

Atmospheric Community Toolkit (ACT) Roadmap 2025-2030

Zachary Sherman, Adam, Theisen, Joseph O'Brien, and Robert Jackson

Introduction and Aims

The Atmospheric data Community Toolkit (ACT) is an Atmospheric Radiation Measurement (ARM) user facility supported python library for working with atmospheric time series and multidimensional datasets. Similar to Python ARM Radar Toolkit (Py-ART)(Helmus and Collis 2016), ACT aims to reduce redundant coding by individual researchers and focus that effort into developing capabilities that can be utilized by the broader community. ACT roadmaps are typically structured in five-year horizons, helping prioritize development and align work with ARM and community needs. With the previous roadmap complete, this document established the next five-year roadmap for ACT.

ACT has benefited from code from 16 individual contributors. These contributors with their commits over time can be seen in **Appendix 2**. This has been enabled by careful implementation of unit tests and continuous integration. Every time a pull request is submitted against the ACT codebase a set of tests run and a report is generated so the developers know if a contribution causes any unit tests to fail. The lead developer as well as the associate developers, review these pull requests and guide the user submitting the code, to make the code acceptable for ACT. When the code is acceptable, it is merged into ACT.

Version 2.0.0 of ACT was released in January of 2024. This release saw another overhaul of the documentation which included separation of examples into specific categories. This version of ACT also saw the movement of test data into its own repository. This was to reduce the download size of the package. This next roadmap will focus on new goals for ACT, from feedback from stakeholders, as well as survey data, but also what the developers have observed over the previous 5 years.

Value of ACT to ARM

Fostering collaboration and shared code development represents one of the primary benefits of ACT to the community. ACT enables individual groups to move beyond isolated development efforts and instead connect with and build upon work from other teams. By doing so, both the science and infrastructure communities can reduce redundant efforts and more effectively advance capabilities by building on a shared foundation, rather than recreating core components from scratch. ACT also promotes transparency in processing workflows and decisions embedded within. ACT has been adopted across many groups both within ARM and the broader community in DOE with over 60 additional code repositories on GitHub utilizing ACT. ACT is being used to streamline workflows, increase transparency, and enable users to directly leverage these codes and contribute improvements back into ARM operations.

The ACT Roadmap Survey and Reviews

Previous ACT roadmaps have been informed by broad input from users, non users, academia, ARM and more. Feedback for the previous roadmap highlighted a strong need for tools that support data access from external APIs such as EPA's AirNow and NOAA FTP servers as well as additional readers such as ICARTT. These requested capabilities were subsequently implemented. While the ACT development team may propose new features based on technical opportunities or emerging trends, priorities are ultimately guided by ARM and community feedback. Survey response and broader community engagement have therefore played a central role in shaping ACT development and ensuring that effort is aligned with demonstrated needs.

As with past roadmaps, a community survey was conducted to capture user and stakeholder perspectives on the future direction of ACT. Based on lessons learned from the previous effort - where survey length likely limited participation - the current survey was intentionally streamlined. The survey was shared by the ASR community as well as in the form of a QR code in posters and presentations throughout the year including the BNF ARM summer school. In total, the survey had **51 respondents**, with most of the respondents from University or other academic settings and in the role of scientist or professor. (Fig. 1, Fig. 2).

What type of organization do you belong to? (Optional)

