PRECIPITATION PROCESSING SYSTEM

TROPICAL RAINFALL MEASURING MISSION

File Specification for TRMM Products

Version 7.005

December 8, 2016

Contents

List of Figures

This document contains file specifications for each algorithm written by PPS.

The introductory sections and algorithms 1A01, 1A11, and 1A21 were transferred from the TSDIS File Specification for Version 6 products. These sections may contain out-of-date references to TSDIS, missing tables or missing figures. These will be corrected in the future.

1 Introduction

The Tropical Rainfall Measuring Mission (TRMM) is an integral part of the National Aeronautics and Space Administration (NASA) Earth Science Enterprise. The TRMM observatory was launched in 1997 into a near circular orbit of approximately 350 kilometers altitude with an inclination of 35 degrees and a period of 91.5 minutes (15.7 orbits per day). During August 2001, the TRMM altitude was raised to approximately 402.5 kilometers with a period of 92.5 minutes (15.6 orbits per day). The Precipitation Processing System (PPS) is the data processing and science information system for TRMM and GPM. The Earth Observing System Data and Information System (EOSDIS) serves as the long-term archive for all TRMM data products.

1.1 Identification

This is the Precipitation Processing System (PPS) TROPICAL RAINFALL MEASURING MISSION File Specification for TRMM Products.

1.2 Scope

This document describes the data file formats for TRMM products. Metadata are described in Metadata for TRMM Products.

1.3 Purpose and Objectives

The purpose of this file specification document is to define the file content and format for the TRMM data products.

The TRMM Science Requirements references intermediate products for the purpose of monitoring instruments and product production, which this document refers to as "Instrument Analysis Products." Since this data is not archived at EOSDIS, these formats are not described in this document.

The specifications in this document incorporate the results from a series of meetings with EOSDIS for agreements on file structure, and TRMM science team members for file contents. Discussions with other missions that have similar data sets have helped to form the initial basis for the file specifications. In addition, characteristics from popular formats were incorporated into the file formats (e.g., the Wentz format is used to help develop the TMI file specification).

1.4 Document Status and Schedule

The file specifications have been reviewed by the algorithm developers and are expected to be fairly stable. The file specifications will be instrumental in developing the algorithm software, the PPS toolkit and refining the Data Volume Estimates. The schedule is as follows:

* Version 1 11/07/94 Complete

* Version 2 12/07/94 Complete

* Version 3 4/14/95 Updated metadata with new EOSDIS definitions and updated orbit definition.

* ICS Release 2 8/31/95 Incorporates comments from algorithm developers

* ICS Release 3 Draft 3/14/96 Incorporates file structure and metadata updates from EOSDIS.

* ICS Release 3 6/28/96 Incorporates Level-1 PR modifications from NASDA

* ICS Release 4 9/01/98 Incorporates PR modifications and removes Metadata information from Appendices.

* ICS Release 5 10/01/99 Updates for Version 5 product.

* ICS Release 6 8/24/04 Updates for Version 6 products.

* Release 7 2010

1.5 Document Organization

This document is organized so that a section is dedicated to each instrument. The organization is as follows:

Section 2.0 EOSDIS AND HDF - This section describes EOSDIS structures, aspects of Hierarchical Data Format (HDF) which are used in the PPS file formats, and the approach to recording metadata.

Section 3.0 FORMATTING CONVENTIONS - This section describes general formatting conventions used in this document.

Section 4.0 LEVEL-1A TRMM MICROWAVE IMAGER (TMI) - This section describes the Level 1A file specifications for the TMI instrument. It provides a brief description of the instrument and the scan geometry as an introduction. The structure of the TMI file is provided, along with descriptions of the data objects and the associated metadata.

Section 5.0 LEVEL-1A VISIBLE AND INFRARED SCANNER (VIRS) - This section describes Level 1A the file specifications for the VIRS instrument. It provides a brief description of the instrument and the scan geometry as an introduction. The structure of the VIRS file is provided, along with descriptions of the data objects and the associated metadata.

Section 6.0 LEVEL-1A PRECIPITATION RADAR (PR) - This section describes the Level 1A file specifications for the PR instrument. It provides a brief description of the instrument and the scan geometry as an introduction. The structure of the PR file is provided, along with descriptions of the data objects and the associated metadata.

Section 7.0 STANDARD TRMM ALGORITHMS - This section describes the file specifications for standard TRMM algorithms written by the PPS Toolkit. The structure of the file is provided, along with descriptions of the data objects and the associated metadata.

Section 8.0 NON-STANDARD TRMM ALGORITHMS - This section describes the file specifications for non-standard TRMM algorithms written by the PPS Toolkit. The structure of the file is provided, along with descriptions of the data objects and the associated metadata.

2 EOS and HDF

2.1 EOSDIS Structures

2.1.1 Swath Structure

The swath structure was created by EOSDIS to store satellite data which are organized by scans. TSDIS implements the swath structure in Levels 1B, 1C, 2A, and 2B satellite products. Figure 2.1.1-1 shows a generic swath structure as it is used in TSDIS data products. More description TBD.

Figure 2.1.1-1 Generic Swath Structure TBD

More description TBD.

2.1.2 Planetary Grid Structure

The Planetary Grid Structure is a structure created by EOSDIS to store earth located grids. The grid is an array of grid boxes, rather than grid points. TSDIS employs the Planetary Grid Structure in Level 3A and 3B satellite products. Figure 2.1.2-1 shows a generic Planetary Grid Structure as it is used in TSDIS formats. The Planetary Grid Structure occupies part of a file. This structure is contained in a Vgroup, with the name PlanetaryGrid and the class PlanetaryGrid. In that Vgroup appear one GridStructure, one or more Data Grids, and other Data. GridStructure is a single Vdata which allows the geometric interpretation of the grids. GridStructure is an object that mimics an attribute, since HDF has not yet defined attributes for Vgroups. This imitation of an attribute is implemented as a Vdata with the name GridStructure and the class "Attr0.0" , one field named "VALUES", number type of DFNT_CHAR8, and order equal to the length of the text. This specification of GridStructure anticipates the HDF development of attributes for Vgroups. The maximum expected size for GridStructure is 5000 bytes. Since the purpose of GridStructure is to allow EOSDIS to ingest data into their archive, Algorithm Developers do not need to read from or write to GridStructure. Table 2.1.2-1 specifies the fields within GridStructure. Six of the fields (the resolutions and bounding coordinates) are also found in Core Metadata. Three fields (bin_meth, registration, and Origin) are not found in Core Metadata.

Figure 2.1.2-1 Generic Planetary Grid Structure TBD

Table 2.1.2-1 GridStructure Fields TBD

Unless otherwise specified, bin_meth = "ARITHMEAN" and registration = "CENTER." TSDIS Planetary Grids will always have North Bounding Coordinate = 40, South Bounding Coordinate = -40, West Bounding Coordinate = -180, East Bounding Coordinate = 180, and Origin = "SOUTHWEST". Unless otherwise specified, Latitude Resolution = Longitude Resolution = 5.

Each Data Grid is an SDS with dimensions Y x nlat x nlon, where Y is the number of variables and nlat and nlon are the number of North-South and East-West grid points, respectively. The names of the latitude and longitude dimensions are Latitude_X and Longitude_X, where X is the name of the Data Grid SDS. Other dimensions have the names specified in the Swath description. The name of the SDS is the name of the variable contained in the grid.

To avoid repetitious text, certain defaults are used in this document for the formats of a Data Grid SDS: unless otherwise specified, the names of the dimensions are as above.

2.2 Hierarchical Data Format (HDF)

Level-1B and higher data products, except 2A-52, produced by TSDIS will be sent to EOSDIS in the Hierarchical Data Format (HDF). HDF was developed by the National Center for Supercomputing Applications (NCSA) at the University of Illinois at Urbana-Champaign and is the selection for the common data format for EOSDIS. HDF manuals and software may be obtained via anonymous ftp at ftp.ncsa.uiuc.edu Communications to NCSA concerning HDF may be made by email to softdev@ncsa.uiuc.edu or by postal mail to:

```
NCSA Software Tools Group HDF
152 Computing Applications Bldg.
605 E. Springfield Ave.
Champaign, IL  61820
USA
```

HDF is a multi-object file format for the transfer of graphical and numerical data between heterogeneous machines. The design of this format allows self-definition of data content, and easy extensibility for future enhancements or compatibility with other standard formats. HDF provides these benefits to algorithm developers and data users:

* versatile file format: HDF supports six different data models (8-bit raster model, palette model, 24-bit raster model, scientific data (SDS) model, annotation model, and the virtual data (Vdata) model).

* extensible file format: Although HDF currently supports six different data models, new data models will need to be incorporated in the future. HDF can easily accommodate new data models either through NCSA or HDF users.

* self-describing: HDF permits the inclusion of a data index (metadata) inside the data file which describes the contents of the file. This simplifies data sharing among diverse scientists.

* portable file format: HDF files are portable across heterogeneous computer systems. This allows for an HDF file created on one system (e.g., UNIX) to be used on a different system (e.g., Macintosh).

* data abstraction: When either reading or writing to an HDF file, the user does not have to be concerned with the phsical layout of the file.

2.2.1 Relevant HDF Data Models

The EOSDIS data structures are composed of Vgroups and simple HDF models that are well tested and common to the science community. A Vgroup is like a node in a tree structure. The members of a Vgroup can be data models or more Vgroups. Of the six defined data models, the TSDIS file specifications use the following models:

a) Scientific Data Set - An SDS is an HDF structure that stores rectangular gridded arrays of data, together with information about the data. An SDS can contain data that are multi-dimensional, but must be of a singular data type and composed of the following elements:

* The actual data values.

* The number of dimensions (rank) and the size of each dimension

* The type of the data (e.g., integer or floating point)

* (optional) Scales to be used along the different dimensions when interpreting or displaying the data

* (optional) Labels for all dimensions and for the data

* (optional) Units for all dimensions and for the data

* (optional) Format specifications to be used when displaying values for the dimensions and for the data

* (optional) A range, attributing maximum and minimum values for the data set.

* (optional) Calibration information including an offset and scale factor.

* (optional) A fill value for representing missing data in a data set.

* (optional) The coordinate system to be used when interpreting or displaying the data

b) VData - The HDF VData model provides a framework for storing related data in a table-like structure in HDF files. Each table is comprised of a collection of similar records called VData records, whose values are stored in fixed-length fields. Each field can support its own data types, however, every VData record in a table must contain the same types of fields.

2.2.2 HDF Structures for Metadata

Metadata are defined as information about data sets which provide a description of the content, format, and utility of the data set. The word metadata is derived from the Greek word Meta, meaning "with". Metadata may be used to select data for a particular scientific investigation. Each metadata element will have the "label = value" structure. The "label" will be defined by EOSDIS for the core metadata, and by TSDIS for the product specific metadata; the "value" will be supplied by TSDIS. For some core fields though, the values will need to comply with EOSDIS guidelines.

EOSDIS has decided that metadata should be stored in file attributes. Attributes are blocks of text that are attached to either files or SDSs.

2.3 Metadata Implementation

TBD.

3 Formatting Conventions

3.1 File Structure Figures

The figures that illustrate file structure contain either Vgroups or data objects (metadata objects, SDSs, or Vdatas). Figure 3.1-1 is an example of a product structure with annotations shown in italics. Vgroups are represented as the name of the Vgroup without a box. Data objects are represented as the name of the object inside a box. The order of the data objects in the file may differ from their order in the file specification. For metadata objects the estimated maximum total size appears on the right hand side of the box. If the object is a Vdata table, the size of one record appears on the right side of the box and the number of records appears next to the box. If the object is a SDS array, the size of one element appears on the right side of the box and the dimensions of the array appear next to the box.

Figure 3.1-1 Example Product Structure TBD

The sizes for the metadata objects are estimated maxima since the values of many metadata are free text and may vary in length and not all metadata elements are used for all products. None of the sizes take HDF overhead into account. Previous (unpublished) experience gained in the TSDIS prototype study and the HDF internal feasibility study has shown HDF overhead to be less than 10% of the total file size for TSDIS products.

3.2 File Contents

In the description of the contents of each object within a file, each object is defined in the following format:

Name (Type of HDF structure, Dimensions, word size and type): Description

3.3 Missing Data and Empty Granules

Missing satellite scans are filled with standard values denoting missing data. Missing satellite scans also have the "missing" byte in Scan Status set to 1. Values less than or equal to -99, -9999, -9999, -9999.9, -9999.9 denote missing or invalid data for 1-byte integer, 2-byte integer, 4-byte integer, 4-byte float, and 8-byte float, respectively. Any exceptions to the use of these standard values are explicitly noted in the description of the object. For the PR instrument, scans whose mode is other than observation mode are filled with missing values.

If an entire granule is missing, an empty granule may be created. If an entire orbit of Level 1B, 1C, 2A, or 2B satellite data is missing, scan data is omitted and the PS metadata named "Orbit Size" has the value zero. If an entire hour of Level 1B, 1C, 2A53, 2A54, or 2A55 GV data is missing, volume scan data is omitted and the PS metadata named "Number of VOS" has the value zero. An empty granule is not defined for pentad or monthly averaged data.

3.4 Array Dimension Order

In the definition of array dimensions, e.g., npixel x nscan, the first dimension (npixel) has the most rapidly varying index and the last dimension (nscan) has the least rapidly varying index. To implement this format in FORTRAN, declare an array with dimensions as they appear in this document. To implement the format in C, declare an array with dimensions reversed from their appearance in this document.

3.5 Orbit and Granule Definition

The beginning and ending time of an orbit is defined as the time when the sub-satellite track reaches its southernmost latitude. This time is determined from the definitive ephemeris data. A scan is included in an orbit when its Scan Time is greater than or equal to the Orbit Start Time and less than the Orbit End Time. The average orbit is 91.5 minutes or 5490 seconds before August 7, 2001 and 92.5 minutes or 5550s after

August 24, 2001 The first partial orbit after launch will be orbit 1, so the first full orbit will be orbit 2.

A granule is defined as one orbit for the VIRS and PR instruments. For the TMI instrument, a granule is defined as one orbit plus an overlap before the orbit, known as the Preorbit Overlap, plus an overlap after the orbit, known as the Postorbit Overlap. The overlap size is fixed at exactly 50 scans. Since there are two overlap periods per granule, each granule will contain 100 overlap scans. Thus the last 100 scans of any granule are duplicated by the first 100 scans of the next granule. See Figure 3.5-1

Overlaps are used to allow algorithm 2B-31 to open only one input granule in order to output one granule. The overlap is needed because 2B-31 requires both TMI and PR measurements at the same location. Since PR points at nadir and TMI points at a 49o angle off of nadir, the colocated measurements will occur around a minute apart.

Figure 3.5-1 Granule Structure Time Increases Toward the Right TBD

The formats in this document for products with overlap (1A-11, 1B-11, and 2A-12) follow the assumption that uniformity within one granule is preferable to uniformity for the same pixel across granules. Therefore one ephemeris file and one UTCF are used in one granule. In a similar vein, the calibration is started at the beginning of the granule and reaches satisfactory values within 10 scans. The advantage of granule uniformity is that there are no discontinuities within a granule and processing has only to input one granule in order to output one granule. The disadvantage is that a pixel in one granule may have a different value, location, and time from the same pixel in another granule. When such a difference occurs, the pixel is in an overlap region in one of the granules. According to the TRMM requirements, the location and time differences will be less than 1 km and 1 ms, respectively.

In Level-1A, extra (usually one) ACS and instrument housekeeping packets are added to ensure that each science packet has an ACS and instrument housekeeping packet before and after the science packet.

3.6 Scans in a Granule

The number of scans in a granule is shown in the structure diagrams and array dimensions as nscan. For VIRS and PR, the average nscan is calculated from the average number of seconds in an orbit as follows:

AVNSCAN = SS x SO.

For TMI, the average nscan is calculated from the average number of seconds in an orbit as follows:

AVNSCAN = SS x SO + 100, where

Table TBD

3.7 Time

Scan Times and Orbit Start Times are stored in the Level-1A headers, the metadata, and in the object named "Scan Time". The Orbit Start Time is determined from ephemeris data and the definition of the orbit start – it is independent of any scans. In contrast, a Scan Time is a time associated with a scan of a particular instrument. The Scan Time is the time tag stamped on each science telemetry packet. In particular, the Orbit First Scan Time is the Scan Time of the first scan in an orbit, which occurs at or later than the Orbit Start Time. The Level-1A header stores the Universal Time Correlation Factor (UTCF) derived from the first ACS packet in the orbit. This UTCF is used to translate the Orbit Start Time from UTC to spacecraft clock time. In normal processing, the UTCF, the Scan Times in UTC, and the Scan Times in spacecraft clock time are repeated exactly in Level-1B and higher levels. In the unusual circumstance that the UTCF is found to be incorrect, a corrected UTCF will be stored in Level-1B and higher data products and a flag set to indicate that a corrected UTCF was used. When a corrected UTCF is applied, the UTC Scan Times will be different between (1) Level-1A and (2) Level-1B and higher levels, although the spacecraft clock Scan Times will be the same in Level-1A and Level-1B and higher levels. Another flag in Level-1B and higher levels shows whether a leap second occurred in the granule.

Times are expressed in five formats:

(1) UTC times in Core or PS metadata or a Level-1A header are written in three words: Date, Time, and Milliseconds. For the Begin and End Times in Core metadata, milliseconds is omitted.

Date is a 10 character string with the following characters:

```
YYYY/MM/DD, where
YYYY = year,
MM = month number,
DD = day of month, and
"/" is a literal.

Time is an 8 character string with the following characters:

HH:MM:SS, where
HH = hour,
MM = minute,
SS = second, and
":" is a literal.

Milliseconds is a 3 character string with the following characters:

MMM, where
MMM = the number of milliseconds later than the last whole second.
```

(2) In 2A-52, UTC time is stored as in (1) except "/" is replaced by "-".

(3) In 1B-11 and 2A-12, UTC time is stored in separate words for year, month, day of month, hour, minute, and second.

(4) UTC Scan Time in the body of the data is in seconds of the day. The UTC date and time in the metadata can be combined with the Scan Time to get a complete date and time for every scan.

(5) Spacecraft clock time and UTCF have the same format, which is specified in the TRMM Telemetry and Command Handbook, Section 3.3.2.

Spacecraft clock time is the accumulated time count since the power-up of the clock card in the TRMM Spacecraft Data System onboard the satellite. Spacecraft time is correlated to UTC time by the UTCF. The sum of the UTCF and Spacecraft time results in a time that represents the total number of seconds since January 1, 1993 at 00:00:00 (UTC) if one assumes that each day has exactly 86400 seconds, even days with leap seconds. This total number of seconds allows easy computation of days since Janulary 1, 1993. However, to accurately compute time differences one must use more complicated methods, which account for leap seconds. For a more thorough discussion of Spacecraft time and UTCF, see the TRMM Telemetry and Command Handbook, Section 3.3.2.

Scan Time is a time associated with each satellite science data scan. It is the time tag written in each science telemetry packet. There is one scan per science telemetry packet. The relationship of Scan Time to the time at each IFOV varies by instrument. A description of the relationship between Scan Time and measurement time for each of the three satellite instruments follows. In each description, T is the beginning sample time and i is the IFOV number:

(1) For TMI, the equations shown in Table 3.7-1 were obtained by personal communication with the instrument scientist.

Table 3.7-1 TMI Equations TBD

(2) For VIRS, the following equation was derived from a viewgraph produced at Hughes and presented by Bruce Love on January 20, 1995:

T = Scan Time + 107.6 ms + (OFFSET + (i - 1)) * Sample Time,

where i = 1, 261, Sample Time = 0.29157 ms, and OFFSET values are shown in Table 3.7-2.

Table 3.7-2 OFFSET Values TBD

The value of Sample Time was derived from the viewgraph using the time of the starting and ending channel 1 science data as follows:

Sample Time = (183.7 ms - 107.6 ms) / 261

(3) For PR, the information was obtained by personal communication with a representative of NASDA.

T = Scan Time + 3.41 ms + (i - 1) * 11.768 ms, where i = 1 to 49

3.8 QAC Error Type

This 1 byte of error information is produced at SDPF only for each packet for which an anomaly is detected. This byte contains 8 fields, shown if Table 3.8-1, each of which is a flag.

Table 3.8-1 Error Fields TBD

A complete description is found in the Interface Control Document Between the SDPF and the TRMM Consumers.

3.9 Satellite Coordinates and Flight Modes

The TRMM satellite structural axes are defined so that +Z is the side where PR is mounted and is the direction normally pointed toward the nadir(straight down toward the Earth). +X is the side toward which the TMI and VIRS instruments are mounted, and the side toward which the TMI instruments takes measurements. Solar arrays are mounted on the + and - Y sides, and the +Y axis is such that the +X, +Y and +Z complete a right-hand system.

The two common flight modes for TRMM are +X forward and -X forward. With +X forward, the +X axis of the spacecraft is pointed along the velocity direction (i.e., in the direction the spacecraft is moving toward), +Y is pointed opposite the orbit normal (orbit spin axis), and +Z is toward the nadir. With -X forward the +X axis is opposite the velocity (i.e., in the direction the spacecraft is moving away from), +Y is along orbit normal, and +Z is still toward the nadir. The spacecraft spends about half its time in each of these two positions. It switches between +X and -X Forward every two to four weeks with a yaw maneuver taking about 20 minutes. This is done in order to keep the sun always in the -Y Hemisphere, as orbit precession and seasonal changes move the sun slowly above and below the orbit plane. (This maintains passive cooling and power design constraints). The flight mode for each scan is written in all Level 1B, 1C, 2A, and 2B satellite products in Scan Status, Current Spacecraft Orientation.

As discussed in the instrument scan geometry sections, the scan directions relative to the flight path change as the mode changes. For TMI, scans are always left to right in the +X spacecraft direction as the microwave antenna rotates about the +Z axis. Thus TMI scans are left to right looking forward along the ground track in the +X forward mode, and are right to left of the ground track in the -X forward mode. For VIRS, scans are right to left of the ground track direction in the +X forward mode, and left to right of the ground track direction in the -X forward mode. PR scans electronically the same direction as VIRS : right to left with +X forward and left to right of the flight path direction with -X forward.

4 Level 1A TRMM Microwave Imager (TMI)

4.1 Instrument and Scan Geometry

The TRMM Microwave Imager (TMI) is one of 5 instruments flown on the TRMM satellite. TMI is similar to the SSM/I instrument flown on the DMSP satellites with several key differences: (1) the addition of vertically and horizontally polarized 10 GHz channels, (2) the scan geometry is the same for every scan instead of alternating between an A scan and a B scan, and (3) there are about twice as many pixels per scan.

The TMI is a 9 channel, 5 frequency, linearly polarized, passive microwave radiometric system. The instrument measures atmospheric and surface brightness temperatures at 10.7, 19.4, 21.3, 37.0, and 85.5 GHz. Each frequency has one vertically (V) and one horizontally (H) polarized channel, except for the 21.3 GHz frequency, which has only vertical polarization. The 10.7, 19.4, 21.3, and 37.0 GHz channels are considered low resolution and the 85.5 GHz channels are considered high resolution. TMI has a conical scanning geometry, rotating continuously about a vertical axis, receiving upwelling radiation from 49? off nadir. Scene radiation is recorded from left to right looking in the +x direction. (See Section 3.9) over an annular sector of 130? about the sub-satellite track. The swath width is 758.5 km, covered by 104 low resolution pixels or 208 high resolution pixels. A complete description of the TMI instrument is available in the TMI Technical Description Document. Figure 4.1-1 shows a simplified version of the scan geometry for two scans. The centers of pixels 1, 3, 5, ... of the high resolution channels (unshaded circles) are collocated with the centers of pixels 1, 2, 3, ... of the low resolution channels (shaded circles).

Low resolution channels appear as shaded circles. High resolution channels appear as open circles in front of the shaded circles. The dots indicate continuation.

Figure 4.1-1 TMI Scan Geometry TBD

4.2 Design Considerations

The TMI formats for the Level-1A and Level-1B products were designed in consultation with Dr. James Shiue (the TMI instrument scientist), Dr. Christian Kummerow (a TMI algorithm scientist), and Mr. Ted Meyer, Mr. Doug Ilg and Dr. Brand Fortner (of EOSDIS). The Level-1A format was designed to minimize processing time and storage volume and to clearly demonstrate reversibility to Level-0. For a discussion of reversibility, see Appendix A. The Level-1A product is a simple concatenation of Level-0 data with a Header Record, which could easily be reversed back to Level-0. Level-1A remains in a binary format and is not in HDF. A detailed description of the Level-1A components is provided in Section 4.3.

The Level-1B format groups like data together. For example, high resolution channel data are in one object rather than, for example, channel, geolocation, and calibration data being interleaved in one object. The Level-1B format includes navigation and calibration information, which was requested by the instrument scientist for purposes of checking the calibration and navigation. Topographic data are not included in Level-1B design at the request of Dr. Kummerow to save disk space. A topographic data base will be provided which users of TSDIS may access. The Surface Type is also not included at the request of Dr. Kummerow since users would have different definitions of Surface Type. The Spacecraft Position and Velocity data are represented in 4-byte floating point precision. Spacecraft Geocentric Position should be represented to 10m accuracy, which equates to 6 significant digits since the earths radius is about 6,370,000 m. Satellite Local Zenith angle is given for every twentieth high resolution pixel to save disk space. Considerations were given to two existing formats, the "Wentz" format and the MSFC format, both currently used for SSM/I data. The TMI Level-1B format is presented in a Swath Structure and formatted in HDF.

Wentz vs TSDIS format The Level-1B format is based on the Wentz format, which is described in Users Manual SSM/I Antenna Temperature Tapes by Frank J. Wentz of Remote Sensing Systems, published in 1988. Since the Wentz format packs most elements, the scales and biases are listed in the Users Manual. The TSDIS format only packs a few elements, whose scales are included in the description of the element. Table 4.2-1 provides a mapping between the Wentz format and the corresponding elements in the TSDIS Level-1B format. Some of the differences in format are due to the difference in scanning geometry between SSM/I and TMI: SSM/I alternates between an A-scan and B-scan, but TMI has only one type of scan. Any references to B-scan by the Wentz format are omitted in the TSDIS format.

MSFC Data The Pathfinder Daily Antenna Temperature Files archived at MSFC were examined as a possible model for the TSDIS format but not used because their format does not use EOSDIS metadata, and has an unusual way to pack orbits into arrays. Row one of an array contains scan one for the orbit one, then a delimiter, then scan one for orbit two, then a delimiter, and so on for all the orbits in a day. Succeeding rows each contain one particular scan for all of the orbits.

Table 4.2-1 Comparison of Wentz with TSDIS Format TBD

4.3 1A11

The Level-1A product consists of two files: the Level-1A Product file and the SFDU header file. The Level-1A Product file, "1A-11," is a concatenation of Header record, Spacecraft Attitude packets, TMI Housekeeping packets, TMI Science Data packets, QACs and an MDUL. The SFDU header is a separate file whose format is specified in the Interface Control Document Between EOSDIS Core System (ECS) and TRMM Science Data and Information System (TSDIS) for the ECS Project. The data granule size is discussed in section 3. Sizing parameters used are:

nsec = the number of seconds in an average granule (includes the seconds in an orbit plus the seconds for 100 scans of overlap). See Sections 3-5 and 3-6. Table 4.3-1 describes 1A-11 and Table 4.3-2 is a description of the 1A-11 Header.

Please note that prior to June 1, 2008 all PR, TMI and VIRS binary 1A products were produced on an SGI (big-endian). PPS is currently producing the TMI and VIRS products on the Linux (little-endian) platform. The TMI and VIRS 1A file formats remain the same. The endian order of the data in the headers is written in NATIVE endian (little-endian) format. The CCSDS packets contained in the 1A files remain unchanged.

Table 4.3-1 Description of 1A-11 TBD

Table 4.3-2 Description of 1A11 Header TBD ** Spacecraft Time format described in the TRMM Telemetry and Command Handbook, section 3.3.2

4.4 Instrument Analysis Products

Instrument Analysis Data and Instrument Analysis Report are intermediate products that are by-products of the Level-1B processing. These products will be made available only to the instrument scientist to assess the health of the TMI. Since these products are not sent to EOSDIS, the format is not specified in this document.

5 Level 1A Visible and Infrared Scanner (VIRS)

5.1 Instrument and Scan Geometry

The Visible and Infrared Scanner (VIRS) sensor is one of the five instruments on the TRMM satellite. The VIRS instrument has a swath width of 720 km and a horizontal resolution of 2 km at nadir. VIRS is similar to the Advanced Very High Resolution Radiometer (AVHRR) now in operation on polar-orbital environmental satellites.

The VIRS measures radiance values in the channels shown in the Table 5.1-1. All five Channels will be in operation during daytime, but only Channels 3, 4 and 5 are in operation during nighttime. The VIRS is a cross-track scanning system and records scene radiation from right to left looking in the +x direction (See Section 3.9) over a scan angle of +/- 450 from the nadir (subpoint view). The swath width of 720 km is covered by 261 pixels and every scan has the same geometry. The Level-1B product stores radiance for every pixel measured.

Table 5.1-1 Channels where VIRS Measures Radiance Values TBD

5.2 Design Considerations

The VIRS formats for the Level-1A and Level-1B products were designed in consultation with Dr. William Barnes (VIRS Instrument Scientist), Mr. Ted Meyer, Mr. Doug Ilg and Dr. Brand Fortner (of EOSDIS). The Level-1A format was designed to minimize processing time and storage volume and to clearly demonstrate reversibility to Level-0. The Level-1A product is a simple concatenation of Level-0 data with a Header, which could easily be reversed back to Level-0. For a discussion of reversibility, see Appendix

A. Level-1A remains in a binary format and is not in HDF. A detailed description of the Level-1A components is provided in Section 5.3.

The Level-1B format groups like data together. For example, channel data are in one object rather than, for example, channel, geolocation, and calibration data being interleaved in one object. The Level-1B format includes navigation and calibration information, which was requested by Dr. Barnes. The Satellite and Solar Local Direction are supplied for every tenth pixel in a scan line to save space. Spacecraft Position and Velocity data are represented in 4-byte floating point precision. Spacecraft Geocentric Position is represented to 10m accuracy, which equates to 6 significant digits since the earths radius is about 6,370,000 m. Data elements are based on the NOAA AVHRR format. The VIRS Level-1B format is are presented in a Swath Structure and formatted in HDF

5.3 1A01

The Level-1A product consists of two files: the Level-1A Product file and the SFDU header file. The Level-1A Product file, "1A-01," is a concatenation of Header record, Spacecraft Attitude packets, VIRS Housekeeping Data packets, VIRS Science Data packets, QACs, and an MDUL. The SFDU header is a separate file whose format is specified in the Interface Control Document Between EOSDIS Core System (ECS) and TRMM Science Data and Information System (TSDIS) for the ECS Project. The data granule size is discussed in section 3. Sizing parameters used are:

nsec = the number of seconds in a granule. See Section 3-6. Tables 5.3-1 and 5.3-2 describe 1A-01.

Table 5.3-1 Description of 1A-01 TBD

Table 5.3-2 Description of 1A-01 Header TBD

** Spacecraft Time format described in the TRMM Telemetry and Command Handbook, Section 3.3.2

Please note that prior to June 1, 2008 all PR, TMI and VIRS binary 1A products were produced on an SGI (big-endian). PPS is currently producing the TMI and VIRS products on the Linux (little-endian) platform. The TMI and VIRS 1A file formats remain the same. The endian order of the data in the headers is written in NATIVE endian (little-endian) format. The CCSDS packets contained in the 1A files remain unchanged.

5.4 Instrument Analysis Products

Instrument Analysis Data and Instrument Analysis Report are intermediate products that are by-products of the Level-1B processing. These products will be made available only to the instrument scientist to assess the health of the VIRS. Since these products are not sent to EOSDIS, the format is not specified in this document.

6 Level 1A Precipitation Radar (PR)

6.1 Instrument and Scan Geometry

The Precipitation Radar (PR) is one of five instruments on the TRMM satellite. The PR is an active 13.8 GHz radar, recording energy reflected from atmospheric and surface targets. The PR electronically scans right to left looking in the +x direction (see Section 3.9) of the satellite every 0.6 s with a swath width of 215 km before August 7, 2001 and 247 km after August 24, 2001.

The complex scan geometry is represented in Figure 6.1-1. Each scan contains 49 rays sampled over an angular sector of 34. For a given ray, the satellite begins recording samples at a fixed distance from the satellite and records a certain number of samples every 125 m along the ray. The starting distance and the number of samples are different for each ray. Assuming the satellite altitude is 350 km, the sampling begins about 23 km above mean sea level and extends for a certain distance along the ray. This distance along the ray is 33.5 km at the two rays farthest from nadir, monotonically declining to 30.25 km at the two rays adjacent to nadir, and jumping to 34.75 km at the single nadir ray. The extra data in the nadir ray is known as "the mirror," because it records energy reflected not once from a target, but three times (ground to target to ground). Rays other than the nadir ray also sample "below" the surface. The purpose of this extension is "to see" below the surface to clearly detect the location of the surface.

The satellite saves data in three samples. Every other data point in the vertical is saved in the "normal sample," shown in Figure 6.1-2. Thus the normal sample has a spacing of 250 m along the ray. The mirror is contained in the normal sample. A subset of the remaining data points is saved in two oversamples: the "surface oversample" and the "rain echo oversample." Both oversamples have a spacing of 250 m along a ray, but a region with both normal sample and oversample has a spacing of 125 m. The PR determines which levels to save in the oversamples based on its on board determination of the surface bin. No data are saved as oversample in rays 1-10 and 40-49. Five levels are saved from rays 11-39 in the surface oversample. If the PR detects the surface in an oversample bin, the surface oversample is centered on the detected surface. If, on the other hand, the PR detects the surface in a normal sample bin, 3 oversample bins are above and 2 oversample bins are below the detected surface. In addition, 28 levels (immediately above the surface oversample) are saved from rays 20-30 in the rain echo oversample.

NASDA defines a range bin number, which is related to distance from the satellite along the ray. It starts at 1 roughly 327 km (381 km after August 24, 2001) from the satellite, increments by 1 roughly every 125 m, and increases to a maximum of 400 at roughly 377 km (431 km after August 24, 2001) from the satellite. The exact value for the starting distance is written in the 1B21 and 1C21 files in Ray Header, Starting Bin Distance for the nadir ray. Twice the exact value for the increment is written in the 1B21 and 1C21 files in Ray Header, Range Bin Size for any ray.

Figure 6.1-1 PR Scan Geometry TBD

Figure 6.1-2 Normal Sample Data Array TBD

6.2 Design Considerations

The Precipitation Radar (PR) formats for the Level-1A, Level-1B, and Level-1C products were designed in consultation with NASDA, CRL, Dr. Robert Meneghini (a PR algorithm scientist) and Mr. Ted Meyer, Mr. Doug Ilg and Dr. Brand Fortner (EOSDIS).

TSDIS, with guidance from Dr. Meneghini, created a PR synthetic data set in a binary format. The TSDIS Level-1B and Level-1C formats consist of the binary format augmented by satellite information and modified by information from NASDA. The TSDIS Level-1B and Level-1C formats are presented in a Swath Structure and formatted in HDF.

The satellite information includes navigation and calibration information, which was requested by the scientists. The Spacecraft Position and Velocity data are represented in 4-byte floating point precision. Spacecraft Geocentric Position is represented to 10m accuracy, which equates to 6 significant digits since the earths radius is about 6,370,000 m.

6.3 1A21

The Level-1A Product file, "1A-21," is a concatenation of a TSDIS Header, Spacecraft Attitude packets, PR Housekeeping Data packets, PR Science Data packets, Calibration Coefficients, QACs, and a Missing Data Units List (MDUL). Tables 6.3-1, Table 6.3-2, and Table 6.3-3 describe 1A-21. There is an additional file: a detached SFDU header. The SFDU header is described in the Interface Control Document Between EOSDIS Core System (ECS) and TRMM Science Data and Information System (TSDIS) for the ECS Project. The sizing parameters used are:

nsec = the number of seconds in a granule. See Section 3-6. Table 6.3-1 Description of 1A-21 TBD

Table 6.3-2 Description of 1A-21 Header TBD ** Spacecraft Time format described in the TRMM Telemetry and Command Handbook, section 3.3.2

Table 6.3-3 Description of the Modified PR Science Data Packet (APID 53) TBD

6.4 Instrument Analysis Products

Since NASDA will monitor the PR instrument and instrument analysis products will not be created in TSDIS, TSDIS will not supply a format for PR instrument analysis products.

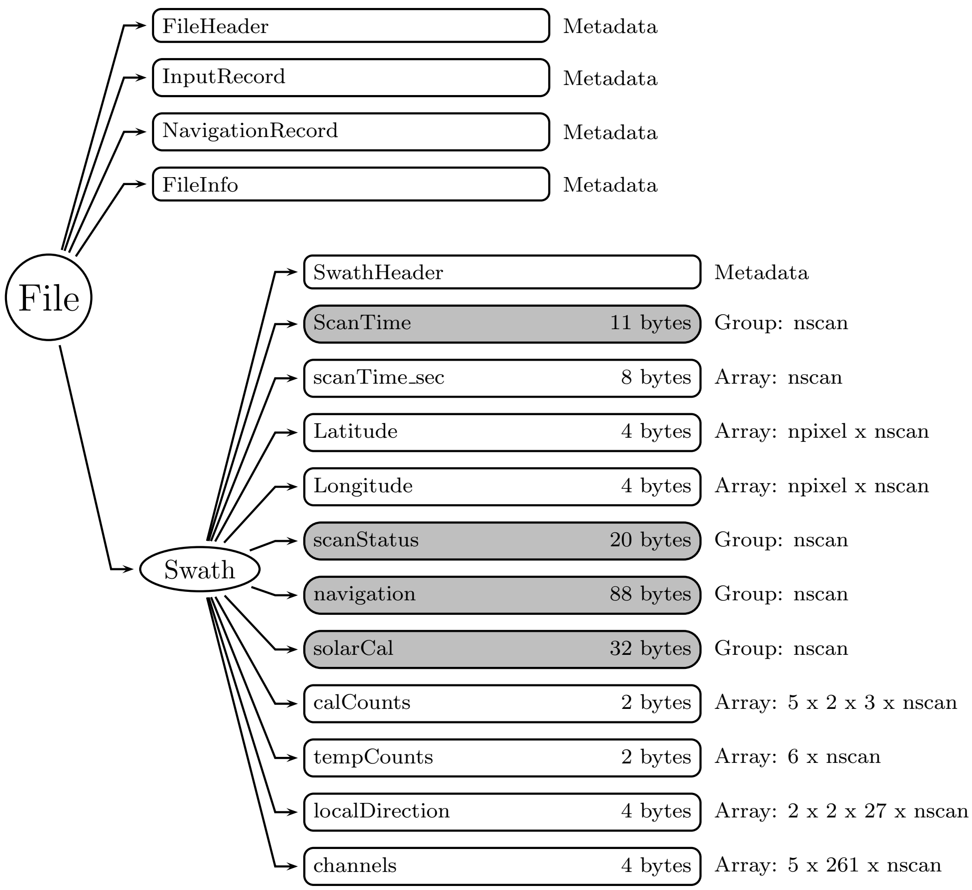


Figure 1: Data Format Structure for 1B01, VIRS Radiance

7 Standard TRMM Algorithms

7.1 1B01 - VIRS Radiance

The VIRS Level-1B Product, 1B01, "VIRS Radiance," is written in HDF. The following sizing parameter is used in describing these formats:

Dimension definitions:

nscan	var	Number of scans in the granule.
npixel	261	Number of pixels in each scan.

Figure 1 through Figure 5 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for TRMM Products for details.

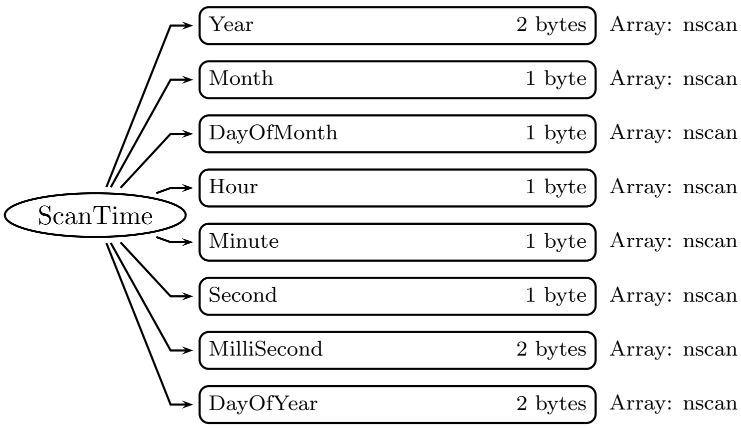


Figure 2: Data Format Structure for 1B01, ScanTime

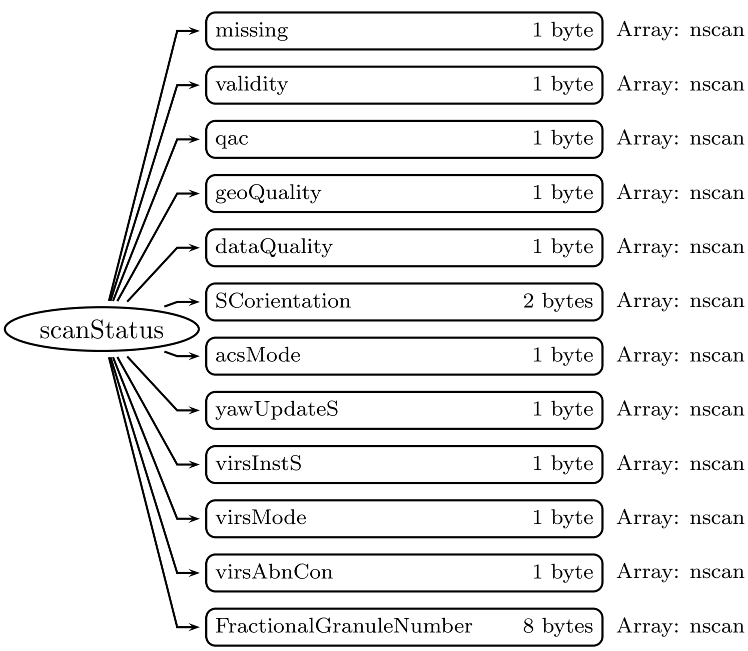


Figure 3: Data Format Structure for 1B01, scanStatus

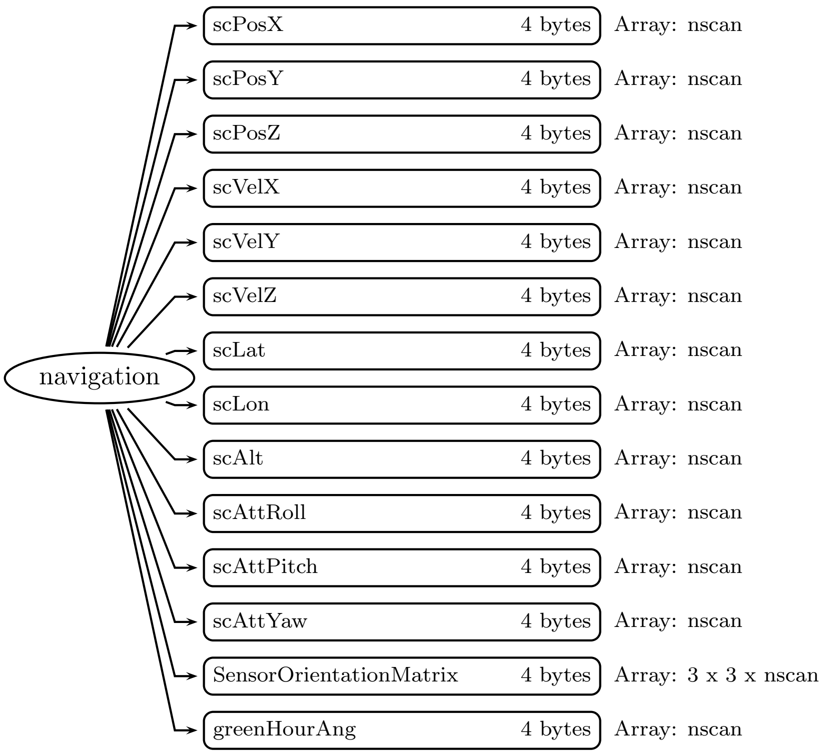


Figure 4: Data Format Structure for 1B01, navigation

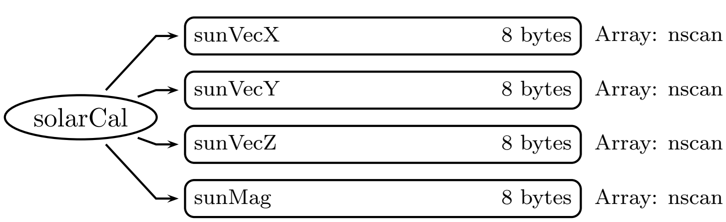


Figure 5: Data Format Structure for 1B01, solarCal

InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for TRMM Products for details.

NavigationRecord (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for TRMM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for TRMM Products for details.

Swath (Swath)

SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for TRMM Products for details.

ScanTime (Group)

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

scanTime_sec (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: npixel x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npixel x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scanStatus (Group)

missing (1-byte integer, array size: nscan):
Missing indicates whether information is contained in the scan data. The values are:

```
0 Scan data elements contain information
1 Scan was missing in the telemetry data
2 Scan data contains no elements with rain
```

validity (1-byte integer, array size: nscan):
Validity is a summary of status modes. If all status modes are routine, all bits in Validity = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. Validity does not assess data or geolocation quality. Validity is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

```
Bit Meaning if bit = 1
0   Spare (always 0)
1   Non-routine spacecraft orientation (2 or 3)
2   Non-routine ACS mode (other than 4)
3   Non-routine yaw update status (0 or 1)
4   Non-routine instrument status (Bit 0 = 0 or bit 1 = 0)
5   Non-routine QAC (non-zero)
6   Spare (always 0)
7   Spare (always 0)
```

qac (1-byte integer, array size: nscan):
The Quality and Accounting Capsule of the Science packet as it appears in Level-0 data. If no QAC is given in Level-0, which means no decoding errors occurred, QAC in this format has a value of zero.

geoQuality (1-byte integer, array size: nscan):
geoQuality is broken into 8 one-bit flags. Some flags represent problems but other flags are informational. Bits 0, 5, and 6 represent problems: 0 = 'good' quality and 1 = 'bad' quality. It is recommended not to use scans when any problem flag is 1. The informational flags have: 0 = routine conditions and 1 = non-routine conditions. Bit 0 is the most significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**(7-i). Note that good scans may have non-zero geoQuality. Each flag is listed below.

```
Bit Meaning if bit = 1
0   Grossly bad geolocation results:
    Spacecraft position vector magnitude outside range 6715 to 6790 km.
    Z component of midpoint of scan outside range -4100 to 4100 km.
    Distance from S/C to midpoint of scan outside range 500 to 750 km.
1   Unexpectedly large scan to scan jumps in geolocated positions in along and
    cross track directions for first, middle, and last pixels in each scan.
    Allowed deviation from nominal jump in along track motion = 3.0 km (first
    pixel), 3.0 km (middle pixel), and 3.0 km (last pixel).
    Allowed deviation from nominal jump in cross track motion = 3.0 km (first
    pixel), 3.0 km (middle pixel), and 3.0 km (last pixel).
    Bit set in normal mode only.
2   Scan to scan jumps in yaw, pitch, and roll exceed maximum values.  Values
    are :  yaw = 0.005 radians; pitch = 0.005 radians; roll = 0.005 radians.
    Bit set in normal control mode only.
3   In normal mode, yaw outside range (-0.005, 0.005) radians; pitch outside
    range (-0.005, 0.005) radians; roll outside range (-0.005, 0.005) radians.
4   Satellite undergoing maneuvers during which geolocation will be less
    accurate.
5   Summary QA flag for dataQuality:  Set to 1 if bit 0 is 1 or bit 6 is 1,
    i.e. Grossly bad or failed geolocation calculations.
    Science data use not recommended.
6   Geolocation calculations failed (fill values inserted in the per pixel
    geolocation products, but not in metadata).
7   Missing attitude data.  ACS data gap larger than 1.0 seconds.
      Pitch, roll, and yaw are interpolated or extrapolated from nearby data.
```

dataQuality (1-byte integer, array size: nscan):
dataQuality is a flag for overall scan quality. Unless this is 0, the scan data is meaningless to higher science processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

```
Bit  Meaning if bit = 1
0    missing
5    geoQuality indicates bad or missing values
6    validity is not normal
```

SCorientation (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the TMI scan. If +X is forward, SCorientation is 0. If -X is forward, SCorientation is 180. If -Y is forward, SCorientation is 90. Values range from 0 to 360 degrees. Special values are defined as:

-8003	Inertial
-8004	Unknown
-9999	Missing value

acsMode (1-byte integer, array size: nscan):

```
Value Meaning
0     Standby
1     Sun Acquire
2     Earth Acquire
3     Yaw Acquire
4     Nominal
5     Yaw Maneuver
6     Delta-H (Thruster)
7     Delta-V (Thruster)
8     CERES Calibration
```

yawUpdateS (1-byte integer, array size: nscan):

```
Value Meaning
0     Inaccurate
1     Indeterminate
2     Accurate
```

virsInstS (1-byte integer, array size: nscan):

```
Value Meaning
0     Day (no calibration occurring)
1     Night
2     Monitor Scan Stability
3     Day with Calibration
```

virsMode (1-byte integer, array size: nscan):

```
Value Meaning
0     mission mode
1     safehold mode
2     outgas mode
3     activation mode
```

virsAbnCon (1-byte integer, array size: nscan):
Bit 0 is the most significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**(8-i) - 1).

```
Bit Value Meaning
0   0     normal
    1     scan phase error
1   0     normal
    1     selftest error
2   0     normal
    1     thermal data missing
3   0     normal
    1     moon in space view
4   0     normal
    1     H/K data drop-out suspected
5   0     normal
    1     SV counts for channel 4 or 5 greater than
          L1B01_MIN_DNSV
6   0     not used
7   0     not used
```

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft's trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

navigation (Group)

scPosX (4-byte float, array size: nscan):
The x component of the position (m) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Geocentric Inertial Coordinates are also commonly known as Earth Centered Inertial coordinates. These coordinates will be True of Date (rather than Epoch 2000 which

are also commonly used), as interpolated from the data in the Flight Dynamics Facility ephemeris files generated for TRMM.

scPosY (4-byte float, array size: nscan):
The y component of the position (m) of the spacecraft in Geocentric Inertial Coordinates. See scPosX.

scPosZ (4-byte float, array size: nscan):
The z component of the position (m) of the spacecraft in Geocentric Inertial Coordinates. See scPosX.

scVelX (4-byte float, array size: nscan):
The x component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scVelY (4-byte float, array size: nscan):
The y component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scVelZ (4-byte float, array size: nscan):
The z component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scLat (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time.

scLon (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time.

scAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time.

scAttRoll (4-byte float, array size: nscan):
The satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

scAttPitch (4-byte float, array size: nscan):
The satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the

rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

scAttYaw (4-byte float, array size: nscan):
The satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

SensorOrientationMatrix (4-byte float, array size: 3 x 3 x nscan):
SensorOrientationMatrix is the rotation matrix from the instrument coordinate frame to Geocentric Inertial Coordinates at the Scan mid-Time. It is unitless.

greenHourAng (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates.

solarCal (Group)

sunVecX (8-byte float, array size: nscan):
Solar Position (X-component) (Geocentric Inertial Coord).

sunVecY (8-byte float, array size: nscan):
Solar Position (Y-component) (Geocentric Inertial Coord).

sunVecZ (8-byte float, array size: nscan):
Solar Position (Z-component) (Geocentric Inertial Coord).

sunMag (8-byte float, array size: nscan):
Sun-Earth Distance (m).

calCounts (2-byte integer, array size: 5 x 2 x 3 x nscan):
Raw calibration counts are given in four dimensions. The first dimension is the channel

number, the second dimension is the data word, the third dimension is blackbody, space view and solar diffuser, in that order, and the fourth dimension is the number of scans.

tempCounts (2-byte integer, array size: 6 x nscan):
Temperatures of the black body, primary and redundant, the radiant cooler temperatures, primary and redundant, the mirror temperature, and the electronics module temperature. All quantities have units of counts, and have minimum values of 0, and maximum values of 4095.

localDirection (4-byte float, array size: 2 x 2 x 27 x nscan):
Angles (degrees) to the satellite and sun from the IFOV pixel position on the earth are given in 4 dimensions. The first dimension is zenith and azimuth angles, in that order. The zenith angle is measured between the local pixel geodetic zenith and the direction to the satellite. The azimuth angle is measured clockwise from the local North direction around toward the local East direction. The second dimension is the object to which the directions point, namely the satellite and the sun, in that order. The third dimension is the pixel number. Angles are given only for every tenth pixel along a scan: pixel 1, 11, 21, ..., and 261. For the pixel dimension, Offset = 0 and Increment = -10. The fourth dimension is the scan number.

channels (4-byte float, array size: 5 x 261 x nscan):
Scene data for the channels, measured in Radiance ($mWcm^{-2}\mu m^{-1}sr^{-1}$). sr means steradian. The three dimensions are channel, pixel, and scan. The range, accuracy and wavelength for each channel are as follows:

Channel	Minimum	Maximum	Accuracy	Wavelength (micrometers)
1	0	65.5	10%	0.63
2	0	32.7	10%	1.6
3	0	0.111	2%	3.75
4	0	1.371	2%	10.8
5	0	1.15	2%	12.0

C Structure Header file:

```
#ifndef _TK_1B01_H_
#define _TK_1B01_H_

#ifndef _L1B01_SOLARCAL_
#define _L1B01_SOLARCAL_

typedef struct {
    double sunVecX;
    double sunVecY;
    double sunVecZ;
    double sunMag;
```

```
} L1B01_SOLARCAL;

#endif

#ifndef _L1B01_NAVIGATION_
#define _L1B01_NAVIGATION_

typedef struct {
    float scPosX;
    float scPosY;
    float scPosZ;
    float scVelX;
    float scVelY;
    float scVelZ;
    float scLat;
    float scLon;
    float scAlt;
    float scAttRoll;
    float scAttPitch;
    float scAttYaw;
    float SensorOrientationMatrix[3][3];
    float greenHourAng;
} L1B01_NAVIGATION;

#endif

#ifndef _L1B01_SCANSTATUS_
#define _L1B01_SCANSTATUS_

typedef struct {
    signed char missing;
    signed char validity;
    signed char qac;
    signed char geoQuality;
    signed char dataQuality;
    short SCorientation;
    signed char acsMode;
    signed char yawUpdateS;
    signed char virsInstS;
    signed char virsMode;
    signed char virsAbnCon;
    double FractionalGranuleNumber;
} L1B01_SCANSTATUS;
```

```
#endif

#ifndef _L1B01_SCANTIME_
#define _L1B01_SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
} L1B01_SCANTIME;

#endif

#ifndef _L1B01_SWATH_
#define _L1B01_SWATH_

typedef struct {
    L1B01_SCANTIME ScanTime;
    double scanTime_sec;
    float Latitude[261];
    float Longitude[261];
    L1B01_SCANSTATUS scanStatus;
    L1B01_NAVIGATION navigation;
    L1B01_SOLARCAL solarCal;
    short calCounts[3][2][5];
    short tempCounts[6];
    float localDirection[27][2][2];
    float channels[261][5];
} L1B01_SWATH;

#endif

#endif
```

Fortran Structure Header file:

```
STRUCTURE /L1B01_SOLARCAL/
```

```
    REAL*8 sunVecX
    REAL*8 sunVecY
    REAL*8 sunVecZ
    REAL*8 sunMag
END STRUCTURE

STRUCTURE /L1B01_NAVIGATION/
    REAL*4 scPosX
    REAL*4 scPosY
    REAL*4 scPosZ
    REAL*4 scVelX
    REAL*4 scVelY
    REAL*4 scVelZ
    REAL*4 scLat
    REAL*4 scLon
    REAL*4 scAlt
    REAL*4 scAttRoll
    REAL*4 scAttPitch
    REAL*4 scAttYaw
    REAL*4 SensorOrientationMatrix(3,3)
    REAL*4 greenHourAng
END STRUCTURE

STRUCTURE /L1B01_SCANSTATUS/
    BYTE missing
    BYTE validity
    BYTE qac
    BYTE geoQuality
    BYTE dataQuality
    INTEGER*2 SCorientation
    BYTE acsMode
    BYTE yawUpdateS
    BYTE virsInstS
    BYTE virsMode
    BYTE virsAbnCon
    REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /L1B01_SCANTIME/
    INTEGER*2 Year
    BYTE Month
    BYTE DayOfMonth
    BYTE Hour
```

```
    BYTE Minute
    BYTE Second
    INTEGER*2 MilliSecond
    INTEGER*2 DayOfYear
END STRUCTURE

STRUCTURE /L1B01_SWATH/
    RECORD /L1B01_SCANTIME/ ScanTime
    REAL*8 scanTime_sec
    REAL*4 Latitude(261)
    REAL*4 Longitude(261)
    RECORD /L1B01_SCANSTATUS/ scanStatus
    RECORD /L1B01_NAVIGATION/ navigation
    RECORD /L1B01_SOLARCAL/ solarCal
    INTEGER*2 calCounts(5,2,3)
    INTEGER*2 tempCounts(6)
    REAL*4 localDirection(2,2,27)
    REAL*4 channels(5,261)
END STRUCTURE
```

7.2 1B11 - TMI Brightness Temperatures

The TMI Level 1B Product, 1B11, "TMI Brightness Temperatures," is written as a Swath Structure. The following sections describe the structure and contents of the format.

Dimension definitions:

nscan	var	Number of scans in the granule.
nchan	9	Number of channels.
nchanlo	7	Number of channels.
nchanhi	2	Number of channels.
npixel	208	Number of high frequency pixels in each scan.
npixlo	104	Number of low frequency pixels in each scan.

Figure 6 through Figure 12 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for TRMM Products for details.

InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. Level 3 time averaged products have the

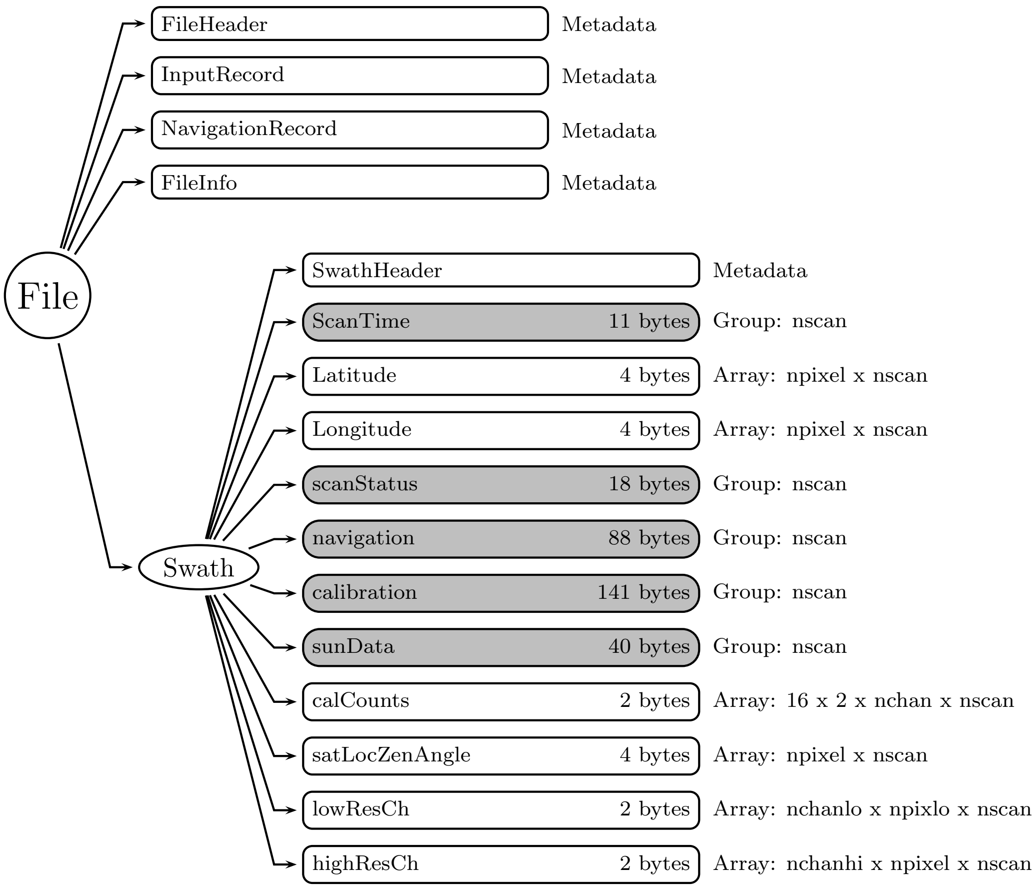


Figure 6: Data Format Structure for 1B11, TMI Brightness Temperatures

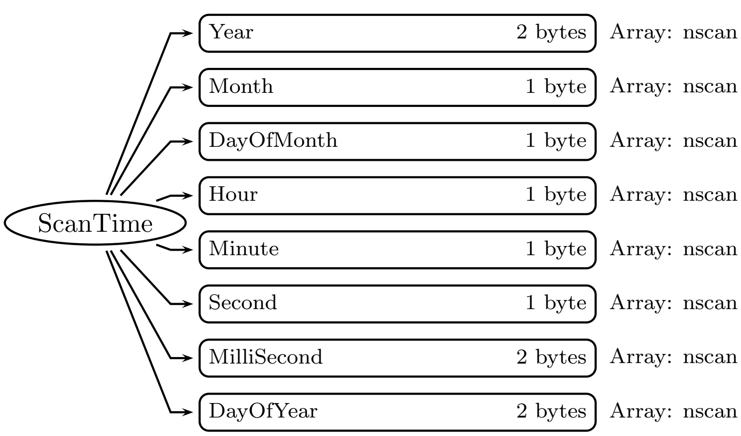


Figure 7: Data Format Structure for 1B11, ScanTime

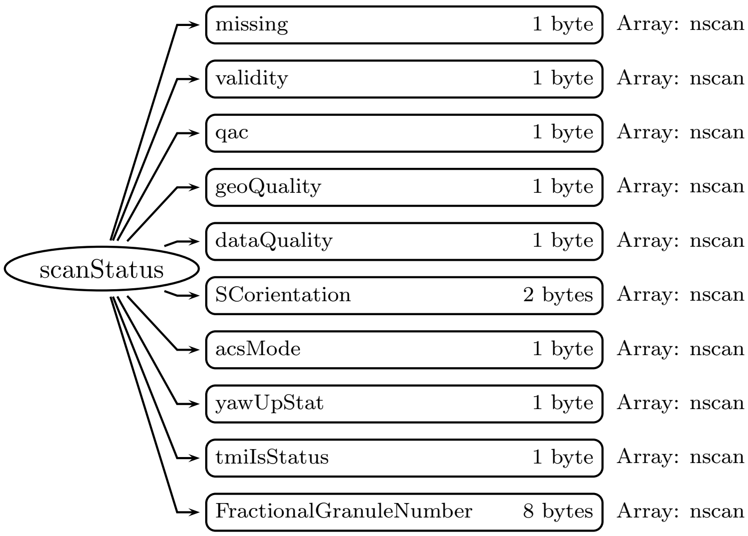


Figure 8: Data Format Structure for 1B11, scanStatus

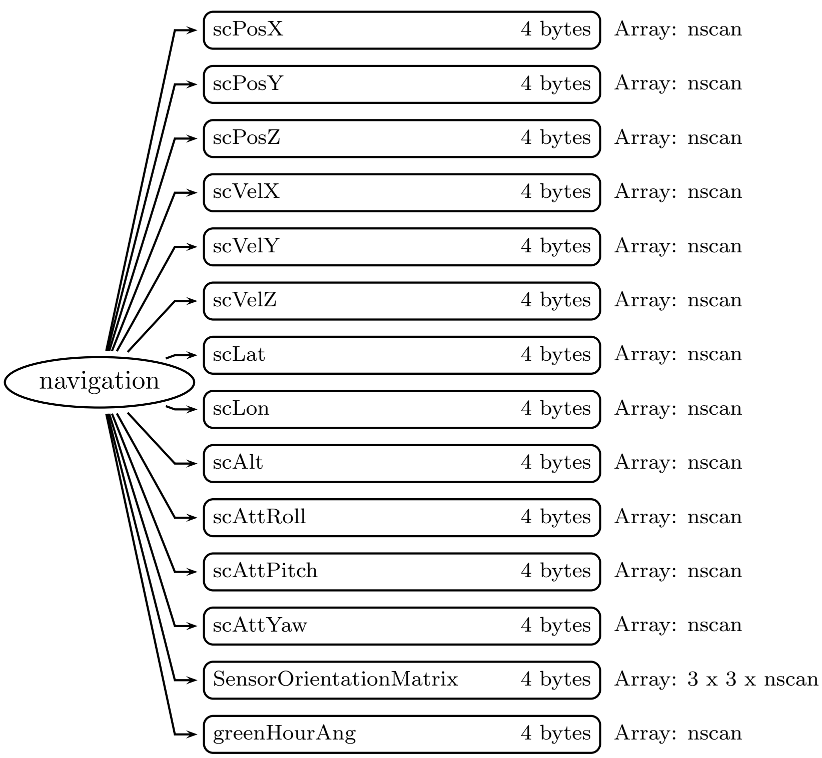
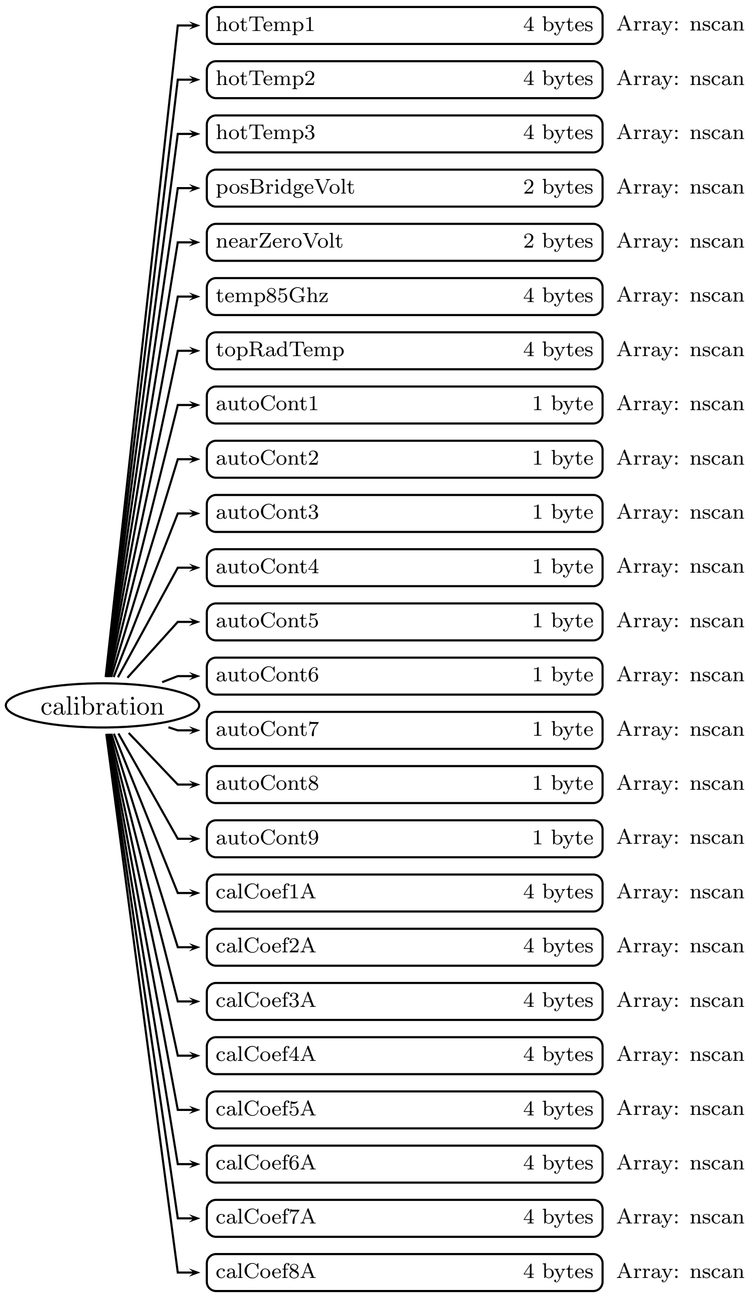


Figure 9: Data Format Structure for 1B11, navigation



continued on next figure

•
•
•

Figure 10: Data Format Structure for 1B11, calibration

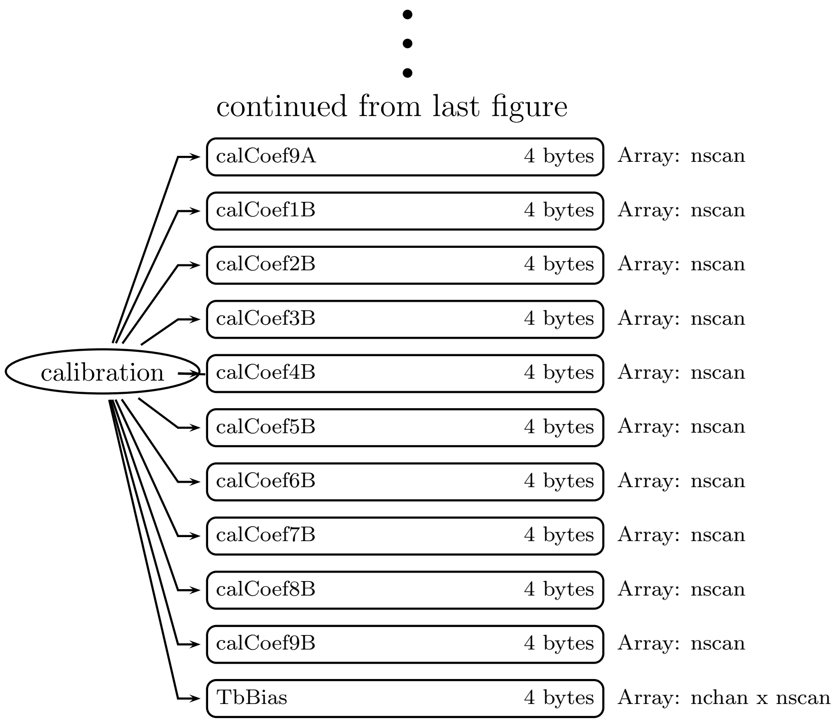


Figure 11: Data Format Structure for 1B11, calibration

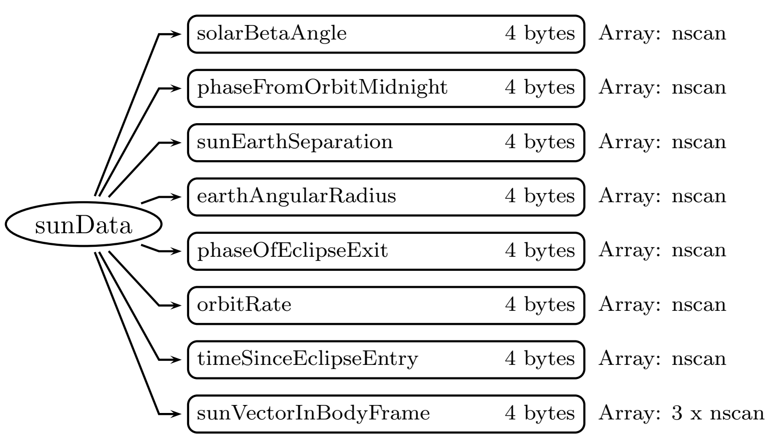


Figure 12: Data Format Structure for 1B11, sunData

same information separated into 3 groups since they have many inputs. See Metadata for TRMM Products for details.

NavigationRecord (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for TRMM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for TRMM Products for details.

Swath (Swath)

SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for TRMM Products for details.

ScanTime (Group)

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

Latitude (4-byte float, array size: npixel x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npixel x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scanStatus (Group)

missing (1-byte integer, array size: nscan):
Missing indicates whether information is contained in the scan data. The values are:

```
0 Scan data elements contain information
1 Scan was missing in the telemetry data
```

validity (1-byte integer, array size: nscan):
Validity is a summary of status modes. If all status modes are routine, all bits in Validity = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. Validity does not assess data or geolocation quality. Validity is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

```
Bit Meaning if bit = 1
0   Spare (always 0)
1   Non-routine spacecraft orientation (2 or 3)
2   Non-routine ACS mode (other than 4)
3   Non-routine yaw update status (0 or 1)
4   Non-routine instrument status (Bit 0 = 0 or bit 1 = 0)
5   Non-routine QAC (non-zero)

6   21 GHz Cold Count Flag (1 if Flag set)
7   Spare (always 0)
```

qac (1-byte integer, array size: nscan):
The Quality and Accounting Capsule of the Science packet as it appears in Level-0 data.

