Thermal Emission Imaging System
2001 Mars Odyssey

THM-EDR & THM-RDR SOFTWARE INTERFACE SPECIFICATION

October 1, 2002

Thermal Emission Imaging System
2001 Mars Odyssey

THM-EDR & THM-RDR SOFTWARE INTERFACE SPECIFICATION

Prepared by:

Kimberly C. Murray Date
THEMIS Data Archivist

Approved by:

Philip R. Christensen Date
THEMIS Principal Investigator

Greg L. Mehall Date
THEMIS Instrument Manager

Noel S. Gorelick Date
THEMIS Principal Systems Programmer

R. Stephen Saunders Date
Mars Odyssey Project Scientist

Betty J. Sword Date
PDS Mars Data Engineer

October 1, 2002

DOCUMENT CHANGE LOG

Date	Description	Sections affected
10/01/01	Initial draft	All
02/25/02	Response to PDS Imaging Node review	All
04/01/02	Response to PDS Imaging Node review; QUBE label corresponds with SFDU2CUBE V-1.45	Most
05/31/02	Response to PDS Imaging Node review and external content review; QUBE label corresponds with SFDU2CUBE V-1.50	Most
07/24/02	Revision of TIME keywords in all labels; revision of IREDR CORE_NULL value; revision of RDR label; revision of center wavelngths of the VIS filters; QUBE label corresponds with SFDU2CUBE V-1.54	Section 2.3.3; Appendicies
10/01/02	Revision of IR-RDR QUBEs calibration (V-4.5) and data format; QUBE label correponds with SFDU2CUBE V-1.54 Response to PDS Peer Review	Sections 2.2, 3.2; Appendicies

CONTENTS

ACRONYMS

ASU	Arizona State University
DN	Data Number
EDR	Experiment Data Record
IR	Infrared
IRIS	Infrared Imaging System
IRS	Infrared Subsystem
ISIS	Integrated Software for Imaging Spectrometers
JPL	Jet Propulsion Laboratory
NASA	National Aeronautics and Space Administration
PDS	Planetary Data System
RDR	Reduced Data Record
ODY	2001 Mars Odyssey
SBRS	Santa Barbara Remote Sensing
SFDU	Standard Formatted Data Unit
SIS	Software Interface Specification
TDI	Time-Delay Integration
TE	Thermal Electric
THEMIS	THermal EMission Imaging System
TLM	Telemetry
VIS	Visible

1. INTRODUCTION

1.1 Purpose and Scope

The purpose of this Data Product SIS is to provide users of the Thermal Emission Imaging System (THEMIS) Visible and Infrared experimental and reduced data products (VISEDR, IREDR, VISRDR, and IRRDR) with enough information to enable them to read and understand the data products. These products are spectral image QUBEs consisting of one layer per each visible or infrared band collected. The format and content specifications presented here apply to all data collection phases of the 2001 Mars Odyssey Project for which the data products are available. This SIS is intended to be used by the scientists who will analyze the data, including those associated with the 2001 Mars Odyssey Project and those in the general planetary science community.

1.2 Contents

This Data Product SIS describes in detail how the visible and infrared data products are acquired by the THEMIS instrument, and how the data are processed, formatted, labeled, and uniquely identified. The document discusses standards used in generating the product and the software that may be used to access the product. The data product structure and organization is described in sufficient detail to enable a user to read the product. Finally, examples of product labels are provided.

1.3 Applicable Documents and Constraints

This Data Product SIS is responsive to the following 2001 Mars Odyssey documents:

1. Mars Exploration Program Data Management Plan, R. E. Arvidson and S. Slavney, Rev. 2, Nov. 2, 2000.
2. 2001 Mars Odyssey Orbiter Archive Generation, Validation and Transfer Plan, R. E. Arvidson, R. S. Saunders, and S. Slavney, JPL D-20679, November 3, 2000.

This SIS is also consistent with the following Planetary Data System documents:

3. Planetary Data System Data Preparation Workbook, February 1, 1995, Version 3.1, JPL D-7669, Part 1.
4. Planetary Data System Data Standards Reference, June 15, 2001, Version 3.4, JPL D-7669, Part 2.
5. Planetary Data System Data Dictionary, July 27, 2001, JPL D-6184.
   (Note: The Data Dictionary is being updated to include several THEMIS specific changes.)

The user is referred to the following THEMIS documents for additional information:

6. The Thermal Emission Imaging System (THEMIS) for the Mars 2001 Odyssey Mission, P.R. Christensen, et. Al., Submitted to *Space Science Review*, 2001.
7. Calibration Report for the Thermal Emission Imaging System (THEMIS) for the 2001 Mars Odyssey Mission, P.R. Christensen.

8. Mars Odyssey THEMIS: Data Processing User’s Guide, P.R. Christensen.
9. Mars Odyssey THEMIS: Geometric Processing User’s Guide.

Finally, this SIS is meant to be consistent with the contract negotiated between the 2001 Mars Odyssey Project and the THEMIS Principal Investigator (PI) in which reduced data records and documentation are explicitly defined as deliverable products.

1.4 Relationships with Other Interfaces

Changes in the THM-EDR and THM-RDR data products would require changes to this document. Changes to the data products will most likely also affect the processing software described in THEMIS Data Processing User’s Guide [8].

2. DATA PRODUCT CHARACTERISTICS AND ENVIRONMENT

2.1 Instrument Overview

The THEMIS instrument is a combined infrared (IR) and visible (VIS) multi-spectral pushbroom imager. The imaging system is comprised of a three-mirror, off-axis, reflecting telescope in a rugged enclosure, a visible/infrared beamsplitter, a silicon focal plane for visible detection, and a microbolometer for infrared detection. The telescope has a 12-cm effective aperture, speed of f/1.6, and co-aligned VIS-IR detector arrays. A major feature of this instrument is the uncooled IR microbolometer array which can be operated at ambient temperature. A small thermal electric (TE) cooler is used to stabilize the detector temperature to ±0.001 K. The calibration flag is the only moving part in the instrument, allowing for thermal calibration and protection of the detectors from unintentional direct Sun illumination when the instrument is not in use.

THEMIS IR images are acquired at selectable image lengths and in combinations of ten selectable bands. The image width is 320 pixels (32 km, based on the nominal 400 km mapping orbit) and the length is variable, in multiples of 256 line increments, with a minimum and maximum image lengths of 272 and 65,296 lines respectively (27.2 km and 6,530 km, based on the nominal mapping orbit). The IR focal plane is covered by ten ~1 µm-bandwidth strip filters, producing ten band images with bands 1 and 2 having the same wavelength range. VIS images can be acquired simultaneously with IR images, but the spacecraft can only transfer data from one of the two THEMIS imagers at a time. The IR imager transfers data as it is being collected, while the VIS images are stored within an internal THEMIS buffer for later transfer to the spacecraft computer.

THEMIS VIS images are acquired in framelets of size 1024 pixels crosstrack by 192 lines downtrack, for a total image size of 3.734 Mbytes or less. The number of framelets is determined by the number of bands selected (five available) and the spatial resolution selected (three summing modes available). The size of an image is given by:

$$[((1024 * 192) * \#framelets * \#bands) \div summing^2] \leq 3.734 \text{ Mbytes}$$

For example, if spatial summing is not applied (summing=1), either a single-band, 19-framelet (65.6 km) image or a 5-band 3-framelet (10.3 km) image can be collected. Each VIS image collected is stored in the THEMIS internal buffer and must be transferred to the spacecraft computer before a subsequent image can be acquired.

The IR and VIS cameras share the instrument optics and housing, but have independent power and data interfaces to the spacecraft. Internal THEMIS data processing may be applied to each image type before storage. IR images may be processed with a time-delay integration (TDI) algorithm which co-adds 16 independent measurements of each point on the ground to improve the signal-to-noise ratio of each pixel. VIS images may be compressed with a lossey data compression algorithm. Both the IR and VIS data streams are downlinked through the spacecraft after lossless data compression has been applied by the hardware Rice data compression algorithm chip. Final data stream formatting for both the IR and VIS data is performed by the spacecraft processor. Further information about onboard processing is available in the THEMIS *Space Science Review* paper [6].

2.2 Data Product Overview

The four THEMIS standard data products (referred to collectively as the THM-EDR and THM-RDR data products) include raw and radiometrically calibrated image QUBEs at either thermal infrared or visible wavelengths. As discussed in the Instrument Overview (Section 2.1), one THEMIS observation results in either a visible image, an infrared image, or both an infrared image and a visible image with overlapping spatial coverage. Additional infrared images, called "reset" and "shutter" images, are collected throughout each orbit for calibration purposes. All images are stored in binary format with an attached ASCII label and header data objects.

All THEMIS standard data products are image QUBEs: VISEDR and IREDR contain raw data values; VISRDR and IRRDR contain radiometrically corrected radiance data. The label attached to each product contains identification and observation parameters associated with the image. A HISTORY data object, in ASCII format, follows the label within each product header. For raw infrared products (IREDR), the header includes a second data object containing binary telemetry information sampled regularly throughout the observation. In an image QUBE each layer contains the data from one instrument band; thus, a three band observation will result in a three layer QUBE. Available bands for each camera are listed in Table 1a&b. VIS layers are sorted into ascending wavelength order during QUBE generation. All standard data products are represented in raw raster order; geometric correction of the THM-RDR products is discussed in the THEMIS Geometric Processing User's Guide [9].

Tables 1a&b: THEMIS available bands

INFRARED BANDS		
Band Numbers	Center (µm)	FWHM (µm)
IR-1	6.78	1.01
IR-2	6.78	1.01
IR-3	7.93	1.09
IR-4	8.56	1.16
IR-5	9.35	1.20
IR-6	10.21	1.10
IR-7	11.04	1.19
IR-8	11.79	1.07
IR-9	12.57	0.81
IR-10	14.88	0.87

VISIBLE BANDS		
Band Numbers	Center (µm)	FWHM (µm)
V-1	0.425	0.049
V-2	0.540	0.051
V-3	0.654	0.053
V-4	0.749	0.053
V-5	0.860	0.045

2.3 Standards Used in Generating Data Products

2.3.1 PDS Standards

The THM-EDR and THM-RDR data products comply with Planetary Data System standards for file formats and labels, as specified in the PDS Standards Reference [4].

