



## NEON CODE RESOURCES OVERVIEW

The National Ecological Observatory Network (NEON) provides over 180 different data products that cover a wide range of variables of interest to researchers across the earth and life sciences. The diversity of data and data types made available by NEON create significant challenges to data users, and the NEON program is exploring:

- 1) Providing open source code to perform common tasks required for NEON data
- 2) Supporting code-sharing by NEON data users

An engaged community of scientific programmers is an invaluable asset to any open data provider, and facilitating such a community is a priority for the NEON project.



Background on the NEON project:

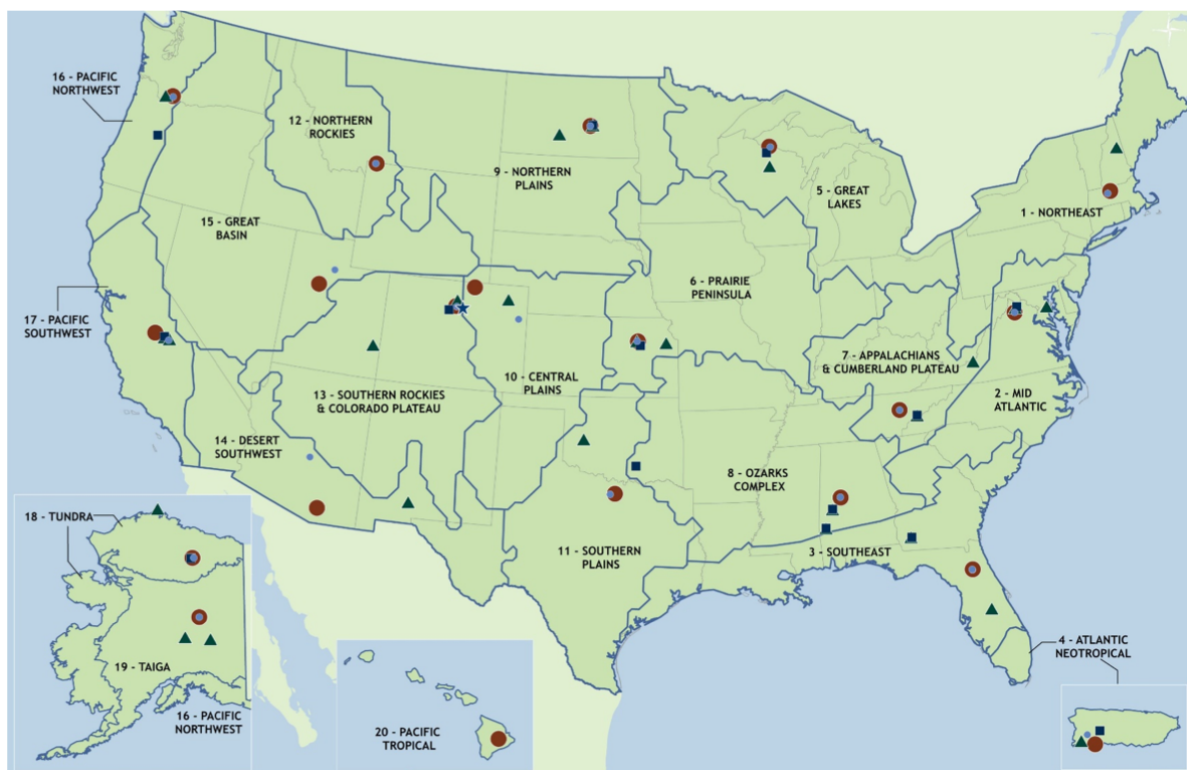
NEON is a continental-scale observation facility, funded by the National Science Foundation and operated by Battelle, and designed to collect long-term open access ecological data to better understand how U.S. ecosystems are changing. The comprehensive data, spatial extent and remote sensing technology provided by NEON will enable a large and diverse user community to tackle new questions at scales not accessible to previous generations of ecologists.

