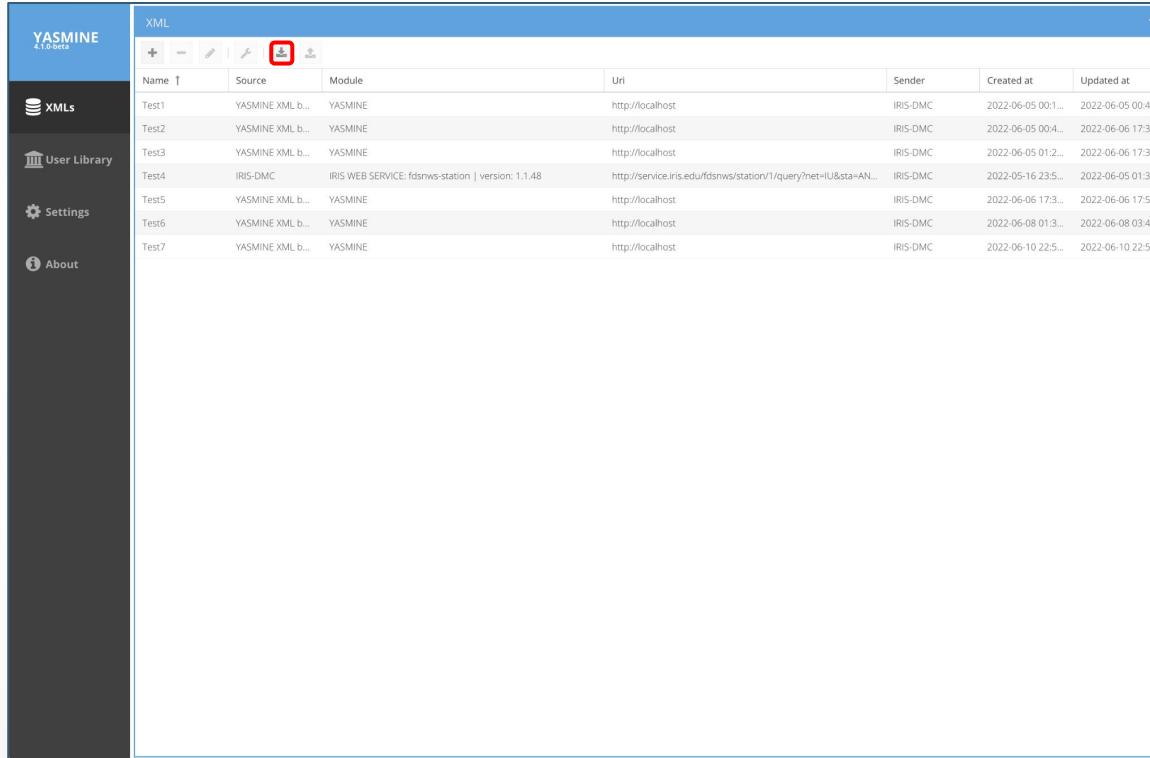
The screenshot shows a software window titled "Channel > Final Step". The main content area displays a message: "You have created 3 channels for ABC1 station of ZZ network." Below this, there are three checked options with corresponding icons:

-  Add the ZZ network, its station and its channels to the network user library
-  Add the ABC1 station and its channels to the station user library
-  Add 3 created channels to the channel user library

Below these options is a "User Library:" dropdown menu set to "Library1". At the bottom of the window are two buttons: "« Previous" and "Complete Wizard".

