

ObsPy: A Python Toolbox for Seismology

Moritz Beyreuther

Department of Earth and Environmental Sciences (Geophysics)
Ludwig-Maximilians-Universität München

ROSE School
2009-12-01

ObsPy: A Python Toolbox for Seismology

Simplify Python programming for seismologists (no GUI). Functionality include:

- obspy.gse2 - GSE2 read and write support (CM6)
- obspy.sac - SAC read and write support
- obspy.mseed - MiniSEED read and write support
- obspy.xseed - Converter from Dataless SEED to XML-SEED and RESP files
- obspy.seishub - SeisHub client
- obspy.arclink - ArcLink/WebDC request client
- obspy.fissures - DHI/Fissures request client (experimental)
- ...

Why Python?

- Free and Open Source (replacement for MATLAB)
 - ▶ No licence for every process necessary
 - ▶ Available for third world countries
- Interpreter Language (steep learning curve)
 - ▶ Python shell for interactive learning
 - ▶ Easy debugging
 - ▶ Fast prototyping
- Direct access to existing shared libraries (C & Fortran)
- Large collection of open source scientific modules (matrix based)
- "Batteries included", large standard library

Object Structure

- stream object consists of multiple trace objects
- `trace = stream[i]` object of one contiguous data block
- `trace.data` contains data as C style contiguous memory block, easy passing to unmodified C and Fortran libraries
- `trace.stats` contains meta information as dictionary
 - ▶ `trace.stats.starttime` contains starttime UTCDateTime object
 - ▶ `trace.stats.sampling_rate` contains the sampling rate
 - ▶ ...

Standard for the Exchange of Earthquake Data (SEED)

- 1985 IASPEI formed working group on Digital Data Exchange
- 1987 FDSN (who felt responsible) adopted SEED format developed/proposed by USGS
- 1991 FDSN meeting introduce Data only SEED (MiniSEED)
- Standard archive format of seismological data centers

Details MiniSEED:

- Simpler to understand than SEED
- Storage as int32, float possible (not often used)
- Compression (STEIM1-3)
- Data stored in blocks, each block has its header → streaming

```
import obspy
st = obspy.read("miniseed.volume")
```


Group of Scientific Experts (GSE)

- 1976 GSE research monitoring and data analysis methods for verification of a nuclear test ban.
- format developed for GSETT-3 Group of Scientific Experts Third Technical Test
- Technical and personal foundation of the CTBTO (1996)

Details:

- Storage as int32
- Compression (CM6)
- Data stored in blocks, each block has its header → streaming

```
import obspy
st = obspy.read("gse.volume")
```

GSE2.0, Example

```
WID2 2004/06/09 20:05:59.850 RNON      Z RNON CM6      1200..  
STA2          -999.0000 -999.00000      -.999 ..  
DAT2
```

```
JTVCKPGGHUS19UMUO1762kKUTUI13MHUSH10VPkG13UF-UL10AUM1AUMUL  
FMI0UL13UOKkFUH13V+9kFUFSNUHR--2UF1HV6UO1TVJkKkMUTkKUQkEkF  
RVC16kIVETkMUIJ3-RUFJJIUGkKkIVI14V-9m2W5kSkTVP1D1VC14OVDkT  
UFV-11PUKH1-GKOKUEUGkQUFCkL8kFUI1kTUN+2kGSUJL9UF+kQI2CUI1C  
MSUSRUM13kFW0kFTkFkHV1UI118RVHJ15UOV-1P1VG1CBUI1-VFkK18VBk  
SUJUR1KV5JS+GF4UHkHR3V6S1CV-H-0TUK-FNJI85RUIkKMUHIUJkP4814  
JkK7HUKkFkE8GJGUSO15V9UGkJOkNUS0kMUSQKBR+DRkHV11kN8AIkLD3k  
CO8HLO6UPkJMFV8kHkRUT1FVPL1SVJG-kLFUGMV8kRkN31VP15mJWPUNm0  
6UPUT146DkMNUEUJ+TA14JUH9V01631kKUNR1DMNkFUGUMkILUIkL8US1-  
KUNKJ5kH9V912PAIUM1B9VJ1QV241IV1P2HK06VCm-GW71KkHUM0FPUKOk  
O106V0kI3UH13D5MUS17SUHAUJ1IUQUR1CUEkNV+UK1-UP1I8V1kIDkF7I  
-RV-QNULkNUH+12VFkRLUNkPUJ-4PKUKkOJUFK76kSDUHKG0K0-QA5MM3U
```

