**EUMETSAT Satellite Application Facility on Climate Monitoring**

# CLAAS-3 Auxiliary Data User Guide

**DOI:10.5676/EUM_SAF_CM/CLAAS/V003**

| | |
|---|---|
| Reference Number: | SAF/CM/DWD/AUX/SEV/CLD |
| Issue/Revision Index: | 3.0 |
| Date: | 15.11.2022 |

### Document Signature Table

| | Name | Function | Signature | Date |
|---|---|---|---|---|
| Authors | Irina Solodovnik<br>Martin Stengel | CM SAF scientist (DWD)<br>CM SAF scientist (DWD) | | 15.11.2022 |
| Editor | | | | |
| Approval | | | | |
| Release | | | | |

### Distribution List

| Internal Distribution | |
|---|---|
| **Name** | **No. Copies** |
| DWD Archive | 1 |
| CM SAF Team | 1 |

| External Distribution | | |
|---|---|---|
| **Company** | **Name** | **No. Copies** |
| PUBLIC | | 1 |

This document is barrier-free.

## Document Change Record

| Issue/ Revision | Date | DCN No. | Changed Pages/Paragraphs |
|---|---|---|---|
| 3.0 | 15.11.2022 | SAF/CM/DWD/AUX/SEV/CLD/3.0 | Final version – applicable to the CLAAS-3 data record |

## Reference Documents

| Reference | Title | Code |
|---|---|---|
| RD 1 | Product User Manual SEVIRI Cloud Products Edition 3 (CLAAS-3) | SAF/CM/KNMI/PUM/SEV/CLD/3.1, 08.08.2022 |
| RD 2 | | |
| RD 3 | | |

## Table of Contents

## List of Tables

## List of Figures

# 1 Introduction

The third edition of CM SAF’s **CL**oud property d**A**t**A**set using **S**EVIRI (CLAAS-3) provides cloud properties derived from the SEVIRI imager onboard geostationary METEOSAT Second Generation (MSG) satellites. Two categories of products are available:

- **Level 2 (L2)**: 15 minutes repeat cycle, native SEVIRI grid (3km NADIR resolution)
- **Level 3 (L3)**: Daily and monthly aggregations (averages and histograms) on regular 0.05° and 0.25° grids

The spatial resolution of the different L3 products is given in Table 1-1. For detailed information about the CLAAS-3 product suite, see the Product User Manual RD 1.

**Table 1-1** Spatial resolution of CLAAS-3 products.

| Product | Resolution |
|---|---|
| Daily Mean | 0.05° |
| Monthly Mean | 0.05° |
| Monthly Histogram (CTO, CWP) | 0.05° |
| Monthly Histogram (JCH) | 0.25° |
| Monthly Mean Diurnal Cycle | 0.25° |

In order to facilitate work with the CLAAS-3 dataset, we provide the following auxiliary data:

**Table 1-2** Available auxiliary data and corresponding netCDF variable names. In the L2 file, y and x denote the SEVIRI image rows and columns, respectively.

| Parameter | netCDF Variable Name | | |
|---|---|---|---|
| | **L2** | **L3, 0.05°** | **L3, 0.25°** |
| *Latitude* | lat(georef_offset_corrected, y, x) | lat(lat) | lat(lat) |
| *Longitude* | lon(georef_offset_corrected,y, x) | lon(lon) | lon(lon) |
| *Land-Sea-Mask* | lsm(georef_offset_corrected,y, x) | lsm(lat, lon) | / |
| *Land Fraction* | / | / | land_fraction(lat, lon) |
| *Altitude* | alt(georef_offset_corrected,y, x) | alt(lat, lon) | alt(lat, lon) |
| *Satellite-Zenith-Angle* | satzen(georef_offset_corrected,lon0, y, x) | satzen(lon0, lat, lon) | satzen(lon0, lat, lon) |
| *Surface area covered by each* | pixel_area(georef_offset_corrected,y, x) | pixel_area(lat, lon) | pixel_area(lat, lon) |
| *Dimensions* | x/y=3712, lon0=5, georef_offset_corrected=2 | lat/lon=3600, lon0=5 | lat/lon=720, lon0=5 |

As the satellite position changes over time, Satellite Zenith Angles are available for the most prominent satellite positions (see Section 0 for details).

