Cassini
Composite Infrared Spectrometer
(CIRS)

Planetary Data System (PDS)
Time Sequential Data Record (TSDR)
Data Product
Software Interface Specification (SIS)

July 6, 2005

IO-AR-002

Prepared by:

____

Conor A. Nixon, University of Maryland, CIRS Team

Approved by:

____

F. Michael Flasar, CIRS Principal Investigator

Concurred by:

____

Diane Conner, Cassini Archive Data Engineer

____

Lyle Huber, PDS Atmospheres Node Data Engineer

Contents

List of Figures

List of Tables

1 Introduction

This document describes the format and content of the Cassini Composite Infrared Spectrometer (CIRS) Time Sequential Data Record (TSDR) products, as archived to the Planetary Data System (PDS). The aim of this document is to provide a user guide for persons accessing and using the archived data.

Acknowledgment

The heritage of the CIRS ground data system draws heavily on the experience and tools developed by the Mars Global Surveyor Thermal Emission Spectrometer (MGS TES) team. Their database utility 'Vanilla' is used as a core access tool by the CIRS team. In addition, the authors have drawn on the TES-TSDR Standard Data Product SIS as a template and starting point for this document.

2 Data Overview

2.1 CIRS Instrument

CIRS is a Fourier Transform Spectrometer, and a dual interferometer by design (see Kunde et al 1996, Flasar et al 2004). In the mid-infrared, a conventional Michelson mirror arrangement is terminated by dual 10-element detector arrays. The 10 photoconductive (PC) detectors of focal plane 3 (FP3) have a bandpass of 600–1100 cm−1 approximately, and the 10 photovoltaic detectors of focal plane 4 (FP4) are sensitive from 1000–1500 cm−1. Each of the 20 MIR detectors has a square field of view (FOV) which is 0.273 milliradian across.

In the far-infrared, a Martin-Puplett type (Martin and Puplett 1969) polarizing interferometer conveys the radiation onto two redundant thermopile detectors, one sensitive to the transmitted and one to the reflected beam at the final polarizer-analyzer. Each detector has a circular FOV of full width to half maximum (FWHM) 2.4 milliradians. Each covers the spectral range 10–600 cm−1, and are known as FP1.[1]

CIRS records interferograms (IFMs) at varying resolution: from 0.5 cm−1 to 15.5 cm−1. The resolution is determined by the scan distance (time), normally commanded in units of 1/8 second (=1 RTI, or 'Real Time Interrupt'). The valid range of RTIs is 36 to 416. The resolution is approximately inversely proportional to the scan time, although in fact the relationship is non-linear, especially at short scan times (see Figure 1). This is due to (i) mirror flyback time, which is included in the commanded scan time, and (ii) mirror acceleration and deceleration to constant-velocity scanning.

---

[1] In the original design for CIRS, a second FIR detector (FP2) was planned; but this was later removed during a de-scope exercise.

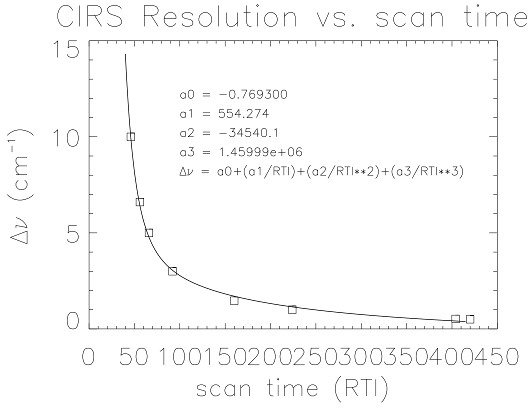

Figure 1: CIRS resolution vs scan time.

2.2 CIRS MIR Observing Modes

In the mid-IR, CIRS has only five solid-state 'channels' for post-detection signal processing, for each of FP3 and FP4. These arrays each have 10 detectors, so the solution is that exactly five detectors can be activated at a time from each array.

There are four pre-programmed observing modes (see figure 2):

ODD Detectors 1, 3, 5, 7, 9 active on each array.

EVEN Detectors 2, 4, 6, 8, 10 active on each array.

CENTER Detectors 4–8 (FP3) and 3–7 (FP4) active on each array.

PAIRS The signal from the detectors is combined pairwise before signal processing; so 1&2 are paired etc, through 9&10.