2.3.2 QUBE Object

All THEMIS data products make use of the PDS spectral QUBE object, adapted from the ISIS cube object and defined in the PDS Standards Reference [4]. A QUBE is an array of sample values in two or more dimensions. The “core” of a THEMIS QUBE is three-dimensional, with two spatial dimensions (samples and lines) and one spectral dimension (bands), as shown conceptually in Figure 1a. This format allows THEMIS data to be simultaneously a set of images (at different wavelengths) of the same target area, and also a multi-point spectrum at each spatially registered pixel in the target area. Additional information may be stored in “suffix” planes (back, side, or bottom) as shown in Figure 1b.

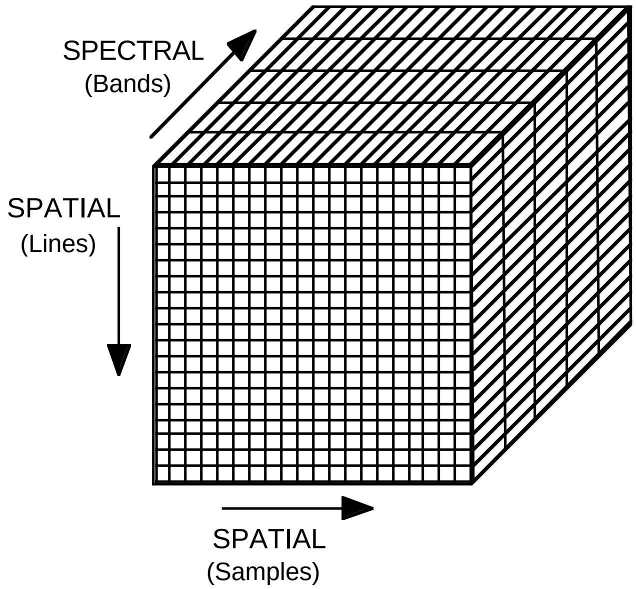

Figure 1a: THEMIS QUBE core structure

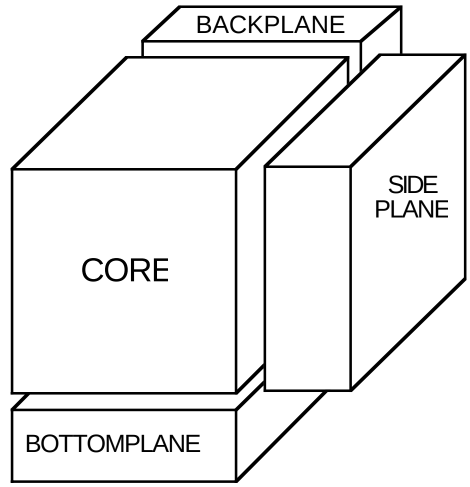

Figure 1b. Exploded view of PDS QUBE

The QUBE object has an attached label containing pertinent observation information, and header data objects (Figure 2). Required keywords, in the "keyword=value" text format of PDS labels, define QUBE structure, CORE parameters, and BAND_BIN information. The header data objects contain information related to the image; for THEMIS QUBEs these may include a HISTORY object and a telemetry TABLE object.

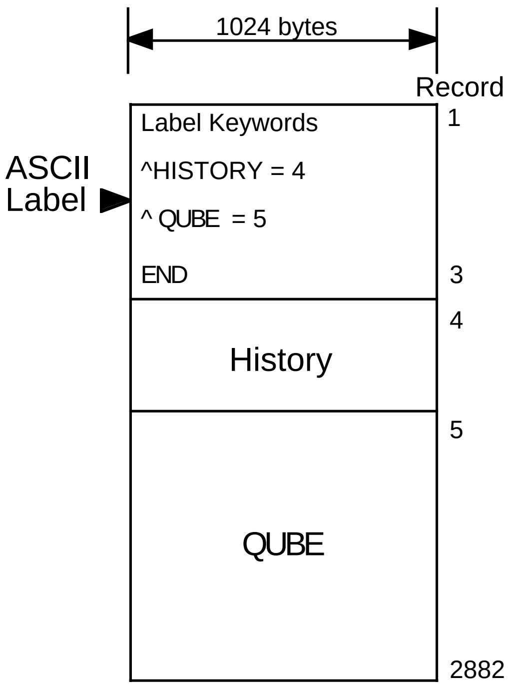

Figure 2: Example of a THEMIS VISEDR QUBE: attached label, header data object, and image QUBE

2.3.3 Time Standards

The time stamp (SPACECRAFT_CLOCK_START_COUNT) stored with each THM-EDR or THM-RDR data product is the value of the spacecraft clock at the time of data acquisition of the leading edge of the first detector in the array (filter 1), even if filter 1 is not downlinked. For VIS QUBEs, this time is calculated from the UNCORRECTED_START_TIME and may differ by as much as 4 seconds, depending on which bands are acquired in the observation.

The spacecraft clock value is equal to the number of seconds since 12:00 a.m. 1/01/1980 GMT. This number can vary from the number of seconds recorded on earth due to variations in the spacecraft's oscillator or relativistic effects. The portion of the number that occurs after the decimal point is a count of "clock tics" which are 1/256th of a second long; the decimal portion will always be between 0 and 255. THM-EDR and THM-RDR data products also contain time values in UTC (Universal Time Coordinated) and ET (Ephemeris Time) formats, translated from the spacecraft event times. UTC is the date (year, month, day) and time (hour, minute, second) in GMT. ET is the time in seconds since January 1, 2000 at 12:00:00 in Barycentric Dynamical Time (TDB).

2.3.4 Coordinate Systems

The THM-EDR and THM-RDR data products are not projected into any coordinate system. The image QUBEs are maintained in the raw raster order produced by the instrument, reorganized to group together the data from each band. The QUBE layers are not spatially registered. Layers within a single QUBE can be out of registration with each other by up to 10 lines and/or columns.

2.3.5 Orbit Numbering Conventions

The orbit number (ORBIT_NUMBER) stored with each THEMIS data product follows the convention established by the 2001 Mars Odyssey Project. During aerobraking, orbits are counted from the periapsis pass, with orbit 1 being the Mars Orbit Insertion pass. During mapping, orbits are counted from the descending equator crossing, incrementing from the last aerobraking orbit counted.

2.4 Data Product Contents

2.4.1 Data Processing Level

The THM-EDR and THM-RDR data products comply with NASA processing levels standards. THM-EDR are Level-0 spectral image QUBEs of raw THEMIS science data at the full resolution returned from the spacecraft, time ordered, with duplicates and transmission errors removed. THM-RDR are Level-1A spectral image QUBEs, radiometrically calibrated versions of the THM-EDR products.

2.4.2 Data Product Generation

The THEMIS data products will be generated by the staff at the ASU Mars Space Flight Facility. The data received on the ground are in the form of compressed, scaled, 8-bit "data numbers"

(DN). Data processing will consist of decompression, radiometric calibration, and systematic noise removal. The instrument response functions necessary to perform calibration were acquired prior to launch using a thermal vacuum chamber at the SBRS facility (see THEMIS Calibration Report [7]). A detailed discussion of the processing techniques summarized below is available in the THEMIS Data Processing User's Guide [8].

For IR data, the DN values represent the delta signal between the scene and the internal reference calibration flag. After decompression, the data is converted to scene radiance by: (1) adjusting for the gain and offset levels used during data collection; (2) correcting for drift or offset that occurs between observations of the calibration flag; and (3) converting signal to radiance using the instrument response function determined prior to launch.

For VIS data, the DN values represent relative radiance values which are converted to scene radiance by: (1) correcting for the CCD dark current with nighttime Mars images; and (2) converting signal to radiance using the instrument response function determined prior to launch. Both of the above VIS calibration steps are functions of the exposure setting of the camera, which is one of the defined image parameters available in the image label.

2.4.3 Data Product Archive

Data will be accumulated, calibrated, and validated at the ASU Mars Space Flight Facility. The size of individual data products depends on several factors: image type (VIS vs. IR), length of an image, number of bands in the image, and data type (8-bit raw vs. 16-bit calibrated). Within these parameters, a raw VIS image (VISEDR) can vary in size from 0.38 to 3.7 Mbytes; a raw IR image (IREDR) can vary in size from 0.07 to 199 Mbytes. Calibration of any of these images (VISRDR and IRRDR) increases the size by a factor of two. Validation will be conducted using the latest, best-effort algorithms available.

The estimated total volume of data to be collected over the course of the mission is limited by the available downlink allocated to THEMIS. Many factors affect the actual downlink available on any given day, which can vary from 0 to more than 400 Mbytes per day. THEMIS mission planners will maximize data collection by balancing the day's available allocated downlink against the size-defining parameters of the daily planned observations (VIS/IR, image length, number of bands).

Data products will be archived and released following the agreement outlined in the 2001 Mars Odyssey Orbiter Archive Plan [2]. Due to the large volume of data products expected from the mission, physical copies will be made for PDS long-term archive purposes only. All other data distribution will be facilitated through an online THEMIS data archive service, maintained by the ASU Mars Space Flight Facility.

2.4.4 Labeling and Identification

Each THEMIS data product is stored in a single file following the PDS SPECTRAL_QUBE format. Data products are uniquely identified by the PRODUCT_ID which is based on the abbreviated description of the product type, the data collected time, and the data processing level (see Section 3.1). File names follow the PDS convention of "PRODUCT_ID".QUB Each product has an attached PDS label (see Section 3.3), which includes a

PRODUCT_VERSION_ID keyword in the event that multiple versions of the product may be made available in the future.

