Collaborative Computational Resource Development around ICESat-2 Data: the icepyx Community and Library

Jessica Scheick[1], Kelsey Bisson[2], Tian Li[3], Wei Ji Leong[4], and Anthony Arendt[5]

[1]University of New Hampshire
[2]Oregon State University
[3]University of Bristol
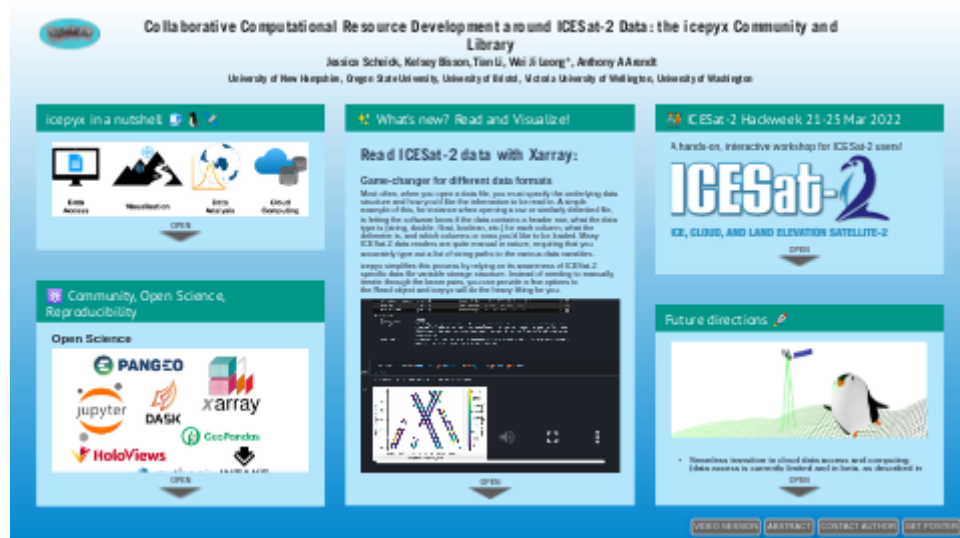[4]Victoria University of Wellington
[5]University of Washington

November 22, 2022

Abstract

Cryospheric data is increasing in size, demanding highly computational analyses. Open science principles, including collaboration, enable efficient, tested, reproducible, and diverse computational resource development. The ICESat-2 science community continues to coalesce around these ideals through contributions to icepyx, a community and open-source Python library for obtaining and working with large (~500 GB/day) data products from the ICESat-2 satellite/ATLAS laser altimeter. Our presentation will focus on the history, motivation, and process of creating this community, developing shared computational tools, and collating a set of example workflows within Jupyter Notebooks focused on ICESat-2 data. We will present new and in-the-works examples and features of the library, including enhanced pre-data-download visualizations, collaborative developments for multi-mission and -sensor data access, and data read-in/merging functionality. We will also highlight the community building events (including hackweeks) that drive this group and showcase some of the research supported and enabled by this software library.

1

Collaborative Computational Resource Development around ICESat-2 Data: the icepyx Community and Library

Jessica Scheick, Kelsey Bisson, Tian Li, Wei Ji Leong*, Anthony A Arendt

University of New Hampshire, Oregon State University, University of Bristol, Victoria University of Wellington, University of Washington
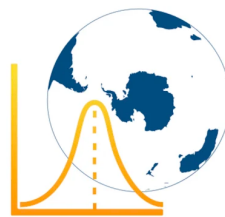
ICEPYX IN A NUTSHELL 📦🐧🛰️

Data Access   **Visualisation**   **Data Analysis**   **Cloud Computing**

What is icepyx?

Icepyx is both a software library and a community composed of ICESat-2 data users, developers, and the scientific community. We are working together to develop a shared library of resources - including existing resources, new code, tutorials, and use-cases/examples - that simplify the process of querying, obtaining, analyzing, and manipulating ICESat-2 datasets to enable scientific discovery. Icepyx also hopes to foster collaboration, open-science, and reproducible workflows by integrating and sharing resources.

Development history

Many of the underlying tools from which icepyx was developed began as Jupyter Notebooks developed for and during the cryosphere-themed ICESat-2 Hackweek at the University of Washington in June 2019 or as scripts written and used by the ICESat-2 Science Team members. This project combines and generalizes these scripts into a unified framework, adding examples, documentation, and testing where necessary and making them accessible for everyone. It also improves interoperability for ICESat-2 datasets with other open-source tools.