The result is that CIRS normally writes 11 IFMs at a time to the Bus Interface Unit (BIU): 1 from FP1, 5 from each of FP3 and FP4. However, there are reduced data modes where only 6 (FP1 plus either FP3 or FP4) or 1 (FP1) channels are written. Note that, in addition, CIRS may be commanded to co-add IFMs pairwise in time, with the result that data are only written on alternate scans. This achieves a 50% reduction in data volume if and when required, at the cost of possible 'smear' of spatial field if the spacecraft is scanning.

2.3 Standards Used in Generating Data Products

2.3.1 Time Standards

The primary key field, used to cross-link CIRS data in different files together, is *spacecraft event time* (SCET) stored as an integer number of seconds since 1/1/1970 00:00:00 UT,

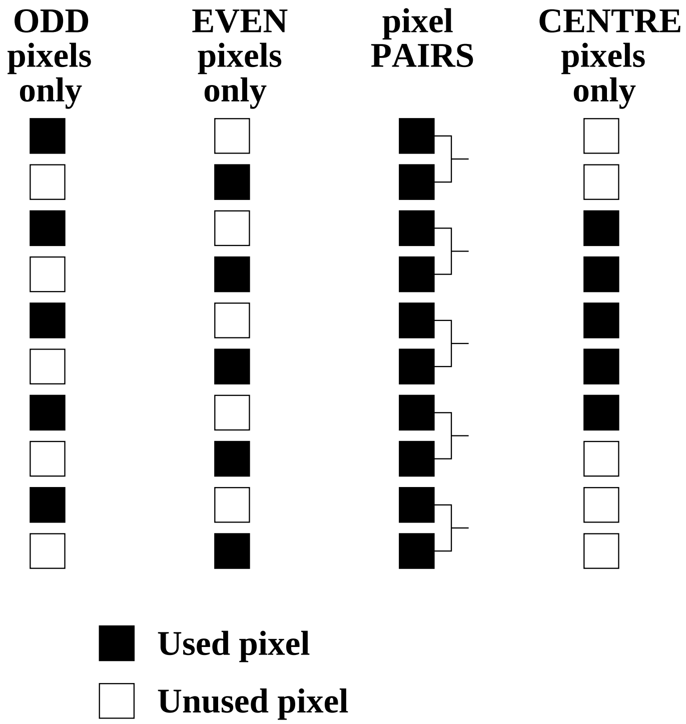

Figure 2: CIRS pixel switching modes.

ignoring leap seconds. Also stored as retrievable field, but not a key field, is *spacecraft clock* (SCLK), and the SCLK partition number. These are stored as two integers.

SCLK is essentially an integer number related to oscillations of the on-board quartz crystal clock. Partition number is advanced by one when the SCLK count is reset, expected to be a rare event, if ever. SCLK ticks are not even linearly related to universal time, and the conversion between the two requires a polynomial to account for drifts in the SCLK oscillation frequency. SCET on the other hand *is* linearly related to universal time.

SCET is used as the primary time key field rather than SCLK for two reasons: firstly, SCLK may reset, and not be monotonically increasing, in which case the partition number also changes; and secondly, SCET may easily and conveniently be converted to text time using standard computer library functions, whereas interpreting SCLK requires the NAIF toolkit and Cassini time kernel (SCLKSCET) files.

At the time of writing, standard computer library routines are available which facilitate the conversion to and from text time to SCET number, such as the `C` call to `gmtime`.

2.3.2 Co-ordinate Systems

In general, our conventions will follow the Cassini Archive Plan for Science Data: refer to that document for further details.

Jupiter: System III west longitude and planetographic latitude.

Titan: Titan longitude is the standard Longitude of Central Meridian (West Longitude) with the origin in the Saturn-facing direction of synchronous rotation. Latitude is relative to a spherical body at the time of writing: hence the planetographic/planetocentric dichotomy does not occur.

Saturn: System III west longitude and planetographic latitude.

Rings: see appendices.

Icy Satellites: standard Cassini definitions will be applied (see the APSD document).

2.3.3 Orbit Numbers (revs)

We use Cassini orbit numbers, as determined by the Cassini project.

2.3.4 Data Storage Conventions

CIRS data is produced on PC-type machines using, at the time of writing, 32 bit Intel CPU's of either the Pentium or Xeon class. These chips use the LSB (least significant byte) storage convention. More details can be found in Appendix C of the PDS Standards Reference document.

