

# ObsPy

A Python Framework for Seismology

mail: [devs@obspy.org](mailto:devs@obspy.org)  
twitter: @obspy  
github: obspy  
www: <https://obspy.org>

Tobias Megies<sup>1</sup>, Lion Krischer<sup>2</sup>, Derrick JA Chambers<sup>3</sup>,  
Tom Eulenfeld<sup>4</sup>, Calum John Chamberlain<sup>5</sup>, Thomas Lecocq<sup>6</sup>, Robert Barsch<sup>7</sup> & The Obspy Development Team

<sup>1</sup>Ludwig Maximilians University of Munich, Germany <sup>2</sup>ETH Zürich, Switzerland <sup>3</sup>Centers for Disease Control and Prevention, WA, USA <sup>4</sup>Friedrich Schiller University of Jena, Germany  
<sup>5</sup>Victoria University of Wellington, New Zealand <sup>6</sup>Royal Observatory of Belgium, Brussels, Belgium <sup>7</sup>EGU Office, Munich, Germany

## ObsPy in a Nutshell

Python combines the power of a fullblown programming language with the flexibility and fast code development of an interactive scripting language. Its extensive standard library and large variety of freely available high quality scientific modules cover most needs in developing scientific processing workflows.

ObsPy extends Python's capabilities to fit the specific needs that arise when working with seismological data. It a) provides **read and write support** for all of the most important waveform, station and event metadata formats b) enables **direct access to all important data centers**, web services and databases to easily retrieve waveform data and station/event metadata and c) comes with a continuously growing **powerful signal processing toolbox** that covers all everyday tasks in seismological analysis.

In combination with mature and free Python packages like NumPy, SciPy, Matplotlib, IPython/Jupyter, sympy, pandas and scikitlearn, ObsPy makes it possible to **develop complete seismological processing workflows**, ranging from reading locally stored data or requesting data from one or more different data centers via signal analysis and data processing to visualization in GUI and web applications, output of modified/derived data and the creation of publication quality figures. All functionality is **extensively documented** and the online ObsPy **Tutorial and Gallery** give a good impression of the wide range of possible use cases. ObsPy is tested and running on Linux, Mac OS X and Windows and comes with installation routines for all platforms. ObsPy is developed in a test-driven approach and is available under the LGPLv3 open source licence.

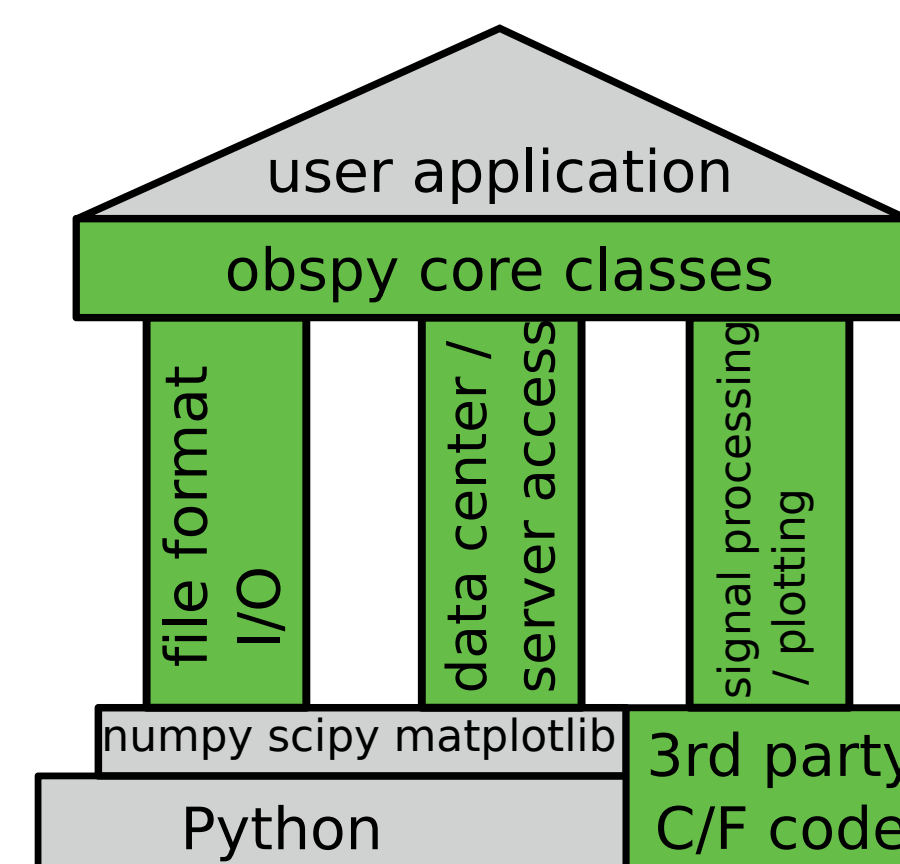
Users are welcome to request help, report bugs, propose enhancements or contribute code via either the user mailing list or the project page on GitHub.