If no QAC is given in Level-0, which means no decoding errors occurred, QAC in this format has a value of zero.

geoQuality (1-byte integer, array size: nscan):
geoQuality is broken into 8 one-bit flags. Some flags represent problems but other flags are informational. Bits 0, 5, and 6 represent problems: 0 = 'good' quality and 1 = 'bad' quality. It is recommended not to use scans when any problem flag is 1. The informational flags have: 0 = routine conditions and 1 = non-routine conditions. Bit 0 is the most significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**(7-i). Note that good scans may have non-zero geoQuality. Each flag is listed below.

```
Bit Meaning if bit = 1
0   Grossly bad geolocation results:
    Spacecraft position vector magnitude outside range 6715 to 6790 km.
    Z component of midpoint of scan outside range -4100 to 4100 km.
    Distance from S/C to midpoint of scan outside range 500 to 750 km.
1   Unexpectedly large scan to scan jumps in geolocated positions in along and
    cross track directions for first, middle, and last pixels in each scan.
    Allowed deviation from nominal jump in along track motion = 3.0 km (first
    pixel), 3.0 km (middle pixel), and 3.0 km (last pixel).
    Allowed deviation from nominal jump in cross track motion = 3.0 km (first
    pixel), 3.0 km (middle pixel), and 3.0 km (last pixel).
    Bit set in normal mode only.
2   Scan to scan jumps in yaw, pitch, and roll exceed maximum values.  Values
    are :  yaw = 0.005 radians; pitch = 0.005 radians; roll = 0.005 radians.
    Bit set in normal control mode only.
3   In normal mode, yaw outside range (-0.005, 0.005) radians; pitch outside
    range (-0.005, 0.005) radians; roll outside range (-0.005, 0.005) radians.
4   Satellite undergoing maneuvers during which geolocation will be less
    accurate.
5   Summary QA flag for dataQuality:  Set to 1 if bit 0 is 1 or bit 6 is 1,
    i.e. Grossly bad or failed geolocation calculations.
    Science data use not recommended.
6   Geolocation calculations failed (fill values inserted in the per pixel
    geolocation products, but not in metadata).
7   Missing attitude data.  ACS data gap larger than 1.0 seconds.
      Pitch, roll, and yaw are interpolated or extrapolated from nearby data.
```

dataQuality (1-byte integer, array size: nscan):
dataQuality is a flag for overall scan quality. Unless this is 0, the scan data is meaningless to higher science processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

```
Bit  Meaning if bit = 1
```

```
0    missing
5    geoQuality indicates bad or missing values

6    validity bits 0-5 not all normal
```

SCorientation (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the TMI scan. If +X is forward, SCorientation is 0. If -X is forward, SCorientation is 180. If -Y is forward, SCorientation is 90. Values range from 0 to 360 degrees. Special values are defined as:

-8003	Inertial
-8004	Unknown
-9999	Missing value

acsMode (1-byte integer, array size: nscan):

```
Value Meaning
0     Standby
1     Sun Acquire
2     Earth Acquire
3     Yaw Acquire
4     Nominal
5     Yaw Maneuver
6     Delta-H (Thruster)
7     Delta-V (Thruster)
8     CERES Calibration
```

yawUpStat (1-byte integer, array size: nscan):

```
Value Meaning
0     Inaccurate
1     Indeterminate
2     Accurate
```

tmiIsStatus (1-byte integer, array size: nscan):
Bit 0 is the most significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**(8-i) - 1).

```
Bit Meaning
00  Receiver Status (1=ON, 0=OFF)
01  Spin-up Status (1=ON, 0=OFF)
02  Spare Command 1 Status
```

```
03  Spare Command 2 Status
04  1 Hz Clock Select (1=A, 0=B)

05  Spare
06  Spare Command 4 Status
07  Spare Command 5 Status
```

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft's trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

navigation (Group)

scPosX (4-byte float, array size: nscan):
The x component of the position (m) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Geocentric Inertial Coordinates are also commonly known as Earth Centered Inertial coordinates. These coordinates will be True of Date (rather than Epoch 2000 which are also commonly used), as interpolated from the data in the Flight Dynamics Facility ephemeris files generated for TRMM.

scPosY (4-byte float, array size: nscan):
The y component of the position (m) of the spacecraft in Geocentric Inertial Coordinates. See scPosX.

scPosZ (4-byte float, array size: nscan):
The z component of the position (m) of the spacecraft in Geocentric Inertial Coordinates. See scPosX.

scVelX (4-byte float, array size: nscan):
The x component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scVelY (4-byte float, array size: nscan):
The y component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scVelZ (4-byte float, array size: nscan):
The z component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scLat (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time.

scLon (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time.

scAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time.

scAttRoll (4-byte float, array size: nscan):
The satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

scAttPitch (4-byte float, array size: nscan):
The satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

scAttYaw (4-byte float, array size: nscan):
The satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

SensorOrientationMatrix (4-byte float, array size: 3 x 3 x nscan):

SensorOrientationMatrix is the rotation matrix from the instrument coordinate frame to Geocentric Inertial Coordinates at the Scan mid-Time. It is unitless.

greenHourAng (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates.

calibration (Group)

hotTemp1 (4-byte float, array size: nscan):
The physical temperatures, in degrees Kelvin, for the 3 temperature sensors attached to the hot load. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

hotTemp2 (4-byte float, array size: nscan):
The physical temperatures, in degrees Kelvin, for the 3 temperature sensors attached to the hot load. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

hotTemp3 (4-byte float, array size: nscan):
The physical temperatures, in degrees Kelvin, for the 3 temperature sensors attached to the hot load. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

posBridgeVolt (2-byte integer, array size: nscan):
The positive bridge voltage of the hot load bridge reference. Values range from 0 to 4095 volts. Special values are defined as:
-9999 Missing value

nearZeroVolt (2-byte integer, array size: nscan):
The near zero voltage of the hot load bridge reference. Values range from 0 to 4095 volts. Special values are defined as:
-9999 Missing value

temp85Ghz (4-byte float, array size: nscan):
The receiver shelf temperature of the 85.5 GHz channel. Values range from -273.15 to 126.85 Celsius. Special values are defined as:
-9999.9 Missing value

topRadTemp (4-byte float, array size: nscan):
The temperature of the top of the radiator channel. Values range from -273.15 to 126.85 Celsius. Special values are defined as:
-9999.9 Missing value

autoCont1 (1-byte integer, array size: nscan):
Automatic gain control for the 9 channels in counts. Values range from 0 to 15. Special values are defined as:
-99 Missing value

autoCont2 (1-byte integer, array size: nscan):
Automatic gain control for the 9 channels in counts. Values range from 0 to 15. Special values are defined as:
-99 Missing value

autoCont3 (1-byte integer, array size: nscan):
Automatic gain control for the 9 channels in counts. Values range from 0 to 15. Special values are defined as:
-99 Missing value

autoCont4 (1-byte integer, array size: nscan):
Automatic gain control for the 9 channels in counts. Values range from 0 to 15. Special values are defined as:
-99 Missing value

autoCont5 (1-byte integer, array size: nscan):
Automatic gain control for the 9 channels in counts. Values range from 0 to 15. Special values are defined as:
-99 Missing value

autoCont6 (1-byte integer, array size: nscan):
Automatic gain control for the 9 channels in counts. Values range from 0 to 15. Special values are defined as:
-99 Missing value

autoCont7 (1-byte integer, array size: nscan):
Automatic gain control for the 9 channels in counts. Values range from 0 to 15. Special values are defined as:
-99 Missing value

autoCont8 (1-byte integer, array size: nscan):
Automatic gain control for the 9 channels in counts. Values range from 0 to 15. Special values are defined as:
-99 Missing value

autoCont9 (1-byte integer, array size: nscan):
Automatic gain control for the 9 channels in counts. Values range from 0 to 15. Special values are defined as:
-99 Missing value

calCoef1A (4-byte float, array size: nscan):
Calibration coefficient A (degrees Kelvin / counts) for the 9 channels. Coefficient A for each channel is used in the following equation to convert counts, C, to antenna temperature, TA: TA = A C + B Values are in K. Special values are defined as:
-9999.9 Missing value

calCoef2A (4-byte float, array size: nscan):
Calibration coefficient A (degrees Kelvin / counts) for the 9 channels. Coefficient A for

each channel is used in the following equation to convert counts, C, to antenna temperature, TA: TA = A C + B Values are in K. Special values are defined as:
-9999.9 Missing value

calCoef3A (4-byte float, array size: nscan):
Calibration coefficient A (degrees Kelvin / counts) for the 9 channels. Coefficient A for each channel is used in the following equation to convert counts, C, to antenna temperature, TA: TA = A C + B Values are in K. Special values are defined as:
-9999.9 Missing value

calCoef4A (4-byte float, array size: nscan):
Calibration coefficient A (degrees Kelvin / counts) for the 9 channels. Coefficient A for each channel is used in the following equation to convert counts, C, to antenna temperature, TA: TA = A C + B Values are in K. Special values are defined as:
-9999.9 Missing value

calCoef5A (4-byte float, array size: nscan):
Calibration coefficient A (degrees Kelvin / counts) for the 9 channels. Coefficient A for each channel is used in the following equation to convert counts, C, to antenna temperature, TA: TA = A C + B Values are in K. Special values are defined as:
-9999.9 Missing value

calCoef6A (4-byte float, array size: nscan):
Calibration coefficient A (degrees Kelvin / counts) for the 9 channels. Coefficient A for each channel is used in the following equation to convert counts, C, to antenna temperature, TA: TA = A C + B Values are in K. Special values are defined as:
-9999.9 Missing value

calCoef7A (4-byte float, array size: nscan):
Calibration coefficient A (degrees Kelvin / counts) for the 9 channels. Coefficient A for each channel is used in the following equation to convert counts, C, to antenna temperature, TA: TA = A C + B Values are in K. Special values are defined as:
-9999.9 Missing value

calCoef8A (4-byte float, array size: nscan):
Calibration coefficient A (degrees Kelvin / counts) for the 9 channels. Coefficient A for each channel is used in the following equation to convert counts, C, to antenna temperature, TA: TA = A C + B Values are in K. Special values are defined as:
-9999.9 Missing value

calCoef9A (4-byte float, array size: nscan):
Calibration coefficient A (degrees Kelvin / counts) for the 9 channels. Coefficient A for each channel is used in the following equation to convert counts, C, to antenna temperature, TA: TA = A C + B Values are in K. Special values are defined as:
-9999.9 Missing value

calCoef1B (4-byte float, array size: nscan):
Calibration coefficient B (degrees Kelvin / counts) for the 9 channels. Coefficient B for

each channel is used in the following equation to convert counts, C, to antenna temperature, TA: TA = A C + B Values are in K. Special values are defined as:
-9999.9 Missing value

calCoef2B (4-byte float, array size: nscan):
Calibration coefficient B (degrees Kelvin / counts) for the 9 channels. Coefficient B for each channel is used in the following equation to convert counts, C, to antenna temperature, TA: TA = A C + B Values are in K. Special values are defined as:
-9999.9 Missing value

calCoef3B (4-byte float, array size: nscan):
Calibration coefficient B (degrees Kelvin / counts) for the 9 channels. Coefficient B for each channel is used in the following equation to convert counts, C, to antenna temperature, TA: TA = A C + B Values are in K. Special values are defined as:
-9999.9 Missing value

calCoef4B (4-byte float, array size: nscan):
Calibration coefficient B (degrees Kelvin / counts) for the 9 channels. Coefficient B for each channel is used in the following equation to convert counts, C, to antenna temperature, TA: TA = A C + B Values are in K. Special values are defined as:
-9999.9 Missing value

calCoef5B (4-byte float, array size: nscan):
Calibration coefficient B (degrees Kelvin / counts) for the 9 channels. Coefficient B for each channel is used in the following equation to convert counts, C, to antenna temperature, TA: TA = A C + B Values are in K. Special values are defined as:
-9999.9 Missing value

calCoef6B (4-byte float, array size: nscan):
Calibration coefficient B (degrees Kelvin / counts) for the 9 channels. Coefficient B for each channel is used in the following equation to convert counts, C, to antenna temperature, TA: TA = A C + B Values are in K. Special values are defined as:
-9999.9 Missing value

calCoef7B (4-byte float, array size: nscan):
Calibration coefficient B (degrees Kelvin / counts) for the 9 channels. Coefficient B for each channel is used in the following equation to convert counts, C, to antenna temperature, TA: TA = A C + B Values are in K. Special values are defined as:
-9999.9 Missing value

calCoef8B (4-byte float, array size: nscan):
Calibration coefficient B (degrees Kelvin / counts) for the 9 channels. Coefficient B for each channel is used in the following equation to convert counts, C, to antenna temperature, TA: TA = A C + B Values are in K. Special values are defined as:
-9999.9 Missing value

calCoef9B (4-byte float, array size: nscan):
Calibration coefficient B (degrees Kelvin / counts) for the 9 channels. Coefficient B for

each channel is used in the following equation to convert counts, C, to antenna temperature, TA: TA = A C + B Values are in K. Special values are defined as:
-9999.9 Missing value

TbBias (4-byte float, array size: nchan x nscan):
Bias in the brightness temperatures which is estimated due to instrument temperature fluctuations as a function of timeSinceEclipseEntry and solarBetaAngle. This bias is applied to reach the brightness temperature in this product. Values range from -10.0 to 10.0 K. In other words,

```
V6 Tb - TbBias = V7 Tb
```

sunData (Group)

solarBetaAngle (4-byte float, array size: nscan):
Sun direction elevation from the orbit plane, positive toward orbit normal which is given by the cross product of the spacecraft position and velocity vectors. Values range from -59.0 to 59.0 degrees. Special values are defined as:
-9999.9 Missing value

phaseFromOrbitMidnight (4-byte float, array size: nscan):
Phase angle of the Sun direction around the orbit plane, with zero phase in the direction of the Earth center from the spacecraft and positive toward the spacecraft velocity direction so the phase increases with time. Zero phase occurs at local orbit midnight, 90 degrees occurs with the spacecraft over the Earth's dawn terminator, 180 degrees occurs at local orbit noon, and -90 degrees occurs with the spacecraft over the Earth's dusk terminator. Values range from -180.0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value

sunEarthSeparation (4-byte float, array size: nscan):
The separation angle between the Sun and Earth directions from the spacecraft. Values range from 0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value

earthAngularRadius (4-byte float, array size: nscan):
The angle between the center of the Earth and the horizon edge. The sun is above the Earth horizon when the sunEarthSeparation is greater than the earthAngularRadius. Values range from 69.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

phaseOfEclipseExit (4-byte float, array size: nscan):
The estimated phaseFromOrbitMidnight where the spacecraft leaves the Earth shadow, based on the instantaneous solarBetaAngle and earthAngularRadius. Values range from 45.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

orbitRate (4-byte float, array size: nscan):
The instantaneous angular rate of the spacecraft around the orbit. Values range from

0.064 to 0.07 degrees/s. Special values are defined as:
-9999.9 Missing value

timeSinceEclipseEntry (4-byte float, array size: nscan):
The estimated duration in seconds since the last entry into the Earth's shadow. Values range from 0 to 5600.0 s. Special values are defined as:
-9999.9 Missing value

sunVectorInBodyFrame (4-byte float, array size: 3 x nscan):
The unit sun vector direction in the TMI instrument body coordinate frame, defined such that +Z is nominally toward the Earth and gives the instrument spin axis, and data is collected nominally centered about the +X direction. Values range from 0 to 1.0. Special values are defined as:
-9999.9 Missing value

calCounts (2-byte integer, array size: 16 x 2 x nchan x nscan):
Calibration measurements, in counts. The dimensions are: samples, load, channel, and scan. The sample dimension has a maximum of 16. The load dimension has first hot load and then cold sky. The low resolution channels (1-7) have 8 samples (the remaining 8 elements in the array are not used for each low resolution channel) and the high resolution channels (8 - 9) have 16 samples.

satLocZenAngle (4-byte float, array size: npixel x nscan):
The angle, in degrees, between the local pixel geodetic zenith and the direction to the satellite.

lowResCh (2-byte integer, array size: nchanlo x npixlo x nscan):
Brightness temperature (K) reduced by 100 K, multiplied by 100, and stored as a 2-byte integer, i.e.

```
Stored value = ( T - 100 K ) * 100
```

The dimensions are: channel, pixel, scan. The pixel dimension has Offset = 0 and Increment = -2. The following channels are included:

Channel	Frequency	Polarization	Minimum	Maximum
1	10 GHz	Vertica1	33	320
2	10 GHz	Horizontal	66	320
3	19 GHz	Vertical	133	320
4	19 GHz	Horizontal	80	320
5	21 GHz	Vertical	133	320
6	37 GHz	Vertical	133	320
7	37 GHz	Horizontal	112	320

highResCh (2-byte integer, array size: nchanhi x npixel x nscan):
Brightness temperature (K) reduced by 100 K, multiplied by 100, and stored as a 2-byte integer, i.e.

```
Stored value = ( T - 100 K ) * 100
```

The dimensions are: channel, pixel, scan. The following channels are included:

Channel	Frequency	Polarization	Minimum	Maximum
8	85 GHz	Vertical	70	320
9	85 GHz	Horizontal	70	320

C Structure Header file:

```
#ifndef _TK_1B11_H_
#define _TK_1B11_H_

#ifndef _L1B11_SUNDATA_
#define _L1B11_SUNDATA_

typedef struct {
    float solarBetaAngle;
    float phaseFromOrbitMidnight;
    float sunEarthSeparation;
    float earthAngularRadius;
    float phaseOfEclipseExit;
    float orbitRate;
    float timeSinceEclipseEntry;
    float sunVectorInBodyFrame[3];
} L1B11_SUNDATA;

#endif

#ifndef _L1B11_CALIBRATION_
#define _L1B11_CALIBRATION_

typedef struct {
    float hotTemp1;
    float hotTemp2;
    float hotTemp3;
    short posBridgeVolt;
    short nearZeroVolt;
    float temp85Ghz;
    float topRadTemp;
    signed char autoCont1;
    signed char autoCont2;
    signed char autoCont3;
```

```
    signed char autoCont4;
    signed char autoCont5;
    signed char autoCont6;
    signed char autoCont7;
    signed char autoCont8;
    signed char autoCont9;
    float calCoef1A;
    float calCoef2A;
    float calCoef3A;
    float calCoef4A;
    float calCoef5A;
    float calCoef6A;
    float calCoef7A;
    float calCoef8A;
    float calCoef9A;
    float calCoef1B;
    float calCoef2B;
    float calCoef3B;
    float calCoef4B;
    float calCoef5B;
    float calCoef6B;
    float calCoef7B;
    float calCoef8B;
    float calCoef9B;
    float TbBias[9];
} L1B11_CALIBRATION;

#endif

#ifndef _L1B11_NAVIGATION_
#define _L1B11_NAVIGATION_

typedef struct {
    float scPosX;
    float scPosY;
    float scPosZ;
    float scVelX;
    float scVelY;
    float scVelZ;
    float scLat;
    float scLon;
    float scAlt;
    float scAttRoll;
```

```
    float scAttPitch;
    float scAttYaw;
    float SensorOrientationMatrix[3][3];
    float greenHourAng;
} L1B11_NAVIGATION;

#endif

#ifndef _L1B11_SCANSTATUS_
#define _L1B11_SCANSTATUS_

typedef struct {
    signed char missing;
    signed char validity;
    signed char qac;
    signed char geoQuality;
    signed char dataQuality;
    short SCorientation;
    signed char acsMode;
    signed char yawUpStat;
    signed char tmiIsStatus;
    double FractionalGranuleNumber;
} L1B11_SCANSTATUS;

#endif

#ifndef _L1B11_SCANTIME_
#define _L1B11_SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
} L1B11_SCANTIME;

#endif

#ifndef _L1B11_SWATH_
```

```
#define _L1B11_SWATH_

typedef struct {
    L1B11_SCANTIME ScanTime;
    float Latitude[208];
    float Longitude[208];
    L1B11_SCANSTATUS scanStatus;
    L1B11_NAVIGATION navigation;
    L1B11_CALIBRATION calibration;
    L1B11_SUNDATA sunData;
    short calCounts[9][2][16];
    float satLocZenAngle[208];
    float lowResCh[104][7];
    float highResCh[208][2];
} L1B11_SWATH;

#endif

#endif
```

Fortran Structure Header file:

```
STRUCTURE /L1B11_SUNDATA/
    REAL*4 solarBetaAngle
    REAL*4 phaseFromOrbitMidnight
    REAL*4 sunEarthSeparation
    REAL*4 earthAngularRadius
    REAL*4 phaseOfEclipseExit
    REAL*4 orbitRate
    REAL*4 timeSinceEclipseEntry
    REAL*4 sunVectorInBodyFrame(3)
END STRUCTURE

STRUCTURE /L1B11_CALIBRATION/
    REAL*4 hotTemp1
    REAL*4 hotTemp2
    REAL*4 hotTemp3
    INTEGER*2 posBridgeVolt
    INTEGER*2 nearZeroVolt
    REAL*4 temp85Ghz
    REAL*4 topRadTemp
    BYTE autoCont1
    BYTE autoCont2
```

```
    BYTE autoCont3
    BYTE autoCont4
    BYTE autoCont5
    BYTE autoCont6
    BYTE autoCont7
    BYTE autoCont8
    BYTE autoCont9
    REAL*4 calCoef1A
    REAL*4 calCoef2A
    REAL*4 calCoef3A
    REAL*4 calCoef4A
    REAL*4 calCoef5A
    REAL*4 calCoef6A
    REAL*4 calCoef7A
    REAL*4 calCoef8A
    REAL*4 calCoef9A
    REAL*4 calCoef1B
    REAL*4 calCoef2B
    REAL*4 calCoef3B
    REAL*4 calCoef4B
    REAL*4 calCoef5B
    REAL*4 calCoef6B
    REAL*4 calCoef7B
    REAL*4 calCoef8B
    REAL*4 calCoef9B
    REAL*4 TbBias(9)
END STRUCTURE

STRUCTURE /L1B11_NAVIGATION/
    REAL*4 scPosX
    REAL*4 scPosY
    REAL*4 scPosZ
    REAL*4 scVelX
    REAL*4 scVelY
    REAL*4 scVelZ
    REAL*4 scLat
    REAL*4 scLon
    REAL*4 scAlt
    REAL*4 scAttRoll
    REAL*4 scAttPitch
    REAL*4 scAttYaw
    REAL*4 SensorOrientationMatrix(3,3)
    REAL*4 greenHourAng
```

```
END STRUCTURE

STRUCTURE /L1B11_SCANSTATUS/
    BYTE missing
    BYTE validity
    BYTE qac
    BYTE geoQuality
    BYTE dataQuality
    INTEGER*2 SCorientation
    BYTE acsMode
    BYTE yawUpStat
    BYTE tmiIsStatus
    REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /L1B11_SCANTIME/
    INTEGER*2 Year
    BYTE Month
    BYTE DayOfMonth
    BYTE Hour
    BYTE Minute
    BYTE Second
    INTEGER*2 MilliSecond
    INTEGER*2 DayOfYear
END STRUCTURE

STRUCTURE /L1B11_SWATH/
    RECORD /L1B11_SCANTIME/ ScanTime
    REAL*4 Latitude(208)
    REAL*4 Longitude(208)
    RECORD /L1B11_SCANSTATUS/ scanStatus
    RECORD /L1B11_NAVIGATION/ navigation
    RECORD /L1B11_CALIBRATION/ calibration
    RECORD /L1B11_SUNDATA/ sunData
    INTEGER*2 calCounts(16,2,9)
    REAL*4 satLocZenAngle(208)
    REAL*4 lowResCh(7,104)
    REAL*4 highResCh(2,208)
END STRUCTURE
```

7.3 1B21 - PR Power

The PR Level-1B product, 1B21, "PR Power," is written as a Swath Structure. See Section 2 in HDF. The following sections describe the structure and contents of the format.

Dimension definitions:

nscan	var	Number of scans in the granule.
nray	49	Number of angle bins in each scan.

Figure 13 through Figure 20 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for TRMM Products for details.

InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for TRMM Products for details.

NavigationRecord (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for TRMM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for TRMM Products for details.

JAXAInfo (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by PR algorithms only. See Metadata for TRMM Products for details.

pr_cal_coef (Group)

transCoef (4-byte float, array size: 1):
Transmission coefficient.

receptCoef (4-byte float, array size: 1):
Reception coefficient.

fcifIOchar (4-byte float, array size: 16):
FCIF I/O Characteristics.

ray_header (Group)

rayStart (2-byte integer, array size: nray):
Starting range bin number of Normal sample.

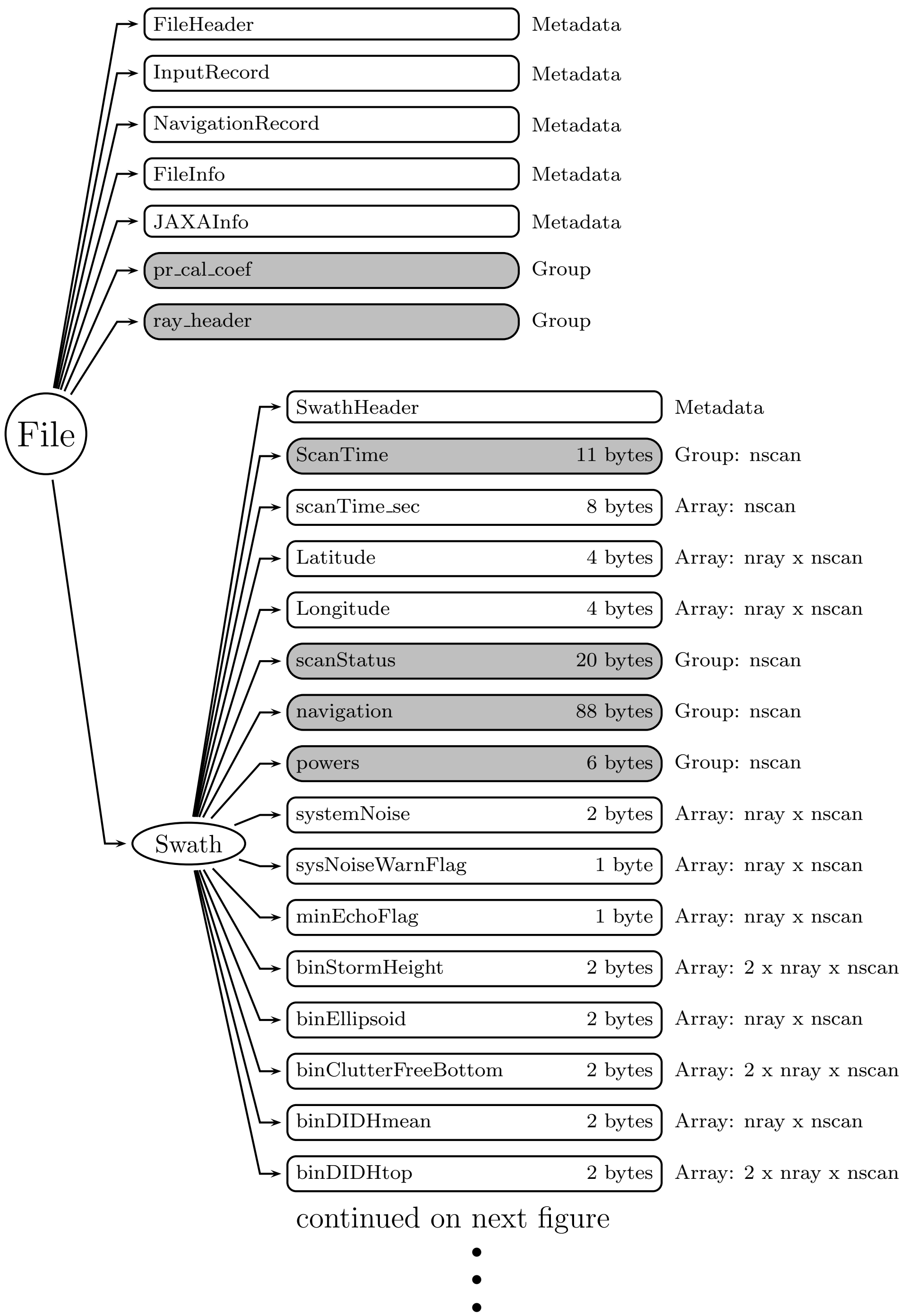


Figure 13: Data Format Structure for 1B21, PR Power

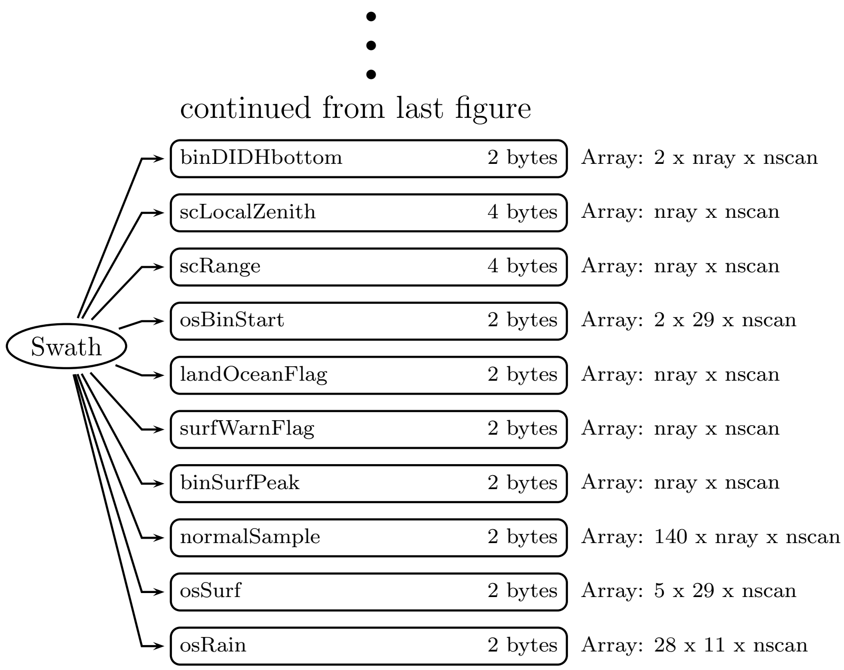


Figure 14: Data Format Structure for 1B21, PR Power

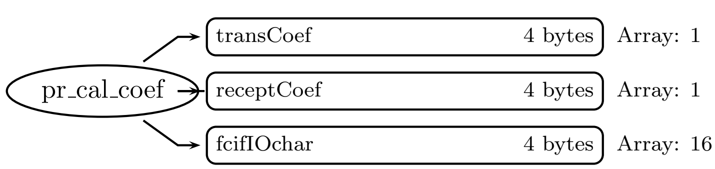


Figure 15: Data Format Structure for 1B21, pr_cal_coef

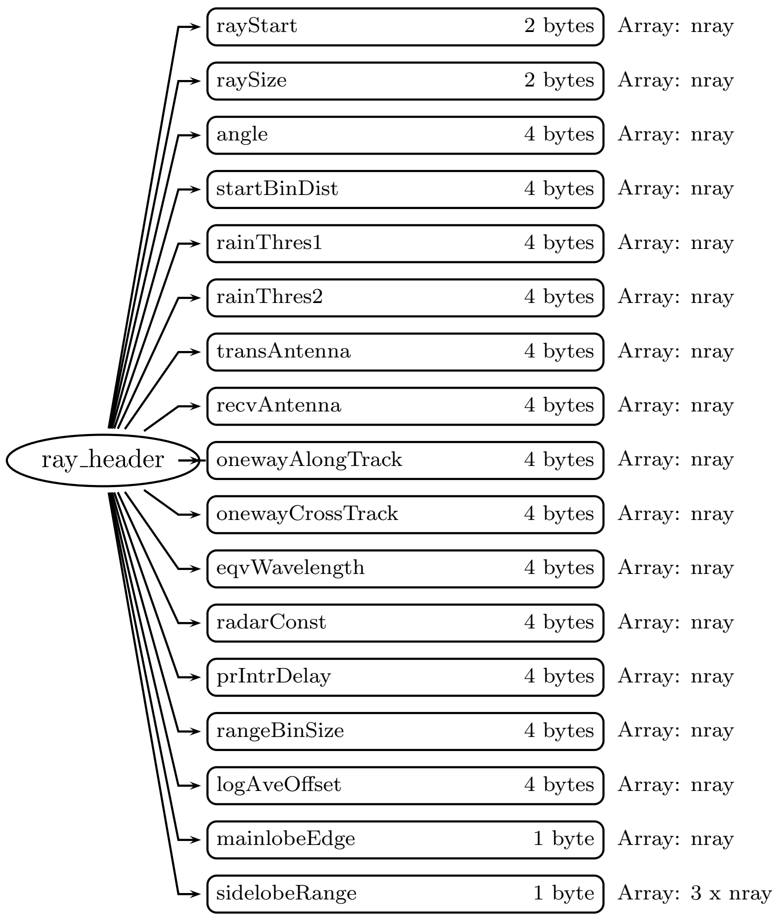


Figure 16: Data Format Structure for 1B21, ray_header

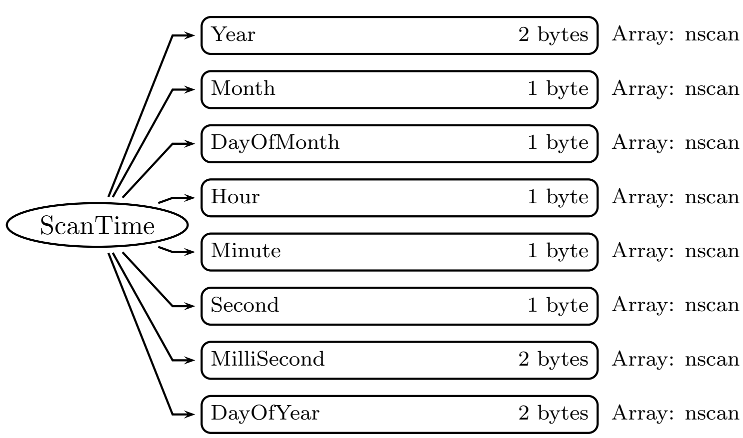


Figure 17: Data Format Structure for 1B21, ScanTime

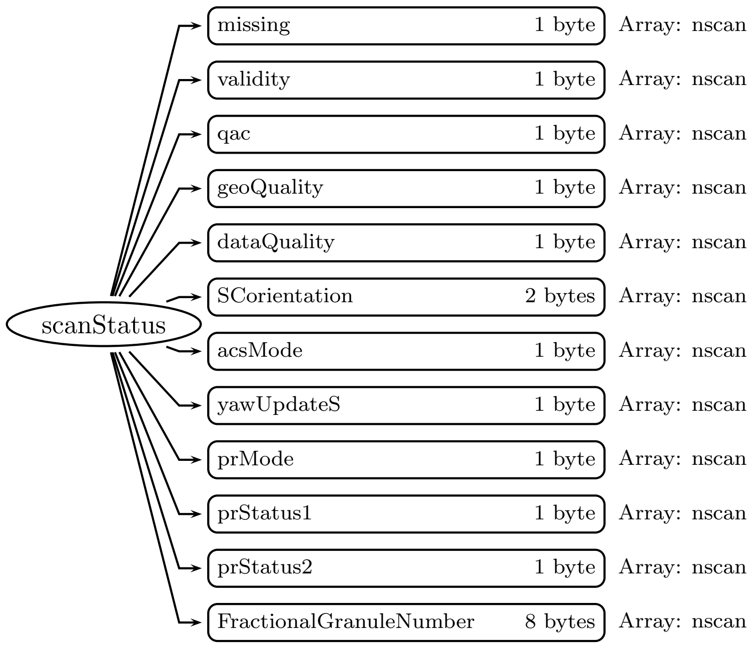


Figure 18: Data Format Structure for 1B21, scanStatus

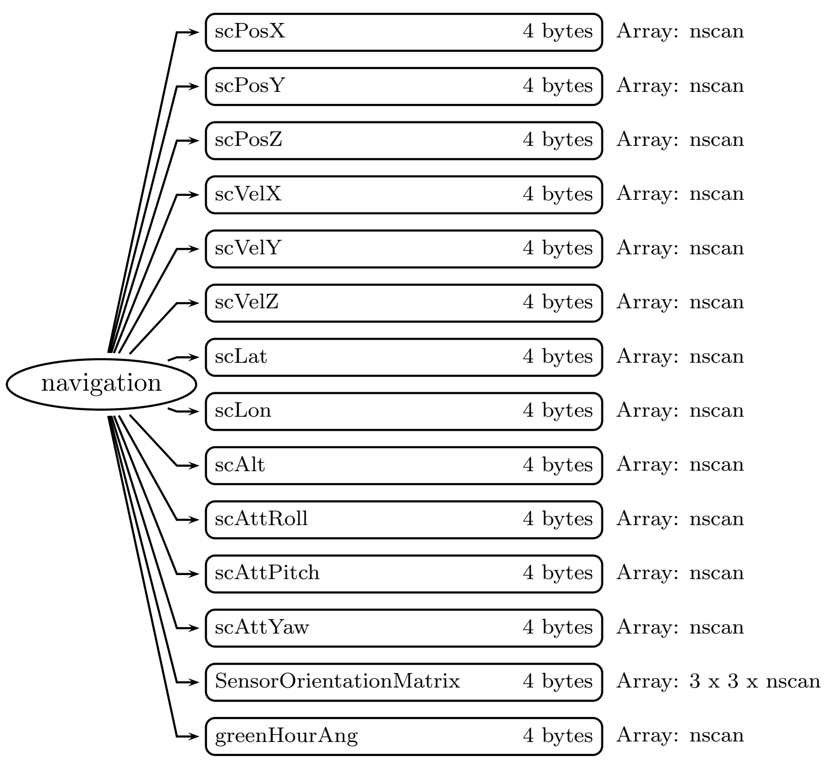


Figure 19: Data Format Structure for 1B21, navigation

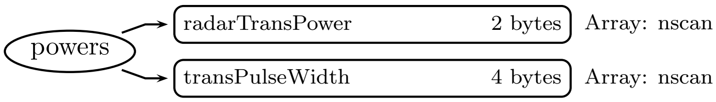


Figure 20: Data Format Structure for 1B21, powers

raySize (2-byte integer, array size: nray):
Number of Normal samples in the ray.

angle (4-byte float, array size: nray):
Angle (degrees) of the ray from nadir. The sign of the angle is consistent with the sensor y-axis, i.e., the angle is positive to the right of the direction of travel if the spacecraft is in normal mode.

startBinDist (4-byte float, array size: nray):
Distance (m) between the satellite and the starting bin number of the Normal sample for the ray.

rainThres1 (4-byte float, array size: nray):
Threshold used in minimum echo test (unitless). Value set by JAXA.

rainThres2 (4-byte float, array size: nray):
Threshold used in minimum echo test (unitless). Value set by JAXA.

transAntenna (4-byte float, array size: nray):
Transmitted radar antenna effectiveness (dB).

recvAntenna (4-byte float, array size: nray):
Received radar antenna effectiveness (dB).

onewayAlongTrack (4-byte float, array size: nray):
Radar beamwidth (radians) at the point transmitted power reaches one half of peak power in the along-track direction.

onewayCrossTrack (4-byte float, array size: nray):
Radar beamwidth (radians) at the point transmitted power reaches one half of peak power along the corss-track.

eqvWavelength (4-byte float, array size: nray):
Equivalent wavelength (m).

radarConst (4-byte float, array size: nray):
Radar constant dC (units are dB), which relates Received Power to Radar Reflectivity. dC depends on angle. ADD EQUATIONS.

prIntrDelay (4-byte float, array size: nray):
The time (seconds) between when echo returns at antenna and when echo is recorded in onboard processor.

rangeBinSize (4-byte float, array size: nray):
The vertical resolution of Normal sample bin (250 m).

logAveOffset (4-byte float, array size: nray):
The offset value (dB) between logarithmic average and normal average (+2.507dB).

mainlobeEdge (1-byte integer, array size: nray):
Absolute value of the difference in Range Bin Numbers between the detected surface and the edge of the clutter from the mainlobe.

sidelobeRange (1-byte integer, array size: 3 x nray):
Absolute value of the difference in Range Bin Numbers between the detected surface and the clutter psition from the sidelobe. A zero means no clutter indicated in this field since less than 3 bins contained significant clutter.

Swath (Swath)

SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for TRMM Products for details.

ScanTime (Group)

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

scanTime_sec (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: nray x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: nray x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scanStatus (Group)

missing (1-byte integer, array size: nscan):
Missing indicates whether information is contained in the scan data. The values are:

```
0 Scan data elements contain information
1 Scan was missing in the telemetry data
2 Scan data contains no elements with rain
```

validity (1-byte integer, array size: nscan):
Validity is a summary of status modes. If all status modes are routine, all bits in Validity = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. Validity does not assess data or geolocation quality. Validity is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

```
Bit Meaning if bit = 1
0   Spare (always 0)
1   Non-routine spacecraft orientation (2 or 3)
2   Non-routine ACS mode (other than 4)
3   Non-routine yaw update status (0 or 1)
4   Non-routine instrument status (other than 1)
5   Non-routine QAC (non-zero)
6   Spare (always 0)
7   Spare (always 0)
```

qac (1-byte integer, array size: nscan):
The Quality and Accounting Capsule of the Science packet as it appears in Level-0 data. If no QAC is given in Level-0, which means no decoding errors occurred, QAC in this format has a value of zero.

geoQuality (1-byte integer, array size: nscan):
Geolocation quality is a summary of geolocation quality in the scan. A zero integer value indicates 'good' geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

```
Bit Meaning if bit = 1
0   latitude limit error
1   geolocation
2   attitude change rate limit error
3   attitude limit error
4   satellite undergoing maneuvers
5   using predictive orbit data
6   geolocation calculation error
7   not used
```

dataQuality (1-byte integer, array size: nscan):
Data quality is a summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

```
Bit Meaning if bit = 1
0   missing
5   Geolocation Quality is not normal
6   Validity is not normal
```

SCorientation (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the TMI scan. If +X is forward, SCorientation is 0. If -X is forward, SCorientation is 180. If -Y is forward, SCorientation is 90. Values range from 0 to 360 degrees. Special values are defined as:

-8003	Inertial
-8004	Unknown
-9999	Missing value

acsMode (1-byte integer, array size: nscan):

```
Value Meaning
```

```
0     Standby
1     Sun Acquire
2     Earth Acquire
3     Yaw Acquire
4     Nominal
5     Yaw Maneuver
6     Delta-H (Thruster)
7     Delta-V (Thruster)
8     CERES Calibration
```

yawUpdateS (1-byte integer, array size: nscan):

```
Value Meaning
0     Inaccurate
1     Indeterminate
2     Accurate
```

prMode (1-byte integer, array size: nscan):

```
Value Meaning
1     Observation Mode
2     Other Mode
```

prStatus1 (1-byte integer, array size: nscan):
This status is a warning for scan data. Unless this is 0, the scan data may include a little questionable value though it is not a problem (such as break of caution limit). This field is used only for NASDA's data analysis.

prStatus2 (1-byte integer, array size: nscan):
Initialization in Onboard Surface Search Algorithm.

```
Value Meaning
0     Not initialized
1     Initialized
```

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft's trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

navigation (Group)

scPosX (4-byte float, array size: nscan):
The x component of the position (m) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Geocentric Inertial Coordinates are also commonly known as Earth Centered Inertial coordinates. These coordinates will be True of Date (rather than Epoch 2000 which are also commonly used), as interpolated from the data in the Flight Dynamics Facility ephemeris files generated for TRMM.

scPosY (4-byte float, array size: nscan):
The y component of the position (m) of the spacecraft in Geocentric Inertial Coordinates. See scPosX.

scPosZ (4-byte float, array size: nscan):
The z component of the position (m) of the spacecraft in Geocentric Inertial Coordinates. See scPosX.

scVelX (4-byte float, array size: nscan):
The x component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scVelY (4-byte float, array size: nscan):
The y component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scVelZ (4-byte float, array size: nscan):
The z component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scLat (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time.

scLon (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time.

scAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time.

scAttRoll (4-byte float, array size: nscan):
The satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

scAttPitch (4-byte float, array size: nscan):
The satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

scAttYaw (4-byte float, array size: nscan):
The satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

SensorOrientationMatrix (4-byte float, array size: 3 x 3 x nscan):
SensorOrientationMatrix is the rotation matrix from the instrument coordinate frame to Geocentric Inertial Coordinates at the Scan mid-Time. It is unitless.

greenHourAng (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates.

powers (Group)

radarTransPower (2-byte integer, array size: nscan):
The total (sum) power of 128 SSPA elements corrected with SSPA temperature in orbit, based on temperature test data of SSPA transmission power. The units are dBm * 100. For this variable, the TSDIS Toolkit does not provide scaling.

transPulseWidth (4-byte float, array size: nscan):
Transmitted pulse width (s) corrected with FCIF temperature in orbit, based on temperature test data of FCIF.

systemNoise (2-byte integer, array size: nray x nscan):
System Noise (dBm) is an average of the 4 measured system noise values, multiplied by 100 and stored as a 2-byte integer. The range is -120 dBm to -20 dBm with an accuracy of 0.9 dBm. Missing data are given the value of -32,734.

sysNoiseWarnFlag (1-byte integer, array size: nray x nscan):
System Noise Warning Flag indicates possible contamination of lower window noise by high towers of rain. 1 means possible contamination; 0 means no possible contamination.

minEchoFlag (1-byte integer, array size: nray x nscan):
Flag to indicate the presence of rain in the ray (angle bin). The values of the flag are:

```
0  = no rain
10 = rain possible
11 = rain possible (Echo greater than rain threshold #1 in clutter range)
12 = rain possible (Echo greater than rain threshold #2 in clutter range)
13 = rain possible but probably sidelobe clutter
20 = rain certain
```

binStormHeight (2-byte integer, array size: 2 x nray x nscan):
Range Bin Number of the storm top. The first dimension is threshold, with values of possible rain threshold and certain rain threshold in that order. The Bin Storm Heights are generated in the procedure to determine the Minimum Echo Flag. The Bin Storm Height is the top range bin of the portion of consecutive range bins that flagged the ray as rain possible or rain certain. The range bin number is defined in this volume in the section on Precipitation Radar, Instrument and Scan Geometry.

binEllipsoid (2-byte integer, array size: nray x nscan):
The range bin number of the earth ellipsoid. The range bin number is defined in this volume in the section on Precipitation Radar, Instrument and Scan Geometry.

binClutterFreeBottom (2-byte integer, array size: 2 x nray x nscan):
The range bin number of the lowest clutter free bin. Clutter free bin numbers are given for clutter free certain and possible, respectively. The clutter free certain bin is always less than or equal to the clutter free possible bin number.

binDIDHmean (2-byte integer, array size: nray x nscan):
The mean range bin number of the DID surface elevation in a 5 km x 5 km box centered on the IFOV. The range bin number is defined in this volume in the section on Precipitation Radar, Instrument and Scan Geometry.

binDIDHtop (2-byte integer, array size: 2 x nray x nscan):
The range bin number of the maximum DID surface elevation in a box centered on the IFOV. The first dimension is the box size, with sizes of 5 km x 5 km and 11 km x 11km. The range bin number is defined in this volume in the section on Precipitation Radar, Instrument and Scan Geometry.

binDIDHbottom (2-byte integer, array size: 2 x nray x nscan):
The range bin number of the minimum DID surface elevation in a box centered on the IFOV. The first dimension is the box size, with sizes of 5 km x 5 km and 11 km x 11km. The range bin number is defined in this volume in the section on Precipitation Radar, Instrument and Scan Geometry.

scLocalZenith (4-byte float, array size: nray x nscan):
The angle, in degrees, between the local zenith and the beam's center line. The local (geodetic) zenith at the intersection of the ray and the earth ellipsoid is used.

scRange (4-byte float, array size: nray x nscan):
Distance (m) between the spacecraft and the center of the footprint of the beam on the earth ellipsoid.

osBinStart (2-byte integer, array size: 2 x 29 x nscan):
The first dimension is the Bin Start of Oversample and Surface Tracker Status. The second dimension is the ray. The number of rays is 29 because this information only applies to the rays that have oversample data (rays 11 to 39). The third dimension is the scan. The Bin Start of Oversample is the starting range bin number of the oversample (either surface or rain) data, counting from the top down. The Surface Tracker Status has the value of 0 (Lock) or 1 (Unlock), where Lock means that (1) the on board surface detection detected the surface and (2) the surface detected later by processing on the ground fell within the oversample bins. Unlock means that Lock was not achieved. The range bin number is defined in this volume in the section on Precipitation Radar, Instrument and Scan Geometry.

landOceanFlag (2-byte integer, array size: nray x nscan):
Land or ocean information. The values of the flag are:

```
0 = Water
1 = Land
2 = Coast
3 = Water (w/ large attenuation)
4 = Land/Coast (w/ large attenuation)
```

surfWarnFlag (2-byte integer, array size: nray x nscan):
Definition TBD by JAXA.

binSurfPeak (2-byte integer, array size: nray x nscan):
The range bin number of the peak surface echo. This peak is determined by the post observation ground processing, not by the on board surface detection. The range bin number is defined in this volume in the section on Precipitation Radar, Instrument and Scan Geometry.

normalSample (2-byte integer, array size: 140 x nray x nscan):
Return power (dBm) of the normal sample, multiplied by 100 and stored as a 2-byte integer. Since each ray has a different size, the elements after the end of each ray are filled with a value of -32767. Other bins where data is not written due to a transmission,

calibration, or other problem, including an entire scan of missing bins, have the value of -32734. See Figure 6.1-2. The size of each ray is specified in Ray Header. The range is -120 dBm to -20 dBm.

osSurf (2-byte integer, array size: 5 x 29 x nscan):
Return power (dBm) of the surface echo oversample for the central 29 rays (rays number 11-39), multiplied by 100 and stored as a 2-byte integer. The range is -120 dBm to -20 dBm. Bins where data is not written due to a transmission, calibration, or other problem, including an entire scan of missing bins, have the value of -32734.

osRain (2-byte integer, array size: 28 x 11 x nscan):
Return power (dBm) of the rain echo oversample for the central 11 rays (rays number 20-30), multiplied by 100 and stored as a 2-byte integer. The range is -120 dBm to -20 dBm. Bins where data is not written due to a transmission, calibration, or other problem, including an entire scan of missing bins, have the value of -32734.

C Structure Header file:

```
#ifndef _TK_1B21_H_
#define _TK_1B21_H_

#ifndef _L1B21_POWERS_
#define _L1B21_POWERS_

typedef struct {
    short radarTransPower;
    float transPulseWidth;
} L1B21_POWERS;

#endif

#ifndef _L1B21_NAVIGATION_
#define _L1B21_NAVIGATION_

typedef struct {
    float scPosX;
    float scPosY;
    float scPosZ;
    float scVelX;
    float scVelY;
    float scVelZ;
    float scLat;
    float scLon;
    float scAlt;
    float scAttRoll;
```

```
    float scAttPitch;
    float scAttYaw;
    float SensorOrientationMatrix[3][3];
    float greenHourAng;
} L1B21_NAVIGATION;

#endif

#ifndef _L1B21_SCANSTATUS_
#define _L1B21_SCANSTATUS_

typedef struct {
    signed char missing;
    signed char validity;
    signed char qac;
    signed char geoQuality;
    signed char dataQuality;
    short SCorientation;
    signed char acsMode;
    signed char yawUpdateS;
    signed char prMode;
    signed char prStatus1;
    signed char prStatus2;
    double FractionalGranuleNumber;
} L1B21_SCANSTATUS;

#endif

#ifndef _L1B21_SCANTIME_
#define _L1B21_SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
} L1B21_SCANTIME;

#endif
```

```
#ifndef _L1B21_SWATH_
#define _L1B21_SWATH_

typedef struct {
    L1B21_SCANTIME ScanTime;
    double scanTime_sec;
    float Latitude[49];
    float Longitude[49];
    L1B21_SCANSTATUS scanStatus;
    L1B21_NAVIGATION navigation;
    L1B21_POWERS powers;
    float systemNoise[49];
    signed char sysNoiseWarnFlag[49];
    signed char minEchoFlag[49];
    short binStormHeight[49][2];
    short binEllipsoid[49];
    short binClutterFreeBottom[49][2];
    short binDIDHmean[49];
    short binDIDHtop[49][2];
    short binDIDHbottom[49][2];
    float scLocalZenith[49];
    float scRange[49];
    short osBinStart[29][2];
    short landOceanFlag[49];
    short surfWarnFlag[49];
    short binSurfPeak[49];
    float normalSample[49][140];
    float osSurf[29][5];
    float osRain[11][28];
} L1B21_SWATH;

#endif

#ifndef _L1B21_RAY_HEADER_
#define _L1B21_RAY_HEADER_

typedef struct {
    short rayStart[49];
    short raySize[49];
    float angle[49];
    float startBinDist[49];
    float rainThres1[49];
```

```
    float rainThres2[49];
    float transAntenna[49];
    float recvAntenna[49];
    float onewayAlongTrack[49];
    float onewayCrossTrack[49];
    float eqvWavelength[49];
    float radarConst[49];
    float prIntrDelay[49];
    float rangeBinSize[49];
    float logAveOffset[49];
    signed char mainlobeEdge[49];
    signed char sidelobeRange[49][3];
} L1B21_RAY_HEADER;

#endif

#ifndef _L1B21_PR_CAL_COEF_
#define _L1B21_PR_CAL_COEF_

typedef struct {
    float transCoef[1];
    float receptCoef[1];
    float fcifIOchar[16];
} L1B21_PR_CAL_COEF;

#endif

#endif
```

Fortran Structure Header file:

```
STRUCTURE /L1B21_POWERS/
    INTEGER*2 radarTransPower
    REAL*4 transPulseWidth
END STRUCTURE

STRUCTURE /L1B21_NAVIGATION/
    REAL*4 scPosX
    REAL*4 scPosY
    REAL*4 scPosZ
    REAL*4 scVelX
    REAL*4 scVelY
    REAL*4 scVelZ
```

```
    REAL*4 scLat
    REAL*4 scLon
    REAL*4 scAlt
    REAL*4 scAttRoll
    REAL*4 scAttPitch
    REAL*4 scAttYaw
    REAL*4 SensorOrientationMatrix(3,3)
    REAL*4 greenHourAng
END STRUCTURE

STRUCTURE /L1B21_SCANSTATUS/
    BYTE missing
    BYTE validity
    BYTE qac
    BYTE geoQuality
    BYTE dataQuality
    INTEGER*2 SCorientation
    BYTE acsMode
    BYTE yawUpdateS
    BYTE prMode
    BYTE prStatus1
    BYTE prStatus2
    REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /L1B21_SCANTIME/
    INTEGER*2 Year
    BYTE Month
    BYTE DayOfMonth
    BYTE Hour
    BYTE Minute
    BYTE Second
    INTEGER*2 MilliSecond
    INTEGER*2 DayOfYear
END STRUCTURE

STRUCTURE /L1B21_SWATH/
    RECORD /L1B21_SCANTIME/ ScanTime
    REAL*8 scanTime_sec
    REAL*4 Latitude(49)
    REAL*4 Longitude(49)
    RECORD /L1B21_SCANSTATUS/ scanStatus
    RECORD /L1B21_NAVIGATION/ navigation
```

```
    RECORD /L1B21_POWERS/ powers
    REAL*4 systemNoise(49)
    BYTE sysNoiseWarnFlag(49)
    BYTE minEchoFlag(49)
    INTEGER*2 binStormHeight(2,49)
    INTEGER*2 binEllipsoid(49)
    INTEGER*2 binClutterFreeBottom(2,49)
    INTEGER*2 binDIDHmean(49)
    INTEGER*2 binDIDHtop(2,49)
    INTEGER*2 binDIDHbottom(2,49)
    REAL*4 scLocalZenith(49)
    REAL*4 scRange(49)
    INTEGER*2 osBinStart(2,29)
    INTEGER*2 landOceanFlag(49)
    INTEGER*2 surfWarnFlag(49)
    INTEGER*2 binSurfPeak(49)
    REAL*4 normalSample(140,49)
    REAL*4 osSurf(5,29)
    REAL*4 osRain(28,11)
END STRUCTURE

STRUCTURE /L1B21_RAY_HEADER/
    INTEGER*2 rayStart(49)
    INTEGER*2 raySize(49)
    REAL*4 angle(49)
    REAL*4 startBinDist(49)
    REAL*4 rainThres1(49)
    REAL*4 rainThres2(49)
    REAL*4 transAntenna(49)
    REAL*4 recvAntenna(49)
    REAL*4 onewayAlongTrack(49)
    REAL*4 onewayCrossTrack(49)
    REAL*4 eqvWavelength(49)
    REAL*4 radarConst(49)
    REAL*4 prIntrDelay(49)
    REAL*4 rangeBinSize(49)
    REAL*4 logAveOffset(49)
    BYTE mainlobeEdge(49)
    BYTE sidelobeRange(3,49)
END STRUCTURE

STRUCTURE /L1B21_PR_CAL_COEF/
    REAL*4 transCoef1
```

```
    REAL*4 receptCoef1
    REAL*4 fcifIOchar(16)
END STRUCTURE
```

7.4 1C21 - PR Reflectivities

The PR Level-1C product, 1C21, "PR Reflectivities," is written as a Swath Structure. See Section 2 in HDF. The following sections describe the structure and contents of the format.

Dimension definitions:

nscan	var	Number of scans in the granule.
nray	49	Number of angle bins in each scan.

Figure 21 through Figure 28 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for TRMM Products for details.

InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for TRMM Products for details.

NavigationRecord (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for TRMM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for TRMM Products for details.

JAXAInfo (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by PR algorithms only. See Metadata for TRMM Products for details.

pr_cal_coef (Group)

transCoef (4-byte float, array size: 1):
Transmission coefficient.

receptCoef (4-byte float, array size: 1):
Reception coefficient.

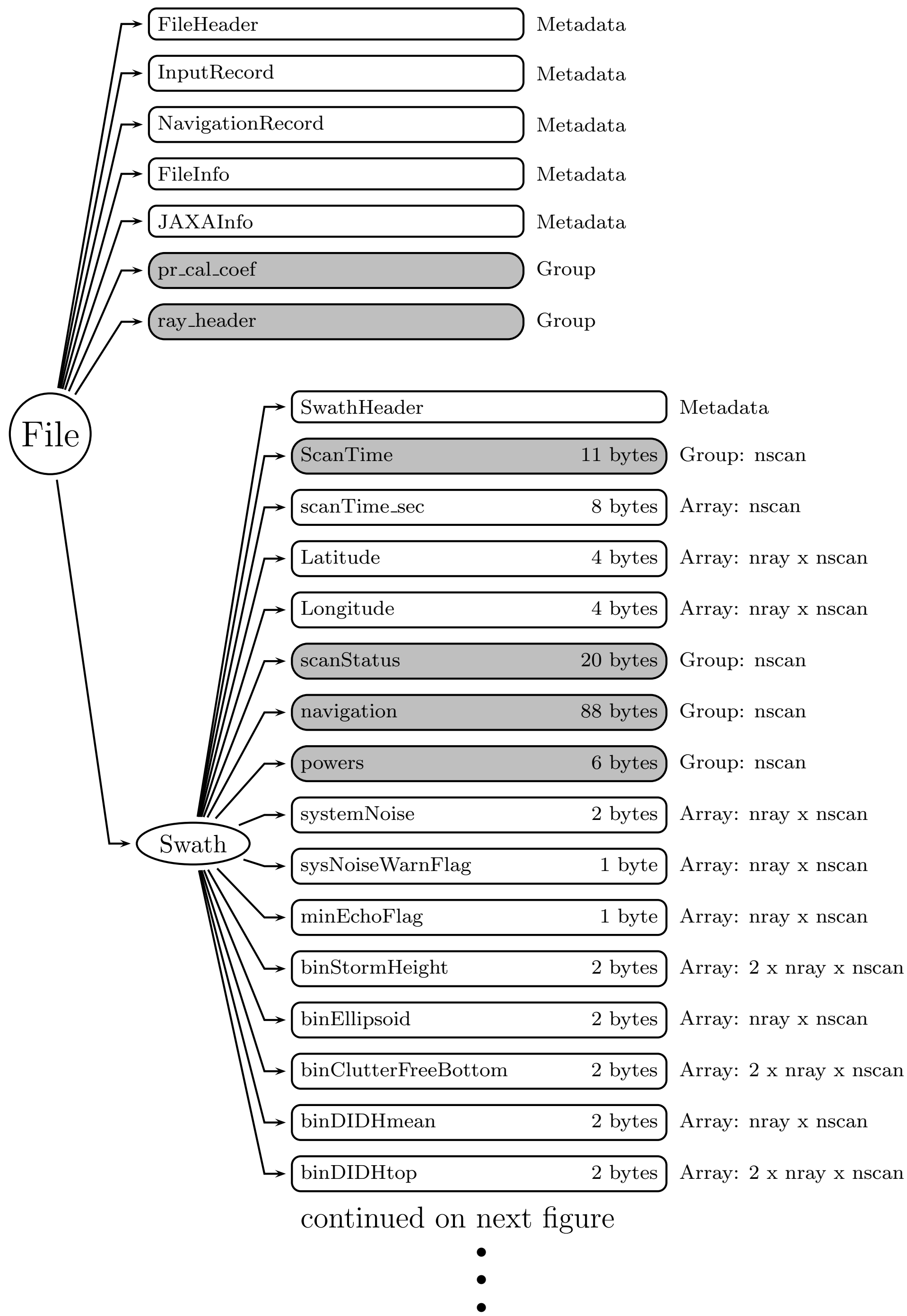


Figure 21: Data Format Structure for 1C21, PR Reflectivities

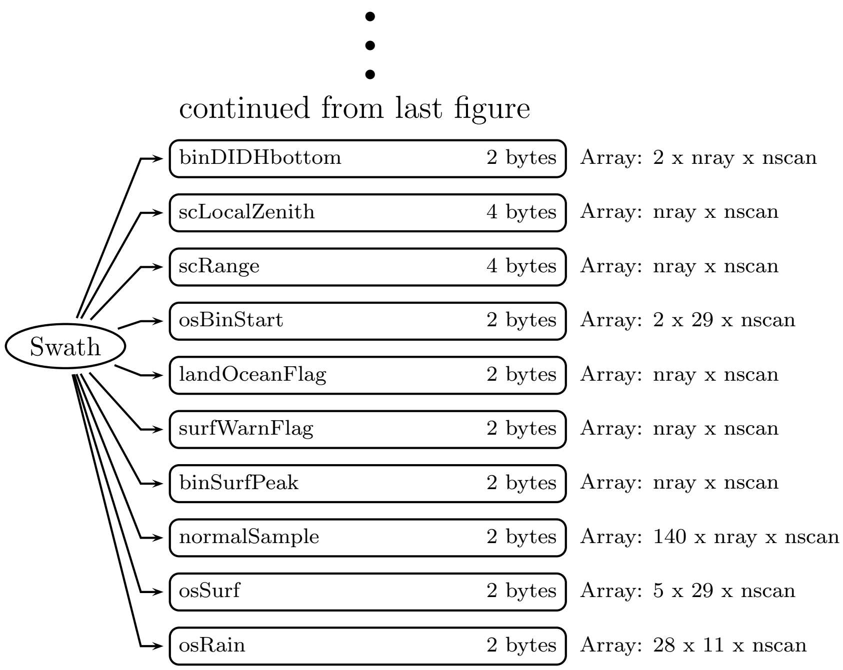


Figure 22: Data Format Structure for 1C21, PR Reflectivities

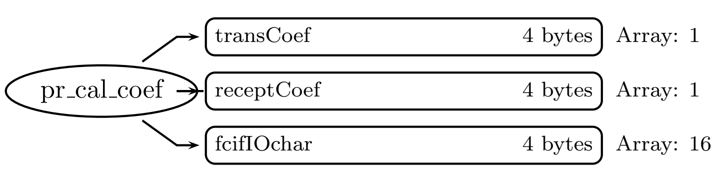


Figure 23: Data Format Structure for 1C21, pr_cal_coef

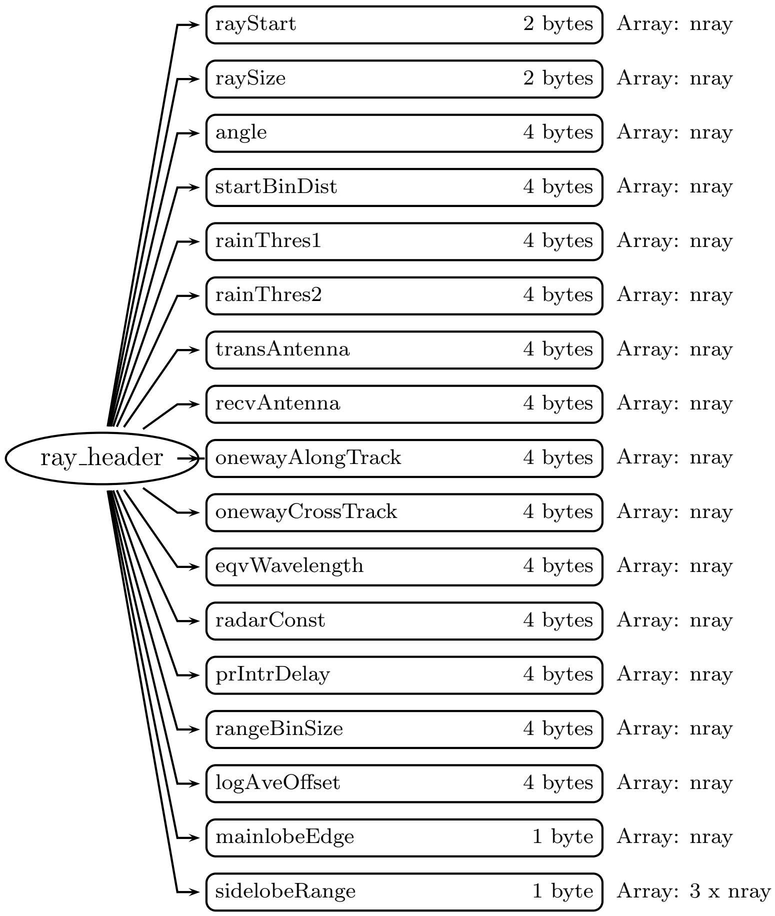


Figure 24: Data Format Structure for 1C21, ray_header

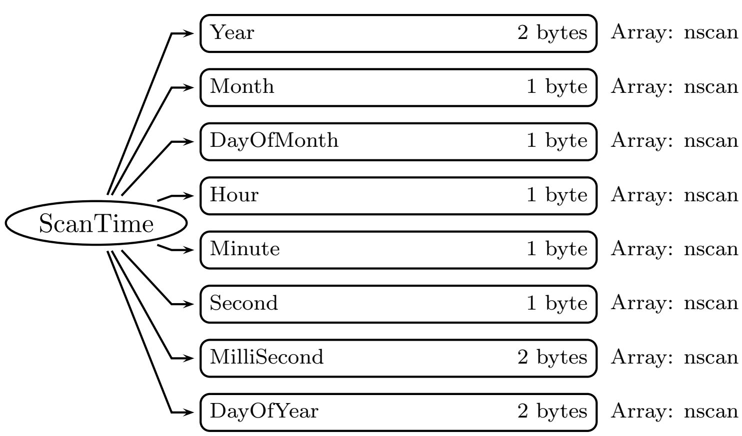


Figure 25: Data Format Structure for 1C21, ScanTime

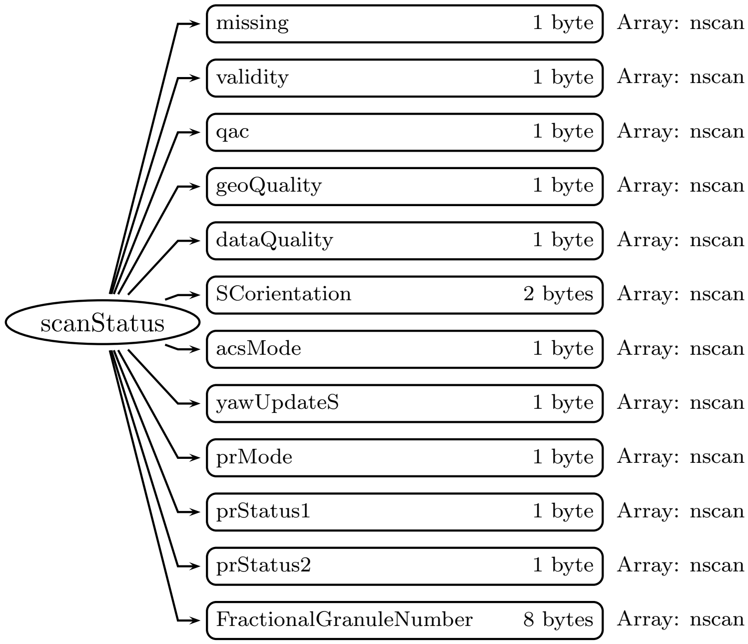


Figure 26: Data Format Structure for 1C21, scanStatus

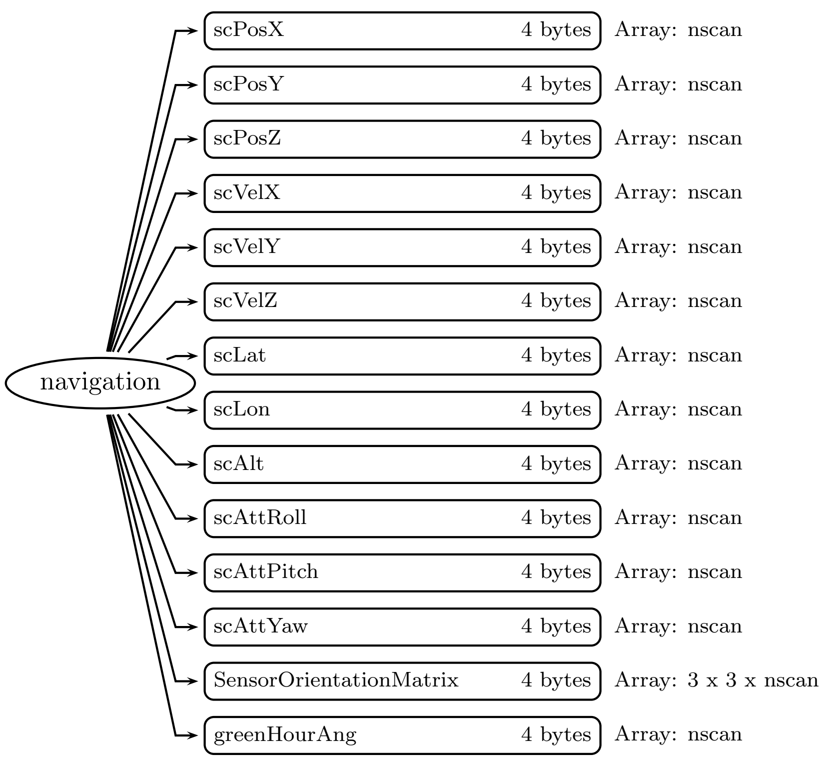


Figure 27: Data Format Structure for 1C21, navigation

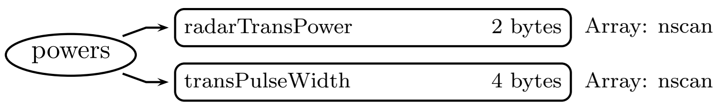


Figure 28: Data Format Structure for 1C21, powers

fcifIOchar (4-byte float, array size: 16):
FCIF I/O Characteristics.

ray_header (Group)

rayStart (2-byte integer, array size: nray):
Starting range bin number of Normal sample.

raySize (2-byte integer, array size: nray):
Number of Normal samples in the ray.

angle (4-byte float, array size: nray):
Angle (degrees) of the ray from nadir. The sign of the angle is consistent with the sensor y-axis, i.e., the angle is positive to the right of the direction of travel if the spacecraft is in normal mode.

startBinDist (4-byte float, array size: nray):
Distance (m) between the satellite and the starting bin number of the Normal sample for the ray.

rainThres1 (4-byte float, array size: nray):
Threshold used in minimum echo test (unitless). Value set by JAXA.

rainThres2 (4-byte float, array size: nray):
Threshold used in minimum echo test (unitless). Value set by JAXA.

transAntenna (4-byte float, array size: nray):
Transmitted radar antenna effectiveness (dB).

recvAntenna (4-byte float, array size: nray):
Received radar antenna effectiveness (dB).

onewayAlongTrack (4-byte float, array size: nray):
Radar beamwidth (radians) at the point transmitted power reaches one half of peak power in the along-track direction.

onewayCrossTrack (4-byte float, array size: nray):
Radar beamwidth (radians) at the point transmitted power reaches one half of peak power along the corss-track.

eqvWavelength (4-byte float, array size: nray):
Equivalent wavelength (m).

radarConst (4-byte float, array size: nray):
Radar constant dC (units are dB), which relates Received Power to Radar Reflectivity. dC depends on angle. ADD EQUATIONS.

prIntrDelay (4-byte float, array size: nray):
The time (seconds) between when echo returns at antenna and when echo is recorded in onboard processor.

rangeBinSize (4-byte float, array size: nray):
The vertical resolution of Normal sample bin (250 m).

logAveOffset (4-byte float, array size: nray):
The offset value (dB) between logarithmic average and normal average (+2.507dB).

mainlobeEdge (1-byte integer, array size: nray):
Absolute value of the difference in Range Bin Numbers between the detected surface and the edge of the clutter from the mainlobe.

sidelobeRange (1-byte integer, array size: 3 x nray):
Absolute value of the difference in Range Bin Numbers between the detected surface and the clutter psition from the sidelobe. A zero means no clutter indicated in this field since less than 3 bins contained significant clutter.

Swath (Swath)

SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for TRMM Products for details.

ScanTime (Group)

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

scanTime_sec (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: nray x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: nray x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scanStatus (Group)

missing (1-byte integer, array size: nscan):
Missing indicates whether information is contained in the scan data. The values are:

```
0 Scan data elements contain information
1 Scan was missing in the telemetry data
2 Scan data contains no elements with rain
```

validity (1-byte integer, array size: nscan):
Validity is a summary of status modes. If all status modes are routine, all bits in Validity = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. Validity does not assess data or geolocation quality. Validity is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

```
Bit Meaning if bit = 1
```

```
0   Spare (always 0)
1   Non-routine spacecraft orientation (2 or 3)
2   Non-routine ACS mode (other than 4)
3   Non-routine yaw update status (0 or 1)
4   Non-routine instrument status (other than 1)
5   Non-routine QAC (non-zero)
6   Spare (always 0)
7   Spare (always 0)
```

qac (1-byte integer, array size: nscan):
The Quality and Accounting Capsule of the Science packet as it appears in Level-0 data. If no QAC is given in Level-0, which means no decoding errors occurred, QAC in this format has a value of zero.

geoQuality (1-byte integer, array size: nscan):
Geolocation quality is a summary of geolocation quality in the scan. A zero integer value indicates 'good' geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

```
Bit Meaning if bit = 1
0   latitude limit error
1   geolocation
2   attitude change rate limit error
3   attitude limit error
4   satellite undergoing maneuvers
5   using predictive orbit data
6   geolocation calculation error
7   not used
```

dataQuality (1-byte integer, array size: nscan):
Data quality is a summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

```
Bit Meaning if bit = 1
0   missing
5   Geolocation Quality is not normal
6   Validity is not normal
```

SCorientation (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the TMI scan. If +X is forward, SCorientation is 0. If -X is forward, SCorientation is 180. If -Y is forward, SCorientation is 90. Values range

from 0 to 360 degrees. Special values are defined as:

-8003	Inertial
-8004	Unknown
-9999	Missing value

acsMode (1-byte integer, array size: nscan):

```
Value Meaning
0     Standby
1     Sun Acquire
2     Earth Acquire
3     Yaw Acquire
4     Nominal
5     Yaw Maneuver
6     Delta-H (Thruster)
7     Delta-V (Thruster)
8     CERES Calibration
```

yawUpdateS (1-byte integer, array size: nscan):

```
Value Meaning
0     Inaccurate
1     Indeterminate
2     Accurate
```

prMode (1-byte integer, array size: nscan):

```
Value Meaning
1     Observation Mode
2     Other Mode
```

prStatus1 (1-byte integer, array size: nscan):
This status is a warning for scan data. Unless this is 0, the scan data may include a little questionable value though it is not a problem (such as break of caution limit). This field is used only for NASDA's data analysis.

prStatus2 (1-byte integer, array size: nscan):
Initialization in Onboard Surface Search Algorithm.

```
Value Meaning
0     Not initialized
1     Initialized
```

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft's trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

navigation (Group)

scPosX (4-byte float, array size: nscan):
The x component of the position (m) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Geocentric Inertial Coordinates are also commonly known as Earth Centered Inertial coordinates. These coordinates will be True of Date (rather than Epoch 2000 which are also commonly used), as interpolated from the data in the Flight Dynamics Facility ephemeris files generated for TRMM.

scPosY (4-byte float, array size: nscan):
The y component of the position (m) of the spacecraft in Geocentric Inertial Coordinates. See scPosX.

scPosZ (4-byte float, array size: nscan):
The z component of the position (m) of the spacecraft in Geocentric Inertial Coordinates. See scPosX.

scVelX (4-byte float, array size: nscan):
The x component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scVelY (4-byte float, array size: nscan):
The y component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scVelZ (4-byte float, array size: nscan):
The z component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scLat (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time.

scLon (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time.

scAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time.

scAttRoll (4-byte float, array size: nscan):
The satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the

rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

scAttPitch (4-byte float, array size: nscan):
The satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

scAttYaw (4-byte float, array size: nscan):
The satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

SensorOrientationMatrix (4-byte float, array size: 3 x 3 x nscan):
SensorOrientationMatrix is the rotation matrix from the instrument coordinate frame to Geocentric Inertial Coordinates at the Scan mid-Time. It is unitless.

greenHourAng (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates.

powers (Group)

radarTransPower (2-byte integer, array size: nscan):
The total (sum) power of 128 SSPA elements corrected with SSPA temperature in orbit, based on temperature test data of SSPA transmission power. The units are dBm * 100. For this variable, the TSDIS Toolkit does not provide scaling.

transPulseWidth (4-byte float, array size: nscan):
Transmitted pulse width (s) corrected with FCIF temperature in orbit, based on temperature test data of FCIF.

systemNoise (2-byte integer, array size: nray x nscan):
System Noise (dBm) is an average of the 4 measured system noise values, multiplied by 100 and stored as a 2-byte integer. The range is -120 dBm to -20 dBm with an accuracy of 0.9 dBm. Missing data are given the value of -32,734.

sysNoiseWarnFlag (1-byte integer, array size: nray x nscan):
System Noise Warning Flag indicates possible contamination of lower window noise by high towers of rain. 1 means possible contamination; 0 means no possible contamination.

minEchoFlag (1-byte integer, array size: nray x nscan):
Flag to indicate the presence of rain in the ray (angle bin). The values of the flag are:

```
0  = no rain
10 = rain possible
11 = rain possible (Echo greater than rain threshold #1 in clutter range)
12 = rain possible (Echo greater than rain threshold #2 in clutter range)
13 = rain possible but probably sidelobe clutter
20 = rain certain
```

binStormHeight (2-byte integer, array size: 2 x nray x nscan):
Range Bin Number of the storm top. The first dimension is threshold, with values of possible rain threshold and certain rain threshold in that order. The Bin Storm Heights are generated in the procedure to determine the Minimum Echo Flag. The Bin Storm Height is the top range bin of the portion of consecutive range bins that flagged the ray as rain possible or rain certain. The range bin number is defined in this volume in the section on Precipitation Radar, Instrument and Scan Geometry.

binEllipsoid (2-byte integer, array size: nray x nscan):
The range bin number of the earth ellipsoid. The range bin number is defined in this volume in the section on Precipitation Radar, Instrument and Scan Geometry.

binClutterFreeBottom (2-byte integer, array size: 2 x nray x nscan):
The range bin number of the lowest clutter free bin. Clutter free bin numbers are given for clutter free certain and possible, respectively. The clutter free certain bin is always less than or equal to the clutter free possible bin number.

binDIDHmean (2-byte integer, array size: nray x nscan):
The mean range bin number of the DID surface elevation in a 5 km x 5 km box centered on the IFOV. The range bin number is defined in this volume in the section on Precipitation Radar, Instrument and Scan Geometry.

binDIDHtop (2-byte integer, array size: 2 x nray x nscan):
The range bin number of the maximum DID surface elevation in a box centered on the IFOV. The first dimension is the box size, with sizes of 5 km x 5 km and 11 km x 11km. The range bin number is defined in this volume in the section on Precipitation Radar, Instrument and Scan Geometry.

binDIDHbottom (2-byte integer, array size: 2 x nray x nscan):
The range bin number of the minimum DID surface elevation in a box centered on the IFOV. The first dimension is the box size, with sizes of 5 km x 5 km and 11 km x 11km. The range bin number is defined in this volume in the section on Precipitation Radar, Instrument and Scan Geometry.

scLocalZenith (4-byte float, array size: nray x nscan):
The angle, in degrees, between the local zenith and the beam's center line. The local (geodetic) zenith at the intersection of the ray and the earth ellipsoid is used.

scRange (4-byte float, array size: nray x nscan):
Distance (m) between the spacecraft and the center of the footprint of the beam on the earth ellipsoid.

osBinStart (2-byte integer, array size: 2 x 29 x nscan):
The first dimension is the Bin Start of Oversample and Surface Tracker Status. The second dimension is the ray. The number of rays is 29 because this information only applies to the rays that have oversample data (rays 11 to 39). The third dimension is the scan. The Bin Start of Oversample is the starting range bin number of the oversample (either surface or rain) data, counting from the top down. The Surface Tracker Status has the value of 0 (Lock) or 1 (Unlock), where Lock means that (1) the on board surface detection detected the surface and (2) the surface detected later by processing on the ground fell within the oversample bins. Unlock means that Lock was not achieved. The range bin number is defined in this volume in the section on Precipitation Radar, Instrument and Scan Geometry.

landOceanFlag (2-byte integer, array size: nray x nscan):
Land or ocean information. The values of the flag are:

```
0 = Water
1 = Land
2 = Coast
3 = Water (w/ large attenuation)
4 = Land/Coast (w/ large attenuation)
```

surfWarnFlag (2-byte integer, array size: nray x nscan):
Definition TBD by JAXA.

binSurfPeak (2-byte integer, array size: nray x nscan):
The range bin number of the peak surface echo. This peak is determined by the post observation ground processing, not by the on board surface detection. The range bin number is defined in this volume in the section on Precipitation Radar, Instrument and Scan Geometry.

normalSample (2-byte integer, array size: 140 x nray x nscan):
Reflectivity (dBZ) of the normal sample, multiplied by 100 and stored as a 2-byte integer. Since each ray has a different size, the elements after the end of each ray are filled with a value of -32767. Other bins where data is not written due to a transmission, calibration, or other problem, including an entire scan of missing bins, have the value of -32734. See Figure 6.1-2. The size of each ray is specified in Ray Header. The range is -20 dBZ to 80 dBZ with an accuracy of 1 dBZ. If a cell (range bin) is determined to have no rain, the reflectivity is set to -32700.

osSurf (2-byte integer, array size: 5 x 29 x nscan):
Reflectivity (dBZ) of the surface echo oversample for the central 29 rays (rays 11-39), multiplied by 100 and stored as a 2-byte integer. The range is -20 dBZ to 80 dBZ with an accuracy of 1 dBZ. Bins where data is not written due to a transmission, calibration, or other problem, including an entire scan of missing bins, have the value of -32734. If a cell (range bin) is determined to have no rain, the reflectivity is set to -32700.

osRain (2-byte integer, array size: 28 x 11 x nscan):
Reflectivity (dBZ) of the rain echo oversample for the central 11 rays (rays 20-30), multiplied by 100 and stored as a 2-byte integer. The range is -20 dBZ to 80 dBZ with an accuracy of 1 dBZ. Bins where data is not written due to a transmission, calibration, or other problem, including an entire scan of missing bins, have the value of -32734. If a cell (range bin) is determined to have no rain, the reflectivity is set to -32700.