2.4 Interferences and Noises

CIRS interferogram data suffers from a number of external interferences, especially:

- A 8 Hz spike pattern due to the spacecraft clock ticks.
- A 1/2 Hz spike pattern due to the Bus Interface Unit, transfer of data.
- A sine wave of variable frequency which appears correlated with the electronics board temperature.
- Scan speed fluctuations which have been traced to two mechanical vibrations on the spacecraft: (a) the MIMI LEMMS actuator (b) the reaction wheels used to turn the spacecraft.

These various effects are described in more detail in the cirs_interferences.pdf document found in the DOC directory.

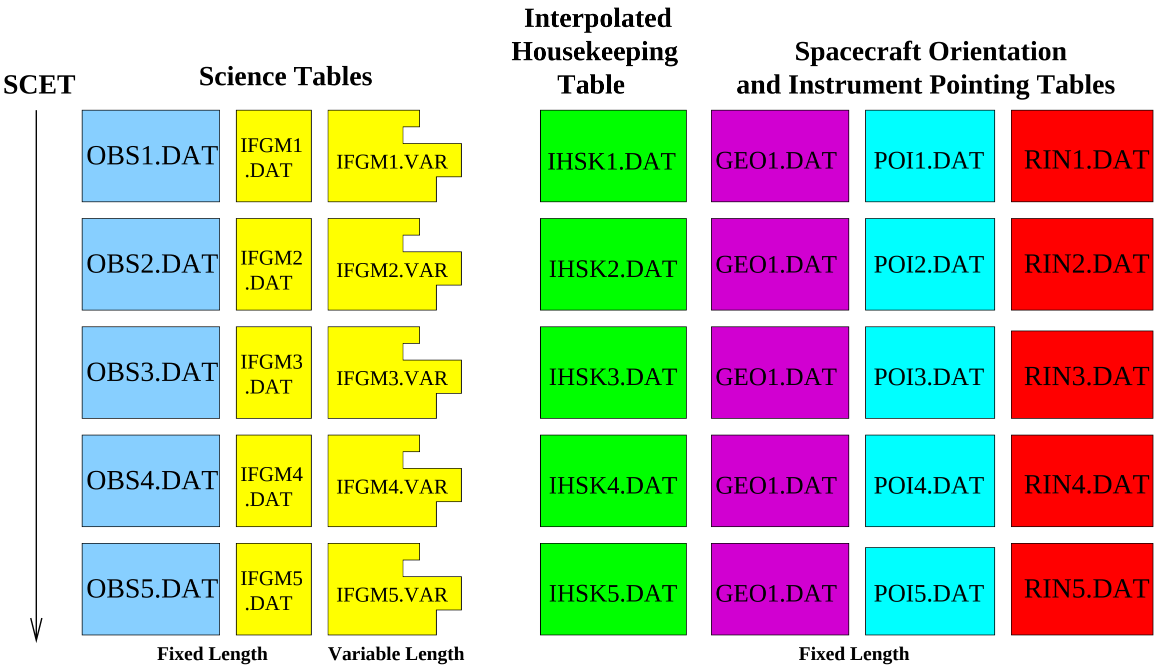

Figure 3: CIRS data file fragments.

3 Data Detailed Description

3.1 Logical Table and Physical Organisation

The dataset is conceptually one huge table, with records (‘rows’) time-ordered, and containing many fields (‘columns’). In practice, this large ‘virtual’ (or ‘logical’) table is stored in multiple files, known as fragments. This ‘physical’ table is fragmented both in the ‘row’ and ‘column’ sense.

Table fragments have name stems of the form NNNYYMMDDHH, where ‘NNN’ is a fragment type descriptor, such as ‘OBS’ indicating observation parameters, and ‘IFGM’ indicating interferogram data. The remainder of the stem is a time period: year (YY), month (MM), day (DD) and hour (HH) corresponding to the start of the data period covered. CIRS data typically is divided into 4-hour blocks at the uncalibrated level and 24-hour blocks at the calibrated level, although there may be exceptions. Normally this division is transparent to the user in any case, because Vanilla treats all separate files as forming part of a single, huge logical search table.

Each fragment stem has a file extension ‘.DAT’ if it contains only fixed-length fields. If it contains both fixed length and variable length fields, then fragments with both ‘.DAT’ and ‘.VAR’ will be written for that stem name. The ‘.DAT’ portion of a variable length fragment specifies the location of the variable length data in the ‘.VAR’ file.

See figure 3 for schematic.