## What Can I Do With ObsPy?

Easy to use helper functions to access local files and online data centers give quick access to all data necessary for seismological data analysis.

All acquired information is exposed to the user in ObsPy's core classes that handle waveform data, station and event metadata in a unified, consistent fashion, regardless of the data source. This makes it easy to combine data from different sources in unified work flows, both interactively and automated.

ObsPy's core classes have many convenience routines for signal processing directly attached for quick, reproducible and well tested execution of common processing tasks.

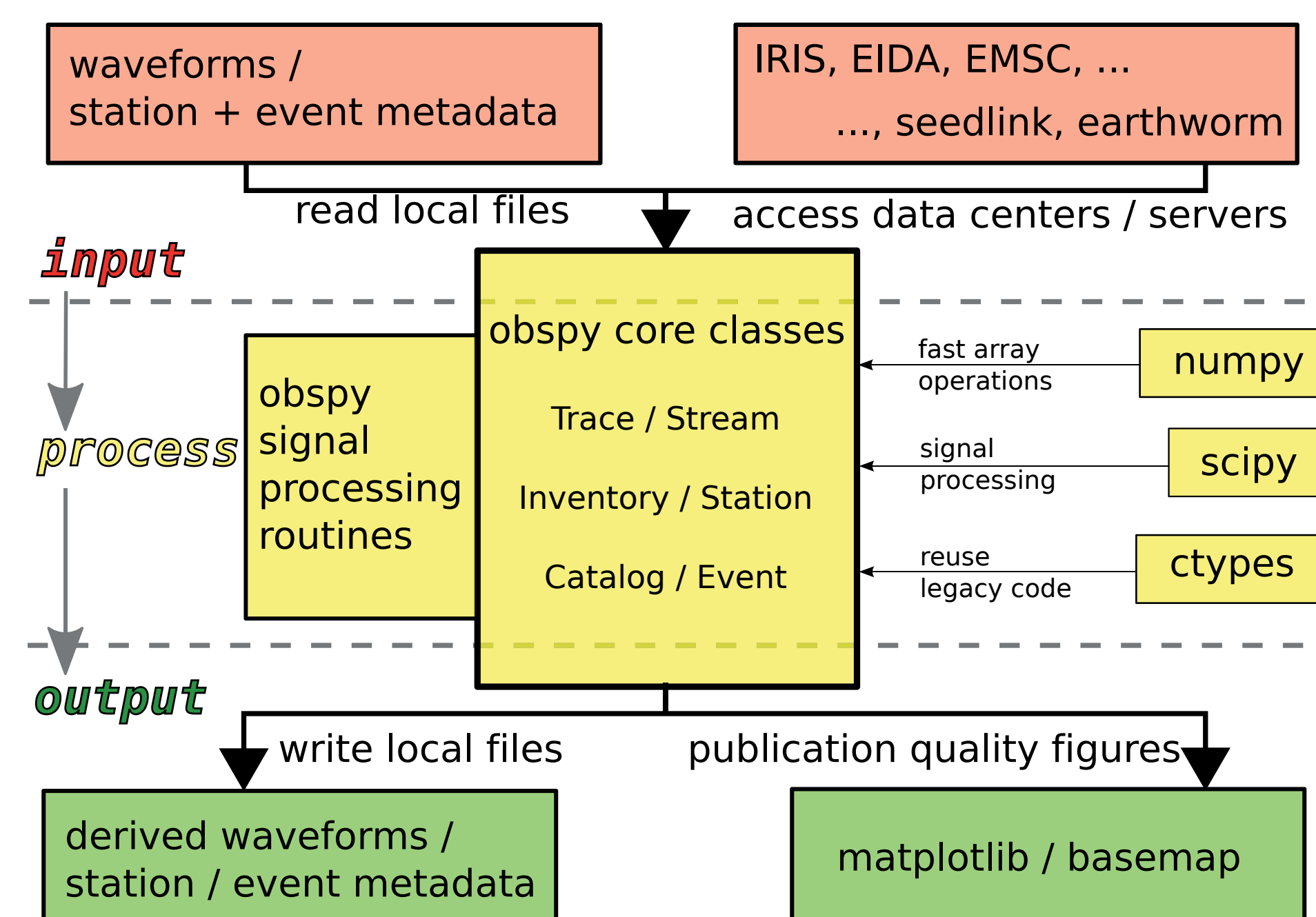


## Supported File Formats

**WAVEFORMS** AH, CSS, Guralp GCF, GSE1/2, KINEMATICS EVT, KNET, MiniSEED, NNSA KB Core, PDAS, RG16, REFTEK 130, SAC, SEISAN, SEG-2, SEG-Y, SH ASC/Q, SLIST, SU, TSPAIR, WAV, WIN, Y

**STATION METADATA** ArcLink XML, CSS, KML, SACPZ, SeiscompML, SH EVT, Shapefile, FDSN StationTXT, Station-XML, XSEED, SEED, RESP