51 responses

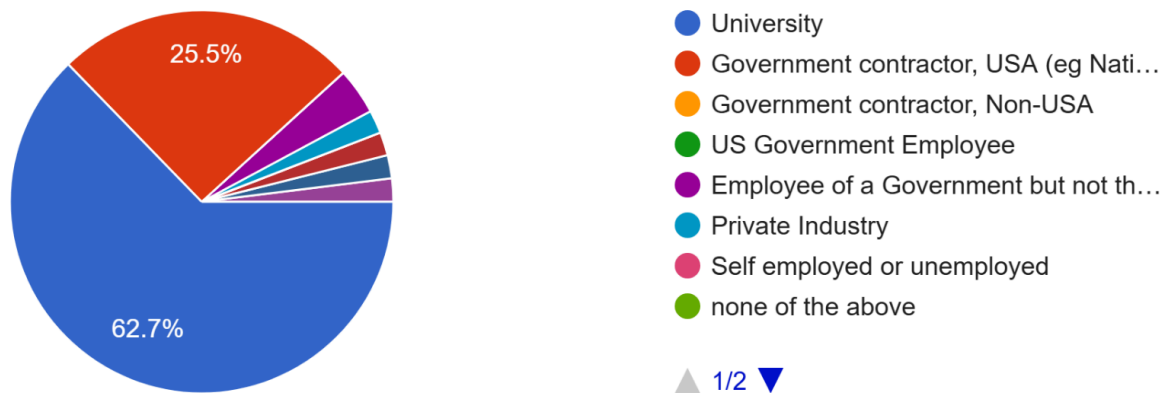


Figure 1: Organization of respondents.

ACT is taught and demonstrated through multiple community training venues, including the PNNL instrument course and the ARM Summer School. Given this outreach, it is not surprising that respondents from university and academic roles represent a large portion of the survey. The survey also showed a large group of respondents from the scientist/professional role in government, which could be many individuals from the ASR community.

What is your role within your organization? (Optional)

51 responses

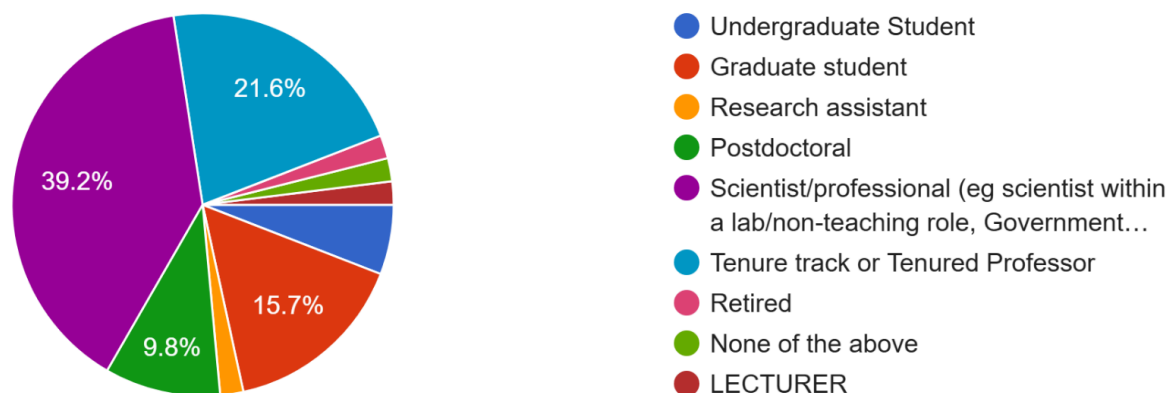


Figure 2: Field of work or study of respondents.

Non ACT Users Survey

First we will start with non users to gauge the barriers to using ACT as well as possible changes to attract more users. Most respondents were non-users with 37 total of the 51 responses we received. With the most common responses from the non-users, we will provide some possible solutions to address why there are many non-users and why they might not use ACT.

The overwhelming response chosen the most to being a barrier to using ACT was that the non-user “Honestly, I have never heard of it”, followed by ‘I have my own software’ **Figure 3**. To address survey respondents never hearing about ACT, we can continue to showcase ACT in tutorials such as the ARM summer school while also utilize ARM communications to showcase ACT as well. Users having their own software is not necessarily a point that needs to be addressed, but if we can provide or even integrate tools with ACT, that might help expand ACT to those respondents.