# 2 L2 Auxiliary Data

CLAAS-3 L2 products are distributed on the native SEVIRI grid, which is an irregular grid. Hence, latitude and longitude coordinates providing the georeferencing of each pixel are two-dimensional. However, *lat* and *lon* fields in the CLAAS-3 L2 auxiliary file are 3-dimensional. The additional dimension is needed, because coordinates change throughout the dataset. Until December 2017 SEVIRI L1.5 data is shifted by 1.5km SSP North and West against the nominal GEOS projection. Since December 2017 this offset has been corrected. CLAAS-3 L2 data is hence provided on the shifted grid before and on the corrected grid after December 2017. Every L2 product contains the corresponding flag denoting the shift correction, so L2 data can be easily matched with the auxiliary fields using that flag.

Although the satellite position changes over time, the SEVIRI images are reprojected so that (lat,lon)=(0,0) is always in the image centre. That means the coordinates - except for the shift - are valid throughout the dataset. However, this is not the case for satellite zenith angles, see Section 0.

Land-Sea-Mask, altitude and acquisition time were taken from output of the the PPSSEV v2018-patchCMSAF-May2021 software package by the NWC SAF (SAF in support of Nowcasting and Very Short Range Forecasting) which was used in CLAAS-3. Pixel area and satellite zenith angle were directly derived from the latitude/longitude coordinates.

Auxiliary data for L2 products is stored in the netCDF4 file *claas3_level2_aux_data.nc*. Figure 2-1 shows its content.

