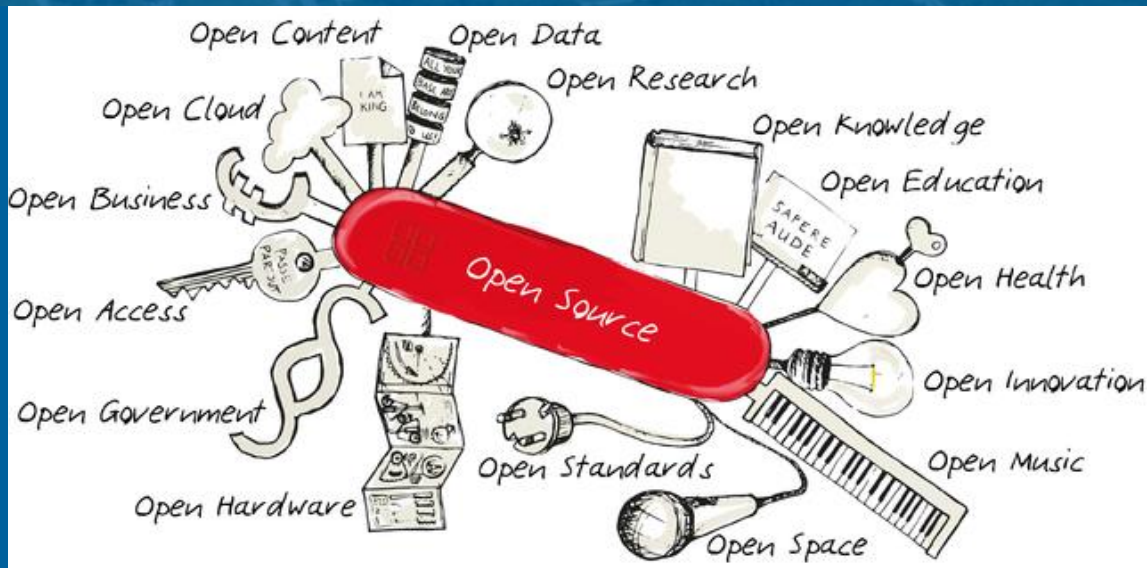
OPEN RADAR SCIENCE FOR FUN AND, YES, EVEN PROFIT.



SCOTT COLLIS
Atmospheric Scientist, Argonne National Laboratory.
Senior Institute Fellow, Northwestern University.

ROBERT JACKSON, ZACH SHERMAN AND MARK PICEL
Argonne National Laboratory

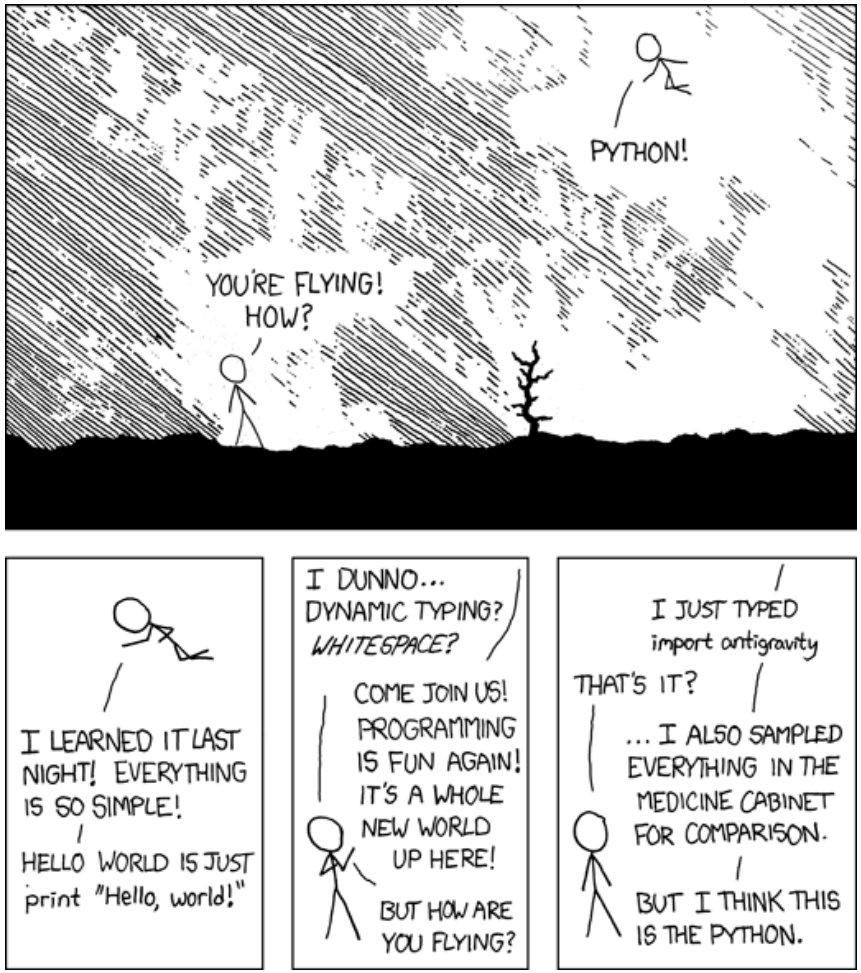
AND SO MANY MORE….

Argonne NATIONAL LABORATORY

WHY AM I GIVING YOU THIS TALK

What started as a way of building a radar retrievals/QC pipeline turned into a popular community radar toolkit