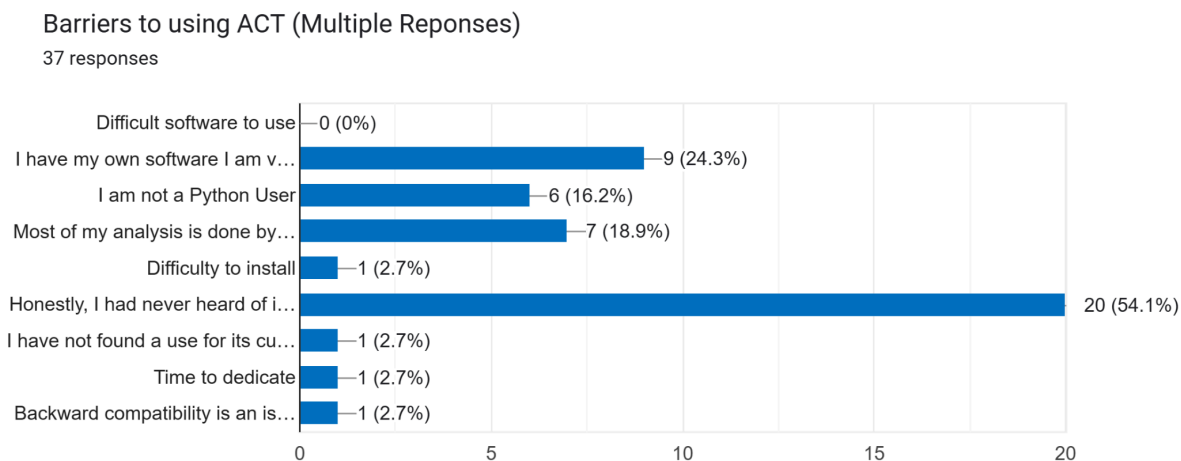


Figure 3: Barriers to using ACT.

The question was also asked, ‘What would get you interested in using ACT?’ **Figure 4.** Overwhelming with 25 responses, “More tutorials on Python ACT” was the most selected response followed by “New reader for a specific dataset”.

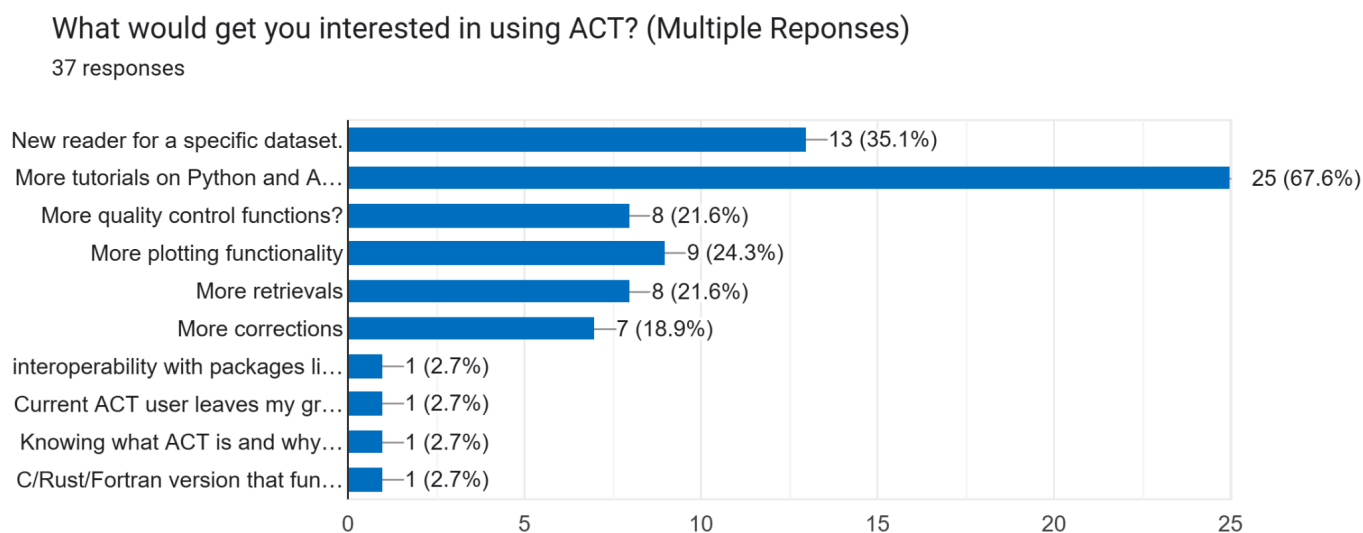


Figure 4: Non-user responses on what would get you interested in ACT.

Again, providing more tutorials for ACT and Python and the ARM summer school as well as getting involved with other events such as the PNNL instrument course or Scipy would be beneficial for increasing the user base. Open-ended responses as well mimic many of the responses in the figures above, that ACT is unknown to many of the respondents or they require tutorials and use cases **Table 1**.

These responses have been quite useful and give the developers insight on how we can expand the user base for ACT for not only those who have not heard of ACT, but also those who have but have barriers to use it.

Table 1: Barriers to using ACT open-ended responses.