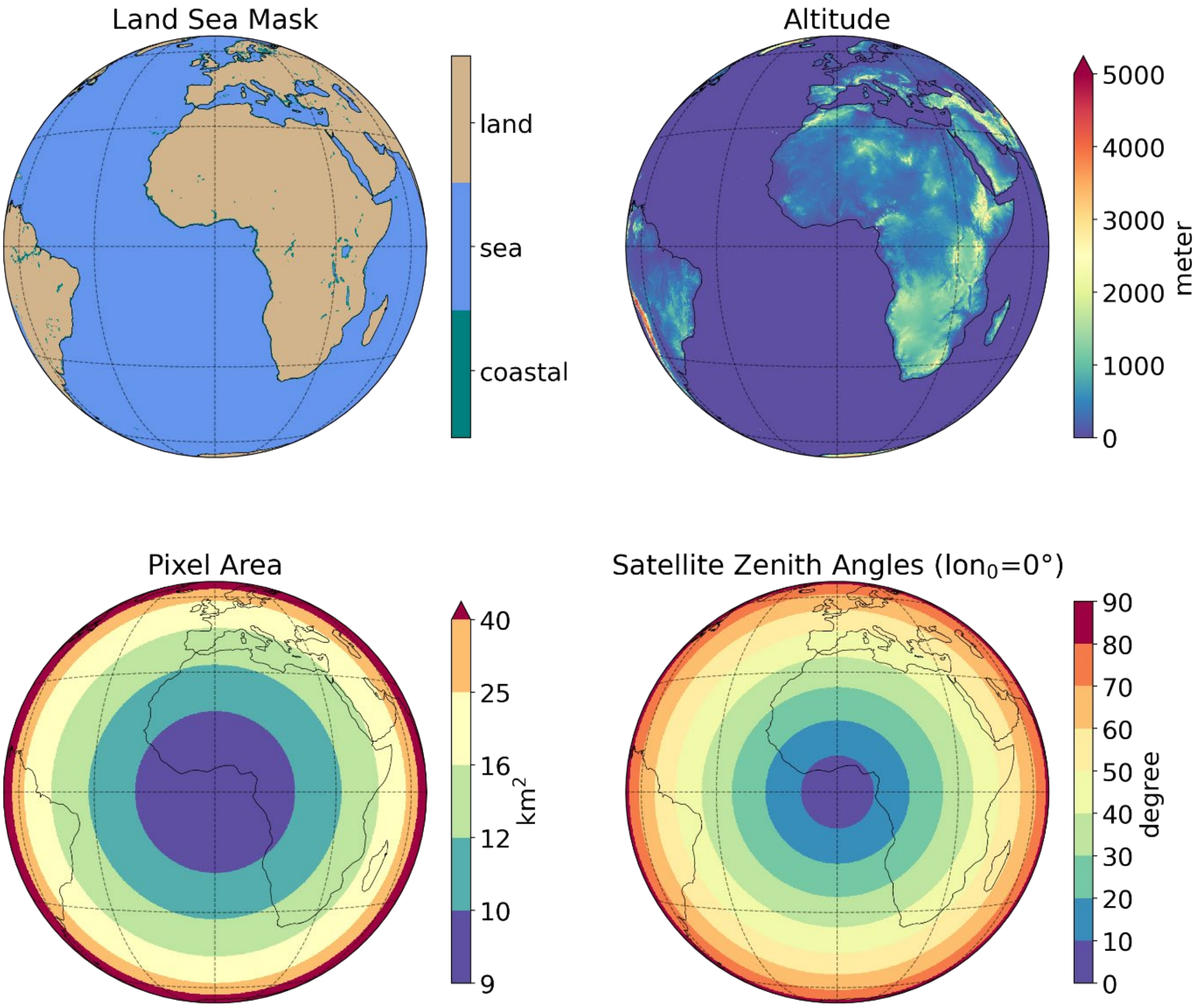

**Figure 2-1** Example plots of available L2 auxiliary data.

# 3 L3 Auxiliary Data

L3 products are distributed on regular 0.05° and 0.25° grids. The corresponding auxiliary data are described in the following two sections. Land-Sea-Mask, altitude and pixel area are valid throughout the dataset. However, the satellite position changes over time so that satellite zenith angles are not constant. See section 4 for details.

## 3.1 Regular 0.05° Grid

The 0.05° auxiliary data are stored in the netCDF4 file *class3_level3_aux_data_005deg.nc*, Figure 3-1 shows example plots of its contents. Land-Sea-Mask and altitude were obtained by remapping the corresponding L2 fields to the 0.05° grid using a nearest neighbour approach. Pixel area as well as satellite zenith angle were derived directly from the latitude/longitude coordinates.