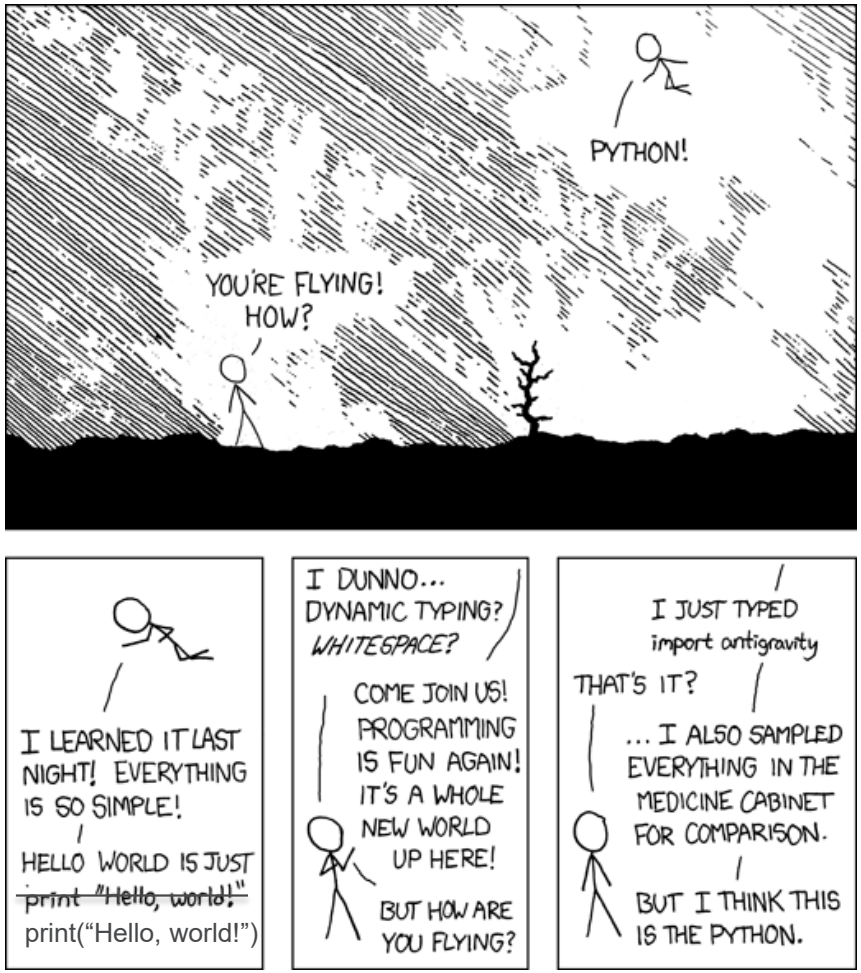
- I have personally benefited greatly from the Python and open source community.
- This quick talk will:
  - Give a bit of history as to how the Python ARM Toolkit came into being.
  - Introduce Py-ART for those that do not know what it is.
  - Show some example usage.
  - Sustainable growth: The Py-ART roadmap.
  - Introduce the Open Radar Partnership.



WHY AM I GIVING YOU THIS TALK

What started as a way of building a radar retrievals/QC pipeline turned into a popular community radar toolkit

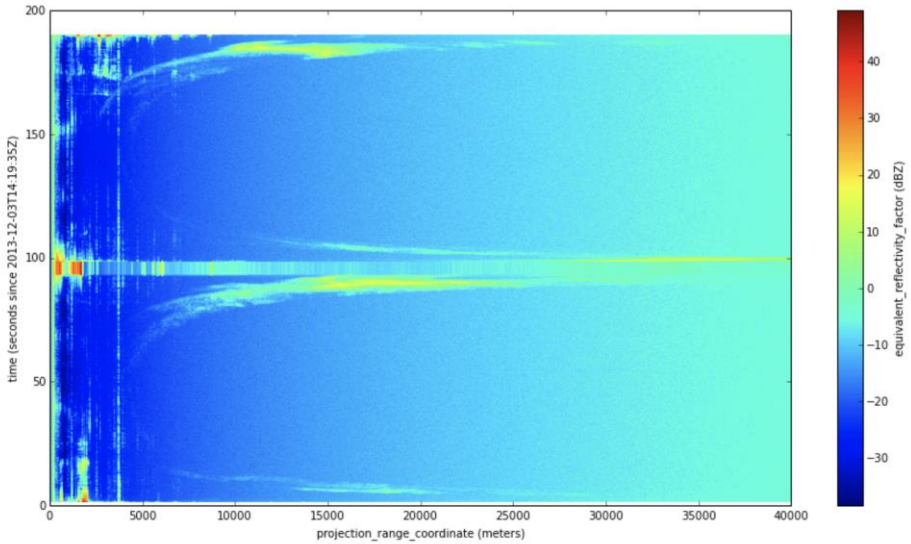
- I have personally benefited greatly from the Python and open source community.
- This quick talk will:
  - Give a bit of history as to how the Python ARM Toolkit came into being.
  - Introduce Py-ART for those that do not know what it is.
  - Show some example usage.
  - Sustainable growth: The Py-ART roadmap.
  - Introduce the Open Radar Partnership.



RADAR DATA

The shape of it….

- Weather radars transmit a pulse of energy (in GHz range, pw ~ 10's of meters) and gate the receiver to achieve ranging.
- The basic data form are rays. Collection of data collected at different (timed) range.
- For most radars the number of gates remain constant so data can be stored in a number of ND-Arrays of (time,range).
- In addition, for steerable (scanning) radars information about the pointing of the antenna must be stored.



A LITTLE HISTORY

Building a retrievals and quality control framework across a diverse network of radars

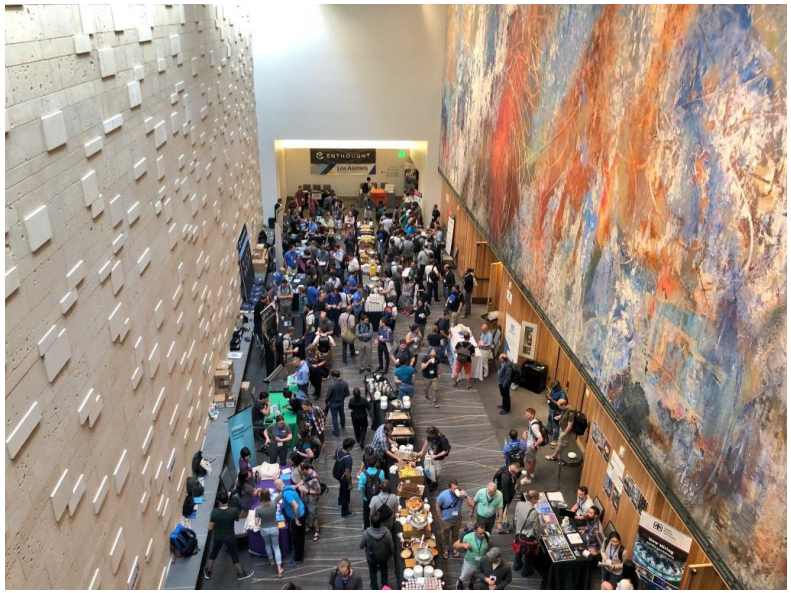
- Our journey into open source software did not start ”properly”. It was not scoped, designed, frame-worked, engineered and tested initially.
- ARM has a network of 20+ radars. Made by four vendors. Each vendor has their own format.
- **Py-ART started as a desire to consistently represent radar data in Python.**

SCIPY WAS THE CATALYST

Scipy showed me what is possible… I had no idea before.

