InSAR Scientific Computing Environment

Eric Gurrola, Gian Franco Sacco, Piyush Agram, Marco Lavalle,
Paul A. Rosen, Mark Powell, Darren Dao
Jet Propulsion Laboratory
California Institute of Technology

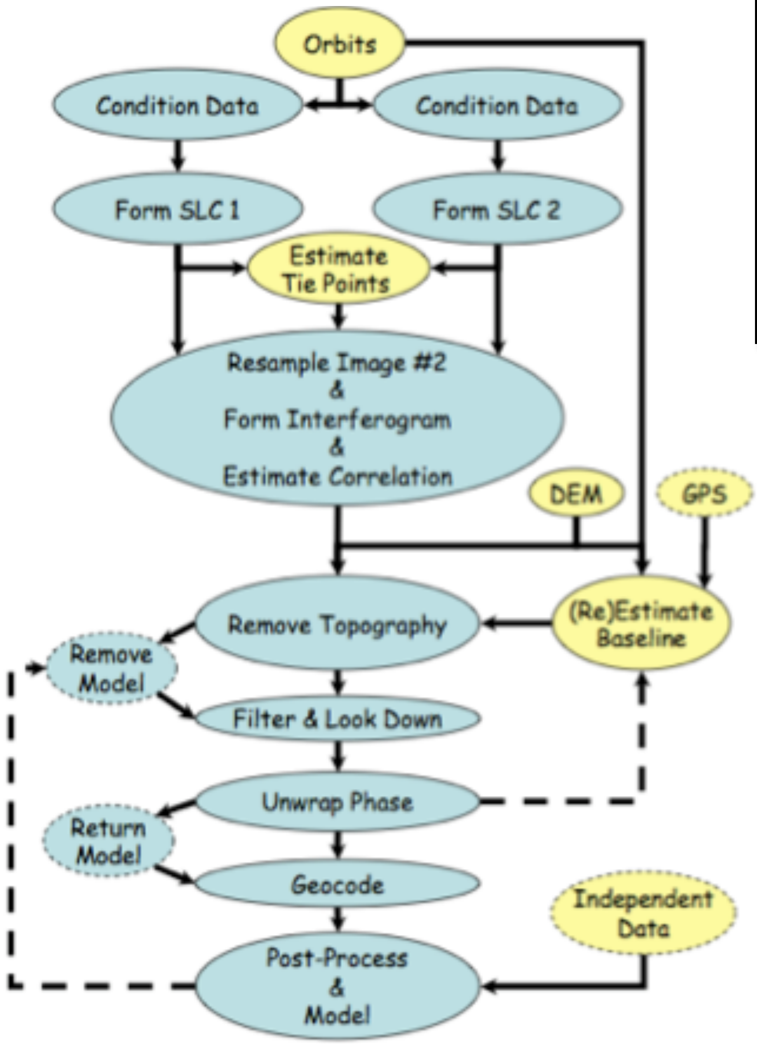
Howard A. Zebker
Stanford University

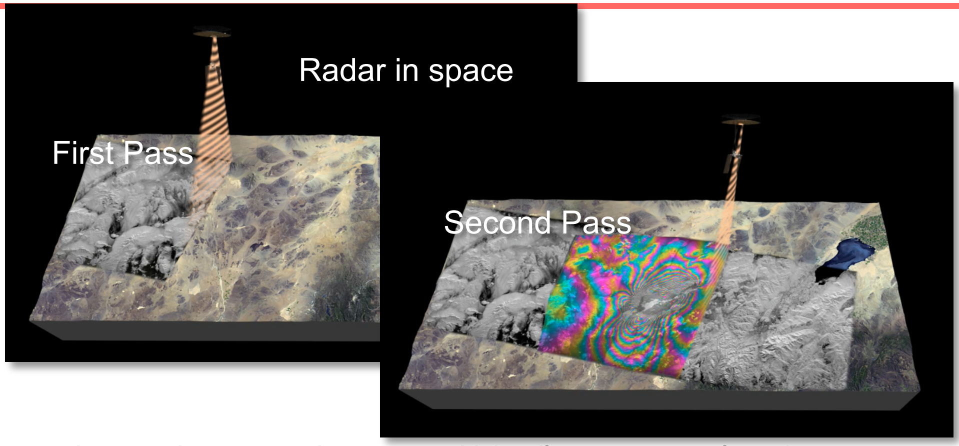
ESTF 2014
Leesburg, Virginia
October 30, 2014

Outline

- Introduction
- Technology Description
- Functionality
- Part of the ARIA-MH Production System
- Core Technology in Earthkit and CESCRE cloud-based computing
- Baseline for proposed NISAR Algorithm Core
- Recently upgraded to support UAVSAR Interferometry and Polarimetry for Time-Series stacks
- Vision for ISCE architecture
  - What international missions it supports and how
  - How does it support the user community

Interferometric Synthetic Aperture Radar (InSAR) From Recipes to Reconfigurable Flow






```
/Users/parosen/Projects/CSK/hawaii> python3
>>> import isce
>>> from applications.insarApp import Insar
>>> a = Insar()
>>> a.configure()

2014-10-18 17:47:51,008 - isce.insar - INFO –
ISCE VERSION = 2.0.0_201409, RELEASE_SVN_REVISION =
1612, RELEASE_DATE = 20140918,
CURRENT_SVN_REVISION = 1647

>>> a.rangeLooks = 4
>>> a.run()
```

Insar Application: Implements InSAR processing flow for a pair of scenes from sensor raw data to geocoded, flattened interferograms.

