

# Introducing FDSN StationXML

StationXML support comes to Antelope

Celso Reyes  
AUG Rome, Italy

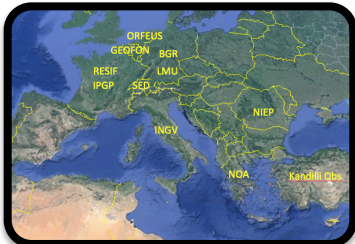
May 2016

# FDSN StationXML:

International Federation of Digital Seismograph Networks



... is an FDSN supported format  
(that will replace dataless SEED)



... is already available from  
major data centers



... can now be exported from  
Antelope CSS 3.0 Databases



## FDSN StationXML Schema

**purpose:** XML representation of the most important and commonly used structures of SEED 2.4 metadata.

**goal:** allow mapping to and from SEED 2.4 dataless SEED volumes

- minimal transformation or loss of information
- simplify station metadata.

Also, where lacking in SEED standard:

- Content and clarification has been added

### [fdsn-station+availability-1.0.xsd](#)

Extension of the base schema that includes time series data availability structures

### [Variations-FDSNSXML-SEED.txt](#)

An overview of major variations between SEED 2.4 and FDSN StationXML

## StationXML Versioning

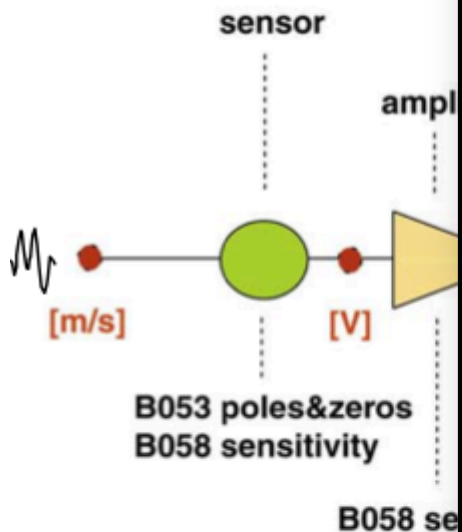
The `version` attribute of the schema definition identifies the version of the schema. This version is not enforced when validating documents.

The required `schemaVersion` attribute of the root element identifies the version of the schema that the document is compatible with. Validation only requires that a value is present but not that it matches the

# Station

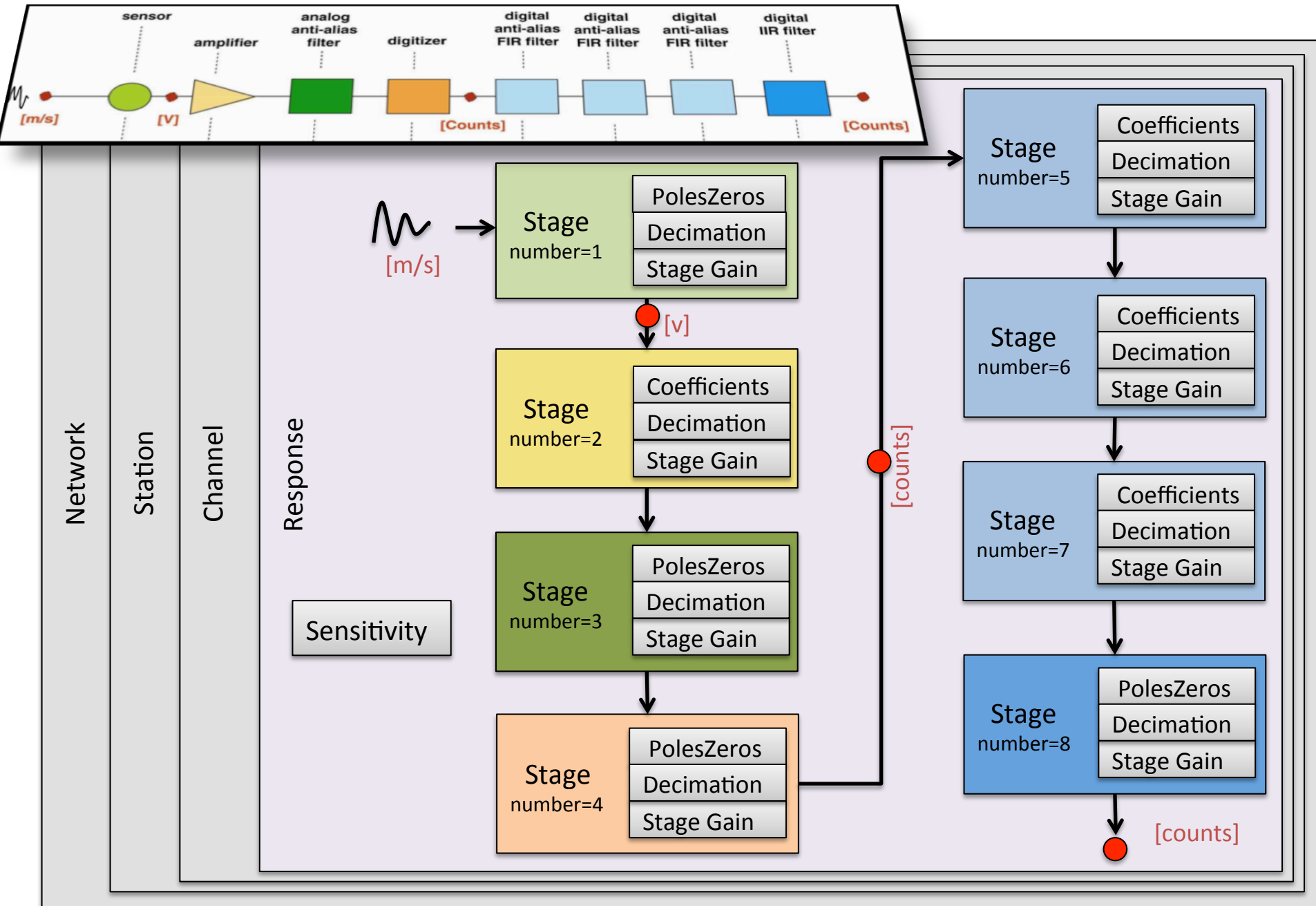
Sample:

```
0530382BΔ1007008Δ7.87395E+00Δ5.00000E-02ΔΔ3
Δ0.00000E+00Δ0.00000E+00Δ0.00000E+00Δ0.00000E+00
Δ0.00000E+00Δ0.00000E+00Δ0.00000E+00Δ0.00000E+00
-1.27000E+01Δ0.00000E+00Δ0.00000E+00Δ0.00000E+00ΔΔ4
-1.96418E-03Δ1.96418E-03Δ0.00000E+00Δ0.00000E+00
S-1.96418E-03-1.96418E-03Δ0.00000E+00Δ0.00000E+00
53-6.23500E+00Δ7.81823E+00Δ0.00000E+00Δ0.00000E+00
-6.23500E+00-7.81823E+00Δ0.00000E+00Δ0.00000E+00
```