C Structure Header file:

```
#ifndef _TK_1C21_H_
#define _TK_1C21_H_

#ifndef _L1C21_POWERS_
#define _L1C21_POWERS_

typedef struct {
    short radarTransPower;
    float transPulseWidth;
} L1C21_POWERS;

#endif

#ifndef _L1C21_NAVIGATION_
#define _L1C21_NAVIGATION_
```

```
typedef struct {
    float scPosX;
    float scPosY;
    float scPosZ;
    float scVelX;
    float scVelY;
    float scVelZ;
    float scLat;
    float scLon;
    float scAlt;
    float scAttRoll;
    float scAttPitch;
    float scAttYaw;
    float SensorOrientationMatrix[3][3];
    float greenHourAng;
} L1C21_NAVIGATION;

#endif

#ifndef _L1C21_SCANSTATUS_
#define _L1C21_SCANSTATUS_

typedef struct {
    signed char missing;
    signed char validity;
    signed char qac;
    signed char geoQuality;
    signed char dataQuality;
    short SCorientation;
    signed char acsMode;
    signed char yawUpdateS;
    signed char prMode;
    signed char prStatus1;
    signed char prStatus2;
    double FractionalGranuleNumber;
} L1C21_SCANSTATUS;

#endif

#ifndef _L1C21_SCANTIME_
#define _L1C21_SCANTIME_
```

```
typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
} L1C21_SCANTIME;

#endif

#ifndef _L1C21_SWATH_
#define _L1C21_SWATH_

typedef struct {
    L1C21_SCANTIME ScanTime;
    double scanTime_sec;
    float Latitude[49];
    float Longitude[49];
    L1C21_SCANSTATUS scanStatus;
    L1C21_NAVIGATION navigation;
    L1C21_POWERS powers;
    float systemNoise[49];
    signed char sysNoiseWarnFlag[49];
    signed char minEchoFlag[49];
    short binStormHeight[49][2];
    short binEllipsoid[49];
    short binClutterFreeBottom[49][2];
    short binDIDHmean[49];
    short binDIDHtop[49][2];
    short binDIDHbottom[49][2];
    float scLocalZenith[49];
    float scRange[49];
    short osBinStart[29][2];
    short landOceanFlag[49];
    short surfWarnFlag[49];
    short binSurfPeak[49];
    float normalSample[49][140];
    float osSurf[29][5];
    float osRain[11][28];
} L1C21_SWATH;
```

```
#endif

#ifndef _L1C21_RAY_HEADER_
#define _L1C21_RAY_HEADER_

typedef struct {
    short rayStart[49];
    short raySize[49];
    float angle[49];
    float startBinDist[49];
    float rainThres1[49];
    float rainThres2[49];
    float transAntenna[49];
    float recvAntenna[49];
    float onewayAlongTrack[49];
    float onewayCrossTrack[49];
    float eqvWavelength[49];
    float radarConst[49];
    float prIntrDelay[49];
    float rangeBinSize[49];
    float logAveOffset[49];
    signed char mainlobeEdge[49];
    signed char sidelobeRange[49][3];
} L1C21_RAY_HEADER;

#endif

#ifndef _L1C21_PR_CAL_COEF_
#define _L1C21_PR_CAL_COEF_

typedef struct {
    float transCoef[1];
    float receptCoef[1];
    float fcifIOchar[16];
} L1C21_PR_CAL_COEF;

#endif

#endif
```

Fortran Structure Header file:

```
STRUCTURE /L1C21_POWERS/
    INTEGER*2 radarTransPower
    REAL*4 transPulseWidth
END STRUCTURE

STRUCTURE /L1C21_NAVIGATION/
    REAL*4 scPosX
    REAL*4 scPosY
    REAL*4 scPosZ
    REAL*4 scVelX
    REAL*4 scVelY
    REAL*4 scVelZ
    REAL*4 scLat
    REAL*4 scLon
    REAL*4 scAlt
    REAL*4 scAttRoll
    REAL*4 scAttPitch
    REAL*4 scAttYaw
    REAL*4 SensorOrientationMatrix(3,3)
    REAL*4 greenHourAng
END STRUCTURE

STRUCTURE /L1C21_SCANSTATUS/
    BYTE missing
    BYTE validity
    BYTE qac
    BYTE geoQuality
    BYTE dataQuality
    INTEGER*2 SCorientation
    BYTE acsMode
    BYTE yawUpdateS
    BYTE prMode
    BYTE prStatus1
    BYTE prStatus2
    REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /L1C21_SCANTIME/
    INTEGER*2 Year
    BYTE Month
    BYTE DayOfMonth
    BYTE Hour
    BYTE Minute
```

```
    BYTE Second
    INTEGER*2 MilliSecond
    INTEGER*2 DayOfYear
END STRUCTURE

STRUCTURE /L1C21_SWATH/
    RECORD /L1C21_SCANTIME/ ScanTime
    REAL*8 scanTime_sec
    REAL*4 Latitude(49)
    REAL*4 Longitude(49)
    RECORD /L1C21_SCANSTATUS/ scanStatus
    RECORD /L1C21_NAVIGATION/ navigation
    RECORD /L1C21_POWERS/ powers
    REAL*4 systemNoise(49)
    BYTE sysNoiseWarnFlag(49)
    BYTE minEchoFlag(49)
    INTEGER*2 binStormHeight(2,49)
    INTEGER*2 binEllipsoid(49)
    INTEGER*2 binClutterFreeBottom(2,49)
    INTEGER*2 binDIDHmean(49)
    INTEGER*2 binDIDHtop(2,49)
    INTEGER*2 binDIDHbottom(2,49)
    REAL*4 scLocalZenith(49)
    REAL*4 scRange(49)
    INTEGER*2 osBinStart(2,29)
    INTEGER*2 landOceanFlag(49)
    INTEGER*2 surfWarnFlag(49)
    INTEGER*2 binSurfPeak(49)
    REAL*4 normalSample(140,49)
    REAL*4 osSurf(5,29)
    REAL*4 osRain(28,11)
END STRUCTURE

STRUCTURE /L1C21_RAY_HEADER/
    INTEGER*2 rayStart(49)
    INTEGER*2 raySize(49)
    REAL*4 angle(49)
    REAL*4 startBinDist(49)
    REAL*4 rainThres1(49)
    REAL*4 rainThres2(49)
    REAL*4 transAntenna(49)
    REAL*4 recvAntenna(49)
    REAL*4 onewayAlongTrack(49)
```

```
    REAL*4 onewayCrossTrack(49)
    REAL*4 eqvWavelength(49)
    REAL*4 radarConst(49)
    REAL*4 prIntrDelay(49)
    REAL*4 rangeBinSize(49)
    REAL*4 logAveOffset(49)
    BYTE mainlobeEdge(49)
    BYTE sidelobeRange(3,49)
END STRUCTURE

STRUCTURE /L1C21_PR_CAL_COEF/
    REAL*4 transCoef1
    REAL*4 receptCoef1
    REAL*4 fcifIOchar(16)
END STRUCTURE
```

7.5 2A12 - TMI Profiling

2A-12, "TMI Profiling", generates surface rainfall and vertical hydrometeor profiles on a pixel by pixel basis from the TRMM Microwave Imager (TMI) brightness temperature data using the Goddard Profiling algorithm GPROF2010. Because the vertical information comes from a radiometer, it is not written out in independent vertical layers like the TRMM Precipitation Radar. Instead, the output is referenced to one of 100 typical structures for each hydrometeor or heating profile. These vertical structures are referenced as clusters in the output structure. Vertical hydrometeor profiles can be reconstructed to 28 layers by knowing the cluster number (i.e. shape) of the profile and a scale factor that is written for each pixel.

GPROF 2010 (2A12, V7) vs GPROF 2004 (2A12, V6)
(A short explanation of the differences)

The basic change from 2A12, V6 (corresponding to GRPOF2004) and 2A12, V7 (Corresponding to GPROF2010) is the use of the observed databases instead of CRM databases. GPROF 2004 used a set of eight cloud resolving model simulations to create the a-priori databases for the Bayesian retrieval scheme. This methodology had issues over oceans due primarily to a lack of completeness or representativeness of this database. GPROF 2010 replaced these simulations with combined radar/radiometer/CRM retrievals that are supposed to better represent the actual profiles seen over oceans. One year of combined retrievals (detailed in Kummerow et al., J. Atmos. and Oceanic Tech., 2011) are used to construct the a-priori database. Note that while the a-priori database in GPROF2004 was stratified by SST, the GPROF2010 database is stratified by both SST and total precipitable water (TPW). The retrieval was also changed over land. Here, new regressions have been established to match PR rainfall. These are done separately for convective and stratiform precipitation. The regressions are global. Changes to the algorithm led to some natural changes in the output format. Below is a basic description of the new or modified fields.

pixelStatus flags any pixel that was eliminated due to QC screens. The value itself, if not equal to zero, contains information about the specific QC procedure that identified the pixel as bad. Values are available below. If PixelStatus is not equal to zero, all other fields are set to missing.

qualityFlag is a qualitative flag in GPROF2010 to guide users in areas where GPROF 2010 has retrieved rain but with less than optimal confidence. QualityFlag = 0 means highest confidence. QualityFlag = 1 is used to identify pixels that are probably good but climate trends and very small signals should not be taken from these. Pixels with a small sun glint angle, for instance, are identified in this category. QualityFlag = 2 should be used primarily for qualitative purposes. This category, for instance, identifies areas where GPROF databases were searched far beyond the appropriate SST and TPW bin before a solution could be found.

surfacePrecipitation is the precipitation rate corresponding to each pixel. This value should be accumulated if area or temporal averages are sought. SurfaceRain represents the liquid component of SurfacePrecipitation.

New/Additional diagnostic parameters over ocean

oceanSearchRadius contains the specific value of how far the algorithm searched the a-priori database beyond the nominal SST and TPW bin on the pixel. This field is intended for users willing to parse the QualityFlag further and then use certain subsets of the solution.

chiSquared (ocean only): In addition to precipitation, GPROF2010 also retrieves and outputs surface wind speed, water vapor and cloud water. The retrieval uses an optimal estimation methodology (similar to 1D Var). ChiSquared is the diagnostic that measures the fit of the non-raining solution. A ChiSquared value of less than 5 is equivalent to a quality flag of 0, implying that the wind, water vapor and cloud water are good. ChiSquared values between 5 and 10 are generally trustworthy but they can be contaminated with light rain and should probably not be used for studying small trends or correlations. Values above 10 are equivalent to qualityFlag = 2 and should not be used for non-raining pixels. Precipitation can be retrieved simultaneously with these parameters. A typical scenario is one where chiSquared is less than 5 for no rain or very light (less than 0.1 mm/hr) rain, and increases as the rainfall increases. For significant rain, chiSquared is often very high, while qualityFlag = 0 – simply stating that background fields cannot be retrieved reliably in heavy rainfall conditions.

probabilityOfPrecip (ocean only): Because the ocean database now follows PR/TMI observations, including the proper ratio of raining and non-raining pixels, the retrieval no longer screens for raining pixels before applying the Bayesian scheme. All pixels are compared to the a-priori database. This leads to an unexpected, but correct result that nearly all pixels have some very small probability of rainfall as well as rainfall rate. This occurs because a given set of observed Tbs has similarities to one or more entry in the database that has some rain, even if the majority of matches are to non-raining pixels. To allow the user to determine if a single pixel is likely raining or not, GPROF 2010 writes out the probability of precipitation in addition to the mean rainfall rate. Using a threshold of 50% for the probability of rain was found to be a fairly good indicator of rain in the FOV when compared to PR (at the lower TMI resolution). For monthly accumulations over oceans, the user should count all pixels (i.e. use surfacePrecipitation). For making instantaneous rainfall maps, or maps of the probability of rain, higher probabilityOfPrecip threshold should be used or maps will reflect all pixels in which rain is possible (instead of likely). A probability of precipitation of 50% seems to track raining areas pretty well but other thresholds can be chosen as well. The transition between 0% rain and 100% rain is quite steep so that the exact threshold for the probability of precipitation is not critical.

Other Changes to Output

The final major change over oceans deals with the format of the output structure. GPROF2004 (2A12, V6) produced output in 14 layers for rain water, cloud water, precipitation ice, cloud ice and latent heating. GPROF 2010 uses 28 layers but does not write these out explicitly. Instead, it uses only 100 typical profiles (clusters) for each hydrometeor (or LH) species. The profile shapes for each cluster are given in the DataHeader while the pixel carries only the profile cluster number and the scaling factor for each hydrometeor type. Freezing levels are not mixed so the freezing level index is also given for each pixel. This reduces the size of the files significantly although the profiles have to be reconstructed from the shape and scale information before it can be used.

Land retrievals use a regression against PR data as described above. The surface parameters (wind, water vapor and cloud water) are not retrieved over land. As such, chi squared does not exist. Land still employs a screen to determine if it is raining or not. Unfortunately, the screens are not always unambiguous and an ambiguousFlag is therefore defined. If a pixel is defined as Ambiguous, it is given a low qualityFlag of 2. Ambiguous pixels should be avoided by all except expert users who know each of the ambiguous screens. Because the ambiguous definition does not translate into a probability of precipitation as it does over ocean, the probabilityOfPrecip is left as missing over land and coast. Land and Coast retrievals do not retrieve any structure for TMI. All vertical information is also set to missing at this time.

The format of this product was designed in consultation with the TMI algorithm scientists. The following sections describe the structure and contents of the format.

Dimension definitions:

nscan	var	Number of scans in the granule.
npixel	208	Number of pixels in each scan.
nspecies	6	Number of hydrometeor species. Species are: 1=cloud liquid water content (g/m^3), 2=rain water content (g/m^3), 3=cloud ice water content (g/m^3), 4=snow water content (g/m^3), 5=graupel water content (g/m^3), 6=latent heating (K/h).
nfindex	13	Number of freezing height indeces.
nlayer	28	Number of profiling layers. The top height of each layer is specified in heightLayerTop.
ncluster	100	Number of clusters at each freezing height.

Figure 29 through Figure 34 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for TRMM Products for details.

InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for TRMM Products for details.

NavigationRecord (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for TRMM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for TRMM Products for details.

GprofInfo (Metadata):
GprofInfo contains metadata required by Gprof. Used by 2A12 only. See Metadata for TRMM Products for details.

DataHeader (Group)

heightLayerTop (4-byte float, array size: nlayer):
Height of the top of each atmospheric layer in the rain profile. Values range from 0 to 18.0 km. Special values are defined as:

-9999.9	Missing value

cluster (4-byte float, array size: nspecies x nfindex x nlayer x ncluster):
Hydrometeor profile shapes. Dimensions are hydrometeor/heating species (6) consisting of cloud water, rain water, cloud ice, snow, graupel, and latent heating; freezing height index (13) for freezing levels starting at 250m and going to 4.5 km in 250 m intervals, vertical layers (28) starting at 500 m and going to 18 km at 500 m layers in the lower troposphere

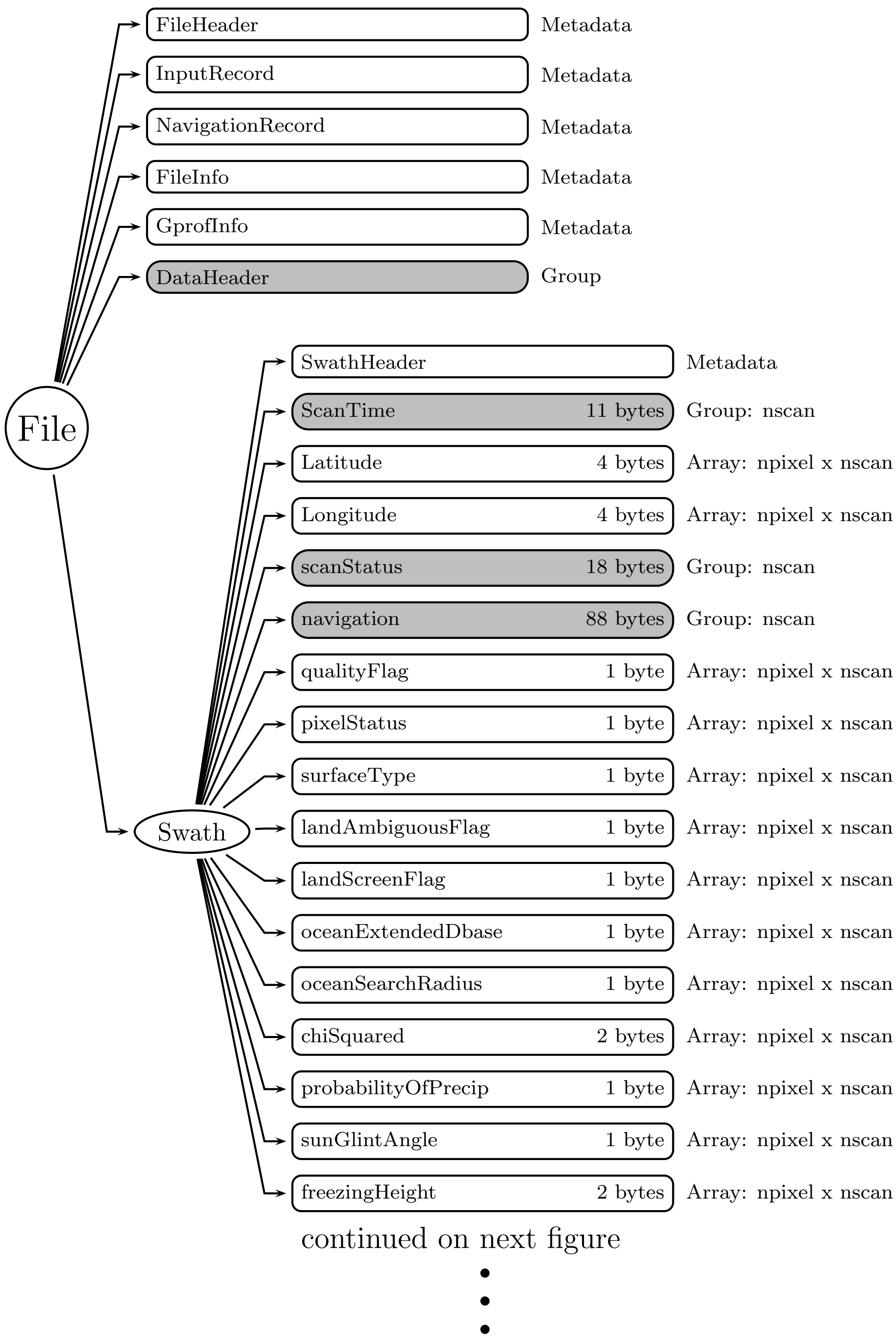


Figure 29: Data Format Structure for 2A12, TMI Profiling

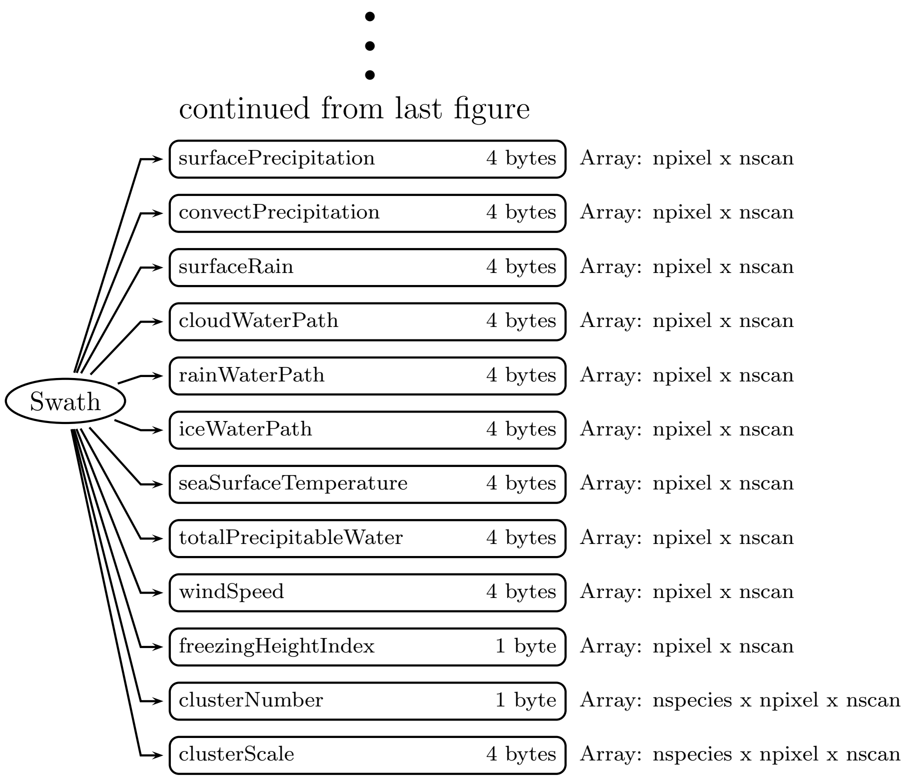


Figure 30: Data Format Structure for 2A12, TMI Profiling

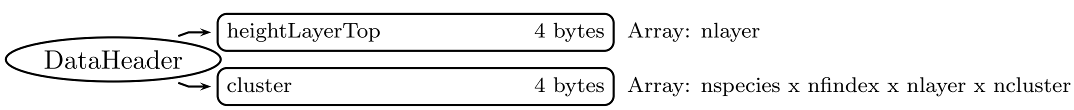


Figure 31: Data Format Structure for 2A12, DataHeader

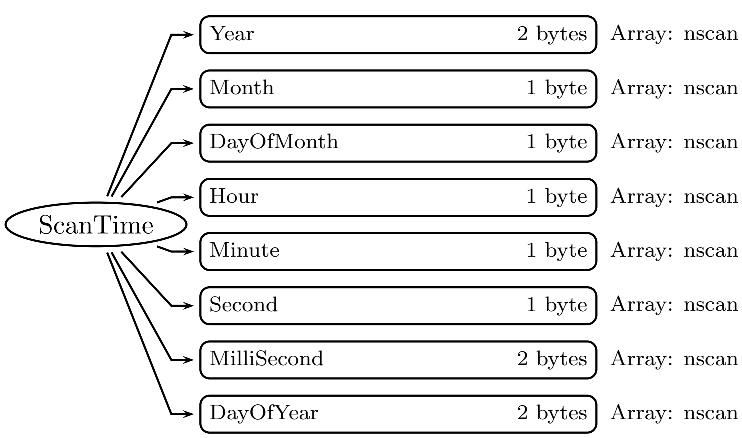


Figure 32: Data Format Structure for 2A12, ScanTime

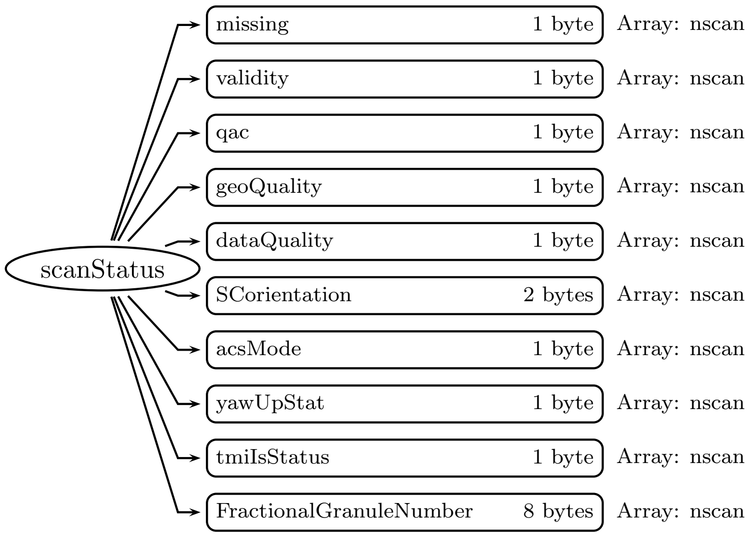


Figure 33: Data Format Structure for 2A12, scanStatus

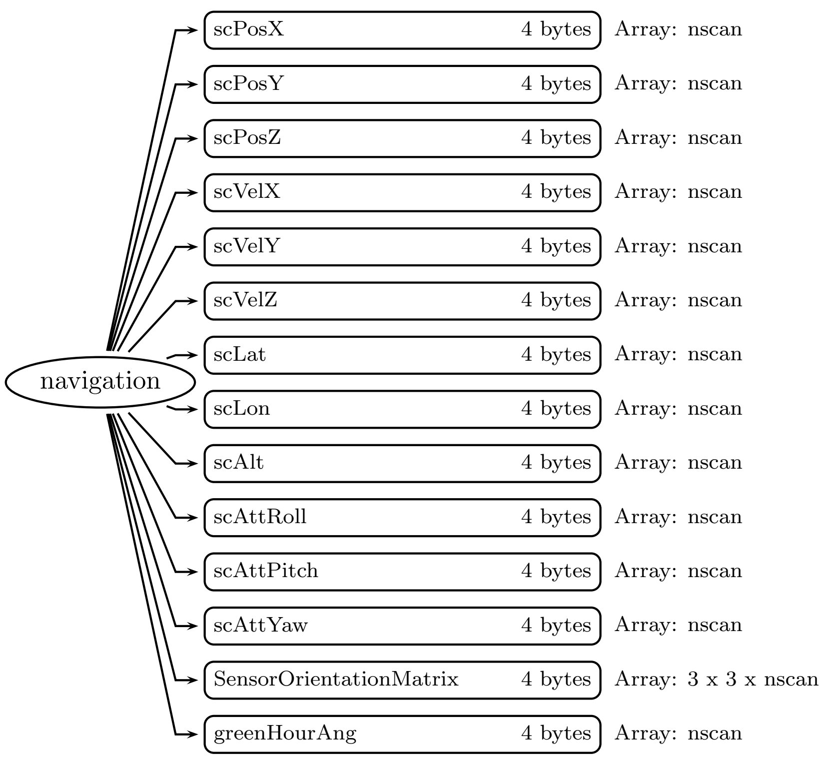


Figure 34: Data Format Structure for 2A12, navigation

and 1 km in the upper troposphere; and cluster number (currently 100 shapes). To recover values in a profile see the description below the variable clusterScale.

Special values are defined as:
-9999.9 Missing value

Swath (Swath)

SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for TRMM Products for details.

ScanTime (Group)

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

Latitude (4-byte float, array size: npixel x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are

defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npixel x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scanStatus (Group)

missing (1-byte integer, array size: nscan):
Missing indicates whether information is contained in the scan data. The values are:

```
0 Scan data elements contain information
1 Scan was missing in the telemetry data
```

validity (1-byte integer, array size: nscan):
Validity is a summary of status modes. If all status modes are routine, all bits in Validity = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. Validity does not assess data or geolocation quality. Validity is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

```
Bit Meaning if bit = 1
0   Spare (always 0)
1   Non-routine spacecraft orientation (2 or 3)
2   Non-routine ACS mode (other than 4)
3   Non-routine yaw update status (0 or 1)
4   Non-routine instrument status (Bit 0 = 0 or bit 1 = 0)
5   Non-routine QAC (non-zero)

6   21 GHz Cold Count Flag (1 if Flag set)
7   Spare (always 0)
```

qac (1-byte integer, array size: nscan):
The Quality and Accounting Capsule of the Science packet as it appears in Level-0 data. If no QAC is given in Level-0, which means no decoding errors occurred, QAC in this format has a value of zero.

geoQuality (1-byte integer, array size: nscan):
geoQuality is broken into 8 one-bit flags. Some flags represent problems but other flags are informational. Bits 0, 5, and 6 represent problems: 0 = 'good' quality and 1 = 'bad' quality. It is recommended not to use scans when any problem flag is 1. The

informational flags have: 0 = routine conditions and 1 = non-routine conditions. Bit 0 is the most significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**(7-i). Note that good scans may have non-zero geoQuality. Each flag is listed below.

```
Bit Meaning if bit = 1
0   Grossly bad geolocation results:
    Spacecraft position vector magnitude outside range 6715 to 6790 km.
    Z component of midpoint of scan outside range -4100 to 4100 km.
    Distance from S/C to midpoint of scan outside range 500 to 750 km.
1   Unexpectedly large scan to scan jumps in geolocated positions in along and
    cross track directions for first, middle, and last pixels in each scan.
    Allowed deviation from nominal jump in along track motion = 3.0 km (first
    pixel), 3.0 km (middle pixel), and 3.0 km (last pixel).
    Allowed deviation from nominal jump in cross track motion = 3.0 km (first
    pixel), 3.0 km (middle pixel), and 3.0 km (last pixel).
    Bit set in normal mode only.
2   Scan to scan jumps in yaw, pitch, and roll exceed maximum values.  Values
    are :  yaw = 0.005 radians; pitch = 0.005 radians; roll = 0.005 radians.
    Bit set in normal control mode only.
3   In normal mode, yaw outside range (-0.005, 0.005) radians; pitch outside
    range (-0.005, 0.005) radians; roll outside range (-0.005, 0.005) radians.
4   Satellite undergoing maneuvers during which geolocation will be less
    accurate.
5   Summary QA flag for dataQuality:  Set to 1 if bit 0 is 1 or bit 6 is 1,
    i.e. Grossly bad or failed geolocation calculations.
    Science data use not recommended.
6   Geolocation calculations failed (fill values inserted in the per pixel
    geolocation products, but not in metadata).
7   Missing attitude data.  ACS data gap larger than 1.0 seconds.
      Pitch, roll, and yaw are interpolated or extrapolated from nearby data.
```

dataQuality (1-byte integer, array size: nscan):
dataQuality is a flag for overall scan quality. Unless this is 0, the scan data is meaningless to higher science processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

```
Bit  Meaning if bit = 1
0    missing
5    geoQuality indicates bad or missing values

6    validity bits 0-5 not all normal
```

SCorientation (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft

axis +X, which is also the center of the TMI scan. If +X is forward, SCorientation is 0. If -X is forward, SCorientation is 180. If -Y is forward, SCorientation is 90. Values range from 0 to 360 degrees. Special values are defined as:
-8003 Inertial
-8004 Unknown
-9999 Missing value

acsMode (1-byte integer, array size: nscan):

```
Value Meaning
0     Standby
1     Sun Acquire
2     Earth Acquire
3     Yaw Acquire
4     Nominal
5     Yaw Maneuver
6     Delta-H (Thruster)
7     Delta-V (Thruster)
8     CERES Calibration
```

yawUpStat (1-byte integer, array size: nscan):

```
Value Meaning
0     Inaccurate
1     Indeterminate
2     Accurate
```

tmiIsStatus (1-byte integer, array size: nscan):
Bit 0 is the most significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**(8-i) - 1).

```
Bit Meaning
00  Receiver Status (1=ON, 0=OFF)
01  Spin-up Status (1=ON, 0=OFF)
02  Spare Command 1 Status
03  Spare Command 2 Status
04  1 Hz Clock Select (1=A, 0=B)

05  Spare
06  Spare Command 4 Status
07  Spare Command 5 Status
```

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft's trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

navigation (Group)

scPosX (4-byte float, array size: nscan):
The x component of the position (m) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Geocentric Inertial Coordinates are also commonly known as Earth Centered Inertial coordinates. These coordinates will be True of Date (rather than Epoch 2000 which are also commonly used), as interpolated from the data in the Flight Dynamics Facility ephemeris files generated for TRMM.

scPosY (4-byte float, array size: nscan):
The y component of the position (m) of the spacecraft in Geocentric Inertial Coordinates. See scPosX.

scPosZ (4-byte float, array size: nscan):
The z component of the position (m) of the spacecraft in Geocentric Inertial Coordinates. See scPosX.

scVelX (4-byte float, array size: nscan):
The x component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scVelY (4-byte float, array size: nscan):
The y component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scVelZ (4-byte float, array size: nscan):
The z component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scLat (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time.

scLon (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time.

scAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time.

scAttRoll (4-byte float, array size: nscan):
The satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the

rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

scAttPitch (4-byte float, array size: nscan):
The satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

scAttYaw (4-byte float, array size: nscan):
The satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

SensorOrientationMatrix (4-byte float, array size: 3 x 3 x nscan):
SensorOrientationMatrix is the rotation matrix from the instrument coordinate frame to Geocentric Inertial Coordinates at the Scan mid-Time. It is unitless.

greenHourAng (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates.

qualityFlag (1-byte integer, array size: npixel x nscan):
qualityFlag indicates a generalized quality of the retrieved pixel (Range 0 - 99).

```
Ocean Algorithm:
    High:    Good retrieval (uses only entries from TRMM apriori
             database)
    Medium:  Retrieval used extended database (created by lowering SST
             and Freezing level by 3K from TRMM observations)  and/or
             expanded search radius beyond 2K in SST and 3 mm in TPW
    Low:     Retrieval used excessive search radius to find matches in
             apriori database
Land/Coast Algorithm:
    High:    Good retrieval
    Medium:  Not currently used
    Low:     Pixel is ambiguous (Tb depression due to precipitation or
             surface effect)

Valid values include:
    0 : High quality (retrieval is good)
    1 : Medium quality (use with caution)
    2 : Low quality (recommended qualitative use only)
  -99 : Missing value
```

pixelStatus (1-byte integer, array size: npixel x nscan):
If there is no retrieval at a given pixel, pixelStatus explains the reason (Range 0 - 99).

```
  0 : Valid pixel
  1 : Boundary error in landmask
  2 : Boundary error in sea-ice check
  3 : Boundary error in sea surface temperature
  4 : Invalid time
  5 : Invalid latitude/longitude
  6 : Invalid brightness temperature
  7 : Invalid sea surface temperature
  8 : No retrieval due to sea-ice over water
  9 : No retrieval due to sea-ice over coast
 10 : Land/coast screens not able to be applied
 11 : Failure in ocean rain - no match with database profile Tbs
-99 : Missing value
```

surfaceType (1-byte integer, array size: npixel x nscan):
Indicates the type of surface (Range 0 - 99).

```
 10 : Ocean
 11 : Sea ice
 12 : Partial sea ice
 20 : Land
 30 : Coast
-99 : Missing value
```

landAmbiguousFlag (1-byte integer, array size: npixel x nscan):
Defines codes for uncertain/ambiguous retrievals over land (Range 0 - 99). Valid values are:

```
  0 : No information
 13 : Ambiguous T22V / 2 different scattering screens
 14 : Cannot discriminate precip from cold surface
 63 : Light precipitation
 64 : Cold surface
 65 : Grody light precipitation
 66 : Huffman ambiguous
-99 : Missing value
```

landScreenFlag (1-byte integer, array size: npixel x nscan):
Diagnostic codes for rainfall screens over land (Range 0 - 99). Valid values are:

```
  0 : No information
-31 : Land retrieval found ice likely
-41 : Land retrieval found large polarization
      difference due to ice or sand
-51 : Warm 85H and Low 22V, or clear ocean likely in coast retrieval
-61 : Probable coastline in coast retrieval
-99 : Missing value
```

oceanExtendedDbase (1-byte integer, array size: npixel x nscan):
Percent of the extended database entries (i.e., beyond the TRMM database) used in the retrieval (Range 0 - 100). Valid values are:

```
  0 : Only the TRMM database entries are used
  N : N% of the entries from the extended database are used
100 : Only the extended database entries are used
-99 : Missing value
```

oceanSearchRadius (1-byte integer, array size: npixel x nscan):
Expansion of the search radius of the apriori database beyond the initial SST and TPW search range. The profiles for the rain_ocean procedure are grouped by SST and TPW. The individual pixels TPW and SST are used to retrieve a group of pixels from the database.

If there are fewer than 1000 profile clusters found, the search radius is expanded. (Range 0 - 99). Valid values are:

```
   0 : Default search radius used
   1 : Search radius expanded by +/- 1 mm in TPW and +/- 1 degree in SST
   N : Search radius expanded by +/- N mm in TPW and +/- N degrees in SST
 -99 : Missing value
```

chiSquared (2-byte integer, array size: npixel x nscan):
Error diagnostic for Optimal Estimation calculation of TPW and wind speed. Values greater than the number of channels (9 for TMI) should be considered suspect, with values greater than 18 of limited use. Rainfall is possible above these values. Values could range from 0 to 10000, but should be less than 100. The missing value is -9999.9.

probabilityOfPrecip (1-byte integer, array size: npixel x nscan):
A diagnostic variable, in percent, defining the fraction of raining vs. non-raining Dbase profiles that make up the final solution. Values range from 0 to 100 percent. Special values are defined as:
-99 Missing value

sunGlintAngle (1-byte integer, array size: npixel x nscan):
Conceptually, the angle between the sun and the instrument view direction as reflected off the Earth's surface. More specifically, define a Sun Vector from the viewed pixel location on the earth ellipsiod-model surface to the sun. Also define an Inverse Satellite Vector from the pixel to the satellite. Then reflect the Inverse Satellite Vector off the earth's surface at the pixel location to form the Reflected Satellite View Vector. sunGlintAngle is the angular separation between the Reflected Satellite View Vector and the Sun Vector. When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like) sun reflection. Values range from 0 to 180 degrees. Special values are defined as:
-99 Missing value

freezingHeight (2-byte integer, array size: npixel x nscan):
The height, in meters, of the 0^{o}C isotherm above the earth ellipsoid. The missing value is -9999.

surfacePrecipitation (4-byte float, array size: npixel x nscan):
The instantaneous precipitation rate at the surface for each pixel. Check pixelStatus for a valid retrieval. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

convectPrecipitation (4-byte float, array size: npixel x nscan):
The instantaneous convective precipitation rate at the surface for each pixel. Check pixelStatus for a valid retrieval. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

surfaceRain (4-byte float, array size: npixel x nscan):
The instantaneous rain rate (liquid portion of precipitation) at the surface for each pixel. Check pixelStatus for a valid retrieval. Values are in mm/hr. Special values are defined

as:
-9999.9 Missing value

cloudWaterPath (4-byte float, array size: npixel x nscan):
Total cloud liquid water in the column. Values range from 0 to 3000 kg/m^2. Special values are defined as:
-9999.9 Missing value

rainWaterPath (4-byte float, array size: npixel x nscan):
Total rain water in the column. Values range from 0 to 3000 kg/m^2. Special values are defined as:
-9999.9 Missing value

iceWaterPath (4-byte float, array size: npixel x nscan):
Total of all ice species (including cloud ice and precipitation ice) in the column. Values range from 0 to 3000 kg/m^2. Special values are defined as:
-9999.9 Missing value

seaSurfaceTemperature (4-byte float, array size: npixel x nscan):
Sea surface temperature. Values in degrees K. The missing value is -9999.9.

totalPrecipitableWater (4-byte float, array size: npixel x nscan):
Liquid equivalent of the total water vapor column. Values range from 0 to 75 mm. Special values are defined as:
-9999.9 Missing value

windSpeed (4-byte float, array size: npixel x nscan):
Wind speed at the sea surface. Values in m/s, 20m above the surface. The missing value is -9999.9.

freezingHeightIndex (1-byte integer, array size: npixel x nscan):
Freezing Height Index in the cluster array. See description below clusterScale. Values range from 1 to 13. Special values are defined as:
-99 Missing value

clusterNumber (1-byte integer, array size: nspecies x npixel x nscan):
Cluster Number in the cluster array. See clusterScale description below clusterScale. Values range from 1 to 100. Special values are defined as:
-99 Missing value

clusterScale (4-byte float, array size: nspecies x npixel x nscan):
clusterScale is used to scale the values of the cluster array.

```
In order to recover values in a profile use
the clusterNumber, clusterScale and freezingHeightIndex
parameters and select your species and level:

Where:
  L = profile level (1-18) Top of each level
      specified in HgtLayerTop
```

```
  S = species(1-6)
      1 = cloud water content
      2 = rain water content
      3 = cloud ice content
      4 = snow water content
      5 = graupel water content
      6 = latent heat
  F = freezingHeightIndex
  C = clusterNumber

In a Fortran program,
  Profile Value = clusterScale * cluster(S,F,L,C)

In a C program,
  Profile Value = clusterScale * cluster[C-1][L-1][F-1][S-1]
```

C Structure Header file:

```
#ifndef _TK_2A12_H_
#define _TK_2A12_H_

#ifndef _L2A12_NAVIGATION_
#define _L2A12_NAVIGATION_

typedef struct {
    float scPosX;
    float scPosY;
    float scPosZ;
    float scVelX;
    float scVelY;
    float scVelZ;
    float scLat;
    float scLon;
    float scAlt;
    float scAttRoll;
    float scAttPitch;
    float scAttYaw;
    float SensorOrientationMatrix[3][3];
    float greenHourAng;
} L2A12_NAVIGATION;

#endif
```

```
#ifndef _L2A12_SCANSTATUS_
#define _L2A12_SCANSTATUS_

typedef struct {
    signed char missing;
    signed char validity;
    signed char qac;
    signed char geoQuality;
    signed char dataQuality;
    short SCorientation;
    signed char acsMode;
    signed char yawUpStat;
    signed char tmiIsStatus;
    double FractionalGranuleNumber;
} L2A12_SCANSTATUS;

#endif

#ifndef _L2A12_SCANTIME_
#define _L2A12_SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
} L2A12_SCANTIME;

#endif

#ifndef _L2A12_SWATH_
#define _L2A12_SWATH_

typedef struct {
    L2A12_SCANTIME ScanTime;
    float Latitude[208];
    float Longitude[208];
    L2A12_SCANSTATUS scanStatus;
    L2A12_NAVIGATION navigation;
```

```
    signed char qualityFlag[208];
    signed char pixelStatus[208];
    signed char surfaceType[208];
    signed char landAmbiguousFlag[208];
    signed char landScreenFlag[208];
    signed char oceanExtendedDbase[208];
    signed char oceanSearchRadius[208];
    short chiSquared[208];
    signed char probabilityOfPrecip[208];
    signed char sunGlintAngle[208];
    short freezingHeight[208];
    float surfacePrecipitation[208];
    float convectPrecipitation[208];
    float surfaceRain[208];
    float cloudWaterPath[208];
    float rainWaterPath[208];
    float iceWaterPath[208];
    float seaSurfaceTemperature[208];
    float totalPrecipitableWater[208];
    float windSpeed[208];
    signed char freezingHeightIndex[208];
    signed char clusterNumber[208][6];
    float clusterScale[208][6];
} L2A12_SWATH;

#endif

#ifndef _L2A12_DATAHEADER_
#define _L2A12_DATAHEADER_

typedef struct {
    float heightLayerTop[28];
    float cluster[100][28][13][6];
} L2A12_DATAHEADER;

#endif

#endif
```

Fortran Structure Header file:

```
STRUCTURE /L2A12_NAVIGATION/
    REAL*4 scPosX
```

```
    REAL*4 scPosY
    REAL*4 scPosZ
    REAL*4 scVelX
    REAL*4 scVelY
    REAL*4 scVelZ
    REAL*4 scLat
    REAL*4 scLon
    REAL*4 scAlt
    REAL*4 scAttRoll
    REAL*4 scAttPitch
    REAL*4 scAttYaw
    REAL*4 SensorOrientationMatrix(3,3)
    REAL*4 greenHourAng
END STRUCTURE

STRUCTURE /L2A12_SCANSTATUS/
    BYTE missing
    BYTE validity
    BYTE qac
    BYTE geoQuality
    BYTE dataQuality
    INTEGER*2 SCorientation
    BYTE acsMode
    BYTE yawUpStat
    BYTE tmiIsStatus
    REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /L2A12_SCANTIME/
    INTEGER*2 Year
    BYTE Month
    BYTE DayOfMonth
    BYTE Hour
    BYTE Minute
    BYTE Second
    INTEGER*2 MilliSecond
    INTEGER*2 DayOfYear
END STRUCTURE

STRUCTURE /L2A12_SWATH/
    RECORD /L2A12_SCANTIME/ ScanTime
    REAL*4 Latitude(208)
    REAL*4 Longitude(208)
```

```
    RECORD /L2A12_SCANSTATUS/ scanStatus
    RECORD /L2A12_NAVIGATION/ navigation
    BYTE qualityFlag(208)
    BYTE pixelStatus(208)
    BYTE surfaceType(208)
    BYTE landAmbiguousFlag(208)
    BYTE landScreenFlag(208)
    BYTE oceanExtendedDbase(208)
    BYTE oceanSearchRadius(208)
    INTEGER*2 chiSquared(208)
    BYTE probabilityOfPrecip(208)
    BYTE sunGlintAngle(208)
    INTEGER*2 freezingHeight(208)
    REAL*4 surfacePrecipitation(208)
    REAL*4 convectPrecipitation(208)
    REAL*4 surfaceRain(208)
    REAL*4 cloudWaterPath(208)
    REAL*4 rainWaterPath(208)
    REAL*4 iceWaterPath(208)
    REAL*4 seaSurfaceTemperature(208)
    REAL*4 totalPrecipitableWater(208)
    REAL*4 windSpeed(208)
    BYTE freezingHeightIndex(208)
    BYTE clusterNumber(6,208)
    REAL*4 clusterScale(6,208)
END STRUCTURE

STRUCTURE /L2A12_DATAHEADER/
    REAL*4 heightLayerTop(28)
    REAL*4 cluster(6,13,28,100)
END STRUCTURE
```

7.6 2A21 - Surface Cross Section

2A21, "Surface Cross Section," computes the normalized surface cross section. If rain is present, it will also compute path attenuation and its associated reliability factor. Figure 1.2.1-1 shows the structure of the 2A21 product in terms of the component objects and their sizes.

Dimension definitions:

nscan	var	Number of scans in the granule.
nray	49	Number of angle bins in each scan.
refmethod	5	Number of reference methods.
direction	2	Number of refScanID directions.
distance	2	Number of refScanID distances.

Figure 35 through Figure 39 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for TRMM Products for details.

InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for TRMM Products for details.

NavigationRecord (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for TRMM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for TRMM Products for details.

JAXAInfo (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by PR algorithms only. See Metadata for TRMM Products for details.

Swath (Swath)

SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for TRMM Products for details.

ScanTime (Group)

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

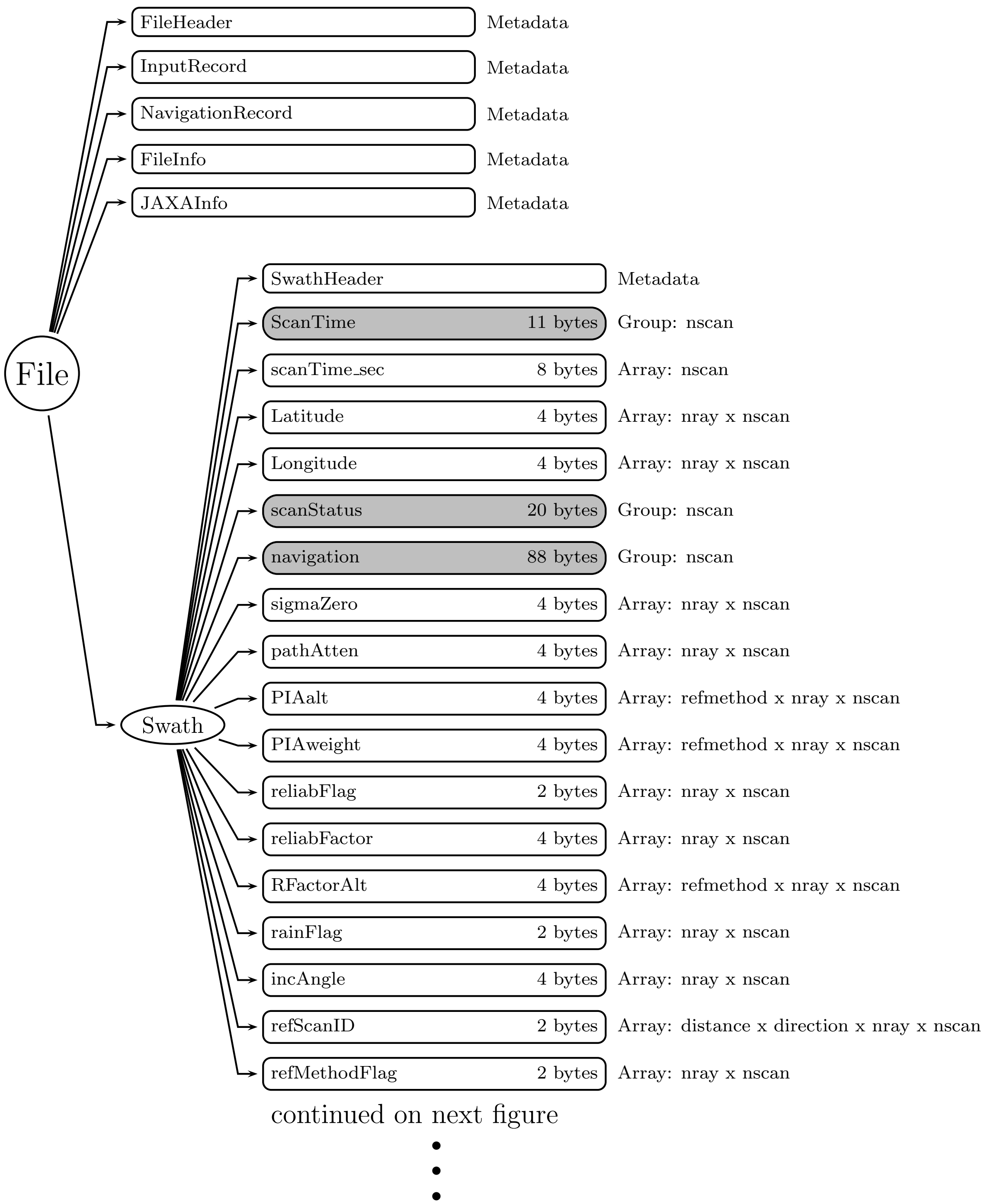


Figure 35: Data Format Structure for 2A21, Surface Cross Section

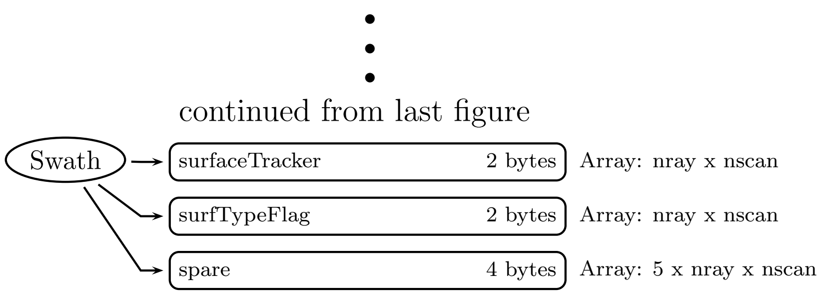


Figure 36: Data Format Structure for 2A21, Surface Cross Section

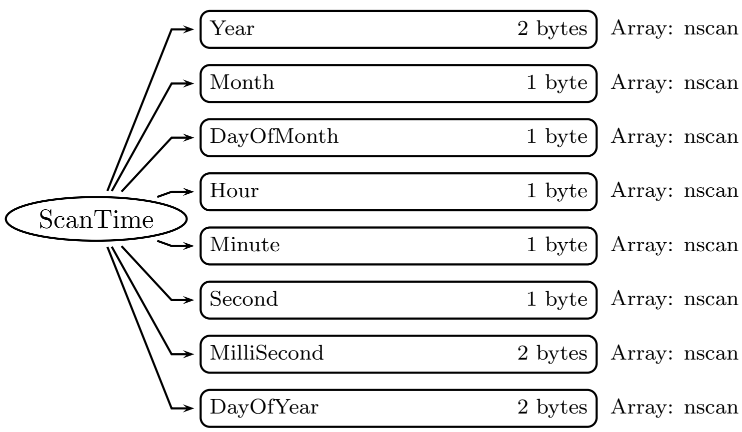


Figure 37: Data Format Structure for 2A21, ScanTime

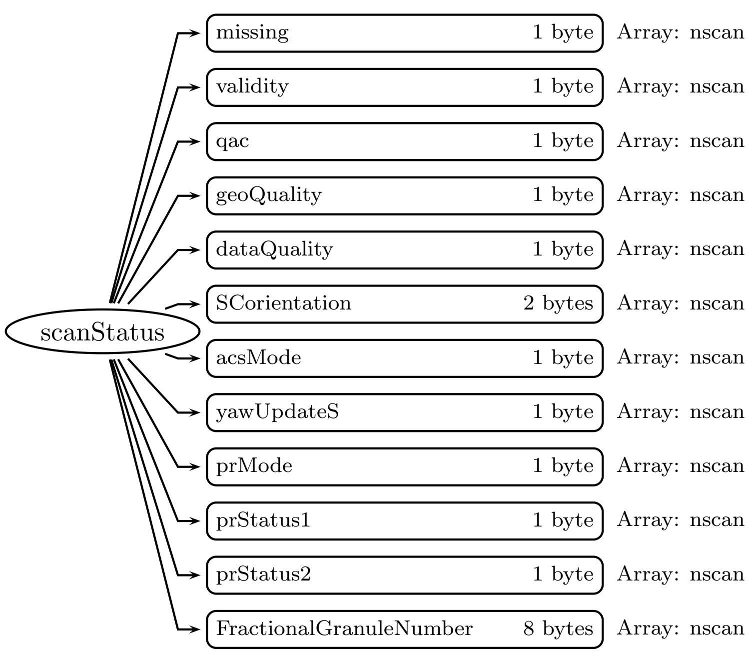


Figure 38: Data Format Structure for 2A21, scanStatus

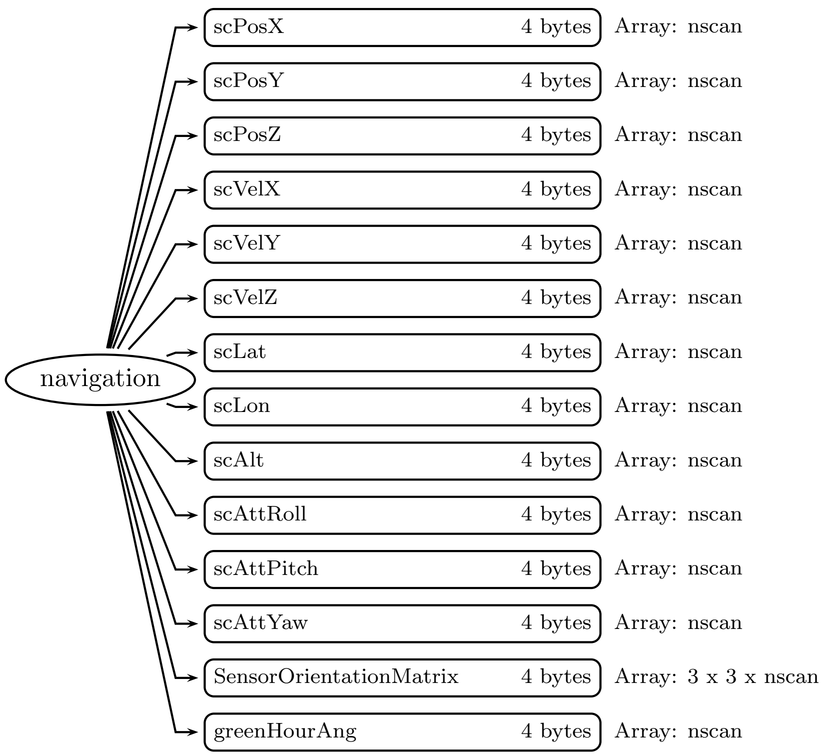


Figure 39: Data Format Structure for 2A21, navigation

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

scanTime_sec (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: nray x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: nray x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scanStatus (Group)

missing (1-byte integer, array size: nscan):
Missing indicates whether information is contained in the scan data. The values are:

```
0 Scan data elements contain information
1 Scan was missing in the telemetry data
2 Scan data contains no elements with rain
```

validity (1-byte integer, array size: nscan):
Validity is a summary of status modes. If all status modes are routine, all bits in Validity = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. Validity does not assess data or geolocation quality. Validity is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

```
Bit Meaning if bit = 1
0   Spare (always 0)
1   Non-routine spacecraft orientation (2 or 3)
2   Non-routine ACS mode (other than 4)
3   Non-routine yaw update status (0 or 1)
4   Non-routine instrument status (other than 1)
5   Non-routine QAC (non-zero)
6   Spare (always 0)
7   Spare (always 0)
```

qac (1-byte integer, array size: nscan):
The Quality and Accounting Capsule of the Science packet as it appears in Level-0 data.

If no QAC is given in Level-0, which means no decoding errors occurred, QAC in this format has a value of zero.

geoQuality (1-byte integer, array size: nscan):
Geolocation quality is a summary of geolocation quality in the scan. A zero integer value indicates 'good' geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

```
Bit Meaning if bit = 1
0   latitude limit error
1   geolocation
2   attitude change rate limit error
3   attitude limit error
4   satellite undergoing maneuvers
5   using predictive orbit data
6   geolocation calculation error
7   not used
```

dataQuality (1-byte integer, array size: nscan):
Data quality is a summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

```
Bit Meaning if bit = 1
0   missing
5   Geolocation Quality is not normal
6   Validity is not normal
```

SCorientation (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the TMI scan. If +X is forward, SCorientation is 0. If -X is forward, SCorientation is 180. If -Y is forward, SCorientation is 90. Values range from 0 to 360 degrees. Special values are defined as:

-8003	Inertial
-8004	Unknown
-9999	Missing value

acsMode (1-byte integer, array size: nscan):

```
Value Meaning
0     Standby
1     Sun Acquire
```

```
2     Earth Acquire
3     Yaw Acquire
4     Nominal
5     Yaw Maneuver
6     Delta-H (Thruster)
7     Delta-V (Thruster)
8     CERES Calibration
```

yawUpdateS (1-byte integer, array size: nscan):

```
Value Meaning
0     Inaccurate
1     Indeterminate
2     Accurate
```

prMode (1-byte integer, array size: nscan):

```
Value Meaning
1     Observation Mode
2     Other Mode
```

prStatus1 (1-byte integer, array size: nscan):
This status is a warning for scan data. Unless this is 0, the scan data may include a little questionable value though it is not a problem (such as break of caution limit). This field is used only for NASDA's data analysis.

prStatus2 (1-byte integer, array size: nscan):
Initialization in Onboard Surface Search Algorithm.

```
Value Meaning
0     Not initialized
1     Initialized
```

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft's trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

navigation (Group)

scPosX (4-byte float, array size: nscan):
The x component of the position (m) of the spacecraft in Geocentric Inertial Coordinates

at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Geocentric Inertial Coordinates are also commonly known as Earth Centered Inertial coordinates. These coordinates will be True of Date (rather than Epoch 2000 which are also commonly used), as interpolated from the data in the Flight Dynamics Facility ephemeris files generated for TRMM.

scPosY (4-byte float, array size: nscan):
The y component of the position (m) of the spacecraft in Geocentric Inertial Coordinates. See scPosX.

scPosZ (4-byte float, array size: nscan):
The z component of the position (m) of the spacecraft in Geocentric Inertial Coordinates. See scPosX.

scVelX (4-byte float, array size: nscan):
The x component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scVelY (4-byte float, array size: nscan):
The y component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scVelZ (4-byte float, array size: nscan):
The z component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scLat (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time.

scLon (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time.

scAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time.

scAttRoll (4-byte float, array size: nscan):
The satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

scAttPitch (4-byte float, array size: nscan):
The satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

scAttYaw (4-byte float, array size: nscan):
The satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

SensorOrientationMatrix (4-byte float, array size: 3 x 3 x nscan):
SensorOrientationMatrix is the rotation matrix from the instrument coordinate frame to Geocentric Inertial Coordinates at the Scan mid-Time. It is unitless.

greenHourAng (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates.

sigmaZero (4-byte float, array size: nray x nscan):
The normalized surface cross section. Values range from -50 to 50 dB. Special values are defined as:
-9999.9 Missing value

pathAtten (4-byte float, array size: nray x nscan):
This is the best estimate of 2-way path integrated attenuation when rain is present and the reference is reliable or marginally reliable (see reliabFlag). This best estimate is a combination of various reference methods (see PIAweight). Values range from -50 to 50 dB. Special values are defined as:
-9999.9 Missing value

PIAalt (4-byte float, array size: refmethod x nray x nscan):
Alternate estimates of the 2-way path integrated attenuation when rain is present and the reference is reliable or marginally reliable (see RFactorAlt). The order in zero-based notation of the reference methods (refmethod) is:

```
0 - spatial, forward
1 - hybrid, forward (ocean only)
2 - spatial, backward
3 - hybrid, backward (ocean only)
4 - temporal (should be the same forward or backward)
```

Values range from -50 to 50 dB. Special values are defined as:
-9999.9 Missing value

PIAweight (4-byte float, array size: refmethod x nray x nscan):
A vector containing floating point weights indicating the combination of PIA estimates used to establish the best 2-way path integrated attenuation (pathAtten). Each value gives the relative weight used for each reference method. The order in zero-based notation of the reference methods (refmethod) is:

```
0 - spatial, forward
1 - hybrid, forward (ocean only)
2 - spatial, backward
3 - hybrid, backward (ocean only)
4 - temporal (should be the same forward or backward)
```

Values range from 0 to 1. Special values are defined as:
-9999.9 Missing value

reliabFlag (2-byte integer, array size: nray x nscan):
This flag indicates the reliability of the pathAtten estimate. Flag values are:

```
1 - PIA estimate is reliable
2 - PIA estimate is marginally reliable
3 - PIA estimate is unreliable
4 - PIA estimate is a lower bound to the path attenuation
9 - No PIA estimate, no-rain in ifov.
```

Values range from 1 to 9. Special values are defined as:
-9999 Missing value

reliabFactor (4-byte float, array size: nray x nscan):
The Reliability Factor is the ratio of the estimated value of path attenuation to the standard deviation associated with the mean value of the reference estimate. Values range from -10 to 10. Special values are defined as:
-9999.9 Missing value

RFactorAlt (4-byte float, array size: refmethod x nray x nscan):
Reliability factor for the alternate PIA estimates in PIAalt for each refmethod. The Reliability Factor is the ratio of the estimated value of path attenuation to the standard deviation associated with the mean value of the reference estimate. The order in zero-based notation of the reference methods (refmethod) is:

```
0 - spatial, forward
1 - hybrid, forward (ocean only)
2 - spatial, backward
3 - hybrid, backward (ocean only)
4 - temporal (should be the same forward or backward)
```

Values range from -10 to 10. Special values are defined as:
-9999.9 Missing value

rainFlag (2-byte integer, array size: nray x nscan):
The Rain Flag has the following values:

```
0: no rain;
1: rain present.
```

Values range from 0 to 1. Special values are defined as:
-9999 Missing value

incAngle (4-byte float, array size: nray x nscan):
The Incident Angle is the angle, in degrees, between the PR nadir and the radar beam. Values range from -30 to 30 degrees. Special values are defined as:
-9999.9 Missing value

refScanID (2-byte integer, array size: distance x direction x nray x nscan):
Provides scan information for the nearest and farthest reference points for the along-track methods. The values are (Current Scan) - (Reference Scan). These values are positive for the Forward estimates and negative for the Backward estimates. The dimensions in C notation are:

```
[0][0] - Forward - Near reference
[0][1] - Forward - Far reference
[1][0] - Backward - Near reference
[1][1] - Backward - Far reference
```

Values range from -9300 to 9300 scanNum. Special values are defined as:
-9999 Missing value

refMethodFlag (2-byte integer, array size: nray x nscan):
Flag for the reference method associated with the best PIA estimate (pathAtten).

```
3 - insufficient number of data points
4 - unknown background type
5 - no-rain case and low SNR, no update to reference data
9 - no rain case
```

Values range from 3 to 9. Special values are defined as:
-9999 Missing value

surfaceTracker (2-byte integer, array size: nray x nscan):
Status of surface tracker as a flag value.

```
1 - surface tracker locked - central angle bin
2 - surface tracker unlocked - central angle bin
3 - peak surface return at normally sampled gate,
    outside central swath
4 - peak surface return not at normally sampled gate,
    outside central swath
```

Values range from 1 to 4. Special values are defined as:
-9999 Missing value

surfTypeFlag (2-byte integer, array size: nray x nscan):
Flag value for the surface type for a given ifov. Flag values are:

```
0 - Ocean
1 - Land
2 - Coast
3 - Unknown or of a category other than those above or 'mixed'
```

Values range from 0 to 3. Special values are defined as:
-9999 Missing value

spare (4-byte float, array size: 5 x nray x nscan):
Contains developer output.

C Structure Header file:

```
#ifndef _TK_2A21_H_
#define _TK_2A21_H_

#ifndef _L2A21_NAVIGATION_
#define _L2A21_NAVIGATION_

typedef struct {
    float scPosX;
    float scPosY;
```

```
    float scPosZ;
    float scVelX;
    float scVelY;
    float scVelZ;
    float scLat;
    float scLon;
    float scAlt;
    float scAttRoll;
    float scAttPitch;
    float scAttYaw;
    float SensorOrientationMatrix[3][3];
    float greenHourAng;
} L2A21_NAVIGATION;

#endif

#ifndef _L2A21_SCANSTATUS_
#define _L2A21_SCANSTATUS_

typedef struct {
    signed char missing;
    signed char validity;
    signed char qac;
    signed char geoQuality;
    signed char dataQuality;
    short SCorientation;
    signed char acsMode;
    signed char yawUpdateS;
    signed char prMode;
    signed char prStatus1;
    signed char prStatus2;
    double FractionalGranuleNumber;
} L2A21_SCANSTATUS;

#endif

#ifndef _L2A21_SCANTIME_
#define _L2A21_SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
```

```
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
} L2A21_SCANTIME;

#endif

#ifndef _L2A21_SWATH_
#define _L2A21_SWATH_

typedef struct {
    L2A21_SCANTIME ScanTime;
    double scanTime_sec;
    float Latitude[49];
    float Longitude[49];
    L2A21_SCANSTATUS scanStatus;
    L2A21_NAVIGATION navigation;
    float sigmaZero[49];
    float pathAtten[49];
    float PIAalt[49][5];
    float PIAweight[49][5];
    short reliabFlag[49];
    float reliabFactor[49];
    float RFactorAlt[49][5];
    short rainFlag[49];
    float incAngle[49];
    short refScanID[49][2][2];
    short refMethodFlag[49];
    short surfaceTracker[49];
    short surfTypeFlag[49];
    float spare[49][5];
} L2A21_SWATH;

#endif

#endif
```

Fortran Structure Header file:

```
STRUCTURE /L2A21_NAVIGATION/
    REAL*4 scPosX
```

```
    REAL*4 scPosY
    REAL*4 scPosZ
    REAL*4 scVelX
    REAL*4 scVelY
    REAL*4 scVelZ
    REAL*4 scLat
    REAL*4 scLon
    REAL*4 scAlt
    REAL*4 scAttRoll
    REAL*4 scAttPitch
    REAL*4 scAttYaw
    REAL*4 SensorOrientationMatrix(3,3)
    REAL*4 greenHourAng
END STRUCTURE

STRUCTURE /L2A21_SCANSTATUS/
    BYTE missing
    BYTE validity
    BYTE qac
    BYTE geoQuality
    BYTE dataQuality
    INTEGER*2 SCorientation
    BYTE acsMode
    BYTE yawUpdateS
    BYTE prMode
    BYTE prStatus1
    BYTE prStatus2
    REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /L2A21_SCANTIME/
    INTEGER*2 Year
    BYTE Month
    BYTE DayOfMonth
    BYTE Hour
    BYTE Minute
    BYTE Second
    INTEGER*2 MilliSecond
    INTEGER*2 DayOfYear
END STRUCTURE

STRUCTURE /L2A21_SWATH/
    RECORD /L2A21_SCANTIME/ ScanTime
```

```
    REAL*8 scanTime_sec
    REAL*4 Latitude(49)
    REAL*4 Longitude(49)
    RECORD /L2A21_SCANSTATUS/ scanStatus
    RECORD /L2A21_NAVIGATION/ navigation
    REAL*4 sigmaZero(49)
    REAL*4 pathAtten(49)
    REAL*4 PIAalt(5,49)
    REAL*4 PIAweight(5,49)
    INTEGER*2 reliabFlag(49)
    REAL*4 reliabFactor(49)
    REAL*4 RFactorAlt(5,49)
    INTEGER*2 rainFlag(49)
    REAL*4 incAngle(49)
    INTEGER*2 refScanID(2,2,49)
    INTEGER*2 refMethodFlag(49)
    INTEGER*2 surfaceTracker(49)
    INTEGER*2 surfTypeFlag(49)
    REAL*4 spare(5,49)
END STRUCTURE
```

7.7 2A23 - PR Qualitative

2A23, "PR Qualitative", produces a Rain/No-rain flag. If rain is present, this algorithm will detect the bright band, determine the heights of the bright band and the storm, and classify rain types. The following sections describe the structure and contents of the format.

Dimension definitions:

nscan	var	Number of scans in the granule.
nray	49	Number of angle bins in each scan.

Figure 40 through Figure 44 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for TRMM Products for details.

InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for TRMM Products for details.