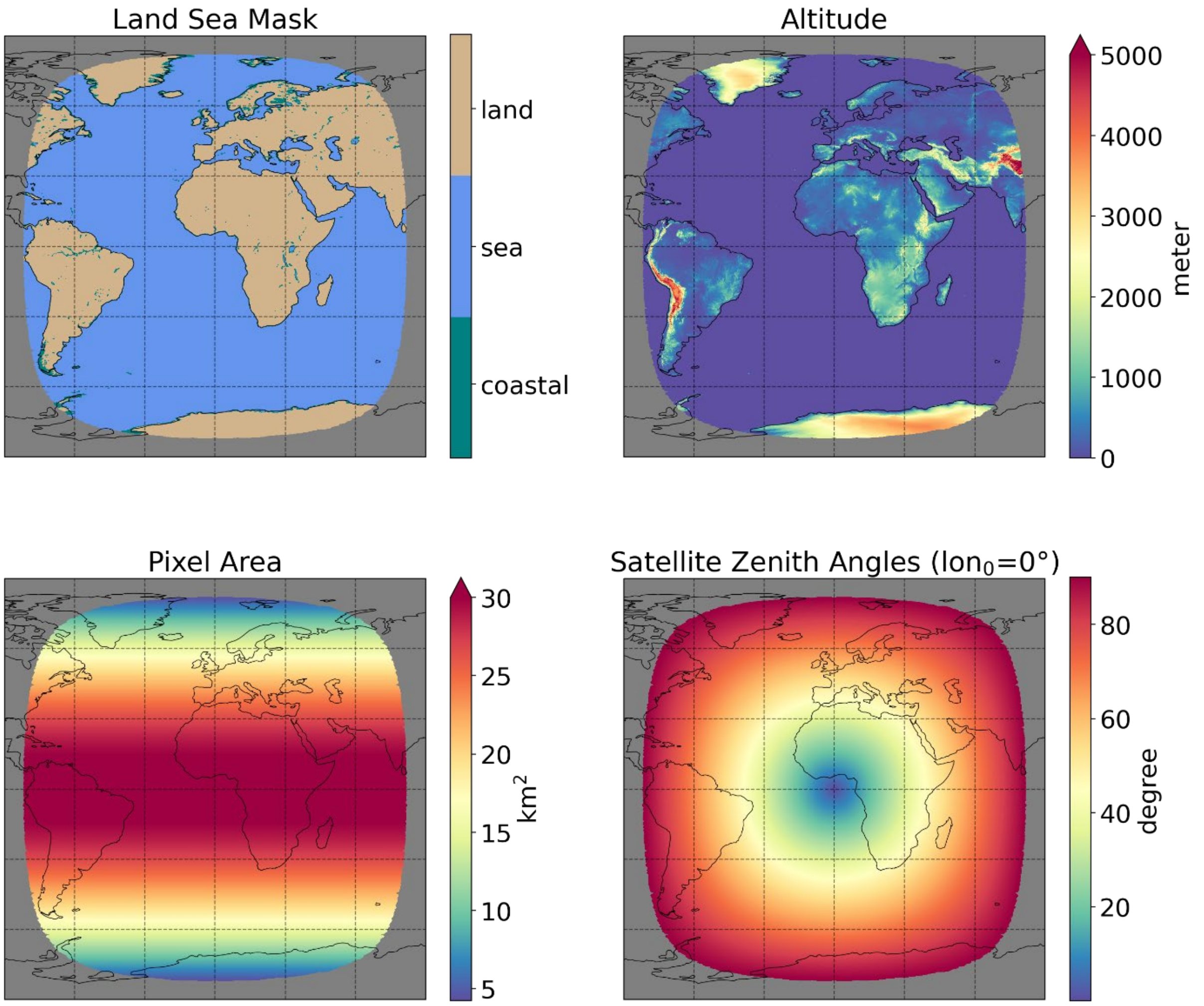

**Figure 3-1** Example plots of available L3 auxiliary data on the 0.05° grid.

## 3.2 Regular 0.25° Grid

The 0.25° auxiliary data are stored in the netCDF4 file *class3_level3_aux_data_025deg.nc*, its contents are shown in Figure 3-2. In contrast to the other grids, we provide mean quantities here (except for pixel area), which were obtained by area-weighted averaging of the 0.05° auxiliary data in 5x5 pixel boxes. This average is representing the 25 pixels more accurate than just the nearest neighbour. The discrete land sea mask is consequently transformed into a continuous land fraction. The pixel area was derived directly from the latitude/longitude coordinates.

