BEAM

Blackness Data System (BDS)

Instrument Response handling – 3.0.x – 2022-05-19

Table of Contents

1. Introduction

The BDS system stores channels of time sampled data from a number of measurement instrument types. As well as the normal earth movement seismic sensors there are also pressure, hydrophone and other sensor data.

This data is stored as a set of time domain numerical values on a per channel basis in sensor data files. For each channel there is Metadata, stored in a database, to describe the scaling, frequency response and data unit domain transformations of these data sets allowing the user to convert raw instrument data values into the measured quantities in SI units.

Note that the scaling and SI units are only valid after all individual response stage compensation algorithms have been applied to the raw data. Raw data from a seismometer may be in ADC counts describing metres displacement, metres/sec velocity or metres/second^2 acceleration. The response PoleZeros, FIR tables or Polynomial transformations can include the necessary data stream integrations to produce the data as a metres displacement quantity for example.

The BDS system stores, manages and retrieves instrument responses from its internal database. Its model of the response metadata encompasses that from all known seismic Metadata formats in the most generic way possible so that it can be used to store all of the information available as well as import and export this Metadata in the various different formats.

This document briefly describes this process and the external formats supported.

2. Calibration Data

The fundamental channel scaling Metadata is held in the Calibration database table. There are three parameters for each channel stored that provide the overall scaling and SI units of the data after the response has been compensated before and, optionally, three values for the raw data that can be used for simple graph or event finding algorithms. These are:

Type	*Name*	*Description*
Float64	samplingFrequency	The sample rate used in Hz
Float64	calibrationFrequency	The frequency that the CalibrationFactor value is valid for.
Float64	calibrationFactor	The scaling value to apply to the response processed data to normalise data to the units. This value is valid at the calibrationFrequency.
String	calibrationUnits	The measurement SI units such as “m”. These units should be the primary units such as “m” not “nm”. Current core units are: m, m/s, m/s^2, s, Pa, K. degrees (for wind direction sensors), V, A (for hydroacoustic state-of-health sensors)
Float64	rawCalibrationFrequency	The frequency that the RawCalibrationFactor value is valid for.
Float64	rawCalibrationFactor	The scaling value to apply to the raw data to normalise data to the units. This is valid at the rawCalibration frequency.
String	rawCalibrationUnits	The measurement SI units of the raw data such as “m”. These units should be the primary units such as “m” not “nm”.
Float64	depth	The depth of the sensor below ground level in meters
Float64	waterLevel	Elevation of the water surface in meters for underwater sites, where 0 is sea level
Float64	horizontalAngle	The Sensors channel placement horizontal angle in degrees clockwise from north
Float64	verticalAngle	The Sensors channel placement vertical angle in degrees degrees with zero = vertically up

The “rawCalibration*” attributes are provided so that the units of scaled raw data can be given for simple uses such as graphically displaying the raw data without having to apply the response inverse algorithms to the data to obtain the data transformed to particular unit domains with frequency and responses flattened.

3. Frequency Response Information

As well as the basic scaling information, for each channel there can be a set of amplitude/phase frequency response information and other necessary data transformations. The amplitude/phase responses are normally related to the physical instrument processing chain. For a seismic instrument the first response is typically that of the sensor of the instrument and normally has the most variation. Following this are typically the responses of the digitiser and anti-aliasing filters. The raw data should have inverse transformations applied in order to compensate for the frequency amplitude/phase response of the overall instrument. As an approximation just the response of the sensor can be used if wanted as the other response stages normally do not vary the signal response by significant amounts for some usages.

As well as frequency amplitude/phase response compensation, the response information can include transformations such as integration. This is often used for seismic movement channels as seismic sensors often measure velocity or even acceleration rather than displacement which is often required.

The frequency response and transformation information is normally provided either as a single overall response or as a set of frequency responses for each individual processing stage. There can be a frequency response for the sensor, the analogue anti-aliasing filter, the digitiser with digital anti-aliasing filter and digital decimation filter stages etc. Channels vary in the type of response transformation information and number of stages of frequency responses provided. To obtain the most accurate overall frequency response the user should transform the data with the inverse effect of all of the stage responses, although typically it is the stage 1 sensor's frequency response that is dominant.

There are exceptions to this, for example, certain infra-sound sensors in the IMS network, e.g. I52, for which the first stage of the response is not the pressure sensor but the wind-noise reduction system, represented as a FAP table, and the pressure sensor response is shown as the second stage.

With seismic data channels, typically the stage 1 response is in PoleZero format and will also include the necessary integration transformations (inverse of) in order to produce a set of displacement data samples from whatever physical quantity the sensor actually measures.