Fixed Prescriptive Flow → **Flexible "Sandbox" ISCE Flow**

ESTO Earth Science Technology Office

ISCE Algorithms

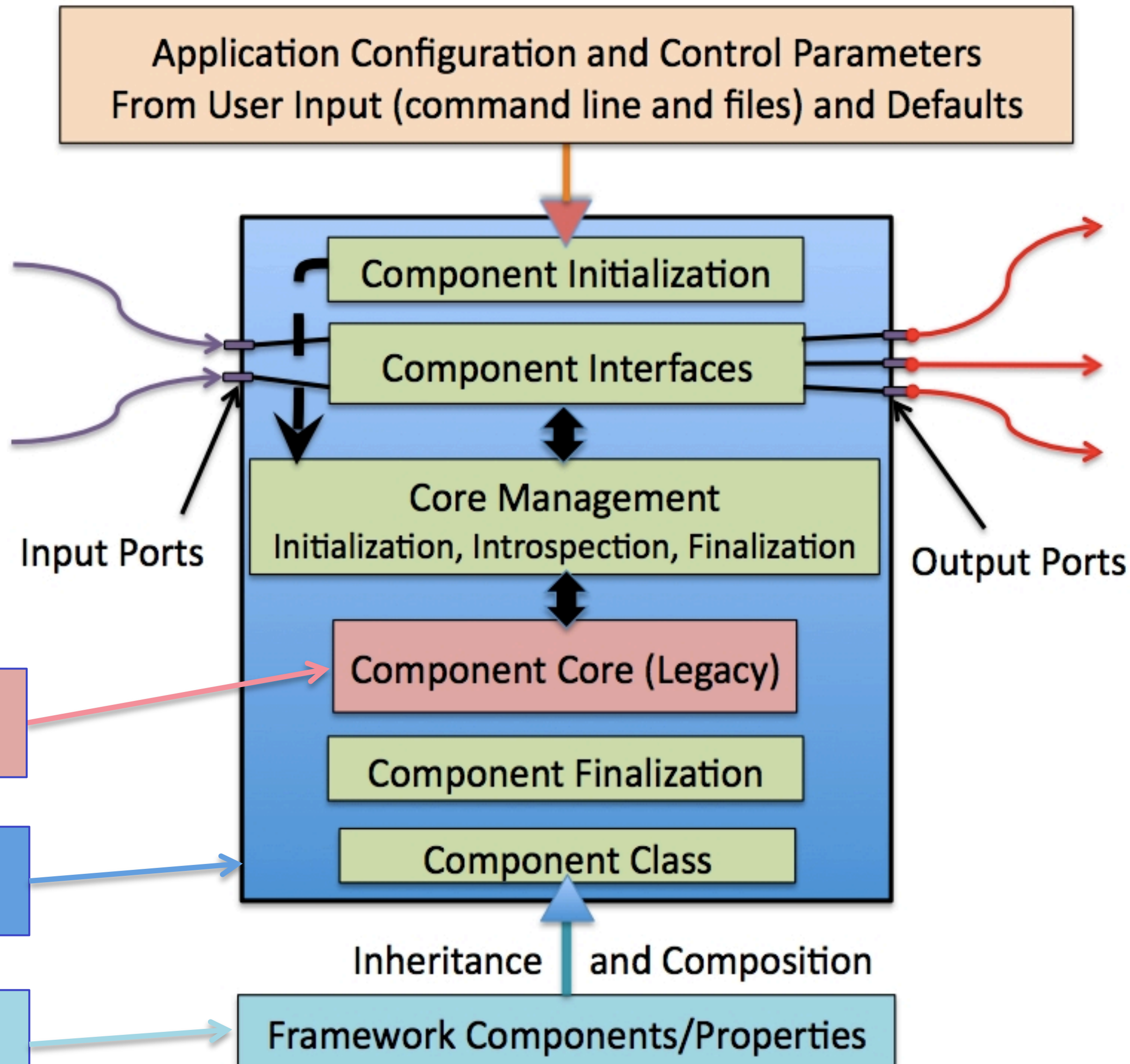
- Created accurate and fast algorithms based on GEOSAR and SRTM heritage

- Advantages of improved accuracy and speed:
  - Enable InSAR and time series methods with precise SLC alignment requirements
  - Straightforward merging SAR data with other types
  - Feasible processing of dozens of scenes by a desktop user
- Approach
  - Define a rigorous geometric framework tied to local spherical coordinate system
  - Adjust collection of images to a common reference trajectory
  - Condition data for ingestion in post-processing time-series applications such as GIAnT