3. DETAILED DATA PRODUCT SPECIFICATIONS

3.1 Data Product Structure and Organization

Each THM-EDR or THM-RDR data product is an individual image QUBE with a unique label. Data products are organized in the time-sequential order that they were collected during the mission. Each file name consists of an alphanumeric identifier following the pattern "AxxxxxnnnPPP.QUB", where

A	is a 1-letter description of the type of image collected; [V = visible image; I = infrared image; R = infrared reset image; S = infrared shutter image]
xxxxx	is a 5-digit mission orbit number when the image was collected; [01000 = mapping orbit number example]
nnn	is a 3-digit image sequence number indicating the order that images were collected each orbit; [001 = first image collected in the xxxxx orbit]
PPP	is a 3-letter description of the processing level of the image data; [EDR = raw data; RDR = radiometrically calibrated data]
.QUB	is a 3-letter extension describing this product as a spectral QUBE

More information, including mission orbit numbers, spacecraft clock times, processing dates, and version numbers, are accessible in the ASCII label described in Section 3.3 below.

3.2 Data Format

The THM-EDR data products are uncompressed, binary, band-sequential QUBEs of 8-bit integers. The image width is fixed (320 pixels for IR, 1024 pixels for VIS), but the length varies proportional to the duration of the observation. The number of layers in a THM-EDR QUBE corresponds to the number of bands selected for the observation: an IREDR may have up to 10 layers; a VISEDR may have up to 5 layers.

The format of the THM-RDR QUBEs is identical to the source THM-EDR QUBE, except that the data are stored as 32-bit IEEE real.

For IR QUBEs, missing data pixels are set to the CORE_NULL value and the total count of missing lines is stored in the MISSING_SCAN_LINES keyword. For VISEDR QUBEs, missing data pixels are either filled with zero values, if several complete lines are missing, or they are filled with a pattern of values, if a section of a line is missing. In VISRDR QUBEs, the missing data pixels are set to zero.

3.3 Labels

The PDS label describes the structure, content, and observation specifications of the data. It is attached as ASCII text at the beginning of each image QUBE. Information in the label are stored in a "keyword=value" text format and structured in the Object Definition Language (ODL) of PDS. Example labels are shown in Appendices A.1-A.4; individual keyword items are defined in Appendices A.5-A.8.

3.3.1 File Identification and Structure Label

The first lines of the label are the file identification keywords and associated values. Next are the file structure keywords, which define the number and size of records in the data file, followed by the pointer keywords, which define the start byte of the header data objects and the image QUBE. Finally, "identification data elements" define parameters of the mission, spacecraft, instrument team, and data stream. See Appendix A.5 for a detailed description of these keywords.

3.3.2 HISTORY Object Structure

The HISTORY object structure keywords define the size and format of the data object stored later in the header. The HISTORY object itself is a structured series of text entries identifying all previous computer manipulations of the data in the file; the format is not intended to be complient with PDS-ODL standards. HISTORY entries may include identification of source data, processes performed, processing parameters, and dates and times of processing. See Appendix A.6 for a detailed description of the entries and keywords used with THEMIS HISTORY objects.

3.3.3 Telemetry Table Object Structure

The telemetry (TLM) table is only available in the raw infrared data products (IREDR). The TLM table object structure keywords define the size and format of the table object stored later in the header. See Appendix A.7 for a detailed content description of the TLM table.

The TLM table itself follows the PDS TABLE structure using fixed length binary records sorted time-sequentially. The table structure is defined in an external, ASCII file identified in the pointer keyword as "tlm.fmt". It contains details such as the table dimensions, a general description of the telemetry data source, and definitions of each table column. Column definitions include the following details: name, starting position (in bytes), size (in bytes), data type, description, and scaling factors if applicable. In some cases, the column being described is composed of multiple bit-fields; the individual meaning of each bit-field is described with the same details listed above.

The TLM table records can be accessed using the DAVINCI software package described in Section 4.1 below.

3.3.4 QUBE Object Label

The QUBE object keywords make up the bulk of the label and are organized by the following sub-structure descriptions:

QUBE structure	- parameters of the multidimensional array (image)
CORE description	- parameters of the array elements (pixels)
Observation parameters	- operational modes of the instrument for this image
Band-bins	- parameters of the layers (bands) in the array

See Appendix A.8 for a detailed description of the keywords used in the QUBE label.

4. APPLICABLE SOFTWARE

4.1 Utility Programs

The THEMIS team uses the software tools DAVINCI and ISIS to display and analyze the image QUBEs. DAVINCI is a data analysis package for working with images and image QUBEs. DAVINCI is distributed by ASU and is available at http://tes.asu.edu/software. ISIS is an image processing package produced by USGS - Flagstaff and is available at http://astrogeology.usgs.gov/Projects/ISIS.

The software tool VANILLA is used to extract the telemetry (TLM) table object embedded in the image header. Vanilla was produced by the MGS-TES team at ASU to read and manipulate PDS tables and the variable-length records. Since DAVINCI can extract and read the TLM table, most users will not need to acquire VANILLA, however, the software is available at http://tes.asu.edu/software.

4.2 Applicable PDS Software Tools

The THEMIS team uses no PDS software to view, manipulate or process the data. However, the images are stored and labeled using the PDS QUBE standard structure and any tool that understands that structure should be able to view them.

A. APPENDICIES

Appendices A.1-4 contains example labels from THEMIS IREDR, VISEDR, IRRDR, and VISRDR image QUBEs. Definitions of individual items contained in the label are given in the following appendices. The items defined in Appendices A.5 File and Data Identification Items and A.8 QUBE Object Items, are listed in the order of appearance within a QUBE label; "valid values" for each item are shown in [ ] at end of each description, as appropriate. Appendix A.6 contains example THM-EDR and IRRDR HISTORY objects and definitions for the basic HISTORY items used. Appendix A.7, Telemetry Table Structure contains a copy of the "tlm.fmt" file which defines and describes that object.

A.1 Example Label: IREDR

An example IREDR label is shown below:

```
PDS_VERSION_ID = PDS3

/* File Identification and Structure */
RECORD_TYPE = “FIXED_LENGTH”
RECORD_BYTES = 320
FILE_RECORDS = 18087
LABEL_RECORDS = 8

/* Pointers to Data Objects */
^HISTORY = 9
^TABLE = 11
^SPECTRAL_QUBE = 12

/* Identification Data Elements */
MISSION_NAME = “2001 MARS ODYSSEY”
INSTRUMENT_HOST_NAME  = “2001 MARS ODYSSEY”
INSTRUMENT_NAME  = “THERMAL EMISSION IMAGING SYSTEM”
INSTRUMENT_ID  = “THEMIS”
DETECTOR_ID = “IR”
MISSION_PHASE_NAME = “MAPPING”
TARGET_NAME = “MARS”
PRODUCT_ID = “I00013007EDR”
PRODUCER_ID = “ODY_THM_TEAM”
DATA_SET_ID = “ODY-M-THM-2-IREDR-V1.0”
PRODUCT_CREATION_TIME = 2002-03-08T21:54:02
PRODUCT_VERSION_ID = 1.0
START_TIME = 2001-11-02T14:38:30.010
STOP_TIME = 2001-11-02T14:39:30.271
SPACECRAFT_CLOCK_START_COUNT = 689179146.000
SPACECRAFT_CLOCK_STOP_COUNT = 689179206.067
START_TIME_ET = 57983974.192
```

```
STOP_TIME_ET = 57984034.453
ORBIT_NUMBER = 00013

/* History Object Structure */
OBJECT = HISTORY
   BYTES = 640
   HISTORY_TYPE = CUSTOM
   INTERCHANGE_FORMAT = ASCII
END_OBJECT = HISTORY

/* Telemetry Table Structure */
OBJECT = TABLE
   NAME = TLM
   ROWS =  2
   ^STRUCTURE = “tlm.fmt”
END_OBJECT = TABLE

OBJECT = SPECTRAL_QUBE

   /* QUBE Structure */
   AXES = 3
   AXIS_NAME = (SAMPLE, LINE, BAND)

   /* Core Description */
   CORE_ITEMS= (320, 1808, 10)
   CORE_NAME = “RAW_DATA_NUMBER”
   CORE_ITEM_BYTES = 1
   CORE_ITEM_TYPE = MSB_UNSIGNED_INTEGER
   CORE_BASE = 0.0
   CORE_MULTIPLIER = 1.0
   CORE_UNIT = “DIMENSIONLESS”
   CORE_NULL = 0

   /* Observation Parameters */
   FLIGHT_SOFTWARE_VERSION_ID = “1.00”
   COMMAND_SEQUENCE_NUMBER = 13
   IMAGE_ID = 7
   DESCRIPTION = “Example IR image”
   INST_CMPRS_RATIO = 2.70
   UNCORRECTED_START_TIME = 689179146.000
   GAIN_NUMBER = 8
   OFFSET_NUMBER = 0
   TIME_DELAY_INTEGRATION_FLAG = “ENABLED”
   MISSING_SCAN_LINES = 0
   MD5_CHECKSUM = “fe027fe2ca98562a1d61e0d6be3284d0”
```

```
    /*Band Bins */
    GROUP = BAND_BIN
      BAND_BIN_FILTER_NUMBER = (1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
      BAND_BIN_ BAND_NUMBER = (1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
      BAND_BIN_CENTER = (6.78, 6.78, 7.93, 8.56, 9.35, 10.21, 11.04, 11.79, 12.57,
                         14.88)
      BAND_BIN_WIDTH = ( 1.01, 1.01, 1.09, 1.16, 1.20, 1.10, 1.19, 1.07, 0.81, 0.87 )
      BAND_BIN_UNIT = “MICROMETER”
    END_GROUP = BAND_BIN

  END_OBJECT = SPECTRAL_QUBE

  END
```

