Marine data and metadata standards and software

Wayne Crawford[1], Mechita Schmidt-Aursch[2] and the FDSN OBS Standards Action Group[3]
[1]Institut de Physique du Globe de Paris and CNRS, Paris, France; [2]Alfred-Wegener-Institut, Helmholtz-Zentrum für Polar- und Meeresforschung, Bremerhaven, Germany; [3]https://github.com/FDSN/OBS-standards

ABSTRACT

Mobile marine seismology instruments such as ocean bottom seismometers have some specific issues that are not integrated into seismology standards. We are developing international standards and software for mobile marine data and metadata. The standards are being defined through a Federation of Digital Seismometer Networks Action Group. The tools are at various stages of development or conceptualization: we present each one and its current state.

THE FDSN OBS-standards site

At https://github.com/FDSN/OBS-standards

Two main files

- standards.md: proposed standards
- software.md: existing and needed software

An issues page to add suggestions and corrections

- https://github.com/FDSN/OBS-standards/issues

CONTACT

Wayne Crawford
Email: crawford@ipgp.fr
Phone: +33 6 5151 1054
Website: http://www.ipgp.fr/~crawford

PLEASE ADD ANY INFORMATION, TOOLS OR IDEAS YOU HAVE OR KNOW OF.

STANDARDS

Standards developed by the FDSN Action Group on OBS Standards are available for review and comment at https://github.com/FDSN/OBS-standards.

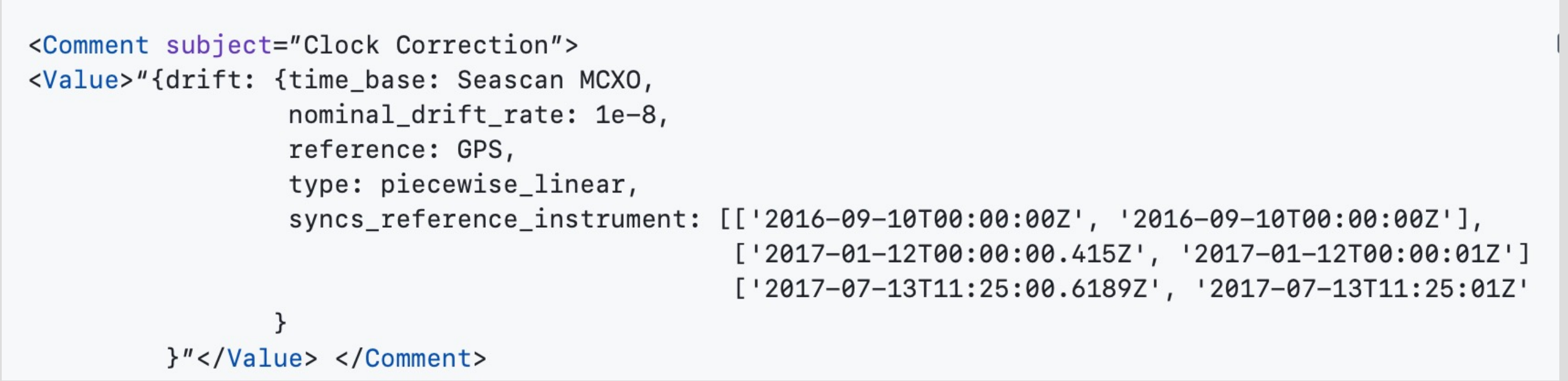
Metadata standards concern entering OBS-specific information into StationXML and miniSEED files. Data standards concern how to implement some necessary steps such as time adjustment. Proposed standards are:

StationXML standards

Entering clock correction information

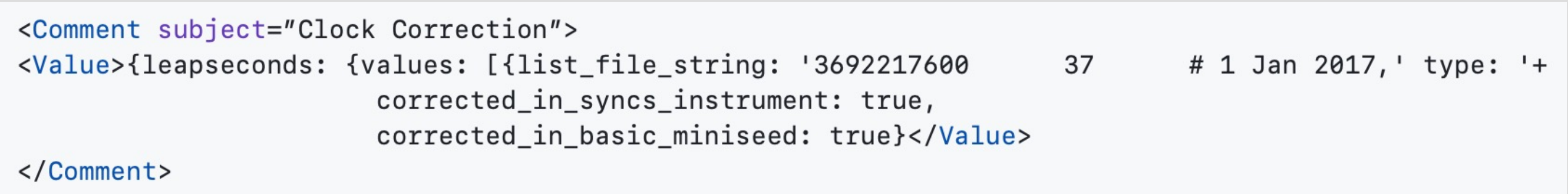
JSON-formatted strings in <Comment subject=clock_correction> elements