- Scientific Computing with Python in 2012 is where I learned how to “Open Source” properly.
- My first ever distributed version control experience (Git/GitHub) was during the sprints.
- It gave a formula for building a code base and then user base we were able to take to DoE and request support for engineering Py-ART for community use.
- Inadvertently it led our first lead developer, Jonathan Helmus, joining us.

SciPy2012



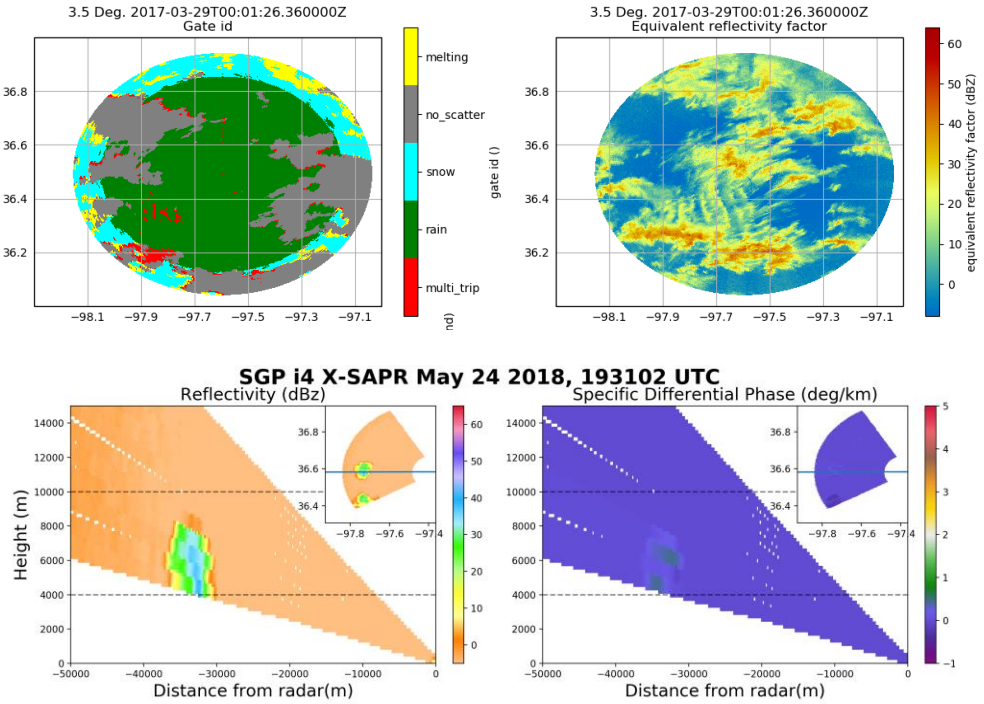
THE PYTHON ARM RADAR TOOLKIT

Philosophy: It's all about the data model.

- Py-ART's central core is a data model for gated data with pointing information.
- Py-ART created a way of representing radar data in the Python programming language that mirrors the CF-Radial standard.
- Py-ART has a cloud functions to correct, retrieve and grid radar data.
- By keeping a limited scope Py-ART aims to "do less better".
- There is now a **rich ecosystem** of packages that interact: ART-View, CSU tools, PyTDA, Multidop to name a few. Oh and we are working on TINT!

```
In [2]: import pyart
        radar = pyart.io.read('/data/cmac/raw/XSW170519002005.RAW54Y0')
        print(radar.ngates, radar.nrays, radar.nsweeps)

501 9200 23
```



Animation courtesy of users Marcus van Lier-Walqui and Sara E. Lytle
Data: Andrei Lindenmaeir – ARM Mentor

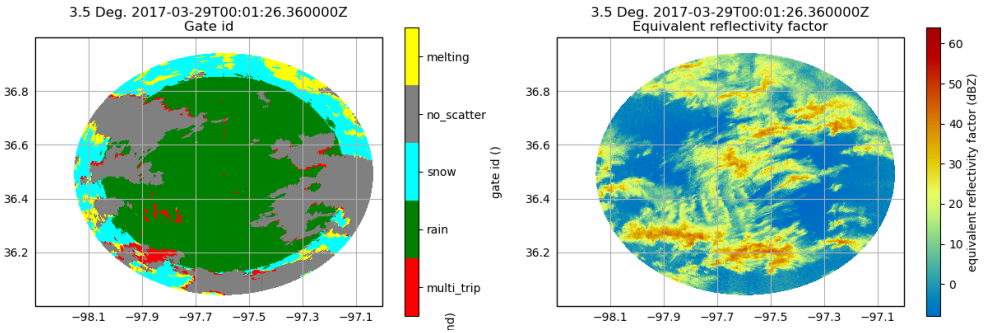
THE PYTHON ARM RADAR TOOLKIT

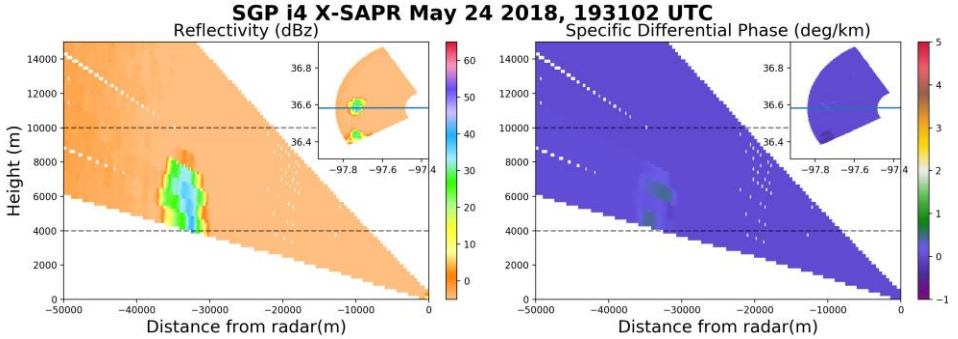
Data model. <= From scratch would use X-Array (todo)

