Productos y Aplicaciones de Satélite en Latitudes Medias - Fase Presencial

Acceso y Procesamiento de Datos Satelitales - Prácticas

EUMETSAT

AEMet CENTRO REGIONAL DE FORMACIÓN DE LA OMM

Diego Souza

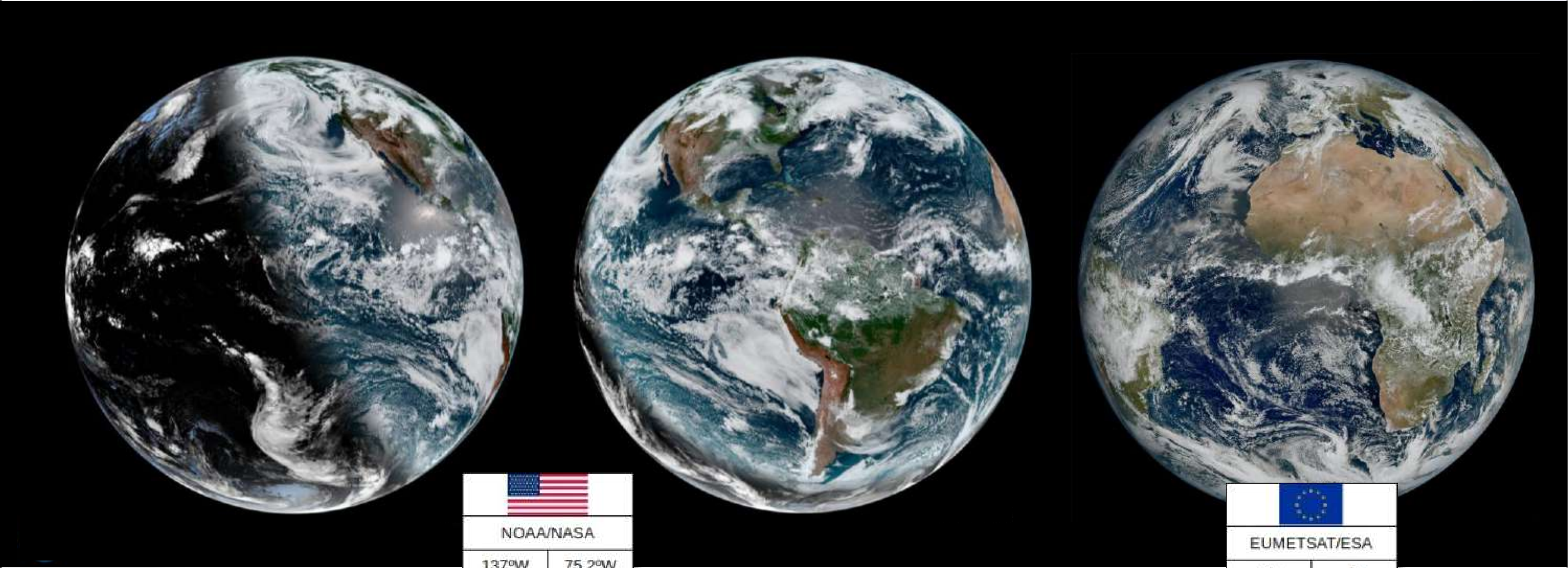
diego.souza@inpe.br

DISSM - División de Satélites y Sensores Meteorológicos
CGCT - Coordinación General de Ciencias de la Tierra
INPE - Instituto Nacional de Investigaciones Espaciales

MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E INOVAÇÃO

GOVERNO FEDERAL BRASIL UNIÃO E RECONSTRUÇÃO

Satélites GOES-Oeste | GOES-Este | Meteosat



NOAA/NASA	
137ºW	75.2ºW
GOES-18	GOES-16

EUMETSAT/ESA	
0º	0º
MSG-10	MTG-I1

Estructura de Nuestro Notebook Colab

Los pasos abajo se ejecutan una sola vez

- **Paso 1:** Características de la Máquina Virtual
- **Paso 2:** Instalación de Bibliotecas
- **Paso 3:** Importando las Bibliotecas Necesarias
- **Paso 4:** Credenciales EUMDAC
- **Paso 5:** Cargando las Funciones

Los ejemplos pueden ejecutarse varias veces, con parámetros diferentes	Fechas disponibles para cada dato
Mosaico Global (EUMETVIEW) : GOES-W + GOES-E + MSG 0° + MSG 45.5° + Himawari	Desde 06 de Junio de 2021 18:00 UTC, cada 3 horas
Imágenes GOES-Este (AWS)	Desde 10 de Julio de 2017
Imágenes GOES-Oeste (AWS)	Desde 28 de agosto de 2018 (G17) Desde 02 de agosto de 2022 (G18)
Imágenes MSG 0° (EUMDAC)	Desde 26 de marzo de 2004
Imágenes MSG 45.5° (EUMDAC)	Desde 01 de febrero de 2017
Descarga de Datos + Explicando Como se Procesan con Satpy	Cualquier fecha arriba
Imágenes Modelos Numéricos (varias fuentes)	Desde 21 de Enero de 2022 (ECMWF) Desde 26 de Febrero de 2021 (GFS)
Imágenes METAR (Unidata)	Últimos 10 días
Animaciones	Cualquier intervalo entre fechas arriba
Acceso y Procesamiento SAF	Cualquier intervalo de los datos SAF

Paso 1: Verificando la Configuración de la Máquina Virtual

Paso 1: Verificando la Configuración de la Máquina

```
[ ] # verifying the installed O.S.
    !cat /etc/issue
    !uname -a
    print('\n')

    # verifying the available RAM
    !grep MemTotal /proc/meminfo
    print('\n')

    # verifying the available HD space
    !df -h
    print('\n')

    # verifying the default Python installation directory
    !which python
    print('\n')

    # verifying the Python version
    !python --version
```

```
Ubuntu 22.04.2 LTS \n \l

Linux c013493fb1e6 5.15.120+ #1 SMP Wed Aug 30 11:19:59 UTC 2023 x86_64 x86_64 x86_64 GNU/Linux


MemTotal:       13294200 kB


Filesystem      Size  Used Avail Use% Mounted on
overlay         108G   28G   81G  26% /
tmpfs            64M     0   64M   0% /dev
shm             5.8G     0  5.8G   0% /dev/shm
/dev/root       2.0G  1.1G  885M  55% /usr/sbin/docker-init
tmpfs           6.4G   40K  6.4G   1% /var/colab
/dev/sda1        44G   28G   16G  65% /etc/hosts
tmpfs           6.4G     0  6.4G   0% /proc/acpi
tmpfs           6.4G     0  6.4G   0% /proc/scsi
tmpfs           6.4G     0  6.4G   0% /sys/firmware


/usr/local/bin/python


Python 3.10.12
```

Comandos Linux que "preguntan":

- **Sistema Operativo**
- **Memoria RAM disponible**
- **Espacio en Disco Duro**
- **Versión Python instalada**

Este paso es totalmente opcional, pero es interesante conocer las características de la máquina Linux de la versión gratuita de Google Colab

Salida con las informaciones:

Sistema Operativo: Ubuntu 22.04.2 LTS
Memoria RAM: 13 GB
Disco Duro: +100 GB
Python 3.10.2 instalado en /usr/local/bin/python

Paso 2: Instalando Bibliotecas y Descargando Archivos

Paso 2: Instalando Bibliotecas Python y Descargando Archivos Auxiliares

En este paso instalaremos todas las bibliotecas necesarias para esta sesión práctica y bajaremos algunos archivo auxiliares.

Nota: Sólo es necesario ejecutar esta celda una vez (tarda algunos minutos para completar la instalación).

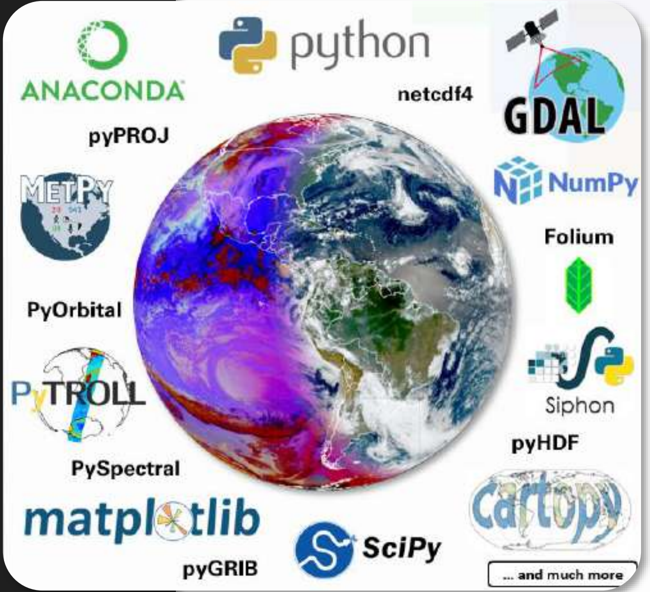
```
[ ] #======================================================================================================#
    # INSTALL THE REQUIRED LIBRARIES AND MODULES AND IMPORT ANCILLARY FILES
    #------------------------------------------------------------------------------------------------------#

    # install libraries with conda (system package manager)
    !pip install -q condacolab                                              # install conda and friends on Google Colab, easily
    import condacolab                                                       # import condacolab
    condacolab.install()                                                    # install condacolab
    !conda config --add channels conda-forge                                # configure conda channels
    !conda config --set channel_priority strict                             # configure conda channel priorities
    !mamba install -c conda-forge satpy netcdf4 pyproj pyspectral hdf5plugin # install libraries with conda

    # install libraries with pip (package installe
    !pip install herbie-data                                                # retrieve NWP model data
    !pip install eccodes                                                    # package developed by ECMWF for decoding and encoding messages
    !pip install ecmwflibs                                                  # wraps some of ECMWF libraries to be used by Python interfaces to ECMWF software
    !pip install git+https://github.com/blaylockbk/Carpenter_Workshop.git   # download utilities from past workshops
    !pip install cffi==1.16.0                                               # foreign function interface for Python calling C code.
    !pip install eumdac                                                     # provides simple access to the EUMETSAT data of all satellite missions
    !pip install boto3                                                      # amazon web services (AWS) SDK for python
    !pip install owslib                                                     # client programming with Open Geospatial Consortium (OGC) web service
    !pip install rioxarray                                                  # rasterio xarray extension
    !pip install cartopy                                                    # produce maps and other geospatial data analyses
    !pip install shapely --no-binary shapely --force                        # for cartopy to run OK on Colab

    # download a shapefile with states and provinces of the world
    !wget -c https://www.naturalearthdata.com/http//www.naturalearthdata.com/download/10m/cultural/ne_10m_admin_1_states_provinces.zip

    # unzip the shapefile
    !unzip -o ne_10m_admin_1_states_provinces.zip

    # download some example "CPT" files
    !wget -c https://github.com/diegormsouza/aemet_eumetsat_uruguay_2023/raw/main
    !unzip -o cpt_color_tables.zip

    # download the utilities.py script with the load_CPT function
    !wget -c https://raw.githubusercontent.com/diegormsouza/aemet_eumetsat_ur

    # change some files (necessary for Satpy to work with CMIPF and goes2go to
    !wget -c https://raw.githubusercontent.com/diegormsouza/aemet_eumetsat_uruguay
    !cp -f /content/abi_l2_nc.py /usr/local/lib/python3.10/site-packages/satpy/readers/abi_l2_nc.py
```

Usando "conda"

Usando "pip"

Nota: Este paso es el que más tarda, y siempre debe ejecutarse algunos minutos antes de empezar las prácticas

Vamos a trabajar con datos bien variados (GOES, MSG, Modelos, METAR, etc), entonces un numero considerable de bibliotecas Python son instaladas.



Paso 3: Importando las Bibliotecas Necesarias

Paso 3: Importando las Bibliotecas Necesarias

En este paso importaremos las bibliotecas instaladas en el paso anterior. Cada biblioteca tiene una aplicación especifica, a ser explorada en los scripts Python. Estaremos trabajando con datos Meteosat, GOES-Este, GOES-Oeste, datos de modelo y datos METAR, por lo tanto, usaremos una cantidad considerable de bibliotecas.

Nota: Sólo es necesario ejecutar esta celda una vez.

```python
#-----------------------------------------------------------------------------------------------------------#
# REQUIRED MODULES
#-----------------------------------------------------------------------------------------------------------#

import os                                                  # miscellaneous operating system interfaces
import glob                                                # unix style pathname pattern expansion
import zipfile                                             # tools to create, read, write, append, and list a ZIP file
import shutil                                              # high-level file operations
import requests                                            # elegant and simple HTTP library for Python
import numpy as np                                         # import the Numpy package
import cartopy, cartopy.crs as ccrs                        # produce maps and other geospatial data analyses
import cartopy.feature as cfeature                         # common drawing and filtering operations
import cartopy.io.shapereader as shpreader                 # import shapefiles
import pyproj                                              # python interface to PROJ (cartographic projections and coordinate transformations library)
import time                                                # time access and conversions
import owslib                                              # client programming with Open Geospatial Consortium (OGC) web service
from owslib.wms import WebMapService                       # owslib webmapservice
from datetime import timedelta, date, datetime             # basic date and time types
from pyresample import geometry                            # classes for describing different geographic areas using a mesh of points or pixels
import matplotlib.pyplot as plt                            # plotting library
import matplotlib.patheffects as PathEffects               # define classes for path effects
from matplotlib.offsetbox import AnchoredText              # adds an anchored text box in the corner
from matplotlib.offsetbox import OffsetImage               # change the image size (zoom)
from matplotlib.offsetbox import AnnotationBbox            # creates an annotation using an OffsetBox
from matplotlib import cm                                  # Colormap handling utilities
from utilities import loadCPT                              # import the CPT convert function
from utilities import download_MCMIPF                      # function to download MCMIPF files
import eumdac                                              # simple access to the EUMETSAT data of all satellite missions
import hdf5plugin                                          # for reading compressed data, a decompression library is needed
from satpy import Scene                                    # scene object to hold satellite data
from satpy import available_readers                        # available readers from satpy
from satpy.writers import get_enhanced_image               # get an enhanced version of dataset as an XRImage instance
from herbie import Herbie                                  # retrieve NWP model data
from toolbox import EasyMap, pc                            # utilities from previous workshops
from metpy.io import metar                                 # parse METAR-formatted data
from metpy.plots import current_weather, sky_cover, StationPlot  # contains functionality for making meteorological plots
from metpy.calc import reduce_point_density                # return a mask to reduce the density of points in irregularly-spaced data
import rioxarray                                           # rasterio xarray extension
import imageio                                             # python interface to read and write a wide range of image data
import warnings                                            # warning control
warnings.filterwarnings("ignore")                          # disable warnings
```

```
I'm building Herbie's default config file.
```

```
You're ready to go.
You may edit the config file here:
/root/.config/herbie/config.toml
```

```
/usr/local/lib/python3.10/site-packages/gribapi/__init__.py:23: UserWarning: ecCodes 2.31.0 or higher is recommended. You are running version 2.30.0
  warnings.warn(
```

Después de instalar es necesario importar bibliotecas. Algunos ejemplos de uso

Cartopy para añadir mapas

Boto3 para descargar datos GOES desde AWS

EUMDAC para descargar datos MSG

Satpy para graficar datos GOES y MSG

Herbie para descargar datos de modelo

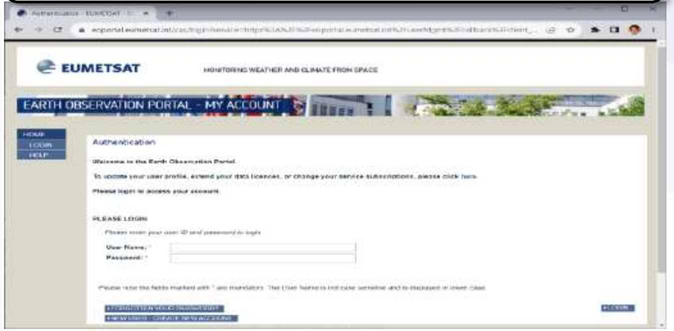
Metpy para graficar datos METAR

y mucho más...

Paso 4: Credenciales de Usuario EUMDAC

Para descargar datos de EUMETSAT con EUMDAC, es necesario crear un "Access Token".

1. Crear una cuenta en: eoportal.eumetsat.int y hacer el login



2. Acceder a la página: https://api.eumetsat.int/api-key/, copiar y pegar el "consumer key" y "consumer secret" en tu copia del notebook Colab (paso 4), entre comillas

Paso 4: Credenciales de Usuario EUMDAC (EUMETSAT Data Services Client)

Para descargar datos usando el cliente EUMDAC (EUMETSAT Data Access Client), primero es necesario registrarse en https://eoportal.eumetsat.int/ y despues acceder al siguiente enlace: https://api.eumetsat.int/api-key/. En esta página, es necesario copiar las llaves **"Consumer key"** y **"Consumer secret"** y ingresarlas en las variables de la siguiente celda.

Nota: Sólo es necesario ejecutar esta celda una vez.

```
# EUMDAC (EUMETSAT Data Services Client) Access Token
# insert your personal key and secret into the single quotes below.
# first, create an account at this link: https://eoportal.eumetsat.int/
# and get the consumer key and secret at this link: https://api.eumetsat.int/api-key/
consumer_key = [redacted]
consumer_secret = [redacted]

# verifying whether the creation of the Access Token was successful.
credentials = (consumer_key, consumer_secret)
token = eumdac.AccessToken(credentials)
try:
    print(f"This token '{token}' expires {token.expiration}")
except requests.exceptions.HTTPError as error:
    print(f"Error when tryng the request to the server: '{error}'")
```

```
This token '3ca7faf3-4af8-3422-8935-1a3935e58c08' expires 2023-10-31 14:52:37.708641
```

Al ejecutar la celda de código, esta salida indica que las credenciales están correctas

Paso 5: Cargando las Funciones

El “trabajo sucio” es hecho por funciones! Si no tienes experiencia con programación, sin problemas, podrás cambiar pocas variables en los ejemplos, fácilmente. Si tienes experiencia con Python y programación, puedes expandir esta celda y ver como las cosas se hacen.

Paso 5: Cargando las Funciones

Para generar imagenes necesitamos hacer algunas operaciones complejas. Para facilitar la manipulación de los datos, se han creado algunas funciones que hacen el trabajo más difícil. Estas funciones son cargadas al ejecutar la siguiente celda.

Nota: Sólo es necesario ejecutar esta celda una vez.

Al hacer click acá, la celda se expande. No es necesario expandir (hay bastante código Python), a no ser que estés interesado en conocer como las imágenes son generadas

Algunas funciones para descargar y graficar datos (no es necesario expandir esta celda, solo ejecutarla)

0 s

Mostrar código

Al ejecutar esta celda de código, todas las funciones necesarias (creadas previamente por el creador del notebook) serán cargadas

Importante: Acciones Necesarias

En cada día, antes de las actividades prácticas Python, es necesario ejecutar las celdas de los pasos 2 a 5 algunos minutos antes, para que los notebooks estén listos cuando empecemos.

Referencia: Latitudes y Longitudes

En todos los scripts es posible seleccionar una región especifica, en la variable "extent" (min lon, min lat, max lon, max lat).
En el notebook hay esta imagen de referencia con las coordinadas.

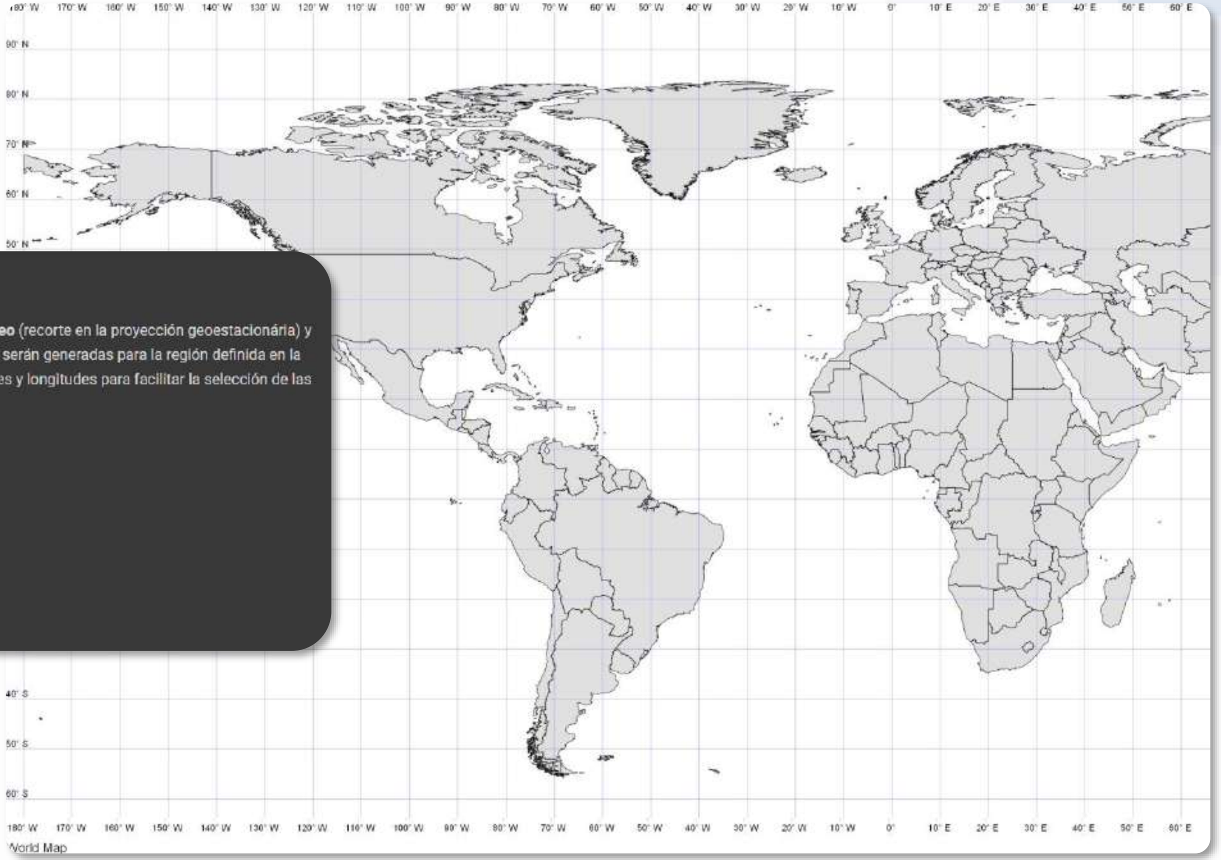
Referencia: Latitudes y Longitudes

En cada script, en variable **"area"** puedes elegir la opción **"full_disk"** (Disco Completo), **custom_geo** (recorte en la proyección geoestacionária) y **"custom"** (recorte en la proyección equidistante). Al elegir **custom_geo** o **"custom"**, las imágenes serán generadas para la región definida en la variable **"extent"** (min lon, min lat, max lon, max lat). Abajo, una imagen de referencia con latitudes y longitudes para facilitar la selección de las coordenadas del usuario, así como algunos ejemplos de "extent".

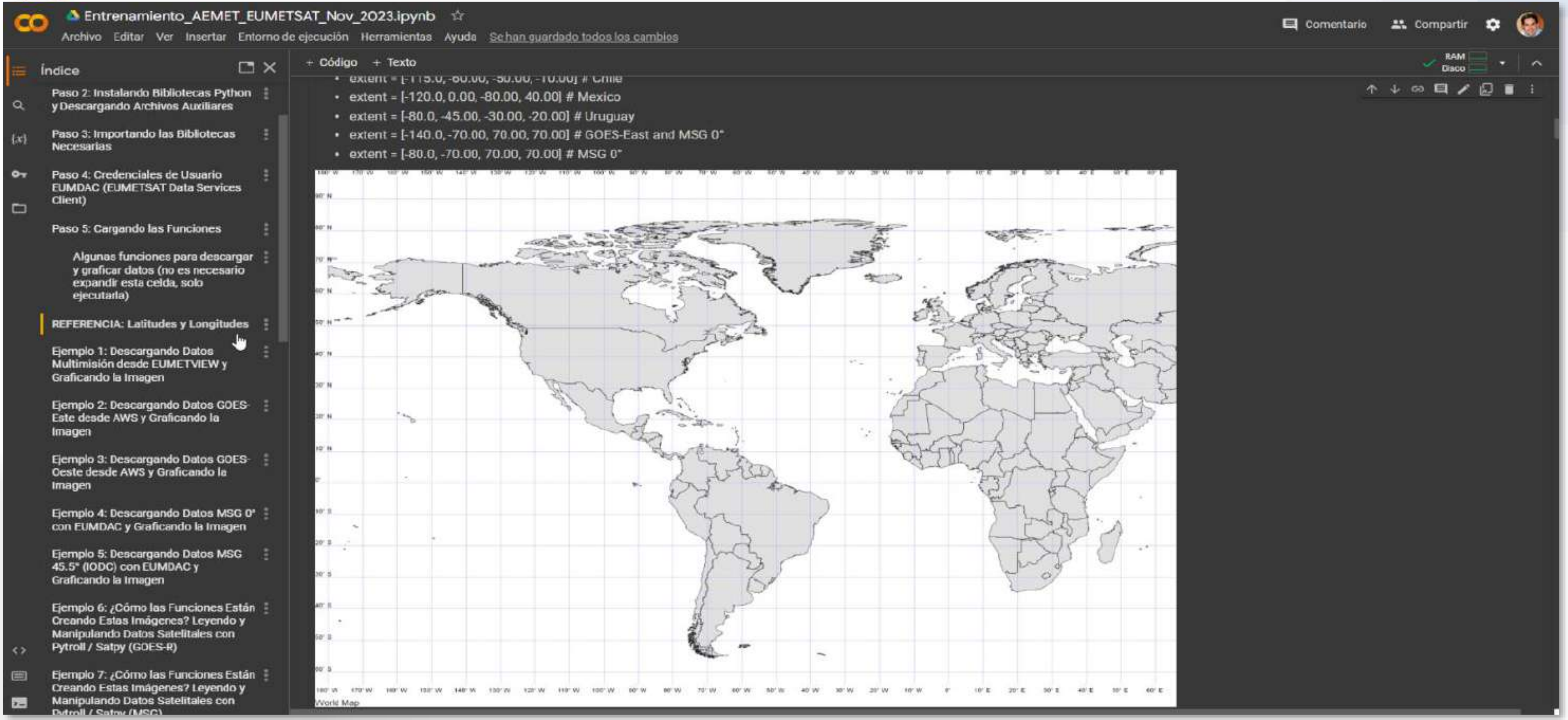
Algunos extent de ejemplo:

- extent = [-90.0, -60.00, -30.00, 15.00] # South America
- extent = [-110.0, 0.00, -60.00, 30.00] # Central America
- extent = [-90.0, -60.00, -40.00, -10.00] # Argentina
- extent = [-80.0, -40.00, -30.00, 10.00] # Brazil
- extent = [-115.0, -60.00, -50.00, -10.00] # Chile
- extent = [-120.0, 0.00, -80.00, 40.00] # Mexico
- extent = [-80.0, -45.00, -30.00, -20.00] # Uruguay
- extent = [-140.0, -70.00, 70.00, 70.00] # GOES-East and MSG 0°

Además, se ha dejado algunos extent estándar para regiones especificas. Nota: Con el índice lateral de Colab es fácil acceder a esta referencia.

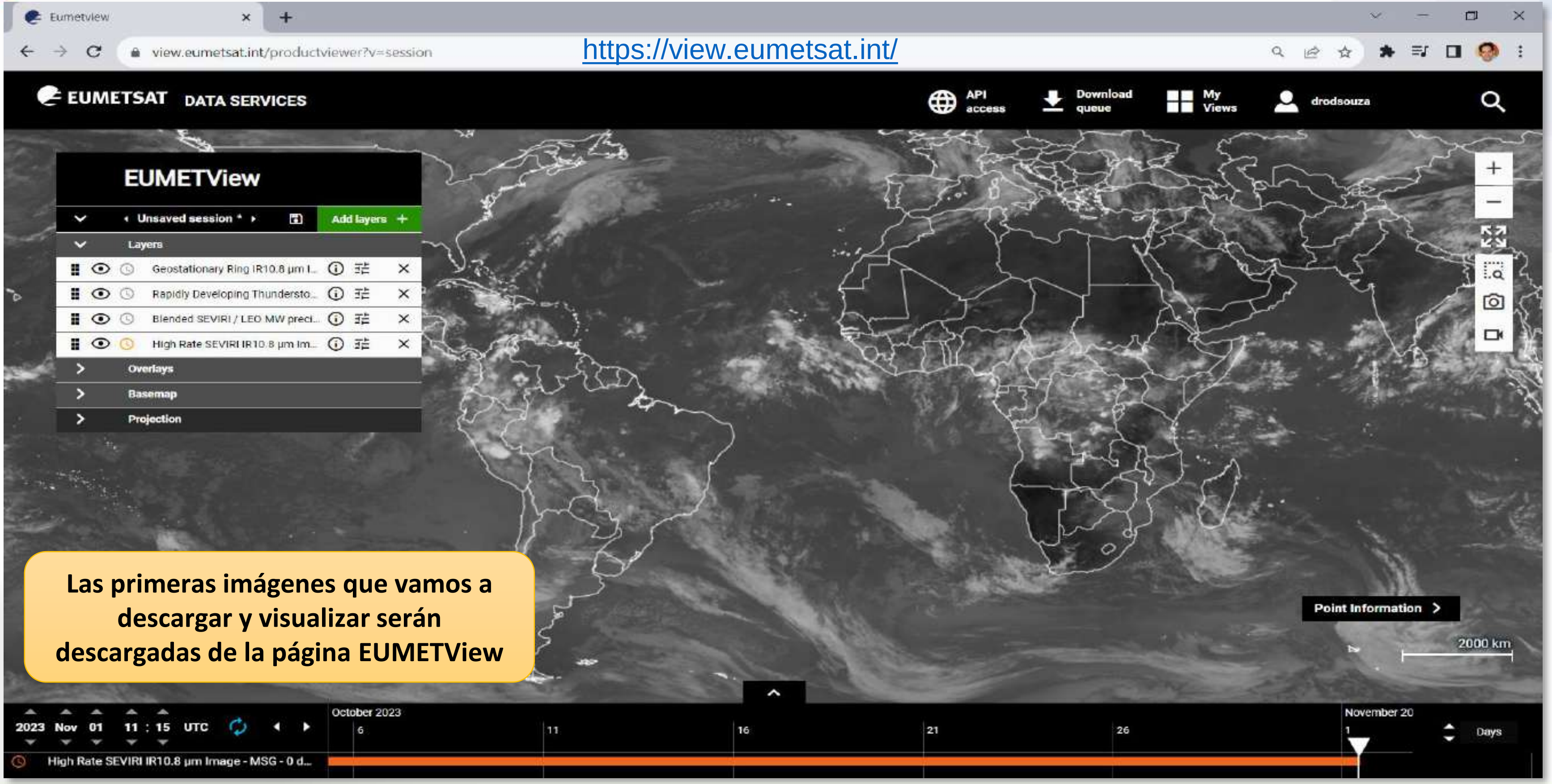


Vamos Usar el Índice Frecuentemente



¡Estamos listos para empezar!

Ejemplo 1: Datos Multimisión EUMETView



https://view.eumetsat.int/

Las primeras imágenes que vamos a descargar y visualizar serán descargadas de la página EUMETView

Ejemplo 1: Datos Multimisión EUMETView

EUMETView

www.eumetsat.int

Automated access to the imagery and data

WMS, WFS and WCS (OGC API)

Possibility for further customized downstream use

Web Pages (Javascript+HTML)

Python Scripts

Desktop GIS Applications

EUM/USC/VWG/23/1371051, v1 Draft, 19 July 2023

11

Ejemplo 1: Datos Multimisión EUMETView

Ejemplo 1: Descargando Datos Multimisión desde EUMETVIEW y Graficando la Imagen

```
[4] #-----------------------------------------------------------------------------------------------------
    # Training - Satellite Data Access and Processing - Example 1: Downloading and Visualizing Data from EUMETVIEW
    # Author: Diego Souza (INPE/CGCT/DISSM)
    #-----------------------------------------------------------------------------------------------------
    # georing available since June 06 2021 18:00 UTC '2021-06-06 18:00' (note: data available
    # available composites: 'ir108', 'airmass', 'natural_color', 'ash', 'dust'
    # max. res: 6
    plot_ring(date_sat='2023-10-28 15:00',
              composite='airmass',
              resolution=12,
              area='custom',
              extent=[-140.0, -70.00, 70.00, 70.00],
              coast_color='white',
              countries_color='white',
              grid_color='white',
              overwrite=True,
              figsize=[20,20])
```

```
--------------------------------------------------------------------------------
Downloading the following file from EUMETView: GLOBAL_mumi_wideareacoverage_rgb_airmass_20

Plotting the file:
/content/sample_data/GLOBAL_mumi_wideareacoverage_rgb_airmass_2023-10-28T15-00Z.tif

Total processing time (satellite): 9.56 seconds.
--------------------------------------------------------------------------------
'output/georing_airmass_2023-10-28_15:00.png'
```

Nombre de la función

Parámetros

Salida con informaciones

Parámetros

- Fecha 'YYYY-MM-DD HH:MM'
- Compuesto deseado
- Resolución (mínimo 6)
- Área deseada
- Coordinadas
- Color de las líneas de costa
- Color de los mapas de países
- Color de las líneas de ref.
- Sobrescribir archivo
- Tamaño de la figura (pulgadas)

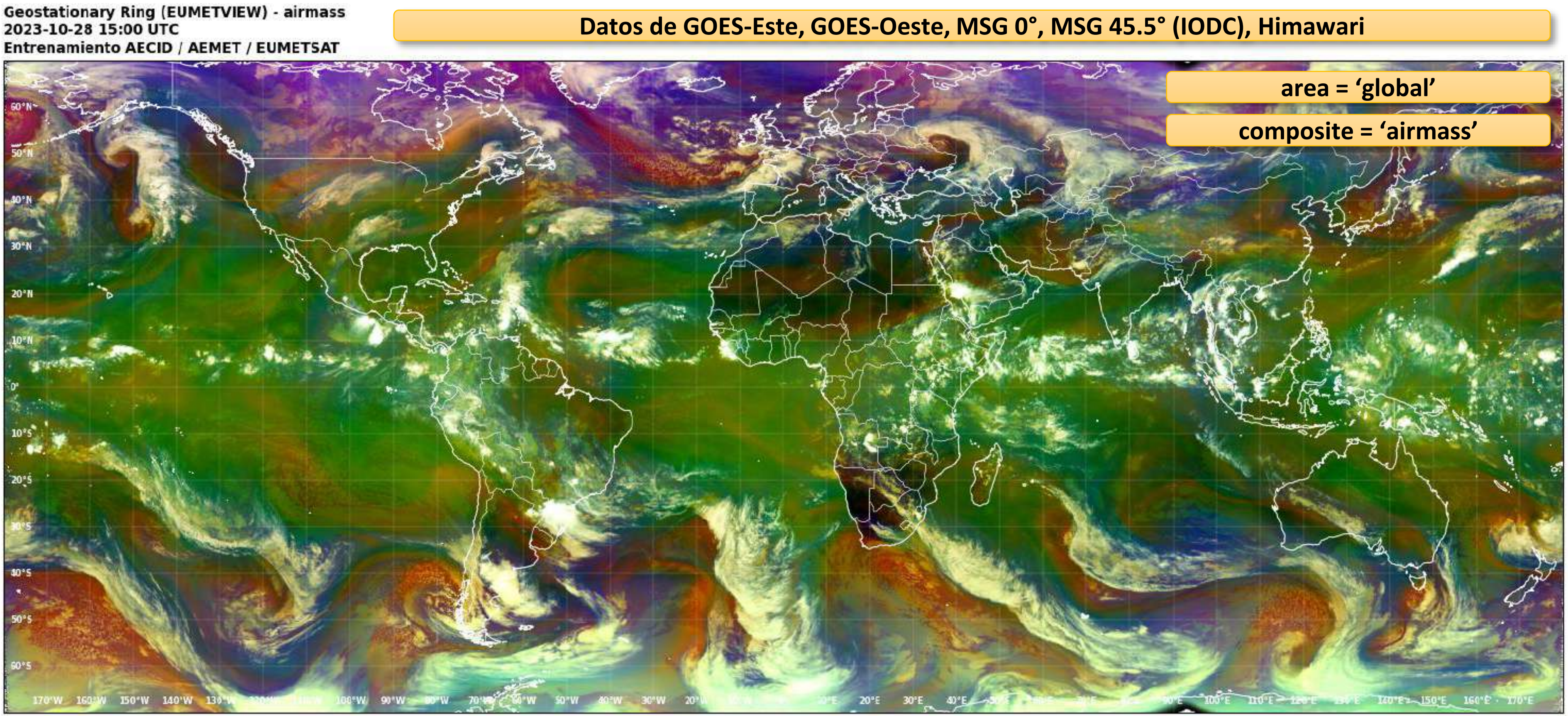
Desde 06 de junio de 2021, cada tres horas

'ir108', 'airmass', 'natural_color', 'ash', 'dust'

'custom' o 'global'

Ejemplo 1: Datos Multimisión EUMETView

Datos de GOES-Este, GOES-Oeste, MSG 0°, MSG 45.5° (IODC), Himawari



area = ‘global’

composite = ‘airmass’

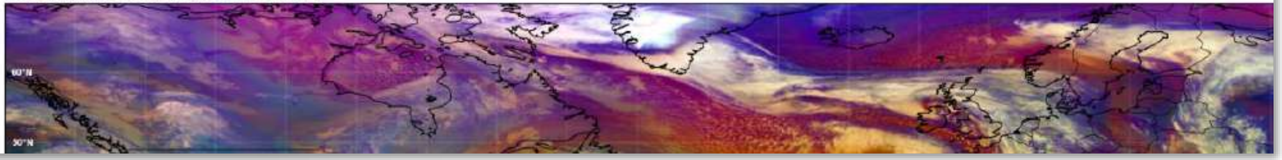
Ejemplo 1: Datos Multimisión EUMETView

Ejemplo 1: Descargando Datos Multimisión desde EUMETVIEW y Graficando la Imagen

```
#-----------------------------------------------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example 1: Downloading and Visualizing Data from EUMETVIEW
# Author: Diego Souza (INPE/CGCT/DISSM)
#-----------------------------------------------------------------------------------------------------------

# georing available since June 06 2021 18:00 UTC '2021-06-06 18:00' (note: data available every 3 hours)
# available composites: 'ir108', 'airmass', 'natural_color', 'ash', 'dust'
# max. res: 6
plot_ring(date_sat='2023-11-01 15:00',
          composite='airmass',
          resolution=6,
          area='custom',
          extent=[-140.0, -70.00, 70.00, 70.00],
          coast_color='black',
          countries_color='black',
          grid_color='white',
          overwrite=True,
          figsize=[20,20])
```

```
-----------------------------------------------------------------------------------------------------------
Downloading the following file from EUMETView: GLOBAL_mumi_wideareacoverage_rgb_airmass_2023-11-01T15-00Z.tif

Plotting the file:
/content/sample_data/GLOBAL_mumi_wideareacoverage_rgb_airmass_2023-11-01T15-00Z.tif

Total processing time (satellite): 22.13 seconds.
-----------------------------------------------------------------------------------------------------------
'output/georing_airmass_2023-11-01_15:00.png'
```

Geostationary Ring (EUMETVIEW) - airmass
2023-11-01 15:00 UTC
Entrenamiento AECID / AEMET / EUMETSAT



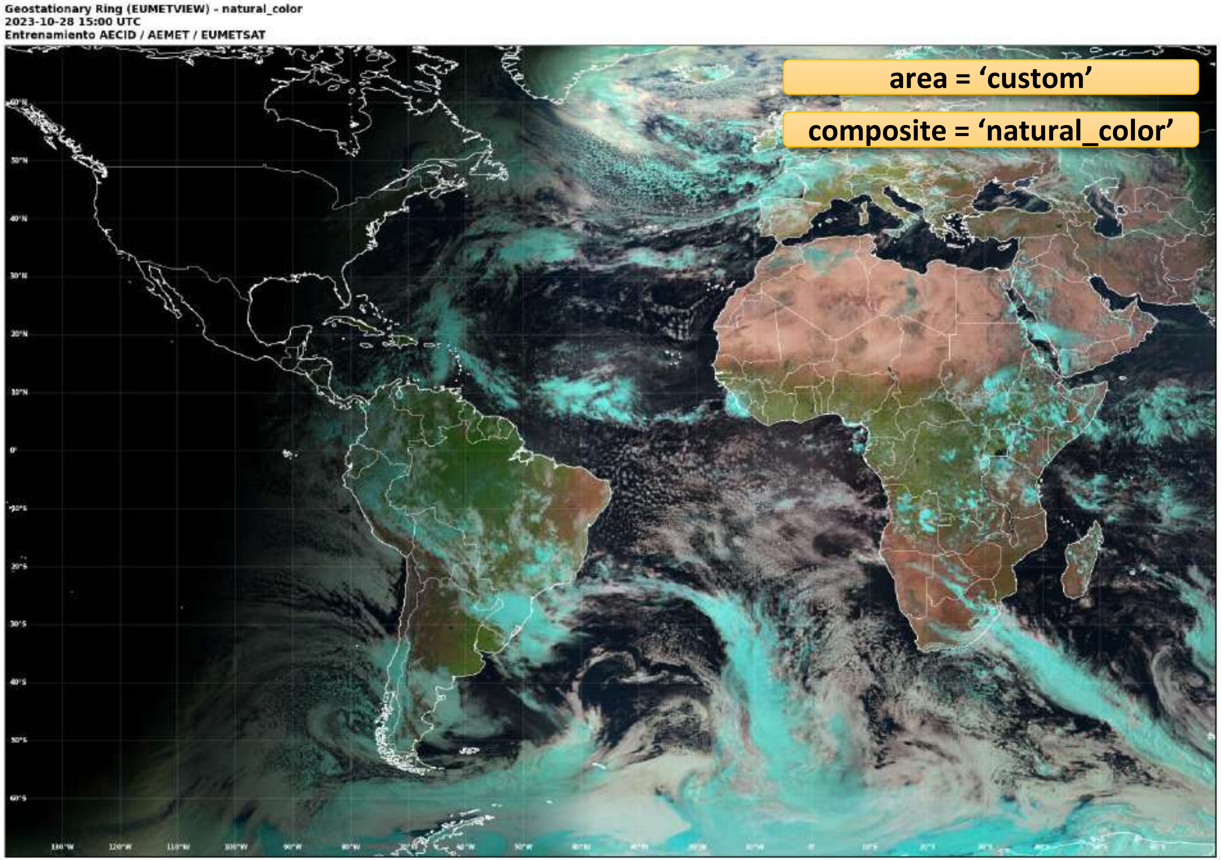
Archivo a ser descargado

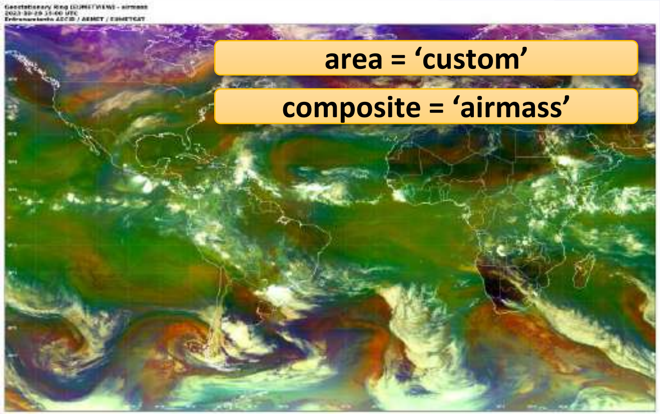
Graficando la información

Tiempo necesario

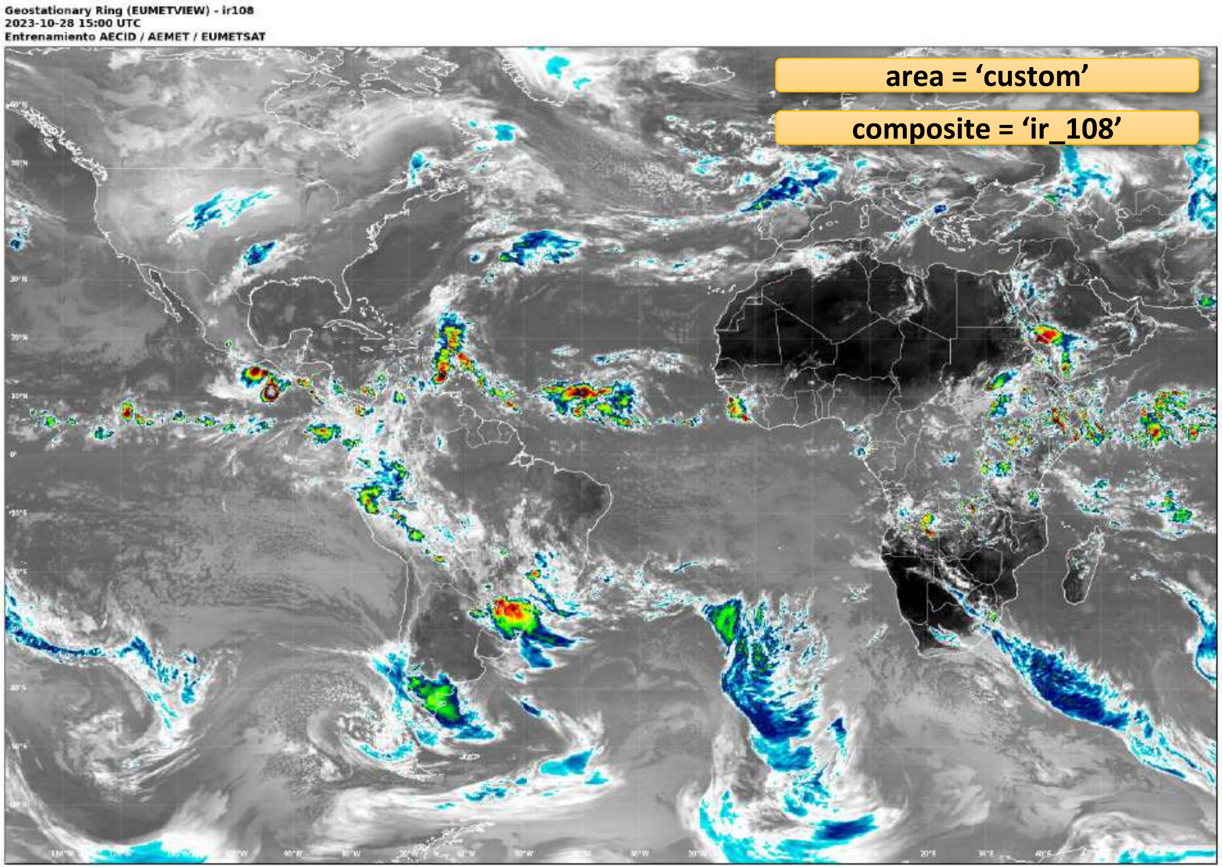
Nombre de la imagen generada (carpeta output)

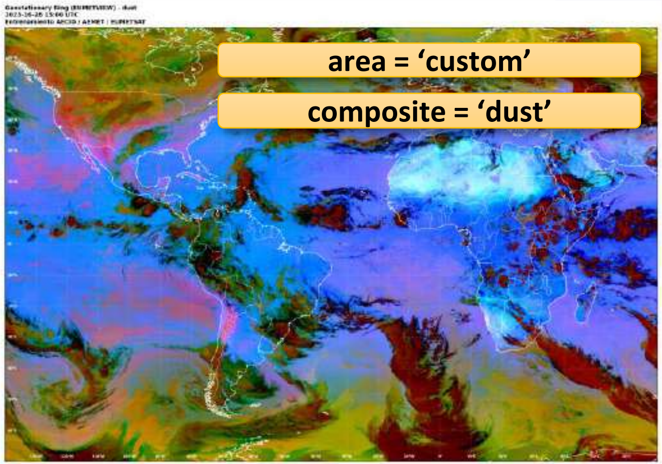
Ejemplo 1: Datos Multimisión EUMETView



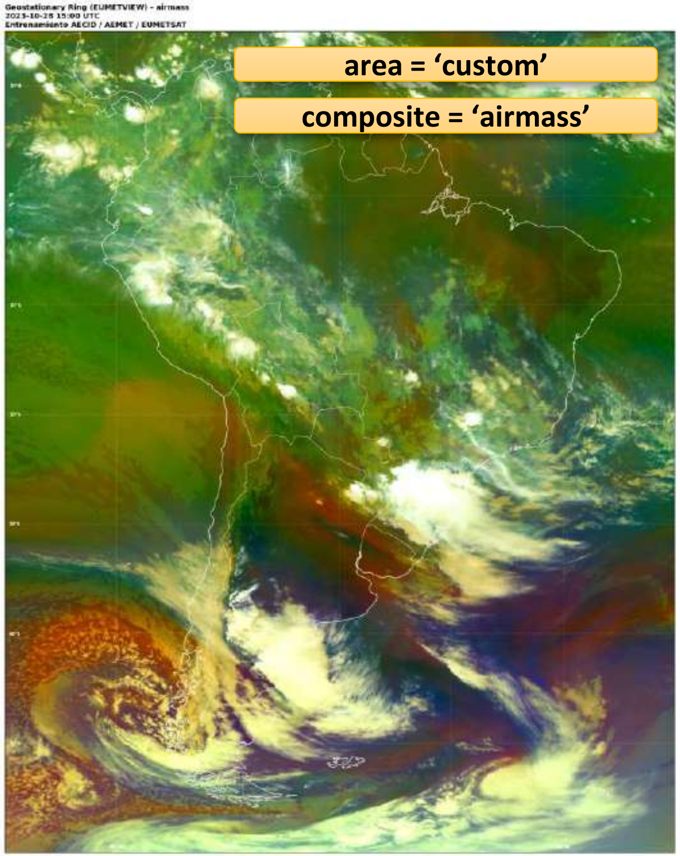


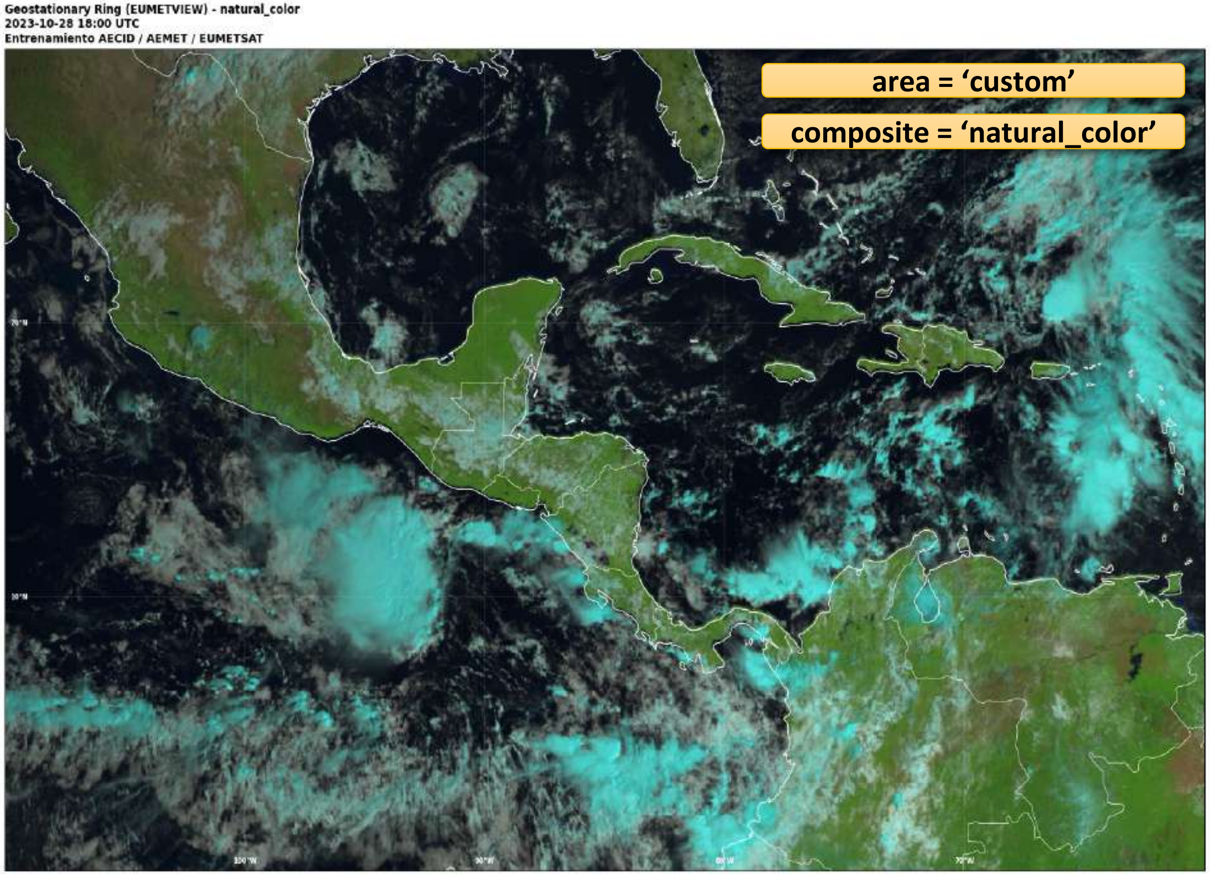
Ejemplo 1: Datos Multimisión EUMETView



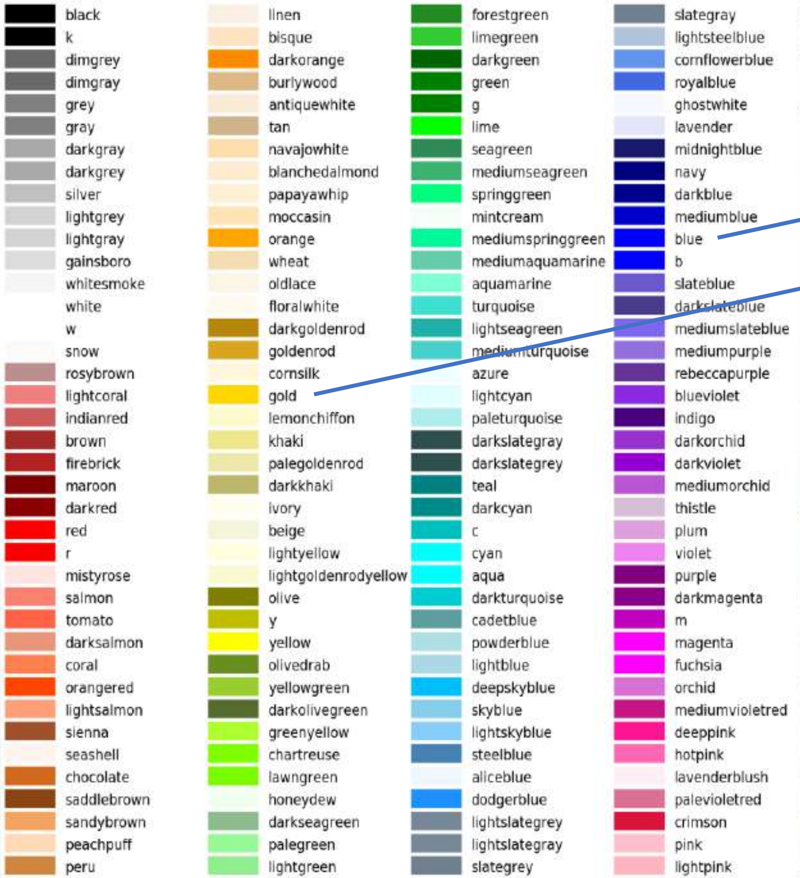


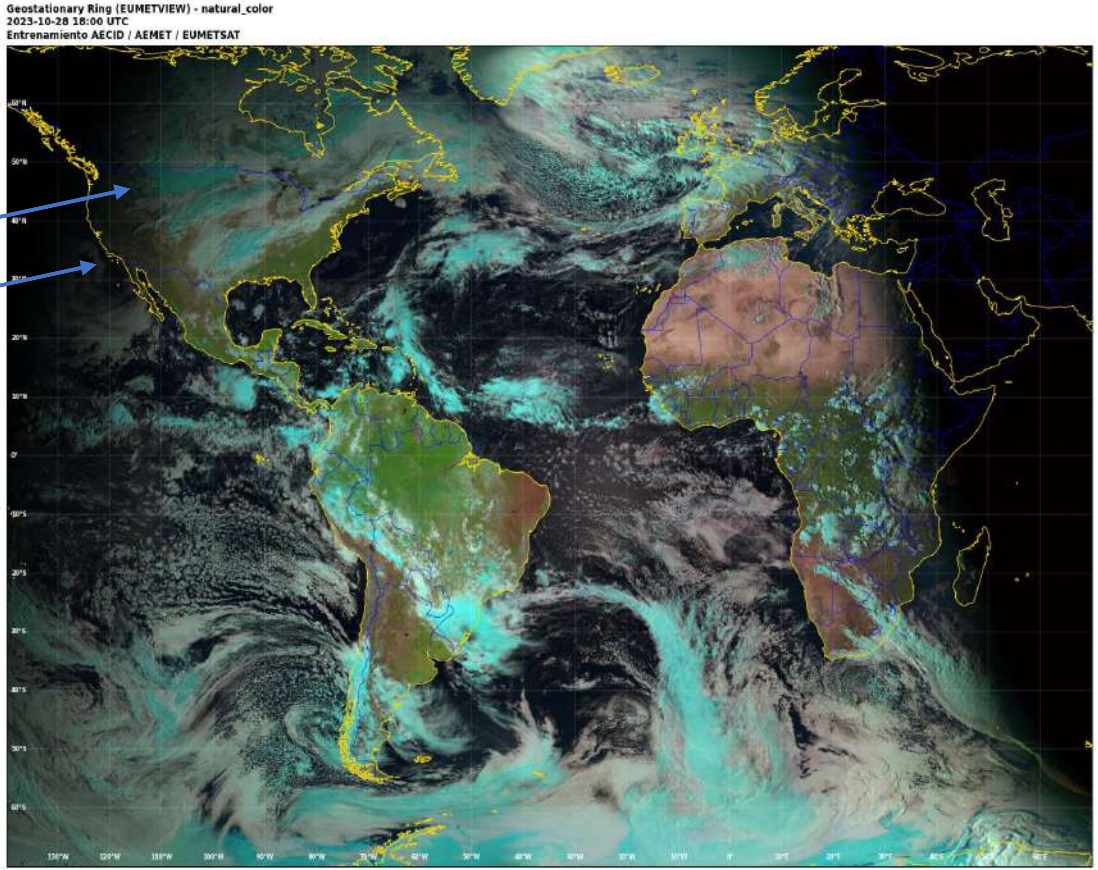
Ejemplo 1: Datos Multimisión EUMETView





Ejemplo 1: Nombres de Colores Estándar Matplotlib





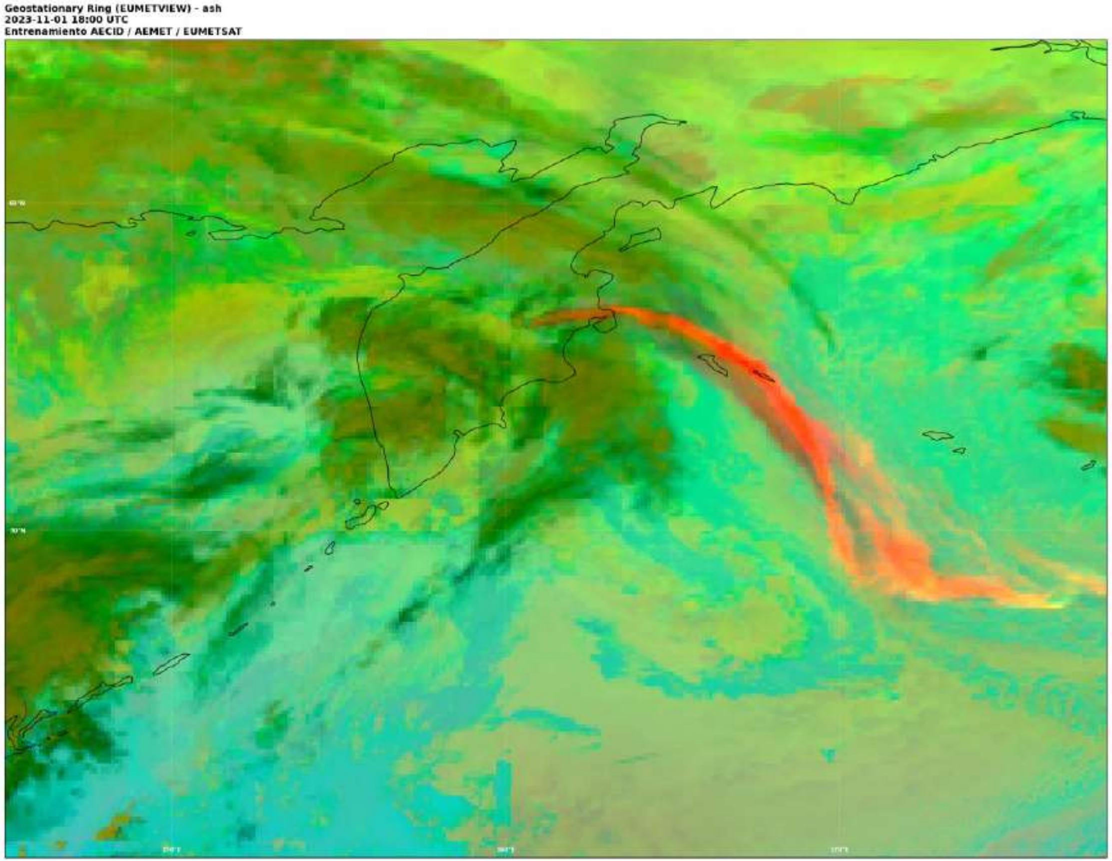
Ejercicio: Erupción de Klyuchevskaya en la Península de Kamchatka

Fecha:
2023-11-01 18:00 UTC

Región:
area = 'custom'
extent = [145.0, 40.00, 178.00, 65.00]

Compuesto:
'ash'

Referencia:
https://cimss.ssec.wisc.edu/satellite-blog/archives/55417



Ejemplo 1: Datos Multimisión EUMETView

Input del Usuario:
Fecha, Compuesto, Resolución, Área, Extensión, Colores y Tamaño

Descarga de Datos desde EUMETView
Con el Open Geospatial Consortium (OGC) Web Service (OWS)

Procesa y Grafica la Imagen

```
#######################################################################################################################
# plot_ring: Function to download and visualize geostationary ring data from EUMETView
# Author: Diego Souza (INPE/CGCT/DISSM)
#######################################################################################################################

def plot_ring(date_sat, composite, resolution, area, extent, coast_color, countries_color, grid_color, overwrite, figsize):
  print(f'----------------------------------------------------------------------------------------------------------------
```

Ejemplo 2: Datos GOES-Este

Ejemplo 2: Descargando Datos GOES-Este desde AWS y Graficando la Imagen

```
#-----------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example 2: Downloading GOES-
# Author: Diego Souza (INPE/CGCT/DISSM)
#-----------------------------------------------------------------------

#                       2017, 'goes17' since August 28 2018, 'go
#                       erated images every 15 minutes (instead of 10) until a few months after launch

#                       ptions below have been tested):
#                       5', 'C07', 'C08', 'C09', 'C10', 'C11', 'C
# 'airmass', 'ash', 'cimss_cloud_type', 'cimss_true_color', 'cimss_true_color_su
# 'cloudtop', 'color_infrared', 'colorized_ir_clouds', 'convection', 'day_microphysics', 'dust', 'fire_temperature_awips', 'fog', 'green_snow
# 'night_microphysics', 'overview', 'rocket_plume_day', 'rocket_plume_night', 's

plot_sat(sat='goes16',
         date_sat='2023-10-28 15:00',
         composite='airmass',
         area='full_disk',
         extent=[-90.0, -60.00, -30.00, 15.00],
         coast_color='white',
         countries_color='white',
         grid_color='white',
         figsize=[15,15])
```

