GeoIPS Breakout Session: *Sectors*

2025 Fall Workshop

Andrew Thorpe

Overview

- Geographic sectors in GeoIPS
- Capabilities of sectors in GeoIPS:
  - Static Sectors
  - Dynamic Sectors
- Sectors and Sectoring backend
- Sector Examples:
  - Static: Cape Horn
  - Dynamic: Tonga Volcanic activity

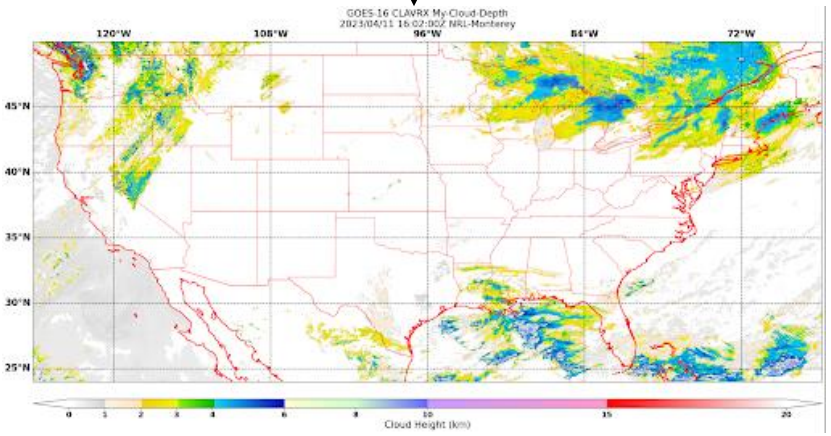
GeoIPS sectors

- YAML defined sectors
- Set style helps interchange projection type, size, resolution, and central point
- Wide coverage between static and dynamic sectors
- Interchangeable; same sector for different products, readers, and data sources
- 55 static sectors and 23 dynamic sectors
- Defined structure and format: see Extend GeoIPS with a Static Sector

Static Sector Capabilities

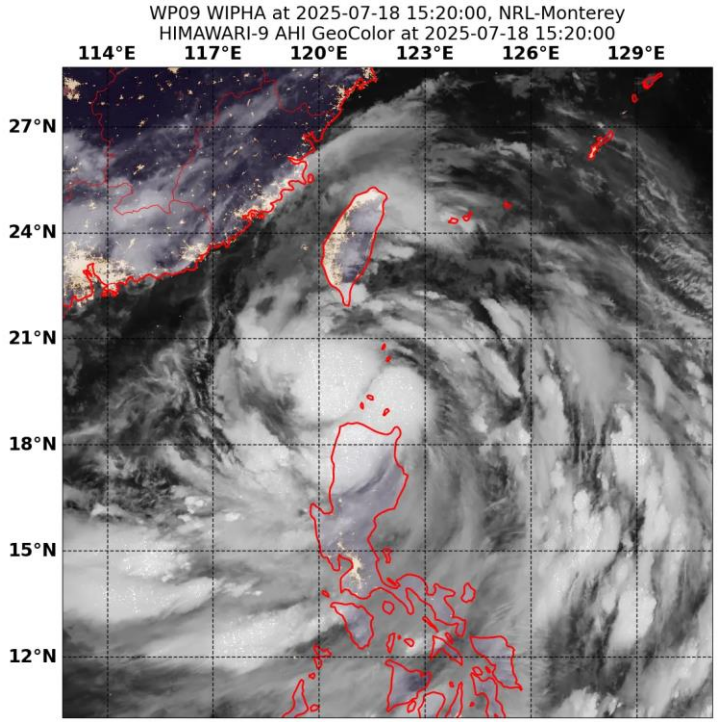
- Tunable parameters:
  - Center latitude & longitude
  - Projection type (EQC)
  - Units (meters), used in resolution
  - Resolution of each pixel (in user defined units; km/m)
  - Shape (height/width in pixels)
  - Center (almost always 0,0)
- Tunable metadata:
  - Region description; used in output metadata and formatting
  - Continent (EX: SouthAmerica)
  - Country (EX: Chile)
  - Area (EX: AtacamaDesert)
  - Subarea (EX: Huasco)
  - State (EX: Atacama)
  - City (EX: Vallenar)
- Downsides:
  - Custom projections are hard to implement

```yaml
interface: sectors
family: area_definition_static
name: my_conus_sector
docstring: "My CONUS Sector"
metadata:
  region:
    continent: NorthAmerica
    country: UnitedStates
    area: x
    subarea: x
    state: x
    city: x
spec:
  area_id: my_conus_sector
  description: CONUS
  projection:
    a: 6371228.0  # The average radius of the Earth in Meters
    lat_0: 37.0  # The center latitude point
    lon_0: -96.0  # The center longitude point
    proj: eqc # Describes the Projection Type (from PROJ Projections)
    units: m
  resolution:
    - 3000 # The resolution of each pixel in meters (x, y)
    - 3000
  shape:
    height: 1000  # The height of your sector in pixels
    width: 2200  # The width of your sector in pixels
  center: [0, 0]  # The center x/y point of your sector. Almost always [0, 0]
```