Our goal

- Open-source Python code to save people from reinventing the wheel

- Build a community for sharing and cooperating on ICESat-2 related science

- Making science reproducible and accessible for everyone!
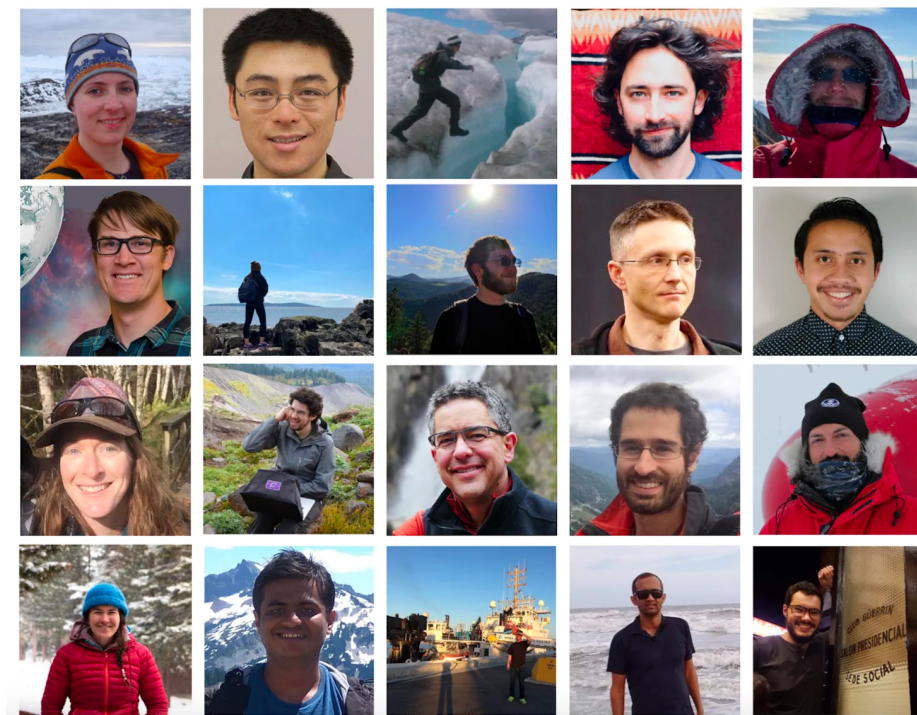
Find out more at https://icepyx.readthedocs.io

⚛ COMMUNITY, OPEN SCIENCE, REPRODUCIBILITY

Open Science

Community

Some (but not all!) of our amazing contributors and community members are pictured below. We invite you to join the team (no coding experience necessary)!

- **GitHub (https://github.com/icesat2py/icepyx)**
- **Pangeo forum (https://discourse.pangeo.io/c/science/icesat-2/16)**
- **ICESat-2 Hackweek Tutorials (https://github.com/ICESAT-2HackWeek /ICESat2_hackweek_tutorials)**
- **ICESat-2 online viewer by** OpenAltimetry (https://openaltimetry.org/data/icesat2/)
- Jupyter meets the Earth (https://workflow.jupytearth.org/icesat2-crevasses/getdata.html) **by Whyjay Zheng**

Sponsors:

NASA ICESat-2 Mission

University of Washington eScience Institute

✨ WHAT'S NEW? READ AND VISUALIZE!

Read ICESat-2 data with Xarray:

Game-changer for different data formats

Most often, when you open a data file, you must specify the underlying data structure and how you'd like the information to be read in. A simple example of this, for instance when opening a csv or similarly delimited file, is letting the software know if the data contains a header row, what the data type is (string, double, float, boolean, etc.) for each column, what the delimeter is, and which columns or rows you'd like to be loaded. Many ICESat-2 data readers are quite manual in nature, requiring that you accurately type out a list of string paths to the various data variables.

icepyx simplifies this process by relying on its awareness of ICESat-2 specific data file variable storage structure. Instead of needing to manually iterate through the beam pairs, you can provide a few options to the Read object and icepyx will do the heavy lifting for you.

[VIDEO] https://res.cloudinary.com/amuze-interactive/video/upload/vc_auto/v1638452824/agu-fm2021/87-3A-85-E7-6F-78-E6-C8-F6-05-A6-7A-77-54-F0-94/Video/Xarray_lq4cmo.mp4

Notebook link:

https://github.com/icesat2py/icepyx/blob/development/examples/ICESat-2_Data_Read-in_Example.ipynb (https://github.com/icesat2py/icepyx/blob/development/examples/ICESat-2_Data_Read-in_Example.ipynb)

Visualization with OpenAltimetry:

Interactive visualizations and data download all in one package

Create your icepyx Query object with your desired search parameters. Then, visualize the data right within your workflow (behind the scenes, icepyx is using OpenAltimetry). Continue working with the heights from OpenAltimetry, or order more variables and download the data files.