If a barrier to using ACT hasn't been mentioned, what is a barrier that is stopping you from using ACT? (Optional)
Time to sit down and learn how to use it! If only there were 25hours/day.
Maybe
I'm not sure what ACT does, or why I need it.
I am often not sure what is available for the different datasets I use. It would be nice if DataDiscovery lists which ACT functions are available for each dataset.

ACT Users Survey

Another part of the ACT survey is asking current users different questions about ACT as well. We received 14 responses from users for this survey. One of the questions we ask is how long users have been using ACT **Figure 5**. The reason we ask this question is to see if users have been increasing over the past year and how many users continue to use ACT.

How long have you been using ACT? (Optional)

14 responses

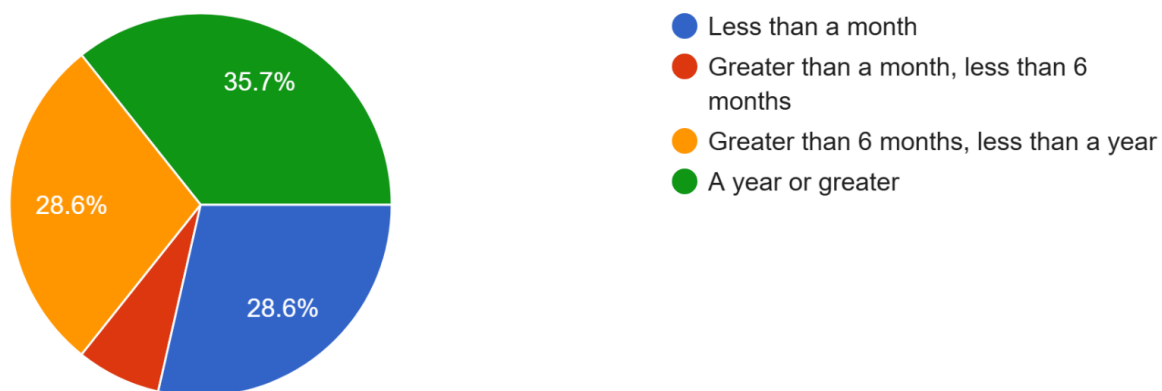


Figure 5: How long have you been using ACT.

We also gauge what users are using ACT for and also what their favorite features are (**Figure 6**). The reason we ask these questions is to not only see which features are the most used within ACT, but also to know which features to prioritize maintenance and development to ensure that user workflows are not interrupted. The most popular features are plotting, downloading ARM data, and aerosol research.

What is your favorite/most beneficial feature within ACT? (Optional).

13 responses

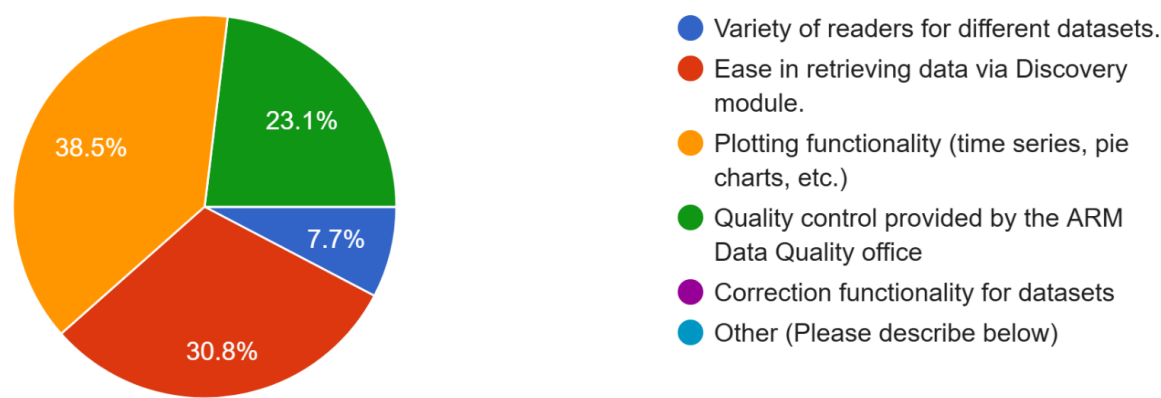


Figure 6: Favorite/most beneficial features in ACT according to respondents.