Dynamic Sector Capabilities

- Capture transient activity, can be dynamic in space and/or time
- Work in conjunction with track files and parsers
- Tunable parameters:
  - Pixel width (meters)
  - Pixel height (meters)
  - Number lines (pixels)
  - Number samples (pixels)
- Tunable metadata:
  - Docstring
- Downsides:
  - Not well documented



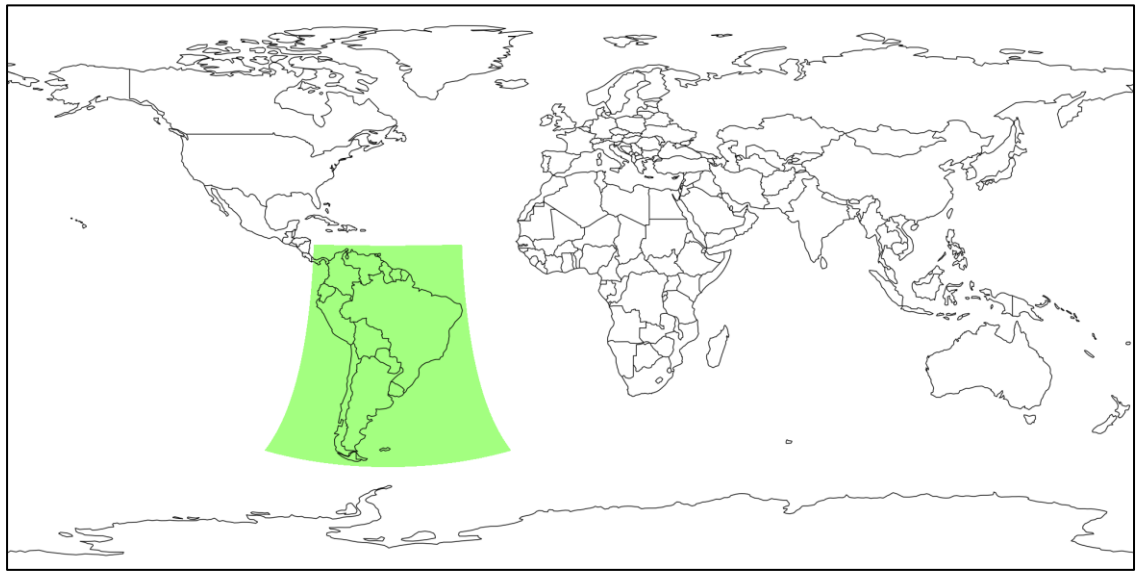
GeoIPS utilizes dynamic sectors for Tropical Cyclones
(imagery via TCWeb)

Sectoring backend

- Uses a backend of PROJ and pyresample to generate sectors
- Sectors are transformed to Area Definition object to help interpolate, resample, and perform calculations
- Sectoring and interpolation steps within GeoIPS utilize the user-defined sector to then manipulate input data
- Projections are tied to PROJ projections: PROJ Projection Documentation
- Units are tied to: PROJ Units Documentation