The Architectural Core of ISCE

By adopting a general object oriented framework for processing, ISCE

- Enables extensibility and flexibility by abstracting functions that are typically embedded in scientific codes
- Allows straightforward inclusion of scientific methods developed over many years without redevelopment of algorithms



Legacy code in Fortran, C, or other domain expert language

Object oriented processing components in Python 2.X

Framework components in Python 2.X and C++

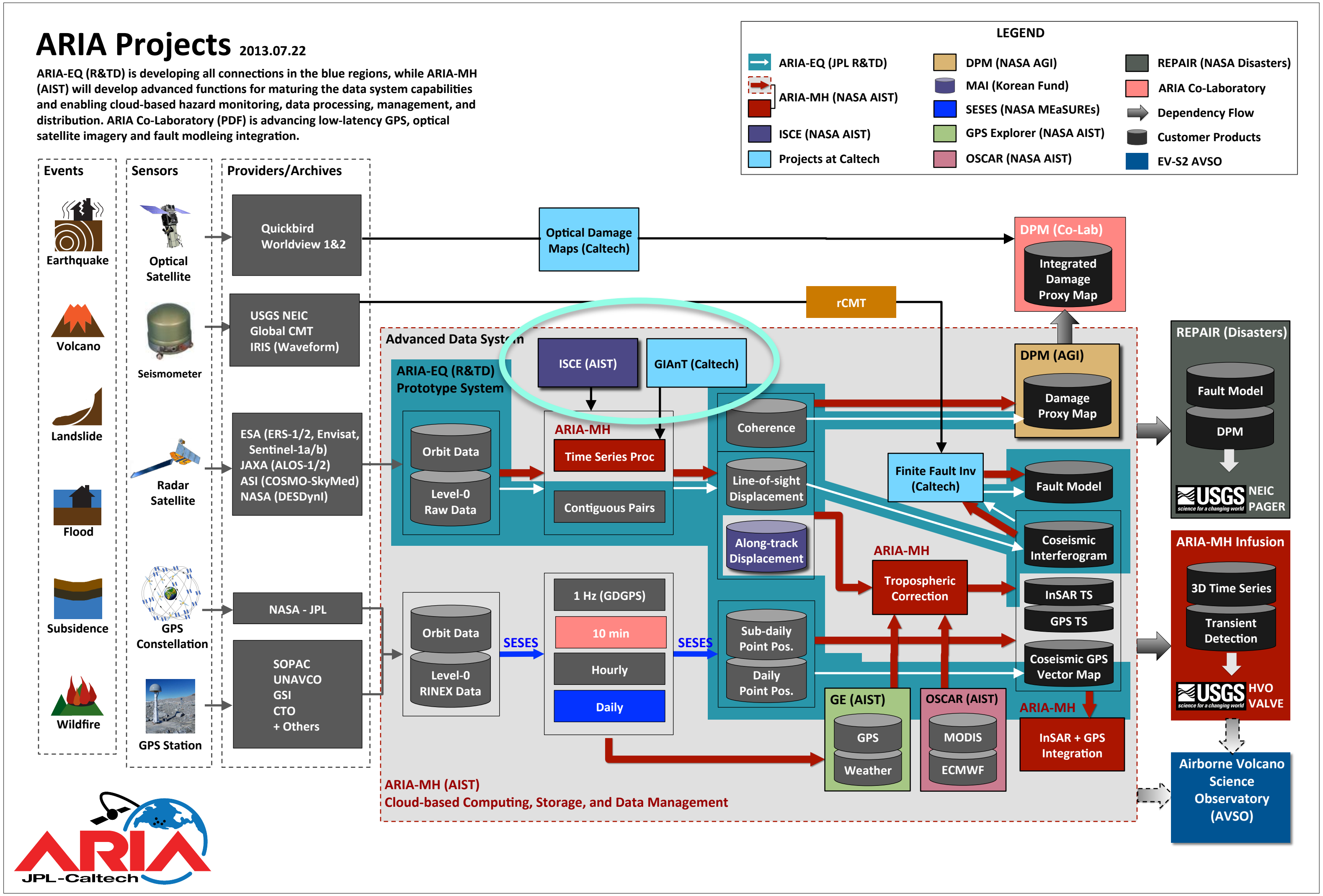
ISCE Features

- Object orientation using Python
- A common hierarchical structure for defining components and applications
- An Image API & I/O API to abstract access to data and metadata
- Runtime polymorphism through factory pattern creation of components
- Provenance through xml metadata and python *pickle* mechanisms