- Py-ART's central core is a data model for gated data with pointing information.
- Py-ART created a way of representing radar data in the Python programming language that mirrors the CF-Radial standard.
- Py-ART has a cloud functions to correct, retrieve and grid radar data.
- By keeping a limited scope Py-ART aims to "do less better".
- There is now a **rich ecosystem** of packages that interact: ART-View, CSU tools, PyTDA, Multidop to name a few. Oh and we are working on TINT!

```
In [2]: import pyart
        radar = pyart.io.read('/data/cmac/raw/XSW170519002005.RAW54Y0')
        print(radar.ngates, radar.nrays, radar.nsweeps)

501 9200 23
```





Animation courtesy of users Marcus van Lier-Walqui and Sara E. Lytle
Data: Andrei Lindenmaeir – ARM Mentor

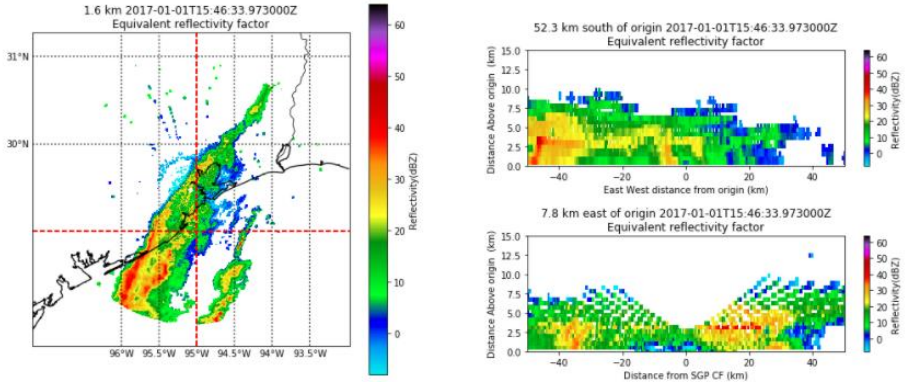
TYPICAL USAGE OF PY-ART

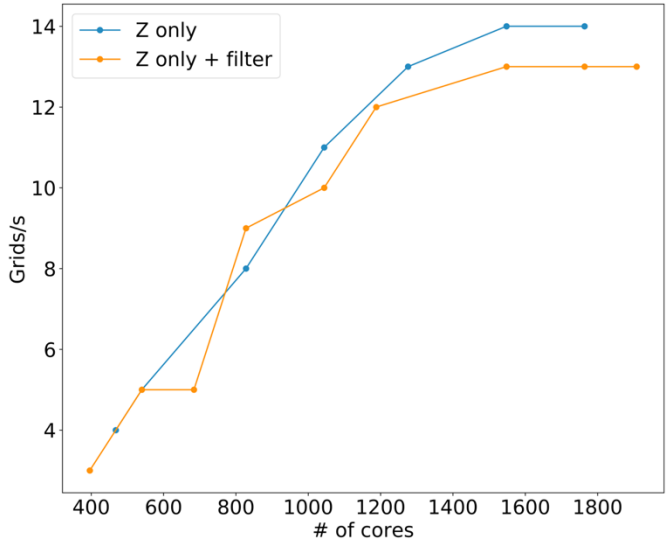
Notebooks – Single file application Scale to a cluster

- Jupyter notebooks give you the ability to interact in a tight feedback cycle with radar data.
- Great for adjusting parameters like membership functions or weights.
- Get that right for a single file and then port (using your favorite IDE, PyCharm!) to an executable.
- Many of us have access to clusters or at least multi-core machines. Tools like IPyCluster and Dask allow single “granules” to be mapped to many workers.
- ~20M voxels/grid = ~300M voxels/s inc. reading wsr88d and writing CF-Radial.

```
gf = pyart.filters.GateFilter(radar)
gf.exclude_masked('reflectivity')
```