NEON collects environmental data and archival samples that characterize plant, animals, soil, nutrients, freshwater and atmosphere from 81 field sites strategically located in terrestrial and freshwater ecosystems across the U.S.

Collection methods are standardized across field sites to provide high quality datasets from in situ automated instrument measurements, observational sampling and airborne remote sensing surveys.

- Over 180 open access data products are available on the NEON data portal

- NEON also provides a variety of open access data tutorials, code packages and other resources to enable use of NEON data.
- NEON also archives over 100,000 biological, genomic and geological samples each year which are available upon request from the NEON Biorepository.



## NEONUTILITIES: NEON DATA ACCESS AND WRANGLING

There is a small set of functionality that is essential for downloading and handling nearly all NEON data products. Rather than force each user to reinvent the wheel, NEON provides the `neonUtilities` R package to handle this basic functionality. `neonUtilities` contains functions for:

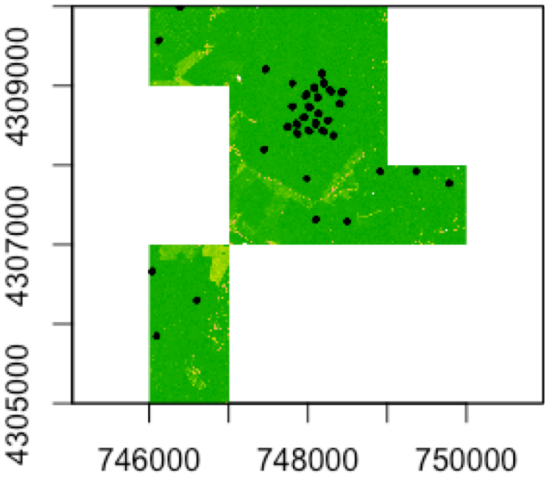
1. Downloading data directly from the NEON API
2. Merging data files downloaded from the NEON Data Portal or API
3. Extracting tabular data from NEON flux data HDF5 files
4. Downloading data based on specific criteria (e.g., downloading remote sensing data based on ground sampling locations)

The `neonUtilities` package is under continual development and improvement. The latest release is available at the Comprehensive R Archive Network (CRAN) (<https://CRAN.R-project.org/package=neonUtilities>), and the development version and full code history are available on GitHub (<https://github.com/NEONScience/NEON-utilities/tree/master/neonUtilities>).

Merge data files:



Download remote sensing data based on locations of ground sampling:



## SUPPORTING A COMMUNITY OF CODE

### Got NEON-related code? We can help you share it!

Beyond the basics provided by neonUtilities, there are a broad range of common transformations and calculations that many users will want to perform, and that would benefit from a shared code base. Some of that code base can be developed by NEON staff, but there are also extensive resources currently being developed by the user community.

To facilitate both sharing and discovery of code, NEON provides an index of code resources, including both in-house and externally developed code. Submissions are open! [neonscience.org/resources/code-resources](https://neonscience.org/resources/code-resources)  
(<https://neonscience.org/resources/code-resources>)

To help guide both developers and users of code, NEON has developed a set of standards for different types of shared code, and the level of review each type will be subject to before it is promoted on the NEON website. This figure gives a general outline, more detailed information can be found at [neonscience.org/code-resource-guidelines](https://neonscience.org/code-resource-guidelines) (<https://neonscience.org/code-resource-guidelines>)

NEON Code Guidelines:

	<b>Tier 1: Contributed Code</b>	<b>Tier 2: NEON Certified Code</b>	<b>Tier 3: NEON Production Code</b>
<b>Purpose of code</b>	Work with published NEON data	Work with published NEON data	Generate NEON data products
<b>Standards for review</b>	Relevant to NEON mission  Available under a public license	Packagized  Passes standard automated checks (e.g. R CMD)	Requires extensive collaboration with NEON scientists
<b>Examples</b>	Code associated with NEON-related publications	neonUtilities geoNEON neonNTrans, reaRate, etc	eddy4R  stageQCurve