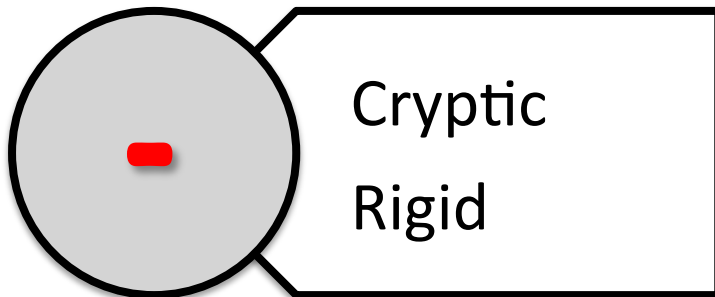
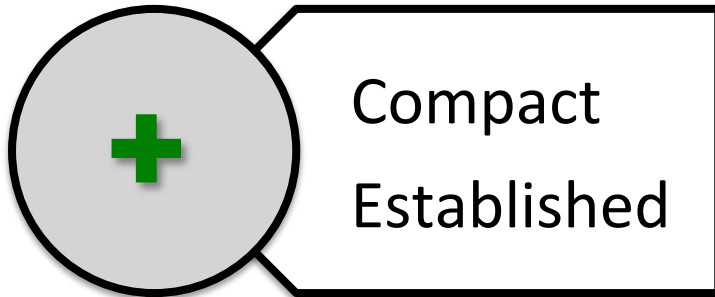
Note	Field name	Type	Length	Mask or Flags
1	Blockette type — 053	D	3	"###"
2	Length of blockette	D	4	"####"
3	Transfer function type	A	1	[U]
4	Stage sequence number	D	2	"##"
5	Stage signal input units	D	3	"###"
6	Stage signal output units	D	3	"###"
7	AO normalization factor (1.0 if none)	F	12	"-#.#####E-##"
8	Normalization frequency fn(Hz)	F	12	"-#.#####E-##"
9	Number of complex zeros	D	3	"###"
	REPEAT fields 10 — 13 for the Number of complex zeros:			
10	Real zero	F	12	"-#.#####E-##"
11	Imaginary zero	F	12	"-#.#####E-##"
12	Real zero error	F	12	"-#.#####E-##"
13	Imaginary zero error	F	12	"-#.#####E-##"
14	Number of complex poles	D	3	"###"
	REPEAT fields 15 — 18 for the Number of complex poles:			
15	Real pole	F	12	"-#.#####E-##"
16	Imaginary pole	F	12	"-#.#####E-##"
17	Real pole error	F	12	"-#.#####E-##"
18	Imaginary pole error	F	12	"-#.#####E-##"

# ...as StationXML

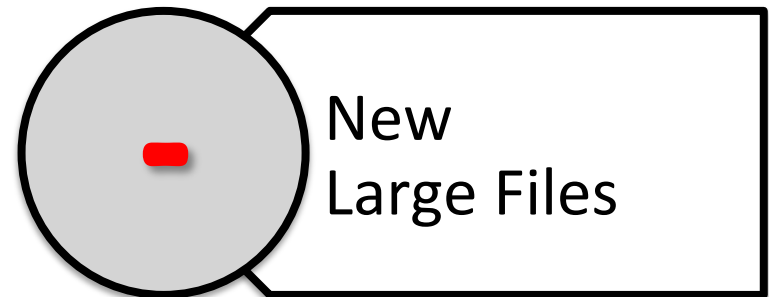
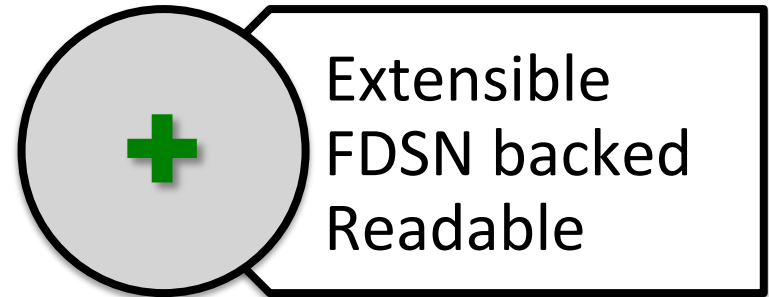


