



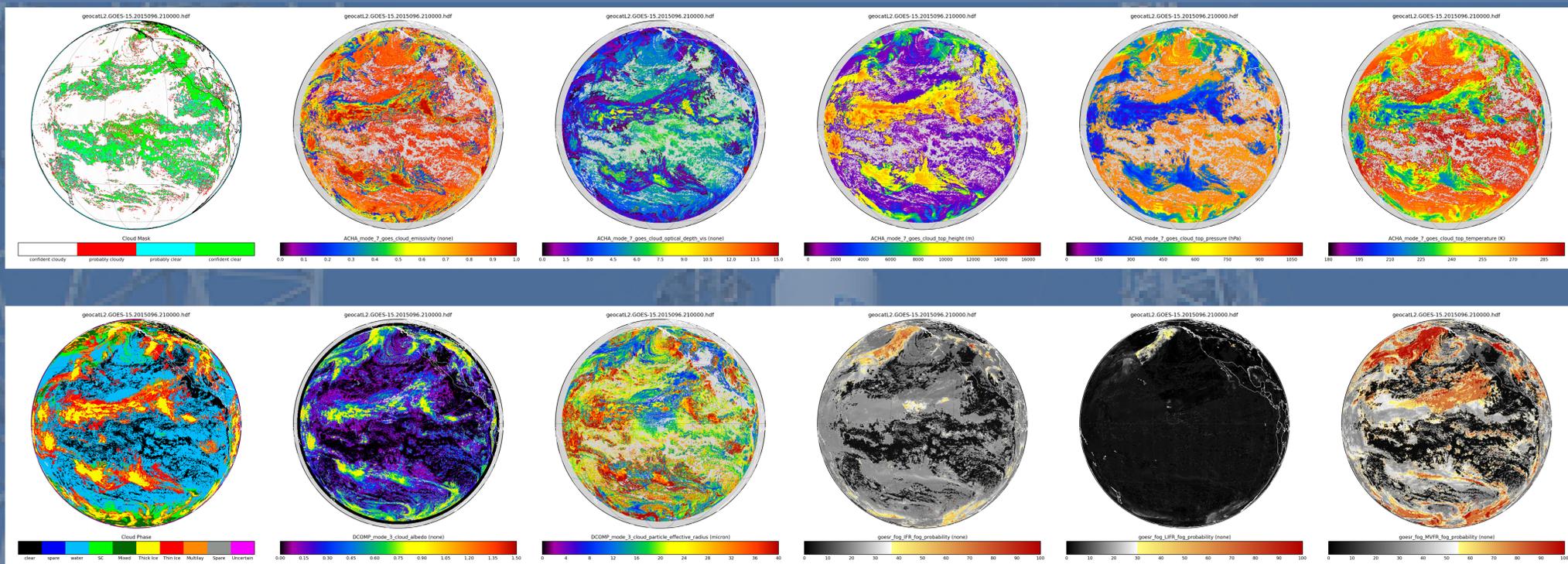
Level-2 Products in the CSPP-GEO Direct Broadcast Package

Geoff Cureton, Scott Mindock, Graeme Martin, Liam Gumley

Cooperative Institute for Meteorological Satellite Studies
Space Science and Engineering Center, University of Wisconsin - Madison
1225 W. Dayton St., Madison, WI 53706, USA



GOES-15 Level-2 Products for 2015-04-06 @ 2100h UTC



CSPP-GEO

- The Community Satellite Processing Package-Geostationary (CSPP-GEO) generates products from geostationary satellite data.
- Allows Direct Broadcast users to process current-generation GOES data received on their own antennas.
- Includes cloud and fog detection algorithms developed under the GOES-R Algorithm Working Group.

Features

- Processes current GOES and MSG imager data.
- Will run on area files generated by the CSPP-GVAR package, or on area files obtained from CLASS.
- Automatic retrieval and transcoding of GFS ancillary data.
- Able to run different implementations of an algorithm for development and testing.

What's to Come?

- GOES-R and HIMAWARI support..
- Introduction of JSON configuration files for CSPP-GEO.
- Quicklooks (see above).
- Upcoming beta release of CSPP-GEO!

Direct Broadcast

- The CSPP-GEO package will allow users with an antenna to access data directly through the GRB stream, rather than over the internet, and process through to level-1 and level-2 data products.
- Direct Broadcast may be attractive to users with limited internet access or restricted bandwidth. This will soon become more relevant as the GOES-R and HIMAWARI imagers have significantly more channels than current GOES imagers.
- The CSPP-GEO package is likely to contain newer versions of the operational product algorithms that include recent science improvements.

Current GOES Imager Support

- Software to ingest GVAR data from the current generation of GOES imagers and generate area files has been released.
- An upcoming release will generate level-1 products, and level-2 products using the research versions of the level-2 cloud and fog detection algorithms that were developed for GOES-R.
- Direct Broadcast users will be able to run the GOES-R research algorithms on data received at their own antennas, giving them an early look at GOES-R products (albeit generated with current GOES channels and lower spatial and temporal resolution).

GEOCAT

- The Geostationary Cloud Algorithm Testbed (Geocat) was developed by the GOES-R Algorithm Working Group (AWG) Cloud Application Team to serve as a cloud retrieval algorithm development testbed.
- Geocat provides a convenient interface to measured radiances, ancillary data (NWP profiles, surface emissivity, surface type, snow, etc.), fast model generated clear sky radiance profiles, and measured and/or generated data from previous image time steps.
- Geocat provides the choice of using either CRTM or PFAAST radiative transfer models (CRTM will be the default).
- Geocat also provides a common algorithm output structure, whose definition is transparent to the algorithm developer. Geocat is currently capable of processing GOES (current generation), MSG, MTSAT, or simulated GOES-R ABI data.
- Geocat currently outputs level-1 level-2 and RTM data in HDF4 format.

GOES-R ABI Imager Support

- We plan to release a version of the level-2 software that can process ABI data. The initial version will produce cloud and fog detection products. Subsequent versions will include additional products, and algorithm updates as they become available.

HIMAWARI AHI Imager Support

- We plan to provide a version of the level-2 software that can process AHI data from the Japanese Himawari mission. AHI offers an excellent proxy for ABI data, having similar bands, resolution and data volume. Existing ABI product algorithms are expected to run on AHI data with little or no modification.
- By running the GOES-R product algorithms on data from an ABI-like instrument, we hope to identify potential issues with software, hardware and data products before ABI is launched, and to allow users to generate potentially useful products from AHI data.