It is expected that multiple stage sets have the core instrument response as stage 1 and is named "instrument" as per the standard IDC naming scheme. Following this can be any number of named stages. Typically these will be named "digitizer".

The frequency response of each stage is defined by one of: a set of poles and zeros, a set of FIR coefficients, a FAP (frequency, amplitude, phase) table, or a polynomial (non-linear) response.

When a set of responses contain a non-linear Polynomial response there should also be a stage 0 Polynomial response. This provides an approximate channels overall response and can be used on its own to transform the raw data.

As well as stage 0 storing an overall Polynomial response the BDS has the option of having a response stage 0 named "Overall" to approximately describe the channel's overall response even when there is no Polynomial stage. This special, BDS stage 0 response will be used, if available, when exporting Metadata in file formats that only allow a single response to be defined. This stage 0 overall response can be configured manually or automatically by a software program. If this special BDS stage 0 overall response is not present the BDS will return the stage 1 response. If there are multiple response stages available then the stage 1 response will be returned with a warning. This special BDS stage 0 also allows a particular response to be returned for special cases such as channels that have the first stage describing a wind noise reduction system.

Each BDS channel can have multiple sets of responses stored. This could be due to changes based on time (such as Sensor or Digitiser changes).

The network:stations:channel:source fields are used to link the response to a particular BDS channel. The startTime and endTime fields are used for time related response changes. The multiple processing stages are defined using the stage and name fields.

Individual stages are defined by number starting at stage 1 which will normally be the response of the sensor itself and will often be in a PoleZero form. For each stage the BDS stores the following core information:

Name	***Type***	***Description***
id	Integer	Unique ID for this entry
startTime	String	The Start time
endTime	String	The End time
network	String	The Network name
station	String	The Station/Array name

channel	String	The Channel's primary identifier
source	String	The Source of the data this metadata is for
stage	Integer	The stage number of the response. 0 is for an override for overall response, 1, 2, 3 … for the individual stages.
name	String	The responses name (Overall, instrument, digitiser, anti-aliasing filter, post filter, wind-noise reduction system (WNRS), etc)
description	String	General description
type	String	The type of response (PoleZero, Polynomial, FAP or FIR)
data	Blob	PoleZero, Polynomial, FAP or FIR Coefficient data. This is stored in an encoded HEX/ASCII format. (We could store as a more readable ASCII)
gain	Double	Overall stage gain at gain frequency if known. 0 Otherwise.
gainFrequency	Double	Frequency that gain is valid for. 0 if unkown.
stageType	String	The stage type: A - Analog (rad/sec), B - Analog (Hz), C - Composite, D - Digital
symmetry	String	Symmetry for FIR coefficients (A = asymmetric, B = symmetric[odd], C = symmetric[even]). The BDS only stores full asymmetric cooeficients.
decimation	Double	Decimation performed post filter. if 0 unknown. Can be calculated if sampleRate is given for all stages.
decimationOffset	Double	Decimation sample offset
decimationDelay	Double	Decimation delay
decimationCorr	Double	Decimation correction performed
sampleRate	Double	The stage's sample rate if set. generally for Digitiser FIR filters. Set to 0.00 for analogue or unknown. Can be calculated if decimation value is given for all stages.
inputUnits	String	The units of the data as input from the perspective of data acquisition. After correcting data for this response, these would be the resulting units. Note that in output StationXML and SEED format this and outputUnits should contain meaningful values as some software (e.g. ObsPy) uses them to change the domain of a deconvolved response, e.g. from velocity to acceleration.
outputUnits	String	The units of the data as output from the perspective of data acquisition. These would be the units of the data prior to correcting for this response.
measured	Boolean	Set if response was a measured response.
lastUpdate	Timestamp	DateTime this record was last updated
Polynomial Response		
transferType	String	Transfer Type
approximationType	String	Polynomial approximation type, usually "MACLAURIN" – see SEED manual, page 85.
validFrequencyUnits	String	Valid Frequency Units

frequencyLowerBound	Double	Lower frequency bound for which the sensor response is valid, or zero if unspecified.
frequencyUpperBound	Double	Upper frequency bound for which the sensor response is valid, or if unspecified, the Nyquist frequency.
approximationLowerBound	Double	Lower bound of polynomial approximation, in units of the incoming signal (i.e. InputUnits, see below)
approximationUpperBound	Double	Upper bound of polynomial approximation, in units of the incoming signal (InputUnits)

Most of the legacy Metadata imported into the BDS does not have the full information for these fields. In the case of numerical fields a value of 0 means undefined and null strings also mean undefined.