Fixed length files are designed to contain data fields of unvarying length. These items include fixed byte-size scalar values such as detector number, scet time, number of points in the IFM etc, and also fixed-size vectors such as the distance to the target surface for each of the nine Q-points in each detector.

The variable length files contain vectors types which may vary in length. The archetypal examples of variable-length vector fields are the interferograms and spectra. The number of points in an FTS spectrum depends is proportional to the spectral resolution (approximately): the higher the spectral resolution, the longer the interferogram (i.e. greater path difference sampled), and the more spectral points in the final spectrum. I.e. the fixed bandpass is being covered by more and more points closer and closer together.

CIRS has a variable spectral resolution from 0.5 cm-1 to 15.5 cm-1 (apodized), depending on commanded scan length, and hence the need for variable-length vector fields. If these were stored as fixed-length vectors padded with zeroes, it would be extremely inefficient storage.

3.2 Fields and Key Fields

As mentioned above, each fragment type stores different fields of the table. For example, while the 'OBS' (observation parameters) fragment may store the MIR shutter open/closed status, the 'HSK' (housekeeping) fragment may store the temperature of the primary mirror. All fragments generally store unique information, not contained in other fragments. The exceptions to this are the so-called 'key fields', which provide the row indexing.

The primary key field is **scet** - the spacecraft event time, stored as a number of seconds since the start of 1970. However, as CIRS simultaneously records 11 IFMs, the time alone is not enough to uniquely label an IFM. Hence, a unique detector id is used, as listed in table 1. Note that the same physical detector is treated separately depending on whether it is being used in ODD/EVEN or CENTER mode. This is due to the fact that the signal will be processed by a different receiver channel in the electronics in the two cases, and so they must be calibrated separately.

As well as spectral information, the table stores pointing information for the detectors at the time of observation. This information is generated on the ground from the reconstructed c-kernels provided by JPL, in association with the instrument frames kernel.

However, a further degeneracy may be introduced into the data, if multiple targets appear in a single FOV. We must find a way to specify multiple pointings for say, a moon, which appears superimposed on the parent planet. Hence, the third key field is the target id.[2]

Thus a maximum of 3 key fields uniquely determine a row in the table. A given data fragment will store the key fields sufficient to uniquely identify the data within. E.g, the OBS fragment stores SCET only, the IFGM fragment stores SCET and DET, while the POI stores SCET, DET and TARGET_ID.

---

[2] The target id field is called 'body id' when it occurs in the GEO tables, to distinguish objects which may or may not be in the FOV ('bodies') from objects which are definitely in the FOV ('targets').

Focal Plane	Observing Mode	Detector Number	Database ID Code
1	ALL	1	0
3	PAIR	1 & 2	1
3	PAIR	3 & 4	2
3	PAIR	5 & 6	3
3	PAIR	7 & 8	4
3	PAIR	9 & 10	5
3	CENTER	4	6
3	CENTER	5	7
3	CENTER	6	8
3	CENTER	7	9
3	CENTER	8	10
3	ODD	1	11
3	EVEN	2	12
3	ODD	3	13
3	EVEN	4	14
3	ODD	5	15
3	EVEN	6	16
3	ODD	7	17
3	EVEN	8	18
3	ODD	9	19
3	EVEN	10	20
4	ODD	1	21
4	EVEN	2	22
4	ODD	3	23
4	EVEN	4	24
4	ODD	5	25
4	EVEN	6	26
3	ODD	7	27
4	EVEN	8	28
4	ODD	9	29
4	EVEN	10	30
4	CENTER	3	31
4	CENTER	4	32
4	CENTER	5	33
4	CENTER	6	34
4	CENTER	7	35
4	PAIR	1 & 2	36
4	PAIR	3 & 4	37
4	PAIR	5 & 6	38
4	PAIR	7 & 8	39
4	PAIR	9 & 10	40

Table 1: Detector ID specifications in the database.