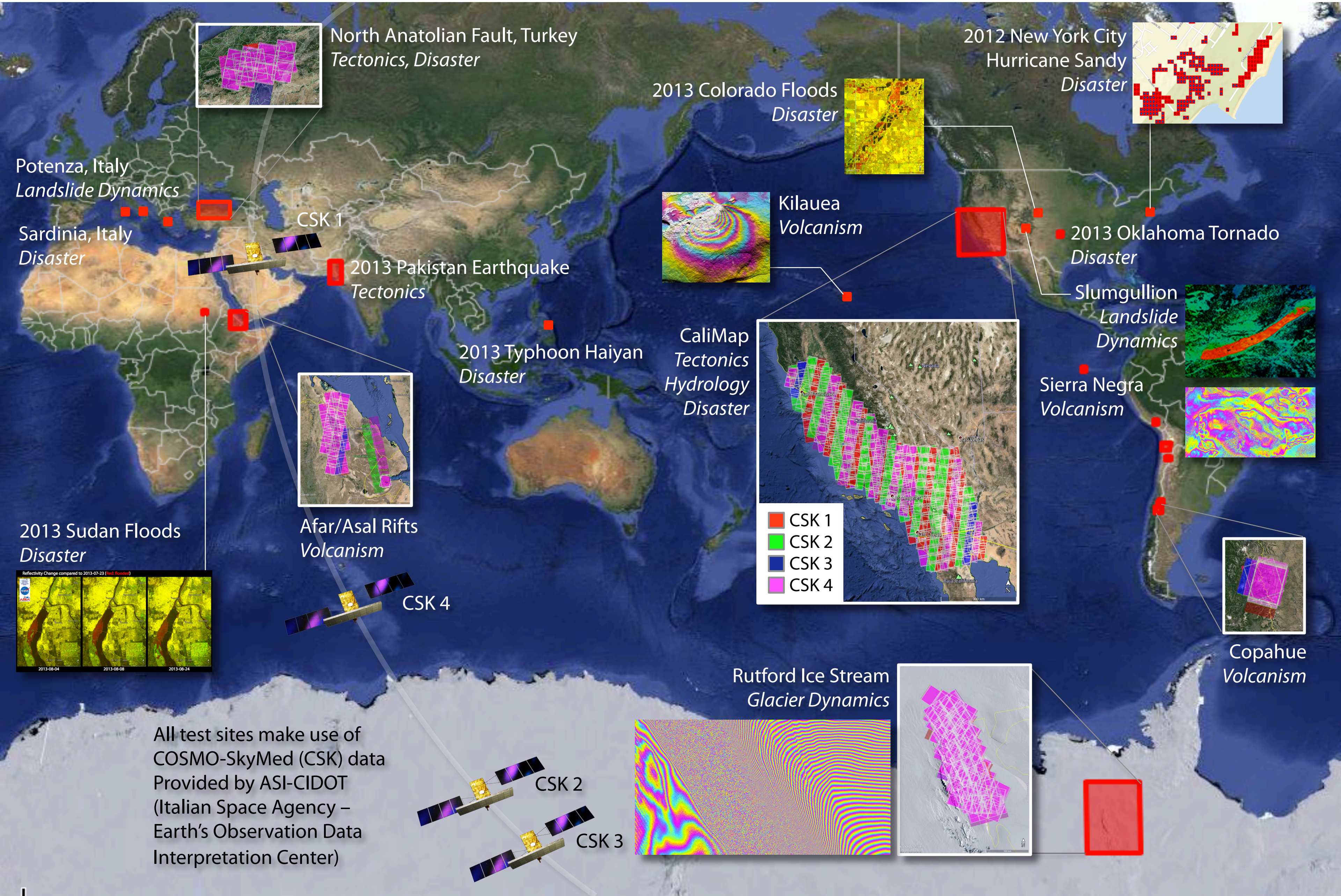
Code Status and Availability

- Code supports strip mode data from
  - ERS-1, ERS-2, ENVISAT, RADARSAT-1, RADARSAT-2, Sentinel-1
  - JERS, ALOS-1, ALOS-2
  - TerraSAR-X, Cosmo-Skymed (CSK)
  - UAVSAR
- Unix/Linux build environment requires
  - gcc 4.7+, Python 3.3+, numpy, fftw 3.2, scons 2.0.1, motif
  - spiceypy (RadarSAT1 only)
  - gdal (RadarSAT2 only)
  - hdf5 (CSK only)
- UNAVCO is licensed to distribute the ISCE software to members of the WInSAR consortium
  - WInSAR members comprise a significant portion of the ISCE user base
  - WInSAR maintains a large database of InSAR raw data, as well as processing tools
- Individual research licenses can be provided by Caltech