3.1. Response Stages

- **Stage 0:** This is a special response stage. It is used to store a channels overall response in Polynomial format if any of the response stages is a Polynomial. In the BDS it can also store an "Overall" response in any format to be used when exporting Metadata in data formats that only allow one response stage to be exported. In principle the Stage 0 response is the product of the responses of all the stages of signal modification by the instrument and would normally be an approximation. Also the BDS stage 0 response could in principle be used to store an overall gain determined experimentally (e.g. by applying a calibration signal) rather than theoretically as the product of the manufacturer's specifications of the individual parts. With a typical seismic channel it might be equal to the sensor's stage 1 PoleZero response which is normally the dominant response and also contains the integration transformations to convert velocity or acceleration raw data to displacement samples. The special BDS stage 0 response should be named "Overall".
- **Stage 1:** This will normally be the dominant sensor's response and is often given in pole-zero form and named "instrument". For IMS infrasound sensors Stage 1 might be the wind-noise reduction system, named "wnrs" and represented as a FAP table, in which case the sensor is stage 2 and subsequent stage numbers are increased by one. For seismic channels it will often contain the integration transformations to convert velocity or acceleration raw data to displacement samples.
- **Stage 2:** This may be the analogue anti-aliasing filter's response. Often will be provided in pole-zero form. Would be named "instrument" in this case.
- **Stage 2 and up:** This will normally be the frequency responses of the digitiser and following digital anti-aliasing filters and decimation stages. Often will be provided in Finite-Impulse-Response (FIR) filter form and named "digitizer". Occasionally supplied as a FAP table.

It is expected that, for normal channels, there will be stages 1 and possible stages with numbers incrementally upwards. However there is no requirement for stage numbering to be incrementally sequential, there can be gaps in the stage numbering.

3.2. Response Types

The type field defines the type of response. At the moment PoleZero, Polynomial, FAP and FIR types are supported. The actual response data is stored in the data blob using the BOAP data serialising scheme. The BDS supports the following response types:

- **PoleZero:** This consists of a set of poles and a set of zeros each represented as complex numbers. The units are assumed to be radians/second [or: the units are radians/second or Hz as denoted by the "poleZeroUnits" variable.][or whatever it is called in the end]. An explanation of poles-and-zeros and other response types is given at http://docs.fdsn.org/projects/stationxml/en/latest/response.html

- **FAP:** This consists of a set of frequency, amplitude and phase values. The amplitudes are expected to be dimensionless.
- **FIR:** This consists of a set of "b" (numerator) coefficients and a set of "a" denominator coefficients with an error value for each. The BDS always stores and exports a complete set of coefficients. If a reduced odd or even symmetric set of coefficients is imported it is converted to a full set for storage.
- **Polynomial**: This consists of a list of coefficients (with error values), along with the frequency bounds and the polynomial approximation used (usually "MacLaurin"). It will often describe a non-linear response. For StationXml see: http://docs.fdsn.org/projects/stationxml/en/latest/reference.html#instrumentpolynomial and for SEED see the manual page 4, and Blockette 62 page 84. If a channels response contains a Polynomial response stage then there should also be an overall Polynomial response in stage 0.

3.3. Response Units

The overall channels signal units and its response stages input and output units describe the signal quantities of these inputs and outputs normally in SI units. There is some variation in the units given in old Metadata format files however. Where possible, the BDS converts the units of import file to SI unit name in the appropriate ASCII case so that BDS Metadata has units in SI format where ever possible.

The standard unit names used by the BDS are: m, m/s, m/s**2, s, Pa, K, counts, V, W, %, a, t, deg, Hz.

Note that the units should be stored with the correct letter case which is mainly lower case. They may be converted to upper case when exporting to certain formats.

There are also special unit names such as "counts" typically used for raw ADC output values etc.

3.4. Channel Data Types

To help distinguish the types of data a channel is associated with there is a dataType field on each Channel. This can be one of the following values: seismic, seismicUnknown, data, log, unknown, empty, ignore.

4. Response Gains

Each response stage has a gain and gainFrequency value as well as a set of parameters defining the amplitude/phase response in PoleZero, FIR, FAP or Polynomial form.

Normally the frequency response parameters define a frequency response normalised for a gain of 1 at the calibrationFrequency, but this is not always the case.

The gain and gainFrequency parameters are normally imported with the response parameters. However for some formats, like IDC format, their value is unknown and so are set to 0.