A.2 Example Label: VISEDR

An example VISEDR label is shown below:

```
PDS_VERSION_ID = PDS3

/* File Identification and Structure */
RECORD_TYPE = “FIXED_LENGTH”
RECORD_BYTES = 1024
FILE_RECORDS = 2882
LABEL_RECORDS = 2

/* Pointers to Data Objects */
^HISTORY = 3
^SPECTRAL_QUBE = 4

/* Identification Data Elements */
MISSION_NAME = “2001 MARS ODYSSEY”
INSTRUMENT_HOST_NAME  = “2001 MARS ODYSSEY”
INSTRUMENT_NAME  = “THERMAL EMISSION IMAGING SYSTEM”
INSTRUMENT_ID  = “THEMIS”
DETECTOR_ID = “VIS”
MISSION_PHASE_NAME = “MAPPING”
TARGET_NAME = “MARS”
PRODUCT_ID = “V00013003EDR”
PRODUCER_ID = “ODY_THM_TEAM”
DATA_SET_ID = “ODY-M-THM-2-VISEDR-V1.0”
PRODUCT_CREATION_TIME = 2002-03-08T21:45:02
PRODUCT_VERSION_ID = 1.0
START_TIME = 2001-11-02T14:38:49.010
STOP_TIME = 2001-11-02T14:39:08.209
```

```
SPACECRAFT_CLOCK_START_COUNT = 689179165.000
SPACECRAFT_CLOCK_STOP_COUNT = 689179184.051
START_TIME_ET = 57983993.192
STOP_TIME_ET = 57984012.391
ORBIT_NUMBER = 00013

/* History Object Structure */
OBJECT = HISTORY
   BYTES = 1024
   HISTORY_TYPE = CUSTOM
   INTERCHANGE_FORMAT = ASCII
END_OBJECT = HISTORY

OBJECT = SPECTRAL_QUBE

   /* QUBE Structure */
   AXES = 3
   AXIS_NAME = (SAMPLE, LINE, BAND)

   /* Core Description */
   CORE_ITEMS= (1024, 576, 5)
   CORE_NAME = “RAW_DATA_NUMBER”
   CORE_ITEM_BYTES = 1
   CORE_ITEM_TYPE = MSB_UNSIGNED_INTEGER
   CORE_BASE = 0.0
   CORE_MULTIPLIER = 1.0
   CORE_UNIT = “DIMENSIONLESS”
   CORE_NULL = 0

   /* Observation Parameters */
   FLIGHT_SOFTWARE_VERSION_ID = “1.00”
   COMMAND_SEQUENCE_NUMBER = 13
   IMAGE_ID = 3
   DESCRIPTION = “Example VIS image”
   INST_CMPRS_RATIO = 1.93
   UNCORRECTED_START_TIME = 689179165.000
   INST_CMPRS_NAME = “PREDICTIVE”
   FOCAL_PLANE_TEMPERATURE = -0.42
   EXPOSURE_DURATION = 3.000
   INTERFRAME_DELAY = 1.000
   SPATIAL_SUMMING = 1
   MD5_CHECKSUM = “851ab2a81c55db940fc59200d9ba6f6f”

   /*Band Bins */
   GROUP = BAND_BIN
```

```
        BAND_BIN_FILTER_NUMBER = (2, 5, 3, 4, 1)
        BAND_BIN _BAND_NUMBER = (1, 2, 3, 4, 5)
        BAND_BIN_CENTER = (0.425, 0.540, 0.654, 0.749, 0.860)
        BAND_BIN_WIDTH = ( 0.049, 0.051, 0.053, 0.053, 0.045 )
        BAND_BIN_UNIT = "MICROMETER"
    END_GROUP = BAND_BIN

END_OBJECT = SPECTRAL_QUBE

END
```

A.3 Example Label: IRRDR

An example IRRDR label is shown below:

```
PDS_VERSION_ID = PDS3


/* File Identification and Structure */
RECORD_TYPE = "FIXED_LENGTH"
RECORD_BYTES = 644
FILE_RECORDS = 18087
LABEL_RECORDS = 5

/* Pointers to Data Objects */
^HISTORY = 6
^SPECTRAL_QUBE = 10

/* Identification Data Elements */
MISSION_NAME = "2001 MARS ODYSSEY"
INSTRUMENT_HOST_NAME  = "2001 MARS ODYSSEY"
INSTRUMENT_NAME  = "THERMAL EMISSION IMAGING SYSTEM"
INSTRUMENT_ID  = "THEMIS"
DETECTOR_ID = "IR"
MISSION_PHASE_NAME = "MAPPING"
TARGET_NAME = "MARS"
PRODUCT_ID = "I00013007RDR"
PRODUCER_ID = "ODY_THM_TEAM"
DATA_SET_ID = "ODY-M-THM-3-IRRDR-V1.0"
PRODUCT_CREATION_TIME = 2002-03-08T22:00:02
PRODUCT_VERSION_ID = 1.0
START_TIME = 2001-11-02T14:38:30.010
STOP_TIME = 2001-11-02T14:39:30.271
SPACECRAFT_CLOCK_START_COUNT = 689179146.000
SPACECRAFT_CLOCK_STOP_COUNT = 689179206.067
START_TIME_ET = 57983974.192
STOP_TIME_ET = 57984034.453
```

```
ORBIT_NUMBER = 00013

/* History Object Structure */
OBJECT = HISTORY
   BYTES = 1932
   HISTORY_TYPE = CUSTOM
   INTERCHANGE_FORMAT = ASCII
END_OBJECT = HISTORY

OBJECT = SPECTRAL_QUBE

   /* QUBE Structure */
   AXES = 3
   AXIS_NAME = (SAMPLE, LINE, BAND)

   /* Core Description */
   CORE_ITEMS= (320, 1808, 10)
   CORE_NAME = “CALIBRATED_SPECTRAL_RADIANCE”
   CORE_ITEM_BYTES = 4
   CORE_ITEM_TYPE = SUN_REAL
   CORE_BASE = 0.000000
   CORE_MULTIPLIER = 1.000000
   CORE_UNIT = “WATT*CM**-2*SR**-1*UM**-1”
   CORE_NULL = 16#FF7FFFFB#
   CORE_VALID_MINIMUM = 16#FF7FFFFA#
   CORE_LOW_REPR_SATURATION = 16#FF7FFFFC#
   CORE_LOW_INSTR_SATURATION = 16#FF7FFFFD#
   CORE_HIGH_REPR_SATURATION = 16#FF7FFFFF#
   CORE_HIGH_INSTR_SATURATION = 16#FF7FFFFE#

   /* Suffix Description */
   SUFFIX_ITEMS = (1, 1, 0)
   SUFFIX_BYTES = 4
   SAMPLE_SUFFIX_NAME = HORIZONTAL_DESTRIPE
   SAMPLE_SUFFIX_ITEM_BYTES = 2
   SAMPLE_SUFFIX_ITEM_TYPE = MSB_INTEGER
   SAMPLE_SUFFIX_BASE = -0.001143
   SAMPLE_SUFFIX_MULTIPLIER = 0.002281
   SAMPLE_SUFFIX_VALID_MINIMUM = 16#FF7FFFFA#
   SAMPLE_SUFFIX_NULL = 16#FF7FFFFB#
   SAMPLE_SUFFIX_LOW_REPR_SATURATION = 16#FF7FFFFC#
   SAMPLE_SUFFIX_LOW_INSTR_SATURATION = 16#FF7FFFFD#
   SAMPLE_SUFFIX_HIGH_REPR_SATURATION = 16#FF7FFFFF#
   SAMPLE_SUFFIX_HIGH_INSTR_SATURATION = 16#FF7FFFFE#
   LINE_SUFFIX_NAME = VERTICAL_DESTRIPE
```

```
    LINE_SUFFIX_ITEM_BYTES = 2
    LINE_SUFFIX_ITEM_TYPE = MSB_INTEGER
    LINE_SUFFIX_BASE = -0.000626
    LINE _SUFFIX_MULTIPLIER = 0.00747
    LINE _SUFFIX_VALID_MINIMUM = 16#FF7FFFFA#
    LINE _SUFFIX_NULL = 16#FF7FFFFB#
    LINE _SUFFIX_LOW_REPR_SATURATION = 16#FF7FFFFC#
    LINE _SUFFIX_LOW_INSTR_SATURATION = 16#FF7FFFFD#
    LINE _SUFFIX_HIGH_REPR_SATURATION = 16#FF7FFFFF#
    LINE _SUFFIX_HIGH_INSTR_SATURATION = 16#FF7FFFFE#

    /* Observation Parameters */
    FLIGHT_SOFTWARE_VERSION_ID = “1.00”
    COMMAND_SEQUENCE_NUMBER = 13
    IMAGE_ID = 7
    DESCRIPTION = “Example IR image”
    INST_CMPRS_RATIO = 2.70
    UNCORRECTED_START_TIME = 689179146.000
    GAIN_NUMBER = 8
    OFFSET_NUMBER = 0
    TIME_DELAY_INTEGRATION_FLAG = “ENABLED”
    MISSING_SCAN_LINES = 0
    MD5_CHECKSUM = “cbfa3fbc6b5304ffa2976fe795e4931f”

    /*Band Bins */
    GROUP = BAND_BIN
       BAND_BIN_FILTER_NUMBER = (1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
       BAND_BIN _BAND_NUMBER = (1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
       BAND_BIN_CENTER = (6.78, 6.78, 7.93, 8.56, 9.35, 10.21, 11.04, 11.79, 12.57,
                          14.88)
       BAND_BIN_WIDTH = ( 1.01, 1.01, 1.09, 1.16, 1.20, 1.10, 1.19, 1.07, 0.81, 0.87 )
       BAND_BIN_UNIT = “MICROMETER”
    END_GROUP = BAND_BIN

  END_OBJECT = SPECTRAL_QUBE

END
```

A.4 Example Label: VISRDR

An example VISRDR label is shown below:

The contents of this label are to be determined.

A.5 File and Data Identification Items

PDS_VERSION_ID
PDS version number for the label format. [PDS3]

RECORD_TYPE
Style of records in this file. [“FIXED_LENGTH”]

RECORD_BYTES
Number of bytes per record. [320 (for IREDR), 1024 (for VISEDR), 644 (for IRRDR), or 2048 (for VISRDR)]

FILE_RECORDS
Number of records in this file, including labels and data.

