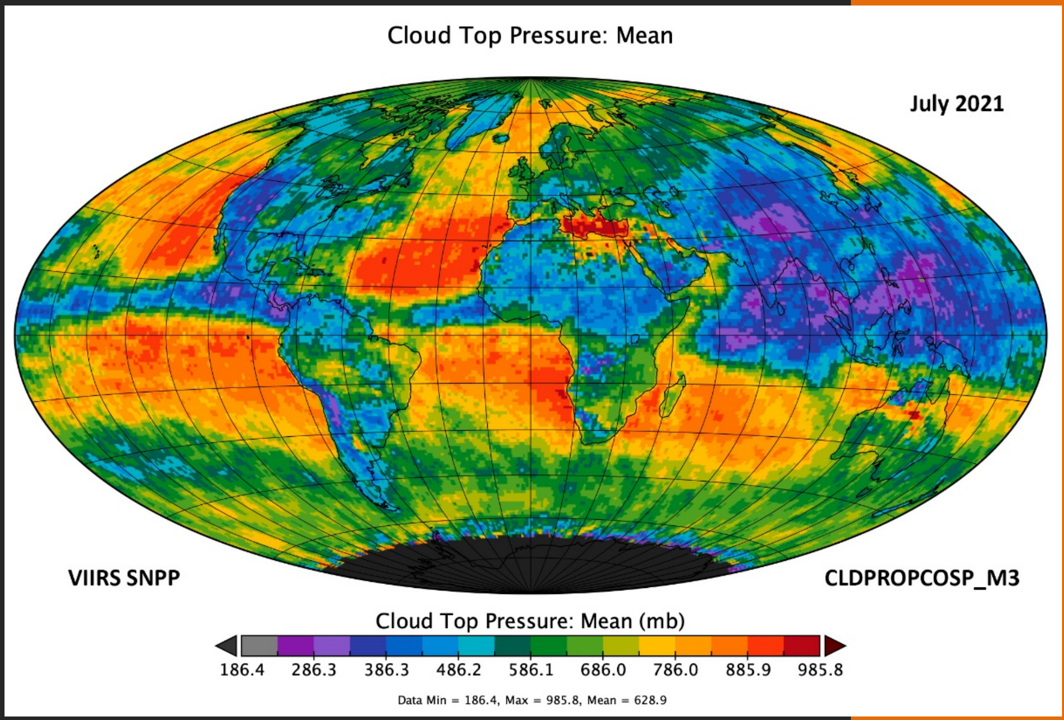
Level-3 COSP Cloud Properties (CLDPROPCOSP_L3) (2023/2024 Production)

VIIRS SNPP, VIIRS NOAA20, MODIS Aqua Global Gridded Product for Climate Studies

User Guide



MODIS Standard Atmosphere Level-3 CLDPROPCOSP (v1.1, 2023/2024 Production)
L3 VIIRS SNPP, VIIRS NOAA20, & MODIS Aqua
Global Gridded Products for Daily (D3) and Monthly (M3)
User Guide

L3 CLDPROPCOSP User Guide, Version 3.0, 26 June, 2024

Applies to COSP Files Version 011 (v1.1)

PAUL HUBANKS[1], ROBERT PINCUS[2], STEVEN PLATNICK[3], KERRY MEYER[3]

[1] Adnet Systems, Lanham, MD

[2] Lamont-Doherty Earth Observatory, Earth Institute, Columbia University, Palisades, NY

[3] Earth Sciences Division, NASA Goddard Space Flight Center, Greenbelt, MD

TABLE OF CONTENTS

1.0. Introduction

1.0.1. Background

NASA ushered in a new generation of global imager observations of the Earth with MODIS on the EOS Terra and Aqua missions, which provided unique spectral capability relative to earlier global imagers, allowing for the retrieval of geophysical parameters key to understanding changes in the Earth's land surface, ocean, and atmosphere. To date, these imagers have proved remarkably successful, exceeding their design lives to produce 24-year (Terra) and 22-year (Aqua) climate data records that are expected to continue into the mid 2020s.

Nevertheless, detecting climate trends, including those related to clouds, is a multi-decadal endeavor (typically greater than 2-3 decades) given the uncertainties of current sensors [*Wielicki et al.*, 2013]. Therefore, while the EOS Aqua and Terra missions are projected to extend into the early 2020s, those data records alone are not sufficient for establishing climate trends. Extending these EOS data records to new sensors is therefore imperative.

For continuing MODIS-like global cloud records, the only viable imager is VIIRS (375-750 m nadir resolution) on Suomi NPP (SNPP) and the operational NOAA JPSS series (starting with JPSS-1/NOAA-20 in late 2017, with subsequent satellites expected to provide coverage into the mid-2030s). For this reason, a merged MODIS-VIIRS cloud record to serve the science community in the coming decades requires different algorithm approaches than those used for MODIS alone.

Starting in 2013, NASA formed the SNPP Science Team to develop continuity algorithms that would bridge the EOS and SNPP observation era. Due to the differing characteristics between MODIS and VIIRS, the approach taken was to develop a continuity algorithm that uses common (or near-common) logic to be run on both the MODIS and VIIRS data streams.

To advance research on detecting climate trends, including those related to clouds, a new project began development: Level-3 Atmosphere for **C**FMIP (Cloud Feedback Model Intercomparison Project) **O**bservation **S**imulator **P**ackage or **COSP**.

Remote satellite observations provide a valuable source of information for understanding errors in climate models. Retrievals of particle size, in particular, offer a unique view of cloud-aerosol interactions. A "MODIS simulator", part of the COSP package, facilitates the comparison of models and observations by producing synthetic MODIS observations from climate model integrations. Output from the MODIS simulator is part of the formal request for data in the current generation of intercomparison projects (CMIP6). The MODIS request is consistent with similar requests from other platforms and minimizes data volumes, but the output requested unfortunately does not map easily onto standard MODIS Level-3 monthly products.

Thus a new dataset (CLDPROPCOSP) was devised, which is described fully in this User Guide. This new dataset, which processes CLDPROP data (separately and individually) from 3 instruments: MODIS Aqua, VIIRS SNPP, and VIIRS NOAA-20. This data includes cloud mask, cloud top, and cloud optical retrieval data over daily and monthly timeframes, provides a set of custom cloud-related parameters, using specific dataset definitions, for better comparison with climate model output. This new dataset is provided in the Network Common Data Format Version 4 (NetCDF4) -- a data format widely used by the climate modeling community.

1.0.2. Level-3 (L3) Atmosphere Data Production & Archive

Level-3 (L3) CLDPROPCOSP Atmosphere data users should note that the inputs to COSP are the Continuity CLDPROP_L2 products. These cloud products have been in production since 2017.

Starting by mid-year 2024, this new L3 CLDPROPCOSP product (for all 3 of the aforementioned instruments, for the entire available data record) will be archived and made available from the LAADS DAAC distribution facility:

https://ladsweb.modaps.eosdis.nasa.gov

1.0.3. Definition of "Level"

A quick definition of the "Level" term for those not familiar. The "Level" terminology is used to denote broad categories of satellite data products. Level-0 (L0) denotes raw spectral channel counts, Level-1B (L1B) denotes calibrated and geolocated radiances, Level-2 (L2) denotes orbital-swath science products, typically organized into files with only several minutes of data, and finally Level-3 (L3) denotes global-gridded science products, most commonly compiled on longer time frames like daily or monthly.

1.0.4. Scope of this Document

This document describes the Atmosphere Level-3 (L3) CLDPROPCOSP product. Topics covered include computational approaches and scenarios, subsampling and gridding characteristics, caveats to be aware of, and finally a breakdown of the parameters and statistics computed in this new L3 product.

1.1. L3 CLDPROPCOSP File Characteristics

There are currently three L3 CLDPROPCOSP Cloud products (See Table 1) derived from three L2 CLDPROP data streams from MODIS Aqua, VIIRS SNPP, and VIIRS NOAA20 -- at two temporal timeframes (daily & monthly). L3 Daily CLDPROPCOSP products are tagged D3, which is short for L3 Daily. L3 Monthly CLDPROPCOSP products are tagged M3, which is short for L3 Monthly.

The acronym CLDPROPCOSP (used at the beginning of the L3 COSP filename) is a combination of the input to the product (L2) CLDPROP; and the output format COSP.

1.1.1. *File Format*

The data contained within these files are organized into (broad) Groups (e.g., Cloud_Top_Pressure) -- and statistics are stored as Variables under that Group (e.g., Mean, Standard_Deviation, Pixel_Counts, etc.).

1.1.2. *Resolution*

The statistics in L3 CLDPROPCOSP files are at 1×1° resolution on a rectangular lat-lon global map. This means that all L3 statistics have an array size of 360 columns by 180 rows (360x180), with each L3 grid cell representing one degree of latitude and longitude.

1.1.3. *Filename Convention*

Level-3 (L3) CLDPROPCOSP Atmosphere Product NetCDF4 files are named using a standardized convention as displayed below in Table 2. Note that for the sample NetCDF4 file-name shown:

- The first part of the new standardized Product Prefix is the Product Type (CLDPROPCOSP).
- The second part is the Level (D3 for Daily or M3 for Monthly).
- The third part is the Instrument (MODIS or VIIRS)
- The fourth part is the Satellite Platform (Aqua, SNPP, NOAA20)
- The DDD in the date denotes the Day of Year (001-366).
- All times are UTC time, not local time.

1.1.4. *Versioning: The relationship between Filename (File) Version and Data Version*

The latest CLDPROPCOSP is v1.1, which is represented in the filename as 011. A specific version of data generally stays "current" (is distributed) for anywhere from 1 to as many as 6 years. When reprocessing with an updated CLDPROPCOSP algorithm version is undertaken, or when a new version of inputs becomes available, the previous version of data often continues to be distributed while a new version is being processed and distributed. Hence, in some instances in the future, users may encounter two versions of CLDPROPCOSP data in the public archive. Users should always attempt to acquire and use the latest version.

Finally, while most users may think of versions typically incrementing by one (001 to 002) when reprocessing is undertaken -- versions can sometimes increment by *point one* (denoting a more minor update) which leads to unusual increments in the filename version: 001 to 011 (due to the unusual 3-digit filename version convention of the LAADS DAAC, which distributes COSP data). The related (parallel) data version, used by the Science Algorithm Teams, will always follow a more logical (easier to interpret) pattern: v1.0 to v1.1 to v2.0.

During the processing of a particular version, an attempt is made to use the same science algorithm, sometimes known as the Product Generation Executive (PGE). However, occasionally several new PGE's (algorithm updates) are released during a particular data version to fix minor bugs or make improvements to the data. For bug fixes, if the bug is not serious, forward processing will continue with the newly corrected PGE, while old previously processed data (which was produced with an older version algorithm (PGE)) are left in the archive.

Note that issues, anomalies, and problems in the L2 CLDPROP Product for Version 1.1, which is the input to the L3 CLDPROPCOSP Product, are outlined in the Data Issues section of the Atmosphere Imager website as they are discovered:

https://atmosphere-imager.gsfc.nasa.gov/continuity/issues/cldprop_l2

1.1.5. Start Date for the COSP Products Data Record

The start date for CLDPROPCOSP product data record varies based on the input satellite platform used.

For MODIS Aqua, the start year 2002 -- and the data record continues into the present day, or until the instrument data becomes unviable due to instrument aging. For VIIRS SNPP, the start is year 2012 -- and the data record continues into the present day, or until the instrument data becomes unviable due to instrument aging. For VIIRS NOAA20, the start is year

2018 -- and the data record continues into the present day, or until the instrument data becomes unviable due to instrument aging.

1.2. L3 CLDPROPCOSP File Metrics

1.2.1. Group (Parameter) and Variable (Statistic) Metrics

The current version (v1.1) of the L3 CLDPROPCOSP Global Product (both Daily and Monthly) contains 114 Variables (Statistics) that are computed for 23 Scientific Groups (Parameters) derived from their associated Level-2 input products.

All L3 statistics are sorted into 1×1° cells on an equal-angle global grid (see section 2.0). A number of statistical summaries are computed for each of these L3 parameters, depending on the parameter being considered.

Statistics for a given L2 parameter or measurement might include:

- Simple (Mean, Standard_Deviation, Pixel_Count, Sum, and Sum_Squares) statistics
- Parameters of log-normal distributions
- Fraction of pixels that satisfy some condition (e.g., liquid water clouds, ice clouds)
- Joint Histograms derived from comparing one science parameter to another, statistics may be computed for a subset that satisfies some condition.

1.2.2. File Sizes for NetCDF4 Daily and Monthly files

For the current version denoted as v1.1 (011) of L3 CLDPROPCOSP Product Files, the Daily (D3) NetCDF4 file is around 32 MB in size, the Monthly (M3) NetCDF4 file is around 44 MB in size. These files used standard packing tools to reduce the file sizes as much as possible to aid users in downloading and storing data.

1.3. L3 COSP Definition of "Day": Time span of the Daily Product

The issue of how to define a *single day of data* in a Daily file is more complex than one might initially think; but before we delve into this topic, let's first review how L2 granules are time stamped.

1.3.1. Time Stamping of L2 Files

The L2 input data granules are always time-stamped using Coordinated Universal Time. Coordinated Universal Time is International Atomic Time (TAI) with leap seconds added at irregular intervals to compensate for the Earth's slowing rotation. These added leap seconds allow Coordinated Universal Time (UTC) to track the mean solar time at the Royal Observatory, Greenwich (GMT).

Coordinated Universal Time is abbreviated UTC. The reason for this seemingly unusual abbreviation is the International Telecommunication Union wanted Coordinated Universal Time to have a single abbreviation in all languages. English speakers and French speakers each wanted the initials of their respective language terms to be used internationally: "CUT" for *coordinated universal time* in English, and "TUC" for *temps universel coordonné* in French. This resulted in the final compromise (which likely pleased no one) of using "UTC."

Each L2 granule contains 5 minutes of data for MODIS -- and 6 minutes of data for VIIRS. The time stamp (which is part of the L2 file name) shows the start minute of the data collection time period. For example, a L2 MODIS granule might have a time stamp of 1430. This means the start time of data acquisition was 14:30:00 UTC and the end time was 14:34:59 UTC. The subsequent HDF4 granule would have a time stamp of 1435, which corresponds to an acquisition start time of 14:35:00 UTC.

1.3.2. Definition of "Day" for L3 Daily COSP Products (CLDPROPCOSP_D3)

For CLDPROPCOSP the method to define a "Day" is using a simple 0000 to 2400 UTC period. This can lead to some data gaps near the international date line or data observed (at mid latitudes) nearly 24 hours apart being mixed together in the same geographic region.

Since the monthly CLDPROPCOSP data was deemed the most useful to do climate studies, how we define a single-data-day became less of a factor, since the determination of that single-data-day will average out (or smear out) over a month. A final benefit is the simpler definition of a single-data-day, which is much easier to describe and document.

2.0. Gridding

All COSP Atmosphere L3 statistics are stored on an equal-angle latitude-longitude grid. The grid cells on this projection are 1×1° in size, which means the CLDPROPCOSP Atmosphere L3 output grid is always 360 pixels in width and 180 pixels in height. The left-most boundary of the grid (or map) is located at 180° W longitude, the exact center is the Greenwich Meridian 0°, and the right-most boundary of the grid (or map) is located at 180° E longitude. (See Figure 1).

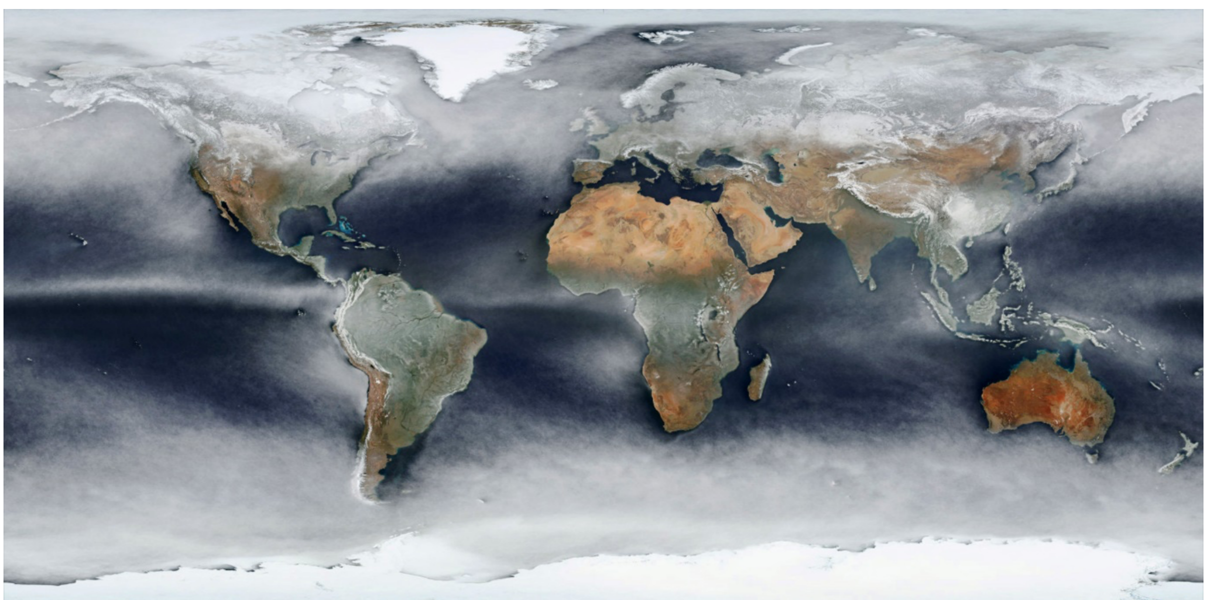


Figure 1. A sample 1x1 degree latitude-longitude rectangular grid. Displayed here is the result of collecting daily RGB Corrected Reflectance satellite data and calculating the per-pixel median over the full year of 2018.

L3 grid cells are indexed (0,0) (since indexing starts at 0 in NetCDF4 files) at the lower left corner of the map and corresponds to a grid box with boundaries of 89° to 90°S latitude and

179° to 180°W longitude. L3 indexing increases as you move up and to the right on the map.

An example of a L3 COSP parameter mapped on this standard 1×1° latitude-longitude grid is shown in Figure 2. Also shown in Figure 2 is a corresponding (matching) Corrected Reflectance image to compare science algorithm results with real-world conditions (views).

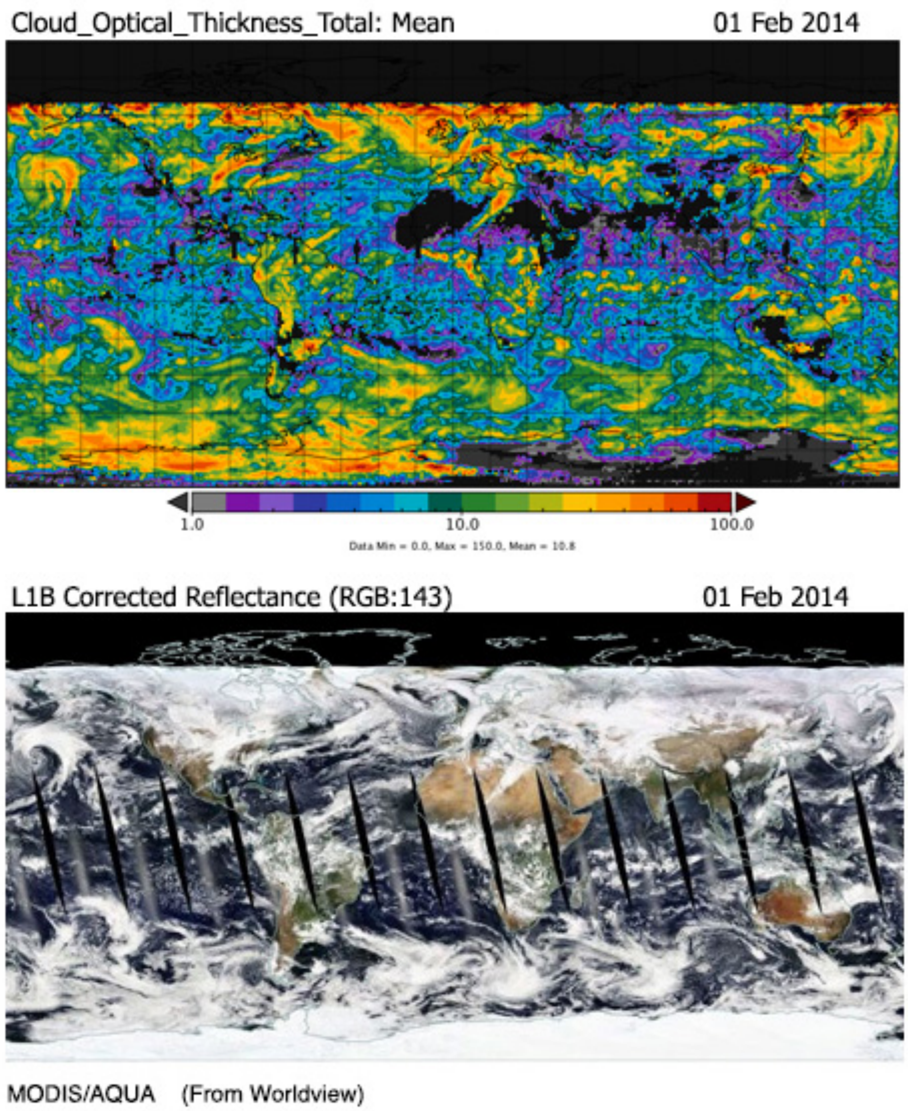


Figure 2. Top is a sample COSP Daily (D3) Cloud Optical Thickness (all clouds) image for combined Terra and Aqua data shown in its native latitude-longitude projection format. Bottom is a corresponding L3 global image of Corrected L1B Radiances (RGB:143), shown for the Aqua platform only, for comparison.

The details of how L2 pixels get binned at L3 is important to understand, especially exactly at the boundaries of 1x1° grid cells. The assignment of L2 pixels (to a L3 grid cell) that fall exactly on a 1° L3 grid boundary is performed using the following convention: L2 pixels that fall exactly on the first whole degree boundary 90°S (-90.0) latitude and 180°W (-180.0) longitude are binned in the first L3 grid column and row (0,0). L2 pixels that fall exactly on the second whole degree boundary 89°S (-89.0) and 179°W (-179.0) are binned in second L3 grid column and row (1,1). The exception to this logic occurs in the last L3 grid row (89° to 90°N), which contains both whole degree latitude boundary pixels (that fall on exactly 89.0 as well as 90.0). There is no exception for the last L3 grid column or longitude (179°E to 180°E) since +180.0 and -180.0 represent the same physical location (these L2 pixels are binned in the first L3 grid column).

An important property to note when considering L3 gridding occurs due to distortion in the latitude-longitude (rectangular) map projection as one moves poleward. The actual (real-world) size of each 1° × 1° grid box physically shrinks when moving from the equator toward the pole due to the convergence of longitude lines on the globe (see Figure 3). At the equator each 1×1° grid cell is roughly 12,321 km^2 in size. At the pole each 1×1° grid cell is only 107 km^2 or less than 1/100th the size.

Figure 3. Convergence of longitude lines produces shrinking 1×1° rectangular grid cells toward the poles.

Please note that there is also a variation of pixel size in L2 (input) products due to viewing (scan angle) distortion. For example, in 1 km (nadir) resolution L2 data, the L2 pixels expand due to view-angle distortion when moving from nadir towards the limb (high scan angles) of an instrument scan (see Figures 4 and 5).

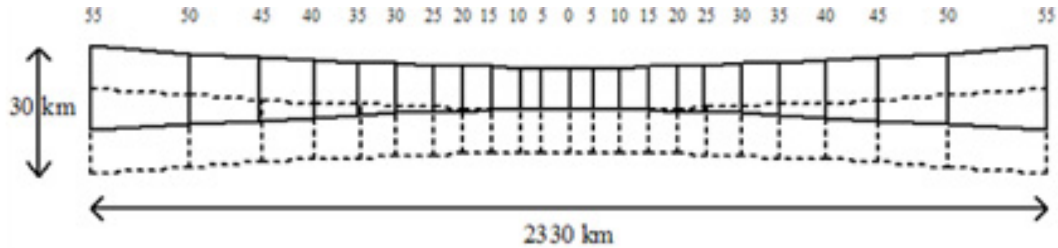
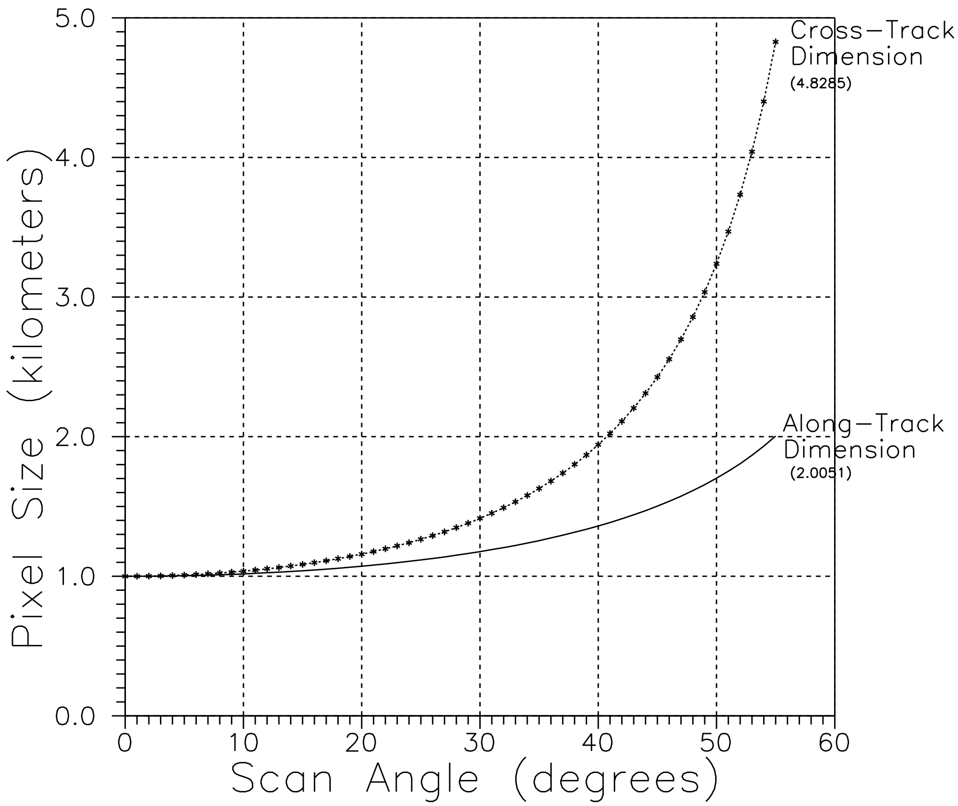


Figure 4. L2 pixel overlap due to scan width growth as a function of scan angle for the MODIS Instrument



....*..*.. cross-track dimension

_______ along-track dimension

Figure 5. Estimated growth of a 1-km resolution L2 pixel as a function of scan angle for the MODIS Instrument.

This distortion of L2 input pixels means that there are fewer L2 pixels to average in the computation of L3 daily statistics for L3 1×1° grid boxes on a Daily (D3) product map that cover regions of high scan angles in the L2 orbital products, than for those that cover L2 orbital products computed closer to nadir view. For the Monthly (M3) products, this effect is averaged out and is mitigated.

Also note that this "growth" of the L2 pixel also causes some pixel (scene) overlap at high scan angles – meaning that the L2 pixels used to compute L3 daily statistics in these "orbit edge" regions are not as statistically independent as those computed with near-nadir-view pixels. So not only are there fewer L2 pixels to average at high MODIS scan angles, but they are less independent.

A third impact, these high scan-angle L2 pixels may be less reliable in general as they are viewed by the sensor through more atmosphere, which complicates the retrieval process.

A fourth impact is that they could have more inherent geolocation error depending on the local terrain.

L3 gridding issues at the poles (due to very small (area-wise) L3 grid boxes) are exacerbated by orbital tracks that typically go no further north or south than roughly 82° – which means the pixels at the poles (90° latitude) are always viewed at higher scan angles. Also, there are a number of overlapping orbits near 82° latitude that cause "time averaging" to occur for daily statistics, for each instrument, computed poleward of about 77° (that is, in polar regions, statistics tend to be daily average statistics). In contrast to those at mid-latitudes (that typically can be pinned down to within 20 minutes of an instrument overpass).

So for the first and last few latitude bins (88° to 90° latitude), one has the dual drawback of (i) very small (area-wise) L3 grid cells containing small populations of relatively large input L2 pixels that overlap spatially (due to view-angle distortion) in a single orbital pass; and (ii) multiple orbital passes (up to 16 on a given day for MODIS) that take measurements of the region, which causes the L3 Daily data to be more representative of a daily average instead of a single overpass snapshot at the poles.

Users should note an important property of L3 gridding: in order to simplify the L3 operational production software, each L2 pixel is "located" or placed in the L3 grid cell where its center latitude/longitude falls within the L3 grid cell. In cases where a larger (view-angle distorted) L2 pixel actually covers (or intersects) more than one 1×1° L3 grid cell, (a situation that typically occurs at high L2 scan angles located over higher latitudes), the L2 pixel is still only assigned (placed) in the single L3 grid cell where the geolocated center point of the L2 pixel is located. Although this shortcoming is relatively minor in the operational L3 software due to the relatively large 1×1° L3 grid size and the much smaller L2 pixel size (750 m or 1km at nadir view), when the code is used in "research mode" to produce products at higher resolution (0.25° or 0.1° for example), it can produce unusual visual effects in the imagery where unnaturally appearing gaps are introduced between adjacent L2 pixels, as illustrated in Figure 8.

Finally, note that most of these pixel count variation issues are mitigated in the Monthly (M3) product since the orbital gap positions shift from day to day, which allows data from subsequent days to fill in the previous day's gaps. This will tend to provide complete global coverage over these longer time periods.

3.0. Sampling

3.1 The Decision to Sample L2 data for L3 Computation

Level-2 (L2) CLDPROP Cloud Products always have geolocation and data arrays stored at 1km resolution, Since the L3 grid size (1°) is so much larger than the L2 resolution (1km), sampling the L2 input data to compute L3 statistics is always performed. The main reason for this is to make the L3 computation faster. In addition, we found in a number of studies (early on) for MODIS that sampling 1km L2 data at 5km had little effect on computed L3 statistics, at the L3 operational grid resolution of 1x1°.

4.0. Computation of L3 Daily Statistics

4.1. Aggregation of Statistics

All Atmosphere L3 products (Daily (D3) and Monthly (M3)) use aggregation capabilities. Aggregation information is typically based upon L2 QA bit flag arrays.

For some parameters, it is useful to aggregate results based on a physical characteristic of the parameter or of the scene. Aggregation refers to the ability to separate L2 input pixel information into various scientifically relevant categories such as liquid water clouds only, ice clouds only, daytime only, etc. These aggregations utilize L2 runtime QA flags that are designed to convey information on retrieval processing path, input data source, and scene characteristics of the physical parameters retrieved.

In addition, this broad group of flags also includes Cloud Mask flags (derived at 1×1 km resolution) to determine cloudy/clear or daytime-only scene characteristics.

Aggregated L3 statistics are often identified by a suffix to the Group (Parameter) name such as "_Liquid" or "_Day". However, to simplify the L3 COSP Product, the _Day suffix was dropped since all the Groups in the COSP product reflect daytime-only scenes. Table 4 lists the various aggregations that occur in L3 COSP.

Product Family	Aggregation Property
Cloud Top Properties & Cloud Mask	Daytime Only.
Cloud Optical Properties	Liquid Water, Ice, Total or Combined Phase Clouds. Partly Cloudy (PCL) Scenes vs. Std. Regular Cloudy Scenes.

Table 4. Aggregations that are performed in L3 COSP.

Additional details and documentation about each Group (Parameter) are always provided in the local attribute "long_name" attached to each Group within the NetCDF4 file.

An interesting observation to note is that aggregations of L3 statistics are performed in Yori through the use of true/false (T/F) Masks that are defined in the pre-Yori code and triggered by particular settings in the YAML (Yet Another Markup Language) Configuration File during processing. Multiple Masks may be applied to Statistics in any Group (Parameter).