In the case of a PoleZero response the PoleZero response may not have a gain of 1. It is normalised to a gain of 1 by the PoleZero stages A0 normalisation factor at the calibrationFrequency. The BDS calculates the PoleZero A0 normalisation factor on the fly when exporting the Metadata or the user can calculate this from the set of PoleZeros. This needs to be applied whenever the PoleZero transformation is applied.

The overall gain of a stage is its gain value multiplied by the response parameter transformation (which typically has again of 1) at the gainFrequency. The gainFrequency typically is the same as the calibrationFrequency but may not be.

Normally, assuming each stages response transformation is normalised to a gain of 1 at calibrationFrequency, the product of all of the stages gain values should equal the overall calibrationFactor gain. However if some stages have their gain value given at a different frequency, or there response parameters do not result in a normalised gain at the calibrationFrequency then this may not be so.

Typically, with a seismic data channel, most of the stages, apart from the instrument sensors stage, have a fairly flat amplitude response wrt frequency. These stages typically have an overall gain of 1. Because of this the gain of the stage 1 response, typically a PoleZero response, equals the calibrationFactor.

When exporting Response metadata the stored stage gain/gainFrequency values are used. However if some of these values are not known then the stage gains will be set to 1 for all stages apart from stage 1. The stage 1 gain value will be hen set to the calibrationFactor.

5. Response Import and Export

In the long past there didn't used to be any well named and documented response Metadata file formats. More recently the SEED Metadata and StationXML formats have become more dominant.

This section describes the Metadata formats supported and any particular peculiarities when importing from or exporting to these formats.

In general it is assumed that Response Metadata is checked on import and any missing information added at that stage. Thus it is assumed that the Metadata in the BDS Metadata storage is valid and correct and will be fixed whenever issues are found. However the system allows some information to be not present or set. In the case of ASCII string based parameters a null string means this value is unknown. In the case of numeric values normally a value of 0 indicates unknown apart from a few specific cases.

On export the system will provide the Metadata it has. In the case of certain formats it has to manipulate or calculate certain aspects of the Metadata. Where this is done it is described below.

The BDS supports the following external representations of response Metadata:

Type	***Description***
STATIONXML	New standard format for a Stations channel's response Metadata. See: http://docs.fdsn.org/projects/stationxml/en/latest
SEED-METADATA SEED	SEED files supported for export of responses and sensor data with response metadata. See: SEEDManual_V2.4_2017-04-20.pdf
IMS-RESPONSE IMS-POLEZERO IMS-FAP IMS	IMS files supported for export. Described in document IDC-ENG-SPC-103-Rev.7.3, "Formats and Protocols for Messages", CTBTO, May 2017. Note all IMS channel responses are designed to transform the raw sensor data to the displacement domain (for seismometers) or pressure domain (for infrasound and hydroacoustic sensors).
SAC SAC-POLEZERO	This is a simple ASCII format for PoleZero response import and export, described in the SAC manual at http://ds.iris.edu/files/sac-manual/, see the page http://ds.iris.edu/files/sac-manual/commands/transfer.html#from-polezero.
IDC IDC-POLEZERO IDC-FAP	Generic IDC ASCII response for import/export. This is described in the informal document IDC_inst_resp.pdf.
ASCII ASCII-METADATA	Simple BDS ASCII format for responses and data.

Type	*Description*
RESPONSE	Old generic for import of any of the supported formats. Scans the file to determine the format. No longer supported.

Note that some of the older ASCII text based Response formats only support a single channel's response info. In these cases the BDS might generate an ASCII file with multiple Responses for multi channel exports. The user can split this file manually if needed or request the response file for one channel at a time.

5.1. Response Metadata Storage

The BDS can provide the response Metadata in various seismic formats as well as through its direct access API as used by its client programs. Where possible, for consistency, the Response Metadata obtained via these various routes should be the same. Thus the BDS stored Metadata should have as much information as possible so that little or no Metadata manipulation is required on export to any of the formats.
If some response Metadata is not known, strings are set to null strings and numeric values set to 0.

5.2. Response Handling for Particular Formats

Generally the BDS will simply export the response Metadata it has. In some cases it has to manipulate or calculate some parameters.

When importing response Metadata some older file formats do not provide the full set of response Metadata. In this case the import programs or the user should set these parameters to either unknown (blank strings or 0 numerical values in most cases) or to suitable approximations.

For example in some formats the input sampling rate and a decimation factor and offset are not provided. Where these are not available, they can be either set to 0 or set to the final sampling rate, one and zero respectively. Stage gains can either be set to 0 as unknown or set to one so that in principle the product of all the stage gains can be checked against the overall gain.