[VIDEO] https://res.cloudinary.com/amuze-interactive/video/upload/vc_auto/v1638177592/agu-fm2021/87-3A-85-E7-6F-78-E6-C8-F6-05-A6-7A-77-54-F0-94/Video/icepyx_visualization_module_skk32i.mp4

Notebook link:

https://github.com/icesat2py/icepyx/blob/main/examples/ICESat-2_Data_Visualization_Example.ipynb (https://github.com/icesat2py/icepyx/blob/main/examples/ICESat-2_Data_Visualization_Example.ipynb)

Customized Data Subsetting:

Smaller, faster downloads and storage

Anyone who's worked with geospatial data has probably encountered subsetting. Typically, we search for data wherever it is stored and download the chunks (aka granules, scenes, passes, swaths, etc.) that contain something we are interested in. Then, we have to extract from each chunk the pieces we actually want to analyze. Those pieces might be geospatial (i.e. an area of interest), temporal (i.e. certain months of a time series), and/or certain variables. This process of extracting the data we are going to use is called subsetting.

In the case of ICESat-2 data coming from the NSIDC DAAC, we can do this subsetting step on the data prior to download, reducing our number of data processing steps and resulting in smaller, faster downloads and storage.

[VIDEO] https://res.cloudinary.com/amuze-interactive/video/upload/vc_auto/v1638455194/agu-fm2021/87-3A-85-E7-6F-78-E6-C8-F6-05-A6-7A-77-54-F0-94/Video/subset_wkncbv.mp4

Notebook link:

https://github.com/icesat2py/icepyx/blob/development/examples/ICESat-2_DAAC_DataAccess2_Subsetting.ipynb (https://github.com/icesat2py/icepyx/blob/development/examples/ICESat-2_DAAC_DataAccess2_Subsetting.ipynb)

Stay tuned for more exciting updates!
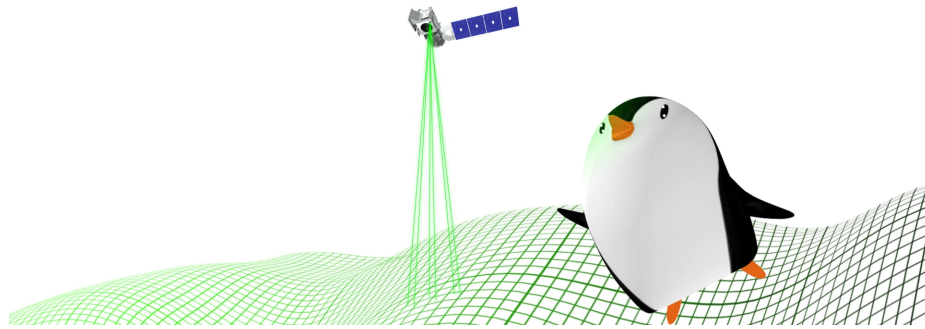
👫 ICESAT-2 HACKWEEK 21-25 MAR 2022

A hands-on, interactive workshop for ICESat-2 users!

- **Virtual** on a variety of interactive platforms
- **In person** @ University of Washington, Seattle.

More info and applications at https://icesat-2hackweek.github.io (https://icesat-2hackweek.github.io). Apply by 10 Jan 2022!

FUTURE DIRECTIONS 🚀

- Seamless transition to cloud data access and computing (data access is currently limited and in beta, as described in this notebook (https://github.com/icesat2py/icepyx /blob/development/examples/ICESat-2_cloud_data_access_example.ipynb))

- Template for integration of new datasets
  - Uses a generic class structure to recycle spatial and temporal data
  - Provides templates for adding new datasets and integrating them with ICESat-2 data
  - 2020 Hackweek Phytoplankton team initiated and continues this work

- Check out our GitHub issues (https://github.com/icesat2py/icepyx/issues) for some other features in the works, and send us your ideas/requests!

ABSTRACT

Cryospheric data is increasing in size, demanding highly computational analyses. Open science principles, including collaboration, enable efficient, tested, reproducible, and diverse computational resource development. The ICESat-2 science community continues to coalesce around these ideals through contributions to icepyx, a community and open-source Python library for obtaining and working with large (~500 GB/day) data products from the ICESat-2 satellite/ATLAS laser altimeter. Our presentation will focus on the history, motivation, and process of creating this community, developing shared computational tools, and collating a set of example workflows within Jupyter Notebooks focused on ICESat-2 data. We will present new and in-the-works examples and features of the library, including enhanced pre-data-download visualizations, collaborative developments for multi-mission and -sensor data access, and data read-in/merging functionality. We will also highlight the community building events (including hackweeks) that drive this group and showcase some of the research supported and enabled by this software library.