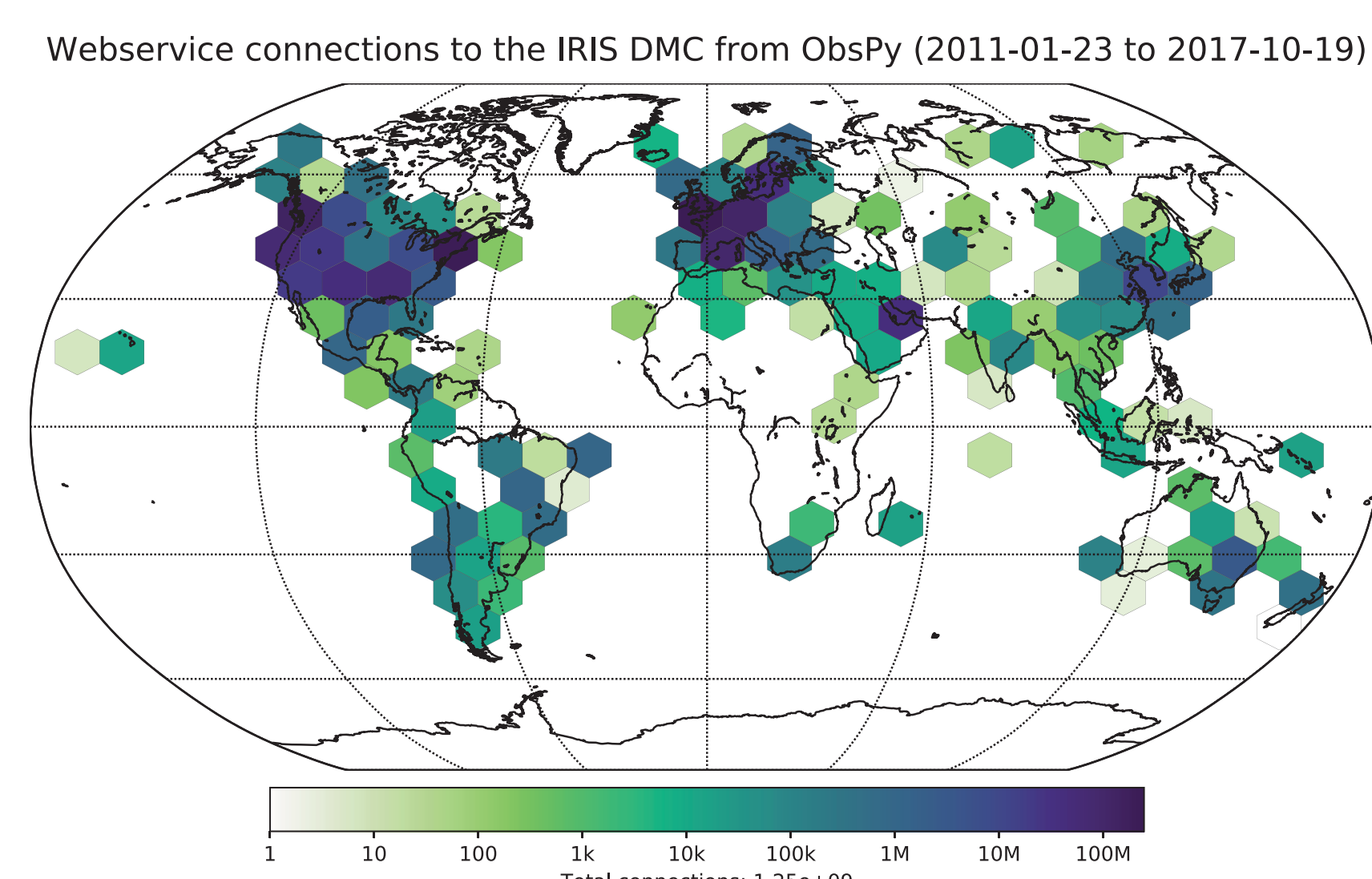
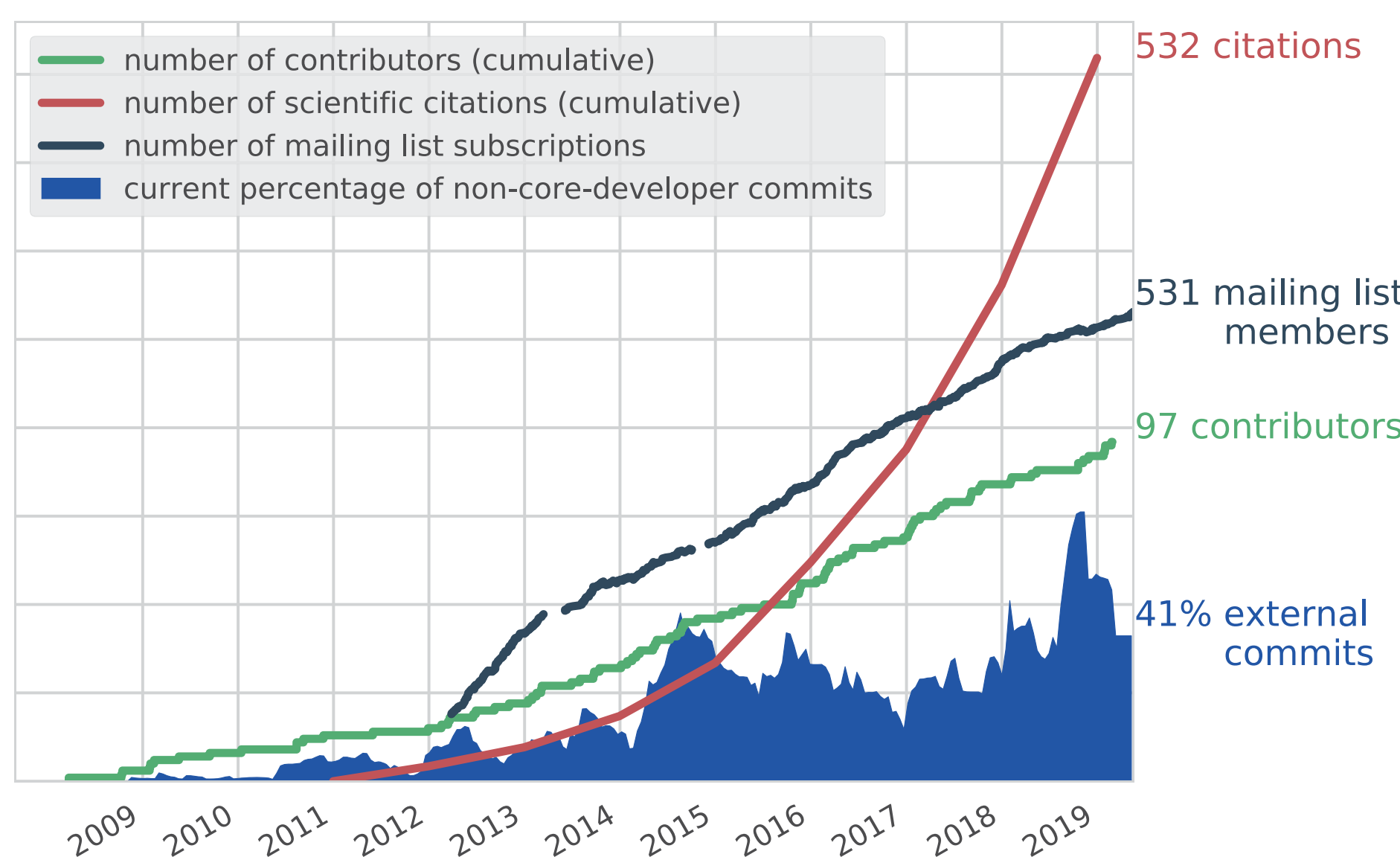
**EVENT METADATA** CMTSOLUTION, CNV, F-NETMT, GSE2, HypoDD PHA, IASPEI, JSON, KML, MCHEDR, NDK, NLLOC, Nordic, PDE, SC3ML, SCARDEC, Shapefile, QuakeML, Zmap



## Impact & Sustainability

Nine years after the beginnings of the project, ObsPy is used by seismologists all around the world. With tens of thousands conda downloads in the last year (only one of many ways to install ObsPy), we estimate -- including Mac, Windows and other Linux/Unix users -- an active user base of several thousand people.