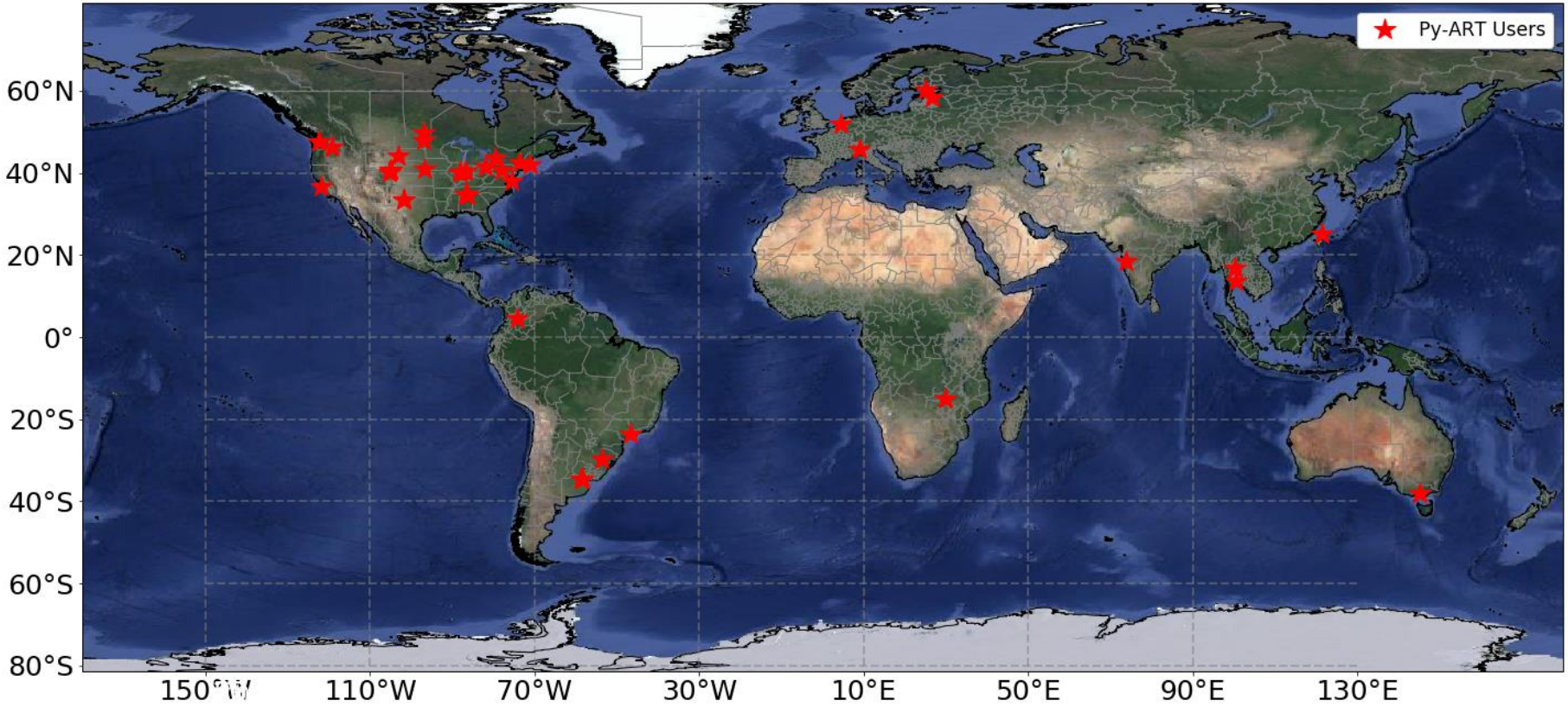
```
grids = pyart.map.grid_from_radars(radar,(31,801,801),
                ((0.,15000.),(-200000.,200000.),(-200000.,200000.)),
                fields=['differential_phase', 'differential_reflectivity',
                        'cross_correlation_ratio', 'reflectivity', 'unfolded_differential_phase',
                        'velocity', 'cylp_processed_phase', 'spectrum_width',
                        'specific_differential_phase'],
                refl_field='reflectivity',roi_func='dist_beam',
                h_factor=0.,nb=0.6,bsp=1.,min_radius=200.,
                            gatefilters=(gf,))
```



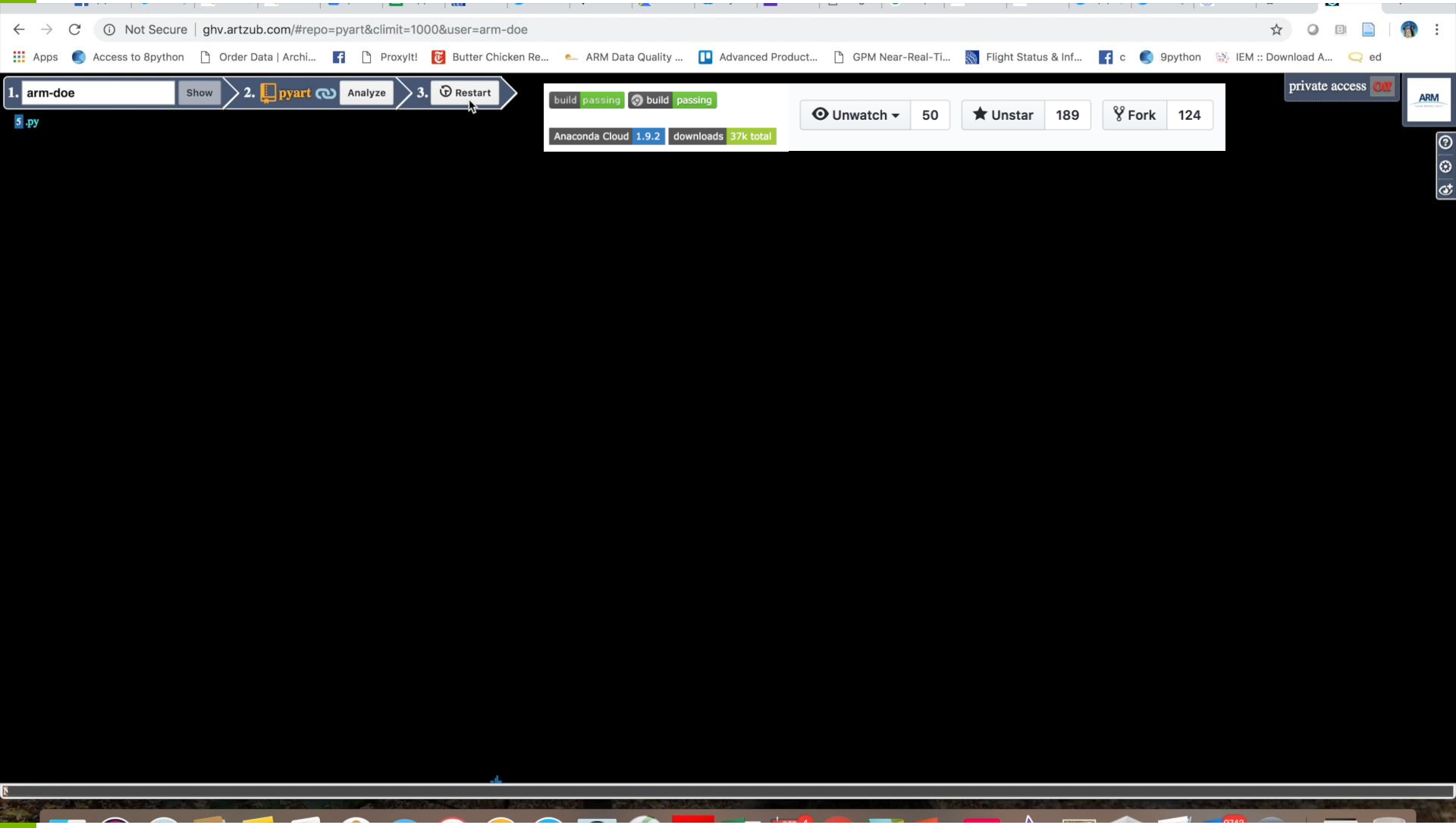


PY-ART IS GLOBAL

Self reported installations of Py-ART



The power of community software.. But, you need standards, unit testing and continuous integration.