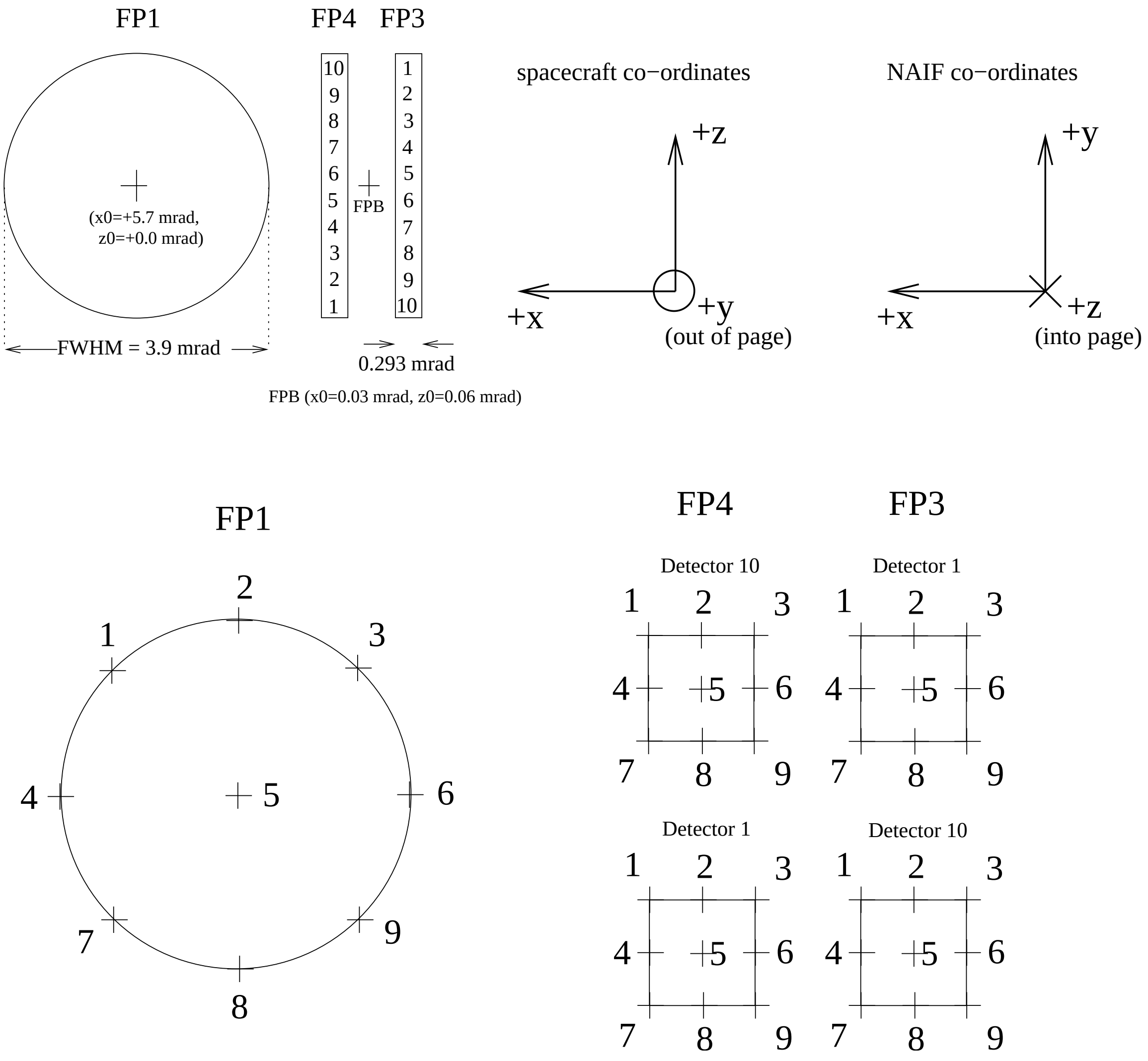

Figure 4: Q-points for CIRS detectors.

Table	Key fields			
Type	SCET	DET	TARGET_ID	BODY_ID
AIFM	X	X		
GEO	X			X
IFGM/FIFM	X	X		
IHSK/HSK	X			
OBS	X			
POI	X	X	X	
RIN	X	X		
SPM/ISPM	X	X		
TAR	X	X		

Table 2: Key fields for different file types.

The field BODY_ID, as used in the GEO tables, is similar to the TARGET_ID used in the POI tables. The main difference is that GEO table contains positional information for a subset of large bodies (primary planet, larger moons) with respect to the spacecraft, *whether or not* they are in the instrument FOV. The POI tables on the other hand store pointing information for all bodies which enter one or more of the FOVs. Hence, the list of bodies is different in the two cases, and a different keyword must be used to separate between the two.

3.3 Fixed-Length Data File Format

When we say 'fixed-length' file format in this document, we refer to the record, not the file. I.e., we mean that the record may consist of say 10, 4-byte numbers, for a total fixed width of 40 bytes. No field varies in width. However, the length of the file (number of records) can and will vary between instances of the file type. One may have 100 fixed-length records, another 300.