LABEL_RECORDS
Number of records used for label data; value does not include records in the Telemetry table or HISTORY object.

MISSION_NAME
Name of the mission including the THEMIS instrument. [“2001 MARS ODYSSEY”]

INSTRUMENT_HOST_NAME
Name of the host spacecraft for the THEMIS instrument. [“2001 MARS ODYSSEY”]

INSTRUMENT_NAME
Proper name of the instrument. [“THERMAL EMISSION IMAGING SYSTEM”]

INSTRUMENT_ID
Abbreviated name of instrument used to collect this image. [“THEMIS”]

DETECTOR_ID
Abbreviated name of camera used to collect this image. [“IR” or “VIS”]

MISSION_PHASE_NAME
Mission phase during which this image was collected. [“MAPPING”]

TARGET_NAME
The name of the target observed in the image. [“MARS”]

PRODUCT_ID
Unique identifier for each image commanded. [“AxxxxxnnnEDR” or “AxxxxxnnnRDR”]

PRODUCER_ID
Identity of the producer of this dataset. [“ODY_THM_TEAM”]

DATA_SET_ID
Unique alphanumeric identifier of this dataset. [“ODY-M-THM-2-IREDR-V1.0” or “ODY-M-THM-2-VISEDR-V1.0” or “ODY-M-THM-3-IRRDR-V1.0” or “ODY-M-THM-3-VISRDR-V1.0”]

PRODUCT_CREATION_TIME
Time of creation of this QUBE on the ground (in UTC). [yyyy-mm-ddThh:mm:ss]

PRODUCT_VERSION_ID
    Version identification of this QUBE.

START_TIME
    The time of data acquisition of the leading edge of the detector array (filter 1), even if filter 1 is not downlinked; given in spacecraft event time (SCET), UTC format. [yyyy-mm-ddThh:mm:ss.fff]

STOP_TIME
    The time of the end of data acquisition in spacecraft event time (SCET), UTC format. [yyyy-mm-ddThh:mm:ss.fff]

SPACECRAFT_CLOCK_START_COUNT
    The value of the spacecraft clock at the time of data acquisition of the leading edge of the detector array (filter 1), even if filter 1 is not downlinked; given in seconds.

SPACECRAFT_CLOCK_STOP_COUNT
    The time on the spacecraft clock at the end of data acquisition (in seconds).

START_TIME_ET
    The time of data acquisition of the leading edge of the detector array (filter 1), even if filter 1 is not downlinked; given in spacecraft event time (SCET), ET format.

STOP_TIME_ET
    The time of the end of data acquisition in spacecraft event time (SCET), ET format.

ORBIT_NUMBER
    Spacecraft orbit during which this image was observed.

A.6 HISTORY Object Items and Examples

The HISTORY data object is described within the THM-EDR and THM-RDR labels by the following keywords:

BYTES
    Number of bytes in the HISTORY object.

HISTORY_TYPE
    Identifies the software compliance of the HISTORY object format. [CUSTOM]

INTERCHANGE_FORMAT
    Identifies the manner in which the HISTORY object data items are stored. [ASCII]

Each program that operates on the data product will generate a new "history entry", similar in format to the examples shown here, and will concatenate the new entry onto the existing HISTORY object.

An example THM-EDR HISTORY object is shown below:

```
GROUP                 = SFDU2CUBE
  DATE_TIME           = 2002-02-15T13:15:00
  SOFTWARE_DESC       = "Translation of data format from SFDU into raw image
                         QUBE (THM-EDR).  Removes SFDU headers and
```

```
                                   unpackages data; returns an individual spectral image
                                   QUBE (THM-EDR) containing raw DN, with  missing data
                                   CORE_NULL filled and an attached PDS label."
        VERSION_ID               = 1.54
        USER_NAME                = "murray@east"
        USER_NOTE                = ""

        GROUP                      = PARAMETERS
            START_SFDU_ID          = 689179146-2
            STOP_SFDU_ID           = 689179206-4
            MISSING_PACKETS        = 0
        END_GROUP                  = PARAMETERS
    END_GROUP                    = SFDU2CUBE
END
```

An example IRRDR HISTORY object is shown below:

```
    GROUP                        = CAL_IR_IMAGE
        DATE_TIME                = 2002-02-15T13:20:00
        SOFTWARE_DESC            = "Calibration of a raw, infrared image (IREDR).  Uses DN,
                                   gain, and offset values from raw image with the instrument
                                   response function and a calibration flag image (IREDR);
                                   returns a calibrated spectral radiance image (IRRDR) in (W
                                   cm-2 str-1 μm-1)."
        VERSION_ID               = 1.0
        USER_NAME                = "murray@east"
        USER_NOTE                = ""

        GROUP                        = PARAMETERS
            IREDR_FILE               = "I0013007EDR.QUB"
            IR_IMG_CAL_QUBE_VER      = 4.5
            IRF_FILE                 = "/themis/calib/irf_fit_all_v2.0_tv6_1_1_v3.0"
            TEMP2RAD_FILE            = "/themis/calib/temp_rad_v4"
            CALIB_FLAG_IMAGE         = "S0013008EDR.QUB"
            CALIB_FLAG_DN            = (193.034, 193.656, 193.353, 192.725, 192.453,
                                         193.044, 193.044, 193.044, 193.044, 193.044)
            CALIB_FLAG_FILTER        = 1
            CALIB_FLAG_OPTION        = 2
            CALIB_FLAG_TEMP          = -7.66
            DESTRIPE_OPTION_X        = 3
            DESTRIPE_OPTION_Y        = 3
            DESTRIPE_FILTER_X        = 9
            DESTRIPE_FILTER_Y        = 9
            DEGHOST_OPTION           = 0
        END_GROUP                    = PARAMETERS
    END_GROUP                    = CAL_IR_IMAGE
END
```

Basic HISTORY object keywords are defined below:

GROUP
: The first use of this item identifies the name of the program that generated the history entry. The second use is followed with the value "PARAMETER" and is used to delineate the statements specifying the parameters of the program.

DATE_TIME
: Date and time, in UTC standard format, that the program was executed. [yyyy-mm-ddThh:mm:ss]

SOFTWARE_DESC
: Program generated description and execution notes.

VERSION_ID
: Program version number.

USER_NAME
: Username and name of computer. ["smith@east"]

USER_NOTE
: User supplied brief description of program; may be blank.

"PARAMETERS"
: Multiple "KEYWORD = value" style statements specifying the values of significant parameters used in the execution of the program.