Final step - optional addition to Library for later re-use

# Importing Existing StationXML File



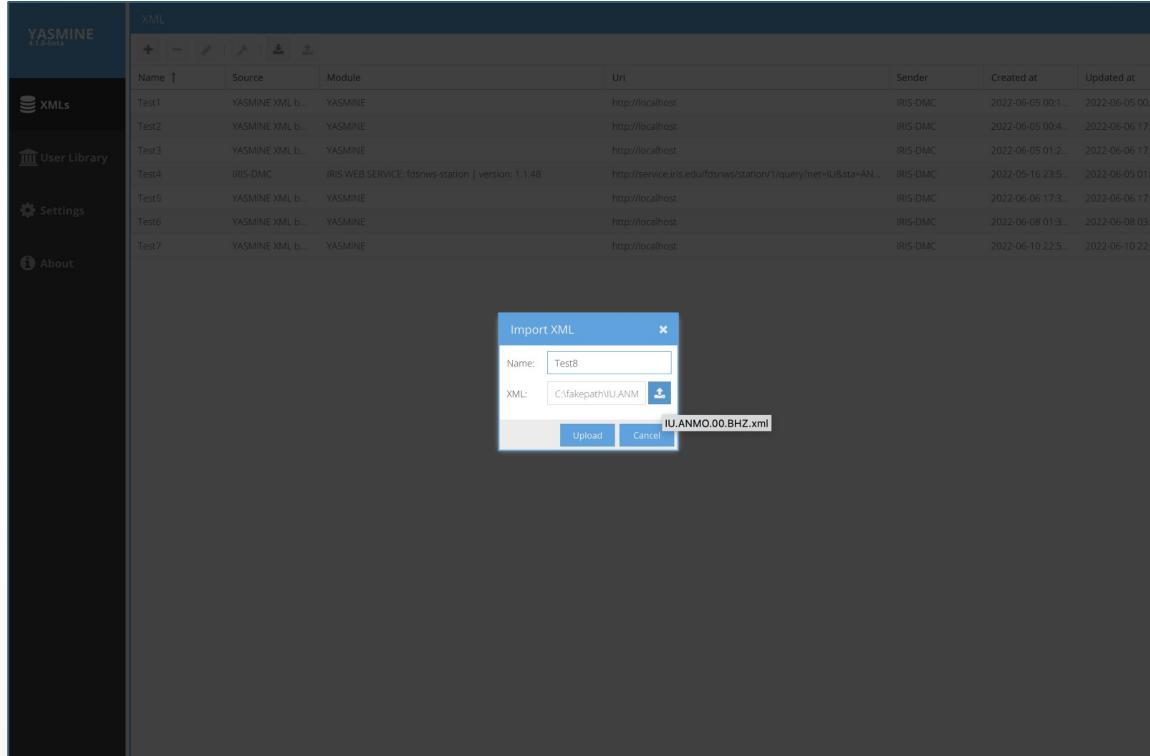
The screenshot shows the YASMINE 4.1.0-beta software interface. On the left is a dark sidebar with the following navigation options:

- YASMINE** (4.1.0-beta)
- XMLs**
- User Library**
- Settings**
- About**

The main area is titled "XML" and contains a table with the following data:

Name	Source	Module	Uri	Sender	Created at	Updated at
Test1	YASMINE XML b...	YASMINE	http://localhost	IRIS-DMC	2022-06-05 00:1...	2022-06-05 00:4...
Test2	YASMINE XML b...	YASMINE	http://localhost	IRIS-DMC	2022-06-05 00:4...	2022-06-06 17:3...
Test3	YASMINE XML b...	YASMINE	http://localhost	IRIS-DMC	2022-06-05 01:2...	2022-06-06 17:3...
Test4	IRIS-DMC	IRIS WEB SERVICE: fdsnws-station   version: 1.1.48	http://service.iris.edu/fdsnws/station/1/query?net=IU&sta=AN...	IRIS-DMC	2022-05-16 23:5...	2022-06-05 01:3...
Test5	YASMINE XML b...	YASMINE	http://localhost	IRIS-DMC	2022-06-06 17:3...	2022-06-06 17:5...
Test6	YASMINE XML b...	YASMINE	http://localhost	IRIS-DMC	2022-06-08 01:3...	2022-06-08 03:4...
Test7	YASMINE XML b...	YASMINE	http://localhost	IRIS-DMC	2022-06-10 22:5...	2022-06-10 22:5...