The Masks which are currently defined in the Yori/YAML code are:

- Mask_Day: Daytime pixels based on the day/night flag
- Mask_CloudMaskDetermined: Cloud mask determined flag
- Mask_Liquid_Water_Phase_Clouds: Liquid water phase cloud retrievals
- Mask_Ice_Phase_Clouds: Ice phase cloud retrievals
- Mask_Undetermined_Phase_Clouds: Undetermined phase cloud retrievals
- Mask_Combined_Phase_Clouds: Combined phase cloud retrievals

4.2. Types of Statistics computed

4.2.1. L3 Statistics Computed by Yori (the L3 Processing System)

Yori is the system/package that is used to process L3 COSP Atmosphere Data. A total of 7 general types of statistics are computed in the COSP Atmosphere L3 products within the Yori Processing System for each 1x1° grid cell. These seven statistics shown below are always found as individual Variables (tagged exactly as shown in the bullets below) under specific Groups within the NetCDF4 file. The Statistics are:

- **Mean** = the mean of the sampled L2 pixels in each L3 grid
- **Standard_Deviation** = the standard deviation of the sampled L2 pixels in each L3 grid
- **Pixel_Counts** = the counts of the sampled L2 pixels in each L3 grid
- **Sum** = the sum of the values of the L2 pixels in each L3 grid
- **Sum_Squares** = the sum of the squares of the values of the L2 pixels in each L3 grid
- **Histogram_Counts** = 1D marginal histogram counts of binned values of the L2 pixels
- **JHisto_vs_xxx** (where xxx is a secondary Parameter name) = 2D joint histogram counts of binned values of one set of L2 pixels vs. another set of L2 pixels of a different parameter

Note that in the Monthly COSP file (product) the Mean is always pixel-count weighted, since it's based on Daily values (derived from 06_L2 input) stored as Sum and Pixel_Counts. It is not an unweighted Mean of the Daily Means. Likewise, for Standard_Deviation, it's the standard deviation of the L2 input pixels.

4.2.2. Details of the Math Behind Yori (Section 4.2.2.is from documentation on Yori L3 Gridding Tools)

Since Yori can aggregate files that have already been aggregated (e.g., using 28, 29, 30 or 31 daily files to create a monthly product), it is important to clarify how the aggregation is performed and how the quantities stored in the files are computed. In this section we go through the details of the Yori aggregation phase where statistics are computed, starting from the trivial quantities and then working our way up from there.

4.2.2.1. Computation of Sum, Sum_Squares, and Pixel_Counts

Let's first briefly introduce how sums and sums of squares are computed during the L3 gridding phase. Although these definitions should be fairly obvious, they will be helpful as things progress in the following section.

For any given variable, the **sum** (Sum) represents the sum of its values v_j within a grid cell c and is computed as:

$$s_c = \sum_j v_j \qquad (1)$$

where the subscript j is the j-th pixel and v_j is the value of the quantity under consideration for that pixel j. The summation is performed over all the valid pixels, that is, all those pixels that aren't filtered out via masking in L3.

Similar to eq. (1) we can also compute the **sum of squares** (Sum_Squares), that is, the sum of the squared values of all the valid pixels inside a grid cell:

$$ss_c = \sum_j v_j^2 \qquad (2)$$

where the subscript j denotes the j-th pixel and v_j is the value of the quantity under consideration for that pixel.

The **number of points** (Pixel_Counts) at the L3 gridding stage is simply the count of the number of valid pixels n for any given grid cell c.

During the aggregation phase, the quantities introduced in the previous paragraphs are summed in order to compute what will be stored in the aggregated files. Hence, the aggregated **sum** (Sum) **is** derived as:

$$S_c = \sum_i s_{i,c} \qquad (3)$$

where the subscripts i and c represent the i-th file and the c-th grid cell respectively and S_c is the total sum for the cell c. Likewise, the aggregated **sum of squares** (Sum_Squares) SS_c is computed by just replacing $s_{i,c}$ with $ss_{i,c}$ in eq. (3):

$$SS_c = \sum_i ss_{i,c} \qquad (4)$$

Finally, the **number of points** (Pixel_Counts) for any given grid cell c is:

$$N_c = \sum_i n_{i,c} \qquad (5)$$

where, once again, the subscripts i and c indicate the i-th file and the c-th grid cell respectively, so that n is the number of points of the cell c for the file i and N_c is the total number of points for the same cell c.

4.2.2.2. *Computation of Mean and Standard_Deviation*

In this section we will explain how the mean (Mean) and standard deviation (Standard_Deviation) are computed during the aggregation and computation process. In the following, the subscript c will be omitted for sake of simplicity but, unless otherwise stated, all quantities refer to a single grid cell.

The **mean** (Mean) M of a quantity is computed as:

$$M = \frac{\sum_i s_i}{\sum_i n_i} = \frac{S}{N} \qquad (6)$$

where the subscript i indicates the i-th file, while S and N are the sum and number of points respectively, introduced earlier in eqs. (1) and (5).

The **standard deviation** (Standard_Deviation) is derived from:

$$\sqrt{\frac{\sum_i ss_i}{\sum_i n_i} - \left(\frac{\sum_i s_i}{\sum_i n_i}\right)^2} \qquad (7)$$

where, again, i denotes the i-file and s_i, ss_i and n_i are sum, sum of squares and number of points described previously in eqs. (1), (2), and (5), respectively.

By saving sum, sum of squares, and number of points at every stage, and using eqs. (6) and (7), it is possible to correctly compute mean and standard deviation of all the elements within a grid cell without the need to also store all those values (in case further aggregation is required at a later stage).

For instance, this allows deriving monthly aggregations from daily data, which wouldn't be possible if the mean and standard deviation were computed directly from the values provided in the input files.

4.2.3. *L3 Statistics computed by manual set-up in the Pre-Yori Step*

Some desired L3 statistics were not directly computable by Yori (such as Fractions or Log Statistics). Fortunately, a part of the L3 Yori Processing System is a step called pre-Yori, where

L2 files are "massaged" into a format that can be read by the packaged (canned) routines of Yori (which are mostly fixed and unchanging). Because this pre-Yori step exists, Science Algorithm Developers have the opportunity to create "set up" arrays, which can be used by the relatively basic Yori system to create more interesting and useful statistics. The L3 COSP product uses a pre-Yori step to preprocess or set up new L2 intermediate arrays, which Yori later uses to produce both Fraction (Cloud Fraction) and Log statistics.

4.2.3.1. Cloud Fraction from Cloud Mask

A number of different "flavors" of cloud fractions with different meanings and properties are included in the L3 COSP Product.

The L3 cloud fraction that appears to garner the most interest from L3 Atmosphere data users is cloud fraction derived directly from the L2 cloud mask flags (in the 35_L2 product). Note that these cloud mask flags are duplicated in the L2 Cloud Properties (06_L2) product.

In the L3 COSP product, the following aggregations of Cloud Fraction (from Cloud Mask) are available in the following L3 Groups (or Parameters):

- Cloud_Mask_Fraction (daytime only, all pressure levels (total column))
- Cloud_Mask_Fraction_Low (daytime only, CTP $\geq$ 680 hPa)
- Cloud_Mask_Fraction_Mid (daytime only, 680 hPa $>$ CTP $\geq$ 440 hPa)
- Cloud_Mask_Fraction_High (daytime only, CTP $<$ 440 hPa)

The first parameter listed above contains the cloud fraction at all pressure levels computed from daytime retrievals (Solar Zenith Angle (SZA) $\leq$ 85°). The second through fourth parameters listed above have additional aggregations of atmospheric pressure levels: low-only, mid-level-only, and high-level-only clouds.

Users should note that the daytime cutoff for Cloud Mask (SZA $\leq$ 85°) is less strict than for Cloud Top or Cloud Optical Properties (SZA $\leq$ 80°), so the data extends a bit farther pole-ward for Cloud Mask.

The Yori processing system was not designed to compute fraction statistics directly; an intermediate L2 array called *Cloud_Mask_Cloudiness* is created as a pre-Yori L3 processing step to enable Yori to compute Cloud Fraction from Cloud Mask.

This intermediate array is basically an array filled with 1s (True) and 0s (False). The 1s represent L2 pixels that are true (or cloudy) for the computation of the particular cloud fraction. The 0s represent L2 pixels that are false (because they were deemed clear or did not meet some other specified criteria).

To compute these 1s and 0s in the intermediate array, two L2 QA flags are read from the Cloud_Mask QA array in L2 COSP. The two L2 QA flags are: the *Cloud Mask Status Flag* and the *Cloud Mask Cloudiness Flag* (see Table 5).

L2 QA Flag	Flag Value	Meaning
Cloud Mask Status Flag	0 1	Undetermined Determined
Cloud Mask Cloudiness Flag	0 1 2 3	Confident Cloudy (or Fill if Status Flag = 0) Probably Cloudy Probably Clear Confident Clear

Table 5. The two key L2 Cloud Mask Flags used to compute the intermediate pre-Yori array *Cloud_Mask_Cloudiness*; and finally in the final L3 processing step, the Cloud_Fraction parameter in the L3 COSP product.

To properly set the 1s and 0s in the intermediate *Cloud_Mask_Cloudiness* array, the following logic is used. If the Cloud Mask Status Flag is set to 1 (determined), then L2 pixels with Cloud Mask Cloudiness Flags set to either 0 (confident cloudy) or 1 (probably cloudy), are assigned a 1 or True. If the Cloud Mask Status Flag is set to 1 (determined), then L2 pixels with Cloud Mask Cloudiness Flags set to either 2 (probably clear) or 3 (confident clear), are assigned a 0 or False.

Finally, the L3 cloud fraction from cloud mask (Cloud_Fraction) is calculated in Yori and written to the L3 COSP product by taking the mean value of the 1s and 0s in the intermediate *Cloud_Mask_Cloudiness* array.

For example, consider the 1x1 degree L3 grid box shown in Figure 10. Then the L3 Yori code does the following:

1. Toss out any Fill Value (-9999) encountered, and
2. with the remaining 1s and 0s (in each 1x1 degree L3 grid box) simply take the Mean of all the values -- that Mean is deemed the cloud fraction for that L3 grid box

In this example of a single L3 1x1 degree grid box shown in Figure 10, there are 2 fill (missing) values (-9999) that are tossed out or ignored. This leaves 10 valid L2 pixels or observations in the L3 Grid Box (0s and 1s). There were 3 pixels or observations assigned cloudy or true (1.0) and 7 pixels or observations assigned clear or false (0.0). The Mean of those 0s and 1s is 0.300.

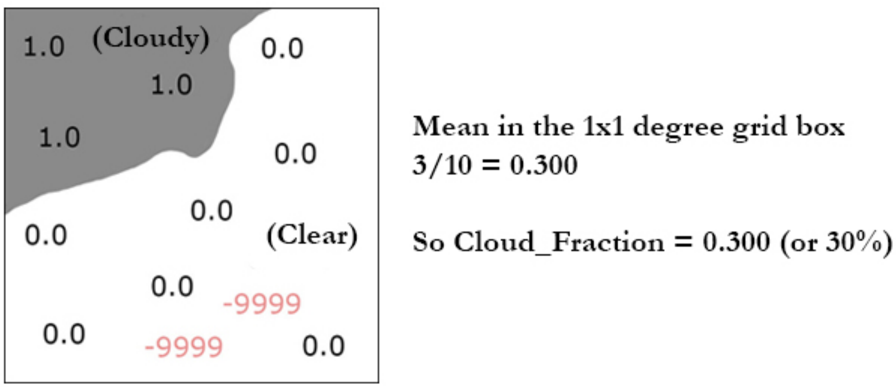


Figure 10. A theoretical example of a 1x1 degree L3 grid cell displaying a hypothetical sample of the intermediate Cloud_Mask_Cloudiness array of 1s (true or cloudy) and 0s (false or clear) that are used to compute cloud fractions in L3 by computing the mean of those numbers.

Cloud Fraction = Sum / Pixel_Counts = Mean = 3 / 10 = 0.300 (or 30% cloudy)

So the output Cloud Fraction, in this theoretical 1x1 degree L3 Grid Box, will be 0.300. To complete the job, Yori simply continues to build up the remaining grid cells in the 360 x 180 degree L3 Map in the same manner.

4.2.3.2. Cloud Fraction based on Successful Cloud Optical Property Retrievals (by cloud phase)

The second-most utilized L3 cloud fraction is that derived from the Cloud Optical Properties retrieval. All optical property cloud fractions are computed for **daytime scenes only** (Solar

Zenith Angle (SZA) ≤ 81.3731°), which is slightly more restrictive than the definition of daytime for Cloud Mask fractions (SZA ≤ 85°).

The L3 COSP product provides the following aggregations of Cloud Optical Property Cloud Fractions Groups (or Parameters):

- Cloud_Retrieval_Fraction_Liquid
- Cloud_Retrieval_Fraction_Ice
- Cloud_Retrieval_Fraction_Total

The first parameter above represents the cloud retrieval fraction for liquid water phase clouds only; the second, ice phase clouds only; and the third, total or combined phase clouds. Note that the total cloud fraction (Cloud_Retrieval_Fraction_Total) includes Undetermined phase clouds, so it's not just a sum of the Liquid phase and Ice phase.

The optical property cloud fraction is computed in L3 from QA Flags read from a Level-2 Cloud Product (06_L2) SDS called *Quality_Assurance*. These two QA Flags are: the *Primary Cloud Retrieval Phase Flag and the Primary Cloud Retrieval Outcome Flag*. See Table 6.

To compute Cloud Retrieval Fractions (from the Cloud Optical Property Algorithm) using Yori, intermediate L2 arrays are created in the pre-Yori step called *COPR_** (where * is an acronym for the cloud phase).

These intermediate *COPR_** arrays are (as before for the *Cloud_Mask_Cloudiness arrays*) filled with 1s (True) and 0s (False). The 1s represent L2 pixels that are true (or cloudy that also meet some preset requirement specified, e.g., liquid water cloud phase) for the computation of the particular cloud fraction. The 0s represent L2 pixels that are either false (because they were deemed clear) or false (because they did not meet the requirements specified for that particular fraction (e.g., ice phase clouds). Therefore, if one is computing Liquid Water Cloud Fraction, 1s (T) can only be set if the L2 pixel is both cloudy and if the clouds were tagged Liquid Water Phase.

The logic used was as follows: every sampled L2 grid point that has a *Primary Cloud Retrieval Outcome Flag* = 1 (Retrieval Successful) and a *Primary Cloud Retrieval Phase Flag* of 2 (Liquid Water

Cloud), 3 (Ice Cloud), or 4 (Undetermined Phase Cloud) are taken as 100% cloudy for the cloud phase category in question.

L2 QA Flag	Flag Value	Meaning
Primary Cloud Retrieval Phase Flag	0 1 2 3 4	Cloud Mask Undetermined (Missing or Fill) Not Processed (Typically Clear) Liquid Water Cloud Ice Cloud Undetermined Phase Cloud
Primary Cloud Retrieval Outcome Flag	0 1	Retrieval Not Attempted or Unsuccessful Retrieval Successful

Table 6. Two key 06_L2 QA Flags used to compute the Cloud Optical Properties Cloud Fraction.

Key point to remember there was a change in how the Cloud Optical Property L3 Cloud Fractions were computed between MODIS Standard Collection 5/51 and Collection 6/61. In MODIS Standard Collection 6 (C6) and later, it was decided to include failed retrievals (in the counts used in the denominator of the L3 cloud fraction computation). This same MODIS Standard C6 logic (to include failed retrievals in this computation as false or 0) is used in L3 COSP.

One note about the *Primary Cloud Retrieval Phase Flag* in Table 6. Flag values of 1 (Not Processed) are counted as clear sky. These retrievals will always have a *Retrieval Outcome Flag* of 0 (Unsuccessful).

In summary, in the L3 COSP files, the Cloud Optical Properties Cloud Fraction uses clear pixels, successfully retrieved pixels, **and unsuccessfully retrieved pixels**, to derive the computational denominator. Only successfully retrieved pixels that meet the set cloud-phase aggregation logic are used in the computational numerator.

Keeping in mind the logic described previously for the Cloud Fraction Computation in L3 COSP, the Mean of those T/F pixels are tallied to compute the fraction. See Figure 10.

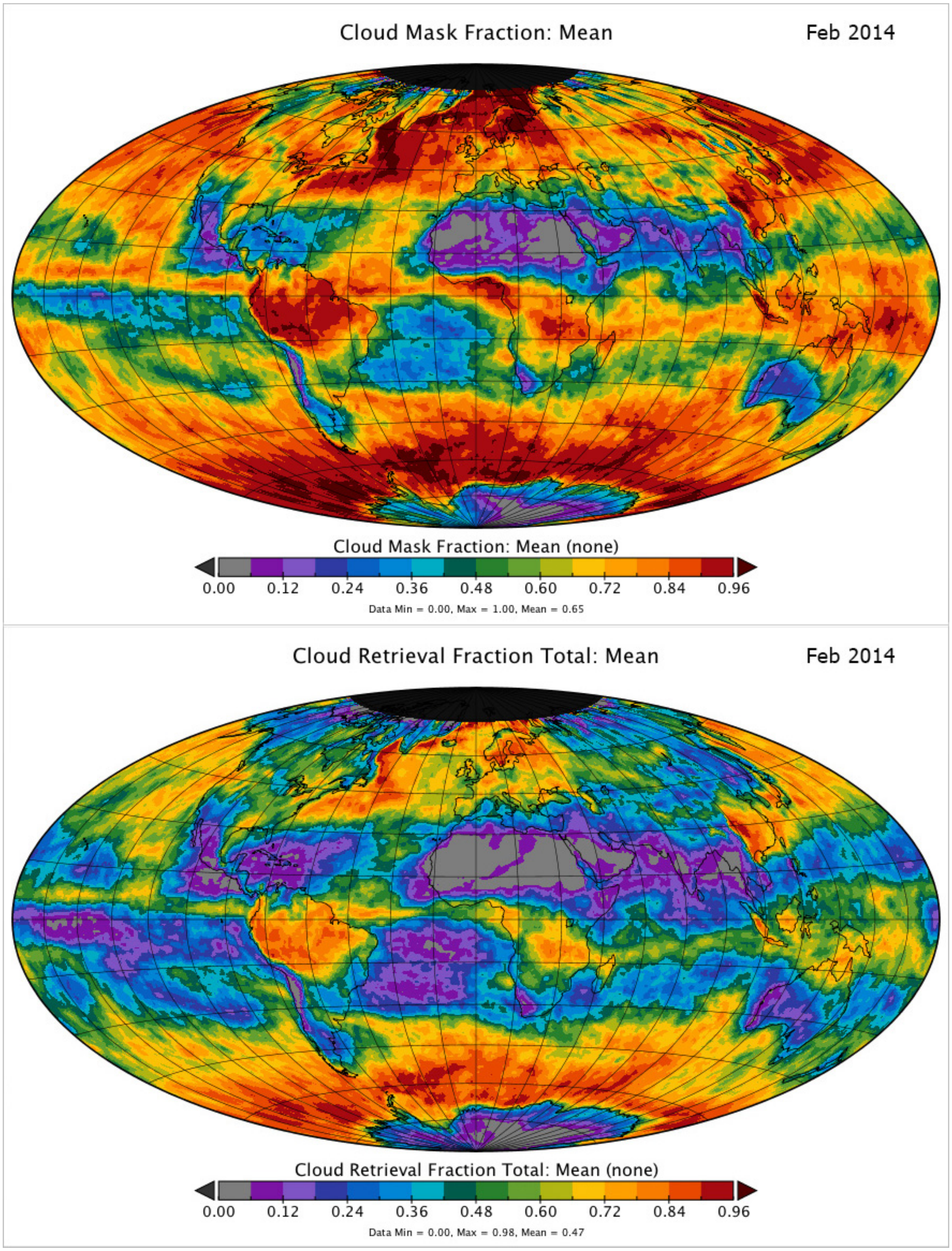


Figure 11. L3 COSP Monthly Images for Feb 2014 (a) Cloud Fraction from Cloud Mask for Daytime only (b) Successful Cloud Optical Property Retrieval Fraction for Combined Phase Clouds (Daytime) The scale on both images span the full range of data: 0.0 (0%) to 1.0 (100%). There are more clouds in the top image due to failed retrievals in the bottom image.

4.2.3.3. Logarithm Statistics.

Logarithms are another type of statistic not directly computable by Yori. Again, the pre-Yori step was utilized to pre-process (or set up) new L2 intermediate arrays, which Yori could later use to produce Log statistics.

This patch does precipitate a very minor array difference in MODIS Standard (MOD08) vs. COSP (CLDPROPCOSP) Products. In the MODIS Standard Products, Log was a 'statistic type'; however, with this patch required in Yori, Log is now an actual new Group (or Parameter) in the L3 COSP product.

Please note that these Log statistics are only computed for cloud optical thickness (τ_c) parameters. The reason for this is as follows: because of the curvature of cloud reflectance as a function of optical thickness, the mean optical thickness of an ensemble of pixels does not correspond to the mean reflectance (or albedo) of those pixels. However, the mean of log(τ_c) approximates the radiatively-averaged optical thickness because reflectance plotted as a function of log(τ_c) is linear over a wide range of optical thickness (excluding small and large values). That is, the mean of log(τ_c) gives an optical thickness that approximately corresponds to the average reflectance of the pixels that comprise the mean. The accuracy of this approximation depends on the nature of the optical thickness probability density function (PDF). Studies on the validity of this approximation for MODIS scenes is reported by Oreopoulos et al. (2007). A similar study on ice clouds by the same authors is ongoing.

Relative to the more complex Cloud Fraction patch noted in the previous section, this pre-Yori Log software patch was much easier to code. The Log set-up arrays were made by simply taking the Log (base 10) of Cloud Optical Thickness and creating a new intermediate input L2 array called *Cloud_Optical_Thickness_Log10*. Next, that new array is simply run through the canned Yori routines to produce the suite of 5 simple Log statistics (Mean, Standard_Deviation, etc.).

Since for the Log computation, L2 cloud optical thicknesses (τ_c) are converted to base 10 logs, the following mapping can be laid out. A τ_c of 100 would be converted to a log value of

2.0, a τ_c of 10 would be converted to a log value of 1.0, a τ_c of 1.0 would be converted to a log value of 0, a τ_c of 0.1 would be converted to a log value of –1.0, and finally a τ_c of 0.01 (the smallest reported value in L2) would be converted to a log value of –2.0. So the valid range of this Log statistic would normally be -2.0 to 2.0 (corresponding to regular Cloud Optical Thickness data values ranging from 0.01 to 100). However, at the start of MODIS Standard C6, the maximum reported cloud optical thickness was increased from 100 to 150 in the cloud retrieval algorithm, which means the maximum log value is now 2.176 instead of 2.0. Once the log values of the L2 input pixel data are calculated in the pre-Yori step, a daily mean value of all the log values is computed.

Figure 12 shows an illustrative comparison of traditional (regular linear scale) Cloud Optical Thickness imagery vs. Cloud Optical Thickness **Log** imagery. These images are for the exact same month of data, February 2014.

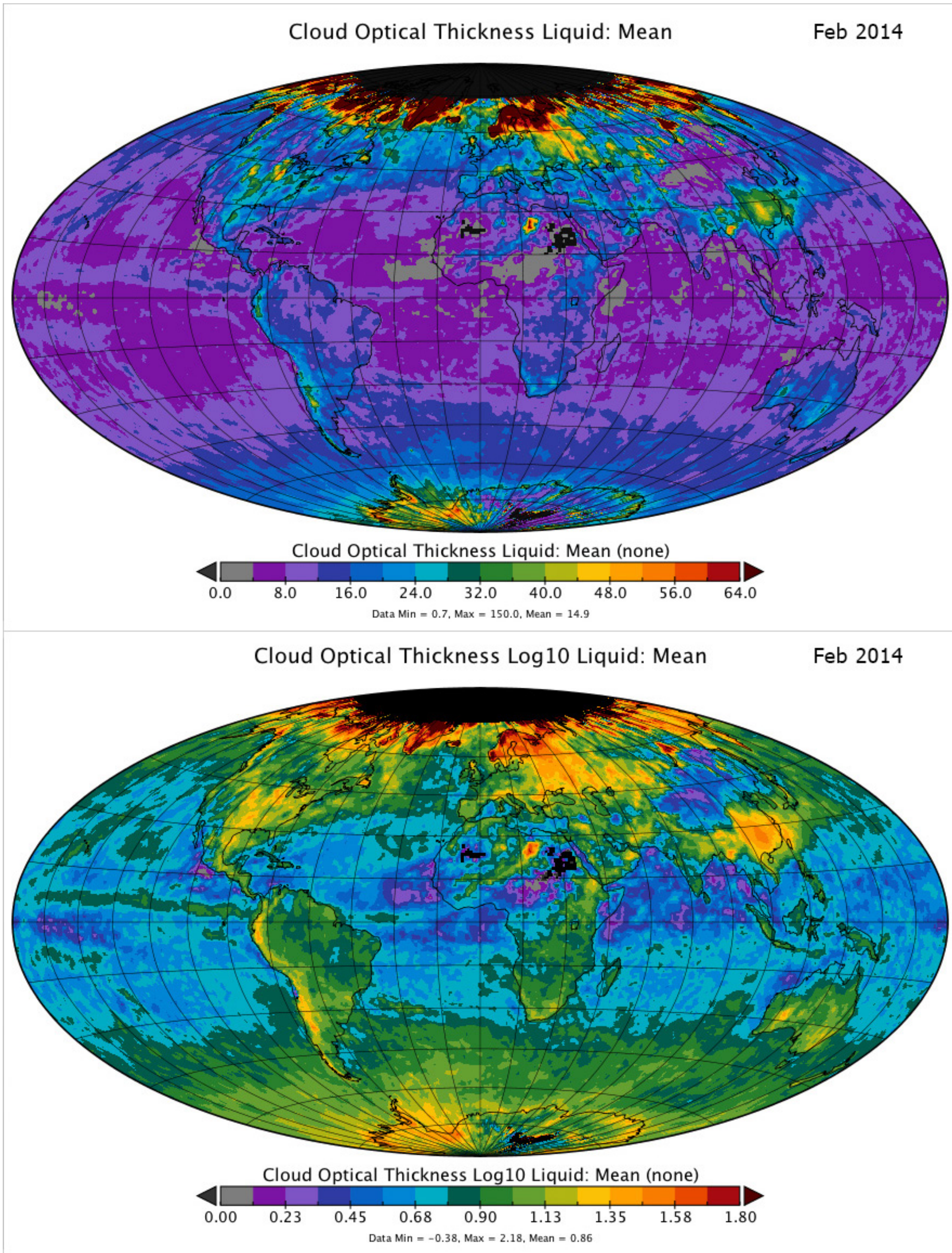


Figure 12. L3 COSP Monthly Images for Feb 2014 (a) Cloud Optical Thickness for Liquid Water Phase Clouds (full valid range from 0.01 to 150.0) (b) The Logarithm (Log) of that exact same data (full valid range from -2.0 to 2.176). Note the image scale ranges were tweaked on both images to provide more detail.

5.0. Computation of L3 Monthly Statistics

Since the exact same pre-Yori code, YAML (a recursive acronym that stands for Yet Another Markup Language) Configuration File, and Yori Processing System is used to produce Monthly (M3) files that were used to produce Daily (D3) files, there is little chance of anything going askew in the Monthly product only. If the Daily is being computed correctly and producing valid results, then the Monthly will follow. If the Daily is not, then likewise, the Monthly will not. That being said, both the Daily and Monthly products rely on correct and valid upstream L2 input products.

Because of this processing algorithm matching paradigm, L3 COSP Daily and Monthly products have identical mapping grids and resolution, identical parameter and statistic inventory lists, and identical histograms and joint histograms (including bin boundary definitions).

It's interesting to note that only the L3 Daily files are used as direct input into the Monthly product, however because L2-tied quantities Sum and Sum_Squares are carried through the Daily and Monthly file, statistics in the Monthly file continue to reflect the L2 input data directly. This means quantities of standard deviation in the Monthly (M3) file are actually the standard deviation of the L2 pixels, which is the most useful and meaningful way to compute standard deviation.

Users should also note that in L3 COSP Monthly file (as well as Daily file), all mean and standard deviation quantities are pixel-count weighted.

Finally, true in both the Daily and Monthly files, there is no "valid_range" check, only "Fill_Value" L2 grid cells are universally excluded. Valid_range is reported in the YAML configuration file and in the NetCDF4 file as local attributes attached to each Group, however it's for documentation purposes only.

6.0. Reading and Unpacking NetCDF4 File Data

NetCDF4 (Network Common Data Form, v4) is a set of software libraries and self-describing, machine-independent, multi-object data formats that support the creation, access, and sharing of array-oriented scientific data. NetCDF was developed by and is maintained at Unidata, which is part of the University Corporation for Atmospheric Research (UCAR) Community Programs (UCP). NetCDF4 is the standard data format for Atmosphere COSP data products.

6.1. Descaling data

The standard NetCDF4 file local attributes "scale_factor" and "add_offset," attached to every Group, would typically only apply to the Mean and Standard_Deviation statistics (which are the only two statistics in L3 COSP that could be realistically packed) and in that case, would be used to convert stored (packed) integer data to geophysical floating-point numbers through the use of the following equation:

Geophysical Value = scale_factor * (Stored Integer - add_offset) (8)

However, in L3 COSP files, the Mean and Standard_Deviation statistics are stored as double-precision real (R*8) numbers that are not natively packed, therefore no end-user unpacking needs to occur. So, for all Groups (Parameters) in L3 COSP, the scale_factor local attribute is set to 1.0 and the add_offset local attribute is set to 0.0.

All L3 count-related statistics (Pixel_Counts, Histogram_Counts, and JHisto_vs_xxx) are stored as 4-byte Integer (I*4) or INT. INT numbers can range up to 2,147,483,647, which is enough headroom to store any possible real-world count from sampled MODIS instrument data at larger L3 grid sizes, even at longer time intervals. The Pixel_Counts will always contain the largest numbers of these three count-related statistics, since those counts are not subdivided into data-value bins.

The Sum and Sum_Squares statistics are stored as double-precision Real (R*8), the same format as the Mean and Standard_Deviation. The Sum_Squares numbers can get astronomically large. The Sum and Sum_Squares data is really only provided (available) in order to compute other statistics (e.g.: Mean and Standard_Deviation) as Yori aggregate data, and rarely need to be read by the science or user community.

The scientific measuring units of the geophysical floating-point Mean and Standard Deviation values are indicated by the "units" local attribute that is also provided with each Group.

Valid range local attributes (given as a min (value_min) and max (value_max) number) is also provided with each Group. The two valid range values provided are the expected low and high values of valid (non-fill) L2 input to the L3 statistic data.

Note that **no** valid range screening on the input L2 data or the output L3 data is performed. The reason for this is sometimes absolute valid ranges are difficult to determine in advance and the algorithm developers want to avoid the potential loss of good data. Therefore, users should not be surprised to find non-fill data points that fall outside the documented valid range; however, it should raise a flag for the user to make sure they are unpacking the data correctly.

6.2. Definitions of local attributes

As partially described above, attached to each Group within a NetCDF4 file are a suite of local attributes (some described in the previous section). These local attributes serve as a key to interpret the data, drive the logic in the Yori system, and provide documentation for end users.

6.2.1. Local attributes used in L3 COSP files.

Given below is the full set of local attributes, which can be attached to specific Groups (Parameters) or Variables (Statistics) in the L3 COSP files. Note that not all of these are attached to every Group and Variable (only the local attributes necessary to compute or properly document the statistics in the Group).

- **long_name**. A longer character string description of the Group.
- **units**. Scientific measurement units of the data (if there are no units, "none" is

specified).

- **valid_min, valid_max.** An array of two numbers that describe the expected low and high values (valid range) of the L2 data (before applying any scale and offset in L2, if L2 data was packed); ignoring the L2 Fill Value.
- **scale_factor.** If the data is packed, this is the scaling factor used to unpack the data. Note for L3 COSP files, this is always set to 1.0 (which means no packing was done).
- **add_offset.** If the data is packed, this is the offset used to unpack the data. Note for L3 COSP files, this is always set to 0.0 (which means no packing was done).
- **_FillValue.** The value of missing or fill data in the Group (Parameter) or Variable (Statistics). Typically set to -9999 in the L3 COSP file.
- **title.** A character array providing a succinct description of what is in the variable. Note that this attribute is attached to all variables in the L3 COSP NetCDF4 file and provides information to various visualization packages (e.g., Panoply) on how to explicitly title images produced from that variable (typically this is a combination of the Group name and the Variable name).
- **Histogram_Bin_Boundaries.** An array of numbers describing the histogram bin boundaries of the primary parameter. (Only used when an optional histogram (Histogram_Count) statistic is defined.)
- **JHisto_Bin_Boundaries.** An array of numbers describing the joint histogram bin boundaries of the primary parameter. (Only used when an optional joint histogram (JHisto_vs_xxx) statistic is defined.)
- **JHisto_Bin_Boundaries_Joint_Parameter.** An array of numbers describing the histogram bin boundaries of any secondary (or jointed) parameter. (Only used when an optional joint histogram (JHisto_vs_xxx) statistic is defined.)

