



UNITED STATES DEPARTMENT OF COMMERCE
National Oceanic and Atmospheric Administration
Office of Oceanic and Atmospheric Research
Earth System Research Laboratory
325 Broadway – David Skaggs Research Center
Boulder, Colorado 80305-3328

MEMORANDUM FOR: John Clyne, VAST Section Head, NCAR

Mary Haley, NCL Project Lead, NCAR

FROM: NCL Advisory Panel

SUBJECT: NCL Advisory Panel Report

This letter is being sent on behalf of the NCAR Command Language (NCL) Advisory Panel, to report our findings from the 1.5-day meeting hosted at the NCAR Mesa VisLab August 2–3, 2018.

Integrating NCL into the ambient Python ecosystem is necessary for maintaining NCL’s utility and relevance as a tool for data analysis and visualization going forward. This realization comes from a recognition that the greater scientific community is moving towards Python and that early career scientists are increasingly likely to be Python users. As a community resource, the NCL team plays a vital role in enabling community science, which is a fundamental NCAR/UCAR mission as a national center. The proposed ‘pivot to Python’ will allow NCL to ride this wave by ensuring that young scientists continue to have access to the vast utility of the NCL core language. A key element in the popular movement toward Python is the culture of open development where users have access to (and can contribute to) third-party modules that are designed to play well together and that can be called by multiple languages. Within that context, a pivot to Python and a simultaneous move to open development builds a framework to ensure that the graphical and computational abilities provided by NCL, particularly those unique to atmospheric and earth system sciences, are continued into the future irrespective of the language in vogue. The process of transitioning to Python requires that NCL functions be packaged into libraries that can be read by Python and, if developed in an agnostic framework as proposed by the NCL developers, can be called by other languages now and into the future.

By embracing an open development model within the Python ecosystem, NCL will gain two additional benefits: (1) NCL can leverage existing third-party tools, eliminating the need to support in-house tools that are effectively redundant, and (2) outside contributors can augment the development efforts of the in-house NCL team. Given the enormous scope of scientific analysis and visualization, coupled with realistic funding expectations going forward, NCL will



benefit from embracing an open development model where some of NCL's current functionality can be ceded to existing software, allowing NCL to better focus on supporting the core computational and visualization strengths that best serve the NCL user community.

The intent of this document is to report on the findings of the NCL Advisory panel, including advising the NCL development team on community priorities, and on how best to serve the (very committed) NCL user community during this necessary period of transition to Python, understanding that there will be a perceived cost to the user community for moving to a Python ecosystem where there is little defined user support. Support for these findings comes from the NCL user survey conducted May 2 – June 5, 2018. Over 7300 emails were sent, and 699 respondents completed the survey. In broadcasting the survey to the community, the NCL team was able to reach a diverse audience, including Python users, allowing the NCL developers to:

- Better understand the workflow of NCL and Python users
- Identify NCL features that are missing from Python tools
- Identify which NCL features are considered to be crucial by users
- Determine what other software tools are being used
- Solicit general thoughts on open development

Two key outcomes from the user survey were that (1) the NCL user community is active and thriving, and that (2) user-support was key to this continued success and vital for the scientific researcher community. This support spans many things: email, help desk, documentation, manuals, workshops and online examples. These strengths will be crucial for NCL's successful 'Pivot to Python', discussed in the findings below. We break these findings into four broad categories: (1) support for NCL core language, (2) support for the Python transition (including messaging), (3) transition to open development model (including governance), (4) what functionality can be deprecated/downgraded.

Our findings are as follows:

1. Continue Support for the NCL Core Language:

The findings of our panel include the strong recommendation that CISL/NCAR continue to provide support for the core NCL language, including outreach support for the NCL user community. As a community resource, the NCL team has played a vital role in enabling community science, which is a fundamental NCAR/UCAR mission as a national center.

- 1.1. Maintain the NCL core language in its current state indefinitely, without necessarily adding new functionality.
- 1.2. Maintain access to the existing NCL function library indefinitely.

- 1.3. Continue to support and develop specific libraries (e.g., wrf-python), that are critical to a range of community users and unique to NCL.
- 1.4. Continue to support the current state of online NCL documentation and support the NCL user community through continuation of ncl-talk.