Fixed-length data record files (.DAT type) store much of the CIRS pointing, geometry, housekeeping and some science data fields. The other main type of data files are the variable-length data files (.VAR type), which act as extensions to the fixed-length data files and are described below. Both file types are binary, written in PC encoding (LSB).

Each record in a fixed-length record file consists of a number of fields. The field widths are fixed, and every field is written for every record. The field layout of each type of fixed-format data files is give in a format file (.FMT), which are re-produced in Appendix B. Format files also form an integral part of the Vanilla database, and hence there is always a copy of the requisite format files in each data subdirectory.

Important Note: fields are written to the file with the exact byte-lengths described in the format files. Depending on which system, language and compiler is used to read/write the binary data, this will usually mean that each field must be read/written by a single command statement, rather than read/writing the entire record at a time, using a data 'structure'. Structures are often padded, so that a 1-byte number may be read/written as 4-bytes for example, if it occurs next to 4-byte integers in a structure. The code used to write CIRS data is all written in the (ANSI) C-language, and the subroutines are all included in the SOFTWARE/SRC directory of each volume.

3.4 PDS Label Files

Each fixed-format binary data file (.DAT type) is accompanied by an ASCII label file (.LBL type) containing key fields written in PDS3 Object Description Language (ODL) format. An example is:

```
PDS_VERSION_ID               = PDS3
INSTRUMENT_HOST_NAME         = "CASSINI ORBITER"
INSTRUMENT_NAME              = "COMPOSITE INFRARED SPECTROMETER"
MISSION_PHASE_NAME           = "JUPITER CRUISE"
DATA_SET_ID                  = "CO-J-CIRS-2/3/4-TSDR-V1.0"
PRODUCT_ID                   = "CIRS-ISPM01013000"
STANDARD_DATA_PRODUCT_ID     = "CIRS-ISPM"
NOTE                         = "Created by Conor A Nixon.
      MD5 checksum of table  = 0fe68fbba98e8869724a7f58e60232f2
      NAIF toolkit version   = N0054"
PRODUCT_CREATION_TIME        = 2003-10-29T22:21:58
SPACECRAFT_CLOCK_START_COUNT = "1/1359504733.204"
SPACECRAFT_CLOCK_STOP_COUNT  = "1/1359590206.099"
START_TIME                   = 2001-01-30T00:00:18
STOP_TIME                    = 2001-01-30T23:44:50
SPICE_PRODUCT_ID             = { "cas_cirs_v05.ti", "cas_v28_mod.tf",
                                 "naif0007.tls", "pck00007.tpc",
                                 "010420R_SCPSE_EP1_JP83.bsp", "SCLK.ker",
                                 "010129_010201rb.bc" }
TARGET_NAME                  = { JUPITER }
OBJECT                       = FILE
    ^TABLE                   = "ISPM01013000.DAT"
    FILE_NAME                = "ISPM01013000.DAT"
    RECORD_TYPE              = FIXED_LENGTH
    RECORD_BYTES             = 45
    FILE_RECORDS             = 3478
    OBJECT                   = TABLE
        NAME                 = ISPM
        INTERCHANGE_FORMAT   = BINARY
        ^STRUCTURE           = "ISPM.FMT"
        START_PRIMARY_KEY    = (980812818)
        STOP_PRIMARY_KEY     = (980898290)
        PRIMARY_KEY          = ( "SCET", "DET" )
        ROWS                 = 3478
    END_OBJECT               = TABLE
END_OBJECT                   = FILE
```

```
OBJECT                        = FILE
    FILE_NAME                 =    "ISPM01013000.VAR"
    RECORD_TYPE               = UNDEFINED
    DESCRIPTION               =  "This file contains variable length spectra
                                   which can be read using the Vanilla software
                                   package. See the Vanilla software
                                   documentation for details."
END_OBJECT                    = FILE
END
```

These fields are interpreted as follows:

PDS_VERSION_ID, RECORD_TYPE	standard items.
INSTRUMENT_HOST_NAME, INSTRUMENT_NAME	spacecraft and instrument name.
MISSION_PHASE_NAME	mission phase.
DATA_SET_ID, PRODUCT_ID,	data set id; product id.
STANDARD_DATA_PRODUCT_ID	standard data product id.
NOTE	data creating person; MD5 checksum; NAIF version
PRODUCT_CREATION_TIME	data creation time.
SPACECRAFT_CLOCK_START_COUNT, SPACECRAFT_CLOCK_STOP_COUNT	values of the spacecraft clock for the first and last records in this file.
START_TIME, STOP_TIME	as previous, but in UTC time rather than seconds since epoch.
SPICE_PRODUCT_ID	IDs of SPICE kernels used in pointing.
TARGET_NAME	target name.
Lines 21-36	description of table object (.DAT file contents).
T̂ABLE	pointer to start record of table object.
FILE_NAME	name of this file, in the format: [type]YYMMDDHH where YYMMDD is the year-month-day of the science data, and HH is the hour of the data start.
RECORD_TYPE	file is fixed-length records.
RECORD_BYTES	number of bytes per record.
FILE_RECORDS	number of data records in file.
ŜTRUCTURE	file name of table format description.
PRIMARY_KEY	names of key fields, primary key first.
Lines 37-44	description of variable-length record file.

3.5 Variable-length Data File Format

When a particular table fragment type contains variable-length data, then only a pointer is stored in the '.DAT' file. The actual variable-length data is contained in a file having the same name, except that the extension is '.VAR'. E.g. a table fragment 'IFGM0001.DAT' would be accompanied by 'IFGM0001.VAR'.

In the variable-length data file, each record starts at the byte position indicated for that record in the '.DAT' file. Preceding, and following the actual data is a 2-byte integer giving the length in bytes. E.g. if the first record in the file was a spectrum of 32 4-byte floats, the first two bytes in the file would be the short integer '32', followed by the 32 4-byte floats, followed by the 2-byte number '32' again.

In the fixed-length file, the first pointer would be 1, the second pointer would be 133 = (32 * 4) + 2+ 2 + 1, pointing to the next spectrum, and so on. Note that although the fixed-format '.DAT' files can exist without a '.VAR' accompanying, the reverse is not true: a '.VAR' file must always have a '.DAT' counterpart, to store the byte offset positions of the variable-length record data.

3.6 Table Types

The following sections describe the different types of CIRS data tables. Field names given are the exact field names which are used when interrogating the database using `Vanilla`. Field names followed by '[ ]' are variable width fields, i.e. an array, whose number of elements may vary from record to record. Normally, the number of elements in the array for each record is given in the preceding field. Some fields are fixed-width arrays, e.g. 'LATITUDE_ZPD[9]', an array of nine elements. Please see the Vanilla Users Guide document for more information on querying arrays.

3.6.1 OBS tables

These tables store information relating to the instrument data-taking settings for each scan, and also the values of warning or indicator flags. Fields:

`SCET`

Spacecraft Event Time, in UT seconds from epoch (01/01/1970). This field is the primary key for all table fragments, and is a required field in all table types. Note that there is a utility `cirs_time` for converting text time to SCET.

`SCLK`

Spacecraft clock ticks, approximately from launch. Not guaranteed to exact one-second intervals. SCLK is not used a primary key field because unlike SCET, it cannot be simply converted to UT without the NAIF library routines, and the SCLKSCET kernels.

`RTI` The length of the scan, in real time interrupts (1/8 second).

`FP3_MODE` Detectors enabled on FP3. Four possible values: (O)dd, (E)ven, (C)enter or (P)airs.

`FP4_MODE` Detectors enabled on FP4. Values as for FP3_MODE.

`FIR_OVERFLOW` Overflow detected on FIR accumulation.

`FP1_OVERFLOW` Overflow on FP1 average buffer.

`FP3_OVERFLOW` Overflow on FP3 average buffer.

`FP4_OVERFLOW` Overflow on FP4 average buffer.

`FP1_COMP_OVERFLOW` Overflow on compression buffer 1.

`FP3_COMP_OVERFLOW` Overflow on compression buffer 3.

`FP4_COMP_OVERFLOW` Overflow on compression buffer 4.

`FP3_ZERO` Detected 0000H on FP3 raw data.

`FP3_OFFF` Detected 0FFFH on FP3 raw data.

`FP1_ZERO` Detected 0000H on FP1 raw data.

`FP1_OFFF` Detected 0FFFH on FP1 raw data.

`FP4_ZERO` Detected 0000H on FP4 raw data.

`FP4_OFFF` Detected 0FFFH on FP4 raw data.

`COADD` Coadding enabled.

`SCIENCE_OVERWRITTEN` Science data overwritten condition detected.

`FLYBACK_EXCEEDED` Flyback exceeded.