7.0. Complete Group & Variable Inventory of L3 COSP Files

Table 7 (shown on the following page) shows a complete inventory of Groups and Variables in the L3 COSP products. The Groups (or Parameters) are shown in the rows of the table. The Variables (or Statistics) are shown in the columns of the table. Statistics that are computed for each Group are represented by colored dots in the table. A dot means that statistic exists (shown in the column) for that particular parameter (shown in the row).

The color coding of the statistic colums was done to help differentiate groups of statistics. This table was compiled based on the latest version and should remain relatively stable for future versions. This version matches that of the source data.

The number of Groups (Parameters) total 32, the number of possible Variables (Statistics) that can be defined for each Group number 11. The total number of Group/Variable (or Parameter/Statistic) combinations total 174. Of these 174 Parameter/Statistic combinations, 160 are scalar statistics and 14 are 2D Joint Histograms.

Variable	Mean	Standard_Deviation	Sum	Sum_Squares	Pixel_Counts	Histogram_Counts (n)	JHisto_vs_Cloud_Particle_Size_Liquid (nxm)	JHisto_vs_Cloud_Particle_Size_PCL_Liquid (nxm)	JHisto_vs_Cloud_Particle_Size_Ice (nxm)	JHisto_vs_Cloud_Particle_Size_PCL_Ice (nxm)	JHisto_vs_Cloud_Top_Pressure (nxm)
Angles *4 (20)*											
01. Solar_Zenith [Day Mask: SZA ≤ 85°]	•	•	•	•	•						
02. Solar_Azimuth [Day Mask: SZA ≤ 85°]	•	•	•	•	•						
03. Sensor_Zenith [Day Mask: SZA ≤ 85°]	•	•	•	•	•						
04. Sensor_Azimuth [Day Mask: SZA ≤ 85°]	•	•	•	•	•						
Cloud Top Properties (CTP) *5km input 5 (25)*											
01. Cloud_Top_Pressure [Day Mask: SZA ≤ 85°]	•	•	•	•	•						
02. Cloud_Mask_Fraction [Derived from CM, includes CTP fail pixels]	•	•	•	•	•						
03. Cloud_Mask_Fraction_Low [CTP≥680 hPa] (2022 Production Fix)	•	•	•	•	•						
04. Cloud_Mask_Fraction_Mid [680 hPa>CTP≥440 hPa] (2022 Fix)	•	•	•	•	•						
05. Cloud_Mask_Fraction_High [CTP<440 hPa] (2022 Fix)	•	•	•	•	•						
Cloud Optical Properties (COP) *13 (68) JH=3*											
3.7µm Retrieval (Regular Cloudy Retr.)											
Fully Cloudy Scenes [COP Limit 06_L2 Algo: SZA ≤ 81.3731°]											
01. Cloud_Optical_Thickness_Liquid	•	•	•	•	•		•				•
02. Cloud_Optical_Thickness_Ice [PC = Success CLDY Retr.]	•	•	•	•	•				•		•
03. Cloud_Optical_Thickness_Total [Includes Undet.]	•	•	•	•	•						•
04. Cloud_Optical_Thickness_Log10_Liquid	•	•	•	•	•						
05. Cloud_Optical_Thickness_Log10_Ice	•	•	•	•	•						
06. Cloud_Optical_Thickness_Log10_Total [Includes Undet.]	•	•	•	•	•						
07. Cloud_Particle_Size_Liquid [PC = COTL PC]	•	•	•	•	•						
08. Cloud_Particle_Size_Ice	•	•	•	•	•						
09. Cloud_Water_Path_Liquid [PC = COTL PC]	•	•	•	•	•		•				
10. Cloud_Water_Path_Ice	•	•	•	•	•				•		
11. Cloud_Retrieval_Fraction_Liquid [COPR derived]	•	•	•	•	•						
12. Cloud_Retrieval_Fraction_Ice [Pixel_Counts include Clears]	•	•	•	•	•						
13. Cloud_Retrieval_Fraction_Total [Includes Undetermined Phase]	•	•	•	•	•						

3.7µm PCL Retrieval (Partly Cloudy Retr.) Partly Cloudy Scenes	Mean	Standard_Deviation	Sum	Sum_Squares	Pixel_Counts	Histogram_Counts (n)	JHisto_vs_Cloud_Particle_Size_Liquid (nxm)	JHisto_vs_Cloud_Particle_Size_PCL_Liquid (nxm)	JHisto_vs_Cloud_Particle_Size_Ice (nxm)	JHisto_vs_Cloud_Particle_Size_PCL_Ice (nxm)	JHisto_vs_Cloud_Top_Pressure (nxm)
01. Cloud_Optical_Thickness_PCL_Liquid	•	•	•	•	•			•			•
02. Cloud_Optical_Thickness_PCL_Ice [PC = PCL Success]	•	•	•	•	•					•	•
03. Cloud_Optical_Thickness_PCL_Total [Incl. Undetermined Phase]	•	•	•	•	•						•
04. Cloud_Particle_Size_PCL_Liquid [PC = COTPCLL PC]	•	•	•	•	•						
05. Cloud_Particle_Size_PCL_Ice	•	•	•	•	•						
06. Cloud_Water_Path_PCL_Liquid [PC = COTPCLL PC]	•	•	•	•	•			•			
07. Cloud_Water_Path_PCL_Ice	•	•	•	•	•					•	
08. Cloud_Retrieval_Fraction_PCL_Liquid [COPR derived]	•	•	•	•	•						
08. Cloud_Retrieval_Fraction_PCL_Ice [PC incl. Clears]	•	•	•	•	•						
10. Cloud_Retrieval_Fraction_PCL_Total [Incl. Undet. Phase]	•	•	•	•	•						

Table 7. A complete inventory of Groups (Parameters) and Variables (Statistics) in the L3 COSP product. There are 32 Groups (or Parameters) in the product shown in the rows of the table. A dot in the table means that the particular Variable (Statistic) exists for that particular Group (Parameter). So, the 174 dots in the table above, represent the 174 group/variable (a.k.a. parameter/statistic) combinations defined in the L3 COSP product.

The CLDPROPCOSP NetCDF4 file size is roughly 60 MB for Daily (D3) and 80 MB for Monthly (M3). The file size is slightly larger for monthly since the more completely filled global grids for Monthly (M3) don't compress as well as the less-filled global grids for Daily (D3). Finally, note that the file format and structure is exactly the same between the Daily (D3) and Monthly (M3) NetCDF4 COSP files.

7.0.1. Histogram bin boundaries

The new CLDPROPCOSP files brought with them some changes to the joint histogram bin boundaries as compared to older datasets. The table below shows those changes. This was done primarily to provide a better match (correspond) to the domain of the L2 input data used. Note that Cloud Water Path was not provided at all in the original version, so those parameters are new and unique to the 2022 production version.

Parameter	Old Bin Boundaries	COSP Bin Boundaries
Cloud Optical Thickness Liquid	[0.0, 0.3, 1.3, 3.6, 9.4, 23.0, 60.0, 10000.0]	[0.0, 0.3, 1.3, 3.6, 9.4, 23.0, 60.0, 150.0]
Cloud Particle Size Liquid	[0, 8, 10, 13, 15, 20, 30]	[4, 8, 10, 12.5, 15, 20, 30]
Cloud Water Path Liquid	*Not Available*	[0, 10, 30, 60, 100, 150, 250, 20000]
Cloud Optical Thickness Ice	[0.0, 0.3, 1.3, 3.6, 9.4, 23.0, 60.0, 10000.0]	[0.0, 0.3, 1.3, 3.6, 9.4, 23.0, 60.0, 150.0]
Cloud Particle Size Ice	[0, 10, 20, 30, 40, 60, 90]	[5, 10, 20, 30, 40, 50, 60]
Cloud Water Path Ice	*Not Available*	[0, 20, 50, 100, 200, 400, 1000, 20000]
Cloud Optical Thickness Total	[0.0, 0.3, 1.3, 3.6, 9.4, 23.0, 60.0, 10000.0]	[0.0, 0.3, 1.3, 3.6, 9.4, 23.0, 60.0, 150.0]
Cloud Top Pressure	[0, 180, 310, 440, 560, 680, 800, 10000]	[0, 180, 310, 440, 560, 680, 800, 1100]

Units: Cloud Optical Thickness (none), Cloud Particle Size (microns), Cloud Water Path (g/m2), Cloud Top Pressure (hPa)

Table 7.0.1 New COSP bin boundaries are shown in the far-right column. Any changes as compared to old heritage numbers are shown in red.

7.1. Mapping Parameter Names from heritage 08_L3 output to new L3 COSP output

There were some slight changes made to the parameter names and keywords between the heritage 08_L3 products and the new L3 COSP products. COSP users should keep these details in mind in case some users were more familiar with the 08_L3 products. Since the COSP product is a daytime only product, the explicit _Day suffix was dropped on some parameters that previously had that suffix. The parameter Cloud_Fraction was expanded to

Cloud_Mask_Fraction in L3 COSP to add clarity. The keyword _Combined, previously used to denote parameters that combine all cloud phases (Liquid Water, Ice, and Undetermined Phase), was changed to _Total in the new L3 COSP product. The previous keyword _Log was changed to _Log10 for L3 COSP. The heritage 08_L3 parameter name Cloud_Effective_Radius was changed to Cloud_Particle_Size for L3 COSP.

Input 06_L2 SDS Name	Heritage 08_L3 Parameter Name	New L3 COSP Group Name
Cloud_Top_Pressure_Day	Cloud_Top_Pressure_Day	Cloud_Top_Pressure
Cloud_Fraction_Day	Cloud_Fraction_Day	Cloud_Mask_Fraction
Cloud_Optical_Thickness_37	Cloud_Optical_Thickness_37_Liquid	Cloud_Optical_Thickness_Liquid
Cloud_Optical_Thickness_37	Cloud_Optical_Thickness_37_Ice	Cloud_Optical_Thickness_Ice
Cloud_Optical_Thickness_37	Cloud_Optical_Thickness_37_Combined	Cloud_Optical_Thickness_Total
Cloud_Optical_Thickness_37	Cloud_Optical_Thickness_37_Log	Cloud_Optical_Thickness_Log10
Cloud_Effective_Radius_37	Cloud_Effective_Radius_37_Liquid	Cloud_Particle_Size_Liquid
Cloud_Effective_Radius_37	Cloud_Effective_Radius_37_Ice	Cloud_Particle_Size_Ice
Cloud_Water_Path_37	Cloud_Water_Path_37_Liquid	Cloud_Water_Path_Liquid
Cloud_Water_Path_37	Cloud_Water_Path_37_Ice	Cloud_Water_Path_Ice
Cloud_Phase_Optical_Properties	Cloud_Retrieval_Fraction_37_Combined	Cloud_Retrieval_Fraction_Total

Table 8. Users should note some changes in parameter names between the heritage 08_L3 (orange shading) and L3 COSP (violet shading) in case any comparisons are made between those two products.

8.0. Important Notes, Caveats, and Issues to Consider in L3 COSP

8.1. The logic used to compute the LMH aggregations of Cloud Fraction from Cloud Mask in the CLDPROPCOSP files

After a number of attempts of finding the most appropriate way to compute the Low, Middle, High Cloud (LMH) aggregations of Cloud Fraction, it was finally determined that the best method was to set any non-zero cloud fraction, which had a fill or missing cloud top pressure (a quantity needed to perform the cloud height aggregation), to zero cloud fraction (clear), before computing the LMH aggregations of cloud fraction. This change kept the fractions and pixel counts "well behaved" and made it more logically clear to users what the cloud fraction numbers mean. To provide more detail on this change, the new logic forces the pixel counts of Cloud Fraction Low+Medium+High to be exactly equal to the pixel counts of Cloud Fraction Total. The new logic also ensures that the actual Cloud Fractions of Low+Medium+High will always be less than or equal (more typically less than) to the Cloud Fraction Total.

8.2. Important: The 3.7-micron Cloud Optical Properties Retrieval, rather than the Primary 2.1 Retrieval, is used in COSP

It should be noted that there are four different flavors of Cloud Optical Property Retrieval Algorithms used in the MODIS Standard 06_L2 product: a.) a primary retrieval using 2.1 microns (really band 7 and either band 1, 2, or 5, depending); b.) an alternate retrieval using a combination of 1.6 and 2.1 microns (really band 7 and band 6, two channel); c.) an alternate retrieval using 1.6 microns (really band 6 and either band 1, 2, or 5, depending); and finally d.) an alternate retrieval using 3.7 microns (really band 20 and either band 1, 2, or 5, depending).

For CLDPROPCOSP, it was decided to use option d.) above, the alternate retrieval using 3.7 microns. This was done to better match the COSP Simulator data.

Therefore, COSP data users should be aware that the Cloud Optical Thickness, Cloud Particle Size, Cloud Water Path, and Cloud Retrieval Fraction parameters in the COSP files were all derived from this 3.7-micron Cloud Optical Properties Retrieval Algorithm sourced from the MODIS Standard 06_L2 files.

It should also be noted that the SDS names in the COSP files do not have an "_37" suffix appended, like these alternate retrievals do in the MODIS Standard (06_L2 and 08_L3) files, but they do have a reference to this 3.7-micron data source in the "long_name" local attribute, so this is documented in the file itself.

8.3. Cloud Retrieval Fraction Pixels Counts: MOD08 vs. COSP

A difference in the statistics for Cloud Retrieval Fraction-related parameters is noted in the Pixel Counts statistic. In MODIS Standard MOD08, the Cloud Retrieval Fraction Pixel Counts is the numerator of the fraction (the number of cloudy pixels). While in L3 COSP, the Cloud Retrieval Fraction Pixel Counts is the denominator of the fraction (the total number of clear + cloudy pixels). This difference in the meaning of Pixel Counts for Cloud Retrieval Fraction Groups results from the different way this quantity was calculated in MODIS Standard vs. COSP, the former using the heritage L3 Code run at MODAPS, and the latter run with the new YORI L3 Code run at SIPS (computed in the Yori framework).

If a user would like to see the numerator (the total number of cloudy pixels by cloud phase) for Cloud Retrieval Fractions in L3 COSP, simply multiply the Fraction (ranging from 0.0 to 1.0) by the Pixel Counts for that particular parameter, stored in the product.

Note that there is no change in the meaning of Pixel Counts for Cloud Mask Fraction in L3 COSP, because for that parameter, cloud fractions are computed in advance on a 5x5 km grid in the input L2 product (06_L2).

For all other types of Parameters (that are not Cloud Retrieval Fraction), the Pixel Count has the traditional meaning. That is, it's the number of valid or successful L2 sampled retrievals for each particular Parameter.

8.4. Monthly Cloud Top Property (CTP) Weighting: MOD08 vs. COSP

Users comparing CTP-related data in the monthly MODIS COSP file (CLDPROPCOSP) to the MODIS Standard data file (MOD08_M3) should be aware of a difference in how the monthly mean and standard deviation statistics were computed for all Cloud Top Property (CTP)-related parameters.

The monthly MODIS Standard (MOD08_M3) product uses an *Unweighted Weighting Scheme* to compute Cloud Top Property-related statistics. That is, monthly MODIS Standard (MOD08_M3) CTP-related statistics (e.g.: Cloud_Top_Temperature_Mean_Mean) is an un-weighted *mean of the daily mean* without regard to pixel counts observed on each day.

The monthly COSP (CLDPROPCOSP_M3) product uses a *Pixel Count Weighting Scheme* to compute CTP-related statistics (e.g.: Cloud_Top_Temperature, Mean). So, there is a difference between how the CTP parameters are being computed in a MODIS Standard M3 file (un-weighted) vs. a COSP M3 file (pixel-count weighted). Since the COSP Simulator data uses a pixel-count weighting scheme, it seemed appropriate to switch to that scheme for all Cloud Top Property-related groups.

8.5. Simple Stats most users can ignore: Sum & Sum_Squares

There are five standard simple statistics (computed as a set) in almost all L3 COSP Groups: Mean, Standard_Deviation, Pixel_Counts, Sum, and Sum_Squares. Users can generally ignore the last two statistics in that list (Sum and Sum_Squares) as they are generally not deemed useful to most users. However, they are a required element in Yori to facilitate recomputing the Mean and Standard_Deviation on-the-fly as data is aggregated (spatially and temporally) in L3.

8.6. Pixel_Count Array Initialization in COSP

Note that Pixel Count statistics in L3 COSP (produced by the Yori System) are initialized with zero -- and not fill (missing) as was done for MODIS Standard MOD08. This means orbital gaps and day/night boundary regions are shown as 0 and not fill in all count-related data/imagery. In other words, Pixel_Count images will not show (as clearly) data gap (missing data) regions (typically seen between orbital swaths in Daily images for MODIS and over the poles for day- or night-only data).

This statistic initialization rule also applies to histogram count and joint histogram count arrays – so basically all count arrays are initialized with zero (and not fill) in L3 COSP. If a user wanted to show those missing data gaps on their Pixel_Count images, they would need to mask those images with the fill value from the Mean image in that same group. This visualization issue of fill vs. zero is less important for Histograms and Joint Histograms since those statistics tend not to be displayed on a geographic map -- so initializing with zero is less noticeable.

The L3 Development Team did attempt to get the UW Yori Development Team to change this procedure for initializing Pixel_Counts in Yori, but they seemed set on this decision saying that it made more sense for them to do it that way. Looking back on it now, it probably had a lot to do with how Yori is computing statistics on the fly and simplifying the Yori computational logic.

This issue does not impact the Mean and Standard Deviation statistics because they are initialized with fill-values (missing).

8.7. Histogram Bin Boundary definitions in COSP

There is a slight bin boundary handling change between MODIS Standard and L3 COSP. For L3 COSP (CLDPROPCOSP), the first through the next-to-last (penultimate) histogram bin includes L2 data points that fall on the lower bin boundary. The last (ultimate) histogram bin includes L2 data points that fall on both the lower and the higher bin boundary.

For L3 MODIS Standard MOD08, the first histogram bin includes L2 data points that fall on both the lower and higher bin boundaries. All subsequent histogram bins include points that fall on the higher bin boundary.

For both streams, any L2 data point that falls outside the specified range of L3 histogram bin boundaries is not counted.

At first the L3 Development Team attempted to get the Yori developers to match the old MODIS Standard convention, but in the end it was decided it was only a minor issue that would not affect any science results, so it was left as is. It remains unclear if there is a widely accepted standard for how the binning in histograms and joint histograms is handled.

8.8. A Quick Review of Day/Night Algorithm Cut-Off: Cloud Top/Cloud Mask vs. Cloud Optical

The Day/Night separation for Cloud Top Parameter-related parameters in MODIS Standard MOD06/MOD08 was based on the L2 Cloud Mask QA Day/Night Flag. Note that this Day/Night Flag uses Solar Zenith Angle ≤ 85 degrees as a daytime cutoff.

For Cloud Optical Property L2 MODIS Standard products, the daytime cutoff is set to 81.3731 degrees. This means the Cloud Top-related parameters will extend slightly farther north on a global grid.

Algorithm	**MODIS Standard Atmosphere** Day/Night Cut off
Cloud Mask Algorithm	Solar Zentih ≤ 85°
Cloud Top Property Algorithm	Solar Zentih ≤ 85°
Cloud Optical Property Algorithm	Solar Zentih ≤ 81.3731°

Table 9. The Different ways the three Cloud Algorithm Teams (Cloud Mask, Cloud Top, and Cloud Optical Properties) have defined the Day/Night Cutoff for MODIS Standard Products (MOD06, MOD08). L3 COSP follows this definition.

8.9. File format tweaks to improve interaction with Visualization Tools (e.g.: Panoply)

Early in the development of L3 COSP products, Panoply (a popular visualization and analysis tool) was not handling the NetCDF4 files as expected in creating optimal image titles by default. The primary reason for this was the new Group/Variable format of L3 COSP vs. the old standalone Scientific Data Set (SDS) format of L3 MODIS Standard MOD08.

When Panoply titles images from a NetCDF4 file with the Group/Variable structure, the Group information is ignored, and instead, Panoply uses information attached to the Variable only. In this paradigm, Panoply creates image titles based on the following list of items, any of which may be associated with a specific variable -- in descending order of priority: 1.) the *long_name* attribute, 2.) the *title* attribute, 3.) the *standard_name* attribute, and 4.) the variable's *simple name*. The first three items in the list are defined through the use of local attributes attached to the variable.

Option 1, the *long_name* attribute, which is, by default, attached to the Group as a group attribute in the L3 COSP NetCDF4 file, was too wordy and long-winded to use as an image title, so we decided not to transfer (or pass-through) that string to a local attribute attached to variables in that Group (so that Panoply would "pick it up" when it read each variable). Option 4, the variable's *simple name*, was too short (too cryptic) with not enough information included -- it's simply the name of the variable (e.g.: Mean).

This left only two options on the table to remedy this shortcoming. The *title* attribute (option 2) and the *standard-name* attribute (option 3).

The *title* attribute is described in the NetCDF4 User Guide as *a character array providing a succinct description of what is in the variable*. The *standard_name* attribute is described in the NetCDF4 User Guide as *a long descriptive name for the variable taken from a controlled vocabulary of variable names*.

Of these two remaining choices, we chose option 2, the ***title*** attribute, to solve our titling problems in Panoply. We made this choice for three primary reasons. First, the ***title*** attribute was a higher priority than the ***standard_name*** attribute in Panoply's internal rankng system – and we thought it was prudent to use the higher ranking vehicle. Second, we found the name of the keyword itself, "title", the most descriptive, since this whole process was done to fix how Panoply *titles* images from L3 COSP files -- so why not use an attribute name that captures that meaning literally. (This also makes it easier to for users to determine what that attribute means without added documentation). Third, when we read the meaning of the ***title*** attribute in the NetCDF User Guide, it used the word *succinct* (meaning a concise and compact description). So a ***title*** local attribute is defined to be a shorter laconic string (while still defining the variable fully), and therefore felt it would best fit in the limited space available for most image titles, while retaining enough information to fully describe the variable.

Therefore, in our final production version of the L3 COSP NetCDF4 file, an attribute called ***title*** is attached to **all** variables and provides information that can be passed to various visualization packages (e.g.: Panoply) telling the package how to explicitly title images produced from that variable. Under most circumstances, these ***title*** local attributes are a simple combination of the Group name and the Variable name, which matches fairly closely how images were generally titled for variables (statistics) imaged from MODIS Standard Products. This new ***title*** local attribute was implemented in Yori Software version 1.3.8 and later, which was launched on 9 July 2019, and is reflected in all L3 COSP files.

Note that custom software tools, manually written in Python, IDL, or other languages, by individual users, can be designed to do this sort of Group and Variable combining on the fly to make well-behaved image titles, however we were looking for a way to standardize how our files were going to pass titling information to pre-packaged commercial or standardized (widely distributed) visualization tools (like Panoply).

8.10. Data Types of Variables with the L3 COSP NetCDF4 file

All the statistics in this L3 (except for Pixel_Counts, Histograms, and Joint_Histograms) are stored in Floating-Point Double-Precision Real in the final gridded NetCDF4 COSP output file (by default) – that is, R*8 output is hardcoded within the Yori processing system for those statistics. For all the "count"-related statistics such as Pixel_Counts, Histogram_Counts, and Joint Histograms, they are stored in Integer I*4.

The new NetCDF4 format and advanced compression tools pose no issues with file size. Note that the Fill Value in these new NetDF4 files is hardwired to -999.0 and is documented in the local attributes of the file itself.

8.11. Intermittent Bug in 06_L2 Cloud Top Properties impacts L3 Pixel Counts

In July 2020, a COSP data user found a few cases in the CLDPROPCOSP Daily (D3) product where the pixel counts for Cloud_Mask_Fraction were less than the pixel counts for Cloud_Retrieval_Fraction_Total, something that should never occur given the logical structure of the two source retrieval algorithms.

After an intensive investigation, it was discovered that the Cloud Top Property Retrieval algorithm was, very intermittently, producing fill (missing) data in regions where the Cloud Mask was valid and defined as cloudy -- meanwhile, in these same very limited regions, the Cloud Optical Property retrieval algorithm was producing valid data.

Due to the fairly large grid size in Level-3 (1x1 degree), it was determined that these small areas of missing Level-2 data did not impact the gridded Level-3 mean (or mean fraction) values (e.g. Cloud_Mask_Fraction) -- however, the pixel count values were impacted and pixel count differences are noticeable. This caused, in some very limted regions, the L3 gridded pixel count values for the parameter Cloud_Mask_Fraction to be less than the pixel count values for Cloud_Retrieval_Fraction_Total, which should never occur with bug-free source algorithms.

8.12. Summary webpage outlining known issues

Users should be aware that they stay apprised of known issues in the Continuity Cloud Product (L2 used as input into the L3 COSP files) by visiting the webpages at

https://atmosphere-imager.gsfc.nasa.gov/issues/continuity/cldprop_l2

https://atmosphere-imager.gsfc.nasa.gov/issues/continuity/cldprop_l3

9.0. Using the newly redesigned Atmosphere-Imager web site

The original MODIS-Atmosphere web site (developed in 1999) was initially designed for MODIS Standard Products, such as MOD06_L2 (L2 Cloud Product) and MOD08_M3 (L3 Global Gridded Monthly Product). This site was completely redesigned in 2018 (using the state-of-the-art Drupal framework) and adapted to include the new Continuity Products. (Screen-captures in Figures 14 through 17.) The redesigned website was renamed the **Atmosphere-Imager** (AI) website and can be viewed at: https://atmosphere-imager.gsfc.nasa.gov

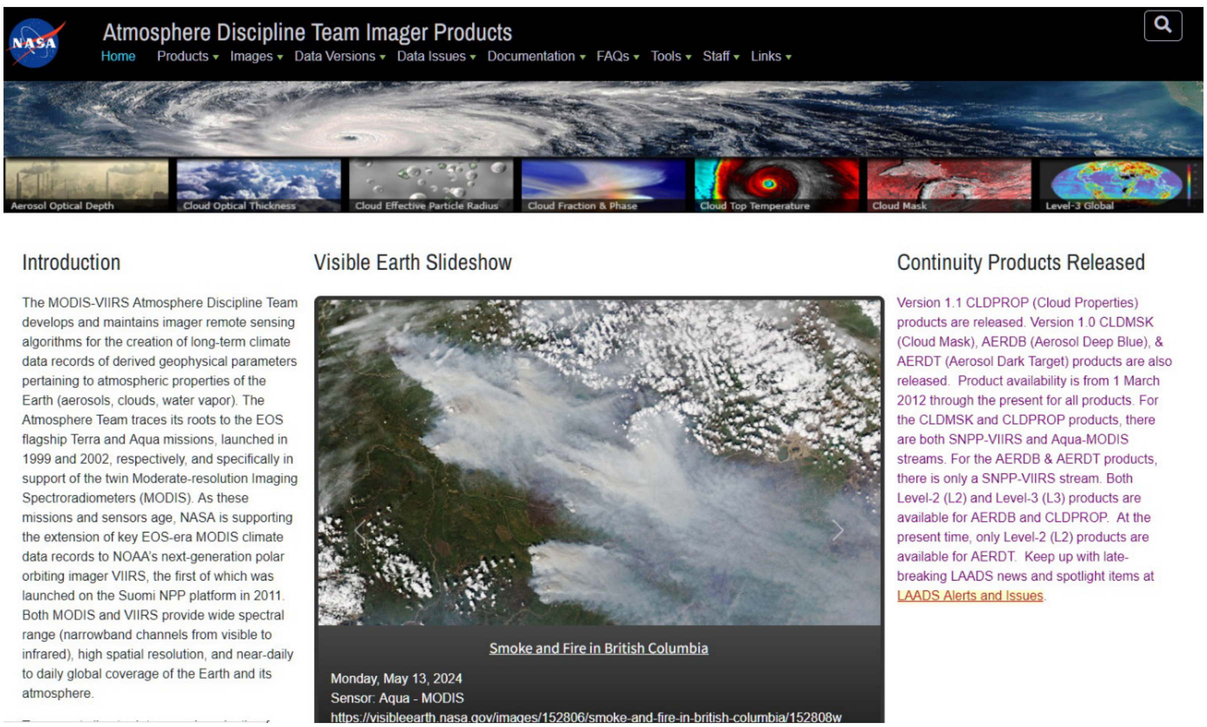


Figure 14. A capture of the new Atmosphere Imager website that was redesigned with a split navigation bar, allowing visitors to view both MODIS Standard content as well as new Continuity content.

The Atmosphere-Imager website has a new user-friendly navigation interface that can be used to view either heritage MODIS Standard content or newer Continuity content.

Note that the COSP Product (CLDPROPCOSP), relates directly to the L2 CLDPROP content, which is the direct input to the L3 CLDPROPCOSP product.

This navigation *split* can be found if a visitor *pulls-down* some of the main topic links displayed at the top of all pages. The main topic links that have this navigation *split* are: 1.) Products, 2.) Images, 3.) Data Versions, 4.) Data Issues, 5.) Documentation, and 6.) FAQs.

By selecting the "MODIS Standard" option on any of these main topic pulldown menus, one branches into sections of the website geared towards the Atmosphere MODIS Standard Products (such as 06_L2, which is the input to COSP (CLDPROPCOSP).

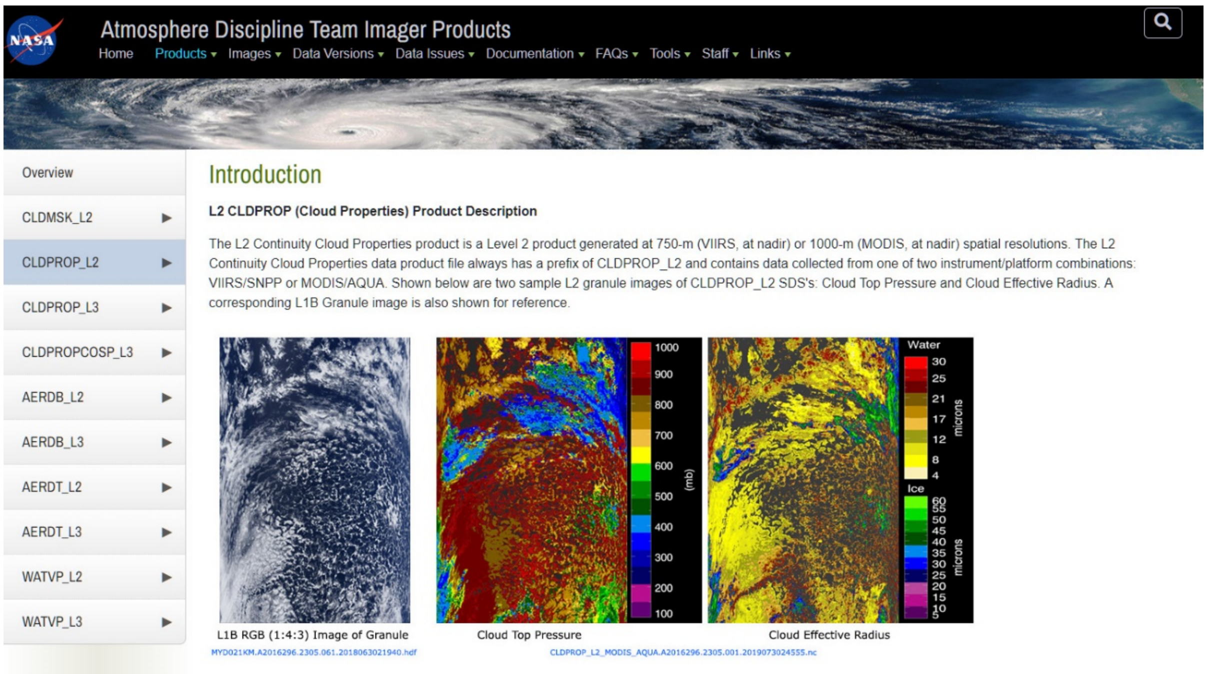


Figure 15. A capture of the Products > Continuity > CLDPROP_L2 > Introduction page. A pull-down sub menu on the left-hand side, allows visitors to view more subpages related to the CLDPROP L2 Product

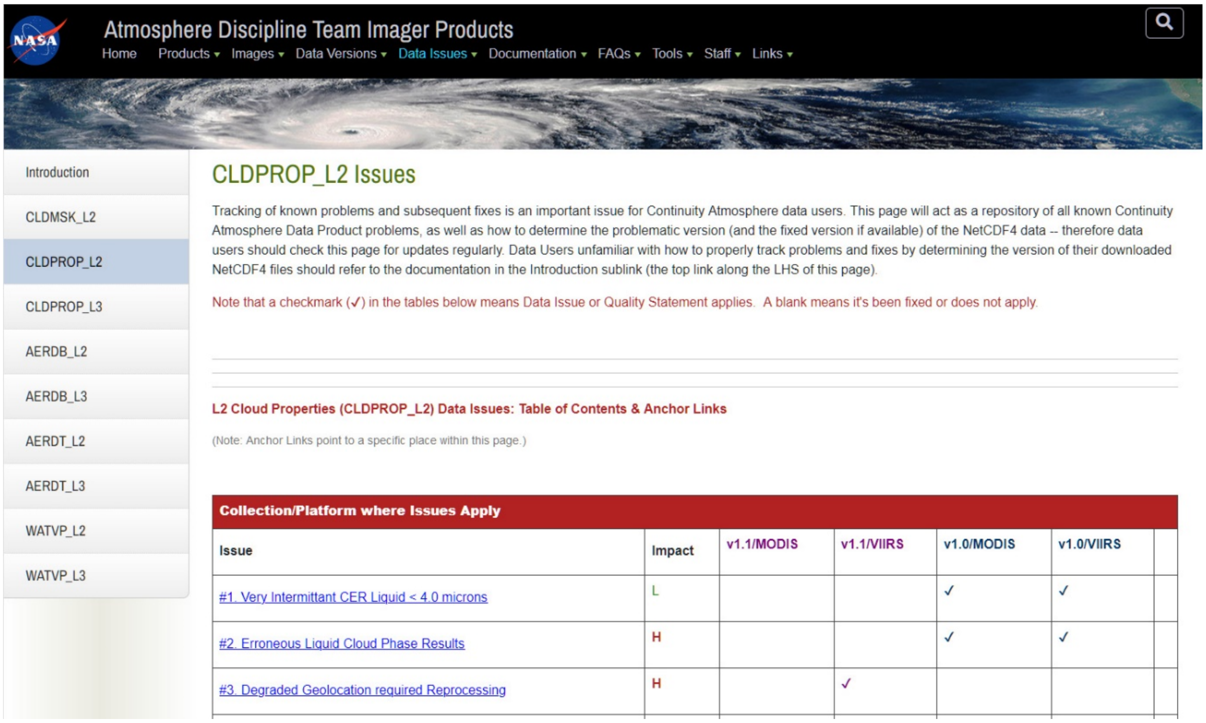


Figure 16. A capture of the Data Issues > Continuity > CLDPROP_L2 page. As problems or issues are discovered in the L2 data, they are posted here to alert the user community. Also provided are fixes (if possible) as well as a graphical *Impact Meter* to give a quick overview as to whether an issue has Low, Medium, or High Impact. A table of contents with jump links to the relevant part of the page below is included at the top of the page for easy navigation.