2. **Training and Support for the Pivot to Python:**

The panel strongly recommends developing a training and support system for guiding NCL users through the pivot to Python transition, with an eye toward maintaining the current user base while expanding it to include the Python user community. This recommendation does not imply twice the effort, but rather working toward an expanded user base that will eventually take some of the weight from the NCL developers by contributing to current (e.g., PyNGL and PyNIO) and future (e.g., NCL computational) Python libraries through the open development model. Moreover, an analysis of the NCL user survey shows that there are more new Python users than new NCL users and so this pivot may result in capturing new NCL users and coming out stronger in the end.

- 2.1. The consensus of the survey was that NCL user support is unparalleled in the software open development world and, coupled with the heavy focus on meteorological, climatological, and model-centric algorithms and examples, provides an unparalleled resource for scientific research. The panel feels that it is in the interest of the scientific community to continue this support in the new Python ecosystem.
- 2.2. NCL has a strong presence in the research community, and the panel is aware that this community will be impacted by a pivot to Python. In particular, we believe that it is crucial to proactively address the perception that a ‘code freeze’, together with the pivot to python, may signal a threat to the ability to use NCL. The panel encourages the NCL team to ease community anxiety by emphasizing that the pivot to Python indicates that NCL’s capabilities continue to evolve and move forward, even as resources move away from expanding the core language.
- 2.3. Given the needs of the current NCL user base the panel specifically recommends that the current support (website and ncl-talk) and training be augmented. Specifically, information about the Python libraries should be added to the NCL website and additional training on NCL-to-Python libraries should be offered. These training sessions could be stand-alone workshops or tagged onto NCL-Python workshops already being offered.
- 2.4. Informational sessions about the benefits of the pivot to Python should be held and advertised on the NCL website. It is strongly recommended that they are also

announced through ncl-talk. Pangeo should also be advertised on the NCL website so that new users of Python have a set of recommended libraries to start with. This type of guidance from NCL will greatly help new users since Python, by nature, is a decentralized language.

- 2.5. Survey respondents expressed a large interest in having side-by-side NCL-Python scripts on the NCL website (50% of respondents) and training in all of the current NCL Python libraries (100% of respondents for PyNGL, PyNIO, and wrf-Python). Additionally, this survey reached a diverse audience which were outspoken about how important NCL support (e.g., website, ncl-talk, and tutorials) is for their ability to learn NCL easily and use it quickly for their work. This low-bar for entry is one of NCL's strengths that needs to be continued.

3. **NCL Graphics:**

In recommending that NCL continue to be supported as a core language, the panel points to the unique role that NCL plays for data analysis and scientific visualization, particularly for atmospheric and related sciences.

- 3.1. Given the uniqueness of NCL graphics in the Python ecosystem, the panel strongly recommends continued development and support of NCL graphics within the PyNGL library and a mirroring of these changes in the NCL language as appropriate.
- 3.2. This finding is supported by the quality and utility of NCL graphics, which was noted multiple times throughout the NCL user survey, with 70–79% of respondents indicating that general or journal publication use of NCL graphics was a 'very important' priority for them. Additionally, several panelists and survey respondents noted that in general matplotlib cannot match NCL's graphic quality, speed or range of plot types.

4. **NCL Computational Functions:**

The NCL language was designed for scientific analysis and data visualization specific to the predominant sciences found at NCAR and the surrounding laboratories, e.g., Atmospheric Science, Climatology, Meteorology, and Oceanography. Therefore, the panel strongly recommends continued development and support for the NCL computational functions unique to this community.

- 4.1. Participants in the NCL user survey indicated that adding performance enhancements to the existing functions and porting them to a Python library are both high software development priorities.

- 4.2. Leveraging the Pangeo community should be part of the development of the NCL computations functions Python library, specifically in identifying computational functions that are not already available in existing Python libraries, and thereby saving the NCL developers time and effort. Additionally, this process could facilitate integration of the library into Pangeo's 'Python Data Stack', which would expand the NCL Python library user base.
- 4.3. NOTE: The panel did not have the time to build a specific list of critical NCL functions so it is recommended that NCL Developers generate a list from the results of the 'Porting NCL Computation Routines to Python' section of the NCL user survey.