• • • •

SAC, Example

- Developed at Lawrence Livermore National Laboratory over the last 20-25 years.
- An interactive analysis tool for research seismologist
- SAC uses the SAC file format

Details SAC:

- Storage as float
- No compression
- Large required header

```
import obspy
st = obspy.read("sac.volume")
```


ArcLink

ArcLink, a distributed data request protocol for accessing archived waveform data. European Datacenter consisting of ORFEUS at KNMI, INGV, IPGP and GFZ.

```
from obspy.arclink import Client
from obspy.core import UTCDateTime

t = UTCDateTime("2009-08-24 00:20:03")
client = Client(host="webdc.eu", port=18001)

data = []
for station in ["RJOB", "RNON", "MANZ", "ROTZ"]:
    data += client.getWaveform("BW", station, "", "EHZ", t, t+30)
```

DHI/Fissures

DHI, a Data Handling Interface allowing users to access seismic data and meta data from IRIS DMC and other participating institutions (CORBA based)

```
from obspy.fissures import Client
from obspy.core import UTCDateTime

t = UTCDateTime("2009-08-24 00:20:03")
client = Client()

data = []
for station in ["BRNL", "PMG", "MORC", "DSB"]:
    data += client.getWaveform("GE", station, "", "EHZ", t, t+30)
```

www.obspy.org

Key Features Ensuring Continuation of ObsPy

- Test-driven development (TDD), currently 177 unit tests
- Modular structure
- Reliance of well-known third-party tools (numpy, scipy, matplotlib)
- Reusing well established existing code, e.g. libmseed, GSE_UTI
- Platform independence (Win, Mac, Linux) and tested
- Free, open source (and available from the very beginning)
- Automatic generated API documentation
- Community webpage www.obspy.org containing: tutorials, installation instructions, complete source code, ...

XML-SEED

Introduced by Tsuboi, Tromp and Komatitsch (2004)

- Converter from:
 - ▶ Dataless SEED to XML-SEED and vice versa
 - ▶ Dataless SEED to RESP files
- Tested against complete:
 - ▶ ORFEUS Dataless SEED archive
 - ▶ IRIS (US) Dataless SEED archive
 - ▶ ArcLink requests

```
from obspy.xseed import Parser
```

```
sp = Parser()
sp.read("data/dataless/bw/dataless.seed.BW_MANZ")
sp.writeRESP(folder="BW_MANZ", zipped=False)
sp.writeXSEED("dataless.seed.BW_MANZ.xml")
```

XML-SEED

```
000001V 010009402.3121970,001,00:00:00.0000~2038,001,00:00:00.0000~
2009,037,04:32:41.0000~ BayernNetz~~0110032002RJOB 000003RJOB 000008
...
```

```
<?xml version='1.0' encoding='utf-8'?>
<xseed version="1.0">
  <volume_index_control_header>
    <volume_identifier blockette="010">
      <version_of_format>2.4</version_of_format>
      <logical_record_length>12</logical_record_length>
      <beginning_time>1970-01-01T00:00:00</beginning_time>
      <end_time>2038-01-01T00:00:00</end_time>
      <volume_time>2009-02-06T04:32:41</volume_time>
      <originating_organization>BayernNetz</
        originating_organization>
    ...
```