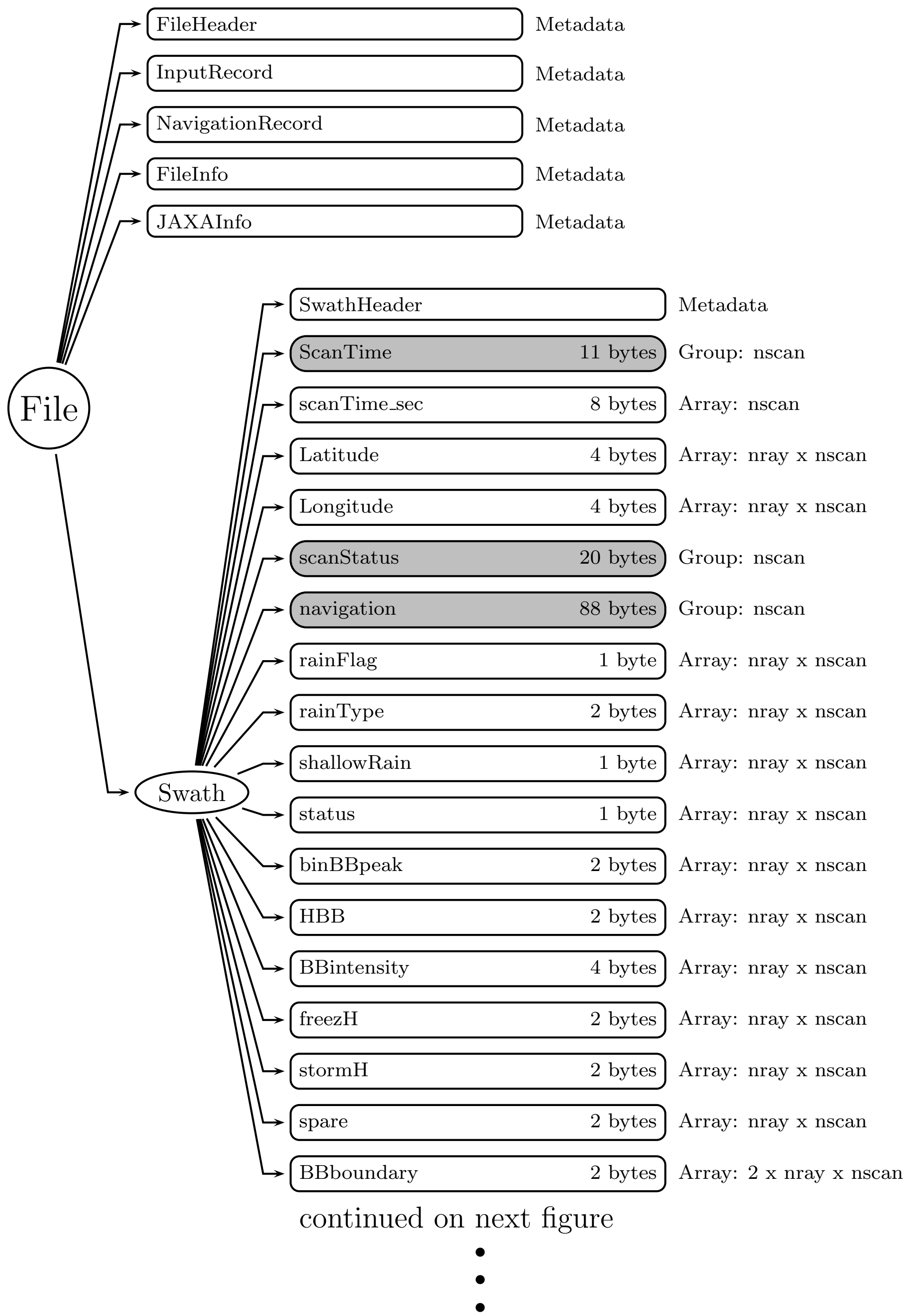


Figure 40: Data Format Structure for 2A23, PR Qualitative

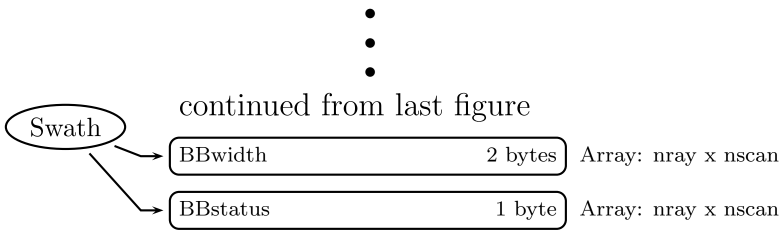


Figure 41: Data Format Structure for 2A23, PR Qualitative

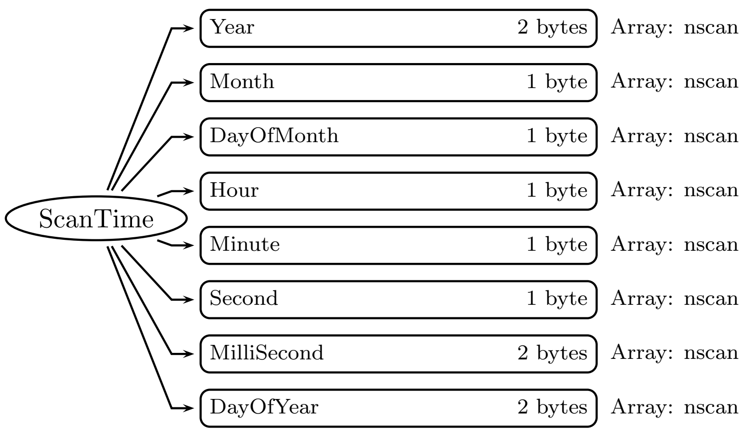


Figure 42: Data Format Structure for 2A23, ScanTime

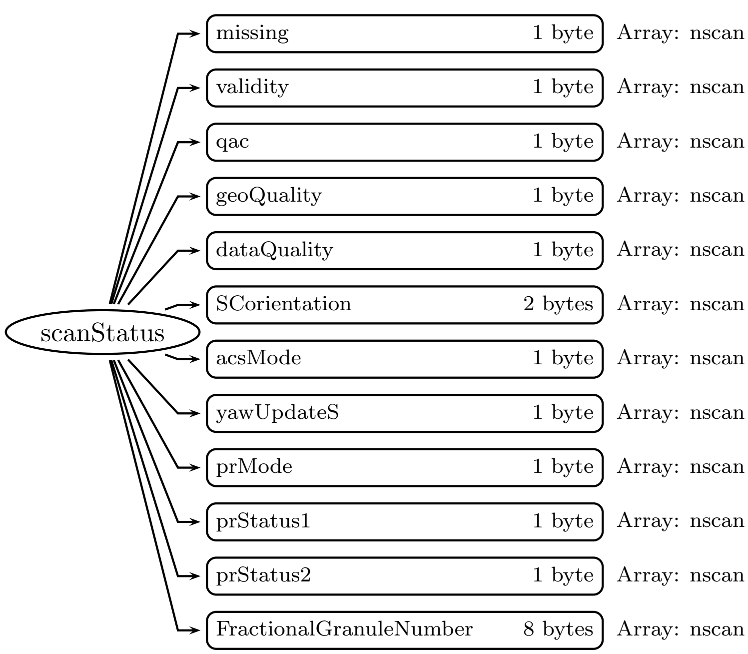


Figure 43: Data Format Structure for 2A23, scanStatus

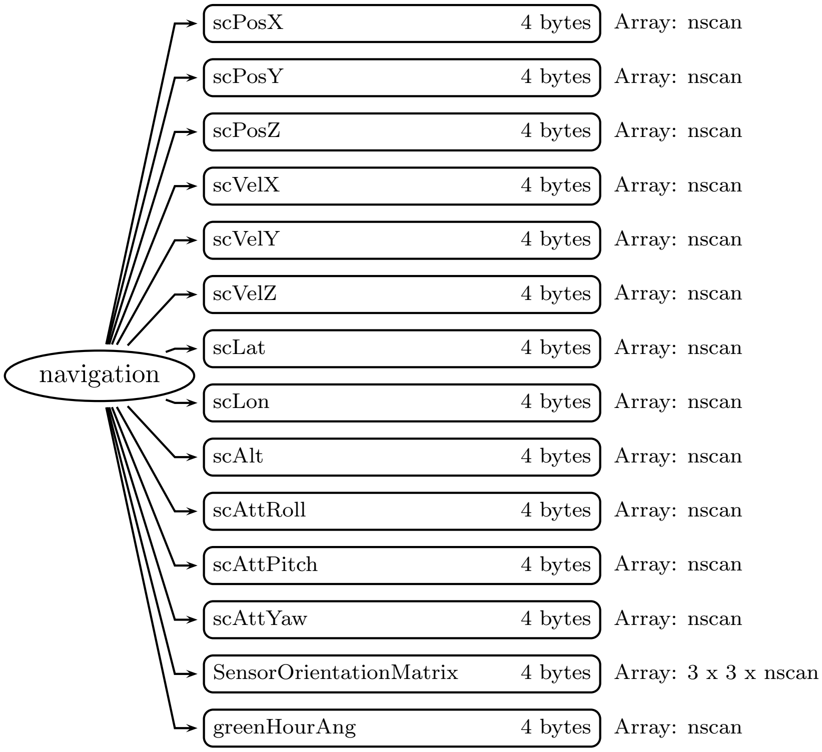


Figure 44: Data Format Structure for 2A23, navigation

NavigationRecord (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for TRMM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for TRMM Products for details.

JAXAInfo (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by PR algorithms only. See Metadata for TRMM Products for details.

Swath (Swath)

SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for TRMM Products for details.

ScanTime (Group)

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined

as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

scanTime_sec (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: nray x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: nray x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scanStatus (Group)

missing (1-byte integer, array size: nscan):
Missing indicates whether information is contained in the scan data. The values are:

```
0 Scan data elements contain information
1 Scan was missing in the telemetry data
2 Scan data contains no elements with rain
```

validity (1-byte integer, array size: nscan):
Validity is a summary of status modes. If all status modes are routine, all bits in Validity = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. Validity does not assess data or geolocation quality. Validity is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

```
Bit Meaning if bit = 1
0   Spare (always 0)
1   Non-routine spacecraft orientation (2 or 3)
2   Non-routine ACS mode (other than 4)
3   Non-routine yaw update status (0 or 1)
4   Non-routine instrument status (other than 1)
5   Non-routine QAC (non-zero)
6   Spare (always 0)
7   Spare (always 0)
```

qac (1-byte integer, array size: nscan):
The Quality and Accounting Capsule of the Science packet as it appears in Level-0 data. If no QAC is given in Level-0, which means no decoding errors occurred, QAC in this format has a value of zero.

geoQuality (1-byte integer, array size: nscan):
Geolocation quality is a summary of geolocation quality in the scan. A zero integer value indicates 'good' geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

```
Bit Meaning if bit = 1
0   latitude limit error
1   geolocation
2   attitude change rate limit error
3   attitude limit error
4   satellite undergoing maneuvers
5   using predictive orbit data
6   geolocation calculation error
7   not used
```

dataQuality (1-byte integer, array size: nscan):
Data quality is a summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

```
Bit Meaning if bit = 1
0   missing
5   Geolocation Quality is not normal
6   Validity is not normal
```

SCorientation (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the TMI scan. If +X is forward, SCorientation is 0. If -X is forward, SCorientation is 180. If -Y is forward, SCorientation is 90. Values range from 0 to 360 degrees. Special values are defined as:

-8003	Inertial
-8004	Unknown
-9999	Missing value

acsMode (1-byte integer, array size: nscan):

```
Value Meaning
0     Standby
1     Sun Acquire
2     Earth Acquire
3     Yaw Acquire
4     Nominal
5     Yaw Maneuver
6     Delta-H (Thruster)
7     Delta-V (Thruster)
8     CERES Calibration
```

yawUpdateS (1-byte integer, array size: nscan):

```
Value Meaning
0     Inaccurate
1     Indeterminate
2     Accurate
```

prMode (1-byte integer, array size: nscan):

```
Value Meaning
1     Observation Mode
2     Other Mode
```

prStatus1 (1-byte integer, array size: nscan):
This status is a warning for scan data. Unless this is 0, the scan data may include a little questionable value though it is not a problem (such as break of caution limit). This field is used only for NASDA's data analysis.

prStatus2 (1-byte integer, array size: nscan):
Initialization in Onboard Surface Search Algorithm.

```
Value Meaning
0     Not initialized
1     Initialized
```

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft's trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

navigation (Group)

scPosX (4-byte float, array size: nscan):
The x component of the position (m) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Geocentric Inertial Coordinates are also commonly known as Earth Centered Inertial coordinates. These coordinates will be True of Date (rather than Epoch 2000 which are also commonly used), as interpolated from the data in the Flight Dynamics Facility ephemeris files generated for TRMM.

scPosY (4-byte float, array size: nscan):
The y component of the position (m) of the spacecraft in Geocentric Inertial Coordinates. See scPosX.

scPosZ (4-byte float, array size: nscan):
The z component of the position (m) of the spacecraft in Geocentric Inertial Coordinates. See scPosX.

scVelX (4-byte float, array size: nscan):
The x component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scVelY (4-byte float, array size: nscan):
The y component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scVelZ (4-byte float, array size: nscan):
The z component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scLat (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time.

scLon (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time.

scAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time.

scAttRoll (4-byte float, array size: nscan):
The satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

scAttPitch (4-byte float, array size: nscan):
The satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

scAttYaw (4-byte float, array size: nscan):
The satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is

toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

SensorOrientationMatrix (4-byte float, array size: 3 x 3 x nscan):
SensorOrientationMatrix is the rotation matrix from the instrument coordinate frame to Geocentric Inertial Coordinates at the Scan mid-Time. It is unitless.

greenHourAng (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates.

rainFlag (1-byte integer, array size: nray x nscan):
The Rain Flag is identical to the Minimum Echo Flag of 1C21:

```
 0: no rain
10: rain possible
11: rain possible (echo greater than rain threshold 1 in clutter region)
12: rain possible (echo greater than rain threshold 2 in clutter region)
13: rain possible
15: rain probable
20: rain certain
```

rainType (2-byte integer, array size: nray x nscan):
The Rain Type is set as follows:

```
100: Stratiform.
When R\_type\_V[i] = T\_stra,
and R\_type\_H[i] = T\_stra.
(BB detected.)

105: Stratiform.
--- added in V7.
When R\_type\_V[i] = T\_stra,
(BB detected.)
and R\_type\_H[i] = T\_stra,
But storm top (determined by 2A23) is too high.

110: Stratiform.
When R\_type\_V[i] = T\_stra,
(BB detected.)
and R\_type\_H[i] = T\_other.
```

```
115: Stratiform.
--- added in V7.
When R\_type\_V[i] = T\_stra,
(BB detected.)
and R\_type\_H[i] = T\_other.
But storm top (determined by 2A23) is too high.

120: Probably stratiform. (BB may exist but not detected.)
When R\_type\_V[i] = T\_other,
and R\_type\_H[i] = T\_stra.

130: Maybe stratiform.
When R\_type\_V[i] = T\_stra,
and R\_type\_H[i] = T\_conv.
(BB detected.)

135: Maybe stratiform.
--- added in V7.
When R\_type\_V[i] = T\_stra,
(BB detection certain.)
and R\_type\_H[i] = T\_conv.
But storm top (determined by 2A23) is too high.

140: Maybe stratiform. (BB hardly expected.)
When R\_type\_V[i] = T\_other,
and R\_type\_H[i] = T\_stra.

152: Maybe stratiform:
When R\_type\_V[i] = T\_other,
R\_type\_H[i] = T\_stra,
and shallowRain[i] = 20 or 21.
(Shallow non-isolated is detected.)

160: Maybe stratiform, but rain hardly expected near surface.
BB may exist but is not detected.
When R\_type\_V[i] = T\_other,
and R\_type\_H[i] = T\_stra.

170: Maybe stratiform, but rain hardly expected near surface.
BB hardly expected. Maybe cloud only.
Distinction between 170 and 300 is very small.
When R\_type\_V[i] = T\_other,
```

```
and R\_type\_H[i] = T\_stra.

200: Convective.
When R\_type\_V[i] = T\_conv,
and R\_type\_H[i] = T\_conv.
210: Convective.
When R\_type\_V[i] = T\_other,
and R\_type\_H[i] = T\_conv;

220: Convective
When R\_type\_V[i] = T\_conv,
and R\_type\_H[i] = T\_other;

230: Probably convective. ---> Re-introduced in V7.
When R\_type\_V[i] = T\_stra;
(BB exists)
R\_type\_H[i] = T\_conv;
and Z below BB is strong.

235: Probably convective. ---> Added in V7.
When R\_type\_V[i] = T\_other;
R\_type\_H[i] = T\_stra;
But storm top (determined by 2A23) is too high.

237: Probably convective. ---> Added in V7.
When R\_type\_V[i] = T\_other;
R\_type\_H[i] = T\_stra;
But the cell size is small.

240: Maybe convective.
When R\_type\_V[i] = T\_conv,
and R\_type\_H[i] = T\_stra;

251: Convective.
When R\_type\_V[i] = T\_conv,
R\_type\_H[i] = T\_conv,
and shallowRain[i] = 10 or 11;
(Shallow isolated is detected)

252: Convective.
When R\_type\_V[i] = T\_conv,
R\_type\_H[i] = T\_conv,
and shallowRain[i] = 20 or 21;
```

```
(Shallow non-isolated is detected)

261: Convective.
When R\_type\_V[i] = T\_conv,
R\_type\_H[i] = T\_conv;
and shallowRain[i] = 10 or 11;
(Shallow isolated is detected)

262: Convective.
When R\_type\_V[i] = T\_conv,
R\_type\_H[i] = T\_other;
and shallowRain[i] = 20 or 21;
(Shallow non-isolated is detected)

271: Convective.
When R\_type\_V[i] = T\_other,
R\_type\_H[i] = T\_conv;
and shallowRain[i] = 10 or 11;
(Shallow isolated is detected)

272: Convective.
When R\_type\_V[i] = T\_other,
R\_type\_H[i] = T\_conv;
and shallowRain[i] = 20 or 21;
(Shallow non-isolated is detected)

281: Convective.
When R\_type\_V[i] = T\_conv,
R\_type\_H[i] = T\_stra;
and shallowRain[i] = 10 or 11;
(Shallow isolated is detected)

282: Convective.
When R\_type\_V[i] = T\_conv,
R\_type\_H[i] = T\_stra;
and shallowRain[i] = 20 or 21;
(Shallow non-isolated is detected)

291: Convective:
When R\_type\_V[i] = T\_other;
R\_type\_H[i] = T\_stra;
and shallowRain[i] = 10 or 11;
(Shallow isolated is detected)
```

```
292: Convective:
--- added in V7.
When R\_type\_V[i] = T\_other;
R\_type\_H[i] = T\_stra;
and shallowRain[i] = 20 or 21;
Though this is shallow non-isolated, the appearance is
'sporadic', hence convective.

297: Convective:
--- added in V7.
When R\_type\_V[i] = T\_other;
R\_type\_H[i] = T\_stra;
shallowRain[i] = 20 or 21;
(Shallow non-isolated is detected)
But the cell size is small.

300: Other.
When R\_type\_V[i] = T\_other;
and R\_type\_H[i] = T\_other;
This category includes very weak echo (possibly noise)
and/or cloud.

311: Others.
--- added in V7.
When R\_type\_V[i] = T\_other,
R\_type\_H[i] = T\_other;
and shallowRain[i] = 10 or 11;
(Shallow isolated is detected)

312: Other.
When R\_type\_V[i] = T\_other,
R\_type\_H[i] = T\_other;
and shallowRain[i] = 20 or 21;
(Shallow non-isolated is detected)

313: Other.
When R\_type\_V[i] = T\_other,
R\_type\_H[i] = T\_other;
If sidelobe clutter were not rejected.

-88: no rain
```

```
-99: missing
```

The above assignment of numbers has the following meaning:

shallowRain (1-byte integer, array size: nray x nscan):
The Shallow Rain Flag takes the following values:

```
shallowRain(i)  = 10: maybe shallow, isolated,
                = 11: shallow isolated (with confidence),
                = 20: maybe shallow but not isolated,
                = 21: shallow but not isolated (width confidence)
                =  0: when not shallow.
                less than  0: when not rain certain or data missing.
```

status (1-byte integer, array size: nray x nscan):
The Status Flag indicates whether the data are obtained over sea or land and the confidence of 2A-23 product data. It is set as follows:

```
  0: good (over ocean)
 10: BB detection may be good (over ocean)
 20: R-type classification may be good (over ocean)
     (BB detection is good or BB does not exist)
 30: Both BB detection and R-type classification may be
     good (over ocean)
 50: not good (because of warnings)  (over ocean)
100: bad (possible data corruption)  (over ocean)

  1: good (over land)
 11: BB detection may be good (over land)
 21: R-type classification may be good (over land)
     (BB detection is good or BB does not exist)
 31: Both BB detection and R-type classification may be
     good (over land)
 51: not good (because of warnings)  (over land)
101: bad (possible data corruption)  (over land)

  2: good (over coastline)
 12: BB detection may be good (over coastline)
 22: R-type classification may be good (over coastline)
     (BB detection is good or BB does not exist)
 32: Both BB detection and R-type classification may be
     good (over coastline)
 52: not good (because of warnings)  (over coastline)
```

```
102: bad (possible data corruption)  (over coastline)

  4: good (over inland lake)
 14: BB detection may be good (over inland lake)
 24: R-type classification may be good (over inland lake)
     (BB detection is good or BB does not exist)
 34: Both BB detection and R-type classification may be
     good (over inland lake)
 54: not good (because of warnings)  (over inland lake)
104: bad (possible data corruption)  (over inland lake)

  9: may be good (land/sea unknown)
 19: BB detection may be good  (land/sea unknown)
 29: R-type classification may be good (BB detection is
     good or BB does not exist)      (land/sea unknown)
 39: Both BB detection and R-type classification may be
     good (land/sea unknown)
 59: not good (because of warnings)  (land/sea unknown)
109: bad (possible data corruption)  (land/sea unknown)

When it is "no rain" or "data missing",
Status Flag contains the following values:
   -88:    no rain
   -99:    data missing

Assignment of the above numbers are based on the following rules:
   When Status
   Status/100   =   0:  good, may be good, or not good
                    1:  doubtful
```

binBBpeak (2-byte integer, array size: nray x nscan):
A positive range bin number that corresponds to the peak of the bright band. This bin number is in the Level-1 bin numbering scheme (125m, see Level-1 PR description). Values range from 1 to 400. Special values are defined as:

-8888 No rain
-1111 No bright band
-9999 Missing value

HBB (2-byte integer, array size: nray x nscan):
A positive Height of Bright Band is defined in meters above mean sea level. Values are in m. Special values are defined as:

-8888 No rain
-1111 No bright band
-9999 Missing value

BBintensity (4-byte float, array size: nray x nscan):
The maximum value of the bright band obtained from normal samples. Values range from 0.00 to 100.0 dBZ. Special values are defined as:

-8888	No rain
-1111	No bright band
-9999	Missing value

freezH (2-byte integer, array size: nray x nscan):
A positive Height of Freezing Level is the height of the 0^{o}C isotherm above mean sea level, estimated from GANAL (Global analysis data by Japanese Meteorological Agency) surface temperature data. Values are in m. Special values are defined as:

-8888	No rain
-5555	When error occurred in the estimation of Height of Freezing Level
-9999	Missing value

stormH (2-byte integer, array size: nray x nscan):
A positive Height of Storm is the height of the storm top above mean sea level. A positive Height of Storm is given only when rain is present with a high degree of confidence in 1C21 (i.e., the Minimum Echo Flag in 1C21 has the value of 2 (rain certain)). Values range from 0 to 30000 m. Special values are defined as:

-8888	No rain
-1111	Rain is not present with a high level of confidence in 1C21
-9999	Missing value

spare (2-byte integer, array size: nray x nscan):
Contains developer output.

BBboundary (2-byte integer, array size: 2 x nray x nscan):
Positive bin numbers of the boundary of the bright band. The first index indicates the top of the bright band, the second index indicates the bottom. These bin numbers are in the Level-1 bin numbering scheme (125m, see Level-1 PR description). Values range from 0.00 to 100.0. Special values are defined as:

-8888	No rain
-1111	No bright band
-9999	Missing value

BBwidth (2-byte integer, array size: nray x nscan):
Width of the bright band. Values are in m. Special values are defined as:

-8888	No rain
-1111	No bright band
-9999	Missing value

BBstatus (1-byte integer, array size: nray x nscan):
Indicates the status of the bright band detection. This flag is a composite of three internal status flags:

```
BB\_status(j) = BB\_detection\_status(j) * 16
              + BB\_boundary\_status(j) * 4
```

```
                 + BB\_width\_status(j)

    where each status on the right hand side takes the following
    values:
          1: poor,
          2: fair,
          3: good.


    These three internal flags would be computed from BB\_status(j),
    for example, by something like as follows:

    if (BB\_status(j)>0)
    {
        BB\_detection\_status(j) = BB\_status(j) / 16;
        BB\_boundary\_status(j) = (BB\_status(j)%16) / 4;
        BB\_width\_status(j) = BB\_status(j)%4;
    }

    where % means MOD in FORTRAN;
```

C Structure Header file:

```
#ifndef _TK_2A23_H_
#define _TK_2A23_H_

#ifndef _L2A23_NAVIGATION_
#define _L2A23_NAVIGATION_

typedef struct {
    float scPosX;
    float scPosY;
    float scPosZ;
    float scVelX;
    float scVelY;
    float scVelZ;
    float scLat;
    float scLon;
    float scAlt;
    float scAttRoll;
    float scAttPitch;
    float scAttYaw;
    float SensorOrientationMatrix[3][3];
```

```
    float greenHourAng;
} L2A23_NAVIGATION;

#endif

#ifndef _L2A23_SCANSTATUS_
#define _L2A23_SCANSTATUS_

typedef struct {
    signed char missing;
    signed char validity;
    signed char qac;
    signed char geoQuality;
    signed char dataQuality;
    short SCorientation;
    signed char acsMode;
    signed char yawUpdateS;
    signed char prMode;
    signed char prStatus1;
    signed char prStatus2;
    double FractionalGranuleNumber;
} L2A23_SCANSTATUS;

#endif

#ifndef _L2A23_SCANTIME_
#define _L2A23_SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
} L2A23_SCANTIME;

#endif

#ifndef _L2A23_SWATH_
#define _L2A23_SWATH_
```

```
typedef struct {
    L2A23_SCANTIME ScanTime;
    double scanTime_sec;
    float Latitude[49];
    float Longitude[49];
    L2A23_SCANSTATUS scanStatus;
    L2A23_NAVIGATION navigation;
    signed char rainFlag[49];
    short rainType[49];
    signed char shallowRain[49];
    signed char status[49];
    short binBBpeak[49];
    short HBB[49];
    float BBintensity[49];
    short freezH[49];
    short stormH[49];
    short spare[49];
    short BBboundary[49][2];
    short BBwidth[49];
    signed char BBstatus[49];
} L2A23_SWATH;

#endif

#endif
```

Fortran Structure Header file:

```
STRUCTURE /L2A23_NAVIGATION/
    REAL*4 scPosX
    REAL*4 scPosY
    REAL*4 scPosZ
    REAL*4 scVelX
    REAL*4 scVelY
    REAL*4 scVelZ
    REAL*4 scLat
    REAL*4 scLon
    REAL*4 scAlt
    REAL*4 scAttRoll
    REAL*4 scAttPitch
    REAL*4 scAttYaw
    REAL*4 SensorOrientationMatrix(3,3)
```

```
    REAL*4 greenHourAng
END STRUCTURE

STRUCTURE /L2A23_SCANSTATUS/
    BYTE missing
    BYTE validity
    BYTE qac
    BYTE geoQuality
    BYTE dataQuality
    INTEGER*2 SCorientation
    BYTE acsMode
    BYTE yawUpdateS
    BYTE prMode
    BYTE prStatus1
    BYTE prStatus2
    REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /L2A23_SCANTIME/
    INTEGER*2 Year
    BYTE Month
    BYTE DayOfMonth
    BYTE Hour
    BYTE Minute
    BYTE Second
    INTEGER*2 MilliSecond
    INTEGER*2 DayOfYear
END STRUCTURE

STRUCTURE /L2A23_SWATH/
    RECORD /L2A23_SCANTIME/ ScanTime
    REAL*8 scanTime_sec
    REAL*4 Latitude(49)
    REAL*4 Longitude(49)
    RECORD /L2A23_SCANSTATUS/ scanStatus
    RECORD /L2A23_NAVIGATION/ navigation
    BYTE rainFlag(49)
    INTEGER*2 rainType(49)
    BYTE shallowRain(49)
    BYTE status(49)
    INTEGER*2 binBBpeak(49)
    INTEGER*2 HBB(49)
    REAL*4 BBintensity(49)
```

```
    INTEGER*2 freezH(49)
    INTEGER*2 stormH(49)
    INTEGER*2 spare(49)
    INTEGER*2 BBboundary(2,49)
    INTEGER*2 BBwidth(49)
    BYTE BBstatus(49)
END STRUCTURE
```

7.8 2A25 - PR Profile

2A25, "PR Profile", produces an estimate of vertical rainfall rate profile for each radar beam. The rainfall rate estimate is given at each resolution cell of the PR radar. To compare with ground-based radar data, the attenuation corrected Z profile is also given. The average rainfall rate between the two pre-defined altitudes is calculated for each beam position. Other output data include parameters of Z-R relationships, integrated rain rate of each beam, range bin numbers of rain layer boundaries, and many intermediate parameters. The following sections describe the structure and contents of the format.

Dimension definitions:

nscan	var	Number of scans in the granule.
nray	49	Number of angle bins in each scan.
ncell1	80	Number of radar range cells at which the rain rate is estimated. The cells range from 0 to 79. Each cell is 250m apart, with cell 79 at the earth ellipsiod.
ncell2	5	Number of radar range cells at which the Z-R parameters are output.
nmeth	2	Number of methods used.
nestmeth	6	Number of estimation methods.

Figure 45 through Figure 51 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for TRMM Products for details.

InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for TRMM Products for details.

NavigationRecord (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for TRMM Products for details.

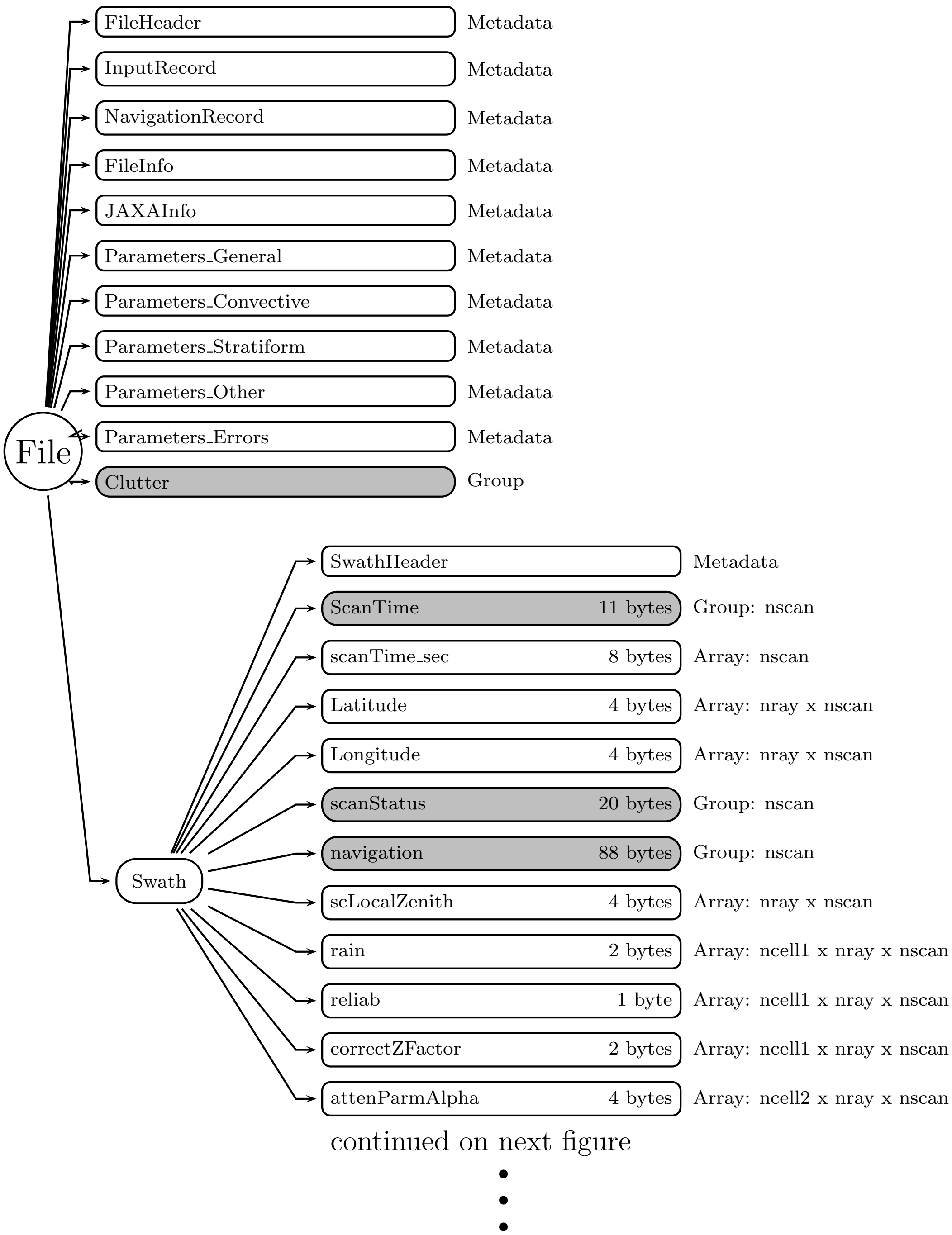


Figure 45: Data Format Structure for 2A25, PR Profile

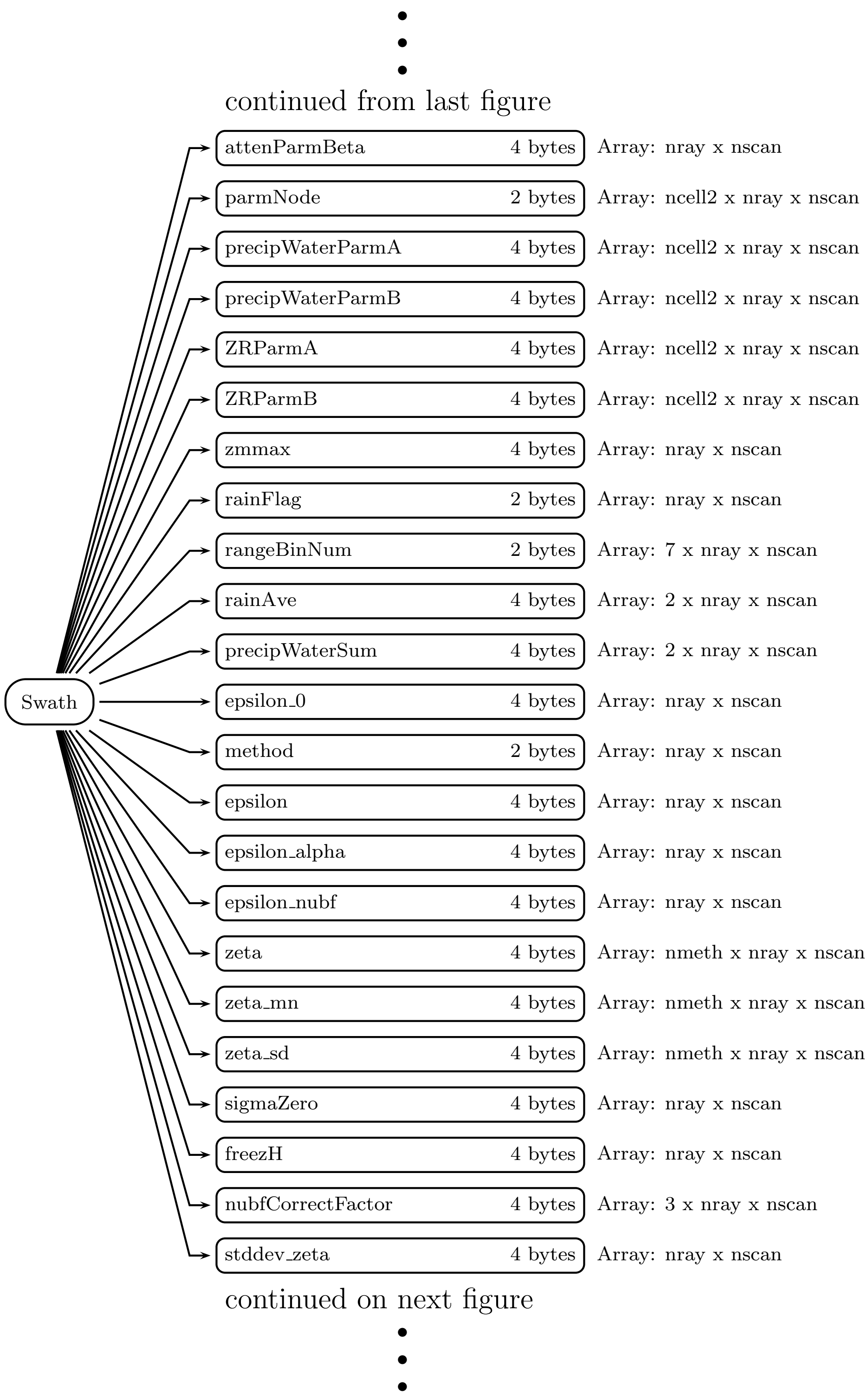


Figure 46: Data Format Structure for 2A25, PR Profile

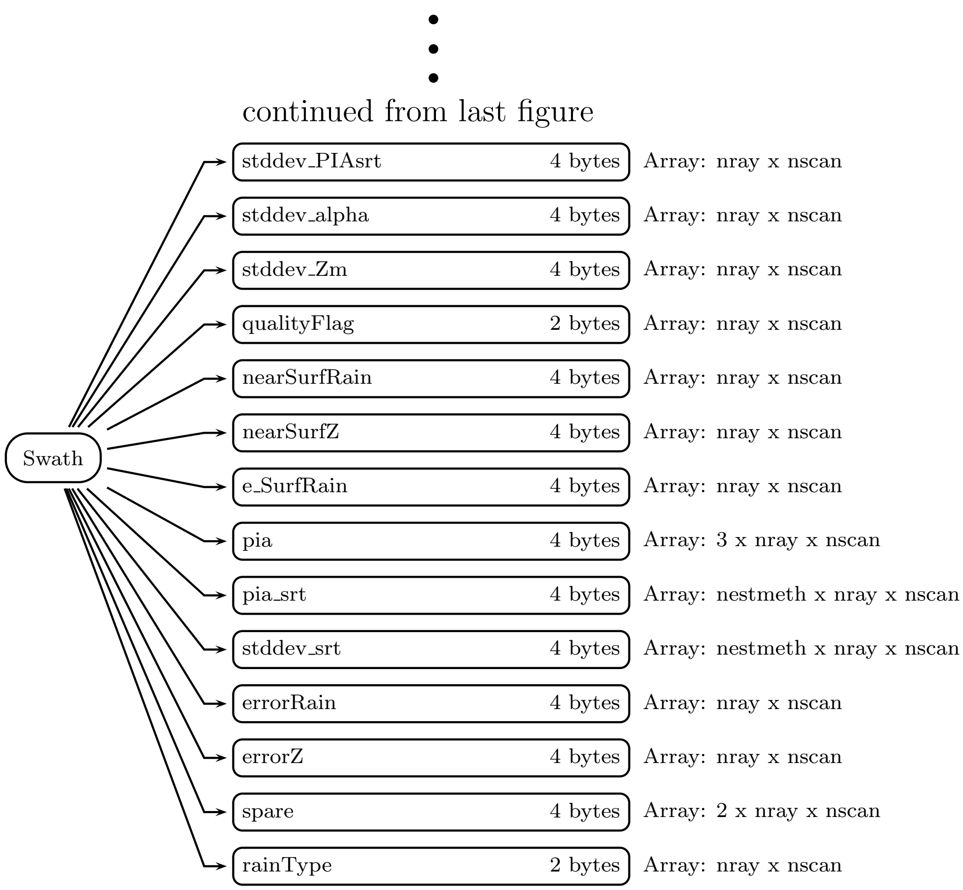


Figure 47: Data Format Structure for 2A25, PR Profile

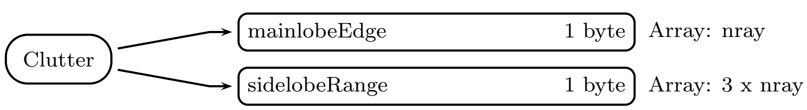


Figure 48: Data Format Structure for 2A25, Clutter

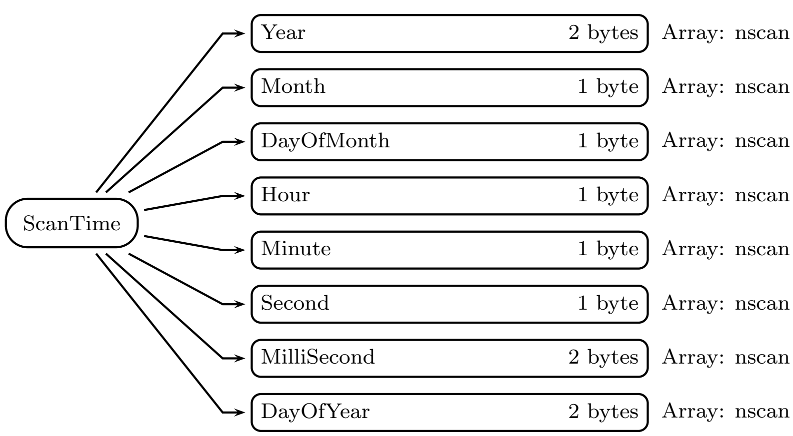


Figure 49: Data Format Structure for 2A25, ScanTime

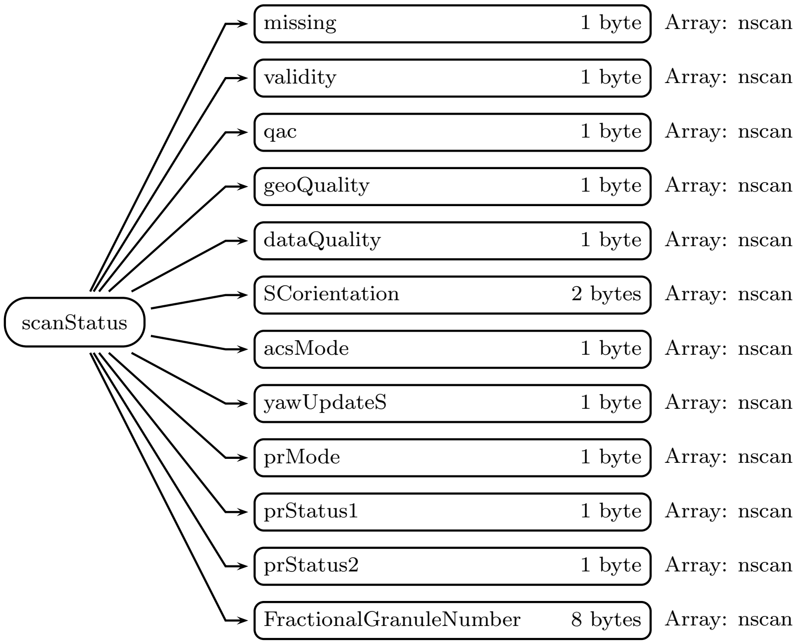


Figure 50: Data Format Structure for 2A25, scanStatus

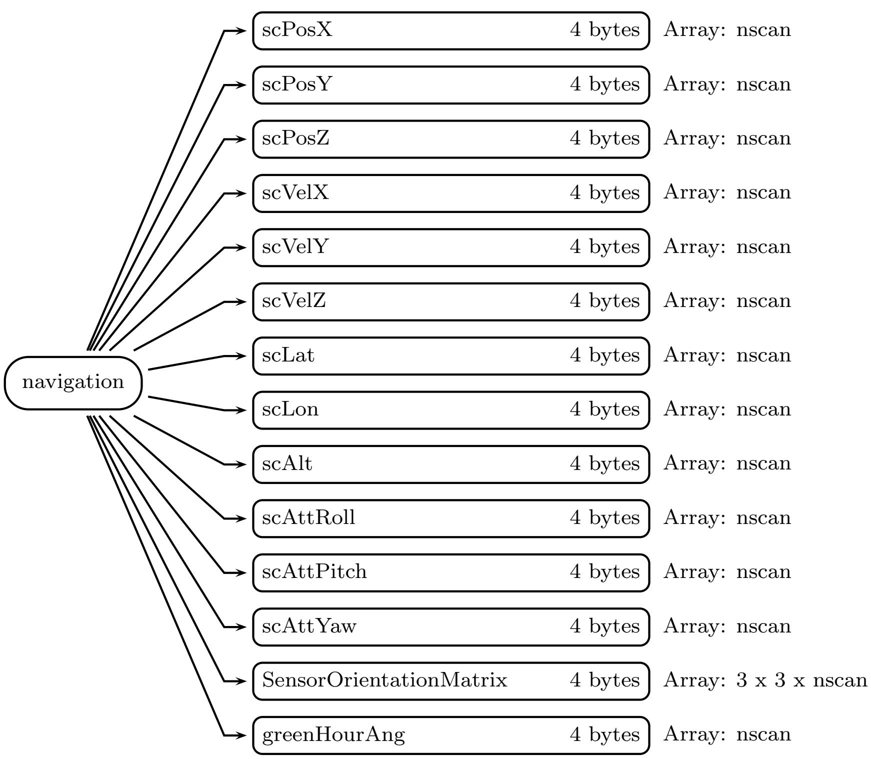


Figure 51: Data Format Structure for 2A25, navigation

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for TRMM Products for details.

JAXAInfo (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by PR algorithms only. See Metadata for TRMM Products for details.

Parameters_General (Metadata):
ASCII text of the general parameters used by 2A25 at runtime.

Parameters_Convective (Metadata):
ASCII text of the parameters for Convective rain used by 2A25 at runtime.

Parameters_Stratiform (Metadata):
ASCII text of the parameters for Stratiform rain used by 2A25 at runtime.

Parameters_Other (Metadata):
ASCII text of the parameters for Other rain used by 2A25 at runtime.

Parameters_Errors (Metadata):
ASCII text of the Error parameters used by 2A25 at runtime.

Clutter (Group)

mainlobeEdge (1-byte integer, array size: nray):
Absolute value of the difference in Range Bin Numbers between the detected surface and the edge of the clutter from the mainlobe.

sidelobeRange (1-byte integer, array size: 3 x nray):
Absolute value of the difference in Range Bin Numbers between the detected surface and the clutter position from the sidelobe. A zero means no clutter indicated in this field since less than 3 bins contained significant clutter.

Swath (Swath)

SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for TRMM Products for details.

ScanTime (Group)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

scanTime_sec (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: nray x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: nray x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scanStatus (Group)

missing (1-byte integer, array size: nscan):
Missing indicates whether information is contained in the scan data. The values are:

```
0 Scan data elements contain information
1 Scan was missing in the telemetry data
2 Scan data contains no elements with rain
```

validity (1-byte integer, array size: nscan):
Validity is a summary of status modes. If all status modes are routine, all bits in Validity = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. Validity does not assess data or geolocation quality. Validity is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

```
Bit Meaning if bit = 1
0   Spare (always 0)
1   Non-routine spacecraft orientation (2 or 3)
2   Non-routine ACS mode (other than 4)
3   Non-routine yaw update status (0 or 1)
4   Non-routine instrument status (other than 1)
5   Non-routine QAC (non-zero)
6   Spare (always 0)
7   Spare (always 0)
```

qac (1-byte integer, array size: nscan):
The Quality and Accounting Capsule of the Science packet as it appears in Level-0 data. If no QAC is given in Level-0, which means no decoding errors occurred, QAC in this format has a value of zero.

geoQuality (1-byte integer, array size: nscan):
Geolocation quality is a summary of geolocation quality in the scan. A zero integer value indicates 'good' geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

```
Bit Meaning if bit = 1
0   latitude limit error
1   geolocation
2   attitude change rate limit error
```

```
3   attitude limit error
4   satellite undergoing maneuvers
5   using predictive orbit data
6   geolocation calculation error
7   not used
```

dataQuality (1-byte integer, array size: nscan):
Data quality is a summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

```
Bit Meaning if bit = 1
0   missing
5   Geolocation Quality is not normal
6   Validity is not normal
```

SCorientation (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the TMI scan. If +X is forward, SCorientation is 0. If -X is forward, SCorientation is 180. If -Y is forward, SCorientation is 90. Values range from 0 to 360 degrees. Special values are defined as:

-8003	Inertial
-8004	Unknown
-9999	Missing value

acsMode (1-byte integer, array size: nscan):

```
Value Meaning
0     Standby
1     Sun Acquire
2     Earth Acquire
3     Yaw Acquire
4     Nominal
5     Yaw Maneuver
6     Delta-H (Thruster)
7     Delta-V (Thruster)
8     CERES Calibration
```

yawUpdateS (1-byte integer, array size: nscan):

```
Value Meaning
0     Inaccurate
```

```
1     Indeterminate
2     Accurate
```

prMode (1-byte integer, array size: nscan):

```
Value Meaning
1     Observation Mode
2     Other Mode
```

prStatus1 (1-byte integer, array size: nscan):
This status is a warning for scan data. Unless this is 0, the scan data may include a little questionable value though it is not a problem (such as break of caution limit). This field is used only for NASDA's data analysis.

prStatus2 (1-byte integer, array size: nscan):
Initialization in Onboard Surface Search Algorithm.

```
Value Meaning
0     Not initialized
1     Initialized
```

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft's trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

navigation (Group)

scPosX (4-byte float, array size: nscan):
The x component of the position (m) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Geocentric Inertial Coordinates are also commonly known as Earth Centered Inertial coordinates. These coordinates will be True of Date (rather than Epoch 2000 which are also commonly used), as interpolated from the data in the Flight Dynamics Facility ephemeris files generated for TRMM.

scPosY (4-byte float, array size: nscan):
The y component of the position (m) of the spacecraft in Geocentric Inertial Coordinates. See scPosX.

scPosZ (4-byte float, array size: nscan):
The z component of the position (m) of the spacecraft in Geocentric Inertial Coordinates. See scPosX.

scVelX (4-byte float, array size: nscan):
The x component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scVelY (4-byte float, array size: nscan):
The y component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scVelZ (4-byte float, array size: nscan):
The z component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scLat (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time.

scLon (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time.

scAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time.

scAttRoll (4-byte float, array size: nscan):
The satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

scAttPitch (4-byte float, array size: nscan):
The satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system

following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

scAttYaw (4-byte float, array size: nscan):
The satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

SensorOrientationMatrix (4-byte float, array size: 3 x 3 x nscan):
SensorOrientationMatrix is the rotation matrix from the instrument coordinate frame to Geocentric Inertial Coordinates at the Scan mid-Time. It is unitless.

greenHourAng (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates.

scLocalZenith (4-byte float, array size: nray x nscan):
The angle, in degrees, between the local zenith and the PR ray center line. The local (geodetic) zenith at the intersection of the ray and the earth ellipsoid is used. Ranges from - 30.0 to +30.0. This is an exact copy of the satellite local zenith angle in the 1C21 product.

rain (2-byte integer, array size: ncell1 x nray x nscan):
This is the estimate of rain rate at the radar range gates from 0 to 20 km along the slant range. It ranges from 0.0 to 300.0 mm/hr and is multiplied by 100 and stored as a 2-byte integer. A value of -88.88 mm/hr (stored as -8888) means ground clutter.

reliab (1-byte integer, array size: ncell1 x nray x nscan):
The Reliability is that for estimated rain rates at the radar range gates from 0 to 20 km. It ranges from 0 to 255. If data are missing, the reliability will be set as 10000000 in binary. The default value is 0 (measured signal below noise). Bit 0 is the least significant bit (i.e., if bit i =1 and other bits =0, the unsigned integer value is 2**i). The following meanings are assigned to each bit in the 8-bit integer if the bit = 1.

```
bit 0   rain possible
bit 1   rain certain
bit 2   bright band
bit 3   large attenuation
```

```
bit 4   weak return (Zm less than 20 dBZ)
bit 5   estimated Z less than 0 dBZ
bit 6   main-lobe clutter or below surface
bit 7   missing data
```

correctZFactor (2-byte integer, array size: ncell1 x nray x nscan):
This is the attenuation corrected reflectivity factor (Z) at the radar range gates from 0 to 20 km along the slant range. It ranges from 0.0 to 80.0 dBZ and is multiplied by 100 and stored as a 2-byte integer. Values of reflectivity less than 0.0 dBZ are set to 0.0 dBZ. A value of -88.88 dB (stored as -8888) is a ground clutter flag.

attenParmAlpha (4-byte float, array size: ncell2 x nray x nscan):
The attenuation parameter alpha relates the attenuation coefficient, k (dB/km) to the Z-factor: k = $alpha * Z^{beta}$. Alpha is computed at ncell2 radar range gates for each ray. It ranges from 0.000100 to 0.002000.

attenParmBeta (4-byte float, array size: nray x nscan):
The attenuation parameter Beta in k(dB/km) = $alpha * Z^{beta}$. Computed at ncell2 radar range gates for each ray.

parmNode (2-byte integer, array size: ncell2 x nray x nscan):
The Parameter Node gives the range bin numbers of the nodes at which the values of Attenuation and Z-R Parameters are given (see below). The values of the parameters between the nodes are linearly interpolated. This variable ranges from 0 and 79 and is unitless.

precipWaterParmA (4-byte float, array size: ncell2 x nray x nscan):
Parameter A in the M = AZ^B relationship. A is computed at each node (ncell2) for each ray. It ranges from ? to ?

precipWaterParmB (4-byte float, array size: ncell2 x nray x nscan):
Parameter B in the M = AZ^B relationship. B is computed at each node (ncell2) for each ray. It ranges from ? to ?

ZRParmA (4-byte float, array size: ncell2 x nray x nscan):
Parameter a for Z-R relationship (R=aZb) is determined from the rain type and the height relative to the freezing level, the non-uniformity parameter (?) and the correction factor (?) for the surface reference technique. a is computed at ncell2 radar range gates for each ray. It ranges from 0.0050 to 0.2000.

ZRParmB (4-byte float, array size: ncell2 x nray x nscan):
Parameter b for Z-R relationship (R=aZb) is determined from the rain type and the height relative to the freezing level, the non-uniformity parameter (?) and the correction factor (?) for the surface reference technique. b is computed at ncell2 radar range gates for each ray. It ranges from 0.500 to 1.000.

zmmax (4-byte float, array size: nray x nscan):
This is the maximum value of measured reflectivity factor at each IFOV. It ranges from 0.0 to 100.0 dBZ.

rainFlag (2-byte integer, array size: nray x nscan):
The Rain Flag indicates rain or no rain status and the rain type assumed in rain rate retrieval. The default value is 0 (no rain). Bit 0 is the least significant bit (i.e., if bit i=1 and other bits =0, the unsigned integer value is 2**i). The following meanings are assigned to each bit in the 16-bit integer if the bit = 1.

```
bit 0   rain possible
bit 1   rain certain
bit 2   zeta**beta greater than 0.5
        (Path Integrated Attenuation (PIA) larger than 3 dB)
bit 3   large attenuation (PIA larger than 10 dB)
bit 4   stratiform
bit 5   convective
bit 6   bright band exists
bit 7   warm rain
bit 8   rain bottom above 2 km
bit 9   rain bottom above 4 km
bit 10  not used
bit 11  not used
bit 12  not used
bit 13  not used
bit 14  data missing between rain top and bottom
bit 15  not used
```

rangeBinNum (2-byte integer, array size: 7 x nray x nscan):
This array gives the Range Bin Number of various quantities for each ray in every scan. The definitions are:

```
- top range bin number of the interval that is processed as meaningful
  data in 2A-25
- bottom range bin number of the interval that is processed as meaningful
  data in 2A-25
- actual surface range bin number (can be larger than 79)
- range bin number of the bright band if it exits
- range bin number at which the path-integrated Z-factor first exceeds
  the given threshold
- range bin number at which the measured Z-factor is maximum
- range bin number of near surface bin
```

The Range Bin Numbers in this algorithm are different from the NASDA definition of Range Bin Number described in the ICS, Volume 3. The Range Bin Numbers in the algorithm range from 0 to 82 and have an interval of 250m. The earth ellipsoid is defined as range bin 79.

rainAve (4-byte float, array size: 2 x nray x nscan):
There are two kinds of Average Rain Rate. The first one is the average rain rate for each ray between the two predefined heights of 2 and 4 km. It ranges from 0.0 to 3000.0 mm h-1 and is multiplied by 10 and stored as a 2-byte integer. The second one is the integral of rain rate from rain top to rain bottom. It ranges from 0.0 to 300 mm h-1 km.

precipWaterSum (4-byte float, array size: 2 x nray x nscan):
Vertically integrated value of sum precipitation water content calculated from Ze at each range bin. The first index is the precipitation liquid water content from the freezing height to the actual surface. The second index is the sum of precipitation ice content from the top of the storm to the freezing height. Units are $gkm/m^3(kg/m^2)$ and it ranges from 0.0 to 50.0.

epsilon_0 (4-byte float, array size: nray x nscan):
The adjustment parameter computed from the filtered surface reference PIA (2A21). Unitless and it ranges from 0.0 to 100.0.

method (2-byte integer, array size: nray x nscan):
This flag indicates which method is used to derive the rain rate. The default value is 0 (including no rain case). Bit 0 is the least significant bit (i.e., if bit i =1 and other bits =0, the unsigned integer value is 2**i). The default value is 0 (including no rain case). The following meanings are assigned to each bit in the 16-bit integer.

```
         0: (bit  1)    no rain
    if rain
         0: (bit  1)    over ocean
         1: (bit  1)    over land
         2: (bit  2)    over coast, river, etc.
         3: (bit  2)    others (impossible)
        +4: (bit  3)    PIA from constant-Z-near-surface assumption
        +8: (bit  4)    spatial reference
       +16: (bit  5)    temporal reference
       +32: (bit  6)    global reference
       +64: (bit  7)    hybrid reference
      +128: (bit  8)    good to take statistics of epsilon.
      +256: (bit  9)    HB method used, SRT totally ignored
      +512: (bit 10)    very large pia\_srt for given zeta
     +1024: (bit 11)    very small pia\_srt for given zeta
     +2048: (bit 12)    no ZR adjustment by epsilon
     +4096: (bit 13)    no NUBF correction because NSD unreliable
     +8192: (bit 14)    surface attenuation > 60 dB
    +16384: (bit 15)    data partly missing between rain top and bottom
```

epsilon (4-byte float, array size: nray x nscan):
The Epsilon (?) is the final correction factor applied to the assumed drop size distribution. Unitless and it ranges from 0.0 to 100.0.

epsilon_alpha (4-byte float, array size: nray x nscan):
Value used as a multiplicative correction to alpha used to modify R-Ze and LWC-Ze relations. Unitless and ranges from 0.0 to 100.0.

epsilon_nubf (4-byte float, array size: nray x nscan):
NUBF correction factor for k-Ze relation. Unitless and it ranges from 0.8 to 1.0.

zeta (4-byte float, array size: nmeth x nray x nscan):
Integral of $0.2 * ln(10) * beta * alpha * Zm^{beta}$ from rain top to the clutter-free bottom. First index is from the rain top to the bottom. Second index is PIA_est from epsilon corrected zeta. It ranges from 0.0 to 100.0 and is unitless.

zeta_mn (4-byte float, array size: nmeth x nray x nscan):
Zeta_mn is the average of zeta in the vicinity of each beam position (average over three scans and three IFOVs). First index is mean of zeta. Second index is mean of PIA_est. It ranges from 0.0 to 100.0 and is unitless.

zeta_sd (4-byte float, array size: nmeth x nray x nscan):
Zeta_sd is the standard deviation of zeta in the vicinity of each beam position (using three scans and three IFOVs). First index is the standard deviation of zeta. Second index is the standard deviation of PIA_est. It ranges from 0.0 to 100.0 and is unitless.

sigmaZero (4-byte float, array size: nray x nscan):
The Sigma-zero is the normalized surface cross section. It ranges from -50.00 to 20.00 dB. This field is copied from the 2A21 product file.

freezH (4-byte float, array size: nray x nscan):
A positive Height of Freezing Level is the height of the 0^{o}C isotherm above mean sea level in meters, estimated from GANAL (Global analysis data by Japanese Meteorological Agency) surface temperature data. This field is copied from the 2A23 product. Negative values are defined as:

```
-5555: When error occurred in the estimation of Height of Freezing Level
-8888: No rain
-9999: Data missing
```

nubfCorrectFactor (4-byte float, array size: 3 x nray x nscan):

```
The Non-Uniform Beam Filling (NUBF) Correction Factor
is used as a correction to reflectivity and attenuation calculations.
It's range is between 1.0 and 10.0 and is unitless.
The dimension of 3 has the following meanings (in C):
  0    NUBF correction factor for the surface  reference
  1    NUBF correction factor for the R-Ze relation
  2    NUBF correction factor for the LWC-Ze relation
Note that the NUBF correction factor for the k-Ze relation
is stored in epsilon nubf.
```

stddev_zeta (4-byte float, array size: nray x nscan):
Standard deviation of zeta. Unitless.

stddev_PIAsrt (4-byte float, array size: nray x nscan):
Standard deviation of PIAsrt given by surface reference technique. Units are dB.

stddev_alpha (4-byte float, array size: nray x nscan):
Standard deviation of epsilon total corresponds to estimated uncertainty of alpha. Unitless.

stddev_Zm (4-byte float, array size: nray x nscan):
Standard deviation of epsilon_f. Unitless.

qualityFlag (2-byte integer, array size: nray x nscan):
This is a quality flag and ranges from 0 to 32767. The default value is 0 (normal). Bit 0 is the least significant bit (i.e., if bit i =1 and other bits =0, the unsigned integer value is 2**i). The following meanings are assigned to each bit in the 16-bit integer if the bit = 1.

```
     0:   normal
    +1:   unusual situation in rain average
    +2:   NSD of zeta (xi) calculated from less than 6 points
    +4:   NSD of PIA calculated from less than 6 points
    +8:   NUBF for Z-R below lower bound
   +16:   NUBF for PIA above upper bound
   +32:   epsilon not reliable, epsi\_sig less than or equal to 0.0
   +64:   2A21 input data not reliable
  +128:   2A23 input data not reliable
  +256:   range bin error
  +512:   sidelobe clutter removal
 +1024:   probability=0 for all tau
 +2048:   pia\_surf\_ex less than or equal to 0.0
 +4096:   const Z is invalid
 +8192:   reliabFactor in 2A21 is NaN
+16384:   data missing
```

nearSurfRain (4-byte float, array size: nray x nscan):
Rainfall rate near the surface. The range is 0 to 300 mm/hr. A value of -99.99 mm/hr. is a missing flag.

nearSurfZ (4-byte float, array size: nray x nscan):
Reflectivity near the surface. The range is 0.0 to 100.0 dBZ. A value of -99.99 dBZ is a missing flag.

e_SurfRain (4-byte float, array size: nray x nscan):
The rainfall estimate at the true (detected) surface bin. Units are mm/hr and ranges from 0 to 300 mm/hr. A value of -99.99 mm/hr is a missing flag.

pia (4-byte float, array size: 3 x nray x nscan):
Path integrated attenuation (PIA) (two-way) estimates for three cases: - The final adjusted PIA estimate - The difference between the PIA at the surface and near surface range bins - The PIA estimate from 2A21 The units are dB. Values are in dB. Special values are defined as:
-9999.9 Missing value

pia_srt (4-byte float, array size: nestmeth x nray x nscan):
Path integrated attenuation estimates from 2A21. First index is the method of estimation:

```
0 - best estimate (copy of 2A21 pathAtten)
1 - spatial, forward
2 - hybrid, forward (ocean only)
3 - spatial, backward
4 - hybrid, backward (ocean only)
5 - temporal (should be the same forward or backward)
```

Values range from -50 to 50 dB. Special values are defined as:
-9999.9 Missing value

stddev_srt (4-byte float, array size: nestmeth x nray x nscan):
Standard deviation of the SRT PIA estimates computed from the SRT PIA and the 2A21 reliability factor. First index is the method of estimation:

```
0 - best estimate (copy of 2A21 pathAtten)
1 - spatial, forward
2 - hybrid, forward (ocean only)
3 - spatial, backward
4 - hybrid, backward (ocean only)
5 - temporal (should be the same forward or backward)
```

Values range from -50 to 50 dB. Special values are defined as:
-9999.9 Missing value

errorRain (4-byte float, array size: nray x nscan):
The error in Near Surface Rain Rate. The units are dB.

errorZ (4-byte float, array size: nray x nscan):
The error in Near Surface Z. The range is 0.0 to 100.0 dBZ.

spare (4-byte float, array size: 2 x nray x nscan):
Contents and ranges are not public.

rainType (2-byte integer, array size: nray x nscan):
This is a copy of the 2A23 Rain Type field, See 2A23 description.

C Structure Header file:

```
#ifndef _TK_2A25_H_
#define _TK_2A25_H_

#ifndef _L2A25_NAVIGATION_
#define _L2A25_NAVIGATION_

typedef struct {
    float scPosX;
    float scPosY;
    float scPosZ;
    float scVelX;
    float scVelY;
    float scVelZ;
    float scLat;
    float scLon;
    float scAlt;
    float scAttRoll;
    float scAttPitch;
    float scAttYaw;
    float SensorOrientationMatrix[3][3];
    float greenHourAng;
} L2A25_NAVIGATION;

#endif

#ifndef _L2A25_SCANSTATUS_
#define _L2A25_SCANSTATUS_

typedef struct {
    signed char missing;
    signed char validity;
    signed char qac;
    signed char geoQuality;
    signed char dataQuality;
    short SCorientation;
    signed char acsMode;
    signed char yawUpdateS;
    signed char prMode;
    signed char prStatus1;
    signed char prStatus2;
    double FractionalGranuleNumber;
} L2A25_SCANSTATUS;
```

```
#endif

#ifndef _L2A25_SCANTIME_
#define _L2A25_SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
} L2A25_SCANTIME;

#endif

#ifndef _L2A25_SWATH_
#define _L2A25_SWATH_

typedef struct {
    L2A25_SCANTIME ScanTime;
    double scanTime_sec;
    float Latitude[49];
    float Longitude[49];
    L2A25_SCANSTATUS scanStatus;
    L2A25_NAVIGATION navigation;
    float scLocalZenith[49];
    float rain[49][80];
    signed char reliab[49][80];
    float correctZFactor[49][80];
    float attenParmAlpha[49][5];
    float attenParmBeta[49];
    short parmNode[49][5];
    float precipWaterParmA[49][5];
    float precipWaterParmB[49][5];
    float ZRParmA[49][5];
    float ZRParmB[49][5];
    float zmmax[49];
    short rainFlag[49];
    short rangeBinNum[49][7];
```

```
    float rainAve[49][2];
    float precipWaterSum[49][2];
    float epsilon_0[49];
    short method[49];
    float epsilon[49];
    float epsilon_alpha[49];
    float epsilon_nubf[49];
    float zeta[49][2];
    float zeta_mn[49][2];
    float zeta_sd[49][2];
    float sigmaZero[49];
    float freezH[49];
    float nubfCorrectFactor[49][3];
    float stddev_zeta[49];
    float stddev_PIAsrt[49];
    float stddev_alpha[49];
    float stddev_Zm[49];
    short qualityFlag[49];
    float nearSurfRain[49];
    float nearSurfZ[49];
    float e_SurfRain[49];
    float pia[49][3];
    float pia_srt[49][6];
    float stddev_srt[49][6];
    float errorRain[49];
    float errorZ[49];
    float spare[49][2];
    short rainType[49];
} L2A25_SWATH;

#endif

#ifndef _L2A25_CLUTTER_
#define _L2A25_CLUTTER_

typedef struct {
    signed char mainlobeEdge[49];
    signed char sidelobeRange[49][3];
} L2A25_CLUTTER;

#endif

#endif
```

Fortran Structure Header file:

```
STRUCTURE /L2A25_NAVIGATION/
    REAL*4 scPosX
    REAL*4 scPosY
    REAL*4 scPosZ
    REAL*4 scVelX
    REAL*4 scVelY
    REAL*4 scVelZ
    REAL*4 scLat
    REAL*4 scLon
    REAL*4 scAlt
    REAL*4 scAttRoll
    REAL*4 scAttPitch
    REAL*4 scAttYaw
    REAL*4 SensorOrientationMatrix(3,3)
    REAL*4 greenHourAng
END STRUCTURE

STRUCTURE /L2A25_SCANSTATUS/
    BYTE missing
    BYTE validity
    BYTE qac
    BYTE geoQuality
    BYTE dataQuality
    INTEGER*2 SCorientation
    BYTE acsMode
    BYTE yawUpdateS
    BYTE prMode
    BYTE prStatus1
    BYTE prStatus2
    REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /L2A25_SCANTIME/
    INTEGER*2 Year
    BYTE Month
    BYTE DayOfMonth
    BYTE Hour
    BYTE Minute
    BYTE Second
    INTEGER*2 MilliSecond
    INTEGER*2 DayOfYear
```

```
END STRUCTURE

STRUCTURE /L2A25_SWATH/
    RECORD /L2A25_SCANTIME/ ScanTime
    REAL*8 scanTime_sec
    REAL*4 Latitude(49)
    REAL*4 Longitude(49)
    RECORD /L2A25_SCANSTATUS/ scanStatus
    RECORD /L2A25_NAVIGATION/ navigation
    REAL*4 scLocalZenith(49)
    REAL*4 rain(80,49)
    BYTE reliab(80,49)
    REAL*4 correctZFactor(80,49)
    REAL*4 attenParmAlpha(5,49)
    REAL*4 attenParmBeta(49)
    INTEGER*2 parmNode(5,49)
    REAL*4 precipWaterParmA(5,49)
    REAL*4 precipWaterParmB(5,49)
    REAL*4 ZRParmA(5,49)
    REAL*4 ZRParmB(5,49)
    REAL*4 zmmax(49)
    INTEGER*2 rainFlag(49)
    INTEGER*2 rangeBinNum(7,49)
    REAL*4 rainAve(2,49)
    REAL*4 precipWaterSum(2,49)
    REAL*4 epsilon_0(49)
    INTEGER*2 method(49)
    REAL*4 epsilon(49)
    REAL*4 epsilon_alpha(49)
    REAL*4 epsilon_nubf(49)
    REAL*4 zeta(2,49)
    REAL*4 zeta_mn(2,49)
    REAL*4 zeta_sd(2,49)
    REAL*4 sigmaZero(49)
    REAL*4 freezH(49)
    REAL*4 nubfCorrectFactor(3,49)
    REAL*4 stddev_zeta(49)
    REAL*4 stddev_PIAsrt(49)
    REAL*4 stddev_alpha(49)
    REAL*4 stddev_Zm(49)
    INTEGER*2 qualityFlag(49)
    REAL*4 nearSurfRain(49)
    REAL*4 nearSurfZ(49)
```

```
    REAL*4 e_SurfRain(49)
    REAL*4 pia(3,49)
    REAL*4 pia_srt(6,49)
    REAL*4 stddev_srt(6,49)
    REAL*4 errorRain(49)
    REAL*4 errorZ(49)
    REAL*4 spare(2,49)
    INTEGER*2 rainType(49)
END STRUCTURE

STRUCTURE /L2A25_CLUTTER/
    BYTE mainlobeEdge(49)
    BYTE sidelobeRange(3,49)
END STRUCTURE
```

7.9 2B31 - TRMM Combined

2B31, "TRMM Combined", derives vertical hydrometeor profiles using data from PR radar and TMI. It also computes the correlation-corrected mass-weighted mean drop diameter and its standard deviation, and latent heating. The following sections describe the structure and contents of the format.

Dimension definitions:

nscan	var	Number of scans in the granule.
nray	49	Number of angle bins in each scan.
Nradarrange	80	Number of radar range cells at which the rain rate is estimated. The cells range from 0 to 79. Each cell is 250m apart, with cell 79 at the earth ellipsiod.
nlayer	13	Number of layers of latent heating.

Figure 52 through Figure 55 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for TRMM Products for details.

InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for TRMM Products for details.

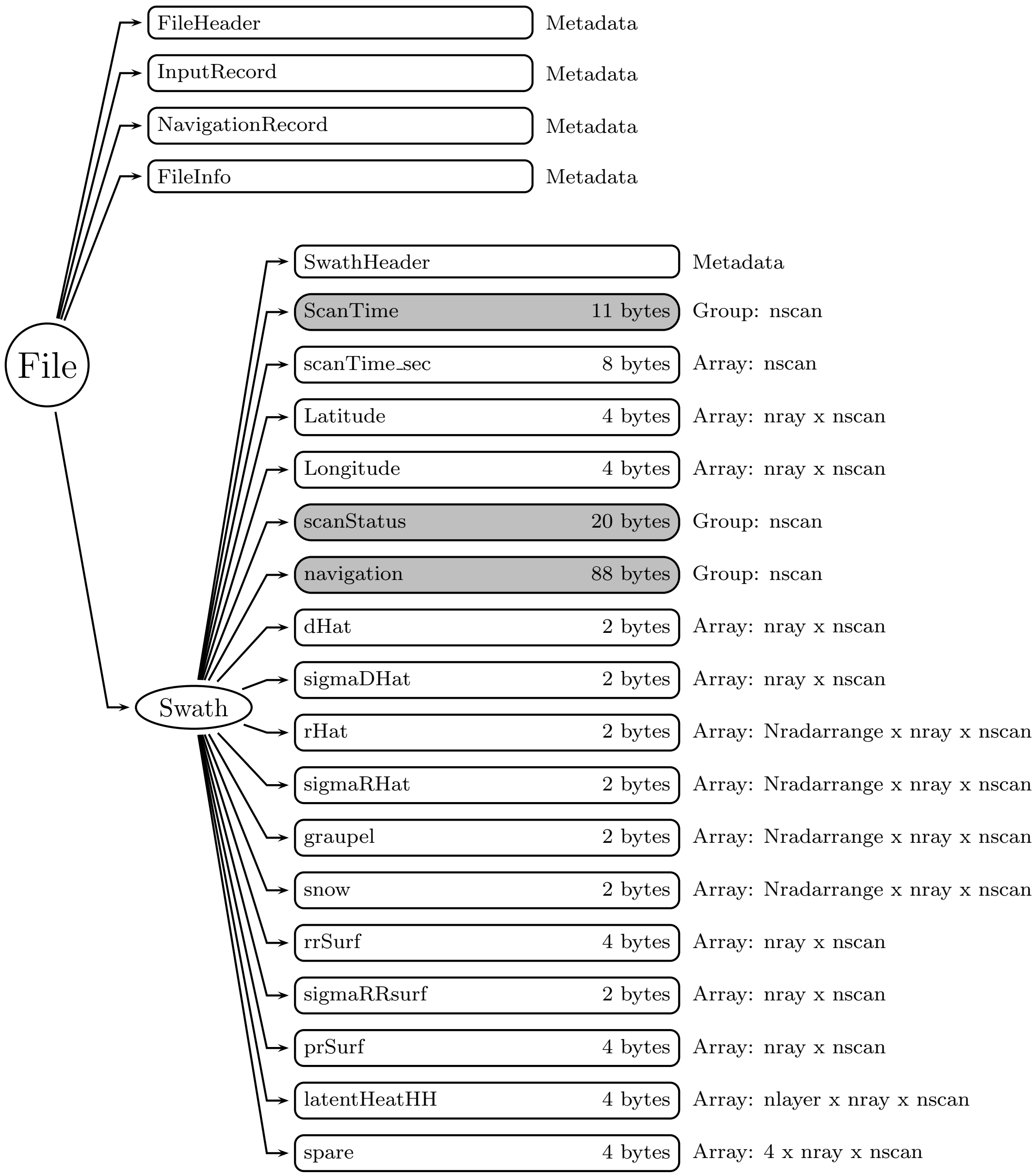


Figure 52: Data Format Structure for 2B31, TRMM Combined

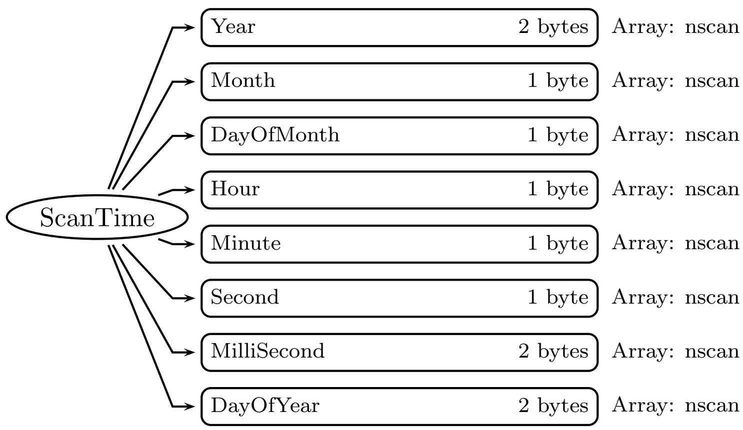


Figure 53: Data Format Structure for 2B31, ScanTime

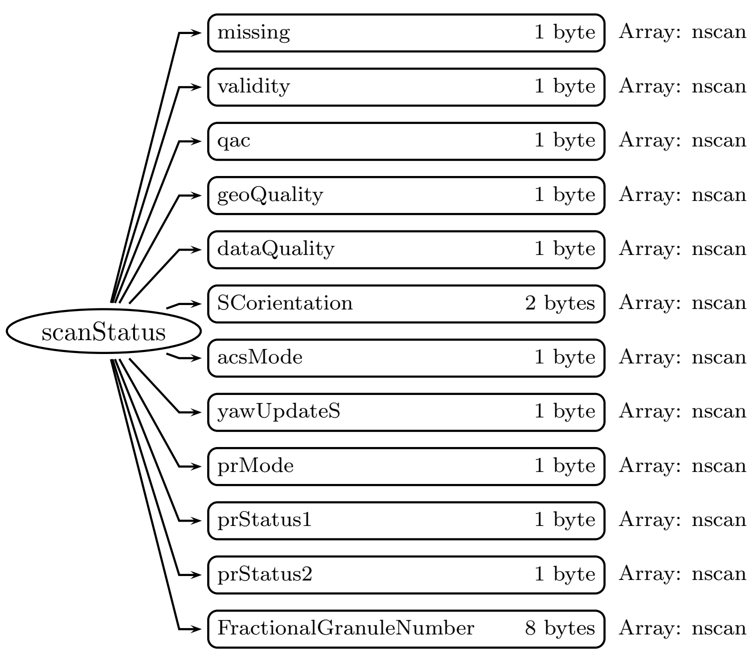


Figure 54: Data Format Structure for 2B31, scanStatus

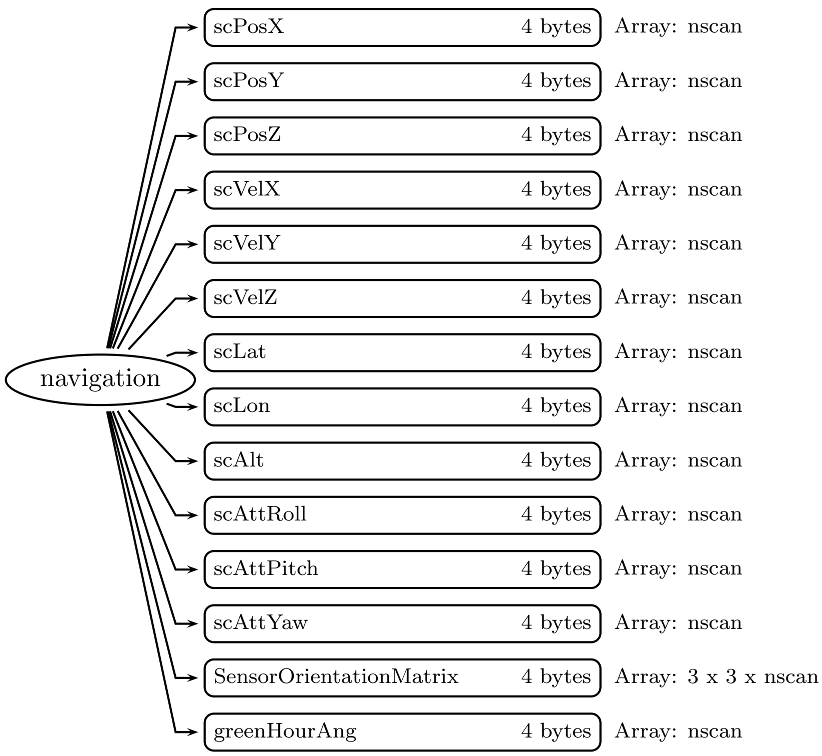


Figure 55: Data Format Structure for 2B31, navigation

NavigationRecord (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for TRMM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for TRMM Products for details.