# Comparison

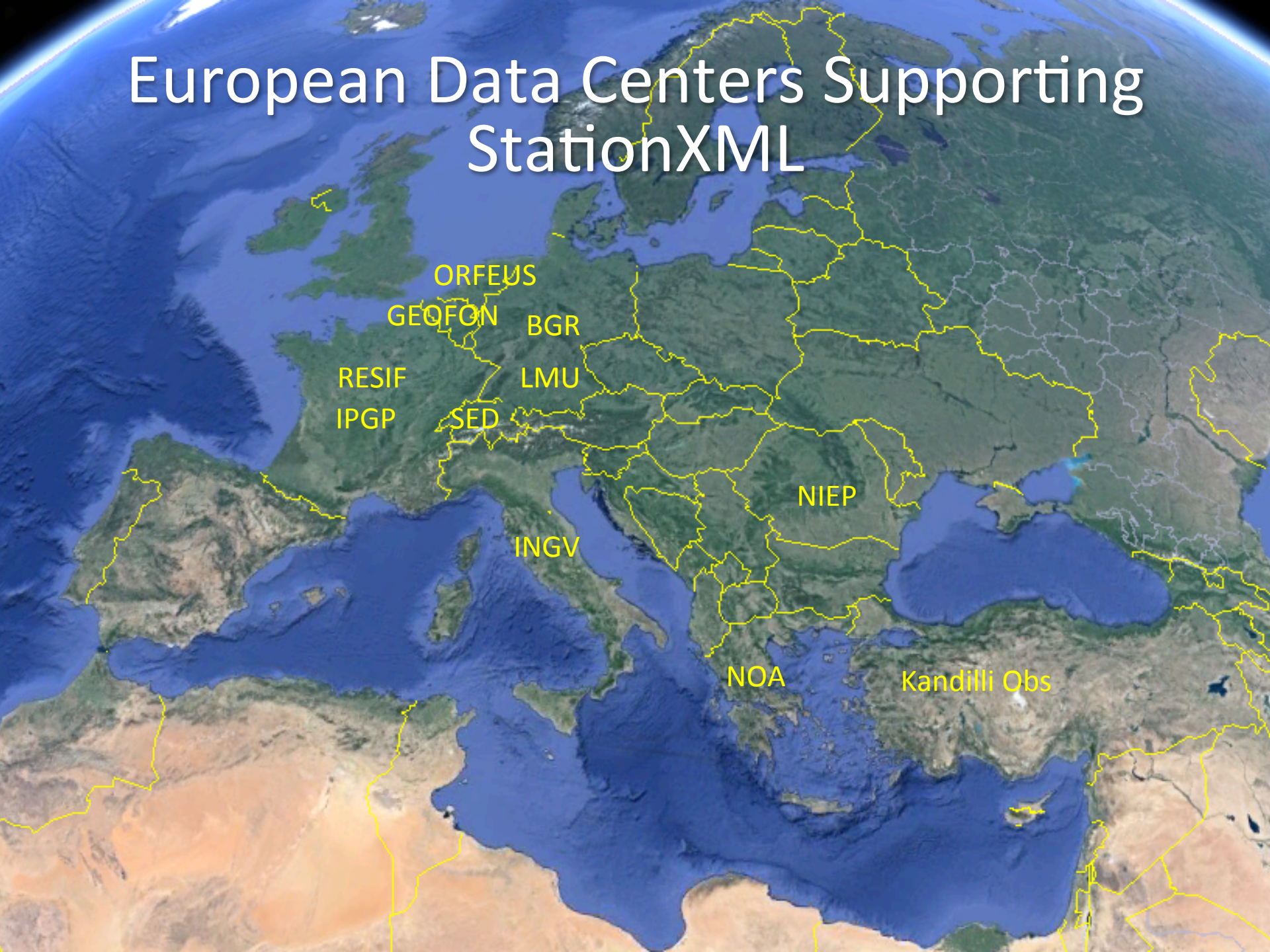
## Dataless SEED



## FDSN StationXML



# European Data Centers Supporting StationXML



ORFEUS

GEOFON

BGR

RESIF

LMU

IPGP

SED

NIEP

INGTV

NOA

Kandilli Obs

# Worldwide Data Centers Supporting StationXML

CNSN (?)

IRIS

NCEDC

SSIN (?)

ORFEUS

IRIS

INGV

INEP

NCA andilli Obs

?

USP

?

FDSN StationXML is already here

NZ GeoNet



# XML: eXtensible Markup Language

```
<Root>
```

```
  <Slide number="1">
```

```
    <Title>SPOILERS</Title>
```

```
    <Item type="definition">
```

```
      XML is eXtensible Markup Language
```

```
    </Item>
```

```
    <Item type="purpose">
```

```
      Provide organization to data
```

```
    </Item>
```

```
  </Slide>
```

```
</Root>
```

# XML: eXtensible Markup Language

```
<ROOT>
```

```
  <Network Code="A1">
```

```
    <Description>Comedy Net</Description>
```

```
    <Station code="ABOT"></Station>
```

```
    <Station code="CSTL"></Station>
```

```
  </Network>
```

```
  <Network Code="B2">
```

```
    <Description>Cartoon Net</Description>
```

```
    <Station code="TOM"></Station>
```

```
    <Station code="JRRY"></Station>
```

```
  </Network>
```

```
</ROOT>
```

# XML Elements: building blocks of XML

**<ROOT>**

**<Network Code="A1">**

**<Description>Comedy Net</Description>**

**<Station code="ABOT"></Station>**

**<Station code="CSTL"></Station>**

**</Network>**

**<Network Code="B2">**

**<Description>Cartoon Net</Description>**

**<Station code="TOM"></Station>**

**<Station code="JRRY"></Station>**

**</Network>**

**</ROOT>**

# XML Attributes: provide detail

```
<ROOT>
  <Network Code="A1">
    <Description>Comedy Net</Description>
    <Station code="ABOT"></Station>
    <Station code="CSTL"></Station>
  </Network>
  <Network Code="B2">
    <Description>Cartoon Net</Description>
    <Station code="TOM"></Station>
    <Station code="JRRY"></Station>
  </Network>
</ROOT>
```

# FDSNStationXML is defined by a SCHEMA

A **SCHEMA** defines:

- Elements and Attribute names
- Intent
- data types and ranges
- order of elements
- number of times each element may appear

# Sample of StationXML SCHEMA

```
<schema xmlns="http://www.w3.org/2001/XMLSchema" xmlns:fsx="http://www.fdsn.org/xml/
...>
<annotation><documentation>
  FDSN StationXML schema. Designed as an XML representation of SEED metadata, the
  schema maps to the most important and commonly used structures of SEED 2.4. When
  definitions and usage are underdefined the SEED manual should be referred to for
  clarification.
</documentation></annotation>
<!-- Root element -->
<element name="FDSNStationXML" type="fsx:RootType"/>
<!-- Type definitions -->
<complexType name="RootType">
  <annotation><documentation>
    Top-level type for Station XML. Required field are Source (network ID of the i
    sending the message) and one or more Network containers or one or more Station
  </documentation></annotation>
  <sequence>
    <element name="Source" type="string">
      <annotation>
        <documentation>Network ID of the institution sending the message.</documenta
      </annotation>
    </element>
  </sequence>
</complexType>
</schema>
[...]
```