Nombre de la función

Parámetros

Parámetros

- Satélite
- Fecha 'YYYY-MM-DD HH:MM'
- Compuesto deseado
- Área deseada
- Coordinadas
- Color de las líneas de costa
- Color de los mapas de países
- Color de las líneas de ref.
- Tamaño de la figura (pulgadas)

'goes16', 'goes17', 'msg0' o 'msg45'

Canales: 'C01, 'C02', 'C03', 'C04', 'C05', 'C06', 'C07', 'C08', 'C09', 'C10', 'C11', 'C12', 'C13', 'C14', 'C15', 'C16'

Composites: 'airmass', 'ash', 'cimss_cloud_type', 'cimss_true_color', 'cimss_true_color_sunz_rayleigh', 'cira_day_convection', 'cira_fire_temperature', 'cloud_phase', 'cloud_phase_distinction', 'cloudtop', 'color_infrared', 'colorized_ir_clouds', 'convection', 'day_microphysics', 'dust', 'fire_temperature_awips', 'fog', 'green_snow', 'land_cloud', 'natural_color', 'night_fog', 'night_microphysics', 'overview', 'rocket_plume_day', 'rocket_plume_night', 'snow', 'so2', 'tropical_airmass', 'true_color', 'water_vapors1', 'water_vapors2'

'full_disk', 'custom_geo', 'custom'

```
-----------------------------------------------------------------------
File /content/sample_data/OR_ABI-L2-MCMIPF-M6_G16_s20233011500204_e20233011509525_c20233011509594.nc already exists

Plotting the file:
/content/sample_data/OR_ABI-L2-MCMIPF-M6_G16_s20233011500204_e20233011509525_c20233011509594.nc


WARNING:matplotlib.image:Clipping input data to the valid range for imshow with RGB data ([0..1] for floats or [0..255] for integers).
Total processing time (satellite): 21.81 seconds.
-----------------------------------------------------------------------
'output/goes16_airmass_2023-10-28_15:00.png'
```

Salida con informaciones

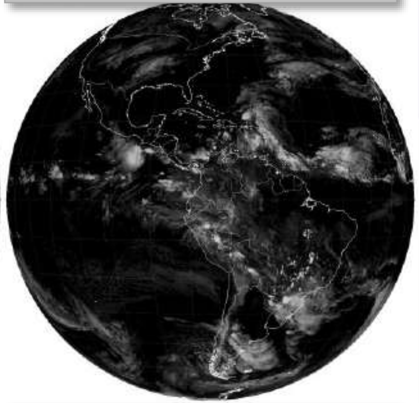
Datos GOES-Este: Algunos “Composites” Disponibles

C01

C02

C03

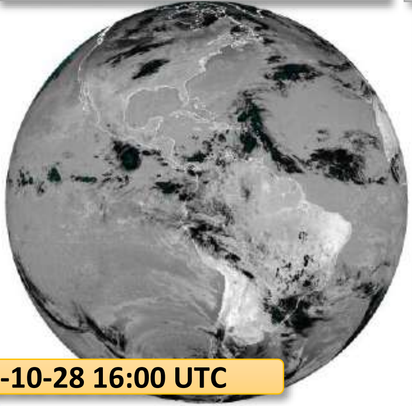
C04



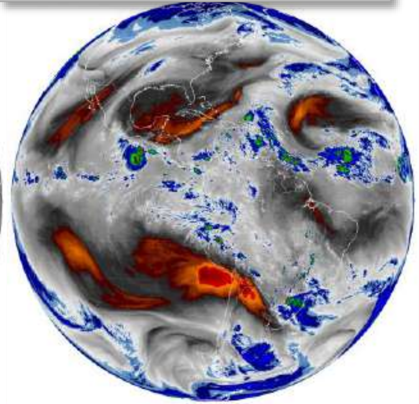
C05

C06

C07

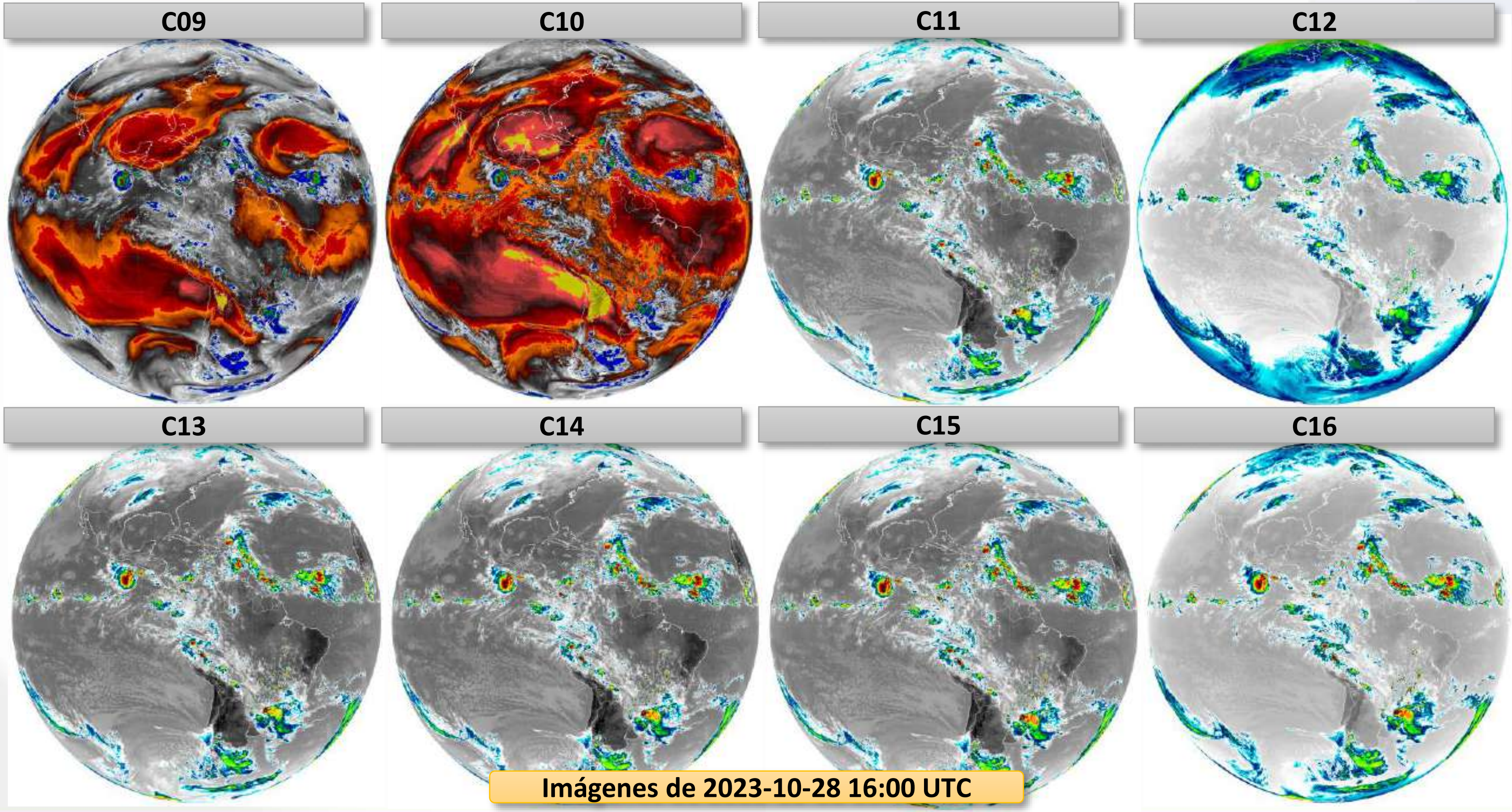


C08



Imágenes de 2023-10-28 16:00 UTC

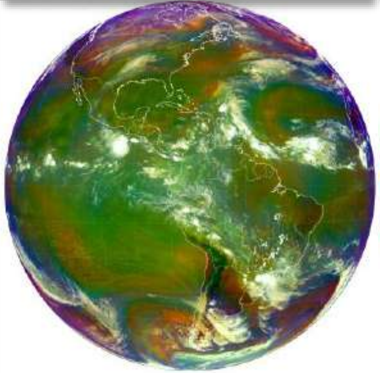
Datos GOES-Este: Algunos “Composites” Disponibles



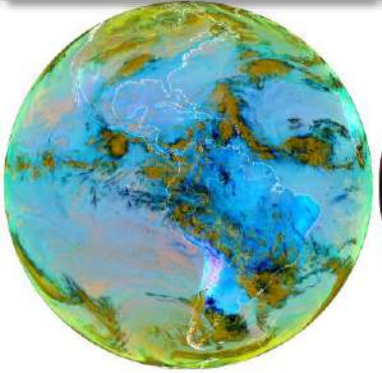
Imágenes de 2023-10-28 16:00 UTC

Datos GOES-Este: Algunos “Composites” Disponibles

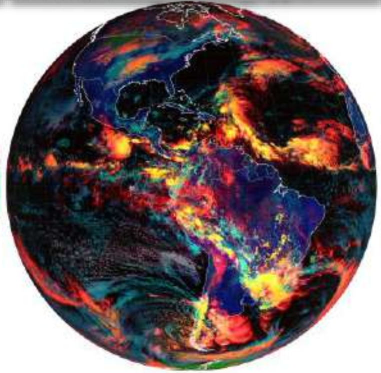
airmass



ash

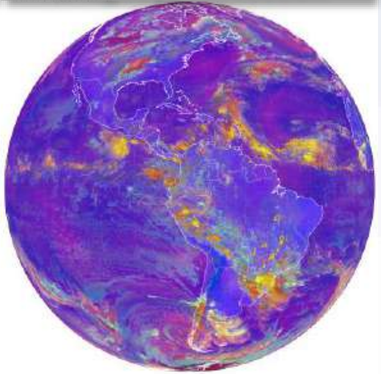


cimss_cloud_type

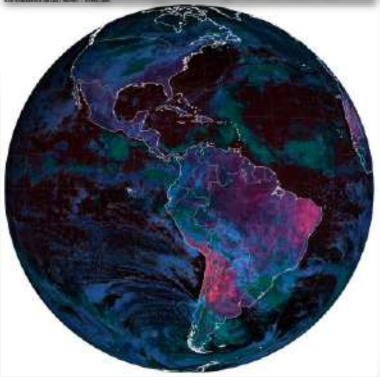


cimss_true_color

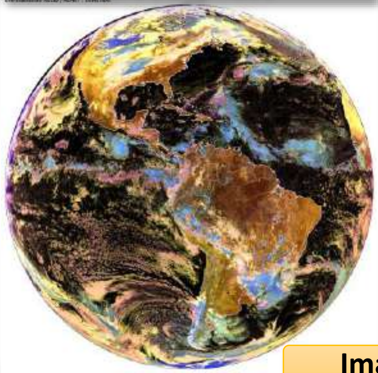
cira_day_convection



cira_fire_temperature

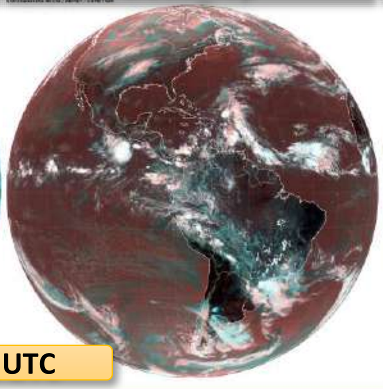


cloud_phase

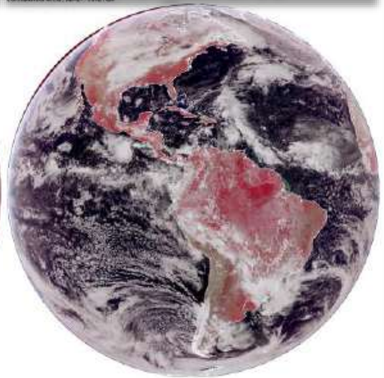


cloud_phase_distinction

cloud_top



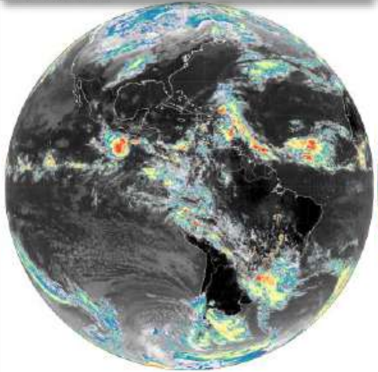
color_infrared



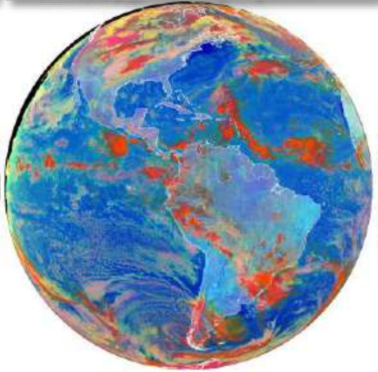
Imágenes de 2023-10-28 16:00 UTC

Datos GOES-Este: Algunos “Composites” Disponibles

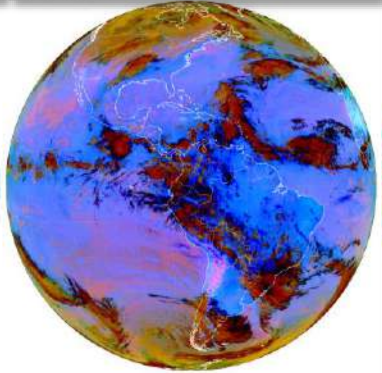
colorized_ir_clouds



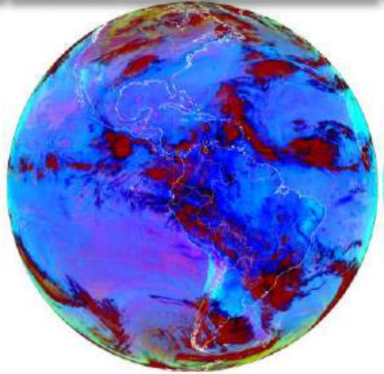
day_microphysics



dust

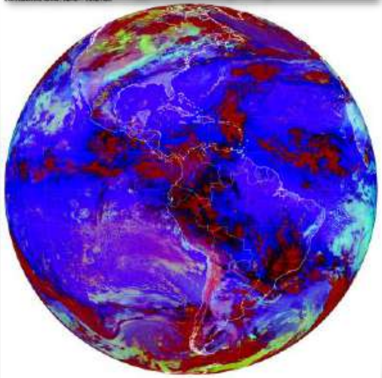


fog



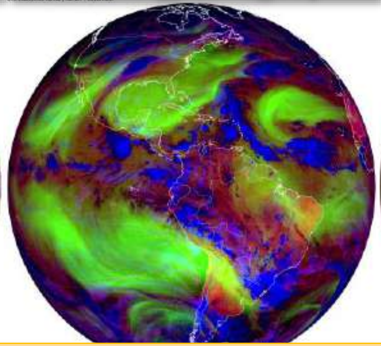
land_cloud

night_microphysics

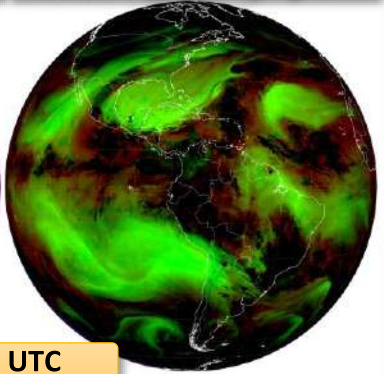


overview

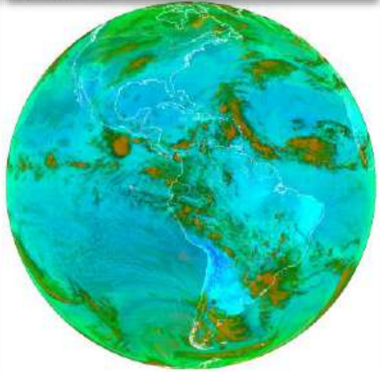
rocket_plume_day



rocket_plume_night



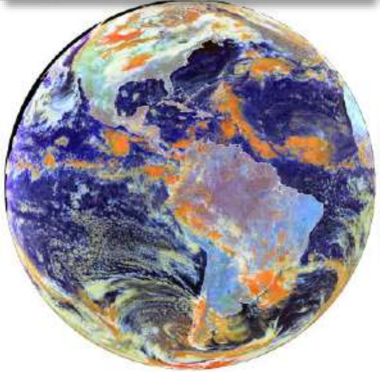
so2



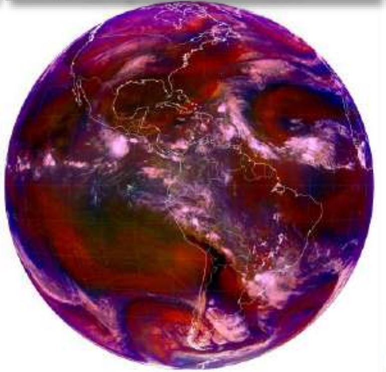
Imágenes de 2023-10-28 16:00 UTC

Datos GOES-Este: Algunos “Composites” Disponibles

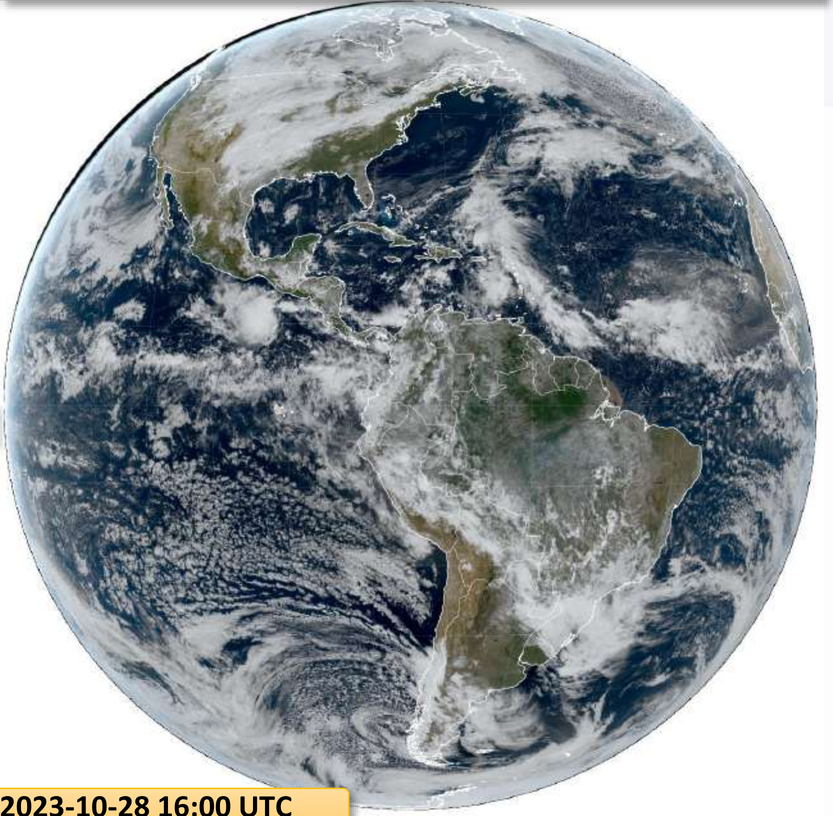
snow



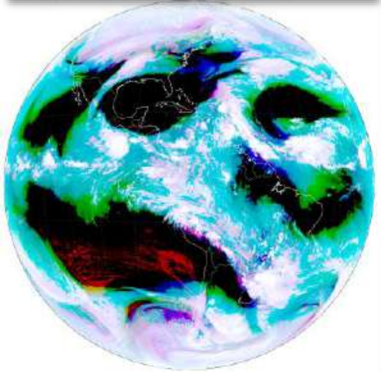
tropical_airmass



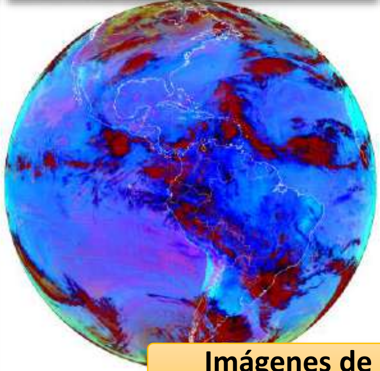
true_color



water_vapors1



water_vapors2

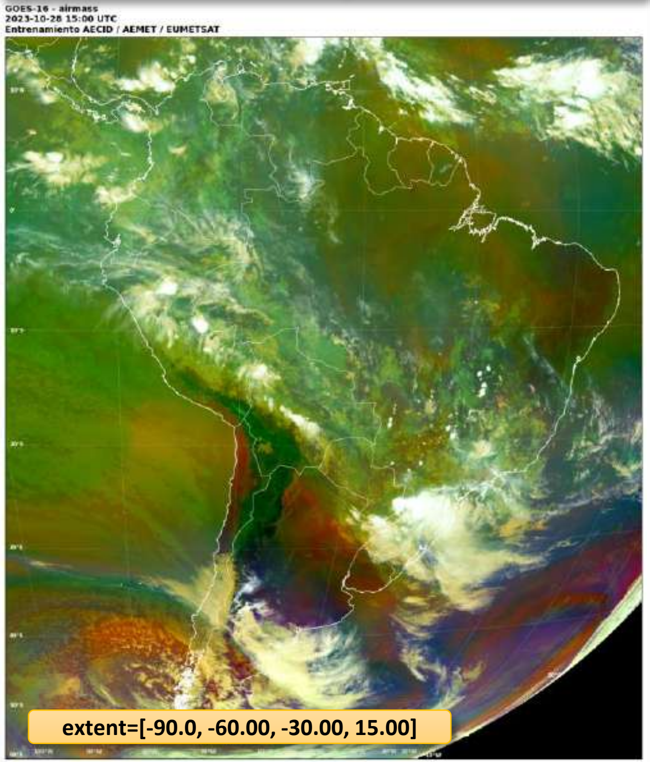


Imágenes de 2023-10-28 16:00 UTC

Ejemplo 2: Datos GOES-Este

area = 'custom_geo' (recorte en la proyección original)

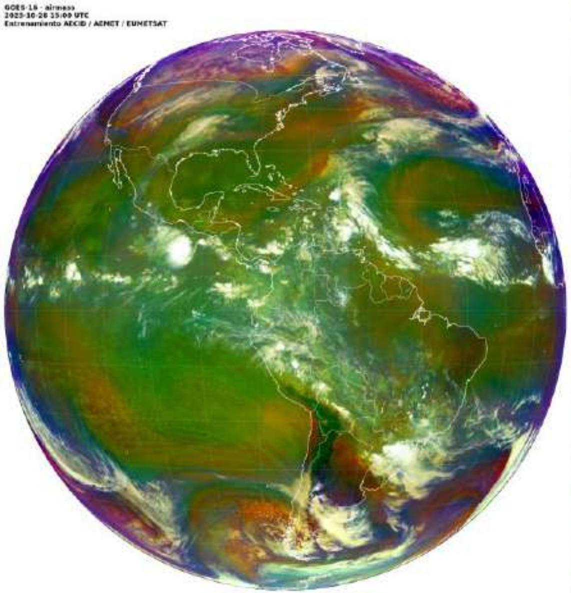
Más rápido



extent=[-90.0, -60.00, -30.00, 15.00]

area = 'full_disk'

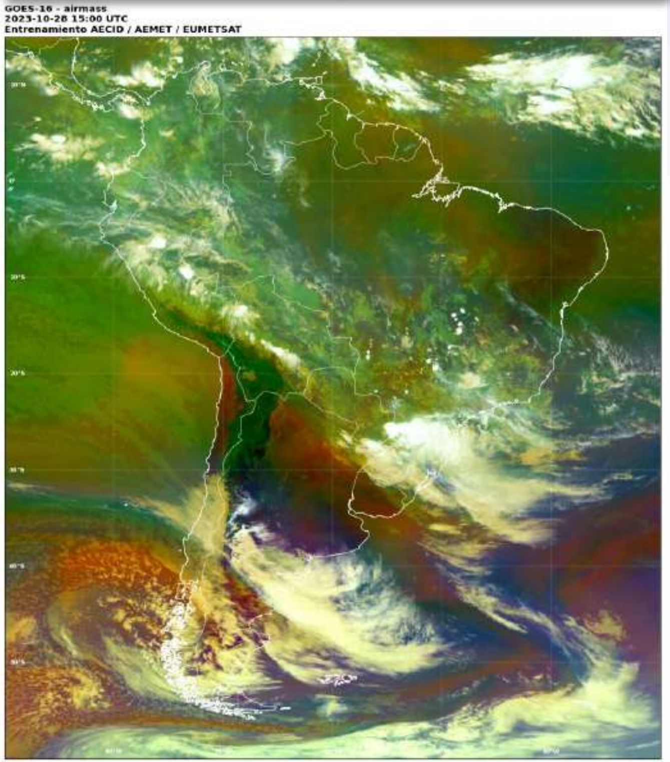
Tiempo intermedio



Datos desde 10 de julio de 2017,
cada 15 o 10 minutos

area = 'custom' (recorte en la proyección equidistante)

Tarda un poco más



Ejercicio: Huracanes Katia, Irma y José

Satélite:
'goes16’

Fecha:
2017-09-08 17:45 UTC

Región (Sudamerica):
area = 'custom'
extent = [-110.0, 0.00, -50.00, 40.00]

Compuestos RGB:
'true_color'
'cloud_phase_distinction'
'convection’

Referencia:
https://cimss.ssec.wisc.edu/satellite-blog/archives/25537

GOES-16 - true_color
2017-09-08 17:45 UTC
Entrenamiento AECID / AEMET / EUMETSAT

Ejercicio: Polvo del Sahara

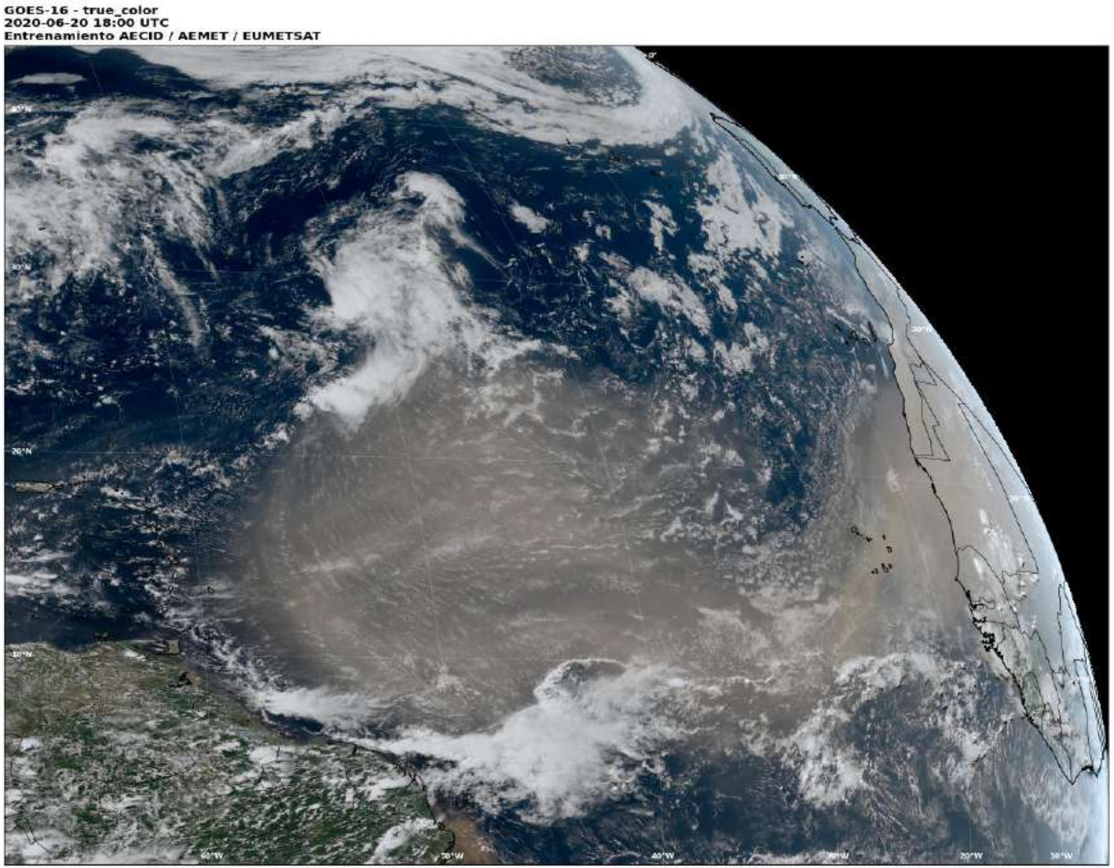
Satélite:
'goes16’

Fecha:
2020-06-20 18:00 UTC

Región:
area = 'custom_geo'
extent = [-65.0, 0.00, 0.00, 45.00]

Compuestos RGB:
'true_color’

Referencia:
https://www.severe-weather.eu/global-weather/massive-saharan-dust-cloud-atlantic-usa-fa/



Ejemplo 3: Datos GOES-Oeste

Ejemplo 3: Descargando Datos GOES-Oeste desde AWS y Graficando la Imagen

```
#-----------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example 3: Downloading GOES-
# Author: Diego Souza (INPE/CGCT/DISSM)
#-----------------------------------------------------------------------

plot_sat(sat='goes18',
         date_sat='2023-10-28 15:00',
         composite='airmass',
         area='full_disk',
         extent=[-90.0, -60.00, -30.00, 15.00],
         coast_color='white',
         countries_color='white',
         grid_color='white',
         figsize=[15,15])
```

Nombre de la función

Parámetros

Parámetros

- Satélite
- Fecha 'YYYY-MM-DD HH:MM'
- Compuesto deseado
- Área deseada
- Coordinadas
- Color de las líneas de costa
- Color de los mapas de países
- Color de las líneas de ref.
- Tamaño de la figura (pulgadas)

'goes16', 'goes17', 'msg0' o 'msg45'

Canales: 'C01, 'C02', 'C03', 'C04', 'C05', 'C06', 'C07', 'C08', 'C09', 'C10', 'C11', 'C12', 'C13', 'C14', 'C15', 'C16'

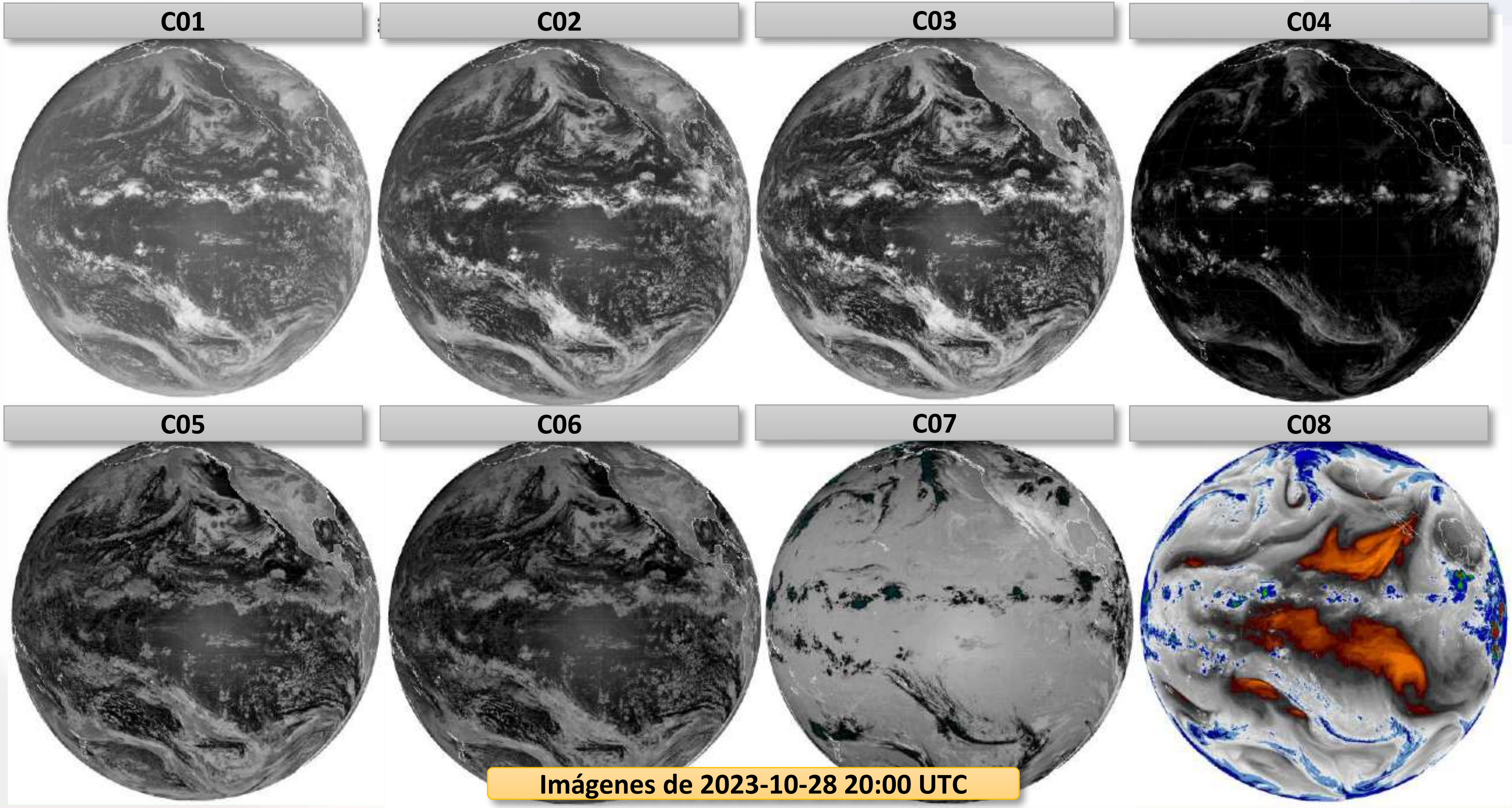
Composites: 'airmass', 'ash', 'cimss_cloud_type', 'cimss_true_color', 'cimss_true_color_sunz_rayleigh', 'cira_day_convection', 'cira_fire_temperature', 'cloud_phase', 'cloud_phase_distinction', 'cloudtop', 'color_infrared', 'colorized_ir_clouds', 'convection', 'day_microphysics', 'dust', 'fire_temperature_awips', 'fog', 'green_snow', 'land_cloud', 'natural_color', 'night_fog', 'night_microphysics', 'overview', 'rocket_plume_day', 'rocket_plume_night', 'snow', 'so2', 'tropical_airmass', 'true_color', 'water_vapors1', 'water_vapors2'

'full_disk', 'custom_geo', 'custom'

```
-----------------------------------------------------------------------
Downloading file /content/sample_data/OR_ABI-L2-MCMIPF-M6_G18_s20233011500224_e20233011509544_c20233011510011.nc

Plotting the file:
/content/sample_data/OR_ABI-L2-MCMIPF-M6_G18_s20233011500224_e20233011509544_c20233011510011.nc


Clipping input data to the valid range for imshow with RGB data ([0..1] for floats or [0..255] for integers).
Total processing time (satellite): 28.02 seconds.
-----------------------------------------------------------------------
'output/goes18_airmass_2023-10-28_15:00.png'
```

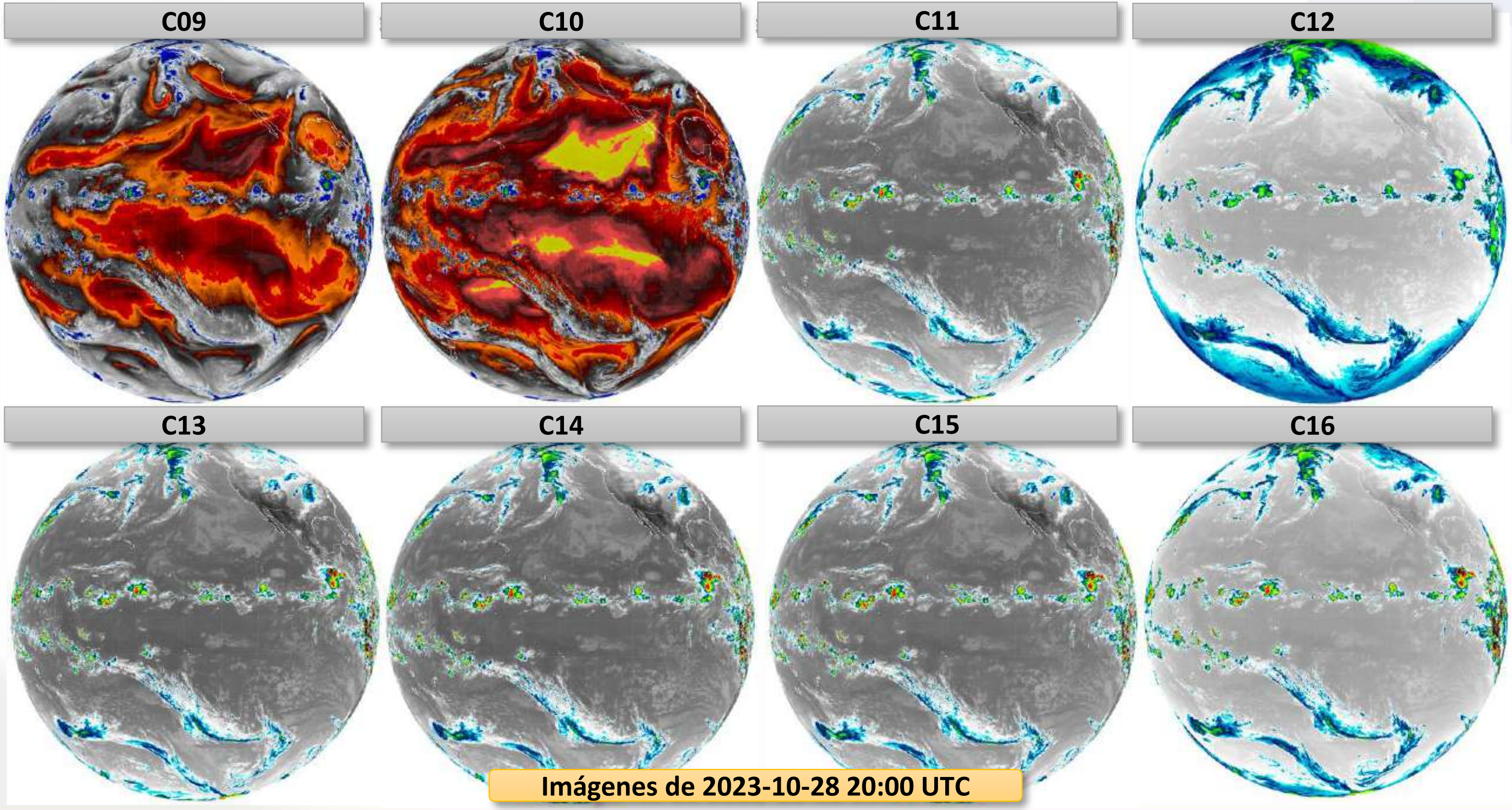
Salida con informaciones

Datos GOES-Oeste: Algunos "Composites" Disponibles



Imágenes de 2023-10-28 20:00 UTC

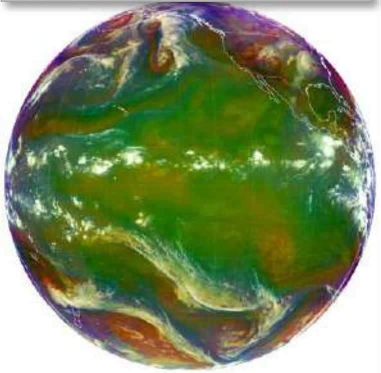
Datos GOES-Oeste: Algunos “Composites” Disponibles



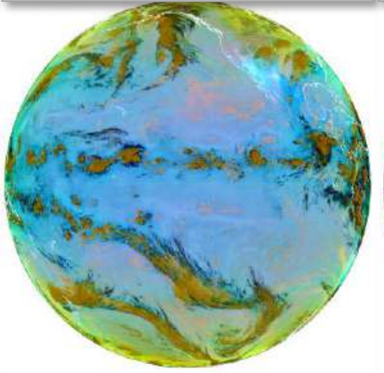
Imágenes de 2023-10-28 20:00 UTC

Datos GOES-Oeste: Algunos “Composites” Disponibles

airmass



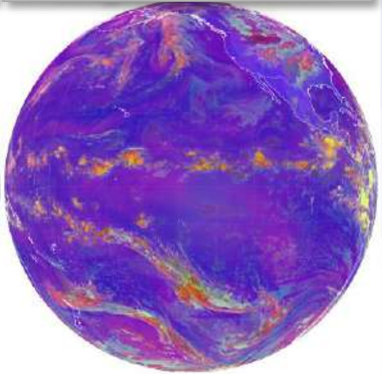
ash



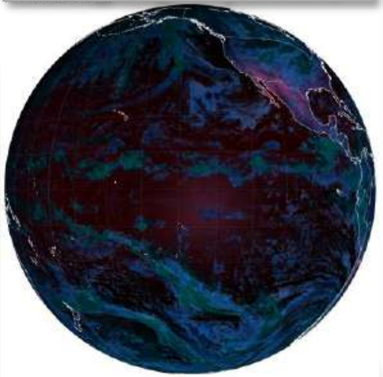
cimss_cloud_type

cimss_true_color

cira_day_convection

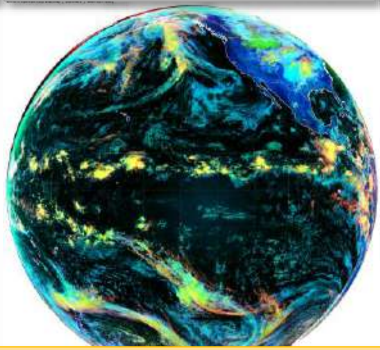


cira_fire_temperature

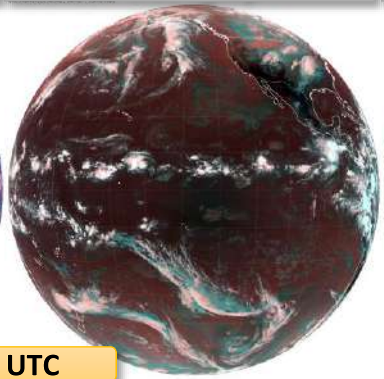


cloud_phase

cloud_phase_distinction



cloud_top

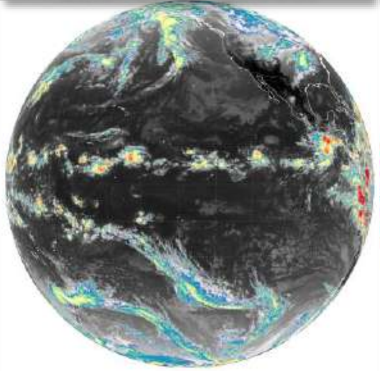


color_infrared

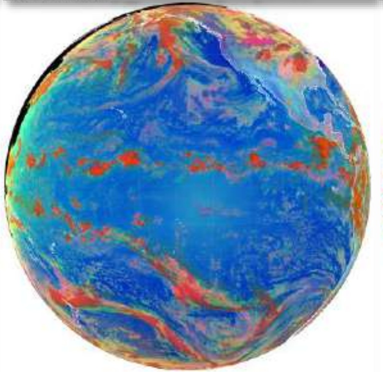
Imágenes de 2023-10-28 20:00 UTC

Datos GOES-Oeste: Algunos “Composites” Disponibles

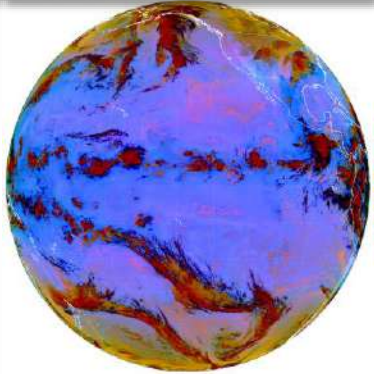
colorized_ir_clouds



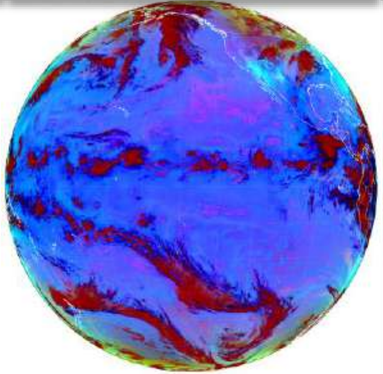
day_microphysics



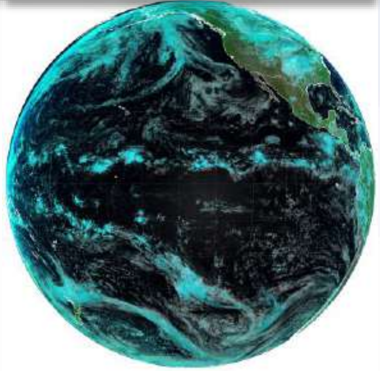
dust



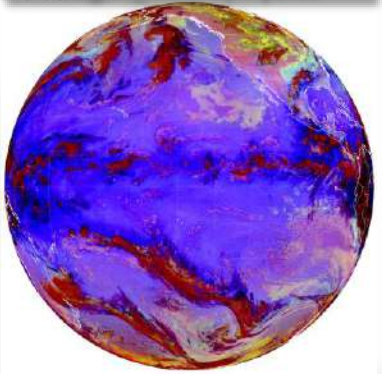
fog



land_cloud

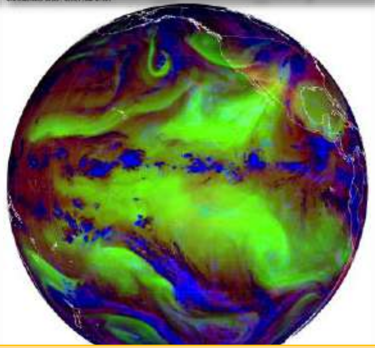


night_microphysics

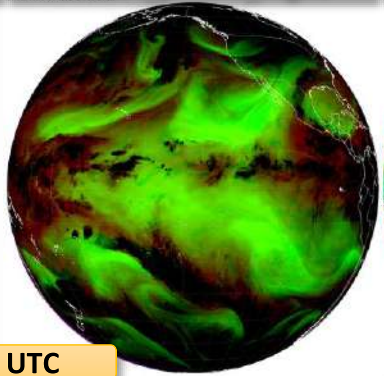


overview

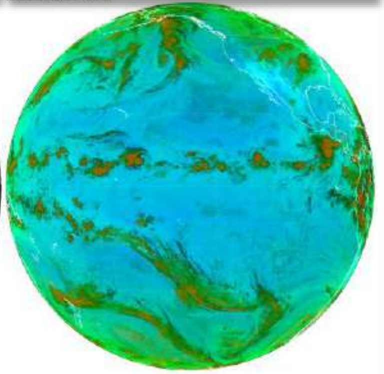
rocket_plume_day



rocket_plume_night



so2



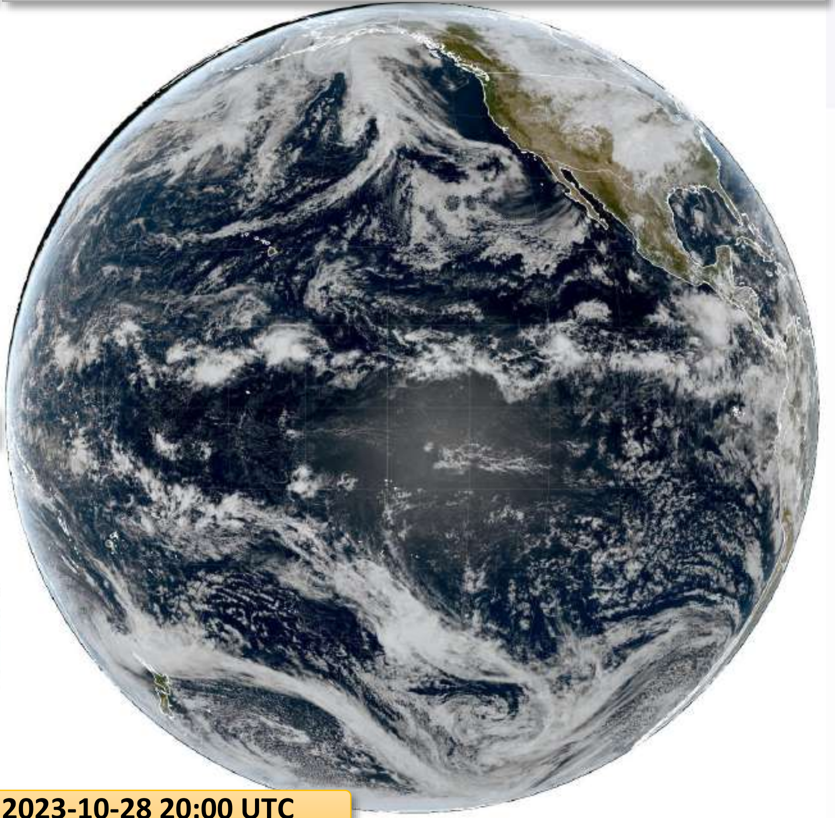
Imágenes de 2023-10-28 20:00 UTC

Datos GOES-Oeste: Algunos “Composites” Disponibles

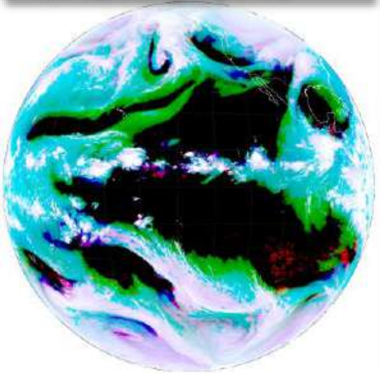
snow

tropical_airmass

true_color



water_vapors1

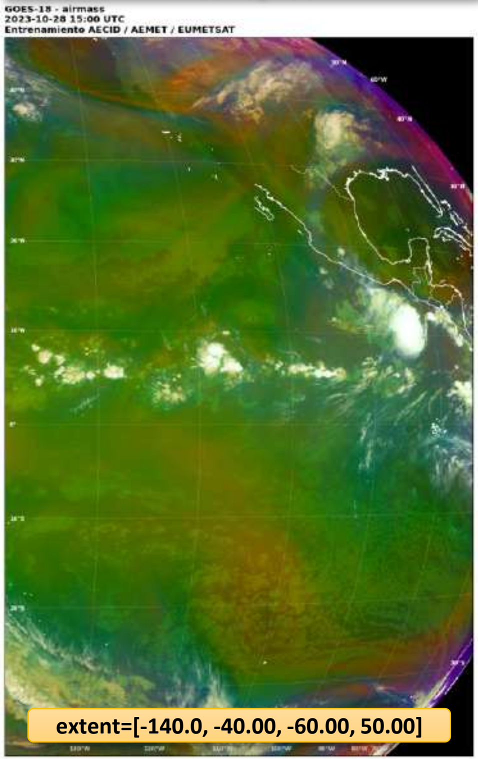


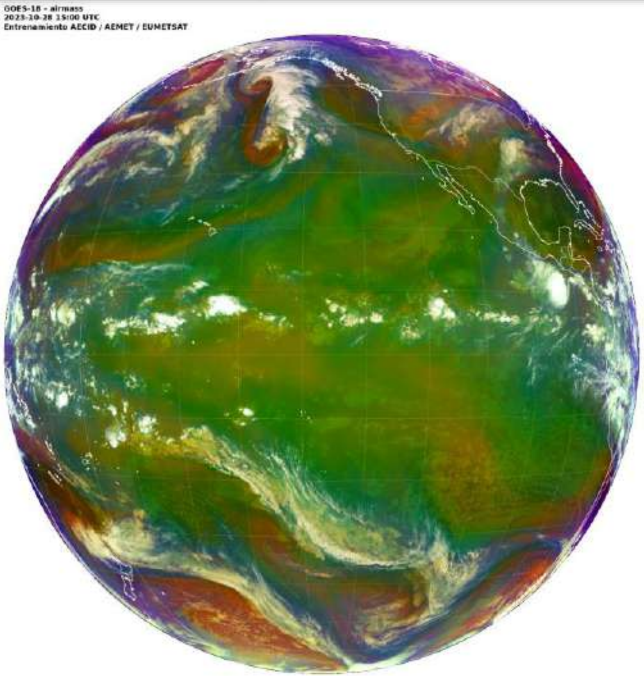
water_vapors2

Imágenes de 2023-10-28 20:00 UTC

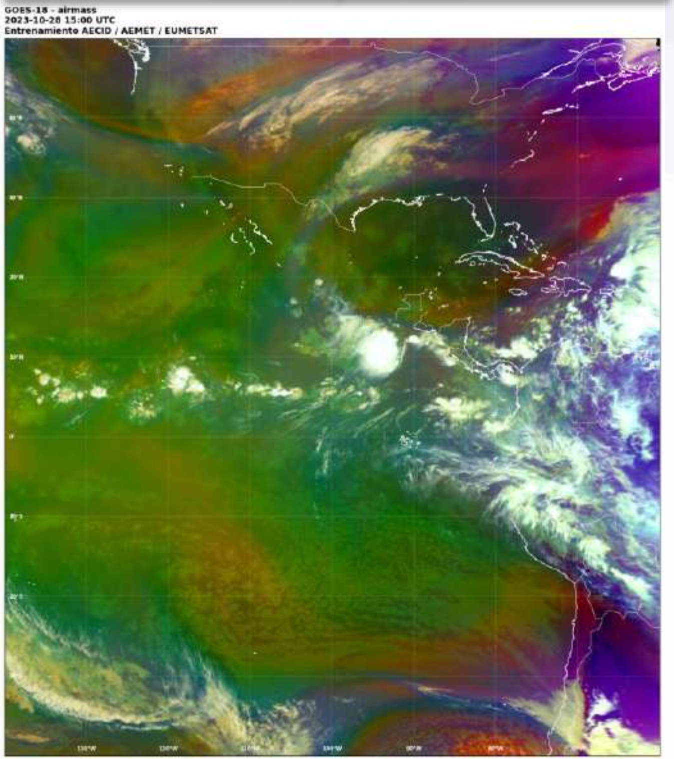
Ejemplo 3: Datos GOES-Oeste

area = 'custom_geo' (recorte en la proyección original)	area = 'full_disk'	area = 'custom' (recorte en la proyección equidistante)
Más rápido	Tiempo intermedio	Tarda un poco más





Desde 28 de agosto de 2018 (G17)
Desde 02 de agosto de 2022 (G18)



Ejercicio: Erupción de Klyuchevskaya en la Península de Kamchatka

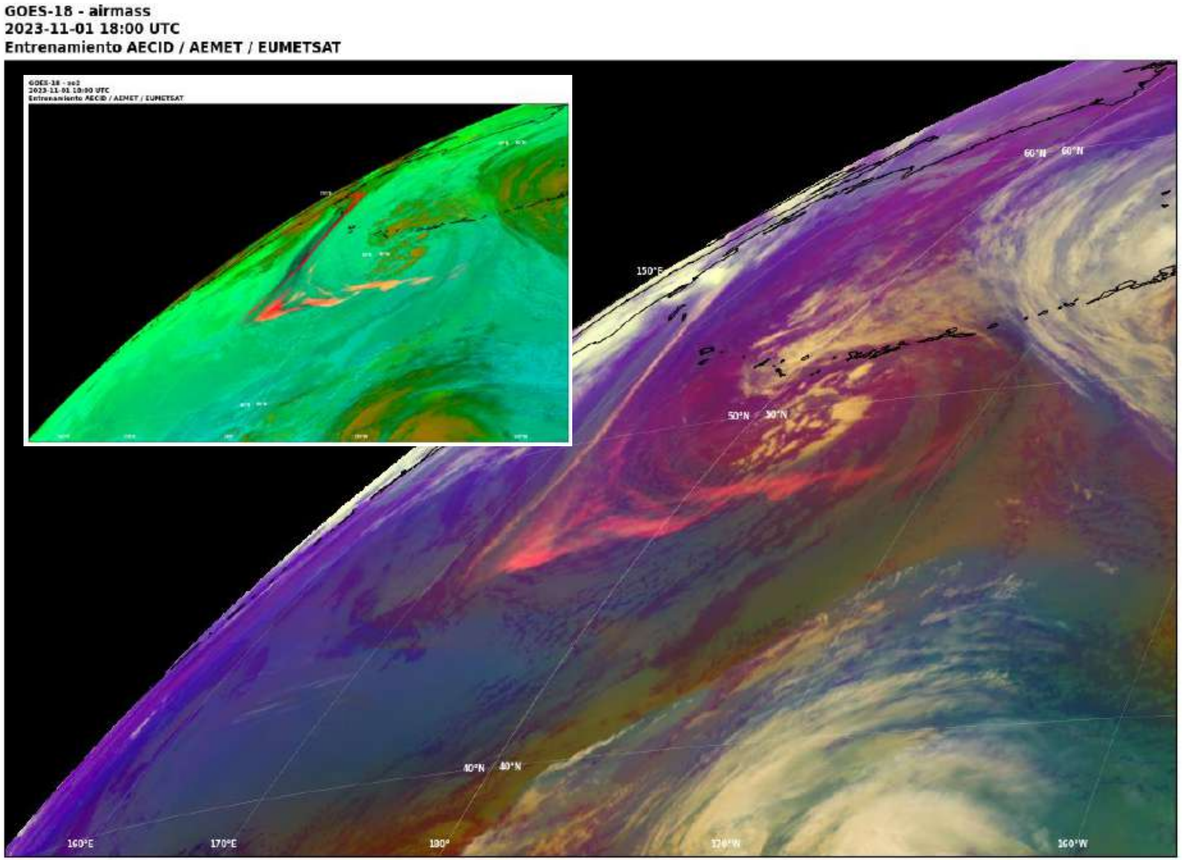
Satélite:
'goes18’

Fecha:
2023-11-01 18:00 UTC

Región:
area = 'custom_geo'
extent = [145.0, 40.00, 178.00, 65.00]

Compuestos:
'airmass'
'so2'

Referencia:
https://cimss.ssec.wisc.edu/satellite-blog/archives/55417



Ejemplo 4: Datos MSG-0°

Ejemplo 4: Descargando Datos MSG 0° con EUMDAC y Graficando la Imagen

```
#------------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example 4: Downloading MSG 0° Data with E
#      So
#------------------------------------------------------------------------
#   wi                                                every 15 minutes)

#      po                                              been tested):
# 'VIS006', 'VIS008', 'IR_016', 'IR_039', 'WV_062', 'WV_073', 'IR_087', 'IR_097', 'IR_108', '
# 'airmass', 'ash', 'cloudtop', 'cloudtop_daytime', 'colorized_ir_clouds', 'convection', 'day   reen_snow'
# 'natural_color', 'natural_color_nocorr', 'natural_color_raw', 'natural_color_raw_with_night_ir', 'natural_color_with_night_ir', 'natural_enh', 'natural_with_nig
# 'night_ir_alpha', 'night_microphysics', 'overview', 'overview_raw', 'rocket_plume_night', '

plot_sat(sat='msg0',
         date_sat='2023-10-28 15:00',
         composite='airmass',
         area='full_disk',
         extent=[-90.0, -60.00, -30.00, 15.00],
         coast_color='white',
         countries_color='white',
         grid_color='white',
         figsize=[15,15])
```

Nombre de la función

Parámetros

Parámetros

- Satélite
- Fecha 'YYYY-MM-DD HH:MM'
- Compuesto deseado
- Área deseada
- Coordinadas
- Color de las líneas de costa
- Color de los mapas de países
- Color de las líneas de ref.
- Tamaño de la figura (pulgadas)

'msg0', 'msg45'

Canales: 'VIS006', 'VIS008', 'IR_016', 'IR_039', 'WV_062', 'WV_073', 'IR_087', 'IR_097', 'IR_108', 'IR_120', 'IR_134'

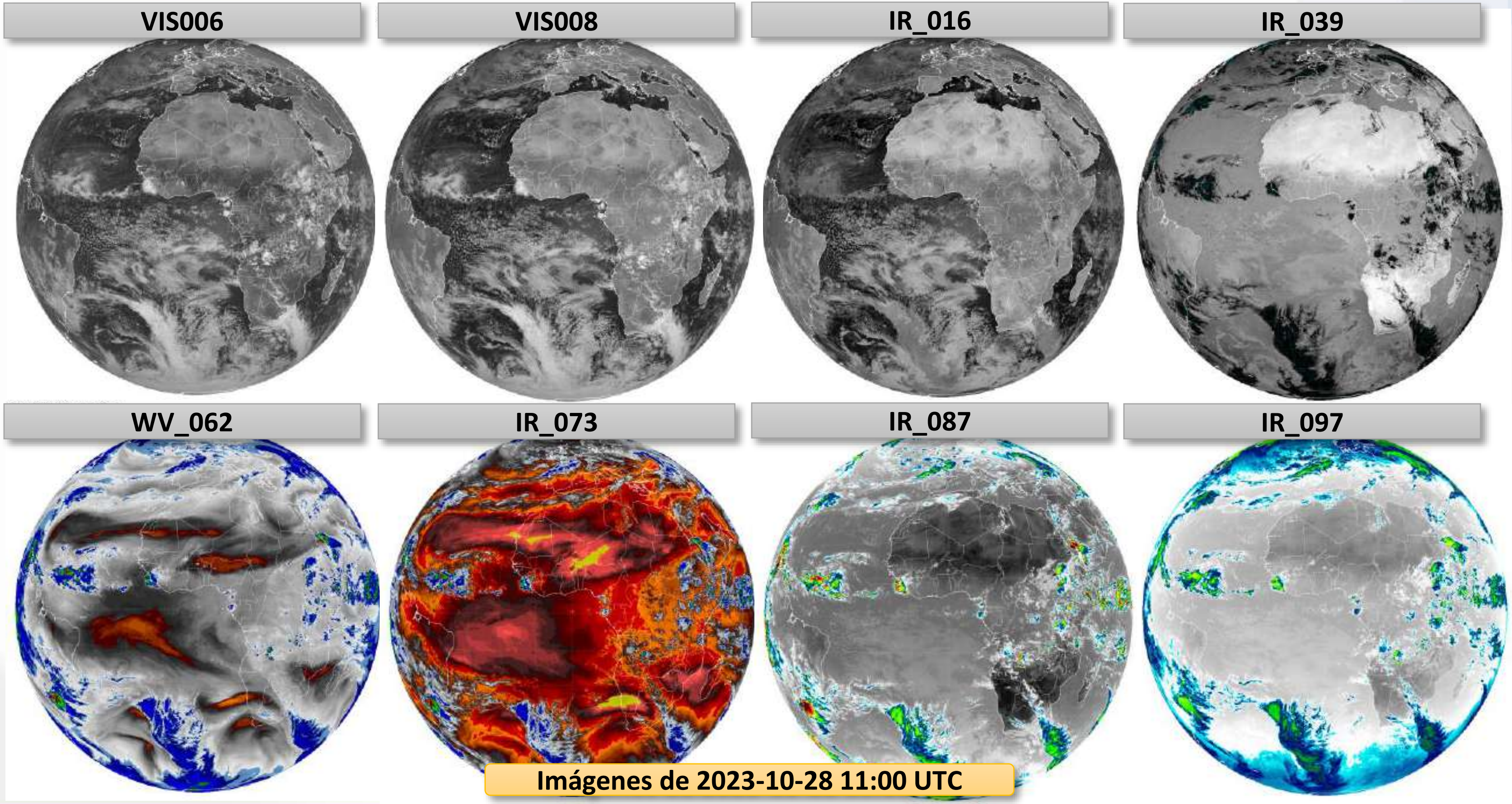
Composites: 'airmass', 'ash', 'cloudtop', 'cloudtop_daytime', 'colorized_ir_clouds', 'convection', 'day_microphysics' 'day_microphysics_winter', 'dust', 'fog', 'green_snow', 'ir108_3d', 'ir_overview', 'natural_color', 'natural_color_nocorr', 'natural_color_raw', 'natural_color_raw_with_night_ir', 'natural_color_with_night_ir', 'natural_enh', 'natural_with_night_fog', 'night_fog', 'night_ir_alpha', 'night_microphysics', 'overview', 'overview_raw', 'rocket_plume_night', 'snow'