Swath (Swath)

SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for TRMM Products for details.

ScanTime (Group)

A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:

-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:

-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

scanTime_sec (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: nray x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: nray x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scanStatus (Group)

missing (1-byte integer, array size: nscan):
Missing indicates whether information is contained in the scan data. The values are:

```
0 Scan data elements contain information
1 Scan was missing in the telemetry data
```

```
2 Scan data contains no elements with rain
```

validity (1-byte integer, array size: nscan):
Validity is a summary of status modes. If all status modes are routine, all bits in Validity = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. Validity does not assess data or geolocation quality. Validity is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

```
Bit Meaning if bit = 1
0   Spare (always 0)
1   Non-routine spacecraft orientation (2 or 3)
2   Non-routine ACS mode (other than 4)
3   Non-routine yaw update status (0 or 1)
4   Non-routine instrument status (other than 1)
5   Non-routine QAC (non-zero)
6   Spare (always 0)
7   Spare (always 0)
```

qac (1-byte integer, array size: nscan):
The Quality and Accounting Capsule of the Science packet as it appears in Level-0 data. If no QAC is given in Level-0, which means no decoding errors occurred, QAC in this format has a value of zero.

geoQuality (1-byte integer, array size: nscan):
Geolocation quality is a summary of geolocation quality in the scan. A zero integer value indicates 'good' geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

```
Bit Meaning if bit = 1
0   latitude limit error
1   geolocation
2   attitude change rate limit error
3   attitude limit error
4   satellite undergoing maneuvers
5   using predictive orbit data
6   geolocation calculation error
7   not used
```

dataQuality (1-byte integer, array size: nscan):
Data quality is a summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

```
Bit Meaning if bit = 1
0   missing
5   Geolocation Quality is not normal
6   Validity is not normal
```

SCorientation (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the TMI scan. If +X is forward, SCorientation is 0. If -X is forward, SCorientation is 180. If -Y is forward, SCorientation is 90. Values range from 0 to 360 degrees. Special values are defined as:

-8003	Inertial
-8004	Unknown
-9999	Missing value

acsMode (1-byte integer, array size: nscan):

```
Value Meaning
0     Standby
1     Sun Acquire
2     Earth Acquire
3     Yaw Acquire
4     Nominal
5     Yaw Maneuver
6     Delta-H (Thruster)
7     Delta-V (Thruster)
8     CERES Calibration
```

yawUpdateS (1-byte integer, array size: nscan):

```
Value Meaning
0     Inaccurate
1     Indeterminate
2     Accurate
```

prMode (1-byte integer, array size: nscan):

```
Value Meaning
1     Observation Mode
2     Other Mode
```

prStatus1 (1-byte integer, array size: nscan):
This status is a warning for scan data. Unless this is 0, the scan data may include a little questionable value though it is not a problem (such as break of caution limit). This field is used only for NASDA's data analysis.

prStatus2 (1-byte integer, array size: nscan):
Initialization in Onboard Surface Search Algorithm.

```
Value Meaning
0     Not initialized
1     Initialized
```

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft's trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

navigation (Group)

scPosX (4-byte float, array size: nscan):
The x component of the position (m) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Geocentric Inertial Coordinates are also commonly known as Earth Centered Inertial coordinates. These coordinates will be True of Date (rather than Epoch 2000 which are also commonly used), as interpolated from the data in the Flight Dynamics Facility ephemeris files generated for TRMM.

scPosY (4-byte float, array size: nscan):
The y component of the position (m) of the spacecraft in Geocentric Inertial Coordinates. See scPosX.

scPosZ (4-byte float, array size: nscan):
The z component of the position (m) of the spacecraft in Geocentric Inertial Coordinates. See scPosX.

scVelX (4-byte float, array size: nscan):
The x component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scVelY (4-byte float, array size: nscan):
The y component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scVelZ (4-byte float, array size: nscan):
The z component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scLat (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time.

scLon (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time.

scAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time.

scAttRoll (4-byte float, array size: nscan):
The satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

scAttPitch (4-byte float, array size: nscan):
The satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

scAttYaw (4-byte float, array size: nscan):
The satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system

following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

SensorOrientationMatrix (4-byte float, array size: 3 x 3 x nscan):
SensorOrientationMatrix is the rotation matrix from the instrument coordinate frame to Geocentric Inertial Coordinates at the Scan mid-Time. It is unitless.

greenHourAng (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates.

dHat (2-byte integer, array size: nray x nscan):
dHat is the correlation-corrected mass-weighted mean drop diameter (liquid only). It is multiplied by 100 and stored as a two-byte integer. The accuracy is 0.01 "normalized" mm. It ranges from 0.7 to 1.8 "normalized" mm (the value 0 indicates no rain or bad data). The accuracy is 0.01 "normalized" mm.

The parameters Λ, μ and No of the corresponding drop size distribution $N(D)dD = NoD^{\mu}e^{-\Lambda D}dD$, giving the number per cubic-meter of drops of diameter between D and D + dD mm, can be obtained from dHat and the rain rate rHat using the formulas:

$$\mu = -4 + 1/(0.1521 dHat^{0.23} rHat^{0.074})$$

$$\Lambda = 1/(0.1521 dHat^{1.33} rHat^{0.23})$$

$$No = 55 rHat \Lambda^{\mu+4}/(\Gamma(\mu+4)(1-(1+0.53/\Lambda)^{-\mu-4}))$$

Similarly, the rain rate rHat mm/hr can be converted into a liquid (no solid) water M (g/m3) using the formula:

$$M = \frac{0.02878 rHat}{1-(1+0.53/\Lambda)^{-\mu-4}}$$

The average value of dHat is around 1.1 "normalized" mm, a unit which comes from the fact that dHat is related to the true mass-weighted mean drop diameter D* mm by the formula dHat = D* rHat -0.155 (with rHat in mm/hr).

"normalized units" are defined as follows: If a variable X, expressed in grams, is correlated with the rain rate R and a variable Y is defined where Y = X * R0.37 R, then the unit of Y is called "normalized grams".

sigmaDHat (2-byte integer, array size: nray x nscan):
Sigma-D-hat is the RMS uncertainty in D-Hat. It ranges from 0.00 to 2.00 "normalized"* mm and is multiplied by 100 and stored as a two-byte integer. The accuracy is 0.01 "normalized" mm. See description of dHat.

rHat (2-byte integer, array size: Nradarrange x nray x nscan):
R-hat is the instantaneous rain rate (liquid only) at the radar range gates. It ranges from 0.0 to 500.0 mm/hr and is multiplied by 10 and stored as a two-byte integer. The accuracy is 0.1 mm/hr.

sigmaRHat (2-byte integer, array size: Nradarrange x nray x nscan):
Sigma-R-hat is the RMS uncertainty in the R-hat estimated at the radar range gates. It is multiplied by 10 and stored as a two-byte integer. It ranges from -125 to 125 mm/hr (the negative sign indicating estimates based on a "rain-possible" detection by the radar rather than the "rain-certain" associated with positive values). The values -125 and 125 are reserved for cases where the RMS uncertainty could not be accurately estimated. The accuracy is 0.5 mm/hr.

graupel (2-byte integer, array size: Nradarrange x nray x nscan):
"graupel" content estimated at the radar range gates. It is multiplied by 1000 and stored as a two-byte integer. It ranges from 0 to 10 g/m^3. "graupel" is defined as frozen hydrometeors with a density of 600 Kg/m^3. "graupel" thus is mostly bigger, denser frozen hydrometeors, and would represent aggregates, graupel, and hail.

snow (2-byte integer, array size: Nradarrange x nray x nscan):
"snow" content estimated at the radar range gates. It is multiplied by 1000 and stored as a two-byte integer. It ranges from 0 to 10 g/m^3. "snow" is defined as frozen hydrometeors with a density of 100 Kg/m^3. "snow" thus is mostly smaller, fluffier frozen hydrometeors.

rrSurf (4-byte float, array size: nray x nscan):
RR-Surf is the surface rainfall rate (liquid only). It ranges from 0.0 to 500.0 mm/hr. The accuracy is 0.1 mm/hr.

sigmaRRsurf (2-byte integer, array size: nray x nscan):
Sigma-RR-Surf is the RMS uncertainty in RR-Surf. It is multiplied by 100 and stored as a two-byte integer. It ranges from -125 to 125 mm/hr (the negative sign indicating estimates based on a "rain-possible" detection by the radar rather than the "rain-certain" associated with positive values). The values -125 and 125 are reserved for cases where the RMS uncertainty could not be accurately estimated. The accuracy is 0.5mm/hr.

prSurf (4-byte float, array size: nray x nscan):
The surface precipitation rate (liquid plus solid). It ranges from 0.0 to 500.0 mm/hr. The accuracy is 0.1 mm/hr.

latentHeatHH (4-byte float, array size: nlayer x nray x nscan):
latentHeatHH is the "hydrometeor heating" in K/hr calculated from the vertical fluxes of the different hydrometeor species and using average archival temperature/pressure/humidity soundings which depend on longitude and latitude only. In version 6 all the precipitation is assumed to be liquid. Heating is listed for 13 layers. The first entry in the array is the heating in the layer between 18 km and 16 km above the earth ellipsoid. The layers have the following upper and lower boundaries defined in km above the earth ellipsoid:

```
First layer       18      16
Second layer      16      14
Third layer       14      12
Fourth layer      12      10
Fifth layer       10       8
Sixth layer        8       7
```

```
    Seventh layer       7        6
    Eighth layer        6        5
    Ninth layer         5        4
    Tenth layer         4        3
    Eleventh layer      3        2
    Twelvth layer       2        1
    Thirteenth layer    1        0
```

spare (4-byte float, array size: 4 x nray x nscan):
Contents and ranges are not public.

C Structure Header file:

```
#ifndef _TK_2B31_H_
#define _TK_2B31_H_

#ifndef _L2B31_NAVIGATION_
#define _L2B31_NAVIGATION_

typedef struct {
    float scPosX;
    float scPosY;
    float scPosZ;
    float scVelX;
    float scVelY;
    float scVelZ;
    float scLat;
    float scLon;
    float scAlt;
    float scAttRoll;
    float scAttPitch;
    float scAttYaw;
    float SensorOrientationMatrix[3][3];
    float greenHourAng;
} L2B31_NAVIGATION;

#endif

#ifndef _L2B31_SCANSTATUS_
#define _L2B31_SCANSTATUS_

typedef struct {
    signed char missing;
    signed char validity;
```

```
    signed char qac;
    signed char geoQuality;
    signed char dataQuality;
    short SCorientation;
    signed char acsMode;
    signed char yawUpdateS;
    signed char prMode;
    signed char prStatus1;
    signed char prStatus2;
    double FractionalGranuleNumber;
} L2B31_SCANSTATUS;

#endif

#ifndef _L2B31_SCANTIME_
#define _L2B31_SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
} L2B31_SCANTIME;

#endif

#ifndef _L2B31_SWATH_
#define _L2B31_SWATH_

typedef struct {
    L2B31_SCANTIME ScanTime;
    double scanTime_sec;
    float Latitude[49];
    float Longitude[49];
    L2B31_SCANSTATUS scanStatus;
    L2B31_NAVIGATION navigation;
    float dHat[49];
    float sigmaDHat[49];
    float rHat[49][80];
```

```
    float sigmaRHat[49][80];
    float graupel[49][80];
    float snow[49][80];
    float rrSurf[49];
    float sigmaRRsurf[49];
    float prSurf[49];
    float latentHeatHH[49][13];
    float spare[49][4];
} L2B31_SWATH;

#endif

#endif
```

Fortran Structure Header file:

```
STRUCTURE /L2B31_NAVIGATION/
    REAL*4 scPosX
    REAL*4 scPosY
    REAL*4 scPosZ
    REAL*4 scVelX
    REAL*4 scVelY
    REAL*4 scVelZ
    REAL*4 scLat
    REAL*4 scLon
    REAL*4 scAlt
    REAL*4 scAttRoll
    REAL*4 scAttPitch
    REAL*4 scAttYaw
    REAL*4 SensorOrientationMatrix(3,3)
    REAL*4 greenHourAng
END STRUCTURE

STRUCTURE /L2B31_SCANSTATUS/
    BYTE missing
    BYTE validity
    BYTE qac
    BYTE geoQuality
    BYTE dataQuality
    INTEGER*2 SCorientation
    BYTE acsMode
    BYTE yawUpdateS
    BYTE prMode
```

```
    BYTE prStatus1
    BYTE prStatus2
    REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /L2B31_SCANTIME/
    INTEGER*2 Year
    BYTE Month
    BYTE DayOfMonth
    BYTE Hour
    BYTE Minute
    BYTE Second
    INTEGER*2 MilliSecond
    INTEGER*2 DayOfYear
END STRUCTURE

STRUCTURE /L2B31_SWATH/
    RECORD /L2B31_SCANTIME/ ScanTime
    REAL*8 scanTime_sec
    REAL*4 Latitude(49)
    REAL*4 Longitude(49)
    RECORD /L2B31_SCANSTATUS/ scanStatus
    RECORD /L2B31_NAVIGATION/ navigation
    REAL*4 dHat(49)
    REAL*4 sigmaDHat(49)
    REAL*4 rHat(80,49)
    REAL*4 sigmaRHat(80,49)
    REAL*4 graupel(80,49)
    REAL*4 snow(80,49)
    REAL*4 rrSurf(49)
    REAL*4 sigmaRRsurf(49)
    REAL*4 prSurf(49)
    REAL*4 latentHeatHH(13,49)
    REAL*4 spare(4,49)
END STRUCTURE
```

7.10 3A11 - TMI Emission

3A11, "TMI Emission", produces 5deg x 5deg monthly oceanic rainfall maps using TMI Level-1 data. Statistics of the monthly rainfall will also be calculated. The following sections describe the structure and contents of the format.

Dimension definitions:

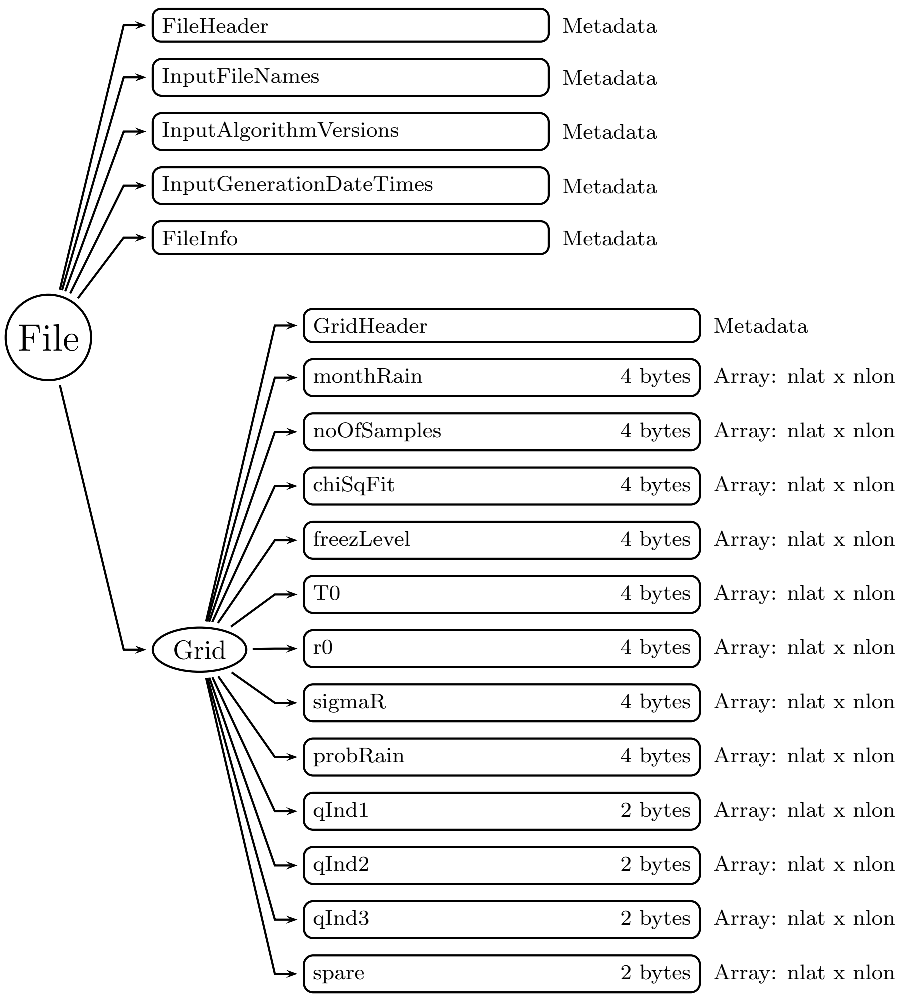


Figure 56: Data Format Structure for 3A11, TMI Emission

nlat	16	Number of 5 degree grid intervals of latitude from 40 N to 40 S.
nlon	72	Number of 5 degree grid intervals of longitude from 180 W to 180 E.

Figure 56 shows the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for TRMM Products for details.

InputFileNames (Metadata):
InputFileNames contains a list of input file names for this granule. See Metadata for TRMM Products for details.

InputAlgorithmVersions (Metadata):

InputAlgorithmVersions contains a list of input algorithm versions for this granule. See Metadata for TRMM Products for details.

InputGenerationDateTimes (Metadata):
InputGenerationDateTimes contains a list of input generation datetimes. See Metadata for TRMM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for TRMM Products for details.

Grid (Grid)

GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for TRMM Products for details.

monthRain (4-byte float, array size: nlat x nlon):
The Monthly Rainfall is the surface rainfall over oceans in 5 x 5 boxes from 40N to 40S. It ranges from 0.0 to 3000.0 mm. Data on land areas are assigned the value -9999.9

noOfSamples (4-byte integer, array size: nlat x nlon):
The Number of Samples is that over oceans in 5 x 5 boxes for one month. It ranges from 0 to 500,000. Data on land areas are assigned the value -9999.

chiSqFit (4-byte integer, array size: nlat x nlon):
The Chi Square Fit indicates how well the histogram of brightness temperatures fits the lognormal distribution function in 5 x 5 boxes for one month. It ranges from 1 to 1,000,000,000. Data on land areas are assigned the value -9999.

freezLevel (4-byte float, array size: nlat x nlon):
The Freezing Level is the estimated height of 0C isotherm over oceans in 5 x 5 boxes for one month. It ranges from 0.00 to 6.00 km. Data on land areas are assigned the value -9999.9

T0 (4-byte float, array size: nlat x nlon):
The T_0 is the mean of non-raining brightness temperatures over oceans in 5 x 5 boxes for one month. It ranges from 160.0 to 180.0 K. Data on land areas are assigned the value -9999.9

r0 (4-byte float, array size: nlat x nlon):
The r_0 is the logarithmic mean rain rate over oceans in 5 x 5 boxes for one month. It ranges from 0.00 to 15.00 mm/hr. Data on land areas are assigned the value -9999.9

sigmaR (4-byte float, array size: nlat x nlon):
The Sigma_r(sigmar) is the standard deviation of logarithmic rain rates over oceans in 5 x 5 boxes for one month. It ranges from 0.00 to 1.00 mm h-1. Data on land areas are assigned the value -9999.9

probRain (4-byte float, array size: nlat x nlon):
The Probability of Rain is that over oceans in 5 x 5 boxes for one month. It ranges from 0.000 to 1.000. Data on land areas are assigned the value -9999.9

qInd1 (2-byte integer, array size: nlat x nlon):
TBD

qInd2 (2-byte integer, array size: nlat x nlon):
TBD

qInd3 (2-byte integer, array size: nlat x nlon):
TBD

spare (2-byte integer, array size: nlat x nlon):
TBD.

C Structure Header file:

```
#ifndef _TK_3A11_H_
#define _TK_3A11_H_

#ifndef _L3A11_GRID_
#define _L3A11_GRID_

typedef struct {
    float monthRain[72][16];
    int noOfSamples[72][16];
    int chiSqFit[72][16];
    float freezLevel[72][16];
    float T0[72][16];
    float r0[72][16];
    float sigmaR[72][16];
    float probRain[72][16];
    short qInd1[72][16];
    short qInd2[72][16];
    short qInd3[72][16];
    short spare[72][16];
} L3A11_GRID;

#endif

#endif
```

Fortran Structure Header file:

```
STRUCTURE /L3A11_GRID/
```

```
    REAL*4 monthRain(16,72)
    INTEGER*4 noOfSamples(16,72)
    INTEGER*4 chiSqFit(16,72)
    REAL*4 freezLevel(16,72)
    REAL*4 T0(16,72)
    REAL*4 r0(16,72)
    REAL*4 sigmaR(16,72)
    REAL*4 probRain(16,72)
    INTEGER*2 qInd1(16,72)
    INTEGER*2 qInd2(16,72)
    INTEGER*2 qInd3(16,72)
    INTEGER*2 spare(16,72)
END STRUCTURE
```

7.11 3A12 - Monthly TMI Profiling

3A12, "Monthly TMI Profiling", produces global 0.5^o x 0.5^o monthly gridded means using 2A12 data. Vertical hydrometeor profiles and surface rainfall means are computed. Various pixel counts are also reported. The PI is Joyce Chou. The granule size is one month. The following sections describe the structure and contents of the format.

Dimension definitions:

nlat	160	Number of 0.5^o grid intervals of latitude from 40^oN to 40^oS.
nlon	720	Number of 0.5^o grid intervals of longitude from 180^oW to 180^oE.
nlayer	28	Number of profiling layers. The top of each layer is 0.5, 1.0, 1.5, ..., 9.5, 10.0, 11.0, ..., 18.0 km. The layer tops are heights above the earth's surface.

Figure 57 shows the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for TRMM Products for details.

InputFileNames (Metadata):
InputFileNames contains a list of input file names for this granule. See Metadata for TRMM Products for details.

InputAlgorithmVersions (Metadata):
InputAlgorithmVersions contains a list of input algorithm versions for this granule. See Metadata for TRMM Products for details.

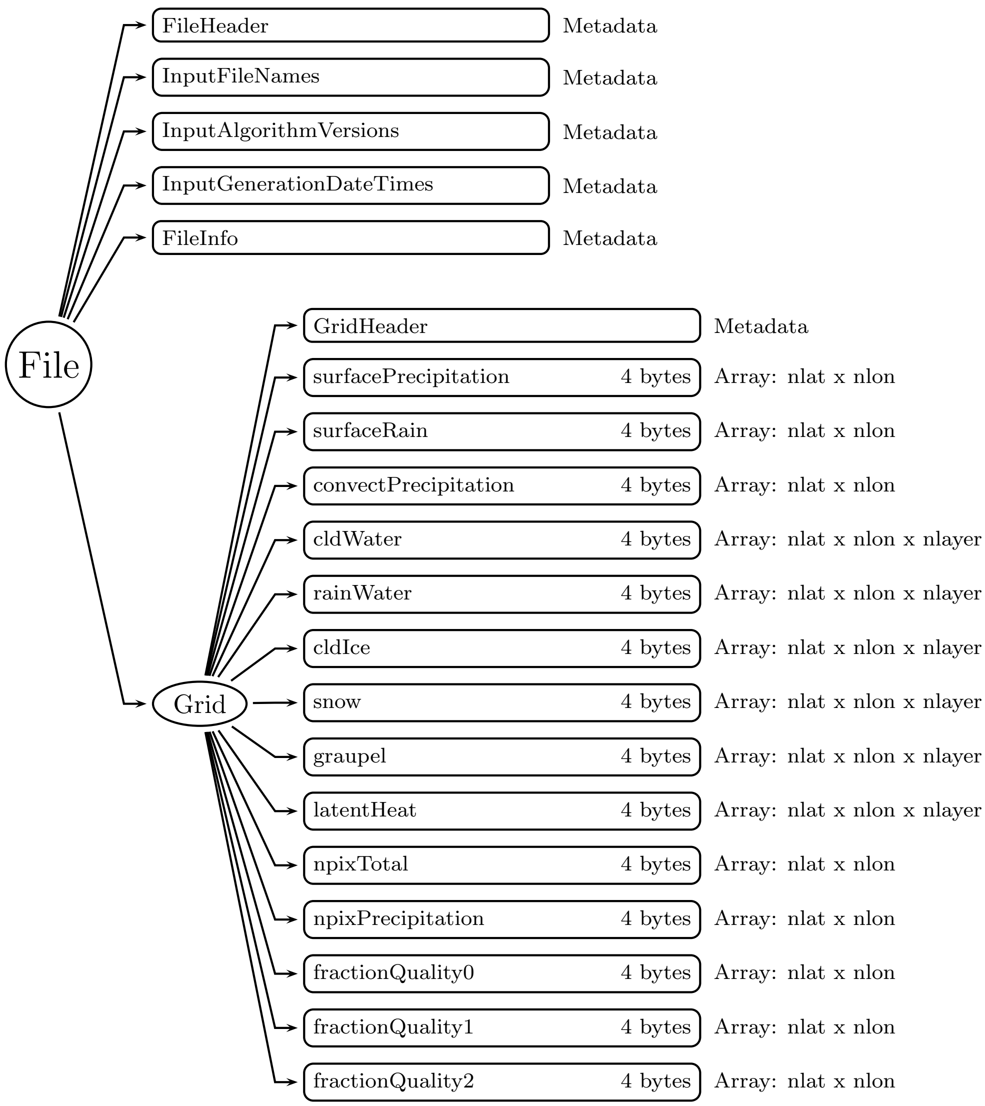


Figure 57: Data Format Structure for 3A12, Monthly TMI Profiling

InputGenerationDateTimes (Metadata):
InputGenerationDateTimes contains a list of input generation datetimes. See Metadata for TRMM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for TRMM Products for details.

Grid (Grid)

GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for TRMM Products for details.

surfacePrecipitation (4-byte float, array size: nlat x nlon):
The monthly mean of the instantaneous precipitation rate at the surface for each grid. Values range from 0 to 3000 mm/hr. Special values are defined as:
-9999.9 Missing value

surfaceRain (4-byte float, array size: nlat x nlon):
The monthly mean of the instantaneous rain rate (liquid portion of precipitation) at the surface for each grid. Values range from 0 to 3000 mm/hr. Special values are defined as:
-9999.9 Missing value

convectPrecipitation (4-byte float, array size: nlat x nlon):
The monthly mean of the instantaneous convective precipitation rate at the surface for each grid. Values range from 0 to 3000 mm/hr. Special values are defined as:
-9999.9 Missing value

cldWater (4-byte float, array size: nlat x nlon x nlayer):
The monthly mean of the cloud liquid water content for each grid at each vertical layer. Values range from 0 to 10 g/m^3. Special values are defined as:
-9999.9 Missing value

rainWater (4-byte float, array size: nlat x nlon x nlayer):
The monthly mean of the rain water content for each grid at each vertical layer. Values range from 0 to 10 g/m^3. Special values are defined as:
-9999.9 Missing value

cldIce (4-byte float, array size: nlat x nlon x nlayer):
The monthly mean of the cloud ice liquid water content for each grid at each vertical layer. Values range from 0 to 10 g/m^3. Special values are defined as:
-9999.9 Missing value

snow (4-byte float, array size: nlat x nlon x nlayer):
The monthly mean of the snow liquid water content for each grid at each vertical layer. Values range from 0 to 10 g/m^3. Special values are defined as:
-9999.9 Missing value

graupel (4-byte float, array size: nlat x nlon x nlayer):
The monthly mean of the graupel liquid water content for each grid at each vertical layer. Values range from 0 to 10 g/m^3. Special values are defined as:
-9999.9 Missing value

latentHeat (4-byte float, array size: nlat x nlon x nlayer):
The monthly mean of the latent heating release for each grid at each vertical layer. Values range from -256 to 256 C/hr. Special values are defined as:
-9999.9 Missing value

npixTotal (4-byte integer, array size: nlat x nlon):
The monthly number of pixels with pixelStatus equal to zero for each grid. The major effect of the pixelStatus requirement is to remove sea ice. npixTotal is used to compute the monthly means described above. Values range from 0 to 10000. Special values are defined as:
-9999 Missing value

npixPrecipitation (4-byte integer, array size: nlat x nlon):
The monthly number of pixels with surfacePrecipitation greater than 0 for each grid. For ocean, a pixel is also required to have probabilityOfPrecip greater than 50 percent. Values range from 0 to 10000. Special values are defined as:
-9999 Missing value

fractionQuality0 (4-byte float, array size: nlat x nlon):
The fraction of total pixels with qualityFlag equal to 0 (high quality) for each grid. Values range from 0 to 100 percent. Special values are defined as:
-9999.9 Missing value

fractionQuality1 (4-byte float, array size: nlat x nlon):
The fraction of total pixels with qualityFlag equal to 1 (medium quality) for each grid. Values range from 0 to 100 percent. Special values are defined as:
-9999.9 Missing value

fractionQuality2 (4-byte float, array size: nlat x nlon):
The fraction of total pixels with qualityFlag equal to 2 (low quality) for each grid. Values range from 0 to 100 percent. Special values are defined as:
-9999.9 Missing value

C Structure Header file:

```
#ifndef _TK_3A12_H_
#define _TK_3A12_H_

#ifndef _L3A12_GRID_
#define _L3A12_GRID_

typedef struct {
    float surfacePrecipitation[720][160];
```

```
    float surfaceRain[720][160];
    float convectPrecipitation[720][160];
    float cldWater[28][720][160];
    float rainWater[28][720][160];
    float cldIce[28][720][160];
    float snow[28][720][160];
    float graupel[28][720][160];
    float latentHeat[28][720][160];
    int npixTotal[720][160];
    int npixPrecipitation[720][160];
    float fractionQuality0[720][160];
    float fractionQuality1[720][160];
    float fractionQuality2[720][160];
} L3A12_GRID;

#endif

#endif
```

Fortran Structure Header file:

```
STRUCTURE /L3A12_GRID/
    REAL*4 surfacePrecipitation(160,720)
    REAL*4 surfaceRain(160,720)
    REAL*4 convectPrecipitation(160,720)
    REAL*4 cldWater(160,720,28)
    REAL*4 rainWater(160,720,28)
    REAL*4 cldIce(160,720,28)
    REAL*4 snow(160,720,28)
    REAL*4 graupel(160,720,28)
    REAL*4 latentHeat(160,720,28)
    INTEGER*4 npixTotal(160,720)
    INTEGER*4 npixPrecipitation(160,720)
    REAL*4 fractionQuality0(160,720)
    REAL*4 fractionQuality1(160,720)
    REAL*4 fractionQuality2(160,720)
END STRUCTURE
```

7.12 3A25 - PR Rainfall

3A25, "PR Rainfall", computes monthly statistics of the PR measurements at both a low horizontal resolution (5 x 5 latitude/longitude) and a high horizontal resolution (0.5o x

0.5o latitude/longitude). The low resolution grids are in the Planetary Grid 1 structure and include 1) mean and standard deviation of the rain rate, reflectivity, path-integrated attenuation (PIA), storm height, Xi, bright band height and the NUBF (Non-Uniform Beam Filling) correction; 2) rain fractions; 3) histograms of the storm height, bright-band height, snow-ice layer, reflectivity, rain rate, path-attenuation and NUBF correction; 4) correlation coefficients. For the high resolution grids in the Planetary Grid 2 structure, mean rain rate along with standard deviation and rain fractions are computed. The following sections describe the structure and contents of the format.

Dimension definitions:

nlat	16	Number of 5^o grid intervals of latitude from 40^oN to 40^oS.
nlon	72	Number of 5^o grid intervals of longitude from 180^oW to 180^oE.
nlath	148	Number of 0.5^odegree grid intervals of latitude from 37^oN to 37^oS.
nlonh	720	Number of 0.5^odegree grid intervals of longitude from 180^oW to 180^oE.
nh1	6	Number of fixed heights above the earth ellipsoid, at 2, 4, 6, 10, and 15 km plus one for path-average.
nh3	4	Number of fixed heights above the earth ellipsoid, at 2, 4, and 6 km plus one for path-average.
nang	5	Number of fixed incidence angles, at 0, 5, 10 and 15 degree.
ncat2	30	Second number of categories for histograms (30). Note that the number of thresholds is one greater than the number of categories. Thresholds are given below for several variables, others are TBD. Reflectivity (dBZ) (bhz): 0.01, 12., 14., 16., 18., 20., 22., 24., 26., 28., 30., 32., 34., 36., 38., 40., 42., 44., 46., 48., 50., 52., 54., 56., 58., 60., 62., 64., 66., 68., 70. Rainfall rate (mm/hr): 0.01 0.205, 0.273, 0.364, 0.486, 0.648, 0.864, 1.153, 1.537, 2.050, 2.734, 3.646, 4.862, 6.484, 8.646, 11.53, 15.37, 20.50, 27.34, 36.46, 48.62, 64.84, 86.46, 115.3, 153.7, 205.0, 273.4, 364.6, 486.2, 648.4, 864.6 Bright Band Height (km) (bhbb): 0.01, 0.25, 0.5, 0.75, 1., 1.25, 1.5, 1.75, 2., 2.25, 2.5, 2.75, 3., 3.25, 3.5, 3.75, 4., 4.25, 4.5, 4.75, 5., 5.25, 5.5, 5.75, 6., 6.25, 6.5, 6.75, 7., 7.5, 20. Storm Height (km) (bhstorm): 0.01, 0.5, 1., 1.5, 2., 2.5, 3., 3.5, 4., 4.5, 5., 5.5, 6., 6.5, 7., 7.5, 8., 8.5, 9., 9.5, 10., 10.5, 11., 11.5, 12., 12.5, 13., 14., 15., 16., 20. snowIceH, bbNadirWidthH: 0.0, 125.0, 250.0, 375.0, 500.0, 625.0, 750.0, 875.0, 1000.0, 1125.0, 1250.0, 1375.0, 1500.0, 1625.0, 1750.0, 1875.0, 2000.0, 2125.0, 2250.0, 2375.0, 2500.0, 2625.0, 2750.0, 2875.0, 3000.0, 3125.0, 3250.0, 3375.0, 3500.0, 3625.0, 3750.0 Snow Depth (km) (bhdepth): 0.01, 0.5, 0.75, 1., 1.25, 1.5, 1.75, 2., 2.25, 2.5, 2.75, 3., 3.25, 3.5, 3.75, 4., 4.25, 4.5, 4.75, 5., 5.25, 5.5, 5.75, 6., 6.25, 6.5, 6.75, 7., 7.25, 7.5, 20. Zpzm (km) (bhzpzm): 0., 1., 2., 3., 4., 5., 6., 7., 8., 9., 10., 11., 12., 13., 14., 15., 16., 17., 18., 19., 20., 22., 24., 26., 28., 30., 32., 34., 36., 38., 50. All PIA (dB) (bhpia): 0.01, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.8, 1.0, 1.2, 1.4, 1.6, 1.8, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0, 5.5, 6.0, 6.5, 7.0, 7.5, 8.0, 8.5, 9.0, 9.5, 10., 100. NUBF or Non-Uniform Beam Filling Factor (unitless) (bhnubf): 1., 1.05, 1.1, 1.15, 1.2, 1.25, 1.3, 1.35, 1.4, 1.45, 1.5, 1.55, 1.6, 1.65, 1.7, 1.75, 1.8, 1.85, 1.9, 1.95, 2., 2.1, 2.2, 2.3, 2.4, 2.5, 2.6, 2.7, 2.8, 2.9, 3.0 Xi or Horizontal Non-Uniformity Parameter (unitless) (bhxi): 0., 0.2, 0.4, 0.6, 0.8, 1., 1.2, 1.4, 1.6, 1.8, 2., 2.2, 2.4, 2.6, 2.8, 3., 3.2, 3.4, 3.6, 3.8, 4., 4.2, 4.4, 4.6, 4.8, 5., 10., 20., 30., 50., 10000. Epsilon conditioned on use of SRT (unitless) (bhepsilon): 0., 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1., 1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7, 1.8, 1.9, 2., 2.1, 2.2, 2.3, 2.4, 2.5, 2.6, 2.7, 2.8, 2.9, 3.0

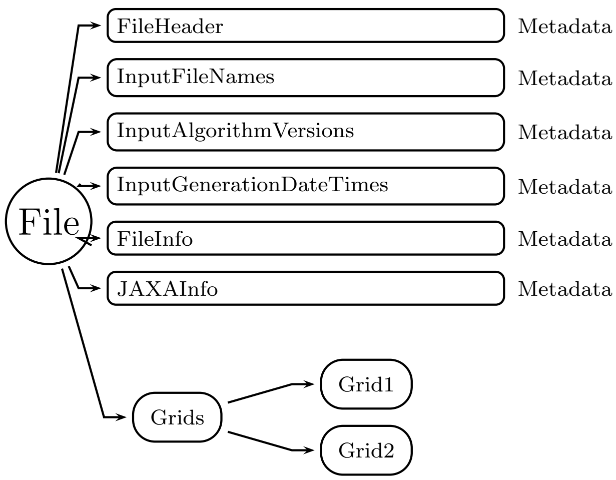


Figure 58: Data Format Structure for 3A25, PR Rainfall

Figure 58 through Figure 68 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for TRMM Products for details.

InputFileNames (Metadata):
InputFileNames contains a list of input file names for this granule. See Metadata for TRMM Products for details.

InputAlgorithmVersions (Metadata):
InputAlgorithmVersions contains a list of input algorithm versions for this granule. See Metadata for TRMM Products for details.

InputGenerationDateTimes (Metadata):
InputGenerationDateTimes contains a list of input generation datetimes. See Metadata for TRMM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for TRMM Products for details.

JAXAInfo (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by PR algorithms only. See Metadata for TRMM Products for details.

Grids (Group)

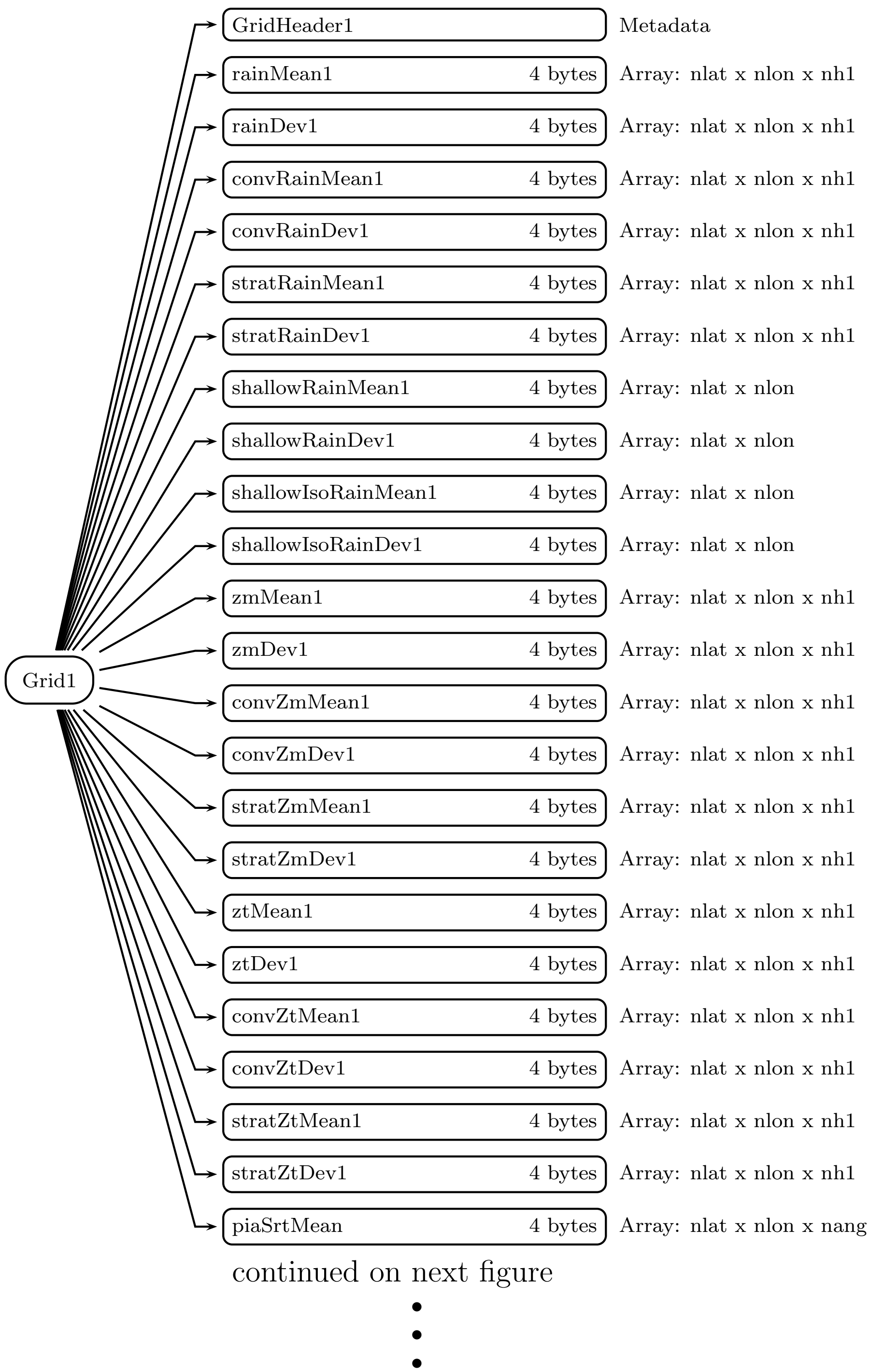


Figure 59: Data Format Structure for 3A25, Grid1, Grid1

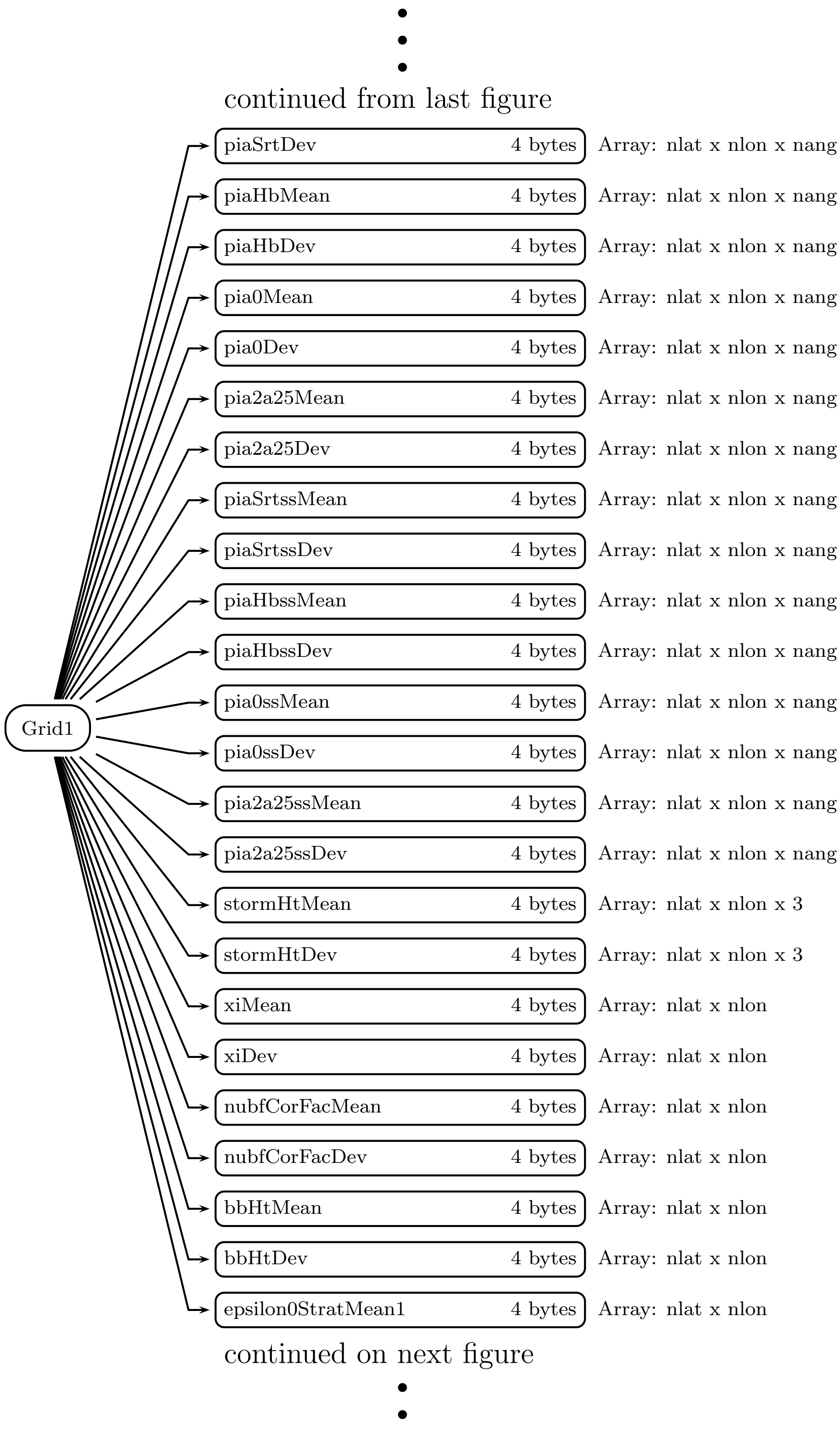


Figure 60: Data Format Structure for 3A25, Grid1, Grid1

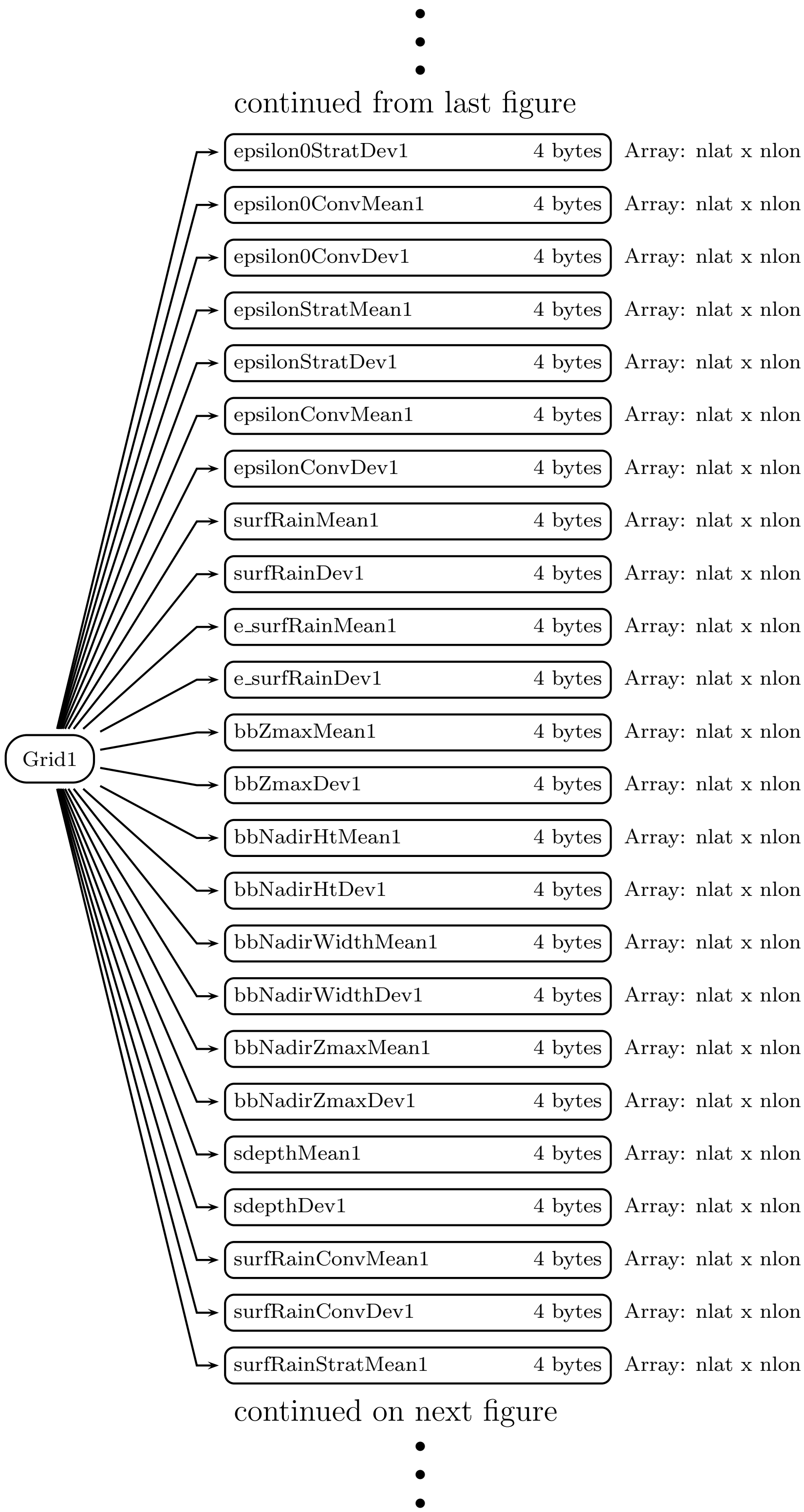


Figure 61: Data Format Structure for 3A25, Grid1, Grid1

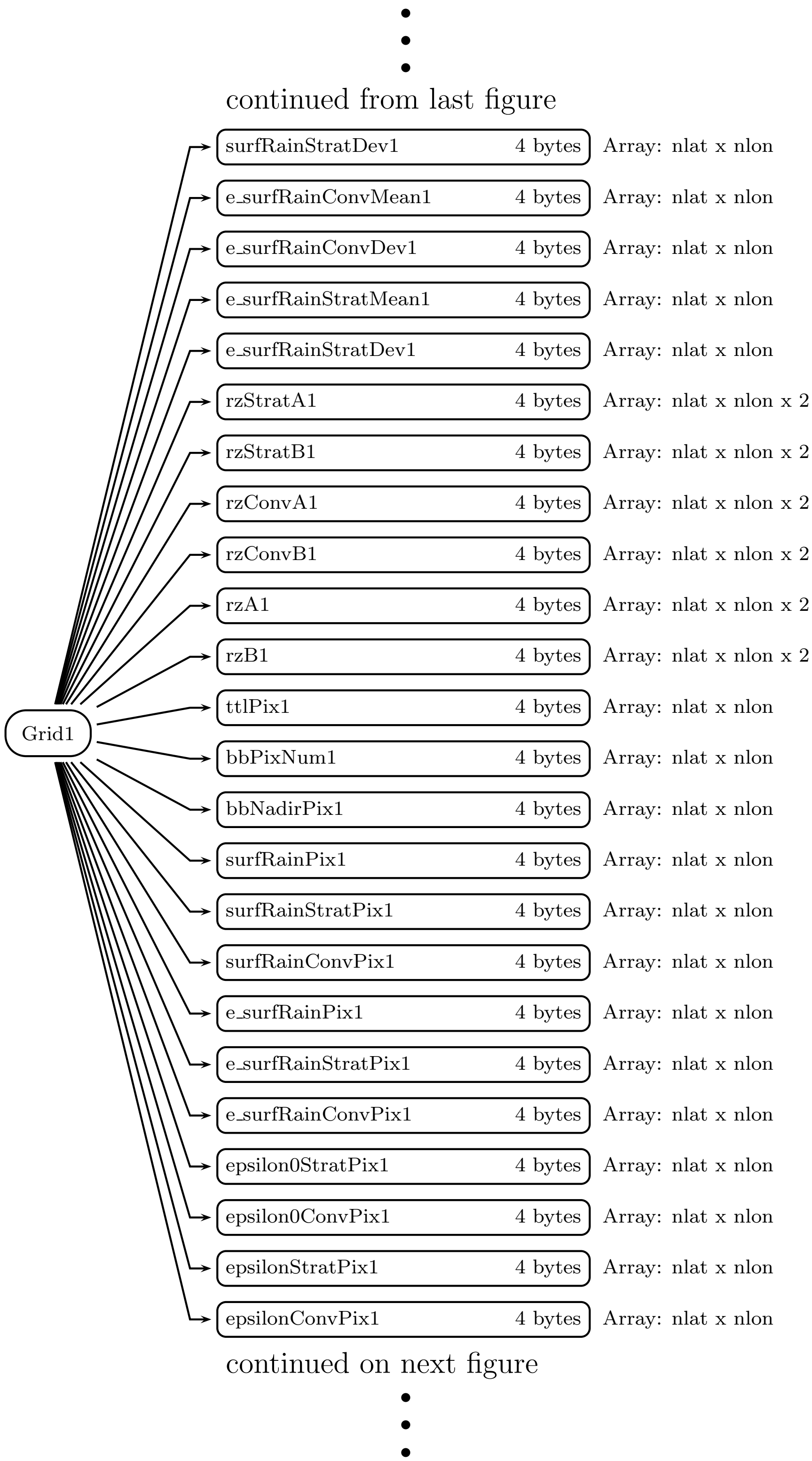


Figure 62: Data Format Structure for 3A25, Grid1, Grid1

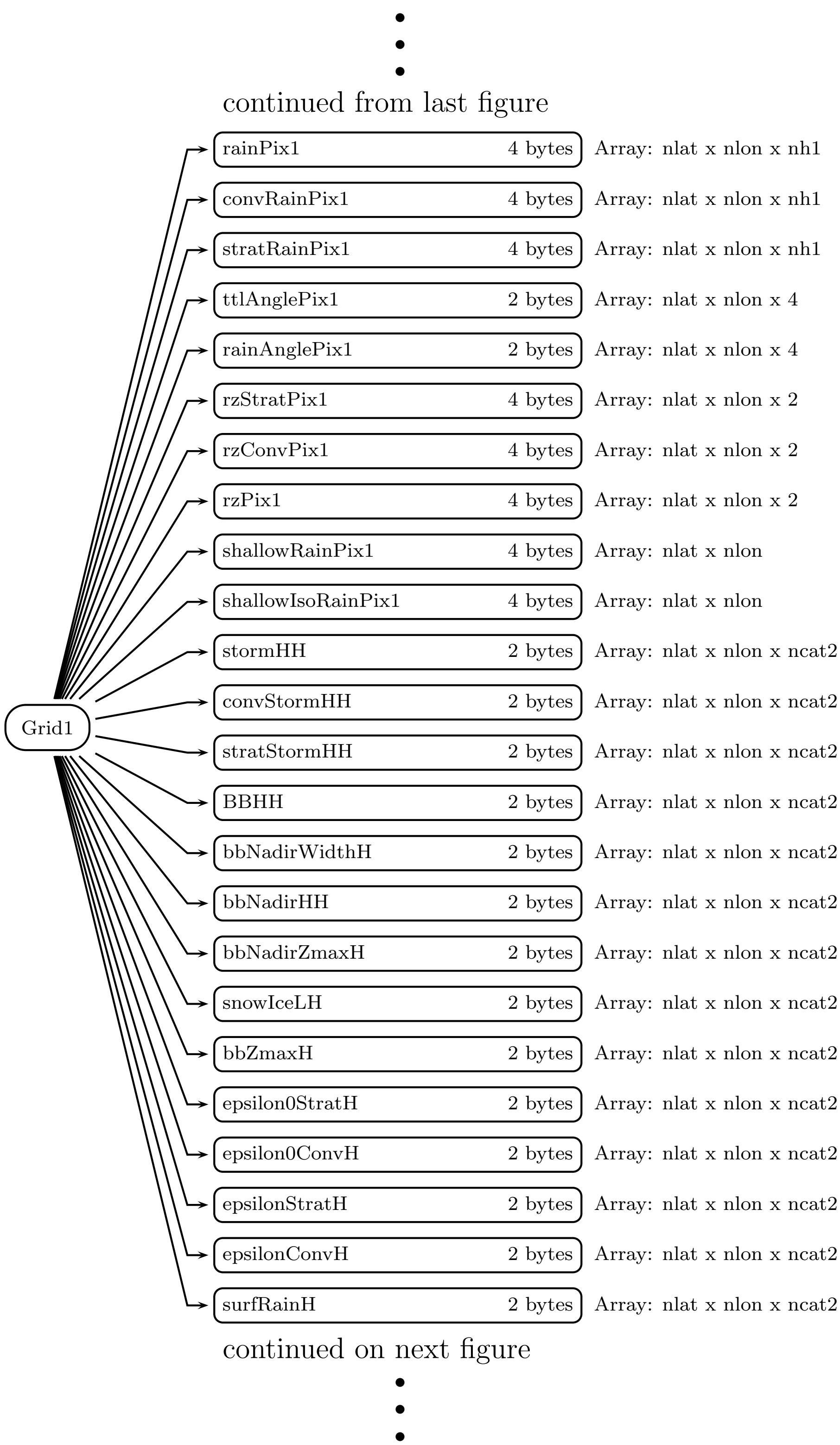


Figure 63: Data Format Structure for 3A25, Grid1, Grid1

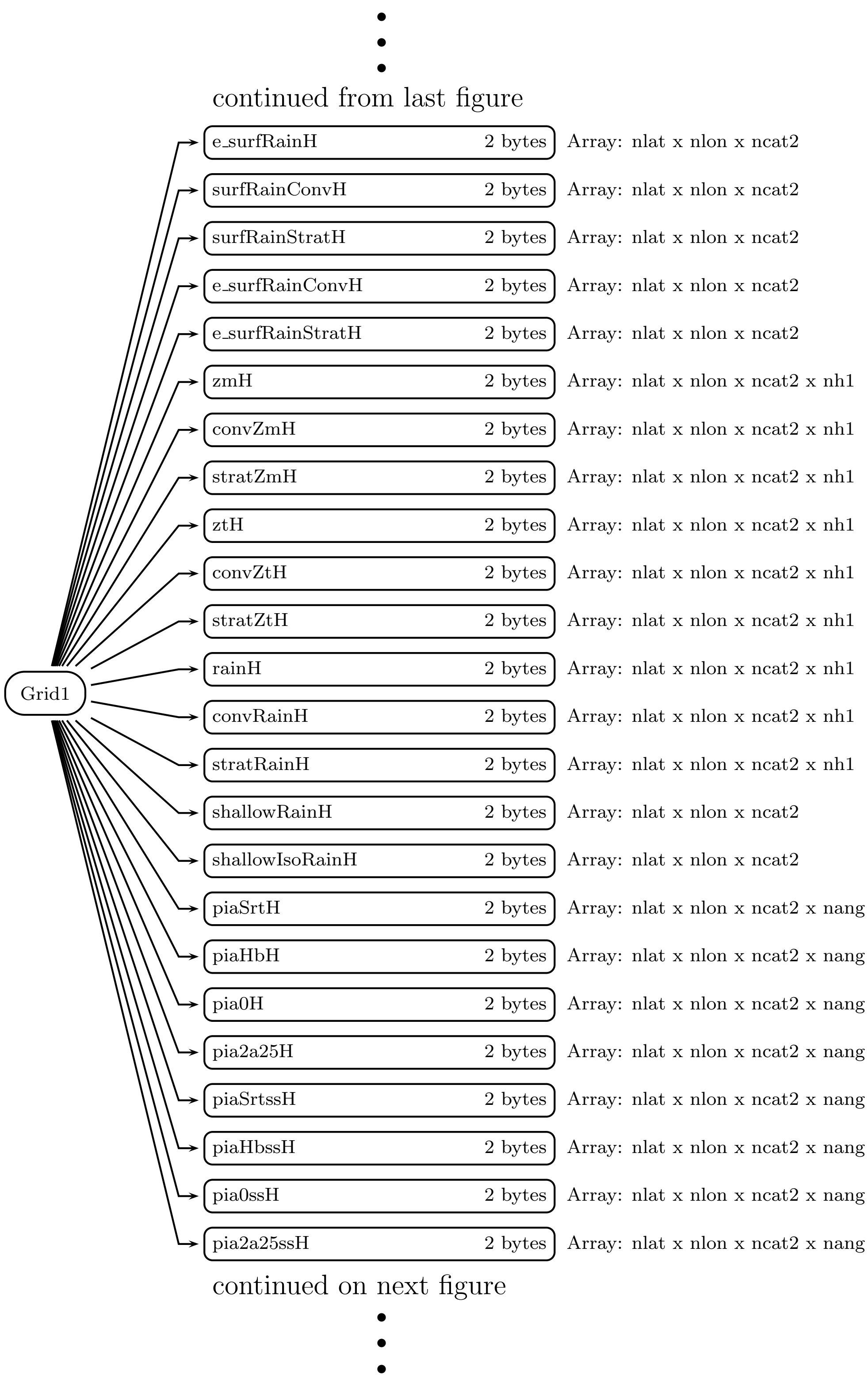


Figure 64: Data Format Structure for 3A25, Grid1, Grid1

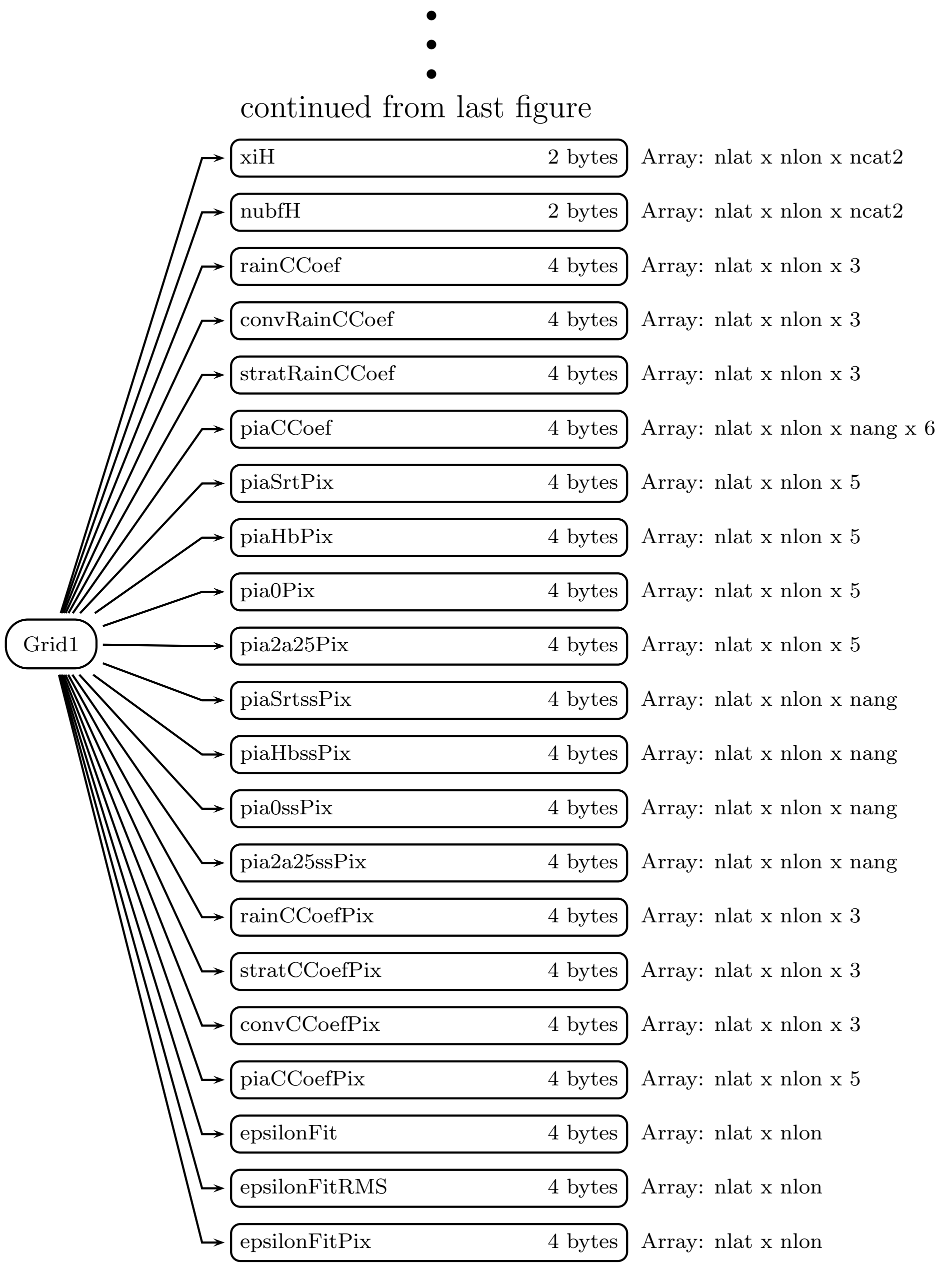


Figure 65: Data Format Structure for 3A25, Grid1

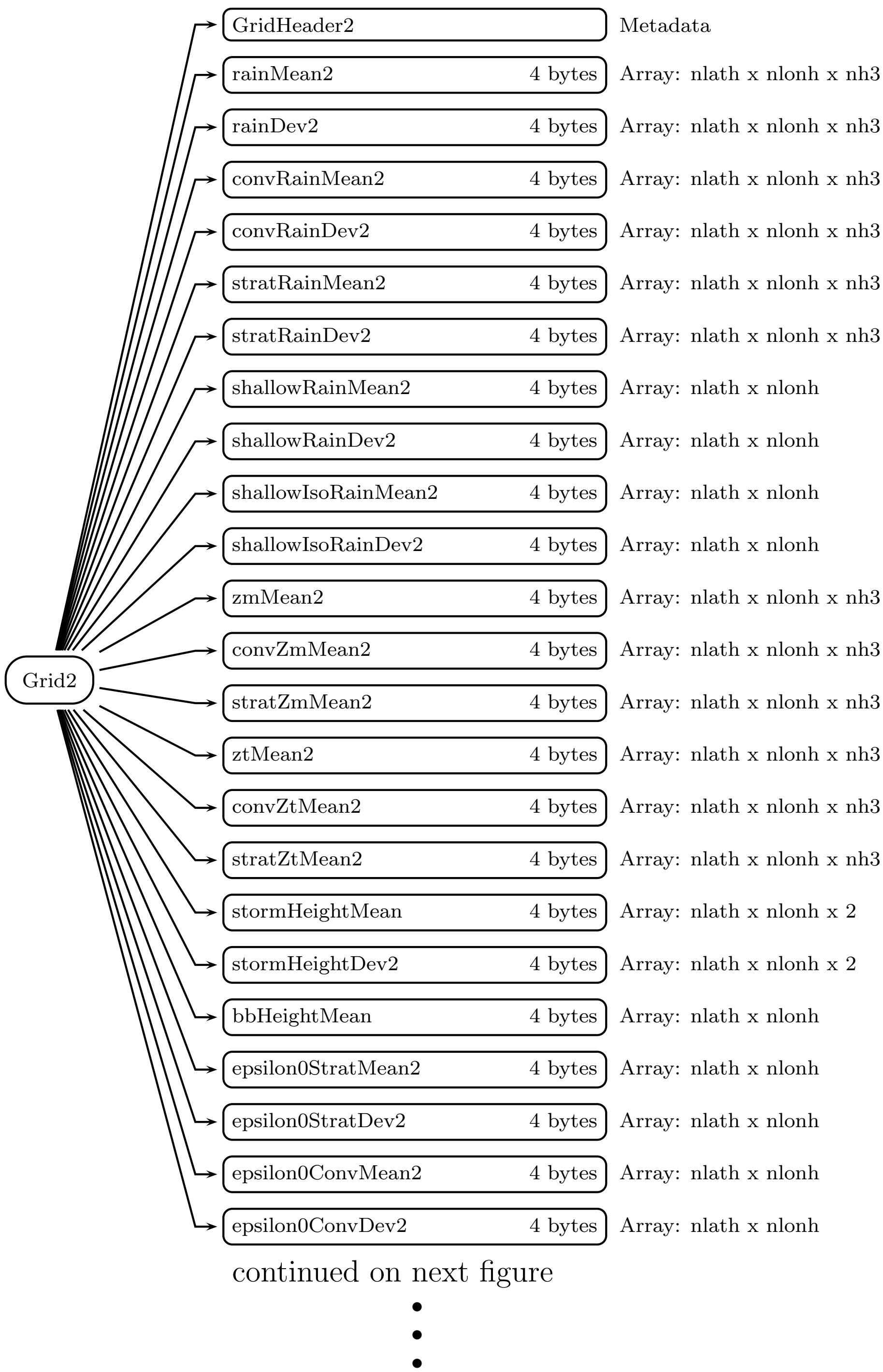


Figure 66: Data Format Structure for 3A25, Grid2, Grid2

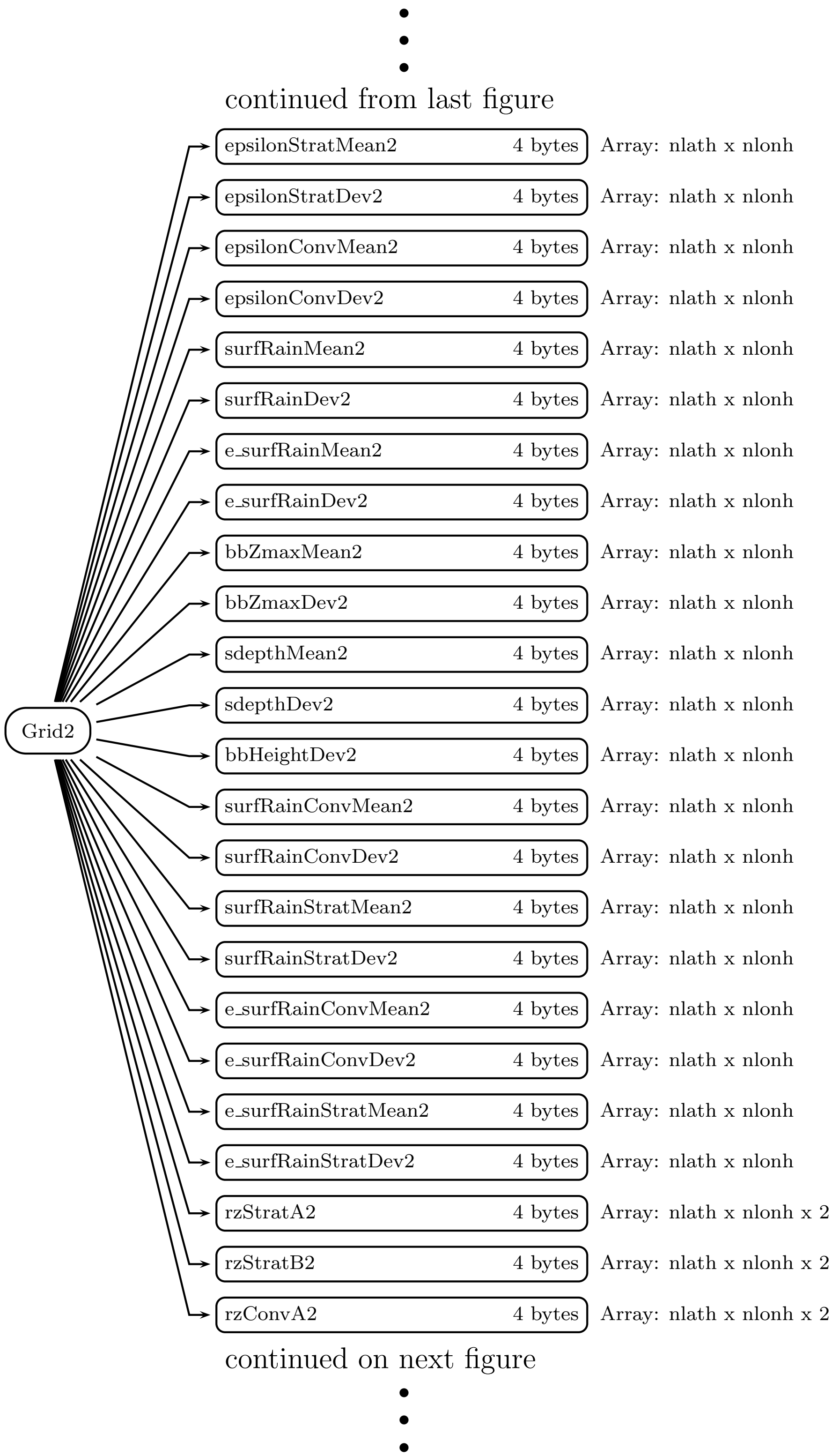


Figure 67: Data Format Structure for 3A25, Grid2, Grid2

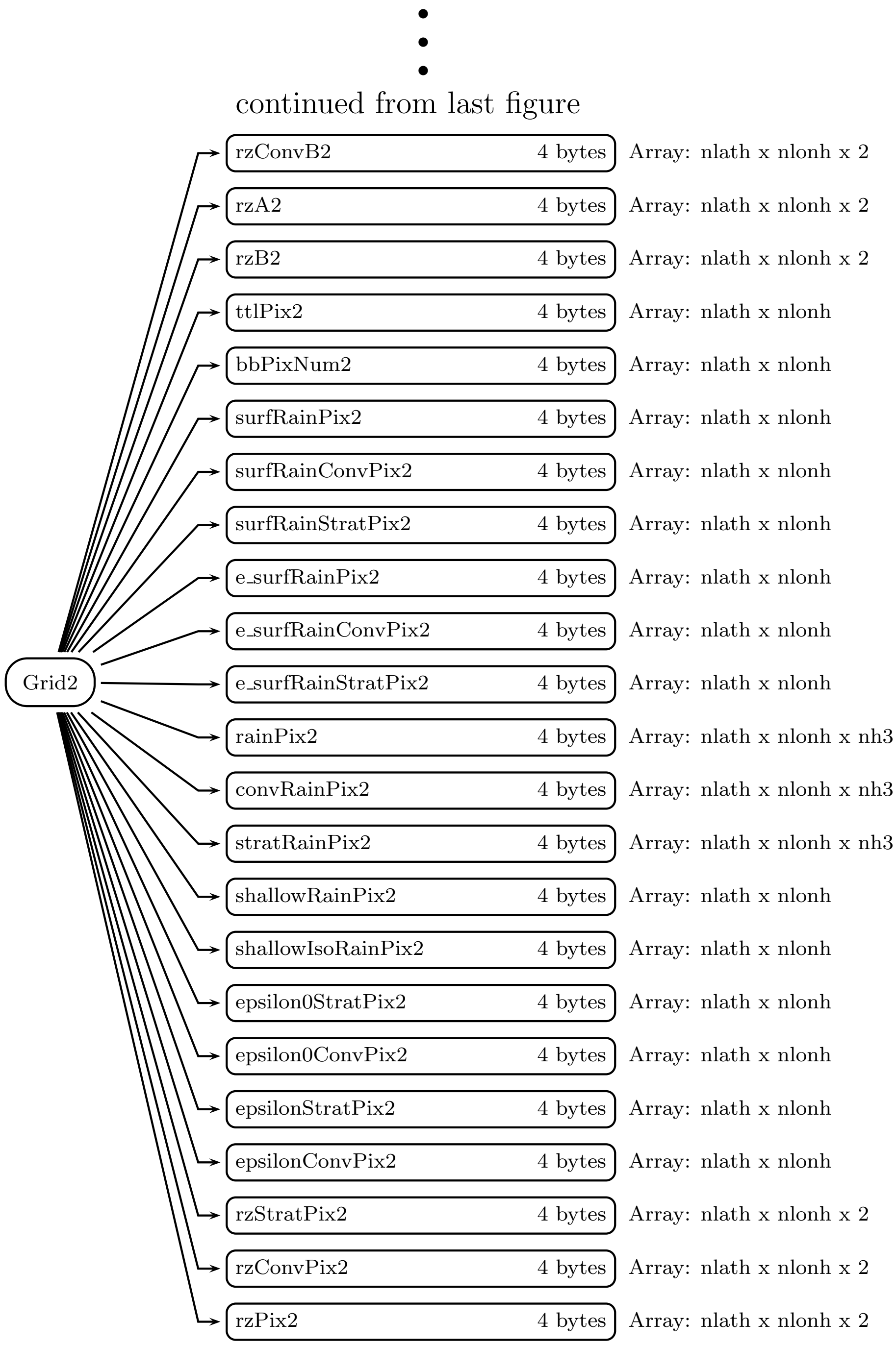


Figure 68: Data Format Structure for 3A25, Grid2

Grid1 (Grid)