5.3. Response Import Fixes

During import, ideally, the input Metadata should be cleaned and validated before import with missing information set as needed. However the BDS will perform the following fixes using a function called bdsFixMetadataImport(). This will:

1. If there is a stage numbered 0 it will increment this and all subsequent stages by 1.
2. As it works through the ordered list of stages, if it finds a duplicate stage number, it will increment this and all subsequent stage numbers. (If the stage number has gone backwards the appropriate offset will be applied to this and all following stage numbers).
3. This will result in the stages numbers being set from 1 onwards with no duplicates. There may still be gaps in the stage numbering reflecting the positions of these in the original info.
4. If any FIR coefficients are imported in odd or even symmetric form they will be converted to a full set of coefficients for storage as type "A".
5. [Maybe we should also do stage gain fixing (as done at export) here ?]

5.4. Response Export Fixes

During export a certain set of default Metadata "fixes" will be applied to the Metadata and then some specific changes as needed by the export format in question. The generic fixes are:

1. If any stage has a gain of 0, then the system will set the gains of all stages to 1 except the first stage

which will be set to the calibrationFactor. All gainFrequency values will be set to the calibrationFrequency. (Maybe we should issue an error if some stages do have gain values other than 0 ?)

2. On export if the BDS finds any stage's inputUnits or outputUnits is not set then it will set the inputUnits and outputUnits of all stages to something that is normally acceptable. Typically this it sets all stages input and output units to "counts" apart from the fist stage where the inputUnits is set to the calibrationUnits value.
3. Where the sampleRate and/or decimation factor is not provided for any one of the stages, set the sampleRate for all stages to the calibration.samplingFrequency and set the decimation to 1.
4. PoleZero normalisation values will always be calculated on the fly.

5.4.1. StationXML

This is a fairly recent format and can provide a fairly complete representation of the Metadata for all seismic data channels currently available. It has a number of extra fields compared with very old seismic data and so some fields need to be generated from what information is available when exporting to this format.

On import all fields are stored in the BDS as they are apart from:

- calibration.calibrationFactor is set to 1.0 / InstrumentSensitivity.value
- calibration.verticalAngle is set to Channel.Dip + 90.0;
- The Channel.Dip value is set to the calibration.verticalAngle – 90.0.
- The InstrumentSensitivity.value is set to 1 / calibration.calibrationFactor.
- StationXML expects a block <InstrumentSensitivity> containing the overall gain and gain frequency of the system, to appear separately from the individual stages assuming no Polynomial stages are present.
- For single-stage responses it expects the stageGain to be equal to the InstrumentSensitivity.value and the gainFrequency to equal the frequency of the Instrument Sensitivity.
- Polynomial responses in StationXML do not include a value for ValidFrequencyUnits because StationXML requires this to be "Hz", hence ValidFrequencyUnits will be set to "Hz" on import.???On export in StationXML, if the stored ValidFrequencyUnits is "rad/s" the frequency bounds are divided by 2pi to convert their units to Hz, and the polynomial coefficients are divided by (2pi)**(order of the coefficient). OR no attempt is made to change the units of the polynomial or its frequency bounds to Hz ?? but a warning is issued to the user and a comment inserted into the StationXML file.

5.4.2. SEED

The SEED format is relatively old although much newer than some of the seismic Metadata formats used. It supports most of the seismic Metadata available in the BDS.

In general most of the Metadata stored in the BDS is written verbatim to the SEED format files. Particular changes include:

- Sensor angles and calibration factors are modified as:

```
if(ci.calibration.calibrationFactor < 0){
      ci.calibration.calibrationFactor = -ci.calibration.calibrationFactor;
      if((ci.calibration.verticalAngle == 0) ||
(ci.calibration.verticalAngle == 180)){
            dip = ci.calibration.verticalAngle + 90;
            azimuth = 0;
      }
```

```
            else {
                  dip = 0;
                  if(ci.calibration.horizontalAngle < 0)
                        azimuth = 360 + ci.calibration.horizontalAngle;
                  else
                        azimuth = ci.calibration.horizontalAngle;
                  azimuth += 180;
                  if(azimuth > 360)
                        azimuth -= 360;

            }
      }
      else {
            dip = ci.calibration.verticalAngle - 90;
            if(ci.calibration.horizontalAngle < 0)
                  azimuth = 360 + ci.calibration.horizontalAngle;
            else
                  azimuth = ci.calibration.horizontalAngle;
      }
```

- The SEED manual (page 78) states that for non-polynomial responses, the overall gain, in a blockette 58 with stage number 0, is necessary only if there are >1 response stages. Where there is only one stage, then it states that at least one blockette 58 must be present, with stage number either 0 or 1, although both may be present. Where a polynomial response is present as either a single stage or one of a number of stages, two blockettes 62 are needed, the second with stage number 0 and specifying the overall conversion of ground motion into digital samples. There should be no blockette 58 with stage number 0 in this case, as the gain of a polynomial response is not single-valued (SEED manual page 84; but we have an example violating the latter condition, with a single blockette 62 with stage number 1 followed by a blockette 58 with stage number 0).