Time base drift



```
<Comment subject="Clock Correction">
<Value>"{drift: {time_base: Seascan MCXO,
                 nominal_drift_rate: 1e-8,
                 reference: GPS,
                 type: piecewise_linear,
                 syncs_reference_instrument: [['2016-09-10T00:00:00Z', '2016-09-10T00:00:00Z'],
                                               ['2017-01-12T00:00:00.415Z', '2017-01-12T00:00:01Z']
                                               ['2017-07-13T11:25:00.6189Z', '2017-07-13T11:25:01Z'
                }
        }"</Value> </Comment>
```

Leapseconds



```
<Comment subject="Clock Correction">
<Value>{leapseconds: {values: [{list_file_string: '3692217600      37      # 1 Jan 2017,' type: '+
                        corrected_in_syncs_instrument: true,
                        corrected_in_basic_miniseed: true}</Value>
</Comment>
```

<WaterLevel> element

If the deployment is in a lake, set to elevation of the lake's surface. Otherwise 0 or no element

Specifying unknown or non-standard orientation

code	<Dip>	<Azimuth>	Comment
1	0.0	0.0, minusError=180.0, plusError=180.0	
2	0.0	90.0, minusError=180.0, plusError=180.0	
3	90.0	0.0	If value DECREASES on upward motion
N	0.0	0.0	Azimuth must be within 5° of 0
E	0.0	90.0	Azimuth must be within 5° of 0
Z	-90.0	0.0	Dip must be within 5° of -90
D[H,G,O]	90.0	0.0	If value DECREASES on pressure increase
	-90.0	0.0	If value INCREASES on pressure increase

Data completeness

Set station `<startdate>` and `<enddate>` to expected data collection bounds.
Set channel `<startdate>` and `<enddate>` to actual data collection bounds

miniSEED standards for clock corrections

Type	miniSEED2	miniSEED3
NOT CLOCK CORRECTED	Data quality = D	Use the "Extra Header Field" (define an FDSN-standard, if possible) and allow ws-dataselect to use this field for selection
CLOCK CORRECTED	Data quality = Q	
RESAMPLED	Change channel name (to what?)	
leapseconds	Do not apply to "D" quality data	