GridHeader1 (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for TRMM Products for details.

rainMean1 (4-byte float, array size: nlat x nlon x nh1):
Rain Rate Mean 1 gives means of non-zero rain rates over 5 x 5 boxes for one month. The rain rates are determined in 2A-25 and evaluated for the path-average and at the fixed heights of 2, 4, 6, 10 and 15 km. It ranges from 0.0 to 3000.0 mm/hr.

rainDev1 (4-byte float, array size: nlat x nlon x nh1):
These are standard deviations of non-zero rain rates over 5 x 5 boxes for one month. The rain rates are determined in 2A-25 and evaluated for path-average and at the fixed heights of 2, 4, 6, 10 and 15 km. It ranges from 0.0 to 3000.0 mm/hr.

convRainMean1 (4-byte float, array size: nlat x nlon x nh1):
Conv. Rain Rate Mean 1 gives means of non-zero rain rates for convective rain over 5 x 5 boxes for one month. The rain rates are determined in 2A-25 and evaluated for path-average and at the fixed heights of 2, 4, 6, 10 and 15 km. It ranges from 0.0 to 3000.0 mm/hr.

convRainDev1 (4-byte float, array size: nlat x nlon x nh1):
Conv. Rain Rates Dev. 1 gives standard deviations of non-zero rain rates for convective rain over 5 x 5 boxes for one month. The rain rates are determined in 2A-25 and evaluated for path-average and at the fixed heights of 2, 4, 6, 10 and 15 km. It ranges from 0.0 to 3000.0 mm/hr.

stratRainMean1 (4-byte float, array size: nlat x nlon x nh1):
Strat. Rain Rate Mean 1 gives means of non-zero rain rates for stratiform rain over 5 x 5 boxes for one month. The rain rates are determined in 2A-25 and evaluated for path-average and at the fixed heights of 2, 4, 6, 10 and 15 km. It ranges from 0.0 to 3000.0 mm/hr.

stratRainDev1 (4-byte float, array size: nlat x nlon x nh1):
Strat. Rain Rates Dev. 1 gives standard deviations of non-zero rain rates for stratiform rain over 5 x 5 boxes for one month. The rain rates are determined in 2A-25 and evaluated for path-average and at the fixed heights of 2, 4, 6, 10 and 15 km. It ranges from 0.0 to 3000.0 mm/hr.

shallowRainMean1 (4-byte float, array size: nlat x nlon):
Mean of non-zero shallow rain at a horizontal resolution of 5 x 5. It ranges from 0.0 to 3000.0 mm/hr.

shallowRainDev1 (4-byte float, array size: nlat x nlon):
Standard deviation of non-zero shallow rain at a horizontal resolution of 5^o x 5^o. It ranges from 0.0 to 3000.0 mm/hr.

shallowIsoRainMean1 (4-byte float, array size: nlat x nlon):
Mean of non-zero isolated shallow rain at a horizontal resolution of 5^o x 5^o. It ranges from 0.0 to 3000.0 mm/hr.

shallowIsoRainDev1 (4-byte float, array size: nlat x nlon):
Standard deviation of non-zero shallow rain at a horizontal resolution of 5^o x 5^o. It ranges from 0.0 to 3000.0 mm/hr.

zmMean1 (4-byte float, array size: nlat x nlon x nh1):
The Zm Mean 1 gives means of measured radar reflectivity at the fixed heights of 2, 4, 6, 10 and 15 km and for path-average over 5^o x 5^o boxes for one month using data from 1C-21. It ranges from 0 to 100 dBZ.

zmDev1 (4-byte float, array size: nlat x nlon x nh1):
The Zm Dev. 1 gives standard deviations of measured radar reflectivity at the fixed heights of 2, 4, 6, 10 and 15 km and for path-average over 5^o x 5^o boxes for one month using data from 1C-21. It ranges from 0 to 100 dBZ.

convZmMean1 (4-byte float, array size: nlat x nlon x nh1):
Conv. Zm Mean 1 gives the monthly means of measured radar reflectivity for convective rain at a horizontal resolution of 5^o x 5^o. The path-averaged mean and means at the fixed heights of 2, 4, 6, 10 and 15 km are calculated using data from 1C-21. It ranges from 0 to 100 dBZ.

convZmDev1 (4-byte float, array size: nlat x nlon x nh1):
Conv. Zm Dev. 1 gives the monthly standard deviations of measured radar reflectivity for convective rain at a horizontal resolution of 5^o x 5^o. The path-averaged standard deviation and those at the fixed heights of 2, 4, 6, 10 and 15 km are calculated using data from 1C-21. It ranges from 0 to 100 dBZ.

stratZmMean1 (4-byte float, array size: nlat x nlon x nh1):
Strat. Zm Mean 1 gives the monthly means of measured radar reflectivity for stratiform rain at a horizontal resolution of 5^o x 5^o. The path-averaged mean and means at the fixed heights of 2, 4, 6, 10 and 15 km are calculated using data from 1C-21. It ranges from 0 to 100 dBZ.

stratZmDev1 (4-byte float, array size: nlat x nlon x nh1):
Strat. Zm Dev. 1 gives the monthly standard deviations of measured radar reflectivity for stratiform rain at a horizontal resolution of 5^o x 5^o. The path-averaged standard deviation and those at the fixed heights of 2, 4, 6, 10 and 15 km are calculated using data from 1C-21. It ranges from 0 to 100 dBZ.

ztMean1 (4-byte float, array size: nlat x nlon x nh1):
The Zt Mean 1 gives means of corrected radar reflectivity factors at the fixed heights of 2, 4, 6, 10 and 15 km and for path-average over 5^o x 5^o boxes for one month using data from 2A-25. It ranges from 0.1 to 80 dBZ.

ztDev1 (4-byte float, array size: nlat x nlon x nh1):
The Zt Dev. 1 gives standard deviations of corrected radar reflectivity factors at the fixed

heights of 2, 4, 6, 10 and 15 km and for path-average over 5^o x 5^o boxes for one month using data from 2A-25. It ranges from 0.0 to 80 dBZ.

convZtMean1 (4-byte float, array size: nlat x nlon x nh1):
Conv. Zt Mean 1 gives the monthly means of corrected radar reflectivity for convective rain at a horizontal resolution of 5^o x 5^o. The path-averaged mean and means at the fixed heights of 2, 4, 6, 10 and 15 km are calculated using data from 2A-25. It ranges from 0.1 to 80 dBZ.

convZtDev1 (4-byte float, array size: nlat x nlon x nh1):
Conv. Zt Dev. 1 gives the monthly standard deviations of corrected radar reflectivity for convective rain at a horizontal resolution of 5 x 5. The path-averaged standard deviation and those at the fixed heights of 2, 4, 6, 10 and 15 km are calculated using data from 2A-25. It ranges from 0.0 to 80 dBZ.

stratZtMean1 (4-byte float, array size: nlat x nlon x nh1):
Strat. Zt Mean 1 gives the monthly means of measured radar reflectivity for stratiform rain at a horizontal resolution of 5 x 5. The path-averaged mean and means at the fixed heights of 2, 4, 6, 10 and 15 km are calculated using data from 2A-25. It ranges from 0.1 to 80 dBZ.

stratZtDev1 (4-byte float, array size: nlat x nlon x nh1):
Strat. Zt Dev. 1 gives the monthly standard deviations of corrected radar reflectivity for stratiform rain at a horizontal resolution of 5^o x 5^o. The path-averaged standard deviation and those at the fixed heights of 2, 4, 6, 10 and 15 km are calculated using data from 2A-25. It ranges from 0.0 to 80.0 dBZ.

piaSrtMean (4-byte float, array size: nlat x nlon x nang):
PIA srt Mean gives the monthly means of SRT (surface reference technique) path-integrated attenuation calculated at four fixed incidence angles. It has a horizontal resolution of 5 x 5. It has units of dB and a range from 0 dB to 100 dB.

piaSrtDev (4-byte float, array size: nlat x nlon x nang):
PIA srt Dev. gives the monthly standard deviation of SRT path-integrated attenuation calculated at four fixed incidence angles. It has a horizontal resolution of 5 x 5. It has units of dB and a range from 0 dB to 100 dB.

piaHbMean (4-byte float, array size: nlat x nlon x nang):
PIA hb Mean gives the monthly means of HB path-integrated attenuation calculated at four fixed incidence angles. It has a horizontal resolution of 5 x 5. It has units of dB and a range from 0 dB to 100 dB.

piaHbDev (4-byte float, array size: nlat x nlon x nang):
PIA hb Dev. gives the monthly standard deviation of HB path-integrated attenuation calculated at four fixed incidence angles. It has a horizontal resolution of 5 x 5. It has units of and a range from 0 dB to 100 dB.

pia0Mean (4-byte float, array size: nlat x nlon x nang):
PIA 0th Mean gives the monthly means of the 0th-order path-integrated attenuation

calculated at four fixed incidence angles. It has a horizontal resolution of 5 x 5. It has units of and a range from 0 dB to 100 dB.

pia0Dev (4-byte float, array size: nlat x nlon x nang):
PIA 0th Dev. gives the monthly standard deviation of the 0th-order path-integrated attenuation calculated at four fixed incidence angles. It has a horizontal resolution of 5 x 5. It has units of dB and a range from 0 dB to 100 dB.

pia2a25Mean (4-byte float, array size: nlat x nlon x nang):
pia2a25Mean gives the monthly means of 2A25 path-integrated attenuation calculated at four fixed incidence angles. It has a horizontal resolution of 5 x 5. It has units of dB and a range from 0 dB to 100 dB.

pia2a25Dev (4-byte float, array size: nlat x nlon x nang):
pia2a25Dev. gives the monthly standard deviation of 2A25 path-integrated attenuation calculated at four fixed incidence angles. It has a horizontal resolution of 5 x 5. It has units of and a range from 0 dB to 100 dB.

piaSrtssMean (4-byte float, array size: nlat x nlon x nang):
Mean of PIA (path integrated attenuation, one-way) for SRT for a sub-set of data where the 2A25 method flag has been set (see 2A25/3A25 algorithm users guide). Over 5 x 5 deg. boxes. Units are dB and it ranges from 0 to 100dB.

piaSrtssDev (4-byte float, array size: nlat x nlon x nang):
Standard deviation of PIA (path integrated attenuation, one-way) for SRT for a sub-set of data where the 2A25 method flag has been set (see 2A25/3A25 algorithm users guide). Over 5 x 5 deg. boxes. Units are dB and it ranges from 0 to 100dB.

piaHbssMean (4-byte float, array size: nlat x nlon x nang):
Mean of PIA (path integrated attenuation, one-way) for HB method for a sub-set of data where the 2A25 method flag has been set (see 2A25/3A25 algorithm users guide). Over 5 x 5 deg. boxes. Units are dB and it ranges from 0 to 100dB.

piaHbssDev (4-byte float, array size: nlat x nlon x nang):
Standard deviation of PIA (path integrated attenuation, one-way) for HB method for a sub-set of data where the 2A25 method flag has been set (see 2A25/3A25 algorithm users guide). Over 5 x 5 deg. boxes. Units are dB and it ranges from 0 to 100dB.

pia0ssMean (4-byte float, array size: nlat x nlon x nang):
Mean of PIA (path integrated attenuation, one-way) for 0th-order method for a sub-set of data where the 2A25 method flag has been set (see 2A25/3A25 algorithm users guide). Over 5 x 5 deg. boxes. Units are dB and it ranges from 0 to 100dB.

pia0ssDev (4-byte float, array size: nlat x nlon x nang):
Standard deviation of PIA (path integrated attenuation, one-way) for 0th-order method for a sub- set of data where the 2A25 method flag has been set (see 2A25/3A25 algorithm users guide). Over 5 x 5 deg. boxes. Units are dB and it ranges from 0 to 100dB.

pia2a25ssMean (4-byte float, array size: nlat x nlon x nang):
Mean of final PIA (path integrated attenuation, one-way) from 2A25 for a sub-set of data

where the 2A25 method flag has been set (see 2A25/3A25 algorithm users guide). Over 5 x 5 deg. boxes. Units are dB and it ranges from 0 to 100dB.

pia2a25ssDev (4-byte float, array size: nlat x nlon x nang):
Standard deviation of final PIA (path integrated attenuation, one-way) from 2A25 for a sub-set of data where the 2A25 method flag has been set (see 2A25/3A25 algorithm users guide). Over 5 x 5 deg. boxes. Units are dB and it ranges from 0 to 100dB.

stormHtMean (4-byte float, array size: nlat x nlon x 3):
Storm Height Mean is the mean of the storm height for conditions of stratiform rain, convective rain and unconditional rain. It has units of meters and ranges from 0.0 to 20,000.

stormHtDev (4-byte float, array size: nlat x nlon x 3):
Storm Height Dev. is the standard deviation of the storm height for conditions of stratiform rain, convective rain and unconditional rain. It has units of meters and ranges from 0.0 to 20,000.

xiMean (4-byte float, array size: nlat x nlon):
Xi Mean gives the monthly means of the horizontal non-uniformity parameter of the rain field within a ray at a horizontal resolution of 5 x 5. It has no units and ranges from 0.0 to 99.0.

xiDev (4-byte float, array size: nlat x nlon):
Xi Dev. gives the monthly standard deviation of the horizontal non-uniformity parameter of the rain field within a ray at a horizontal resolution of 5 x 5. It has no units and ranges from 0.0 to 99.0.

nubfCorFacMean (4-byte float, array size: nlat x nlon):
The NUBF (Non-Uniform Beam Filling) Correction Factor Mean gives the monthly mean of NUBF correction for Z-factor and Rain Rate at a horizontal resolution of 5 x 5. It has no units and a range of 0 to 2.0.

nubfCorFacDev (4-byte float, array size: nlat x nlon):
The NUBF (Non-Uniform Beam Filling) Correction Factor Dev. gives the monthly standard deviation of the NUBF correction for Z-factor and Rain Rate at a horizontal resolution of 5 x 5. It has no units and ranges from 0 to 2.0.

bbHtMean (4-byte float, array size: nlat x nlon):
BB Height Mean gives the monthly means of the bright band height at a horizontal resolution of 5 x 5. It has units of meters and ranges from 0 to 20,000.

bbHtDev (4-byte float, array size: nlat x nlon):
BB Height Dev. gives the monthly deviation of the bright band height at a horizontal resolution of 5 x 5. It has units of meters and ranges from 0 to 20,000.

epsilon0StratMean1 (4-byte float, array size: nlat x nlon):
Mean of epsilon conditioned on stratiform rain and use of 2A21 SRT at a horizontal resolution of 5 x 5deg. It ranges from 0.0 to 5.0 (unitless).

epsilon0StratDev1 (4-byte float, array size: nlat x nlon):
Standard deviation of epsilon0 conditioned on stratiform rain and use of 2A21 SRT at a horizontal resolution of 5 x 5deg. It ranges from 0.0 to 5.0 (unitless).

epsilon0ConvMean1 (4-byte float, array size: nlat x nlon):
Mean of epsilon0 conditioned on convective rain and use of 2A21 SRT at a horizontal resolution of 5 x 5deg. It ranges from 0.0 to 5.0 (unitless).

epsilon0ConvDev1 (4-byte float, array size: nlat x nlon):
Standard deviation of epsilon0 conditioned on convective rain and use of 2A21 SRT at a horizontal resolution of 5 x 5deg. It ranges from 0.0 to 5.0 (unitless).

epsilonStratMean1 (4-byte float, array size: nlat x nlon):
Mean of epsilon conditioned stratiform rain and use of 2A21 SRT at a horizontal resolution of 5 x 5deg. It ranges from 0.0 to 5.0 (unitless).

epsilonStratDev1 (4-byte float, array size: nlat x nlon):
Mean of epsilon conditioned stratiform rain and use of 2A21 SRT at a horizontal resolution of 5 x 5deg. It ranges from 0.0 to 5.0 (unitless).

epsilonConvMean1 (4-byte float, array size: nlat x nlon):
Mean of epsilon conditioned on convective rain and use of 2A21 SRT at a horizontal resolution of 5 x 5deg. It ranges from 0.0 to 5.0 (unitless).

epsilonConvDev1 (4-byte float, array size: nlat x nlon):
Standard deviation of epsilon conditioned on convective rain and use of 2A21 SRT at a horizontal resolution of 5 x 5deg. It ranges from 0.0 to 5.0 (unitless).

surfRainMean1 (4-byte float, array size: nlat x nlon):
Mean of non-zero near-surface rain rate at a horizontal resolution of 5 x 5. It ranges from 0.0 to 3000.0 mm/hr.

surfRainDev1 (4-byte float, array size: nlat x nlon):
Standard deviation of non-zero near-surface rain rate at a horizontal resolution of 5 x 5. It ranges from 0.0 to 3000.0 mm/hr.

e_surfRainMean1 (4-byte float, array size: nlat x nlon):
Mean of non-zero estimated surface rain below clutter (See 2A25 algorithm user guide) in mm/hr. Over 5 x 5 deg. boxes. It ranges from 0.0 to 400.0 mm/hr.

e_surfRainDev1 (4-byte float, array size: nlat x nlon):
Standard deviation of non-zero estimated surface rain below clutter (See 2A25 algorithm user guide) in mm/hr. Over 5 x 5 deg. boxes. It ranges from 0.0 to 400.0 mm/hr.

bbZmaxMean1 (4-byte float, array size: nlat x nlon):
Mean of maximum reflectivity in bright band at a horizontal resolution of 5 x 5. It ranges from 0.0 to 100.0 dBZ.

bbZmaxDev1 (4-byte float, array size: nlat x nlon):
Standard Deviation of maximum reflectivity in bright band at a horizontal resolution of 5 x 5. It ranges from 0.0 to 100.0 dBZ.

bbNadirHtMean1 (4-byte float, array size: nlat x nlon):
Height of bright band from nadir ray in meters for 5 x 5 deg. boxes. It ranges from 0 to 20,000 meters.

bbNadirHtDev1 (4-byte float, array size: nlat x nlon):
Standard deviation of the bright band height from nadir ray in meters for 5 x 5 deg. boxes. It ranges from 0 to 20,000 meters.

bbNadirWidthMean1 (4-byte float, array size: nlat x nlon):
Width of bright band from nadir ray in meters for 5 x 5 deg. boxes. It ranges from 0 to 10,000 meters.

bbNadirWidthDev1 (4-byte float, array size: nlat x nlon):
Standard deviation of the width of the bright band from nadir ray in meters for 5 x 5 deg. boxes. It ranges from 0 to 10,000 meters.

bbNadirZmaxMean1 (4-byte float, array size: nlat x nlon):
Mean of maximum Z in bright band from the nadir ray in dBZ for 5 x 5 deg. boxes. It ranges from 0 to 70dBZ.

bbNadirZmaxDev1 (4-byte float, array size: nlat x nlon):
Standard deviation of maximum Z in bright band from the nadir ray in dBZ for 5 x 5 deg. boxes. It ranges from 0 to 70dBZ.

sdepthMean1 (4-byte float, array size: nlat x nlon):
Mean of snow depth at a horizontal resolution of 5 x 5. It ranges from 0.0 to 20,000.0 m.

sdepthDev1 (4-byte float, array size: nlat x nlon):
Standard deviation of snow depth at a horizontal resolution of 5 x 5. It ranges from 0.0 to 20,000.0 m.

surfRainConvMean1 (4-byte float, array size: nlat x nlon):
Mean of non-zero near-surface rain rate conditioned on convective rain in 5 x 5 deg. boxes. It ranges from 0.0 to 3000.0 mm/hr.

surfRainConvDev1 (4-byte float, array size: nlat x nlon):
Standard deviation of non-zero near-surface rain rate conditioned on convective rain in 5 x 5 deg. boxes. It ranges from 0.0 to 3000.0 mm/hr.

surfRainStratMean1 (4-byte float, array size: nlat x nlon):
Mean of non-zero near-surface rain rate conditioned on stratiform rain in 5 x 5 deg. boxes. It ranges from 0.0 to 3000.0 mm/hr.

surfRainStratDev1 (4-byte float, array size: nlat x nlon):
Standard deviation of non-zero near-surface rain rate conditioned on convective rain in 5 x 5 deg. boxes. It ranges from 0.0 to 3000.0 mm/hr.

e_surfRainConvMean1 (4-byte float, array size: nlat x nlon):
Mean of non-zero estimated surface rain below clutter (See 2A25 algorithm user guide) conditioned on convective rain in mm/hr. Over 5 x 5 deg. boxes. It ranges from 0.0 to 400.0 mm/hr.

e_surfRainConvDev1 (4-byte float, array size: nlat x nlon):
Standard deviation of non-zero estimated surface rain below clutter (See 2A25 algorithm user guide) conditioned on convective rain in mm/hr. It ranges from 0.0 to 400.0 mm/hr.

e_surfRainStratMean1 (4-byte float, array size: nlat x nlon):
Mean of non-zero estimated surface rain below clutter (See 2A25 algorithm user guide) conditioned on stratiform rain in mm/hr. Over 5 x 5 deg. boxes. It ranges from 0.0 to 400.0 mm/hr.

e_surfRainStratDev1 (4-byte float, array size: nlat x nlon):
Standard deviation of non-zero estimated surface rain below clutter (See 2A25 algorithm user guide) conditioned on stratiform rain in mm/hr. Over 5 x 5 deg. boxes. It ranges from 0.0 to 400.0 mm/hr.

rzStratA1 (4-byte float, array size: nlat x nlon x 2):
The A parameter in rainfall-reflectivity relation $R = AZ^B$ from fitting of instantaneous R,Z pairs conditioned on stratiform rain. Computed for near-surface and 2km heights. It ranges from 0 to 1.0.

rzStratB1 (4-byte float, array size: nlat x nlon x 2):
The B parameter in rainfall-reflectivity relation $R = AZ^B$ from fitting of instantaneous R,Z pairs conditioned on stratiform rain. Computed for near-surface and 2km heights. Over 5 x 5 deg. boxes. It ranges from 0 to 1.0.

rzConvA1 (4-byte float, array size: nlat x nlon x 2):
The A parameter in rainfall-reflectivity relation $R = AZ^B$ from fitting of instantaneous R,Z pairs conditioned on convective rain. Computed for near-surface and 2km heights. Over 5 x 5 deg. boxes. It ranges from 0 to 1.0.

rzConvB1 (4-byte float, array size: nlat x nlon x 2):
The B parameter in rainfall-reflectivity relation $R = AZ^B$ from fitting of instantaneous R,Z pairs conditioned on convective rain. Computed for near-surface and 2km heights. Over 5 x 5 deg. boxes. It ranges from 0 to 1.0.

rzA1 (4-byte float, array size: nlat x nlon x 2):
The A parameter in rainfall-reflectivity relation $R = AZ^B$ from fitting of instantaneous R,Z pairs. Computed for near-surface and 2km heights. Over 5 x 5 deg. boxes. It ranges from 0 to 1.0.

rzB1 (4-byte float, array size: nlat x nlon x 2):
The B parameter in rainfall-reflectivity relation $R = AZ^B$ from fitting of instantaneous R,Z pairs. Computed for near-surface and 2km heights. Over 5 x 5 deg. boxes. It ranges from 0 to 1.0.

ttlPix1 (4-byte integer, array size: nlat x nlon):
The Total Pixel Number 1 is the number of total pixels over 5 x 5 boxes for one month. The range is 0 to 2,000,000.

bbPixNum1 (4-byte integer, array size: nlat x nlon):
The number of bright band counts over each 5 x 5 box for one month. The range is 0 to

2,000,000.

bbNadirPix1 (4-byte integer, array size: nlat x nlon):
The number of bright band nadir pixel counts. Over 5 x 5 deg. boxes. The range is 0 to 2,000,000.

surfRainPix1 (4-byte integer, array size: nlat x nlon):
Near-surface rain counts at a horizontal resolution of 5 x 5. It ranges from 0 to 2,000,000.

surfRainStratPix1 (4-byte integer, array size: nlat x nlon):
Counts of non-zero near-surface rainfall conditioned on convective rain in 5 x 5 deg. boxes. Ranges from 0 to 32,767.

surfRainConvPix1 (4-byte integer, array size: nlat x nlon):
Counts of non-zero near-surface rain fall conditioned on convective rain in 5 x 5 deg. boxes. Ranges from 0 to 32,767.

e_surfRainPix1 (4-byte integer, array size: nlat x nlon):
The number of non-zero estimated surface rain pixel counts. Over 5 x 5 deg. boxes. The range is 0 to 2,000,000.

e_surfRainStratPix1 (4-byte integer, array size: nlat x nlon):
The number of non-zero estimated surface rain pixel counts conditioned on stratiform. Over 5 x 5 deg. boxes. The range is 0 to 2,000,000.

e_surfRainConvPix1 (4-byte integer, array size: nlat x nlon):
The number of non-zero estimated surface rain pixel counts conditioned on convective rain. Over 5 x 5 deg. boxes. The range is 0 to 2,000,000.

epsilon0StratPix1 (4-byte integer, array size: nlat x nlon):
Counts of epsilon0 conditioned on stratiform rain and use of 2A21 SRT at a horizontal resolution of 5 x 5deg. It ranges from 0 to 2,000,000.

epsilon0ConvPix1 (4-byte integer, array size: nlat x nlon):
Counts of epsilon0 conditioned on convective rain and use of 2A21 SRT at a horizontal resolution of 5 x 5deg. It ranges from 0 to 2,000,000.

epsilonStratPix1 (4-byte integer, array size: nlat x nlon):
Counts of epsilon conditioned on stratiform rain and use of 2A21 SRT at a horizontal resolution of 5 x 5deg. It ranges from 0 to 2,000,000.

epsilonConvPix1 (4-byte integer, array size: nlat x nlon):
Counts of epsilon conditioned on convective rain and use of 2A21 SRT at a horizontal resolution of 5 x 5deg. It ranges from 0 to 2,000,000.

rainPix1 (4-byte integer, array size: nlat x nlon x nh1):
The Rain Pixel Number 1 is the number of non-zero rain rate pixels at the fixed heights of 2, 4, 6, 10 and 15 km and for path-average over 5 x 5 boxes for one month. The range is 0 to 2,000,000.

convRainPix1 (4-byte integer, array size: nlat x nlon x nh1):
The Convective Rain Pixel Number 1 is the number of non-zero rain rate pixels for

convective rain at the fixed heights of 2, 4, 6, 10 and 15 km and for path-average over 5 x 5 boxes for one month. The range is 0 to 2,000,000.

stratRainPix1 (4-byte integer, array size: nlat x nlon x nh1):
The Stratiform Rain Pixel Number 1 is the number of non-zero rain rate pixels for stratiform rain at the fixed heights of 2, 4, 6, 10 and 15 km and for path-average over 5 x 5 boxes for one month. The range is 0 to 2,000,000.

ttlAnglePix1 (2-byte integer, array size: nlat x nlon x 4):
Total Angle Pixel Number 1 is the total number of pixels over each 5 x 5 latitude-longitude grid box for a month. This parameter is accumulated at four different angles (i.e., 0, 5, 10, and 15). The range is 0 to 30,000.

rainAnglePix1 (2-byte integer, array size: nlat x nlon x 4):
Rain Angle Pixel Number 1 is the total number of non-zero rain rate pixels over each 5 x 5 latitude-longitude grid box for a month. This parameter is accumulated at four different angles (i.e., 0, 5, 10, and 15). The range is 0 to 30,000.

rzStratPix1 (4-byte integer, array size: nlat x nlon x 2):
The number of R-Z coefficient pixel counts for stratiform rain for near-surface and 2km heights. Over 5 x 5 deg. boxes. The range is 0 to 2,000,000.

rzConvPix1 (4-byte integer, array size: nlat x nlon x 2):
The number of R-Z coefficient pixel counts for convective rain for near-surface and 2km heights. Over 5 x 5 deg. boxes. The range is 0 to 2,000,000.

rzPix1 (4-byte integer, array size: nlat x nlon x 2):
The number of R-Z coefficient pixel counts for near-surface and 2km heights. Over 5 x 5 deg. boxes. The range is 0 to 2,000,000.

shallowRainPix1 (4-byte integer, array size: nlat x nlon):
Counts of non-zero shallow rain over 5 x 5 deg. boxes. The range is 0 to 2,000,000.

shallowIsoRainPix1 (4-byte integer, array size: nlat x nlon):
Counts of non-zero isolated shallow rain over 5 x 5 deg. boxes. The range is 0 to 2,000,000.

stormHH (2-byte integer, array size: nlat x nlon x ncat2):
These are histograms of the 'effective' storm heights for 30 categories over a 5 x 5 box for one month. It ranges from 0 to 32,767.

convStormHH (2-byte integer, array size: nlat x nlon x ncat2):
These are histograms of the 'effective' storm heights for convective rain for 30 categories over a 5 x 5 box for one month. It ranges from 0 to 32,767.

stratStormHH (2-byte integer, array size: nlat x nlon x ncat2):
These are histograms of the 'effective' storm heights for stratiform rain for 30 categories over a 5 x 5 box for one month. It ranges from 0 to 32,767.

BBHH (2-byte integer, array size: nlat x nlon x ncat2):
These are histograms of the bright-band heights for 30 categories over a 5 x 5 box for one month, given that the bright band is detected. It ranges from 0 to 32,767.

bbNadirWidthH (2-byte integer, array size: nlat x nlon x ncat2):
Histogram in counts of bright band widths from nadir ray for 5 x 5 deg. boxes. There are 30 categories. It ranges from 0 to 32,767.

bbNadirHH (2-byte integer, array size: nlat x nlon x ncat2):
Histogram in counts of bright band heights from nadir ray for 5 x 5 deg. boxes. There are 30 categories. It ranges from 0 to 32,767.

bbNadirZmaxH (2-byte integer, array size: nlat x nlon x ncat2):
Histogram in counts of maximum Z in bright band from nadir ray for 5 x 5 deg. boxes. There are 30 categories. It ranges from 0 to 32,767.

snowIceLH (2-byte integer, array size: nlat x nlon x ncat2):
These are histograms of the depth of snow-ice layer for 30 categories over a 5 x 5 box for one month. The depth of snow-ice layer is defined as the difference between effective storm height and estimated height of 0^{o}C isotherm. It ranges from 0 to 32,767.

bbZmaxH (2-byte integer, array size: nlat x nlon x ncat2):
Histogram of maximum Zt in bright band at a horizontal resolution of 5 x 5. It ranges from 0 to 32,000.

epsilon0StratH (2-byte integer, array size: nlat x nlon x ncat2):
Histogram of epsilon0 conditioned stratiform rain and use of 2A21 SRT at a horizontal resolution of 5 x 5 deg. It ranges from 0 to 32,000.

epsilon0ConvH (2-byte integer, array size: nlat x nlon x ncat2):
Histogram of epsilon0 conditioned convective rain and use of 2A21 SRT at a horizontal resolution of 5 x 5 deg. It ranges from 0 to 32,000.

epsilonStratH (2-byte integer, array size: nlat x nlon x ncat2):
Histogram of epsilon conditioned stratiform rain and use of 2A21 SRT at a horizontal resolution of 5 x 5 deg. It ranges from 0 to 32,000.

epsilonConvH (2-byte integer, array size: nlat x nlon x ncat2):
Histogram of epsilon conditioned convective rain and use of 2A21 SRT at a horizontal resolution of 5 x 5 deg. It ranges from 0 to 32,000.

surfRainH (2-byte integer, array size: nlat x nlon x ncat2):
Histogram of near-surface rain rate at a horizontal resolution of 5 x 5. It ranges from 0 to 32,000.

e_surfRainH (2-byte integer, array size: nlat x nlon x ncat2):
Histogram in counts of non-zero estimated surface rain for 5 x 5 deg. boxes. There are 30 categories. It ranges from 0 to 32,767.

surfRainConvH (2-byte integer, array size: nlat x nlon x ncat2):
Histogram in counts of non-zero near-surface rainfall conditioned on convective rain in 5 x 5 deg. boxes. Binned into 20 categories. Ranges from 0 to 32,767.

surfRainStratH (2-byte integer, array size: nlat x nlon x ncat2):
Histogram in counts of non-zero near-surface rainfall conditioned on stratiform rain in 5 x 5 deg. boxes. Binned into 20 categories. Ranges from 0 to 32,767.

e_surfRainConvH (2-byte integer, array size: nlat x nlon x ncat2):
Histogram in counts of non-zero estimated surface rain conditioned on convective rain for 5 x 5 deg. boxes. There are 30 categories. It ranges from 0 to 32,767.

e_surfRainStratH (2-byte integer, array size: nlat x nlon x ncat2):
Histogram in counts of non-zero estimated surface rain conditioned on stratiform rain for 5 x 5 deg. boxes. There are 30 categories. It ranges from 0 to 32,767.

zmH (2-byte integer, array size: nlat x nlon x ncat2 x nh1):
The Zm Histograms are histograms of measured reflectivities of rain pixels at five heights (2, 4, 6, 10 and 15 km) and path-average for 20 categories over a 5 x 5 box for one month. It ranges from 0 to 32,767.

convZmH (2-byte integer, array size: nlat x nlon x ncat2 x nh1):
The Convective Zm Histograms are histograms of measured reflectivities of convective rain pixels at five heights (2, 4, 6, 10 and 15 km) and path-average for 20 categories over a 5^o x 5^o box for one month. It ranges from 0 to 32,767.

stratZmH (2-byte integer, array size: nlat x nlon x ncat2 x nh1):
The Stratiform Zm Histograms are histograms of measured reflectivities of stratiform rain pixels at five heights (2, 4, 6, 10 and 15 km) and path-average for 20 categories over a 5^o x 5^o box for one month. It ranges from 0 to 32,767.

ztH (2-byte integer, array size: nlat x nlon x ncat2 x nh1):
The Zt Histograms are histograms of corrected reflectivity factors for rain pixels at five heights (2, 4, 6, 10 and 15 km) and path-average for 20 categories over a 5^o x 5^o box for one month. It ranges from 0 to 32,767.

convZtH (2-byte integer, array size: nlat x nlon x ncat2 x nh1):
The Convective Zt Histograms are histograms of corrected reflectivity factors for convective rain pixels at five heights (2, 4, 6, 10 and 15 km) and path-average for 20 categories over a 5^o x 5^o box for one month. It ranges from 0 to 32,767.

stratZtH (2-byte integer, array size: nlat x nlon x ncat2 x nh1):
The Stratiform Zt Histograms are histograms of corrected reflectivity factors for stratiform rain pixels at five heights (2, 4, 6, 10 and 15 km) and path-average for 20 categories over a 5^o x 5^o box for one month. It ranges from 0 to 32,767.

rainH (2-byte integer, array size: nlat x nlon x ncat2 x nh1):
These are histograms of non-zero rain rate pixels at five heights (2, 4, 6, 10 and 15 km) and path-average for 20 categories over a 5^o x 5^o box for one month. It ranges from 0 to 32,767.

convRainH (2-byte integer, array size: nlat x nlon x ncat2 x nh1):
These are histograms of non-zero rain rate pixels for convective rain at five heights (2, 4, 6, 10 and 15 km) and path-average for 20 categories over a 5^o x 5^o box for one month. It ranges from 0 to 32,767

stratRainH (2-byte integer, array size: nlat x nlon x ncat2 x nh1):
These are histograms of non-zero rain rate pixels for stratiform rain at five heights (2, 4,

6, 10 and 15 km) and path-average for 20 categories over a 5^o x 5^o box for one month. It ranges from 0 to 32,767

shallowRainH (2-byte integer, array size: nlat x nlon x ncat2):
Histogram of non-zero shallow rain at a horizontal resolution of 5 x 5 deg. It ranges from 0 to 32,000.

shallowIsoRainH (2-byte integer, array size: nlat x nlon x ncat2):
Histogram of non-zero isolated shallow rain at a horizontal resolution of 5 x 5 deg. It ranges from 0 to 32,000.

piaSrtH (2-byte integer, array size: nlat x nlon x ncat2 x nang):
PIA srt Hist. gives histograms of path-attenuation as determined by the surface reference technique (SRT) at 4 incidence angles (0^o, 5^o, 10^o and 15^o) for 30 categories over a 5^o x 5^o box for one month. It ranges from 0 to 32,767.

piaHbH (2-byte integer, array size: nlat x nlon x ncat2 x nang):
These are histograms of path-attenuation using an estimate derived from measured reflectivity (Zm) and a k-Z relationship at 4 incidence angles (0^o, 5^o, 10^o and 15^o) for 30 categories over a 5^o x 5^o box for one month. It ranges from 0 to 32,767.

pia0H (2-byte integer, array size: nlat x nlon x ncat2 x nang):
PIA 0th Hist. is the histogram of the 0th order path-integrated attenuation with a horizontal resolution of 5^o x 5^o. This histogram is calculated for 30 categories at 4 different incident angles (0^o, 5^o, 10^o and 15^o). It ranges from 0 to 32,767

pia2a25H (2-byte integer, array size: nlat x nlon x ncat2 x nang):
These are histograms of path-attenuation as determined by 2A25 at 4 incidence angles (0^o, 5^o, 10^o and 15^o) for 30 categories over a 5^o x 5^o box for one month. It ranges from 0 to 32,767.

piaSrtssH (2-byte integer, array size: nlat x nlon x ncat2 x nang):
Histogram in counts of PIA from SRT subsetted by 2A25 method flag at 5 angles (0^o, 5^o, 10^o, 15^o and all 49 angle bins) for 30 categories over 5^o x 5^o deg. boxes. It ranges from 0 to 32,767.

piaHbssH (2-byte integer, array size: nlat x nlon x ncat2 x nang):
Histogram in counts of PIA from HB method subsetted by 2A25 method flag at 5 angles (0^o, 5^o, 10^o, 15^o and all 49 angle bins) for 30 categories over 5^o x 5^o deg. boxes. It ranges from 0 to 32,767.

pia0ssH (2-byte integer, array size: nlat x nlon x ncat2 x nang):
Histogram in counts of PIA from 0th-order method subsetted by 2A25 method flag at 5 angles (0^o, 5^o, 10^o, 15^o and all 49 angle bins) for 30 categories over 5^o x 5^o deg. boxes. It ranges from 0 to 32,767.

pia2a25ssH (2-byte integer, array size: nlat x nlon x ncat2 x nang):
Histogram in counts of final PIA from 2A25 subsetted by 2A25 method flag at 5 angles (0^o, 5^o, 10^o, 15^o and all 49 angle bins) for 30 categories over 5^o x 5^o deg. boxes. It ranges from 0 to 32,767.

xiH (2-byte integer, array size: nlat x nlon x ncat2):
The Xi Histograms is the histogram of non-uniformity parameter determined in 2A-25 for 30 categories over a 5 x 5 box for one month. It ranges from 0 to 32,767.

nubfH (2-byte integer, array size: nlat x nlon x ncat2):
NUBF (Non-Uniform Beam Filling) Hist. gives the histogram of the NUBF correction for Z-factor and rain rate of 30 different categories over 5o x 5o grid boxes. It ranges from 0 to 32,767.

rainCCoef (4-byte float, array size: nlat x nlon x 3):
These are correlation coefficients of non-zero rain rates between 3 heights (i.e., correlation coefficient of rain rates at 2 km vs 4 km, 2 km vs 6 km, and 4 km vs 6 km) for a 5 x 5 box for one month.. It ranges from -1.000 to 1.000.

convRainCCoef (4-byte float, array size: nlat x nlon x 3):
These are correlation coefficients of non-zero rain rates for convective rain between 3 heights (i.e., correlation coefficient of rain rates at 2 km vs 4 km, 2 km vs 6 km, and 4 km vs 6 km) for a 5 x 5 box for one month. It ranges from -1.000 to 1.000.

stratRainCCoef (4-byte float, array size: nlat x nlon x 3):
These are correlation coefficients of non-zero rain rates for stratiform rain between 3 heights (i.e., correlation coefficient of rain rates at 2 km vs 4 km, 2 km vs 6 km, and 4 km vs 6 km) for a 5 x 5 box for one month. It ranges from -1.000 to 1.000.

piaCCoef (4-byte float, array size: nlat x nlon x nang x 6):
This is the correlation coefficient of three path-integrated attenuations (SRT, HB, and 0th order PIAs) at angles of 0, 5, 10 and 15 for a 5 x 5 box for one month. It ranges from -1.000 to 1.000.

piaSrtPix (4-byte float, array size: nlat x nlon x 5):
Counts of PIA using the SRT method. Over 5 x 5 deg. boxes, calculated at four incidence angles and for all angles. It ranges from 0 to 32,767.

piaHbPix (4-byte float, array size: nlat x nlon x 5):
Counts of PIA using the HB method. Over 5 x 5 deg. boxes, calculated at four incidence angles and for all angles. It ranges from 0 to 32,767.

pia0Pix (4-byte float, array size: nlat x nlon x 5):
Counts of PIA using the 0th order method. Over 5 x 5 deg. boxes, calculated at four incidence angles and for all angles. It ranges from 0 to 32,767.

pia2a25Pix (4-byte float, array size: nlat x nlon x 5):
Counts of PIA from 2A25. Over 5 x 5 deg. boxes, calculated at four incidence angles and for all angles. It ranges from 0 to 32,767.

piaSrtssPix (4-byte float, array size: nlat x nlon x nang):
Counts of PIA using SRT method for a sub-set of data where the 2A25 method flag has been set (see 2A25/3A25 algorithm users guide). Over 5 x 5 deg. boxes. It ranges from 0 to 32,767.

piaHbssPix (4-byte float, array size: nlat x nlon x nang):
Counts of PIA using HB method for a sub-set of data where the 2A25 method flag has been set (see 2A25/3A25 algorithm users guide). Over 5 x 5 deg. boxes. It ranges from 0 to 32,767.

pia0ssPix (4-byte float, array size: nlat x nlon x nang):
Counts of PIA using 0th-order method for a sub-set of data where the 2A25 method flag has been set (see 2A25/3A25 algorithm users guide). Over 5 x 5 deg. boxes. It ranges from 0 to 32,767.

pia2a25ssPix (4-byte float, array size: nlat x nlon x nang):
Counts of final PIA from 2A25 for a sub-set of data where the 2A25 method flag has been set (see 2A25/3A25 algorithm users guide). Over 5 x 5 deg. boxes. It ranges from 0 to 32,767.

rainCCoefPix (4-byte float, array size: nlat x nlon x 3):
Counts for correlation coefficients of rain at the 3 heights. Over 5 x 5 deg. boxes. It ranges from 0 to 32,767.

stratCCoefPix (4-byte float, array size: nlat x nlon x 3):
Counts for correlation coefficients of rain conditioned on stratiform rain at the 3 heights. Over 5 x 5 deg. boxes. It ranges from 0 to 32,767.

convCCoefPix (4-byte float, array size: nlat x nlon x 3):
Counts for correlation coefficients of rain conditioned on convective rain at the 3 heights. Over 5 x 5 deg. boxes. It ranges from 0 to 32,767.

piaCCoefPix (4-byte float, array size: nlat x nlon x 5):
Counts for correlation coefficients of PIA for the 5 angle categories (0, 5, 10, 15 degrees and all 49 angle bins). It ranges from 0 to 32,767.

epsilonFit (4-byte float, array size: nlat x nlon):
Estimate of epsilon from a linear fit of zeta from 2A25 and SRT PIA from 2A21. Values range from -10 to 10. Special values are defined as:
-9999.9 Missing value

epsilonFitRMS (4-byte float, array size: nlat x nlon):
RMS uncertainty from the fit of zeta and SRT PIA. Values range from -50 to 50. Special values are defined as:
-9999.9 Missing value

epsilonFitPix (4-byte integer, array size: nlat x nlon):
Number of measurements used in the fit of zeta and SRT PIA. Values range from 0 to 5000. Special values are defined as:
-9999 Missing value

Grid2 (Grid)

GridHeader2 (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for TRMM Products for details.

rainMean2 (4-byte float, array size: nlath x nlonh x nh3):
Rain Rate Mean 2 gives means of non-zero rain rates over 0.5 x 0.5 boxes for one month. The rain rates are determined in 2A-25 and evaluated at the fixed heights of 2 km, 4 km, 6 km, and path average. It ranges from 0 to 3000.0 mm/hr.

rainDev2 (4-byte float, array size: nlath x nlonh x nh3):
Rain Rate Dev. 2 gives standard deviations of non-zero rain rates over 0.5 x 0.5 boxes for one month. The rain rates are determined in 2A-25 and evaluated at the fixed heights of 2 km, 4 km, 6 km, and path average. It ranges from 0 to 3000.0 mm/hr.

convRainMean2 (4-byte float, array size: nlath x nlonh x nh3):
Conv. Rain Rate Mean 2 gives means of non-zero rain rates for convective rain over 0.5 x 0.5 boxes for one month. The rain rates are determined in 2A-25 and evaluated at the fixed heights of 2 km, 4 km, 6 km, and path average. It ranges from 0 to 3000.0 mm/hr.

convRainDev2 (4-byte float, array size: nlath x nlonh x nh3):
Conv. Rain Rate Dev. 2 gives standard deviations of non-zero rain rates for convective rain over 0.5 x 0.5 boxes for one month. The rain rates are determined in 2A-25 and evaluated at the fixed heights of 2 km, 4 km, 6 km, and path average. It ranges from 0 to 3000.0 mm/hr.

stratRainMean2 (4-byte float, array size: nlath x nlonh x nh3):
Strat. Rain Rate Mean 2 gives means of non-zero rain rates for stratiform rain over 0.5 x 0.5 boxes for one month. The rain rates are determined in 2A-25 and evaluated at the fixed heights of 2 km, 4 km, 6 km, and path average. It ranges from 0 to 3000.0 mm/hr.

stratRainDev2 (4-byte float, array size: nlath x nlonh x nh3):
Strat/ Rain Rate Dev. 2 gives standard deviations of non-zero rain rates for stratiform rain over 0.5 x 0.5 boxes for one month. The rain rates are determined in 2A-25 and evaluated at the fixed heights of 2 km, 4 km, 6 km, and path average. It ranges from 0 to 3000.0 mm/hr.

shallowRainMean2 (4-byte float, array size: nlath x nlonh):
Mean of shallow rain at a horizontal resolution of 0.5 x 0.5 deg. It ranges from 0.0 to 3000 mm/hr.

shallowRainDev2 (4-byte float, array size: nlath x nlonh):
Standard deviation of shallow rain at a horizontal resolution of 0.5 x 0.5 deg. It ranges from 0.0 to 3000 mm/hr.

shallowIsoRainMean2 (4-byte float, array size: nlath x nlonh):
Mean of shallow isolated rain at a horizontal resolution of 0.5 x 0.5 deg. It ranges from 0.0 to 3000 mm/hr.

shallowIsoRainDev2 (4-byte float, array size: nlath x nlonh):
Standard deviation of shallow isolated rain at a horizontal resolution of 0.5 x 0.5 deg. It

ranges from 0.0 to 3000 mm/hr.

zmMean2 (4-byte float, array size: nlath x nlonh x nh3):
Zm Mean 2 gives the monthly means of the measured reflectivity at the fixed height levels of 2 km, 4 km, 6 km, and path average over 0.5^o x 0.5^o grid boxes. It ranges from -20 to 80 dBZ.

convZmMean2 (4-byte float, array size: nlath x nlonh x nh3):
Conv. Zm Mean 2 gives the monthly means of the measured reflectivity of convective rain at the fixed height levels of 2 km, 4 km, 6 km, and path average over 0.5^o x 0.5^o grid boxes. It ranges from -20 to 80 dBZ.

stratZmMean2 (4-byte float, array size: nlath x nlonh x nh3):
Strat. Zm Means gives the monthly means of the measured reflectivity of stratiform rain at the fixed heights of 2 km, 4 km, 6 km, and path average over 0.5^o x 0.5^o grid boxes. It ranges from -20 to 80 dBZ.

ztMean2 (4-byte float, array size: nlath x nlonh x nh3):
Zt Mean 2 gives the monthly means of the corrected reflectivity at the fixed heights of 2 km, 4 km, 6 km, and path average over 0.5^o x 0.5^o grid boxes. It ranges from 0.1 to 80 dBZ.

convZtMean2 (4-byte float, array size: nlath x nlonh x nh3):
Conv. Zm Mean 2 gives the monthly means of the corrected reflectivity of convective rain at the fixed height levels of 2 km, 4 km, 6 km, and path average over 0.5^o x 0.5^o grid boxes. It ranges from 0.1 to 80 dBZ.

stratZtMean2 (4-byte float, array size: nlath x nlonh x nh3):
Strat. Zm Means gives the monthly means of the corrected reflectivity of stratiform rain at the fixed heights of 2 km, 4 km, 6 km, and path average over 0.5^o x 0.5^o grid boxes. It ranges from 0.1 to 80 dBZ.

stormHeightMean (4-byte float, array size: nlath x nlonh x 2):
Storm Height Mean gives the monthly means of the storm height for stratiform and convective rain over 0.5^o x 0.5^o grid boxes. It has units of meters and ranges from 0 to 20,000.

stormHeightDev2 (4-byte float, array size: nlath x nlonh x 2):
Standard deviation of storm height, conditioned for stratiform and convective rain at a horizontal resolution of 0.5 x 0.5. It ranges from 0.0 to 20,000.0 m.

bbHeightMean (4-byte float, array size: nlath x nlonh):
BB Height Mean gives the monthly means of bright-band height over grid boxes of 0.5^o x 0.5^o. It has units of meters and ranges from 0 to 20,000.

epsilon0StratMean2 (4-byte float, array size: nlath x nlonh):
Mean of epsilon0 conditioned on stratiform rain and use of 2A21 SRT at a horizontal resolution of 0.5 x 0.5 deg. It ranges from 0.0 to 5.0 (unitless).

epsilon0StratDev2 (4-byte float, array size: nlath x nlonh):
Standard deviation of epsilon0 conditioned on stratiform rain and use of 2A21 SRT at a horizontal resolution of 0.5 x 0.5 deg. It ranges from 0.0 to 5.0 (unitless).

epsilon0ConvMean2 (4-byte float, array size: nlath x nlonh):
Mean of epsilon0 conditioned on convective rain and use of 2A21 SRT at a horizontal resolution of 0.5 x 0.5 deg. It ranges from 0.0 to 5.0 (unitless).

epsilon0ConvDev2 (4-byte float, array size: nlath x nlonh):
Standard deviation of epsilon0 conditioned on convective rain and use of 2A21 SRT at a horizontal resolution of 0.5 x 0.5 deg. It ranges from 0.0 to 5.0 (unitless).

epsilonStratMean2 (4-byte float, array size: nlath x nlonh):
Mean of epsilon conditioned on stratiform rain and use of 2A21 SRT at a horizontal resolution of 0.5 x 0.5 deg. It ranges from 0.0 to 5.0(unitless).

epsilonStratDev2 (4-byte float, array size: nlath x nlonh):
Standard deviation of epsilon conditioned on stratiform rain and use of 2A21 SRT at a horizontal resolution of 0.5 x 0.5 deg. It ranges from 0.0 to 5.0 (unitless).

epsilonConvMean2 (4-byte float, array size: nlath x nlonh):
Mean of epsilon conditioned on convective rain and use of 2A21 SRT at a horizontal resolution of 0.5 x 0.5 deg. It ranges from 0.0 to 5.0 (unitless).

epsilonConvDev2 (4-byte float, array size: nlath x nlonh):
Standard deviation of epsilon conditioned on convective rain and use of 2A21 SRT at a horizontal resolution of 0.5 x 0.5 deg. It ranges from 0.0 to 5.0 (unitless).

surfRainMean2 (4-byte float, array size: nlath x nlonh):
Mean of non-zero near-surface rain rate at a horizontal resolution of 0.5 x 0.5. It ranges from 0.0 to 3000.0 mm/hr.

surfRainDev2 (4-byte float, array size: nlath x nlonh):
Standard Deviation of non-zero near-surface rain rate at a horizontal resolution of 0.5 x 0.5. It ranges from 0.0 to 3000.0 mm/hr.

e_surfRainMean2 (4-byte float, array size: nlath x nlonh):
Mean of non-zero estimated surface rain below clutter (See 2A25 algorithm user guide) in mm/hr. Over 0.5 x 0.5 deg. boxes. It ranges from 0.0 to 400.0mm/hr.

e_surfRainDev2 (4-byte float, array size: nlath x nlonh):
Standard deviation of non-zero estimated surface rain below clutter (See 2A25 algorithm user guide) in mm/hr. Over 0.5 x 0.5 deg. boxes. It ranges from 0.0 to 400.0 mm/hr.

bbZmaxMean2 (4-byte float, array size: nlath x nlonh):
Mean of maximum reflectivity in bright band at a horizontal resolution of 0.5 x 0.5. It ranges from 0.0 to 100.0 dBZ.

bbZmaxDev2 (4-byte float, array size: nlath x nlonh):
Mean of maximum reflectivity in bright band at a horizontal resolution of 0.5 x 0.5. It ranges from 0.0 to 100.0 dBZ.

sdepthMean2 (4-byte float, array size: nlath x nlonh):
Mean of snow depth at a horizontal resolution of 0.5^o x 0.5^o. It ranges from 0.0 to 20,000.0 m.

sdepthDev2 (4-byte float, array size: nlath x nlonh):
Standard deviation of snow depth at a horizontal resolution of 0.5 x 0.5. It ranges from 0.0 to 20,000.0 m.

bbHeightDev2 (4-byte float, array size: nlath x nlonh):
Standard deviation of bright band height at a horizontal resolution of 0.5 x 0.5. It ranges from 0.0 to 20,000.0 m.

surfRainConvMean2 (4-byte float, array size: nlath x nlonh):
Mean of non-zero near-surface rain conditioned on convective rain in mm/hr. Over 0.5 x 0.5 deg. boxes. It ranges from 0.0 to 400.0 mm/hr.

surfRainConvDev2 (4-byte float, array size: nlath x nlonh):
Standard deviation of non-zero near-surface rain conditioned on convective rain in mm/hr. Over 0.5 x 0.5 deg. boxes. It ranges from 0.0 to 400.0 mm/hr.

surfRainStratMean2 (4-byte float, array size: nlath x nlonh):
Mean of non-zero near-surface rain conditioned on stratiform rain in mm/hr. Over 0.5 x 0.5 deg. boxes. It ranges from 0.0 to 400.0 mm/hr.

surfRainStratDev2 (4-byte float, array size: nlath x nlonh):
Standard deviation of non-zero near-surface rain conditioned on stratiform rain in mm/hr. Over 0.5 x 0.5 deg. boxes. It ranges from 0.0 to 400.0 mm/hr.

e_surfRainConvMean2 (4-byte float, array size: nlath x nlonh):
Mean of non-zero estimated surface rain below clutter (See 2A25 algorithm user guide) conditioned on convective rain in mm/hr. Over 0.5 x 0.5 deg. boxes. It ranges from 0.0 to 400.0 mm/hr.

e_surfRainConvDev2 (4-byte float, array size: nlath x nlonh):
Standard deviation of non-zero estimated surface rain below clutter (See 2A25 algorithm user guide) conditioned on convective rain in mm/hr. Over 0.5 x 0.5 deg. boxes. It ranges from 0.0 to 400.0 mm/hr.

e_surfRainStratMean2 (4-byte float, array size: nlath x nlonh):
Mean of non-zero estimated surface rain below clutter (See 2A25 algorithm user guide) conditioned on stratiform rain in mm/hr. Over 0.5 x 0.5 deg. boxes. It ranges from 0.0 to 400.0 mm/hr.

e_surfRainStratDev2 (4-byte float, array size: nlath x nlonh):
Standard deviation of non-zero estimated surface rain below clutter (See 2A25 algorithm user guide) conditioned on stratiform rain in mm/hr. Over 0.5 x 0.5 deg. boxes. It ranges from 0.0 to 400.0 mm/hr.

rzStratA2 (4-byte float, array size: nlath x nlonh x 2):
The A parameter in rainfall-reflectivity relation $R = AZ^B$ from fitting of instantaneous

R,Z pairs conditioned on stratiform rain. Computed for near-surface and 2km heights. Over 0.5 x 0.5 deg. boxes. It ranges from 0 to 1.0.

rzStratB2 (4-byte float, array size: nlath x nlonh x 2):
The B parameter in rainfall-reflectivity relation R = AZ^B from fitting of instantaneous R,Z pairs conditioned on stratiform rain. Computed for near-surface and 2km heights. Over 0.5 x 0.5 deg. boxes. It ranges from 0 to 1.0.

rzConvA2 (4-byte float, array size: nlath x nlonh x 2):
The A parameter in rainfall-reflectivity relation R = AZ^B from fitting of instantaneous R,Z pairs conditioned on convective rain. Computed for near-surface and 2km heights. Over 0.5 x 0.5 deg. boxes. It ranges from 0 to 1.0.

rzConvB2 (4-byte float, array size: nlath x nlonh x 2):
The B parameter in rainfall-reflectivity relation R = AZ^B from fitting of instantaneous R,Z pairs conditioned on convective rain. Computed for near-surface and 2km heights. Over 0.5 x 0.5 deg. boxes. It ranges from 0 to 1.0.

rzA2 (4-byte float, array size: nlath x nlonh x 2):
The A parameter in rainfall-reflectivity relation R = AZ^B from fitting of instantaneous R,Z pairs. Computed for near-surface and 2km heights. Over 0.5 x 0.5 deg. boxes. It ranges from 0 to 1.0.

rzB2 (4-byte float, array size: nlath x nlonh x 2):
The B parameter in rainfall-reflectivity relation R = AZ^B from fitting of instantaneous R,Z pairs. Computed for near-surface and 2km heights. Over 0.5 x 0.5 deg. boxes. It ranges from 0 to 1.0.

ttlPix2 (4-byte integer, array size: nlath x nlonh):
The Total Pixel Number 2 is the number of total pixels over 0.5 x 0.5 boxes for one month. The range is 0 to 2,000,000.

bbPixNum2 (4-byte integer, array size: nlath x nlonh):
The number of bright band counts over each 0.5 x 0.5 box for one month. The range is 0 to 2,000,000.

surfRainPix2 (4-byte integer, array size: nlath x nlonh):
Near-surface rain counts at a horizontal resolution of 0.5 x 0.5. It ranges from 0 to 2,000,000,000.

surfRainConvPix2 (4-byte integer, array size: nlath x nlonh):
Counts of non-zero near-surface rain conditioned convective rain at a horizontal resolution of 0.5 x 0.5deg. It ranges from 0 to 2,000,000.

surfRainStratPix2 (4-byte integer, array size: nlath x nlonh):
Counts of non-zero near-surface rain conditioned statiform rain at a horizontal resolution of 0.5 x 0.5deg. It ranges from 0 to 2,000,000.

e_surfRainPix2 (4-byte integer, array size: nlath x nlonh):
Counts of non-zero estimated surface rain at a horizontal resolution of 0.5 x 0.5deg. It ranges from 0 to 2,000,000.

e_surfRainConvPix2 (4-byte integer, array size: nlath x nlonh):
Counts of non-zero estimated surface rain conditioned on convective rain at a horizontal resolution of 0.5^o x 0.5^o. It ranges from 0 to 2,000,000.

e_surfRainStratPix2 (4-byte integer, array size: nlath x nlonh):
Counts of non-zero estimated surface rain conditioned on statiform rain at a horizontal resolution of 0.5^o x 0.5^o. It ranges from 0 to 2,000,000.

rainPix2 (4-byte integer, array size: nlath x nlonh x nh3):
The Rain Pixel Number 2 is the monthly number of non-zero rain rate pixels for path-averaged rainfall and rainfall at the fixed heights of 2 km, 4 km, 6 km, and path average over 0.5 x 0.5 boxes. The range is 0 to 2,000,000.

convRainPix2 (4-byte integer, array size: nlath x nlonh x nh3):
The Convective Rain Pixel Number 2 is the number of non-zero rain rate pixels for convective rain at the fixed heights of of 2 km, 4 km, 6 km, and path average over 0.5x 0.5 boxes for one month. The range is 0 to 2,000,000.

stratRainPix2 (4-byte integer, array size: nlath x nlonh x nh3):
The Stratiform Rain Pixel Number 2 is the number of non-zero rain rate pixels for stratiform rain at the fixed heights of of 2 km, 4 km, 6 km, and path average over 0.5 x 0.5 boxes for one month. The range is 0 to 2,000,000.

shallowRainPix2 (4-byte integer, array size: nlath x nlonh):
Counts of shallow rain rain at a horizontal resolution of 0.5 x 0.5deg. It ranges from 0 to 2,000,000.

shallowIsoRainPix2 (4-byte integer, array size: nlath x nlonh):
Counts of shallow isolated rain rain at a horizontal resolution of 0.5 x 0.5deg. It ranges from 0 to 2,000,000.

epsilon0StratPix2 (4-byte integer, array size: nlath x nlonh):
Counts of epsilon0 conditioned on stratiform rain and use of 2A21 SRT at a horizontal resolution of 0.5^o x 0.5^o. It ranges from 0 to 2,000,000.

epsilon0ConvPix2 (4-byte integer, array size: nlath x nlonh):
Counts of epsilon0 conditioned on convective rain and use of 2A21 SRT at a horizontal resolution of 0.5^o x 0.5^o. It ranges from 0 to 2,000,000.

epsilonStratPix2 (4-byte integer, array size: nlath x nlonh):
Counts of epsilon conditioned on stratiform rain and use of 2A21 SRT at a horizontal resolution of 0.5^o x 0.5^o. It ranges from 0 to 2,000,000.

epsilonConvPix2 (4-byte integer, array size: nlath x nlonh):
Counts of epsilon conditioned on convective rain and use of 2A21 SRT at a horizontal resolution of 0.5^o x 0.5^o. It ranges from 0 to 2,000,000.

rzStratPix2 (4-byte integer, array size: nlath x nlonh x 2):
The number of R-Z coefficent pixel counts conditioned on stratiform rain for near-surface and 2km heights. Over 0.5^o x 0.5^o boxes. It ranges from 0 to 2,000,000.

rzConvPix2 (4-byte integer, array size: nlath x nlonh x 2):
The number of R-Z coefficent pixel counts conditioned on convective rain for near-surface and 2km heights. Over 0.5^o x 0.5^o boxes. It ranges from 0 to 2,000,000.

rzPix2 (4-byte integer, array size: nlath x nlonh x 2):
The number of R-Z coefficent pixel counts for near-surface and 2km heights. Over 0.5^o x 0.5^o boxes. It ranges from 0 to 2,000,000.