<http://www.fdsn.org/xml/station/fdsn-station-1.0.xsd>

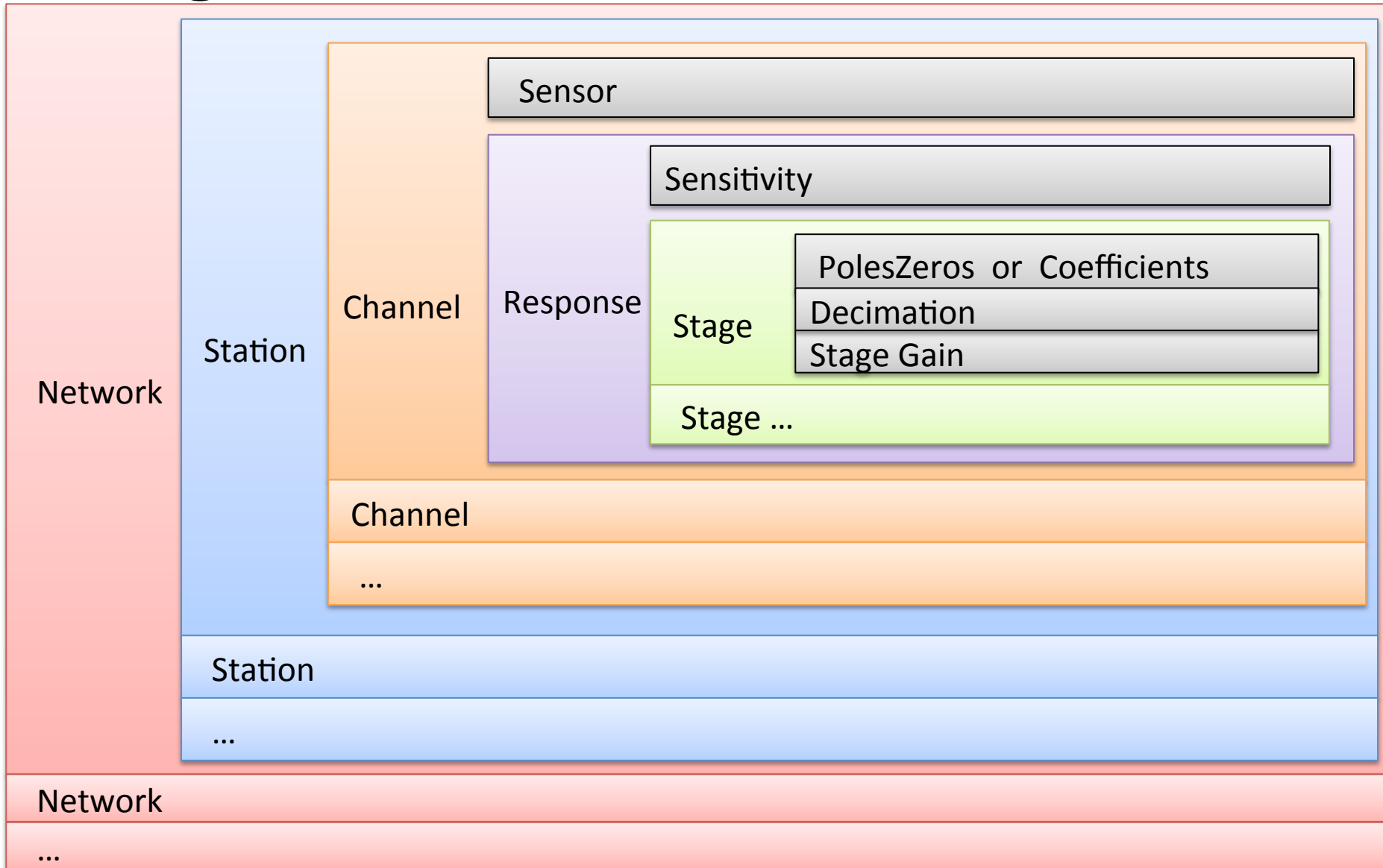
```

▶<ModuleURI>...</ModuleURI>
  <Created>2016-05-05T14:45:11.63006</Created>
▼<Network code="AZ" css30:netType="-">
  <Description>Anza Real-Time Broadband Network</Description>
  <SelectedNumberStations>38</SelectedNumberStations>
  ▼<Station code="BSAP" startDate="2011-08-12T00:00:00" endDate="2599-12-31T23:59:59.999">
    <Latitude>33.260200</Latitude>
    <Longitude>-116.322300</Longitude>
    <Elevation>160.000000</Elevation>
    ▼<Site>
      <Name>Borrego Springs Airport, CA, USA</Name>
    </Site>
    <Vault>-</Vault>
    <CreationDate>2011-08-12T00:00:00</CreationDate>
    <SelectedNumberChannels>3</SelectedNumberChannels>
    ▼<Channel code="HNE" locationCode="" startDate="2011-08-12T00:00:00" endDate="2599-12-31T23:59:59.999">
      ▼<Comment>
        <Value>345 269</Value>
      </Comment>
      <Latitude>33.260200</Latitude>
      <Longitude>-116.322300</Longitude>
      <Elevation>160.000000</Elevation>
      <Depth>9.000000</Depth>
      <Azimuth>90.000000</Azimuth>
      <Dip>0.000000</Dip>
      <SampleRate>250</SampleRate>
      ▼<CalibrationUnits>
        <Name>A</Name>
      </CalibrationUnits>
      ▼<Sensor css30:responseFrequencyBand="b">
        <Type>A</Type>
        <Description>Episensor 200 Hz 10 Volt per g/Reftek 72A-08 Datal</Description>
        <InstallationDate>2011-08-12T22:55:00.00000</InstallationDate>
        <CalibrationDate>2011-08-12T22:55:00.00000</CalibrationDate>
      </Sensor>
      <Response></Response>
    </Channel>
  ▶<Channel code="HNN" locationCode="" startDate="2011-08-12T00:00:00" endDate="2599-12-31T23:59:59.999">
    <Description>Anza Real-Time Broadband Network</Description>
    <SelectedNumberStations>38</SelectedNumberStations>
    ▼<Station code="BSAP" startDate="2011-08-12T00:00:00" endDate="2599-12-31T23:59:59.999">
      <Latitude>33.260200</Latitude>
      <Longitude>-116.322300</Longitude>
      <Elevation>160.000000</Elevation>
      ▼<Site>
        <Name>Borrego Springs Airport, CA, USA</Name>
      </Site>
      <Vault>-</Vault>
      <CreationDate>2011-08-12T00:00:00</CreationDate>
      <SelectedNumberChannels>3</SelectedNumberChannels>
      ▼<Channel code="HNE" locationCode="" startDate="2011-08-12T00:00:00" endDate="2599-12-31T23:59:59.999">
        ▼<Comment>
          <Value>345 269</Value>
        </Comment>
        <Latitude>33.260200</Latitude>
        <Longitude>-116.322300</Longitude>
        <Elevation>160.000000</Elevation>
        <Depth>9.000000</Depth>
        <Azimuth>90.000000</Azimuth>
        <Dip>0.000000</Dip>
        <SampleRate>250</SampleRate>
        ▼<CalibrationUnits>
          <Name>A</Name>
        </CalibrationUnits>
        ▼<Sensor css30:responseFrequencyBand="b">
          <Type>A</Type>
          <Description>Episensor 200 Hz 10 Volt per g/Reftek 72A-08 Datal</Description>
          <InstallationDate>2011-08-12T22:55:00.00000</InstallationDate>
          <CalibrationDate>2011-08-12T22:55:00.00000</CalibrationDate>
        </Sensor>
        <Response></Response>
      </Channel>
    </Station>
  </Network>

```

# FDSN StationXML (output) Sample

# Organization of FDSN StationXML





# FDSN StationXML:



... can now be exported from  
Antelope CSS 3.0 Databases

Through **db2stationxml**, available as:

- Command-Line program
- C++ Function

# Export from Antelope (Command Line)

- **db2stationxml** [options] databaseName

## where options can be:

- **v** verbose mode
- **o** *filename* output file
- **L** *level* detail level
- **p** *filename* alternate parameter file
- **s** *regexp* regular-expression to select stations

# Level (-L) Options

## **Network**

- Network description and [station count]