'full_disk', 'custom_geo', 'custom'

```
------------------------------------------------------------------------
Checking the EUMDAC Access Token...
This Token 082cc401-57e8-3b07-bead-0888544c708d expires 2023-10-31 19:12:41.772760

Checking if the file already exists: /content/sample_data/MSG3-SEVI-MSG15-0100-NA-20231028151241.812000000Z-NA.zip
This file doesn't exist.

Downloading the following file: /content/sample_data/MSG3-SEVI-MSG15-0100-NA-20231028151241.812000000Z-NA.zip
Download finished.

The file has been uncompressed.

Plotting the file: /content/sample_data/MSG3-SEVI-MSG15-0100-NA-20231028151241.812000000Z-NA.nat

Clipping input data to the valid range for imshow with RGB data ([0..1] for floats or [0..255] for integers).
Total processing time (satellite): 158.14 seconds.
------------------------------------------------------------------------
'output/msg0_airmass_2023-10-28_15:00.png'
```

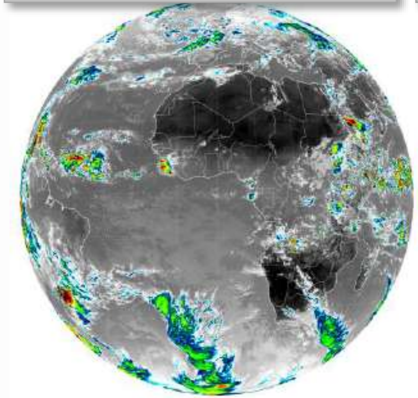
Salida con informaciones

Datos MSG-0°: Algunos "Composites" Disponibles

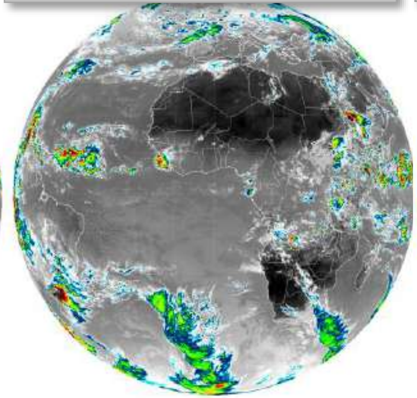


Datos MSG-0°: Algunos “Composites” Disponibles

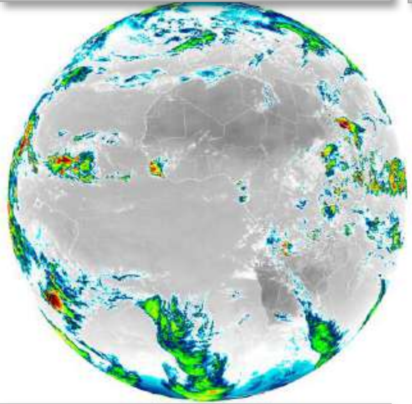
IR_108



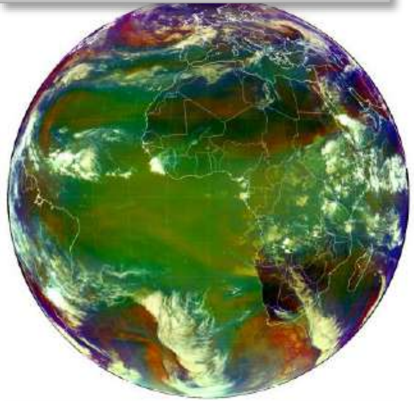
IR_120



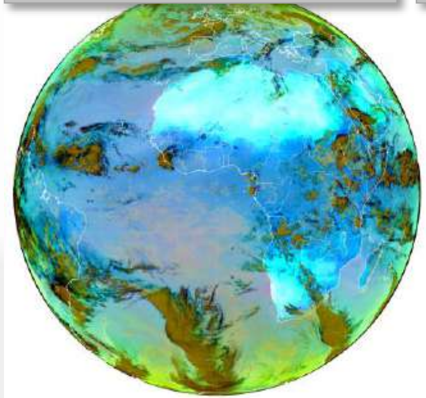
IR_134



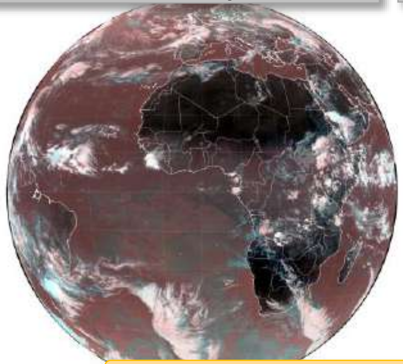
airmass



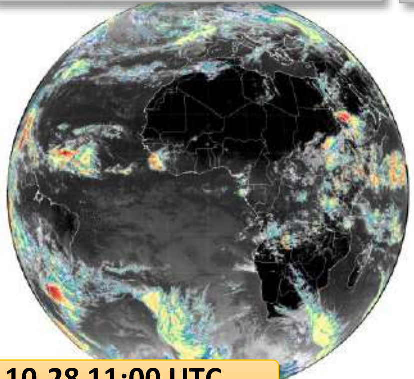
ash



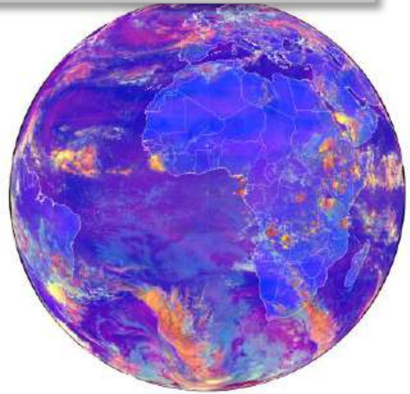
cloudtop



colorized_ir_clouds



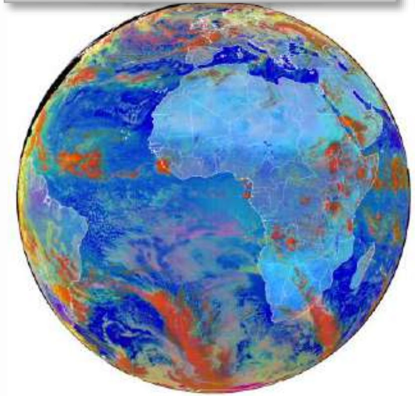
convection



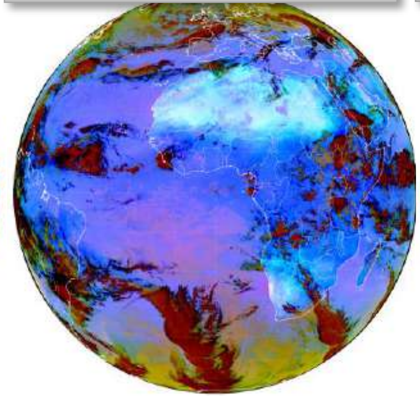
Imágenes de 2023-10-28 11:00 UTC

Datos MSG-0°: Algunos “Composites” Disponibles

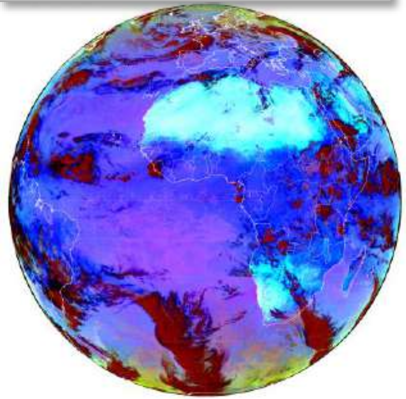
day_microphysics



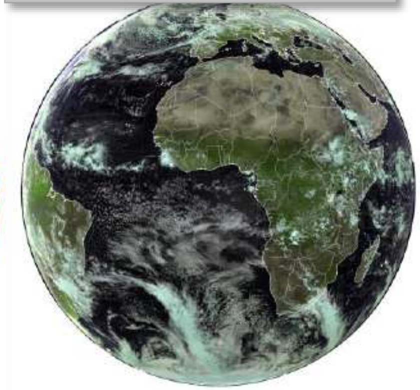
dust



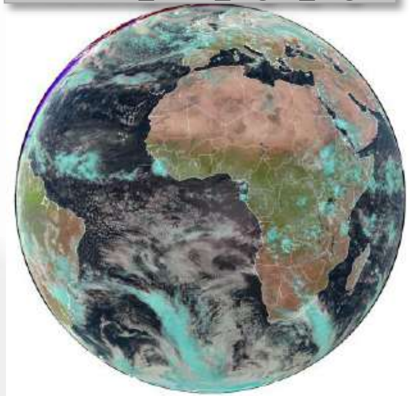
fog



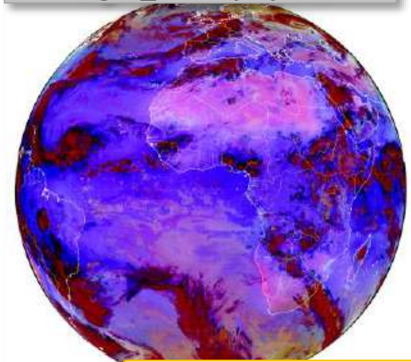
natural_enh



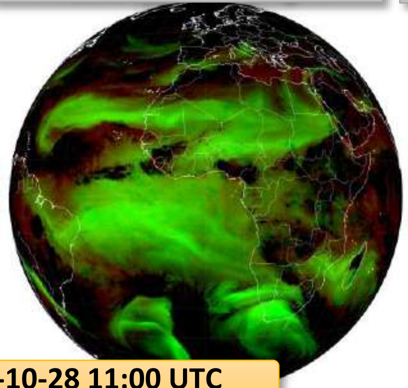
natural_with_night_fog



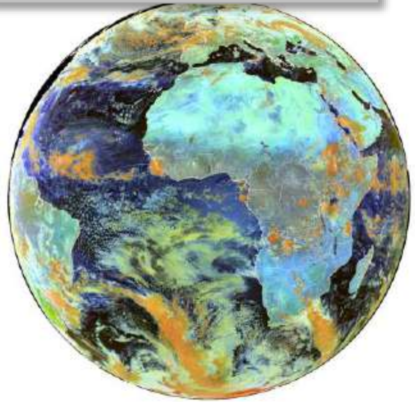
night_microphysics



rocket_plume_night



snow

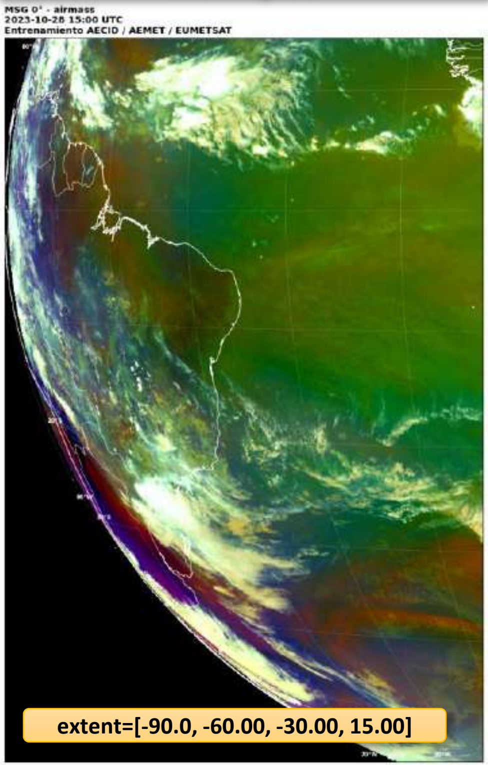


Imágenes de 2023-10-28 11:00 UTC

Ejemplo 4: Datos MSG-0°

area = 'custom_geo' (recorte en la proyección original)

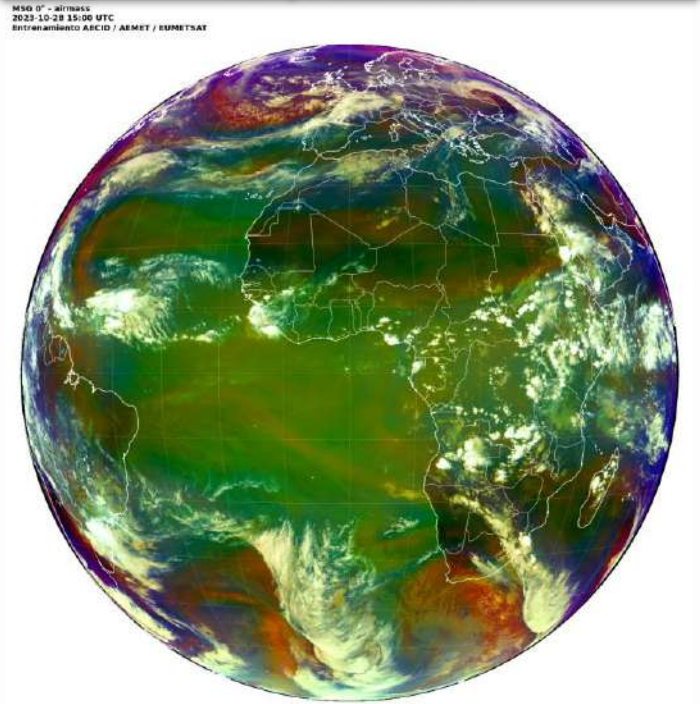
Más rápido



extent=[-90.0, -60.00, -30.00, 15.00]

area = 'full_disk'

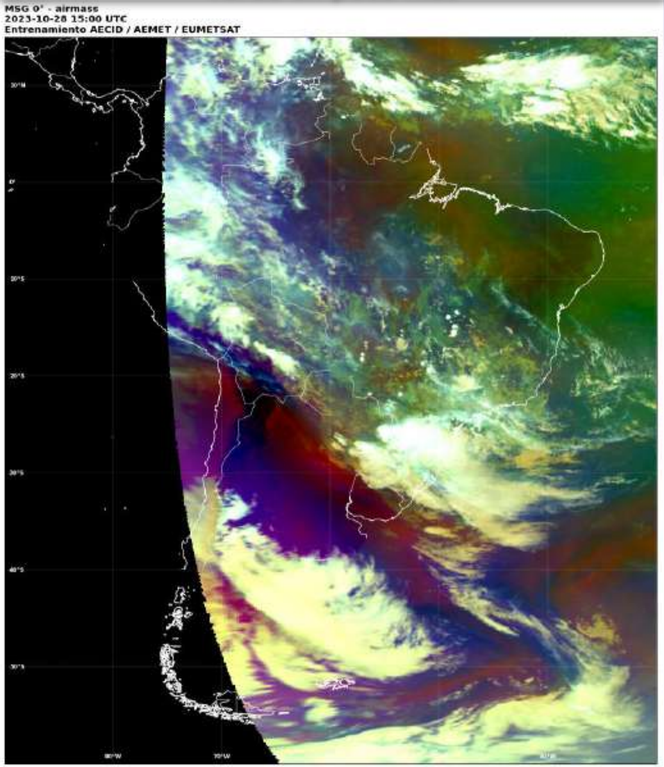
Tiempo intermedio



Datos desde 26 de Marzo de 2004

area = 'custom' (recorte en la proyección equidistante)

Tarda un poco más



Ejercicio: Huracán Catarina

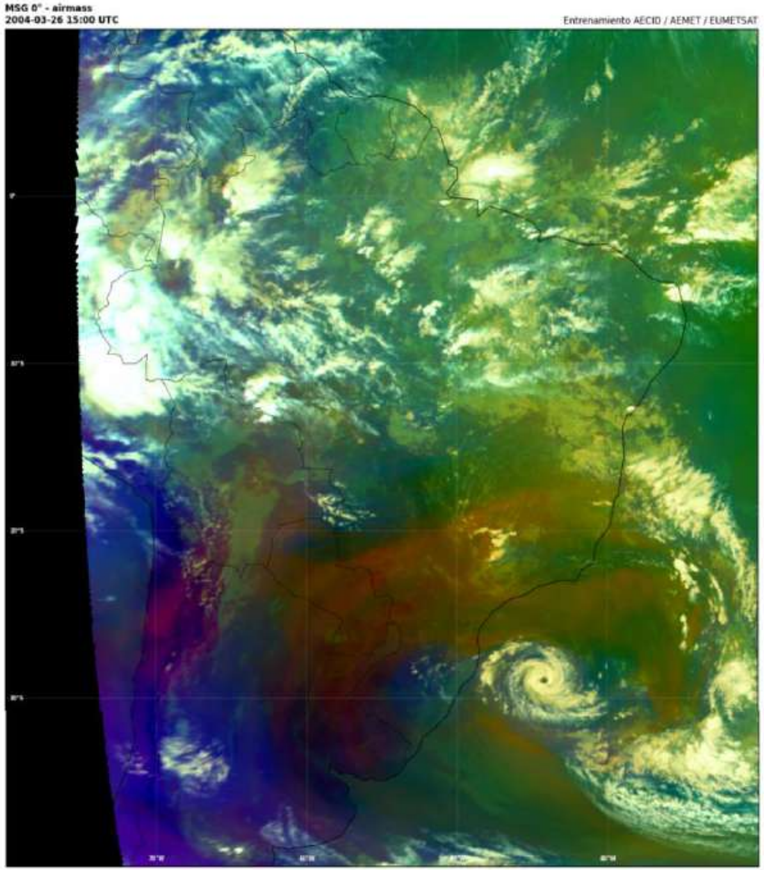
Satélite:
'msg0'

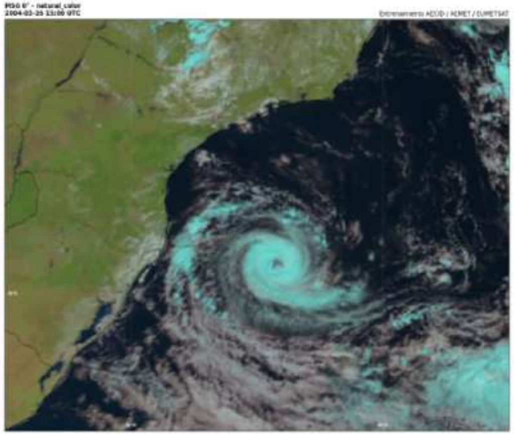
Fecha:
2004-03-26 15:00 UTC

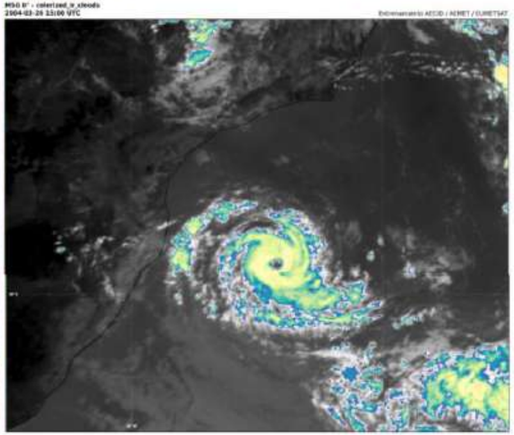
Región:
area = 'custom'
extent = [145.0, 40.00, 178.00, 65.00]

Compuestos:
'airmass'
'natural_color'
'colorized_ir_clouds'

Referencia:
https://en.wikipedia.org/wiki/Hurricane_Catarina







Ejercicio: Erupción Eyjafjallajökull

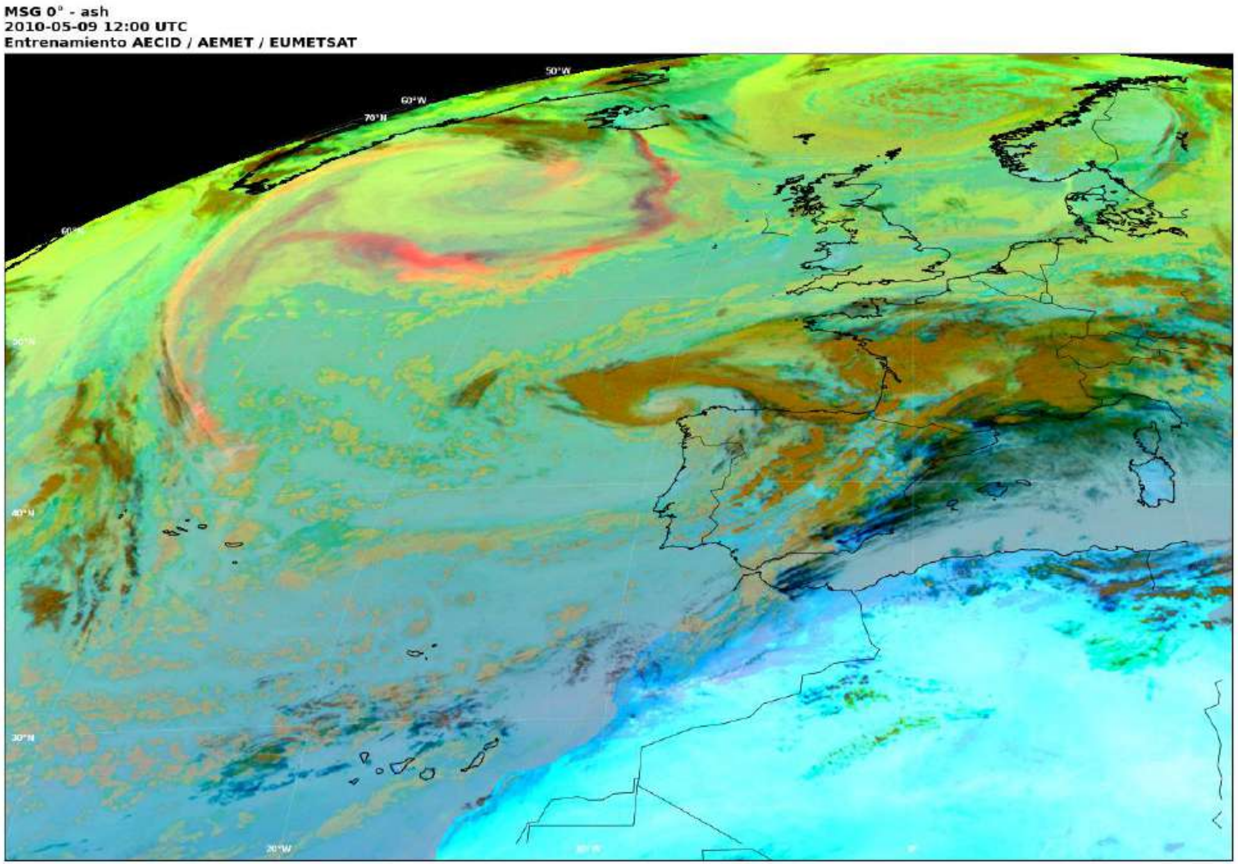
Satélite:
'msg0'

Fecha:
2010-05-09 12:00 UTC

Región:
area = 'custom_geo'
extent = [-30.0, 25.00, 10.00, 75.00]

Compuestos:
'ash'

Referencia:
https://cimss.ssec.wisc.edu/satellite-blog/archives/5212



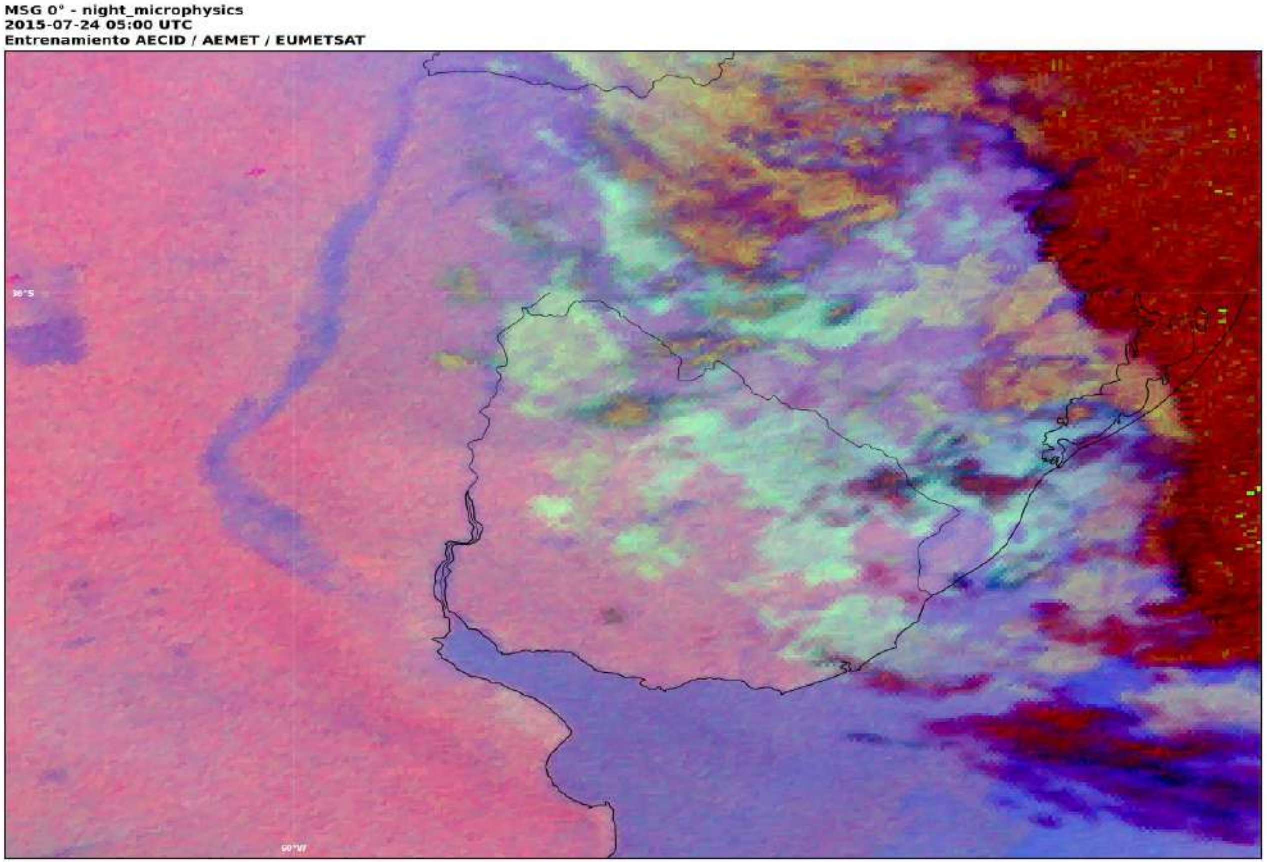
Ejercicio: Niebla en Uruguay

Satélite:
'msg0'

Fecha:
2015-07-24 05:00 UTC

Región:
area = 'custom'
extent = [-63.0, -37.0, -50.00, -27.00]

Compuesto:
'night_microphysics'



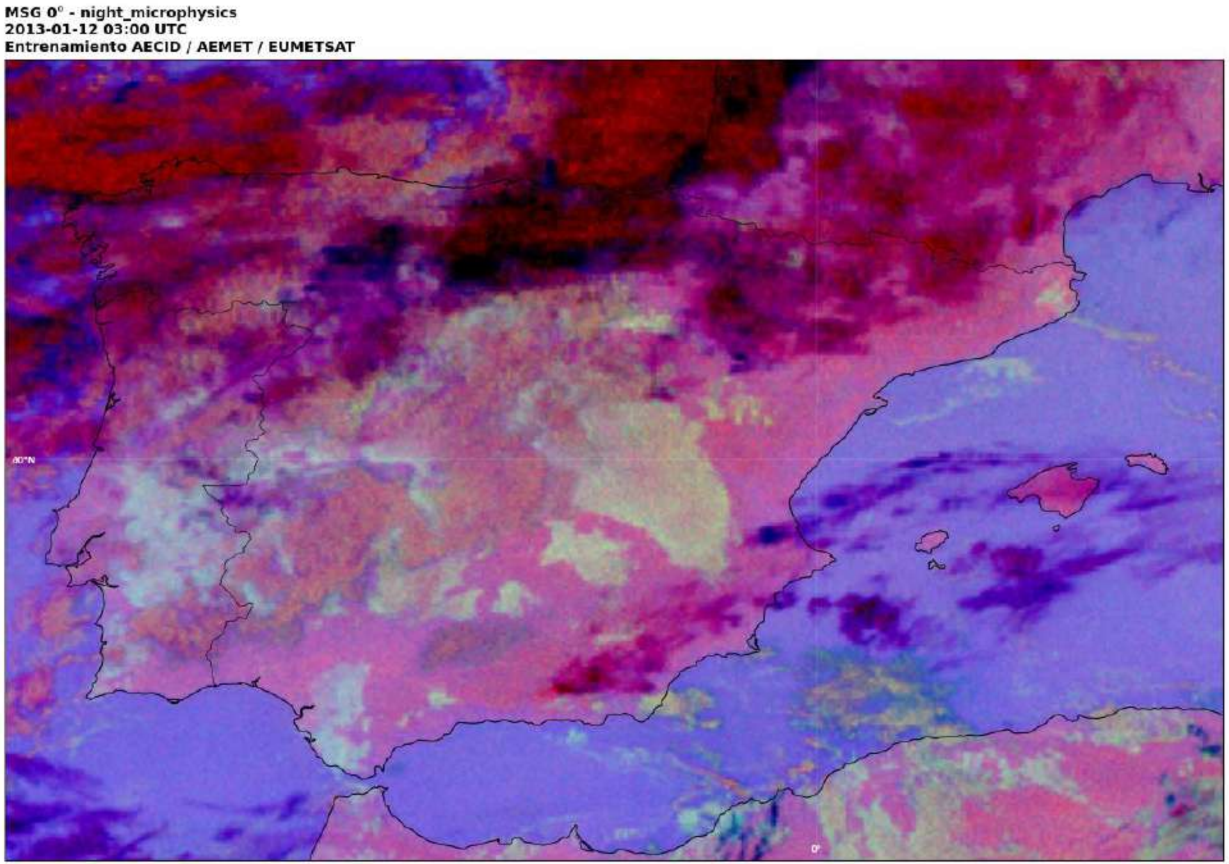
Ejercicio: Niebla en España

Satélite:
'msg0'

Fecha:
2013-01-12 03:00 UTC

Región:
area = 'custom'
extent = [-10.0, 35.0, 5.00, 45.00]

Compuesto:
'night_microphysics'



Ejemplo 5: Datos MSG-45.5°

Ejemplo 5: Descargando Datos MSG 45.5° (IODC) con EUMDAC y Graficando la Imagen

Nombre de la función

Parámetros

```
#-----------------------------------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example 5: Downloading MSG 4
# Author: Diego Souza (INPE/CGCT/DISSM)
#-----------------------------------------------------------------------------------------------

plot_sat(sat='msg45',
         date_sat='2023-10-28 15:00',
         composite='airmass',
         area='full_disk',
         extent=[55.0, 0.00, 105.00, 45.00], #extent=[-90.0, -60.00, -30.00, 15
         coast_color='white',
         countries_color='white',
         grid_color='white',
         figsize=[15,15])
```

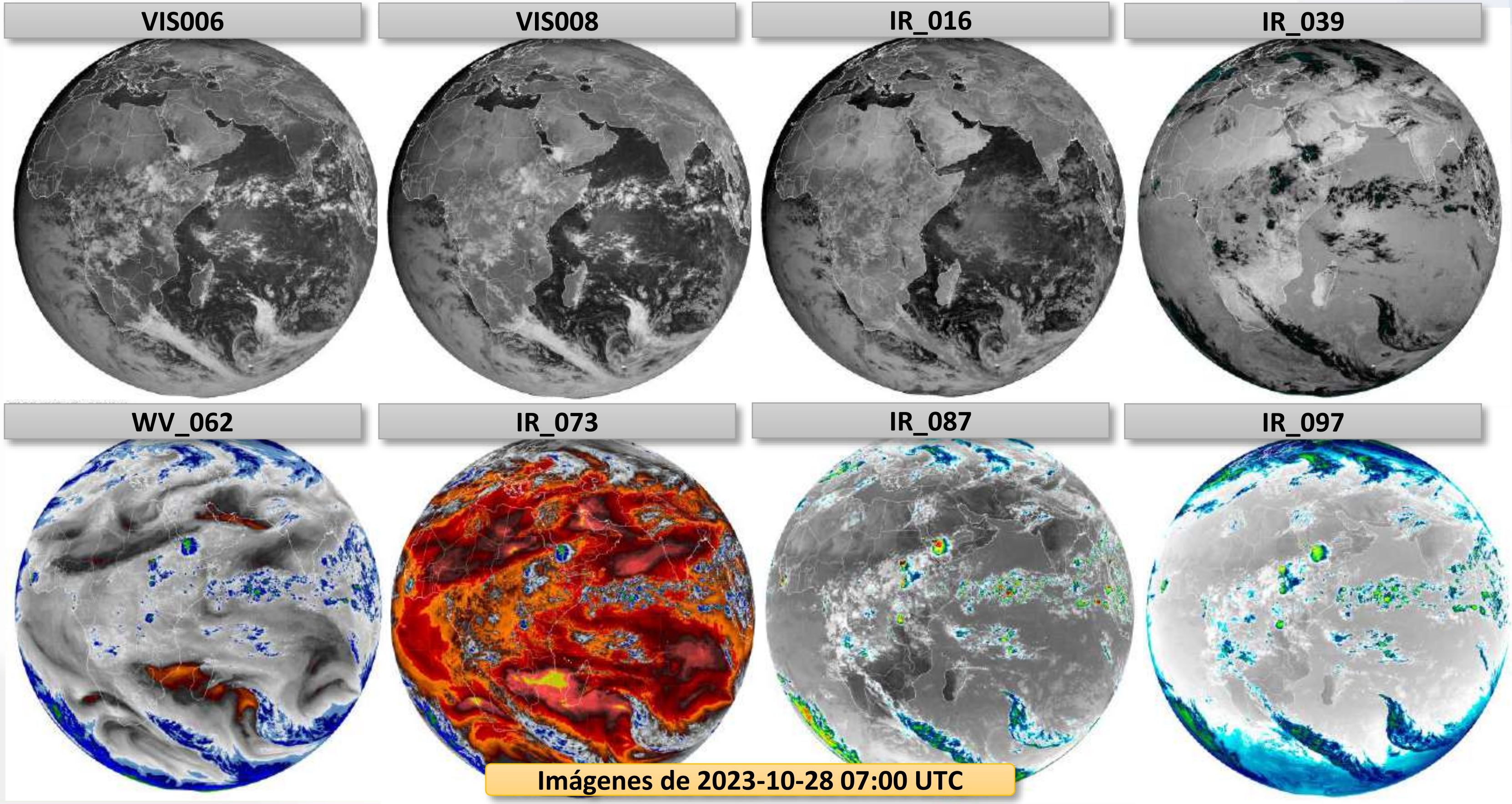
Parámetros

- Satélite: 'msg0', 'msg45'
- Fecha 'YYYY-MM-DD HH:MM'
- Compuesto deseado: Canales: 'VIS006', 'VIS008', 'IR_016', 'IR_039', 'WV_062', 'WV_073', 'IR_087', 'IR_097', 'IR_108', 'IR_120', 'IR_134'; Composites: 'airmass', 'ash', 'cloudtop', 'cloudtop_daytime', 'colorized_ir_clouds', 'convection', 'day_microphysics' 'day_microphysics_winter', 'dust', 'fog', 'green_snow', 'ir108_3d', 'ir_overview', 'natural_color', 'natural_color_nocorr', 'natural_color_raw', 'natural_color_raw_with_night_ir', 'natural_color_with_night_ir', 'natural_enh', 'natural_with_night_fog', 'night_fog', 'night_ir_alpha', 'night_microphysics', 'overview', 'overview_raw', 'rocket_plume_night', 'snow'
- Área deseada: 'full_disk', 'custom_geo', 'custom'
- Coordinadas
- Color de las líneas de costa
- Color de los mapas de países
- Color de las líneas de ref.
- Tamaño de la figura (pulgadas)

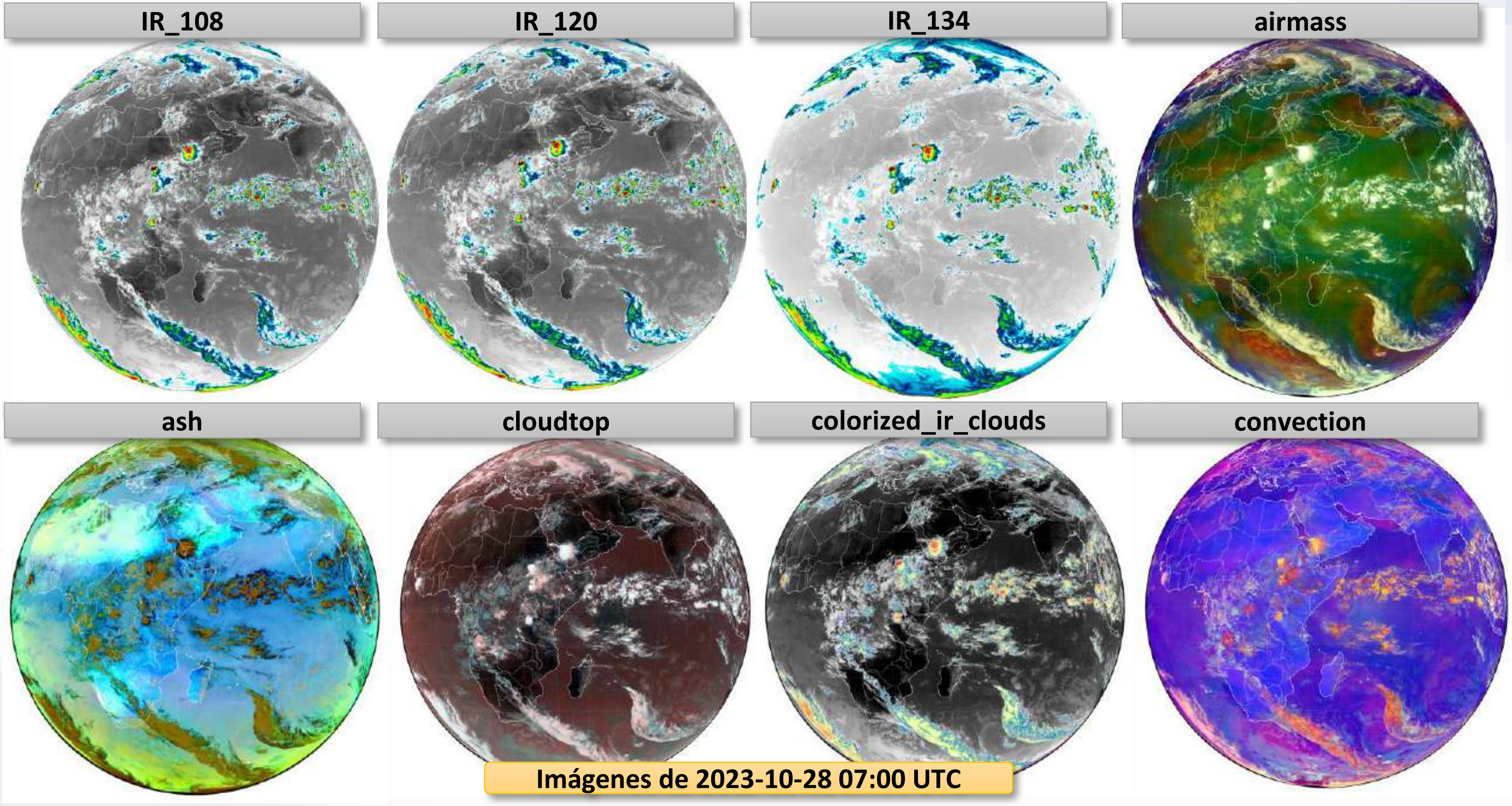
```
-----------------------------------------------------------------------------------------------
Checking the EUMDAC Access Token...
This Token 082cc401-57e8-3b07-bead-0888544c708d expires 2023-10-31 19:12:40.852459

Checking if the file already exists: /content/sample_data/MSG2-SEVI-MSG15-0100-NA-20231028151240.059000000Z-NA.zip
This file already exists.

Plotting the file: /content/sample_data/MSG2-SEVI-MSG15-0100-NA-20231028151240.059000000Z-NA.nat

Clipping input data to the valid range for imshow with RGB data ([0..1] for floats or [0..255] for integers).
Total processing time (satellite): 18.28 seconds.
-----------------------------------------------------------------------------------------------
'output/msg45_airmass_2023-10-28_15:00.png'
```

Salida con informaciones

Datos MSG-45.5°: Algunos "Composites" Disponibles

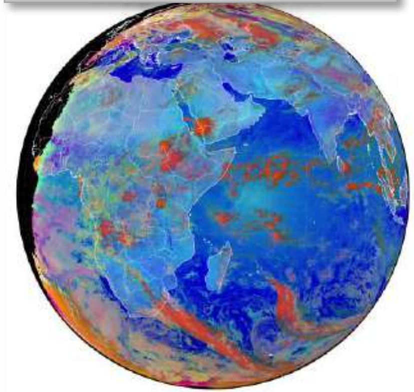


Datos MSG-45.5°: Algunos “Composites” Disponibles

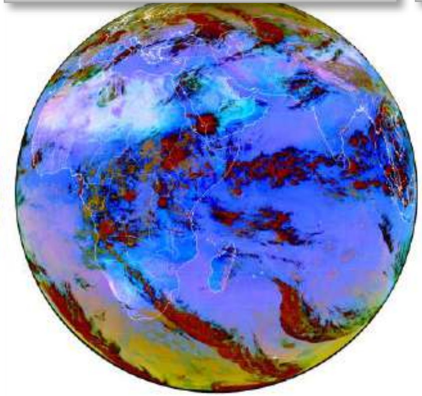


Datos MSG-45.5°: Algunos "Composites" Disponibles

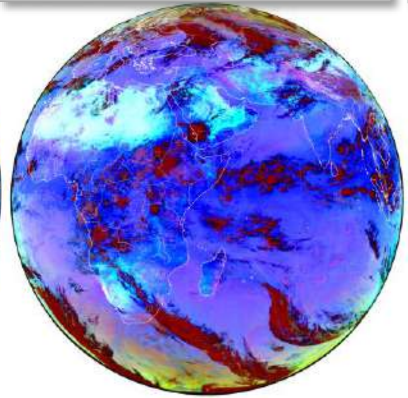
day_microphysics



dust

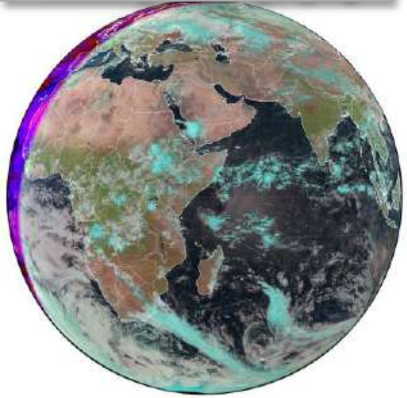


fog

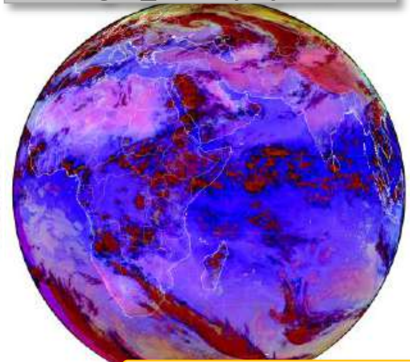


natural_enh

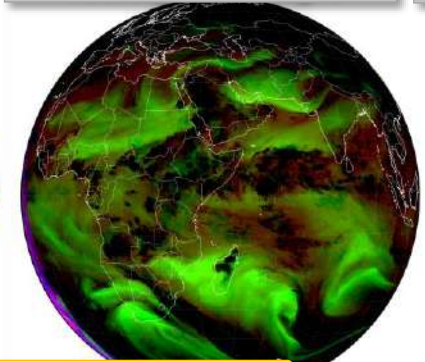
natural_with_night_fog



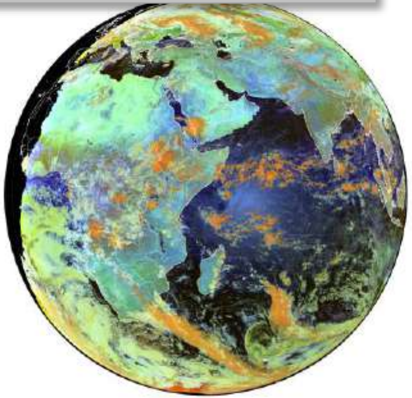
night_microphysics



rocket_plume_night



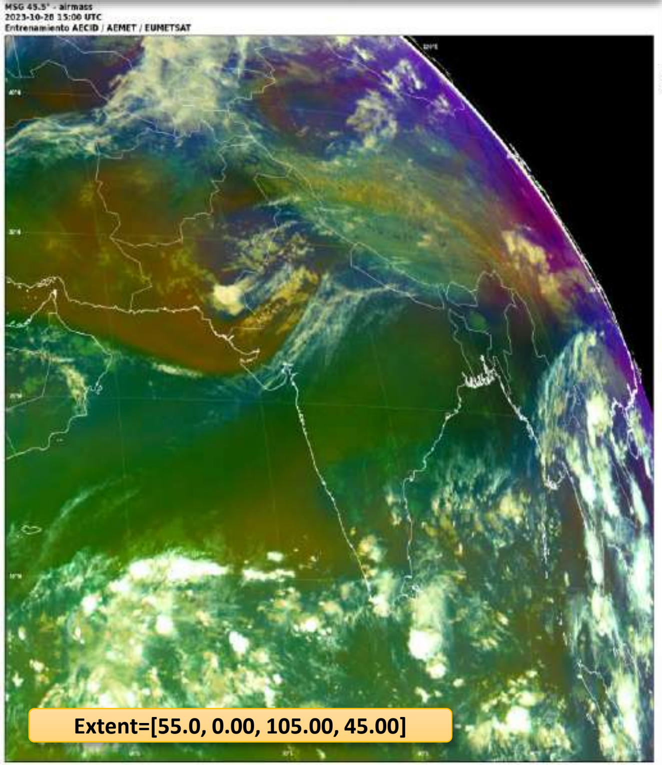
snow



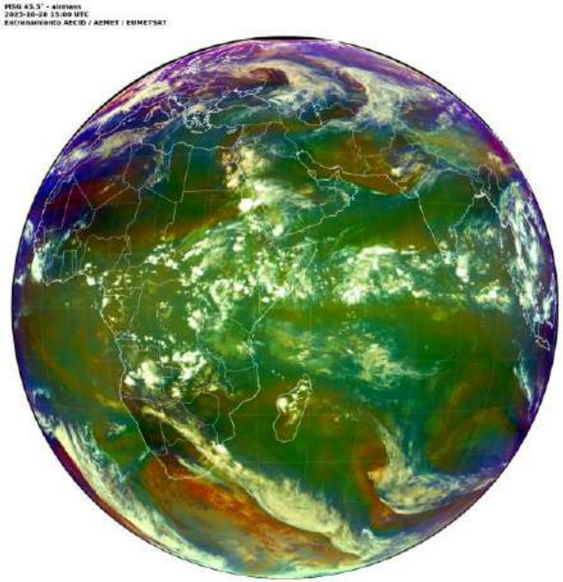
Imágenes de 2023-10-28 07:00 UTC

Ejemplo 5: Datos MSG-45.5°

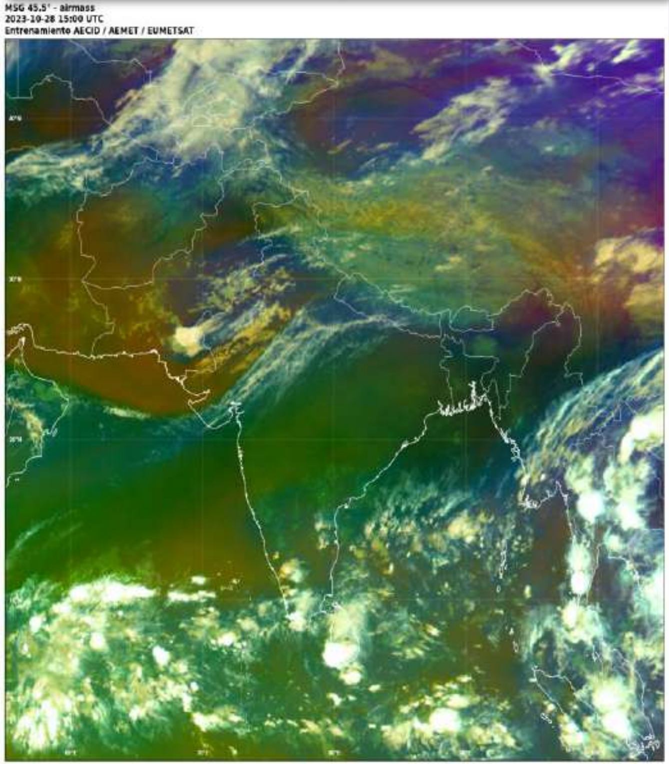
area = ‘custom_geo’ (recorte en la proyección original)	area = ‘full_disk’	area = ‘custom’ (recorte en la proyección equidistante)
Más rápido	Tiempo intermedio	Tarda un poco más



Extent=[55.0, 0.00, 105.00, 45.00]



Datos desde 26 de Marzo de 2004



Estudios de Caso de los Alumnos

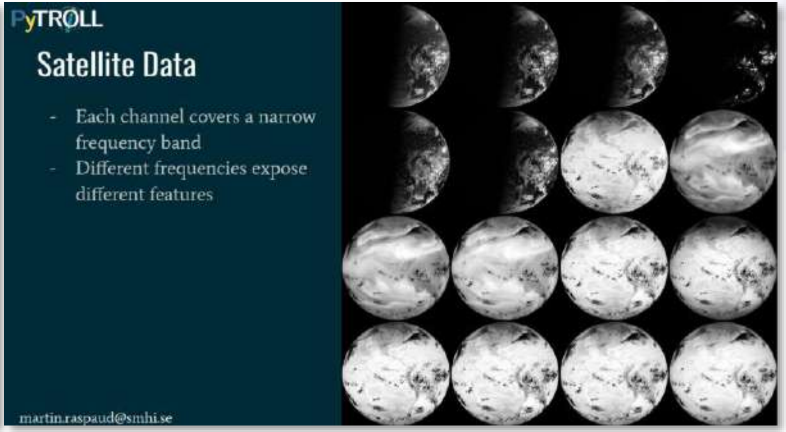
Elegir cualquier evento significativo en la región y generar imágenes (canales y RGBs)

Ejemplo 6: Manipulando Datos con Satpy

https://github.com/pytroll/satpy

Más informaciones en: [Link]()

Ejemplo 6: Manipulando Datos con Satpy



https://github.com/pytroll/satpy

Más informaciones en: Link

Ejemplo 6: Manipulando Datos con Satpy (“Readers”)

1. Verificando los “Readers” (lectores disponibles)

Ejemplo 6: ¿Cómo las Funciones Están Creando Estas Imágenes? Leyendo y Manipulando Datos Satelitales con Satpy (GOES-R)

Lectores disponibles en Satpy

Estos dos “readers” son usados en las funciones de demonstración de este notebook Colab

```
available_readers()
```

```
['abi_l1b', 'abi_l1b_scmi', 'abi_l2_nc', 'acspo', 'agri_fy4a_l1', 'agri_fy4b_l1', 'ahi_hrit', 'ahi_hsd',
'ahi_l1b_gridded_bin', 'ahi_l2_nc', 'ami_l1b', 'amsr2_l1b', 'amsr2_l2', 'amsr2_l2_gaasp', 'amsub_l1c_aapp',
'ascat_l2_soilmoisture_bufr', 'atms_l1b_nc', 'atms_sdr_hdf5', 'avhrr_l1b_aapp', 'avhrr_l1b_eps',
'avhrr_l1b_hrpt', 'avhrr_l1c_eum_gac_fdr_nc', 'cmsaf-claas2_l2_nc', 'electrol_hrit', 'epic_l1b_h5',
'fci_l1c_nc', 'fci_l2_nc', 'generic_image', 'geocat', 'gerb_l2_hr_h5', 'ghi_l1', 'ghrsst_l2', 'glm_l2', 'gms5-
vissr_l1b', 'goes-imager_hrit', 'goes-imager_nc', 'gpm_imerg', 'grib', 'hsaf_grib', 'hsaf_h5',
'hy2_scat_l2b_h5', 'iasi_l2', 'iasi_l2_cdr_nc', 'iasi_l2_so2_bufr', 'ici_l1b_nc', 'jami_hrit', 'li_l2_nc',
'maia', 'meris_nc_sen3', 'mersi2_l1b', 'mersi_ll_l1b', 'mhs_l1c_aapp', 'mimicTPW2_comp', 'mirs', 'msi_safe',
'msu_gsa_l1b', 'mtsat2-imager_hrit', 'mviri_l1b_fiduceo_nc', 'mwi_l1b_nc', 'mws_l1b_nc', 'nucaps', 'nwcsaf-geo',
'nwcsaf-msg2013-hdf5', 'nwcsaf-pps_nc', 'oceancolorcci_l3_nc', 'olci_l1b', 'olci_l2', 'omps_edr',
'safe_sar_l2_ocn', 'sar-c_safe', 'satpy_cf_nc', 'seviri_l1b_hrit', 'seviri_l1b_native', 'seviri_l1b_nc',
'seviri_l2_bufr', 'seviri_l2_grib', 'slstr_l1b', 'smos_l2_wind', 'tropomi_l2', 'vaisala_gld360', 'vii_l1b_nc',
'vii_l2_nc', 'viirs_compact', 'viirs_edr', 'viirs_edr_active_fires', 'viirs_l1b', 'viirs_sdr',
'viirs_vgac_l1c_nc', 'virr_l1b']
```

Ejemplo 6: Manipulando Datos con Satpy ("Scene")

2. Creando un objecto "Scene", especificando el dato y el lector

Descargando el dato GOES-R y creando una "Scene"

Función para descargar datos

Nombre del archivo

Reader (lector)

Satélite

Fecha

```
# download the satellite data
file_name = download_satellite(sat='goes16', date_sat='2023-10-28 15:00')

# initialise scene
scn = Scene(filenames=glob.glob(os.path.join('/content/sample_data/', file_name)), reader='abi_l2_nc')
```

"Scene" Satpy

```
----------------------------------------------------------------------------------------------------
File /content/sample_data/OR_ABI-L2-MCMIPF-M6_G16_s20233011500204_e20233011509525_c20233011509594.nc already exists
Total download time: 0.35 seconds.
----------------------------------------------------------------------------------------------------
```

Ejemplo 6: Manipulando Datos con Satpy (“Datasets”)

3. Visualizando los “Datasets” disponibles

```
# check the available composites
scn.available_dataset_names()
```

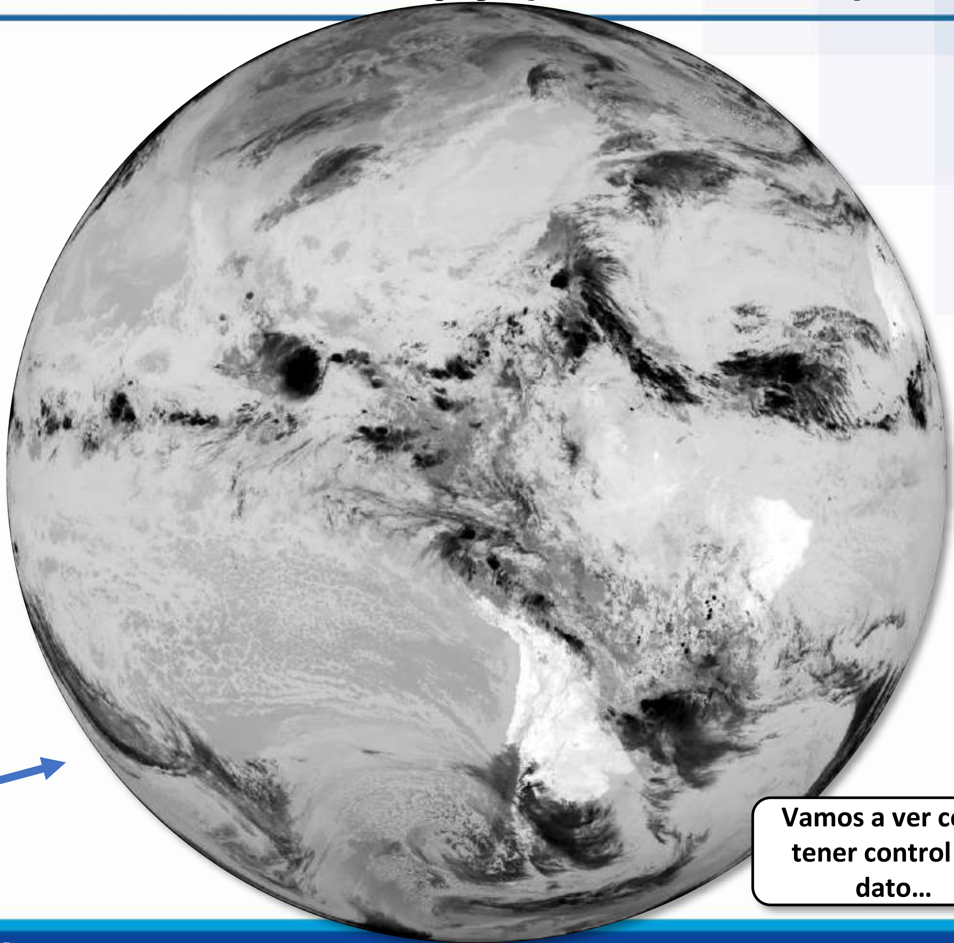
```
['C01',
 'C02',
 'C03',
 'C04',
 'C05',
 'C06',
 'C07',
 'C08',
 'C09',
 'C10',
 'C11',
 'C12',
 'C13',
 'C14',
 'C15',
 'C16']
```

16 canales ABI (o de cualquier otro sensor soportado por Satpy)

4. Cargando y visualizando un dataset

32 s

```
# basic plot
scn.load(['C13'])
scn.show('C13')
```



Vamos a ver como tener control del dato…

Ejemplo 6: Manipulando Datos con Satpy (Plot)

5. Pasando un dataset a una matriz de datos

```
# read the pixel values
data = scn['C13']

# plot size (width x height, in inches)
plt.figure(figsize=(10,10))

# plot the image
plt.imshow(data, vmin=190, vmax=320, cmap='jet')
```

Pasando los valores de pixel a una variable

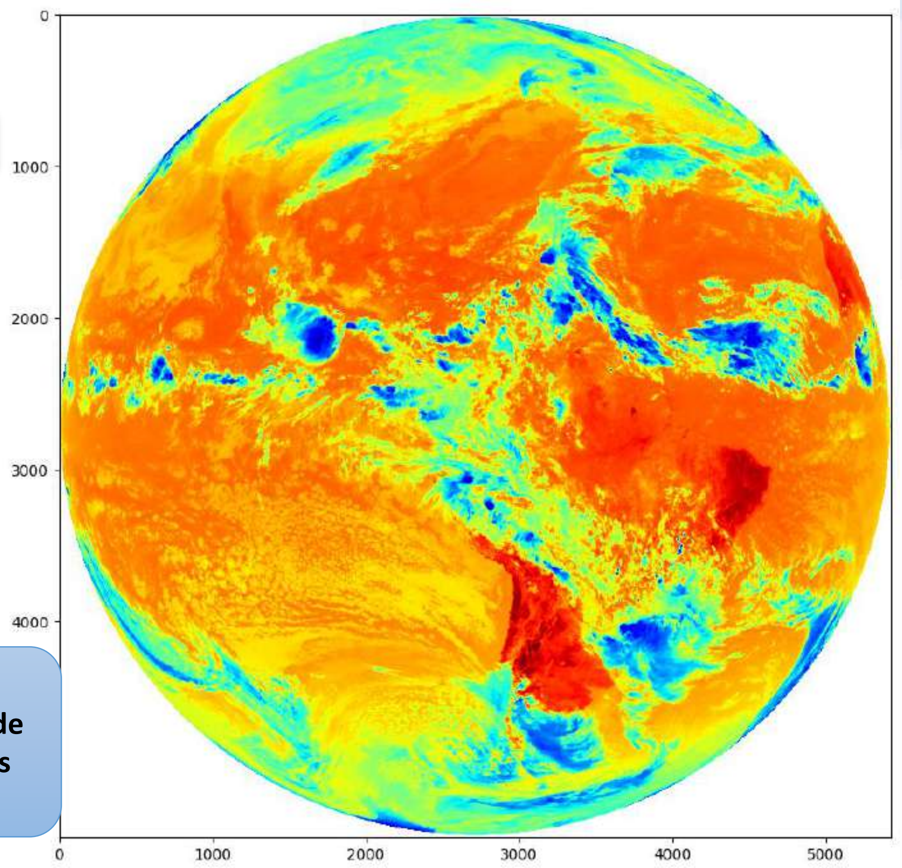
Tamaño del plot

Variable

Valor mínimo de la escala de colores

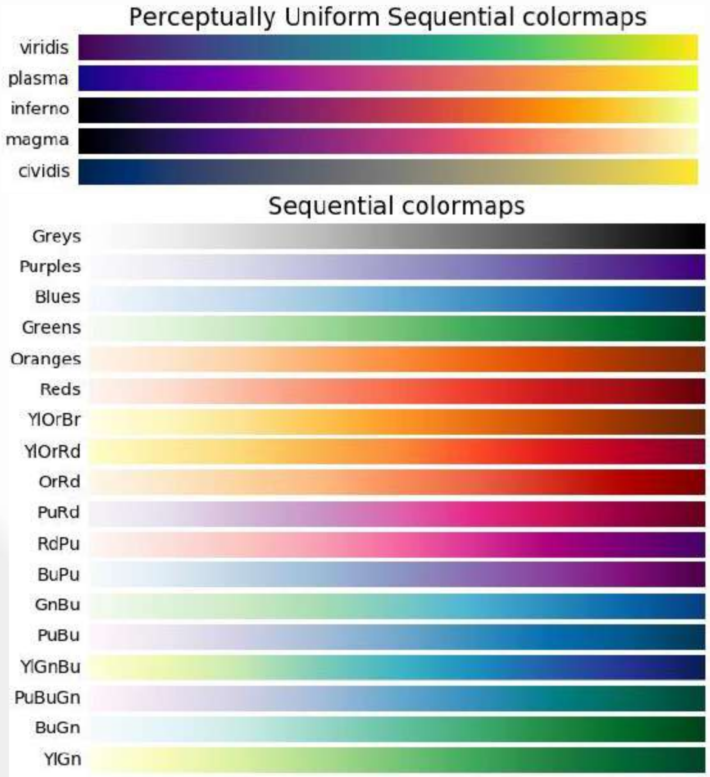
Valor máximo de la escala de colores

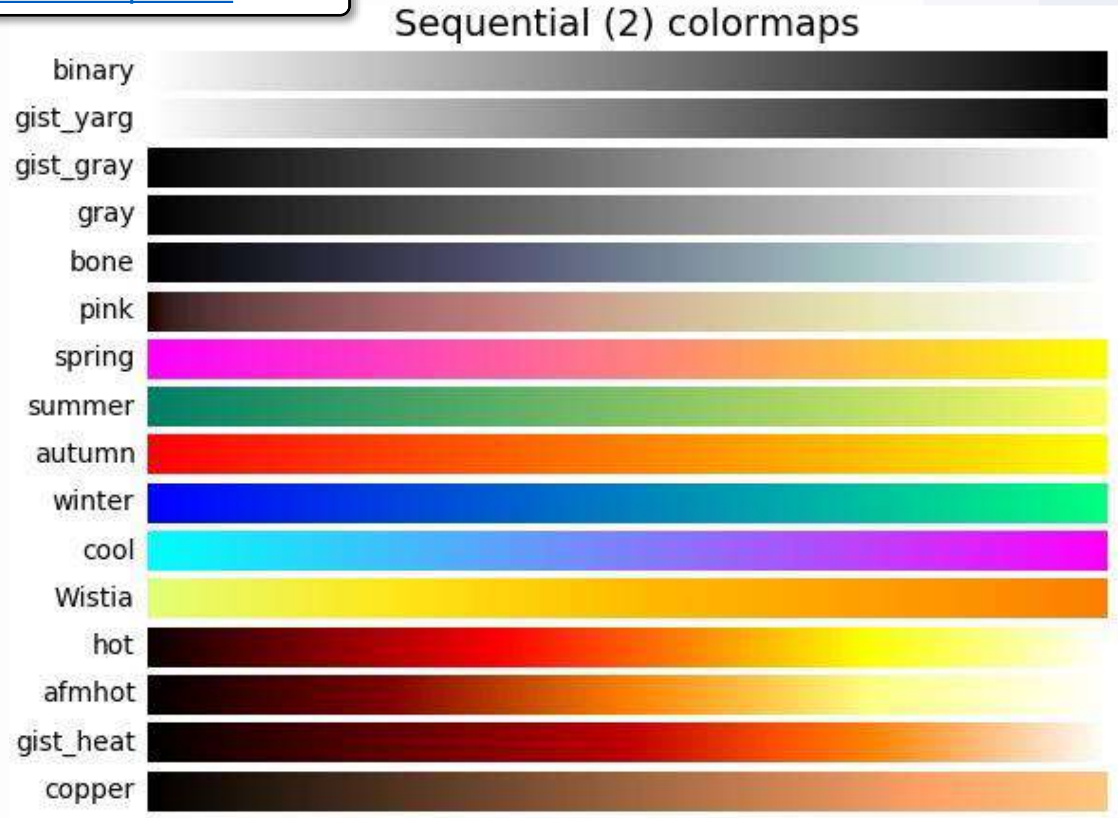
Escala de colores



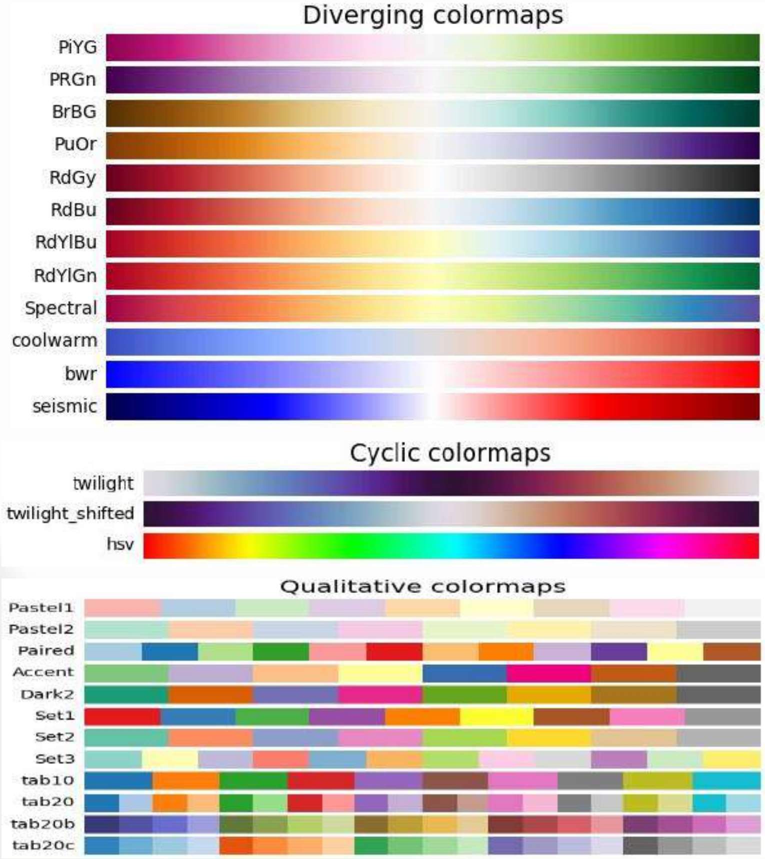
Ejemplo 6: Algunas Escalas de Colores Estándar