A.7 Telemetry Table Structure (tlm.fmt)

```
COLUMNS                  = 41
ROW_BYTES                = 46
INTERCHANGE_FORMAT       = BINARY
DESCRIPTION              = "
  The TLM table stores the THEMIS telemetry parameters downlinked with all IR images in
  the housekeeping telemetry data frame.  One record in the TLM table represents one
  housekeeping telemetry data frame.  For each requested IR image, one housekeeping
  telemetry data frame is collected immediately preceding the first image data frame, another is
  collected every 2048 data frames (68.267 seconds) throughout the image, and a final one is
  collected after the last image data frame.

  Bytes 7, 9-(bits 1-6), 11-(bits 12,4, 9-16), 41, 43-(bits 1-4), and 44-(bits 1-2) are spares
  reserved for future use with a value set to either 0 or 1.  Valid values are defined between [ ]
  in the column description, as appropriate."


  OBJECT                 = COLUMN
    NAME                 = SYNC
    DATA_TYPE            = MSB_UNSIGNED_INTEGER
    START_BYTE           = 1
```

```
        BYTES                 = 2
        DESCRIPTION           = "Indicates frame synchronization at the beginning of each
                                frame.  [1111 0000 1100 1010]"
    END_OBJECT                = COLUMN

    OBJECT                    = COLUMN
        NAME                  = IMAGE_ID
        DATA_TYPE             = MSB_UNSIGNED_INTEGER
        START_BYTE            = 3
        BYTES                 = 1
        DESCRIPTION           = "Number of image counted sequentially within each orbit."
    END_OBJECT                = COLUMN

    OBJECT                    = COLUMN
        NAME                  = TELEMETRY_TYPE
        DATA_TYPE             = MSB_UNSIGNED_INTEGER
        START_BYTE            = 4
        BYTES                 = 1
        DESCRIPTION           = "Identifies packet within datastream as a telemetry frame.
                                [0000 1111] = frame from start or middle of an image
                                [0000 1110] = frame from end of image"
    END_OBJECT                = COLUMN

    OBJECT                    = COLUMN
        NAME                  = FRAME_COUNT
        DATA_TYPE             = MSB_UNSIGNED_INTEGER
        START_BYTE            = 5
        BYTES                 = 2
        DESCRIPTION           = "Frame count from start of image acquisition; increments
                                by 2048 for telemetry frames collected in the middle of the
                                image."
    END_OBJECT                = COLUMN

    OBJECT                    = COLUMN
        NAME                  = SPARE7
        DATA_TYPE             = MSB_UNSIGNED_INTEGER
        START_BYTE            = 7
        BYTES                 = 1
        DESCRIPTION           = "Reserved for future use"
    END_OBJECT                = COLUMN

    OBJECT                    = COLUMN
        NAME                  = IMAGE_LENGTH
        DATA_TYPE             = MSB_UNSIGNED_INTEGER
        START_BYTE            = 8
        BYTES                 = 1
        DESCRIPTION           = "Command value used to define the final size of the image
                                in frames; final image is determined using:
```

```
                                 ((IMAGE_LENGTH+1)*256)-240.
                               [1:255]"
END_OBJECT                   = COLUMN

OBJECT                       = COLUMN
    NAME                     = BAND_ENABLED
    DATA_TYPE                = MSB_BIT_STRING
    START_BYTE               = 9
    BYTES                    = 2
    DESCRIPTION              = "Bit-word defining the band mask used for this image."

    OBJECT                       = BIT_COLUMN
        NAME                     = SPARE9_1
        BIT_DATA_TYPE            = MSB_UNSIGNED_INTEGER
        START_BIT                = 1
        BITS                     = 6
        DESCRIPTION              = "Reserved for future use"
    END_OBJECT                   = BIT_COLUMN

    OBJECT                       = BIT_COLUMN
        NAME                     = BAND_MASK
        BIT_DATA_TYPE            = MSB_UNSIGNED_INTEGER
        START_BIT                = 7
        BITS                     = 10
        DESCRIPTION              = "Flag indicating whether the band is ON [1] or OFF
                                   [0]; one bit per band, stored numerically according to
                                   band number (e.g. bit 7 = Band 1, bit 8 = Band 2 … bit
                                   16 = Band 10)."
    END_OBJECT                   = BIT_COLUMN
END_OBJECT                   = COLUMN

OBJECT                       = COLUMN
    NAME                     = IRS_STATUS
    DATA_TYPE                = MSB_BIT_STRING
    START_BYTE               = 11
    BYTES                    = 2
    DESCRIPTION              = "Bit-word defining calibration flag and latchup status for
                               this image; see individual bit items below"

    OBJECT                       = BIT_COLUMN
        NAME                     = CALIB_FLAG_PRIMARY
        BIT_DATA_TYPE            = MSB_UNSIGNED_INTEGER
        START_BIT                = 1
        BITS                     = 1
        DESCRIPTION              = "Status of calibration flag primary motor control (from
                                   IRS).
                                   [0] = Closed
                                   [1] = Open"
```

```
END_OBJECT              = BIT_COLUMN

OBJECT                  = BIT_COLUMN
    NAME                = SPARE11_2
    BIT_DATA_TYPE       = MSB_UNSIGNED_INTEGER
    START_BIT           = 2
    BITS                = 1
    DESCRIPTION         = “Reserved for future use”
END_OBJECT              = BIT_COLUMN

OBJECT                  = BIT_COLUMN
    NAME                = CALIB_FLAG_REDUNDANT
    BIT_DATA_TYPE       = MSB_UNSIGNED_INTEGER
    START_BIT           = 3
    BITS                = 1
    DESCRIPTION         = “Status of calibration flag redundant motor control
                          (from IRS).
                          [0] = Closed
                          [1] = Open”
END_OBJECT              = BIT_COLUMN

OBJECT                  = BIT_COLUMN
    NAME                = SPARE11_4
    BIT_DATA_TYPE       = MSB_UNSIGNED_INTEGER
    START_BIT           = 4
    BITS                = 1
    DESCRIPTION         = “Reserved for future use”
END_OBJECT              = BIT_COLUMN

OBJECT                  = BIT_COLUMN
    NAME                = LATCHUP_SENSITIVITY
    BIT_DATA_TYPE       = MSB_UNSIGNED_INTEGER
    START_BIT           = 5
    BITS                = 1
    DESCRIPTION         = “Latchup control circuit sensitivity state (from IRS).
                          [0] = Low
                          [1] = High”
END_OBJECT              = BIT_COLUMN

OBJECT                  = BIT_COLUMN
    NAME                = LATCHUP_TRIGGER
    BIT_DATA_TYPE       = MSB_UNSIGNED_INTEGER
    START_BIT           = 6
    BITS                = 1
    DESCRIPTION         = “Latchup protection circuit status (from IRS).
                          [0] = Off
                          [1] = On”
END_OBJECT              = BIT_COLUMN
```

```
    OBJECT                    = BIT_COLUMN
      NAME                    = RICE
      BIT_DATA_TYPE           = MSB_UNSIGNED_INTEGER
      START_BIT               = 7
      BITS                    = 1
      DESCRIPTION             = "Status of onboard, lossless compression algorithm
                                [0] = Enabled
                                [1] = Disabled"
    END_OBJECT                = BIT_COLUMN

    OBJECT                    = BIT_COLUMN
      NAME                    = TDI_ENABLE
      BIT_DATA_TYPE           = MSB_UNSIGNED_INTEGER
      START_BIT               = 8
      BITS                    = 1
      DESCRIPTION             = "Status of onboard Time Delay Integration (TDI)
                                algorithm
                                [0] = Off
                                [1] = On"
    END_OBJECT                = BIT_COLUMN

    OBJECT                    = BIT_COLUMN
      NAME                    = SPARE11_9
      BIT_DATA_TYPE           = MSB_UNSIGNED_INTEGER
      START_BIT               = 9
      BITS                    = 8
      DESCRIPTION             = "Reserved for future use"
    END_OBJECT                = BIT_COLUMN
  END_OBJECT                = COLUMN

  OBJECT                    = COLUMN
    NAME                    = SECONDARY_MIRROR_TEMP
    DATA_TYPE               = MSB_UNSIGNED_INTEGER
    START_BYTE              = 13
    BYTES                   = 1
    OFFSET                  = -50
    SCALING_FACTOR          = 0.3195
    UNIT                    = "C"
    DESCRIPTION             = "Secondary mirror temperature."
  END_OBJECT                = COLUMN

  OBJECT                    = COLUMN
    NAME                    = PRIMARY_MIRROR_TEMP
    DATA_TYPE               = MSB_UNSIGNED_INTEGER
    START_BYTE              = 14
    BYTES                   = 1
    OFFSET                  = -50
```

```
    SCALING_FACTOR      = 0.3195
    UNIT                = "C"
    DESCRIPTION         = "Primary mirror temperature."
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = FLAG_TEMP
    DATA_TYPE           = MSB_UNSIGNED_INTEGER
    START_BYTE          = 15
    BYTES               = 1
    OFFSET              = -50
    SCALING_FACTOR      = 0.3195
    UNIT                = "C"
    DESCRIPTION         = "Calibration flag assembly temperature."
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = IRS_TEMP
    DATA_TYPE           = MSB_UNSIGNED_INTEGER
    START_BYTE          = 16
    BYTES               = 1
    OFFSET              = -50
    SCALING_FACTOR      = 0.3195
    UNIT                = "C"
    DESCRIPTION         = "Infrared Subsystem (IRS) electronics temperature."
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = IR_TEMP
    DATA_TYPE           = MSB_UNSIGNED_INTEGER
    START_BYTE          = 17
    BYTES               = 1
    OFFSET              = -50
    SCALING_FACTOR      = 0.3195
    UNIT                = "C"
    DESCRIPTION         = "Infrared detective assembly (IRDA) temperature."
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = BEAMSPLITTER_TEMP
    DATA_TYPE           = MSB_UNSIGNED_INTEGER
    START_BYTE          = 18
    BYTES               = 1
    OFFSET              = -50
    SCALING_FACTOR      = 0.3195
    UNIT                = "C"
    DESCRIPTION         = "Dichroic beamsplitter assembly temperature."
END_OBJECT              = COLUMN
```

```
OBJECT                = COLUMN
    NAME              = TERT_MIRROR_TEMP
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    START_BYTE        = 19
    BYTES             = 1
    OFFSET            = -50
    SCALING_FACTOR    = 0.3195
    UNIT              = "C"
    DESCRIPTION       = "Tertiary mirror temperature."
END_OBJECT            = COLUMN

OBJECT                = COLUMN
    NAME              = IRIS_1_TEMP
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    START_BYTE        = 20
    BYTES             = 1
    OFFSET            = -50
    SCALING_FACTOR    = 0.3195
    UNIT              = "C"
    DESCRIPTION       = "Infrared Imaging System (IRIS) housing temperature from
                        sensor 1."
END_OBJECT            = COLUMN

OBJECT                = COLUMN
    NAME              = IRIS_2_TEMP
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    START_BYTE        = 21
    BYTES             = 1
    OFFSET            = -50
    SCALING_FACTOR    = 0.3195
    UNIT              = "C"
    DESCRIPTION       = "Infrared Imaging System (IRIS) housing temperature from
                        sensor 2."
END_OBJECT            = COLUMN

OBJECT                = COLUMN
    NAME              = BAFFLE_TEMP
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    START_BYTE        = 22
    BYTES             = 1
    OFFSET            = -50
    SCALING_FACTOR    = 0.3195
    UNIT              = "C"
    DESCRIPTION       = "Main baffle temperature."
END_OBJECT            = COLUMN

OBJECT                = COLUMN
    NAME              = CONVERTER_P12V
```

```
        DATA_TYPE              = MSB_UNSIGNED_INTEGER
        START_BYTE             = 23
        BYTES                  = 1
        OFFSET                 = -1.4634
        SCALING_FACTOR         = 0.09565
        UNIT                   = “VOLT”
        DESCRIPTION            = “+12V voltage measurement at DC/DC converter.”
    END_OBJECT                 = COLUMN

    OBJECT                     = COLUMN
        NAME                   = CONVERTER_P5V
        DATA_TYPE              = MSB_UNSIGNED_INTEGER
        START_BYTE             = 24
        BYTES                  = 1
        OFFSET                 = -1.439
        SCALING_FACTOR         = 0.02869
        UNIT                   = “VOLT”
        DESCRIPTION            = “+5V voltage measurement at DC/DC converter.”
    END_OBJECT                 = COLUMN

    OBJECT                     = COLUMN