Only scollis, jjhelmus, zssherman, rcjackson and rumpkie have received ARM funds. The other 28 have not…

Argonne NATIONAL LABORATORY


THE UNIVERSITY OF ALABAMA IN HUNTSVILLE
PURDUE UNIVERSITY
COLUMBIA UNIVERSITY
TARTU ÜLIKOOL 1632 UNIVERSITAS TARTUENSIS
NATIONAL WEATHER SERVICE
UNIVERSITY OF Nebraska Lincoln
WL
HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI
Universidade Federal de Santa Maria
1960
CLIMACELL
Weather Revealed
Australian Government
Bureau of Meteorology
Argonne NATIONAL LABORATORY
DEPARTMENT OF ENERGY UNITED STATES OF AMERICA
BROOKHAVEN NATIONAL LABORATORY
FLEET NUMERICAL METEOROLOGY AND OCEANOGRAPHY CENTER MONTEREY, CA
MeteoSwiss
THE CLIMATE CORPORATION
THE UNIVERSITY OF QUEENSLAND AUSTRALIA
Universität Potsdam 25 JAHRE
NASA
Pacific Northwest NATIONAL LABORATORY
Proudly Operated by Battelle Since 1965
TEXAS TECH UNIVERSITY
UNIVERSIDAD DE BUENOS AIRES
ILLINOIS
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN
UND
UNIVERSITY OF NORTH DAKOTA.
The UNIVERSITY of OKLAHOMA
INSTITUTO DE ASTRONOMIA, GEOFÍSICA E CIÊNCIAS ATMOSFÉRICAS
Colorado State University
universitätbonn
UNIVERSITY OF WYOMING
Environment Canada
USP Universidade de São Paulo
UNIVERSITY of WASHINGTON
McGill
SIMEPAR Sistema Meteorológico do Paraná
PennState
Servicio Meteorológico Nacional
Ministerio de Defensa Presidencia de la Nación
IBM
UNIVERSITY OF LEICESTER
meteo.cat
Servei Meteorològic de Catalunya
MONASH University
unidata
CRAY THE SUPERCOMPUTER COMPANY
Sky ECHO HD weather nowcasting
國立臺灣大學 National Taiwan University
INDIAN INSTITUTE OF TROPICAL METEOROLOGY PUNE

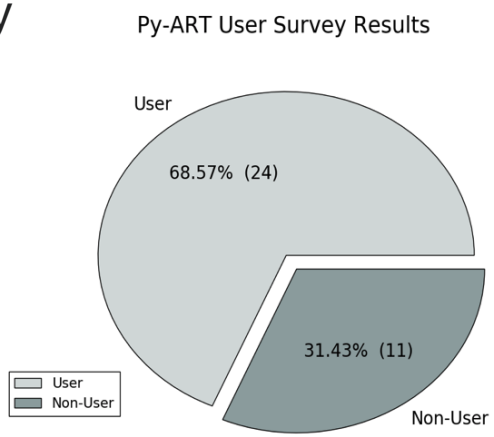
PY-ART IS FREE AS IN LIBRE NOT AS IN BEER

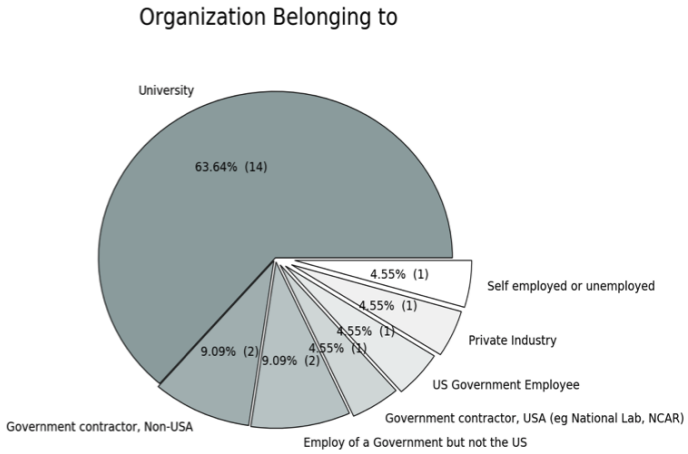
- Making radar codes open source is not free. Non-ARM funded people put in funded and spare time and ARM funded folks have a line item for upkeep.
- We have many automated tools to check when things break.. But things still need fixing. Although we always work to minimize critical failure paths.
- To this end we need to ensure funds spent benefit ARM and its stakeholders. The Py-ART roadmap aims to do this.

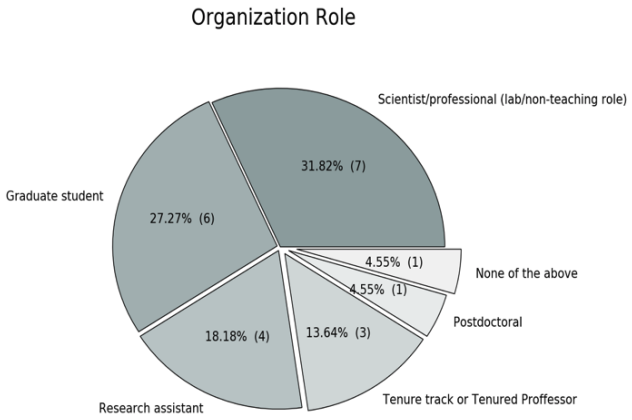
https://commons.wikimedia.org/wiki/File:Isummit_2008,_Japan,_free_beer.jpg

THE ROADMAP: MATCHING COMMUNITY NEEDS TO ARM NEEDS

- The Py-ART roadmap outlines a five year vision for Py-ART.
- Initial drafting was guided by a survey of stakeholders.
- The roadmap was reviewed by the radar science group and by four specifically selected ASR Pis who provided open reviews.
- The roadmap was then revised. In addition a section was written detailing how it changed.







KEY ITEMS.