## **Station**

- Station description, coordinates, time ranges, channel count

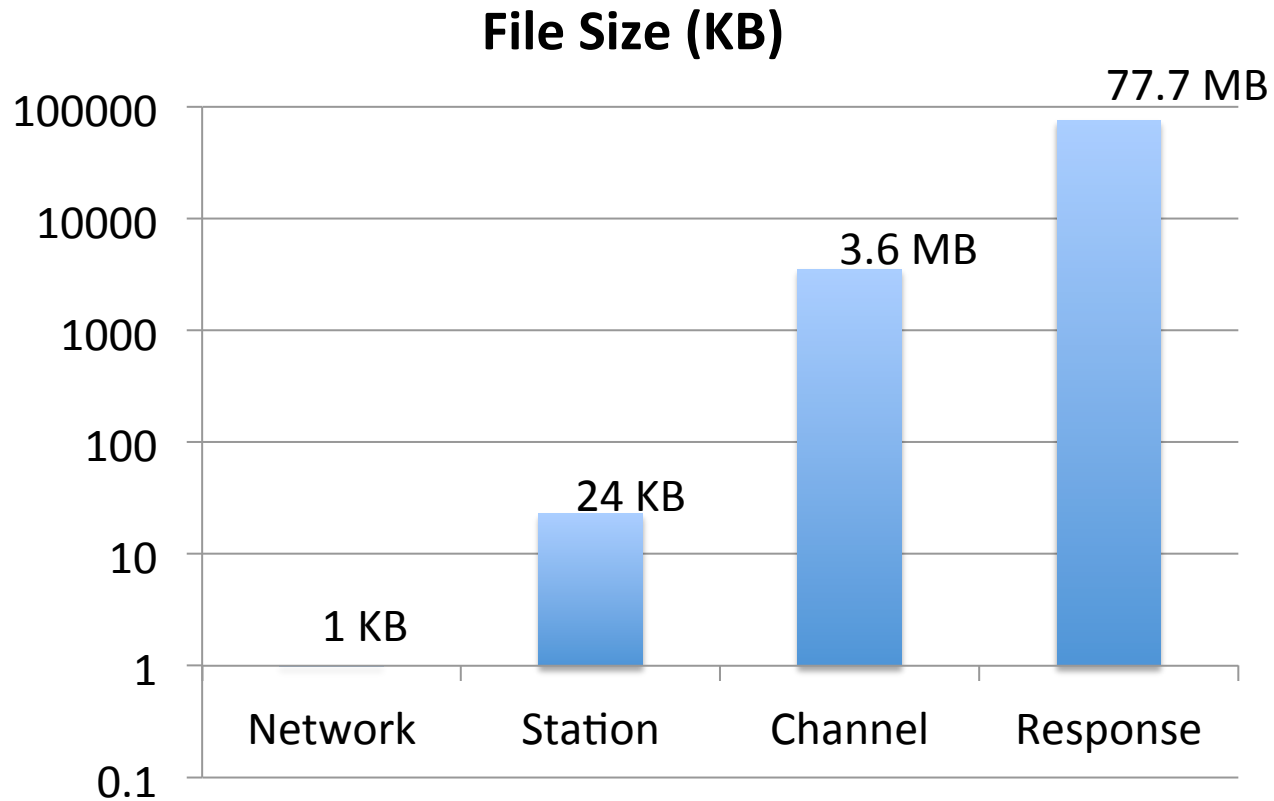
## ***Channel***

- Channel descriptions, coordinates, time ranges, instrument descriptions and channel sensitivity

## **Response**

- Complete channel response information

# File Size by Detail Level



\* Test Database containing 4 networks, 53 stations, 4089 channel epochs

# db2stationxml -L network dbname

celso — db2stationXML sample — -bash — 107x27

```
Knowitall:~ celso$ echo $TESTDB  
/Users/celso/work_brnt/testdata/dbmaster/anza
```

```
Knowitall:~ celso$ db2stationxml -L network $TESTDB
```

```
<FDSNStationXML xmlns="http://www.fdsn.org/xml/station/1" schemaVersion="1.0" xsi:schemaLocation="http://www.fdsn.org/xml/station/1 http://www.fdsn.org/xml/station/fdsn-station-1.0.xsd" xmlns:xsi="http://www.w3.org/2011/XMLSchema-instance" xmlns:css30="http://www.brnt.com/xml/station/css30" >  
  <Source></Source>  
  <Sender></Sender>  
  <Module>db2stationxml</Module>  
  <ModuleURI><!-- UNKNOWN --></ModuleURI>  
  <Created>2016-05-13T10:47:12.18524</Created>  
  <Network code="AZ" startDate="1970-01-01T00:00:00" endDate="2599-12-31T23:59:59" css30:netType="-" >  
    <Description>Anza Real-Time Broadband Network</Description>  
    <SelectedNumberStations>38</SelectedNumberStations>  
  </Network>  
  <Network code="PB" startDate="1970-01-01T00:00:00" endDate="2599-12-31T23:59:59" css30:netType="ww" >  
    <Description>Plate Boundary Observatory Borehole Seismic Network</Description>  
    <SelectedNumberStations>11</SelectedNumberStations>  
  </Network>  
  <Network code="YN" startDate="1970-01-01T00:00:00" endDate="2599-12-31T23:59:59" css30:netType="-" >  
    <Description>San Jacinto Fault Zone (SJFZ)</Description>  
    <SelectedNumberStations>4</SelectedNumberStations>  
  </Network>  
</FDSNStationXML>
```

# db2stationxml -L station -s "A.\*" dbname

```
Knowitall:~ celso$ db2stationxml -L station -s "A.*" $TESTDB
<FDSNStationXML xmlns="http://www.fdsn.org/xml/station/1" schemaVersion="1.0" xsi:schemaLocation="http://www.fdsn.org/xml/station/1 http://www.fdsn.org/xml/station/fdsn-station-1.0.xsd" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:css30="http://www.brtt.com/xml/station/css30" >
  <Source></Source>
  <Sender></Sender>
  <Module>db2stationxml</Module>
  <ModuleURI><!-- UNKNOWN --></ModuleURI>
  <Created>2016-05-13T11:03:48.56772</Created>
  <Network code="YN" startDate="1970-01-01T00:00:00" endDate="2599-12-31T23:59:59" css30:netType="-" >
    <Description>San Jacinto Fault Zone (SJFZ)</Description>
    <SelectedNumberStations>1</SelectedNumberStations>
    <Station code="ALCY" startDate="2012-03-23T00:00:00" endDate="2016-12-31T00:00:00" >
      <Latitude>33.507400</Latitude>
      <Longitude>-116.468600</Longitude>
      <Elevation>1756.000000</Elevation>
      <Site>
        <Name>ALCY, Anza, CA, USA</Name>
      </Site>
      <Vault>--</Vault>
      <CreationDate>2012-03-23T00:00:00</CreationDate>
      <SelectedNumberChannels>21</SelectedNumberChannels>
    </Station>
  </Network>
</FDSNStationXML>
```

# Export from Antelope, C++

```
#include "dbstationxml.h"
```

```
int db2stationxml(  
    Dbptr db ,           // open database  
    Pf *pf,            // open parameter file  
    Arr *subset_rule,  // regexp filters  
    ostream &out,      // output stream  
    int flags)         // level and verbosity
```

# Sample C++ Program

```
// includes go here
int main() {

    // Get Parameter File

    // Open a file to write to

    // Set criteria

    // set options ( level and verbosity)

    // Open Database

    // Request FDSN Station XML

    // Output Results

    // Clean up
}
```

*Include:*

*<stdlib.h>*

*<iostream>*

*<fstream>*

*"db.h"*

*"stock.h"*

*"dbstationxml.h"*



# Tables used by db2stationxml

- *network* - network information
- *snetsta* - SEED network and station mappings
- *site* - station details, such as location
- *sitechan* - channel details, such as instrument orientation
- *instrument* - instrument name and sample rate
- *calibration* - detector details
- *stage* - detailed response information

# Table Mappings

calibration instrument network schanloc sensor  
site sitechan snetsta stage responsefiles .