5.4.3. IMS

Import of IMS Responses by the core BDS client programs is not currently supported.

On export to IMS format the BDS checks the units of any seismic data channels and if they have the calibrationUnits of “m” then it sets this to “nm” and multiplies its calibrationFactor by 1e9 to compensate.

Otherwise it produces the IMS format Metadata using the BDS Response information.

When the IMS format is selected, the Response Metadata has to be provided as a displacement response for seismic sensors (including the mass position sensors used in instrument state-of-health monitoring), or pressure for infrasonic or hydroacoustic sensors, to be compatible with IMS standards. The units are given in the appropriate slots in the output format. Note that only the response, not the digital data, is converted to displacement for output in IMS format, according to the specification of the IMS2.0 format (IDC-ENG-SPC-103-Rev7.3, 2017, page 120).

If the stage 1 (or BDS special Stage 0) response is a PoleZero response and the channel’s output units are in velocity or acceleration units, the BDS will add the appropriate number of Zero’s to the PoleZero Response Metadata to integrate the raw data from either velocity or acceleration values to displacement values.

If the stage 1 (or BDS special Stage 0) response is not a PoleZero response an export error is issued.

The BDS data file converter will use IMS comments to describe changes it has made. These consists of parentheses enclosed text with the opening parentheses in the second textual character column.

So the export changes algorithm when the channels dataType is “seismic” is:

```
If any PoleZero response has a gain set to 0 (unknown) then it calculate the
```

```
PoleZero gain at the calibrationFrequency using the bdsPoleZeroGain() function.

Where the stored units are velocity “m/s”:

- Add one zero at (0.0, 0.0);
- Divide the calibrationFactor by (2pi * calibrationFrequency)
- Multiply it by 10^9 to make it into nanometres/count;

Where the stored units are acceleration “m/s**2”:

- Add two zeros at (0.0, 0.0);
- Divide the calibrationFactor by (2pi * calibrationFrequency)^2
- Multiply it by 10^9 to make it into nanometres/count.
```

In the output PAZ2 line the “scale factor” is equal to the A0 normalisation factor for the modified poles-and-zeros divided by the modified calibrationFactor, so that the response at the calibrationFrequency is equal to the value of the calibrationFactor in nm or Pa.

5.4.4. IDC

An IDC format response Metadata file looks like:

```
theoretical  2    instrument    paz description
1.40568e+12
3
-9904.8 3786 0 0
-9904.8 -3786 0 0
-12507 0 0 0
0
 theoretical  4    digitizer    fir
30000
5
3.788775e-05 0
1.997269e-04 0
5.912768e-04 0
1.198337e-03 0
1.677196e-03 0
```

These are the specific PoleZero and FAP formats for export choice. If the response is stored in PoleZero format, it will be converted and exported as FAP if IDC-FAP is chosen, but if it is stored as a FAP then it cannot be exported in PoleZero format (the BDS is not equipped to derive a set of poles-and-zeros from a FAP table).

IDC-format response files do not contain the overall gain of a system or its calibration frequency (except, unreliably, in the header comments). At or before response import, these have to be provided within a “Calibration” object formed by the API or by using bdsMetadata or the BDS GUI.

The overall gain is specified as its inverse, “ncalib” (nominal calibration factor) in the IDC database table “static.instrument”, along with the calibration period “ncalper”. “Ground motion” in the IDC and IMS formats is always either ground displacement (specifically in nanometres) or pressure (specifically in pascals) (p. 120 of “Formats and Protocols for Messages” says micropascals but there is some inconsistency in IDC practice). They can be obtained from the IDC database table “static.instrument”, items “ncalib” and “ncalper”, or in IMS format from the IDC AutoDRM or VDMS (“nms_client”) services.

> *[NOTE – the above does not cover the case in which a response is imported as a FAP table in units other than displacement or pressure and export in IMS or IDC format is requested. It is easy to “integrate” a set of poles and zeros into the displacement domain, by adding one or two zeros at (0,0), but the BDS should not try to integrate a velocity or acceleration FAP response into*

displacement (too difficult). Hence the options when IMS/IDC output is requested are to serve the FAP table AND the accompanying calibration factor in the units of the FAP table (velocity or acceleration), or to refuse to serve IMS/IDC response output, on the grounds that it would be "out of spec".]