NEON Code Resources Index:




## Code Resources

The NEON-related code resources listed below are designed to make working with all NEON data easier, to perform common algorithms on select data products, and to share the code used to generate select data products. Most NEON-curated code resources can be found in the [NEONScience GitHub organization](#). The code is free and open access to download and utilize. The code found in the NEONScience GitHub organization is published and maintained by NEON project scientists. Other code resources listed below are created by data users interested in sharing their code. If you have requests for coding resources, challenges with NEON data or ideas for creating NEON data-related code, we encourage you to [contact us](#).



[Learn more](#) about how you can submit your own code resource and how we categorize NEON-related code resources

 Search by name or keyword

[RECENT](#)
[A -> Z](#)
[POPULAR](#)
[LANGUAGE](#)
[APPLY](#)

Title	Description	Tier	Language
geoNEON	Use R to handle NEON geolocation data including extracting spatial data from the API based on a named location, and calculating more precise locations for select observational data products. <a href="#">- See less</a> <b>Tier 2:</b> NEON certified code <b>Coding Language:</b> R language <b>Contributor name:</b> NEON <b>License:</b> GNU Affero General Public v3.0 <b>Related collection system:</b> <b>Data products:</b> <a href="#">MORE INFO</a>	Tier 2	R language
metScanR	Access meteorological data from a growing database that contains metadata for >100,000 stations from 219 countries or territories worldwide — including all NEON sites. <a href="#">+ See more</a>	Tier 2	R language

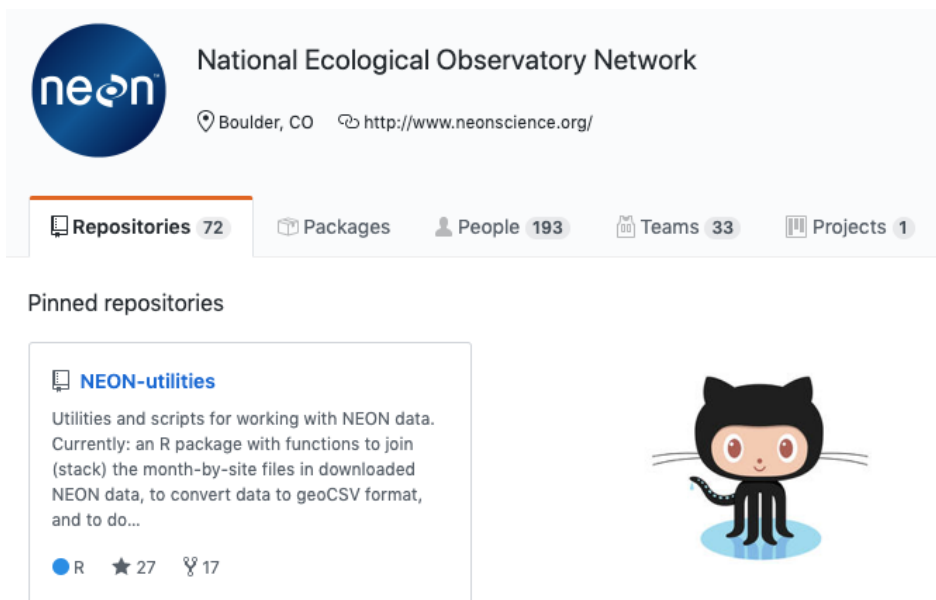
Submit your own code!

[neonscience.org/code-resources-submission](https://neonscience.org/code-resources-submission) (<https://neonscience.org/code-resources-submission>)

## OPEN CODE PLATFORMS

Providing open-source code to the NEON community requires open platforms, and transparency into the code-writing and prioritization process.