FDSNStationXML Source Sender Module ModuleURI Created  
Network code startDate endDate Description SelectedNumberStations  
Station code startDate endDate Latitude Longitude  
Elevation SelectedNumberChannels Site  
Channel code locationCode startDate endDate Latitude Longitude  
Elevation Depth Azimuth Dip SampleRate  
CalibrationUnits  
Sensor Type Description InstallationDate CalibrationDate  
Response  
InstrumentSensitivity Value Frequency InputUnits OutputUnits  
Stage number  
PolesZeros InputUnits OutputUnits PzTransferFunctionType  
NormalizationFactor NormalizationFrequency  
Pole number  
Real minusError plusError  
Imaginary minusError plusError  
Coefficients InputUnits OutputUnits  
Numerator minusError plusError  
Denominator minusError plusError  
Decimation InputSampleRate Factor Offset Delay Correction  
StageGain Value Frequency

## Parameter file:

`root_node`    `root_group`

```
root_group &Arr{  
  name      ROOT  
  
  attributes &Tbl{  
    elements &Tbl{  
      network_group CATEGORIZED_GROUP snetsta.snet  
    }  
  }  
}
```

```
network_group &Arr{  
  name      Network  
  
  attributes &Tbl{  
    Code      lookup      snetsta.snet  
    created   epoch       now()  
  }  
  elements &Tbl{  
    Description lookup      network.description  
    sta_group  CATEGORIZED_GROUP snetsta.fsta  
  }  
}
```

```
sta_group &Arr{  
  name      Station  
  
  attributes &Tbl{  
    code      lookup      snetsta.fsa  
  }  
  elements &Tbl{  
  }  
}
```

## snetsta table:

<u>snet</u>	<u>fsta</u>
A1	ABOT
A1	CSTL
B2	TOM
B2	JRRY

## network table:

<u>net</u>	<u>description</u>
A1	Comedy Net
B2	Cartoon Net

**Output:**

<ROOT>

**Parameter file:**

```

root_node    root_group

root_group &Arr{
  name      ROOT

  attributes &Tbl{}
  elements  &Tbl{
    network_group CATEGORIZED_GROUP snetsta.snet
  }
}

network_group &Arr{
  name      Network

  attributes &Tbl{
    Code      lookup      snetsta.snet
    created   epoch       now()
  }
  elements  &Tbl{
    Description lookup      network.description
    sta_group  CATEGORIZED_GROUP snetsta.fsta
  }
}

sta_group &Arr{
  name      Station

  attributes &Tbl{
    code      lookup      snetsta.fsta
  }
  elements  &Tbl{}
}

```

<i>snetsta.snet</i>
A1
B2

<b>snetsta table:</b>	
<u>snet</u>	<u>fsta</u>
A1	ABOT
A1	CSTL
B2	TOM
B2	JRRY

<b>network table:</b>	
<u>net</u>	<u>description</u>
A1	Comedy Net
B2	Cartoon Net

<ROOT>

**Output:**

**Parameter file:**

```

root_node    root_group

root_group &Arr{
  name      ROOT

  attributes &Tbl{
    elements &Tbl{
      network_group CATEGORIZED_GROUP snetsta.snet
    }
  }

network_group &Arr{
  name      Network

  attributes &Tbl{
    Code     lookup    snetsta.snet
    created  epoch     now()
  }
  elements  &Tbl{
    Description lookup  network.description
    sta_group  CATEGORIZED_GROUP snetsta.fsta
  }
}

sta_group &Arr{
  name      Station

  attributes &Tbl{
    code     lookup  snetsta.fsta
  }
  elements  &Tbl{}
}

```

*snetsta.snet*  
**A1**  
 B2

**snetsta table:**

<u>snet</u>	<u>fsta</u>
A1	ABOT
A1	CSTL
B2	TOM
B2	JRRY

**network table:**

<u>net</u>	<u>description</u>
A1	Comedy Net
B2	Cartoon Net

**Output:**

<ROOT>  
 <**Network**>

**Parameter file:**

```

root_node    root_group

root_group &Arr{
  name      ROOT

  attributes &Tbl{
    elements &Tbl{
      network_group CATEGORIZED_GROUP snetsta.snet
    }
  }

network_group &Arr{
  name      Network

  attributes &Tbl{
    Code      lookup      snetsta.snet
    created  epoch      now()
  }
  elements  &Tbl{
    Description lookup network.description
    sta_group  CATEGORIZED_GROUP snetsta.fsta
  }
}

sta_group &Arr{
  name      Station

  attributes &Tbl{
    code    lookup snetsta.snet
  }
  elements &Tbl{}
}

```

*snetsta.snet*  
**A1**  
 B2

**snetsta table:**

<u>snet</u>	<u>fsta</u>
A1	ABOT
A1	CSTL
B2	TOM
B2	JRRY

**network table:**

<u>net</u>	<u>description</u>
<b>A1</b>	Comedy Net
B2	Cartoon Net

**Output:**

<ROOT>  
 <Network **Code="A1"** **created="2016-05-20"**>

**Parameter file:**

```

root_node    root_group

root_group &Arr{
  name      ROOT

  attributes &Tbl{
  elements  &Tbl{
    network_group CATEGORIZED_GROUP snetsta.snet
  }
}

network_group &Arr{
  name      Network

  attributes &Tbl{
    Code      lookup      snetsta.snet
    created   epoch       now()
  }
  elements  &Tbl{
    Description lookup network.description
    sta_group CATEGORIZED_GROUP snetsta.fsta
  }
}

sta_group &Arr{
  name      Station

  attributes &Tbl{
    code     lookup      snetsta.snet
  }
  elements  &Tbl{}
}

```

*snetsta.snet*  
**A1**  
 B2

**snetsta table:**

<u>snet</u>	<u>fsta</u>
A1	ABOT
A1	CSTL
B2	TOM
B2	JRRY

**network table:**

<u>net</u>	<u>description</u>
A1	<b>Comedy Net</b>
B2	Cartoon Net

**Output:**

```

<ROOT>
  <Network Code="A1" created="2016-05-20">
    <Description>Comedy Net</Description>

```

**Parameter file:**

```

root_node    root_group

root_group &Arr{
  name      ROOT

  attributes &Tbl{
  elements  &Tbl{
    network_group CATEGORIZED_GROUP snetsta.snet
  }
}

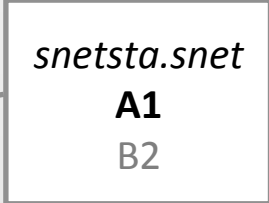
network_group &Arr{
  name      Network

  attributes &Tbl{
    Code      lookup      snetsta.snet
    created   epoch       now()
  }
  elements  &Tbl{
    Description lookup    network.description
    sta_group  CATEGORIZED_GROUP snetsta.fsta
  }
}

sta_group &Arr{
  name      Station

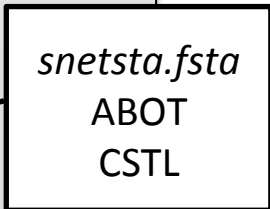
  attributes &Tbl{
    code     lookup      snetsta.fsta
  }
  elements  &Tbl{}
}