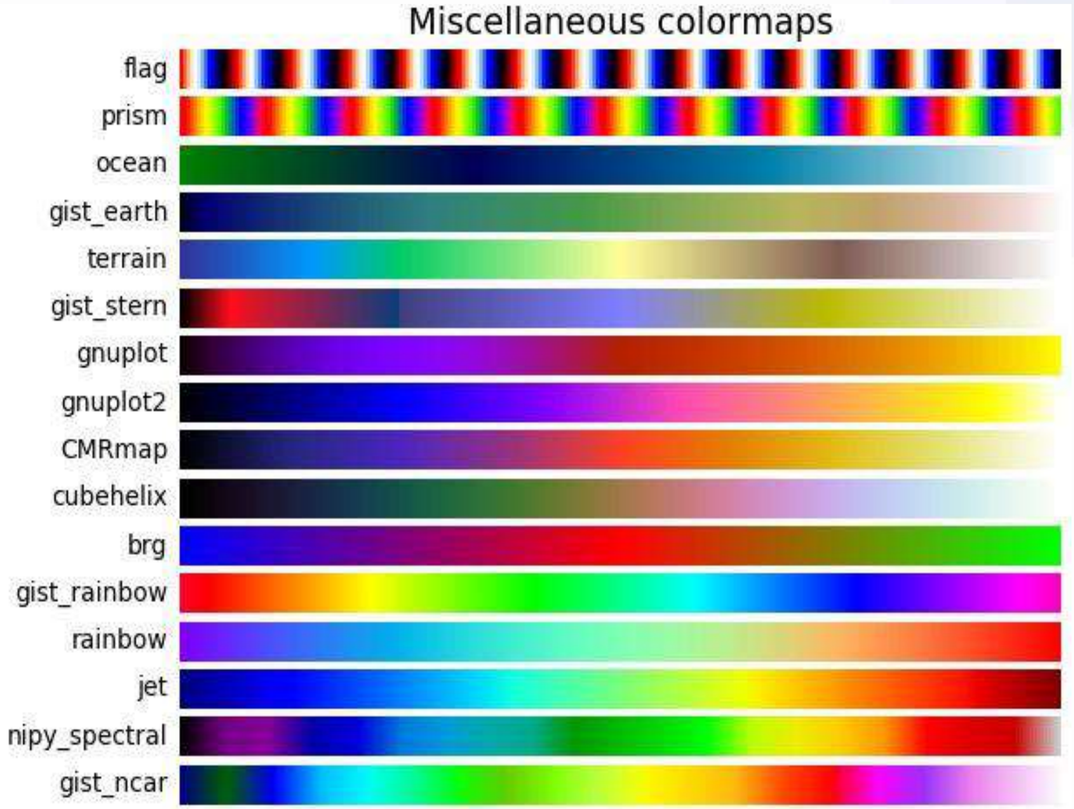
https://matplotlib.org/stable/tutorials/colors/colormaps.html





Ejemplo 6: Algunas Escalas de Colores Estándar





Ejemplo 6: Manipulando Datos con Satpy (Metadatos)

6. Leyendo los atributos de un dataset (metadatos)

```
# reading the attributes
scn['C13'].attrs
```

```
{'orbital_parameters': {'satellite_nominal_latitude': 0.0,
  'satellite_nominal_longitude': -75.19999694824219,
  'satellite_nominal_altitude': 35786023.4375},
 'long_name': 'ABI Cloud and Moisture Imagery brightness temperature at top of atmosphere',
 'standard_name': 'toa_brightness_temperature',
 'sensor_band_bit_depth': 12,
 'units': 'K',
 'resolution': 'y: 0.000056 rad x: 0.000056 rad',
 'grid_mapping': 'goes_imager_projection',
 'cell_methods': 't: point area: point',
 'name': 'C13',
 'wavelength': WavelengthRange(min=10.1, central=10.35, max=10.6, unit='µm'),
 'calibration': 'brightness_temperature',
 'modifiers': (),
 'platform_name': 'GOES-16',
 'sensor': 'abi',
 'scene_abbr': 'F',
 'scan_mode': 'M6',
 'platform_shortname': 'G16',
 'observation_type': 'MCMIP',
 'scene_id': 'Full Disk',
 'orbital_slot': 'GOES-East',
 'instrument_ID': 'FM1',
 'production_site': 'NSOF',
 'timeline_ID': None,
 'start_time': datetime.datetime(2023, 10, 28, 15, 0, 20, 400000),
 'end_time': datetime.datetime(2023, 10, 28, 15, 9, 52, 500000),
 'reader': 'abi_l2_nc',
 'area': Area ID: GOES-East
Description: 2km at nadir
Projection ID: abi_fixed_grid
Projection: {'ellps': 'GRS80', 'h': '35786023', 'lon_0': '-75', 'no_defs': 'None', 'proj': 'geos',
Number of columns: 5424
Number of rows: 5424
Area extent: (-5434894.8851, -5434894.8851, 5434894.8851, 5434894.8851),
 '_satpy_id': DataID(name='C13', wavelength=WavelengthRange(min=10.1, central=10.35, max=10.6, unit
 'ancillary_variables': []}
```

7. Leyendo algunos de los atributos del dataset

```
# get some information from the metadata

# what information will we see?
long_name = scn['C13'].attrs['long_name']
print(long_name)

# pixel unit
units = scn['C13'].attrs['units']
print(units)

# data acquisition time and date
date = scn['C13'].attrs['start_time']
date = date.strftime('%Y-%m-%d %H:%M UTC')
print(date)

# satellite name
satellite = scn['C13'].attrs['platform_name']
print(satellite)

# channel name
channel = scn['C13'].attrs['name']
print(channel)

# wavelenght
wavelenght = scn['C13'].attrs['wavelength']
print(wavelenght)
```

```
ABI Cloud and Moisture Imagery brightness temperature at top of atmosphere
K
2023-10-28 15:00 UTC
GOES-16
C13
10.35 µm (10.1-10.6 µm)
```

Ejemplo 6: Manipulando Datos con Satpy (Decoraciones)

8. Agregando un título y una barra de colores

```
# read the pixel values
data = scn['C13']

# plot size (width x height, in inches)
plt.figure(figsize=(10,10))

# plot the image
plt.imshow(data, vmin=190, vmax=320, cmap='jet')

# add a colorbar
cbar = plt.colorbar(label=f'{long_name} {units}', extend='both', orientation='vertical', pad=0.03, fraction=0.05)

# add a title
plt.title(f'{satellite} - {channel} - {wavelenght}\n{date}' , fontweight='bold', fontsize=10, loc='left')
plt.title('Entrenamiento AECID / AEMET / EUMETSAT', fontsize=10, loc='right')
```

Título de la barra de colores

Extender la barra para ambos los lados

Distancia relativa al plot

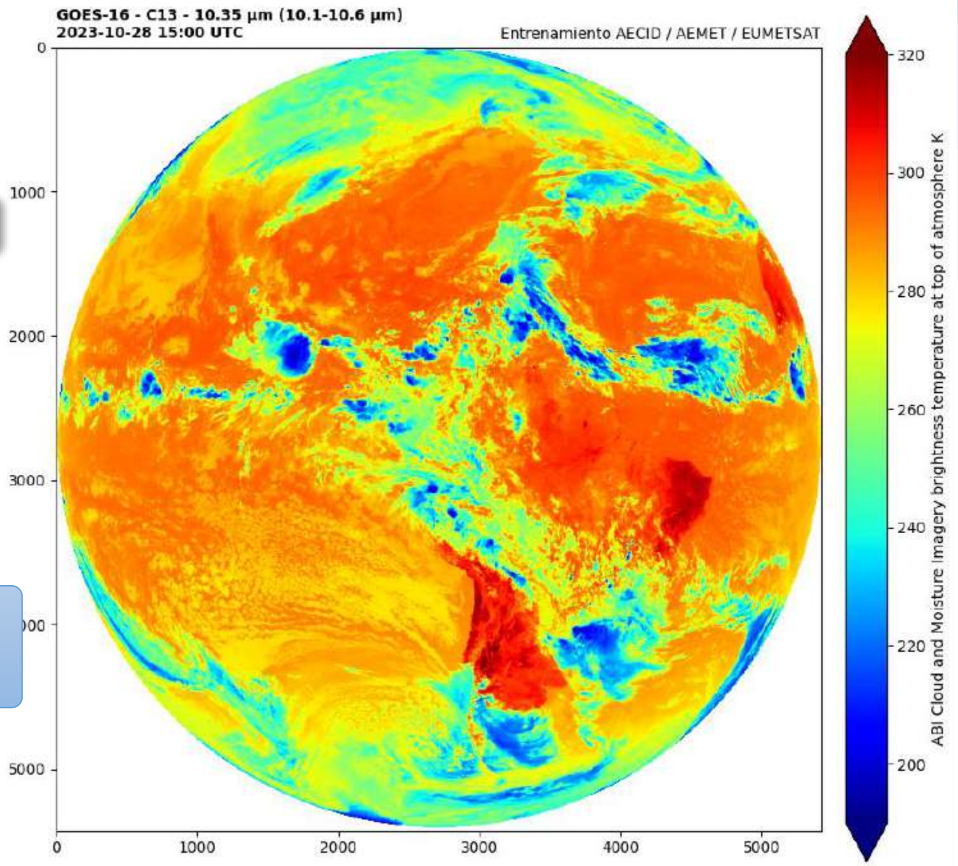
Orientación de la barra

Texto del título

Tamaño del texto

Posición del texto

Extensión de la barra además del plot



Ejemplo 6: Manipulando Datos con Satpy (Cartopy)

9. Usando el atributo “área” como parámetro para la biblioteca Cartopy

```
# read the dataset crs (coordinate reference system)
crs = scn['C13'].attrs['area'].to_cartopy_crs()

# define the projection
ax = plt.axes(projection=crs)

# add coastlines, borders and gridlines
ax.add_feature(cartopy.feature.BORDERS, edgecolor='black', linewidth=0.5)
ax.coastlines(resolution='110m', color='black', linewidth=0.8)
ax.gridlines(color='black', alpha=0.5, linestyle='--', linewidth=0.5)
```

<cartopy.mpl.gridliner.Gridliner at 0x7e4c6b444760>

Lee las informaciones de la proyección

Declara la proyección del dato

https://scitools.org.uk/cartopy/docs/v0.15/crs/projections.html

Añade los limites de los países con Cartopy

Añade las líneas de costa con Cartopy

Añade las líneas de referencia con Cartopy

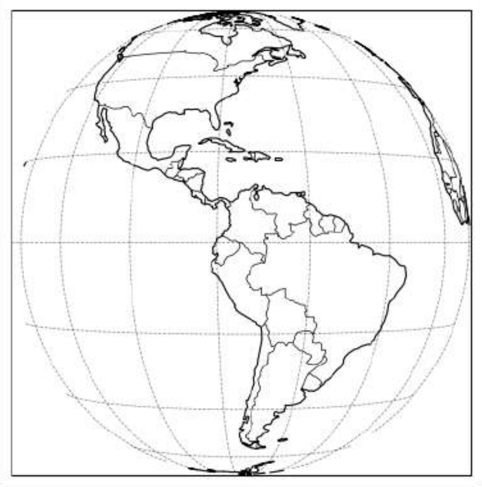


Imagen resultante

```
'area': Area ID: GOES-East
Description: 2km at nadir
Projection ID: abi_fixed_grid
Projection: {'ellps': 'GRS80', 'h': '35786023', 'lon_0': '-75', 'no_defs': 'None', 'proj': 'geos', 'sweep': 'x', 'type': 'crs', 'units': 'm', 'x_0': '0', 'y_0': '0'}
Number of columns: 5424
Number of rows: 5424
Area extent: (-5434894.8851, -5434894.8851, 5434894.8851, 5434894.8851),
```

Ejemplo 6: Manipulando Datos con Satpy

10. Todo en un solo script

```python
#------------------------------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example 6: Learning how to Read and Manipulate Data (GOES-R)
# Author: Diego Souza (INPE/CGCT/DISSM)
#------------------------------------------------------------------------------------------

# download the satellite data
file_name = download_satellite(sat='goes16', date_sat='2023-10-28 15:00')

# initialise scene
scn = Scene(filenames=glob.glob(os.path.join('/content/sample_data/', file_name)), reader='abi_l2_nc')

composite = 'C13'

# load the dataset of interest
scn.load([composite])

# read the dataset crs (coordinate reference system)
crs = scn[composite].attrs['area'].to_cartopy_crs()

# plot size (width x height, in inches)
plt.figure(figsize=(10,10))

# define the projection
ax = plt.axes(projection=crs)

# read the pixel values
data = scn[composite]

# plot the image
plt.imshow(data, extent=crs.bounds, vmin=190, vmax=320, origin='upper', cmap='jet')

# add coastlines and gridlines
ax.coastlines(resolution='110m', color='black', linewidth=0.8)
ax.add_feature(cartopy.feature.BORDERS, edgecolor='black', linewidth=0.5)
gl = ax.gridlines(crs=ccrs.PlateCarree(), color='white', alpha=1.0, linestyle='--', linewidth=0.25,
                  xlocs=np.arange(-180, 181, 10), ylocs=np.arange(-90, 91, 10), draw_labels=False)

# get some information from the metadata
long_name = scn[composite].attrs['long_name']
units = scn[composite].attrs['units']
date = scn[composite].attrs['start_time']
date = date.strftime('%Y-%m-%d %H:%M UTC')
satellite = scn[composite].attrs['platform_name']
channel = scn[composite].attrs['name']
wavelenght = scn[composite].attrs['wavelength']

# add a colorbar
cbar = plt.colorbar(label=f'{long_name} {units}', extend='both', orientation='vertical', pad=0.03, fraction=0.05)

# add a title
plt.title(f'{satellite} - {channel} - {wavelenght}\n{date}' , fontweight='bold', fontsize=10, loc='left')
plt.title('Entrenamiento AECID / AEMET / EUMETSAT', fontsize=10, loc='right')

# remove frame
ax.axis('off')

# show the image
plt.show()
```

Descarga del Dato

Creación del Scene

Cargando un Dataset

Graficando la Información

Leyendo informaciones de proyección

Tamaño del Plot

Proyección

Matriz de Datos

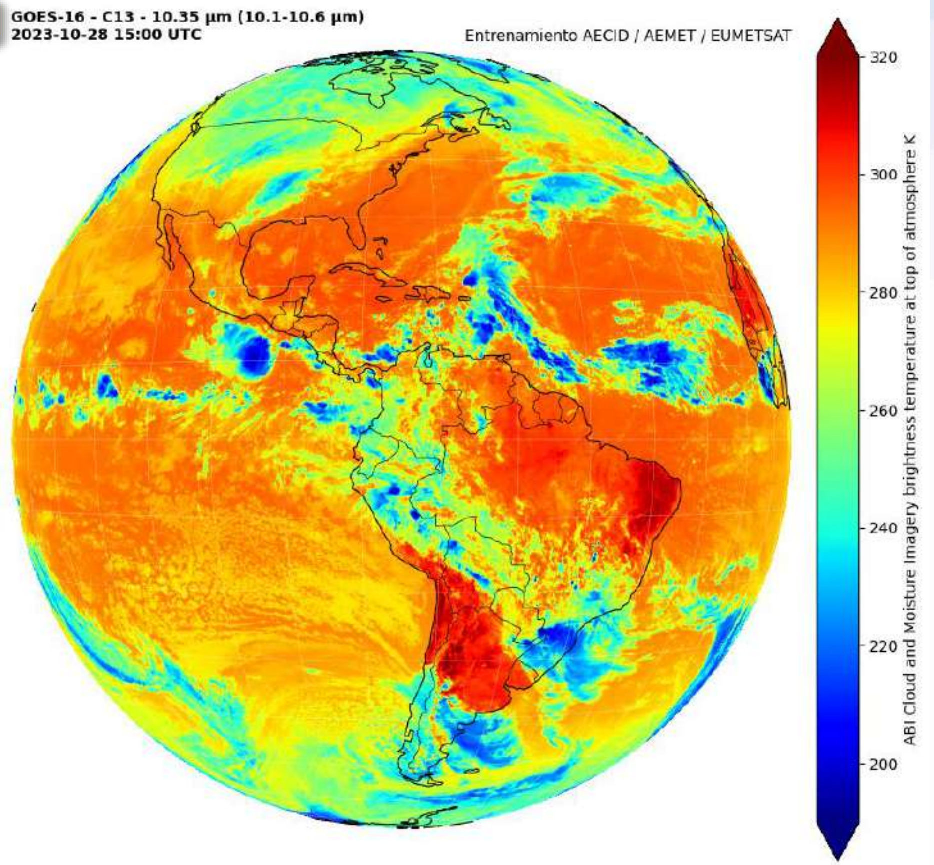
Grafica la matriz

Mapas

Leyendo los metadatos

Barra de colores

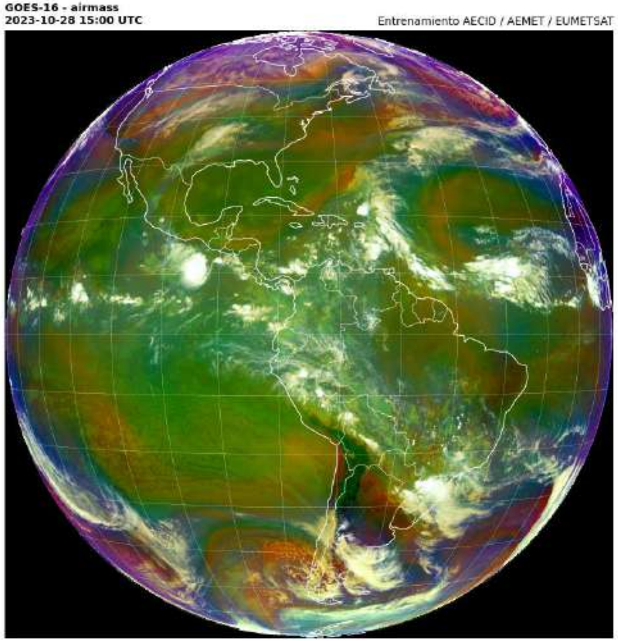
Títulos



Ejemplo 6: Manipulando Datos con Satpy (RGBs)

También es posible generar compuestos RGB

```python
# check the available composites
scn.available_composite_names()
```




```
['airmass',
 'ash',
 'cimss_cloud_type',
 'cimss_cloud_type_raw',
 'cimss_green',
 'cimss_green_sunz',
 'cimss_green_sunz_rayleigh',
 'cimss_true_color',
 'cimss_true_color_sunz',
 'cimss_true_color_sunz_rayleigh',
 'cira_day_convection',
 'cira_fire_temperature',
 'cloud_phase',
 'cloud_phase_distinction',
 'cloud_phase_distinction_raw',
 'cloud_phase_raw',
 'cloudtop',
 'color_infrared',
 'colorized_ir_clouds',
 'convection',
 'day_microphysics',
 'day_microphysics_abi',
 'day_microphysics_eum',
 'dust',
 'fire_temperature_awips',
 'fog',
 'green',
 'green_crefl',
 'green_nocorr',
 'green_raw',
 'green_snow',
 'highlight_C14',
 'ir108_3d',
 'ir_cloud_day',
 'land_cloud',
 'land_cloud_fire',
 'natural_color',
 'natural_color_nocorr',
 'natural_color_raw',
 'natural_color_raw_with_night_ir',
 'night_fog',
 'night_ir_alpha',
 'night_ir_with_background',
 'night_ir_with_background_hires',
 'night_microphysics',
 'night_microphysics_eum',
 'overview',
 'overview_raw',
 'rocket_plume_day',
 'rocket_plume_night',
 'snow',
 'snow_fog',
 'so2',
 'tropical_airmass',
 'true_color',
 'true_color_crefl',
 'true_color_nocorr',
 'true_color_raw',
 'true_color_reproduction',
 'true_color_reproduction_corr',
 'true_color_reproduction_uncorr',
 'true_color_with_night_fires',
 'true_color_with_night_fires_nocorr',
 'true_color_with_night_ir',
 'true_color_with_night_ir_hires',
 'water_vapors1',
 'water_vapors2']
```

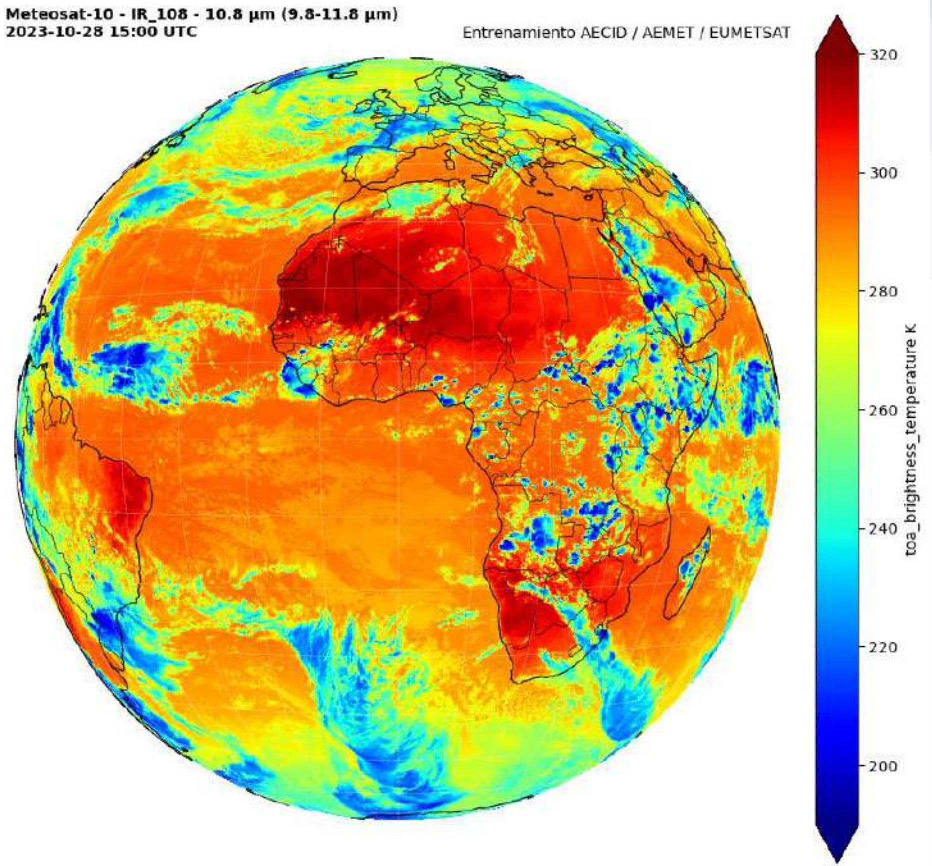
Ejemplo 7: Manipulando Datos con Satpy (MSG 0°)

```python
#-----------------------------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example 7: Learning how to Read and Manipulate Data (MSG)
# Author: Diego Souza (INPE/CGCT/DISSM)
#-----------------------------------------------------------------------------------------

# download the satellite data
file_name = download_satellite(sat='msg0', date_sat='2023-10-28 15:00')

# initialise scene
scn = Scene(filenames=glob.glob(os.path.join('/content/sample_data/', file_name)), reader='seviri_l1b_native')

# dataset name
composite = 'IR_108'

# load the dataset of interest
scn.load([composite])

# read the dataset crs (coordinate reference system)
crs = scn[composite].attrs['area'].to_cartopy_crs()

# plot size (width x height, in inches)
plt.figure(figsize=(10,10))

# define the projection
ax = plt.axes(projection=crs)

# read the pixel values
data = scn[composite]

# plot the image
plt.imshow(data, extent=crs.bounds, vmin=190, vmax=320, origin='upper', cmap='jet')

# add coastlines and gridlines
ax.coastlines(resolution='110m', color='black', linewidth=0.8)
ax.add_feature(cartopy.feature.BORDERS, edgecolor='black', linewidth=0.5)
gl = ax.gridlines(crs=ccrs.PlateCarree(), color='white', alpha=1.0, linestyle='--', linewidth=0.25,
                  xlocs=np.arange(-180, 181, 10), ylocs=np.arange(-90, 91, 10), draw_labels=False)

# get some information from the metadata
long_name = scn[composite].attrs['standard_name']
units = scn[composite].attrs['units']
date = scn[composite].attrs['start_time']
date = date.strftime('%Y-%m-%d %H:%M UTC')
satellite = scn[composite].attrs['platform_name']
channel = scn[composite].attrs['name']
wavelenght = scn[composite].attrs['wavelength']

# add a colorbar
cbar = plt.colorbar(label=f'{long_name} {units}', extend='both', orientation='vertical', pad=0.03, fraction=0.05)

# add a title
plt.title(f'{satellite} - {channel} - {wavelenght}\n{date}' , fontweight='bold', fontsize=10, loc='left')
plt.title('Entrenamiento AECID / AEMET / EUMETSAT', fontsize=10, loc='right')

# remove frame
ax.axis('off')

# show the image
plt.show()
```

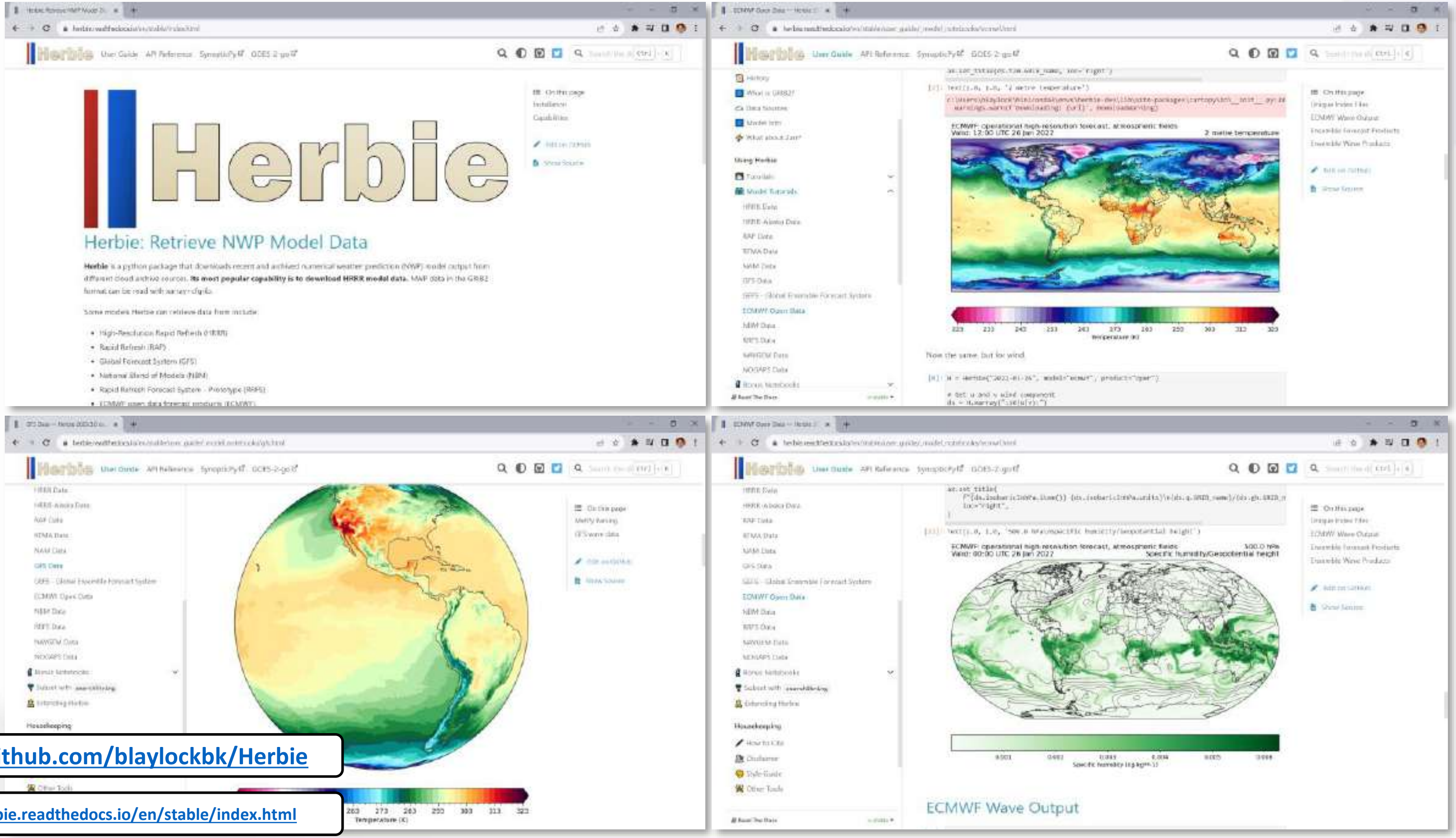
Principales cambios

```python
reader='seviri_l1b_native')
```

```python
composite = 'IR_108'
```



Ejemplo 8: Descargando y Graficando Datos de Modelo



https://github.com/blaylockbk/Herbie

https://herbie.readthedocs.io/en/stable/index.html

Ejemplo 8: Descargando y Graficando Datos de Modelo

Ejemplo 8: Descargando y Graficando Datos de Modelos de Predicción Numérica del Tiempo (PNT)

Ejemplo 8.1: Geopotential Height

Ejemplo 8.1

```
# ... ssing - Example 8: Downloading and Visualizing ...
# ... nce January 21 2022 00:00 ('2022-01-21 00:00')
# gfs data available for download since February 26 2021 00:00 ('2021-02-26 00:00')
# gfs products: 'pgrb2.0p25', 'pgrb2.0p50', 'pgrb2.1p00'

# geopotential height (500 hPa [4800~6000] and 300 hPa [8300~9700])
plot_nwp(model='ecmwf',
         product='oper',
         date_nwp='2023-10-25 12:00',
         area='custom',
         extent=[-140.0, -70.00, 70.00, 70.00],
         fxx='date',
         var='gh', scale=1, offset=0,
         level_hpa='500',
         level_min=4800, level_max=6000, level_int=60,
         plot_type='c_unfilled', contour_color='black',
         apply_cmap=True, cmap='turbo',
         view_clabel=True, fontsize=6,
         linewidth=0.8, linestyle='solid',
         land_ocean=True, land_color = 'lightgray', ocean_color = 'white',
         alpha=1.0, title='new',
         figsize=[15,15])
```

```
Found | model=ecmwf | product=oper | 2023-Oct-25 00:00 UTC F12 | GRIB2 @ azure | IDX @ azure
Created directory: [/root/data/ecmwf/20231025]
Total processing time (nwp): 8.44 seconds.

'output/ecmwf_gh_2023-10-25_12:00.png'
```

- **Nombre de la función**
- **Parámetros**
- **Fecha 'YYYY-MM-DD HH:MM'**
- **Área deseada** → **'custom' o 'previous'**
- **Coordinadas**
- **Step de la salida** → **'date' o algún numero**
- **Variable, escala y offset**
- **Nivel (hPa)**
- **Valores mínimo, máximo y intervalo**
- **Tipo de plot y color de contorno** → **'c_unfilled', 'c_filled', 'pmesh', 'streamlines', 'barbs'**
- **Aplica una colormap? Cual?**
- **Textos en el contorno? Que tamaño?**
- **Espesor y tipo de línea** → **'solid', 'dashed', 'dotted'**
- **Máscara de océano y colores**
- **Transparencia y titulo**
- **Tamaño de la figura (pulgadas)**
- **Salida con informaciones**

Cambiar las funciones de descarga y visualización de datos de modelo no es simple. Vamos hacer algunos cambios juntos.

Ejemplo 8: Descargando y Graficando Datos de Modelo

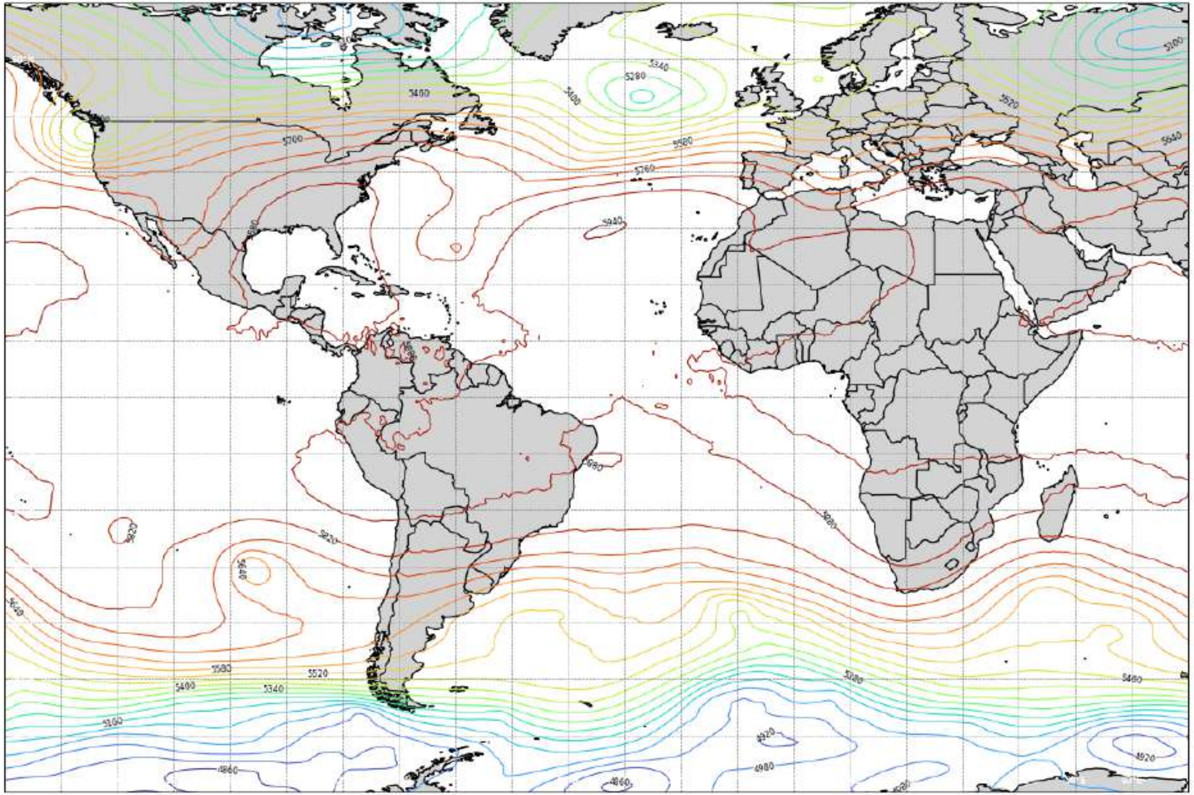
Ejemplo 8.1

Ejemplo 8.1: Geopotential Height

```
#-----------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example 8: Downloading
# Author: Diego Souza (INPE/CGCT/DISSM)
#-----------------------------------------------------------------------

# open ecmwf data available for download since January 21 2022 00:00 ('202
# gfs data available for download since February 26 2021 00:00 ('2021-02-2
# gfs products: 'pgrb2.0p25', 'pgrb2.0p50', 'pgrb2.1p00'

# geopotential height (500 hPa [4800~6000] and 300 hPa [8300~9700])
plot_nwp(model='ecmwf',
         product='oper',
         date_nwp='2023-10-25 12:00',
         area='custom',
         extent=[-140.0, -70.00, 70.00, 70.00],
         fxx='date',
         var='gh', scale=1, offset=0,
         level_hpa='500',
         level_min=4800, level_max=6000, level_int=60,
         plot_type='c_unfilled', contour_color='black',
         apply_cmap=True, cmap='turbo',
         view_clabel=True, fontsize=6,
         linewidth=0.8, linestyle='solid',
         land_ocean=True, land_color = 'lightgray', ocean_color = 'white',
         alpha=1.0, title='new',
         figsize=[15,15])
```

```
-----------------------------------------------------------------------
✅ Found ┆ model=ecmwf ┆ product=oper ┆ 2023-Oct-25 00:00 UTC F12 ┆ GRIB2
Created directory: [/root/data/ecmwf/20231025]
Total processing time (nwp): 8.44 seconds.
-----------------------------------------------------------------------
'output/ecmwf_gh_2023-10-25_12:00.png'
```

ECMWF: operational high-resolution forecast, atmospheric fields
Valid: 12:00 UTC 25 Oct 2023
| Geopotential height (gpm) 500.0 hPa



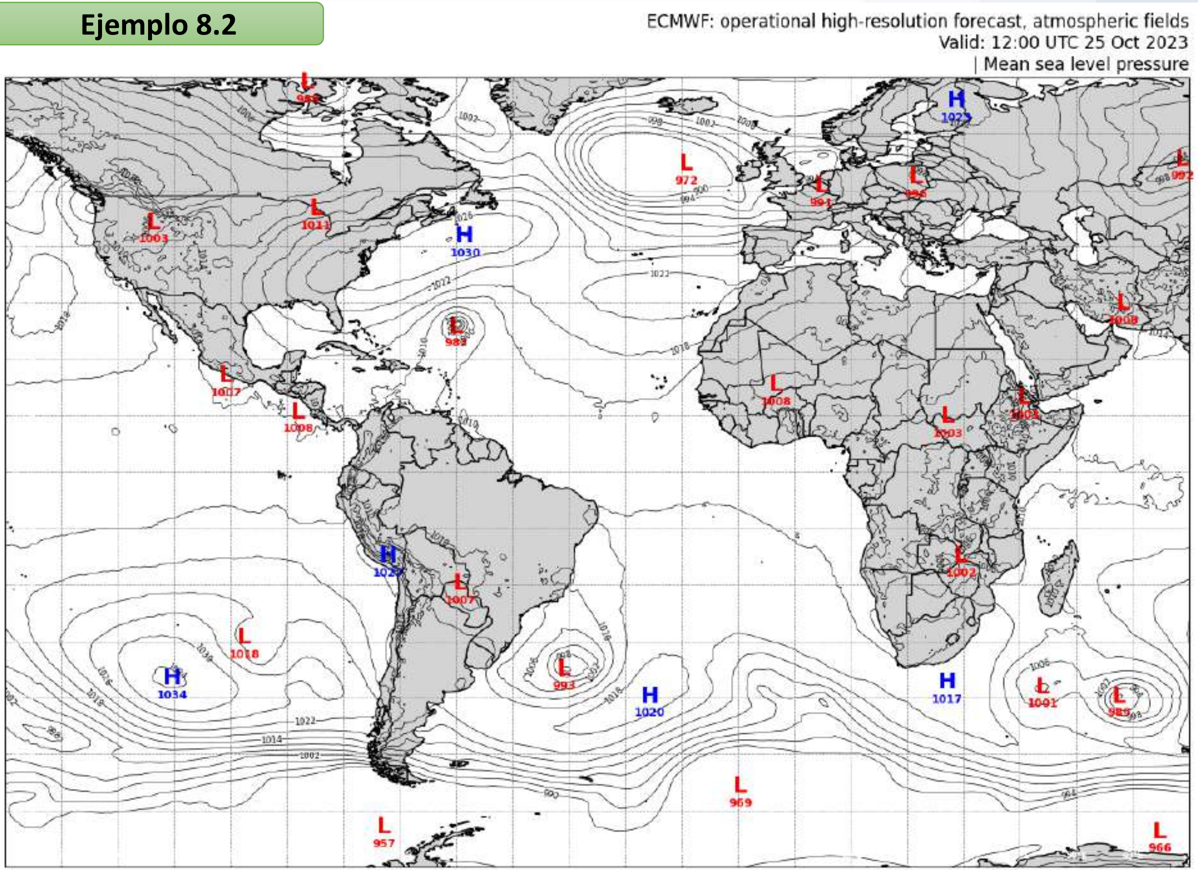
Ejemplo 8: Descargando y Graficando Datos de Modelo

Ejemplo 8.2

Example 8.2: Mean Sea Level Pressure

```
#------------------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example 8: Downloading
# Author: Diego Souza (INPE/CGCT/DISSM)
#------------------------------------------------------------------------------

# open ecmwf data available for download since January 21 2022 00:00 ('2022
# gfs data available for download since February 26 2021 00:00 ('2021-02-26
# gfs products: 'pgrb2.0p25', 'pgrb2.0p50', 'pgrb2.1p00'

# mean sea level pressure
plot_nwp(model='ecmwf',
         product='oper',
         date_nwp='2023-10-25 12:00',
         area='custom',
         extent=[-140.0, -70.00, 70.00, 70.00],
         fxx='date',
         var='msl', scale=0.01, offset=0,
         level_hpa='',
         level_min=990, level_max=1050, level_int=4,
         plot_type='c_unfilled', contour_color='black',
         apply_cmap=False, cmap='jet',
         view_clabel=True, fontsize=6,
         linewidth=0.5, linestyle='solid',
         land_ocean=True, land_color = 'lightgray', ocean_color = 'white',
         alpha=1.0, title='new',
         figsize=[15,15])
```

```
------------------------------------------------------------------------------
✅ Found ┆ model=ecmwf ┆ product=oper ┆ 2023-Oct-25 00:00 UTC F12 ┆ GRIB2
Total processing time (nwp): 9.01 seconds.
------------------------------------------------------------------------------
'output/ecmwf_msl_2023-10-25_12:00.png'
```



Ejemplo 8: Descargando y Graficando Datos de Modelo

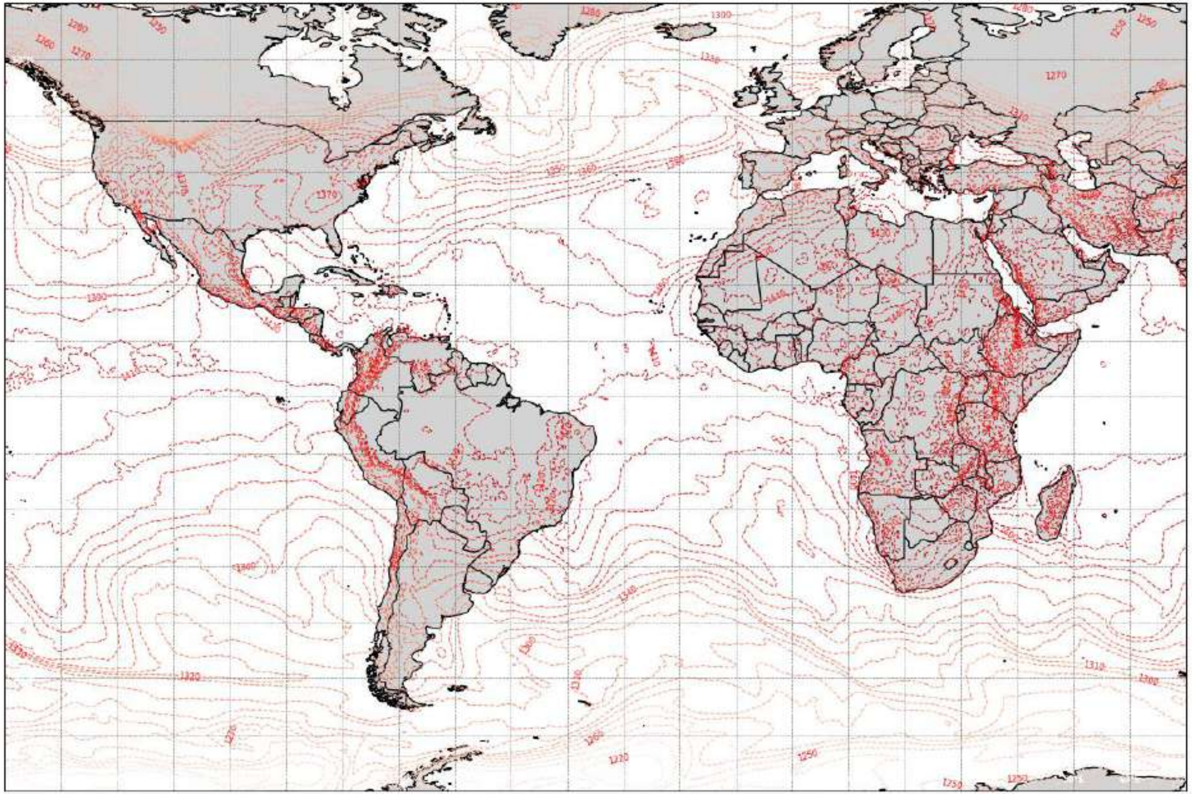
Ejemplo 8.3

Example 8.3: Thickness

```
#--------------------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example 8: Downloading
# Author: Diego Souza (INPE/CGCT/DISSM)
#--------------------------------------------------------------------------------

# open ecmwf data available for download since January 21 2022 00:00 ('2022
# gfs data available for download since February 26 2021 00:00 ('2021-02-26
# gfs products: 'pgrb2.0p25', 'pgrb2.0p50', 'pgrb2.1p00'

# thickness (1000-850 hPa)
plot_nwp(model='ecmwf',
         product='oper',
         date_nwp='2023-10-25 12:00',
         area='custom',
         extent=[-140.0, -70.00, 70.00, 70.00],
         fxx='date',
         var='thickness', scale=1, offset=0,
         level_hpa='1000,850',
         level_min=1200, level_max=1500, level_int=10,
         plot_type='c_unfilled', contour_color='red',
         apply_cmap=True, cmap='Reds',
         view_clabel=True, fontsize=6,
         linewidth=0.8, linestyle='dashed',
         land_ocean=True, land_color = 'lightgray', ocean_color = 'white',
         alpha=1.0, title='new',
         figsize=[15,15])
```

```
--------------------------------------------------------------------------------
✅ Found ┆ model=ecmwf ┆ product=oper ┆ 2023-Oct-25 00:00 UTC F12 ┆ GRIB2
Total processing time (nwp): 8.4 seconds.
--------------------------------------------------------------------------------
'output/ecmwf_thickness_2023-10-25_12:00.png'
```

ECMWF: operational high-resolution forecast, atmospheric fields
Valid: 12:00 UTC 25 Oct 2023
| Thickness 1000.0 - 850.0 hPa



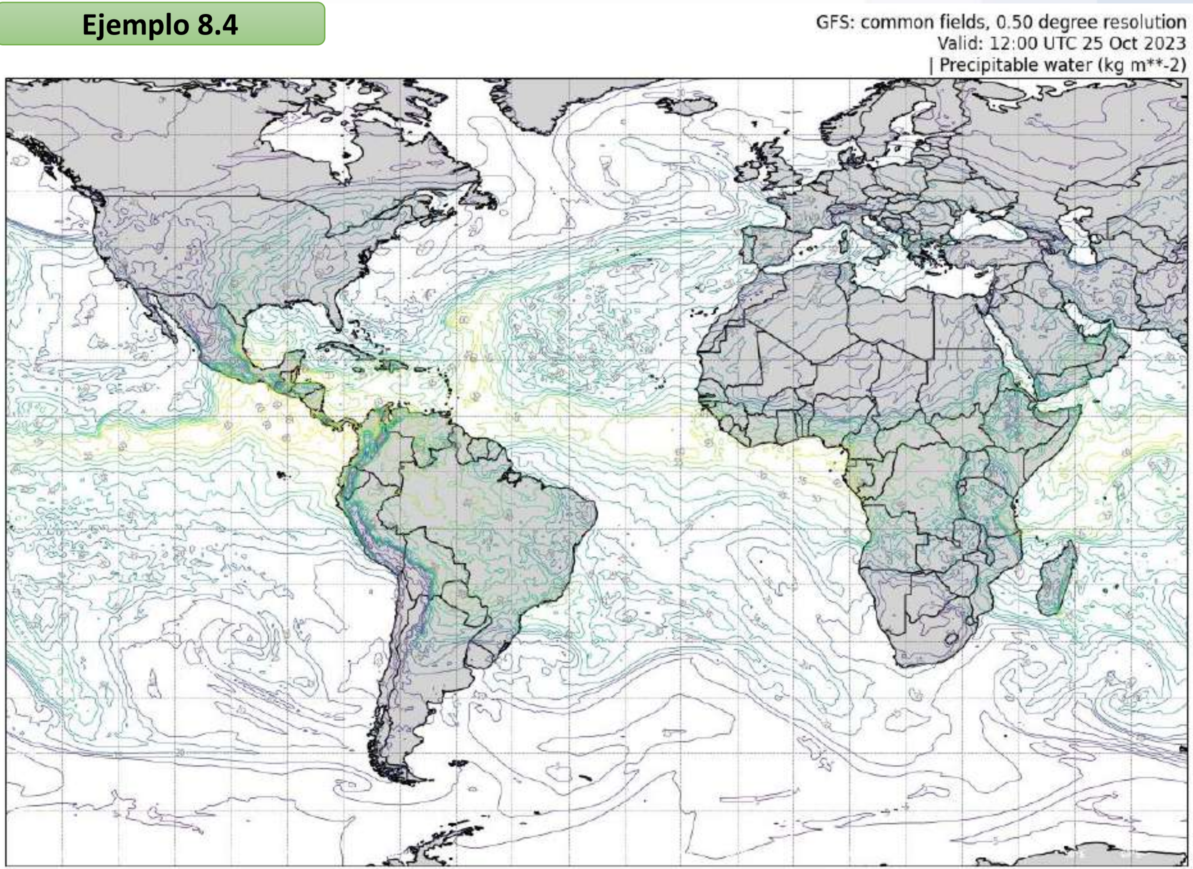
Ejemplo 8: Descargando y Graficando Datos de Modelo

Ejemplo 8.4

Example 8.4: Precipitable Water

```
#-----------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example 8: Downloading an
# Author: Diego Souza (INPE/CGCT/DISSM)
#-----------------------------------------------------------------------

# open ecmwf data available for download since January 21 2022 00:00 ('2022-0
# gfs data available for download since February 26 2021 00:00 ('2021-02-26 0
# gfs products: 'pgrb2.0p25', 'pgrb2.0p50', 'pgrb2.1p00'

# precipitable water
plot_nwp(model='gfs',
         product='pgrb2.0p50',
         date_nwp='2023-10-25 12:00',
         area='custom',
         extent=[-140.0, -70.00, 70.00, 70.00],
         fxx='date',
         var='PWAT', scale=1, offset=0,
         level_hpa='',
         level_min=0, level_max=70, level_int=5,
         plot_type='c_unfilled', contour_color='gray',
         apply_cmap=True, cmap='viridis',
         view_clabel=True, fontsize=6,
         linewidth=0.5, linestyle='solid',
         land_ocean=True, land_color = 'lightgray', ocean_color = 'white',
         alpha=1.0, title='new',
         figsize=[15,15])
```

```
-----------------------------------------------------------------------
✅ Found ┆ model=gfs │ product=pgrb2.0p50 ┆ 2023-Oct-25 00:00 UTC F12 ┆ GRIB2
👷 Created directory: [/root/data/gfs/20231025]
Total processing time (nwp): 9.1 seconds.
-----------------------------------------------------------------------
'output/gfs_PWAT_2023-10-25_12:00.png'
```



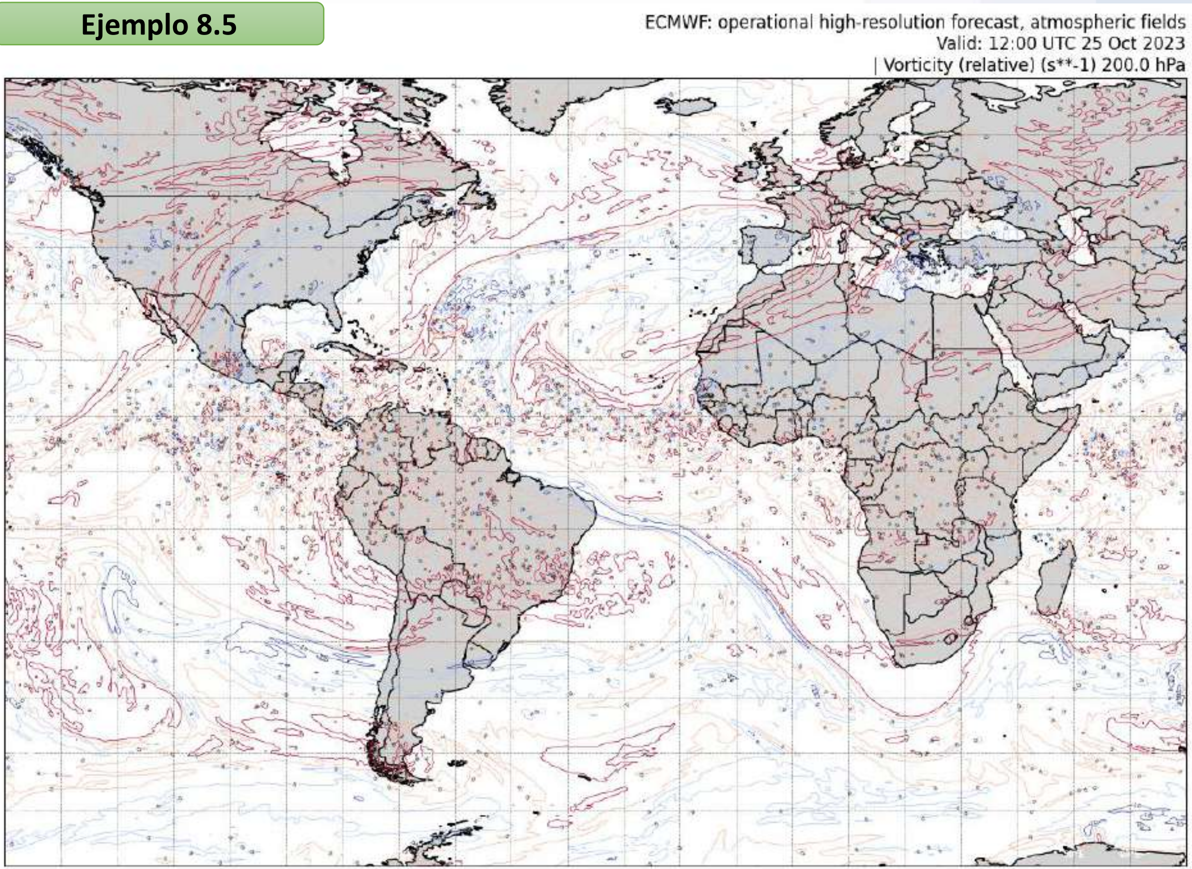
Ejemplo 8: Descargando y Graficando Datos de Modelo

Ejemplo 8.5

Example 8.5: Vorticity (relative)

```
#-----------------------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example 8: Downloading
# Author: Diego Souza (INPE/CGCT/DISSM)
#-----------------------------------------------------------------------------------

# open ecmwf data available for download since January 21 2022 00:00 ('2022
# gfs data available for download since February 26 2021 00:00 ('2021-02-26
# gfs products: 'pgrb2.0p25', 'pgrb2.0p50', 'pgrb2.1p00'

# vorticity (relative) (1000, 850, 700, 500, 300 and 200 hPa)
plot_nwp(model='ecmwf',
         product='oper',
         date_nwp='2023-10-25 12:00',
         area='custom',
         extent=[-140.0, -70.00, 70.00, 70.00],
         fxx='date',
         var='vo', scale=10**5, offset=0,
         level_hpa='200',
         level_min=-10, level_max=10, level_int=5,
         plot_type='c_unfilled', contour_color='black',
         apply_cmap=True, cmap='coolwarm',
         view_clabel=True, fontsize=4,
         linewidth=0.5, linestyle='solid',
         land_ocean=True, land_color = 'lightgray', ocean_color = 'white',
         alpha=1.0, title='new',
         figsize=[15,15])
```

```
-----------------------------------------------------------------------------------
✅ Found ┆ model=ecmwf ┆ product=oper ┆ 2023-Oct-25 00:00 UTC F12 ┆ GRIB2
Total processing time (nwp): 26.81 seconds.
-----------------------------------------------------------------------------------
'output/ecmwf_vo_2023-10-25_12:00.png'
```



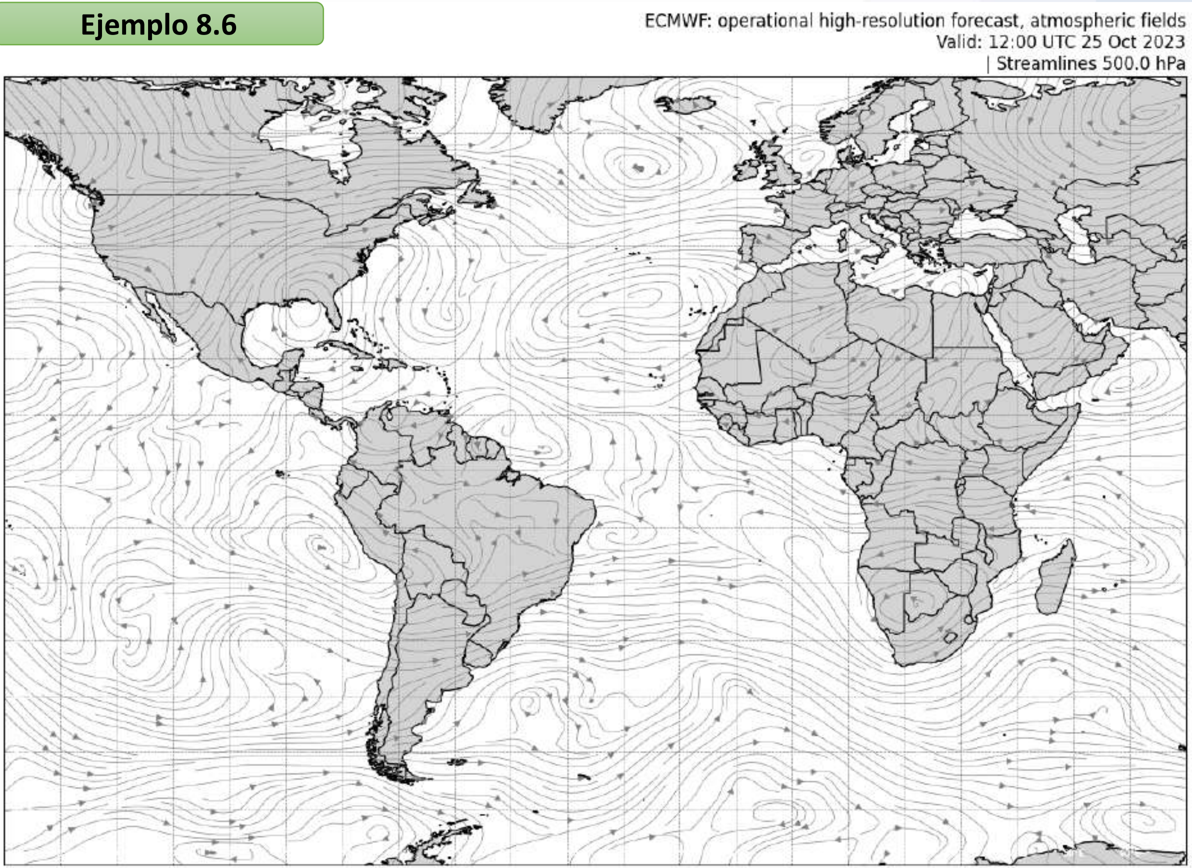
Ejemplo 8: Descargando y Graficando Datos de Modelo

Ejemplo 8.6

Example 8.6: Streamlines

```
#-----------------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example 8: Downloading
# Author: Diego Souza (INPE/CGCT/DISSM)
#-----------------------------------------------------------------------------

# open ecmwf data available for download since January 21 2022 00:00 ('2022
# gfs data available for download since February 26 2021 00:00 ('2021-02-26
# gfs products: 'pgrb2.0p25', 'pgrb2.0p50', 'pgrb2.1p00'

# streamlines (850, 700, 500 and 200 hPa)
plot_nwp(model='ecmwf',
         product='oper',
         date_nwp='2023-10-25 12:00',
         area='custom',
         extent=[-140.0, -70.00, 70.00, 70.00],
         fxx='date',
         var='winds', scale=1, offset=0,
         level_hpa='500',
         level_min=0, level_max=60, level_int=5,
         plot_type='streamlines', contour_color='gray',
         apply_cmap=False, cmap='jet',
         view_clabel=True, fontsize=6,
         linewidth=0.5, linestyle='solid',
         land_ocean=True, land_color = 'lightgray', ocean_color = 'white',
         alpha=1.0, title='new',
         figsize=[15,15])
```

```
-----------------------------------------------------------------------------
✅ Found ¦ model=ecmwf ¦ product=oper ¦ 2023-Oct-25 00:00 UTC F12 ¦ GRIB2 @
Total processing time (nwp): 49.26 seconds.
-----------------------------------------------------------------------------
'output/ecmwf winds 2023-10-25 12:00.png'
```