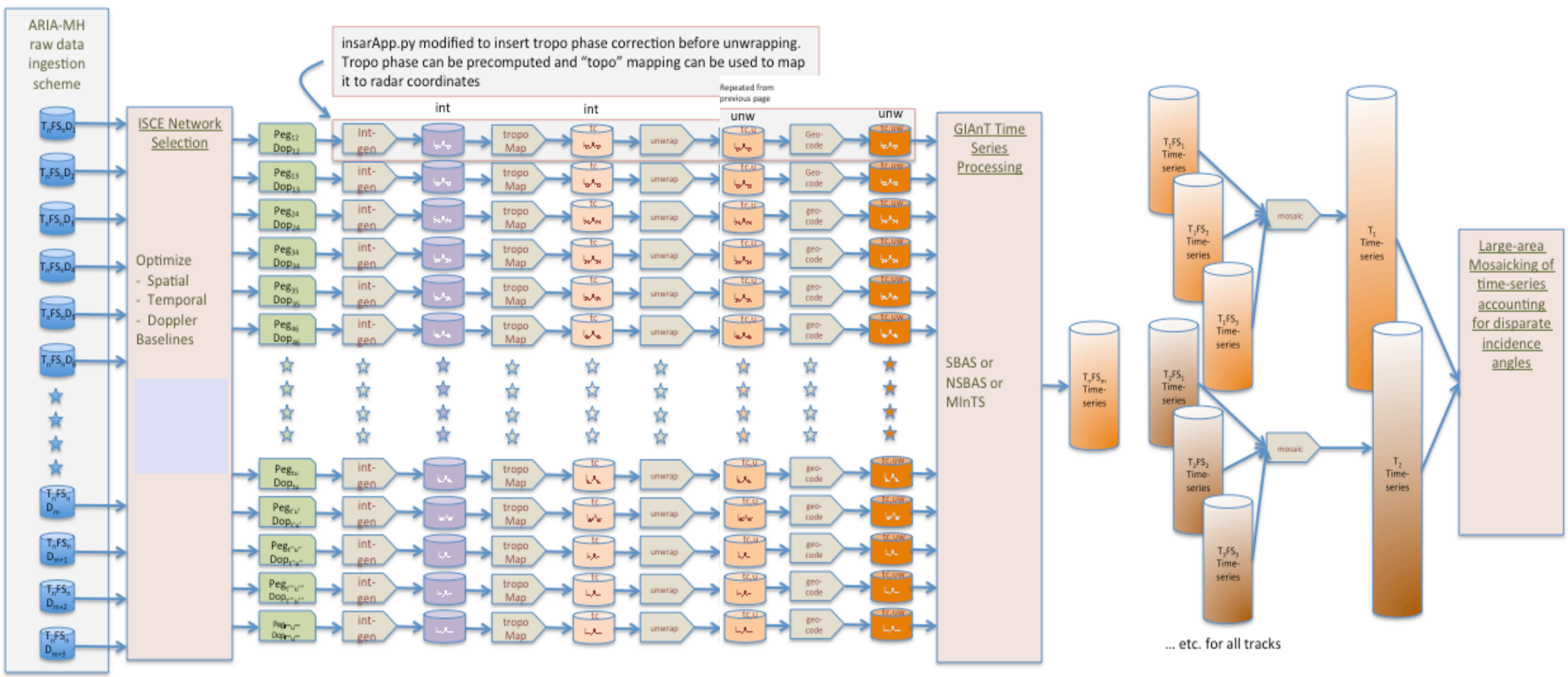
ISCE as the core of ARIA
Advanced Rapid Imaging Analysis system