```



**snetsta table:**

<u>snet</u>	<u>fsta</u>
A1	ABOT
A1	CSTL
B2	TOM
B2	JRRY



**network table:**

<u>net</u>	<u>description</u>
A1	Comedy Net
B2	Cartoon Net

**Output:**

```

<ROOT>
  <Network Code="A1" created="2016-05-20">
    <Description>Comedy Net</Description>

```



**Parameter file:**

```

root_node    root_group

root_group &Arr{
  name      ROOT

  attributes &Tbl{
    elements &Tbl{
      network_group CATEGORIZED_GROUP snetsta.snet
    }
  }

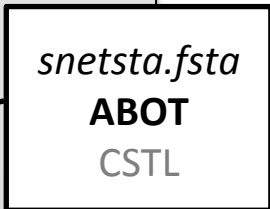
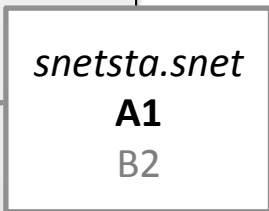
network_group &Arr{
  name      Network

  attributes &Tbl{
    Code      lookup      snetsta.snet
    created   epoch      now()
  }
  elements &Tbl{
    Description lookup      network.description
    sta_group  CATEGORIZED_GROUP snetsta.fsta
  }
}

sta_group &Arr{
  name      Station

  attributes &Tbl{
    code      lookup      snetsta.fsta
  }
  elements &Tbl{}
}

```



**snetsta table:**

<u>snet</u>	<u>fsta</u>
A1	ABOT
A1	CSTL
B2	TOM
B2	JRRY

**network table:**

<u>net</u>	<u>description</u>
A1	Comedy Net
B2	Cartoon Net

**Output:**

```

<ROOT>
  <Network Code="A1" created="2016-05-20">
    <Description>Comedy Net</Description>

```

# Parameter file:

```

root_node    root_group

root_group &Arr{
  name      ROOT

  attributes &Tbl{
    elements &Tbl{
      network_group CATEGORIZED_GROUP snetsta.snet
    }
  }
}

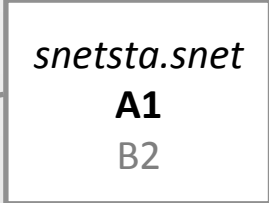
network_group &Arr{
  name      Network

  attributes &Tbl{
    Code      lookup      snetsta.snet
    created   epoch      now()
  }
  elements &Tbl{
    Description lookup      network.description
    sta_group CATEGORIZED_GROUP snetsta.fsta
  }
}

sta_group &Arr{
  name      Station

  attributes &Tbl{
    code      lookup      snetsta.fsta
  }
  elements &Tbl{}
}

```



**snetsta table:**

<u>snet</u>	<u>fsta</u>
A1	ABOT
A1	CSTL
B2	TOM
B2	JRRY



**network table:**

<u>net</u>	<u>description</u>
A1	Comedy Net
B2	Cartoon Net

Output:

```

<ROOT>
  <Network Code="A1" created="2016-05-20">
    <Description>Comedy Net</Description>
    <Station code="ABOT"

```

**Parameter file:**

```

root_node    root_group

root_group &Arr{
  name      ROOT

  attributes &Tbl{
  elements  &Tbl{
    network_group CATEGORIZED_GROUP snetsta.snet
  }
}

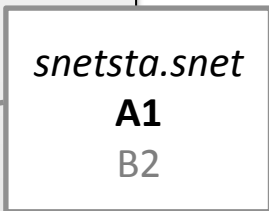
network_group &Arr{
  name      Network

  attributes &Tbl{
    Code     lookup    snetsta.snet
    created  epoch     now()
  }
  elements  &Tbl{
    Description lookup  network.description
    sta_group  CATEGORIZED_GROUP snetsta.fsta
  }
}

sta_group &Arr{
  name      Station

  attributes &Tbl{
    code     lookup    snetsta.fsta
  }
  elements  &Tbl{}
}

```



**snetsta table:**

<u>snet</u>	<u>fsta</u>
A1	ABOT
A1	CSTL
B2	TOM
B2	JRRY



**network table:**

<u>net</u>	<u>description</u>
A1	Comedy Net
B2	Cartoon Net

```

<ROOT>
  <Network Code="A1" created="2016-05-20">
    <Description>Comedy Net</Description>
    <Station code="ABOT""></Station>
    <Station code="CSTL"></Station>

```

**Output:**

**Parameter file:**

```

root_node    root_group

root_group &Arr{
  name      ROOT

  attributes &Tbl{
  elements  &Tbl{
    network_group CATEGORIZED_GROUP snetsta.snet
  }
}

network_group &Arr{
  name      Network

  attributes &Tbl{
    Code      lookup      snetsta.snet
    created   epoch      now()
  }
  elements  &Tbl{
    Description lookup      network.description
    sta_group CATEGORIZED_GROUP snetsta.fsta
  }
}

sta_group &Arr{
  name      Station

  attributes &Tbl{
    code     lookup      snetsta.fsta
  }
  elements  &Tbl{}
}

```

*snetsta.snet*  
**A1**  
 B2

**snetsta table:**

<u>snet</u>	<u>fsta</u>
A1	ABOT
A1	CSTL
B2	TOM
B2	JRRY

**network table:**

<u>net</u>	<u>description</u>
A1	Comedy Net
B2	Cartoon Net

**Output:**

```

<ROOT>
  <Network Code="A1" created="2016-05-20">
    <Description>Comedy Net</Description>
    <Station code="ABOT"></Station>
    <Station code="CSTL"></Station>
  </Network>

```

**Parameter file:**

```

root_node    root_group

root_group &Arr{
  name      ROOT

  attributes &Tbl{
    elements &Tbl{
      network_group CATEGORIZED_GROUP snetsta.snet
    }
  }
}

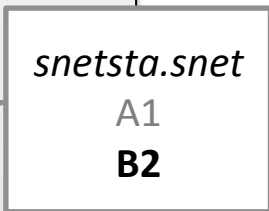
network_group &Arr{
  name      Network

  attributes &Tbl{
    Code      lookup      snetsta.snet
    created   epoch      now()
  }
  elements &Tbl{
    Description lookup      network.description
    sta_group  CATEGORIZED_GROUP snetsta.fsta
  }
}

sta_group &Arr{
  name      Station

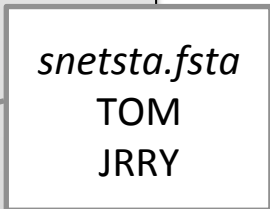
  attributes &Tbl{
    code      lookup      snetsta.fsta
  }
  elements &Tbl{}
}

```



**snetsta table:**

<u>snet</u>	<u>fsta</u>
A1	ABOT
A1	CSTL
B2	TOM
B2	JRRY



**network table:**

<u>net</u>	<u>description</u>
A1	Comedy Net
B2	Cartoon Net

**Output:**

```

<ROOT>
  <Network Code="A1" created="2016-05-20">
    <Description>Comedy Net</Description>
    <Station code="ABOT"></Station>
    <Station code="CSTL"></Station>
  </Network>
  <Network ... /Network>
</ROOT>