C Structure Header file:

```
#ifndef _TK_3A25_H_
#define _TK_3A25_H_

#ifndef _L3A25_GRID2_
#define _L3A25_GRID2_

typedef struct {
    float rainMean2[4][720][148];
    float rainDev2[4][720][148];
    float convRainMean2[4][720][148];
    float convRainDev2[4][720][148];
    float stratRainMean2[4][720][148];
    float stratRainDev2[4][720][148];
    float shallowRainMean2[720][148];
    float shallowRainDev2[720][148];
    float shallowIsoRainMean2[720][148];
    float shallowIsoRainDev2[720][148];
    float zmMean2[4][720][148];
    float convZmMean2[4][720][148];
    float stratZmMean2[4][720][148];
    float ztMean2[4][720][148];
    float convZtMean2[4][720][148];
    float stratZtMean2[4][720][148];
    float stormHeightMean[2][720][148];
    float stormHeightDev2[2][720][148];
    float bbHeightMean[720][148];
    float epsilon0StratMean2[720][148];
    float epsilon0StratDev2[720][148];
    float epsilon0ConvMean2[720][148];
    float epsilon0ConvDev2[720][148];
    float epsilonStratMean2[720][148];
    float epsilonStratDev2[720][148];
    float epsilonConvMean2[720][148];
    float epsilonConvDev2[720][148];
```

```
float surfRainMean2[720][148];
float surfRainDev2[720][148];
float e_surfRainMean2[720][148];
float e_surfRainDev2[720][148];
float bbZmaxMean2[720][148];
float bbZmaxDev2[720][148];
float sdepthMean2[720][148];
float sdepthDev2[720][148];
float bbHeightDev2[720][148];
float surfRainConvMean2[720][148];
float surfRainConvDev2[720][148];
float surfRainStratMean2[720][148];
float surfRainStratDev2[720][148];
float e_surfRainConvMean2[720][148];
float e_surfRainConvDev2[720][148];
float e_surfRainStratMean2[720][148];
float e_surfRainStratDev2[720][148];
float rzStratA2[2][720][148];
float rzStratB2[2][720][148];
float rzConvA2[2][720][148];
float rzConvB2[2][720][148];
float rzA2[2][720][148];
float rzB2[2][720][148];
int ttlPix2[720][148];
int bbPixNum2[720][148];
int surfRainPix2[720][148];
int surfRainConvPix2[720][148];
int surfRainStratPix2[720][148];
int e_surfRainPix2[720][148];
int e_surfRainConvPix2[720][148];
int e_surfRainStratPix2[720][148];
int rainPix2[4][720][148];
int convRainPix2[4][720][148];
int stratRainPix2[4][720][148];
int shallowRainPix2[720][148];
int shallowIsoRainPix2[720][148];
int epsilon0StratPix2[720][148];
int epsilon0ConvPix2[720][148];
int epsilonStratPix2[720][148];
int epsilonConvPix2[720][148];
int rzStratPix2[2][720][148];
int rzConvPix2[2][720][148];
int rzPix2[2][720][148];
```

```
} L3A25_GRID2;

#endif

#ifndef _L3A25_GRID1_
#define _L3A25_GRID1_

typedef struct {
    float rainMean1[6][72][16];
    float rainDev1[6][72][16];
    float convRainMean1[6][72][16];
    float convRainDev1[6][72][16];
    float stratRainMean1[6][72][16];
    float stratRainDev1[6][72][16];
    float shallowRainMean1[72][16];
    float shallowRainDev1[72][16];
    float shallowIsoRainMean1[72][16];
    float shallowIsoRainDev1[72][16];
    float zmMean1[6][72][16];
    float zmDev1[6][72][16];
    float convZmMean1[6][72][16];
    float convZmDev1[6][72][16];
    float stratZmMean1[6][72][16];
    float stratZmDev1[6][72][16];
    float ztMean1[6][72][16];
    float ztDev1[6][72][16];
    float convZtMean1[6][72][16];
    float convZtDev1[6][72][16];
    float stratZtMean1[6][72][16];
    float stratZtDev1[6][72][16];
    float piaSrtMean[5][72][16];
    float piaSrtDev[5][72][16];
    float piaHbMean[5][72][16];
    float piaHbDev[5][72][16];
    float pia0Mean[5][72][16];
    float pia0Dev[5][72][16];
    float pia2a25Mean[5][72][16];
    float pia2a25Dev[5][72][16];
    float piaSrtssMean[5][72][16];
    float piaSrtssDev[5][72][16];
    float piaHbssMean[5][72][16];
    float piaHbssDev[5][72][16];
    float pia0ssMean[5][72][16];
```

```
float pia0ssDev[5][72][16];
float pia2a25ssMean[5][72][16];
float pia2a25ssDev[5][72][16];
float stormHtMean[3][72][16];
float stormHtDev[3][72][16];
float xiMean[72][16];
float xiDev[72][16];
float nubfCorFacMean[72][16];
float nubfCorFacDev[72][16];
float bbHtMean[72][16];
float bbHtDev[72][16];
float epsilon0StratMean1[72][16];
float epsilon0StratDev1[72][16];
float epsilon0ConvMean1[72][16];
float epsilon0ConvDev1[72][16];
float epsilonStratMean1[72][16];
float epsilonStratDev1[72][16];
float epsilonConvMean1[72][16];
float epsilonConvDev1[72][16];
float surfRainMean1[72][16];
float surfRainDev1[72][16];
float e_surfRainMean1[72][16];
float e_surfRainDev1[72][16];
float bbZmaxMean1[72][16];
float bbZmaxDev1[72][16];
float bbNadirHtMean1[72][16];
float bbNadirHtDev1[72][16];
float bbNadirWidthMean1[72][16];
float bbNadirWidthDev1[72][16];
float bbNadirZmaxMean1[72][16];
float bbNadirZmaxDev1[72][16];
float sdepthMean1[72][16];
float sdepthDev1[72][16];
float surfRainConvMean1[72][16];
float surfRainConvDev1[72][16];
float surfRainStratMean1[72][16];
float surfRainStratDev1[72][16];
float e_surfRainConvMean1[72][16];
float e_surfRainConvDev1[72][16];
float e_surfRainStratMean1[72][16];
float e_surfRainStratDev1[72][16];
float rzStratA1[2][72][16];
float rzStratB1[2][72][16];
```

```
float rzConvA1[2][72][16];
float rzConvB1[2][72][16];
float rzA1[2][72][16];
float rzB1[2][72][16];
int ttlPix1[72][16];
int bbPixNum1[72][16];
int bbNadirPix1[72][16];
int surfRainPix1[72][16];
int surfRainStratPix1[72][16];
int surfRainConvPix1[72][16];
int e_surfRainPix1[72][16];
int e_surfRainStratPix1[72][16];
int e_surfRainConvPix1[72][16];
int epsilon0StratPix1[72][16];
int epsilon0ConvPix1[72][16];
int epsilonStratPix1[72][16];
int epsilonConvPix1[72][16];
int rainPix1[6][72][16];
int convRainPix1[6][72][16];
int stratRainPix1[6][72][16];
short ttlAnglePix1[4][72][16];
short rainAnglePix1[4][72][16];
int rzStratPix1[2][72][16];
int rzConvPix1[2][72][16];
int rzPix1[2][72][16];
int shallowRainPix1[72][16];
int shallowIsoRainPix1[72][16];
short stormHH[30][72][16];
short convStormHH[30][72][16];
short stratStormHH[30][72][16];
short BBHH[30][72][16];
short bbNadirWidthH[30][72][16];
short bbNadirHH[30][72][16];
short bbNadirZmaxH[30][72][16];
short snowIceLH[30][72][16];
short bbZmaxH[30][72][16];
short epsilon0StratH[30][72][16];
short epsilon0ConvH[30][72][16];
short epsilonStratH[30][72][16];
short epsilonConvH[30][72][16];
short surfRainH[30][72][16];
short e_surfRainH[30][72][16];
short surfRainConvH[30][72][16];
```

```
short surfRainStratH[30][72][16];
short e_surfRainConvH[30][72][16];
short e_surfRainStratH[30][72][16];
short zmH[6][30][72][16];
short convZmH[6][30][72][16];
short stratZmH[6][30][72][16];
short ztH[6][30][72][16];
short convZtH[6][30][72][16];
short stratZtH[6][30][72][16];
short rainH[6][30][72][16];
short convRainH[6][30][72][16];
short stratRainH[6][30][72][16];
short shallowRainH[30][72][16];
short shallowIsoRainH[30][72][16];
short piaSrtH[5][30][72][16];
short piaHbH[5][30][72][16];
short pia0H[5][30][72][16];
short pia2a25H[5][30][72][16];
short piaSrtssH[5][30][72][16];
short piaHbssH[5][30][72][16];
short pia0ssH[5][30][72][16];
short pia2a25ssH[5][30][72][16];
short xiH[30][72][16];
short nubfH[30][72][16];
float rainCCoef[3][72][16];
float convRainCCoef[3][72][16];
float stratRainCCoef[3][72][16];
float piaCCoef[6][5][72][16];
float piaSrtPix[5][72][16];
float piaHbPix[5][72][16];
float pia0Pix[5][72][16];
float pia2a25Pix[5][72][16];
float piaSrtssPix[5][72][16];
float piaHbssPix[5][72][16];
float pia0ssPix[5][72][16];
float pia2a25ssPix[5][72][16];
float rainCCoefPix[3][72][16];
float stratCCoefPix[3][72][16];
float convCCoefPix[3][72][16];
float piaCCoefPix[5][72][16];
float epsilonFit[72][16];
float epsilonFitRMS[72][16];
int epsilonFitPix[72][16];
```

```
} L3A25_GRID1;

#endif

#ifndef _L3A25_GRIDS_
#define _L3A25_GRIDS_

typedef struct {
    L3A25_GRID1 Grid1;
    L3A25_GRID2 Grid2;
} L3A25_GRIDS;

#endif

#endif
```

Fortran Structure Header file:

```
STRUCTURE /L3A25_GRID2/
    REAL*4 rainMean2(148,720,4)
    REAL*4 rainDev2(148,720,4)
    REAL*4 convRainMean2(148,720,4)
    REAL*4 convRainDev2(148,720,4)
    REAL*4 stratRainMean2(148,720,4)
    REAL*4 stratRainDev2(148,720,4)
    REAL*4 shallowRainMean2(148,720)
    REAL*4 shallowRainDev2(148,720)
    REAL*4 shallowIsoRainMean2(148,720)
    REAL*4 shallowIsoRainDev2(148,720)
    REAL*4 zmMean2(148,720,4)
    REAL*4 convZmMean2(148,720,4)
    REAL*4 stratZmMean2(148,720,4)
    REAL*4 ztMean2(148,720,4)
    REAL*4 convZtMean2(148,720,4)
    REAL*4 stratZtMean2(148,720,4)
    REAL*4 stormHeightMean(148,720,2)
    REAL*4 stormHeightDev2(148,720,2)
    REAL*4 bbHeightMean(148,720)
    REAL*4 epsilon0StratMean2(148,720)
    REAL*4 epsilon0StratDev2(148,720)
    REAL*4 epsilon0ConvMean2(148,720)
    REAL*4 epsilon0ConvDev2(148,720)
    REAL*4 epsilonStratMean2(148,720)
```

```
REAL*4 epsilonStratDev2(148,720)
REAL*4 epsilonConvMean2(148,720)
REAL*4 epsilonConvDev2(148,720)
REAL*4 surfRainMean2(148,720)
REAL*4 surfRainDev2(148,720)
REAL*4 e_surfRainMean2(148,720)
REAL*4 e_surfRainDev2(148,720)
REAL*4 bbZmaxMean2(148,720)
REAL*4 bbZmaxDev2(148,720)
REAL*4 sdepthMean2(148,720)
REAL*4 sdepthDev2(148,720)
REAL*4 bbHeightDev2(148,720)
REAL*4 surfRainConvMean2(148,720)
REAL*4 surfRainConvDev2(148,720)
REAL*4 surfRainStratMean2(148,720)
REAL*4 surfRainStratDev2(148,720)
REAL*4 e_surfRainConvMean2(148,720)
REAL*4 e_surfRainConvDev2(148,720)
REAL*4 e_surfRainStratMean2(148,720)
REAL*4 e_surfRainStratDev2(148,720)
REAL*4 rzStratA2(148,720,2)
REAL*4 rzStratB2(148,720,2)
REAL*4 rzConvA2(148,720,2)
REAL*4 rzConvB2(148,720,2)
REAL*4 rzA2(148,720,2)
REAL*4 rzB2(148,720,2)
INTEGER*4 ttlPix2(148,720)
INTEGER*4 bbPixNum2(148,720)
INTEGER*4 surfRainPix2(148,720)
INTEGER*4 surfRainConvPix2(148,720)
INTEGER*4 surfRainStratPix2(148,720)
INTEGER*4 e_surfRainPix2(148,720)
INTEGER*4 e_surfRainConvPix2(148,720)
INTEGER*4 e_surfRainStratPix2(148,720)
INTEGER*4 rainPix2(148,720,4)
INTEGER*4 convRainPix2(148,720,4)
INTEGER*4 stratRainPix2(148,720,4)
INTEGER*4 shallowRainPix2(148,720)
INTEGER*4 shallowIsoRainPix2(148,720)
INTEGER*4 epsilon0StratPix2(148,720)
INTEGER*4 epsilon0ConvPix2(148,720)
INTEGER*4 epsilonStratPix2(148,720)
INTEGER*4 epsilonConvPix2(148,720)
```

```
    INTEGER*4 rzStratPix2(148,720,2)
    INTEGER*4 rzConvPix2(148,720,2)
    INTEGER*4 rzPix2(148,720,2)
END STRUCTURE

STRUCTURE /L3A25_GRID1/
    REAL*4 rainMean1(16,72,6)
    REAL*4 rainDev1(16,72,6)
    REAL*4 convRainMean1(16,72,6)
    REAL*4 convRainDev1(16,72,6)
    REAL*4 stratRainMean1(16,72,6)
    REAL*4 stratRainDev1(16,72,6)
    REAL*4 shallowRainMean1(16,72)
    REAL*4 shallowRainDev1(16,72)
    REAL*4 shallowIsoRainMean1(16,72)
    REAL*4 shallowIsoRainDev1(16,72)
    REAL*4 zmMean1(16,72,6)
    REAL*4 zmDev1(16,72,6)
    REAL*4 convZmMean1(16,72,6)
    REAL*4 convZmDev1(16,72,6)
    REAL*4 stratZmMean1(16,72,6)
    REAL*4 stratZmDev1(16,72,6)
    REAL*4 ztMean1(16,72,6)
    REAL*4 ztDev1(16,72,6)
    REAL*4 convZtMean1(16,72,6)
    REAL*4 convZtDev1(16,72,6)
    REAL*4 stratZtMean1(16,72,6)
    REAL*4 stratZtDev1(16,72,6)
    REAL*4 piaSrtMean(16,72,5)
    REAL*4 piaSrtDev(16,72,5)
    REAL*4 piaHbMean(16,72,5)
    REAL*4 piaHbDev(16,72,5)
    REAL*4 pia0Mean(16,72,5)
    REAL*4 pia0Dev(16,72,5)
    REAL*4 pia2a25Mean(16,72,5)
    REAL*4 pia2a25Dev(16,72,5)
    REAL*4 piaSrtssMean(16,72,5)
    REAL*4 piaSrtssDev(16,72,5)
    REAL*4 piaHbssMean(16,72,5)
    REAL*4 piaHbssDev(16,72,5)
    REAL*4 pia0ssMean(16,72,5)
    REAL*4 pia0ssDev(16,72,5)
    REAL*4 pia2a25ssMean(16,72,5)
```

```
REAL*4 pia2a25ssDev(16,72,5)
REAL*4 stormHtMean(16,72,3)
REAL*4 stormHtDev(16,72,3)
REAL*4 xiMean(16,72)
REAL*4 xiDev(16,72)
REAL*4 nubfCorFacMean(16,72)
REAL*4 nubfCorFacDev(16,72)
REAL*4 bbHtMean(16,72)
REAL*4 bbHtDev(16,72)
REAL*4 epsilon0StratMean1(16,72)
REAL*4 epsilon0StratDev1(16,72)
REAL*4 epsilon0ConvMean1(16,72)
REAL*4 epsilon0ConvDev1(16,72)
REAL*4 epsilonStratMean1(16,72)
REAL*4 epsilonStratDev1(16,72)
REAL*4 epsilonConvMean1(16,72)
REAL*4 epsilonConvDev1(16,72)
REAL*4 surfRainMean1(16,72)
REAL*4 surfRainDev1(16,72)
REAL*4 e_surfRainMean1(16,72)
REAL*4 e_surfRainDev1(16,72)
REAL*4 bbZmaxMean1(16,72)
REAL*4 bbZmaxDev1(16,72)
REAL*4 bbNadirHtMean1(16,72)
REAL*4 bbNadirHtDev1(16,72)
REAL*4 bbNadirWidthMean1(16,72)
REAL*4 bbNadirWidthDev1(16,72)
REAL*4 bbNadirZmaxMean1(16,72)
REAL*4 bbNadirZmaxDev1(16,72)
REAL*4 sdepthMean1(16,72)
REAL*4 sdepthDev1(16,72)
REAL*4 surfRainConvMean1(16,72)
REAL*4 surfRainConvDev1(16,72)
REAL*4 surfRainStratMean1(16,72)
REAL*4 surfRainStratDev1(16,72)
REAL*4 e_surfRainConvMean1(16,72)
REAL*4 e_surfRainConvDev1(16,72)
REAL*4 e_surfRainStratMean1(16,72)
REAL*4 e_surfRainStratDev1(16,72)
REAL*4 rzStratA1(16,72,2)
REAL*4 rzStratB1(16,72,2)
REAL*4 rzConvA1(16,72,2)
REAL*4 rzConvB1(16,72,2)
```

```
REAL*4 rzA1(16,72,2)
REAL*4 rzB1(16,72,2)
INTEGER*4 ttlPix1(16,72)
INTEGER*4 bbPixNum1(16,72)
INTEGER*4 bbNadirPix1(16,72)
INTEGER*4 surfRainPix1(16,72)
INTEGER*4 surfRainStratPix1(16,72)
INTEGER*4 surfRainConvPix1(16,72)
INTEGER*4 e_surfRainPix1(16,72)
INTEGER*4 e_surfRainStratPix1(16,72)
INTEGER*4 e_surfRainConvPix1(16,72)
INTEGER*4 epsilon0StratPix1(16,72)
INTEGER*4 epsilon0ConvPix1(16,72)
INTEGER*4 epsilonStratPix1(16,72)
INTEGER*4 epsilonConvPix1(16,72)
INTEGER*4 rainPix1(16,72,6)
INTEGER*4 convRainPix1(16,72,6)
INTEGER*4 stratRainPix1(16,72,6)
INTEGER*2 ttlAnglePix1(16,72,4)
INTEGER*2 rainAnglePix1(16,72,4)
INTEGER*4 rzStratPix1(16,72,2)
INTEGER*4 rzConvPix1(16,72,2)
INTEGER*4 rzPix1(16,72,2)
INTEGER*4 shallowRainPix1(16,72)
INTEGER*4 shallowIsoRainPix1(16,72)
INTEGER*2 stormHH(16,72,30)
INTEGER*2 convStormHH(16,72,30)
INTEGER*2 stratStormHH(16,72,30)
INTEGER*2 BBHH(16,72,30)
INTEGER*2 bbNadirWidthH(16,72,30)
INTEGER*2 bbNadirHH(16,72,30)
INTEGER*2 bbNadirZmaxH(16,72,30)
INTEGER*2 snowIceLH(16,72,30)
INTEGER*2 bbZmaxH(16,72,30)
INTEGER*2 epsilon0StratH(16,72,30)
INTEGER*2 epsilon0ConvH(16,72,30)
INTEGER*2 epsilonStratH(16,72,30)
INTEGER*2 epsilonConvH(16,72,30)
INTEGER*2 surfRainH(16,72,30)
INTEGER*2 e_surfRainH(16,72,30)
INTEGER*2 surfRainConvH(16,72,30)
INTEGER*2 surfRainStratH(16,72,30)
INTEGER*2 e_surfRainConvH(16,72,30)
```

```
    INTEGER*2 e_surfRainStratH(16,72,30)
    INTEGER*2 zmH(16,72,30,6)
    INTEGER*2 convZmH(16,72,30,6)
    INTEGER*2 stratZmH(16,72,30,6)
    INTEGER*2 ztH(16,72,30,6)
    INTEGER*2 convZtH(16,72,30,6)
    INTEGER*2 stratZtH(16,72,30,6)
    INTEGER*2 rainH(16,72,30,6)
    INTEGER*2 convRainH(16,72,30,6)
    INTEGER*2 stratRainH(16,72,30,6)
    INTEGER*2 shallowRainH(16,72,30)
    INTEGER*2 shallowIsoRainH(16,72,30)
    INTEGER*2 piaSrtH(16,72,30,5)
    INTEGER*2 piaHbH(16,72,30,5)
    INTEGER*2 pia0H(16,72,30,5)
    INTEGER*2 pia2a25H(16,72,30,5)
    INTEGER*2 piaSrtssH(16,72,30,5)
    INTEGER*2 piaHbssH(16,72,30,5)
    INTEGER*2 pia0ssH(16,72,30,5)
    INTEGER*2 pia2a25ssH(16,72,30,5)
    INTEGER*2 xiH(16,72,30)
    INTEGER*2 nubfH(16,72,30)
    REAL*4 rainCCoef(16,72,3)
    REAL*4 convRainCCoef(16,72,3)
    REAL*4 stratRainCCoef(16,72,3)
    REAL*4 piaCCoef(16,72,5,6)
    REAL*4 piaSrtPix(16,72,5)
    REAL*4 piaHbPix(16,72,5)
    REAL*4 pia0Pix(16,72,5)
    REAL*4 pia2a25Pix(16,72,5)
    REAL*4 piaSrtssPix(16,72,5)
    REAL*4 piaHbssPix(16,72,5)
    REAL*4 pia0ssPix(16,72,5)
    REAL*4 pia2a25ssPix(16,72,5)
    REAL*4 rainCCoefPix(16,72,3)
    REAL*4 stratCCoefPix(16,72,3)
    REAL*4 convCCoefPix(16,72,3)
    REAL*4 piaCCoefPix(16,72,5)
    REAL*4 epsilonFit(16,72)
    REAL*4 epsilonFitRMS(16,72)
    INTEGER*4 epsilonFitPix(16,72)
END STRUCTURE
```

```
STRUCTURE /L3A25_GRIDS/
    RECORD /L3A25_GRID1/ Grid1
    RECORD /L3A25_GRID2/ Grid2
END STRUCTURE
```

7.13 3A26 - Surface Rain

3A26, "Surface Rain", computes the distribution of rainfall on a 5 x 5 grid on a monthly basis. The output products are calculated at three fixed heights (2, 4, and 6 km) and for the path-averaged rain rates. 3A26 will also compute fitting parameters for cumulative probability functions of rain rate as a function of 20 rain categories and 6 thresholds. The following sections describe the structure and contents of the format.

Dimension definitions:

nlat	16	Number of 5 degree grid intervals of latitude from 40^{o}N to 40^{o}S.
nlon	72	Number of 5 degree grid intervals of longitude from 180^{o}W to 180^{o}E.
nh3	4	Number of fixed heights above the earth ellipsoid, at 2, 4, and 6 km plus one for path-average.
ncat3	25	Number of categories for probability distribution functions. Rain rate thresholds (mm/hr) are: 12., 14., 16., 18., 20., 22., 24., 26., 28., 30., 32., 34., 36., 38., 40., 42., 44., 46., 48., 50., 52., 54., 56., 58., 60.
nthrsh	6	Number of thresholds used for probability distribution functions. Q-thresholds for Zero order: 0.1, 0.2, 0.3, 0.5, 0.75, 50. Q-thresholds for HB: 0.1, 0.2, 0.3, 0.5, 0.75, 0.9999 pia-thresholds for SRT: 1.5, 1., 0.8, 0.6, 0.4, 0.1

Figure 69 shows the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for TRMM Products for details.

InputFileNames (Metadata):
InputFileNames contains a list of input file names for this granule. See Metadata for TRMM Products for details.

InputAlgorithmVersions (Metadata):
InputAlgorithmVersions contains a list of input algorithm versions for this granule. See Metadata for TRMM Products for details.

InputGenerationDateTimes (Metadata):
InputGenerationDateTimes contains a list of input generation datetimes. See Metadata for TRMM Products for details.

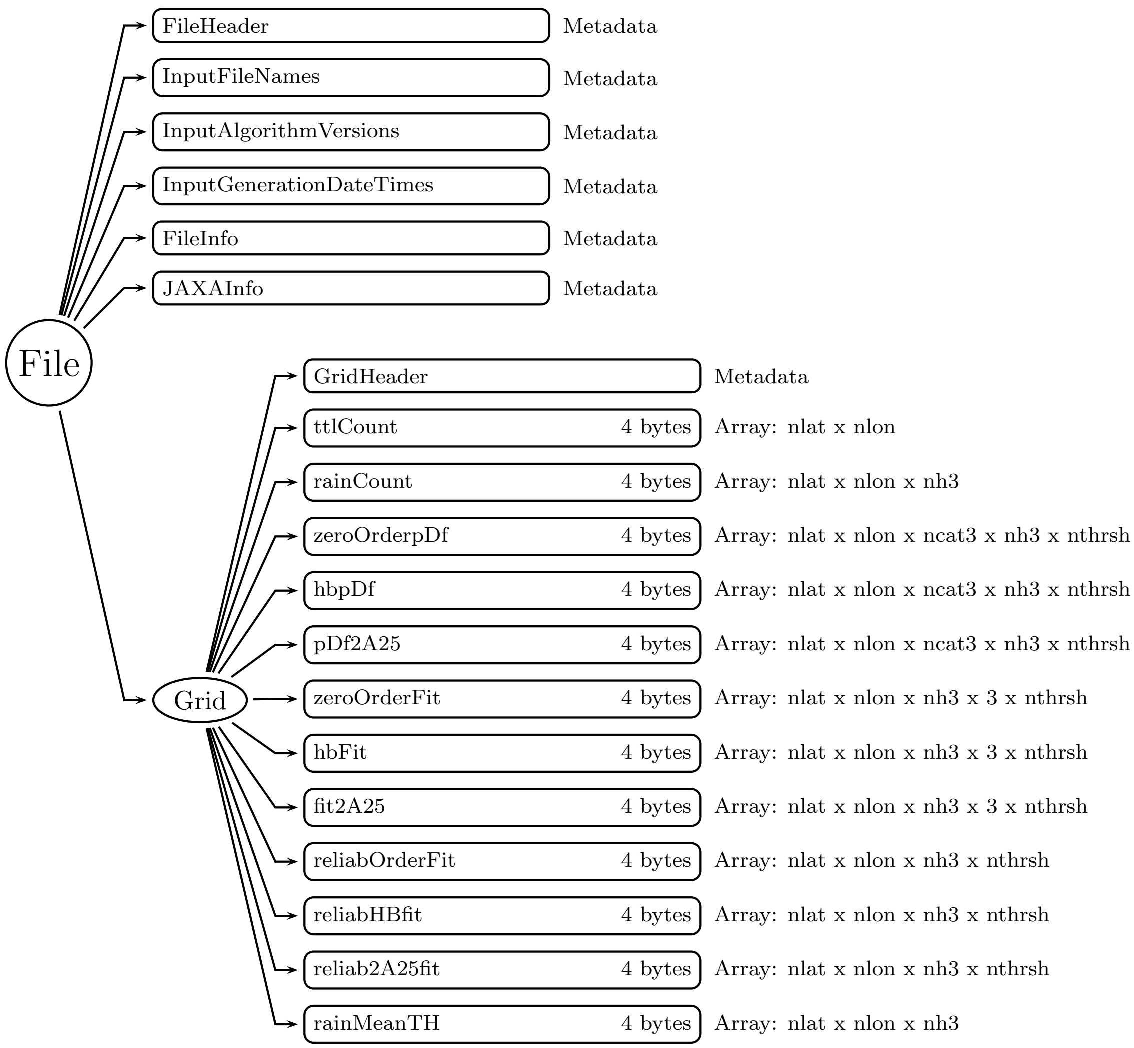


Figure 69: Data Format Structure for 3A26, Surface Rain

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for TRMM Products for details.

JAXAInfo (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by PR algorithms only. See Metadata for TRMM Products for details.

Grid (Grid)

GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for TRMM Products for details.

ttlCount (4-byte integer, array size: nlat x nlon):
This is the total number of counts (measurements) per month at each 5 x 5 box. Ranges are 0 to 2,147,483,647.

rainCount (4-byte integer, array size: nlat x nlon x nh3):
Total number of rain counts per month at each 5 x 5 box. This is computed at 2 km, 4 km, 6 km, and for the path-average. Ranges are 0 to 2,147,483,647.

zeroOrderpDf (4-byte integer, array size: nlat x nlon x ncat3 x nh3 x nthrsh):
Probability distribution function (cumulative) in counts of the zeroth order rain rate estimate at each 5 x 5 box. The pDf is computed at 2 km, 4 km, 6 km, and for the path average. Ranges are 0 to 2,147,483,647.

hbpDf (4-byte integer, array size: nlat x nlon x ncat3 x nh3 x nthrsh):
Probability distribution function (cumulative) in counts of the Hitschfield-Bordan (HB) rain rate estimate at each 5 x 5 box. The pDf is computed at 2 km, 4 km, 6 km, and for the path average. Ranges are 0 to 2,147,483,647.

pDf2A25 (4-byte integer, array size: nlat x nlon x ncat3 x nh3 x nthrsh):
Probability distribution function (cumulative) in counts of the Surface Reference Technique (SRT) rain rate estimate at each 5 x 5 box. The pDf is computed computed at 2 km, 4 km, 6 km, and for the path average. Ranges are 0 to 2,147,483,647.

zeroOrderFit (4-byte float, array size: nlat x nlon x nh3 x 3 x nthrsh):
The mean, variance and probability of rain parameters for the log-normal model obtained from the zeroth order pDf. Fitting parameters are given at 2 km, 4 km, 6 km, and for the path average. In addition, 5 thresholds are used. Ranges are TBD.

hbFit (4-byte float, array size: nlat x nlon x nh3 x 3 x nthrsh):
The 3 fitting parameters for the log-normal model obtained from the HB pDf. Fitting parameters are given at 2 km, 4 km, 6 km, and for the path average. In addition, 5 thresholds are used. Ranges are TBD.

fit2A25 (4-byte float, array size: nlat x nlon x nh3 x 3 x nthrsh):
The 3 fitting parameters for the log-normal model obtained from the SRT pDf. Fitting

parameters are given at 2 km, 4 km, 6 km, and for the path average and 5 thresholds. Ranges are TBD.

reliabOrderFit (4-byte float, array size: nlat x nlon x nh3 x nthrsh):
Reliability parameter for the 0th order fit. Units and ranges are TBD.

reliabHBfit (4-byte float, array size: nlat x nlon x nh3 x nthrsh):
Reliability parameter for the HB fit. Units and ranges are TBD.

reliab2A25fit (4-byte float, array size: nlat x nlon x nh3 x nthrsh):
Reliability parameter for the SRT fit. Units and ranges are TBD.

rainMeanTH (4-byte float, array size: nlat x nlon x nh3):
The mean monthly unconditioned rain rate (mm/h) as determined from the threshold method (in particular, it is determined from the fitting parameters for the '0th-order method' using a single 'Q' threshold for each height level). Range is 0.0 to 3000.0 mm/h.

C Structure Header file:

```
#ifndef _TK_3A26_H_
#define _TK_3A26_H_

#ifndef _L3A26_GRID_
#define _L3A26_GRID_

typedef struct {
    int ttlCount[72][16];
    int rainCount[4][72][16];
    int zeroOrderpDf[6][4][25][72][16];
    int hbpDf[6][4][25][72][16];
    int pDf2A25[6][4][25][72][16];
    float zeroOrderFit[6][3][4][72][16];
    float hbFit[6][3][4][72][16];
    float fit2A25[6][3][4][72][16];
    float reliabOrderFit[6][4][72][16];
    float reliabHBfit[6][4][72][16];
    float reliab2A25fit[6][4][72][16];
    float rainMeanTH[4][72][16];
} L3A26_GRID;

#endif

#endif
```

Fortran Structure Header file:

```
STRUCTURE /L3A26_GRID/
    INTEGER*4 ttlCount(16,72)
    INTEGER*4 rainCount(16,72,4)
    INTEGER*4 zeroOrderpDf(16,72,25,4,6)
    INTEGER*4 hbpDf(16,72,25,4,6)
    INTEGER*4 pDf2A25(16,72,25,4,6)
    REAL*4 zeroOrderFit(16,72,4,3,6)
    REAL*4 hbFit(16,72,4,3,6)
    REAL*4 fit2A25(16,72,4,3,6)
    REAL*4 reliabOrderFit(16,72,4,6)
    REAL*4 reliabHBfit(16,72,4,6)
    REAL*4 reliab2A25fit(16,72,4,6)
    REAL*4 rainMeanTH(16,72,4)
END STRUCTURE
```

7.14 3B31 - Rainfall Combined

3B31, "Rainfall Combined", uses the high quality retrievals done for the narrow swath in 2B31 to calibrate the wide swath retrievals generated in 2A12. For each 0.5^o x 0.5^o latitude/longitude box and each vertical layer, an adjustment ratio will be calculated for the swath overlap region for one month. Only TMI pixels with 2A12 pixelStatus equal to zero are included in monthly averages. The major effect of the pixelStatus requirement is to remove sea ice.

Dimension definitions:

nlat	160	Number of 0.5^o grid intervals of latitude from 40^o N to 40^o S.
nlon	720	Number of 0.5^o grid intervals of longitude from 180^o W to 180^o E.
nlayer	28	Number of profiling layers. The top of each layer is 0.5, 1.0, 1.5, ..., 9.5, 10.0, 11.0, ..., 18.0 km. TMI layer tops are heights above the earth's surface; COMB layer tops are heights above the earth ellipsoid. These layers will have the same height over the ocean but different height over the land, the difference being the land elevation.

Figure 70 shows the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for TRMM Products for details.

InputFileNames (Metadata):
InputFileNames contains a list of input file names for this granule. See Metadata for

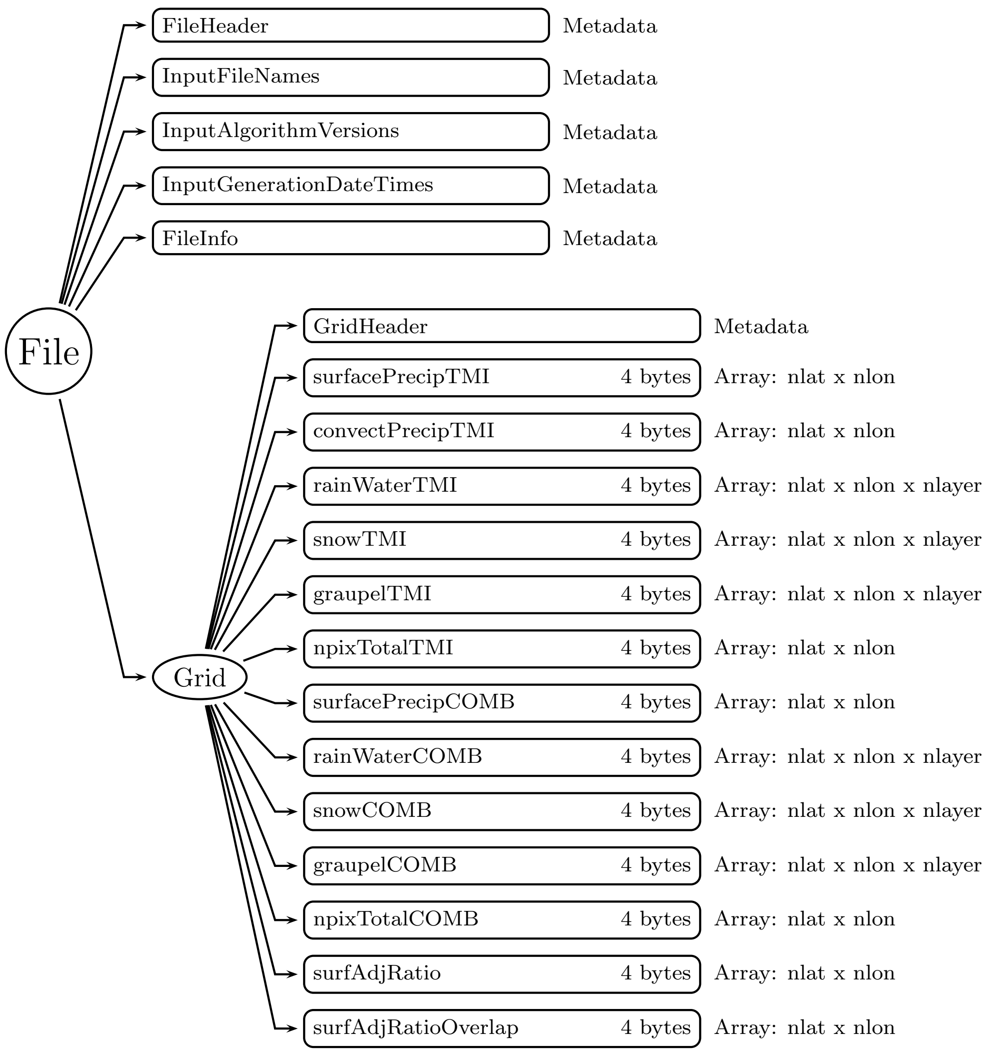


Figure 70: Data Format Structure for 3B31, Rainfall Combined

TRMM Products for details.

InputAlgorithmVersions (Metadata):
InputAlgorithmVersions contains a list of input algorithm versions for this granule. See Metadata for TRMM Products for details.

InputGenerationDateTimes (Metadata):
InputGenerationDateTimes contains a list of input generation datetimes. See Metadata for TRMM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for TRMM Products for details.

Grid (Grid)

GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for TRMM Products for details.

surfacePrecipTMI (4-byte float, array size: nlat x nlon):
Monthly mean surface precipitation from the wide swath of 2A12 accumulated in each 0.5^o x 0.5^o box. Values range from 0 to 3000 mm. Special values are defined as:
-9999.9 Missing value

convectPrecipTMI (4-byte float, array size: nlat x nlon):
Monthly mean sonvective surface precipitation from the wide swath of 2A12 accumulated in each 0.5^o x 0.5^o box. Values range from 0 to 3000 mm. Special values are defined as:
-9999.9 Missing value

rainWaterTMI (4-byte float, array size: nlat x nlon x nlayer):
Monthly mean rain water content from the wide swath of 2A12 at each vertical layer in each 0.5^o x 0.5^o box. Values range from 0 to 10 gm^-3. Special values are defined as:
-9999.9 Missing value

snowTMI (4-byte float, array size: nlat x nlon x nlayer):
Monthly mean snow liquid water content from the wide swath of 2A12 at each vertical layer in each 0.5^o x 0.5^o box. Values range from 0 to 10 gm^-3. Special values are defined as:
-9999.9 Missing value

graupelTMI (4-byte float, array size: nlat x nlon x nlayer):
Monthly mean graupel liquid water content from the wide swath of 2A12 at each vertical layer in each 0.5^o x 0.5^o box. Values range from 0 to 10 gm^-3. Special values are defined as:
-9999.9 Missing value

npixTotalTMI (4-byte integer, array size: nlat x nlon):
The monthly number of pixels with pixelStatus equal to zero for each grid. The major

effect of the pixelStatus requirement is to remove sea ice. npixTotalTMI is used to compute the monthly means described above. Values range from 0 to 10000. Special values are defined as:
-9999 Missing value

surfacePrecipCOMB (4-byte float, array size: nlat x nlon):
Surface precipitation from the narrow swath of 2B31 accumulated in each 0.5^o x 0.5^o box. Values range from 0 to 3000 mm. Special values are defined as:
-9999.9 Missing value

rainWaterCOMB (4-byte float, array size: nlat x nlon x nlayer):
Monthly mean rain water content from the narrow swath of 2B31 accumulated in each at each vertical layer 0.5^o x 0.5^o box. Values range from 0 to 10 gm^-3. Special values are defined as:
-9999.9 Missing value

snowCOMB (4-byte float, array size: nlat x nlon x nlayer):
Monthly mean snow liquid water content from the narrow swath of 2B31 accumulated in each at each vertical layer 0.5^o x 0.5^o box. Values range from 0 to 10 gm^-3. Special values are defined as:
-9999.9 Missing value

graupelCOMB (4-byte float, array size: nlat x nlon x nlayer):
Monthly mean graupel liquid water content from the narrow swath of 2B31 accumulated in each at each vertical layer 0.5^o x 0.5^o box. Values range from 0 to 10 gm^-3. Special values are defined as:
-9999.9 Missing value

npixTotalCOMB (4-byte integer, array size: nlat x nlon):
The monthly number of pixels npixTotalCOMB is used to compute the monthly means described above. Values range from 0 to 10000. Special values are defined as:
-9999 Missing value

surfAdjRatio (4-byte float, array size: nlat x nlon):
The ratio of 2B31 (narrow swath) to 2A12 (wide swath) surface precipitation, calculated for each 0.5^o x 0.5^o box. Special values are defined as:
-9999.9 Missing value

surfAdjRatioOverlap (4-byte float, array size: nlat x nlon):
The ratio of 2B31 to 2A12 surface precipitation, calculated from the overlap region (where both 2B31 and 2A12 have valid samples) for each 0.5^o x 0.5^o box. Special values are defined as:
-9999.9 Missing value

C Structure Header file:

```
#ifndef _TK_3B31_H_
#define _TK_3B31_H_
```

```
#ifndef _L3B31_GRID_
#define _L3B31_GRID_

typedef struct {
    float surfacePrecipTMI[720][160];
    float convectPrecipTMI[720][160];
    float rainWaterTMI[28][720][160];
    float snowTMI[28][720][160];
    float graupelTMI[28][720][160];
    int npixTotalTMI[720][160];
    float surfacePrecipCOMB[720][160];
    float rainWaterCOMB[28][720][160];
    float snowCOMB[28][720][160];
    float graupelCOMB[28][720][160];
    int npixTotalCOMB[720][160];
    float surfAdjRatio[720][160];
    float surfAdjRatioOverlap[720][160];
} L3B31_GRID;

#endif

#endif
```

Fortran Structure Header file:

```
STRUCTURE /L3B31_GRID/
    REAL*4 surfacePrecipTMI(160,720)
    REAL*4 convectPrecipTMI(160,720)
    REAL*4 rainWaterTMI(160,720,28)
    REAL*4 snowTMI(160,720,28)
    REAL*4 graupelTMI(160,720,28)
    INTEGER*4 npixTotalTMI(160,720)
    REAL*4 surfacePrecipCOMB(160,720)
    REAL*4 rainWaterCOMB(160,720,28)
    REAL*4 snowCOMB(160,720,28)
    REAL*4 graupelCOMB(160,720,28)
    INTEGER*4 npixTotalCOMB(160,720)
    REAL*4 surfAdjRatio(160,720)
    REAL*4 surfAdjRatioOverlap(160,720)
END STRUCTURE
```

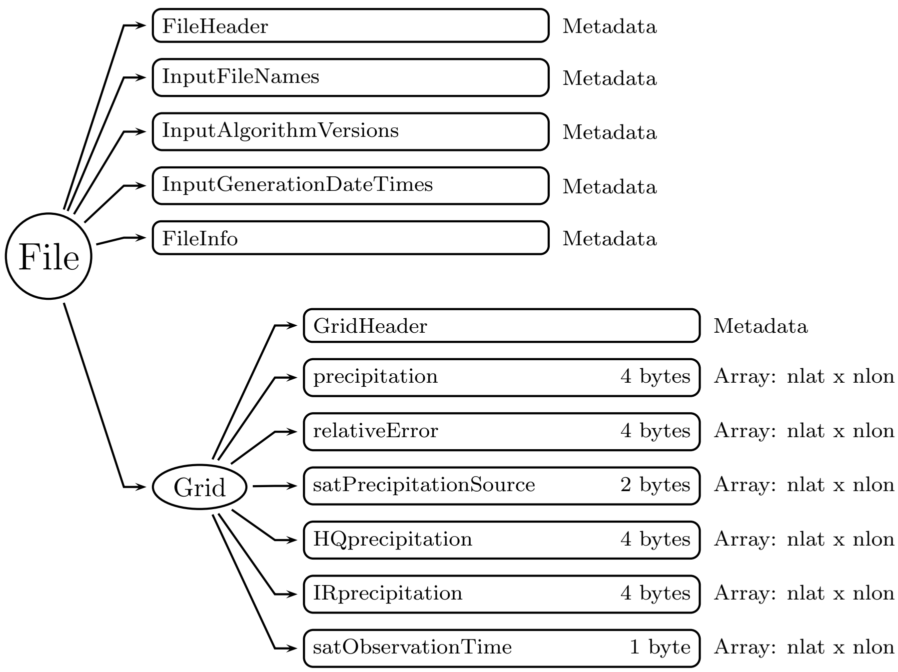


Figure 71: Data Format Structure for 3B42, TRMM and Other Sensors - 3-Hourly

7.15 3B42 - TRMM and Other Sensors - 3-Hourly

3B42, "TRMM and Other Sensors - 3-Hourly", provides precipitation estimates in the TRMM regions that have the (nearly-zero) bias of the "TRMM Combined Instrument" precipitation estimate and the dense sampling of high-quality microwave data with fill-in using microwave-calibrated infrared estimates. The granule size is 3 hours. The following sections describe the structure and contents of the format.

Dimension definitions:

nlat	400	Number of 0.25^o grid intervals of latitude from 50^o N to 50^o S.
nlon	1440	Number of 0.25^o grid intervals of longitude from 180^o W to 180^o E.

Figure 71 shows the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for TRMM Products for details.

InputFileNames (Metadata):
InputFileNames contains a list of input file names for this granule. See Metadata for TRMM Products for details.

InputAlgorithmVersions (Metadata):
InputAlgorithmVersions contains a list of input algorithm versions for this granule. See

Metadata for TRMM Products for details.

InputGenerationDateTimes (Metadata):
InputGenerationDateTimes contains a list of input generation datetimes. See Metadata for TRMM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for TRMM Products for details.

Grid (Grid)

GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for TRMM Products for details.

precipitation (4-byte float, array size: nlat x nlon):
This is the merged microwave/IR precipitation estimate at each 0.25^o x 0.25^o box. Values range from 0 to 100 mm/hr. Special values are defined as:
-9999.9 Missing value

relativeError (4-byte float, array size: nlat x nlon):
This is the merged microwave/IR precipitation relative error estimate at each 0.25^o x 0.25^o box. Values range from 0 to 100 mm/hr. Special values are defined as:
-9999.9 Missing value

satPrecipitationSource (2-byte integer, array size: nlat x nlon):
Flag to show source of data at each 0.25^o x 0.25^o box. Note: "Conical scanner" includes TMI, AMSR, SSMI, and SSMI/S. Flag values are:

```
0   no observation
1   AMSU
2   TMI
3   AMSR
4   SSMI
5   SSMI/S
6   MHS
7   TCI
30  AMSU/MHS average
31  Conical scanner average
50  IR
Add 100 to above if sampling is less than or equal to two pixels
```

HQprecipitation (4-byte float, array size: nlat x nlon):
This is the pre-gauge-adjusted microwave precipitation estimate at each 0.25^o x 0.25^o box. Values range from 0 to 100 mm/hr. Special values are defined as:
-9999.9 Missing value

IRprecipitation (4-byte float, array size: nlat x nlon):
This is the pre-gauge-adjusted infrared precipitation estimate at each 0.25^o x 0.25^o box. Values range from 0 to 100 mm/hr. Special values are defined as:
-9999.9 Missing value

satObservationTime (1-byte integer, array size: nlat x nlon):
The satellite observation time minus the time of the granule at each 0.25^o x 0.25^o box. In case of overlapping satellite observations, the two or more observation times are equal-weighting averaged. Values range from -90 to 90 minutes. Special values are defined as:
-99 Missing value

C Structure Header file:

```
#ifndef _TK_3B42_H_
#define _TK_3B42_H_

#ifndef _L3B42_GRID_
#define _L3B42_GRID_

typedef struct {
    float precipitation[1440][400];
    float relativeError[1440][400];
    short satPrecipitationSource[1440][400];
    float HQprecipitation[1440][400];
    float IRprecipitation[1440][400];
    signed char satObservationTime[1440][400];
} L3B42_GRID;

#endif

#endif
```

Fortran Structure Header file:

```
STRUCTURE /L3B42_GRID/
    REAL*4 precipitation(400,1440)
    REAL*4 relativeError(400,1440)
    INTEGER*2 satPrecipitationSource(400,1440)
    REAL*4 HQprecipitation(400,1440)
    REAL*4 IRprecipitation(400,1440)
    BYTE satObservationTime(400,1440)
END STRUCTURE
```

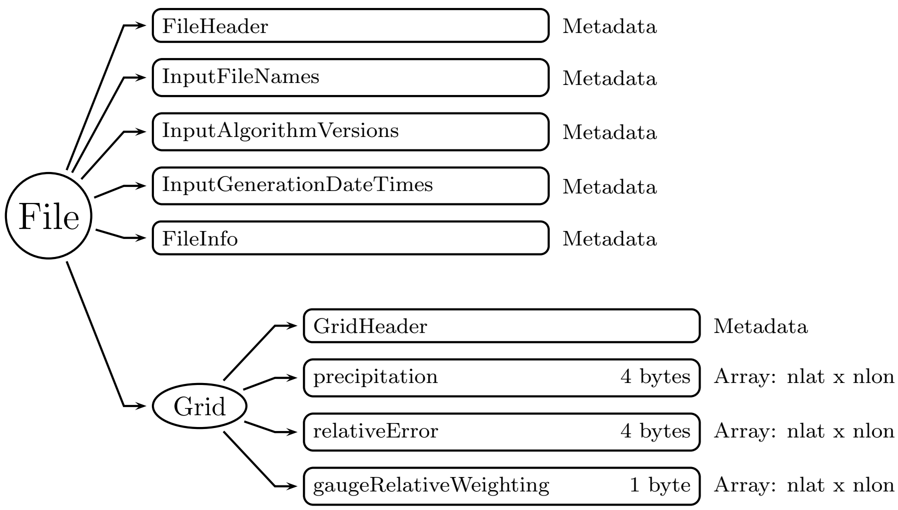


Figure 72: Data Format Structure for 3B43, TRMM and Other Sensors - Monthly

7.16 3B43 - TRMM and Other Sensors - Monthly

3B43, "TRMM and Other Sensors - Monthly", provides a "best" precipitation estimate in a latitude band covering 50^o N to 50^o S, an expansion of the TRMM region, from all global data sources, namely high-quality microwave data, infrared data, and analyses of rain gauges. The granule size is one month. The following sections describe the structure and contents of the format.

Dimension definitions:

nlat	400	Number of 0.25^o grid intervals of latitude from 50^o N to 50^o S.
nlon	1440	Number of 0.25^o grid intervals of longitude from 180^o W to 180^o E.

Figure 72 shows the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for TRMM Products for details.

InputFileNames (Metadata):
InputFileNames contains a list of input file names for this granule. See Metadata for TRMM Products for details.

InputAlgorithmVersions (Metadata):
InputAlgorithmVersions contains a list of input algorithm versions for this granule. See Metadata for TRMM Products for details.

InputGenerationDateTimes (Metadata):
InputGenerationDateTimes contains a list of input generation datetimes. See Metadata for TRMM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for TRMM Products for details.

Grid (Grid)

GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for TRMM Products for details.

precipitation (4-byte float, array size: nlat x nlon):
This is the satellite/gauge precipitation estimate at each 0.25^o x 0.25^o box. Values range from 0 to 100 mm/hr. Special values are defined as:
-9999.9 Missing value

relativeError (4-byte float, array size: nlat x nlon):
This is the satellite/gauge relative error estimate at each 0.25^o x 0.25^o box. Values range from 0 to 100 mm/hr. Special values are defined as:
-9999.9 Missing value

gaugeRelativeWeighting (1-byte integer, array size: nlat x nlon):
TBD Values range from 0 to 100 percent. Special values are defined as:
-99 Missing value

C Structure Header file:

```
#ifndef _TK_3B43_H_
#define _TK_3B43_H_

#ifndef _L3B43_GRID_
#define _L3B43_GRID_

typedef struct {
    float precipitation[1440][400];
    float relativeError[1440][400];
    signed char gaugeRelativeWeighting[1440][400];
} L3B43_GRID;

#endif

#endif
```

Fortran Structure Header file:

```
STRUCTURE /L3B43_GRID/
```

```
    REAL*4 precipitation(400,1440)
    REAL*4 relativeError(400,1440)
    BYTE gaugeRelativeWeighting(400,1440)
END STRUCTURE
```

7.17 2N25 - PR Profile

2N25, "Nadir Only PR Profile", is a simple subset of the PR 2A25 product described in ICS Vol 4. 2N25 contains the same variables as 2A25 but only data from the nadir ray of the PR instrument is present. All 2A25 variables with array indices including 49 (from the 49 PR rays) are reduced to 1. The nadir ray of PR is index 24 in C notation (zero-based) and 25 in FORTRAN. See the 2A25 documentation for an explanation of all variables. The following sections describe the structure and contents of the format.

Dimension definitions:

nscan	var	Number of scans in the granule.
nray	1	Number of pixels in each scan.
ncell1	80	Number of radar range cells at which the rain rate is estimated. The cells range from 0 to 79. Each cell is 250m apart, with cell 79 at the earth ellipsiod.
ncell2	5	Number of radar range cells at which the Z-R parameters are output.
nmeth	2	Number of methods used.
nestmeth	6	Number of estimation methods.

Figure 73 through Figure 79 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for TRMM Products for details.

InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for TRMM Products for details.

NavigationRecord (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for TRMM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for TRMM Products for details.

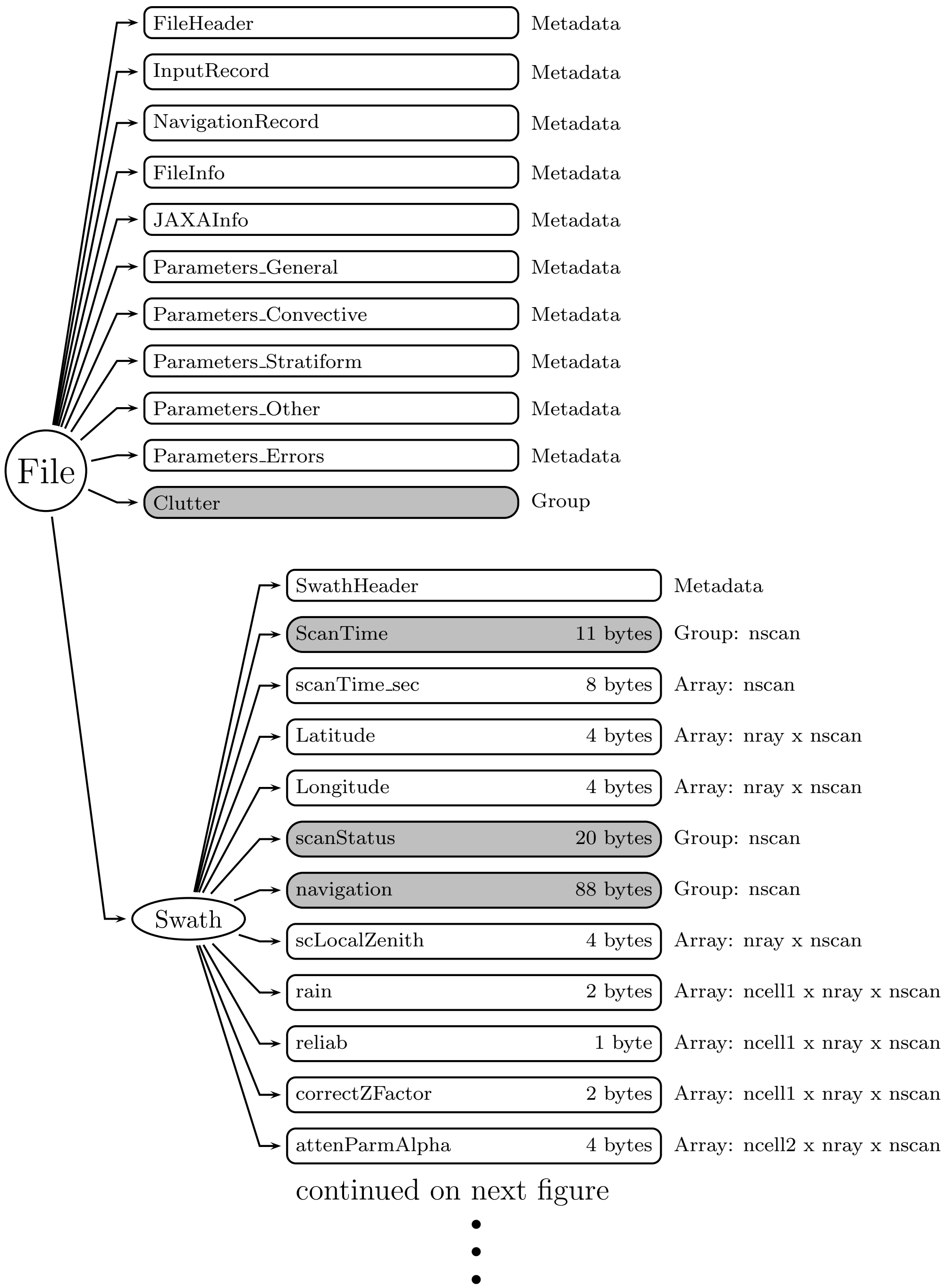


Figure 73: Data Format Structure for 2N25, PR Profile

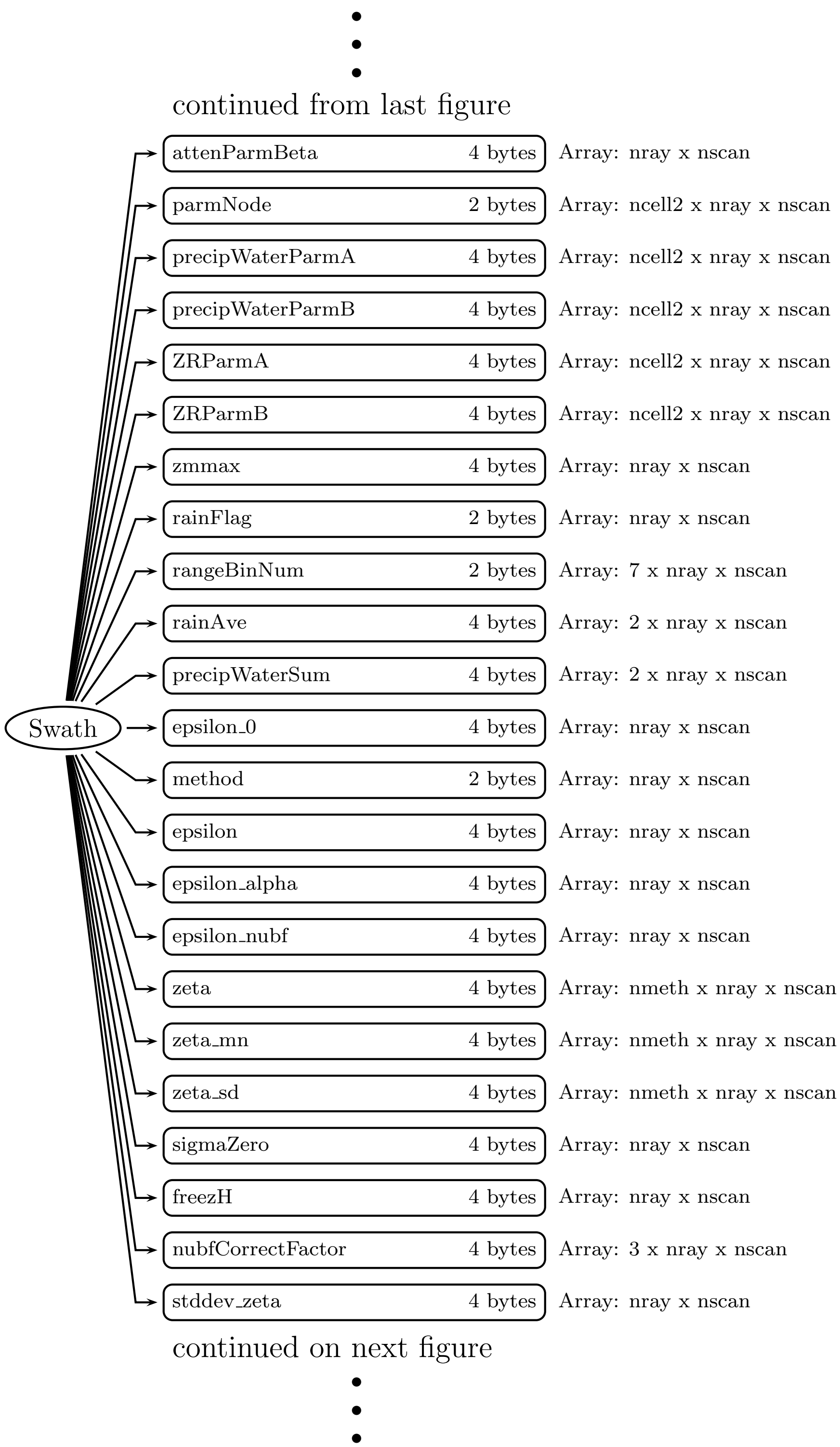


Figure 74: Data Format Structure for 2N25, PR Profile

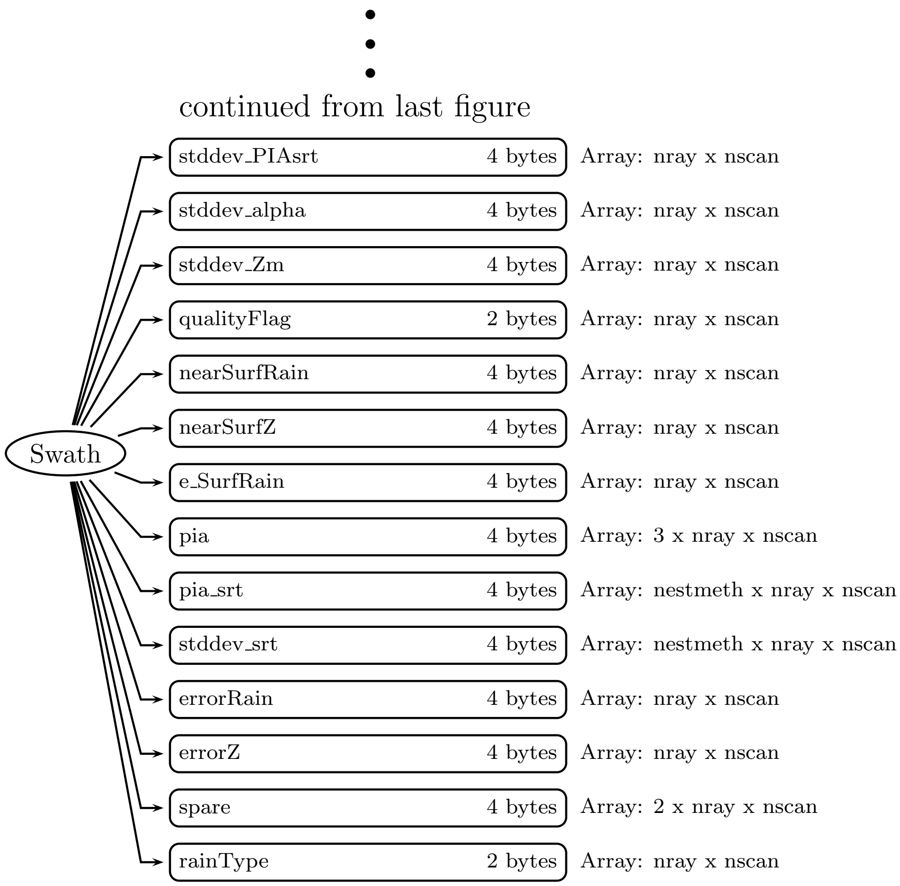


Figure 75: Data Format Structure for 2N25, PR Profile

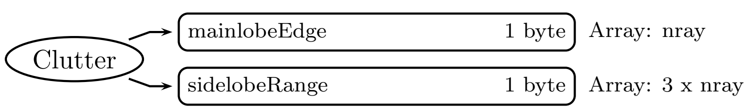


Figure 76: Data Format Structure for 2N25, Clutter

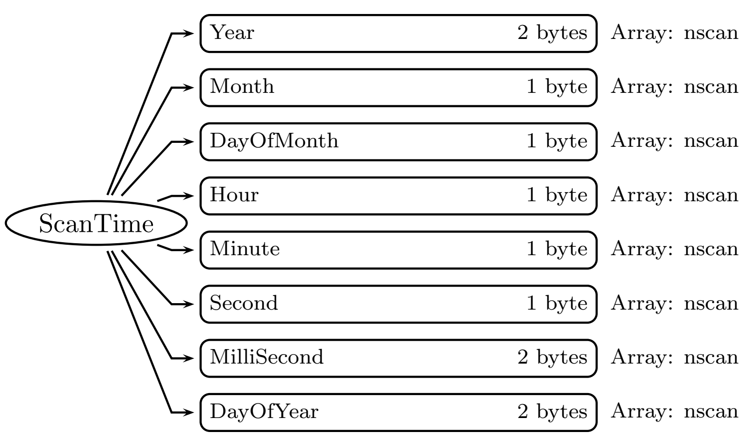


Figure 77: Data Format Structure for 2N25, ScanTime

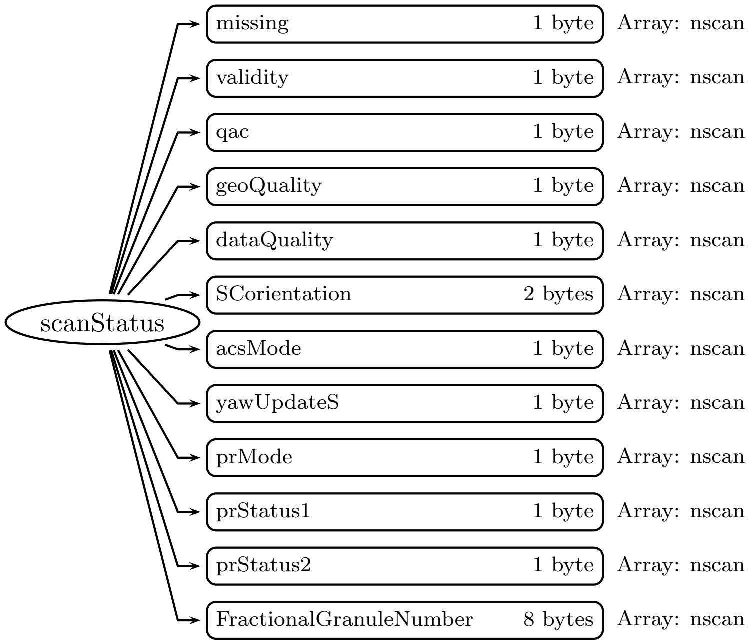


Figure 78: Data Format Structure for 2N25, scanStatus

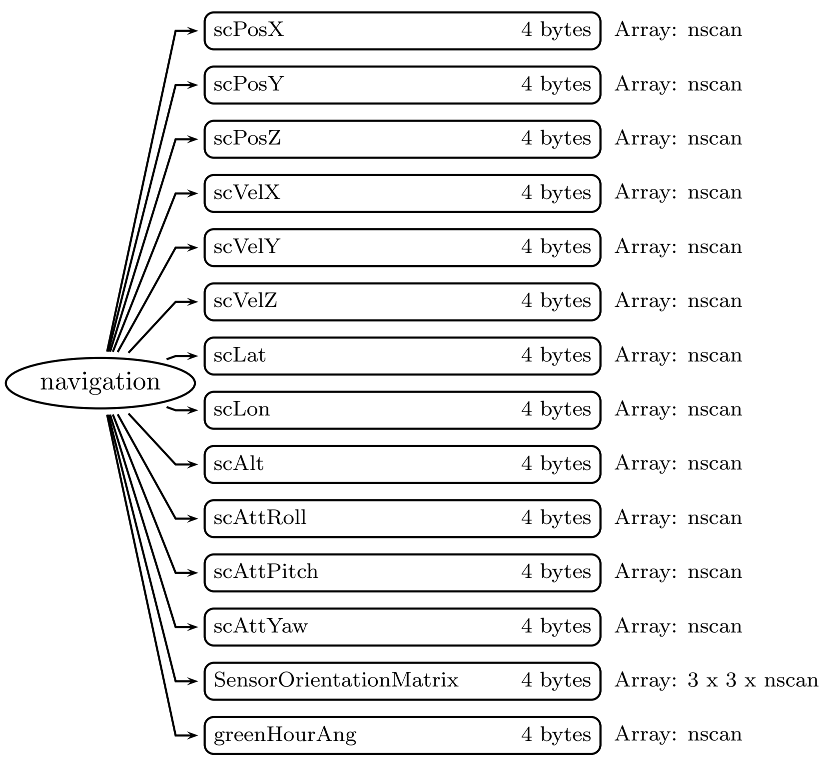


Figure 79: Data Format Structure for 2N25, navigation

JAXAInfo (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by PR algorithms only. See Metadata for TRMM Products for details.

Parameters_General (Metadata):
ASCII text of the general parameters used by 2A25 at runtime.

Parameters_Convective (Metadata):
ASCII text of the parameters for Convective rain used by 2A25 at runtime.

Parameters_Stratiform (Metadata):
ASCII text of the parameters for Stratiform rain used by 2A25 at runtime.

Parameters_Other (Metadata):
ASCII text of the parameters for Other rain used by 2A25 at runtime.

Parameters_Errors (Metadata):
ASCII text of the Error parameters used by 2A25 at runtime.

Clutter (Group)

mainlobeEdge (1-byte integer, array size: nray):
Absolute value of the difference in Range Bin Numbers between the detected surface and the edge of the clutter from the mainlobe.

sidelobeRange (1-byte integer, array size: 3 x nray):
Absolute value of the difference in Range Bin Numbers between the detected surface and the clutter psition from the sidelobe. A zero means no clutter indicated in this field since less than 3 bins contained significant clutter.

Swath (Swath)

SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for TRMM Products for details.

ScanTime (Group)

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

scanTime_sec (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: nray x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: nray x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scanStatus (Group)

missing (1-byte integer, array size: nscan):
Missing indicates whether information is contained in the scan data. The values are:

```
0 Scan data elements contain information
1 Scan was missing in the telemetry data
2 Scan data contains no elements with rain
```

validity (1-byte integer, array size: nscan):
Validity is a summary of status modes. If all status modes are routine, all bits in Validity = 0. Routine means that scan data has been measured in the normal operational situation

as far as the status modes are concerned. Validity does not assess data or geolocation quality. Validity is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

```
Bit Meaning if bit = 1
0   Spare (always 0)
1   Non-routine spacecraft orientation (2 or 3)
2   Non-routine ACS mode (other than 4)
3   Non-routine yaw update status (0 or 1)
4   Non-routine instrument status (other than 1)
5   Non-routine QAC (non-zero)
6   Spare (always 0)
7   Spare (always 0)
```

qac (1-byte integer, array size: nscan):
The Quality and Accounting Capsule of the Science packet as it appears in Level-0 data. If no QAC is given in Level-0, which means no decoding errors occurred, QAC in this format has a value of zero.

geoQuality (1-byte integer, array size: nscan):
Geolocation quality is a summary of geolocation quality in the scan. A zero integer value indicates 'good' geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

```
Bit Meaning if bit = 1
0   latitude limit error
1   geolocation
2   attitude change rate limit error
3   attitude limit error
4   satellite undergoing maneuvers
5   using predictive orbit data
6   geolocation calculation error
7   not used
```

dataQuality (1-byte integer, array size: nscan):
Data quality is a summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

```
Bit Meaning if bit = 1
0   missing
5   Geolocation Quality is not normal
6   Validity is not normal
```

SCorientation (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the TMI scan. If +X is forward, SCorientation is 0. If -X is forward, SCorientation is 180. If -Y is forward, SCorientation is 90. Values range from 0 to 360 degrees. Special values are defined as:

-8003	Inertial
-8004	Unknown
-9999	Missing value

acsMode (1-byte integer, array size: nscan):

Value	Meaning
0	Standby
1	Sun Acquire
2	Earth Acquire
3	Yaw Acquire
4	Nominal
5	Yaw Maneuver
6	Delta-H (Thruster)
7	Delta-V (Thruster)
8	CERES Calibration

yawUpdateS (1-byte integer, array size: nscan):

Value	Meaning
0	Inaccurate
1	Indeterminate
2	Accurate

prMode (1-byte integer, array size: nscan):

Value	Meaning
1	Observation Mode
2	Other Mode

prStatus1 (1-byte integer, array size: nscan):
This status is a warning for scan data. Unless this is 0, the scan data may include a little questionable value though it is not a problem (such as break of caution limit). This field is used only for NASDA's data analysis.

prStatus2 (1-byte integer, array size: nscan):
Initialization in Onboard Surface Search Algorithm.

```
Value Meaning
0     Not initialized
1     Initialized
```

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft's trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

navigation (Group)

scPosX (4-byte float, array size: nscan):
The x component of the position (m) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Geocentric Inertial Coordinates are also commonly known as Earth Centered Inertial coordinates. These coordinates will be True of Date (rather than Epoch 2000 which are also commonly used), as interpolated from the data in the Flight Dynamics Facility ephemeris files generated for TRMM.

scPosY (4-byte float, array size: nscan):
The y component of the position (m) of the spacecraft in Geocentric Inertial Coordinates. See scPosX.

scPosZ (4-byte float, array size: nscan):
The z component of the position (m) of the spacecraft in Geocentric Inertial Coordinates. See scPosX.

scVelX (4-byte float, array size: nscan):
The x component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scVelY (4-byte float, array size: nscan):
The y component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scVelZ (4-byte float, array size: nscan):
The z component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scLat (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time.

scLon (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time.

scAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time.

scAttRoll (4-byte float, array size: nscan):
The satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

scAttPitch (4-byte float, array size: nscan):
The satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

scAttYaw (4-byte float, array size: nscan):
The satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates.

SensorOrientationMatrix (4-byte float, array size: 3 x 3 x nscan):
SensorOrientationMatrix is the rotation matrix from the instrument coordinate frame to Geocentric Inertial Coordinates at the Scan mid-Time. It is unitless.

greenHourAng (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates.

scLocalZenith (4-byte float, array size: nray x nscan):
The angle, in degrees, between the local zenith and the PR ray center line. The local (geodetic) zenith at the intersection of the ray and the earth ellipsoid is used. Ranges from - 30.0 to +30.0. This is an exact copy of the satellite local zenith angle in the 1C21 product.

rain (2-byte integer, array size: ncell1 x nray x nscan):
This is the estimate of rain rate at the radar range gates from 0 to 20 km along the slant range. It ranges from 0.0 to 300.0 mmh-1 and is multiplied by 100 and stored as a 2-byte integer. A value of -88.88 mm/hr (stored as -889) means ground clutter.

reliab (1-byte integer, array size: ncell1 x nray x nscan):
The Reliability is that for estimated rain rates at the radar range gates from 0 to 20 km. It ranges from 0 to 255. If data are missing, the reliability will be set as 10000000 in binary. The default value is 0 (measured signal below noise). Bit 0 is the least significant bit (i.e., if bit i =1 and other bits =0, the unsigned integer value is 2**i). The following meanings are assigned to each bit in the 8-bit integer if the bit = 1.

```
bit 0   rain possible
bit 1   rain certain
bit 2   bright band
bit 3   large attenuation
bit 4   weak return (Zm less than 20 dBZ)
bit 5   estimated Z less than 0 dBZ
bit 6   main-lobe clutter or below surface
bit 7   missing data
```

correctZFactor (2-byte integer, array size: ncell1 x nray x nscan):
This is the attenuation corrected reflectivity factor (Z) at the radar range gates from 0 to 20 km along the slant range. It ranges from 0.0 to 80.0 dBZ and is multiplied by 100 and stored as a 2-byte integer. Values of reflectivity less than 0.0 dBZ are set to 0.0 dBZ. A value of -88.88 dB (stored as -8888) is a ground clutter flag.

attenParmAlpha (4-byte float, array size: ncell2 x nray x nscan):
The attenuation parameter Alpha (?) relates the attenuation coefficient, k (dB/km) to the Z-factor: k = ?Z?. ? is computed at ncell2 radar range gates for each ray. It ranges from 0.000100 to 0.002000.

attenParmBeta (4-byte float, array size: nray x nscan):

parmNode (2-byte integer, array size: ncell2 x nray x nscan):
The Parameter Node gives the range bin numbers of the nodes at which the values of Attenuation and Z-R Parameters are given (see below). The values of the parameters

between the nodes are linearly interpolated. This variable ranges from 0 and 79 and is unitless.

precipWaterParmA (4-byte float, array size: ncell2 x nray x nscan):
Parameter A in the M = AZ^B relationship. A is computed at each node (ncell2) for each ray. It ranges from ? to ?

precipWaterParmB (4-byte float, array size: ncell2 x nray x nscan):
Parameter B in the M = AZ^B relationship. B is computed at each node (ncell2) for each ray. It ranges from ? to ?

ZRParmA (4-byte float, array size: ncell2 x nray x nscan):
Parameter a for Z-R relationship (R=aZb) is determined from the rain type and the height relative to the freezing level, the non-uniformity parameter (?) and the correction factor (?) for the surface reference technique. a is computed at ncell2 radar range gates for each ray. It ranges from 0.0050 to 0.2000.

ZRParmB (4-byte float, array size: ncell2 x nray x nscan):
Parameter b for Z-R relationship (R=aZb) is determined from the rain type and the height relative to the freezing level, the non-uniformity parameter (?) and the correction factor (?) for the surface reference technique. b is computed at ncell2 radar range gates for each ray. It ranges from 0.500 to 1.000.

zmmax (4-byte float, array size: nray x nscan):
This is the maximum value of measured reflectivity factor at each IFOV. It ranges from 0.0 to 100.0 dBZ.

rainFlag (2-byte integer, array size: nray x nscan):
The Rain Flag indicates rain or no rain status and the rain type assumed in rain rate retrieval. The default value is 0 (no rain). Bit 0 is the least significant bit (i.e., if bit i=1 and other bits =0, the unsigned integer value is 2**i). The following meanings are assigned to each bit in the 16-bit integer if the bit = 1.

```
bit 0    rain possible
bit 1    rain certain
bit 2    zeta**beta greater then 0.5
         (Path Integrated Attenuation (PIA) larger than 3 dB)
bit 3    large attenuation (PIA larger than 10 dB)
bit 4    stratiform
bit 5    convective
bit 6    bright band exists
bit 7    warm rain
bit 8    rain bottom above 2 km
bit 9    rain bottom above 4 km
bit 10   not used
bit 11   not used
bit 12   not used
bit 13   not used
```

```
bit 14   data missing between rain top and bottom
bit 15   not used
```

rangeBinNum (2-byte integer, array size: 7 x nray x nscan):
This array gives the Range Bin Number of various quantities for each ray in every scan. The definitions are:

```
- top range bin number of the interval that is processed
  as meaningful data in 2A-25
- bottom range bin number of the interval that is processed
  as meaningful data in 2A-25
- actual surface range bin number (can be larger than 79)
- range bin number of the bright band if it exits
- range bin number at which the path-integrated Z-factor first
  exceeds the given threshold
- range bin number at which the measured Z-factor is maximum
- range bin number of near surface bin
```

The Range Bin Numbers in this algorithm are different from the NASDA definition of Range Bin Number described in the ICS, Volume 3. The Range Bin Numbers in the algorithm range from 0 to 82 and have an interval of 250m. The earth ellipsoid is defined as range bin 79.

rainAve (4-byte float, array size: 2 x nray x nscan):
There are two kinds of Average Rain Rate. The first one is the average rain rate for each ray between the two predefined heights of 2 and 4 km. It ranges from 0.0 to 3000.0 mm h-1. The second one is the integral of rain rate from rain top to rain bottom. It ranges from 0.0 to 300 mm h-1 km.

precipWaterSum (4-byte float, array size: 2 x nray x nscan):
TBD dim 2??? The sum of the precipitation water content calculated from Ze at each range bin. The summation is from the rain top to the actual surface. This includes both liquid and solid phase regions. Units are g km/m^3 and it ranges from 0.0 to 50.0.

epsilon_0 (4-byte float, array size: nray x nscan):
The adjustment parameter computed from the filtered surface reference PIA (2A21). Unitless and it ranges from 0.0 to 100.0.

method (2-byte integer, array size: nray x nscan):
This flag indicates which method is used to derive the rain rate. The default value is 0 (including no rain case). Bit 0 is the least significant bit (i.e., if bit i =1 and other bits =0, the unsigned integer value is 2**i).

The default value is 0 (including no rain case). The following meanings are assigned to each bit in the 16-bit integer.

```
       0: (bit  1)   no rain
    if rain
       0: (bit  1)   over ocean
       1: (bit  1)   over land
       2: (bit  2)   over coast, river, etc.
       3: (bit  2)   others (impossible)
      +4: (bit  3)   PIA from constant-Z-near-surface assumption
      +8: (bit  4)   spatial reference
     +16: (bit  5)   temporal reference
     +32: (bit  6)   global reference
     +64: (bit  7)   hybrid reference
    +128: (bit  8)   good to take statistics of epsilon.
    +256: (bit  9)   HB method used, SRT totally ignored
    +512: (bit 10)   very large pia\_srt for given zeta
   +1024: (bit 11)   very small pia\_srt for given zeta
   +2048: (bit 12)   no ZR adjustment by epsilon
   +4096: (bit 13)   no NUBF correction because NSD unreliable
   +8192: (bit 14)   surface attenuation > 60 dB
  +16384: (bit 15)   data partly missing between rain top and bottom
```

epsilon (4-byte float, array size: nray x nscan):
The Epsilon (?) is the final correction factor applied to the assumed drop size distribution. Unitless and it ranges from 0.0 to 100.0.

epsilon_alpha (4-byte float, array size: nray x nscan):
TBD.

epsilon_nubf (4-byte float, array size: nray x nscan):
TBD

zeta (4-byte float, array size: nmeth x nray x nscan):
The Zeta (?) roughly represents the rain rate integrated along the ray using two different methods. It ranges from 0.0 to 100.0 and is unitless.

zeta_mn (4-byte float, array size: nmeth x nray x nscan):
Zeta_mn (?mn) is the average of zeta (?) in the vicinity of each beam position (average over three scans and three IFOVs). It is calculated using two methods. It ranges from 0.0 to 100.0 and is unitless.

zeta_sd (4-byte float, array size: nmeth x nray x nscan):
Zeta_sd (?sd) is the standard deviation of zeta (?) in the vicinity of each beam position (using three scans and three IFOVs). It is calculated using two methods. It ranges from 0.0 to 100.0 and is unitless.

sigmaZero (4-byte float, array size: nray x nscan):
The Sigma-zero is the normalized surface cross section. It ranges from -50.00 to 20.00 dB. This field is copied from the 2A21 product file.

freezH (4-byte float, array size: nray x nscan):
A positive Height of Freezing Level is the height of the 0C isotherm above mean sea level in meters, estimated from climatological surface temperature data. This field is copied from the 2A23 product. Negative values are defined as:

```
-5555:  When error occurred in the estimation of Height of Freezing Level
-8888:  No rain
-9999:  Data missing
```

nubfCorrectFactor (4-byte float, array size: 3 x nray x nscan):
The Non-Uniform Beam Filling (NUBF) Correction Factor is used as a correction to reflectivity and attenuation calculations. The two NUBF Correction Factors are given for the k-Z and Z-R relations. The ranges are 1.0 to 3.0 and 0.8 to 1.0, respectively. Both are unitless quantities.

stddev_zeta (4-byte float, array size: nray x nscan):
TBD

stddev_PIAsrt (4-byte float, array size: nray x nscan):
TBD

stddev_alpha (4-byte float, array size: nray x nscan):
TBD

stddev_Zm (4-byte float, array size: nray x nscan):
TBD

qualityFlag (2-byte integer, array size: nray x nscan):
This is a quality flag and ranges from 0 to 32767. The default value is 0 (normal). Bit 0 is the least significant bit (i.e., if bit i =1 and other bits =0, the unsigned integer value is 2**i). The following meanings are assigned to each bit in the 16-bit integer if the bit = 1.

```
    0:   normal
   +1:   unusual situation in rain average
   +2:   NSD of zeta (xi) calculated from less than 6 points
   +4:   NSD of PIA calculated from less than 6 points
   +8:   NUBF for Z-R below lower bound
  +16:   NUBF for PIA above upper bound
  +32:   epsilon not reliable, epsi\_sig less than or equal to 0.0
  +64:   2A21 input data not reliable
 +128:   2A23 input data not reliable
 +256:   range bin error
 +512:   sidelobe clutter removal
+1024:   probability=0 for all tau
+2048:   pia\_surf\_ex less than or equal to 0.0
+4096:   const Z is invalid
+8192:   reliabFactor in 2A21 is NaN
```

```
+16384:   data missing

15th bit (sign bit) is not used
```

nearSurfRain (4-byte float, array size: nray x nscan):
Rainfall rate near the surface. The range is 0 to 3000 mm/hr. A value of -99.99 mm/hr. is a missing flag.

nearSurfZ (4-byte float, array size: nray x nscan):
Reflectivity near the surface. The range is 0.0 to 100.0 dBZ. A value of -99.99 dBZ is a missing flag.

e_SurfRain (4-byte float, array size: nray x nscan):
The rainfall estimate at the true (detected) surface bin. Units are mm/hr and ranges from 0 to 3000 mm/hr. A value of -99.99mm/hr is a missing flag.

pia (4-byte float, array size: 3 x nray x nscan):
Path integrated attenuation (PIA) (two-way) estimates for three cases:

```
- The final adjusted PIA estimate
- The difference between the PIA at the surface and near surface range bins
- The PIA estimate from 2A21
```

The units are dB.

pia_srt (4-byte float, array size: nestmeth x nray x nscan):
Path integrated attenuation estimates from 2A21. First index is the method of estimation:

```
0 - best estimate (copy of 2A21 pathAtten)
1 - spatial, forward
2 - hybrid, forward (ocean only)
3 - spatial, backward
4 - hybrid, backward (ocean only)
5 - temporal (should be the same forward or backward)
```

Values range from -50 to 50 dB. Special values are defined as:
-9999.9 Missing value

stddev_srt (4-byte float, array size: nestmeth x nray x nscan):
Standard deviation of the SRT PIA estimates computed from the SRT PIA and the 2A21 reliability factor. First index is the method of estimation:

```
0 - best estimate (copy of 2A21 pathAtten)
1 - spatial, forward
2 - hybrid, forward (ocean only)
3 - spatial, backward
4 - hybrid, backward (ocean only)
5 - temporal (should be the same forward or backward)
```

Values range from -50 to 50 dB. Special values are defined as:
-9999.9 Missing value

errorRain (4-byte float, array size: nray x nscan):
The error in Near Surface Rain Rate. The units are dB.

errorZ (4-byte float, array size: nray x nscan):
The error in Near Surface Z. The range is 0.0 to 100.0 dBZ.

spare (4-byte float, array size: 2 x nray x nscan):
Contents and ranges are not public.

rainType (2-byte integer, array size: nray x nscan):
This is a copy of the 2A23 Rain Type field, See 2A23 description.