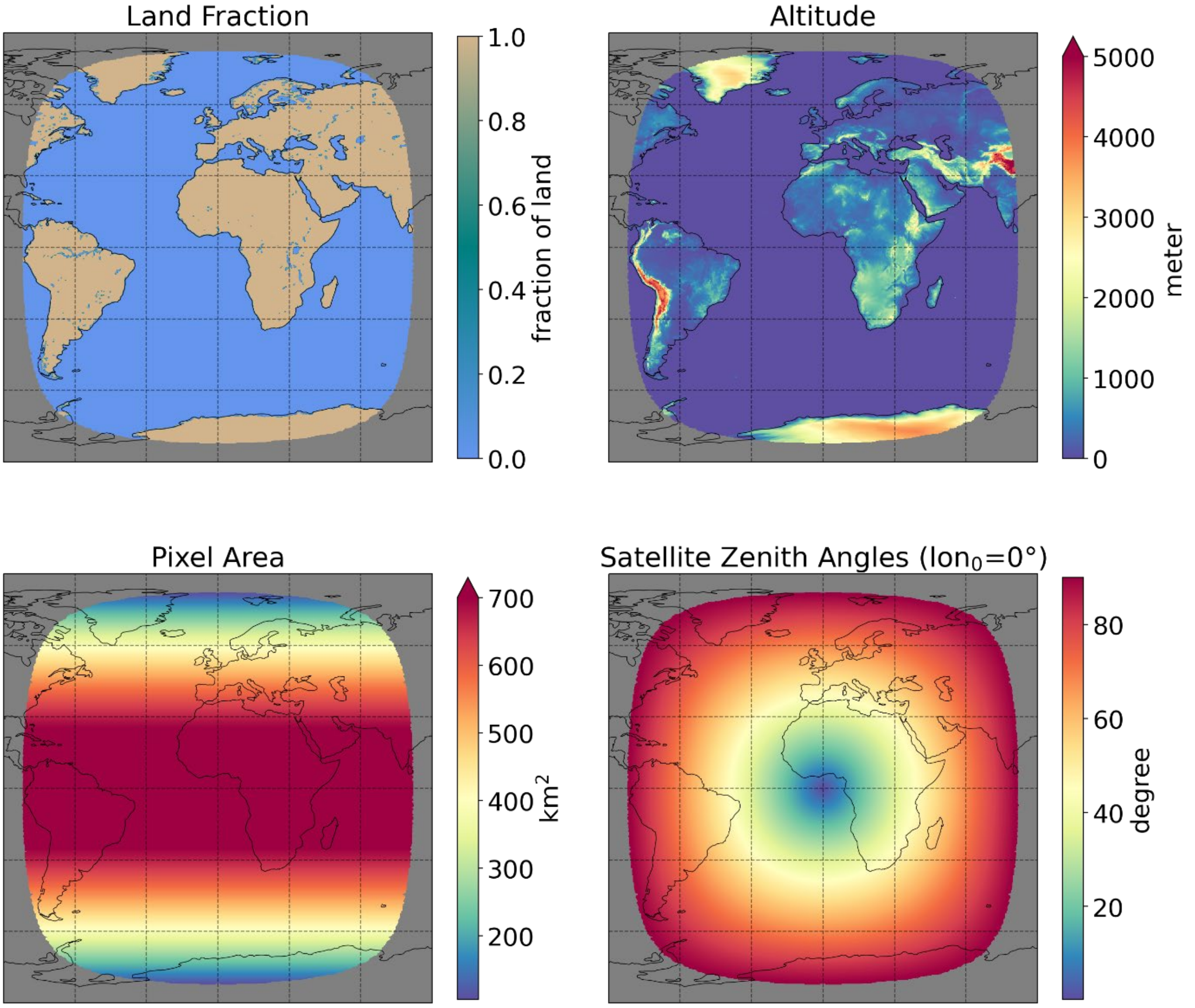

**Figure 3-2** Example plots of available L3 auxiliary data on the 0.25° grid.

# 4 Satellite Zenith Angles

During the CLAAS-3 time series several satellite maneuvers were performed (Figure 4-1). In contrast to the geographical coordinates, which can be reprojected, viewing angles depend on the satellite position. That is why we provide satellite zenith angles for the most prominent subsatellite points: 3.4 °W, 0°, 3.5 °E, 9.5 °E, 41.5 °E.