Ejemplo 8: Descargando y Graficando Datos de Modelo

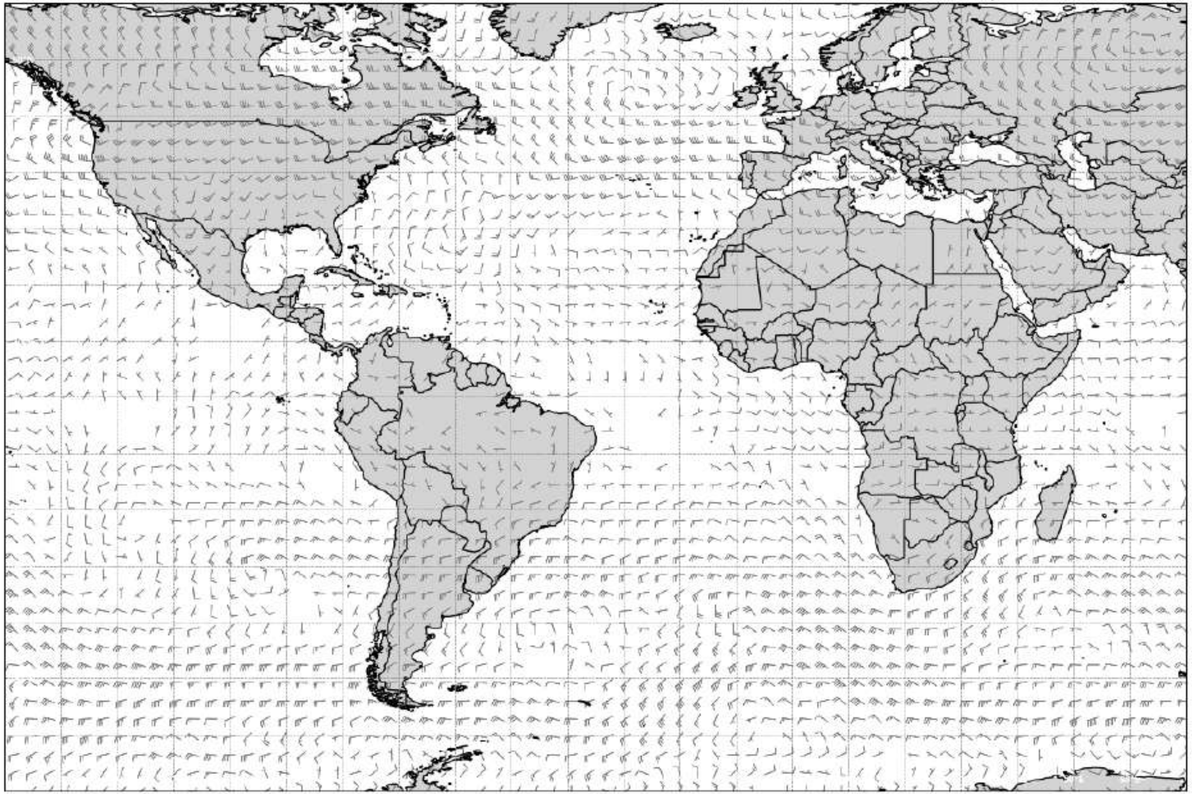
Ejemplo 8.7

Example 8.6: Streamlines

```
#-----------------------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example 8: Downloading
# Author: Diego Souza (INPE/CGCT/DISSM)
#-----------------------------------------------------------------------------------

# open ecmwf data available for download since January 21 2022 00:00 ('2022
# gfs data available for download since February 26 2021 00:00 ('2021-02-26
# gfs products: 'pgrb2.0p25', 'pgrb2.0p50', 'pgrb2.1p00'

# streamlines (850, 700, 500 and 200 hPa)
plot_nwp(model='ecmwf',
         product='oper',
         date_nwp='2023-10-25 12:00',
         area='custom',
         extent=[-140.0, -70.00, 70.00, 70.00],
         fxx='date',
         var='winds', scale=1, offset=0,
         level_hpa='500',
         level_min=0, level_max=60, level_int=5,
         plot_type='streamlines', contour_color='gray',
         apply_cmap=False, cmap='jet',
         view_clabel=True, fontsize=6,
         linewidth=0.5, linestyle='solid',
         land_ocean=True, land_color = 'lightgray', ocean_color = 'white',
         alpha=1.0, title='new',
         figsize=[15,15])
```

```
-----------------------------------------------------------------------------------
✅ Found ┊ model=ecmwf ┊ product=oper ┊ 2023-Oct-25 00:00 UTC F12 ┊ GRIB2 @
Total processing time (nwp): 49.26 seconds.
-----------------------------------------------------------------------------------
'output/ecmwf winds 2023-10-25 12:00.png'
```

ECMWF: operational high-resolution forecast, atmospheric fields
Valid: 12:00 UTC 25 Oct 2023
| Wind Barbs 500.0 hPa



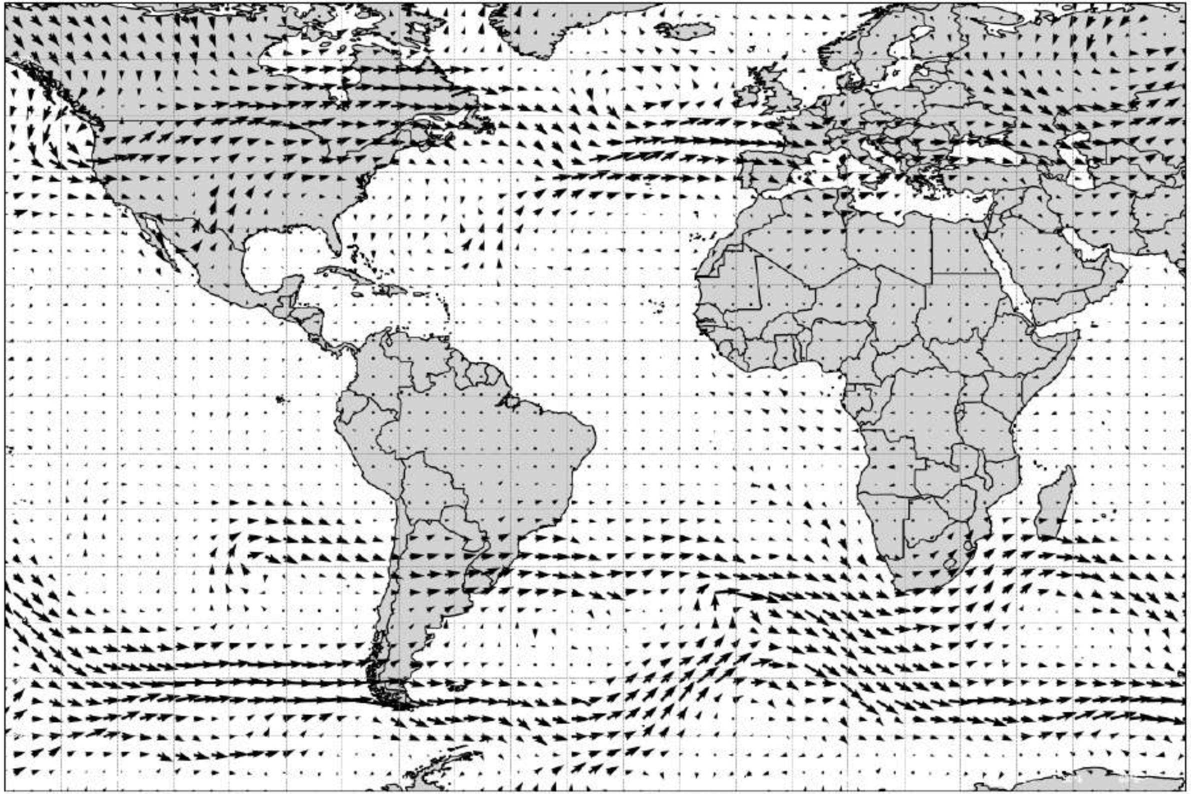
Ejemplo 8: Descargando y Graficando Datos de Modelo

Ejemplo 8.8

Example 8.8: Wind Vectors

```
#-----------------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example 8: Downloading a
# Author: Diego Souza (INPE/CGCT/DISSM)
#-----------------------------------------------------------------------------

# open ecmwf data available for download since January 21 2022 00:00 ('2022-
# gfs data available for download since February 26 2021 00:00 ('2021-02-26
# gfs products: 'pgrb2.0p25', 'pgrb2.0p50', 'pgrb2.1p00'

# wind barbs (250, 300, 500, 700, 850 hPa)
plot_nwp(model='ecmwf',
         product='oper',
         date_nwp='2023-10-25 12:00',
         area='custom',
         extent=[-140.0, -70.00, 70.00, 70.00],
         fxx='date',
         var='winds', scale=1, offset=0,
         level_hpa='500',
         level_min=0, level_max=60, level_int=5,
         plot_type='vectors', contour_color='black',
         apply_cmap=False, cmap='jet',
         view_clabel=True, fontsize=6,
         linewidth=0.5, linestyle='solid',
         land_ocean=True, land_color = 'lightgray', ocean_color = 'white',
         alpha=1.0, title='new',
         figsize=[15,15])
```

```
-----------------------------------------------------------------------------
✅ Found | model=ecmwf | product=oper | 2023-Oct-25 00:00 UTC F12 | GRIB2 @
Total processing time (nwp): 7.6 seconds.
-----------------------------------------------------------------------------
'output/ecmwf_winds_2023-10-25_12:00.png'
```



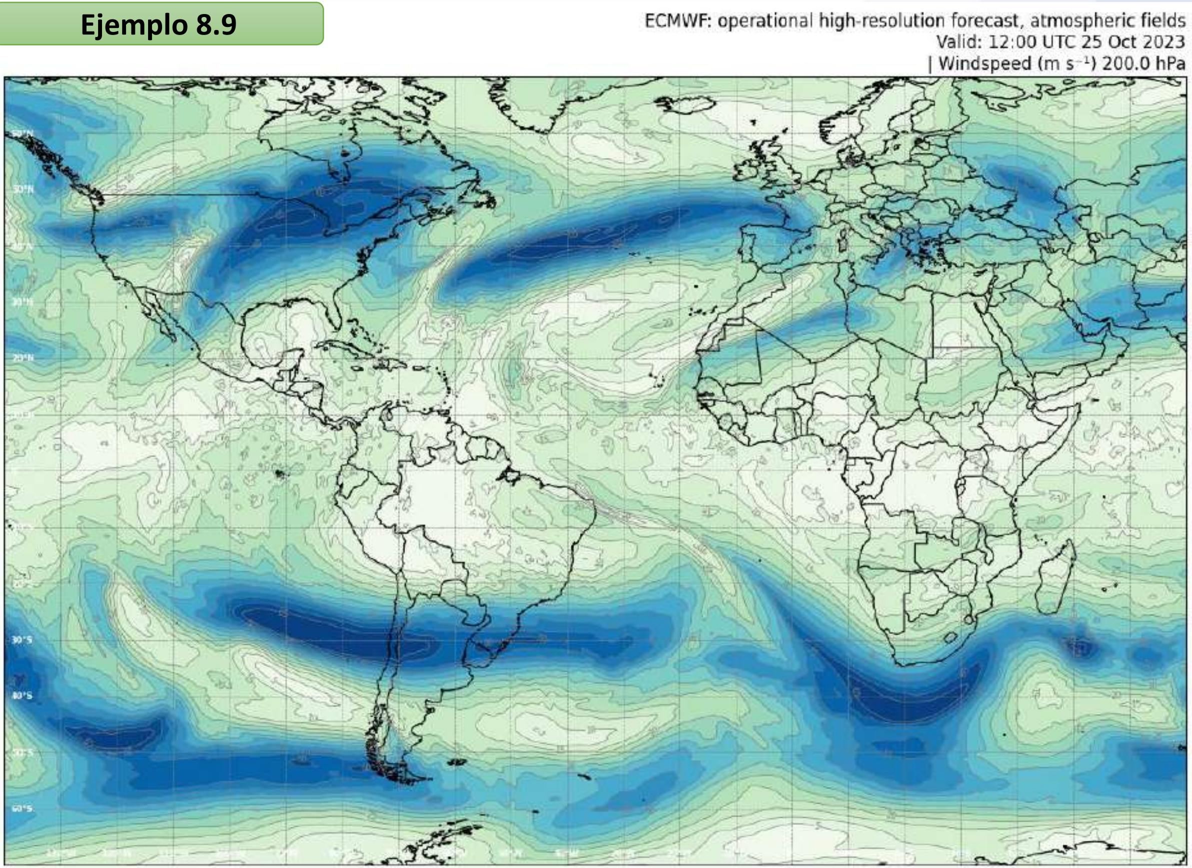
Ejemplo 8: Descargando y Graficando Datos de Modelo

Ejemplo 8.9

Example 8.9: Wind Speed

```
#-----------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example 8: Downloading
# Author: Diego Souza (INPE/CGCT/DISSM)
#-----------------------------------------------------------------------

# open ecmwf data available for download since January 21 2022 00:00 ('2022
# gfs data available for download since February 26 2021 00:00 ('2021-02-26
# gfs products: 'pgrb2.0p25', 'pgrb2.0p50', 'pgrb2.1p00'

# wind speed (200 hPa)
plot_nwp(model='ecmwf',
         product='oper',
         date_nwp='2023-10-25 12:00',
         area='custom',
         extent=[-140.0, -70.00, 70.00, 70.00],
         fxx='date',
         var='windspeed', scale=1, offset=0,
         level_hpa='200',
         level_min=-0, level_max=70, level_int=5,
         plot_type='c_filled', contour_color='gray',
         apply_cmap=True, cmap='GnBu',
         view_clabel=True, fontsize=6,
         linewidth=0.5, linestyle='solid',
         land_ocean=False, land_color = 'lightgray', ocean_color = 'white',
         alpha=1.0, title='new',
         figsize=[15,15])
```

```
-----------------------------------------------------------------------
✅ Found | model=ecmwf | product=oper | 2023-Oct-25 00:00 UTC F12 | GRID2 @
Total processing time (nwp): 7.26 seconds.
-----------------------------------------------------------------------
'output/ecmwf_windspeed_2023-10-25_12:00.png'
```



Ejemplo 8: Descargando y Graficando Datos de Modelo

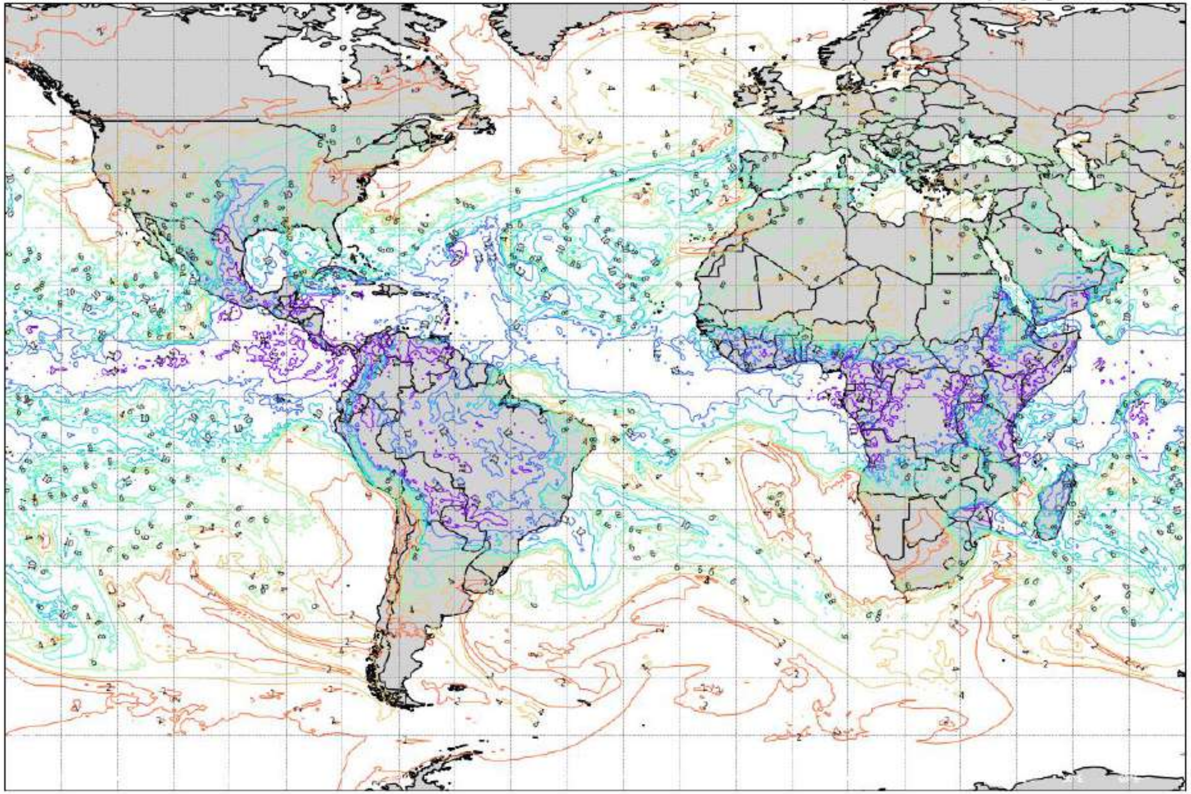
Ejemplo 8.10

Example 8.10: Specific Humidity

```
#-----------------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example 8: Downloading
# Author: Diego Souza (INPE/CGCT/DISSM)
#-----------------------------------------------------------------------------

# open ecmwf data available for download since January 21 2022 00:00 ('2022
# gfs data available for download since February 26 2021 00:00 ('2021-02-26
# gfs products: 'pgrb2.0p25', 'pgrb2.0p50', 'pgrb2.1p00'

# specific humidity (850 and 700 hPa)
plot_nwp(model='ecmwf',
         product='oper',
         date_nwp='2023-10-25 12:00',
         area='custom',
         extent=[-140.0, -70.00, 70.00, 70.00],
         fxx='date',
         var='q', scale=10**3, offset=0,
         level_hpa='850',
         level_min=0, level_max=15, level_int=2,
         plot_type='c_unfilled', contour_color='black',
         apply_cmap=True, cmap='rainbow_r',
         view_clabel=True, fontsize=6,
         linewidth=0.8, linestyle='solid',
         land_ocean=True, land_color = 'lightgray', ocean_color = 'white',
         alpha=1.0, title='new',
         figsize=[15,15])
```

```
-----------------------------------------------------------------------------
✅ Found ┊ model=ecmwf ┊ product=oper ┊ 2023-Oct-25 00:00 UTC F12 ┊ GRIB2
Total processing time (nwp): 10.84 seconds.
-----------------------------------------------------------------------------
'output/ecmwf_q_2023-10-25_12:00.png'
```

ECMWF: operational high-resolution forecast, atmospheric fields
Valid: 12:00 UTC 25 Oct 2023
| Specific humidity (kg kg**-1) 850.0 hPa



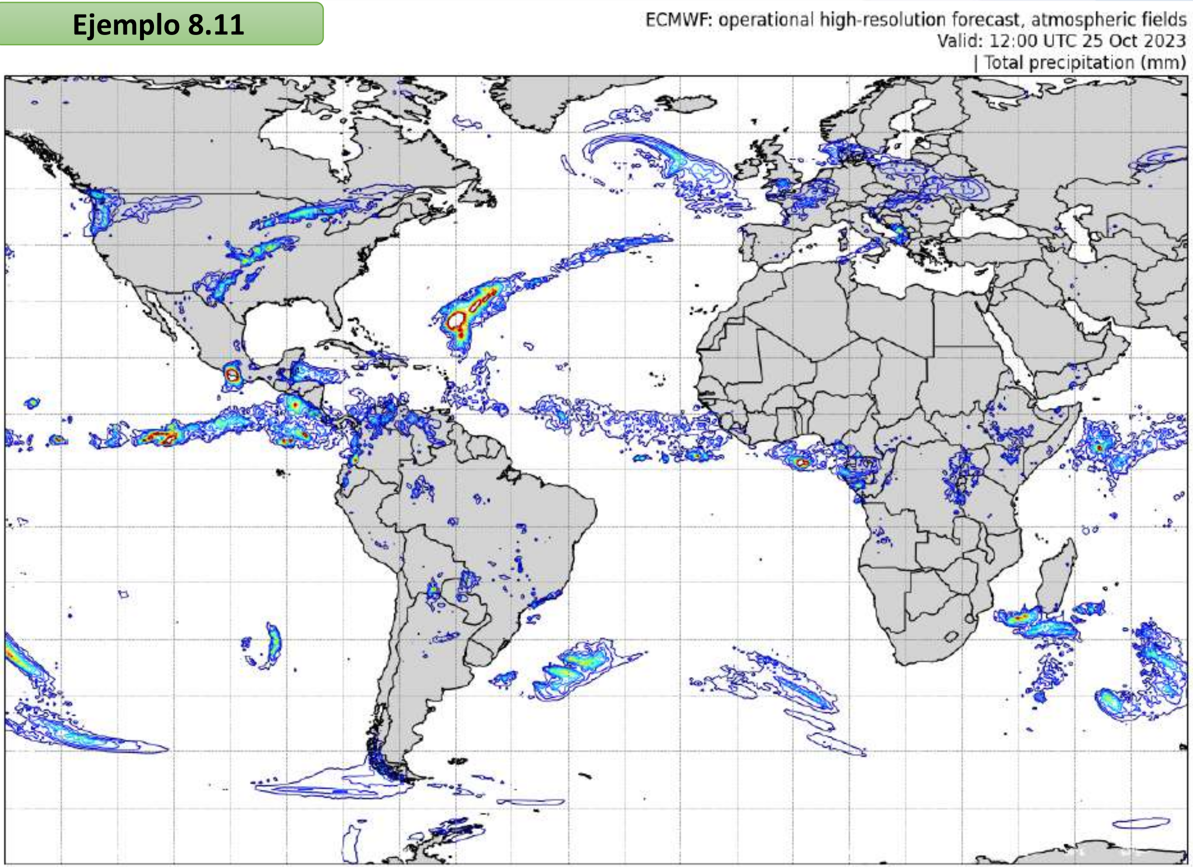
Ejemplo 8: Descargando y Graficando Datos de Modelo

Ejemplo 8.11

Example 8.11: Total Precipitation

```
#-----------------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example 8: Downloading a
# Author: Diego Souza (INPE/CGCT/DISSM)
#-----------------------------------------------------------------------------

# open ecmwf data available for download since January 21 2022 00:00 ('2022
# gfs data available for download since February 26 2021 00:00 ('2021-02-26
# gfs products: 'pgrb2.0p25', 'pgrb2.0p50', 'pgrb2.1p00'

# total precipitation
plot_nwp(model='ecmwf',
         product='oper',
         date_nwp='2023-10-25 12:00',
         area='custom',
         extent=[-140.0, -70.00, 70.00, 70.00],
         fxx='date',
         var='tp', scale=1000, offset=0,
         level_hpa='',
         level_min=5, level_max=70, level_int=5,
         plot_type='c_unfilled', contour_color='black',
         apply_cmap=True, cmap='jet',
         view_clabel=False, fontsize=6,
         linewidth=0.8, linestyle='solid',
         land_ocean=True, land_color = 'lightgray', ocean_color = 'white',
         alpha=1.0, title='new',
         figsize=[15,15])
```

```
-----------------------------------------------------------------------------
✅ Found ┆ model=ecmwf ┆ product=oper ┆ 2023-Oct-25 00:00 UTC F12 ┆ GRIB2 @
Total processing time (nwp): 5.59 seconds.
-----------------------------------------------------------------------------
'output/ecmwf_tp_2023-10-25_12:00.png'
```



Ejemplo 8: Descargando y Graficando Datos de Modelo

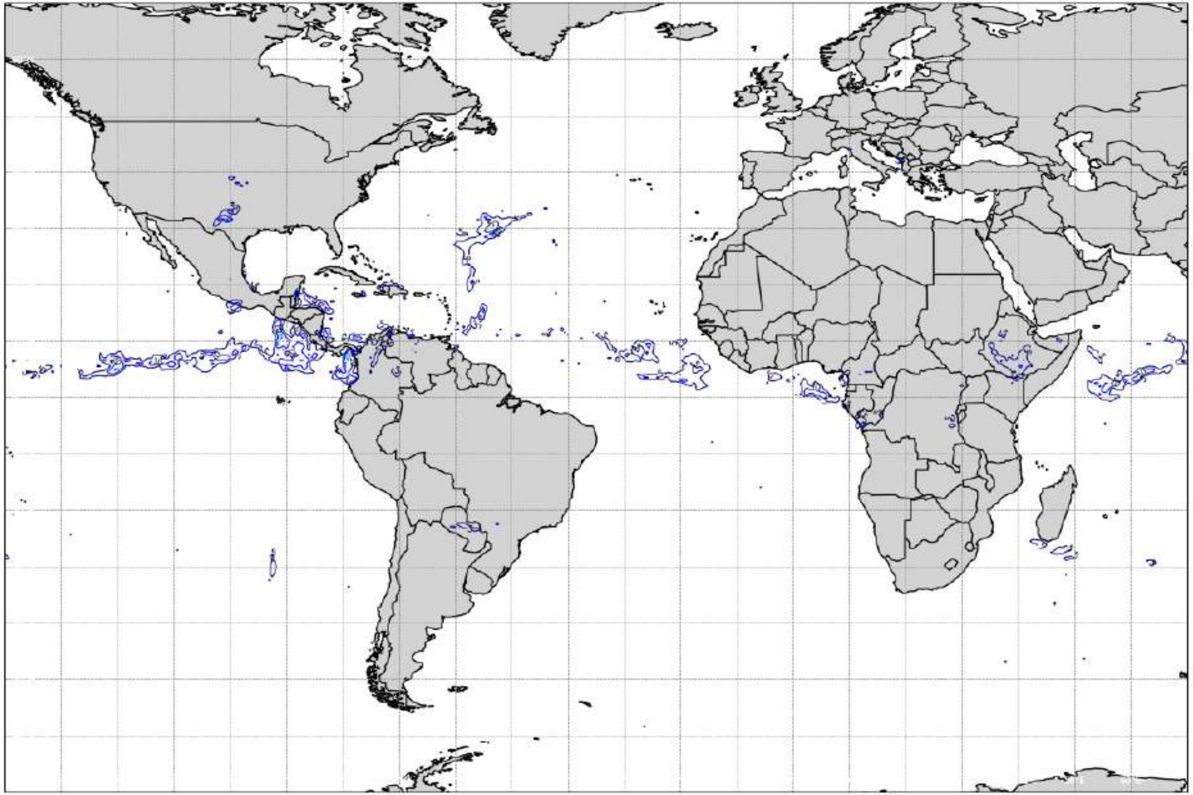
Ejemplo 8.12

Example 8.12: Convective Precipitation

```
#-----------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example 8: Downloading
# Author: Diego Souza (INPE/CGCT/DISSM)
#-----------------------------------------------------------------------

# open ecmwf data available for download since January 21 2022 00:00 ('2022
# gfs data available for download since February 26 2021 00:00 ('2021-02-26
# gfs products: 'pgrb2.0p25', 'pgrb2.0p50', 'pgrb2.1p00'

# convective precipitation (kg/m²)
plot_nwp(model='gfs',
         product='pgrb2.0p50',
         date_nwp='2023-10-25 12:00',
         area='custom',
         extent=[-140.0, -70.00, 70.00, 70.00],
         fxx='date',
         var='ACPCP', scale=1, offset=0,
         level_hpa='',
         level_min=5, level_max=50, level_int=5,
         plot_type='c_unfilled', contour_color='black',
         apply_cmap=True, cmap='jet',
         view_clabel=True, fontsize=6,
         linewidth=0.8, linestyle='solid',
         land_ocean=True, land_color = 'lightgray', ocean_color = 'white',
         alpha=1.0, title='new',
         figsize=[15,15])
```

```
-----------------------------------------------------------------------
✅ Found ┊ model=gfs ┊ product=pgrb2.0p50 ┊ 2023-Oct-25 00:00 UTC F12 ┊ GRI
Total processing time (nwp): 3.28 seconds.
-----------------------------------------------------------------------
'output/gfs_ACPCP_2023-10-25_12:00.png'
```

GFS: common fields, 0.50 degree resolution
Valid: 12:00 UTC 25 Oct 2023
| Convective precipitation (water) (kg m**-2)



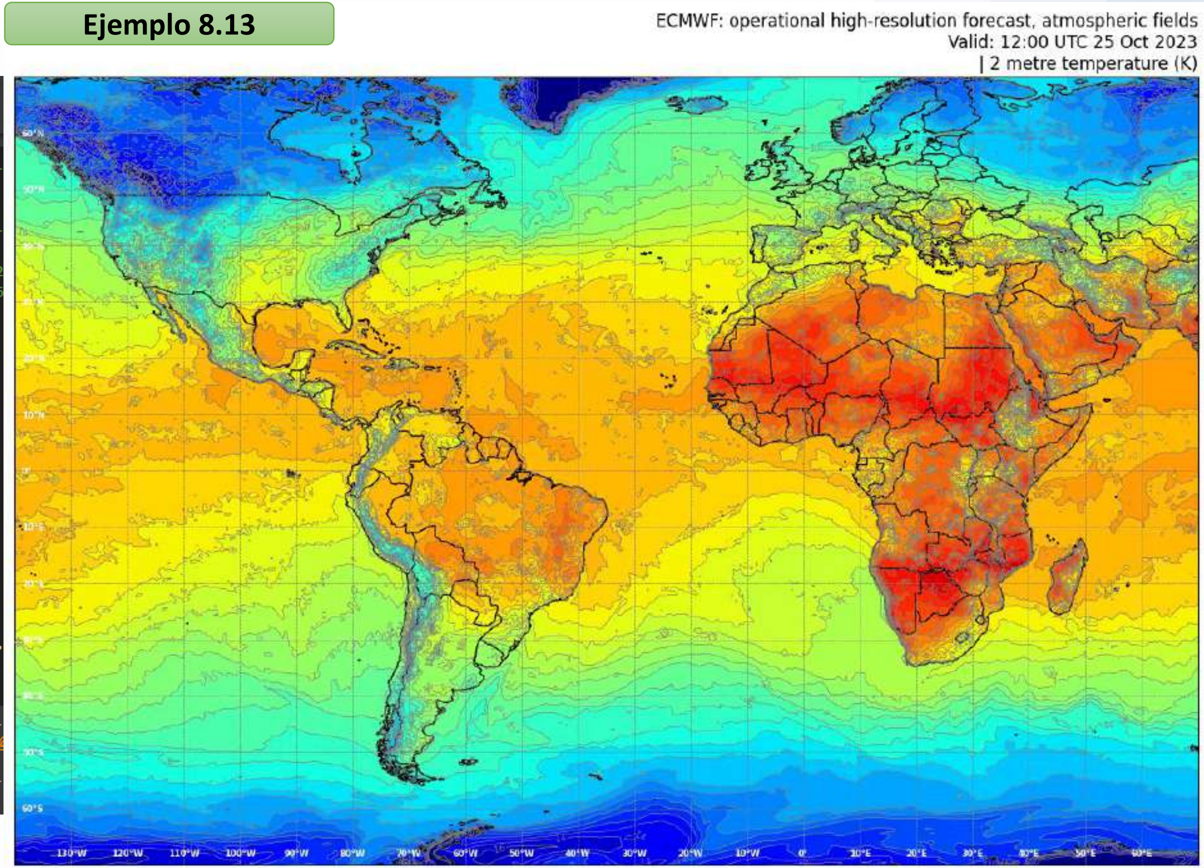
Ejemplo 8: Descargando y Graficando Datos de Modelo

Ejemplo 8.13

Example 8.13: 2m Temperature

```
#-----------------------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example 8: Downloading
# Author: Diego Souza (INPE/CGCT/DISSM)
#-----------------------------------------------------------------------------------

# open ecmwf data available for download since January 21 2022 00:00 ('2022
# gfs data available for download since February 26 2021 00:00 ('2021-02-26
# gfs products: 'pgrb2.0p25', 'pgrb2.0p50', 'pgrb2.1p00'

# 2 m temperature
plot_nwp(model='ecmwf',
         product='oper',
         date_nwp='2023-10-25 12:00',
         area='custom',
         extent=[-140.0, -70.00, 70.00, 70.00],
         fxx='date',
         var='2t', scale=1, offset=-273.15,
         level_hpa='',
         level_min=-20, level_max=48, level_int=2,
         plot_type='c_filled', contour_color='gray',
         apply_cmap=True, cmap='jet',
         view_clabel=True, fontsize=6,
         linewidth=0.5, linestyle='solid',
         land_ocean=False, land_color = 'lightgray', ocean_color = 'white',
         alpha=1.0, title='new',
         figsize=[15,15])
```

```
-----------------------------------------------------------------------------------
✅ Found ¦ model=ecmwf ¦ product=oper ¦ 2023-Oct-25 00:00 UTC F12 ¦ GRIB2 @
Total processing time (nwp): 14.17 seconds.
-----------------------------------------------------------------------------------
'output/ecmwf_t2m_2023-10-25_12:00.png'
```



Ejemplo 8: Descargando y Graficando Datos de Modelo

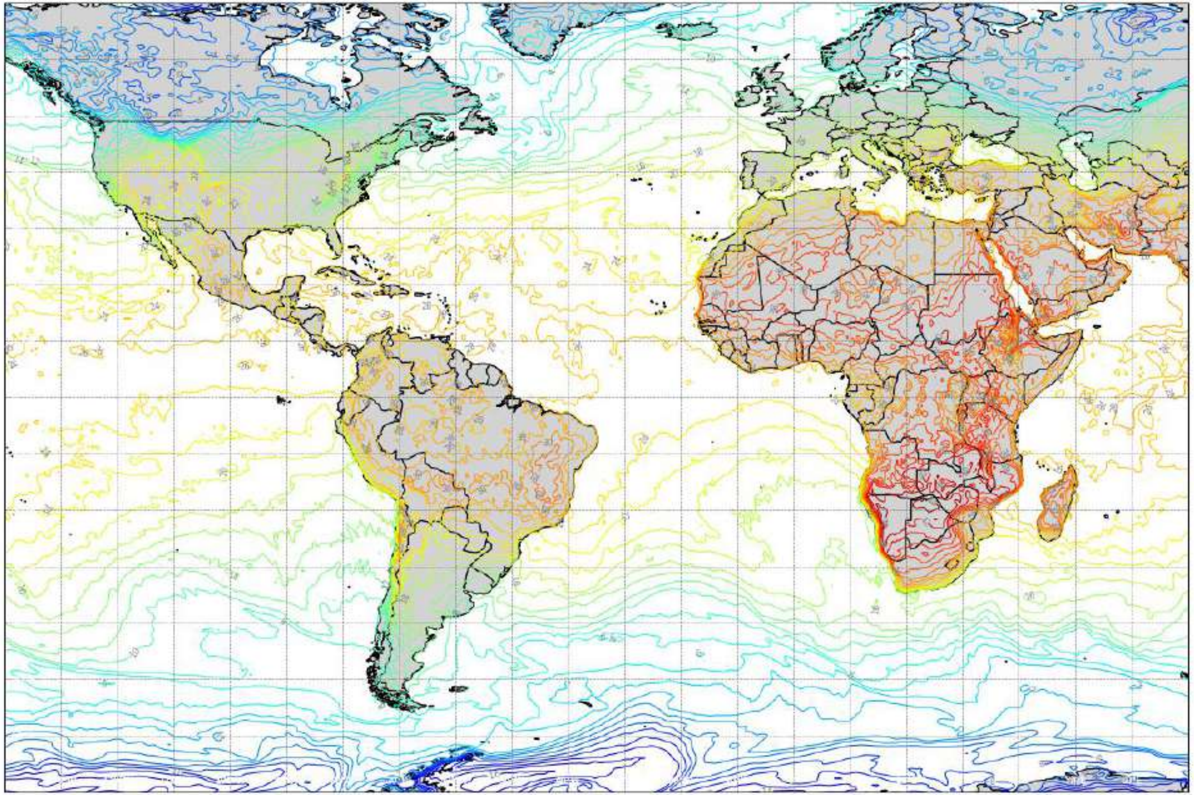
Ejemplo 8.14

Example 8.14: Temperature

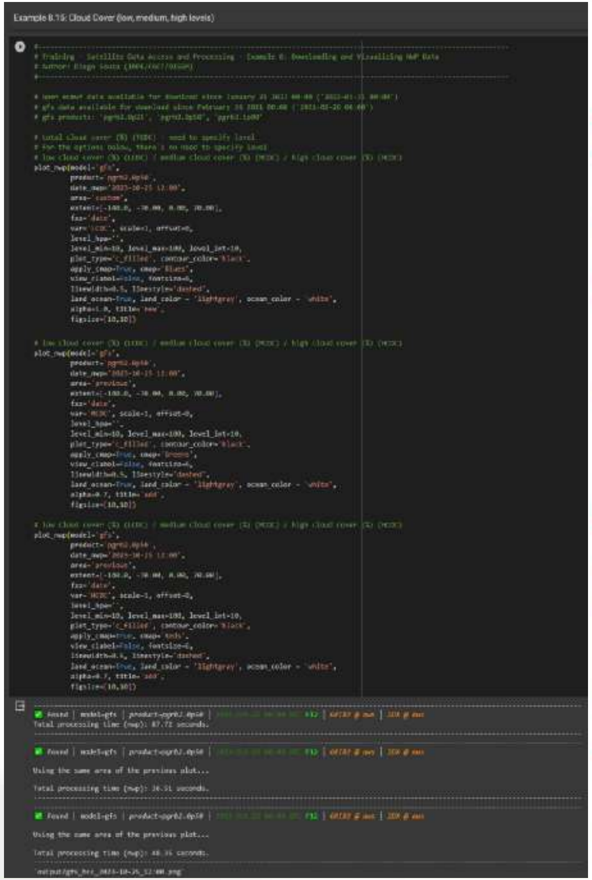
```
#-----------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example 8: Downloading a
# Author: Diego Souza (INPE/CGCT/DISSM)
#-----------------------------------------------------------------------

# open ecmwf data available for download since January 21 2022 00:00 ('2022-
# gfs data available for download since February 26 2021 00:00 ('2021-02-26
# gfs products: 'pgrb2.0p25', 'pgrb2.0p50', 'pgrb2.1p00'

# temperature (1000, 925, 950, 700, 600, 500, 300 and 250 hPa)
plot_nwp(model='ecmwf',
         product='oper',
         date_nwp='2023-10-25 12:00',
         area='custom',
         extent=[-140.0, -70.00, 70.00, 70.00],
         fxx='date',
         var='t', scale=1, offset=-273.15,
         level_hpa='1000',
         level_min=-20, level_max=48, level_int=2,
         plot_type='c_unfilled', contour_color='gray',
         apply_cmap=True, cmap='jet',
         view_clabel=True, fontsize=6,
         linewidth=0.8, linestyle='solid',
         land_ocean=True, land_color = 'lightgray', ocean_color = 'white',
         alpha=1.0, title='new',
         figsize=[15,15])
```

```
-----------------------------------------------------------------------
✅ Found ┆ model=ecmwf ┆ product=oper ┆ 2023-Oct-25 00:00 UTC F12 ┆ GRIB2 @
Total processing time (nwp): 7.4 seconds.
-----------------------------------------------------------------------
'output/ecmwf_t_2023-10-25_12:00.png'
```

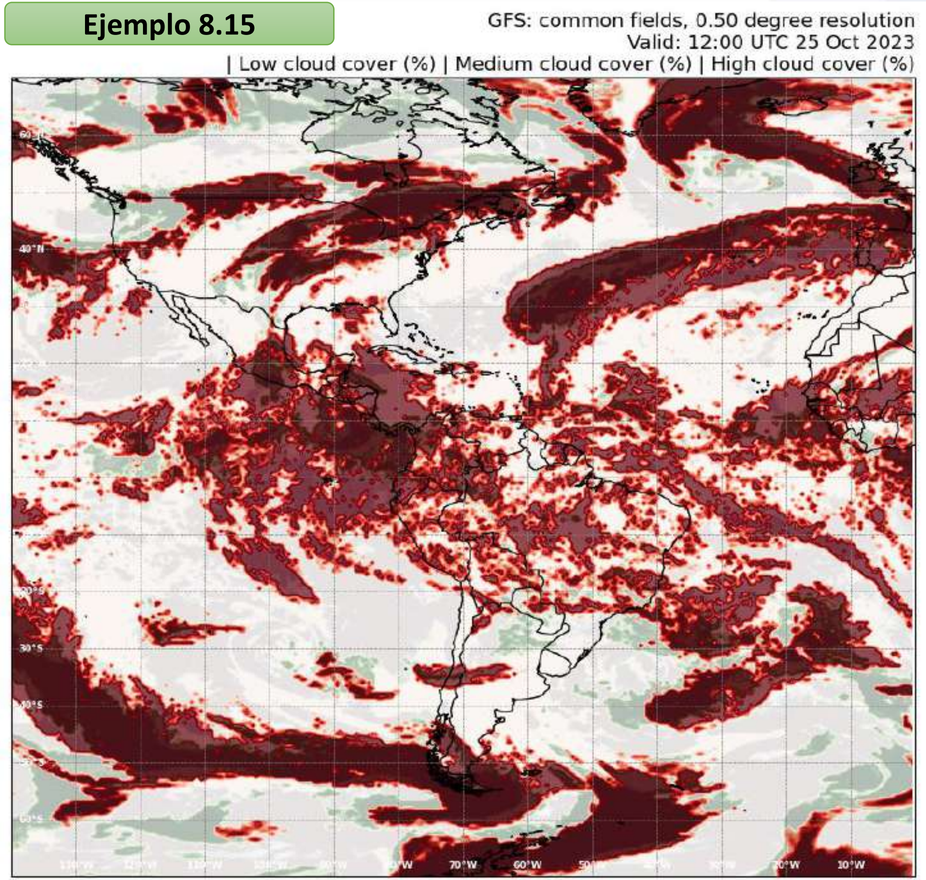
ECMWF: operational high-resolution forecast, atmospheric fields
Valid: 12:00 UTC 25 Oct 2023
| Temperature (°C) 1000.0 hPa



Ejemplo 8: Descargando y Graficando Datos de Modelo



Ejemplo 8.15



Ejemplo 9: Múltiples Variables

Ejemplo 9: Múltiples Variables de Modelo

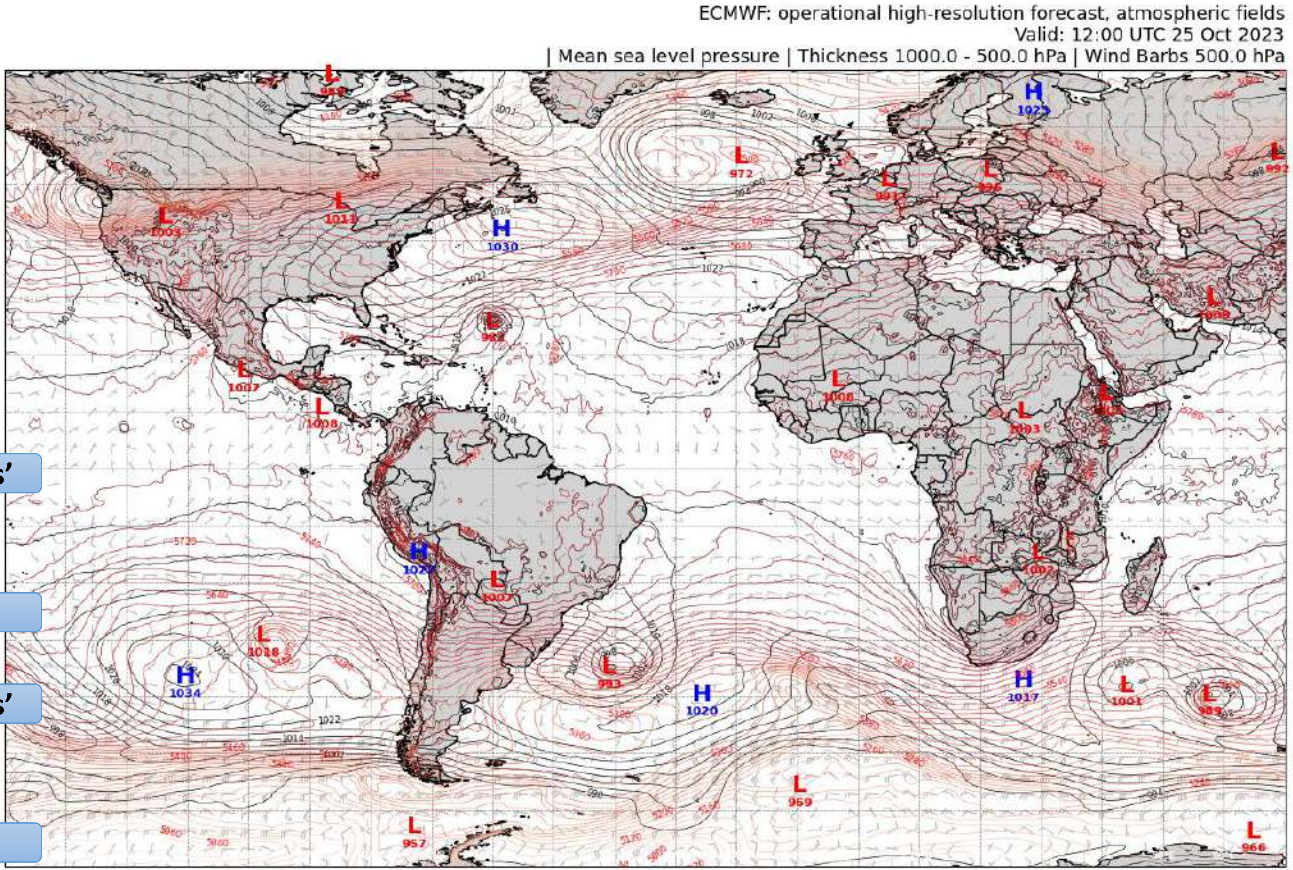
```
#-----------------------------------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example 9: Downloading and Visualizing NWP Data
# Author: Diego Souza (INPE/CGCT/DISSM)
#-----------------------------------------------------------------------------------------------

# open ecmwf data available for download since January 21 2022 00:00 ('2022-01-21 00:00')
# gfs data available for download since February 26 2021 00:00 ('2021-02-26 00:00')
# gfs products: 'pgrb2.0p25', 'pgrb2.0p50', 'pgrb2.1p00'

# mean see level pressure
plot_nwp(model='ecmwf',
         product='oper',
         date_nwp='2023-10-25 12:00',
         area='custom',
         extent=[-140.0, -70.00, 70.00, 70.00],
         fxx='date',
         var='msl', scale=0.01, offset=0,
         level_hpa='',
         level_min=990, level_max=1050, level_int=4,
         plot_type='c_unfilled', contour_color='black',
         apply_cmap=False, cmap='jet',
         view_clabel=True, fontsize=6,
         linewidth=0.5, linestyle='solid',
         land_ocean=True, land_color = 'lightgrey', ocean_color = 'white',
         alpha=1.0, title='new',
         figsize=[15,15])

# thickness (1000-850 hPa)
plot_nwp(model='ecmwf',
         product='oper',
         date_nwp='2023-10-25 12:00',
         area='previous',
         extent=[-140.0, -70.00, 70.00, 70.00],
         fxx='date',
         var='thickness', scale=1, offset=0,
         level_hpa='1000,500',
         level_min=4900, level_max=5900, level_int=20,
         plot_type='c_unfilled', contour_color='red',
         apply_cmap=True, cmap='Reds',
         view_clabel=True, fontsize=6,
         linewidth=0.5, linestyle='solid',
         land_ocean=True, land_color = 'lightgrey', ocean_col
         alpha=1.0, title='add',
         figsize=[15,15])

# wind barbs (250, 300, 500, 700, 850 hPa)
plot_nwp(model='ecmwf',
         product='oper',
         date_nwp='2023-10-25 12:00',
         area='previous',
         extent=[-140.0, -70.00, 70.00, 70.00],
         fxx='date',
         var='winds', scale=1, offset=0,
         level_hpa='500',
         level_min=0, level_max=60, level_int=5,
         plot_type='barbs', contour_color='gray',
         apply_cmap=False, cmap='jet',
         view_clabel=True, fontsize=6,
         linewidth=0.5, linestyle='solid',
         land_ocean=True, land_color = 'lightgrey', ocean_col
         alpha=0.5, title='add',
         figsize=[15,15])
```

area = ‘previous’

title = ‘add’

area = ‘previous’

title = ‘add’

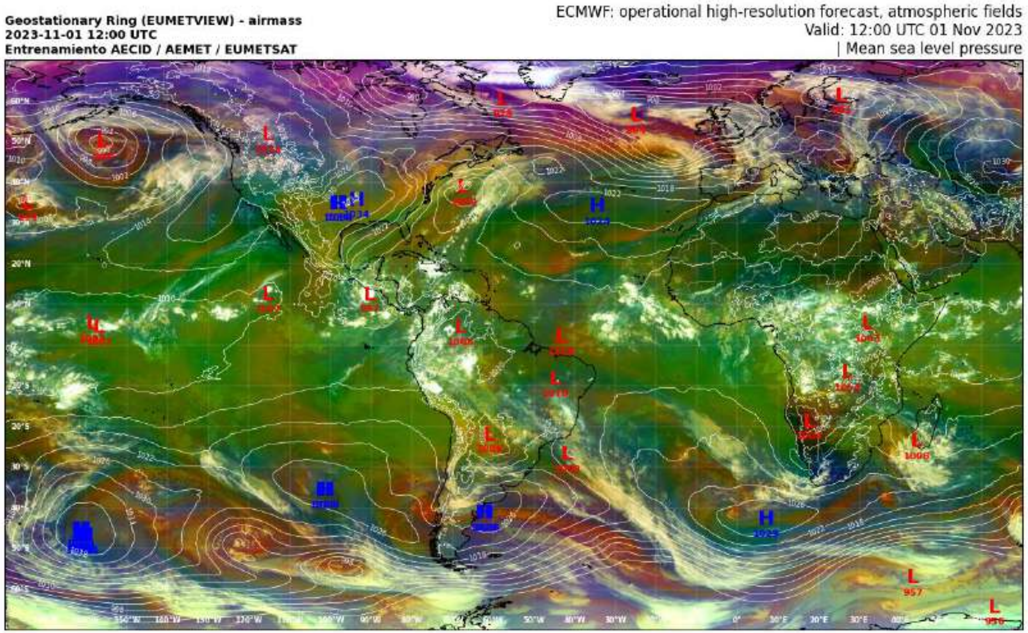


Ejemplo 10: Datos de Satélites (Multimisión) y Modelos

Ejemplo 10: Datos de Satélites (Multimisión) con Datos de Modelo

```
#-----------------------------------------------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example 10: Geostationary Ring and NWP Data
# Author: Diego Souza (INPE/CGCT/DISSM)
#-----------------------------------------------------------------------------------------------------------

# georing available since June 06 2021 18:00 UTC '2021-06-06 18:00' (note: data available every 3 hours)
# available composites: 'ir108', 'airmass', 'natural_color', 'ash', 'dust'
# max. res: 6
plot_ring(date_sat='2023-11-01 12:00',
          composite='airmass',
          resolution=12,
          area='custom',
          extent=[-180.0, -70.00, 70.00, 70.00],
          coast_color='black',
          countries_color='gray',
          grid_color='white',
          overwrite=True,
         figsize=[15,15])

# open ecmwf data available for download since January 21 2022 00:00 ('2022-01-21 00:00')
# gfs data available for download since February 26 2021 00:00 ('2021-02-26 00:00')
# gfs products: 'pgrb2.0p25', 'pgrb2.0p50', 'pgrb2.1p00'

# mean sea level pressure
plot_nwp(model='ecmwf',
         product='oper',
         date_nwp='2023-11-01 12:00',
         area='previous',
         extent=[-180.0, -70.00, 70.00, 70.00],
         fxx='date',
         var='msl', scale=0.01, offset=0,
         level_hpa='',
         level_min=990, level_max=1050, level_int=4,
         plot_type='c_unfilled', contour_color='white',
         apply_cmap=False, cmap='jet',
         view_clabel=True, fontsize=6,
         linewidth=0.5, linestyle='solid',
         land_ocean=False, land_color = 'lightgray', ocean_color = 'white',
         alpha=1.0, title='new',
         figsize=[15,15])
```

area = 'previous'

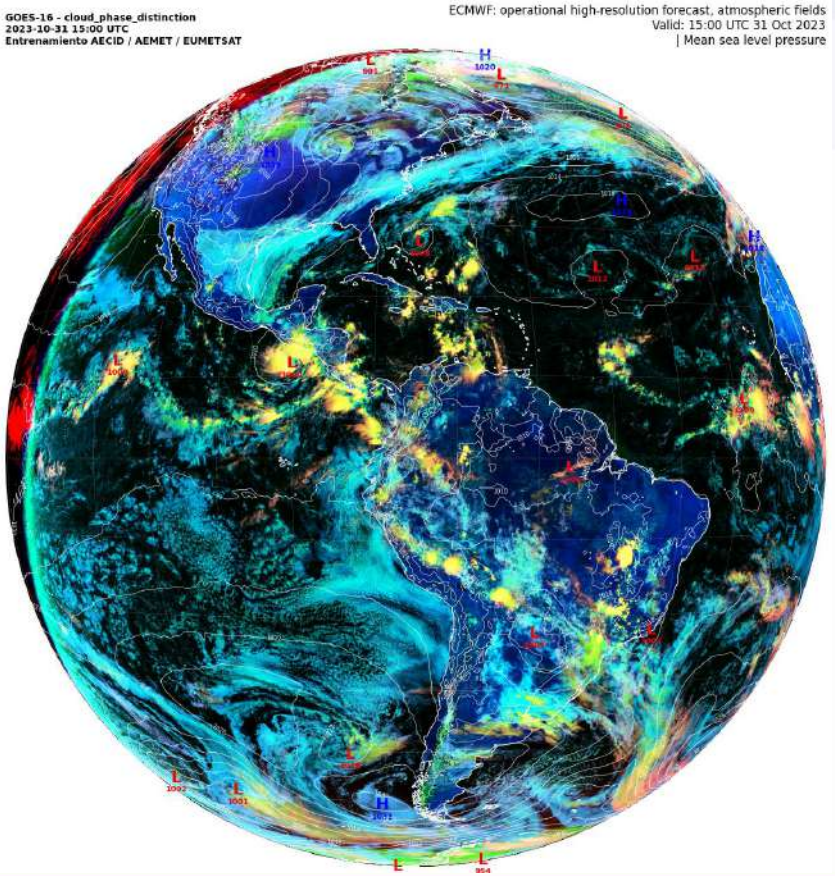


Ejemplo 11: Datos de Satélites (GOES-Este) y Modelos

Ejemplo 11: Datos de Satélites (GOES-Este) con Datos de Modelo

```
#-----------------------------------------------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example 11: GOES-East and NWP Data
# Author: Diego Souza (INPE/CGCT/DISSM)
#-----------------------------------------------------------------------------------------------------------

# 'goes16' data available since July 10 2017, 'goes17' since August 28 2018, 'goes18' since August 02 2022
# note: please consider that goes-16 generated images every 15 minutes (instead of 10) until a few months after launc

# available composites for GOES-R (the options below have been tested):
# 'C01, 'C02', 'C03', 'C04', 'C05', 'C06', 'C07', 'C08', 'C09', 'C10', 'C11', 'C12', 'C13', 'C14', 'C15', 'C16',
# 'airmass', 'ash', 'cimss_cloud_type', 'cimss_true_color', 'cimss_true_color_sunz_rayleigh', 'cira_day_convection',
# 'cloudtop', 'color_infrared', 'colorized_ir_clouds', 'convection', 'day_microphysics', 'dust', 'fire_temperature_aw
# 'night_microphysics', 'overview', 'rocket_plume_day', 'rocket_plume_night', 'snow', 'so2', 'tropical_airmass', 'tru

plot_sat(sat='goes16',
         date_sat='2023-10-31 15:00',
         composite='cloud_phase_distinction',
         area='full_disk',
         extent=[-90.0, -60.00, -30.00, 15.00],
         coast_color='white',
         countries_color='white',
         grid_color='white',
         figsize=[15,15])

# open ecmwf data available for download since January 21 2022 00:00 ('2022-01-21 00:00')
# gfs data available for download since February 26 2021 00:00 ('2021-02-26 00:00')
# gfs products: 'pgrb2.0p25', 'pgrb2.0p50', 'pgrb2.1p00'

# mean sea level pressure
plot_nwp(model='ecmwf',
         product='oper',
         date_nwp='2023-10-31 15:00',
         area='previous',
         extent=[-180.0, -70.00, 70.00, 70.00],
         fxx='date',
         var='msl', scale=0.01, offset=0,
         level_hpa='',
         level_min=990, level_max=1050, level_int=4,
         plot_type='c_unfilled', contour_color='white',
         apply_cmap=False, cmap='jet',
         view_clabel=True, fontsize=6,
         linewidth=0.5, linestyle='solid',
         land_ocean=False, land_color = 'lightgray', ocean_color = 'white',
         alpha=1.0, title='new',
         figsize=[15,15])
```

area = ‘previous’

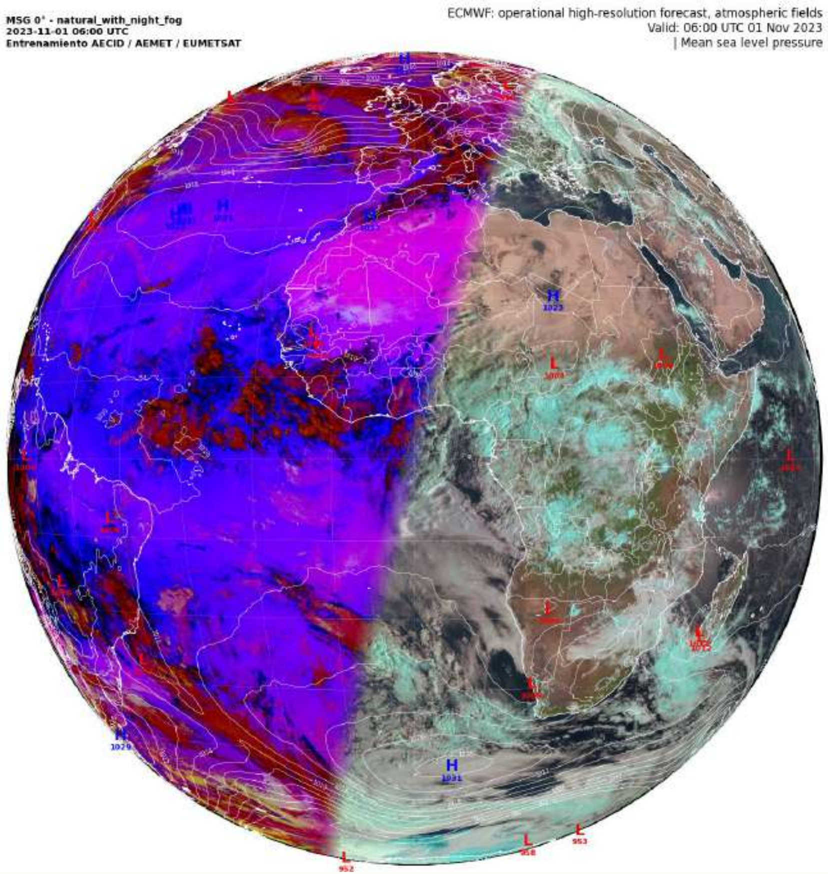


Ejemplo 12: Datos de Satélites (MSG 0°) y Modelos

Ejemplo 12: Datos de Satélites (MSG 0°) con Datos de Modelo

```
#-----------------------------------------------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example 12: MSG 0° and NWP Data
# Author: Diego Souza (INPE/CGCT/DISSM)
#-----------------------------------------------------------------------------------------------------------

# 'msg0' tested with data since March 26 2004 (every 15 minutes)

# select the composite (the options below have been tested):
# 'VIS006', 'VIS008', 'IR_016', 'IR_039', 'WV_062', 'WV_073', 'IR_087', 'IR_097', 'IR_108', 'IR_120', 'IR_134'
# 'airmass', 'ash', 'cloudtop', 'cloudtop_daytime', 'colorized_ir_clouds', 'convection', 'day_microphysics' 'day_micr
# 'natural_color', 'natural_color_nocorr', 'natural_color_raw', 'natural_color_raw_with_night_ir', 'natural_color_wit
# 'night_ir_alpha', 'night_microphysics', 'overview', 'overview_raw', 'rocket_plume_night', 'snow'

plot_sat(sat='msg0',
         date_sat='2023-11-01 06:00',
         composite='natural_with_night_fog',
         area='full_disk',
         extent=[-90.0, -60.00, -30.00, 15.00],
         coast_color='white',
         countries_color='white',
         grid_color='white',
         figsize=[15,15])

# open ecmwf data available for download since January 21 2022 00:00 ('2022-01-21 00:00')
# gfs data available for download since February 26 2021 00:00 ('2021-02-26 00:00')
# gfs products: 'pgrb2.0p25', 'pgrb2.0p50', 'pgrb2.1p00'

# mean sea level pressure
plot_nwp(model='ecmwf',
         product='oper',
         date_nwp='2023-11-01 06:00',
         area='previous',
         extent=[-180.0, -70.00, 70.00, 70.00],
         fxx='date',
         var='msl', scale=0.01, offset=0,
         level_hpa='',
         level_min=990, level_max=1050, level_int=4,
         plot_type='c_unfilled', contour_color='white',
         apply_cmap=False, cmap='jet',
         view_clabel=True, fontsize=6,
         linewidth=0.5, linestyle='solid',
         land_ocean=True, land_color = 'lightgray', ocean_color = 'white',
         alpha=1.0, title='new',
         figsize=[15,15])
```

area = ‘previous’



Ejemplo 13: Descargando y Graficando Datos METAR

Ejemplo 13: Descargando Datos METAR desde Unidata y Graficando la Imagen

```python
#-----------------------------------------------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example 13: METAR Data Access and Visualization
# Author: Diego Souza (INPE/CGCT/DISSM)
#-----------------------------------------------------------------------------------------------------------

# metar data available for the last 10 days, for each hour

# metar plot
plot_metar(date_metar='2023-10-25 12:00',
           area='custom',
           extent= [-90.0, -60.00, -30.00, 15.00],
           radius=3, land_ocean=True, land_color = 'beige', ocean_color = 'lightblue',
           figsize=[15,15])
```

```
-------------------------------------------------------------------------------------------------------------
Downloading the following METAR file from UNIDATA THREDDS Data Server: metar_20231025_1200.txt

Plotting the METAR file: metar_20231025_1200.txt

Total processing time (metar): 20.24 seconds.
-------------------------------------------------------------------------------------------------------------
'output/metar_2023-10-25_12:00.png'
```