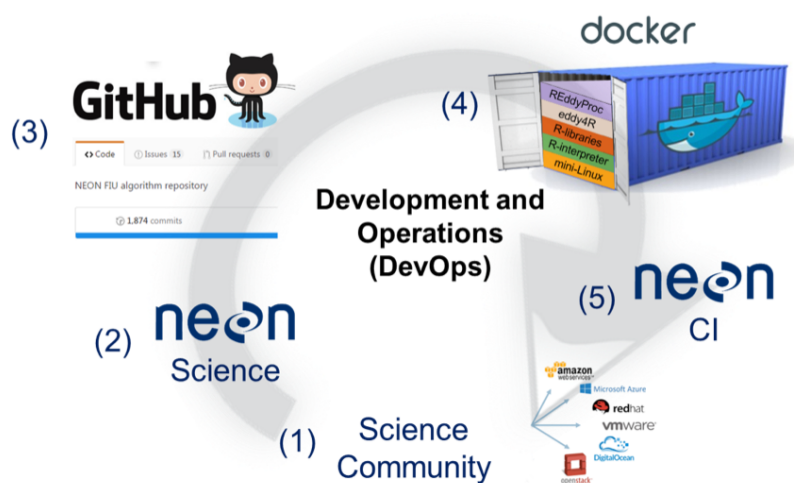
NEON open-source code resources are hosted on GitHub, at [GitHub.com/NEONScience](https://GitHub.com/NEONScience) (<https://GitHub.com/NEONScience>):



Using the GitHub platform enables openness in code development. Users can submit issues to report code bugs, request enhancements, or get help using code resources. They can fork NEON's repositories to adapt code to their own needs, and, for select code projects, can submit pull requests back to NEON for consideration.

The eddy4R package, which is used to generate NEON's surface-atmosphere exchange data products, has been developed via extensive collaboration with the community, using GitHub to manage contributions from different collaborators. To control software version and environment, the code is run in a Docker container in the NEON data pipeline. The package is now available to the public on GitHub.





## LESSONS AND FUTURE DIRECTIONS

One of the major challenges for NEON in an open code environment is the extraordinary diversity of data the observatory provides. For code developed within the observatory, providing resources to support the full range of data products is the highest priority.

Community members developing code to work with NEON data are more likely to focus on a particular challenge or a particular set of data products. As more users join the community and share code, they can develop a catalog of specialized resources, providing a strong complement to the more generalized code resources the observatory focuses on.

Both internal and external developers have expressed strong interest in developing Tier 3 code, that can be used to produce new NEON data products. The eddy4R and stageQCurve packages, and the data products they generate, are only the first examples of what can one day be a very diverse environment of data products derived from NEON's basic measurements.

## ABSTRACT

An engaged community of scientific programmers is an invaluable asset to any open data provider. The National Ecological Observatory Network (NEON) is a long-term observatory focused on collecting and providing open, continental-scale data that characterize and quantify complex and rapidly changing ecological patterns and processes. The observatory provides over 180 different data products that cover a wide range of variables of interest to researchers across the earth and life sciences. NEON creates and provides code and tools to enhance researchers' ability to work with these data. In addition, NEON provides several platforms to help connect researchers sharing open code related to NEON data products with those who are also interested in using them. Code and tools created by NEON scientists are distributed through the NEONScience GitHub organization (<https://github.com/NEONScience>). Current tools include the `neonUtilities` R package that provides basic tools for accessing and working with most NEON data products, as well as the `geoNEON` package that facilitates access to NEON spatial data. Other code packages contain the algorithms used to produce specific data products, including the `eddy4R` package, used to create the bundled eddy-covariance data product. Finally, some code packages are designed to build upon published NEON data to create value-added, derived products. Members of NEON's user community have contributed to some of the packages described above, and others are creating their own open code resources for using NEON data. Use of NEON code packages and development of open code are highly variable within the NEON user community, and NEON has explored several approaches to engage users in this aspect of the observatory, including online tutorials, webinars, workshops, and hackathons. Developing and expanding an engaged community of open code users around NEON data is a continuing and evolving effort for the NEON project.