Since ObsPy's start by a core developer team of 3-5 people at LMU Munich, ObsPy has evolved into a community effort with direct contributions to the code base by over 90 individuals. The busy user mailing list currently has more than 500 subscribers and serves as a place for discussions and asking for help from more experienced users. The impact of ObsPy and the appreciation within the seismological community finds expression in the increasing number of scientific citations, which stands at about 512 as of Dezember 2018.



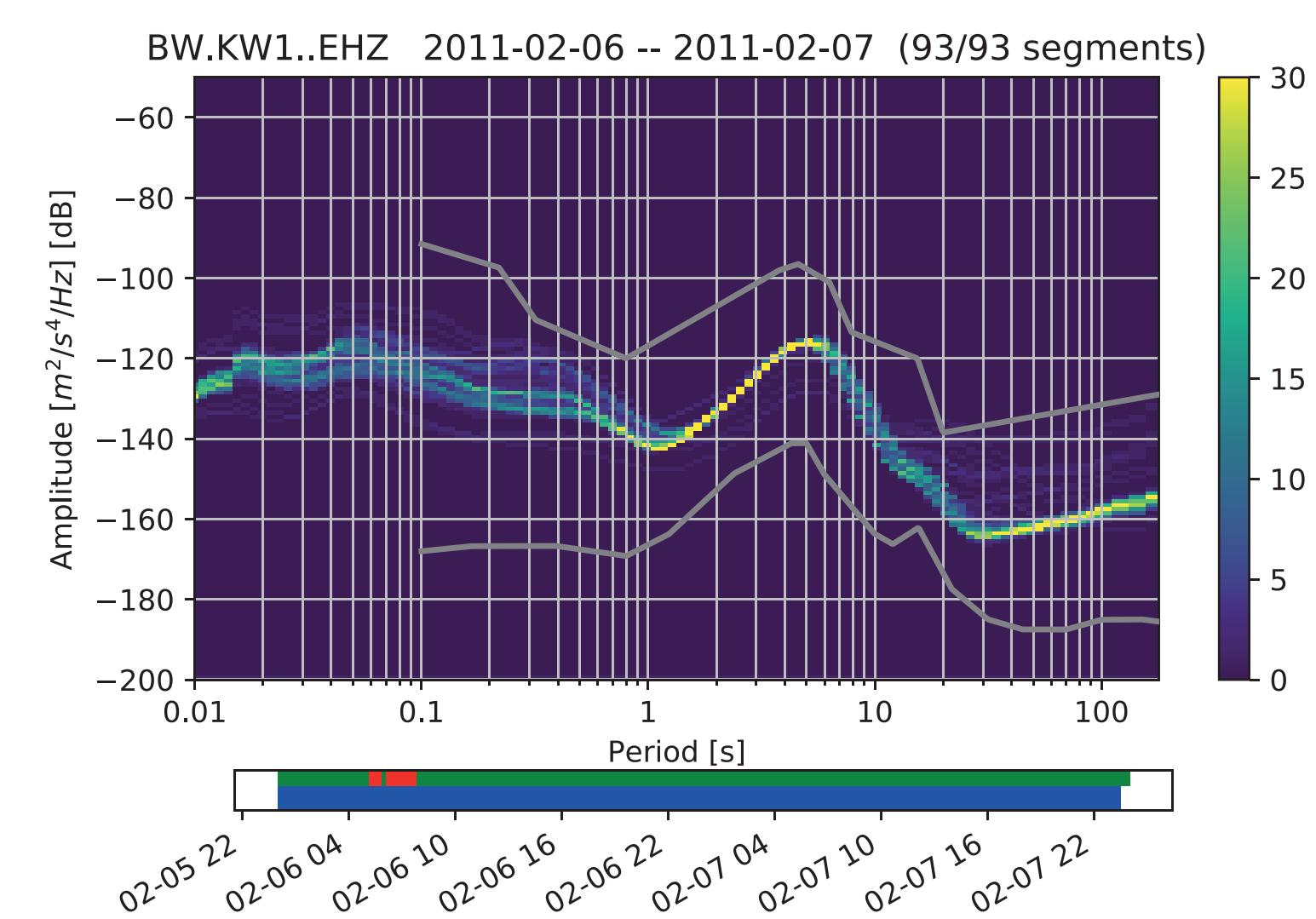
## What's New?