Another question previously asked in prior roadmaps was “What the user would like to see added to ACT”. Many of the responses from the previous survey in the first roadmap on features requested now exist in ACT such as NOAA FTP discovery tools and different readers such as ICARTT. Gaining this feedback from the previous roadmap has allowed the developers to prioritize these new features and add them to ACT. We then again asked this question as an open response **Table 2**.

Table 2: Requested features by respondents to be added to ACT.

What additional features would you like to see in ACT, whether it's new readers or discovery functions or new retrievals or types of visualizations? (Optional)
new retrievals
Feature detection techniques.
More ways of plotting XSACR data

It save me a massive amount of time to have a discovery function to browse ARM data tags. E.g. What TRACER radar data is available between date A & B? Sometimes it takes a long time to debug why the discovery retrievals don't work after looking at data on the PlotBrowsers

Workbooks for BNF tower measurements

Summarizing new features, it seems users want more plotting functionality for X-SACR data, new retrieval methods for working with their data, giving advanced search options for our discovery tools, and examples for working with BNF data. These features, except for radar data which can be handled by Py-ART, requests all seem reasonable and can be added to the roadmap priority list for development.

To streamline the user experience, we also asked an optional question on what examples in the ACT documentation would the user want to see more of **Figure 7**. Plotting was the highest chosen answer followed by corrections and retrievals. When we do add code to ACT we usually add examples as supplements to that new code, however there are areas of the codebase that could be expanded on for examples, or specific use cases that the user might want to see. By adding these new examples, it can also reduce the barriers to using ACT as well.

With ACT's documentation, what examples would you like to see more of? (Optional)

11 responses

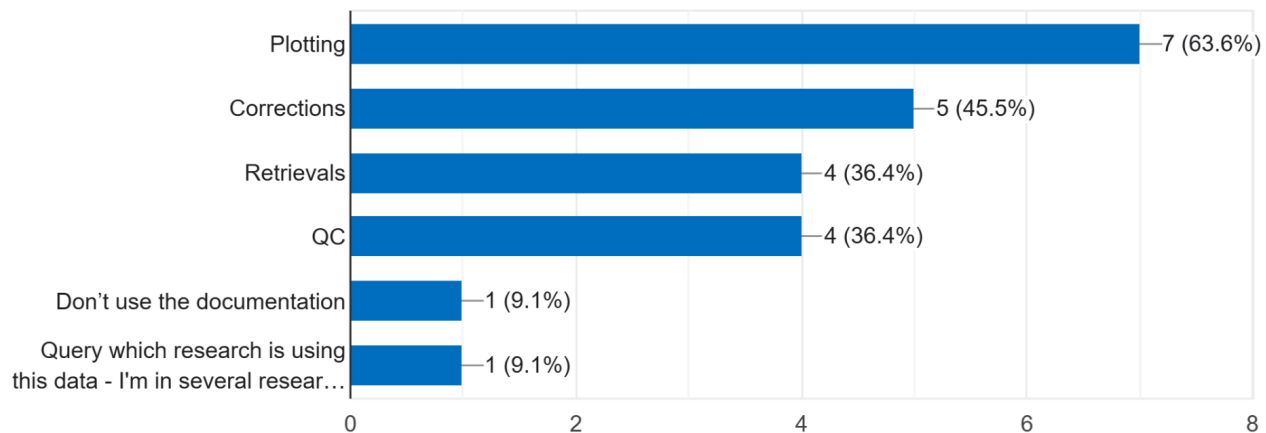


Figure 7: What examples would the user want to see more of.

Finally we asked “Are there any barriers to you contributing to ACT” **Figure 8**. ACT is an open source package, and ideally we would want users to contribute as well, especially if there is a specific use case that does not exist within ACT that could be useful for the broader community. Most responses seem to point to time constraints or nothing worth contributing. We can help the user by updating our guidelines and instructions to help the user. We can also provide Git and GitHub tutorials in our guides for ACT.

These questions allow us as developers to develop ACT in a manner that not only has ACT as a premier package for Atmospheric research, but also to help streamline the user experience while still meeting the needs of the stakeholders.