# Importing Existing StationXML File



The screenshot shows the YASMINE web application interface. On the left is a sidebar with icons for XMLs, User Library, Settings, and About. The main area has a header "XML" with a toolbar containing icons for add, edit, delete, and search. Below is a table listing seven entries:

Name	Source	Module	Url	Sender	Created at	Updated at
Test1	YASMINE XML b...	YASMINE	http://localhost	IRIS-DMC	2022-06-05 00:1...	2022-06-05 00:4...
Test2	YASMINE XML b...	YASMINE	http://localhost	IRIS-DMC	2022-06-05 00:4...	2022-06-05 17:3...
Test3	YASMINE XML b...	YASMINE	http://localhost	IRIS-DMC	2022-06-05 01:2...	2022-06-06 17:3...
Test4	IRIS-DMC	IRIS WEB SERVICE: fdsnws-station   version: 1.1.48	http://service.iris.edu/fdsnws/station/1/query?net=IU&sta=AN...	IRIS-DMC	2022-05-16 23:5...	2022-06-05 01:3...
Test5	YASMINE XML b...	YASMINE	http://localhost	IRIS-DMC	2022-06-06 17:3...	2022-06-06 17:5...
Test6	YASMINE XML b...	YASMINE	http://localhost	IRIS-DMC	2022-06-08 01:3...	2022-06-08 03:4...
Test7	YASMINE XML b...	YASMINE	http://localhost	IRIS-DMC	2022-06-10 22:5...	2022-06-10 22:5...

A modal dialog titled "Import XML" is open in the center. It contains fields for "Name" (set to "Test8") and "XML" (set to "C:\fakepath\IU.ANM"). There is also a user selection dropdown. At the bottom are "Upload" and "Cancel" buttons. A tooltip "IU.ANMO.00.BHZ.xml" is visible near the "Upload" button.

# Editing and Adding Information

The screenshot shows the YASMINE software interface. On the left is a sidebar with icons for XML, User Library, Settings, and About. The main area has a title bar "Test1 > ZZ > ABC1 > BHE" and a dropdown "Epoch: Select Epoch". Below this is a tree view under "Inventory": ZZ (2022-06-04) contains ABC1 (2022-06-04) with 00.MHE, 00.MHN, 00.MHZ, 10.LHE, 10.LHN, 10.LHZ, and 20.BHE (all from 2022-06-04). ABC2 (2022-06-04) contains 00.BH1, 00.BH2, and 00.BHZ (all from 2022-06-04). A "Channel Information" dialog box is open on the right, listing various parameters with their values. The "latitude" field is highlighted with a yellow background and has a value of "49". An arrow points to this value box with the text "select the value box to edit".

Name	Value
* code	BHE
* depth	0
* latitude	49
* longitude	-145
* elevation	0
* location_code	20
start_date	2022-06-04 00:00:00
azimuth	90
dip	0
data_logger	Quanterra, Q330HR-G-1
response	Quanterra, Q330HR-G-1 Guralp, CMG3T
sample_rate	100
sensor	Guralp, CMG3T

← select the value box to edit

(clicking on the response box will allow you to view the response plot as well as edit)

