Development and uses of the Python-ARM Radar Toolkit (Py-ART)

Jonathan Helmus\(^1\), Scott Collis\(^1\), Karen Johnson\(^2\), Scott Giangrande\(^2\), Kirk North\(^3\) and Michael Jensen\(^2\)

1. Argonne National Laboratory, Argonne, IL, United States.
2. Brookhaven National Laboratory, Upton, NY, United States.
3. McGill University, Montreal QC, Canada.

Abstract

The Atmospheric Radiation Measurement (ARM) climate facility routinely deals with large amounts of complex remote sensing data. The measured parameters from this remote sensing data as well as Value Added Products (VAPs) are made available to the cloud and climate modeling communities. Preparing this data for dissemination requires extensive use of computational resources and algorithms which are not well addressed by current software packages. Here we will report on our progress in the development of the Python-ARM Radar Toolkit (Py-ART) to address these needs.

Py-ART offers a powerful interpreted environment for ingesting radar data from a number of formats, correcting for aliasing and attenuation, mapping data to Cartesian grids and performing a number of geophysical retrievals on the data. The package is also capable of writing data to Climate and Forecast (CF) standard NetCDF files as well as the emerging CF-Radial format for antenna coordinate data. Py-ART is written in the Python programming language, taking advantage of the powerful scientific libraries (NumPy, SciPy, matplotlib) available for the language as well as interfacing with legacy C and FORTRAN code. The package will be freely available under an open-source license. Here we will focus on our recent efforts to make the package more accessible to end users by simplifying the interface, increasing code readability, and developing high-quality documentation.

Additional information and Acknowledgements

Py-ART is an open source project which will be released under a BSD license, if you are interested in trying out a beta version please contact the lead developer, Jonathan Helmus (jjhelmus@anl.gov) for access to the GitHub repository and help with installing the package.

Documentation for Py-ART is available online at: http://jjhelmus.github.com/pyart/dev/index.html

Thanks go to the following for providing algorithms, code and support:
Scott Collins, Scott Giangrande, Kirk North, Alexander Ryzhkov, Matthias Steiner, Bart Kelley, Eric Bruning, Michael Jensen, Christian Zelt, Scott Johnson, and many others!

Py-ART would not be possible without the hard work of many other open source Scientific Python packages such as NumPy, SciPy, matplotlib, and python-netcdf4.

Ingest and writing

Py-ART has the ability to natively ingest (read) radar data from MDV and CF-Radial as well as other NetCDF based formats. If the NASA TRMM Radar Software Library is installed Py-ART can also ingest data from any memory as a formatted supported by the library (UF, Sigmet, Lassen, etc). Field data and instrument metadata are stored in based formats. If the NASA TRMM Radar Software Library is installed Py-ART can also ingest data from any memory as a formatted supported by the library (UF, Sigmet, Lassen, etc). Field data and instrument metadata are stored in (PPI) and range-height indicator (RHI) plots as well as plotting individual rays.

Py-ART can quickly create high quality plot of radar moments. Support for creating both plan position indicator (PPI) and range-height indicator (RHI) plots as well as plotting individual rays.

Mapping and Gridding

Py-ART contains a generalized ball-tree mapper which can be used to map data from one or multiple radars to a Cartesian grid.

Plotting

Py-ART can create plats of radar moments. Support for creating both plan position indicator (PPI) and range-height indicator (RHI) plots as well as plotting individual rays.

Coming Soon: https://github.com/ARM-DOE/pyart