JPL/Caltech – ASI/CIDOT Collaboration



Automated InSAR Processing Architecture for Large Scale Analysis



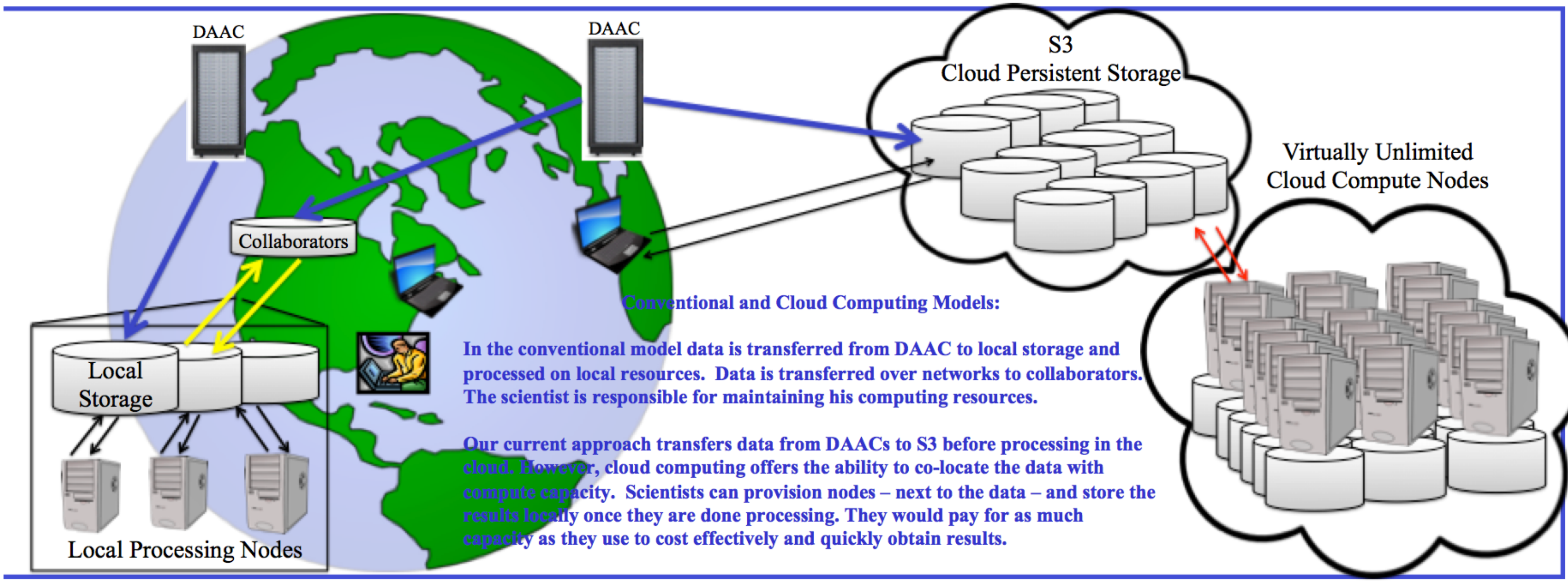
ARIA data system uses

- InSAR Scientific Computing Environment (ISCE) software
- GIAnT software for time series analysis

To generate

- Interferograms
- Tropospheric corrected interferograms
- Velocity maps

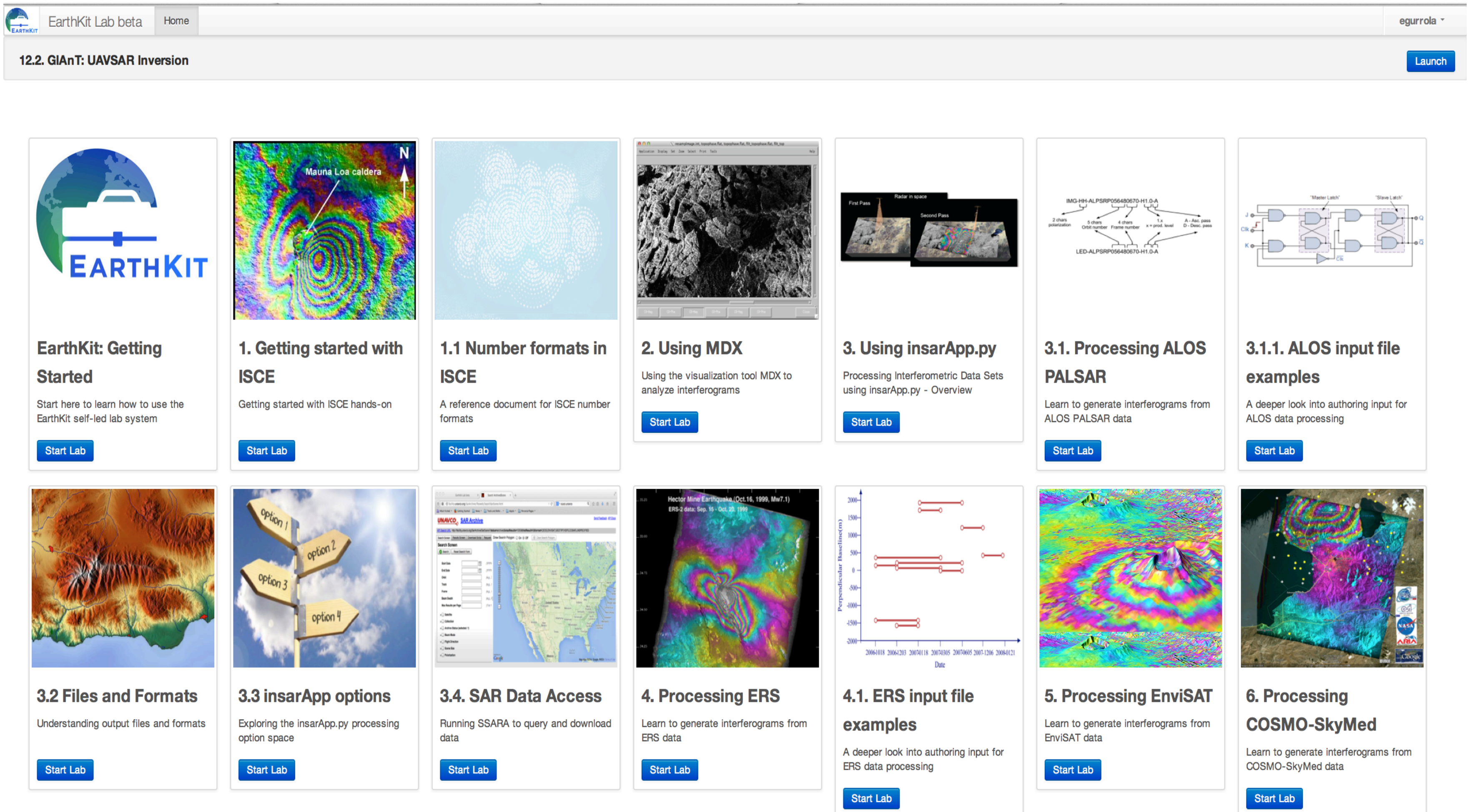
CESCRE – Cloud Enabled Scientific Collaborative Research Environment (Mark Powell, PI)