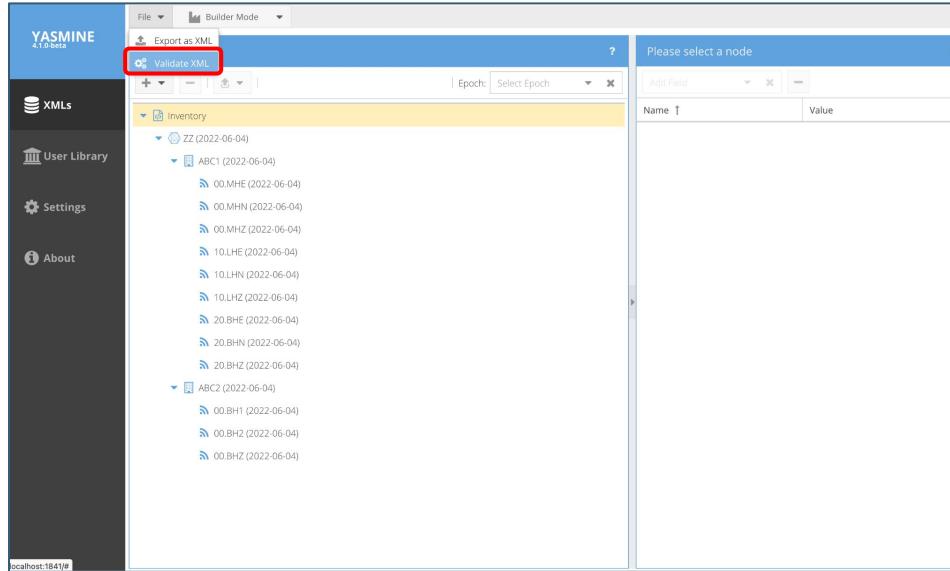
# Editing and Adding Information

The screenshot shows the YASMINE software interface. On the left is a sidebar with options: YASMINE 4.0 beta, XMLs, User Library, Settings, and About. The main window shows a file structure: Test1 > ZZ > ABC1 > BHE. A dropdown menu "File" is open, showing "Builder Mode". The "Inventory" section lists various channels under ZZ, ABC1, and ABC2. One channel, "20 BHE (2022-06-04)", is highlighted with a yellow background. To the right, a "Channel Information" dialog box is open, listing fields like alternate\_code, calibration\_units, etc., with their values. An "Add Field" button is at the top of the list. A red box highlights this button, and a blue arrow points from it to the text "Choose fields to add".

Field	Value
alternate_code	BHE
calibration_units	0
clock_drift_in_seconds_per_sample	45
comments	-145
description	0
end_date	20
equipments	
external_references	2022-06-04 00:00:00
historical_code	90
dip	0
data_logger	Quanterra, Q330HR-G-1
response	Quanterra, Q330HR-G-1 Guralp, CMG3T
sample_rate	100
sensor	Guralp, CMG3T

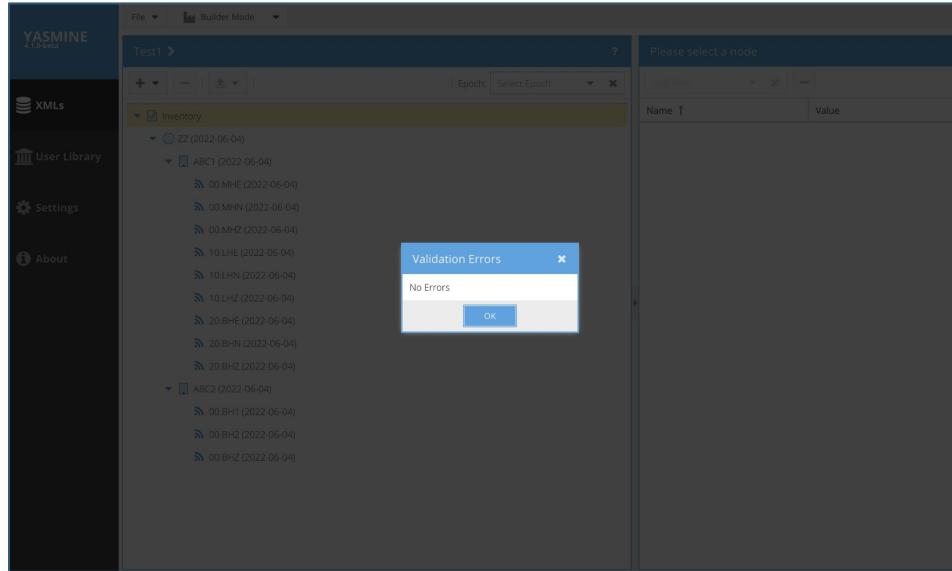
Choose fields to add  
(if you choose 'response' here, it launches the NRL/AROL wizard)