Ejemplo 14: Satélites, Modelos y METAR

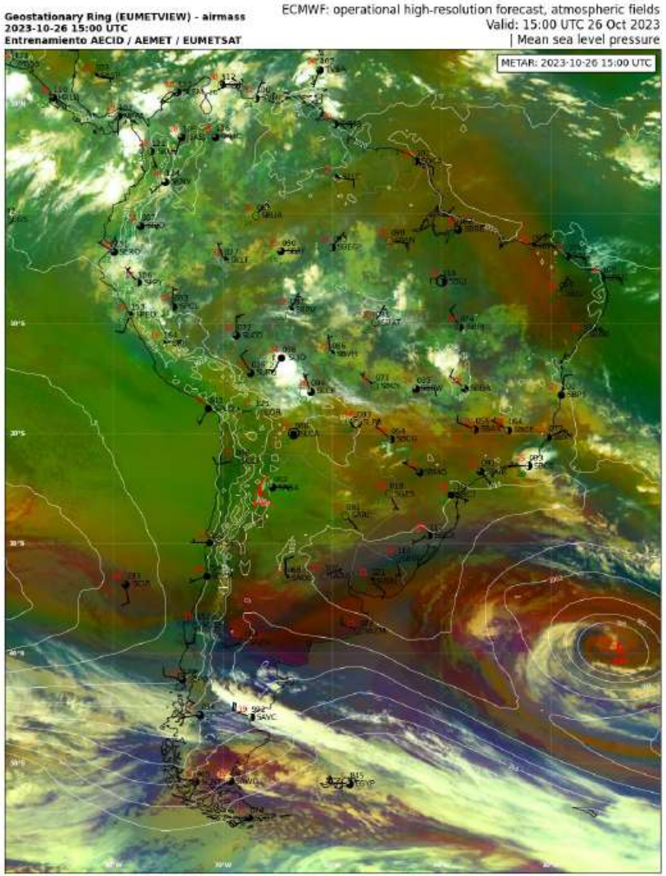
Ejemplo 14: Datos de Satélites, Modelos y METAR

Ejemplo 14.1

```
#-----------------------------------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example 14.1: Satellite, NWP and METAR
# Author: Diego Souza (INPE/CGCT/DISSM)
#-----------------------------------------------------------------------------------------------

# georing available since June 06 2021 18:00 UTC '2021-06-06 18:00' (note: data available every 3 hours)
# available composites: 'ir108', 'airmass', 'natural_color', 'ash', 'dust'
# max. res: 6
plot_ring(date_sat='2023-10-26 15:00',
          composite='airmass',
          resolution=6,
          area='custom',
          extent= [-90.0, -60.00, -30.00, 15.00],
          coast_color='black',
          countries_color='gray',
          grid_color='white',
          overwrite=True,
          figsize=[15,15])

# open ecmwf data available for download since January 21 2022 00:00 ('2022-01-21 00:00')
# gfs data available for download since February 26 2021 00:00 ('2021-02-26 00:00')
# gfs products: 'pgrb2.0p25', 'pgrb2.0p50', 'pgrb2.1p00'

# mean sea level pressure
plot_nwp(model='ecmwf',
         product='oper',
         date_nwp='2023-10-26 15:00',
         area='previous',
         extent=[-90.0, -60.00, -30.00, 15.00],
         fxx='date',
         var='msl', scale=0.01, offset=0,
         level_hpa='',
         level_min=990, level_max=1050, level_int=4,
         plot_type='c_unfilled', contour_color='white',
         apply_cmap=False, cmap='jet',
         view_clabel=True, fontsize=6,
         linewidth=0.5, linestyle='solid',
         land_ocean=False, land_color = 'lightgray', ocean_color = 'white',
         alpha=1.0, title='new',
         figsize=[15,15])

# metar data available for the last 10 days, for each hour

# metar plot
plot_metar(date_metar='2023-10-26 15:00',
           area='previous',
           extent= [-90.0, -60.00, -30.00, 15.00],
           radius=3, land_ocean=False, land_color = 'lightgray', ocean_color = 'white',
           figsize=[15,15])
```

area = 'previous'

area = 'previous'



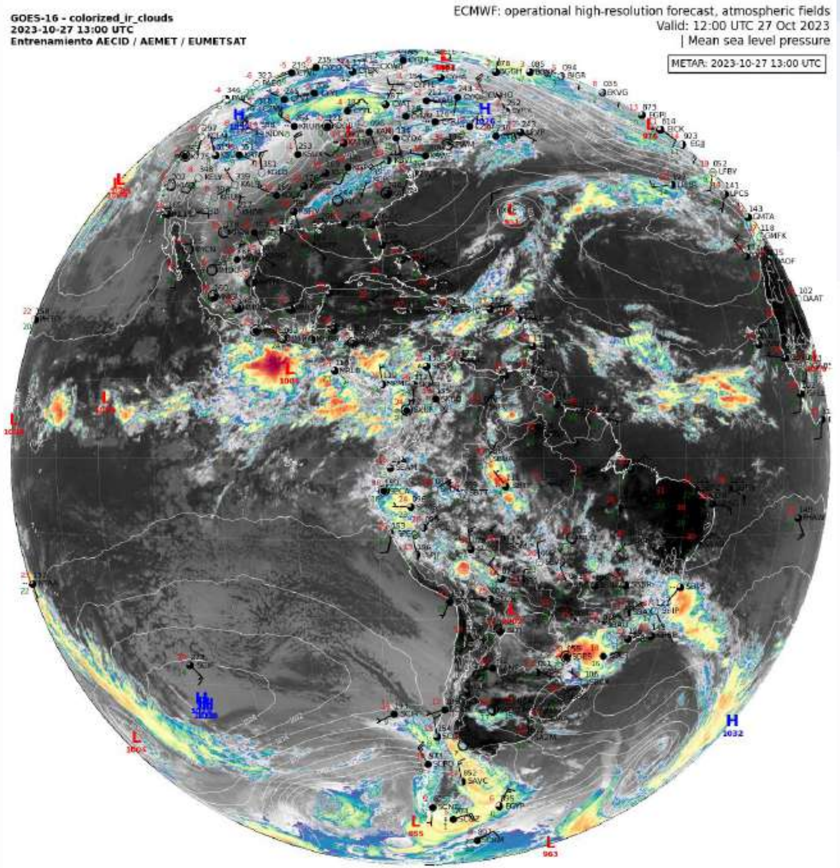
Ejemplo 14: Satélites, Modelos y METAR

Ejemplo 14.2

```
#-----------------------------------------------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example 13.1: Satellite, NWP and METAR
# Author: Diego Souza (INPE/CGCT/DISSM)
#-----------------------------------------------------------------------------------------------------------

# goes-16
plot_sat(sat='goes16',
         date_sat='2023-10-27 13:00',
         composite='colorized_ir_clouds',
         area='full_disk',
         extent=[-90.0, -60.00, -30.00, 15.00],
         coast_color='white',
         countries_color='white',
         grid_color='white',
         figsize=[15,15])

# mean sea level pressure
plot_nwp(model='ecmwf',
         product='oper',
         date_nwp='2023-10-27 13:00',
         area='previous',
         extent=[-110.0, 0.00, -60.00, 30.00],
         fxx='date',
         var='msl', scale=0.01, offset=0,
         level_hpa='',
         level_min=990, level_max=1050, level_int=4,
         plot_type='c_unfilled', contour_color='white',
         apply_cmap=False, cmap='jet',
         view_clabel=True, fontsize=6,
         linewidth=0.5, linestyle='solid',
         land_ocean=False, land_color = 'lightgray', ocean_color = 'white',
         alpha=1.0, title='new',
         figsize=[15,15])

# metar plot
plot_metar(date_metar='2023-10-27 13:00',
           area='previous',
           extent= [-110.0, 0.00, -60.00, 30.00],
           radius=300000, land_ocean=False, land_color = 'lightgray', ocean_color = 'white',
           figsize=[15,15])
```

area = 'previous'

area = 'previous'



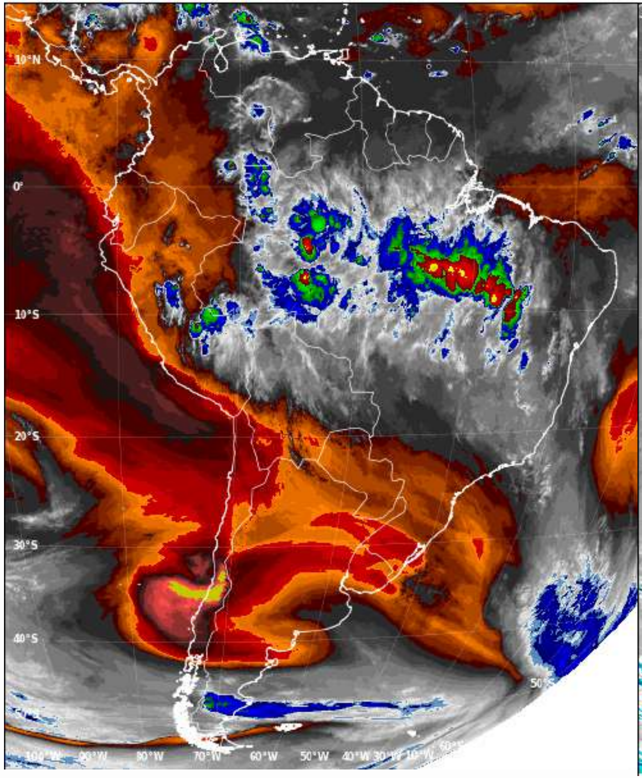
Ejemplo 15: Animaciones (Satélites)

Ejemplo 15: Creando Animaciones

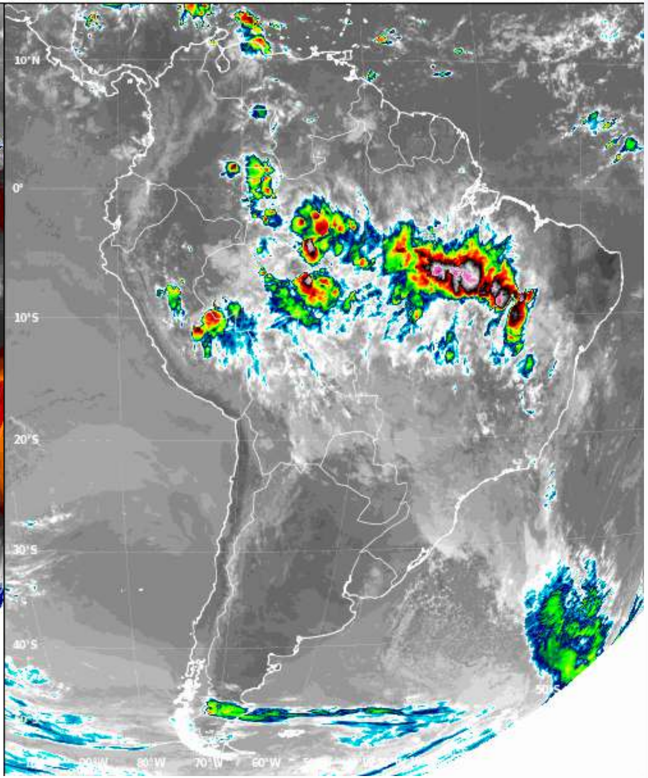
Ejemplo 15.1 Animaciones con Datos de Satélites

```python
#-----------------------------------------------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example: Creating Animations
# Author: Diego Souza (INPE/CGCT/DISSM)
#-----------------------------------------------------------------------------------------------------------

# start date
date_ini = '2022-11-01 12:00'

# end date
date_end = '2022-11-01 15:00'

# interval (minutes)
interval = 20

# convert to datetime objects
date_ini = datetime(int(date_ini[0:4]), int(date_ini[5:7]), int(date_ini[8:10]), int(date_ini[11:13]), int(date_ini[14:16]))
date_end = datetime(int(date_end[0:4]), int(date_end[5:7]), int(date_end[8:10]), int(date_end[11:13]), int(date_end[14:16]))

# create our references for the loop
date_loop = date_ini

# list that will store images
images = []

#-----------------------------------------------------------------------------------------------------------

# loop between dates
while (date_loop <= date_end):

  # convert date time to string
  date = date_loop.strftime('%Y-%m-%d %H:%M')

  ##################################################
  # INSERT YOUR FUNCTION HERE WITH "DATE"
  ##################################################

  # call the function
  image = plot_sat(sat='goes16',
           date_sat=date,
           composite='C09',
           area='custom_geo',
           extent= [-90.0, -60.00, -30.00, 15.00],
           coast_color='white',
           countries_color='white',
           grid_color='white',
           figsize=[8,8])

  ##################################################

  # add the image to the list
  print(f'{image}\n')
  images.append(image)

  # increment the date_ini
  date_loop = date_loop + timedelta(minutes=interval)

#-----------------------------------------------------------------------------------------------------------

# create the GIF with imageio
files = []
for image in images:
    files.append(imageio.imread(image))
imageio.mimsave(f'output/animation_sat.gif', files, loop=0, fps=1)

# open the GIF
print("\nOpening the GIF...\n")
from IPython.display import Image
Image(open(f'output/animation_sat.gif','rb').read())
```

GOES-16 - C09 - 6.95 µm (6.75-7.15 µm)
2022-11-01 12:00 UTC
Entrenamiento AECID / AEMET / EUMETSAT



GOES-16 - C13 - 10.35 µm (10.1-10.6 µm)
2022-11-01 12:00 UTC
Entrenamiento AECID / AEMET / EUMETSAT



Ejemplo 15: Animaciones (Satélites)

1. Elegir fecha inicial, fecha final e intervalo

Ejemplo 15: Creando Animaciones

Ejemplo 15.1 Animaciones con Datos de Satélites

```
#-----------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example: Creating Animations
# Author: Diego Souza (INPE/CGCT/DISSM)
#-----------------------------------------------------------------------

# start date
date_ini = '2022-11-01 12:00'

# end date
date_end = '2022-11-01 15:00'

# interval (minutes)
interval = 20
```

2. Pegar la función en esta región

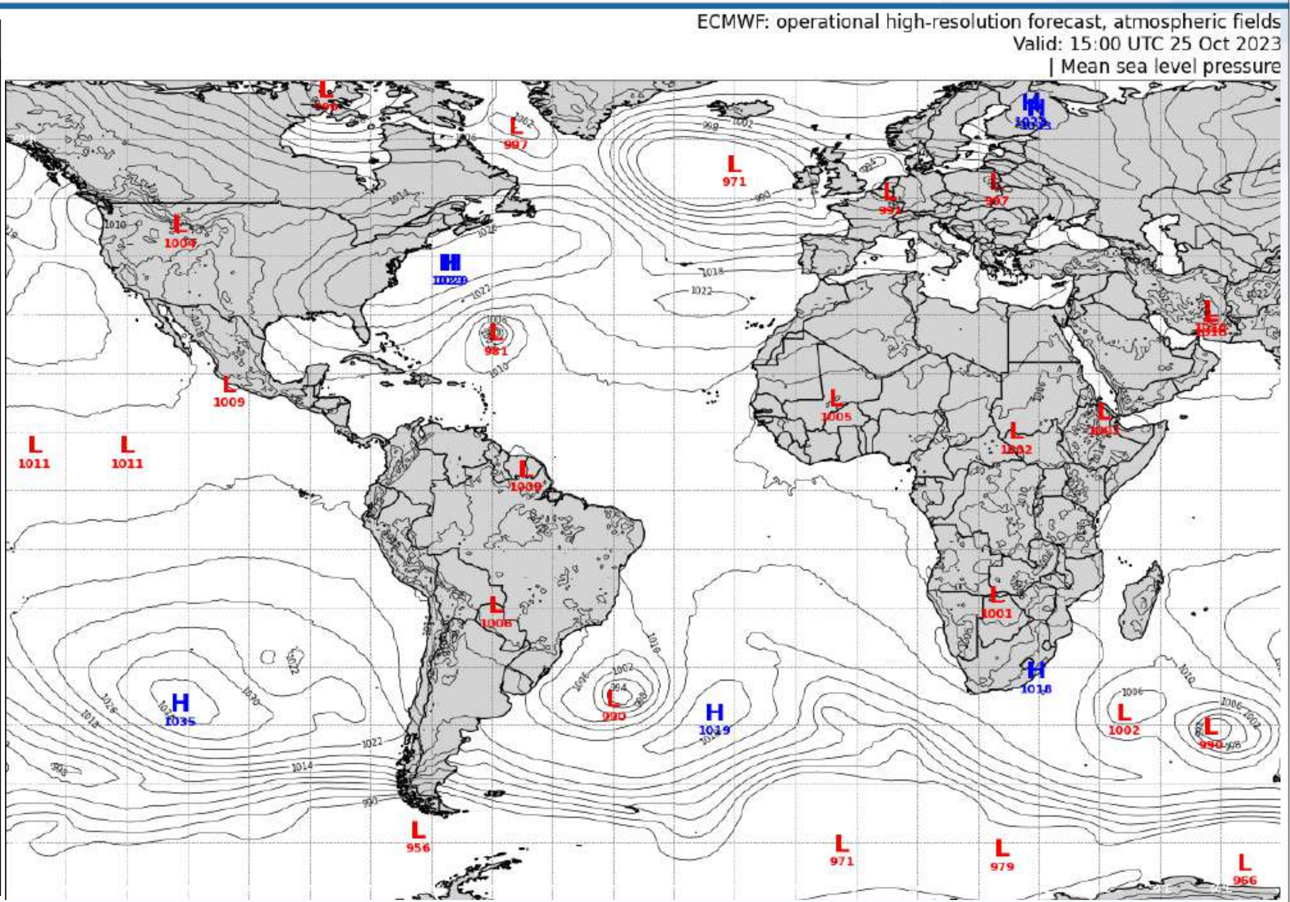
```
while (date_loop <= date_end):

    # convert date time to string
    date = date_loop.strftime('%Y-%m-%d %H:%M')

    ################################################
    # INSERT YOUR FUNCTION HERE WITH "DATE"
    ################################################

    # call the function
    image = plot_sat(sat='goes16',
                     date_sat=date,
                     composite='C09',
                     area='custom_geo',
                     extent= [-90.0, -60.00, -30.00, 15.00],
                     coast_color='white',
                     countries_color='white',
                     grid_color='white',
                     figsize=[8,8])

    ################################################

    # add the image to the list
    print(f'{image}\n')
    images.append(image)

    # increment the date_ini
    date_loop = date_loop + timedelta(minutes=interval)
```

Cambiar para 'date'

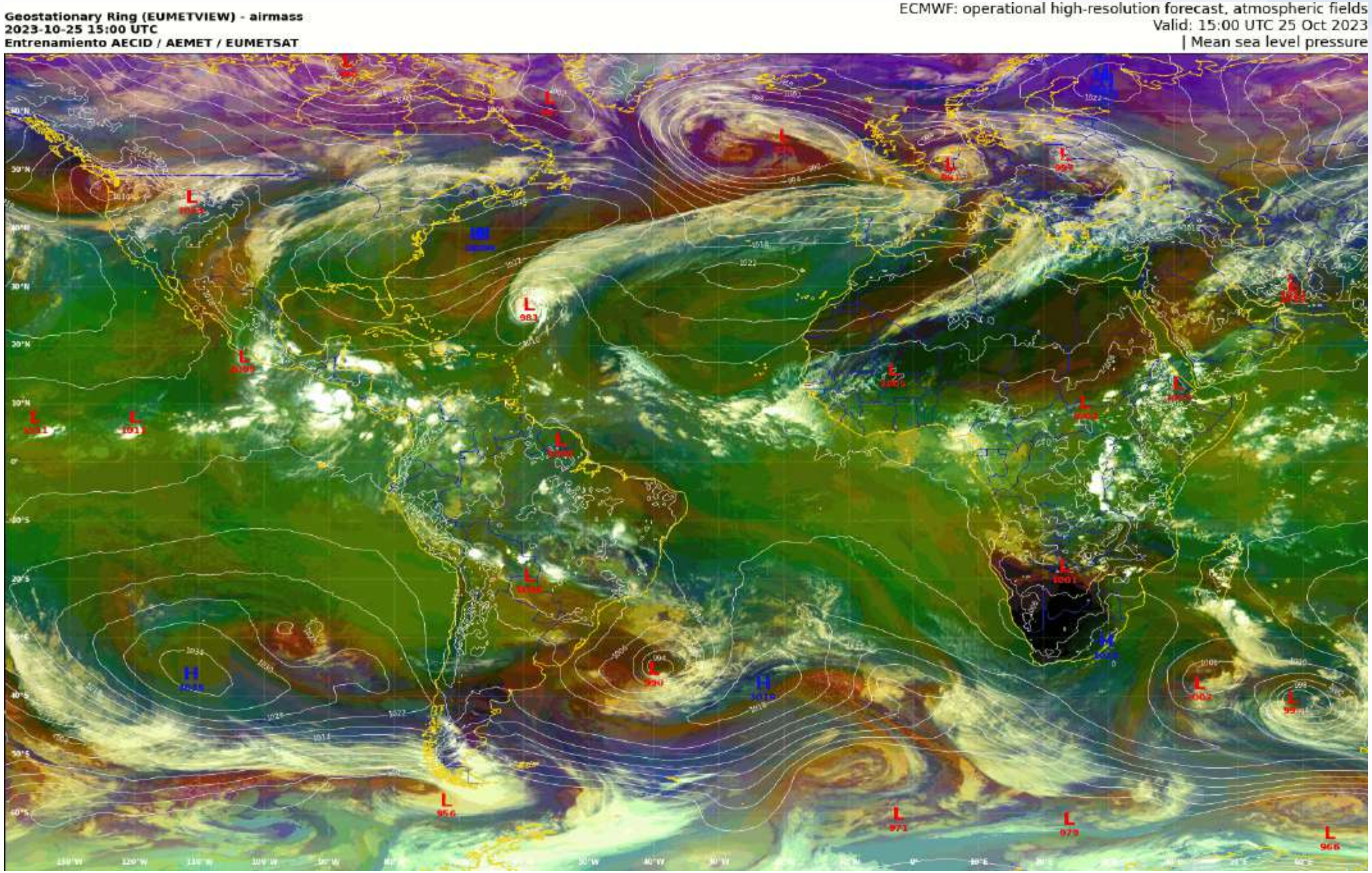
Ejemplo 15: Animaciones (Modelos)

Ejemplo 15.2 Animaciones con Datos de Modelos

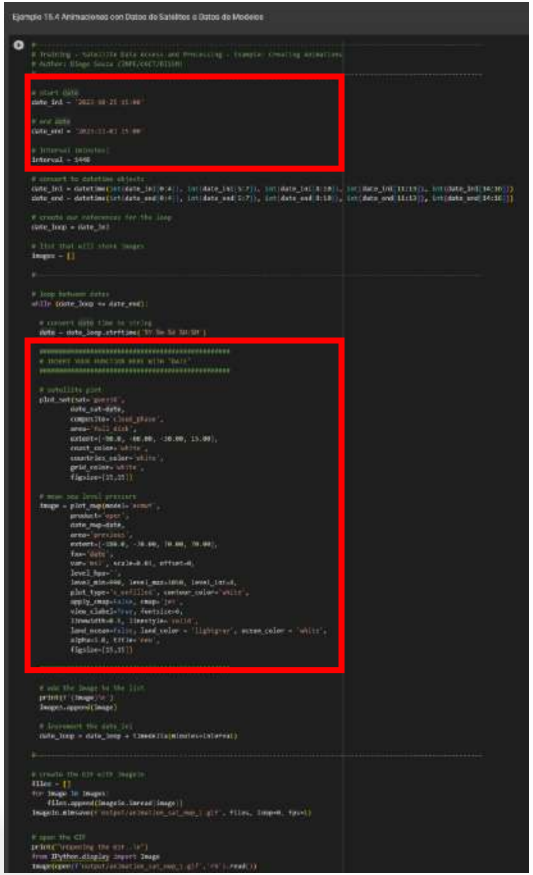
```python
#-----------------------------------------------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example: Creating Animations
# Author: Diego Souza (INPE/CGCT/DISSM)
#-----------------------------------------------------------------------------------------------------------

# start date
date_ini = '2023-10-25 15:00'

# end date
date_end = '2023-11-01 15:00'

# interval (minutes)
interval = 1440

# convert to datetime objects
date_ini = datetime(int(date_ini[0:4]), int(date_ini[5:7]), int(date_ini[8:10]), int(date_ini[11:13]), int(date_ini[14:16]))
date_end = datetime(int(date_end[0:4]), int(date_end[5:7]), int(date_end[8:10]), int(date_end[11:13]), int(date_end[14:16]))

# create our references for the loop
date_loop = date_ini

# list that will store images
images = []

#-----------------------------------------------------------------------------------------------------------

# loop between dates
while (date_loop <= date_end):

  # convert date time to string
  date = date_loop.strftime('%Y-%m-%d %H:%M')

  ##################################################################
  # INSERT YOUR FUNCTION HERE WITH "DATE"
  ##################################################################

  # call the function
  # geopotential height (500 hPa [4800-6000] and 300 hPa [8300-9700])
  # mean sea level pressure
  image = plot_nwp(model='ecmwf',
          product='oper',
          date_nwp=date,
          area='custom',
          extent=[-140.0, -70.00, 70.00, 70.00],
          fxx='date',
          var='msl', scale=0.01, offset=0,
          level_hpa='',
          level_min=990, level_max=1050, level_int=4,
          plot_type='c_unfilled', contour_color='black',
          apply_cmap=False, cmap='jet',
          view_clabel=True, fontsize=6,
          linewidth=0.5, linestyle='solid',
          land_ocean=True, land_color = 'lightgray', ocean_color = 'white',
          alpha=1.0, title='new',
          figsize=(15,15))

  ##################################################################

  # add the image to the list
  print(f'{image}\n')
  images.append(image)

  # increment the date_ini
  date_loop = date_loop + timedelta(minutes=interval)

#-----------------------------------------------------------------------------------------------------------

# create the GIF with imageio
files = []
for image in images:
    files.append(imageio.imread(image))
imageio.mimsave(f'output/animation_nwp.gif', files, loop=0, fps=1)

# open the GIF
print("\nOpening the GIF...\n")
from IPython.display import Image
Image(open(f'output/animation_nwp.gif','rb').read())
```

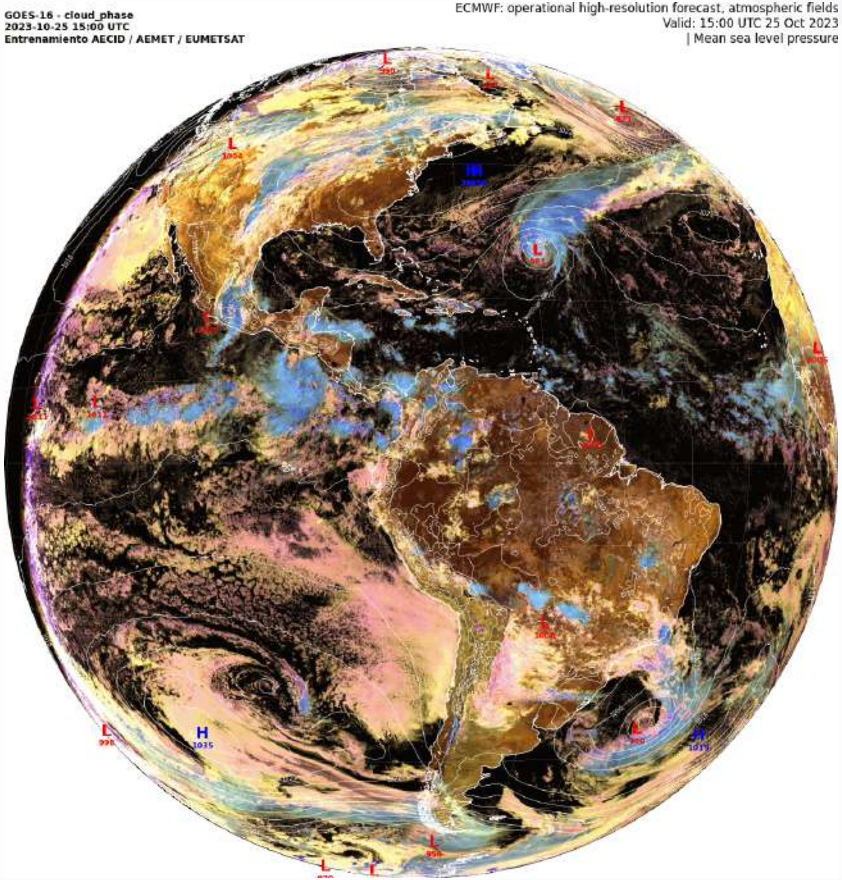


Ejemplo 15: Animaciones (Multimisión + Modelos)

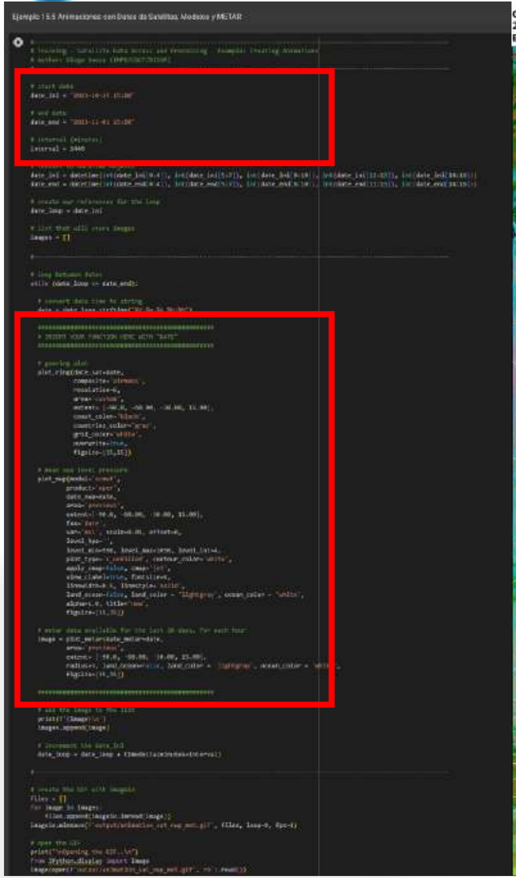


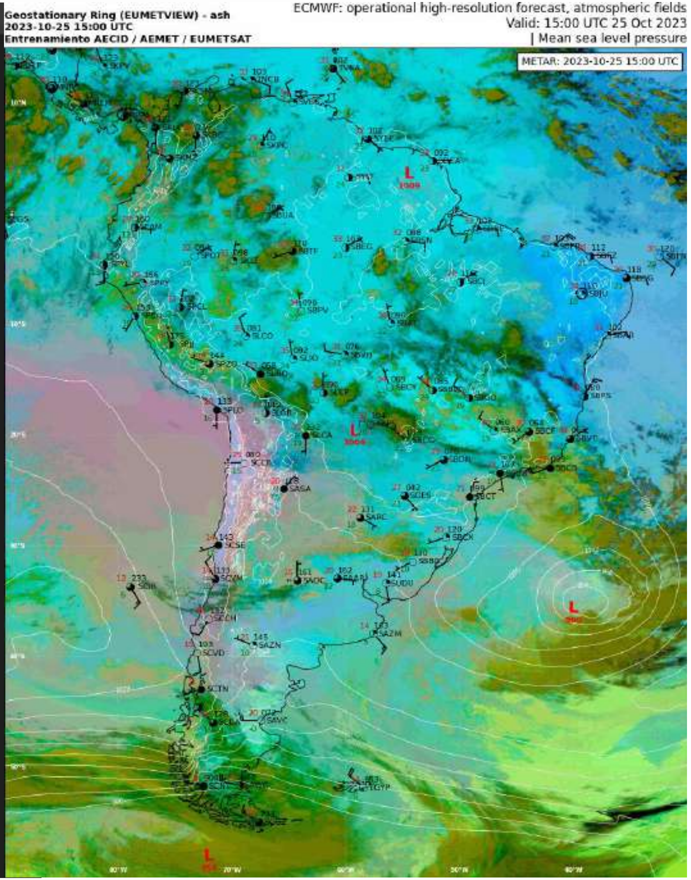
Ejemplo 15: Animaciones (Satélites + Modelos)

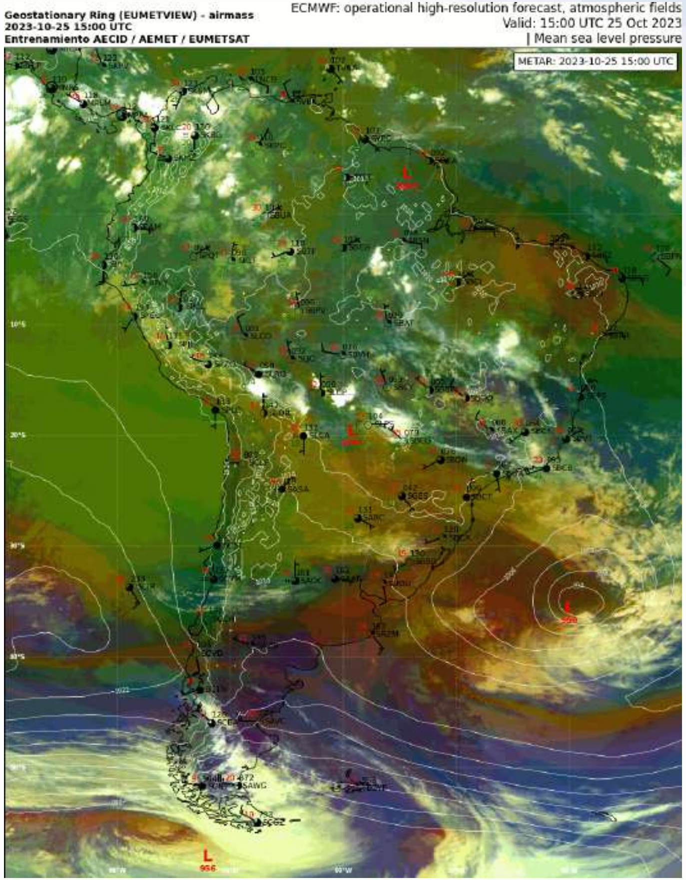




Ejemplo 15: Animaciones (Satélites + Modelos + METAR)







Productos y Aplicaciones de Satélite en Latitudes Medias - Fase Presencial

Acceso y Procesamiento de Datos Satelitales - Prácticas

¡GRACIAS! ¿PREGUNTAS?

Diego Souza
diego.souza@inpe.br

DISSM - División de Satélites y Sensores Meteorológicos
CGCT - Coordinación General de Ciencias de la Tierra
INPE - Instituto Nacional de Investigaciones Espaciales

GOVERNO FEDERAL
MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E INOVAÇÃO
UNIÃO E RECONSTRUÇÃO

Productos y Aplicaciones de Satélite en Latitudes Medias - Fase Presencial

Prácticas LSA SAF y H SAF

INPE

Diego Souza
diego.souza@inpe.br

DISSM - División de Satélites y Sensores Meteorológicos
CGCT - Coordinación General de Ciencias de la Tierra
INPE - Instituto Nacional de Investigaciones Espaciales

MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E INOVAÇÃO

GOVERNO FEDERAL BRASIL UNIÃO E RECONSTRUÇÃO

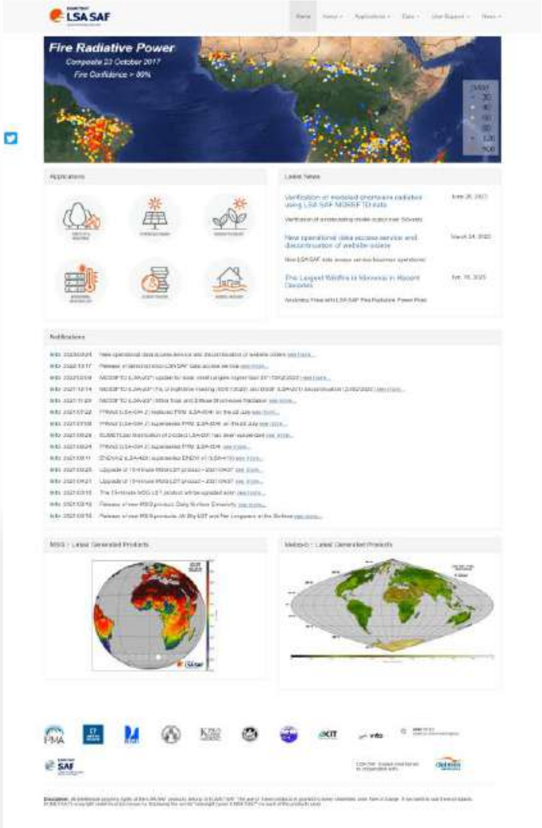
ACCESO A LOS DATOS LSA SAF (LAND SURFACE ANALYSIS)

EUMETSAT
LSA SAF
LAND SURFACE ANALYSIS

Informaciones em la Pagina LSA SAF

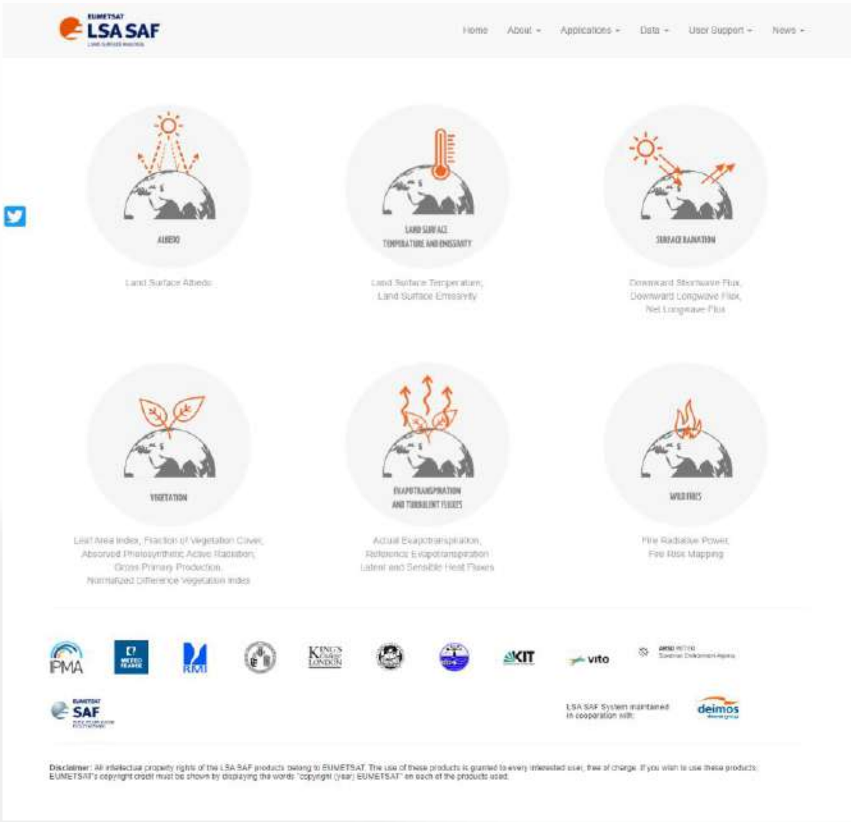
https://landsaf.ipma.pt/

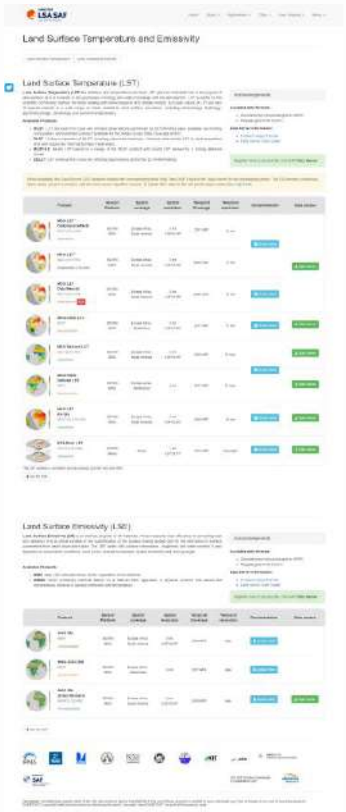
Aplicaciones, descripciones de productos y otras informaciones

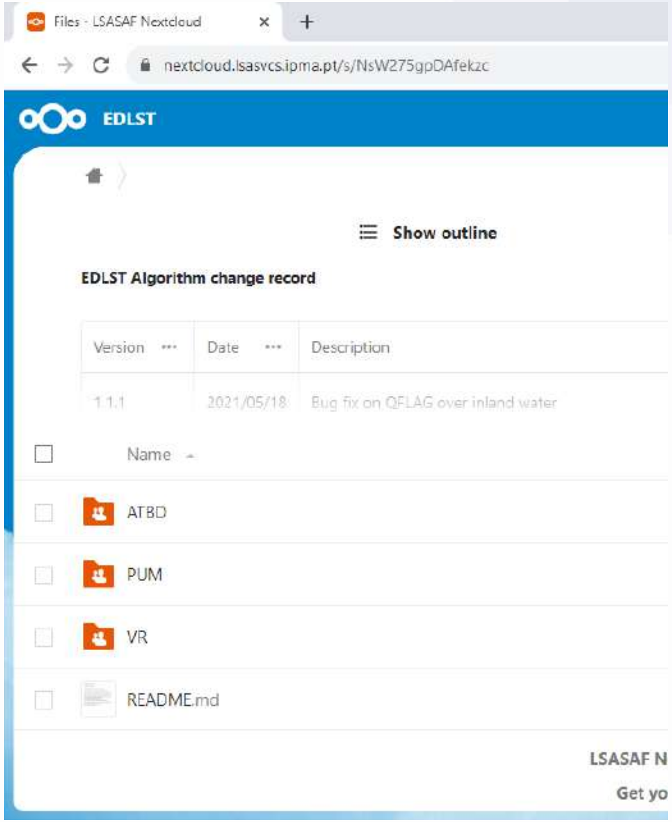


Informaciones Sobre Productos

https://landsaf.ipma.pt/en/data/products/



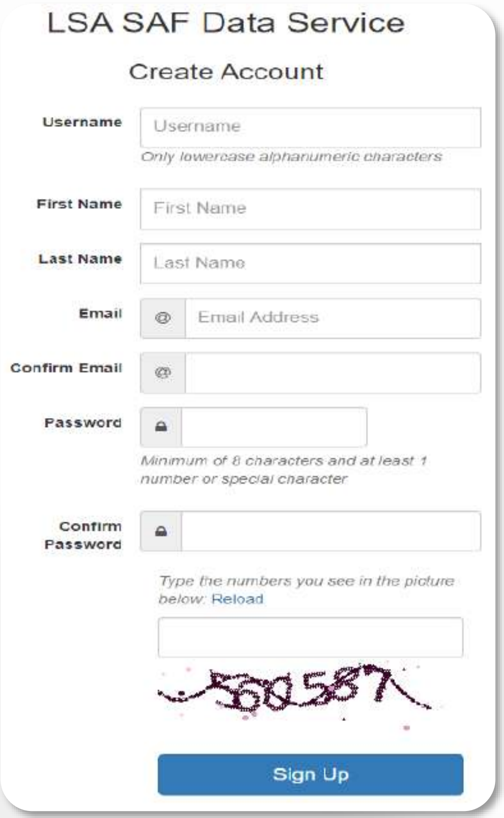


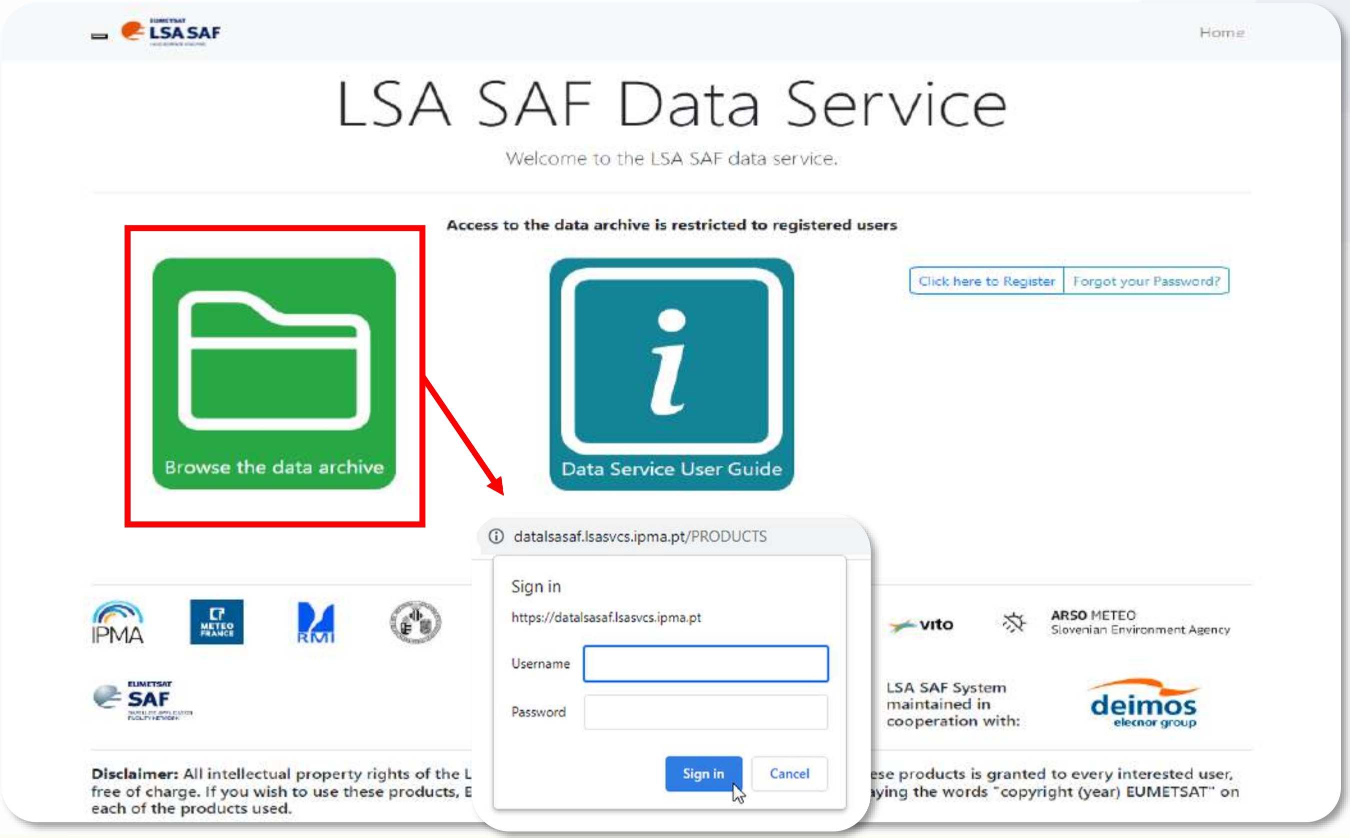


Páginas de Registro y Acceso al LSA SAF Data Service

https://mokey.lsasvcs.ipma.pt/auth/signup

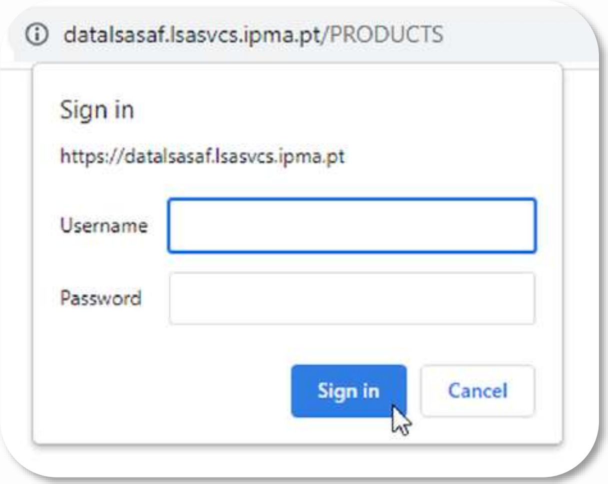
https://datalsasaf.lsasvcs.ipma.pt/





Informaciones: Usuario e Contraseña

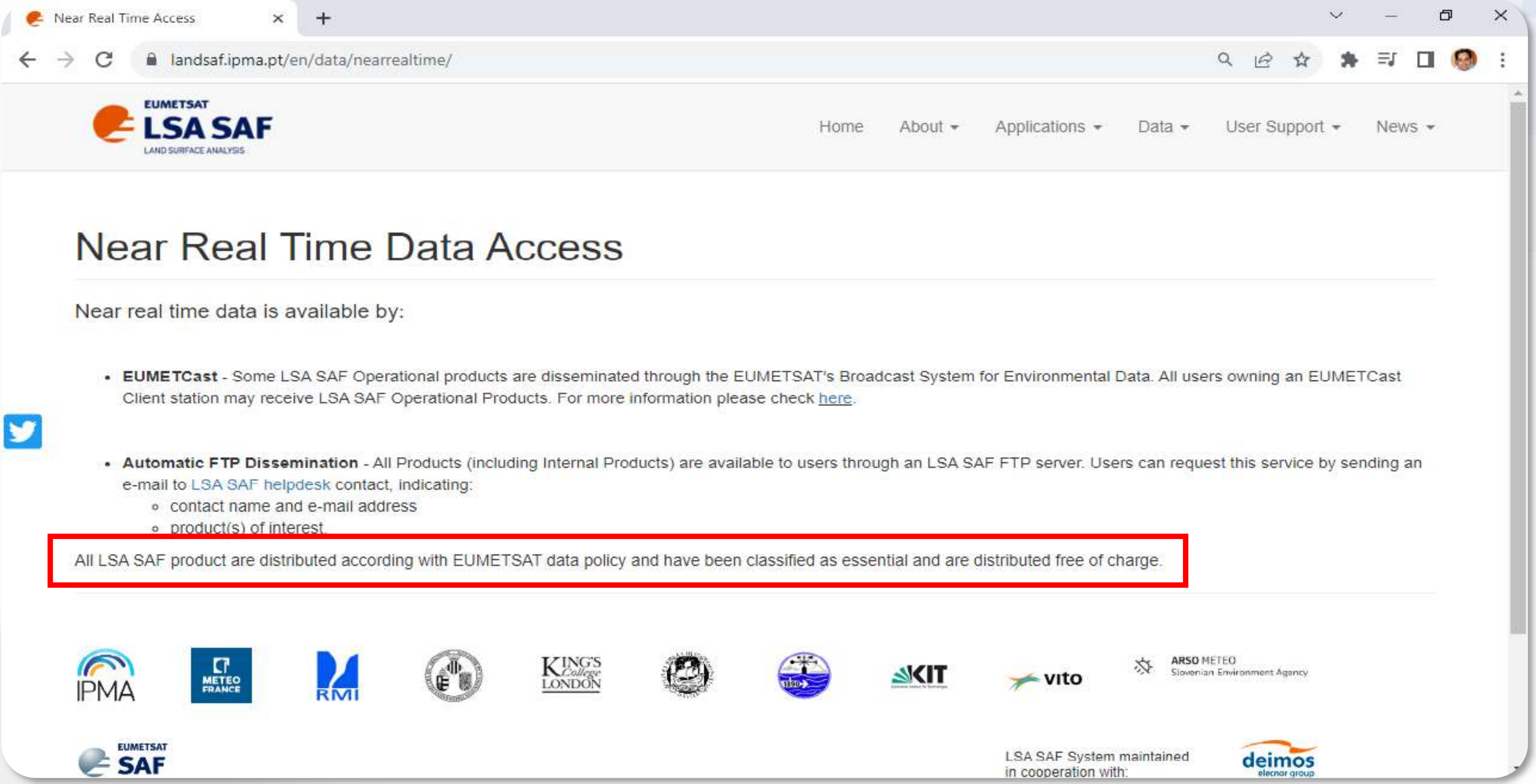
Durante el entrenamiento necesitaremos el nombre de usuario y la contraseña para acceder a los datos a través de scripts python



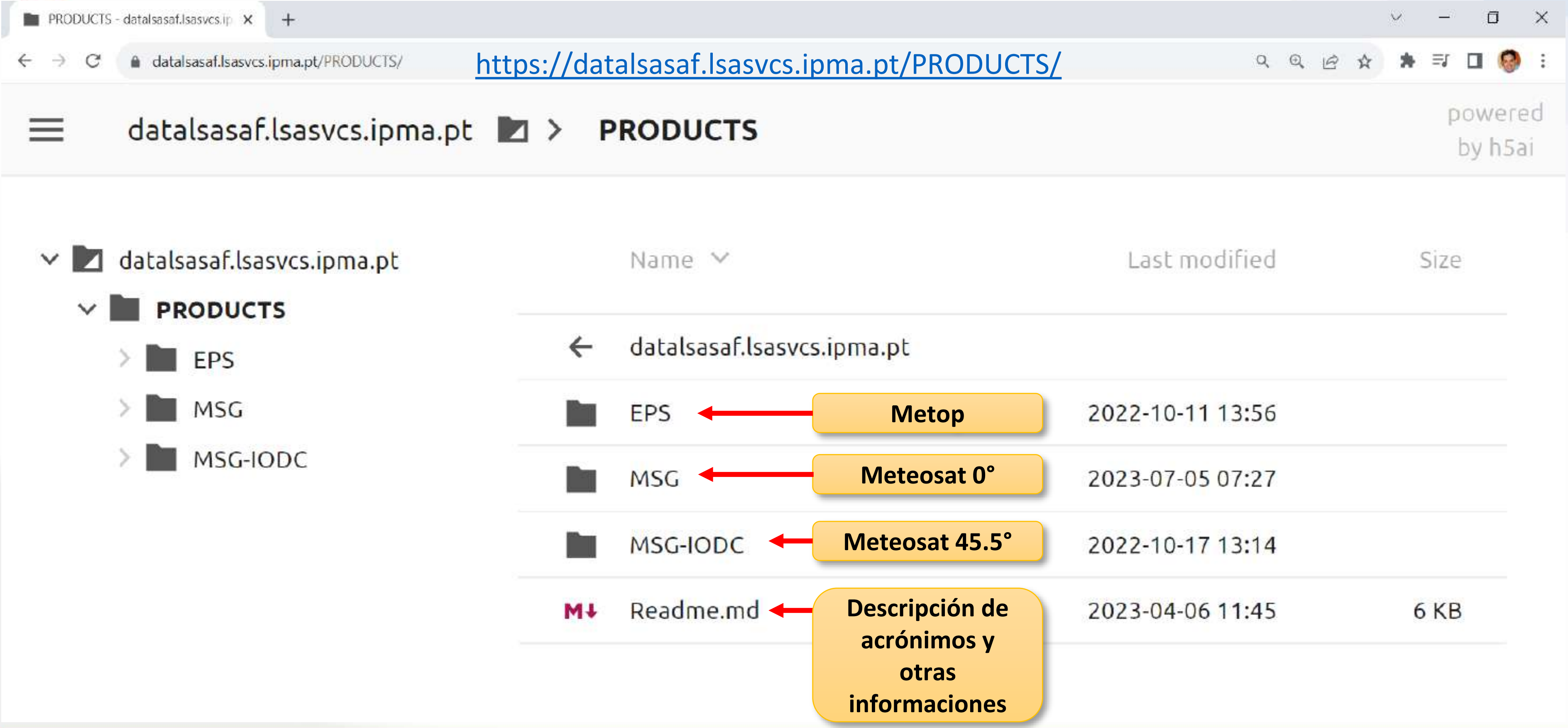
Nota: Después de registrarse, podrá acceder al servidor manualmente o mediante script transcurridos aproximadamente 30 minutos.

```
[ ]  # download a sample of the 10-day composite of MLST (LST retrieval from clear-sky infrared observations performed by SEVIRI/MSG)
     # note: after creating the LSA-SAF account it takes about 30 minutes to be able to use it.

     !wget -c --no-check-certificate \
         --user=[redacted] --password=[redacted] -P /content/samples \
         https://datalsasaf.lsasvcs.ipma.pt/PRODUCTS/MSG/DLST/NETCDF/2023/07/11/NETCDF4_LSASAF_MSG_DLST-MAX10D_MSG-Disk_202307112345.nc
```

Acceso a los Datos en Tiempo Casi Real



Interfaz de Acceso **LSA SAF**



Acrónimos LSA SAF

Welcome to the **LSA SAF data service**.

For queries, contact helpdesk.landsaf@ipma.pt

Check the data service user guide for tips to access the data.

The recomended way to download several products is with the **wget** tool which is available in most linux distributions. Windows binaries can be found here.