An IDC FAP table looks like:

```
5.000000E-02  9.651473E+08  1.303462E+01
6.147467E-02  9.735240E+08  1.026658E+01
7.558270E-02  9.790335E+08  7.895283E+00
9.292844E-02  9.824989E+08  5.838499E+00
1.142549E-01  9.844650E+08  4.018842E+00
1.404756E-01  9.852573E+08  2.364227E+00
1.727139E-01  9.850285E+08  8.069217E-01
2.123506E-01  9.837949E+08  -7.181713E-01
2.610836E-01  9.814743E+08  -2.275690E+00
```

The amplitudes are dimensionless multiplication factors. The BDS will export a FAP response from a single poles-and-zeros response but not from the combined result of a multi-stage response.

IDC-format response metadata files do not contain the full amount of response information the BDS can store. Some of this information, such as stage input/output units need to be either set on import or edited once in the BDS to suitable valid values. The metadata values will be set using some approximations:

1. The Stage Gains of individual stages except the first are set to one; the stage gain of the first stage is set to 1/calibrationFactor for the channel.
2. The gain frequency of individual FIR stages is set to zero but for initial pole-zero stages (representing the instrument and A-D converter analogue responses) the gain frequency is set to the calibrationFrequency for the channel;
3. Input and output units for the analogue stages and input units for the A-D converter stage are set to the units of the calibration factor, usually metres for seismic sensors and pascals for hydroacoustic and infrasound sensors. Output units for the A-D converter and input and output units for subsequent decimation stages are set to "COUNTS".

5.4.5. SAC

A SAC format response Metadata file looks like:

```
ZEROS 5
0.0000 0.0000
0.0000 0.0000
0.0000 0.0000
0.0000 0.0000
-3030.0000 0.0000
POLES 4
-0.3950 0.0000
-3.8330 4.9790
-3.8330 -4.9790
-42.6800 0.0000
CONSTANT 360.992094
```

The "CONSTANT", or normalisation factor, is obtained from bdsPoleZeroGain() function which matches the formula on the web page http://ds.iris.edu/files/sac-manual/commands/transfer.html#from-polezero with frequency set to calibrationFrequency. It is not divided by the calibrationFactor in BDS export.

Note the BDS does not attempt to derive poles-and-zeros from a response that was uploaded in another format, such as FAP. NOTE that although the BDS outputs all the zeros, if SAC reads the line “ZEROS n” but then reads fewer than n pairs of values, it will assume that the remaining zeros are at (0,0).

6. Response Metadata Notes

1. In the special case of an infrasound station the stage 1 response is often that for the wind-noise reduction system (WNRS), usually expressed as a FAP. (We may want to manipulate the Response Metadata on export so that stage 2, representing the infrasound sensor itself (usually a set of poles-and-zeros), is presented as stage 1).

6.1. A note on ObsPy expectations of SEED and StationXML response files

1. ObsPy (version 1.3) software for deconvolving instrument responses from SEED or StationXML files is based on 1990s C code provided by IRIS, and has restrictions accordingly (it converts files of both types into EVALRESP format to ease parsing). It is particularly fussy that stage gains not be zero and that the gain frequency for a pole-zero stage not be zero (it should be set to the calibration frequency of the channel). There is a fixed set of units abbreviations in the code (see the wrapper code in “response.py” in the obspy.core on the ObsPy Gitlab site). Units most likely to be needed for seismic channels are “M”, “M/S” and “M/S**2”, and for hydroacoustic and infrasound channels, “PA”. “V” for volts is also available, but not any units for temperature, electric current or wind direction. After the A-D converter stage the expected units are “COUNTS”. It issues warning messages stating that it is skipping stages if consistent input and output units are not provided for every stage.
2. Consistently with the SEED manual description of Blockette 57 (decimation), it tolerates decimation blocks with input sampling rate zero, decimation factor 1 and offset 0 in analogue or initial A-D stages that do not involve a decimation but need to have a time delay and time correction specified (these are both often zero).
3. In StationXML files it is intolerant of numbers with a decimal point where it is expecting integers, e.g. the decimation “offset”.

7. Export Programs

The bdsAdminGui, bdsUserGui, bdsDataAccess and bdsMetadata programs all support the export of responses. The user supplied format name defines the export format. Responses can be exported on their own or with the data in certain formats such as SEED and IMS.

