Community Infrastructure Breakout

Facts - 1

- Code base fragmented within organization and isolated from rest of the weather enterprise
- It is difficult to find policies on how to get external accounts and relatively difficult to get access to NOAA codes (requires a government sponsor).
- Current security guidance on the use of new tools (e.g., GitHub) within NOAA is inconsistent, but seems to be evolving.
- NWS creates too much data to archive it all.
- Underutilization and mis-utilization of version control and other collaborative tools.
- Access to key datasets (e.g., reforecasts and re-analyses) is unnecessarily limited both within NOAA and with external partners.

Facts - 2

- NOAA’s code, data, and product documentation is inconsistent and widely scattered.
- NOAA doesn’t have policies for documenting code, data, and products.
- NOAA doesn’t have consistent, readily-discoverable policies for software quality.
- The GSA organization 18F has guidelines for openness and collaboration

Findings

- Current code development paradigm encourages stovepipes.
- The burden of approvals discourages right-sizing for collaborative projects.
- Most of the NOAA-specific facts enumerated are found in other met services
- A lack of common vocabulary impedes broad-based collaboration
- There are common security concerns about Git and other novel technologies that arise often.
- There are a number of licensing concerns for publicly sharing of previously internal software.
- Inconsistent and numerous data formats are an obstacle to collaboration.
- Lack of machine-readable metadata and inconsistent metadata are an obstacle to collaboration.

Recommendations - 1

- NOAA should establish modern, widely-utilized distributed version control systems (e.g., Git and GitHub) to facilitate R2O and O2R, including to/from external partners.
- There should be a FAQ regarding version control interfaces (e.g., GitLab/GitHub) addressing common concerns amongst centers and security and how to mitigate risks.
- NOAA should establish a set of basic software guidelines.
- NOAA managers should start encouraging more training for best software practices and learning modern programming languages (e.g. Python) (or putting it into work plans)
- NOAA should allocate resources for code management to assist collaborators and process pull requests to incorporate external code contributions into the main repository.

Recommendations - 2

- NOAA should work with external collaborators to develop a process flow to communicate and collaborate between government, private, academia.
- NOAA should establish requirements for metadata and documentation
- NOAA should establish governance plans to enable groups to work together and methods of making decisions.
- NOAA should create an area where data can be accessed and processed by external collaborators easily, including international collaborators.
- NOAA should establish a centralized location for documentation and data access (e.g., data.gov).
- NOAA should leverage work already done by 18F as well as work with them on creating guidelines and standards.

Recommendations - 3

- NOAA should create requirements and guidelines for meta-data for forecasts, reforecasts, and post-processed data.
- NOAA should work with the community to determine 2 or 3 common data formats that should be used (e.g., GRIB, NetCDF, HDF, geoJSON) that will satisfy operational, research, collaboration, and archival purposes.