- CESCRE developed a framework for processing data and sharing results on the Amazon cloud
- ISCE was the fundamental test environment for CESCRE collaboration

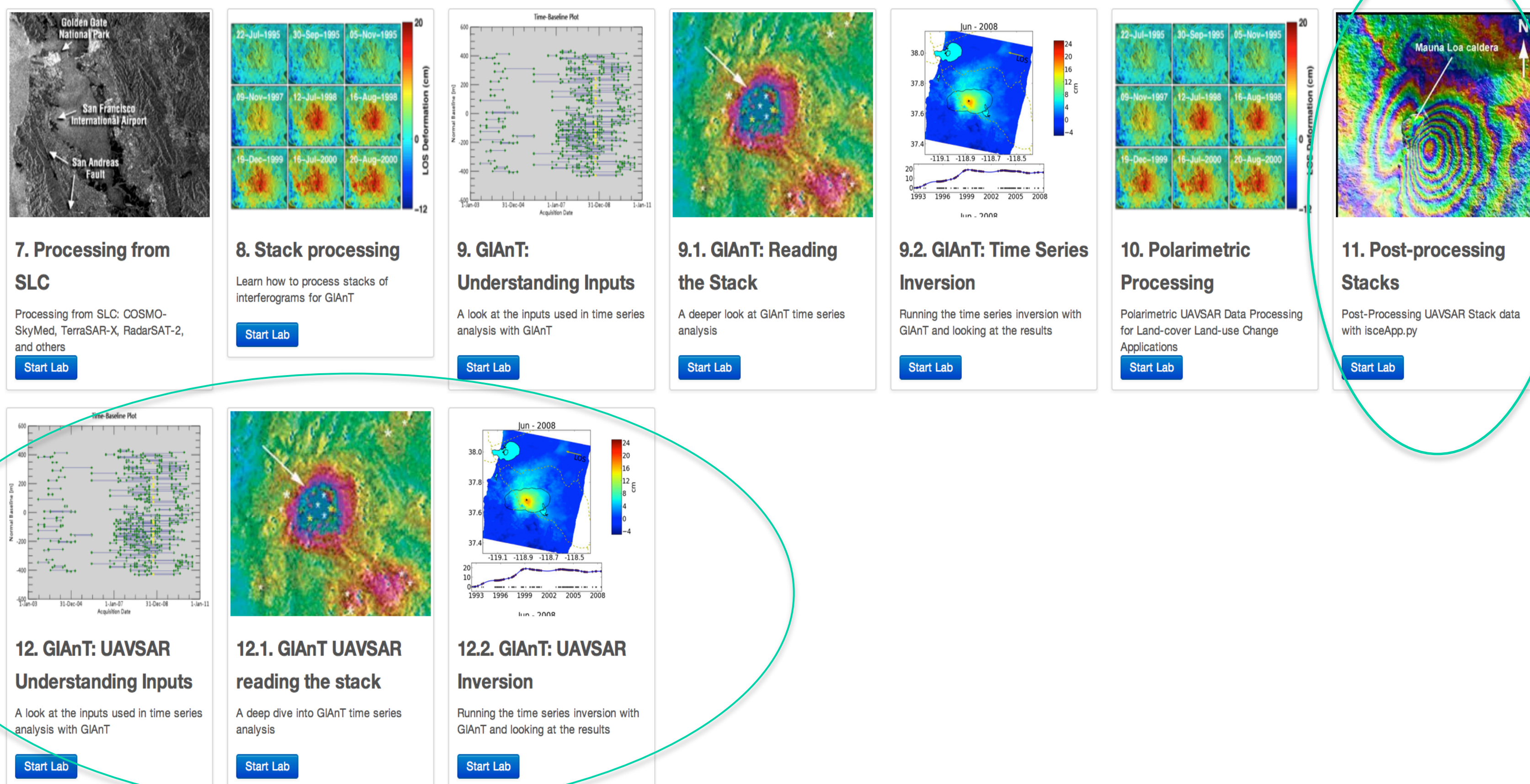
- CESCRE allows a user to sit at their laptop and harness the full power of the amazon cloud