# Validation of StationXML



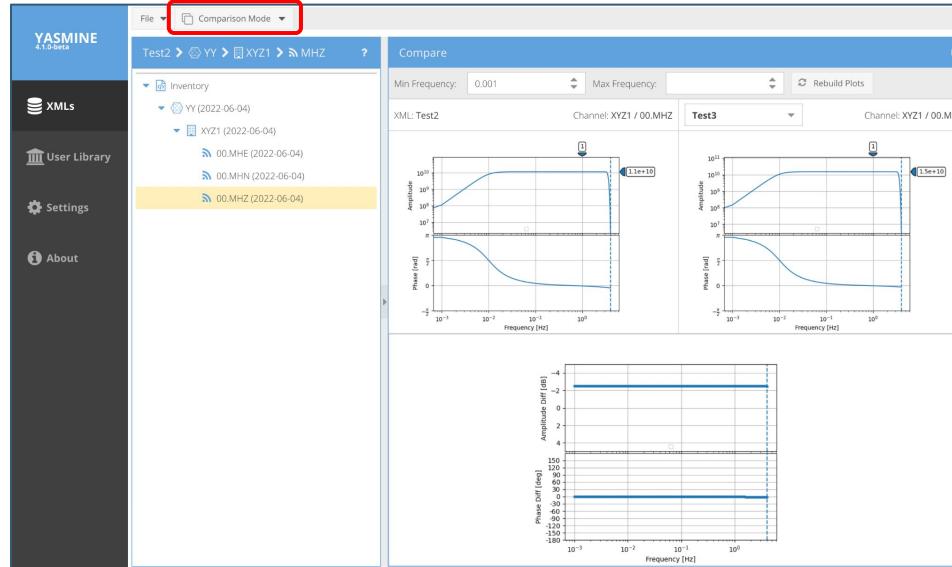
- Yasmine validation only extends to the StationXML .xsd schema
- More extensive validation to meet IRIS StationXML rules is available at:  
<http://github.com/iris-edu/stationxml-validator>

# Validation of StationXML



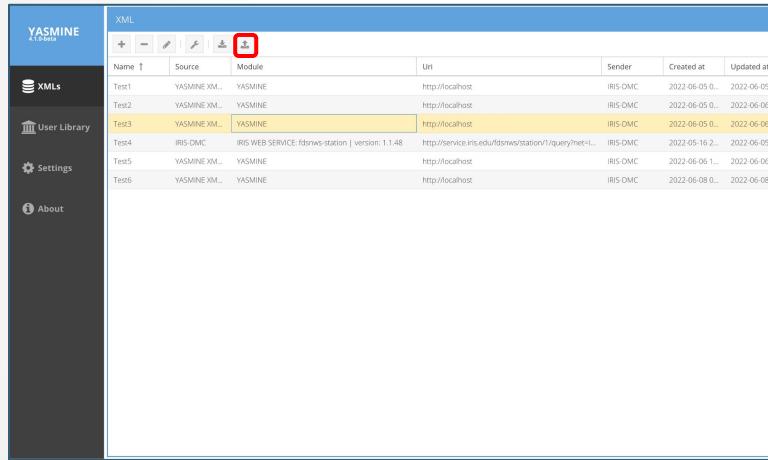
- Yasmine validation only extends to the StationXML .xsd schema
- More extensive validation to meet IRIS StationXML rules is available at: <http://github.com/iris-edu/stationxml-validator>

# Comparison Mode



- must be same network-station-location-channel (SNCL) code
- responses to compare must be in different XML files

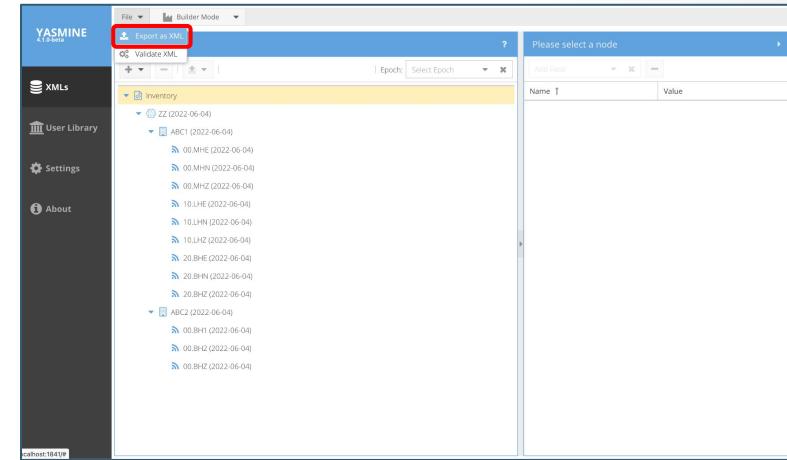
# Exporting StationXML



The screenshot shows the YASMINE web interface. On the left is a sidebar with icons for XMLs, User Library, Settings, and About. The main area is titled "XML" and contains a table with the following data:

Name	Source	Module	URI	Sender	Created at	Updated at
Test1	YASMINE XML...	YASMINE	http://localhost	IRIS DMC	2022-06-05 0...	2022-06-05 0...
Test2	YASMINE XML...	YASMINE	http://localhost	IRIS DMC	2022-06-05 0...	2022-06-06 1...
Test3	YASMINE XML...	YASMINE	http://localhost	IRIS DMC	2022-06-05 0...	2022-06-06 1...
Test4	IRIS DMC	IRIS WEB SERVICE: fdsnws-station   version: 1.1.48	http://service.iris.edu/fdsnws/station/v1/query?net=L...	IRIS DMC	2022-05-16 2...	2022-06-05 0...
Test5	YASMINE XML...	YASMINE	http://localhost	IRIS DMC	2022-06-06 1...	2022-06-06 1...
Test6	YASMINE XML...	YASMINE	http://localhost	IRIS DMC	2022-06-08 0...	2022-06-08 0...

from top-level view



The screenshot shows the YASMINE web interface in Builder/Comparison Mode. The left sidebar includes icons for XMLs, User Library, Settings, and About, along with a "File" menu item. The main area has tabs for "Builder Mode" and "Validate XML". The "Inventory" tab is selected, displaying a tree structure of XML files. The tree shows a root node "ZZ (2022-06-04)" which branches into "ABC1 (2022-06-04)" and "ABC2 (2022-06-04)". Each branch further divides into specific XML files like "00.MHE (2022-06-04)", "10.LHN (2022-06-04)", etc. A right-hand panel titled "Please select a node" contains a table for setting values.

from Builder/Comparison Mode view



# Yasmine Editor Demo

# Yasmine Editor Exercises

1. download StationXML from station service (curl command), then import into YASMINE, then edit

```
wget -O XML1  
http://service.iris.edu/fdsnws/station/1/query?net=IU&sta=ANMO&loc=00&cha=BHZ&level=response&format=xml'
```

2. creation from wizard, set out parameters for people to use

# Yasmine Command Line Interface

→ batch editing and merging of StationXML files

```
usage: yasmine-cli [-h | --help]
                   [--level_network II | --level_station II.* | --level_channel II.ANMO.00.*]
                   [--action [add/delete/update basenode]]
                   [--epoch_station int] [--epoch_channel int]
                   [--field FIELD] [--value VALUE | --from_yml fname.yml]
                   [--infiles] [-o] [-p] [--print_all] [--dont_validate]
                   [--schema_version ver] [--show_fields] [--plot_resp]
                   [--plot_dir path]
```

# Yasmine CLI - Installation

- Use for batch editing and merging of StationXML files

## 1. Create conda environment

(optional, conda available at <https://anaconda.org/conda-forge/conda>)

```
> conda create -n yasmine_test python=3.8  
> conda activate yasmine_test
```

## 2. Install matplotlib backend

```
> pip install pyqt5
```

## 3. Clone the Yasmine CLI Git repository and install:

```
> git clone https://github.com/iris-edu/yasmine-stationxml-cli  
> cd yasmine-stationxml-cli  
> pip install .
```

# Yasmine CLI - Levels and Epochs

→ specify where in the XML document to perform actions

## Levels: Network, Station, Channel

--level\_network=IU (act on this network)

--level\_network=\* (act on all networks)

--level\_station=IU.ANMO (act on this station)

--level\_station=\*.ANMO (act on this station in all networks)

--level\_station=IU.\* (act on all stations in this network)

--level\_station=\*\* (act on all stations in all networks)

