Python ARM Radar Toolkit (Py-ART) Roadmap

Scott Collis, Cory Weber, Jonathan Helmus and Zachary Sherman

Introduction and Aims

Everyone one who uses weather radar to do science uses, in one form or another, radar software. Software is a science enabling piece of infrastructure and good software minimizes frustration and allows the domain expert to get to understanding the phenomena being studied without needing to be an expert on numerics, data informatics and software engineering.

There are several platforms for interacting with radar data, the open source variants are well documented in (Heistermann et al 2014). The Python ARM Radar Toolkit (Py-ART, Helmus and Collis, 2016) is one of these.

Py-ART grew out of a collection of radar algorithms generated in support of the new radar capability in the ARM program (Mather and Voyles, 2012). One of the first contributions was a Linear Programming (LP) technique for separating propagation polarimetric phase shift from other local impacts (Giangrande el at, 2013). As the collection of both algorithms and radars grew it became clear that the problem would become intractable unless a carefully designed architecture was designed that allowed application chains to be developer via a common data model approach.

Shortly after, in early 2012 development on Py-ART began in earnest with the support of the ARM program. In September of 2012 Py-ART was uploaded to the social coding platform GitHub at https://github.com/ARM-DOE/pyart. Py-ART was unofficially bumped to version 1.0.0-Dev in May of 2013 and publicly released. The first "stable" release was 1.2.0 in February of 2015 and the most recent release was 1.7.0 in September of 2016. Release notes can be found here: https://github.com/ARM-DOE/pyart/releases.

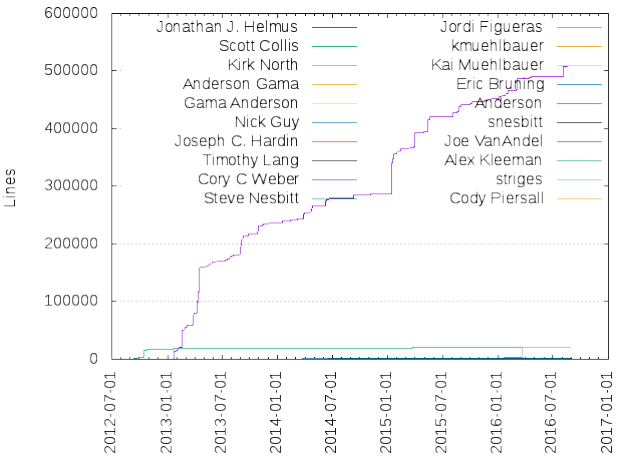

Figure 1: Lines of code by Contributor as a function of time. Note, some lines are automatically generated by Cython to C complilation.

Figure 1 shows the growth of the project as a function of time broken down by contributor. Note that some of the code is C which has been generated by Cython which inflates the number some what.

Py-ART has benefited from code from 15 individual contributors. This has been enabled by careful implementation of unit tests and continuous integration. Every time a pull request is submitted against the Py-ART codebase a set of tests run and a report is generated so the developers know if a contribution causes any unit tests to fail.

Py-ART receives vital support for accepting pull requests, bug fixing, documentation, outreach and education through the ARM program which is part of the Climate and Environmental Sciences Division of the Office of Science in the Department of Energy. Due to this, and to ensure the toolkit has maximal

impact a roadmap to chart development priorities for the next five years is needed and is the subject of this document. The roadmap document is broken down into:

1. This introduction
2. The results of the Py-ART roadmap survey
3. Proposed governance for accepting pull requests
4. Overarching goals for the next five years
5. Specific features that will be a priority for development

We also include a list of papers that have been accepted or are in process that have made use of Py-ART as a reference at the end of this document.

The Py-ART Roadmap Survey

Description and methods and lots of images

Proposed Governance Structure

Overarching Goals for Next Five Years

Freeform discussion of where we want to be

Priority Features

In priority order the features we want added either by ARM or features that if they are in a PR we will be very happy to help with this PR

(Heistermann et al, 2104) Heistermann, M., Collis, S., Dixon, M.J., Giangrande, S., Helmus, J.J., Kelley, B., Koistinen, J., Michelson, D.B., Peura, M., Pfaff, T., Wolff, D.B., 2014. The Emergence of Open Source Software for the Weather Radar Community. Bull. Amer. Meteor. Soc. doi:10.1175/BAMS-D-13-00240.1

(Helmus and Collis, 2016) Helmus, J.J. & Collis, S.M., (2016). The Python ARM Radar Toolkit (Py-ART), a Library for Working with Weather Radar Data in the Python Programming Language. Journal of Open Research Software. 4(1), p.e25. DOI: http://doi.org/10.5334/jors.119

(Mather and Voyles, 2012) Mather, J.H., Voyles, J.W., 2012. The Arm Climate Research Facility: A Review of Structure and Capabilities. Bull. Amer. Meteor. Soc. 94, 377–392. doi:10.1175/BAMS-D-11-00218.1

(Giangrande et al, 2013) Giangrande, S.E., McGraw, R., Lei, L., 2013. An Application of Linear Programming to Polarimetric Radar Differential Phase Processing. Journal of Atmospheric and Oceanic Technology 30, 1716–1729. doi:10.1175/JTECH-D-12-00147.1