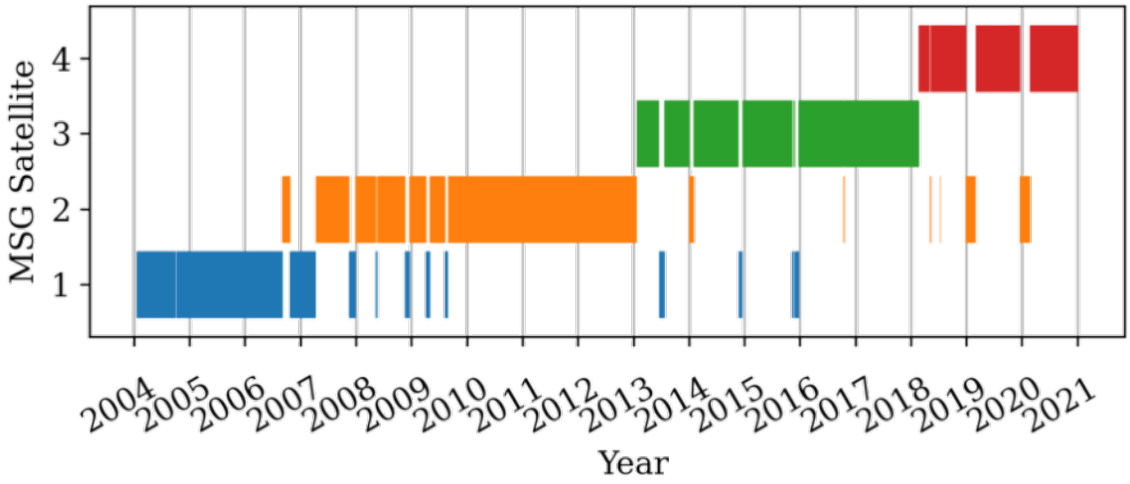

**Figure 4-1** Overview of the SEVIRI measurement record used as basis for the cloud generation of the CLAAS-3 CDR. Short-term data gaps are shown enlarged by a factor of 5 for better visibility.

# 5 Examples

Python and IDL code examples below show how to read the CLAAS-3 L2 auxiliary data and determine the subsatellite point for a specific date.

## 5.1 Python Example

```
import netCDF4
import datetime

# Read MSG1 aux data
ds = netCDF4.Dataset('claas3_level2_aux_data.nc')
msg1_lon0_id = ds.variables['msg1_lon0_id'][:]
msg1_lon0_time_bounds = ds.variables['msg1_lon0_time_bounds'][:]
lon0_vals = ds.variables['lon0'][:]
# Choose point in time
t = netCDF4.date2num(
    datetime.datetime(2009, 7, 1, 12, 15),
    calendar=ds.variables['msg1_lon0_time_bounds'].calendar,
    units=ds.variables['msg1_lon0_time_bounds'].units
)

# Determine subsatellite point
num_pos = len(msg1_lon0_id) # Number of positions occupied by MSG1
idx = 0 # Index of matching time interval
for ipos in range(num_pos):
    if msg1_lon0_time_bounds[ipos, 0]<=t<msg1_lon0_time_bounds[ipos, 1]:
        # Found the matching time interval!
        break
    else:
        idx += 1

if idx == num_pos:
    # No matching time interval found
    raise ValueError('Timestamp is either pre-launch or outside the temporal coverage of the dataset')

lon0 = lon0_vals[msg1_lon0_id[idx]]
satzen = ds.variables['satzen'][msg1_lon0_id[idx], :, :]
print "Subsatellite point: {0} degrees east".format(lon0)
```

## 5.2 IDL Example

```
pro claas_lon0

; Read aux data
fileID = ncdf_open('claas3_level2_aux_data.nc')
```

```
varID = NCDF_Varid(fileID,'msg1_lon0_id')
NCDF_Varget, fileID, varID, msg1_lon0_id
varID = NCDF_Varid(fileID,'msg1_lon0_time_bounds')
NCDF_Varget, fileID, varID, msg1_lon0_time_bounds
varID = NCDF_Varid(fileID,'lon0')
NCDF_Varget, fileID, varID, lon0_vals
varID = NCDF_Varid(fileID,'satzen')
NCDF_Varget, fileID, varID, satzens
NCDF_Close, fileID

; Choose point in time (time unit is 'days since 1970-01-01 00:00')
t = julday(7,1,2009,12,15) - julday(1,1,1970,0,0)

; Determine subsatellite point
num_pos = n_elements(msg1_lon0_id) ; Number of positions occupied by MSG1
idx = 0 ; Index of matching time interval
for ipos = 0,num_pos-1 do begin
    if ( t ge msg1_lon0_time_bounds[0,ipos] and $
        t lt msg1_lon0_time_bounds[1,ipos] ) then begin
        break
    endif else begin
        idx++
    endelse
endfor

if idx eq num_pos then begin
    print,'Timestamp is either pre-launch or outside the '
    print,'temporal coverage of the dataset'
endif else begin
    lon0 = lon0_vals[msg1_lon0_id[idx]]
    satzen = satzens[*, *, msg1_lon0_id[idx]]
    print, "Subsatellite point (degrees east):"
    print, lon0
endelse
end
```