The Journal of Open Source Software

SWEpy: A Python Library for Quick and Easy Access and Manipulation of MEaSUREs CETB Files for SWE Analysis

William M. Norris[1] **and Carson J.Q. Farmer**[1]

1 Earth Lab, University of Colorado Boulder

DOI: 10.21105/joss.01018

Software

- Review
- Repository
- Archive

Submitted: 01 October 2018
Published: 06 December 2018

Summary

Modern analysis of earth's systems requires significant volumes of increasingly fine resolution spatial and temporal information. For example, the *MEaSUREs Calibrated Enhanced Passive Microwave Daily-Grid 2.0 Brightness Temperature* data set, released by the National Snow and Ice Data Center (NSIDC), is no exception (Brodzik & Armstrong, 2018). The MEaSUREs dataset contains twice daily passive microwave imagery from 1978 through 2017, which can be particularly useful for spatial-temporal analysis of snow water equivalent (SWE). In addition to high temporal resolution, the MEaSUREs CETB dataset contains imagery that has been resampled from 25km down to 6.25km wide pixels, allowing for analysis to be done at a finer spatial scale than previously available versions of the data source. While this data is exceedingly useful, it can take considerable time to set up scripts for web access, geographic subsetting, and merging sequential images into useful time cubes. The MEaSUREs data set is also large: accessing a full time series of Northern Hemisphere imagery would require easily two terabytes of disc space, which most researchers do not have available on their local computer. SWEpy was developed in part to deal with these issues, while also providing basic data manipulation tools to facilitate SWE-based research. SWEpy is a Python library that allows for quick and easy web-access, geographic subsetting, and concatenation of MEaSUREs CETB files. The primary goal of SWEpy is to reduce the time needed for researchers to access and begin making use of MEaSUREs CETB data. A user with very little knowledge of Python can use SWEpy to access this data within minutes, and those more familiar with the scientific Python ecosystem can benefit from the wealth of tools available for subsequent analysis of CETB data.

Ease of use is a primary goal for SWEpy. Users can enter as little information as a date range, bounding coordinates, and their Earthdata login credentials, and SWEpy will return two single time cube files with only the data for the user's study area included. SWEpy has intelligent defaults pre-defined for various CETB files used in SWE analysis, such as which sensor is the most stable for a given year, as well as more specific web-access functionality. When using the default settings, in order to avoid running out of disk space, SWEpy will optimize long running data access sessions by automating the whole process, stopping to subset and concatenate files incrementally when the entire dataset will be too large to store on disk. Additionally, if a user has more specific file needs, they can work directly with SWEpy's flexible web access tool, and still make use of the built in geographic subsetting and merging features if desired.

Example Use of Auto Workflow

```
from swepy.swepy import swepy
import datetime
import os

# enter Earthdata login credentials
username = "username"
password = "password"

# enter bounding coordinates
lon_lat_ul = [66,-145]
lon_lat_lr = [73,-166]

# set date range (Y,M,D)
start = datetime.date(2010,1,1)
end = datetime.date(2010,2,1)

# instantiate the class
swe = swepy(os.getcwd(), start=start, end=end,
            ul=lon_lat_ul, lr=lon_lat_lr, username=username,
            password=password)

# Get a time cube for your study area
swe.scrape_all()
```

Example Use of Web Scraper Alone

```
from swepy.nsidcDownloader import nsidcDownloader
nD = nsidcDownloader.nsidcDownloader(username=username,password=password,folder=os
file = {
    "resolution": "3.125km",
    "platform": "F17",
    "sensor": "SSMIS",
    "grid": "N",
    "pass": "M",
    "date1": datetime.date(2015,10,10),
    "date2": datetime.date(2015,10,10),
    "channel": "37H",
    "algorithm": "SIR",
    "input": "CSU"
}

nD.download_file(**file)
```

Accessing SWEpy

SWEpy is available from GitHub (https://github.com/wino6687/SWEpy) and anaconda (https://anaconda.org/wino6687/swepy). Full documentation can be found in the GitHub repository wiki (https://github.com/wino6687/SWEpy/wiki). Any issues or bugs can be reported to the issues page on the GitHub repository: https://github.com/wino6687/SWEpy/issues.

Acknowledgements:

SWEpy was developed with funding from Earth Lab at the University of Colorado Boulder under the supervision of Carson Farmer.

References

Brodzik, D. G. L., M. J., & Armstrong, R. (2018). MEaSUREs calibrated enhanced-resolution passive microwave daily ease-grid 2.0 brightness temperature esdr, version 1.3. https://doi.org/10.5067/MEASURES/CRYOSPHERE/NSIDC-0630.001.