Introducing icepyx, an open source Python library for obtaining and working with ICESat-2 data

Jessica Scheick[1], Anthony Arendt[2], Lindsey Heagy[3], and Fernando Perez[3]

[1]University of Maine
[2]University of Washington
[3]UC - Berkeley

November 22, 2022

Abstract

Within a year of its launch date, the ATLAS altimeter on board ICESat-2 is already providing a wealth of critical data of interest across and beyond the cryospheric sciences. With the satellite returning nearly 1 TB of raw data per day, traditional practices of individual research groups downloading large granules of data and then subsetting, processing, and storing them locally are ultimately impractical. We are leading the development of icepyx (formerly icesat2py), an open source Python library designed to easily query, filter, download, and pre-process ICESat-2 datasets. The project's documentation will include interactive Jupyter Notebook examples, providing a starting point for researchers to create and customize workflows to address their research questions. We actively invite contributions from the community to ensure the project develops in a way that meets a wide range of research needs. The project aims to leverage existing libraries that enable easy parallelization and can be run locally or on cloud-based platforms. As a result, researchers will ultimately download and store minimally-sized subsets of ICESat-2 data and have the opportunity to contribute their code to an established open science project. This presentation will serve to introduce icepyx to the cryosphere community and encourage early adoption of the library by researchers with all levels of coding experience.

C31C-1526

Introducing `icepyx`: an open-source Python library for obtaining and working with ICESat-2 data

Jessica Scheick[1], Anthony Arendt[2], Lindsey Heagy[3], Fernando Pérez[3]

1. University of Maine, Climate Change Institute, 2. University of Washington, Applied Physics Lab, 3. University of California - Berkeley, Department of Statistics

Motivation and Objectives

- The ATLAS altimeter on board ICESat-2 returns ~1 TB of raw data per day
- Traditional practices of downloading large granules of data for local subsetting, processing, and storage are impractical

We are developing `icepyx`, an open-source Python library for cryospheric scientists to query, filter, download, and pre-process ICESat-2 datasets. `icepyx` will:

- Empower the the ICESat-2 user community to utilize advanced computing to answer their research questions without needing to become software developers
- Build a community of cryospheric scientists practicing open science, including contributing to open-source software, regardless of career stage or coding acumen

Software Development Philosophy

- Follow established best practices for software and community development
- Integrate our development with existing open-source services
- Contribute to established communities and their active commitment to open, reproducible science

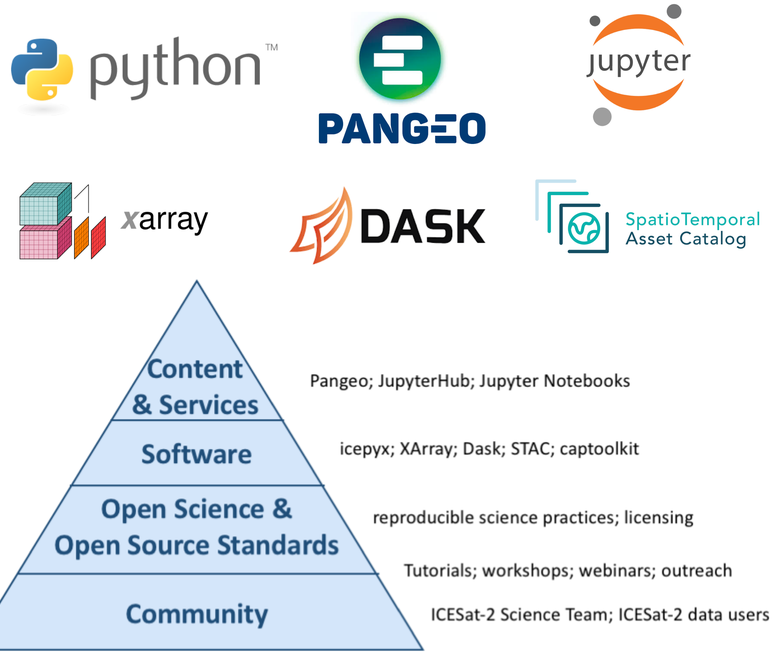

Fig. 1: Hierarchy for successful open-source software development

Community Development - Hackweeks

- In June 2019, the University of Washington hosted the first Cryospheric Sciences ICESat-2 Hackweek
- Attendees noted a need for a unifying framework to collate and document code for working with ICESat-2 data
- `icepyx` began as a Hackweek project, initially called `icesat2py`

Fig. 2: Hackweeks are intensive, interactive workshops centered around three components: interactive lectures on current methods, peer-learning, and collaborative, on-site project work (including software development).