        NAME                   = IRS_P5V
        DATA_TYPE              = MSB_UNSIGNED_INTEGER
        START_BYTE             = 25
        BYTES                  = 1
        OFFSET                 = -15.752
        SCALING_FACTOR         = 1.0295
        UNIT                   = “mAMP”
        DESCRIPTION            = “+5V current measurement of the IRS boards.”
    END_OBJECT                 = COLUMN

    OBJECT                     = COLUMN
        NAME                   = CONVERTER_N12V
        DATA_TYPE              = MSB_UNSIGNED_INTEGER
        START_BYTE             = 26
        BYTES                  = 1
        OFFSET                 = -2.0488
        SCALING_FACTOR         = 0.1339
        UNIT                   = “VOLT”
        DESCRIPTION            = “-12V voltage measurement at DC/DC converter.”
    END_OBJECT                 = COLUMN

    OBJECT                     = COLUMN
        NAME                   = LMS12_P5V
        DATA_TYPE              = MSB_UNSIGNED_INTEGER
        START_BYTE             = 27
        BYTES                  = 1
        OFFSET                 = -3.05
        SCALING_FACTOR         = 0.366
```

```
    UNIT              = "mAMP"
    DESCRIPTION       = "+5V current measurement of the latchup protected part,
                        LMS12."
END_OBJECT            = COLUMN

OBJECT                = COLUMN
    NAME              = EEPROM_P5V
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    START_BYTE        = 28
    BYTES             = 1
    OFFSET            = -3.15
    SCALING_FACTOR    = 0.37
    UNIT              = "mAMP"
    DESCRIPTION       = "+5V current measurement of the latchup protected part,
                        EEPROM."
END_OBJECT            = COLUMN

OBJECT                = COLUMN
    NAME              = TEC_TEMP
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    START_BYTE        = 29
    BYTES             = 1
    OFFSET            = 0.8019
    SCALING_FACTOR    = -0.05241
    UNIT              = "VOLT"
    DESCRIPTION       = "TE cooler temperature voltage; can be converted into
                        temperature using the Table 8 in THEMIS Command and
                        Data Format Description (SBRC document number Y2393-
                        0007)."
END_OBJECT            = COLUMN

OBJECT                = COLUMN
    NAME              = IRIS_P5V
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    START_BYTE        = 30
    BYTES             = 1
    OFFSET            = -38.67
    SCALING_FACTOR    = 2.6124
    UNIT              = "mAMP"
    DESCRIPTION       = "+5V current measurement of IRIS electronics, not latchup
                        protected."
END_OBJECT            = COLUMN

OBJECT                = COLUMN
    NAME              = TOTAL_P5V
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    START_BYTE        = 31
    BYTES             = 1
```

```
    DESCRIPTION         = "The total +5V current count for all boards."
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = TEC_P5V
    DATA_TYPE           = MSB_UNSIGNED_INTEGER
    START_BYTE          = 32
    BYTES               = 1
    OFFSET              = -19.33
    SCALING_FACTOR      = 1.263
    UNIT                = "mAMP"
    DESCRIPTION         = "+5V current measurement of TE cooler."
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = IRIS_N12V
    DATA_TYPE           = MSB_UNSIGNED_INTEGER
    START_BYTE          = 33
    BYTES               = 1
    OFFSET              = -25.14
    SCALING_FACTOR      = 2.30
    UNIT                = "mAMP"
    DESCRIPTION         = "-12V current measurement to the IRIS."
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = IRIS_P12V
    DATA_TYPE           = MSB_UNSIGNED_INTEGER
    START_BYTE          = 34
    BYTES               = 1
    OFFSET              = -64.71
    SCALING_FACTOR      = 4.23
    UNIT                = "mAMP"
    DESCRIPTION         = "+12V current measurement to the IRIS."
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = IRS_N12V
    DATA_TYPE           = MSB_UNSIGNED_INTEGER
    START_BYTE          = 35
    BYTES               = 1
    OFFSET              = -27.93
    SCALING_FACTOR      = 2.413
    UNIT                = "mAMP"
    DESCRIPTION         = "-12V current measurement to the IRS."
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = IRS_P12V
```

```
    DATA_TYPE             = MSB_UNSIGNED_INTEGER
    START_BYTE            = 36
    BYTES                 = 1
    OFFSET                = -36.25
    SCALING_FACTOR        = 2.96
    UNIT                  = "mAMP"
    DESCRIPTION           = "+12V current measurement to the IRS."
  END_OBJECT              = COLUMN

  OBJECT                  = COLUMN
    NAME                  = LATCHUP_V1
    DATA_TYPE             = MSB_UNSIGNED_INTEGER
    START_BYTE            = 37
    BYTES                 = 1
    DESCRIPTION           = "Comparator output voltage count used to determine state
                            of V1 latchup current;  compare count to boundaries: <50 =
                            FAULT, >220 = OKAY."
  END_OBJECT              = COLUMN

  OBJECT                  = COLUMN
    NAME                  = VNSTRIP
    DATA_TYPE             = MSB_UNSIGNED_INTEGER
    START_BYTE            = 38
    BYTES                 = 1
    OFFSET                = 0.38986
    SCALING_FACTOR        = -0.02548
    UNIT                  = "VOLT"
    DESCRIPTION           = "The variable negative bias voltage for the IR focal plane."
  END_OBJECT              = COLUMN

  OBJECT                  = COLUMN
    NAME                  = LATCHUP_5V
    DATA_TYPE             = MSB_UNSIGNED_INTEGER
    START_BYTE            = 39
    BYTES                 = 1
    DESCRIPTION           = "Comparator output voltage count used to determine state
                            of 5V IRIS latchup current;  compare count to boundaries:
                            <50 = FAULT, >220 = OKAY."
  END_OBJECT              = COLUMN

  OBJECT                  = COLUMN
    NAME                  = LATCHUP_V2
    DATA_TYPE             = MSB_UNSIGNED_INTEGER
    START_BYTE            = 40
    BYTES                 = 1
    DESCRIPTION           = "Comparator output voltage count used to determine state
                            of V2 latchup current;  compare count to boundaries: <50 =
                            FAULT, >220 = OKAY."
  END_OBJECT              = COLUMN
```

```
OBJECT                = COLUMN
  NAME                = SPARE41
  DATA_TYPE           = MSB_UNSIGNED_INTEGER
  START_BYTE          = 41
  BYTES               = 1
  DESCRIPTION         = "Reserved for future use"
END_OBJECT            = COLUMN

OBJECT                = COLUMN
  NAME                = TEC_SHUTDOWN_TEMP
  DATA_TYPE           = MSB_UNSIGNED_INTEGER
  START_BYTE          = 42
  BYTES               = 1
  DESCRIPTION         = "Comparator output voltage count used to determine TE
                        cooler temperature shutdown;  compare count to
                        boundaries: <50 = FAULT, >220 = OKAY."
END_OBJECT            = COLUMN

OBJECT                = COLUMN
  NAME                = DIGITAL_WATCHDOG
  DATA_TYPE           = MSB_BIT_STRING
  START_BYTE          = 43
  BYTES               = 1
  DESCRIPTION         = "Bit word flag indicating overcurrent or overtemp of the
                        named components."

  OBJECT                = BIT_COLUMN
    NAME                = SPARE43_1
    BIT_DATA_TYPE       = MSB_UNSIGNED_INTEGER
    START_BIT           = 1
    BITS                = 4
    DESCRIPTION         = "Reserved for future use"
  END_OBJECT            = BIT_COLUMN

  OBJECT                = BIT_COLUMN
    NAME                = TEC_OVERTEMP
    BIT_DATA_TYPE       = MSB_UNSIGNED_INTEGER
    START_BIT           = 5
    BITS                = 1
    DESCRIPTION         = "Status of TE cooler temperature.
                          [0] = Overtemp
                          [1] = OK"
  END_OBJECT            = BIT_COLUMN

  OBJECT                = BIT_COLUMN
    NAME                = IRIS_OVERCURRENT
    BIT_DATA_TYPE       = MSB_UNSIGNED_INTEGER
    START_BIT           = 6
```

```
        BITS                    = 1
        DESCRIPTION             = "Latchup status of IRIS protected parts current.
                                    [0] = Overcurrent
                                    [1] = OK"
    END_OBJECT                  = BIT_COLUMN

    OBJECT                      = BIT_COLUMN
        NAME                    = LMS_OVERCURRENT
        BIT_DATA_TYPE           = MSB_UNSIGNED_INTEGER
        START_BIT               = 7
        BITS                    = 1
        DESCRIPTION             = "Latchup status of LMS12 current.
                                    [0] = Overcurrent
                                    [1] = OK"
    END_OBJECT                  = BIT_COLUMN

    OBJECT                      = BIT_COLUMN
        NAME                    = EEPROM_OVERCURRENT
        BIT_DATA_TYPE           = MSB_UNSIGNED_INTEGER
        START_BIT               = 8
        BITS                    = 1
        DESCRIPTION             = "Latchup status of EEPROM current.
                                    [0] = Overcurrent
                                    [1] = OK"
    END_OBJECT                  = BIT_COLUMN
END_OBJECT                  = COLUMN

OBJECT                      = COLUMN
    NAME                    = IRIS_STATUS
    DATA_TYPE               = MSB_BIT_STRING
    START_BYTE              = 44
    BYTES                   = 1
    DESCRIPTION             = "Bit-word indicating calibration flag or latchup status from
                                IRIS electronics."

    OBJECT                      = BIT_COLUMN
        NAME                    = SPARE44_1
        BIT_DATA_TYPE           = MSB_UNSIGNED_INTEGER
        START_BIT               = 1
        BITS                    = 2
        DESCRIPTION             = "Reserved for future use"
    END_OBJECT                  = BIT_COLUMN

    OBJECT                      = BIT_COLUMN
        NAME                    = LATCHUP_TRIGGER
        BIT_DATA_TYPE           = MSB_UNSIGNED_INTEGER
        START_BIT               = 3
        BITS                    = 1
```

```
    DESCRIPTION        = "Latchup protection circuit status (from IRIS).
                          [0] = On
                          [1] = Off"
END_OBJECT             = BIT_COLUMN

OBJECT                 = BIT_COLUMN
    NAME               = LATCHUP_SENSITIVITY
    BIT_DATA_TYPE      = MSB_UNSIGNED_INTEGER
    START_BIT          = 4
    BITS               = 1
    DESCRIPTION        = "Latchup control circuit sensitivity state (from IRIS).
                          [0] = High
                          [1] = Low"
END_OBJECT             = BIT_COLUMN

OBJECT                 = BIT_COLUMN
    NAME               = CALIB_FLAG_PRI_OPEN
    BIT_DATA_TYPE      = MSB_UNSIGNED_INTEGER
    START_BIT          = 5
    BITS               = 1
    DESCRIPTION        = "Status of calibration flag primary limit switch for open
                          position (from IRIS).
                          [0] = Open
                          [1] = Not open"
END_OBJECT             = BIT_COLUMN

OBJECT                 = BIT_COLUMN
    NAME               = CALIB_FLAG_PRI_CLOSE
    BIT_DATA_TYPE      = MSB_UNSIGNED_INTEGER
    START_BIT          = 6
    BITS               = 1
    DESCRIPTION        = "Status of calibration flag primary limit switch for
                          closed position (from IRIS).
                          [0] = Closed
                          [1] = Not closed "
END_OBJECT             = BIT_COLUMN

OBJECT                 = BIT_COLUMN
    NAME               = CALIB_FLAG_RDT_OPEN
    BIT_DATA_TYPE      = MSB_UNSIGNED_INTEGER
    START_BIT          = 7
    BITS               = 1
    DESCRIPTION        = "Status of calibration flag redundant limit switch for
                          open position (from IRIS).
                          [0] = Open
                          [1] = Not open"
END_OBJECT             = BIT_COLUMN
```

```
        OBJECT                  = BIT_COLUMN
          NAME                  = CALIB_FLAG_RDT_CLOSE
          BIT_DATA_TYPE         = MSB_UNSIGNED_INTEGER
          START_BIT             = 8
          BITS                  = 1
          DESCRIPTION           = "Status of calibration flag redundant limit switch for
                                   closed position (from IRIS).
                                   [0] = Closed
                                   [1] = Not closed "
        END_OBJECT              = BIT_COLUMN
      END_OBJECT                = COLUMN

      OBJECT                    = COLUMN
        NAME                    = END_SYNC
        DATA_TYPE               = MSB_UNSIGNED_INTEGER
        START_BYTE              = 45
        BYTES                   = 2
        DESCRIPTION             = "Indicates frame synchronization at the end of each frame.
                                   [1010 1011 1000 1100]"
      END_OBJECT                = COLUMN
```