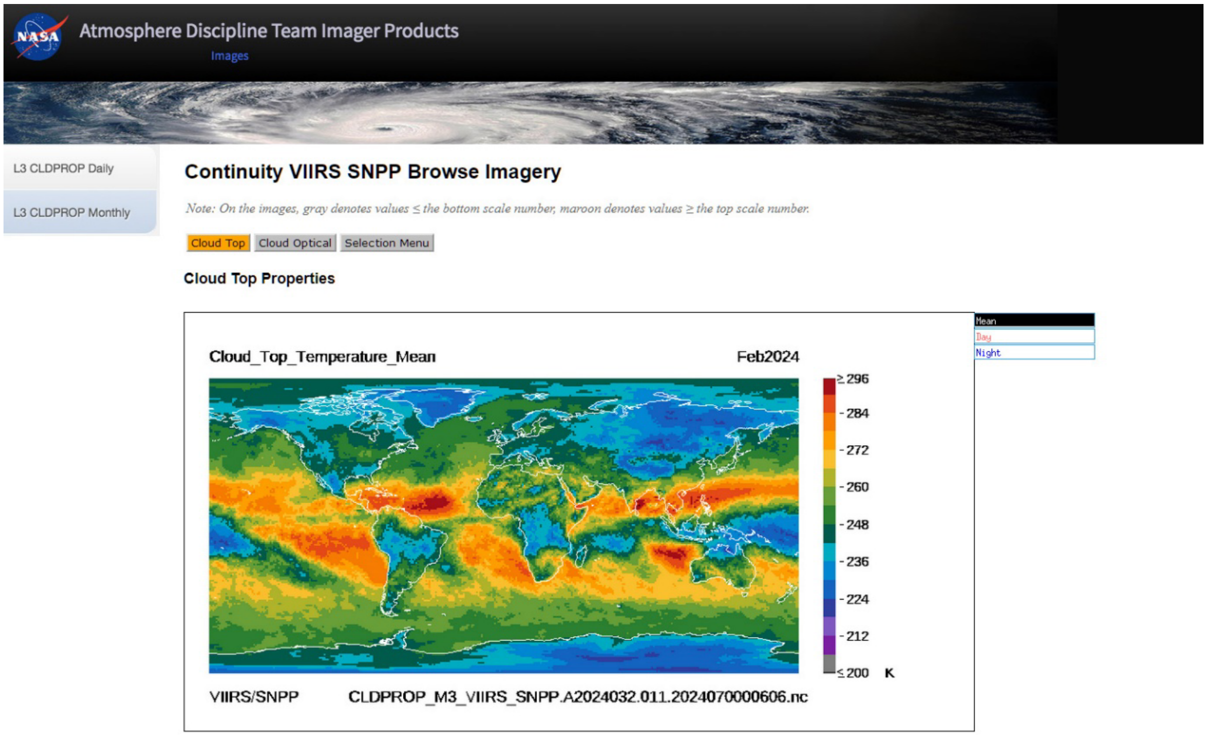


Figure 17. A capture of the Images > Continuity > L3 Monthly page. Here, users can find browse images for L3 CLDPROP Products.

10.0. Interpretation of Data: Frequently Asked Questions

Since the release of L3 Continuity data, users have asked the L3 development team and associated Atmosphere staff a number of questions. This section summarizes the most commonly asked questions and their answers.

10.1. How should joint histograms be converted to cloud fraction?

Joint histograms are reported as integers: the number of pixels observed with the spatial extend (1 degree box) and time period (daily, monthly) with values falling in some range. Reporting in counts allows YORI to produce the monthly files by summing over a collection of daily files, but joint histograms in the scientific literature are often expressed in terms of a cloud fraction.

All COSP joint histograms are associated with cloud optical property retrievals (as opposed to cloud mask retrievals). To express the joint histograms in terms of cloud fraction, users should divide the pixel count value in each histogram bin by the "Pixel_Counts" variable in the "Cloud_Retrieval_Fraction_Total" group for the same grid element. The sum of the resulting histograms agrees with the "Mean" variable in the "Cloud_Retrieval_Fraction_Total" group to within rounding error for joint histograms of cloud optical thickness, cloud water path, and/or cloud particle size. It should be noted that Joint histograms involving cloud top pressure may be slightly (tenths of a percent) smaller than the mean cloud fraction because Cloud_Top_Pressure is only determined where cloud mask cloud fractions are greater than or equal to 16%. In other words, cloud top pressure in a 5x5 km L2 grid is only computed when at least 4 out of the 25 (16%) 1x1 km L2 grids are deemed to be cloudy by the Cloud Mask.

10.2. What is the difference between Cloud Fraction (i) from Cloud Mask, and (ii) from Cloud Optical Property Retrievals?

In order for users to appreciate the meaning of the two basic cloud fractions in L3 COSP products, they should understand how cloud fraction is defined and computed in each L3 cloud fraction parameter. Most simply, L3 cloud fraction is the number of cloudy L2 pixels (that fit pre-determined criteria, such as daytime only or liquid water clouds only) divided by the total number of non-fill L2 pixels within a 1×1° L3 grid box. The key detail is how each parameter defines a pixel as "cloudy" and what domain of pixels is used to compute the denominator. L3 COSP files have two primary types of cloud fractions.

10.2.1. Cloud Fraction from Cloud Mask

The L3 cloud fraction that garners the most interest from Atmosphere data users is Cloud Fraction derived directly from the Cloud Mask. These L3 COSP cloud mask cloud fractions have the Group names:

- Cloud_Fraction
- Cloud_Fraction_Day
- Cloud_Fraction_Night

The first parameter listed above contains both day and night retrievals, the second parameter contains daytime-only retrievals (solar zenith angle less than or equal to 85°), and the third parameter contains nighttime-only retrievals.

The cloud mask cloud fraction is computed in L3 using the Cloud Mask Status Flag and the Cloud Mask Cloudiness Flag, both a part of the Quality_Assurance SDS in L2 COSP. See Table 10.

L2 QA Flag	Flag Value	Meaning
Cloud Mask Status Flag	0 1	Undetermined Determined
Cloud Mask Cloudiness Flag	0 1 2 3	Confident Cloudy (or Fill if Status Flag = 0) Probably Cloudy Probably Clear Confident Clear

Table 10. Two L2 Cloud Mask Flags used to compute the L3 Cloud Mask Cloud Fraction.

In the computation of the L3 cloud mask cloud fraction, the first two flags are assigned 100% cloudy and the last two flags 100% clear. Then in each 1x1 degree L3 grid, the mean L2 cloudiness is computed based on those results.

10.2.2. Cloud Fraction from Cloud Optical Property Retrievals

The second-most utilized L3 cloud fraction is the one derived from the Cloud Optical Properties retrieval. All optical property cloud fractions are computed for **daytime-only scenes** (solar zenith angle ≤ 80°).

These L3 COSP cloud optical property cloud fractions have the Group names:

- Cloud_ Retrieval_Fraction_Liquid
- Cloud_ Retrieval_Fraction_Ice
- Cloud_ Retrieval_Fraction_Undetermined
- Cloud_Retrieval_Fraction_Combined

The first parameter represents the cloud fraction for the liquid water cloud phase only; the second, ice phase clouds only; the third, undetermined phase clouds only; and the fourth, combined (all) cloud phase clouds.

The optical property cloud fraction is computed in L3 using the Primary Cloud Retrieval Phase Flag and the Primary Cloud Retrieval Outcome Flag, both a part of the Quality_Assurance SDS in L2 COSP. See Table 11.

Every sampled L2 grid point that has a Primary Cloud Retrieval Outcome Flag = 1 (Retrieval Successful) and a Primary Cloud Retrieval Phase Flag of 2 (Liquid Water Cloud), 3 (Ice Cloud), or 4 (Undetermined Phase Cloud) are taken as 100% cloudy for the cloud phase category in question. Then in each 1x1 degree L3 grid, the mean L2 cloudiness is computed based on those results.

L2 QA Flag	Flag Value	Meaning
Primary Cloud Retrieval Phase Flag	0 1 2 3 4	Cloud Mask Undetermined (Missing or Fill) Not Processed (Typically Clear) Liquid Water Cloud Ice Cloud Undetermined Phase Cloud
Primary Cloud Retrieval Outcome Flag	0 1	Retrieval Not Attempted or Unsuccessful Retrieval Successful

Table 11. Two L2 Quality Assurance Flags used to compute the L3 Cloud Optical Properties Cloud Fraction.

10.2.3. Formula to recover the total number of cloudy (or true) pixels in L3 COSP

To retrieve the numerator of the cloud fraction, that is, the number of cloudy (or true) pixels (the fraction numerator) in L3 COSP fraction groups, simply multiply the Pixel_Counts by the Mean (Fraction) in the particular cloud fraction group under consideration.

10.3. What is the Cloud Optical Property Retrieval "Flavor" or "Type" used in L3 COSP?

L3 COSP data users should keep in mind that the 06_L2 Cloud Optical Properties (COP) retrieval algorithm has 4 different flavors or types. They include COP Retrievals based on 2.1-micron radiance data, retrieval based on 1.6-micron radiance data, retrievals based on 3.7 radiance data, and retrievals based on 1.6-2.1 radiance data. For L3 COSP, only one of these flavors is used: retrievals based on 3.7-micron radiance data. The reason for this is the COSP model uses the *Look Up Tables* (LUTs) for 3.7 micron in the code.

10.4. What is the meaning of Undetermined and Total (Combined) Cloud Phases?

The undetermined cloud phase means the cloud optical properties retrieval algorithm could not make a determination of the cloud phase (liquid water or ice). This may have been caused by viewing anomalies in the retrieval (sunglint), contamination of the scene by aerosol, or a multi-layer cloud with mixed phases (e.g., thin cirrus overlying liquid water clouds). For these undetermined retrievals the liquid water libraries are used in the cloud optical properties retrievals, but the retrievals are considered to be of lower confidence (and quality) than those that are placed in one of the other primary phase categories (liquid water and ice). The total or combined phase is simply a combination of all cloud phase categories: liquid water, ice, and undetermined.

L3 COSP (CLDPROPCOSP) data users should keep in mind that there is no separate aggregation (Groups) of Undetermined Phase Clouds, however those Undetermined Phase Clouds are included in the Total or Combined Cloud Phase Groups.

10.5. What is the meaning of Partly Cloudy (PCL) Cloud Optical Property parameters?

L3 COSP files contain one PCL (Partly Cloudy) Cloud Optical Property Group or Parameter, which has the string "_PCL" in the Group name. These are slightly less reliable than the regular cloudy retrievals, which are used in all other Groups in the L3 COSP file -- therefore, this PCL Retrieval data was separated into a stand-alone Group so that users could decide either to mix them in with the regular retrievals or exclude them.

10.6. What is the meaning of Low, Mid, & High Cloud aggregations in Cloud Mask Fraction Groups?

The L3 COSP Product contains a Cloud Mask Fraction Group, which is a Cloud Fraction based on the Cloud Mask. This variable is for daytime only and for all pressure levels (total column):

- Cloud_Mask_Fraction (daytime only, all pressure levels)

Next, there are aggregations of that same data for Low, Mid, and High Clouds. The definitions of these aggregations are based on Cloud Top Pressure data. Those aggregations are defined as follows

- Cloud_Mask_Fraction_Low (daytime only, CTP $\geq$ 680 hPa)
- Cloud_Mask_Fraction_Mid (daytime only, 680 hPa $>$ CTP $\geq$ 440 hPa)
- Cloud_Mask_Fraction_High (daytime only, CTP $<$ 440 hPa)

10.7. Why do Joint Histogram bin boundaries sometimes exceed the valid range of data?

L3 COSP data users may notice that some of the Joint Histogram bin boundaries exceed the valid range of the input MODIS data. This occurs because the COSP model and other climate models have certain standardized histogram bin boundaries that are required as input into models. In these cases, the valid range of the data is ignored and joint histograms default to those standardized bin boundaries.

For example, in Cloud Top Pressure, the L3 COSP upper Joint Histogram bin boundary is set to 10,000 hPa or mb. This greatly exceeds real-world values or the valid range of the L2 MODIS input data, which maxes out at 1,100 hPa or mb.

Another example is from Liquid Water Cloud Particle Size (a.k.a. Cloud Effective Radius for Liquid Water Clouds), where the L2 MODIS input data has a valid range of 4 to 30 microns, however the L3 COSP Joint Histogram ranges from 0 to 30. Similarly for Ice Water Cloud Particle Size, the L2 MODIS input data has a valid range of 6 to 60 microns, however the L3 COSP Joint Histogram ranges from 0 to 90 microns.

A final example is from Cloud Optical Thickness, where the L2 MODIS input data has a valid range of 0.01 to 150.0, however the L3 COSP Joint Histogram goes from 0 to 10,000.

The following explanation was offered to account for the difference between the Joint Histogram's bin range and L2 MODIS data's valid range: One can see the boundaries used by the COSP model (and also with ISCCP) in this online reference:

https://github.com/CFMIP/COSPv2.0/blob/master/src/cosp_config.F90

All units are in MKS (meters, kilograms, seconds) except pressure is in hPa/mb instead of Pa:

tau_binBounds = (/0.0, 0.3, 1.3, 3.6, 9.4, 23., 60., 10000./)

pres_binBounds = (/0., 180., 310., 440., 560., 680., 800., 10000./)

reffLIQ_binBounds = (/0., 8e-6, 1.0e-5, 1.3e-5, 1.5e-5, 2.0e-5, 3.0e-5/)

reffICE_binBounds = (/0., 1.0e-5, 2.0e-5, 3.0e-5, 4.0e-5, 6.0e-5, 9.0e-5/)

Given that the original source information is from MODIS, an argument exists to change the particle size boundaries to match the valid range of MODIS data; but the consensus views it as more important to have the boundaries in the L3 COSP data match what the COSP model reports. We may see these bin boundaries revised in a future revision.

10.8. What is the best way to display or visualize 2D joint histogram data?

Common ways to display Joint Histogram data are (i) 3D lego plots, (ii) smoothed contour plots, and (iii) color-coded histogram bin plots.

Some graphics packages render Lego plots (Figure 18); however it's often difficult to read the exact height of each lego block, and some (or most) viewing orientations may cause some blocks to remain hidden by larger blocks in front of them.

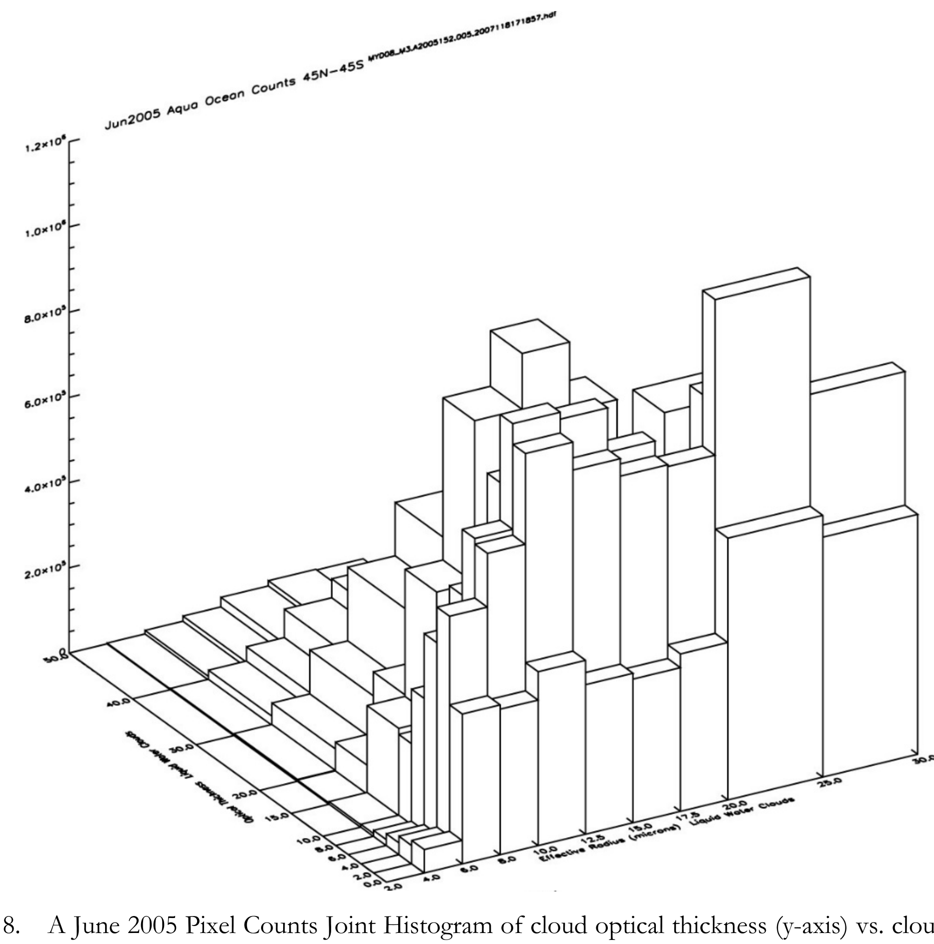


Figure 18. A June 2005 Pixel Counts Joint Histogram of cloud optical thickness (y-axis) vs. cloud effective

radius (x-axis) for liquid water clouds displayed as a "3D lego plot" with post-processing to limit the data to ocean-only L3 grid cells that range from 45°N to 45°S. The top bin of cloud optical thickness from 50 to 100 was chopped off. The height of each Lego bar represents the number of counts in each bin. Orienting all bins to render lego plots visible is often difficult to impossible.

Smoothed contour plots (Figure 19) are useful as they allow quick visual interpretation of data (without an intimate knowledge of the color bar scale) and therefore lend themselves well to time-series animations, where the time to view each joint histogram movie frame is short. However, smoothing options can be difficult to fine-tune, and the result is always a plot that shows a distorted (smoothed) view of the actual data.

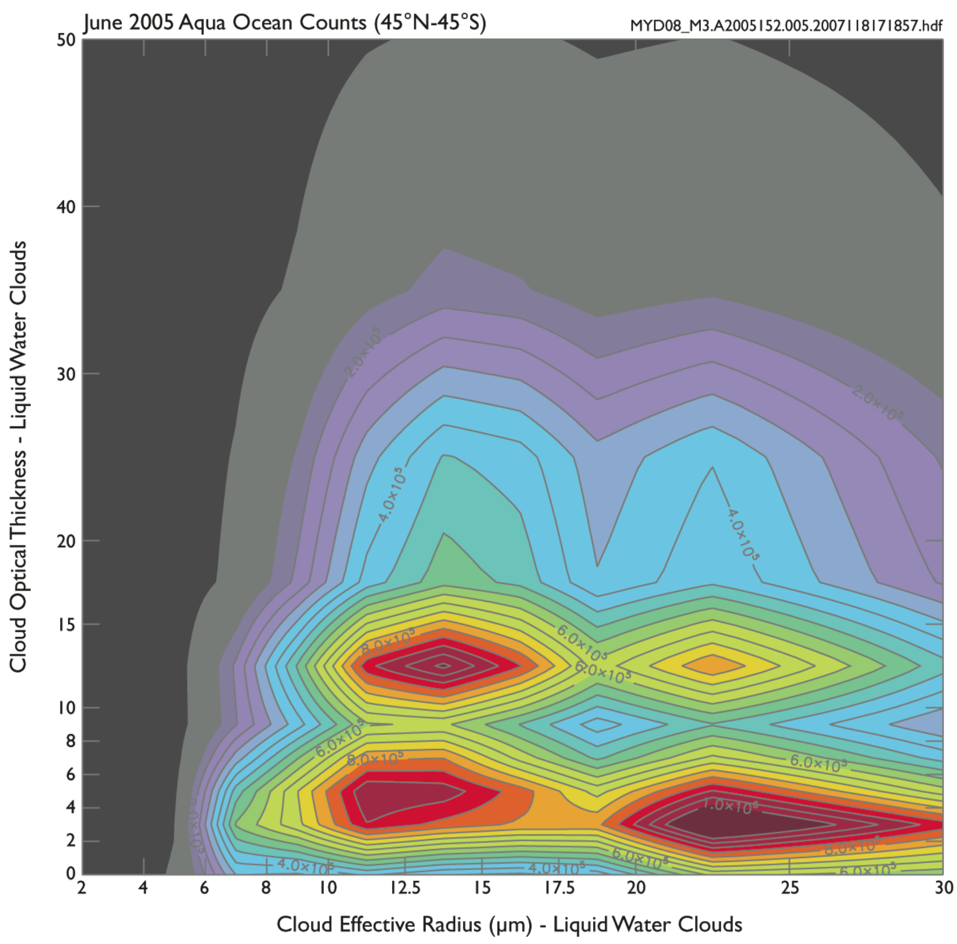


Figure 19. A June 2005 Pixel Counts Joint Histogram of cloud optical thickness (y-axis) vs. cloud effective radius (x-axis) for liquid water clouds displayed as a "colored contour plot" with post-processing to limit the data to ocean-only L3 grid cells that range from 45°N to 45°S. The top bin of cloud optical thickness from 50 to 100 was chopped off. The contour colors represent the number of counts in each bin (low is grey, high is red). Contour plots show a distorted (smoothed) view of the data stored in the joint histogram.

The third option, color-coded histogram bin plots (Figure 20), work best in most cases and have few drawbacks; however, some graphics packages cannot produce these plots without expert knowledge of the language and some user programming. The pros of this final method are there is no distortion (smoothing) of data in the x-y plane (as is the case with smoothed contour plots) and all cells (bins) are viewable without any visual obstruction (as is the case with lego plots). Figures 17, 18, and 19 show identical L3 joint histogram counts data plotted using these three described techniques.

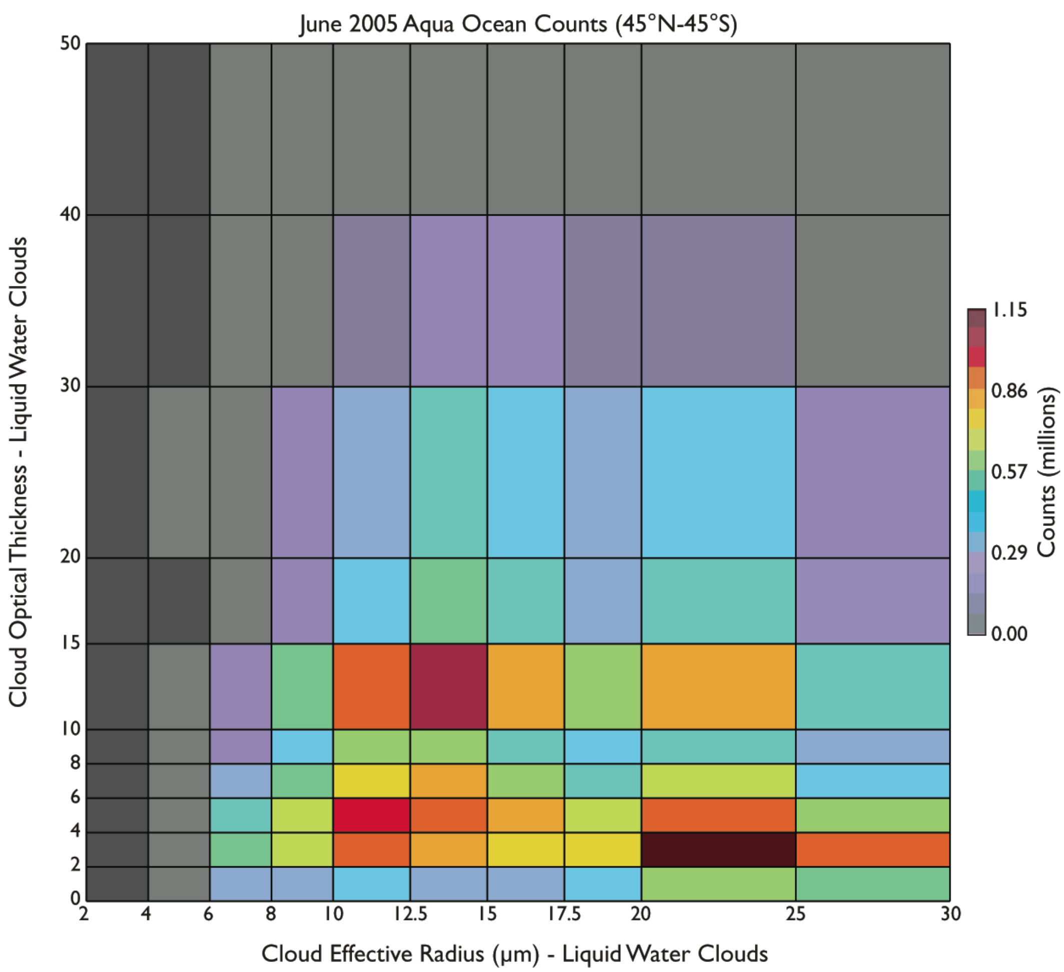


Figure 20. A June 2005 Pixel Counts Joint Histogram of cloud optical thickness (y-axis) vs. cloud effective radius (x-axis) for liquid water clouds displayed as a "colored histogram bin plot" with post-processing to limit the data to ocean-only L3 grid cells that range from 45°N to 45°S. The top bin of cloud optical thickness from 50 to 100 was chopped off. The bin colors represent the number of counts in each bin (low is grey, high is red).

For optimal display, it's best to have the graphics package print tick marks and labels at the exact histogram bin boundaries on both axes, so that the bin boundaries and sizes are clear to the viewer. These bin boundaries can be obtained from local attributes attached to each Joint Histogram variable (statistic) within the L3 COSP file. They are also provided in Appendix A of this User Guide for convenience.

Since all L3 joint histograms are stored as 4D arrays (Lat, Lon, Parameter1Bin, Parameter2Bin), MODIS data users can read (and view) a unique joint histogram for every L3 1° × 1° grid cell. This allows users to easily perform post-processing to narrow the application of the joint histogram to specific global regions to show more focused scientific results.

Users can apply a latitude/longitude screen to the 1°×1° gridded histograms so a newly computed (summed) joint histogram applies to a specified range of latitudes and longitudes only.

Users can also apply land-only or ocean-only masks when summing individual 1° × 1° gridded histograms so the newly summed joint histogram applies to land-only or ocean-only regions.

Occasionally, users may also find it advantageous to delete (remove) the highest bin along one or both data axes (which typically have a large data range and few counts) when displaying joint histograms in order to show more detail in other more scientifically relevant bins.

Another item to keep in mind is one can plot "counts" (which is how the data are stored in the L3 HDF file) in each bin; or sometimes it is more useful to normalize the data by taking into account the bin sizes and then plot "normalized probability" in each bin. This normalized probability calculation is performed as follows:

$$\text{Normalized Probability} = (\text{bin\_box\_counts}) / (\text{bin\_box\_area} * \text{total\_counts}) \qquad (5)$$

Converting the counts to normalized probabilities removes the visual anomalies that occur when comparing bins of different sizes in joint histogram plots. Figure 21 shows how the representation in Figure 20 changes when going from raw counts to normalized probability (where an adjustment is made for bin sizes). It's clear that normalizing the data can really change one's perspective of the results.

Finally, some ask, what do the Probability Density Function (PDF) numbers computed for each histogram bin in Figure 21 actually (physically) represent? If a normalized one-dimensional histogram plot means the area under the curve equals one; then a normalized joint histogram PDF plot means the volume under the surface equals one. (The surface being defined as z-axis heights or PDF values in each bin.) If one calculates the (area of bin) x (height of bin) or, alternatively (area of bin) × (PDF value of bin) then a "volume of the block" calculation is actually being performed. When one sums all the block volumes, one ends up with the volume under the surface, which for the normalized JPDF surface is 1.0 or 100%.

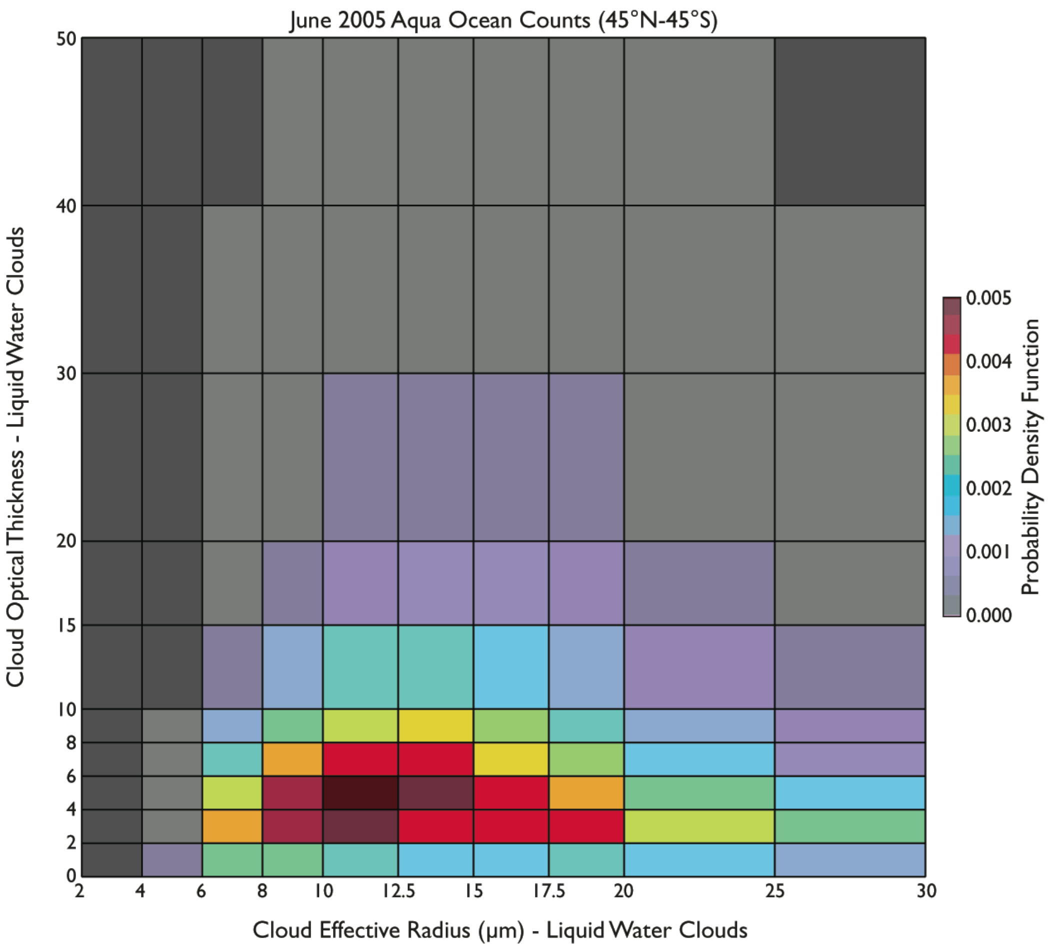


Figure 21. A June 2005 Normalized Probability Density Function (PDF) Joint Histogram of Cloud Optical Thickness (y-axis) vs. Cloud Effective Radius (x-axis) for Liquid Water Clouds displayed as a "colored histogram bin plot" with post-processing to limit the data to Ocean-only L3 grid cells that range from 45N to 45S. The top bin of Cloud Optical Thickness from 50 to 100 was removed. The bin colors represent the PDF in each bin (low probability is grey, high probability

is red).