Interested in participating in a Hackweek?

- The next Cryospheric Sciences ICESat-2 Hackweek will be held at the University of Washington 15-19 June 2020
- Learn about technologies to access and process ICESat-2 data, with a focus on cryospheric applications
- Become a tutorial lead and share your experience
- Visit icesat-2hackweek.github.io to apply!

`icepyx`: Current Framework

Create ICESat-2 data object
- temporal bounds
- dataset (e.g. ATL06)
- region of interest
- optional: start/end time; dataset version

Find granules
- submit search to National Snow and Ice Data Center (NSIDC)

Log in to Earthdata and order data

Download data
- minimally sized dataset
- subset to region and variables of interest (in progress)

```
1  from icepyx import is2class as ipd
2
3  #define search parameters
4  short_name = 'ATL06'
5  spatial_extent = [-64, 66, -55, 72]
6  date_range = ['2019-02-22','2019-02-28']
7
8  #create a data object
9  region_a = ipd.Icesat2Data(short_name, spatial_extent,
   date_range)
10
11 #search for available granules
12 region_a.avail_granules()
13
14 #start an Earthdata session
15 earthdata_uid = 'jane.smith'
16 email = 'somebody@somewhere.edu'
17 session=region_a.earthdata_login(earthdata_uid, email)
18 #the user is prompted for their Earthdata password
19
20 #order the granules, subsetting them (optional,default)
   to your area of interest
21 region_a.order_granules(session)
22
23 #download the data
24 path = './downloads'
25 region_a.download_granules(session, path)
```

Fig. 3: Example searching for and downloading data using `icepyx`. The same tasks without `icepyx` require a minimum of 50 lines of code, in comparison with the 12 shown above.

Use Cases

- Documentation: code structure and interactive Jupyter Notebook examples showcasing common glaciological research tasks
- Currently planned example use cases include:
  - snow height in non-glaciated regions
  - evaluating impacts of blowing snow and clouds
  - sea ice model parameter assimilation
  - glaciological workflow elements: defining a region of interest, downloading and subsetting data, filtering and corrections, trend detection, feature detection, advanced statistical and machine learning methods, validation and integration with other products, visualization
- Please tell us about your ICESat-2 use case!

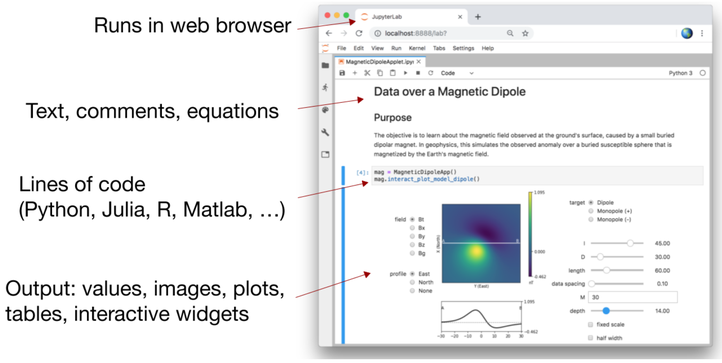

Fig. 4: Jupyter Notebook displayed in the JupyterHub computing environment

How to Find `icepyx` and Get Involved

- Find `icepyx` on GitHub: http://github.com/icesat2py/icepyx
- Installation: fork our repo (pypi and conda are coming soon)
- Contribute to `icepyx`: submit a pull request on GitHub (resources to assist new contributors with this process are coming soon)
- Apply for the next ICESat-2 themed Hackweek (see left)
- Join the conversation: https://discourse.pangeo.io
- What resources would be helpful?
- What examples would you like to see?

Funding: NASA Awards (80NSSC18M0157 and 80NSSC19K0495) to AArendt

In cooperation with NSIDC, Pangeo, and many others