C Structure Header file:

```
#ifndef _TK_2N25_H_
#define _TK_2N25_H_

#ifndef _L2N25_NAVIGATION_
#define _L2N25_NAVIGATION_

typedef struct {
    float scPosX;
    float scPosY;
    float scPosZ;
    float scVelX;
    float scVelY;
    float scVelZ;
    float scLat;
    float scLon;
    float scAlt;
    float scAttRoll;
    float scAttPitch;
    float scAttYaw;
    float SensorOrientationMatrix[3][3];
    float greenHourAng;
} L2N25_NAVIGATION;

#endif

#ifndef _L2N25_SCANSTATUS_
#define _L2N25_SCANSTATUS_

typedef struct {
```

```
    signed char missing;
    signed char validity;
    signed char qac;
    signed char geoQuality;
    signed char dataQuality;
    short SCorientation;
    signed char acsMode;
    signed char yawUpdateS;
    signed char prMode;
    signed char prStatus1;
    signed char prStatus2;
    double FractionalGranuleNumber;
} L2N25_SCANSTATUS;

#endif

#ifndef _L2N25_SCANTIME_
#define _L2N25_SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
} L2N25_SCANTIME;

#endif

#ifndef _L2N25_SWATH_
#define _L2N25_SWATH_

typedef struct {
    L2N25_SCANTIME ScanTime;
    double scanTime_sec;
    float Latitude[1];
    float Longitude[1];
    L2N25_SCANSTATUS scanStatus;
    L2N25_NAVIGATION navigation;
    float scLocalZenith[1];
```

```
    float rain[1][80];
    signed char reliab[1][80];
    float correctZFactor[1][80];
    float attenParmAlpha[1][5];
    float attenParmBeta[1];
    short parmNode[1][5];
    float precipWaterParmA[1][5];
    float precipWaterParmB[1][5];
    float ZRParmA[1][5];
    float ZRParmB[1][5];
    float zmmax[1];
    short rainFlag[1];
    short rangeBinNum[1][7];
    float rainAve[1][2];
    float precipWaterSum[1][2];
    float epsilon_0[1];
    short method[1];
    float epsilon[1];
    float epsilon_alpha[1];
    float epsilon_nubf[1];
    float zeta[1][2];
    float zeta_mn[1][2];
    float zeta_sd[1][2];
    float sigmaZero[1];
    float freezH[1];
    float nubfCorrectFactor[1][3];
    float stddev_zeta[1];
    float stddev_PIAsrt[1];
    float stddev_alpha[1];
    float stddev_Zm[1];
    short qualityFlag[1];
    float nearSurfRain[1];
    float nearSurfZ[1];
    float e_SurfRain[1];
    float pia[1][3];
    float pia_srt[1][6];
    float stddev_srt[1][6];
    float errorRain[1];
    float errorZ[1];
    float spare[1][2];
    short rainType[1];
} L2N25_SWATH;
```

```
#endif

#ifndef _L2N25_CLUTTER_
#define _L2N25_CLUTTER_

typedef struct {
    signed char mainlobeEdge[1];
    signed char sidelobeRange[1][3];
} L2N25_CLUTTER;

#endif

#endif
```

Fortran Structure Header file:

```
STRUCTURE /L2N25_NAVIGATION/
    REAL*4 scPosX
    REAL*4 scPosY
    REAL*4 scPosZ
    REAL*4 scVelX
    REAL*4 scVelY
    REAL*4 scVelZ
    REAL*4 scLat
    REAL*4 scLon
    REAL*4 scAlt
    REAL*4 scAttRoll
    REAL*4 scAttPitch
    REAL*4 scAttYaw
    REAL*4 SensorOrientationMatrix(3,3)
    REAL*4 greenHourAng
END STRUCTURE

STRUCTURE /L2N25_SCANSTATUS/
    BYTE missing
    BYTE validity
    BYTE qac
    BYTE geoQuality
    BYTE dataQuality
    INTEGER*2 SCorientation
    BYTE acsMode
    BYTE yawUpdateS
    BYTE prMode
```

```
    BYTE prStatus1
    BYTE prStatus2
    REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /L2N25_SCANTIME/
    INTEGER*2 Year
    BYTE Month
    BYTE DayOfMonth
    BYTE Hour
    BYTE Minute
    BYTE Second
    INTEGER*2 MilliSecond
    INTEGER*2 DayOfYear
END STRUCTURE

STRUCTURE /L2N25_SWATH/
    RECORD /L2N25_SCANTIME/ ScanTime
    REAL*8 scanTime_sec
    REAL*4 Latitude1
    REAL*4 Longitude1
    RECORD /L2N25_SCANSTATUS/ scanStatus
    RECORD /L2N25_NAVIGATION/ navigation
    REAL*4 scLocalZenith1
    REAL*4 rain(801)
    BYTE reliab(801)
    REAL*4 correctZFactor(801)
    REAL*4 attenParmAlpha(51)
    REAL*4 attenParmBeta1
    INTEGER*2 parmNode(51)
    REAL*4 precipWaterParmA(51)
    REAL*4 precipWaterParmB(51)
    REAL*4 ZRParmA(51)
    REAL*4 ZRParmB(51)
    REAL*4 zmmax1
    INTEGER*2 rainFlag1
    INTEGER*2 rangeBinNum(71)
    REAL*4 rainAve(21)
    REAL*4 precipWaterSum(21)
    REAL*4 epsilon_01
    INTEGER*2 method1
    REAL*4 epsilon1
    REAL*4 epsilon_alpha1
```

```
    REAL*4 epsilon_nubf1
    REAL*4 zeta(21)
    REAL*4 zeta_mn(21)
    REAL*4 zeta_sd(21)
    REAL*4 sigmaZero1
    REAL*4 freezH1
    REAL*4 nubfCorrectFactor(31)
    REAL*4 stddev_zeta1
    REAL*4 stddev_PIAsrt1
    REAL*4 stddev_alpha1
    REAL*4 stddev_Zm1
    INTEGER*2 qualityFlag1
    REAL*4 nearSurfRain1
    REAL*4 nearSurfZ1
    REAL*4 e_SurfRain1
    REAL*4 pia(31)
    REAL*4 pia_srt(61)
    REAL*4 stddev_srt(61)
    REAL*4 errorRain1
    REAL*4 errorZ1
    REAL*4 spare(21)
    INTEGER*2 rainType1
END STRUCTURE

STRUCTURE /L2N25_CLUTTER/
    BYTE mainlobeEdge1
    BYTE sidelobeRange(31)
END STRUCTURE
```

7.18 3G12 - TRAIN Gridded Orbital Heating

3G12, "TRAIN Gridded Orbital Heating", produces 0.5^o x 0.5^o monthly oceanic heating maps using TMI Level-1 data. The PI is Dr. William Olson. The following sections describe the structure and contents of the format.

Dimension definitions:

nlat	160	Number of 0.5^o grid intervals of latitude from 40^oN to 40^oS.
nlon	720	Number of 0.5^o grid intervals of longitude from 180^oW to 180^oE.
nlayer	19	Number of layers at 0.0-0.5 km, 0.5-1 km, 1-2 km, ..., 17-18 km.

Figure 80 through Figure 81 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

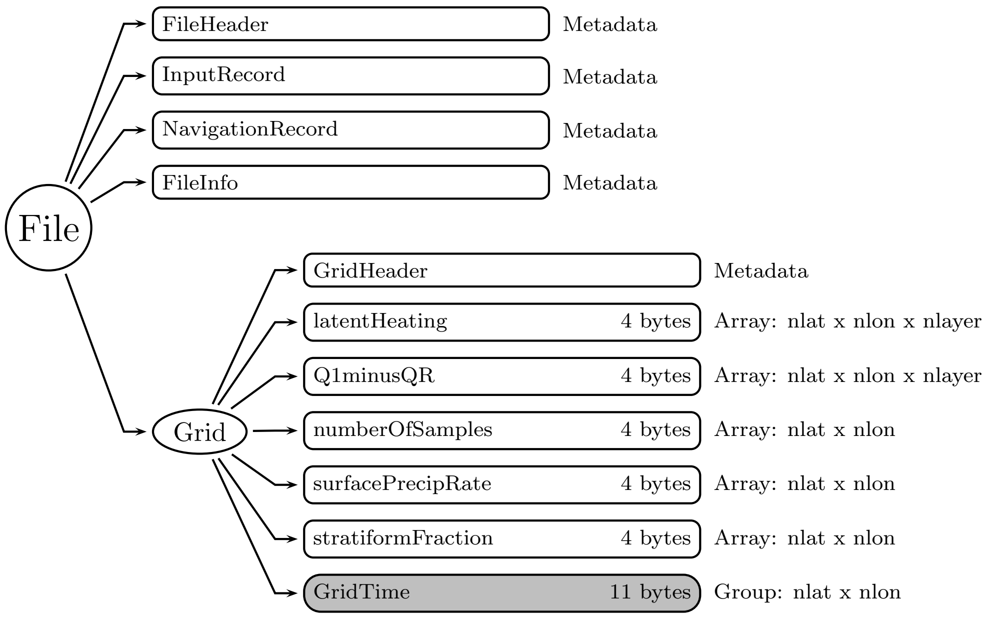


Figure 80: Data Format Structure for 3G12, TRAIN Gridded Orbital Heating

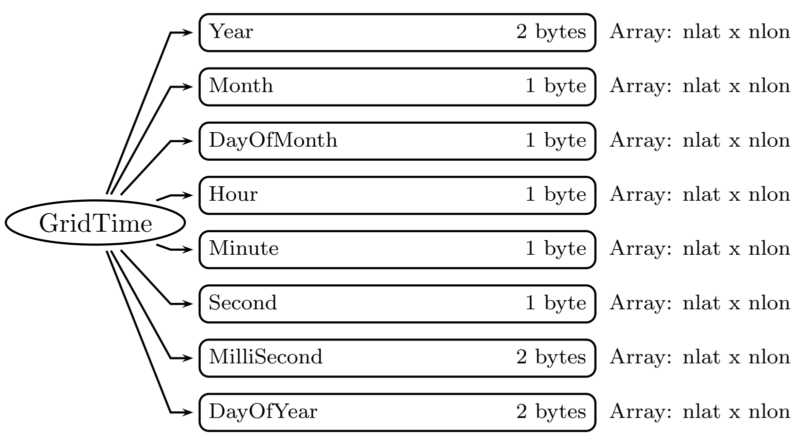


Figure 81: Data Format Structure for 3G12, GridTime

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for TRMM Products for details.

InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for TRMM Products for details.

NavigationRecord (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for TRMM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for TRMM Products for details.

Grid (Grid)

GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for TRMM Products for details.

latentHeating (4-byte float, array size: nlat x nlon x nlayer):
Monthly latent heating in precipitation over oceans in 0.5^o x 0.5^oboxes. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

Q1minusQR (4-byte float, array size: nlat x nlon x nlayer):
Monthly Q1 - QR heating (or latent + eddy heating) in precipitation over oceans in 0.5^o x 0.5^oboxes. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

numberOfSamples (4-byte integer, array size: nlat x nlon):
Number of samples in 0.5^o x 0.5^o boxes. Values range from 0 to 500000. Special values are defined as:
-9999 Missing value

surfacePrecipRate (4-byte float, array size: nlat x nlon):
Monthly surface precipitation rate over oceans from the TRAIN algorithm in 0.5^o x 0.5^oboxes. Values range from 0 to 3000 mm/hr. Special values are defined as:
-9999.9 Missing value

stratiformFraction (4-byte float, array size: nlat x nlon):
Monthly stratiform to total ratio of rainfall rate in precipitation over oceans from the TRAIN algorithm in 0.5^o x 0.5^oboxes. Values range from 0 to 100 percent. Special values are defined as:
-9999.9 Missing value

GridTime (Group)

Year (2-byte integer, array size: nlat x nlon):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nlat x nlon):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nlat x nlon):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nlat x nlon):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nlat x nlon):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nlat x nlon):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nlat x nlon):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nlat x nlon):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

C Structure Header file:

```
#ifndef _TK_3G12_H_
#define _TK_3G12_H_

#ifndef _L3G12_GRIDTIME_
#define _L3G12_GRIDTIME_

typedef struct {
    short Year[720][160];
    signed char Month[720][160];
    signed char DayOfMonth[720][160];
    signed char Hour[720][160];
```

```
    signed char Minute[720][160];
    signed char Second[720][160];
    short MilliSecond[720][160];
    short DayOfYear[720][160];
} L3G12_GRIDTIME;

#endif

#ifndef _L3G12_GRID_
#define _L3G12_GRID_

typedef struct {
    float latentHeating[19][720][160];
    float Q1minusQR[19][720][160];
    int numberOfSamples[720][160];
    float surfacePrecipRate[720][160];
    float stratiformFraction[720][160];
    L3G12_GRIDTIME GridTime;
} L3G12_GRID;

#endif

#endif
```

Fortran Structure Header file:

```
STRUCTURE /L3G12_GRIDTIME/
    INTEGER*2 Year(160,720)
    BYTE Month(160,720)
    BYTE DayOfMonth(160,720)
    BYTE Hour(160,720)
    BYTE Minute(160,720)
    BYTE Second(160,720)
    INTEGER*2 MilliSecond(160,720)
    INTEGER*2 DayOfYear(160,720)
END STRUCTURE

STRUCTURE /L3G12_GRID/
    REAL*4 latentHeating(160,720,19)
    REAL*4 Q1minusQR(160,720,19)
    INTEGER*4 numberOfSamples(160,720)
    REAL*4 surfacePrecipRate(160,720)
    REAL*4 stratiformFraction(160,720)
```

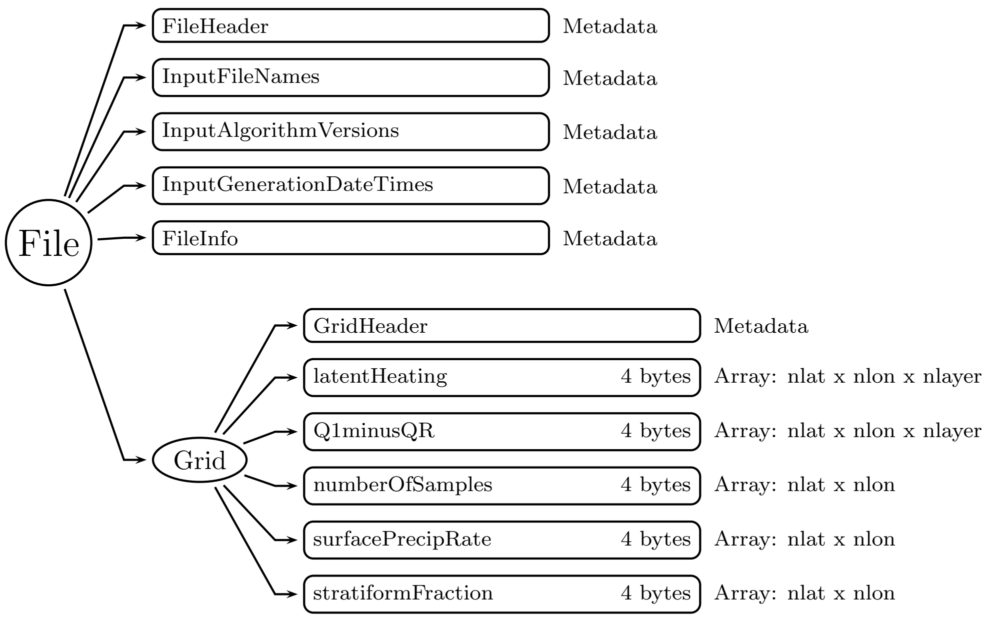


Figure 82: Data Format Structure for 3H12, TRAIN Monthly Heating

```
    RECORD /L3G12_GRIDTIME/ GridTime
END STRUCTURE
```

7.19 3H12 - TRAIN Monthly Heating

3H12, "TRAIN Monthly Heating", produces 0.5^o x 0.5^o monthly oceanic heating maps using TMI Level-1 data. The PI is Dr. William Olson. The following sections describe the structure and contents of the format.

Dimension definitions:

nlat	160	Number of 0.5^o grid intervals of latitude from 40^oN to 40^oS.
nlon	720	Number of 0.5^o grid intervals of longitude from 180^oW to 180^oE.
nlayer	19	Number of layers at 0.0-0.5 km, 0.5-1 km, 1-2 km, ..., 17-18 km.

Figure 82 shows the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for TRMM Products for details.

InputFileNames (Metadata):
InputFileNames contains a list of input file names for this granule. See Metadata for TRMM Products for details.

InputAlgorithmVersions (Metadata):
InputAlgorithmVersions contains a list of input algorithm versions for this granule. See Metadata for TRMM Products for details.

InputGenerationDateTimes (Metadata):
InputGenerationDateTimes contains a list of input generation datetimes. See Metadata for TRMM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for TRMM Products for details.

Grid (Grid)

GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for TRMM Products for details.

latentHeating (4-byte float, array size: nlat x nlon x nlayer):
Monthly latent heating in precipitation over oceans in 0.5^o x 0.5^oboxes. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

Q1minusQR (4-byte float, array size: nlat x nlon x nlayer):
Monthly Q1 - QR heating (or latent + eddy heating) in precipitation over oceans in 0.5^o x 0.5^oboxes. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

numberOfSamples (4-byte integer, array size: nlat x nlon):
Number of samples in 0.5^o x 0.5^o boxes for one month. Values range from 0 to 500000. Special values are defined as:
-9999 Missing value

surfacePrecipRate (4-byte float, array size: nlat x nlon):
Monthly surface precipitation rate over oceans from the TRAIN algorithm in 0.5^o x 0.5^oboxes. Values range from 0 to 3000 mm/hr. Special values are defined as:
-9999.9 Missing value

stratiformFraction (4-byte float, array size: nlat x nlon):
Monthly stratiform to total ratio of rainfall rate in precipitation over oceans from the TRAIN algorithm in 0.5^o x 0.5^oboxes. Values range from 0 to 100 percent. Special values are defined as:
-9999.9 Missing value

C Structure Header file:

```
#ifndef _TK_3H12_H_
#define _TK_3H12_H_
```

```
#ifndef _L3H12_GRID_
#define _L3H12_GRID_

typedef struct {
    float latentHeating[19][720][160];
    float Q1minusQR[19][720][160];
    int numberOfSamples[720][160];
    float surfacePrecipRate[720][160];
    float stratiformFraction[720][160];
} L3H12_GRID;

#endif

#endif
```

Fortran Structure Header file:

```
STRUCTURE /L3H12_GRID/
    REAL*4 latentHeating(160,720,19)
    REAL*4 Q1minusQR(160,720,19)
    INTEGER*4 numberOfSamples(160,720)
    REAL*4 surfacePrecipRate(160,720)
    REAL*4 stratiformFraction(160,720)
END STRUCTURE
```

7.20 2H25 - Spectral Latent Heating

2H25, "Spectral Latent Heating," produces latent heating, Q1-QR, and Q2 profiles from PR rain. The PI is Dr. Takayabu and the Co-PI is Dr. Shige. The granule size is one orbit. The following sections describe the structure and contents of the format.

Dimension definitions:

nscan	var	Number of scans in the granule.
nray	49	Number of angle bins in each scan.
nlayer	19	Number of layers at the fixed heights of 0.0-0.5 km, 0.5-1 km, 1-2 km, ..., 17-18 km above the sea level.

Figure 83 through Figure 85 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for TRMM Products for details.

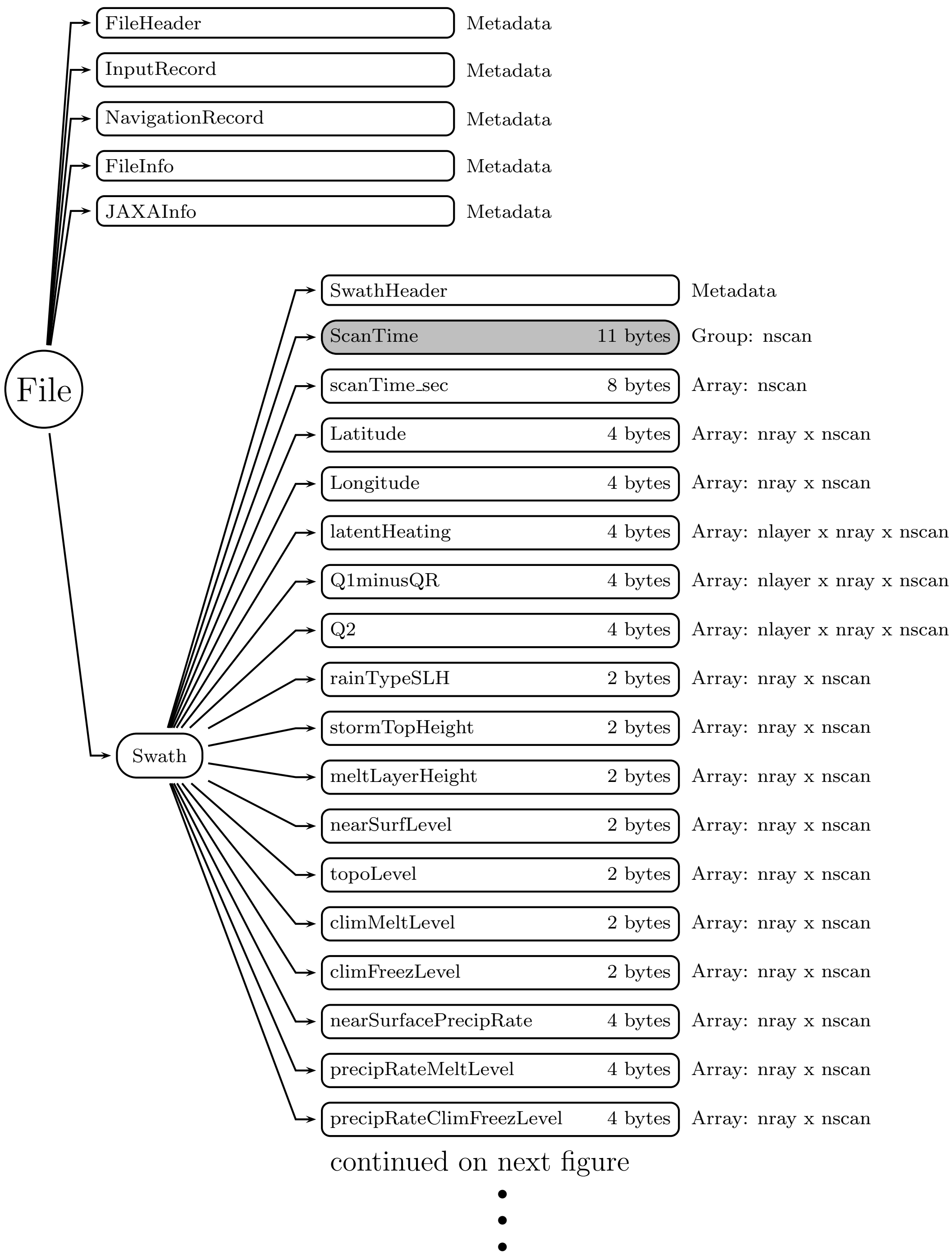


Figure 83: Data Format Structure for 2H25, Spectral Latent Heating

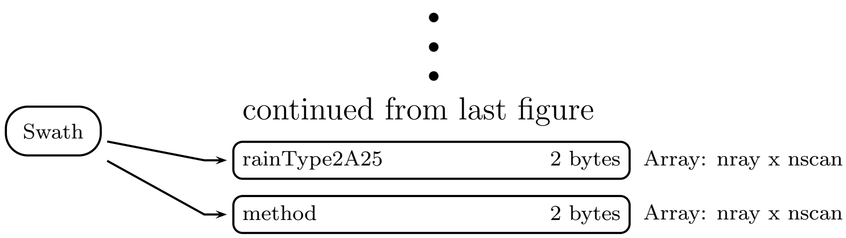


Figure 84: Data Format Structure for 2H25, Spectral Latent Heating

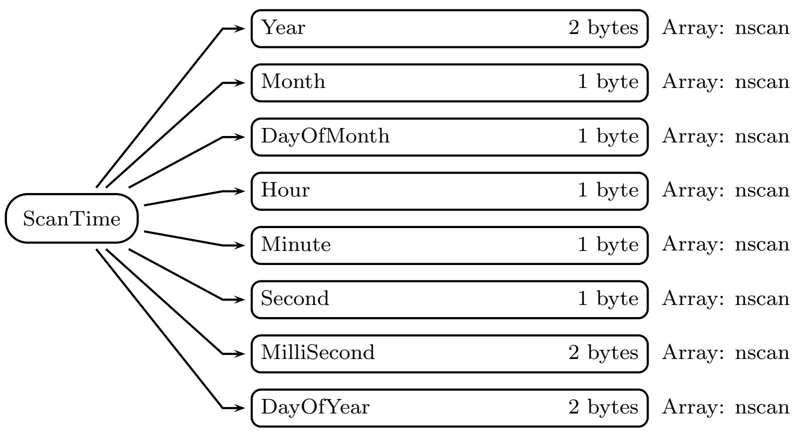


Figure 85: Data Format Structure for 2H25, ScanTime

InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for TRMM Products for details.

NavigationRecord (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for TRMM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for TRMM Products for details.

JAXAInfo (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by PR algorithms only. See Metadata for TRMM Products for details.

Swath (Swath)

SwathHeader (Metadata):

SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for TRMM Products for details.

ScanTime (Group)

A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

scanTime_sec (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: nray x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: nray x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

latentHeating (4-byte float, array size: nlayer x nray x nscan):
Latent Heating. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

Q1minusQR (4-byte float, array size: nlayer x nray x nscan):
Q1 - QR. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

Q2 (4-byte float, array size: nlayer x nray x nscan):
Apparent moisture sink. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

rainTypeSLH (2-byte integer, array size: nray x nscan):
Rain type decided by SLH. Values are as follows:

```
      1: Convective
      2: Shallow strat
      3: Deep strat
      4: Deep strat with low melting level
      5: Intermediary
      6: Other
     80: Mask for Extratropical frontal system regime
     90: Mask for Tibet, winter mid-lat etc.
  -9999: Missing value

  Note that some of pixels classified as rainTypeSLH=4 still remain in the
  version-7A SLH, although the new masks were introduced.
```

stormTopHeight (2-byte integer, array size: nray x nscan):
Height of storm top. Values range from 0 to 32000 m. Special values are defined as:
-9999 Missing value

meltLayerHeight (2-byte integer, array size: nray x nscan):
Height of melting layer. Values range from 0 to 32000 m. Special values are defined as:
-9999 Missing value

nearSurfLevel (2-byte integer, array size: nray x nscan):
Level of near surface rain. Values range from 0 to 32000 m. Special values are defined as:
-9999 Missing value

topoLevel (2-byte integer, array size: nray x nscan):
Level of topography. Values range from 0 to 32000 m. Special values are defined as:
-9999 Missing value

climMeltLevel (2-byte integer, array size: nray x nscan):
Climatological melting level. Values range from 0 to 32000 m. Special values are defined as:
-9999 Missing value

climFreezLevel (2-byte integer, array size: nray x nscan):
Climatological freezing level. Values range from 0 to 32000 m. Special values are defined as:
-9999 Missing value

nearSurfacePrecipRate (4-byte float, array size: nray x nscan):
Precipitation rate at the near surface. Values range from 0 to 500 mm/hr. Special values are defined as:
-9999.9 Missing value

precipRateMeltLevel (4-byte float, array size: nray x nscan):
Precipitation rate at the melting level. Values range from 0 to 500 mm/hr. Special values are defined as:
-9999.9 Missing value

precipRateClimFreezLevel (4-byte float, array size: nray x nscan):
Precipitation rate at the freezing level. Values range from 0 to 500 mm/hr. Special values are defined as:
-9999.9 Missing value

rainType2A25 (2-byte integer, array size: nray x nscan):
Rain Type from 2A25. Special values are defined as:
-9999 Missing value

method (2-byte integer, array size: nray x nscan):
Method from 2A25. Special values are defined as:
-9999 Missing value

C Structure Header file:

```
#ifndef _TK_2H25_H_
#define _TK_2H25_H_

#ifndef _L2H25_SCANTIME_
#define _L2H25_SCANTIME_

typedef struct {
    short Year;
    signed char Month;
```

```
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
} L2H25_SCANTIME;

#endif

#ifndef _L2H25_SWATH_
#define _L2H25_SWATH_

typedef struct {
    L2H25_SCANTIME ScanTime;
    double scanTime_sec;
    float Latitude[49];
    float Longitude[49];
    float latentHeating[49][19];
    float Q1minusQR[49][19];
    float Q2[49][19];
    short rainTypeSLH[49];
    short stormTopHeight[49];
    short meltLayerHeight[49];
    short nearSurfLevel[49];
    short topoLevel[49];
    short climMeltLevel[49];
    short climFreezLevel[49];
    float nearSurfacePrecipRate[49];
    float precipRateMeltLevel[49];
    float precipRateClimFreezLevel[49];
    short rainType2A25[49];
    short method[49];
} L2H25_SWATH;

#endif

#endif
```

Fortran Structure Header file:

```
STRUCTURE /L2H25_SCANTIME/
    INTEGER*2 Year
```

```
    BYTE Month
    BYTE DayOfMonth
    BYTE Hour
    BYTE Minute
    BYTE Second
    INTEGER*2 MilliSecond
    INTEGER*2 DayOfYear
END STRUCTURE

STRUCTURE /L2H25_SWATH/
    RECORD /L2H25_SCANTIME/ ScanTime
    REAL*8 scanTime_sec
    REAL*4 Latitude(49)
    REAL*4 Longitude(49)
    REAL*4 latentHeating(19,49)
    REAL*4 Q1minusQR(19,49)
    REAL*4 Q2(19,49)
    INTEGER*2 rainTypeSLH(49)
    INTEGER*2 stormTopHeight(49)
    INTEGER*2 meltLayerHeight(49)
    INTEGER*2 nearSurfLevel(49)
    INTEGER*2 topoLevel(49)
    INTEGER*2 climMeltLevel(49)
    INTEGER*2 climFreezLevel(49)
    REAL*4 nearSurfacePrecipRate(49)
    REAL*4 precipRateMeltLevel(49)
    REAL*4 precipRateClimFreezLevel(49)
    INTEGER*2 rainType2A25(49)
    INTEGER*2 method(49)
END STRUCTURE
```

7.21 3G25 - Gridded Orbital Spectral Latent Heating

3G25, "Gridded Orbital Spectral Latent Heating", produces 0.5^o x 0.5^o latent heating, Q1-QR, and Q2 profiles from PR rain. The PI is Dr. Takayabu and the Co-PI is Dr. Shige. The granule size is one orbit. The following sections describe the structure and contents of the format.

Dimension definitions:

nlat	148	Number of 0.5^o grid intervals of latitude from 37^oN to 37^oS.
nlon	720	Number of 0.5^o grid intervals of longitude from 180^oW to 180^oE.
nlayer	19	Number of layers at the fixed heights of 0.0-0.5 km, 0.5-1 km, 1-2 km, ..., 17-18 km above the sea level.

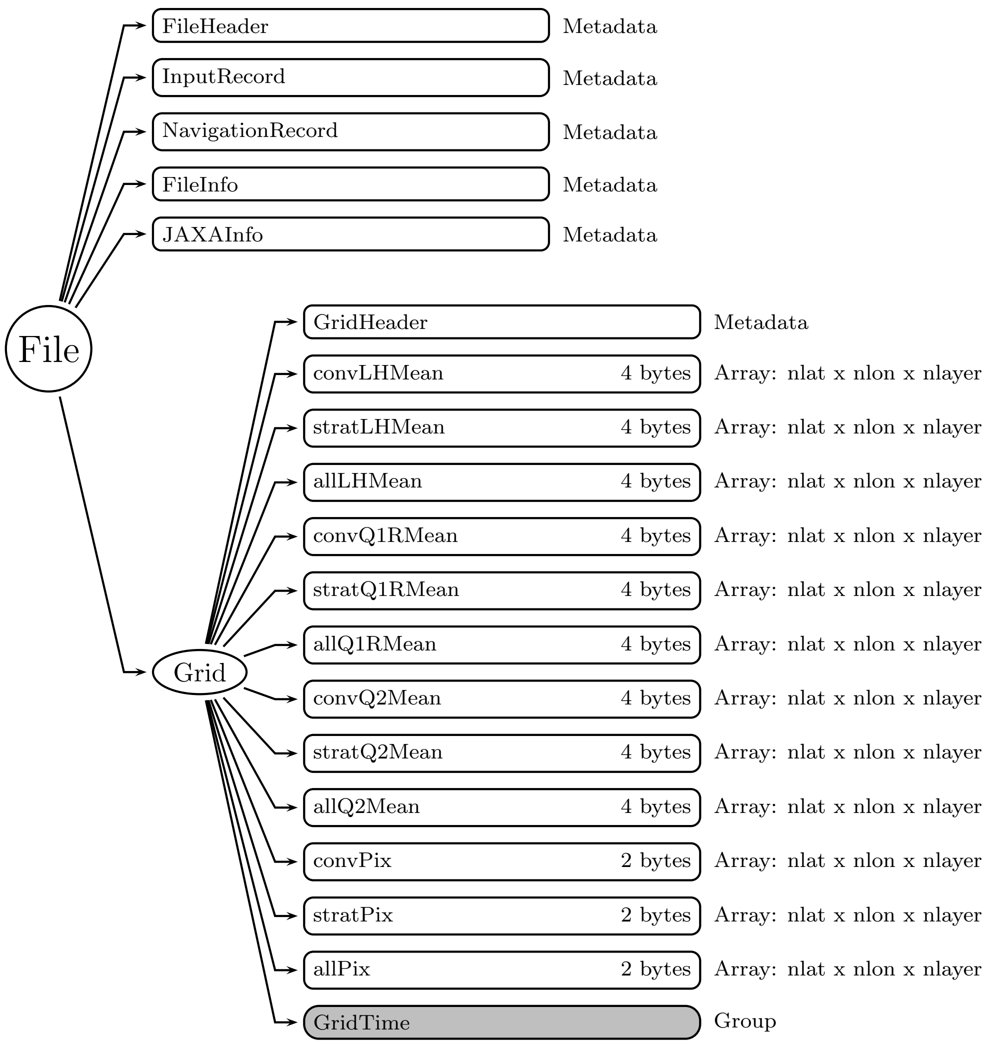


Figure 86: Data Format Structure for 3G25, Gridded Orbital Spectral Latent Heating

Figure 86 through Figure 87 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for TRMM Products for details.

InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for TRMM Products for details.

NavigationRecord (Metadata):

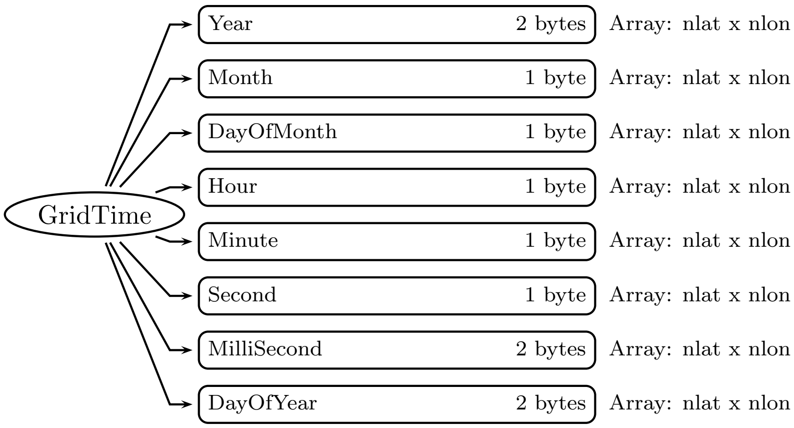


Figure 87: Data Format Structure for 3G25, GridTime

NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for TRMM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for TRMM Products for details.

JAXAInfo (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by PR algorithms only. See Metadata for TRMM Products for details.

Grid (Grid)

GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for TRMM Products for details.

convLHMean (4-byte float, array size: nlat x nlon x nlayer):
Latent heating convective conditional mean. Values range from -400 to 400 K/h. Special values are defined as:
-9999.9 Missing value

stratLHMean (4-byte float, array size: nlat x nlon x nlayer):
Latent heating deep-stratiform and shallow-stratiform conditional mean. Values range from -400 to 400 K/h. Special values are defined as:
-9999.9 Missing value

allLHMean (4-byte float, array size: nlat x nlon x nlayer):
Latent heating all pixel mean. Values range from -400 to 400 K/h. Special values are defined as:
-9999.9 Missing value

convQ1RMean (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR convective conditional mean. Values range from -400 to 400 K/h. Special values

are defined as:
-9999.9 Missing value

stratQ1RMean (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR deep-stratiform and shallow-stratiform conditional mean. Values range from -400 to 400 K/h. Special values are defined as:
-9999.9 Missing value

allQ1RMean (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR all pixel mean. Values range from -400 to 400 K/h. Special values are defined as:
-9999.9 Missing value

convQ2Mean (4-byte float, array size: nlat x nlon x nlayer):
Q2 convective conditional mean. Values range from -400 to 400 K/h. Special values are defined as:
-9999.9 Missing value

stratQ2Mean (4-byte float, array size: nlat x nlon x nlayer):
Q2 deep-stratiform and shallow-stratiform conditional mean. Values range from -400 to 400 K/h. Special values are defined as:
-9999.9 Missing value

allQ2Mean (4-byte float, array size: nlat x nlon x nlayer):
Q2 all pixel mean. Values range from -400 to 400 K/h. Special values are defined as:
-9999.9 Missing value

convPix (2-byte integer, array size: nlat x nlon x nlayer):
Convective pixel counts in the 0.5^o x 0.5^o box. Values range from 0 to 500000. Special values are defined as:
-9999 Missing value

stratPix (2-byte integer, array size: nlat x nlon x nlayer):
Deep-stratiform and shallow-stratiform pixel counts in the 0.5^o x 0.5^o box. Values range from 0 to 500000. Special values are defined as:
-9999 Missing value

allPix (2-byte integer, array size: nlat x nlon x nlayer):
All pixel counts in the 0.5^o x 0.5^o box. Values range from 0 to 500000. Special values are defined as:
-9999 Missing value

GridTime (Group)

Year (2-byte integer, array size: nlat x nlon):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nlat x nlon):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nlat x nlon):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nlat x nlon):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nlat x nlon):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nlat x nlon):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nlat x nlon):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nlat x nlon):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

C Structure Header file:

```
#ifndef _TK_3G25_H_
#define _TK_3G25_H_

#ifndef _L3G25_GRIDTIME_
#define _L3G25_GRIDTIME_

typedef struct {
    short Year[720][148];
    signed char Month[720][148];
    signed char DayOfMonth[720][148];
    signed char Hour[720][148];
    signed char Minute[720][148];
    signed char Second[720][148];
    short MilliSecond[720][148];
    short DayOfYear[720][148];
} L3G25_GRIDTIME;

#endif
```

```
#ifndef _L3G25_GRID_
#define _L3G25_GRID_

typedef struct {
    float convLHMean[19][720][148];
    float stratLHMean[19][720][148];
    float allLHMean[19][720][148];
    float convQ1RMean[19][720][148];
    float stratQ1RMean[19][720][148];
    float allQ1RMean[19][720][148];
    float convQ2Mean[19][720][148];
    float stratQ2Mean[19][720][148];
    float allQ2Mean[19][720][148];
    short convPix[19][720][148];
    short stratPix[19][720][148];
    short allPix[19][720][148];
    L3G25_GRIDTIME GridTime;
} L3G25_GRID;

#endif

#endif
```

Fortran Structure Header file:

```
STRUCTURE /L3G25_GRIDTIME/
    INTEGER*2 Year(148,720)
    BYTE Month(148,720)
    BYTE DayOfMonth(148,720)
    BYTE Hour(148,720)
    BYTE Minute(148,720)
    BYTE Second(148,720)
    INTEGER*2 MilliSecond(148,720)
    INTEGER*2 DayOfYear(148,720)
END STRUCTURE

STRUCTURE /L3G25_GRID/
    REAL*4 convLHMean(148,720,19)
    REAL*4 stratLHMean(148,720,19)
    REAL*4 allLHMean(148,720,19)
    REAL*4 convQ1RMean(148,720,19)
    REAL*4 stratQ1RMean(148,720,19)
```

```
    REAL*4 allQ1RMean(148,720,19)
    REAL*4 convQ2Mean(148,720,19)
    REAL*4 stratQ2Mean(148,720,19)
    REAL*4 allQ2Mean(148,720,19)
    INTEGER*2 convPix(148,720,19)
    INTEGER*2 stratPix(148,720,19)
    INTEGER*2 allPix(148,720,19)
    RECORD /L3G25_GRIDTIME/ GridTime
END STRUCTURE
```

7.22 3H25 - Monthly Spectral Latent Heating

3H25, "Monthly Spectral Latent Heating", produces 0.5^o x 0.5^o latent heating, Q1-QR, and Q2 profiles from PR rain. The PI is Dr. Takayabu and the Co-PI is Dr. Shige. The granule size is one month. The following sections describe the structure and contents of the format.

Dimension definitions:

nlat	148	Number of 0.5^o grid intervals of latitude from 37^oN to 37^oS.
nlon	720	Number of 0.5^o grid intervals of longitude from 180^oW to 180^oE.
nlayer	19	Number of layers at the fixed heights of 0.0-0.5 km, 0.5-1 km, 1-2 km, ..., 17-18 km above the sea level.

Figure 88 through Figure 89 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for TRMM Products for details.

InputFileNames (Metadata):
InputFileNames contains a list of input file names for this granule. See Metadata for TRMM Products for details.

InputAlgorithmVersions (Metadata):
InputAlgorithmVersions contains a list of input algorithm versions for this granule. See Metadata for TRMM Products for details.

InputGenerationDateTimes (Metadata):
InputGenerationDateTimes contains a list of input generation datetimes. See Metadata for TRMM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for TRMM Products for details.

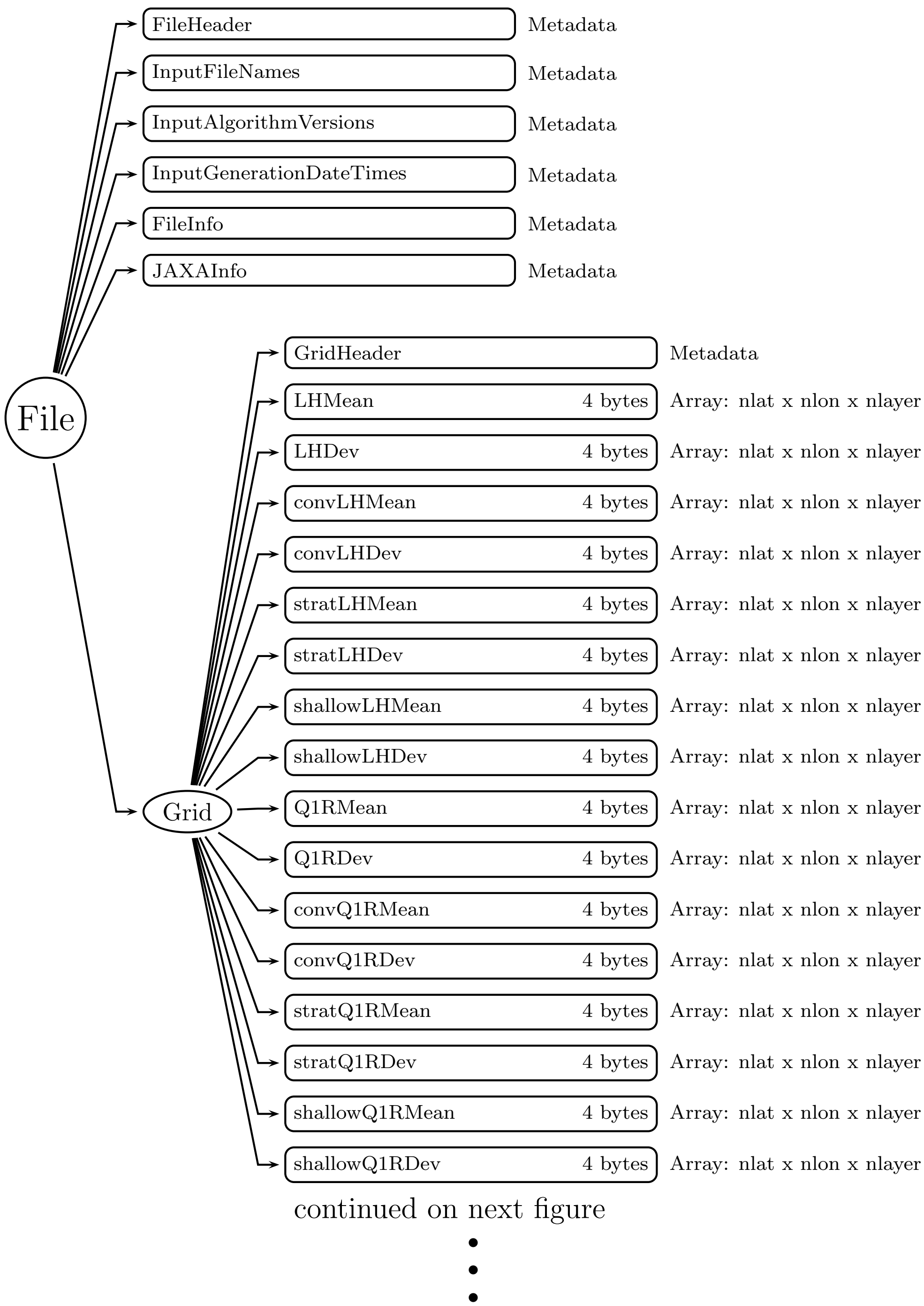


Figure 88: Data Format Structure for 3H25, Monthly Spectral Latent Heating

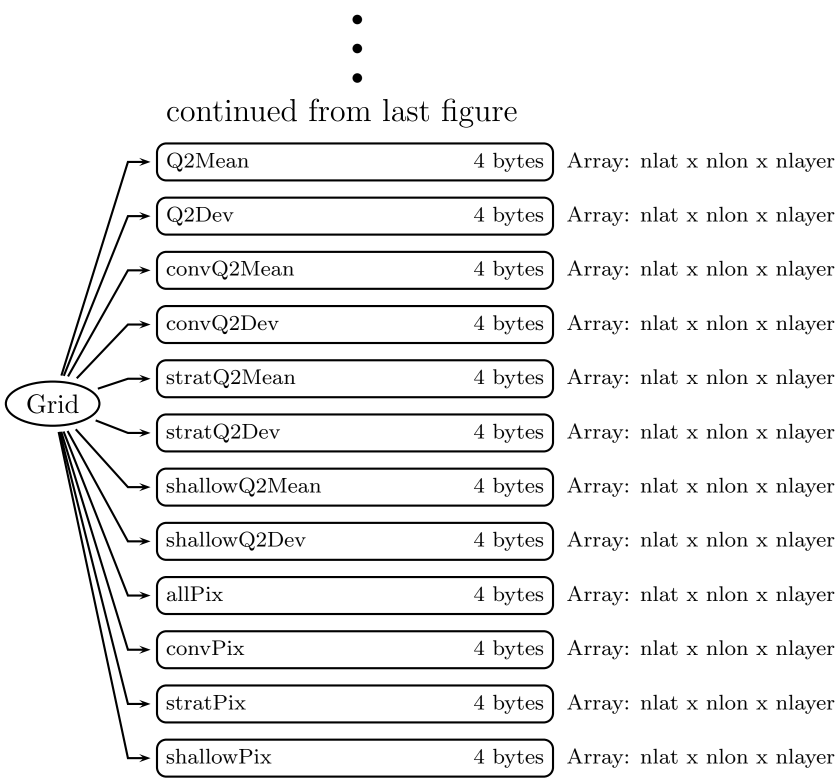


Figure 89: Data Format Structure for 3H25, Monthly Spectral Latent Heating

JAXAInfo (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by PR algorithms only. See Metadata for TRMM Products for details.

Grid (Grid)

GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for TRMM Products for details.

LHMean (4-byte float, array size: nlat x nlon x nlayer):
Latent heating conditional mean. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

LHDev (4-byte float, array size: nlat x nlon x nlayer):
Latent heating conditional standard deviation. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

convLHMean (4-byte float, array size: nlat x nlon x nlayer):
Latent heating convective conditional mean. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

convLHDev (4-byte float, array size: nlat x nlon x nlayer):
Latent heating convective conditional standard deviation. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

stratLHMean (4-byte float, array size: nlat x nlon x nlayer):
Latent heating deep-stratiform conditional mean. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

stratLHDev (4-byte float, array size: nlat x nlon x nlayer):
Latent heating deep-stratiform conditional standard deviation. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

shallowLHMean (4-byte float, array size: nlat x nlon x nlayer):
Latent heating shallow-stratiform conditional mean. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

shallowLHDev (4-byte float, array size: nlat x nlon x nlayer):
Latent heating shallow-stratiform conditional standard deviation. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

Q1RMean (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR conditional mean. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

Q1RDev (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR conditional standard deviation. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

convQ1RMean (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR convective conditional mean. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

convQ1RDev (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR convective conditional standard deviation. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

stratQ1RMean (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR deep-stratiform conditional mean. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

stratQ1RDev (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR deep-stratiform conditional standard deviation. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

shallowQ1RMean (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR shallow-stratiform conditional mean. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

shallowQ1RDev (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR shallow-stratiform conditional standard deviation. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

Q2Mean (4-byte float, array size: nlat x nlon x nlayer):
Q2 conditional mean. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

Q2Dev (4-byte float, array size: nlat x nlon x nlayer):
Q2 conditional standard deviation. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

convQ2Mean (4-byte float, array size: nlat x nlon x nlayer):
Q2 convective conditional mean. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

convQ2Dev (4-byte float, array size: nlat x nlon x nlayer):
Q2 convective conditional standard deviation. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

stratQ2Mean (4-byte float, array size: nlat x nlon x nlayer):
Q2 deep-stratiform conditional mean. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

stratQ2Dev (4-byte float, array size: nlat x nlon x nlayer):
Q2 deep-stratiform conditional standard deviation. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

shallowQ2Mean (4-byte float, array size: nlat x nlon x nlayer):
Q2 shallow-stratiform conditional mean. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

shallowQ2Dev (4-byte float, array size: nlat x nlon x nlayer):
Q2 shallow-stratiform conditional standard deviation. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

allPix (4-byte float, array size: nlat x nlon x nlayer):
All pixel counts. Values range from 0 to 2000000000. Special values are defined as:
-9999.9 Missing value

convPix (4-byte float, array size: nlat x nlon x nlayer):
Convective pixel counts. Values range from 0 to 2000000000. Special values are defined as:
-9999.9 Missing value

stratPix (4-byte float, array size: nlat x nlon x nlayer):
Deep-stratiform pixel counts. Values range from 0 to 2000000000. Special values are defined as:
-9999.9 Missing value

shallowPix (4-byte float, array size: nlat x nlon x nlayer):
Shallow-stratiform pixel counts. Values range from 0 to 2000000000. Special values are defined as:
-9999.9 Missing value

C Structure Header file:

```
#ifndef _TK_3H25_H_
#define _TK_3H25_H_

#ifndef _L3H25_GRID_
#define _L3H25_GRID_

typedef struct {
    float LHMean[19][720][148];
    float LHDev[19][720][148];
    float convLHMean[19][720][148];
    float convLHDev[19][720][148];
    float stratLHMean[19][720][148];
    float stratLHDev[19][720][148];
    float shallowLHMean[19][720][148];
    float shallowLHDev[19][720][148];
    float Q1RMean[19][720][148];
    float Q1RDev[19][720][148];
    float convQ1RMean[19][720][148];
    float convQ1RDev[19][720][148];
    float stratQ1RMean[19][720][148];
    float stratQ1RDev[19][720][148];
    float shallowQ1RMean[19][720][148];
    float shallowQ1RDev[19][720][148];
    float Q2Mean[19][720][148];
    float Q2Dev[19][720][148];
    float convQ2Mean[19][720][148];
    float convQ2Dev[19][720][148];
    float stratQ2Mean[19][720][148];
    float stratQ2Dev[19][720][148];
    float shallowQ2Mean[19][720][148];
    float shallowQ2Dev[19][720][148];
    float allPix[19][720][148];
    float convPix[19][720][148];
    float stratPix[19][720][148];
    float shallowPix[19][720][148];
} L3H25_GRID;

#endif

#endif
```

Fortran Structure Header file:

```
STRUCTURE /L3H25_GRID/
    REAL*4 LHMean(148,720,19)
    REAL*4 LHDev(148,720,19)
    REAL*4 convLHMean(148,720,19)
    REAL*4 convLHDev(148,720,19)
    REAL*4 stratLHMean(148,720,19)
    REAL*4 stratLHDev(148,720,19)
    REAL*4 shallowLHMean(148,720,19)
    REAL*4 shallowLHDev(148,720,19)
    REAL*4 Q1RMean(148,720,19)
    REAL*4 Q1RDev(148,720,19)
    REAL*4 convQ1RMean(148,720,19)
    REAL*4 convQ1RDev(148,720,19)
    REAL*4 stratQ1RMean(148,720,19)
    REAL*4 stratQ1RDev(148,720,19)
    REAL*4 shallowQ1RMean(148,720,19)
    REAL*4 shallowQ1RDev(148,720,19)
    REAL*4 Q2Mean(148,720,19)
    REAL*4 Q2Dev(148,720,19)
    REAL*4 convQ2Mean(148,720,19)
    REAL*4 convQ2Dev(148,720,19)
    REAL*4 stratQ2Mean(148,720,19)
    REAL*4 stratQ2Dev(148,720,19)
    REAL*4 shallowQ2Mean(148,720,19)
    REAL*4 shallowQ2Dev(148,720,19)
    REAL*4 allPix(148,720,19)
    REAL*4 convPix(148,720,19)
    REAL*4 stratPix(148,720,19)
    REAL*4 shallowPix(148,720,19)
END STRUCTURE
```

7.23 2H31 - Convective Stratiform Heating

2H31, "Convective Stratiform Heating," produces orbital apparent heating profiles from surface convective rainfall rate and surface stratiform rainfall rate. The PI is Dr. Tao. The granule size is one orbit. The following sections describe the structure and contents of the format.

Dimension definitions:

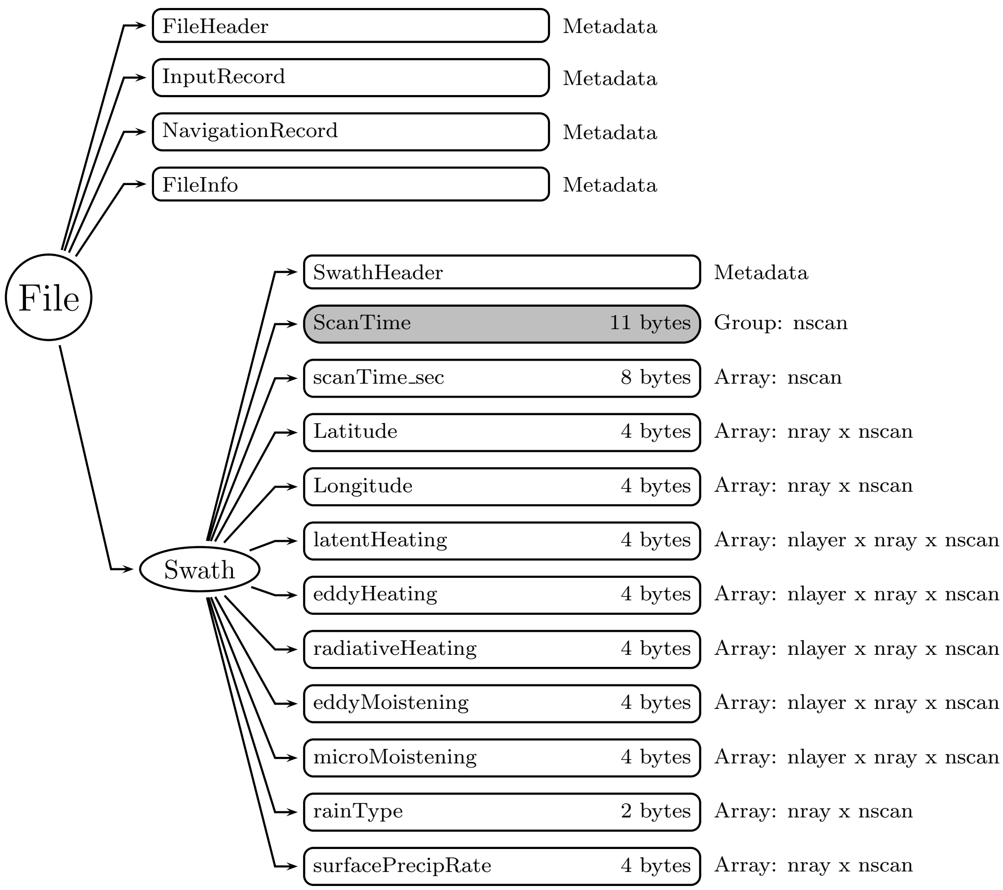


Figure 90: Data Format Structure for 2H31, Convective Stratiform Heating

nscan	var	Number of scans in the granule.
nray	49	Number of angle bins in each scan.
nlayer	19	Number of layers at 0.0-0.5 km, 0.5-1 km, 1-2 km, ..., 17-18 km above the ground level.

Figure 90 through Figure 91 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for TRMM Products for details.

InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for TRMM Products for details.

NavigationRecord (Metadata):

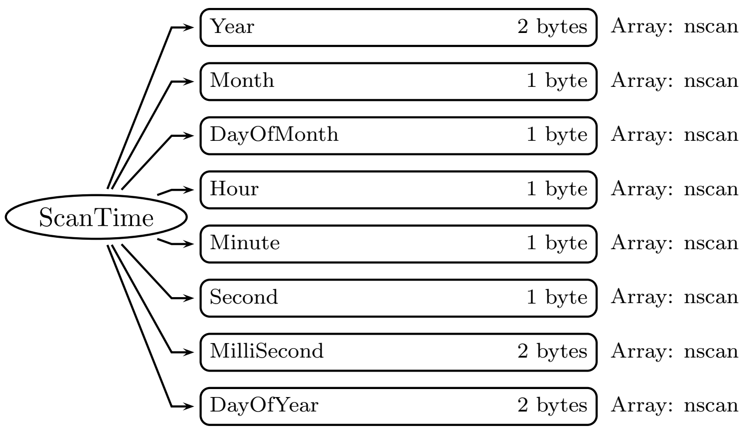


Figure 91: Data Format Structure for 2H31, ScanTime

NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for TRMM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for TRMM Products for details.

Swath (Swath)

SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for TRMM Products for details.

ScanTime (Group)

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

scanTime_sec (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: nray x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: nray x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

latentHeating (4-byte float, array size: nlayer x nray x nscan):
Latent Heating. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

eddyHeating (4-byte float, array size: nlayer x nray x nscan):
Eddy flux heating. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

radiativeHeating (4-byte float, array size: nlayer x nray x nscan):
Radiative heating. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

eddyMoistening (4-byte float, array size: nlayer x nray x nscan):
Apparent moistening due to eddy processes. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

microMoistening (4-byte float, array size: nlayer x nray x nscan):
Apparent moistening due to microphysical processes. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

rainType (2-byte integer, array size: nray x nscan):
Rain type from Level 2 PR Rain Type. Special values are defined as:
-9999 Missing value

surfacePrecipRate (4-byte float, array size: nray x nscan):
Mean estimated surface precipitation rate from Level 2 Combined. Values range from 0 to 500 mm/hr. Special values are defined as:
-9999.9 Missing value

C Structure Header file:

```
#ifndef _TK_2H31_H_
#define _TK_2H31_H_

#ifndef _L2H31_SCANTIME_
#define _L2H31_SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
} L2H31_SCANTIME;

#endif

#ifndef _L2H31_SWATH_
#define _L2H31_SWATH_

typedef struct {
    L2H31_SCANTIME ScanTime;
    double scanTime_sec;
    float Latitude[49];
    float Longitude[49];
    float latentHeating[49][19];
    float eddyHeating[49][19];
```

```
    float radiativeHeating[49][19];
    float eddyMoistening[49][19];
    float microMoistening[49][19];
    short rainType[49];
    float surfacePrecipRate[49];
} L2H31_SWATH;

#endif

#endif
```

Fortran Structure Header file:

```
STRUCTURE /L2H31_SCANTIME/
    INTEGER*2 Year
    BYTE Month
    BYTE DayOfMonth
    BYTE Hour
    BYTE Minute
    BYTE Second
    INTEGER*2 MilliSecond
    INTEGER*2 DayOfYear
END STRUCTURE

STRUCTURE /L2H31_SWATH/
    RECORD /L2H31_SCANTIME/ ScanTime
    REAL*8 scanTime_sec
    REAL*4 Latitude(49)
    REAL*4 Longitude(49)
    REAL*4 latentHeating(19,49)
    REAL*4 eddyHeating(19,49)
    REAL*4 radiativeHeating(19,49)
    REAL*4 eddyMoistening(19,49)
    REAL*4 microMoistening(19,49)
    INTEGER*2 rainType(49)
    REAL*4 surfacePrecipRate(49)
END STRUCTURE
```

7.24 3G31 - Gridded Orbital Convective Stratiform Heating from Combined

3G31, "Gridded Orbital Convective Stratiform Heating from Combined", produces 0.5^o x 0.5^o orbital apparent heating profiles from surface convective rainfall rate and surface

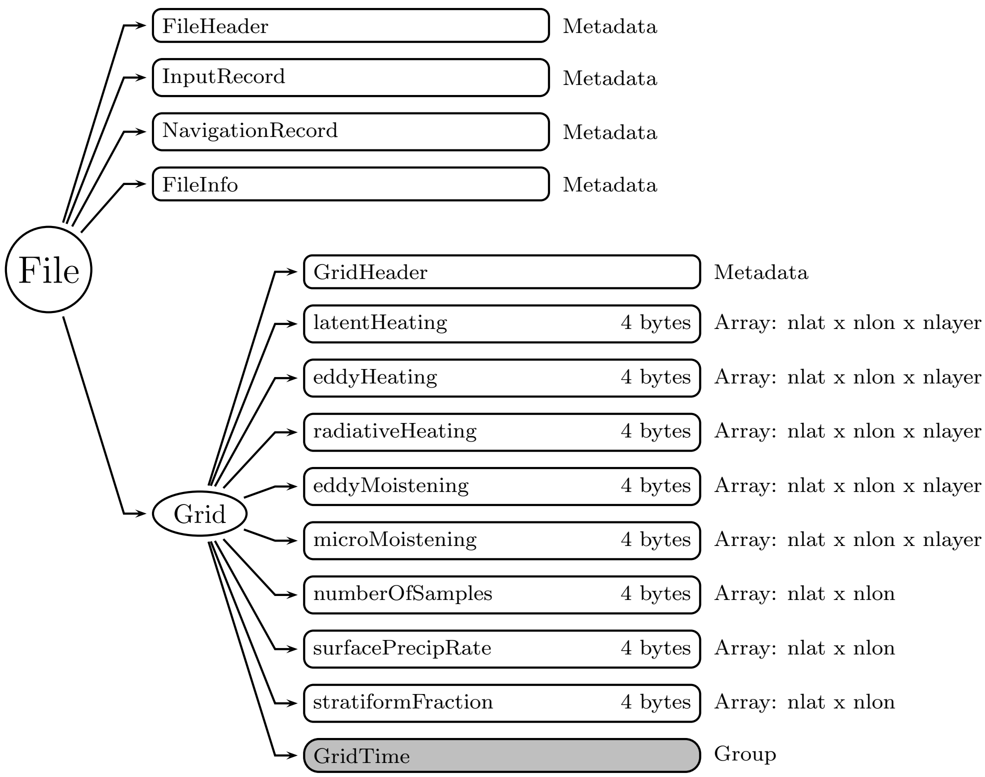


Figure 92: Data Format Structure for 3G31, Gridded Orbital Convective Stratiform Heating from Combined

stratiform rainfall rate. The PI is Dr. Wei-Kuo Tao. The granule size is one orbit. The following sections describe the structure and contents of the format.

Dimension definitions:

nlat	148	Number of 0.5^o grid intervals of latitude from 37^oN to 37^oS.
nlon	720	Number of 0.5^o grid intervals of longitude from 180^oW to 180^oE.
nlayer	19	Number of layers at 0.0-0.5 km, 0.5-1 km, 1-2 km, ..., 17-18 km above the ground level.

Figure 92 through Figure 93 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for TRMM Products for details.

InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for TRMM Products for details.

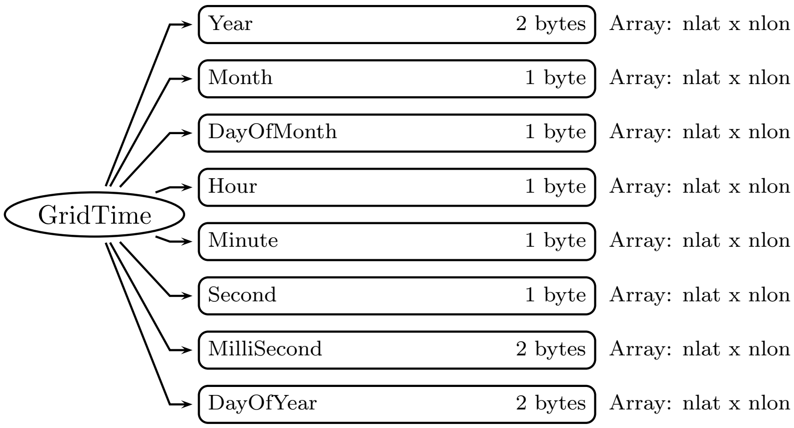


Figure 93: Data Format Structure for 3G31, GridTime

NavigationRecord (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for TRMM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for TRMM Products for details.

Grid (Grid)

GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for TRMM Products for details.

latentHeating (4-byte float, array size: nlat x nlon x nlayer):
Latentheating. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

eddyHeating (4-byte float, array size: nlat x nlon x nlayer):
Eddy flux heating. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

radiativeHeating (4-byte float, array size: nlat x nlon x nlayer):
Radiative heating. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

eddyMoistening (4-byte float, array size: nlat x nlon x nlayer):
Apparent moistening due to eddy processes. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

microMoistening (4-byte float, array size: nlat x nlon x nlayer):
Apparent moistening due to microphysical processes. Values range from -50 to 100 K/hr.

Special values are defined as:
-9999.9 Missing value

numberOfSamples (4-byte integer, array size: nlat x nlon):
Number of samples in 0.5^o x 0.5^o boxes. Values range from 0 to 500000. Special values are defined as:
-9999 Missing value

surfacePrecipRate (4-byte float, array size: nlat x nlon):
Mean estimated surface precipitation rate from Level 2 Combined. Values range from 0 to 3000 mm/hr. Special values are defined as:
-9999.9 Missing value

stratiformFraction (4-byte float, array size: nlat x nlon):
Ratio of stratiform to total surface rain rate from Level 2 PR. Values range from 0 to 1. Special values are defined as:
-9999.9 Missing value

GridTime (Group)

Year (2-byte integer, array size: nlat x nlon):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nlat x nlon):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nlat x nlon):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nlat x nlon):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nlat x nlon):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nlat x nlon):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nlat x nlon):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nlat x nlon):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

C Structure Header file:

```
#ifndef _TK_3G31_H_
#define _TK_3G31_H_

#ifndef _L3G31_GRIDTIME_
#define _L3G31_GRIDTIME_

typedef struct {
    short Year[720][148];
    signed char Month[720][148];
    signed char DayOfMonth[720][148];
    signed char Hour[720][148];
    signed char Minute[720][148];
    signed char Second[720][148];
    short MilliSecond[720][148];
    short DayOfYear[720][148];
} L3G31_GRIDTIME;

#endif

#ifndef _L3G31_GRID_
#define _L3G31_GRID_

typedef struct {
    float latentHeating[19][720][148];
    float eddyHeating[19][720][148];
    float radiativeHeating[19][720][148];
    float eddyMoistening[19][720][148];
    float microMoistening[19][720][148];
    int numberOfSamples[720][148];
    float surfacePrecipRate[720][148];
    float stratiformFraction[720][148];
    L3G31_GRIDTIME GridTime;
} L3G31_GRID;

#endif

#endif
```

Fortran Structure Header file:

```
STRUCTURE /L3G31_GRIDTIME/
    INTEGER*2 Year(148,720)
    BYTE Month(148,720)
    BYTE DayOfMonth(148,720)
    BYTE Hour(148,720)
    BYTE Minute(148,720)
    BYTE Second(148,720)
    INTEGER*2 MilliSecond(148,720)
    INTEGER*2 DayOfYear(148,720)
END STRUCTURE

STRUCTURE /L3G31_GRID/
    REAL*4 latentHeating(148,720,19)
    REAL*4 eddyHeating(148,720,19)
    REAL*4 radiativeHeating(148,720,19)
    REAL*4 eddyMoistening(148,720,19)
    REAL*4 microMoistening(148,720,19)
    INTEGER*4 numberOfSamples(148,720)
    REAL*4 surfacePrecipRate(148,720)
    REAL*4 stratiformFraction(148,720)
    RECORD /L3G31_GRIDTIME/ GridTime
END STRUCTURE
```

7.25 3H31 - Monthly Convective Stratiform Heating from Combined

3H31, "Monthly Convective Stratiform Heating from Combined", produces 0.5^o x 0.5^o monthly apparent heating profiles from surface convective rainfall rate and surface stratiform rainfall rate. The PI is Dr. Wei-Kuo Tao. The granule size is one month. The following sections describe the structure and contents of the format.

Dimension definitions:

nlat	148	Number of 0.5^o grid intervals of latitude from 37^oN to 37^oS.
nlon	720	Number of 0.5^o grid intervals of longitude from 180^oW to 180^oE.
nlayer	19	Number of layers at 0.0-0.5 km, 0.5-1 km, 1-2 km, ..., 17-18 km above the ground level.

Figure 94 shows the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

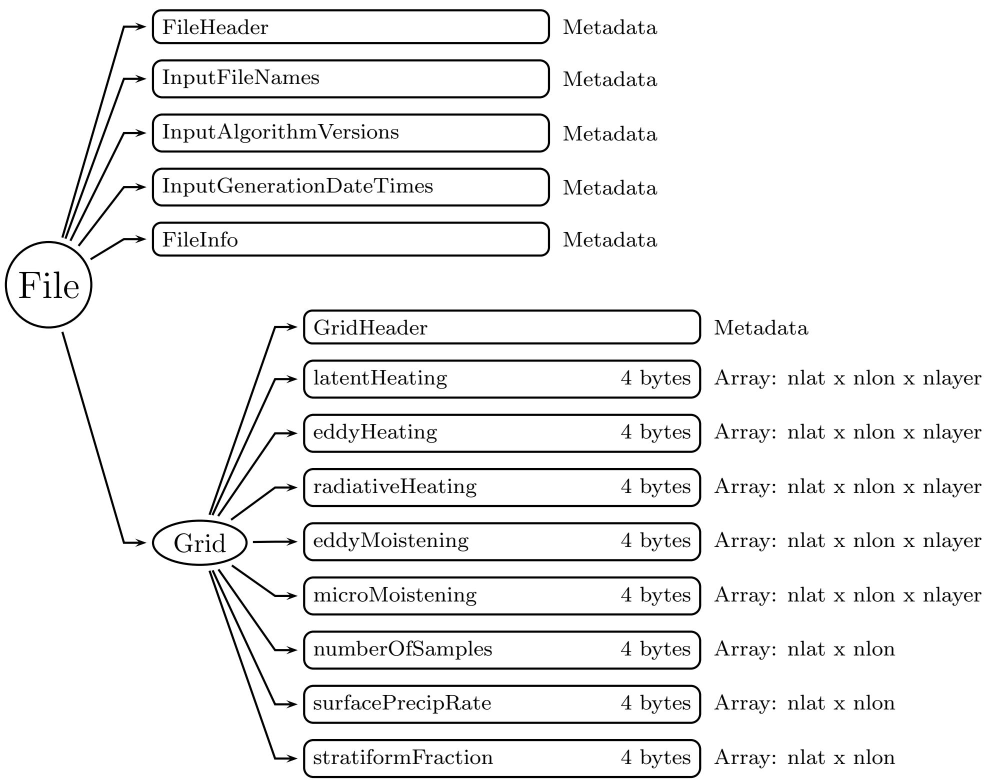


Figure 94: Data Format Structure for 3H31, Monthly Convective Stratiform Heating from Combined

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for TRMM Products for details.

InputFileNames (Metadata):
InputFileNames contains a list of input file names for this granule. See Metadata for TRMM Products for details.

InputAlgorithmVersions (Metadata):
InputAlgorithmVersions contains a list of input algorithm versions for this granule. See Metadata for TRMM Products for details.

InputGenerationDateTimes (Metadata):
InputGenerationDateTimes contains a list of input generation datetimes. See Metadata for TRMM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for TRMM Products for details.

Grid (Grid)

GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for TRMM Products for details.

latentHeating (4-byte float, array size: nlat x nlon x nlayer):
Latentheating. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

eddyHeating (4-byte float, array size: nlat x nlon x nlayer):
Eddy flux heating. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

radiativeHeating (4-byte float, array size: nlat x nlon x nlayer):
Radiative heating. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

eddyMoistening (4-byte float, array size: nlat x nlon x nlayer):
Apparent moistening due to eddy processes. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

microMoistening (4-byte float, array size: nlat x nlon x nlayer):
Apparent moistening due to microphysical processes. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

numberOfSamples (4-byte integer, array size: nlat x nlon):
Number of samples in 0.5^o x 0.5^o boxes for one month. Values range from 0 to 500000.

Special values are defined as:
-9999 Missing value

surfacePrecipRate (4-byte float, array size: nlat x nlon):
Monthly estimated surface precipitation rate from Level 3 combined. Values range from 0 to 3000 mm/hr. Special values are defined as:
-9999.9 Missing value

stratiformFraction (4-byte float, array size: nlat x nlon):
Ratio of stratiform to total surface rain rate from Level 3 PR. Values range from 0 to 1. Special values are defined as:
-9999.9 Missing value

C Structure Header file:

```
#ifndef _TK_3H31_H_
#define _TK_3H31_H_

#ifndef _L3H31_GRID_
#define _L3H31_GRID_

typedef struct {
    float latentHeating[19][720][148];
    float eddyHeating[19][720][148];
    float radiativeHeating[19][720][148];
    float eddyMoistening[19][720][148];
    float microMoistening[19][720][148];
    int numberOfSamples[720][148];
    float surfacePrecipRate[720][148];
    float stratiformFraction[720][148];
} L3H31_GRID;

#endif

#endif
```

Fortran Structure Header file:

```
STRUCTURE /L3H31_GRID/
    REAL*4 latentHeating(148,720,19)
    REAL*4 eddyHeating(148,720,19)
    REAL*4 radiativeHeating(148,720,19)
    REAL*4 eddyMoistening(148,720,19)
    REAL*4 microMoistening(148,720,19)
```

```
    INTEGER*4 numberOfSamples(148,720)
    REAL*4 surfacePrecipRate(148,720)
    REAL*4 stratiformFraction(148,720)
END STRUCTURE
```