Are there any barriers to you contributing to ACT? (Multiple Responses) (Optional)

14 responses

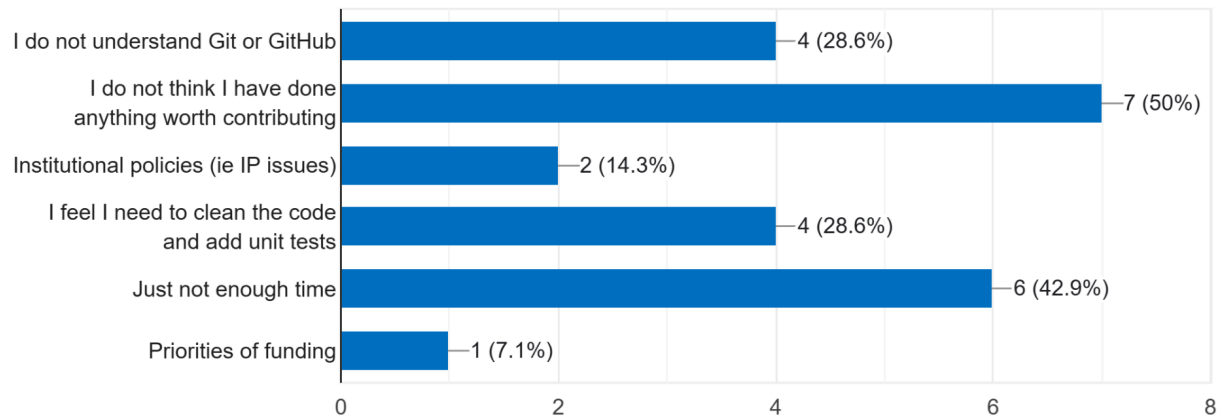


Figure 8: Barriers to contributing to ACT.

Modified Governance Structure

As with previous roadmaps, there is a need to ensure that this effort is responsive to the needs of stakeholders that rely on ACT. The proposed governance structure needs to be flexible and have the ability to expand as the capabilities of the toolkit expand. The roles have not changed since the previous roadmap. The roles required are:

Science Lead: Provides high level leadership for the toolkit, 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 to ensure accuracy to the literature. The science lead will make a judgement on if a pull request requires more review or can be accepted as is.

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). Responds to users on the GitHub issue tracker with the assistance of the associate developers.

Associate Developers: Responsible, as time allows, for doing an initial check of pull requests for suitability and adherence to the Contributors Guide. Associate developers should come from a diverse background to ensure there is no single point of failure in providing support to ACT. The

number and areas of expertise of the associate developers will need to increase and adapt as ACT grows.

Overarching Goals for the Next Five Years

The aim of ACT is to lower barriers to doing science with atmospheric data, in particular for Department of Energy stakeholders. The major goal of this roadmap will be to increase the number of users as well as providing the users with tutorials and examples as most respondents were unaware of ACT or have not used it before. One of the most requested features was increased documentation and examples and this was despite overhauling the documentation during the last roadmap. Despite not seeing responses with regards to AI, us as developers have noticed a trend in the open source space of AI generated pull requests, some either not useful and others possibly malicious. We need to create a guideline in how to approach these PRs as they will most likely grow over time due to exponential growth of AI usage.

Priority Features Summary

The Development team will prioritize the acceptance of Pull requests and perform targeted strategic development that adds the features outlined in the following subsections. As alluded to in the descriptions from previous sections ‘Highest priority’ means that ARM will accept pull requests that need significant (more than a few days) work or even perform some ARM funded work ourselves. "Moderate priority" means we will accept pull requests that may require some clean up and minor development. "Lower priority" are items where we will only advise the requester on changes required. This will change based on reviews from ARM stakeholders. Continued development, such as maintaining continuous integration, bug fix, general maintenance is expected and will be excluded from priority of features below.

Highest Priority

1. **Improved documentation and examples:** Improve ACT’s documentation to include more user cases. Many users might have specific use cases that are not represented in the ACT documentation. The developers can work with these users in getting new examples created. This could help alleviate many issues that are opened on ACT’s GitHub issue tracker as well when users have specific use cases. Ideally we want examples also representing each ARM campaign such as BNF.
2. **Tutorials for yearly summer school and other events:** Continue creating tutorials for ACT and teaching these tutorials at events hosted by ARM. This can help expand the user base.