For example, the peak PDF histogram bin value shown in Figure 21 is 0.005. If you take this number and multiply by the area of the bin ($2.5 \times 2.0 = 5.0$), one gets a bin (block) volume of 0.025. This means that 2.5% of the total volume under the surface of the entire plot fell in this bin (block). You can also interpret this as the probability of any one pixel falling in this peak joint histogram bin box as 2.5%. If you perform this calculation for every bin (block), and sum them, you end up with 1.0 or 100%.

Note that for the joint histogram in Figures 17 through 20, the top bin ($50 <$ Optical Thickness ≤ 100) was cut off in order to show (allow) more visual detail in the remaining bins. In order to get the proper summed value of 1.0 (100%), you need to include this deleted bin in the calculation.

11.0. References

Cho, H.-M., Z. Zhang, K. Meyer, M. Lebsock, S. Platnick, A. S. Ackerman, L. Di Girolamo, L. C.-Labonnote, C. Cornet, J. Riedi, and R. E. Holz (2015), Frequency and causes of failed MODIS cloud property retrievals for liquid phase clouds over global oceans. *J. Geophys. Res.*, 120, 4132-4154, doi:10.1002/2015JD023161.

Oreopoulos, L. (2005), The impact of subsampling on MODIS Level-3 statistics of cloud optical thickness and effective radius. *IEEE Trans. Geosci. Remote Sens.*, **43**, 366–373.

Oreopoulos, L., R. Cahalan, and S. Platnick (2007), The plane-parallel albedo bias of liquid clouds from MODIS observations. *J. Climate*, **20**, 5114–5125.

Pincus, R., S. Platnick, S. Ackerman, R. Hemler, and R. Hofmann (2012), Reconciling Simulated and Observed Views of Clouds: MODIS, ISCCP, and the Limits of Instrument Simulators *J. Climate*, **25**, 4699–4720.

Wielicki, B. A. and 49 Co-authors (2013), Achieving climate change absolute accuracy in orbit, *Bull. Amer. Met. Soc.*, 94(10), 1519–1539, doi:10.1175/BAMS-D-12-00149.1.

Zhang, Z., and S. Platnick (2011), An assessment of differences between cloud effective particle radius for marine water clouds from three MODIS spectral bands. *J. Geophys. Res.*, 116, D20215, doi:10.1029/2011JD016216.

Zhang, Z., A. S. Ackerman, G. Feingold, S. Platnick, R. Pincus, and H. Xue (2012), Effects of cloud horizontal inhomogeneity and drizzle on remote sensing of cloud droplet effective radius: Case studies based on large-eddy simulations. *J. Geophys. Res.,* 117, D19208, doi:10.1029/2012JD017655.

Appendix A:

Joint Histogram Statistics
&
Bin Boundaries

Appendix C:

Joint Histogram Statistics and Bin Boundaries

Identical in Daily & Monthly Products

The new 2022 Production Version of the CLDPROPCOSP files implemented several changes to the Joint Histogram bin boundaries. The updates are shown in the table below, in red. Note that the boundaries are identical between the regular cloudy parameters and the partly cloudy (denoted by a PCL in the parameter name) parameters.

Parameter	**Old Bin Boundaries (previous version)**	**New Bin Boundaries (2022 version)**
Cloud Optical Thickness Liquid	[0.0, 0.3, 1.3, 3.6, 9.4, 23.0, 60.0, 10000.0]	[0.0, 0.3, 1.3, 3.6, 9.4, 23.0, 60.0, 150.0]
Cloud Particle Size Liquid	[0, 8, 10, 13, 15, 20, 30]	[4, 8, 10, 12.5, 15, 20, 30]
Cloud Water Path Liquid	*Not Available*	[0, 10, 30, 60, 100, 150, 250, 20000]
Cloud Optical Thickness Ice	[0.0, 0.3, 1.3, 3.6, 9.4, 23.0, 60.0, 10000.0]	[0.0, 0.3, 1.3, 3.6, 9.4, 23.0, 60.0, 150.0]
Cloud Particle Size Ice	[0, 10, 20, 30, 40, 60, 90]	[5, 10, 20, 30, 40, 50, 60]
Cloud Water Path Ice	*Not Available*	[0, 20, 50, 100, 200, 400, 1000, 20000]
Cloud Optical Thickness Total	[0.0, 0.3, 1.3, 3.6, 9.4, 23.0, 60.0, 10000.0]	[0.0, 0.3, 1.3, 3.6, 9.4, 23.0, 60.0, 150.0]
Cloud Top Pressure	[0, 180, 310, 440, 560, 680, 800, 10000]	[0, 180, 310, 440, 560, 680, 800, 1100]

Units: Cloud Optical Thickness (none), Cloud Particle Size (microns), Cloud Water Path (g/m2), Cloud Top Pressure (hPa)

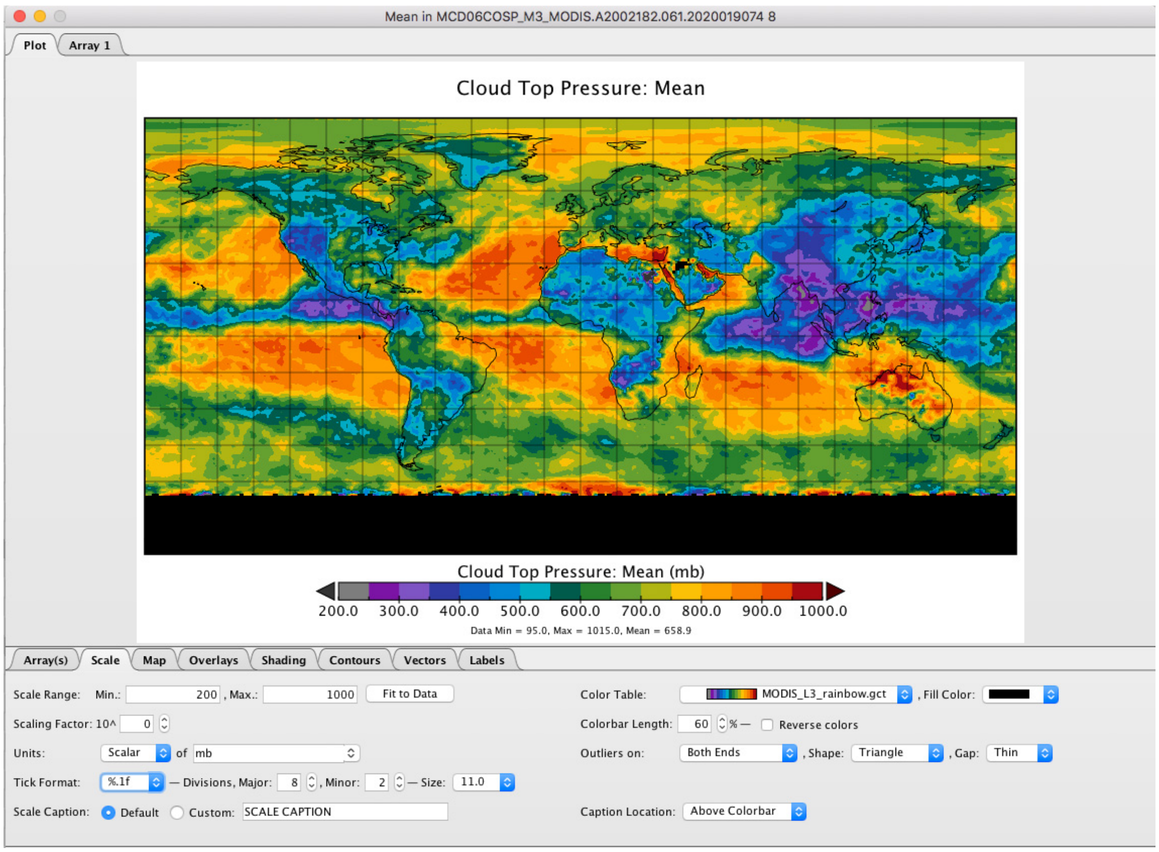
Important to note that some of the histogram bin boundaries changed between the original CLDPROPCOSP file and the new updated 2022 production version. New bin boundaries are shown in the far-right column. Any changes to the numbers are shown in red.

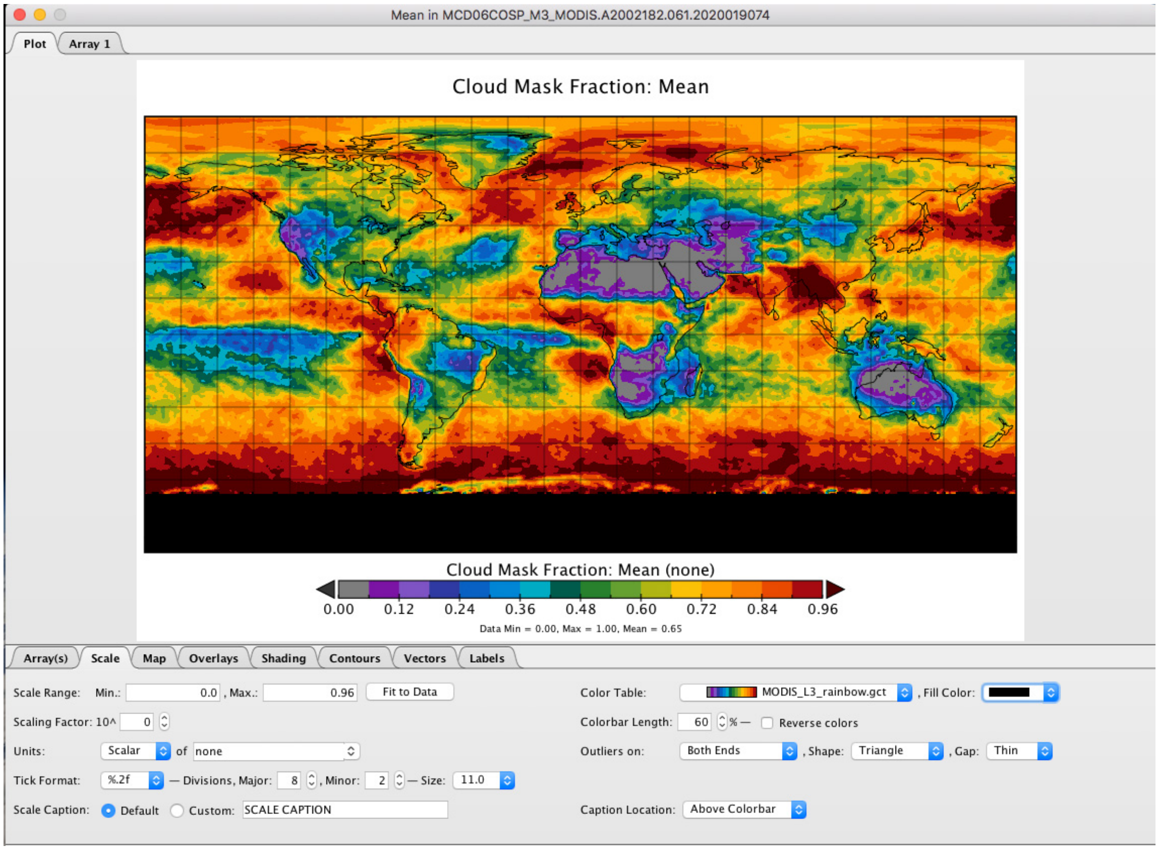
* Bin Boundary Handling: For L3 COSP Joint Histograms, the first through the penultimate histogram bin includes L2 pixel values equal to the lower bin boundary. The **last bin** includes L2 pixel values equal to both the lower and higher bin boundaries. This is slightly different from MODIS Standard MOD08, where the **first bin** included L2 pixel values equal to both the lower and higher bin boundaries.

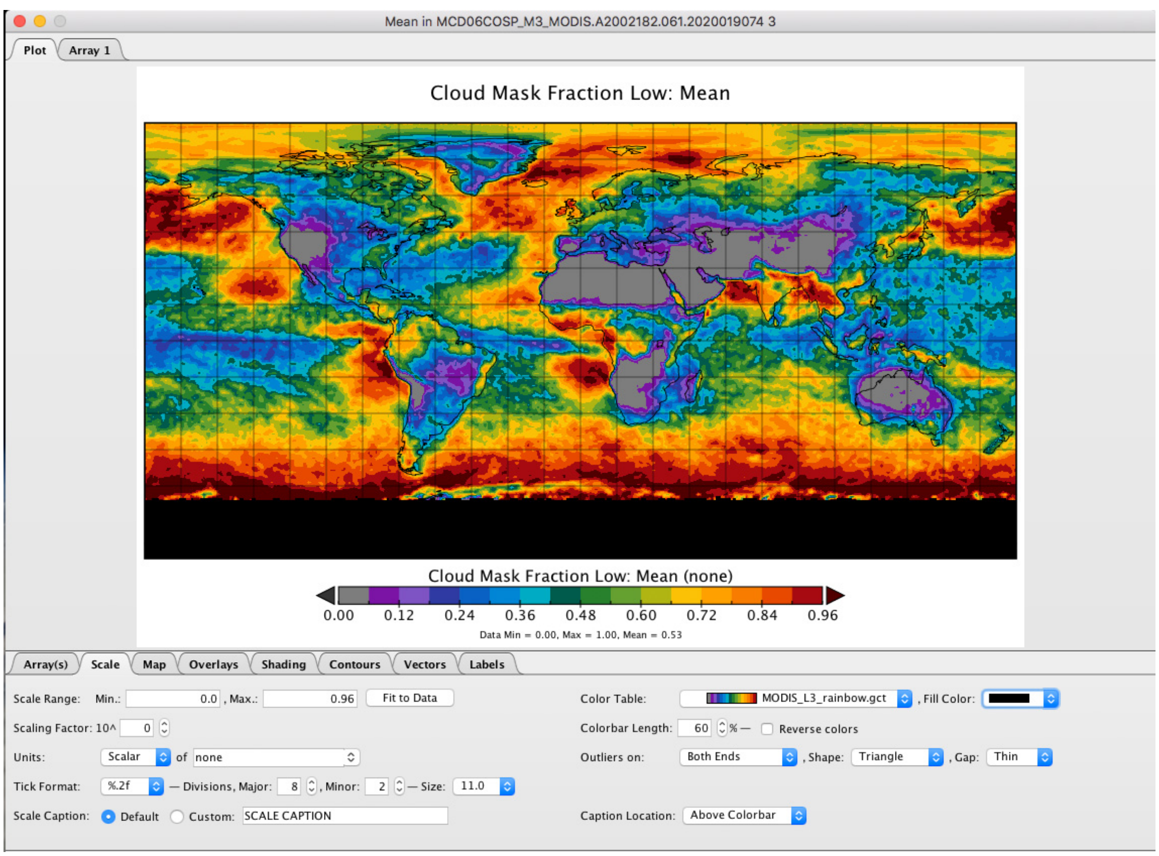
Appendix B:

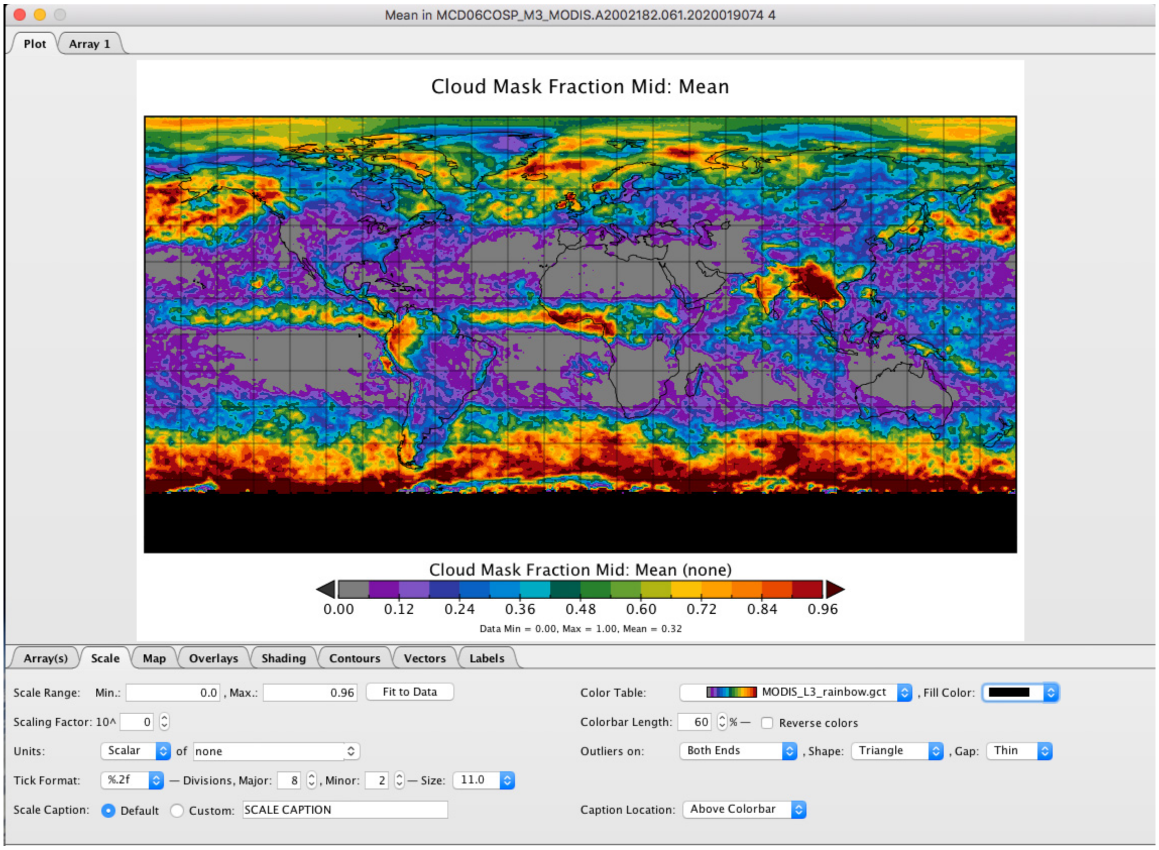
Mean Statistic Images from a Sample L3 COSP File

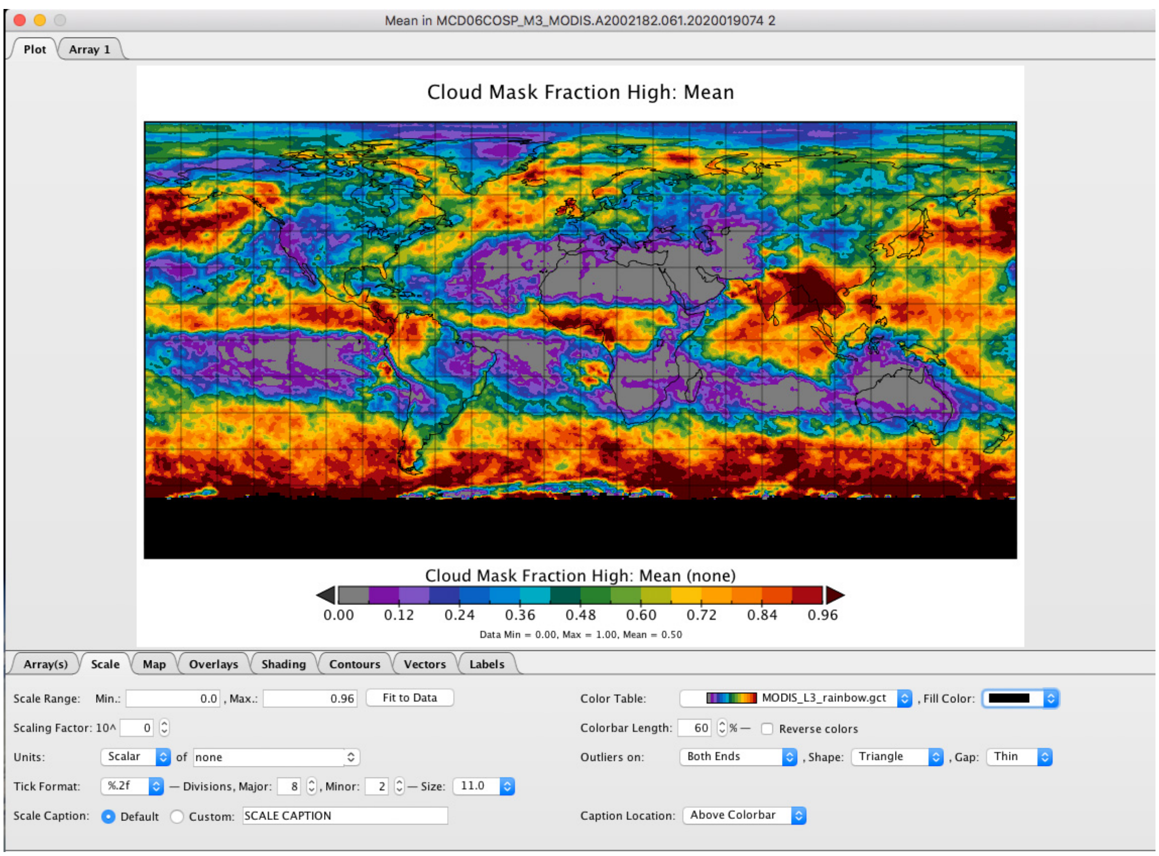
This Appendix shows mean statistic images for the full set of groups in a L3 monthly COSP file. The file being displayed is the L3 monthly COSP file for July 2002. Images were produced using the Panoply tool with a custom high contrast rainbow color bar. The color bar scale range was optimized for each image to show enhanced detail.

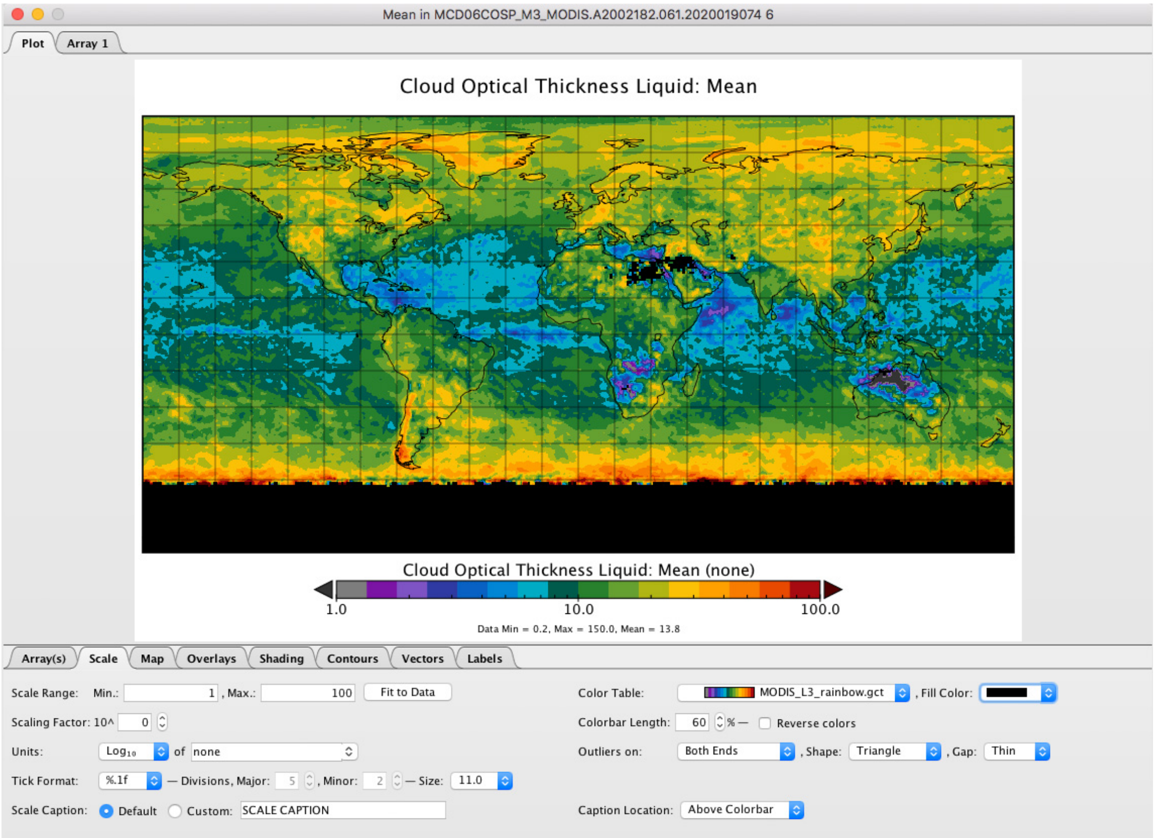


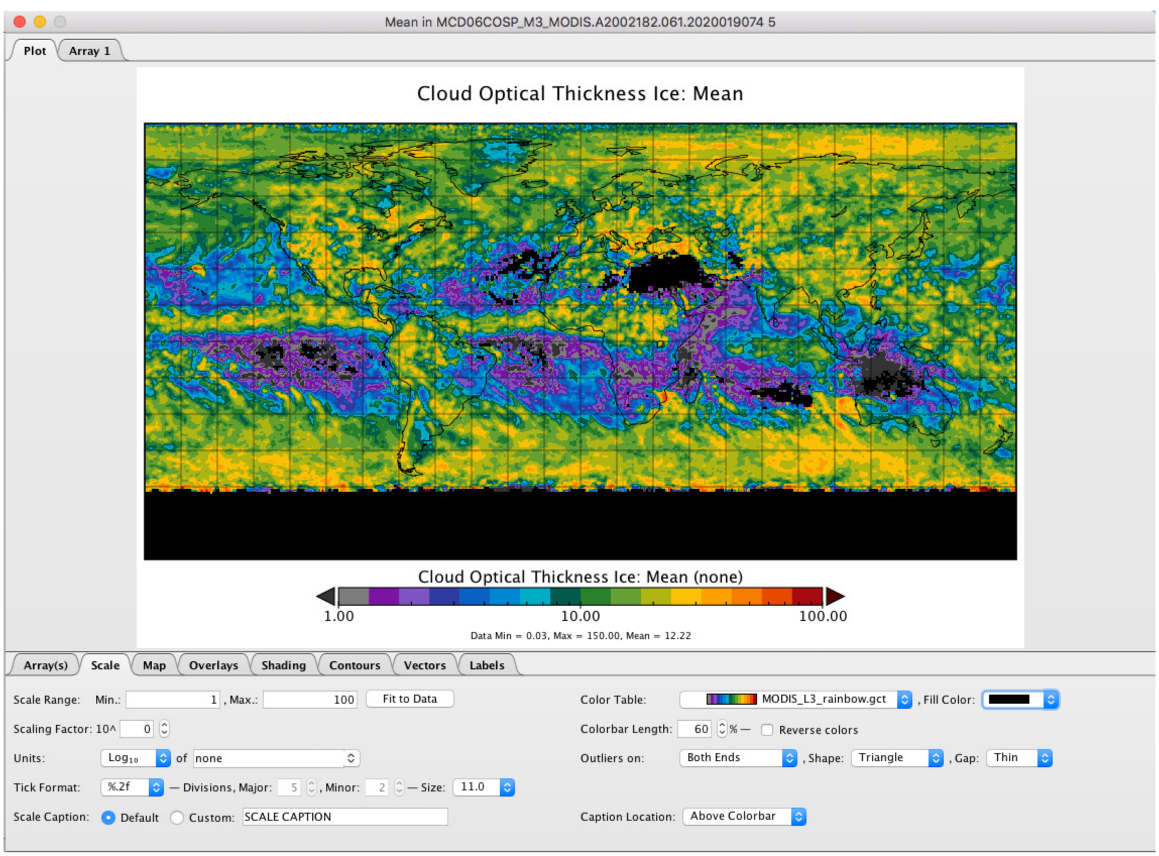

Mean in MCD06COSP_M3_MODIS.A2002182.061.2020019074
Plot
Array 1
Cloud Mask Fraction: Mean
Cloud Mask Fraction: Mean (none)
0.00 0.12 0.24 0.36 0.48 0.60 0.72 0.84 0.96
Data Min = 0.00, Max = 1.00, Mean = 0.65
Array(s) Scale Map Overlays Shading Contours Vectors Labels
Scale Range: Min.: 0.0 , Max.: 0.96 Fit to Data
Color Table: MODIS_L3_rainbow.gct , Fill Color:
Scaling Factor: 10^ 0
Colorbar Length: 60 % — Reverse colors
Units: Scalar of none
Outliers on: Both Ends , Shape: Triangle , Gap: Thin
Tick Format: %.2f — Divisions, Major: 8 , Minor: 2 — Size: 11.0
Scale Caption: Default Custom: SCALE CAPTION
Caption Location: Above Colorbar

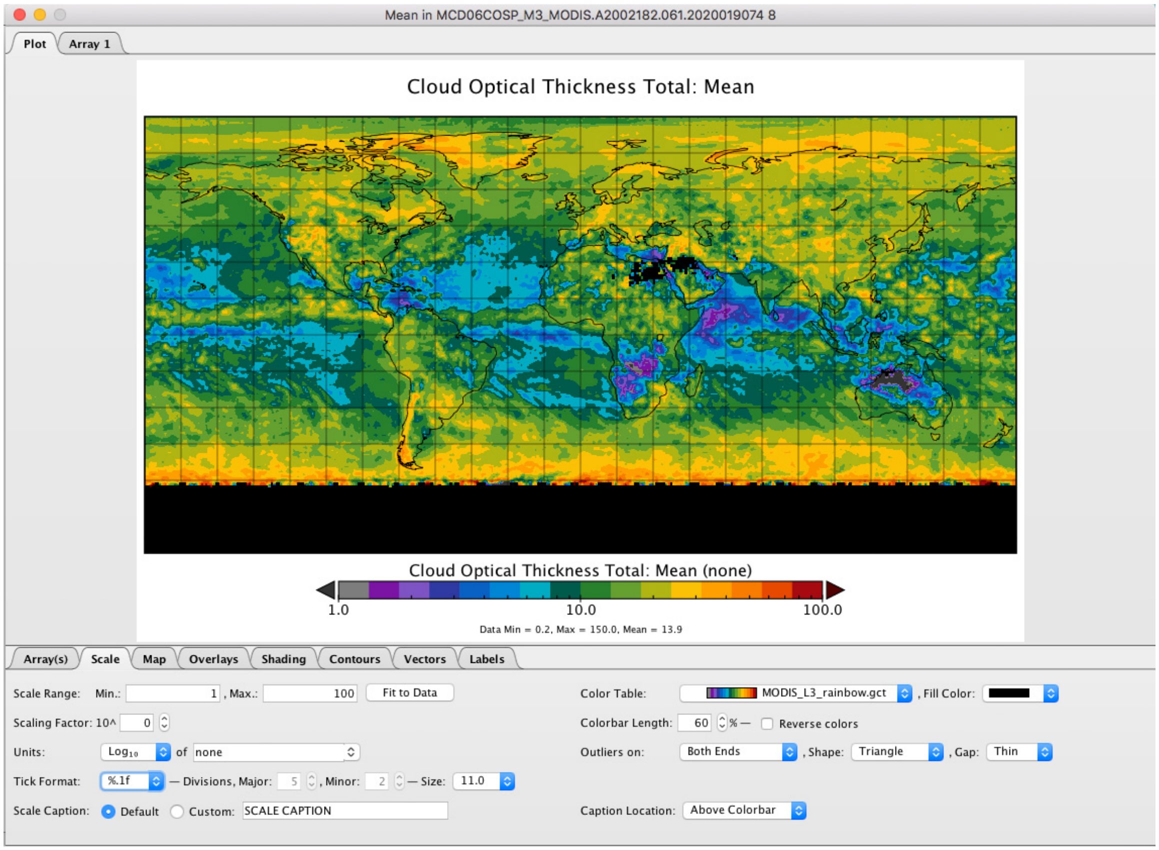

Mean in MCD06COSP_M3_MODIS.A2002182.061.2020019074 3
Plot
Array 1
Cloud Mask Fraction Low: Mean
Cloud Mask Fraction Low: Mean (none)
0.00 0.12 0.24 0.36 0.48 0.60 0.72 0.84 0.96
Data Min = 0.00, Max = 1.00, Mean = 0.53
Array(s) Scale Map Overlays Shading Contours Vectors Labels
Scale Range: Min.: 0.0 , Max.: 0.96 Fit to Data
Color Table: MODIS_L3_rainbow.gct , Fill Color:
Scaling Factor: 10^ 0
Colorbar Length: 60 % — Reverse colors
Units: Scalar of none
Outliers on: Both Ends , Shape: Triangle , Gap: Thin
Tick Format: %.2f — Divisions, Major: 8 , Minor: 2 — Size: 11.0
Scale Caption: Default Custom: SCALE CAPTION
Caption Location: Above Colorbar