A.8 QUBE Object Items

QUBE STRUCTURE & CORE DESCRIPTION

AXES
: Number of dimensions (axes) of the QUBE. [3]

AXIS_NAME
: Names of axes in physical storage order. [(SAMPLE, LINE, BAND)]

CORE_ITEMS
: The length of each of the three axes of the core in pixels.

CORE_NAME
: Name of the value stored in core of QUBE. ["RAW_DATA_NUMBER" (for EDR) or "CALIBRATED_SPECTRAL_RADIANCE" (for RDR)]

CORE_ITEM_BYTES
: Core element size in bytes. [1 (for EDR) or 4 (for RDR)]

CORE_ITEM_TYPE
: Core element type. [MSB_UNSIGNED_INTEGER (for EDR) or SUN_REAL (for RDR)]

CORE_BASE
: Base value of core item scaling.

CORE_MULTIPLIER
: Multiplier for core item scaling.

CORE_UNIT

Unit of the value stored in the core of QUBE. [“DIMENSIONLESS” or “WATT*CM**-2*SR**-1*UM**-1”]

CORE_NULL

Value assigned to “invalid” or missing data. [0 (for EDR) or 16#FF7FFFFB# (for RDR)]

CORE_VALID_MINIMUM

Value of the minimum valid core data in an RDR QUBE. [16#FF7FFFFA#]

CORE_LOW_REPR_SATURATION

Value of representation saturation at the low end in an RDR QUBE. [16#FF7FFFFC#]

CORE_LOW_INSTR_SATURATION

Value of instrument saturation at the low end in an RDR QUBE. [16#FF7FFFFD#]

CORE_HIGH_REPR_SATURATION

Value of representation saturation at the high end in an RDR QUBE. [16#FF7FFFFF#]

CORE_HIGH_INSTR_SATURATION

Value of instrument saturation at the high end in an RDR QUBE. [16#FF7FFFFE#]

SUFFIX DESCRIPTION

SUFFIX_ITEMS

The dimensions of available suffix planes following the order given in AXIS_NAME keyword. [(1, 1, 0)]

SUFFIX_BYTES

The allocation in bytes of each suffix plane defined. [4]

*AXIS*_SUFFIX_NAME

Name of “axis” suffix plane, where “axis” can be either SAMPLE or LINE in IRRDR QUBEs. [HORIZONAL_DESTRIPE (for SAMPLE suffix planes) or VERTICAL_DESTRIPE (for LINE suffix planes)]

*AXIS*_SUFFIX_ITEM_BYTES

Size of “axis” suffix plane elements in bytes, where “axis” can be either SAMPLE or LINE in IRRDR QUBEs. [2]

*AXIS*_SUFFIX_ITEM_TYPE

“Axis” suffix plane element type, where “axis” can be either SAMPLE or LINE in IRRDR QUBEs. [MSB_INTEGER]

*AXIS*_SUFFIX_BASE

Base value of “axis” suffix plane item scaling, where “axis” can be either SAMPLE or LINE in IRRDR QUBEs.

*AXIS*_SUFFIX_MULTIPLIER

Multiplier for “axis” suffix plane item scaling, where “axis” can be either SAMPLE or LINE in IRRDR QUBEs.

*AXIS*_SUFFIX _VALID_MINIMUM

Value of the minimum valid “axis” suppix plane data, where “axis” can be either SAMPLE or LINE in IRRDR QUBEs. [-32752]

*AXIS*_SUFFIX _NULL
Value assigned to "invalid" or missing data in an "axis" suffix plane, where "axis" can be either SAMPLE or LINE in IRRDR QUBEs. [-32768]

*AXIS*_SUFFIX _LOW_REPR_SATURATION
Value of representation saturation at the low end in an "axis" suffix plane, where "axis" can be either SAMPLE or LINE in IRRDR QUBEs. [-32467]

*AXIS*_SUFFIX _LOW_INSTR_SATURATION
Value of instrument saturation at the low end in an "axis" suffix plane, where "axis" can be either SAMPLE or LINE in IRRDR QUBEs. [-32766]

*AXIS*_SUFFIX _HIGH_REPR_SATURATION
Value of representation saturation at the high end in an "axis" suffix plane, where "axis" can be either SAMPLE or LINE in IRRDR QUBEs. [-32764]

*AXIS*_SUFFIX _HIGH_INSTR_SATURATION
Value of instrument saturation at the high end in an "axis" suffix plane, where "axis" can be either SAMPLE or LINE in IRRDR QUBEs. [-32465]

OBSERVATION PARAMETERS

FLIGHT_SOFTWARE_VERSION_ID
Indicates version of instrument flight software used to acquire image. ["1.00"]

COMMAND_SEQUENCE_NUMBER
Numeric identifier for the sequence of commands sent to the spacecraft which include this image.

IMAGE_ID
Numeric identifier for this image within the onboard command sequence.

DESCRIPTION
Description of image written by mission planner.

INST_CMPRS_RATIO
The ratio of the size, in bytes, of the uncompressed data file to the compressed data file.

UNCORRECTED_START_TIME
The spacecraft clock value (in seconds) when the instrument was commanded to acquire an observation. This can differ from the SPACECRAFT_CLOCK_START_COUNT (or the other START_TIME keywords) by as much as 4 seconds, depending on which bands are acquired in the image.

INST_CMPRS_NAME
The type of compression applied to the VIS data and removed before storage in the image QUBE. ["NONE" or "DCT" or "PREDICTIVE"]

FOCAL_PLANE TEMPERATURE
The temperature in Kelvin of the VIS camera focal plane array at the time of the observation.

EXPOSURE_DURATION
The exposure time of a VIS image; given in milliseconds.

INTERFRAME_DELAY
The time between successive frames of a VIS image; given in seconds.

SPATIAL_SUMMING
Onboard spatial average of NxN set of pixels, where N is the value of the keyword. SPATIAL_SUMMING = 1 implies that no spatial averaging has been applied to the VIS image. [1 or 2 or 4]

GAIN_NUMBER
The gain value of the THEMIS IR camera; a multiplicative factor used in the analog to digital conversion.

OFFSET_NUMBER
The offset value of the THEMIS IR camera; the offset value multiplied by a constant voltage is added to the measured voltage in the analog to digital conversion.

TIME_DELAY_INTEGRATION_FLAG
Status of onboard algorithm which applies a temporal average of successive lines in an IR image; when enabled, THEMIS TDI averages 16 detector rows to equal one line in an IR image. [“ENABLED” or “DISABLED”]

MISSING_SCAN_LINES
The total number of scan lines missing from an IR image when it was received at Earth.

MD5_CHECKSUM
A 128-bit checksum identification of the data portion of the QUBE. Corruption of the data QUBE will result in a different value when the MD5 algorithm is reapplied as compared to the value stored in the keyword. An example of the source code applied by ASU is available in SRC/BIN/md5_qube.pl. A complete definition of the MD5 algorithm is available at http://www.ietf.org/rfc/rfc1321.txt. [“fd2781d05bdc0215dc87a0f41035ad77”]

BAND-BINS

BAND_BIN_FILTER_NUMBER
List of filter numbers corresponding to each layer (band) contained in the image; up to 10 entries possible for IR images and up to 5 entries possible for VIS images. The filter number describes the physical location of the band in the detector array; filter 1 is on the leading edge of the detector array.

BAND_BIN _BAND_NUMBER
List of band numbers corresponding to each layer (band) contained in the image; up to 10 entries possible for IR images and up to 5 entries possible for VIS images. The band number is equivalent to the instrument band number listed in Table 1, Section 2.2 of this document (THM-SDPSIS).

BAND_BIN_CENTER
List of wavelength values corresponding to each layer (band) contained in the image; up to 10 entries possible for IR images and up to 5 entries possible for VIS images.

BAND_BIN_WIDTH

Calculated full width, half maximum (in micrometers) for each band listed in the BAND_BIN_ BAND_NUMBER.

BAND_BIN_UNIT

Unit which applies to the values of the BAND_BIN_CENTER keyword. ["MICROMETER"]