3. **New documented procedures and automated countermeasures against the rising issue of AI generated content:** Over recent years, we have seen this issue in Py-ART, but not in ACT as of yet, but Py-ART has had issues and pull requests opened, some possibly AI generated. As the use of ACT grows as does its community, the same issue might occur for ACT and we should ensure that these issues are legitimate and not malicious, we need procedures and documentation alongside unit testing, CI, etc. to ensure that these problematic issues or pull requests are caught.
4. **Continued integration and support with ARM systems:** To meet the needs of the stakeholders, we need to continue to support ACT's usage within the ARM/ASR community. An example of this is continued support of the DQ office who relies heavily on ACT and assisting these individuals.
5. **Continue support for ARM's aerial facilities:** With a new ARM aircraft coming online in 2027 and growing popularity of the tether balloon system, there needs to be continued support for these datasets. New modules and maintenance of current modules will be a high priority.

Moderate Priority

1. **Increased inputs and outputs:** We had a few responses on working with different datasets, however specific datasets were not provided. When these issues arise, we can work with the user in adding these new readers.
2. **Growth of retrievals and correction suites:** ACT's suite of corrections and retrievals has quite few algorithms to improve on datasets, however we should continue updating these modules based on new research and methods.
3. **Data provenance:** Create methods or ways of tracking how data is created or modified in ACT. Possibly creating a new attribute in these datasets that show what methods were applied to the dataset.

Low Priority

1. **Specific discovery selection:** We had one user who specified adding more specific data selections for ACT's discovery tools. As developers we will eventually want to expand the discovery module, however for the moment it is a low priority.
2. **Integrating dependencies within ACT:** Over the past few years of development, we have added a few dependencies for dealing with different datasets, such as ICARTT. Some of these packages are either no longer maintained or need assistance. We might want to reach out to these developers in adding these readers into ACT directly.

3. **DQ Inspector Replacement:** Continue support of the command line tool DQ inspector tool replacement.

Measuring Impact

Developing a roadmap alone does not fully capture the value or success of a Department of Energy-supported software project. To ensure sustained impact and responsible stewardship of resources, ACT must also demonstrate measurable outcomes that reflect both community adoption and scientific utility. Tracking these outcomes provides a means to evaluate progress against the roadmap, guide future development priorities, and justify continued investment. Consistent with practices established during the initial ACT roadmap and informed by impact assessments used for Py-ART, the success and influence of ACT can be assessed using three complementary metrics:

1. **Growing the number of users and installs:** Users and installs can be tracked by the routine statistics we gather from the GitHub site on clones and views, the visitors to the documentation pages as tracked by Google Analytics, and the interest received on GitHub through the use of stars.

Figure 9 shows that average daily unique viewers and cloners and clones have increased since statistics started being tracked. While total views are down on the repo, it is up significantly for ACT's documentation. **Figure 10**, shows that users of ACT's documentation is up over 130% in 2025 compared to the previous period with significant increases in traffic towards the end of 2025. Additionally, GitHub stars have been [consistently increasing](#).

As of writing, downloads of ACT from Anaconda total 185,000 while downloads from pip equal 2,800/month. Note that these are inflated due to unit tests being run on each pull request and merge.

Average Daily Clones, Unique Cloners, Views and Unique Viewers

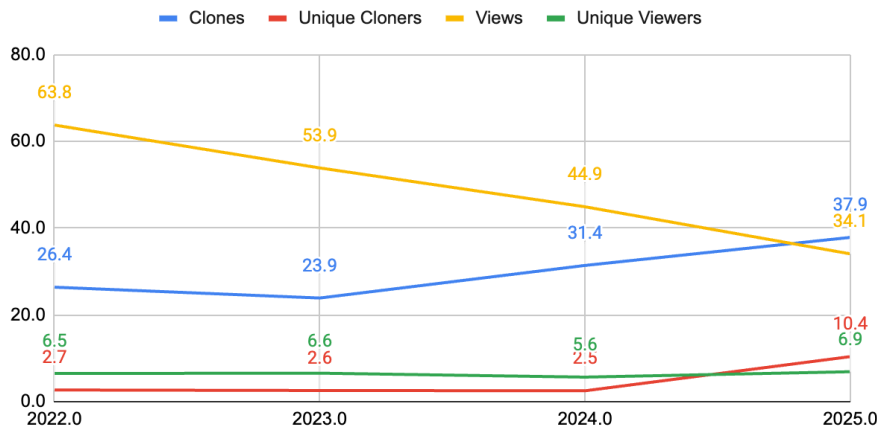


Figure 9: Average daily statistics from GitHub.