5. **Open development:**

- 5.1. The panel strongly supports the movement toward an open development model that can augment continued support for the NCL core language through user contributions. The NCL development team presented a roadmap for implementing open development for NCL and the panel supports this route.
 - 5.1.1. Steps along the path to open development include: refactoring the code, and formal governance in the form of a contributor's guide.
 - 5.1.2. The panel recommends that the repository gatekeepers include NCL developers in order to maintain the stability of the NCL code and keep it connected with UCAR/NCAR, which has a fundamental mission to enable community science.
- 5.2. The panel supports the recommendation to move the NCL open development platform to an issue tracking repository such as Github rather than the external/JIRA system that is being used currently.
- 5.3. As noted above, the Pangeo community should be leveraged to identify computational functions unique to NCL, i.e., that are not already available in existing Python libraries. In addition, the Pangeo community may serve as a model for:
 - 5.3.1. formulating a formal governance framework
 - 5.3.2. the process of vetting and integrating code within an open development framework
 - 5.3.3. a list of libraries recommended by the community

6. Input/Output Support Going Forward:

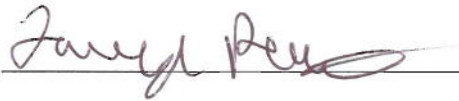
NCL has robust file input and output capability which can already interface with Python through PyNIO. PyNIO is a Python package, developed to interface with the Python environment using NumPy, allowing read and/or write access to a variety of file formats. PyNIO is composed of a C library that contains the same data I/O code used in NCL and it can read and write netCDF-3, netCDF-4 classic, netCDF-4, HDF4, binary, and ASCII data. It can read HDF-EOS2, HDF-EOS5, GRIB1, GRIB2, and OGR files (shapefiles, MapInfo, GMT, and Tiger).

- 6.1. The panel encourages continued PyNIO support for specific NCL capabilities that are not duplicated or well-supported by other Python packages. **In particular, the panel encourages the continued support of uncommon formats such as GRIB and HDF-EOS.**
- 6.2. However, the panel recognizes that existing I/O capability and development outside of NCL does not need to be duplicated by NCL. Going forward, active PyNIO development may best be accomplished as part of an open development community.
- 6.3. Many more common NCL functions are already available in Python libraries. In these cases the Python libraries can be leveraged by the NCL community through the NCL website.

The panel acknowledges that NCL is facing a period of significant change and transition which can be unsettling to users. But the power and prevalence of Python, particularly with young scientists and coders, demands that NCL evolve to exist and thrive within the new Python ecosystem. This evolution will ensure that the graphical and computational abilities provided by NCL, particularly those unique to atmospheric and earth system sciences, are continued into the future irrespective of the programming language. This, along with open development and appropriate governance, will keep NCL relevant and thus available to the scientific community that relies on NCL's unique graphics and computation abilities.

In summary the panel would like to thank you for the opportunity to contribute our input and help inform the future of NCL. If any of the above statements are unclear do not hesitate to reach out to the panel for clarification. Lastly, in light of the paradigm shift manifest in the 'Pivot to Python', we recommend that a second advisory panel meeting be convened in one year.

Tanya Peevey, NOAA / OAR / ESRL / GSD, NCL Advisory Panel chair



8/30/2018

Nan Rosenbloom, NCAR / CGD, NCL Advisory Panel co-chair



8/30/2018

On behalf of the NCL Advisory Panel members:

Dave Allured, NOAA / OAR / ESRL / PSD / CIRES

Sean Birkel, University of Maine

Alan Brammer, ClimaCell

Candida Dewes, NOAA / OAR / ESRL / PSD

Sherrie Fredrick, NOAA / OAR / ESRL / PSD

Soyoung Ha, NCAR / MMM

Laura Landrum, NCAR / CGD

Jared Lee, NCAR / RAL

Karin Meier-Fleischer, DKRZ / Hamburg

Adam Phillips, NCAR / CGD

Clint Rowe, University of Nebraska

Adam Turchioe, Cargill