```

# sampling of programs that access StationXML

- ObsPy – Python Toolbox for seismology / seismological observatories
- Evalresp
  - <http://ds.iris.edu/ds/nodes/dmc/software/downloads/evalresp/>
- SOD – Standing order for data
- [ ... Future ? ... ]

# Feedback

- Uses for StationXML
- Import of FDSN StationXML?
- Feedback appreciated

# Networks Supporting StationXML

## Europe

- BGR Hannover, Germany [eida.bgr.de/fdsnws/station/1/](http://eida.bgr.de/fdsnws/station/1/)
- Boğaziçi University, Kandilli Observatory [eida-service.koeri.boun.edu.tr/fdsnws/station/1/](http://eida-service.koeri.boun.edu.tr/fdsnws/station/1/)
- ETHZ [arclink.ethz.ch/fdsnws/station/1/](http://arclink.ethz.ch/fdsnws/station/1/)
- GEOFON Program, GFZ [geofon.gfz-potsdam.de/fdsnws/station/1/](http://geofon.gfz-potsdam.de/fdsnws/station/1/)
- IPGP Data Center [eida.ipgp.fr/fdsnws/station/1/](http://eida.ipgp.fr/fdsnws/station/1/)
- INGV [webservices.rm.ingv.it/fdsnws/station/1/](http://webservices.rm.ingv.it/fdsnws/station/1/)
- LMU Munich, Germany [erde.geophysik.uni-muenchen.de/fdsnws/station/1/](http://erde.geophysik.uni-muenchen.de/fdsnws/station/1/)
- NIEP, Romania [eida-sc3.infp.ro/fdsnws/station/1/](http://eida-sc3.infp.ro/fdsnws/station/1/)
- NOA, Greece [eida.gein.noa.gr/fdsnws/station/1/](http://eida.gein.noa.gr/fdsnws/station/1/)
- ORFEUS Data Center [www.orfeus-eu.org/fdsnws/station/1/](http://www.orfeus-eu.org/fdsnws/station/1/)
- RESIF [ws.resif.fr/fdsnws/station/1/](http://ws.resif.fr/fdsnws/station/1/)

## North America

- IRIS Data Management Center [service.iris.edu/fdsnws/station/1](http://service.iris.edu/fdsnws/station/1/)
- Northern California Earthquake Data Center [service.ncedc.org/fdsnws/station/1](http://service.ncedc.org/fdsnws/station/1/)

## South America

- USP Seismological Center, Brazil [seisrequest.iag.usp.br/fdsnws/station/1/](http://seisrequest.iag.usp.br/fdsnws/station/1/)

<http://www.fdsn.org/webservices/datacenters/>



# For More Information:

- FDSN StationXML Schema home
  - <http://www.fdsn.org/xml/station/>
- FDSN StationXML Development Site:
  - <https://github.com/FDSN/StationXML>
- db2stationxml
  - Command line: db2stationxml(1)
  - C++ routine (and pf description): db2stationxml(3)

# SEED 2.4 metadata vs StationXML

## SEED

Volume Index [5]

Station Header [50]]

Channel [52]

(For each stage)

- Poles & Zeros[53]

- Coefficients [54]

- Decimation [57]

- Gain [58, 60]

## STATIONXML

<Network>

<Station>

<Channel>

<Response>

<Stage>

<PolesZeroes/>

</ ...

“The purpose of the FDSN StationXML schema is to define an XML representation of the most important and commonly used structures of **SEED 2.4** metadata”

# History

- 2007
  - StationXML schema released at SCEDC
- Roughly 2009 - 2013
  - FDSN StationXML is adapted from StationXML through collaboration with NCEDC, IRIS, and FDSN members.
- 2013
  - IRIS Releases FDSN-STATIONXML Web Services released by IRIS, soon followed by other Data Centers

# Sample C++ Program

```
#include <stdlib.h>
#include <iostream>
#include <fstream>

#include "db.h"
#include "stock.h"
#include "dbstationxml.h"

using namespace std;

int main()
{
    Dbptr    db;
    char     *database = "/opt/antelope/data/demo/gsn/dbmaster/gsn" ;
    Pf       *pf = "db2stationxml.pf";
    char     *outputname = "out.xml";

    if ( pfrear( pfname , &pf ) != 0 ) return -1;
    if ( dbopen( database, "r", &db ) < 0 ) return -1;

    Arr *subset_rules = newarr(1);
    setarr("snetsta.fsta", "/B.*");

    int result = -1;

    ofstream ofs;
    ofs.open ( outputname );
    if (ofs.good())
    {
        string myStationXML;
        result = db2stationxml( db , pf , subset_rules , myStationXML , verbose | STAXML_LEVEL_RESPONSE);
        if (result==0) ofs << myStationXML;
        ofs.close();
    }

    dbclose(db);

    return result;
}
```

# Schema Representation

**FDSNStationXML** Source Sender Module ModuleURI Created

**Network** code startDate endDate Description SelectedNumberStations

**Station** code startDate endDate Latitude Longitude

Elevation SelectedNumberChannels Site

**Channel** code locationCode startDate endDate Latitude Longitude

Elevation Depth Azimuth Dip SampleRate

**CalibrationUnits**

**Sensor** Type Description InstallationDate CalibrationDate

**Response**

**InstrumentSensitivity** Value Frequency InputUnits OutputUnits

**Stage** number

**PolesZeros** InputUnits OutputUnits PzTransferFunctionType

NormalizationFactor NormalizationFrequency

**Pole** number

**Real** minusError plusError

**Imaginary** minusError plusError

**Coefficients** InputUnits OutputUnits

**Numerator** minusError plusError

**Denominator** minusError plusError

**Decimation** InputSampleRate Factor Offset Delay Correction

**StageGain** Value Frequency

## Parameter file:

```
root_node    root_group
```

```
root_group &Arr{  
  name      ROOT  
  
  attributes &Tbl{  
  elements  &Tbl{  
    network_group CATEGORIZED_GROUP snetsta.snet  
  }  
}
```

```
network_group &Arr{  
  name      Network  
  
  attributes &Tbl{  
    Code      lookup      snetsta.snet  
    created   epoch      now()  
  }  
  elements  &Tbl{  
    Description lookup network.description  
    sta_group CATEGORIZED_GROUP snetsta.fsta  
  }  
}
```

```
sta_group &Arr{  
  name      Station  
  
  attributes &Tbl{  
    Code      lookup      snetsta.snet  
  }  
  elements  &Tbl{  
}
```

## snetsta table:

<u>snet</u>	<u>fsta</u>
A1	ABOT
A1	CSTL
B2	TOM
B2	JRRY

## network table:

<u>net</u>	<u>description</u>
A1	Comedy Net
B2	Cartoon Net

## Output:

```
<ROOT>  
<Network Code="A1" created="2016-05-20"  
<Description>Comedy Net</Description>  
<Station code="ABOT"></Station>  
<Station code="CSTL"></Station>  
</Network>  
<Network Code="B2" created="2016-05-20"  
<Description>Cartoon Net</Description>  
<Station code="TOM"></Station>  
<Station code="JRRY"></Station>  
</Network>  
</ROOT>
```