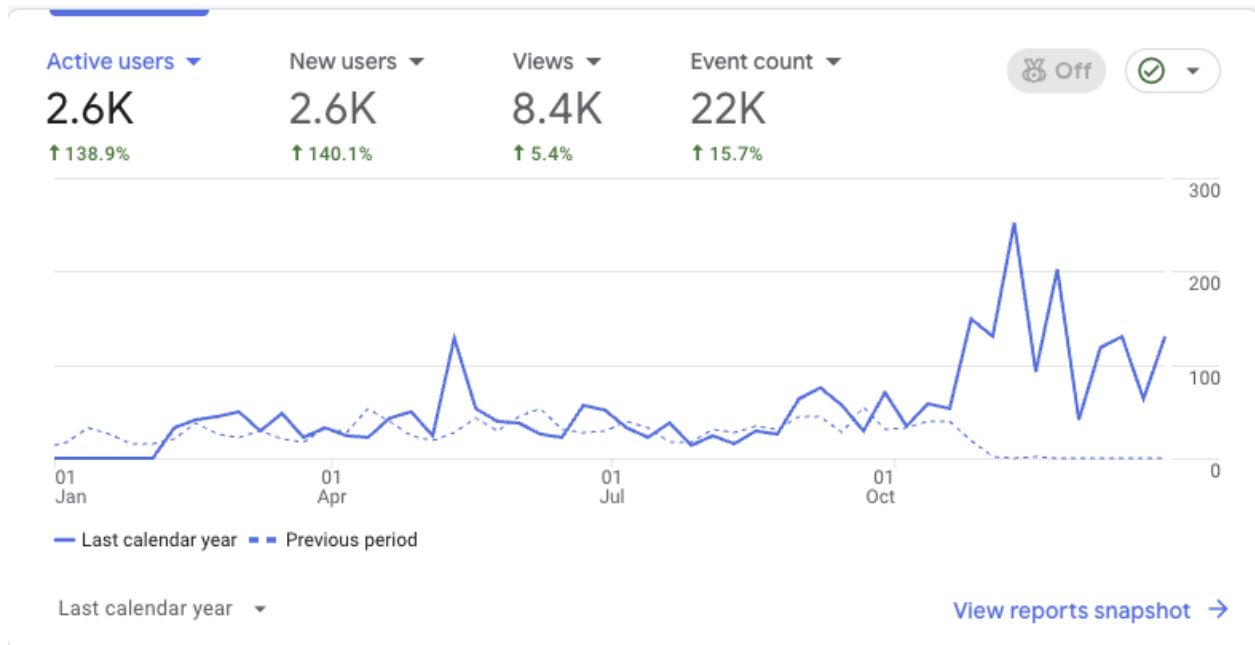


Figure 10: Visits to the ACT documentation pages as reported by Google Analytics.

- Number and success of dependent projects:** Over time the number of dependent repositories who use ACT as a dependency has increased, especially between the previous and current roadmap **Table 3**.

Table 3: Number of dependent repositories for each roadmap.

Roadmap	Number of Dependent Repositories
V1.3	9

V2	14
V3	63

3. **ACT contributors and citations:** As noted earlier, there are 16 contributors to ACT ranging from ARM mentors to developers to ADC staff and DQ Office staff. ACT uses Zenodo for tracking its citation and contributor information. As new contributors add to ACT, in newer releases Zenodo updates the author list automatically. There is currently one ACT citation. More information can be found here:

<https://zenodo.org/records/16423448>

Education and Outreach

The ACT examples and blog posts have been growing. There have also been many tutorials featuring ACT such as the PNNL atmospheric instrument course, the University of Puerto Rico short course, and the ARM summer school. There has also been a lot of education and outreach through conferences, working with others in the programs, and providing sample codes to summer interns at multiple labs.

Overall, not all the goals were fully realized but the roadmap was still very successful in driving development efforts. It should be noted that the ACT budget is for the general maintenance, pull request response, and education and outreach. Development efforts have generally been performed with any remaining effort which is part of the reason for not fully realizing the roadmap.

References

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>