--level\_channel=IU.ANMO.00.BHZ (act on this channel)

--level\_channel=IU.ANMO.00.\* (act on all channels at this location code)

--level\_channel=IU.ANMO.\*.\* (act on all channels at all location codes)

--level\_channel=IU.\*.\*.\* (act on all channels of all stations of this network)

--level\_channel=\*\*.\*.\* (act on all channels of all stations of all networks)

# Yasmine CLI - Levels and Epochs

- specify where in the XML document to perform actions

## Epochs: stations or channels

--epoch\_station=int (act on fields in this station epoch index number)

eg, --epoch\_station=1

--epoch\_channel=int (act on fields in this channel epoch index number)

eg, --epoch\_channel=0

# Yasmine CLI - Input/Output

→ specify input files and output to file or standard out

- infiles comma separated list of input xml files
- o, --output name of output xml file [default=stdout]
- p, --print print out sorted Station/Channel epochs
- print\_all print out sorted Station/Channel epochs + operator/comment lists

# Yasmine CLI - Actions

- specify how to modify or act on the XML document

## Actions: Add, Delete, Select, Update

add a new basenode object (network, station, channel) from a .yml file

```
--action=add --from_yml=yml:/path/to/file.yml --infiles=...
```

Example channel.yml file

```
Channel:  
  code: 'HN1'  
  location_code: '30'  
  description: 'This is an added channel'  
  start_date: 2006-06-30T20:00:01.000000Z  
  end_date: 2399-12-31T23:59:59.000000Z  
  latitude: 34.945911  
  longitude: -106.457199  
  elevation: 1820.0  
  depth: 0.0  
  azimuth: 90.0
```

# Yasmine CLI - Actions

- specify how to modify or act on the XML document

**Actions:** Add, **Delete**, Select, Update

delete basenode object(s) at specified level

--action=delete --level\_network=IU --infiles=... (delete all stations from network)

--action=delete --level\_station=IU.ANMO --infiles=... (delete all channels from station)

# Yasmine CLI - Actions

- specify how to modify or act on the XML document

## **Actions:** Add, Delete, **Update**, Select

update the value of a basenode field or insert value if missing

- set with --field and --value
- where values of type double, int, string can be specified on the command line  
or else contained in a .yml file

```
--action=update --field=latitude --value=45 --level_station=IU.ANMO --infiles=...
```

```
--action=update --field=data_availability --value=yml:yml/dataavailability.yml --level_station=IU.ANMO  
--infiles=...
```

# Yasmine CLI - Actions

- specify how to modify or act on the XML document

**Actions:** Add, Delete, Update, **Select**

filter for basenode object(s) that match the SNCL at specified level

```
--action=select --level_station=IU.ANMO --infiles=... (filter for all station codes matching 'ANMO')
```

If no action is specified, the action will default to **update** if --field and --value are specified  
or to **select** otherwise

# Yasmine CLI - Other Options

--dont_validate	turn OFF StationXML validation on all inputs/outputs
--schema_version	ver {1.0, 1.1}
--show_fields	print out allowable --field, + --value combinations
--show-fields	print out allowable --field, + --value combinations
--plot_resp	plot all channel responses
--plot_dir path	path to dir to save plot responses
--loglevel log level	log level in {DEBUG, INFO, WARN, etc}

# Yasmine CLI - Examples

```
> python yasmine-cli.py --level_network=II --field=description --value='Network description' --infiles=...
> python yasmine-cli.py --level_station=II.* --field=latitude --value=34.97 --infiles=...
> python yasmine-cli.py --level_channel=II.ANMO.00.* --field=comments[0] --value=yml:/path/comment.yml
--infiles=...
> python yasmine-cli.py --level_network=* --action=add --from_yml=path/to/station.yml --infiles=...
```



**SAGE** NSF's Seismological Facility for the Advancement of Geosciences

IRIS



Pittsburgh, PA • June 14–16, 2022 • Organized by IRIS and UNAVCO

## 2022 SAGE/GAGE COMMUNITY SCIENCE WORKSHOP