- Improved Quality Control (QC) algorithms that can be used to create workflows for building more user accessible radar data.
- Full support for the emerging Cartopy mapping engine ensuring sustainability of Py-ARTs geospatial visualization tools.
- Better Documentation, examples and a set of tutorials and courses to allow easy delivery of learning using Py-ART.
- An ingest of WRF produced NetCDF thus allowing efficient comparison between model and radar produced fields.
- Work with a third party application to produce cell tracks. Support this effort with visualizations.

GOVERNANCE

- **Science Lead**: Provides high level leadership for the project, organizes outreach and education, and coordinates contributor and stakeholder input to form a long term vision for the project. The Science Lead will also coordinate reviews of the science behind a pull request where some claim has been made.
- **Lead Developer**: Responsible for overall architecture of the project. Final arbiter in what pull requests to accept. Develops the required style guidelines and coordinates the associate developers. Coordinates contributions from associated developers to a Contributors Guide (and contributes as well).
- **Associate Developers**: Responsible, as time allows, for doing an initial check of pull requests for suitability and adherence to the Contributors Guide. Contributes to the Contributors Guide.

OPEN RADAR PARTNERSHIP

We are all in this together…

- Python efforts in Radar Meteorology nucleated in parallel in Europe and the USA.
- In addition the long standing TITAN software cloud received NSF support for a modernization surge.
- A short course at ERAD 2014 led to a co-written paper which lead to a rotating course between ERAD and the AMS radar conference.
- We now have a web presence and a growing ecosystem of identified packages.

ARM

L-ROSE

baltrad

radlib

THE EMERGENCE OF OPEN-SOURCE SOFTWARE FOR THE WEATHER RADAR COMMUNITY

BY M. HEISTERMANN, S. COLLIS, M. J. DIXON, S. GIANGRANDE, J. J. HELMUS, B. KELLEY, J. KOISTINEN, D. B. MICHELSON, M. PEURA, T. PFAFF, AND D. B. WOLFF

OPEN RADAR PARTNERSHIP

We are all in this together…

- Python efforts in Radar Meteorology nucleated in parallel in Europe and the USA.
- In addition the long standing TITAN software cloud received NSF support for a modernization surge.
- A short course at ERAD 2014 led to a co-written paper which lead to a rotating course between ERAD and the AMS radar conference.
- We now have a web presence and a growing ecosystem of identified packages.

THE EMERGENCE OF OPEN-SOURCE SOFTWARE FOR THE WEATHER RADAR COMMUNITY

BY M. HEISTERMANN, S. COLLIS, M. J. DIXON, S. GIANGRANDE, J. J. HELMUS, B. KELLEY, J. KOISTINEN, D. B. MICHELSON, M. PEURA, T. PFAFF, AND D. B. WOLFF