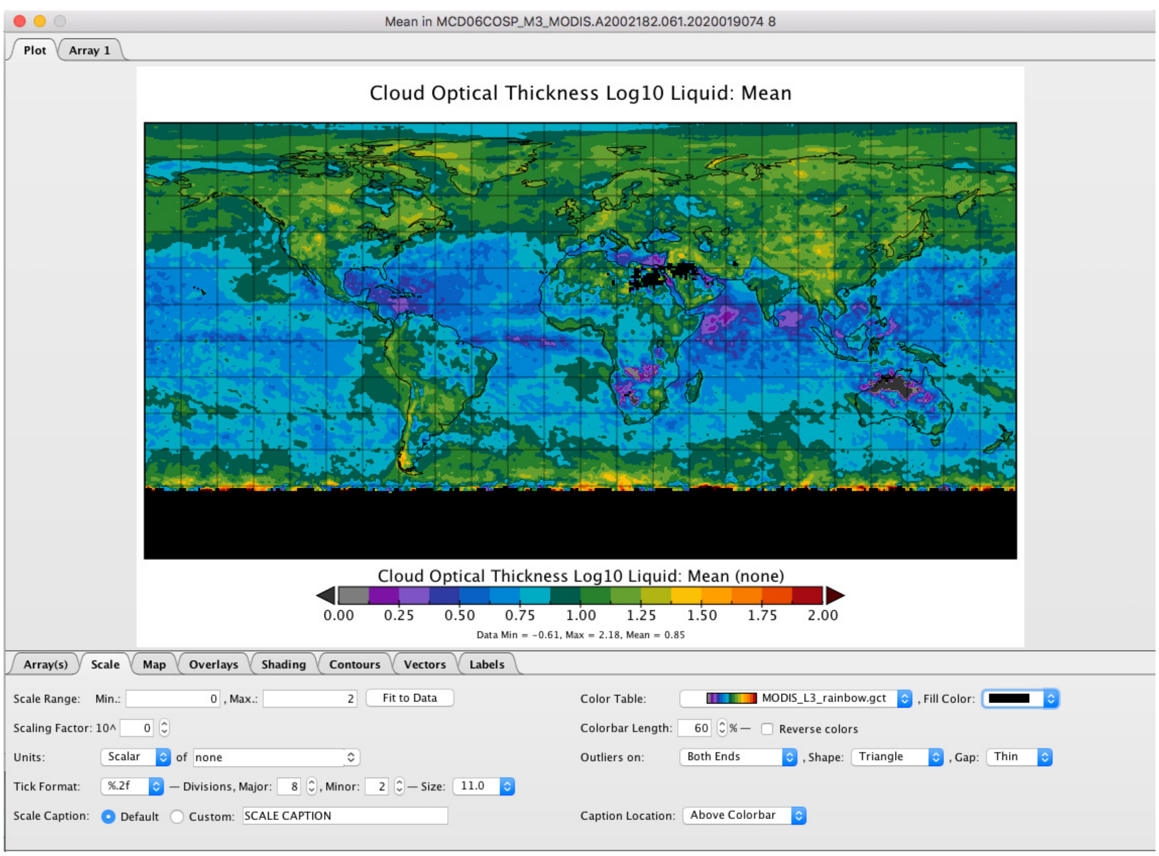

Mean in MCD06COSP_M3_MODIS.A2002182.061.2020019074 4
Plot
Array 1
Cloud Mask Fraction Mid: Mean
Cloud Mask Fraction Mid: Mean (none)
0.00 0.12 0.24 0.36 0.48 0.60 0.72 0.84 0.96
Data Min = 0.00, Max = 1.00, Mean = 0.32
Array(s) Scale Map Overlays Shading Contours Vectors Labels
Scale Range: Min.: 0.0 , Max.: 0.96 Fit to Data
Color Table: MODIS_L3_rainbow.gct , Fill Color:
Scaling Factor: 10^ 0
Colorbar Length: 60 % — Reverse colors
Units: Scalar of none
Outliers on: Both Ends , Shape: Triangle , Gap: Thin
Tick Format: %.2f — Divisions, Major: 8 , Minor: 2 — Size: 11.0
Scale Caption: Default Custom: SCALE CAPTION
Caption Location: Above Colorbar

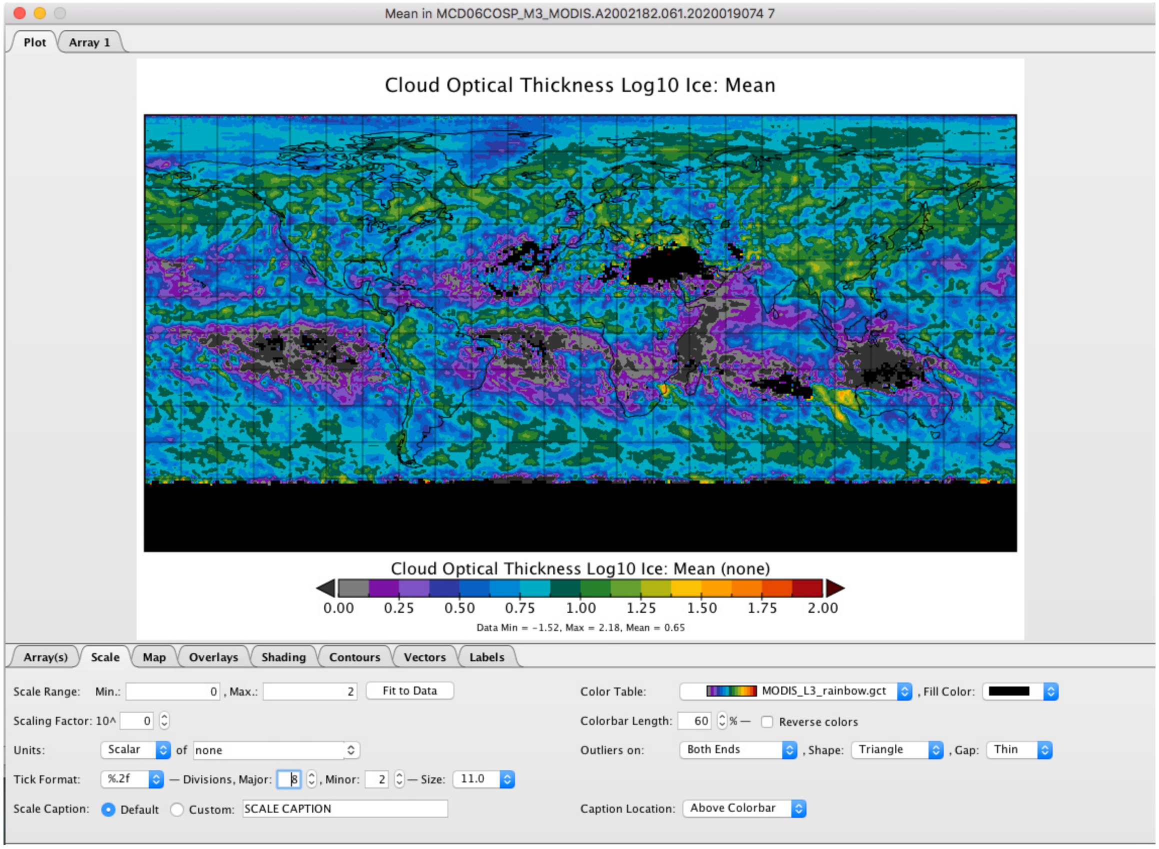

Mean in MCD06COSP_M3_MODIS.A2002182.061.2020019074 2
Plot
Array 1
Cloud Mask Fraction High: Mean
Cloud Mask Fraction High: Mean (none)
0.00 0.12 0.24 0.36 0.48 0.60 0.72 0.84 0.96
Data Min = 0.00, Max = 1.00, Mean = 0.50
Array(s) Scale Map Overlays Shading Contours Vectors Labels
Scale Range: Min.: 0.0 , Max.: 0.96 Fit to Data
Color Table: MODIS_L3_rainbow.gct , Fill Color:
Scaling Factor: 10^ 0
Colorbar Length: 60 % — Reverse colors
Units: Scalar of none
Outliers on: Both Ends , Shape: Triangle , Gap: Thin
Tick Format: %.2f — Divisions, Major: 8 , Minor: 2 — Size: 11.0
Scale Caption: Default Custom: SCALE CAPTION
Caption Location: Above Colorbar

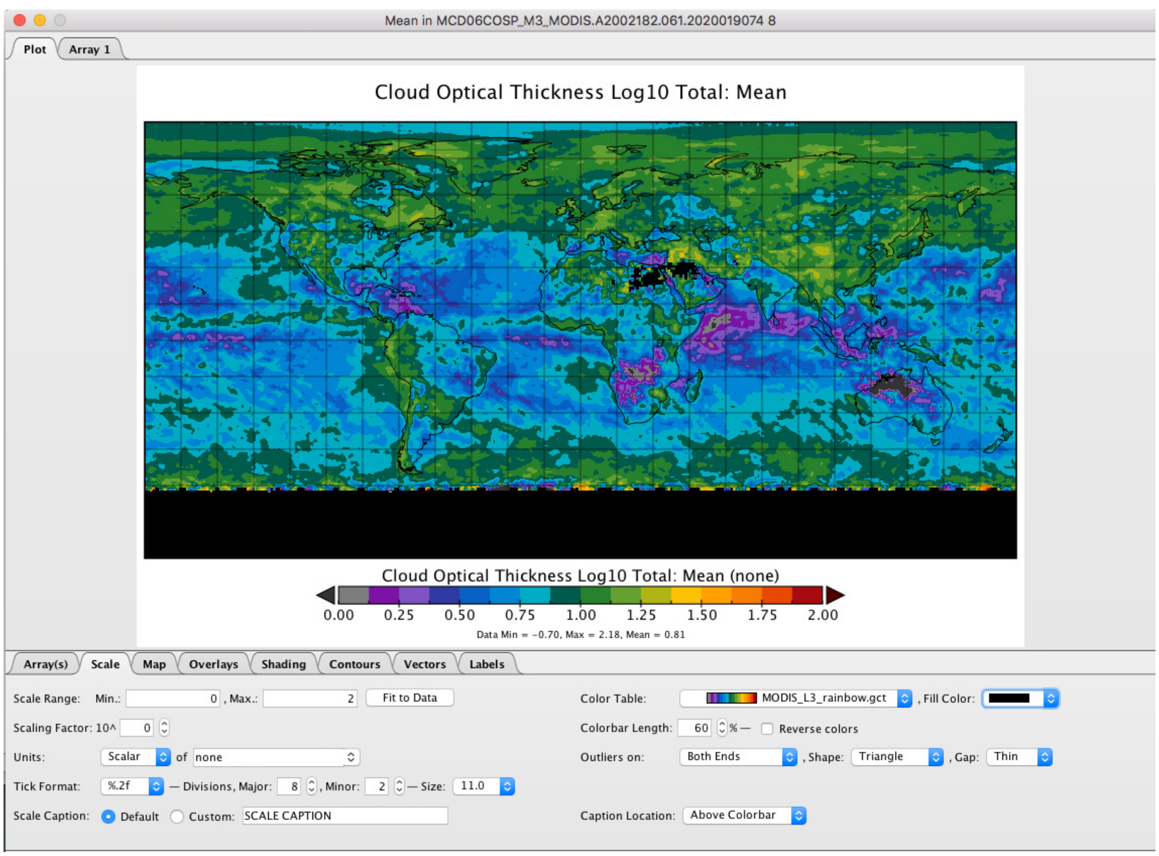

Mean in MCD06COSP_M3_MODIS.A2002182.061.2020019074 6
Plot
Array 1
Cloud Optical Thickness Liquid: Mean
Cloud Optical Thickness Liquid: Mean (none)
1.0
10.0
100.0
Data Min = 0.2, Max = 150.0, Mean = 13.8
Array(s)
Scale
Map
Overlays
Shading
Contours
Vectors
Labels
Scale Range: Min.: 1 , Max.: 100
Fit to Data
Color Table: MODIS_L3_rainbow.gct , Fill Color:
Scaling Factor: 10^ 0
Colorbar Length: 60 % —
Reverse colors
Units: Log10 of none
Outliers on: Both Ends , Shape: Triangle , Gap: Thin
Tick Format: %.1f — Divisions, Major: 5 , Minor: 2 — Size: 11.0
Scale Caption: Default Custom: SCALE CAPTION
Caption Location: Above Colorbar

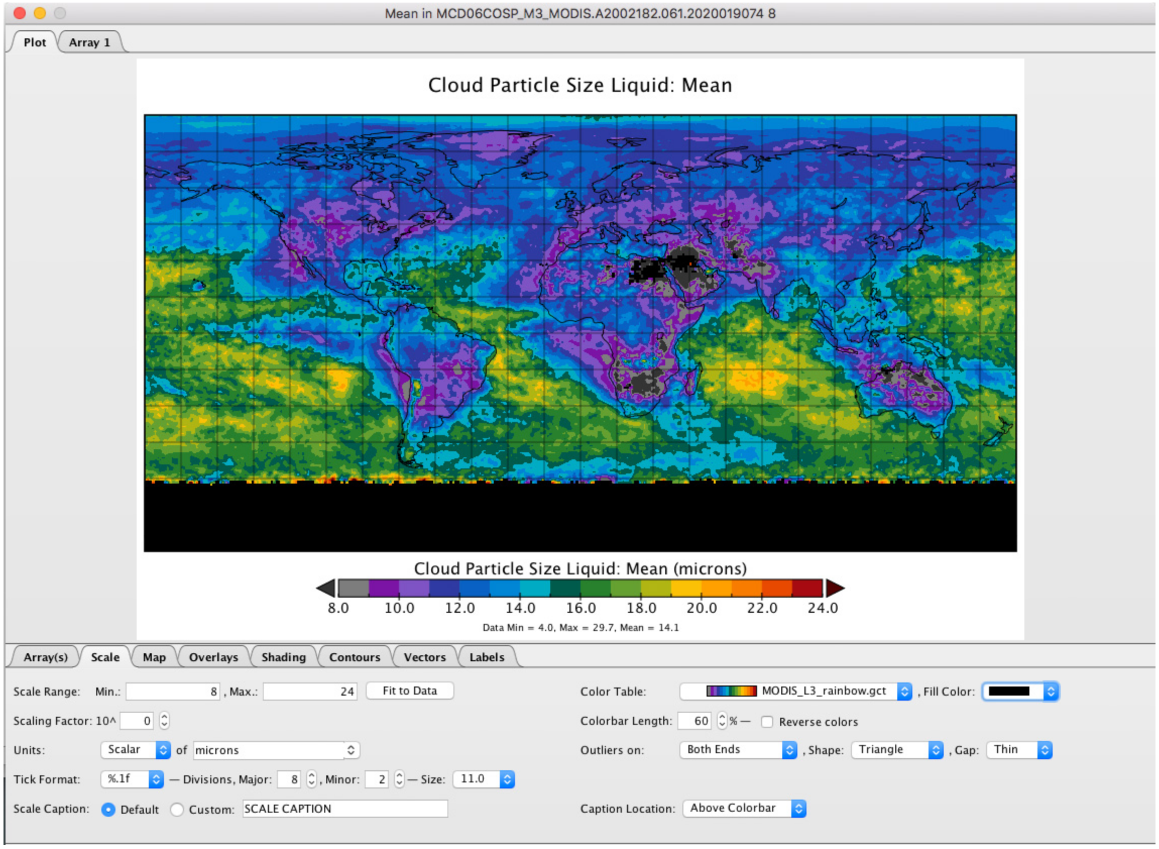

Mean in MCD06COSP_M3_MODIS.A2002182.061.2020019074 5
Plot
Array 1
Cloud Optical Thickness Ice: Mean
Cloud Optical Thickness Ice: Mean (none)
1.00
10.00
100.00
Data Min = 0.03, Max = 150.00, Mean = 12.22
Array(s)
Scale
Map
Overlays
Shading
Contours
Vectors
Labels
Scale Range: Min.: 1 , Max.: 100
Fit to Data
Color Table: MODIS_L3_rainbow.gct , Fill Color:
Scaling Factor: 10^ 0
Colorbar Length: 60 % —
Reverse colors
Units: Log10 of none
Outliers on: Both Ends , Shape: Triangle , Gap: Thin
Tick Format: %.2f — Divisions, Major: 5 , Minor: 2 — Size: 11.0
Scale Caption: Default Custom: SCALE CAPTION
Caption Location: Above Colorbar

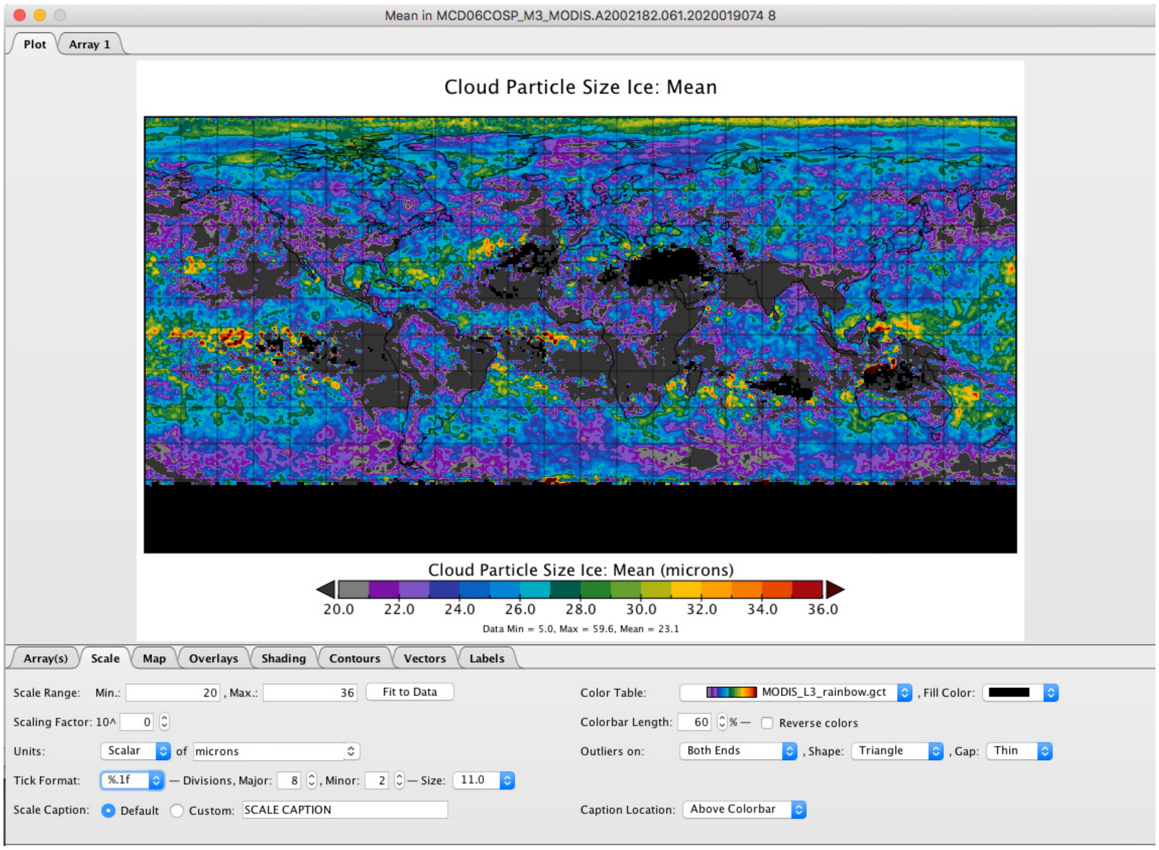

Mean in MCD06COSP_M3_MODIS.A2002182.061.2020019074 8
Plot
Array 1
Cloud Optical Thickness Total: Mean
Cloud Optical Thickness Total: Mean (none)
1.0
10.0
100.0
Data Min = 0.2, Max = 150.0, Mean = 13.9
Array(s)
Scale
Map
Overlays
Shading
Contours
Vectors
Labels
Scale Range: Min.: 1, Max.: 100
Fit to Data
Color Table: MODIS_L3_rainbow.gct, Fill Color:
Scaling Factor: 10^ 0
Colorbar Length: 60 % —
Reverse colors
Units: Log10 of none
Outliers on: Both Ends, Shape: Triangle, Gap: Thin
Tick Format: %.1f — Divisions, Major: 5, Minor: 2 — Size: 11.0
Scale Caption: Default Custom: SCALE CAPTION
Caption Location: Above Colorbar

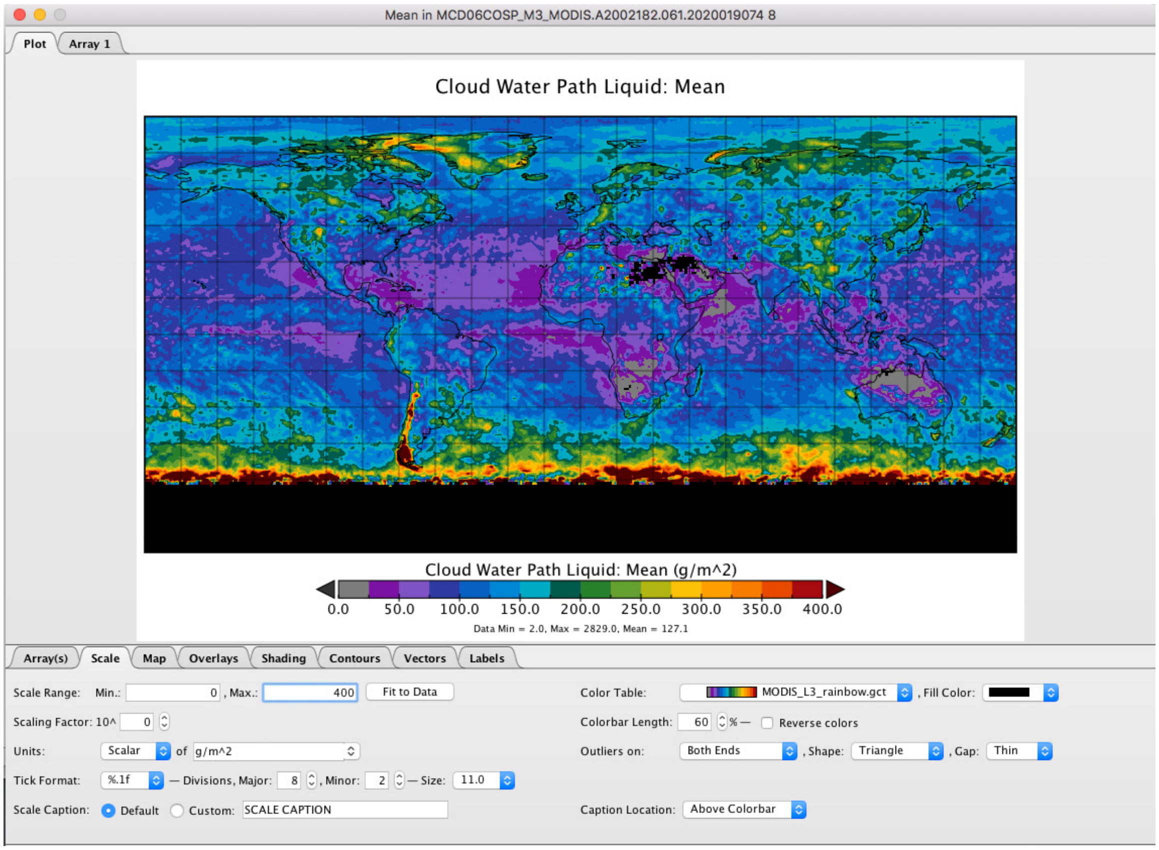

Mean in MCD06COSP_M3_MODIS.A2002182.061.2020019074 8
Plot
Array 1
Cloud Optical Thickness Log10 Liquid: Mean
Cloud Optical Thickness Log10 Liquid: Mean (none)
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
Data Min = −0.61, Max = 2.18, Mean = 0.85
Array(s)
Scale
Map
Overlays
Shading
Contours
Vectors
Labels
Scale Range: Min.: 0, Max.: 2
Fit to Data
Color Table: MODIS_L3_rainbow.gct, Fill Color:
Scaling Factor: 10^ 0
Colorbar Length: 60 % —
Reverse colors
Units: Scalar of none
Outliers on: Both Ends, Shape: Triangle, Gap: Thin
Tick Format: %.2f — Divisions, Major: 8, Minor: 2 — Size: 11.0
Scale Caption: Default Custom: SCALE CAPTION
Caption Location: Above Colorbar

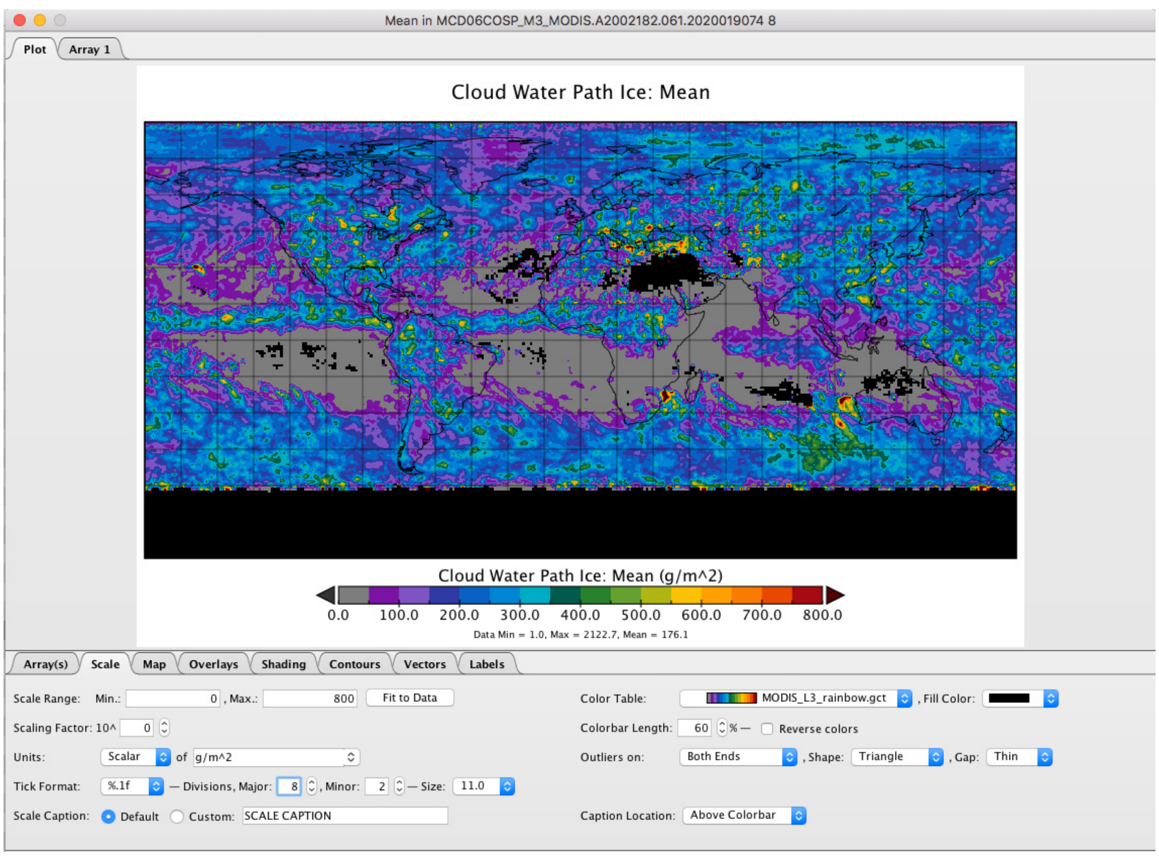

Mean in MCD06COSP_M3_MODIS.A2002182.061.2020019074 7
Plot
Array 1
Cloud Optical Thickness Log10 Ice: Mean
Cloud Optical Thickness Log10 Ice: Mean (none)
0.00 0.25 0.50 0.75 1.00 1.25 1.50 1.75 2.00
Data Min = -1.52, Max = 2.18, Mean = 0.65
Array(s) Scale Map Overlays Shading Contours Vectors Labels
Scale Range: Min.: 0 , Max.: 2 Fit to Data
Color Table: MODIS_L3_rainbow.gct , Fill Color:
Scaling Factor: 10^ 0
Colorbar Length: 60 % — Reverse colors
Units: Scalar of none
Outliers on: Both Ends , Shape: Triangle , Gap: Thin
Tick Format: %.2f — Divisions, Major: 8 , Minor: 2 — Size: 11.0
Scale Caption: Default Custom: SCALE CAPTION
Caption Location: Above Colorbar

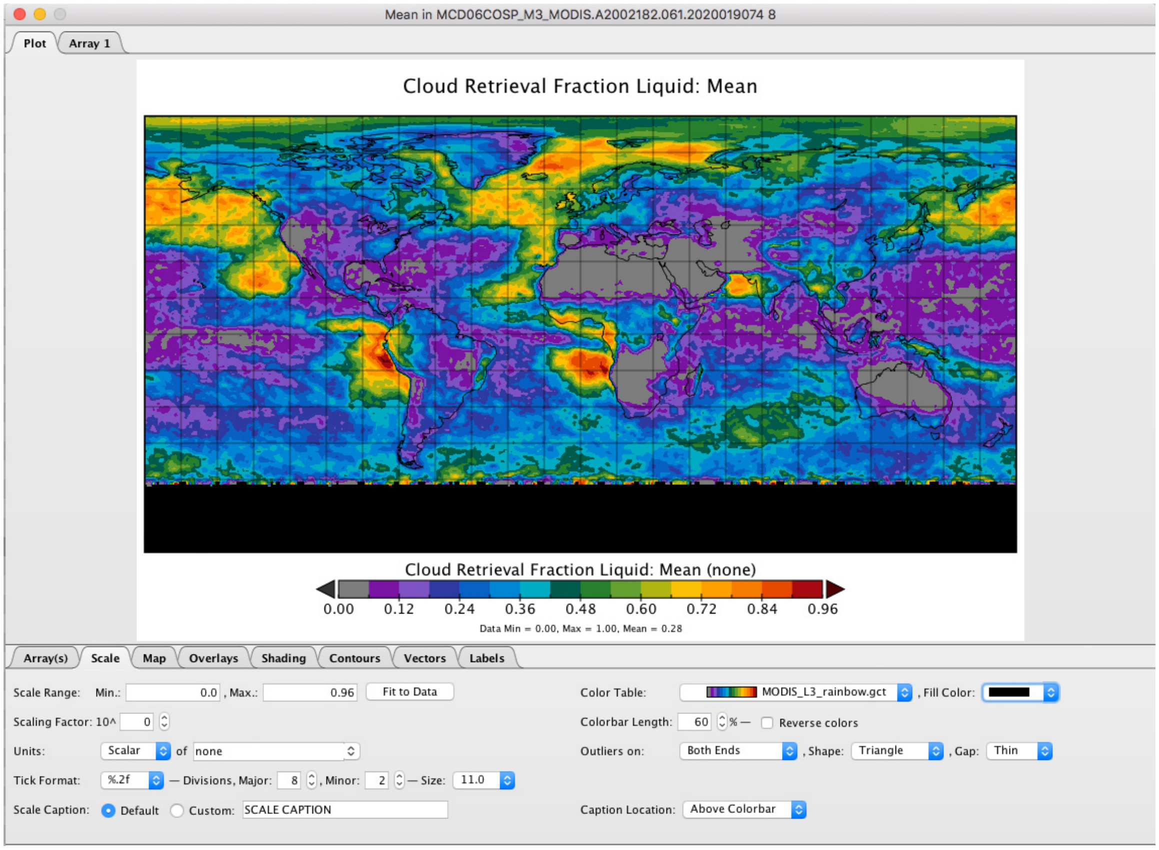

Mean in MCD06COSP_M3_MODIS.A2002182.061.2020019074 8
Plot
Array 1
Cloud Optical Thickness Log10 Total: Mean
Cloud Optical Thickness Log10 Total: Mean (none)
0.00 0.25 0.50 0.75 1.00 1.25 1.50 1.75 2.00
Data Min = -0.70, Max = 2.18, Mean = 0.81
Array(s) Scale Map Overlays Shading Contours Vectors Labels
Scale Range: Min.: 0 , Max.: 2 Fit to Data
Color Table: MODIS_L3_rainbow.gct , Fill Color:
Scaling Factor: 10^ 0
Colorbar Length: 60 % — Reverse colors
Units: Scalar of none
Outliers on: Both Ends , Shape: Triangle , Gap: Thin
Tick Format: %.2f — Divisions, Major: 8 , Minor: 2 — Size: 11.0
Scale Caption: Default Custom: SCALE CAPTION
Caption Location: Above Colorbar

