

Roadmap 2019

In 2019 ASC has several different software targets. These are primarily focused on our core software packages: ISIS, the Community Sensor Model toolchain, plio, PyHAT, and AutoCNet. Additionally, we have tools and processes which rely on these core tools such as the Universal Planetary Coordinates database, Map-A-Planet 2 and Map Projection on the Web, which all utilize ISIS.

We have labeled some tasks as investigations to identify those places where we need additional information before we can estimate the full complexity and scope of a task. These tasks can lead to future work, a decision to defer, or a decision that additional information is required.

Tasks on this roadmap are living. They may move ahead to future years, be identified as being completed, or be identified as not needed. Our ultimate goal is to engage the user community to identify where we can add value.

Tasks status is identified using the following shorthand:

Mark	Meaning
bullet	work not yet started
check	work completed
	on-going work
	stretch goal

Themes

Funded development for ISIS is generally aligned by named projects. We organize the roadmap thematically due to the amount of synergy between projects. In FY19 (October 1, 2018 - September 30, 2019), we are working in the following thematic areas:

- Streamline the build, test, deploy trains
- Begin improving the ISIS testing infrastructure
- Reduce friction for external development on projects

- (Re-)Build the online community around our software packages, all types of contributors.
- Operationalize ASC Community Sensor Model
- Begin re-architecture of ISIS

Release Train 🚂

- ✓ Transition to CMake and anaconda deploys (binaries)
 - 🏃 Transition data area from rsync to version controlled solution
- ✓ Adopt versioning standard. This allows us to alert users to enhancements and breaking changes.
- Separate source/deploy a la conda-forge to reduce the release friction.
- 🏃 Move to 4 releases per year and immediate bug fix releases to get new code out to users at the fastest reasonable cadence.
- 💪 Unpin version on 1+ current dependencies to make installation of ISIS alongside other libraries easier.

Testing

- ✓ Investigate methodology to write unit tests in a testing framework. The goal is to reduce test data volumes and the number of false positive failures.
 - Investigate how to write app tests. The rationale as above.
- ✓ investigate and deploy a testing framework; GTest.
- ✓ Investigate and deploy a CI environment to remove handcrafted cron scripts and remove email chain of build successes.
 - 💪 Transition all unit tests to GTest
 - 💪 Migrate 1+ app tests using the identified framework.

Sensor Models (CSM/ISIS)

- ✓ Finalize Image Support Data Specification
- ✓ Implement CTX sensor model
- ✓ Verify MDIS-NAC/-WAC sensor models
- ✓ Implement Kaguya TC sensor model (CSM/ISIS)
- ✓ Implement Dawn FC sensor model
- ✓ Verify GXP functionality
- ✓ Implement CSM HiRISE sensor model

- 🦷 CSM/ISIS integration

SPICE / Architecture / ALE

- 🏃 Begin ISIS rearchitecture by refactoring SpicePosition and SpiceRotation classes to use ALE
- 🏃 Refactor SPICE management scripts to reduce maintaince costs and rebuild institutional knowledge
- 🏃 Port all spiceinit calls for ISIS sensors to ALE
- 🦷 Refactor all SpicePosition/Rotation calls to use ALE
- ☑ Refactor CSM to utilize common ALE interface
- 🦷 Transition all ISIS spice selection code to ALE.

Documentation

- 🏃 Continue to improve our process, developer, and user documentation, migrating to openly available solutions (e.g., GitHub wikis, discourse forums, etc. as appropriate).
- 🏃 Generate project level roadmaps for all large scope projects that are publicly facing.
- 🦷 Ensure the above are publicized at a monthly cadence

Community Support

- ☑ Get the community support forums turned back on
- ☑ Fully transition from RedMine (performance and usability issues) to GitHub
 - 🏃 Provide support via GitHub, discussion forums, and other communication platforms
- ☑ Solicit, review, and merge community PR

Other ASC Libraries

- PyHAT
 - ☑ Add support for M3 derived products
 - ☑ Add support for CRISM derived products
 - ☑ Build ipynb documentation :muscles: QGIS plugin deploy on 3 major OSes
- AutoCNet
 - ☑ Add capability to tie to ground
 - ☑ Refactor to deprecate server

- ☑️ Improve ipynb documentation for use cases
 - 💪 Identify how to stream new images into a project
- Plio
 - ☑️ Solidify deploy train via conda-forge

Dependent Software

- Universal Planetary Coordinates (UPC) database
 - 🏃 Redesigning the database schema to improve search times and enable level 2 products to be included in the database
- ☑️ Map-A-Planet 2 - uses ISIS to create map projected products from derived products on demand for users from a web interface
- ☑️ Map Projection on the Web (POW) - uses ISIS to create map projected products from level 1 PDS products on demand from a web interface

Summary

Above a high-level overview of the tasks targeted for ASC FY19 development have been enumerated. We are focusing heavily on addressing technical debt, engaging our user base, and making external development more viable. We anticipate that these tasks will continue into FY20 as we transition to a true open source project. We solicit any engagement from those users that we seek to serve.