ARM

L-ROSE

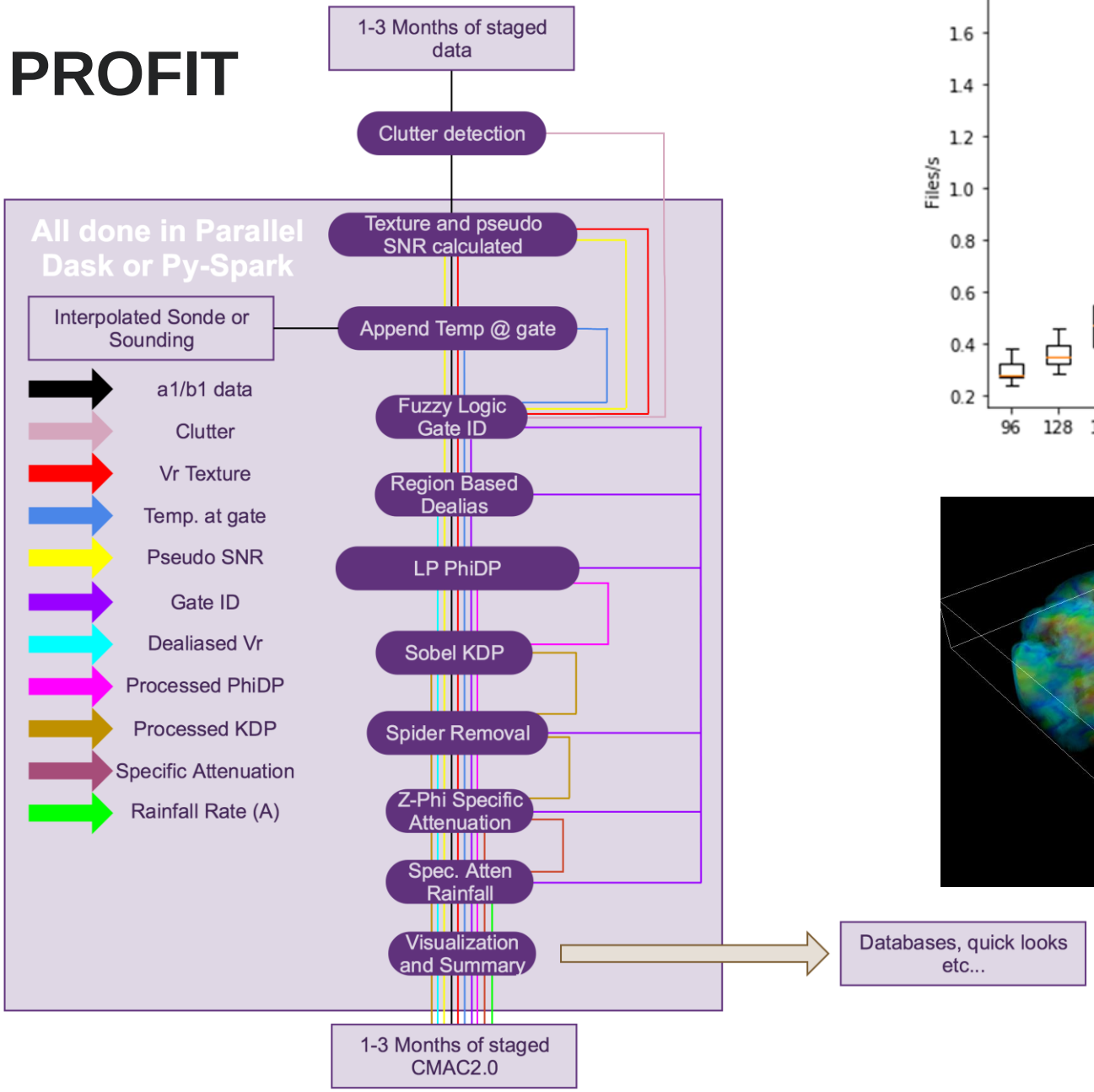
baltrad

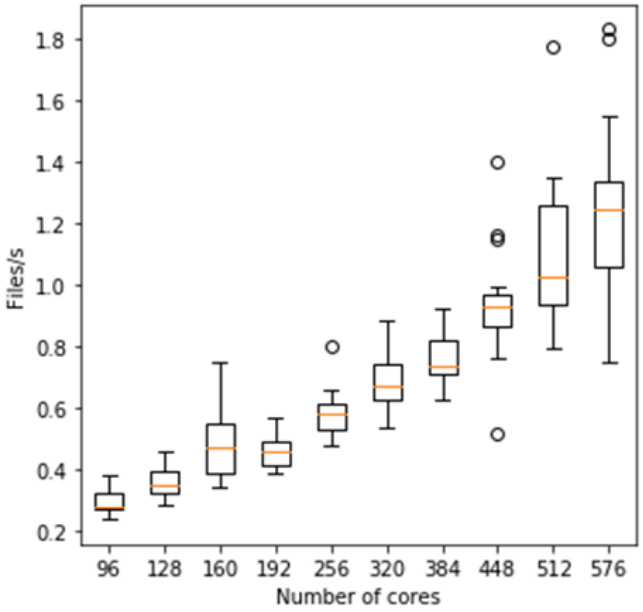
radlib

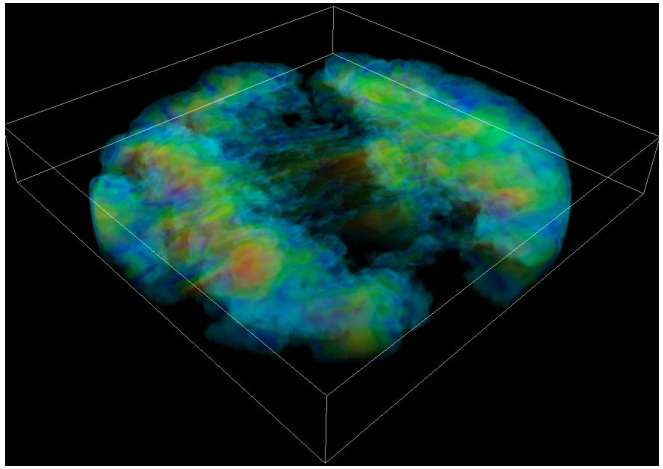
V1.2 out NOW!

Argonne NATIONAL LABORATORY

PROFIT



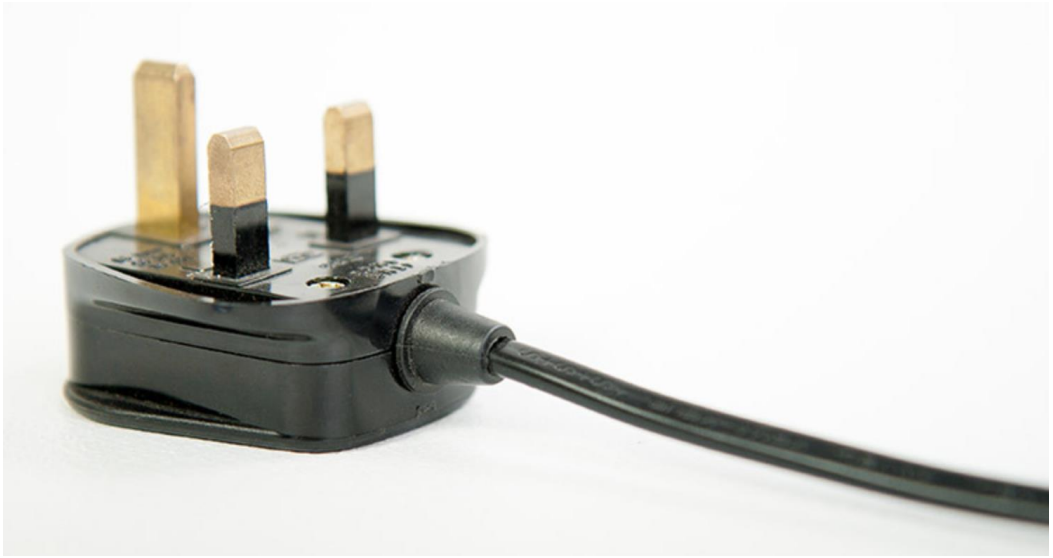




Argonne NATIONAL LABORATORY

A QUICK PLUG

- Scipy 2019: Early stages of planning, July 8-14, Austin, USA.
- AMS Python Symposium 2019. Abstracts closed. January 7-9, Phoenix, USA.
- AMS Python Symposium 2020. Week of 12th of January, Boston, USA.



COME ON IN!

- Official source code repository: https://github.com/ARM-DOE/pyart
- HTML documentation: http://arm-doe.github.io/pyart-docs-travis/
- Examples: http://arm-doe.github.io/pyart/dev/auto_examples/index.html
- Mailing List: http://groups.google.com/group/pyart-users/
- Issue Tracker: https://github.com/ARM-DOE/pyart/issues
- Facebook: https://www.facebook.com/PythonRadar/
- Twitter: https://twitter.com/py_art?lang=en

OPEN RADAR PARTNERSHIP: HTTP://OPENRADARSCIENCE.ORG/

THANK YOU FOR YOUR TIME

SCOLLIS@ANL.GOV

This presentation has been created by UChicago Argonne, LLC, Operator of Argonne National Laboratory (“Argonne”). Argonne, a U.S. Department of Energy Office of Science laboratory, is operated under Contract No. DE-AC02-06CH11357. This research was supported by the Climate Model Development and Validation activity funded by the Office of Biological and Environmental Research in the US Department of Energy Office of Science. Computing resources were provided by the Laboratory Computing Resource Center of Argonne National Laboratory.