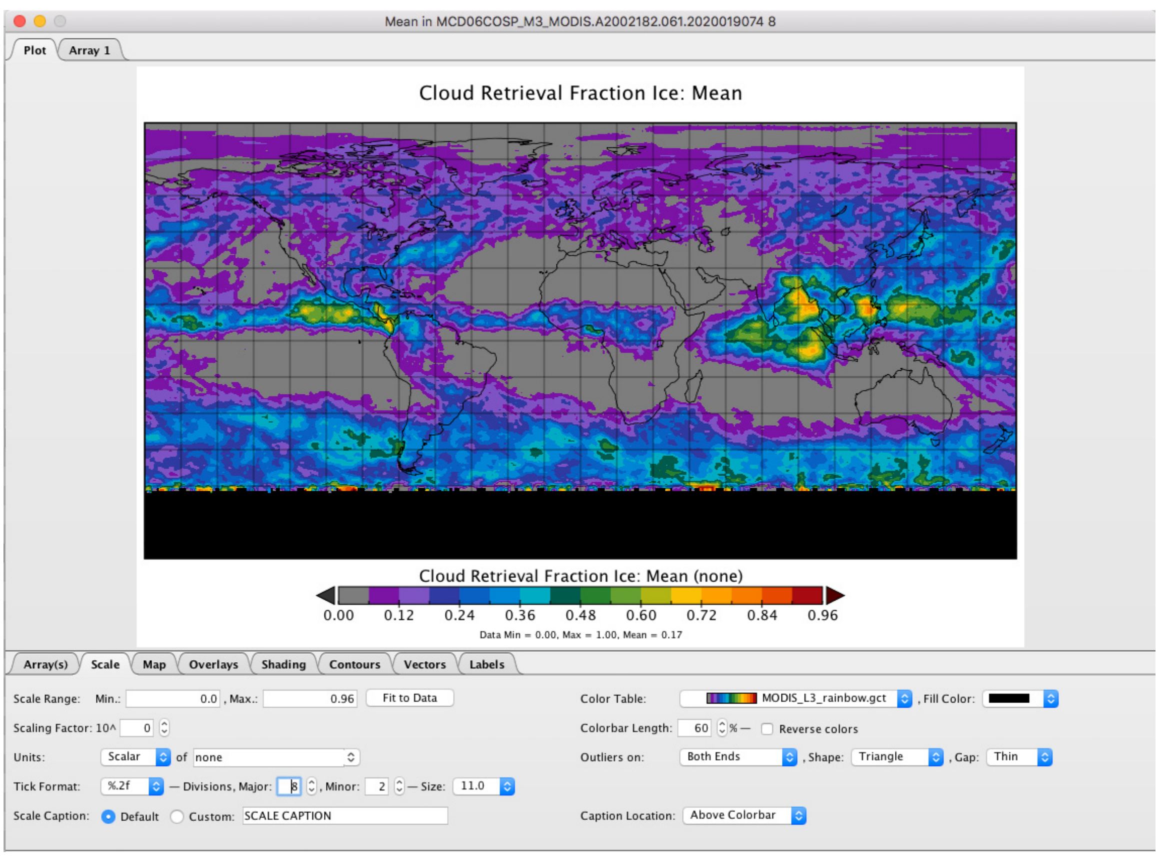

Mean in MCD06COSP_M3_MODIS.A2002182.061.2020019074 8
Plot
Array 1
Cloud Particle Size Liquid: Mean
Cloud Particle Size Liquid: Mean (microns)
8.0 10.0 12.0 14.0 16.0 18.0 20.0 22.0 24.0
Data Min = 4.0, Max = 29.7, Mean = 14.1
Array(s) Scale Map Overlays Shading Contours Vectors Labels
Scale Range: Min.: 8 , Max.: 24 Fit to Data
Color Table: MODIS_L3_rainbow.gct , Fill Color:
Scaling Factor: 10^ 0
Colorbar Length: 60 % — Reverse colors
Units: Scalar of microns
Outliers on: Both Ends , Shape: Triangle , Gap: Thin
Tick Format: %.1f — Divisions, Major: 8 , Minor: 2 — Size: 11.0
Scale Caption: Default Custom: SCALE CAPTION
Caption Location: Above Colorbar

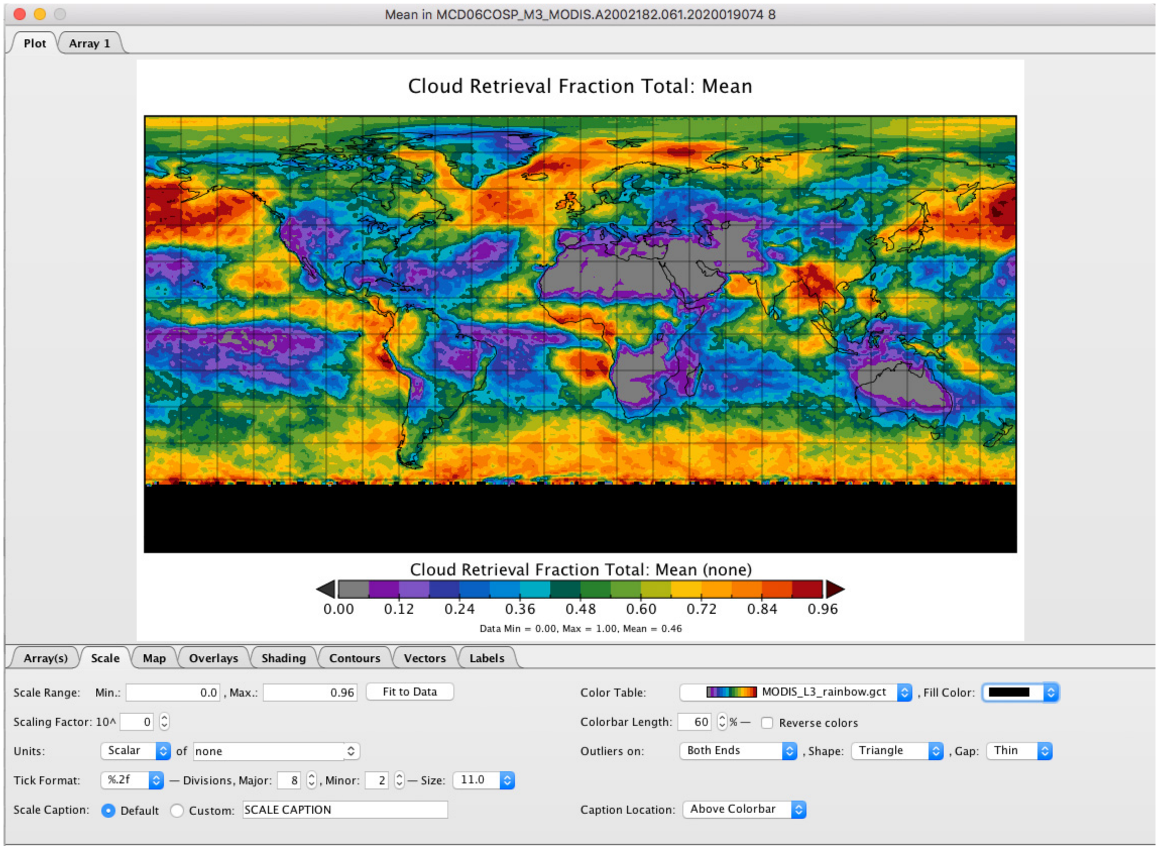

Mean in MCD06COSP_M3_MODIS.A2002182.061.2020019074 8
Plot
Array 1
Cloud Particle Size Ice: Mean
Cloud Particle Size Ice: Mean (microns)
20.0 22.0 24.0 26.0 28.0 30.0 32.0 34.0 36.0
Data Min = 5.0, Max = 59.6, Mean = 23.1
Array(s) Scale Map Overlays Shading Contours Vectors Labels
Scale Range: Min.: 20 , Max.: 36 Fit to Data
Color Table: MODIS_L3_rainbow.gct , Fill Color:
Scaling Factor: 10^ 0
Colorbar Length: 60 % — Reverse colors
Units: Scalar of microns
Outliers on: Both Ends , Shape: Triangle , Gap: Thin
Tick Format: %.1f — Divisions, Major: 8 , Minor: 2 — Size: 11.0
Scale Caption: Default Custom: SCALE CAPTION
Caption Location: Above Colorbar


Mean in MCD06COSP_M3_MODIS.A2002182.061.2020019074 8
Plot
Array 1
Cloud Water Path Liquid: Mean
Cloud Water Path Liquid: Mean (g/m^2)
0.0 50.0 100.0 150.0 200.0 250.0 300.0 350.0 400.0
Data Min = 2.0, Max = 2829.0, Mean = 127.1
Array(s) Scale Map Overlays Shading Contours Vectors Labels
Scale Range: Min.: 0, Max.: 400 Fit to Data
Scaling Factor: 10^ 0
Units: Scalar of g/m^2
Tick Format: %.1f — Divisions, Major: 8, Minor: 2 — Size: 11.0
Scale Caption: Default Custom: SCALE CAPTION
Color Table: MODIS_L3_rainbow.gct, Fill Color:
Colorbar Length: 60 % — Reverse colors
Outliers on: Both Ends, Shape: Triangle, Gap: Thin
Caption Location: Above Colorbar


Mean in MCD06COSP_M3_MODIS.A2002182.061.2020019074 8
Plot
Array 1
Cloud Water Path Ice: Mean
Cloud Water Path Ice: Mean (g/m^2)
0.0 100.0 200.0 300.0 400.0 500.0 600.0 700.0 800.0
Data Min = 1.0, Max = 2122.7, Mean = 176.1
Array(s) Scale Map Overlays Shading Contours Vectors Labels
Scale Range: Min.: 0, Max.: 800 Fit to Data
Scaling Factor: 10^ 0
Units: Scalar of g/m^2
Tick Format: %.1f — Divisions, Major: 8, Minor: 2 — Size: 11.0
Scale Caption: Default Custom: SCALE CAPTION
Color Table: MODIS_L3_rainbow.gct, Fill Color:
Colorbar Length: 60 % — Reverse colors
Outliers on: Both Ends, Shape: Triangle, Gap: Thin
Caption Location: Above Colorbar


Mean in MCD06COSP_M3_MODIS.A2002182.061.2020019074 8
Plot
Array 1
Cloud Retrieval Fraction Liquid: Mean
Cloud Retrieval Fraction Liquid: Mean (none)
0.00
0.12
0.24
0.36
0.48
0.60
0.72
0.84
0.96
Data Min = 0.00, Max = 1.00, Mean = 0.28
Array(s)
Scale
Map
Overlays
Shading
Contours
Vectors
Labels
Scale Range:
Min.:
0.0
, Max.:
0.96
Fit to Data
Color Table:
MODIS_L3_rainbow.gct
, Fill Color:
Scaling Factor: 10^
0
Colorbar Length:
60
%
Reverse colors
Units:
Scalar
of
none
Outliers on:
Both Ends
, Shape:
Triangle
, Gap:
Thin
Tick Format:
%.2f
Divisions, Major:
8
, Minor:
2
Size:
11.0
Scale Caption:
Default
Custom:
SCALE CAPTION
Caption Location:
Above Colorbar


Mean in MCD06COSP_M3_MODIS.A2002182.061.2020019074 8
Plot
Array 1
Cloud Retrieval Fraction Ice: Mean
Cloud Retrieval Fraction Ice: Mean (none)
0.00
0.12
0.24
0.36
0.48
0.60
0.72
0.84
0.96
Data Min = 0.00, Max = 1.00, Mean = 0.17
Array(s)
Scale
Map
Overlays
Shading
Contours
Vectors
Labels
Scale Range:
Min.:
0.0
, Max.:
0.96
Fit to Data
Color Table:
MODIS_L3_rainbow.gct
, Fill Color:
Scaling Factor: 10^
0
Colorbar Length:
60
%
Reverse colors
Units:
Scalar
of
none
Outliers on:
Both Ends
, Shape:
Triangle
, Gap:
Thin
Tick Format:
%.2f
Divisions, Major:
8
, Minor:
2
Size:
11.0
Scale Caption:
Default
Custom:
SCALE CAPTION
Caption Location:
Above Colorbar


Mean in MCD06COSP_M3_MODIS.A2002182.061.2020019074 8
Plot
Array 1
Cloud Retrieval Fraction Total: Mean
Cloud Retrieval Fraction Total: Mean (none)
0.00 0.12 0.24 0.36 0.48 0.60 0.72 0.84 0.96
Data Min = 0.00, Max = 1.00, Mean = 0.46
Array(s) Scale Map Overlays Shading Contours Vectors Labels
Scale Range: Min.: 0.0 , Max.: 0.96 Fit to Data
Color Table: MODIS_L3_rainbow.gct , Fill Color:
Scaling Factor: 10^ 0
Colorbar Length: 60 % — Reverse colors
Units: Scalar of none
Outliers on: Both Ends , Shape: Triangle , Gap: Thin
Tick Format: %.2f — Divisions, Major: 8 , Minor: 2 — Size: 11.0
Scale Caption: Default Custom: SCALE CAPTION
Caption Location: Above Colorbar

Appendix C:

CLDPROPCOSP File Spec

Header Dump, Global Attributes, and Parameter Specification in a L3 COSP File

This Appendix provides users an idea of the kind and scope of metadata information (Dimensions, Group variables and attributes, and Global attributes) available in the header of a L3 daily, CLDPROPCOSP file in NetCDF4 format.

```
netcdf CLDPROPCOSP_D3_MODIS_Aqua.A2023001.011.2024044184622.nc {

dimensions:
	latitude = 180;
	longitude = 360;
	jhisto_cloud_optical_thickness_liquid_7 = 7;
	jhisto_cloud_particle_size_liquid_6 = 6;
	jhisto_cloud_top_pressure_7 = 7;
	jhisto_cloud_optical_thickness_ice_7 = 7;
	jhisto_cloud_particle_size_ice_6 = 6;
	jhisto_cloud_optical_thickness_total_7 = 7;
	jhisto_cloud_optical_thickness_pcl_liquid_7 = 7;
	jhisto_cloud_particle_size_pcl_liquid_6 = 6;
	jhisto_cloud_optical_thickness_pcl_ice_7 = 7;
	jhisto_cloud_particle_size_pcl_ice_6 = 6;
	jhisto_cloud_optical_thickness_pcl_total_7 = 7;
	jhisto_cloud_water_path_liquid_7 = 7;
	jhisto_cloud_water_path_ice_7 = 7;
	jhisto_cloud_water_path_pcl_liquid_7 = 7;
	jhisto_cloud_water_path_pcl_ice_7 = 7;
variables:
	double latitude(latitude=180);
		_FillValue = -999.0; // double
		units = "degrees_north";
		_ChunkSizes = 180U; // uint
	double longitude(longitude=360);
		_FillValue = -999.0; // double
		units = "degrees_east";
		_ChunkSizes = 360U; // uint

group: Solar_Zenith {
variables:
	double Mean(longitude=360, latitude=180);
	_FillValue = -999.0; // double
	title = "Solar_Zenith: Mean";
	units = "degrees";
	_ChunkSizes = 360U, 180U; // uint

	double Sum(longitude=360, latitude=180);
	_FillValue = -999.0; // double
	title = "Solar_Zenith: Sum";
	_ChunkSizes = 360U, 180U; // uint

	double Sum_Squares(longitude=360, latitude=180);
	_FillValue = -999.0; // double
	title = "Solar_Zenith: Sum_Squares";
```

```
        _ChunkSizes = 360U, 180U; // uint

        int Pixel_Counts(longitude=360, latitude=180);
        _FillValue = -999; // int
        title = "Solar_Zenith: Pixel_Counts";
        _ChunkSizes = 360U, 180U; // uint

        double Standard_Deviation(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Solar_Zenith: Standard_Deviation";
        units = "degrees";
        _ChunkSizes = 360U, 180U; // uint

// group attributes:
        _FillValue = -999.0; // double
        valid_min = 0.0; // double
        valid_max = 180.0; // double
        scale_factor = 1.0; // double
        add_offset = 0.0; // double
        long_name = "Solar Zenith Angle (Cell to Sun) for Daytime Scenes";
        units = "degrees";
}

group: Solar_Azimuth {
variables:
        double Standard_Deviation(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Solar_Azimuth: Standard_Deviation";
        units = "degrees";
        _ChunkSizes = 360U, 180U; // uint

        double Mean(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Solar_Azimuth: Mean";
        units = "degrees";
        _ChunkSizes = 360U, 180U; // uint

        double Sum(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Solar_Azimuth: Sum";
        _ChunkSizes = 360U, 180U; // uint

        double Sum_Squares(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Solar_Azimuth: Sum_Squares";
        _ChunkSizes = 360U, 180U; // uint

        int Pixel_Counts(longitude=360, latitude=180);
        _FillValue = -999; // int
        title = "Solar_Azimuth: Pixel_Counts";
        _ChunkSizes = 360U, 180U; // uint

// group attributes:
```

```
        valid_max = 180.0; // double
        scale_factor = 1.0; // double
        add_offset = 0.0; // double
        _FillValue = -999.0; // double
        valid_min = -180.0; // double
        long_name = "Solar Azimuth Angle (Cell to Sun) for Daytime Scenes";
        units = "degrees";
}

group: Sensor_Zenith {
variables:
        double Mean(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Sensor_Zenith: Mean";
        units = "degrees";
        _ChunkSizes = 360U, 180U; // uint

        double Sum(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Sensor_Zenith: Sum";
        _ChunkSizes = 360U, 180U; // uint

        double Sum_Squares(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Sensor_Zenith: Sum_Squares";
        _ChunkSizes = 360U, 180U; // uint

        int Pixel_Counts(longitude=360, latitude=180);
        _FillValue = -999; // int
        title = "Sensor_Zenith: Pixel_Counts";
        _ChunkSizes = 360U, 180U; // uint

        double Standard_Deviation(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Sensor_Zenith: Standard_Deviation";
        units = "degrees";
        _ChunkSizes = 360U, 180U; // uint

// group attributes:
        long_name = "Sensor Zenith Angle (Cell to Sensor) for Daytime Scenes";
        units = "degrees";
        _FillValue = -999.0; // double
        valid_min = 0.0; // double
        valid_max = 180.0; // double
        scale_factor = 1.0; // double
        add_offset = 0.0; // double
}

group: Sensor_Azimuth {
variables:
        double Mean(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Sensor_Azimuth: Mean";
```

```
        units = "degrees";
        ChunkSizes = 360U, 180U; // uint

        double Standard_Deviation(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Sensor_Azimuth: Standard_Deviation";
        units = "degrees";
        _ChunkSizes = 360U, 180U; // uint

        double Sum(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Sensor_Azimuth: Sum";
        _ChunkSizes = 360U, 180U; // uint

        double Sum_Squares(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Sensor_Azimuth: Sum_Squares";
        _ChunkSizes = 360U, 180U; // uint

        int Pixel_Counts(longitude=360, latitude=180);
        _FillValue = -999; // int
        title = "Sensor_Azimuth: Pixel_Counts";
        _ChunkSizes = 360U, 180U; // uint

// group attributes:
        _FillValue = -999.0; // double
        valid_min = -180.0; // double
        valid_max = 180.0; // double
        scale_factor = 1.0; // double
        add_offset = 0.0; // double
        long_name = "Sensor Azimuth Angle (Cell to Sensor) for Daytime Scenes";
        units = "degrees";
}

group: Cloud_Top_Pressure {
variables:
        double Mean(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Top_Pressure: Mean";
        units = "mb";
        _ChunkSizes = 360U, 180U; // uint

        double Standard_Deviation(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Top_Pressure: Standard_Deviation";
        units = "mb";
        _ChunkSizes = 360U, 180U; // uint

        double Sum(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Top_Pressure: Sum";
        _ChunkSizes = 360U, 180U; // uint
```

```
        double Sum_Squares(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Top_Pressure: Sum_Squares";
        _ChunkSizes = 360U, 180U; // uint

        int Pixel_Counts(longitude=360, latitude=180);
        _FillValue = -999; // int
        title = "Cloud_Top_Pressure: Pixel_Counts";
        _ChunkSizes = 360U, 180U; // uint

// group attributes:
        scale_factor = 1.0; // double
        add_offset = 0.0; // double
        _FillValue = -999.0; // double
        valid_min = 1.0; // double
        valid_max = 1100.0; // double
        long_name = "Cloud Top Pressure for Daytime Scenes";
        units = "mb";
}

group: Cloud_Mask_Fraction {
variables:
        double Mean(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Mask_Fraction: Mean";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint

        double Standard_Deviation(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Mask_Fraction: Standard_Deviation";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint

        double Sum(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Mask_Fraction: Sum";
        _ChunkSizes = 360U, 180U; // uint

        double Sum_Squares(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Mask_Fraction: Sum_Squares";
        _ChunkSizes = 360U, 180U; // uint

        int Pixel_Counts(longitude=360, latitude=180);
        _FillValue = -999; // int
        title = "Cloud_Mask_Fraction: Pixel_Counts";
        _ChunkSizes = 360U, 180U; // uint

// group attributes:
        add_offset = 0.0; // double
        _FillValue = -999.0; // double
        valid_min = 0.0; // double
```

```
        valid_max = 1.0; // double
        scale_factor = 1.0; // double
        long_name = "Cloud Fraction from Cloud Mask for Daytime Scenes";
        units = "none";
}

group: Cloud_Mask_Fraction_Low {
variables:
        double Mean(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Mask_Fraction_Low: Mean";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint

        double Standard_Deviation(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Mask_Fraction_Low: Standard_Deviation";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint

        double Sum(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Mask_Fraction_Low: Sum";
        _ChunkSizes = 360U, 180U; // uint

        double Sum_Squares(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Mask_Fraction_Low: Sum_Squares";
        _ChunkSizes = 360U, 180U; // uint

        int Pixel_Counts(longitude=360, latitude=180);
        _FillValue = -999; // int
        title = "Cloud_Mask_Fraction_Low: Pixel_Counts";
        _ChunkSizes = 360U, 180U; // uint

// group attributes:
units = "none";
        _FillValue = -999.0; // double
        valid_min = 0.0; // double
        valid_max = 1.0; // double
        scale_factor = 1.0; // double
        add_offset = 0.0; // double
        long_name = "Cloud Fraction from Cloud Mask (Low Clouds, CTP GE 680 hPa) for Daytime
        Scenes";
}

group: Cloud_Mask_Fraction_Mid {
variables:
        double Standard_Deviation(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Mask_Fraction_Mid: Standard_Deviation";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint
```

```
        double Sum(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Mask_Fraction_Mid: Sum";
        _ChunkSizes = 360U, 180U; // uint

        int Pixel_Counts(longitude=360, latitude=180);
        _FillValue = -999; // int
        title = "Cloud_Mask_Fraction_Mid: Pixel_Counts";
        _ChunkSizes = 360U, 180U; // uint

        double Sum_Squares(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Mask_Fraction_Mid: Sum_Squares";
        _ChunkSizes = 360U, 180U; // uint

        double Mean(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Mask_Fraction_Mid: Mean";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint

// group attributes:
        long_name = "Cloud Fraction from Cloud Mask (Mid Clouds, CTP GE 440 hPa AND CTP LT
        680 hPa) for Daytime Scenes";
        _FillValue = -999.0; // double
        units = "none";
        valid_min = 0.0; // double
        valid_max = 1.0; // double
        scale_factor = 1.0; // double
        add_offset = 0.0; // double
}

group: Cloud_Mask_Fraction_High {
variables:
        double Mean(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Mask_Fraction_High: Mean";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint

        double Standard_Deviation(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Mask_Fraction_High: Standard_Deviation";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint

        double Sum(longitude=360, latitude=180);
        title = "Cloud_Mask_Fraction_High: Sum";
        _FillValue = -999.0; // double
        _ChunkSizes = 360U, 180U; // uint

        double Sum_Squares(longitude=360, latitude=180);
```

```
        _FillValue = -999.0; // double
        title = "Cloud_Mask_Fraction_High: Sum_Squares";
        _ChunkSizes = 360U, 180U; // uint

        int Pixel_Counts(longitude=360, latitude=180);
        _FillValue = -999; // int
        title = "Cloud_Mask_Fraction_High: Pixel_Counts";
        _ChunkSizes = 360U, 180U; // uint

// group attributes:
        scale_factor = 1.0; // double
        add_offset = 0.0; // double
        _FillValue = -999.0; // double
        valid_min = 0.0; // double
        valid_max = 1.0; // double
        units = "none";
        _name = "Cloud Fraction from Cloud Mask (High Clouds, CTP LT 440 hPa) for Daytime
        Scenes";
}

group: Cloud_Optical_Thickness_Liquid {
variables:
        double Mean(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_Liquid: Mean";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint

        double Sum(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_Liquid: Sum";
        _ChunkSizes = 360U, 180U; // uint

        int JHisto_vs_Cloud_Top_Pressure(longitude=360, latitude=180,
        jhisto_cloud_optical_thickness_liquid_7=7, jhisto_cloud_top_pressure_7=7);
        _FillValue = -999; // int
        title = "Cloud_Optical_Thickness_Liquid: JHisto_vs_Cloud_Top_Pressure";
        JHisto_Bin_Boundaries = 0.0, 0.3, 1.3, 3.6, 9.4, 23.0, 60.0, 150.0; // double
        JHisto_Bin_Boundaries_Joint_Parameter = 0.0, 180.0, 310.0, 440.0, 560.0, 680.0, 800.0,
        1100.0; // double
        _ChunkSizes = 360U, 180U, 7U, 7U; // uint

        int JHisto_vs_Cloud_Particle_Size_Liquid(longitude=360, latitude=180,
        jhisto_cloud_optical_thickness_liquid_7=7, jhisto_cloud_particle_size_liquid_6=6);
        _FillValue = -999; // int
        title = "Cloud_Optical_Thickness_Liquid: JHisto_vs_Cloud_Particle_Size_Liquid";
        JHisto_Bin_Boundaries = 0.0, 0.3, 1.3, 3.6, 9.4, 23.0, 60.0, 150.0; // double
        JHisto_Bin_Boundaries_Joint_Parameter = 4.0, 8.0, 10.0, 12.5, 15.0, 20.0, 30.0; // double
        _ChunkSizes = 360U, 180U, 7U, 6U; // uint

        double Sum_Squares(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_Liquid: Sum_Squares";
```

```
        _ChunkSizes = 360U, 180U; // uint

        int Pixel_Counts(longitude=360, latitude=180);
        _FillValue = -999; // int
        title = "Cloud_Optical_Thickness_Liquid: Pixel_Counts";
        _ChunkSizes = 360U, 180U; // uint

        double Standard_Deviation(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_Liquid: Standard_Deviation";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint

// group attributes:
        add_offset = 0.0; // double
        units = "none";
        _FillValue = -999.0; // double
        valid_min = 0.0; // double
        valid_max = 150.0; // double
        scale_factor = 1.0; // double
        long_name = "Cloud Optical Thickness for Liquid Water Clouds (3.7-micron Retrieval for
        Cloudy Scenes)";
}

group: Cloud_Optical_Thickness_Ice {
variables:
        double Standard_Deviation(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_Ice: Standard_Deviation";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint

        double Mean(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_Ice: Mean";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint

        double Sum(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_Ice: Sum";
        _ChunkSizes = 360U, 180U; // uint

        int JHisto_vs_Cloud_Top_Pressure(longitude=360, latitude=180,
        jhisto_cloud_optical_thickness_ice_7=7, jhisto_cloud_top_pressure_7=7);
        _FillValue = -999; // int
        title = "Cloud_Optical_Thickness_Ice: JHisto_vs_Cloud_Top_Pressure";
        JHisto_Bin_Boundaries = 0.0, 0.3, 1.3, 3.6, 9.4, 23.0, 60.0, 150.0; // double
        JHisto_Bin_Boundaries_Joint_Parameter = 0.0, 180.0, 310.0, 440.0, 560.0, 680.0, 800.0,
        1100.0; // double
        _ChunkSizes = 360U, 180U, 7U, 7U; // uint
```

```
        int JHisto_vs_Cloud_Particle_Size_Ice(longitude=360, latitude=180,
        jhisto_cloud_optical_thickness_ice_7=7, jhisto_cloud_particle_size_ice_6=6);
        _FillValue = -999; // int
        title = "Cloud_Optical_Thickness_Ice: JHisto_vs_Cloud_Particle_Size_Ice";
        JHisto_Bin_Boundaries = 0.0, 0.3, 1.3, 3.6, 9.4, 23.0, 60.0, 150.0; // double
        JHisto_Bin_Boundaries_Joint_Parameter = 5.0, 10.0, 20.0, 30.0, 40.0, 50.0, 60.0; // double
        _ChunkSizes = 360U, 180U, 7U, 6U; // uint

        int Pixel_Counts(longitude=360, latitude=180);
        _FillValue = -999; // int
        title = "Cloud_Optical_Thickness_Ice: Pixel_Counts";
        _ChunkSizes = 360U, 180U; // uint

        double Sum_Squares(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_Ice: Sum_Squares";
        _ChunkSizes = 360U, 180U; // uint

// group attributes:
        valid_min = 0.0; // double
        valid_max = 150.0; // double
        scale_factor = 1.0; // double
        add_offset = 0.0; // double
        _FillValue = -999.0; // double
        units = "none";
        long_name = "Cloud Optical Thickness for Ice Clouds (3.7 micron Retrieval for Cloudy
Scenes)";
}

group: Cloud_Optical_Thickness_Total {
variables:
        double Standard_Deviation(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_Total: Standard_Deviation";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint

        double Sum(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_Total: Sum";
        _ChunkSizes = 360U, 180U; // uint

        int Pixel_Counts(longitude=360, latitude=180);
        _FillValue = -999; // int
        title = "Cloud_Optical_Thickness_Total: Pixel_Counts";
        _ChunkSizes = 360U, 180U; // uint

        double Sum_Squares(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_Total: Sum_Squares";
        _ChunkSizes = 360U, 180U; // uint
```

```
        int JHisto_vs_Cloud_Top_Pressure(longitude=360, latitude=180,
        jhisto_cloud_optical_thickness_total_7=7, jhisto_cloud_top_pressure_7=7);
        _FillValue = -999; // int
        title = "Cloud_Optical_Thickness_Total: JHisto_vs_Cloud_Top_Pressure";
        JHisto_Bin_Boundaries = 0.0, 0.3, 1.3, 3.6, 9.4, 23.0, 60.0, 150.0; // double
        JHisto_Bin_Boundaries_Joint_Parameter = 0.0, 180.0, 310.0, 440.0, 560.0, 680.0, 800.0,
        1100.0; // double
        _ChunkSizes = 360U, 180U, 7U, 7U; // uint

        double Mean(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_Total: Mean";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint

// group attributes:
        long_name = "Cloud Optical Thickness for Combined (LiquidWater+Ice+Undetermined)
        Phase Clouds (3.7-micron Retrieval for Cloudy Scenes)";
        units = "none";
        _FillValue = -999.0; // double
        valid_min = 0.0; // double
        valid_max = 150.0; // double
        scale_factor = 1.0; // double
        add_offset = 0.0; // double
}

group: Cloud_Optical_Thickness_PCL_Liquid {
variables:
        double Standard_Deviation(longitude=360, latitude=180);
        title = "Cloud_Optical_Thickness_PCL_Liquid: Standard_Deviation";
        units = "none";
        _FillValue = -999.0; // double
        _ChunkSizes = 360U, 180U; // uint

        double Sum(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_PCL_Liquid: Sum";
        _ChunkSizes = 360U, 180U; // uint

        int JHisto_vs_Cloud_Top_Pressure(longitude=360, latitude=180,
        jhisto_cloud_optical_thickness_pcl_liquid_7=7, jhisto_cloud_top_pressure_7=7);
        _FillValue = -999; // int
        title = "Cloud_Optical_Thickness_PCL_Liquid: JHisto_vs_Cloud_Top_Pressure";
        JHisto_Bin_Boundaries = 0.0, 0.3, 1.3, 3.6, 9.4, 23.0, 60.0, 150.0; // double
        JHisto_Bin_Boundaries_Joint_Parameter = 0.0, 180.0, 310.0, 440.0, 560.0, 680.0, 800.0,
        1100.0; // double
        _ChunkSizes = 360U, 180U, 7U, 7U; // uint

        int JHisto_vs_Cloud_Particle_Size_PCL_Liquid(longitude=360, latitude=180,
        jhisto_cloud_optical_thickness_pcl_liquid_7=7, jhisto_cloud_particle_size_pcl_liquid_6=6);
        _FillValue = -999; // int
        title = "Cloud_Optical_Thickness_PCL_Liquid: JHisto_vs_Cloud_Particle_Size_PCL_Liquid";
        JHisto_Bin_Boundaries = 0.0, 0.3, 1.3, 3.6, 9.4, 23.0, 60.0, 150.0; // double
```

```
        JHisto_Bin_Boundaries_Joint_Parameter = 4.0, 8.0, 10.0, 12.5, 15.0, 20.0, 30.0; // double
        _ChunkSizes = 360U, 180U, 7U, 6U; // uint

        double Sum_Squares(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_PCL_Liquid: Sum_Squares";
        _ChunkSizes = 360U, 180U; // uint

        int Pixel_Counts(longitude=360, latitude=180);
        _FillValue = -999; // int
        title = "Cloud_Optical_Thickness_PCL_Liquid: Pixel_Counts";
        _ChunkSizes = 360U, 180U; // uint

        double Mean(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_PCL_Liquid: Mean";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint

// group attributes:
        long_name = "Cloud Optical Thickness for Liquid Water Phase Clouds (3.7 micron Retrieval
        for Partly Cloudy (PCL) Scenes)";
        units = "none";
        _FillValue = -999.0; // double
        valid_min = 0.0; // double
        valid_max = 150.0; // double
        scale_factor = 1.0; // double
        add_offset = 0.0; // double
}

group: Cloud_Optical_Thickness_PCL_Ice {
variables:
        double Standard_Deviation(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_PCL_Ice: Standard_Deviation";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint

        double Sum(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_PCL_Ice: Sum";
        _ChunkSizes = 360U, 180U; // uint

        int JHisto_vs_Cloud_Top_Pressure(longitude=360, latitude=180,
        jhisto_cloud_optical_thickness_pcl_ice_7=7, jhisto_cloud_top_pressure_7=7);
        _FillValue = -999; // int
        title = "Cloud_Optical_Thickness_PCL_Ice: JHisto_vs_Cloud_Top_Pressure";
        JHisto_Bin_Boundaries = 0.0, 0.3, 1.3, 3.6, 9.4, 23.0, 60.0, 150.0; // double
        JHisto_Bin_Boundaries_Joint_Parameter = 0.0, 180.0, 310.0, 440.0, 560.0, 680.0, 800.0,
        1100.0; // double
        _ChunkSizes = 360U, 180U, 7U, 7U; // uint
```

```
        int JHisto_vs_Cloud_Particle_Size_PCL_Ice(longitude=360, latitude=180,
        jhisto_cloud_optical_thickness_pcl_ice_7=7, jhisto_cloud_particle_size_pcl_ice_6=6);
        _FillValue = -999; // int
        title = "Cloud_Optical_Thickness_PCL_Ice: JHisto_vs_Cloud_Particle_Size_PCL_Ice";
        JHisto_Bin_Boundaries = 0.0, 0.3, 1.3, 3.6, 9.4, 23.0, 60.0, 150.0; // double
        JHisto_Bin_Boundaries_Joint_Parameter = 5.0, 10.0, 20.0, 30.0, 40.0, 50.0, 60.0; // double
        _ChunkSizes = 360U, 180U, 7U, 6U; // uint

        double Sum_Squares(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_PCL_Ice: Sum_Squares";
        _ChunkSizes = 360U, 180U; // uint

        int Pixel_Counts(longitude=360, latitude=180);
        _FillValue = -999; // int
        title = "Cloud_Optical_Thickness_PCL_Ice: Pixel_Counts";
        _ChunkSizes = 360U, 180U; // uint

        double Mean(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_PCL_Ice: Mean";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint

// group attributes:
        long_name = "Cloud Optical Thickness for Ice Phase Clouds (3.7 micron Retrieval for Partly
        Cloudy (PCL) Scenes)";
        units = "none";
        _FillValue = -999.0; // double
        valid_min = 0.0; // double
        valid_max = 150.0; // double
        scale_factor = 1.0; // double
        add_offset = 0.0; // double
}

group: Cloud_Optical_Thickness_PCL_Total {
variables:
        double Mean(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_PCL_Total: Mean";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint

        double Sum(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_PCL_Total: Sum";
        _ChunkSizes = 360U, 180U; // uint

        int Pixel_Counts(longitude=360, latitude=180);
        _FillValue = -999; // int
        title = "Cloud_Optical_Thickness_PCL_Total: Pixel_Counts";
        _ChunkSizes = 360U, 180U; // uint
```

```
        double Sum_Squares(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_PCL_Total: Sum_Squares";
        _ChunkSizes = 360U, 180U; // uint

        int JHisto_vs_Cloud_Top_Pressure(longitude=360, latitude=180,
        jhisto_cloud_optical_thickness_pcl_total_7=7, jhisto_cloud_top_pressure_7=7);
        _FillValue = -999; // int
        title = "Cloud_Optical_Thickness_PCL_Total: JHisto_vs_Cloud_Top_Pressure";
        JHisto_Bin_Boundaries = 0.0, 0.3, 1.3, 3.6, 9.4, 23.0, 60.0, 150.0; // double
        JHisto_Bin_Boundaries_Joint_Parameter = 0.0, 180.0, 310.0, 440.0, 560.0, 680.0, 800.0,
        1100.0; // double
        _ChunkSizes = 360U, 180U, 7U, 7U; // uint

        double Standard_Deviation(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_PCL_Total: Standard_Deviation";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint

// group attributes:
        long_name = "Cloud Optical Thickness for Combined (LiquidWater+Ice+Undetermined)
        Phase Clouds (3.7 micron Retrieval for Partly Cloudy (PCL) Scenes)";
        units = "none";
        _FillValue = -999.0; // double
        valid_min = 0.0; // double
        valid_max = 150.0; // double
        scale_factor = 1.0; // double
        add_offset = 0.0; // double
}

group: Cloud_Optical_Thickness_Log10_Liquid {
variables:
        double Standard_Deviation(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_Log10_Liquid: Standard_Deviation";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint

        double Sum(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_Log10_Liquid: Sum";
        _ChunkSizes = 360U, 180U; // uint

        int Pixel_Counts(longitude=360, latitude=180);
        _FillValue = -999; // int
        title = "Cloud_Optical_Thickness_Log10_Liquid: Pixel_Counts";
        _ChunkSizes = 360U, 180U; // uint

        double Sum_Squares(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_Log10_Liquid: Sum_Squares";
        _ChunkSizes = 360U, 180U; // uint
```

```
        double Mean(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_Log10_Liquid: Mean";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint

// group attributes:
        valid_max = 2.176; // double
        scale_factor = 1.0; // double
        add_offset = 0.0; // double
        units = "none";
        _FillValue = -999.0; // double
        valid_min = -2.0; // double
        long_name = "Cloud Optical Thickness Log10 for Liquid Water Clouds (3.7 micron Retrieval
        for Cloudy Scenes)";
}

group: Cloud_Optical_Thickness_Log10_Ice {
variables:
        double Mean(longitude=360, latitude=180);
        title = "Cloud_Optical_Thickness_Log10_Ice: Mean";
        _FillValue = -999.0; // double
        units = "none";
        _ChunkSizes = 360U, 180U; // uint

        double Sum(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_Log10_Ice: Sum";
        _ChunkSizes = 360U, 180U; // uint

        double Sum_Squares(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_Log10_Ice: Sum_Squares";
        _ChunkSizes = 360U, 180U; // uint

        int Pixel_Counts(longitude=360, latitude=180);
        _FillValue = -999; // int
        title = "Cloud_Optical_Thickness_Log10_Ice: Pixel_Counts";
        _ChunkSizes = 360U, 180U; // uint

        double Standard_Deviation(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_Log10_Ice: Standard_Deviation";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint

// group attributes:
        units = "none";
        _FillValue = -999.0; // double
        valid_min = -2.0; // double
        valid_max = 2.176; // double
        scale_factor = 1.0; // double
```

```
        add_offset = 0.0; // double
        long_name = "Cloud Optical Thickness Log10 for Ice Clouds (3.7 micron Retrieval for Cloudy
        Scenes)";
}

group: Cloud_Optical_Thickness_Log10_Total {
variables:
        double Mean(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_Log10_Total: Mean";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint

        double Sum(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_Log10_Total: Sum";
        _ChunkSizes = 360U, 180U; // uint

        double Sum_Squares(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_Log10_Total: Sum_Squares";
        _ChunkSizes = 360U, 180U; // uint

        int Pixel_Counts(longitude=360, latitude=180);
        _FillValue = -999; // int
        title = "Cloud_Optical_Thickness_Log10_Total: Pixel_Counts";
        _ChunkSizes = 360U, 180U; // uint

        double Standard_Deviation(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Optical_Thickness_Log10_Total: Standard_Deviation";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint

// group attributes:
        long_name = "Cloud Optical Thickness Log10 for Combined
        (LiquidWater+Ice+Undetermined) Phase Clouds (3.7-micron Retrieval for Cloudy Scenes)";
        _FillValue = -999.0; // double
        valid_min = -2.0; // double
        valid_max = 2.176; // double
        scale_factor = 1.0; // double
        add_offset = 0.0; // double
        units = "none";
}

group: Cloud_Particle_Size_Liquid {
variables:
        double Mean(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Particle_Size_Liquid: Mean";
        units = "microns";
        _ChunkSizes = 360U, 180U; // uint
```

```
        double Standard_Deviation(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Particle_Size_Liquid: Standard_Deviation";
        units = "microns";
        _ChunkSizes = 360U, 180U; // uint

        double Sum(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Particle_Size_Liquid: Sum";
        _ChunkSizes = 360U, 180U; // uint

        int Pixel_Counts(longitude=360, latitude=180);
        _FillValue = -999; // int
        title = "Cloud_Particle_Size_Liquid: Pixel_Counts";
        _ChunkSizes = 360U, 180U; // uint

        double Sum_Squares(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Particle_Size_Liquid: Sum_Squares";
        _ChunkSizes = 360U, 180U; // uint

// group attributes:
        units = "microns";
        _FillValue = -999.0; // double
        valid_min = 4.0; // double
        valid_max = 30.0; // double
        scale_factor = 1.0; // double
        add_offset = 0.0; // double
        long_name = "Cloud Effective Radius for Liquid Water Clouds (3.7 micron Retrieval for
        Cloudy Scenes)";
}

group: Cloud_Particle_Size_Ice {
variables:
        double Mean(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Particle_Size_Ice: Mean";
        units = "microns";
        _ChunkSizes = 360U, 180U; // uint

        double Sum(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Particle_Size_Ice: Sum";
        _ChunkSizes = 360U, 180U; // uint

        double Sum_Squares(longitude=360, latitude=180);
        title = "Cloud_Particle_Size_Ice: Sum_Squares";
        _FillValue = -999.0; // double
        _ChunkSizes = 360U, 180U; // uint

        int Pixel_Counts(longitude=360, latitude=180);
        _FillValue = -999; // int
        title = "Cloud_Particle_Size_Ice: Pixel_Counts";
```

```
        _ChunkSizes = 360U, 180U; // uint

        double Standard_Deviation(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Particle_Size_Ice: Standard_Deviation";
        units = "microns";
        _ChunkSizes = 360U, 180U; // uint

// group attributes:
        units = "microns";
        _FillValue = -999.0; // double
        valid_min = 5.0; // double
        valid_max = 60.0; // double
        scale_factor = 1.0; // double
        add_offset = 0.0; // double
        long_name = "Cloud Effective Radius for Ice Clouds (3.7 micron Retrieval for Cloudy
Scenes)";
}

group: Cloud_Particle_Size_PCL_Liquid {
variables:
        double Mean(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Particle_Size_PCL_Liquid: Mean";
        units = "microns";
        _ChunkSizes = 360U, 180U; // uint

        double Sum(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Particle_Size_PCL_Liquid: Sum";
        _ChunkSizes = 360U, 180U; // uint

        double Sum_Squares(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Particle_Size_PCL_Liquid: Sum_Squares";
        _ChunkSizes = 360U, 180U; // uint

        int Pixel_Counts(longitude=360, latitude=180);
        _FillValue = -999; // int
        title = "Cloud_Particle_Size_PCL_Liquid: Pixel_Counts";
        _ChunkSizes = 360U, 180U; // uint

        double Standard_Deviation(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Particle_Size_PCL_Liquid: Standard_Deviation";
        units = "microns";
        _ChunkSizes = 360U, 180U; // uint

// group attributes:
        long_name = "Cloud Effective Radius for Liquid Water Clouds (3.7 micron Retrieval for Partly
        Cloudy (PCL) Scenes)";
        _FillValue = -999.0; // double
        valid_min = 4.0; // double
```

```
        valid_max = 30.0; // double
        scale_factor = 1.0; // double
        add_offset = 0.0; // double
        units = "microns";
}

group: Cloud_Particle_Size_PCL_Ice {
variables:
        double Mean(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Particle_Size_PCL_Ice: Mean";
        units = "microns";
        _ChunkSizes = 360U, 180U; // uint

        double Sum(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Particle_Size_PCL_Ice: Sum";
        _ChunkSizes = 360U, 180U; // uint

        double Sum_Squares(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Particle_Size_PCL_Ice: Sum_Squares";
        _ChunkSizes = 360U, 180U; // uint

        int Pixel_Counts(longitude=360, latitude=180);
        _FillValue = -999; // int
        title = "Cloud_Particle_Size_PCL_Ice: Pixel_Counts";
        _ChunkSizes = 360U, 180U; // uint

        double Standard_Deviation(longitude=360, latitude=180);
        title = "Cloud_Particle_Size_PCL_Ice: Standard_Deviation";
        units = "microns";
        _FillValue = -999.0; // double
        _ChunkSizes = 360U, 180U; // uint

// group attributes:
        units = "microns";
        _FillValue = -999.0; // double
        valid_min = 5.0; // double
        valid_max = 60.0; // double
        scale_factor = 1.0; // double
        add_offset = 0.0; // double
        long_name = "Cloud Effective Radius for Ice Clouds (3.7 micron Retrieval for Partly Cloudy
        (PCL) Scenes)";
}

group: Cloud_Water_Path_Liquid {
variables:
        double Mean(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Water_Path_Liquid: Mean";
        units = "g/m^2";
        _ChunkSizes = 360U, 180U; // uint
```

```
        double Standard_Deviation(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Water_Path_Liquid: Standard_Deviation";
        units = "g/m^2";
        _ChunkSizes = 360U, 180U; // uint

        double Sum(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Water_Path_Liquid: Sum";
        _ChunkSizes = 360U, 180U; // uint

        int JHisto_vs_Cloud_Particle_Size_Liquid(longitude=360, latitude=180,
        jhisto_cloud_water_path_liquid_7=7, jhisto_cloud_particle_size_liquid_6=6);
        title = "Cloud_Water_Path_Liquid: JHisto_vs_Cloud_Particle_Size_Liquid";
        JHisto_Bin_Boundaries = 0.0, 10.0, 30.0, 60.0, 100.0, 150.0, 250.0, 20000.0; // double
        JHisto_Bin_Boundaries_Joint_Parameter = 4.0, 8.0, 10.0, 12.5, 15.0, 20.0, 30.0; // double
        _FillValue = -999; // int
        _ChunkSizes = 360U, 180U, 7U, 6U; // uint

        double Sum_Squares(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Water_Path_Liquid: Sum_Squares";
        _ChunkSizes = 360U, 180U; // uint

        Int Pixel_Counts(longitude=360, latitude=180);
        _FillValue = -999; // int
        title = "Cloud_Water_Path_Liquid: Pixel_Counts";
        _ChunkSizes = 360U, 180U; // uint

// group attributes:
        units = "g/m^2";
        _FillValue = -999.0; // double
        valid_min = 0.0; // double
        valid_max = 3000.0; // double
        scale_factor = 1.0; // double
        add_offset = 0.0; // double
        long_name = "Cloud Water Path for Liquid Water Clouds (3.7 micron Retrieval for Cloudy
        Scenes)";
}

group: Cloud_Water_Path_Ice {
variables:
        double Mean(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Water_Path_Ice: Mean";
        units = "g/m^2";
        _ChunkSizes = 360U, 180U; // uint

        double Sum(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Water_Path_Ice: Sum";
        _ChunkSizes = 360U, 180U; // uint
```

```
        int JHisto_vs_Cloud_Particle_Size_Ice(longitude=360, latitude=180,
        jhisto_cloud_water_path_ice_7=7, jhisto_cloud_particle_size_ice_6=6);
        _FillValue = -999; // int
        title = "Cloud_Water_Path_Ice: JHisto_vs_Cloud_Particle_Size_Ice";
        JHisto_Bin_Boundaries = 0.0, 20.0, 50.0, 100.0, 200.0, 400.0, 1000.0, 20000.0; // double
        JHisto_Bin_Boundaries_Joint_Parameter = 5.0, 10.0, 20.0, 30.0, 40.0, 50.0, 60.0; // double
        _ChunkSizes = 360U, 180U, 7U, 6U; // uint

        double Sum_Squares(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Water_Path_Ice: Sum_Squares";
        _ChunkSizes = 360U, 180U; // uint

        int Pixel_Counts(longitude=360, latitude=180);
        _FillValue = -999; // int
        title = "Cloud_Water_Path_Ice: Pixel_Counts";
        _ChunkSizes = 360U, 180U; // uint

        double Standard_Deviation(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Water_Path_Ice: Standard_Deviation";
        units = "g/m^2";
        _ChunkSizes = 360U, 180U; // uint

// group attributes:
        valid_min = 0.0; // double
        valid_max = 6000.0; // double
        scale_factor = 1.0; // double
        add_offset = 0.0; // double
        long_name = "Cloud Water Path for Ice Clouds (3.7 micron Retrieval for Cloudy Scenes)";
        units = "g/m^2";
        _FillValue = -999.0; // double
}

group: Cloud_Water_Path_PCL_Liquid {
variables:
        double Mean(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Water_Path_PCL_Liquid: Mean";
        units = "g/m^2";
        _ChunkSizes = 360U, 180U; // uint

        double Standard_Deviation(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Water_Path_PCL_Liquid: Standard_Deviation";
        units = "g/m^2";
        _ChunkSizes = 360U, 180U; // uint

        int Pixel_Counts(longitude=360, latitude=180);
        _FillValue = -999; // int
        title = "Cloud_Water_Path_PCL_Liquid: Pixel_Counts";
        _ChunkSizes = 360U, 180U; // uint
```

```
        double Sum_Squares(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Water_Path_PCL_Liquid: Sum_Squares";
        _ChunkSizes = 360U, 180U; // uint

        double Sum(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Water_Path_PCL_Liquid: Sum";
        _ChunkSizes = 360U, 180U; // uint

        int JHisto_vs_Cloud_Particle_Size_PCL_Liquid(longitude=360, latitude=180,
        jhisto_cloud_water_path_pcl_liquid_7=7, jhisto_cloud_particle_size_pcl_liquid_6=6);
        JHisto_Bin_Boundaries = 0.0, 10.0, 30.0, 60.0, 100.0, 150.0, 250.0, 20000.0; // double
        JHisto_Bin_Boundaries_Joint_Parameter = 4.0, 8.0, 10.0, 12.5, 15.0, 20.0, 30.0; // double
        _FillValue = -999; // int
        title = "Cloud_Water_Path_PCL_Liquid: JHisto_vs_Cloud_Particle_Size_PCL_Liquid";
        _ChunkSizes = 360U, 180U, 7U, 6U; // uint

// group attributes:
        valid_min = 0.0; // double
        valid_max = 3000.0; // double
        scale_factor = 1.0; // double
        add_offset = 0.0; // double
        units = "g/m^2";
        _FillValue = -999.0; // double
        long_name = "Cloud Water Path for Liquid Water Clouds (3.7-micron Retrieval for Partly
        Cloudy (PCL) Scenes)";
}

group: Cloud_Water_Path_PCL_Ice {
variables:
        double Mean(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Water_Path_PCL_Ice: Mean";
        units = "g/m^2";
        _ChunkSizes = 360U, 180U; // uint

        double Sum(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Water_Path_PCL_Ice: Sum";
        _ChunkSizes = 360U, 180U; // uint

        int JHisto_vs_Cloud_Particle_Size_PCL_Ice(longitude=360, latitude=180,
        jhisto_cloud_water_path_pcl_ice_7=7, jhisto_cloud_particle_size_pcl_ice_6=6);
        JHisto_Bin_Boundaries = 0.0, 20.0, 50.0, 100.0, 200.0, 400.0, 1000.0, 20000.0; // double
        JHisto_Bin_Boundaries_Joint_Parameter = 5.0, 10.0, 20.0, 30.0, 40.0, 50.0, 60.0; // double
        _FillValue = -999; // int
        title = "Cloud_Water_Path_PCL_Ice: JHisto_vs_Cloud_Particle_Size_PCL_Ice";
        _ChunkSizes = 360U, 180U, 7U, 6U; // uint

        double Sum_Squares(longitude=360, latitude=180);
        _FillValue = -999.0; // double
```

```
        title = "Cloud_Water_Path_PCL_Ice: Sum_Squares";
        _ChunkSizes = 360U, 180U; // uint

        int Pixel_Counts(longitude=360, latitude=180);
        _FillValue = -999; // int
        title = "Cloud_Water_Path_PCL_Ice: Pixel_Counts";
        _ChunkSizes = 360U, 180U; // uint

        double Standard_Deviation(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Water_Path_PCL_Ice: Standard_Deviation";
        units = "g/m^2";
        _ChunkSizes = 360U, 180U; // uint

// group attributes:
        units = "g/m^2";
        _FillValue = -999.0; // double
        valid_min = 0.0; // double
        valid_max = 6000.0; // double
        scale_factor = 1.0; // double
        add_offset = 0.0; // double
        long_name = "Cloud Water Path for Ice Clouds (3.7 micron Retrieval for Partly Cloudy (PCL)
        Scenes)";
}

group: Cloud_Retrieval_Fraction_Liquid {
variables:
        double Mean(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Retrieval_Fraction_Liquid: Mean";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint

        double Sum(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Retrieval_Fraction_Liquid: Sum";
        _ChunkSizes = 360U, 180U; // uint

        double Sum_Squares(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Retrieval_Fraction_Liquid: Sum_Squares";
        _ChunkSizes = 360U, 180U; // uint

        int Pixel_Counts(longitude=360, latitude=180);
        _FillValue = -999; // int
        title = "Cloud_Retrieval_Fraction_Liquid: Pixel_Counts";
        _ChunkSizes = 360U, 180U; // uint

        double Standard_Deviation(longitude=360, latitude=180);
        title = "Cloud_Retrieval_Fraction_Liquid: Standard_Deviation";
        units = "none";
        _FillValue = -999.0; // double
        _ChunkSizes = 360U, 180U; // uint
```

```
// group attributes:
        valid_max = 1.0; // double
        scale_factor = 1.0; // double
        add_offset = 0.0; // double
        _FillValue = -999.0; // double
        valid_min = 0.0; // double
        long_name = "Cloud Optical Properties 3.7 Retrieval Fraction (Liquid Water Clouds)";
        units = "none";
}

group: Cloud_Retrieval_Fraction_Ice {
variables:
        double Mean(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Retrieval_Fraction_Ice: Mean";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint

        double Sum(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Retrieval_Fraction_Ice: Sum";
        _ChunkSizes = 360U, 180U; // uint

        double Sum_Squares(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Retrieval_Fraction_Ice: Sum_Squares";
        _ChunkSizes = 360U, 180U; // uint

        int Pixel_Counts(longitude=360, latitude=180);
        _FillValue = -999; // int
        title = "Cloud_Retrieval_Fraction_Ice: Pixel_Counts";
        _ChunkSizes = 360U, 180U; // uint

        double Standard_Deviation(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Retrieval_Fraction_Ice: Standard_Deviation";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint

// group attributes:
        long_name = "Cloud Optical Properties 3.7 Retrieval Fraction (Ice Clouds)";
        units = "none";
        _FillValue = -999.0; // double
        valid_min = 0.0; // double
        valid_max = 1.0; // double
        scale_factor = 1.0; // double
        add_offset = 0.0; // double
}

group: Cloud_Retrieval_Fraction_Total {
variables:
        double Mean(longitude=360, latitude=180);
```

```
            _FillValue = -999.0; // double
            title = "Cloud_Retrieval_Fraction_Total: Mean";
            units = "none";
            _ChunkSizes = 360U, 180U; // uint

            double Sum(longitude=360, latitude=180);
            _FillValue = -999.0; // double
            title = "Cloud_Retrieval_Fraction_Total: Sum";
            _ChunkSizes = 360U, 180U; // uint

            double Sum_Squares(longitude=360, latitude=180);
            _FillValue = -999.0; // double
            title = "Cloud_Retrieval_Fraction_Total: Sum_Squares";
            _ChunkSizes = 360U, 180U; // uint

            int Pixel_Counts(longitude=360, latitude=180);
            _FillValue = -999; // int
            title = "Cloud_Retrieval_Fraction_Total: Pixel_Counts";
            _ChunkSizes = 360U, 180U; // uint

            double Standard_Deviation(longitude=360, latitude=180);
            _FillValue = -999.0; // double
            title = "Cloud_Retrieval_Fraction_Total: Standard_Deviation";
            units = "none";
            _ChunkSizes = 360U, 180U; // uint

// group attributes:
            long_name = "Cloud Optical Properties 3.7 Retrieval Fraction (Combined
            (LiquidWater+Ice+Undetermined) Phase Clouds)";
            _FillValue = -999.0; // double
            valid_min = 0.0; // double
            valid_max = 1.0; // double
            scale_factor = 1.0; // double
            add_offset = 0.0; // double
            units = "none";
}

group: Cloud_Retrieval_Fraction_PCL_Liquid {
variables:
            double Mean(longitude=360, latitude=180);
            _FillValue = -999.0; // double
            title = "Cloud_Retrieval_Fraction_PCL_Liquid: Mean";
            units = "none";
            _ChunkSizes = 360U, 180U; // uint

            double Sum(longitude=360, latitude=180);
            _FillValue = -999.0; // double
            title = "Cloud_Retrieval_Fraction_PCL_Liquid: Sum";
            _ChunkSizes = 360U, 180U; // uint

            double Sum_Squares(longitude=360, latitude=180);
            _FillValue = -999.0; // double
            title = "Cloud_Retrieval_Fraction_PCL_Liquid: Sum_Squares";
```

```
        _ChunkSizes = 360U, 180U; // uint

        int Pixel_Counts(longitude=360, latitude=180);
        _FillValue = -999; // int
        title = "Cloud_Retrieval_Fraction_PCL_Liquid: Pixel_Counts";
        _ChunkSizes = 360U, 180U; // uint

        double Standard_Deviation(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Retrieval_Fraction_PCL_Liquid: Standard_Deviation";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint

// group attributes:
        long_name = "Cloud Optical Properties 3.7 Retrieval Fraction (Liquid Water Clouds) for Partly
        Cloudy (PCL) Retrievals";
        _FillValue = -999.0; // double
        valid_min = 0.0; // double
        valid_max = 1.0; // double
        scale_factor = 1.0; // double
        add_offset = 0.0; // double
        units = "none";
}

group: Cloud_Retrieval_Fraction_PCL_Ice {
variables:
        double Mean(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Retrieval_Fraction_PCL_Ice: Mean";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint

        double Sum(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Retrieval_Fraction_PCL_Ice: Sum";
        _ChunkSizes = 360U, 180U; // uint

        double Sum_Squares(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Retrieval_Fraction_PCL_Ice: Sum_Squares";
        _ChunkSizes = 360U, 180U; // uint

        int Pixel_Counts(longitude=360, latitude=180);
        _FillValue = -999; // int
        title = "Cloud_Retrieval_Fraction_PCL_Ice: Pixel_Counts";
        _ChunkSizes = 360U, 180U; // uint

        double Standard_Deviation(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Retrieval_Fraction_PCL_Ice: Standard_Deviation";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint
```

```
// group attributes:
        units = "none";
        _FillValue = -999.0; // double
                valid_min = 0.0; // double
        valid_max = 1.0; // double
        scale_factor = 1.0; // double
        add_offset = 0.0; // double
        long_name = "Cloud Optical Properties 3.7 Retrieval Fraction (Ice Clouds) for Partly Cloudy (PCL)
        Retrievals"
}

group: Cloud_Retrieval_Fraction_PCL_Total {
variables:
        double Mean(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Retrieval_Fraction_PCL_Total: Mean";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint

        double Sum(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Retrieval_Fraction_PCL_Total: Sum";
        _ChunkSizes = 360U, 180U; // uint

        int Pixel_Counts(longitude=360, latitude=180);
        _FillValue = -999; // int
        title = "Cloud_Retrieval_Fraction_PCL_Total: Pixel_Counts";
        _ChunkSizes = 360U, 180U; // uint

        double Sum_Squares(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Retrieval_Fraction_PCL_Total: Sum_Squares";
        _ChunkSizes = 360U, 180U; // uint

        double Standard_Deviation(longitude=360, latitude=180);
        _FillValue = -999.0; // double
        title = "Cloud_Retrieval_Fraction_PCL_Total: Standard_Deviation";
        units = "none";
        _ChunkSizes = 360U, 180U; // uint

// group attributes:
        long_name = "Cloud Optical Properties 3.7 Retrieval Fraction (Combined
        (LiquidWater+Ice+Undetermined) Phase Clouds) for Partly Cloudy (PCL) Retrievals";
        _FillValue = -999.0; // double
        valid_min = 0.0; // double
        valid_max = 1.0; // double
        scale_factor = 1.0; // double
        add_offset = 0.0; // double
        units = "none";
}
```

```
// global attributes:

date_created = "2024-02-13T18:45:06Z";
product_name = "CLDPROPCOSP_D3_MODIS_Aqua.A2023001.011.2024044184622.nc";
LocalGranuleID = "CLDPROPCOSP_D3_MODIS_Aqua.A2023001.011.2024044184622.nc";
Conventions = "CF-1.6, ACDD-1.3";
ShortName = "CLDPROPCOSP_D3_MODIS_Aqua";
product_version = "1.1.3";
Yori_version = "1.5.0";
daily_defn_of_day_adjustment = "False";
AlgorithmType = "OPS";
identifier_product_doi = "10.5067/MODIS/CLDPROPCOSP_D3_MODIS_Aqua.011";
identifier_product_doi_authority = "http://dx.doi.org";
DataCenterId = "UWI-MAD/SSEC/ASIPS";
project = "NASA VIIRS Atmosphere SIPS";
creator_name = "NASA VIIRS Atmosphere SIPS";
creator_url = "https://sips.ssec.wisc.edu";
creator_email = "sips.support@ssec.wisc.edu";
creator_institution = "Space Science & Engineering Center, University of Wisconsin - Madison";
publisher_name = "LAADS";
publisher_url = "https://ladsweb.modaps.eosdis.nasa.gov/";
publisher_email = "modis-ops@lists.nasa.gov";
publisher_institution = "NASA Level-1 and Atmosphere Archive & Distribution System";
time_coverage_start = "2023-01-01T00:00:00.000000";
time_coverage_end = "2023-01-01T23:59:59.000000";
platform = "Aqua";
instrument = "MODIS";
processing_level = "L3";
format = "NetCDF4";
title = "Aqua MODIS Cloud Properties Level 3 daily, 1x1 degree grid (CLDPROPCOSP_D3_MODIS_Aqua)";
long_name = "MODIS/Aqua Cloud Properties Level 3 daily, 1x1 degree grid";
version_id = "011";
geospatial_lat_max = 90.0; // double
geospatial_lat_min = -90.0; // double
geospatial_lon_min = 180.0; // double
geospatial_lon_max = -180.0; // double
NorthBoundingCoordinate = 90.0; // double
SouthBoundingCoordinate = -90.0; // double
EastBoundingCoordinate = 180.0; // double
WestBoundingCoordinate = -180.0; // double
latitude_resolution = 1.0; // double
longitude_resolution = 1.0; // double
license = "http://science.nasa.gov/earth-science/earth-science-data/data-information-policy/";
stdname_vocabulary = "NetCDF Climate and Forecast (CF) Metadata Convention";
keywords_vocabulary = "NASA Global Change Master Directory (GCMD) Science Keywords";
keywords = "EARTH SCIENCE > ATMOSPHERE > CLOUDS > CLOUD MICROPHYSICS > CLOUD
OPTICAL DEPTH/THICKNESS
          EARTH SCIENCE > ATMOSPHERE > CLOUDS > CLOUD PROPERTIES > CLOUD TOP HEIGHT
          EARTH SCIENCE > ATMOSPHERE > CLOUDS > CLOUD PROPERTIES > CLOUD FRACTION";
naming_authority = "gov.nasa.gsfc.sci.atmos";
}
```