- \* New File Formats Support: Guralp GCF, Reftek 130, AH (write support added), NNSK KB Core, Nordic s-file, SCARDEC catalog files, GSE2.0 bulletin, SC3ML, IASPEI ISF ISM 1.0 Bulletin, SEED/RESP (now with inventory integration), ArcLink XML, Receiver Gather 1.6, Seismic Handler EVT files
- \* Full normalization in correlations for template matching
- \* Client for the Nominal Response Library (NRL)
- \* Support for two FDSNWS routing services (IRIS federator + EIDAWS routing)
- \* New QC module, ... and much more!

## Example

```
from obspy import read, read_inventory
from obspy.signal import PPSPD

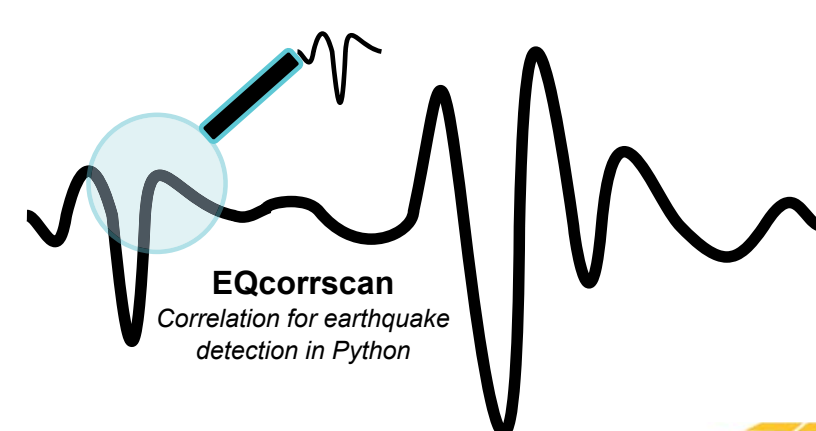
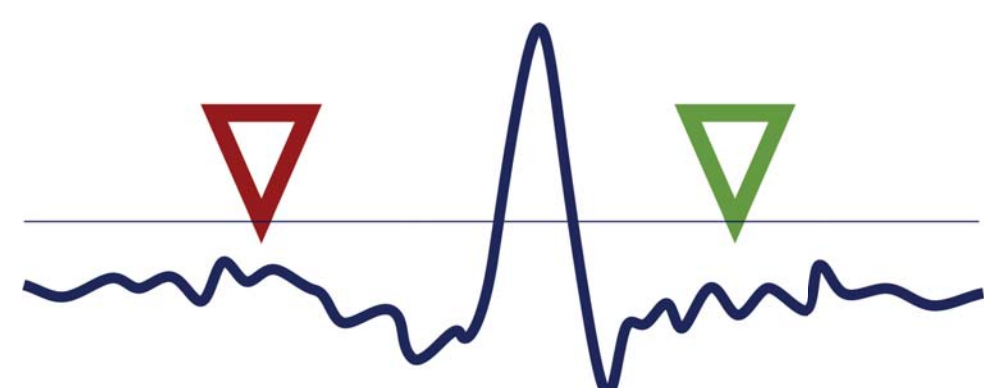
# Read data directly from a web address
st = read(
    "https://examples.obspy.org/BW.KW1..EHZ.D.2011.037")
inv = read_inventory(
    "https://examples.obspy.org/dataless.seed.BW_KW1")
# Init PPSPD object and add meta information
ppsd = PPSPD(st[0].stats, metadata=inv)
# Add data
ppsd.add(st)
st = read(
    "https://examples.obspy.org/BW.KW1..EHZ.D.2011.038")
ppsd.add(st)
# Plot it
ppsd.plot()
```



## Used by...

Applications are numerous and include event (re)locations, ambient seismic noise analysis, seismic tomography, rotational seismology studies, timedependent seismology, and more. Additionally, a large variety of programs built on top of ObsPy are appearing (full list including URLs at <http://obspy.org>)

Right now, ObsPy is used by 180 open-source projects on GitHub: <https://github.com/obspy/obspy/network/dependents>



## Try it Online!

<http://seismo-live.org>

Try ObsPy without installation in your web browser. Seismo-live is hosting a wide variety of instructional resources on all topics of seismology.

## One-Line Installation on All Platforms

```
$ conda install -c conda-forge obspy
```