Data organization

The data is organized as:

satellite: satellite missions: EPS, MSG, MSG-IODC
product: See below the list of products
format: Data format: NETCDF, HDF5
year/month/day: year with 4 digits, month with 2 digits, day with two digits

Satellite missions

EPS: Metop polar orbiting meteorological satellites of the EUMETSAT Polar System

MSG: Meteosat Second Generation, geostationary at 0 degrees East

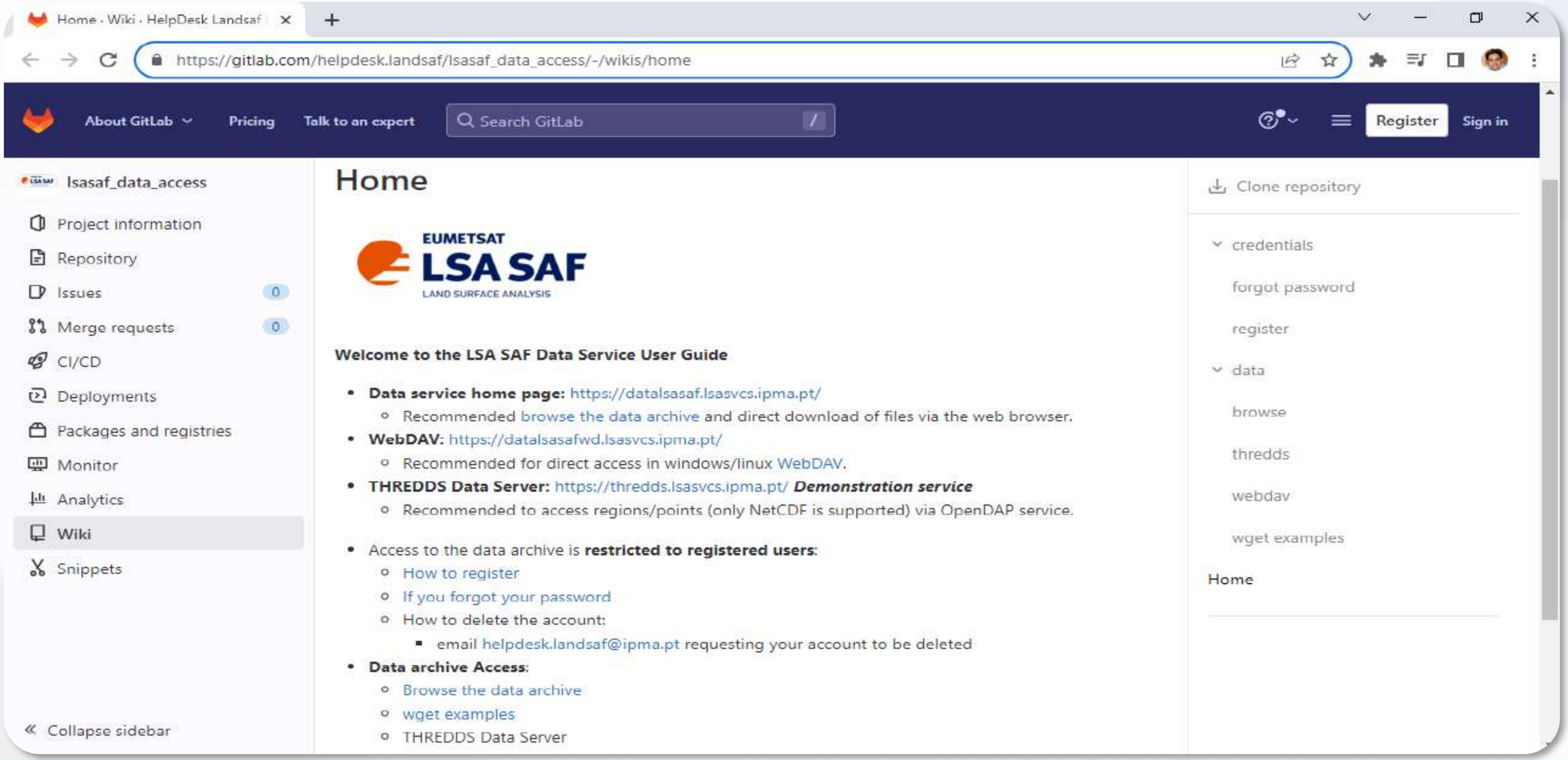
MSG-IOCD: Meteosat Second Generation, geostationary at 41.5/45.5 degrees East Indian Ocean Data Coverage

Products acronyms

Acronym	Description	Satellite missions	Format
Land Surface temperature and emissivity			
DLST	Derived LST: 10-day composites	MSG	HDF5, NETCDF
MEM	Land Surface Emissivity	MSG	HDF5, NETCDF
MLST	Land Surface Temperature	MSG, MSG-IODC	HDF5, NETCDF
MLST-AS	Land Surface Temperature - All Sky	MSG	HDF5, NETCDF
EDLST	Daily Land Surface Temperature	EPS	HDF5, NETCDF
Fire			
FRP-GRID	Fire Radiative Power Gridded	MSG	HDF5
FRP-PIXEL	Fire Radiative Power Pixel	MSG, MSG-IODC	HDF5
FRMv2	Fire Risk Map version 2	MSG	HDF5
Albedo and Vegetation			
MDAL	Daily Surface Albedo	MSG, MSG-IOCD	HDF5, NETCDF
MDFAPAR	Daily Fraction of Absorved Photosynthetic Active Radiation	MSG, MSG-IODC	HDF5, NETCDF
MDFVC	Daily Fraction of Vegetation Cover	MSG, MSG-IODC	HDF5, NETCDF
MDLAI	Daily Leaf Area Index	MSG, MSG-IODC	HDF5, NETCDF
MGPP	10-days Gross Primary Production	MSG	HDF5, NETCDF
MTAL	10-day Surface Albedo	MSG, MSG-IODC	HDF5, NETCDF
MTFAPAR	10-day Fraction of Absorved Photosynthetic Active Radiation	MSG, MSG-IODC	HDF5, NETCDF
MTVFC	10-day Fraction of Vegetation Cover	MSG, MSG-IODC	HDF5, NETCDF
MTLAI	10-day Leaf Area Index	MSG, MSG-IODC	HDF5, NETCDF
ETAL	10-day Surface Albedo	EPS	HDF5, NETCDF
ETFAPAR	10-day Fraction of Absorved Photosynthetic Active Radiation	EPS	HDF5, NETCDF
ETFVC	10-day Fraction of Vegetation Cover	EPS	HDF5, NETCDF
ETLAI	10-day Leaf Area Index	EPS	HDF5, NETCDF
ENDVI10	10-daily synthesis Normalized Difference Vegetation Index	EPS	ENVI
Radiation fluxes			
MDIDSLF	Daily Downward Surface Longwave Flux	MSG, MSG-IODC	HDF5, NETCDF
MDIDSSF	Daily Downward Surface Shortwave Flux	MSG, MSG-IODC	HDF5, NETCDF
MDSLF	Downward Surface Longwave Flux	MSG, MSG-IODC	HDF5, NETCDF
MDSSFTD	Total and Diffuse Downward Surface Shortwave Flux	MSG, MSG-IODC	HDF5, NETCDF
MNSLF	Net Surface Longwave	MSG	HDF5, NETCDF
Evaporation and turbulent fluxes			
METREF	Reference Evapotranspiration	MSG, MSG-IODC	HDF5, NETCDF
MDMET	Daily Evapotranspiration	MSG	HDF5, NETCDF
MET	Evapotranspiration	MSG	HDF5, NETCDF
MH	Sensible heat flux	MSG	HDF5, NETCDF
MLE	Latent heat flux	MSG	HDF5, NETCDF
HSAF products			
MDSC	Daily snow cover (HSAF H31)	MSG	HDF5
EDSC	Daily Snow Cover (HSAF H32)	EPS	HDF5

Referencia Utilizada Para el Desarrollo LSA SAF

https://gitlab.com/helpdesk.landsaf/lsasaf_data_access/-/wikis/home

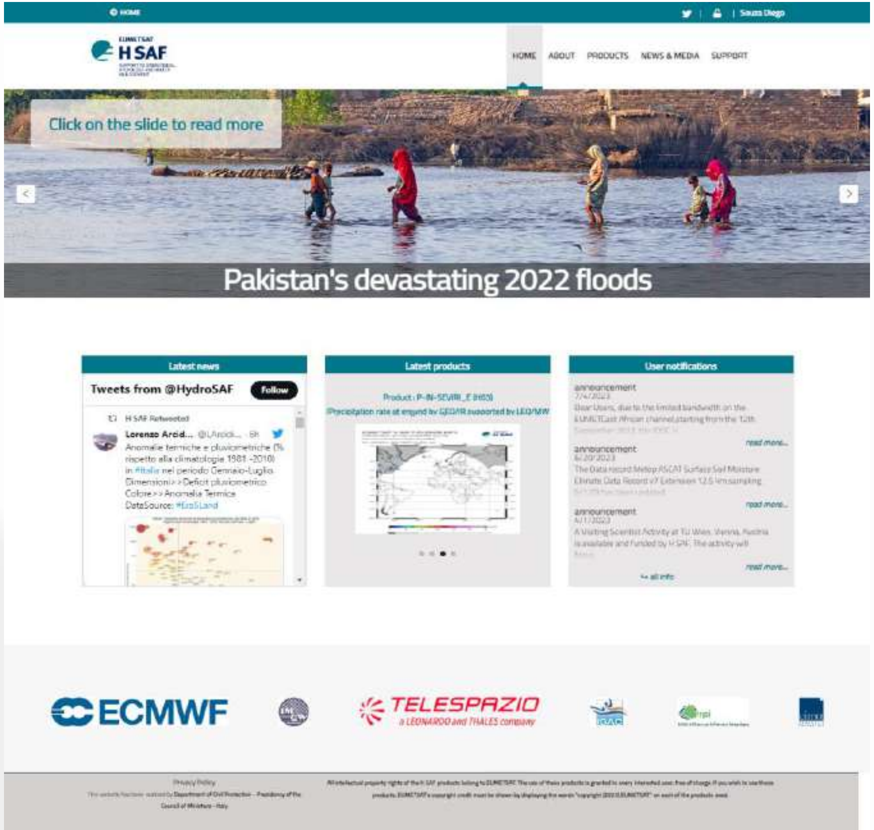


ACESSO A DADOS H SAF (HYDROLOGY AND WATER)

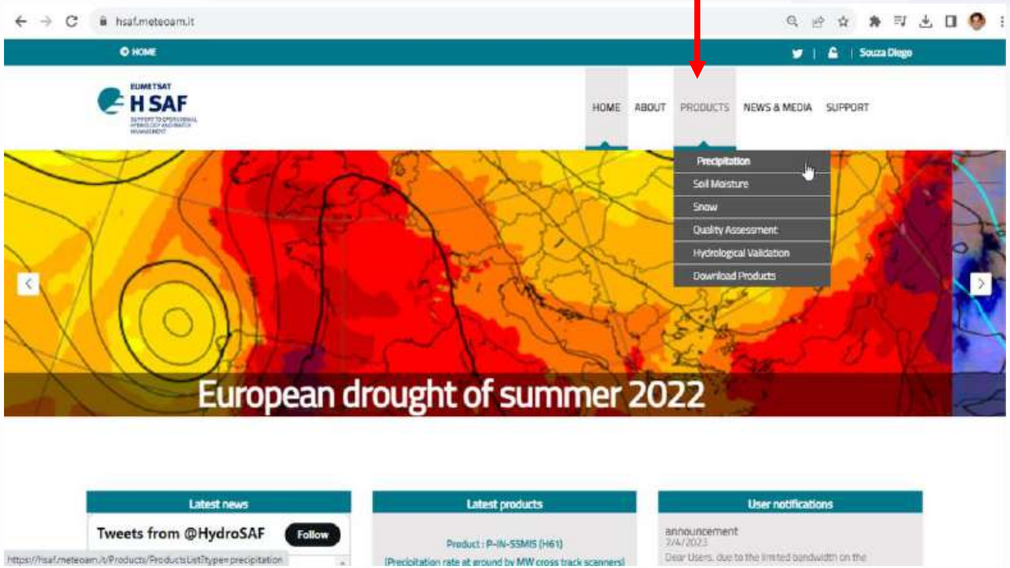
EUMETSAT
H SAF
SUPPORT TO OPERATIONAL HYDROLOGY AND WATER MANAGEMENT

Informaciones en la Página H SAF

https://hsaf.meteoam.it/

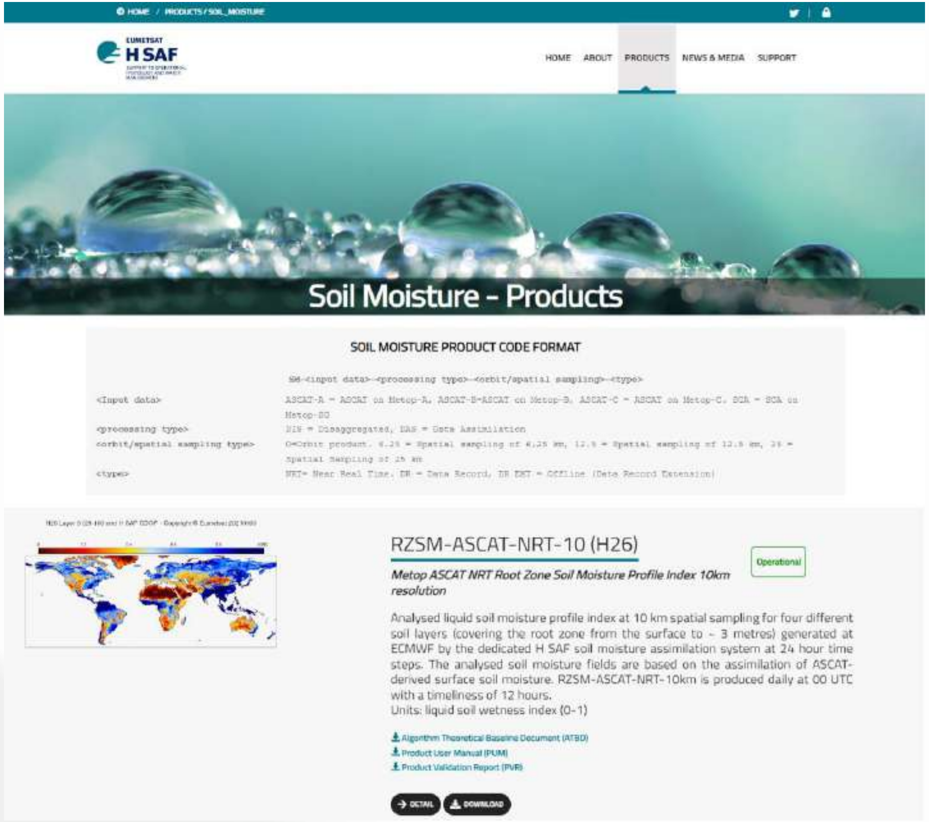


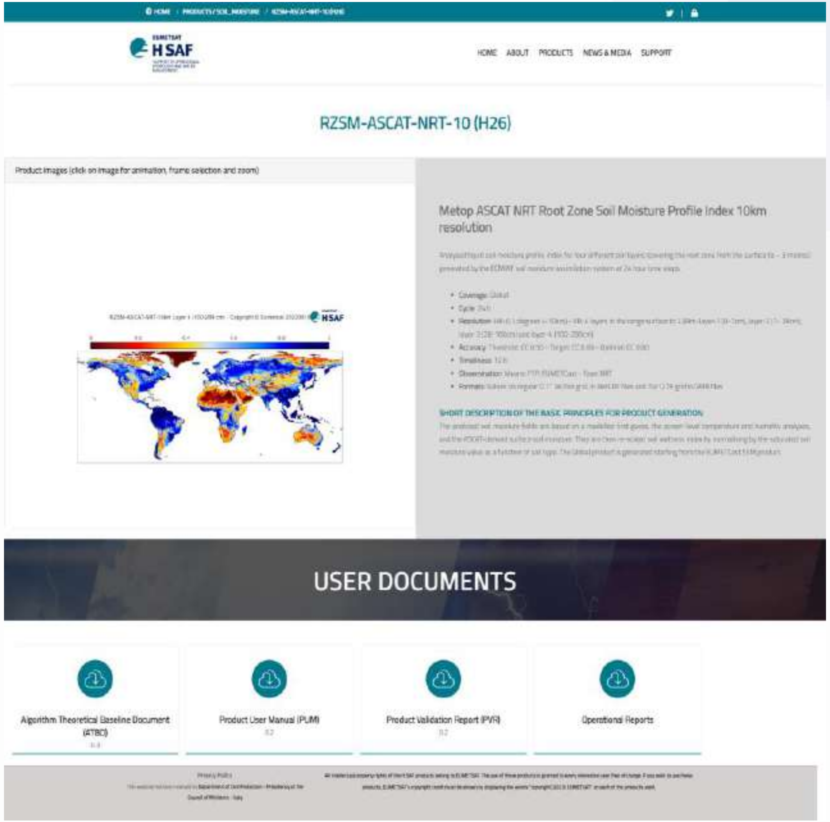
Aplicaciones, descripción de los productos y otras informaciones



Informaciones Sobre Productos

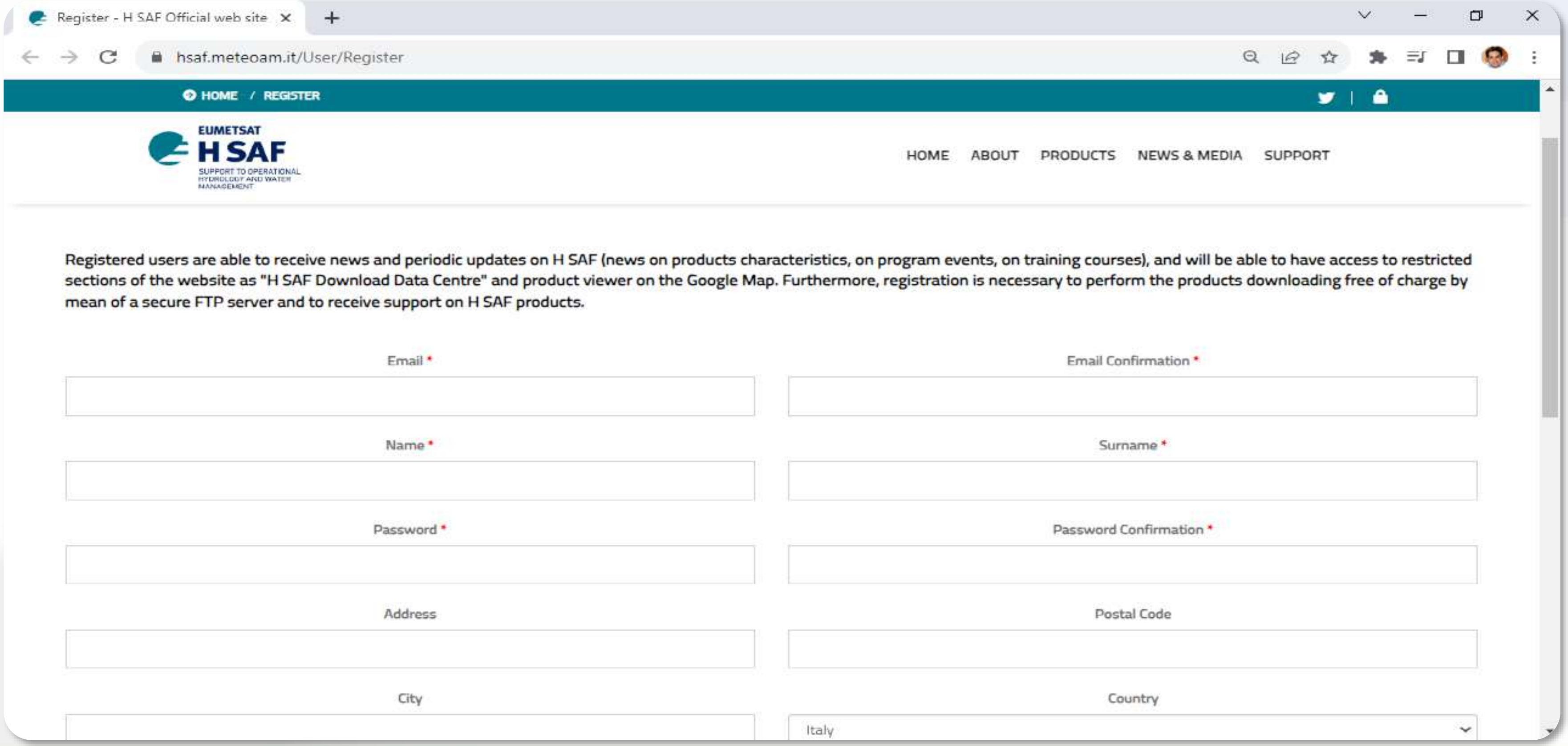
Ejemplo: https://hsaf.meteoam.it/Products/ProductsList?type=soil_moisture





Página de Registro H SAF Download Data Centre

https://hsaf.meteoam.it/User/Register



Informaciones: Usuario y Contraseña

Durante el entrenamiento necesitaremos el nombre de usuario y la contraseña para acceder a los datos a través de scripts python

Username (email)

Password

LOGIN

Forgot your password?

If you are interested in products downloading, a free registration is necessary; completed the registration process, you will receive via email the accounting credentials, as well as useful information to perform the download.

Nota: Después de registrarse, podrá acceder al servidor manualmente o mediante script transcurridos aproximadamente 30 minutos.

Baixando o dado com a biblioteca ascat

```
[ ]  # download a sample of the Root Zone Soil Moisture Profile Index (Metop/ASCAT)
     # note: after creating the H-SAF account it takes about 30 minutes to be able to use it.

     import datetime                                    # basic date and time types
     from ascat.download.interface import hsaf_download # download data from the hsaf server

     credentials = {'user': [redacted], 'password': [redacted]}          # user and password
     local_path = '/content/samples/'                                 # where we want to store the data
     remote_path = '/h26/h26_cur_mon_nc/'                             # FTP server folder
     start_date = datetime.datetime(2023, 7, 25)                      # start date
     end_date = datetime.datetime(2023, 7, 26)                        # end date

     hsaf_download(credentials, remote_path, local_path, start_date, end_date, limit=1) # download the file(s)
```

Página H SAF Products Download Data Centre

https://hsaf.meteoam.it/User/UserSupport

HOME / USER SUPPORT | Souza Diego

Servidor FTP: `ftphsaf.meteoam.it` **Nombre de usuario y contraseña de acuerdo con el registro**

Access To H SAF On-Line Archive (Last 60 Days)

You can access the H-SAF FTP server with your favourite FTP client.

FTP host: `ftphsaf.meteoam.it`

Credentials are the same as for this website.

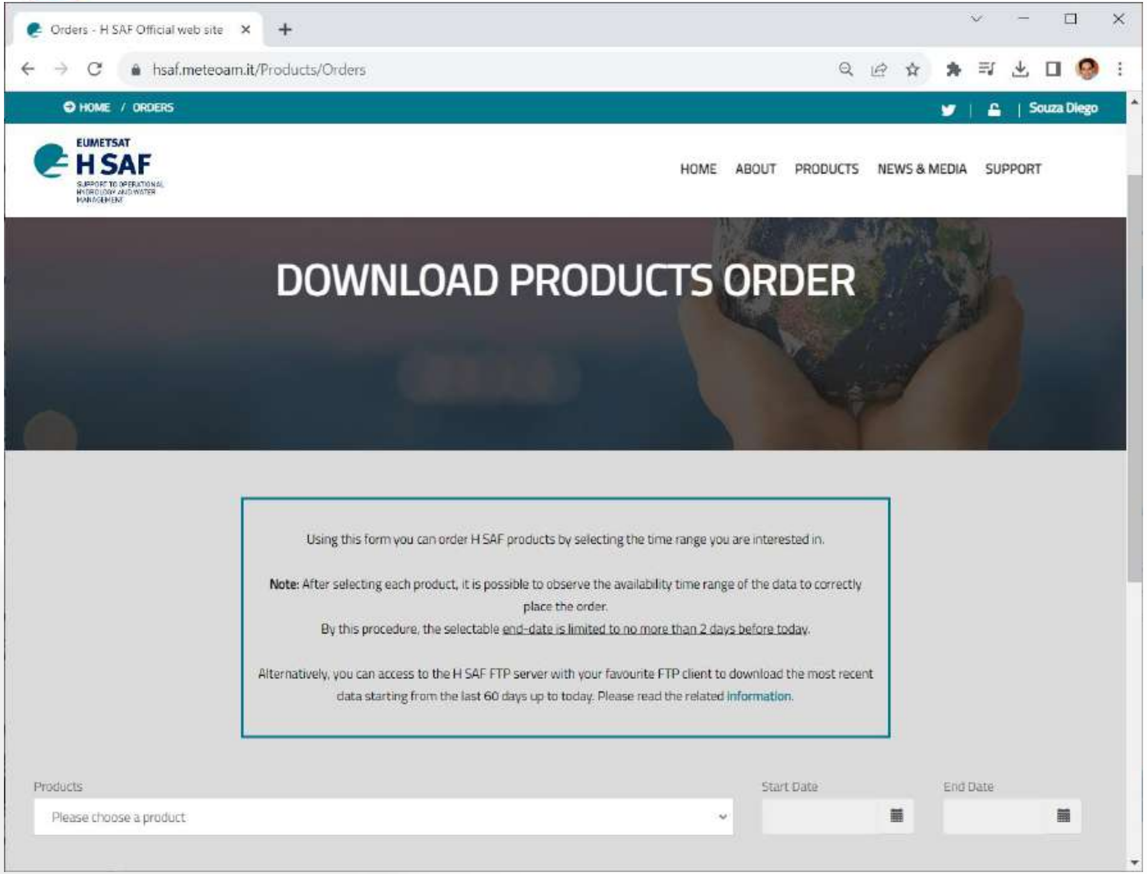
NOTE

In case of just created website accounts or in case of just updated passwords a time lapse up to 30 minutes may be required to have access to the FTP server.

Place An Order To Get The H SAF Archived Data (Basic)

This function allows access to the H SAF internal order centre. This service offers all basic functions to carry out orders on H SAF historical archives (starting from the pre-operation/operational status of products). Orders placed will be submitted for approval and will be processed within three working days

Página H SAF Products Download Data Centre



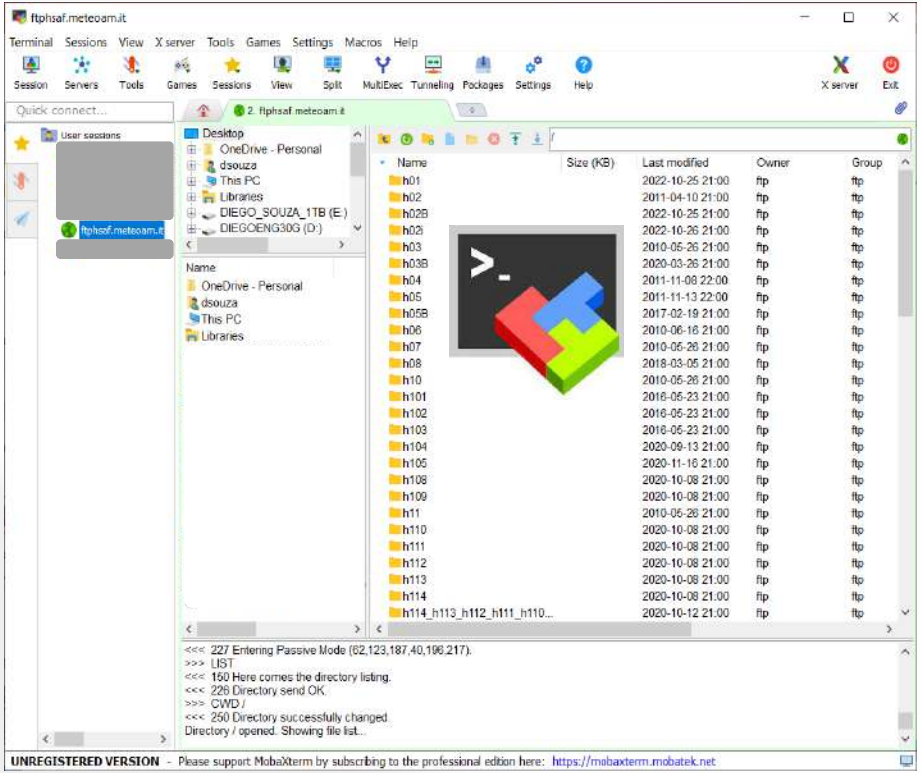
Es posible realizar una solicitud a través de la propia página H SAF (datos de hasta dos días antes de la fecha actual).

Como demostración, no utilizaremos este mecanismo, sino que accederemos a través de scripts. También puede utilizar un cliente FTP.

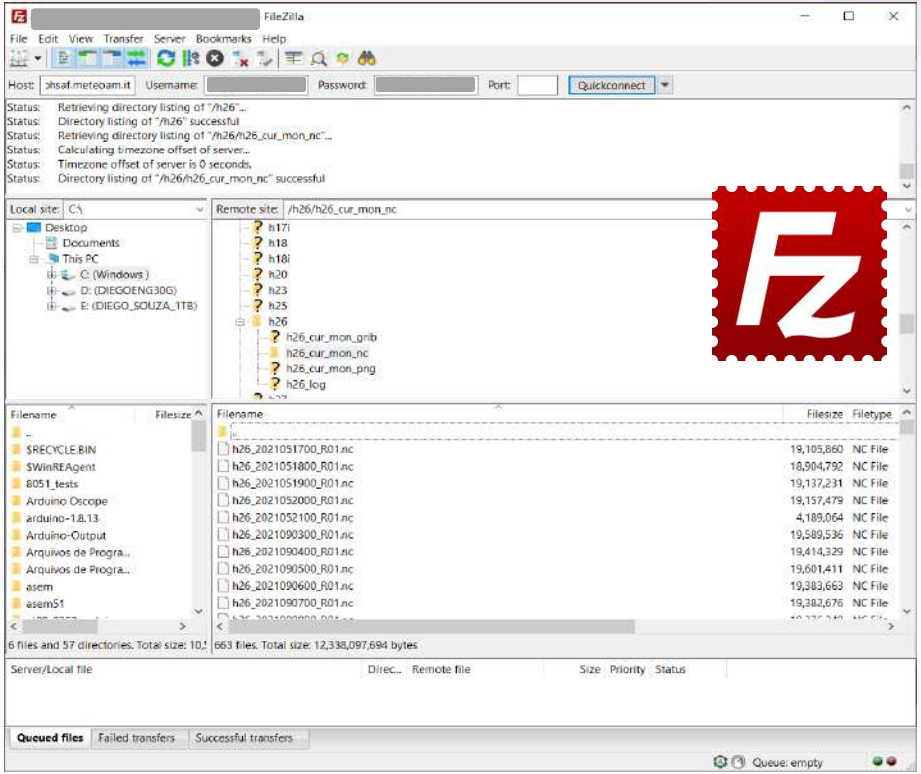
Existen Clientes FTP Gratuitos

Servidor FTP: `ftphsaf.meteoam.it` **Nombre de usuario y contraseña de acuerdo con el registro**

MobaXtern: https://mobaxterm.mobatek.net/



FileZilla: https://filezilla-project.org/



Credenciales LSA-SAF y H-SAF

Prácticas LSA-SAF y H-SAF

EUMETSAT
LSA SAF
LAND SURFACE ANALYSIS

EUMETSAT
H SAF
SUPPORT TO OPERATIONAL HYDROLOGY AND WATER MANAGEMENT

Credenciales LSA-SAF and H-SAF

Para descargar datos desde LSA SAF y H SAF, primero es necesario registrarse en LSA SAF https://mokey.lsasvcs.ipma.pt/auth/signup y H SAF https://hsaf.meteoam.it/User/Register, y inserir tus credenciales en las variables abajo.

Nota: Sólo es necesario ejecutar esta celda una vez.

```
#==========================================================================================#
# LSA SAF AND H SAF CREDENTIALS
#==========================================================================================#

# LSA-SAF credentials
lsa_saf_user = 
lsa_saf_password = 

# H-SAF credentials
h_saf_user = 
h_saf_password = 
```

DEMONSTRACIONES

Así como estamos haciendo con los datos de canales y RGBs, vamos a usar funciones simples para descargar y visualizar datos SAF

EUMETSAT
LSA SAF
LAND SURFACE ANALYSIS

EUMETSAT
H SAF
SUPPORT TO OPERATIONAL HYDROLOGY AND WATER MANAGEMENT

Ejemplo 1: LST - 10 Day Composite (SEVIRI)

Ejemplo 1: Land Surface Temperature - 10 Day Composite (MSG/SEVIRI)

Nombre de la función

Parámetros

```
# Land Surface Temperature - 10 Day Composite (MSG/SEVIRI)
# possible days (01, 11 and 21), since 2015 (MSG
plot_saf(saf_product='LSASAF_MSG_DLST-MAX10D',
         date_saf='2023-10-21 23:45',
         extent=[-80.0, -40.00, -30.00, 10.00],
         cmap='custom', vmin=-20, vmax=48,
         coast_color='black',
         countries_color='black',
         grid_color='white',
         land_ocean=True,
         land_color='beige',
         ocean_color='gray',
         figsize=[10,10])
```

Producto

Fecha 'YYYY-MM-DD HH:MM'

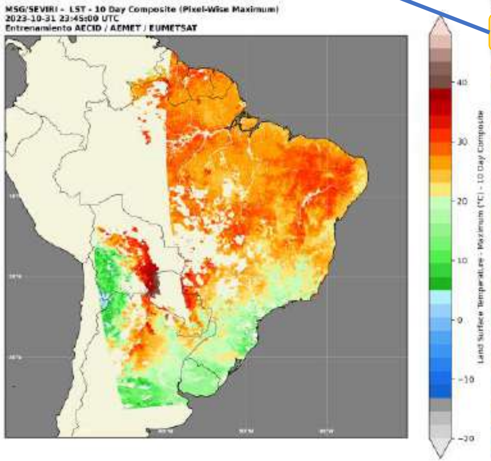
Región deseada

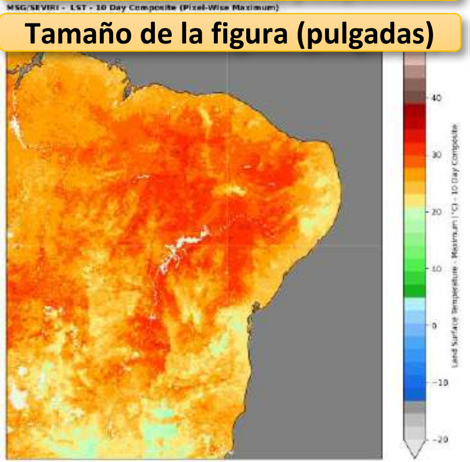
Colormap, mínimo y máximo

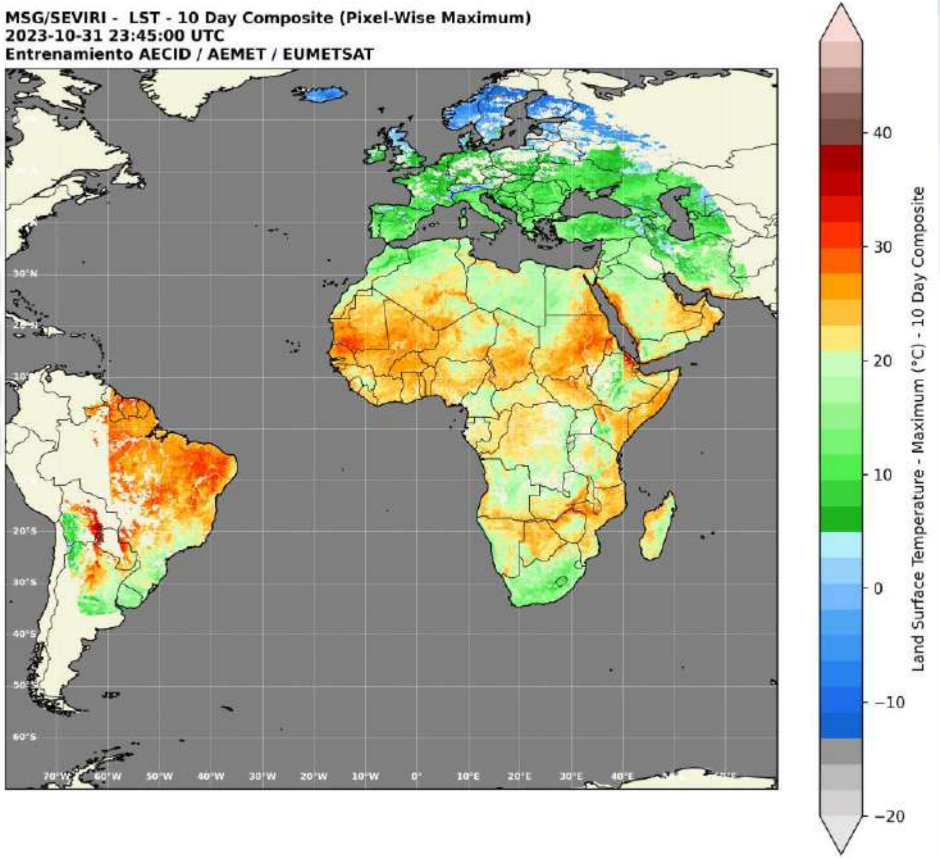
Colores

Máscara de océano y superficie

Tamaño de la figura (pulgadas)







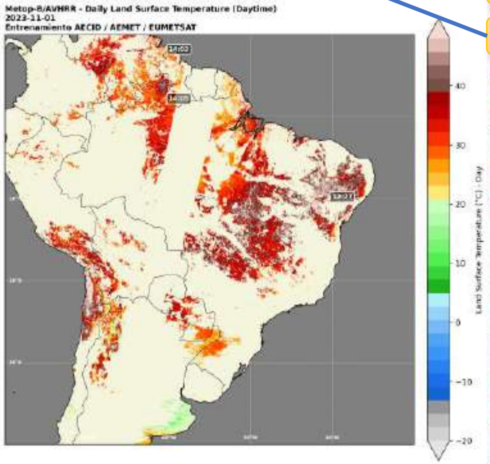
Ejemplo 2: LST (Metop/AVHRR) - Day

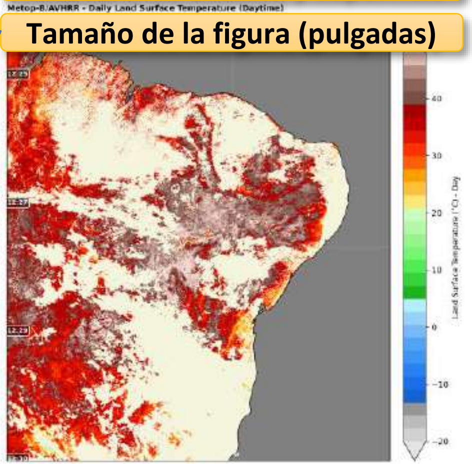
Ejemplo 2: Land Surface Temperature (Metop/AVHRR) - Day

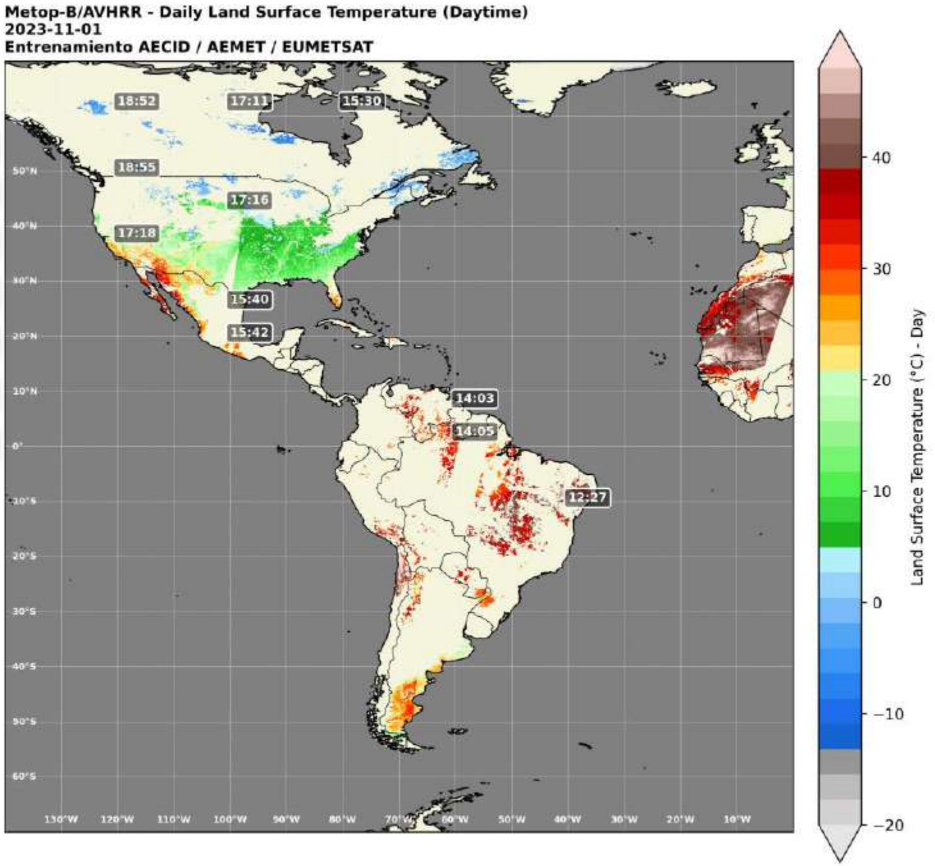
Nombre de la función | Parámetros

```
# Land Surface Temperature (Metop/AVHRR)
# daily product (00:00), since 2015 (global)
plot_saf(saf_product = 'LSASAF_M01-AVHR_EDLST-DAY_GLOBE',
         date_saf='2023-11-01 00:00',
         extent=[-80.0, -40.00, -30.00, 10.00],
         cmap='custom', vmin=-20, vmax=48,
         coast_color='black',
         countries_color='black',
         grid_color='white',
         land_ocean=True,
         land_color='beige',
         ocean_color='gray',
         figsize=[10,10])
```

- Producto
- Fecha 'YYYY-MM-DD HH:MM'
- Región deseada
- Colormap, mínimo y máximo
- Colores
- Máscara de océano y superficie
- Tamaño de la figura (pulgadas)







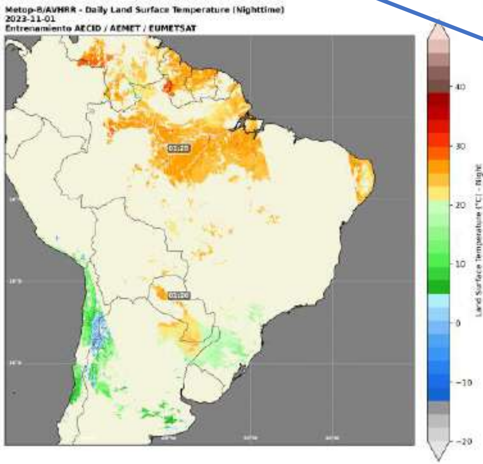
Ejemplo 3: LST (Metop/AVHRR) - Night

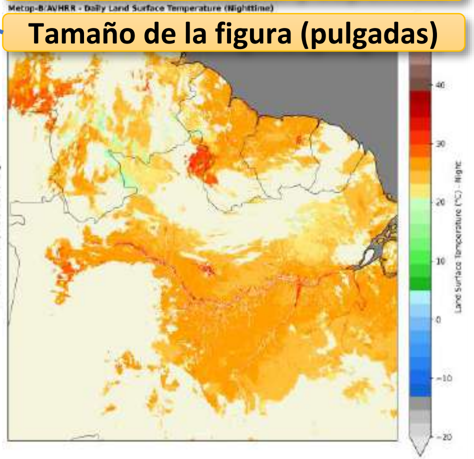
Ejemplo 3: Land Surface Temperature (Metop/AVHRR) - Night

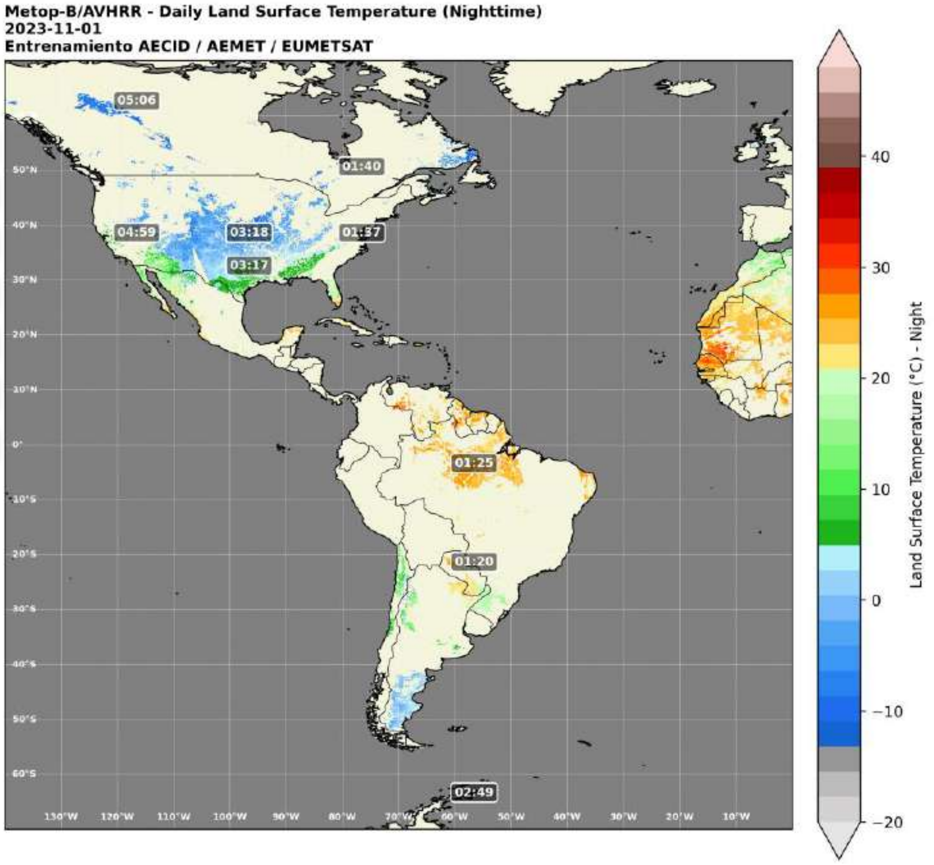
Nombre de la función | Parámetros

```
# Land Surface Temperature (Metop/AVHRR)
# daily product (00:00), since 2015 (global)
plot_saf(saf_product = 'LSASAF_M01-AVHR_EDLST-NIGHT_GLOBE',
         date_saf='2023-11-01 00:00',
         extent=[-80.0, -40.00, -30.00, 10.00],
         cmap='custom', vmin=-20, vmax=48,
         coast_color='black',
         countries_color='black',
         grid_color='white',
         land_ocean=True,
         land_color='beige',
         ocean_color='gray',
         figsize=[10,10])
```

- Producto
- Fecha ‘YYYY-MM-DD HH:MM’
- Región deseada
- Colormap, mínimo y máximo
- Colores
- Máscara de océano y superficie
- Tamaño de la figura (pulgadas)







Ejemplo 4: NDVI (Metop/AVHRR) - South America

Ejemplo 4: Normalized Difference Vegetation Index (Metop/AVHRR) - South America

Nombre de la función

Parámetros

```
# Normalized Difference Vegetation Index (Metop/AVHRR) - South America
# possible days (01, 11 and 21), since 2007. Max
plot_saf(saf_product = 'METOP_AVHRR_S10_AMs_NDV',
         date_saf='2023-10-21 00:00',
         extent=[-75.0, -37.00, -33.00, 8.00],
         cmap='custom', vmin=0.1, vmax=0.8,
         coast_color='black',
         countries_color='black',
         grid_color='white',
         land_ocean=True,
         land_color='beige',
         ocean_color='gray',
         figsize=[10,10])
```

Producto

Fecha ‘YYYY-MM-DD HH:MM’

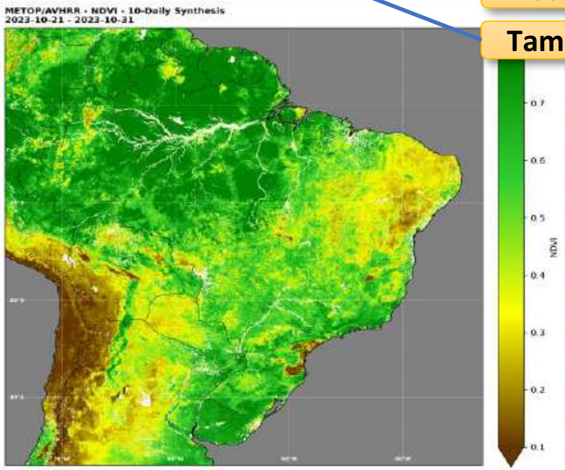
Región deseada

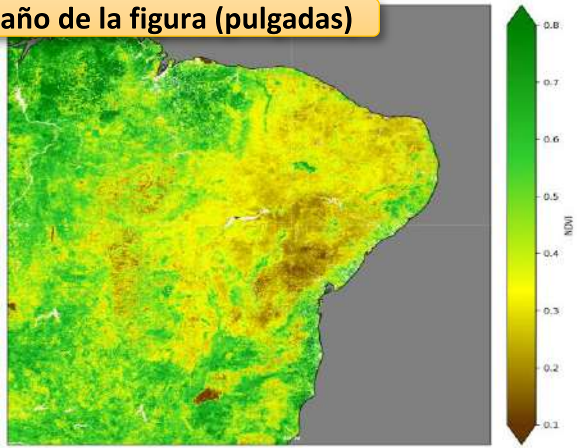
Colormap, mínimo y máximo

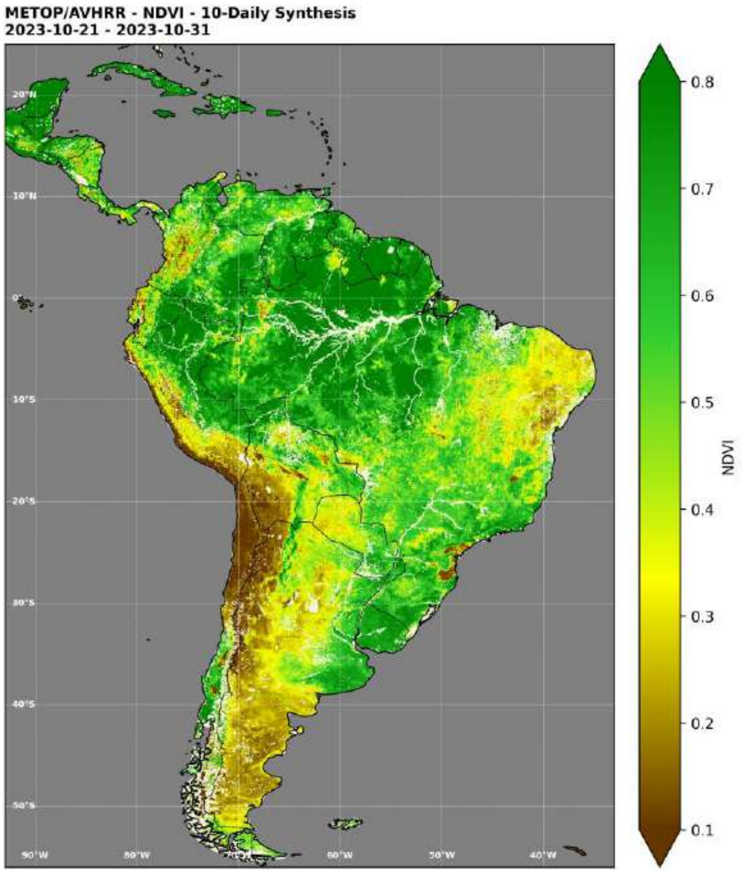
Colores

Máscara de océano y superficie

Tamaño de la figura (pulgadas)







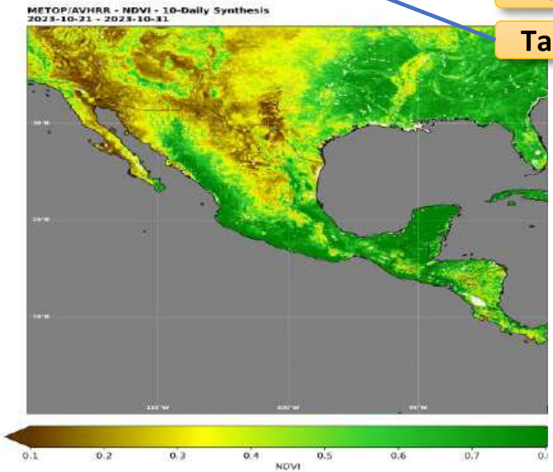
Ejemplo 5: NDVI (Metop/AVHRR) - Central America

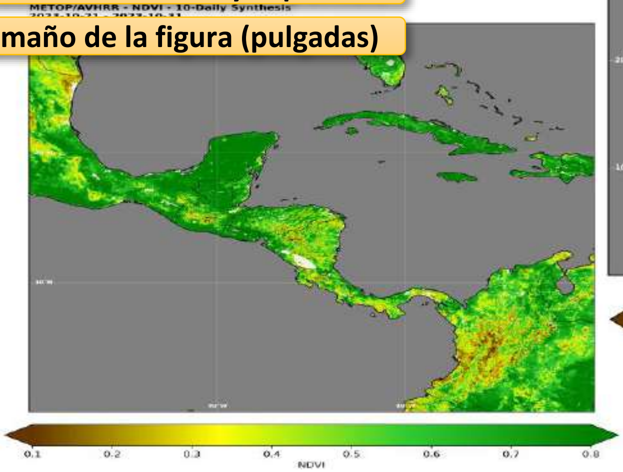
Ejemplo 5: Normalized Difference Vegetation Index (Metop/AVHRR) - Central America

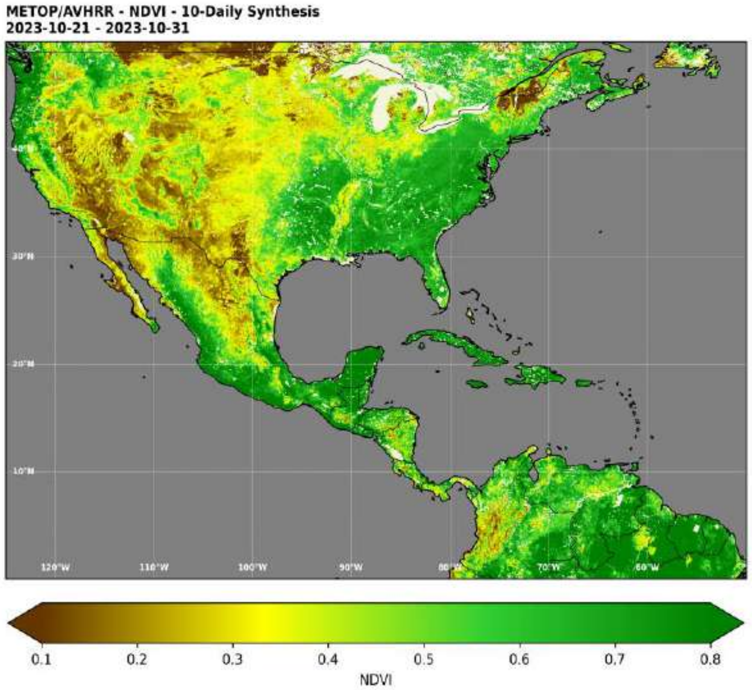
Nombre de la función | Parámetros

```
# Normalized Difference Vegetation Index (Metop/AVHRR) - Central America
# possible days (01, 11 and 21), since 20?7. Max
plot_saf(saf_product = 'METOP_AVHRR_S10_AMc_NDV',
         date_saf='2023-10-21 00:00',
         extent=[-120.0, 0.00, -80.00, 40.00],
         cmap='custom', vmin=0.1, vmax=0.8,
         coast_color='black',
         countries_color='black',
         grid_color='white',
         land_ocean=True,
         land_color='beige',
         ocean_color='gray',
         figsize=[10,10])
```

- Producto
- Fecha 'YYYY-MM-DD HH:MM'
- Región deseada
- Colormap, mínimo y máximo
- Colores
- Máscara de océano y superficie
- Tamaño de la figura (pulgadas)







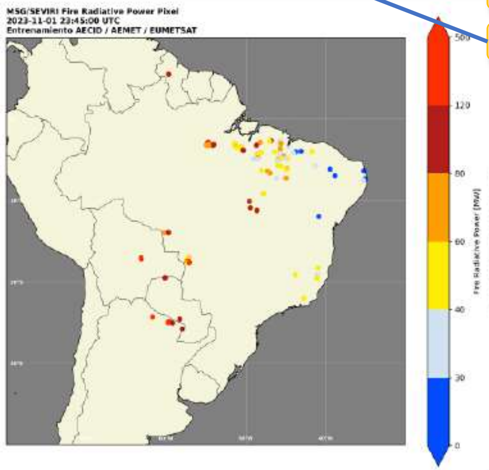
Ejemplo 6: Fire Radiative Power Pixel (MSG/SEVIRI)

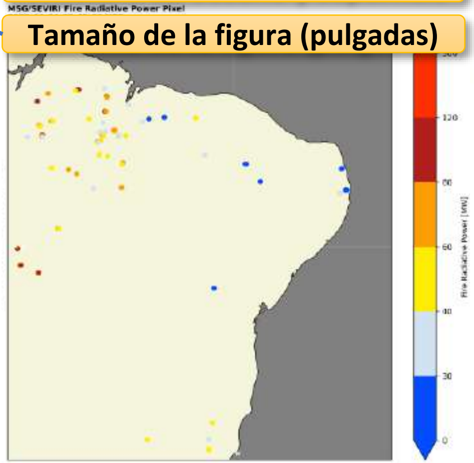
Ejemplo 6: Fire Radiative Power Pixel (MSG/SEVIRI)

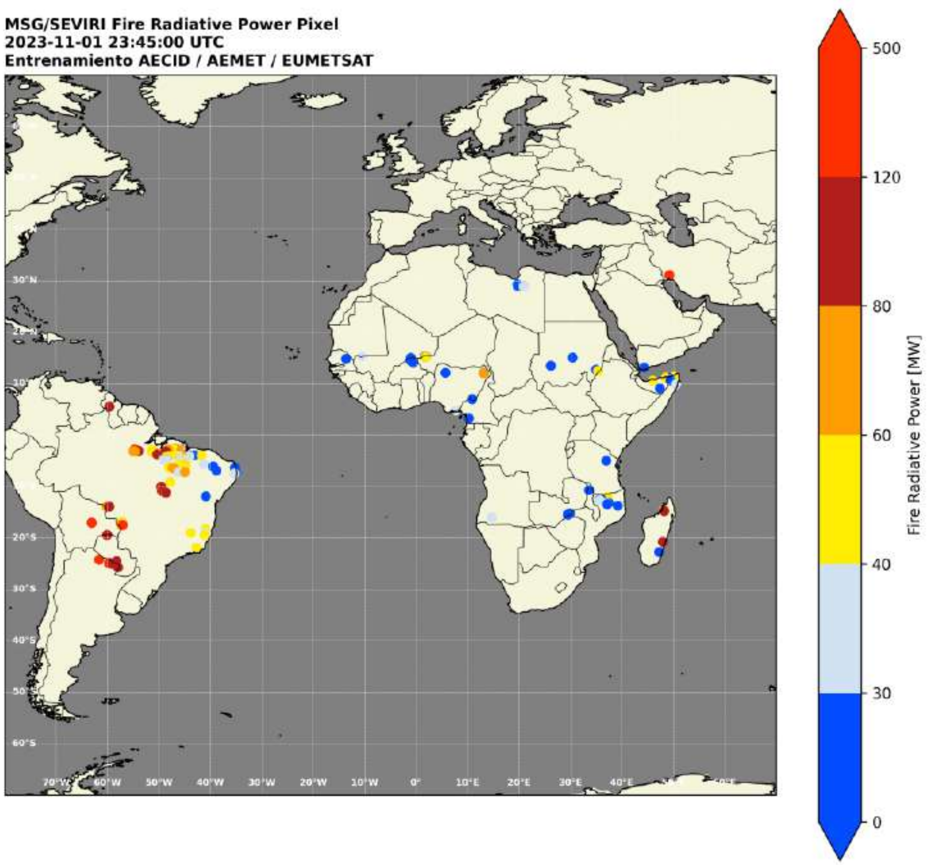
Nombre de la función | Parámetros

```
# Fire Radiative Power Pixel (MSG/SEVIRI)
# daily product (every 15 minutes), since 2004 (M
plot_saf(saf_product = 'LSASAF_MSG_FRP-PIXEL',
         date_saf='2023-11-01 23:45',
         extent=[-80.0, -40.00, -30.00, 10.00],
         cmap='custom', vmin=0, vmax=500,
         coast_color='black',
         countries_color='black',
         grid_color='white',
         land_ocean=True,
         land_color='beige',
         ocean_color='gray',
         figsize=[10,10])
```

- Producto
- Fecha 'YYYY-MM-DD HH:MM'
- Región deseada
- Colormap, mínimo y máximo
- Colores
- Máscara de océano y superficie
- Tamaño de la figura (pulgadas)







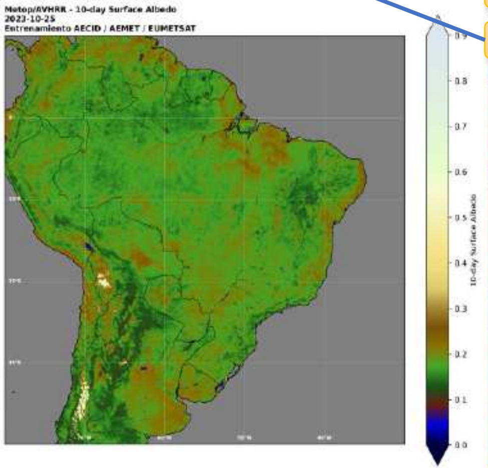
Ejemplo 7: 10-day Surface Albedo

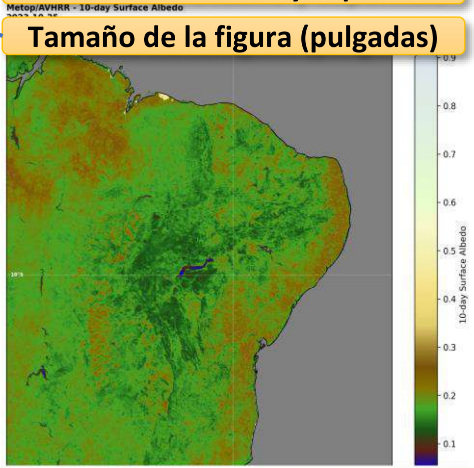
Ejemplo 7: 10-day Surface Albedo

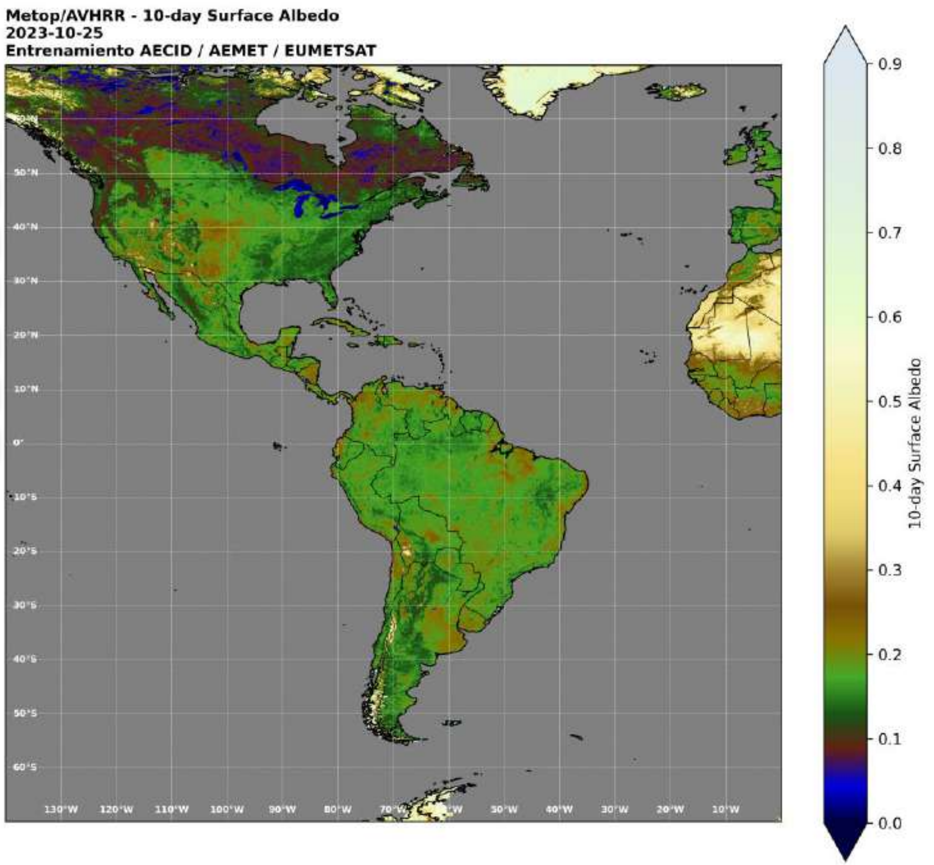
Nombre de la función | Parámetros

```
# 10-day Surface Albedo
# possible days (05, 15 and 25), since 20 5 (glob
plot_saf(saf_product = 'LSASAF_M01-AVHR_ETAL_GLOBE
         date_saf='2023-10-25 00:00',
         extent=[-80.0, -40.00, -30.00, 10.00],
         cmap='custom', vmin=0, vmax=0.9,
         coast_color='black',
         countries_color='black',
         grid_color='white',
         land_ocean=True,
         land_color='beige',
         ocean_color='gray',
         figsize=[10,10])
```

- Producto
- Fecha 'YYYY-MM-DD HH:MM'
- Región deseada
- Colormap, mínimo y máximo
- Colores
- Máscara de océano y superficie
- Tamaño de la figura (pulgadas)







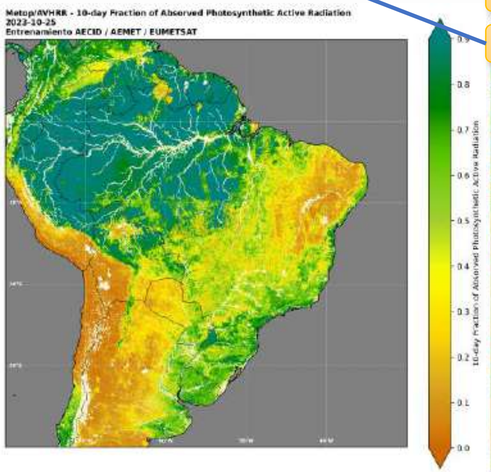
Ejemplo 8: 10-day Fract. of Abs. Photosynthetic Active Rad.