Shared standards

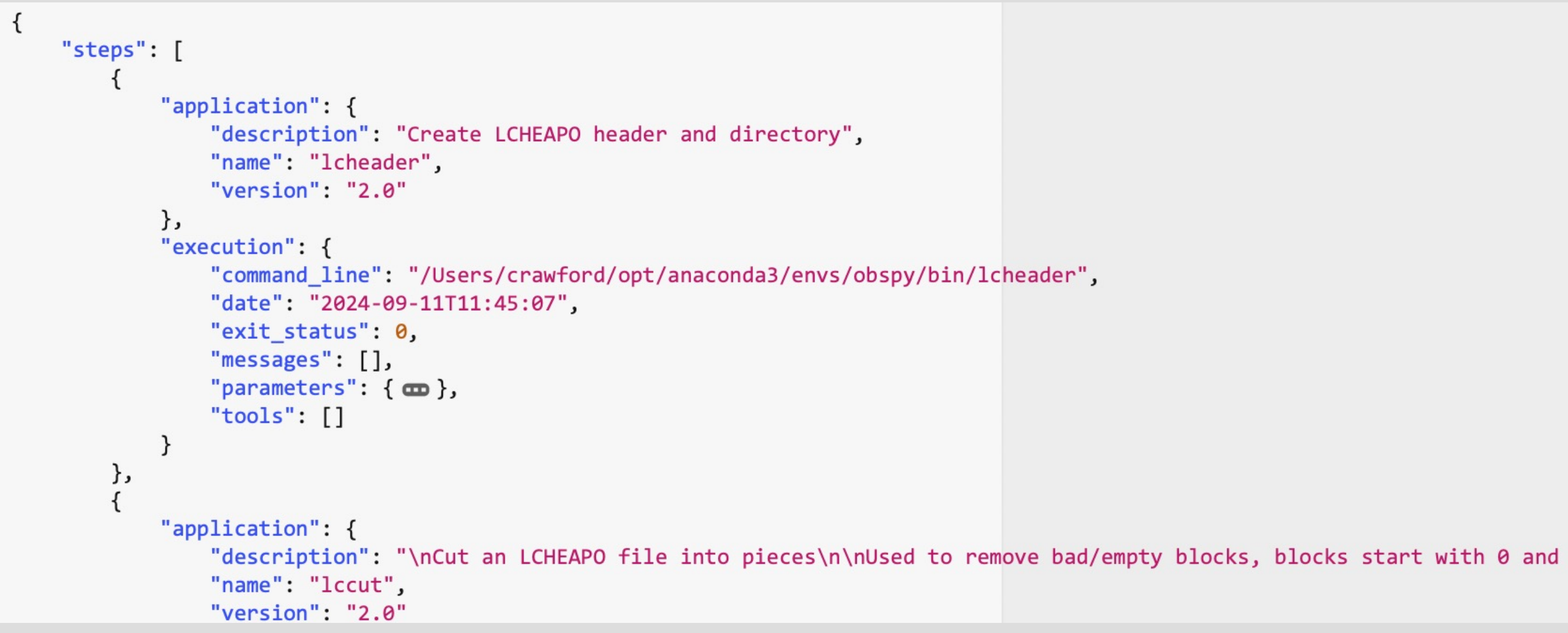
Source identifiers

The following source-subsource codes (see FDSN Source Identifiers documentation) should be used for the following types of sensor/data:

code	description
1	Unoriented seismometer, "N" channel equivalent
2	Unoriented seismometer "E" channel (+90 degrees from "1")
3	Seismometer/geophone with inverted vertical channel (positive voltage is down)
DH	Hydrophone
DG	Differential pressure gauge
DO	"Absolute" bottom pressure recorder

Processing information standards

Processing done on data files (from data download to delivery to the data center) should be recorded in text-based, structured files. No standard been established, the JSON process-steps format shown below is one option, which could possibly be embedded in the proposed "GeoCSV" add-on to StationXML



```
{
    "steps": [
        {
            "application": {
                "description": "Create LCHEAPO header and directory",
                "name": "lcheader",
                "version": "2.0"
            },
            "execution": {
                "command_line": "/Users/crawford/opt/anaconda3/envs/obspy/bin/lcheader",
                "date": "2024-09-11T11:45:07",
                "exit_status": 0,
                "messages": [],
                "parameters": {…},
                "tools": []
            }
        },
        {
            "application": {
                "description": "\nCut an LCHEAPO file into pieces\n\nUsed to remove bad/empty blocks, blocks start with 0 and
                "name": "lccut",
                "version": "2.0"
```

We gratefully acknowledge the support of EPOS, ORFEUS, Geo-Inquire and the FDSN

SOFTWARE

Mobile marine software should be openly available, well documented and work on standard data and metadata **(SeisComp Data Structure** and **StationXML**, at least). Identified software tasks are:

Creating standardized data and metadata

Creating Metatdata : obsinfo (and Yasmine?)

The open-source Pure Python **obsinfo** software (https://obsinfo.readthedocs.io/en/master/) allows the creation of marine-adapted StationXML files using modular text "information" files.

For people who already have their instrumentation responses in another format (RESP, STATIONXML), we are exploring whether it is better develop tools to import these information into obsinfo, or use existing tools((such as **YASMINE**) to add the marine-adapted fields to a StationXML files.

Correcting the instrument Clock : msmod

The Earthscope-distributed **msmod** software will be updated to include time correction (**piecewise linear**, **cubic spline** or **polynomial** fit) and, hopefully, leap-second correction.

Time corrections will be specified in a text file so as not to complicate msmod's interface.

Leap-second correction is possible using two msmod calls: the proposed modification will greatly simplify the correction.
Other Existing software: qedit, GIPP tools

Calculating/transforming data

The following tools should work on standardized data and metadata. These tools are not all publicly available, and their interfaces and levels of documentation are all different. We are looking into a multinational funding to bring all of these tools together under one open-source umbrella, with a consistent interface and documentation, as well as to define interface and documentation requirements for future software integration

Calculating clock drift

Clock drift with respect to a reference station can be calculated

- From ambient noise correlation
- From earthquake hypocenter time errors

Existing software: Hable et al. 2018, JustCorrel, ISP, obspy

Calculating sensor orientation

Sensor orientation can be calculated from located earthquake arrivals, ambient noise(?), active seismic sources, whales and/or ships.
Existing software: ppol, OrientPy, OBSIP orientation

Removing environmental noise

Removing noise from ocean waves and seafloor currents
Existing software: ATACR, tiskitpy, bruit-fm toolbox, ComPy.

Calculating seafloor compliance

Compliance is a measure of the deformation of the seafloor under ocean waves, and is used to determine subsurface structure.
Existing software: ATACR, tiskitpy, bruit-fm toolbox, ComPy

Extracting to active seismic data formats

This will allow us to save active seismic data on standard databases, and to separate of the tasks of locating the seismic sources and correcting the data. correction.
Existing software: obspy?

Other tools

Tools that do not use seismological data or metadata. They should be documented with the other tools and use the same download/packaging tools

Relocating stations

Locate instruments to within ~5m using acoustic interrogation.
Existing software: OBSrange