Earthkit Tutorials for ISCE/GIAnT
Mark Powell, PI



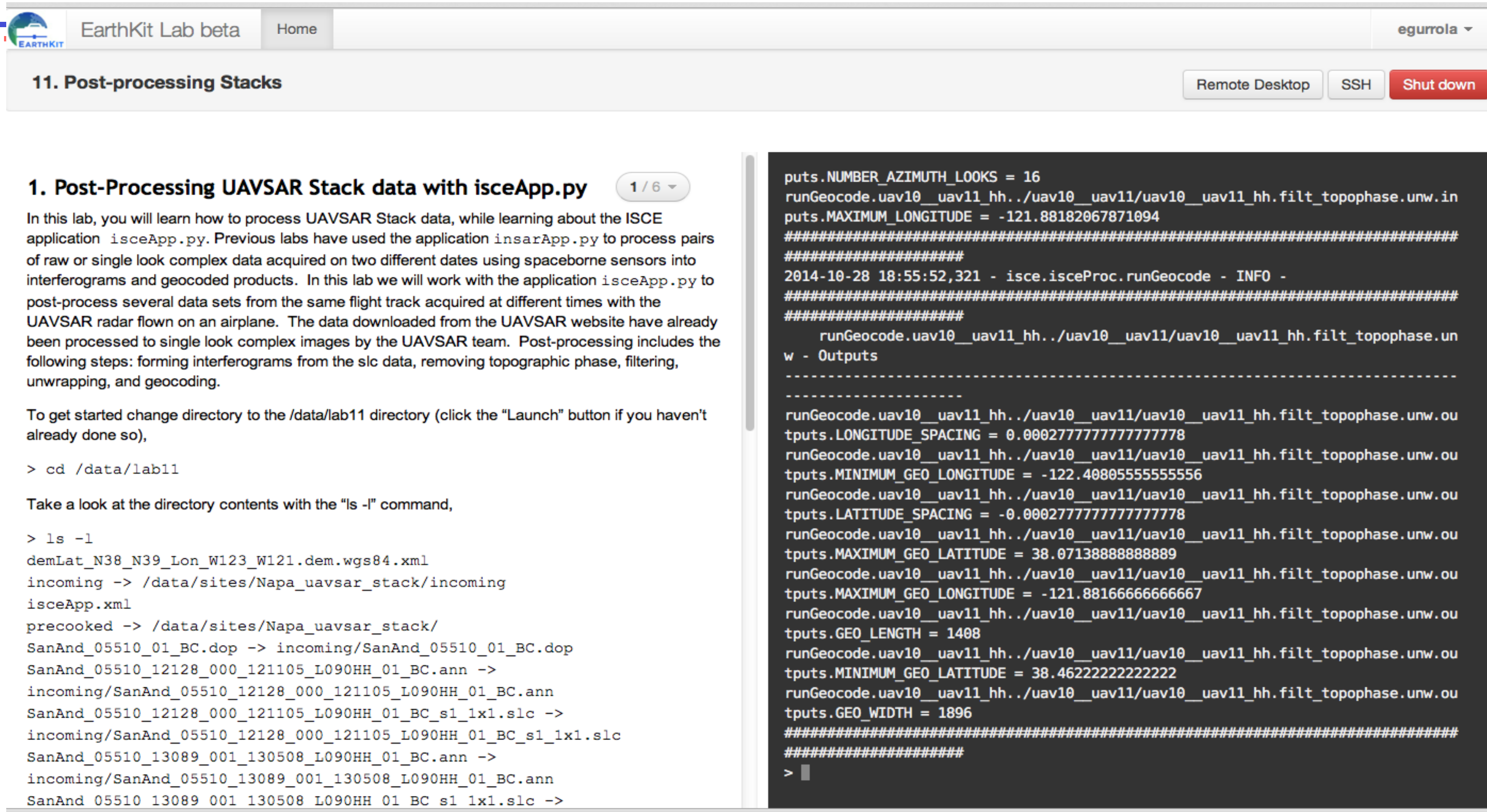
- Extensive, cloud-based, interactive tutorials for learning about ISCE and GIAnT

Earthkit Tutorials for ISCE/GIAnT



- Including new modules for UAVSAR stack processing

Earthkit Tutorials for ISCE/GIAnT



- Guided training through browser interface
  - Integrated step-by-step instructions and a Unix command line prompt to play with code and data.
  - No installation of code and software dependencies

ESTO Earth Science Technology Office

UAVSAR Enhancements for ISCE

ESTO supported enhancements in ISCE to enable scientists to work with UAVSAR products:

- post-process SLC images (provided by the UAVSAR project) into stacks of interferograms with options to filter, unwrap and geocode.
- enable scientists to use the ISCE framework and components to create work flows needed for their work with the data.

ISCE for the NASA-ISRO SAR (NISAR) Mission

- The proposed NISAR mission plans to use the ARIA infrastructure for elastic computing of NISAR data to L0B (raw), L1 (Images), and L2 (interferometric and polarimetric) data levels
- ISCE will be augmented by the project to incorporate new functionality that would be needed for NISAR
  - Image focusing processor for the proposed NISAR mission
  - Calibration/Validation tools
  - Science algorithms for biomass, deformation, time-series analysis, and mosaicking
  - Integrated tutorials and training modules
- This functionality will be available to individual scientists on their local computing devices or through cloud services

Vision for Future Development of ISCE

- The ISCE framework has been designed with flexibility and extensibility that can work in a variety of contexts.
- ISCE can support the community at many levels
  - Individual users working on their laptop or local computing system
  - Users interacting with the cloud through VMs and staging methods for data and processor deployments
  - On-demand processing through web-based queries to a central server
  - Project-sponsored code core for production of multi-petabyte data sets
- Coordinated development across these levels will ensure a robust, community vetted science tool for NASA science – from ROSES to missions like the proposed NISAR