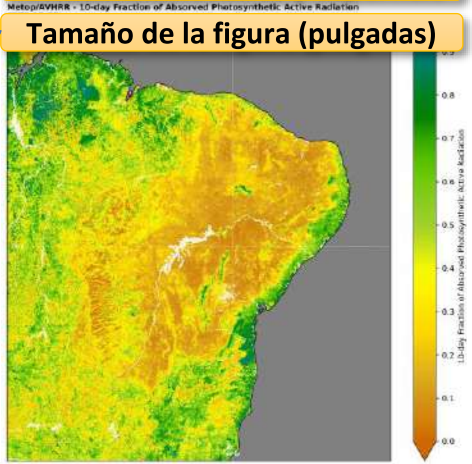
Ejemplo 8: 10-day Fraction of Absorved Photosynthetic Active Radiation

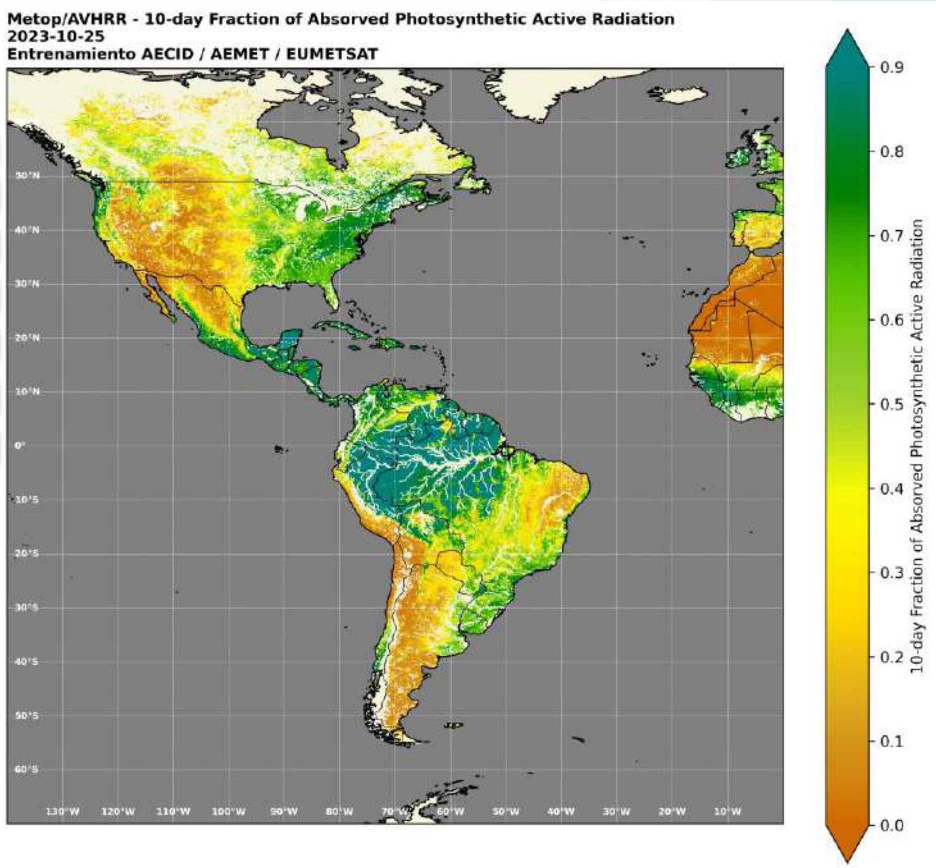
Nombre de la función | Parámetros

```
# 10-day Fraction of Absorved Photosynthetic Active Radiation
# possible days (05, 15 and 25), since 2015 (glob
plot_saf(saf_product = 'LSASAF_M01-AVHR_ETFAPAR_GL
         date_saf='2023-10-25 00:00',
         extent=[-80.0, -40.00, -30.00, 10.00],
         cmap='custom', vmin=0, vmax=0.9,
         coast_color='black',
         countries_color='black',
         grid_color='white',
         land_ocean=True,
         land_color='beige',
         ocean_color='gray',
         figsize=[10,10])
```

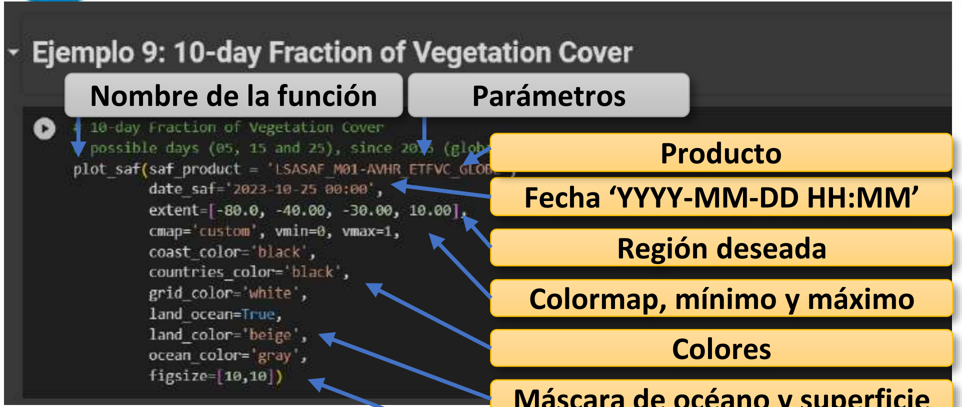
- Producto
- Fecha 'YYYY-MM-DD HH:MM'
- Región deseada
- Colormap, mínimo y máximo
- Colores
- Máscara de océano y superficie
- Tamaño de la figura (pulgadas)

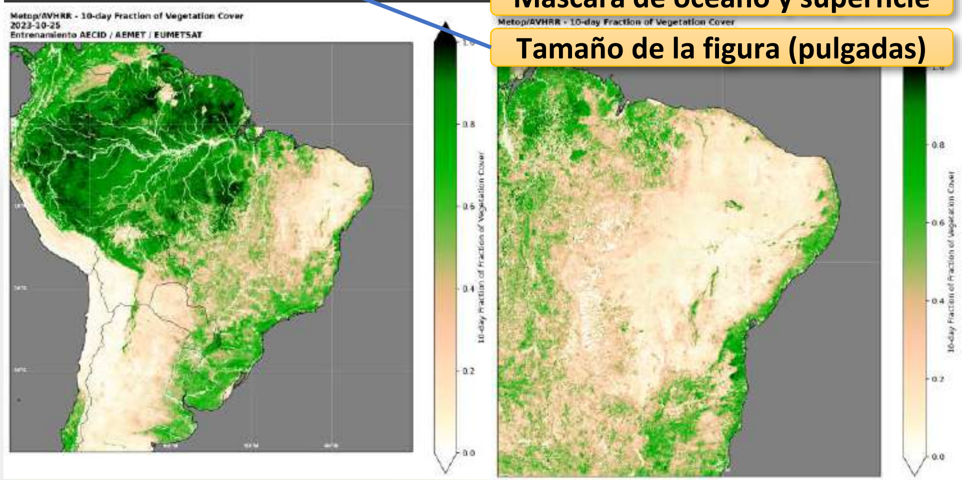


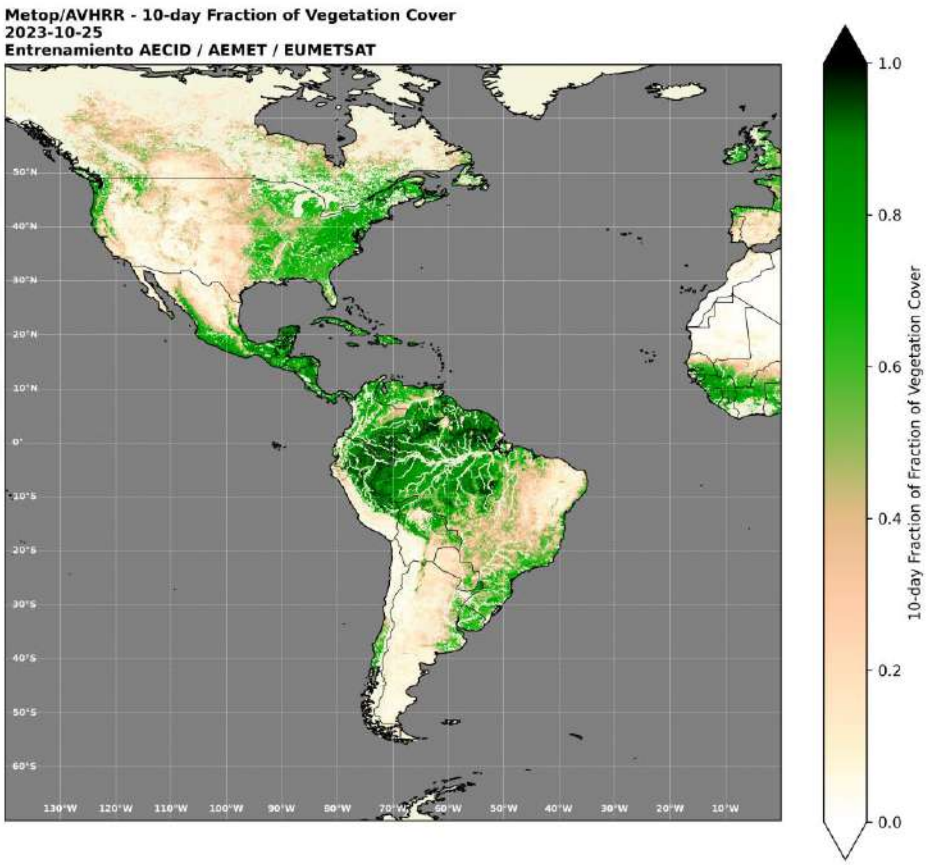




Ejemplo 9: 10-day Fraction of Vegetation Cover







Ejemplo 10: 10-day Leaf Area Index

Ejemplo 10: 10-day Leaf Area Index

Nombre de la función | Parámetros

```
# 10-day Leaf Area Index
# possible days (05, 15 and 25), since 2015 (glob
plot_saf(saf_product = 'LSASAF_M01-AVHR_ETLAI_GLOB
         date_saf='2023-10-25 00:00',
         extent=[-80.0, -40.00, -30.00, 10.00],
         cmap='custom', vmin=0, vmax=6,
         coast_color='black',
         countries_color='black',
         grid_color='white',
         land_ocean=True,
         land_color='beige',
         ocean_color='gray',
         figsize=[10,10])
```

Producto

Fecha 'YYYY-MM-DD HH:MM'

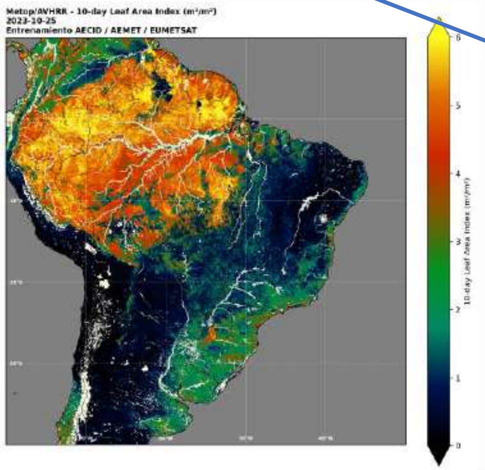
Región deseada

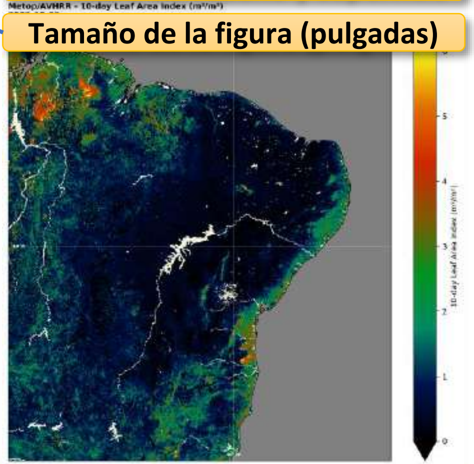
Colormap, mínimo y máximo

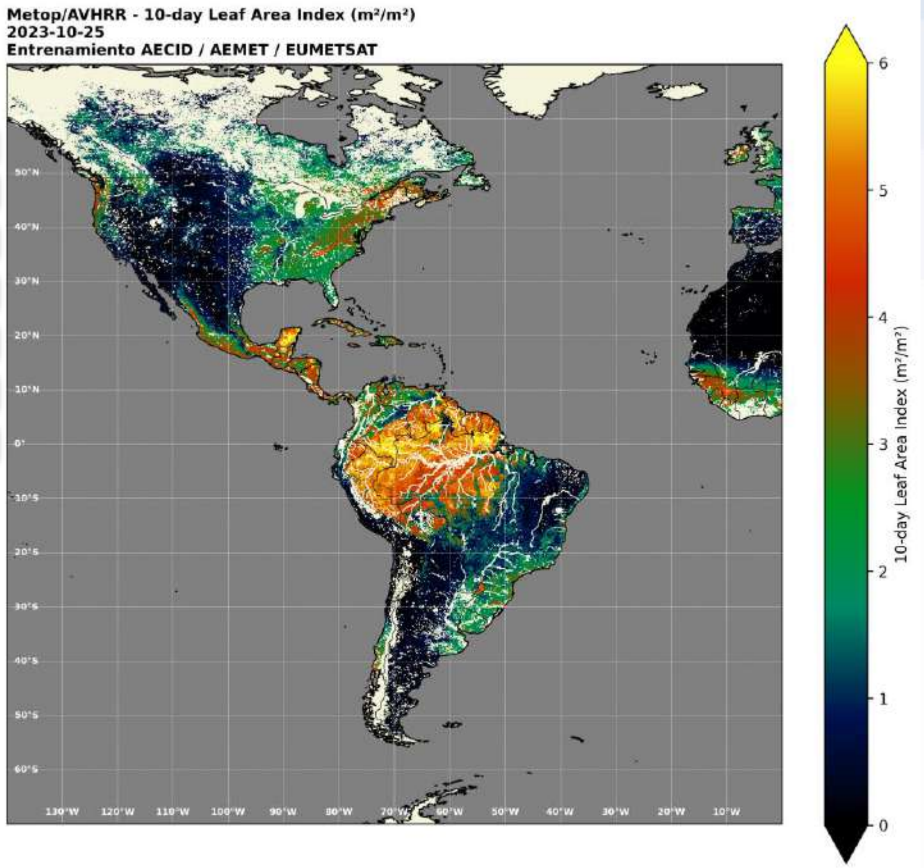
Colores

Máscara de océano y superficie

Tamaño de la figura (pulgadas)







Ejemplo 11: Evapotranspiration (MSG)

Ejemplo 11: Evapotranspiration (MSG)

Nombre de la función

Parámetros

```
# Evapotranspiration (MSG)
# daily product (every 30 minutes), since 2004 (MS
plot_saf(saf_product = 'LSASAF_MSG_ETv3_MSG-Disk',
         date_saf='2023-11-01 15:00',
         extent=[-80.0, -40.00, -30.00, 10.00],
         cmap='jet', vmin=0, vmax=1,
         coast_color='black',
         countries_color='black',
         grid_color='white',
         land_ocean=True,
         land_color='beige',
         ocean_color='gray',
         figsize=[10,10])
```

Producto

Fecha 'YYYY-MM-DD HH:MM'

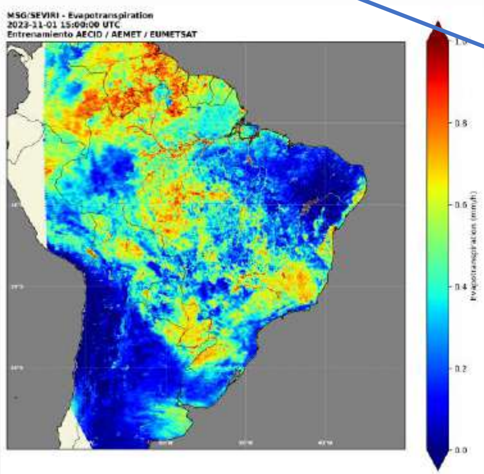
Región deseada

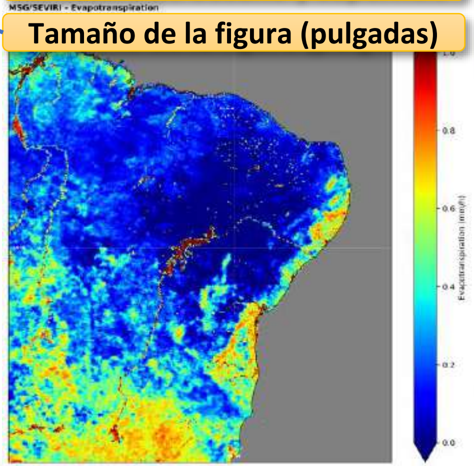
Colormap, mínimo y máximo

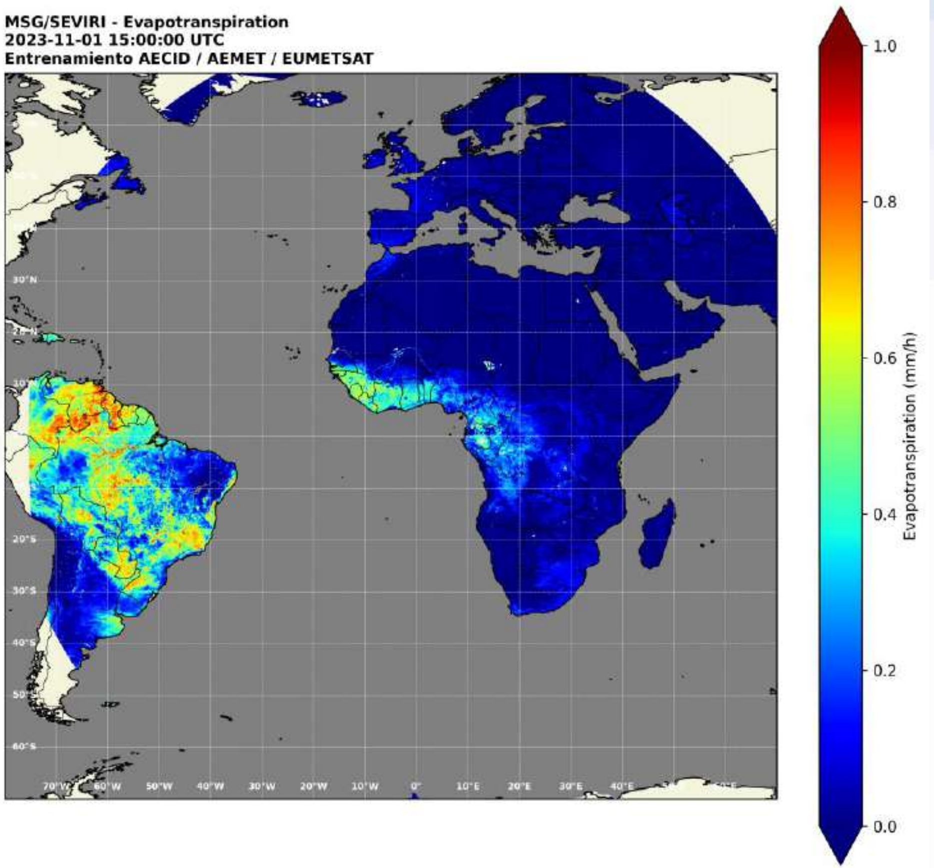
Colores

Máscara de océano y superficie

Tamaño de la figura (pulgadas)







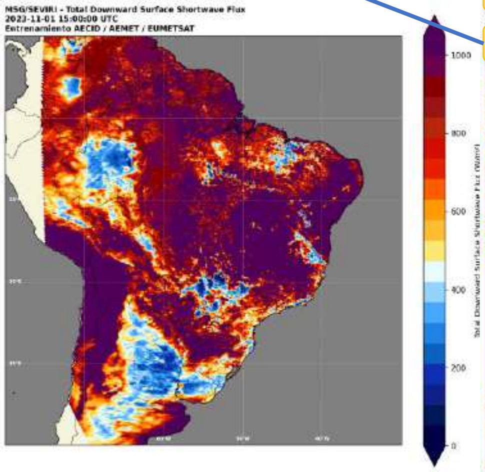
Ejemplo 12: Total Downward Surface Shortwave Flux (MSG)

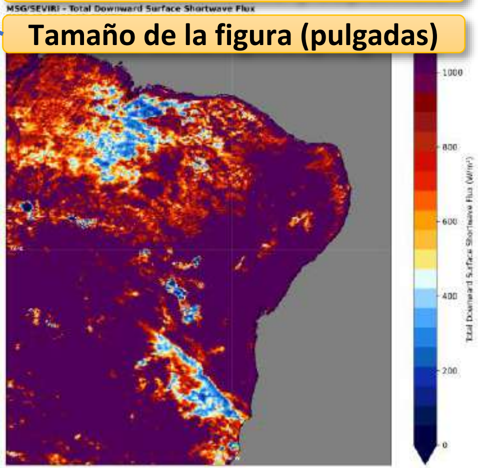
Ejemplo 12: Total Downward Surface Shortwave Flux (MSG)

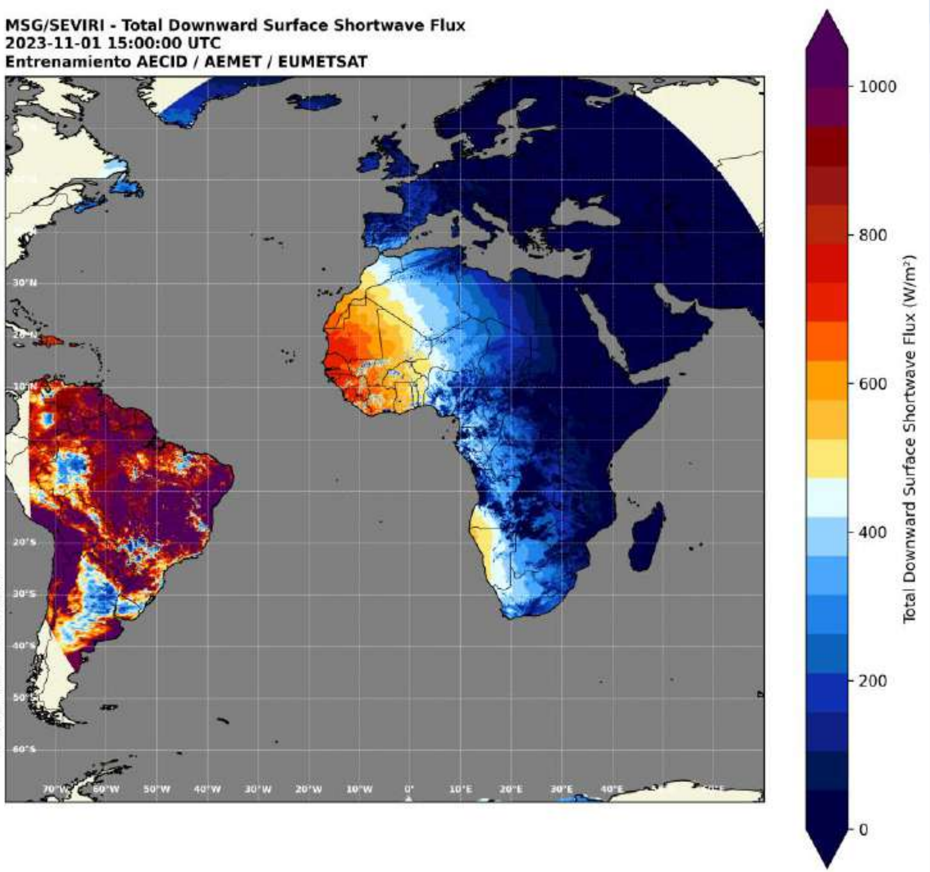
Nombre de la función | Parámetros

```
# Total Downward Surface Shortwave Flux (MSG)
# daily product (every 15 minutes), since 2004 (MS
plot_saf(saf_product = 'LSASAF_MSG_MDSSFTD_MSG-Dis
         date_saf='2023-11-01 15:00',
         extent=[-80.0, -40.00, -30.00, 10.00],
         cmap='custom', vmin=0, vmax=1050,
         coast_color='black',
         countries_color='black',
         grid_color='white',
         land_ocean=True,
         land_color='beige',
         ocean_color='gray',
         figsize=[10,10])
```

- Producto
- Fecha 'YYYY-MM-DD HH:MM'
- Región deseada
- Colormap, mínimo y máximo
- Colores
- Máscara de océano y superficie
- Tamaño de la figura (pulgadas)







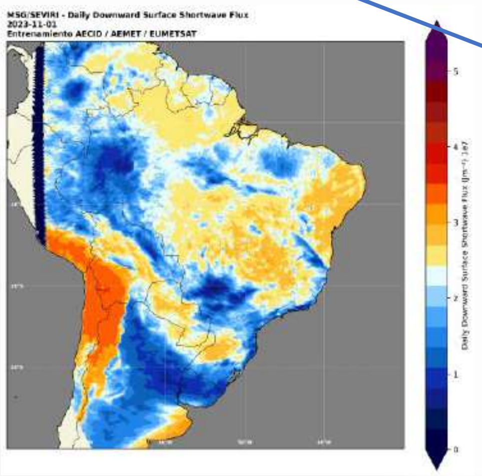
Ejemplo 13: Daily Downward Surface Shortwave Flux (MSG)

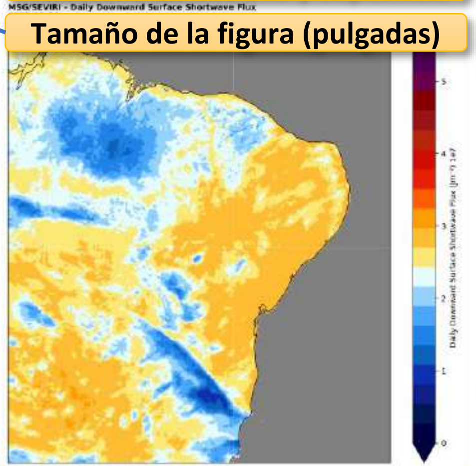
Ejemplo 13: Daily Downward Surface Shortwave Flux (MSG)

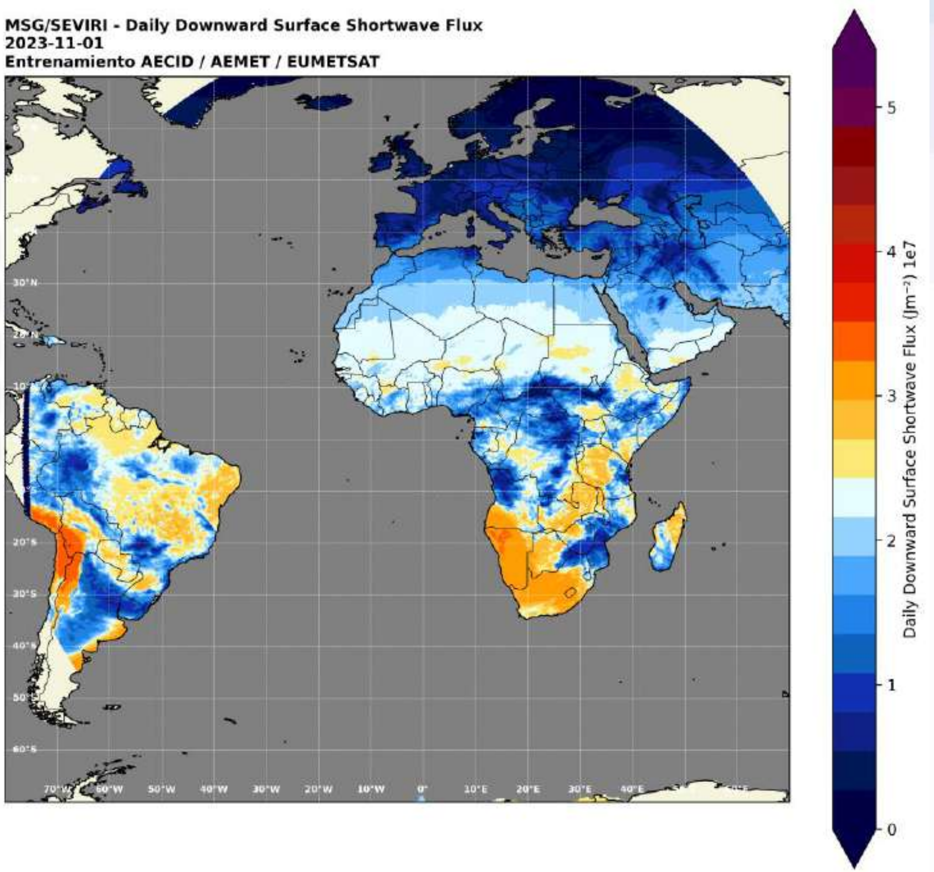
Nombre de la función | Parámetros

```
# Daily Downward Surface Shortwave Flux (MSG)
# daily product (00:00), since 2004 (MSG footprint
plot_saf(saf_product = 'LSASAF_MSG_DIDSSF_MSG-Disk',
         date_saf='2023-11-01 00:00',
         extent=[-80.0, -40.00, -30.00, 10.00],
         cmap='custom', vmin=0, vmax=5.4,
         coast_color='black',
         countries_color='black',
         grid_color='white',
         land_ocean=True,
         land_color='beige',
         ocean_color='gray',
         figsize=[10,10])
```

- Producto
- Fecha 'YYYY-MM-DD HH:MM'
- Región deseada
- Colormap, mínimo y máximo
- Colores
- Máscara de océano y superficie
- Tamaño de la figura (pulgadas)







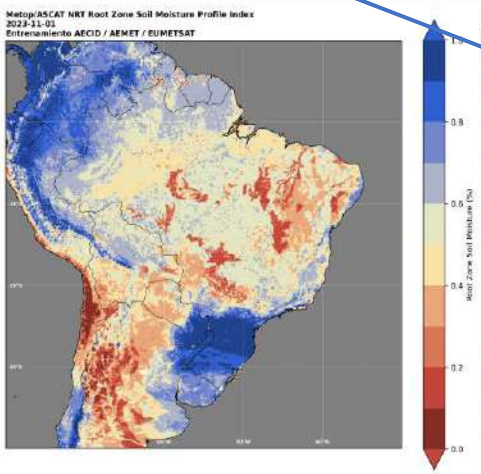
Ejemplo 14: Root Zone Soil Moist. Prof. Index (Metop/ASCAT)

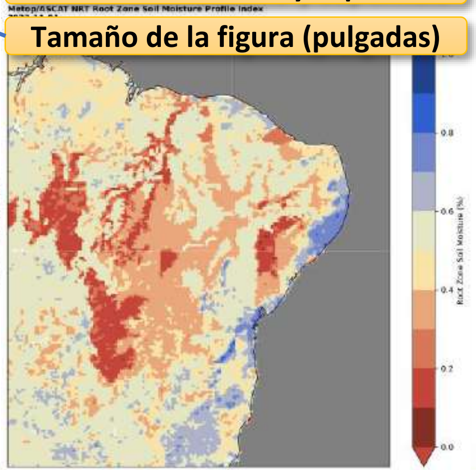
Ejemplo 14: Root Zone Soil Moisture Profile Index (Metop/ASCAT)

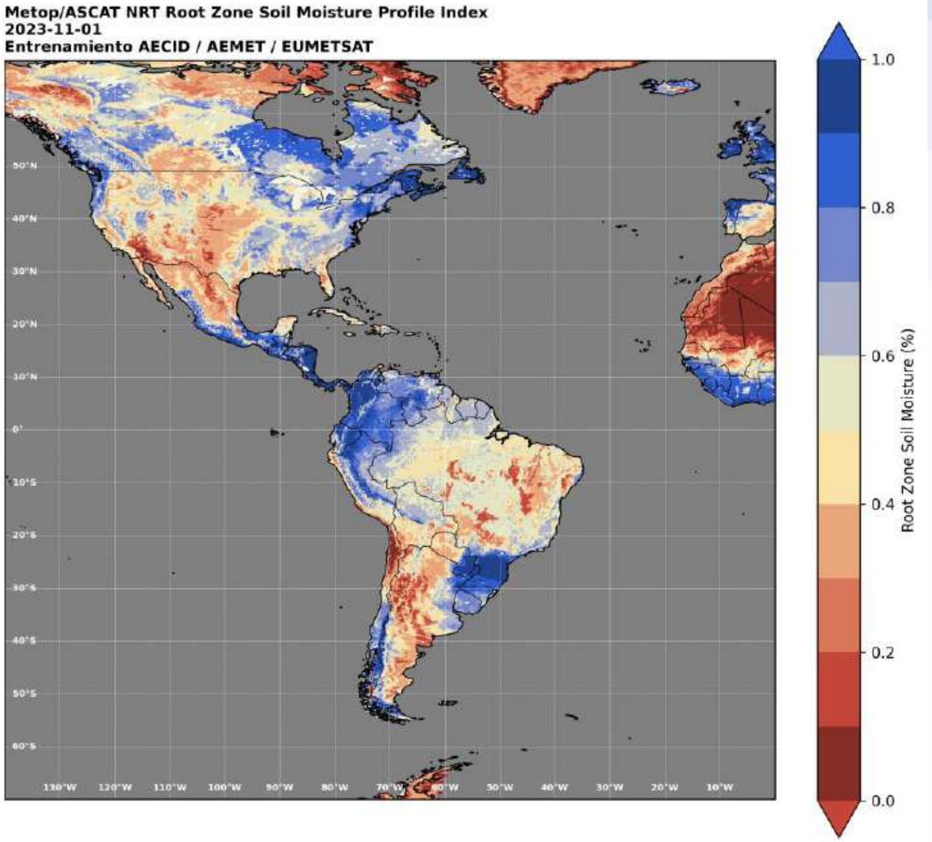
Nombre de la función | Parámetros

```
# Root Zone Soil Moisture Profile Index (Metop/ASCAT)
# daily product
plot_saf(saf_product = 'h26_cur_mon_nc',
         date_saf='2023-11-01 00:00',
         extent=[-140.0, -70.00, 0.00, 70.00],
         cmap='custom', vmin=0, vmax=1,
         coast_color='black',
         countries_color='black',
         grid_color='white',
         land_ocean=True,
         land_color='beige',
         ocean_color='gray',
         figsize=[10,10])
```

- Producto
- Fecha 'YYYY-MM-DD HH:MM'
- Región deseada
- Colormap, mínimo y máximo
- Colores
- Máscara de océano y superficie
- Tamaño de la figura (pulgadas)







Ejemplo 15: Precip. / Soil Moisture Int. Product (Multimission)

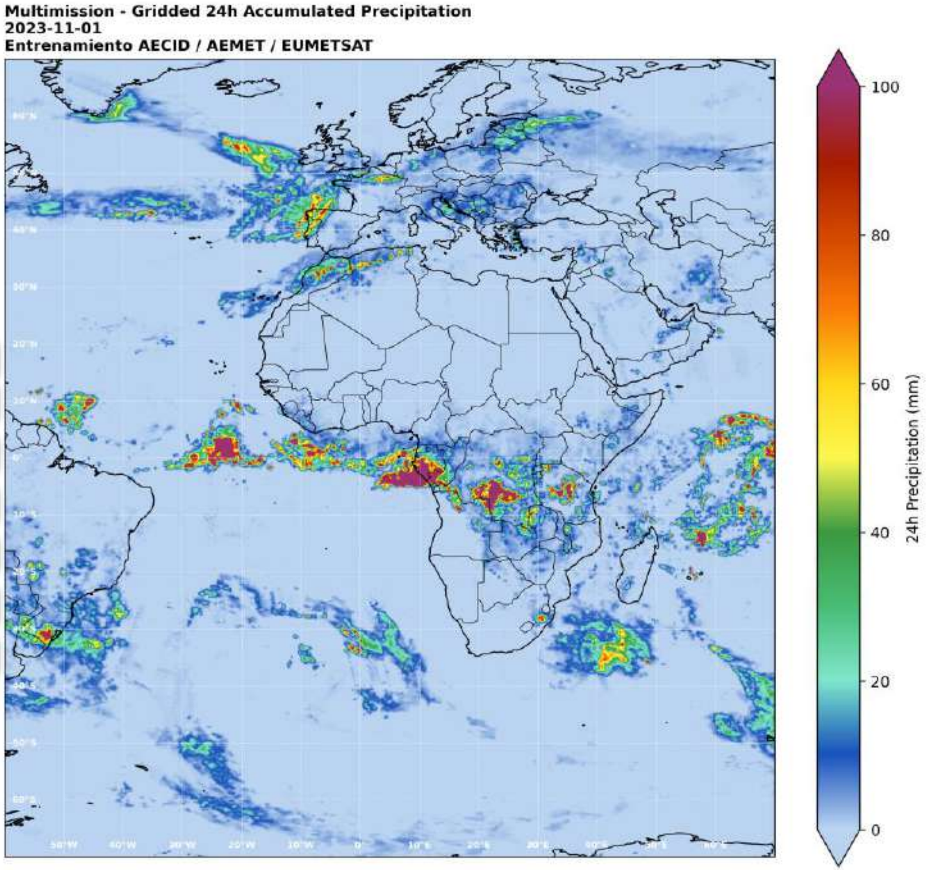
Ejemplo 15: Precipitation / Soil Moisture Integrated Product (Multimission)

Nombre de la función

Parámetros

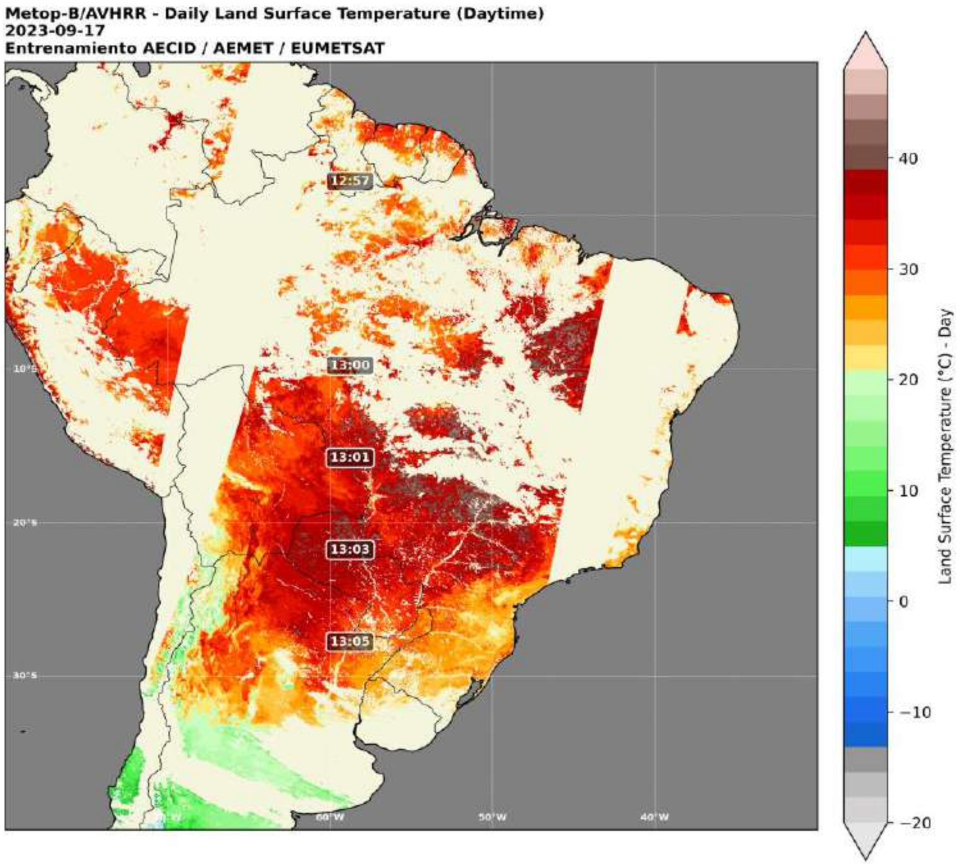
```
# daily product (min lon: -60)
plot_saf(saf_product = 'h64_cur_mon_data',
         date_saf='2023-11-01 00:00',
         extent=[-60.0, -60.00, 20.00, 20.00],
         cmap='custom', vmin=0, vmax=100,
         coast_color='black',
         countries_color='black',
         grid_color='white',
         land_ocean=False,
         land_color='beige',
         ocean_color='skyblue',
         figsize=[10,10])
```

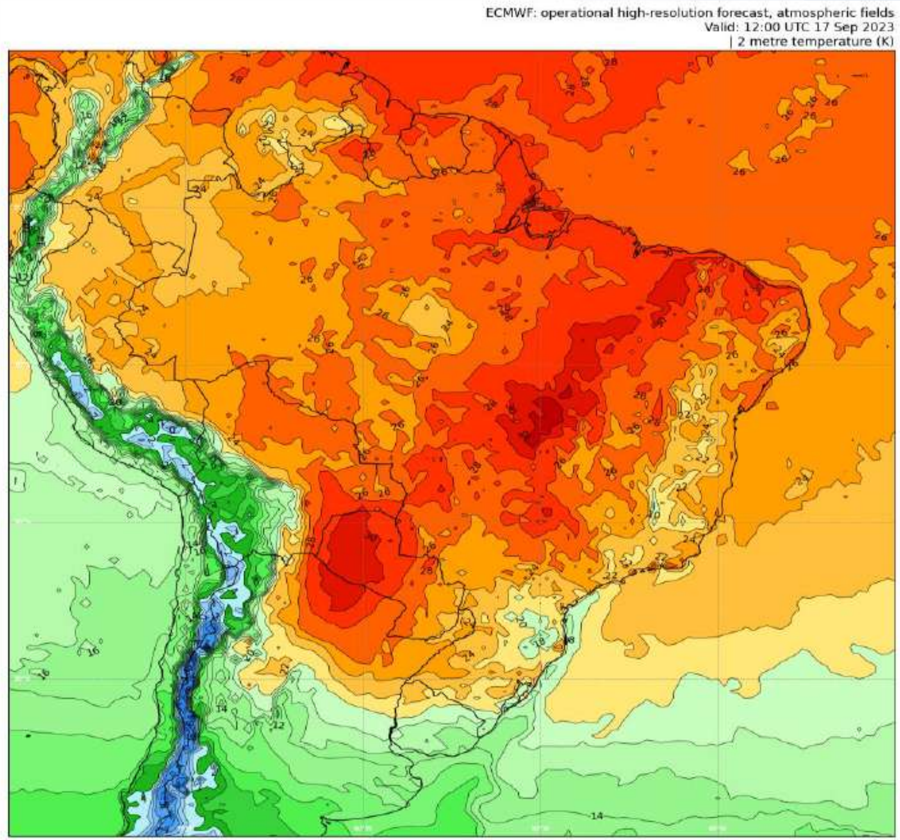
- Producto
- Fecha 'YYYY-MM-DD HH:MM'
- Región deseada
- Colormap, mínimo y máximo
- Colores
- Máscara de océano y superficie
- Tamaño de la figura (pulgadas)



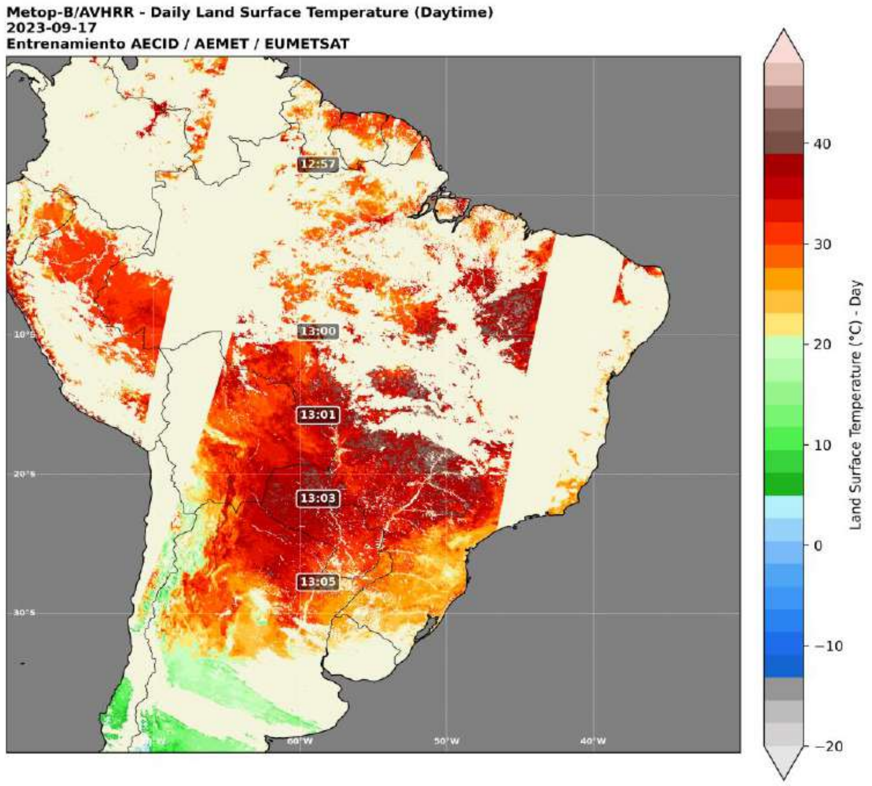
http://lsa-saf.eumetsat.int

Ejercicio 1: Ejemplo





Ejercicio 1: Ejemplo

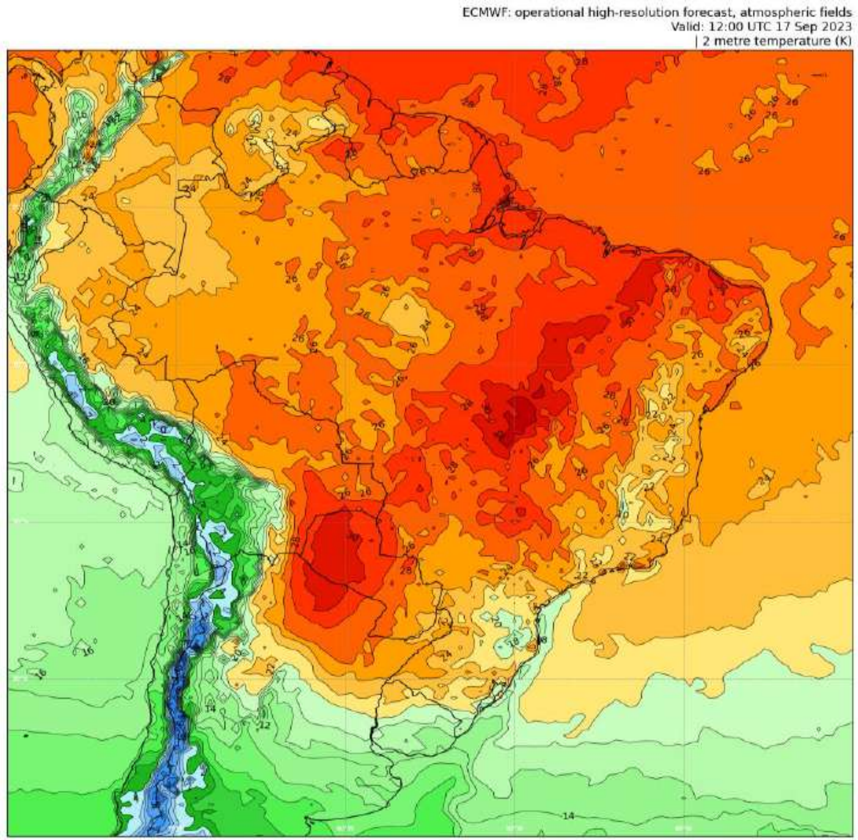


Ejemplo 2 (LSA-SAF)

Ejemplo 2: Land Surface Temperature (Metop/AVHRR) - Day

```
[6]  # Land Surface Temperature (Metop/AVHRR)
     # daily product (00:00), since 2015 (global)
     plot_saf(saf_product = 'LSASAF_M01-AVHR_EDLST-DAY_GLOBE',
              date_saf='2023-09-17 00:00',
              extent=[-80.0, -40.00, -30.00, 10.00],
              cmap='custom', vmin=-20, vmax=48,
              coast_color='black',
              countries_color='black',
              grid_color='white',
              land_ocean=True,
              land_color='beige',
              ocean_color='gray',
              figsize=[10,10])
```

Ejercicio 1: Ejemplo

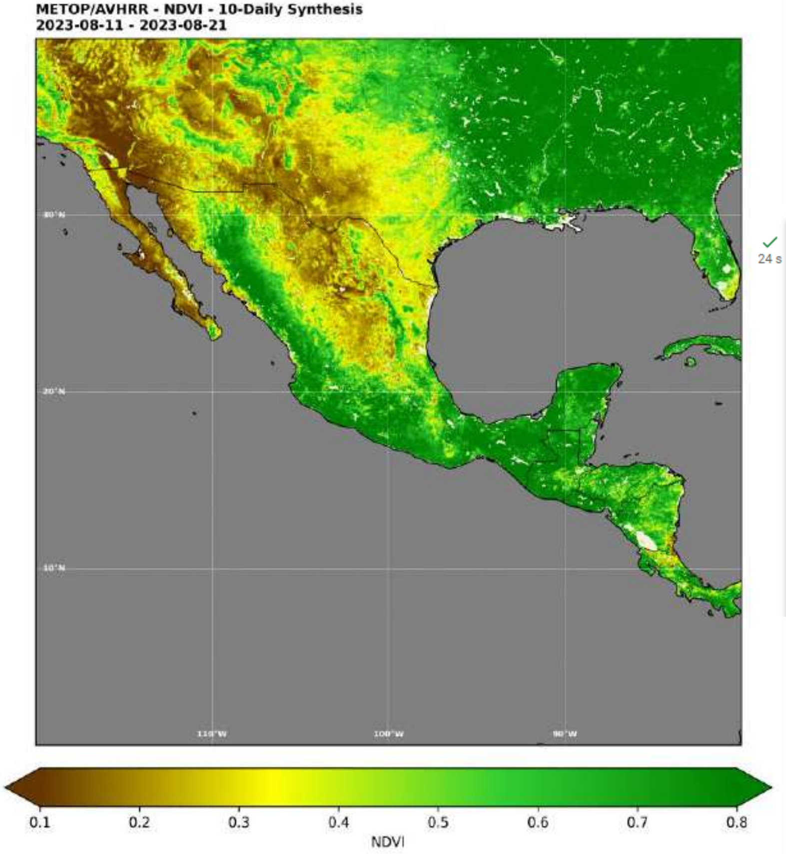


Ejemplo 8.13 (ECMWF)

Example 8.13: 2m Temperature

```
[7]
#-----------------------------------------------------------------------------------------------
# Training - Satellite Data Access and Processing - Example 8: Downloading and Visualizing NWP Data
# Author: Diego Souza (INPE/CGCT/DISSM)
#-----------------------------------------------------------------------------------------------

# open ecmwf data available for download since January 21 2022 00:00 ('2022-01-21 00:00')
# gfs data available for download since February 26 2021 00:00 ('2021-02-26 00:00')
# gfs products: 'pgrb2.0p25', 'pgrb2.0p50', 'pgrb2.1p00'

# 2 m temperature
plot_nwp(model='ecmwf',
         product='oper',
         date_nwp='2023-09-17 12:00',
         area='custom',
         extent=[-80.0, -40.00, -30.00, 10.00],
         fxx='date',
         var='2t', scale=1, offset=-273.15,
         level_hpa='',
         level_min=-20, level_max=48, level_int=2,
         plot_type='c_filled', contour_color='black',
         apply_cmap=True, cmap='custom',
         view_clabel=True, fontsize=10,
         linewidth=0.5, linestyle='solid',
         land_ocean=False, land_color = 'lightgray', ocean_color = 'white',
         alpha=1.0, title='new',
         figsize=[15,15])
```

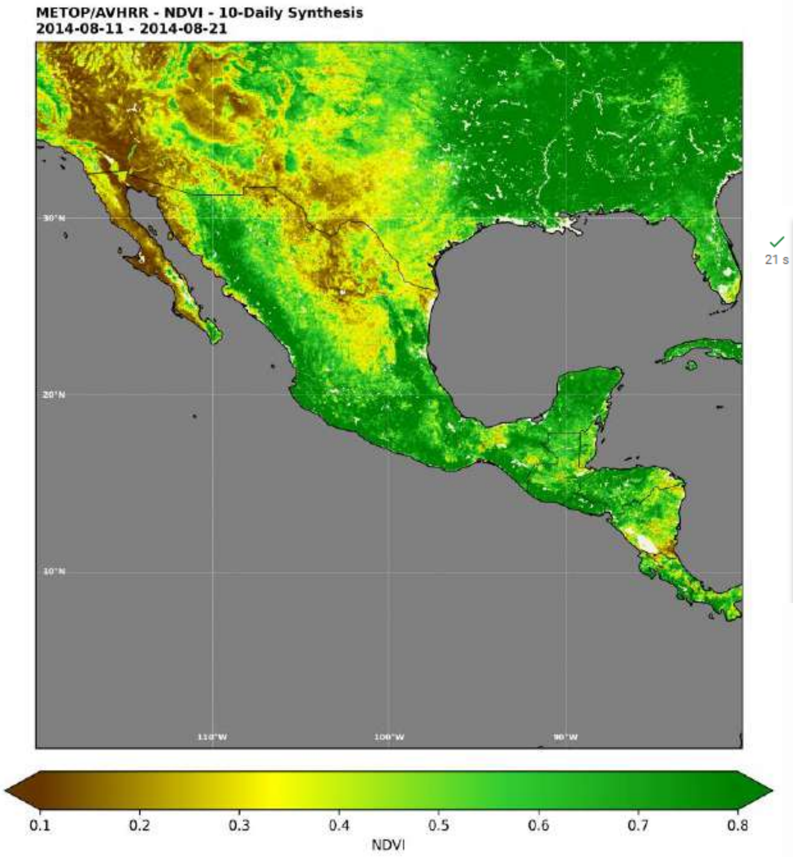
Ejercicio 2: Ejemplo



Ejemplo 5 (LSA-SAF)

```
# Normalized Difference Vegetation Index (Metop/AVHRR) - Central America
# possible days (01, 11 and 21), since 2007. Max. coordinates - extent = [-125, 0, -50, 50]
plot_saf(saf_product = 'METOP_AVHRR_S10_AMc_NDV',
         date_saf='2023-08-11 00:00',
         extent=[-120.0, 0.00, -80.00, 40.00],
         cmap='custom', vmin=0.1, vmax=0.8,
         coast_color='black',
         countries_color='black',
         grid_color='white',
         land_ocean=True,
         land_color='beige',
         ocean_color='gray',
         figsize=[10,10])
```

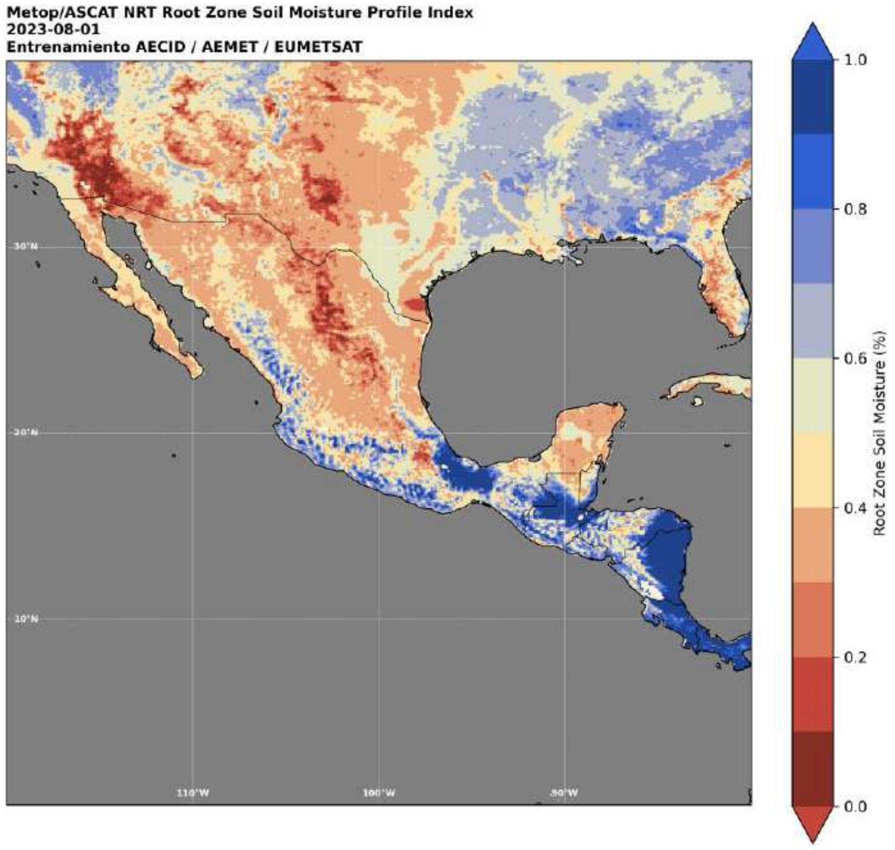
Ejercicio 2: Ejemplo



Ejemplo 5 (LSA-SAF)

```
# Normalized Difference Vegetation Index (Metop/AVHRR) - Central America
# possible days (01, 11 and 21), since 2007. Max. coordinates - extent = [-125, 0, -50, 50]
plot_saf(saf_product = 'METOP_AVHRR_S10_AMc_NDV',
         date_saf='2014-08-11 00:00',
         extent=[-120.0, 0.00, -80.00, 40.00],
         cmap='custom', vmin=0.1, vmax=0.8,
         coast_color='black',
         countries_color='black',
         grid_color='white',
         land_ocean=True,
         land_color='beige',
         ocean_color='gray',
         figsize=[10,10])
```

Ejercicio 2: Ejemplo

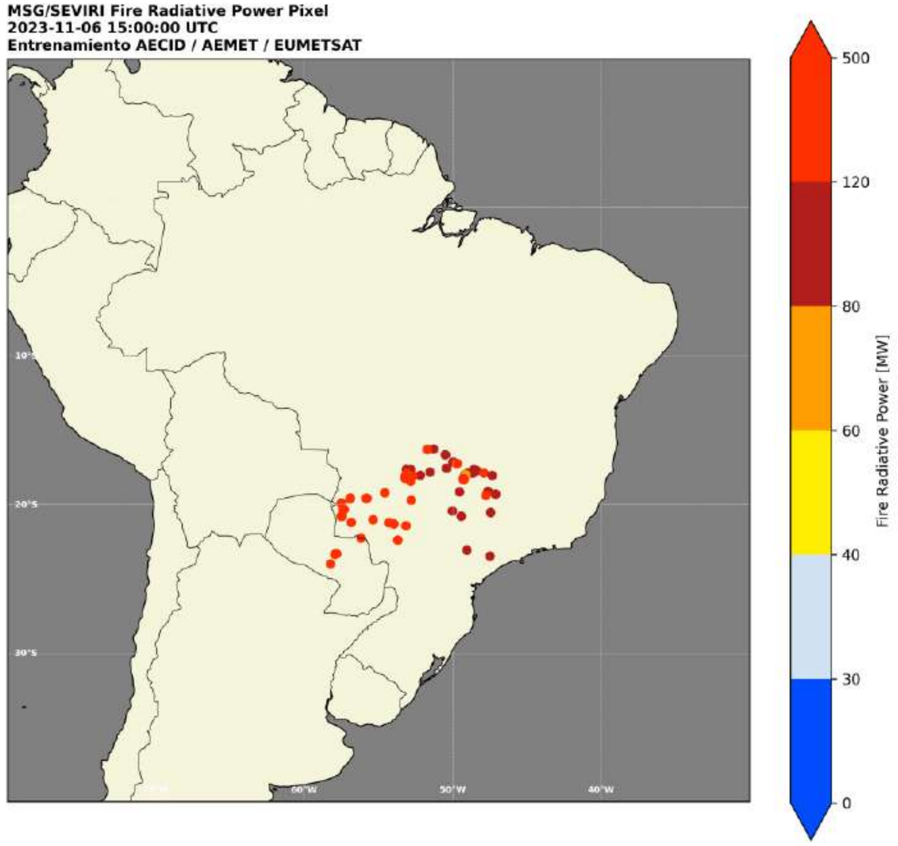


Ejemplo 14 (H-SAF)

14 s

```
# Root Zone Soil Moisture Profile Index (Metop/ASCAT)
# daily product
plot_saf(saf_product = 'h26_cur_mon_nc',
         date_saf='2023-08-01 00:00',
         extent=[-120.0, 0.00, -80.00, 40.00],
         cmap='custom', vmin=0, vmax=1,
         coast_color='black',
         countries_color='black',
         grid_color='white',
         land_ocean=True,
         land_color='beige',
         ocean_color='gray',
         figsize=[10,10])
```

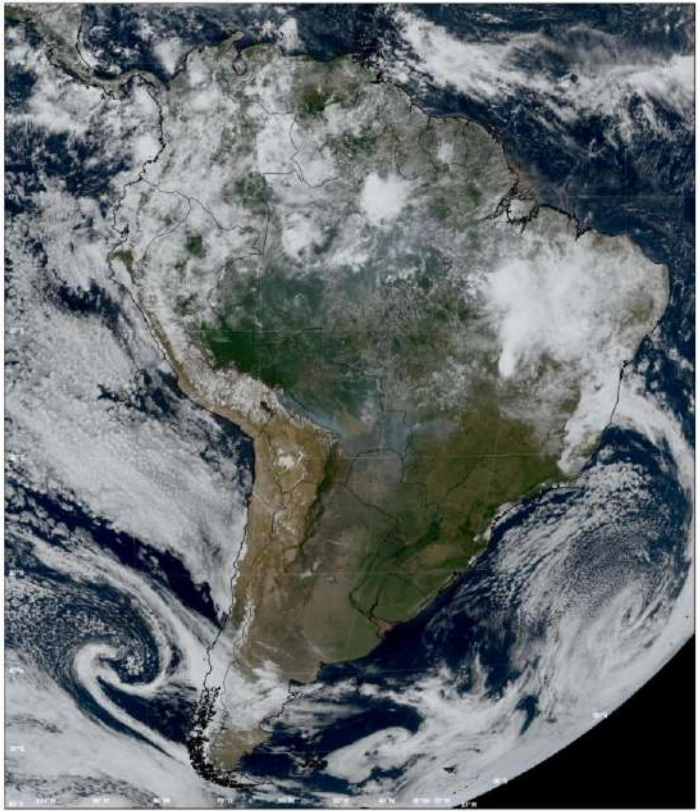
Ejercicio 3: Ejemplo



Ejemplo 6 (LSA SAF)

```
# Fire Radiative Power Pixel (MSG/SEVIRI)
# daily product (every 15 minutes), since 2004 (MSG footprint)
plot_saf(saf_product = 'LSASAF_MSG_FRP-PIXEL',
         date_saf='2023-11-06 15:00',
         extent=[-80.0, -40.00, -30.00, 10.00],
         cmap='custom', vmin=0, vmax=500,
         coast_color='black',
         countries_color='black',
         grid_color='white',
         land_ocean=True,
         land_color='beige',
         ocean_color='gray',
         figsize=[10,10])
```

Ejercicio 3: Ejemplo

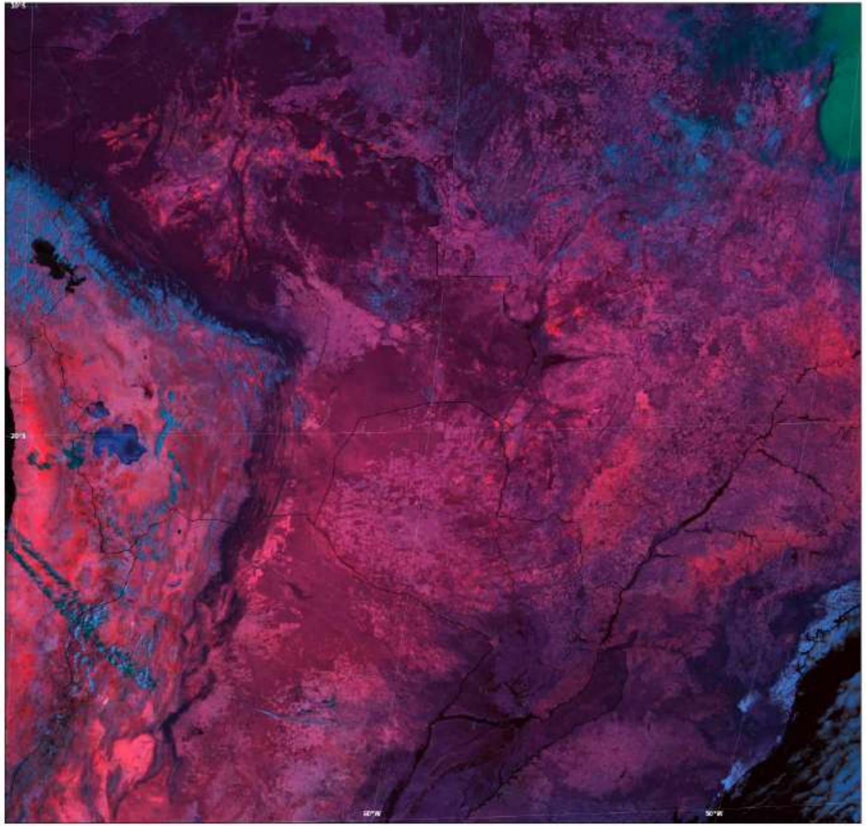


Ejemplo 2 (SATÉLITES)

```
plot_sat(sat='goes16',
         date_sat='2023-11-06 15:00',
         composite='true_color',
         area='custom_geo',
         extent=[-90.0, -60.00, -30.00, 15.00],
         coast_color='black',
         countries_color='black',
         grid_color='white',
         figsize=[15,15])
```

Ejercicio 3: Ejemplo

GOES-16 - cira_fire_temperature
2023-11-06 15:00 UTC
Entrenamiento AECID / AEMET / EUMETSAT

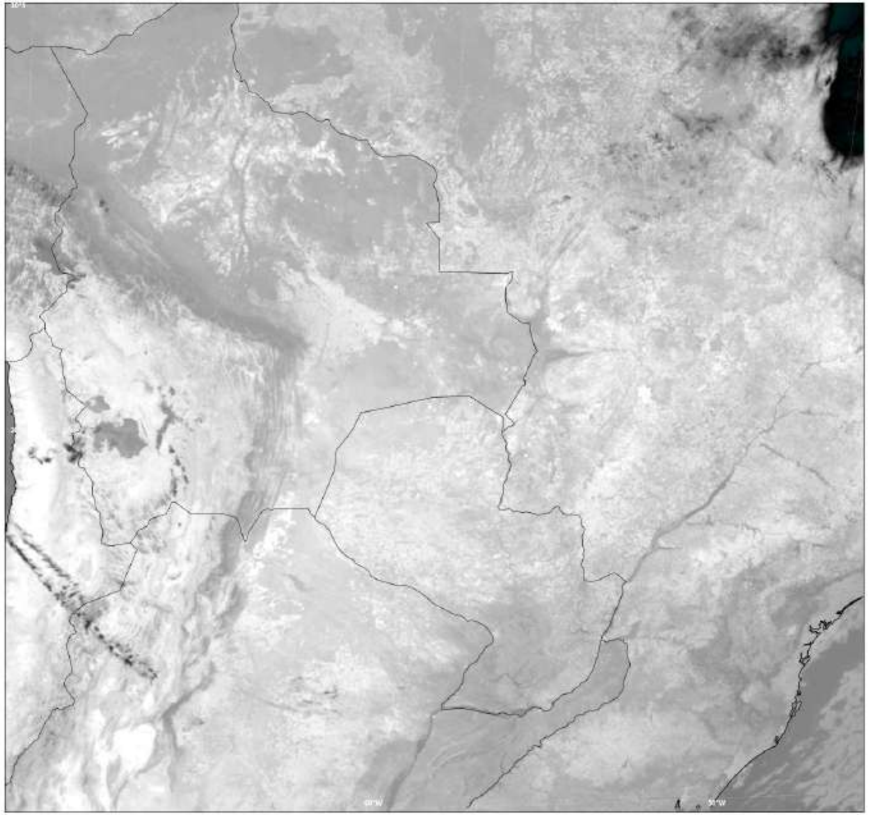


Ejemplo 2 (SATÉLITES)

```
plot_sat(sat='goes16',
         date_sat='2023-11-06 15:00',
         composite='cira_fire_temperature',
         area='custom_geo',
         extent=[-70.0, -30.00, -50.00, -10.00],
         coast_color='black',
         countries_color='black',
         grid_color='white',
         figsize=[15,15])
```

Ejercicio 3: Ejemplo

GOES-16 - C07 - 3.9 µm (3.8-4.0 µm)
2023-11-06 15:00 UTC
Entrenamiento AECID / AEMET / EUMETSAT



Ejemplo 2 (SATÉLITES)

```
plot_sat(sat='goes16',
         date_sat='2023-11-06 15:00',
         composite='C07',
         area='custom_geo',
         extent=[-70.0, -30.00, -50.00, -10.00],
         coast_color='black',
         countries_color='black',
         grid_color='white',
         figsize=[15,15])
```

Productos y Aplicaciones de Satélite en Latitudes Medias - Fase Presencial

Prácticas LSA SAF y H SAF

¡GRACIAS! ¿PREGUNTAS?

Diego Souza
diego.souza@inpe.br

DISSM - División de Satélites y Sensores Meteorológicos
CGCT - Coordinación General de Ciencias de la Tierra
INPE - Instituto Nacional de Investigaciones Espaciales

MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E INOVAÇÃO

GOVERNO FEDERAL BRASIL UNIÃO E RECONSTRUÇÃO