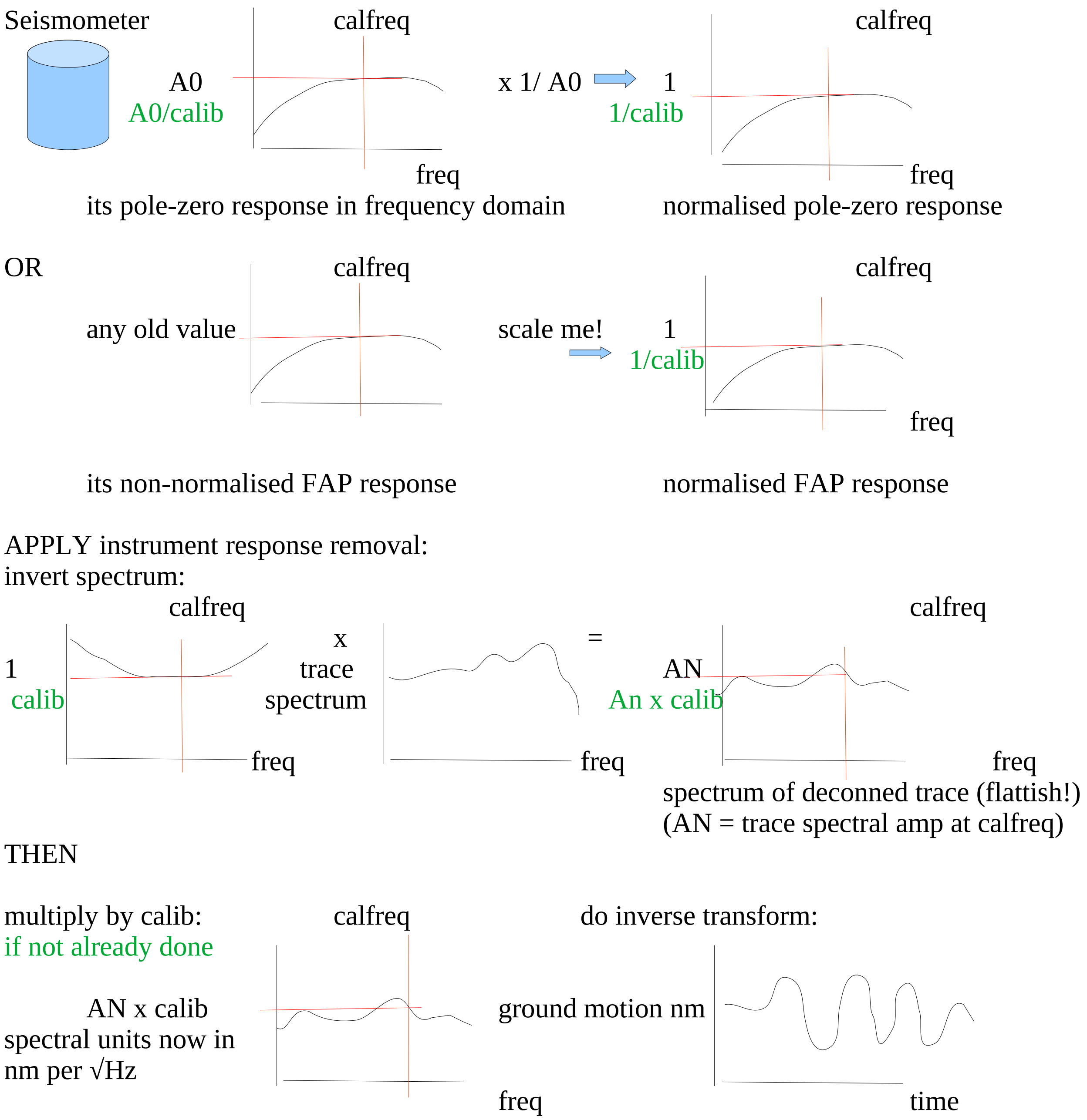
8. Import Programs

The bdsAdminGui, bdsUserGui, bdsMetadata and new bdsImportMetadata programs all support the import of responses from the various response file formats: SEED, StationXML, IDC format and SAC-polezero format. These programs will fail or issue warnings if the imported file deviates from the format specification, and the user might have to correct the file, for instance re-number misnumbered response stages in IDC-format multistage response files. The BDS programs will issue a warning of poles-and-zeros that violate causality by having positive real parts of poles.

9. Further Information

For further information please look at the BDS system documentation at: https://portal.beam.ltd.uk/support/blacknest.

10. Concepts of compensating for seismometer frequency response



Green is when the response has been pre-multiplied by the calibration factor.
The frequency axes are logarithmic.

Concepts of compensating for seismometer frequency response specified as a set of poles-and-zeros or a frequency-amplitude-phase (FAP) table. SP 2022-05-18.