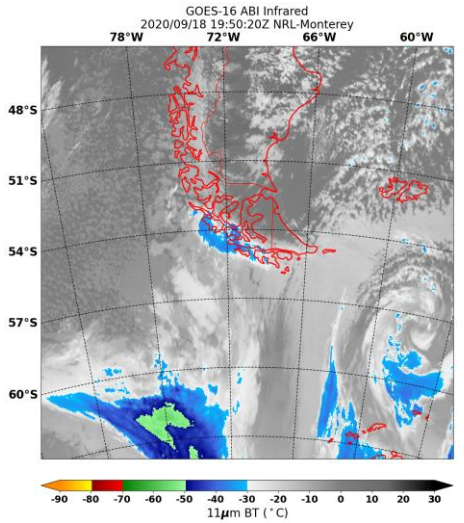
Tools to visualize sectors

- Tools to visualize sectors:
  - geoips test sector <sector_name>
    - Displays sector in defined projection
  - geoips test sector <sector_name> --overlay:
    - Overlays sector onto global equidistant cylindrical projection



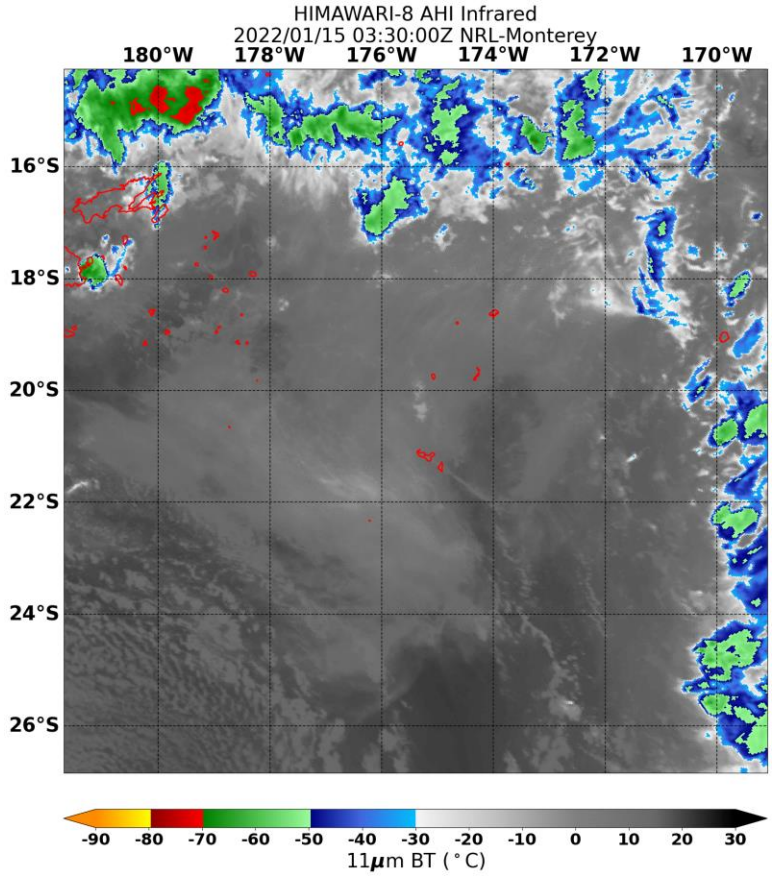
Static example

- Area of interest: Cape Horn
- Steps:
  - Get needed geographic information
    - Center Latitude: 55S
    - Center Longitude: -70 (70W)
    - Projection: Stere
    - Units: KM
    - Resolution: 2km, 2km
    - Shape: 1000,1000
  - Create sector YAML file
  - Test sector: Visualize using GeoIPS testing tools
  - Implement sector: Display GOES-16 Data



Dynamic example

- Tonga Volcano explosion
- Steps:
  - Get needed geographic information
    - Center: 20° 33′ 0″ S, 175° 23′ 6″ W
  - Generate dynamic sector yaml file
  - Generate trackfile:
    - Times: 2022-01-15 03:30 – 06:30 UTC
    - Static Center
  - Implement trackfile parser, and parser to AreaDefinition generator



Dynamic Changes

- Create dynamic yaml sector
- Create trackfile and trackfile parser
- Modify sector_metadata_generators interfaces (to accept volc family)
- Create a function to generate multiple AreaDefinitions from a trackfile (EX: tc_tracks.py)