`SCAN_EXCEEDED` Scan exceeded.

`MECHANISM_ENABLED` Mechanism enabled.

`MECH_OUT_OF_PHASE` Mechanism out of phase some time.

`LASER_A_ENABLED` Laser A enabled.

`LASER_B_ENABLED` Laser B enabled.

`MECH_CMDED` Enable mechanism cmded.

`WHITE_OVERRIDE` White light override.

`OPTICAL_SENSE_MODE` Optical sensor (0), decs (1) mode.

`SHUTTER` Shutter open (0), closed (1).

`RIE_LASCMD_LSB` RIE: LASCMD_LSB Lamp current select.

`RIE_LASCMD_MSB` RIE: LASCMD_MSB Lamp current select.

`RIE_LASCMD_A` RIE: LASCMD_A Lamp A select.

`RIE_LASCMD_B` RIE: LASCMD_B Lamp B select.

`RAW_NO_SET` Count of raw samples without set control bits.

`RAW_FP1_COUNT` Number of FP1 samples read in this scan.

`RAW_FP3_COUNT` Number of FP3 samples read in this scan.

`RAW_FP4_COUNT` Number of FP4 samples read in this scan.

`FIRST_SAMPLE_RTI` RTI value when 1st sample in scan collected.

3.6.2 FRV tables

These tables store the voltages of the central fringe of the CIRS white-light interferometer, sampled at 1 second intervals after mirror flyback. See Kunde *et. al.* (1996) for details of this device.

`SCET`

The Spacecraft Event Time of start of scan (seconds from epoch).

`NFRV`

The number of data points in the fringe voltage record.

`FRV`

A pointer to the location of the data in the variable length record file.

3.6.3 IFGM tables

These tables store the actual IFM records. Note that there will be eleven IFM records (one per signal channel) for each OBS record.

Each record consists of a fixed-length part and a variable-length part corresponding to the actual IFM, which is stored in a separate file. The header stores the detector number and a pointer to the position of the variable-length data record in the VAR file.

Fields:

`SCET`

The Spacecraft Event Time of start of scan (seconds from epoch 1/1/1970 neglecting leap seconds).

`DET`

The number of the detector which recorded the IFM (see table 1).

`NPTS`

The number of data points in the interferogram.

`IFGM`

A pointer to the location of the IFM in the variable length record file.

3.6.4 HSK tables

These tables store the data from the housekeeping packets, including temperature information which is required for the calibration of IFMs. Note that the housekeeping packets are created every 64 seconds, and records do not have a direct one-to-one or one-to-many correspondence to science scans, as all other table records do. A full description of these data is in the CIRS flight software documentation.

Fields:

`SCET` Spacecraft event time of this record.

`SMERIESTAT` SME/RIE status echo word.

`FP3LASTCMD` FP3 last command.

`FP4LASTCMD` FP4 last command.

`TCMLASTCMD` TCM last command.

`FP1MAX` FP1 maximum raw sample.

`FP3MAX` FP3 maximum raw sample.

`FP4MAX` FP4 maximum raw sample.

`FRINGEMAX` Fringe maximum voltage.

`FRINGEMIN` Fringe minimum voltage.

`PENDINGTTAG` Number of pending time tagged commands.

`CODECHK` Code checksum.

`REJECTOP` Last command rejected opcode.

`PCA_ATEM` PCA ‘A’ board temperature.

`PCA_BTEM` PCA ‘B’ board temperature.

`PCA_CTEM` PCA ‘C’ board temperature.

`SME_ATEM` SME ‘A’ board temperature.

`SME_BTEM` SME ‘B’ board temperature.

`RIE_ATEM` RIE ‘A’ board temperature.

`FEE_ATEM` FEE ‘A’ board temperature.

`FEE_BTEM` FEE ‘B’ board temperature.

`FEE_CTEM` FEE ‘C’ board temperature.

`FEE_DTEM` FEE ‘D’ board temperature.

`TCM_ATEM` TCM ‘A’ board temperature.

`TCM_BTEM` TCM ‘B’ board temperature.

`BASEPLATETEM` Base plate temperature.

`PROCESSORTEM` Processor board temperature.

`BIUTEM` Bus Interface Unit (BIU) board temperature.

`ADCTEM` Analog to digital converter (ADC) board temperature.

`MOTORCURRENT` Scan motor current.

`DECSPOSN` DECS postion.

`VELERR` Velocity percentage error.

`PHASEERR` Phase error.

`FRINGEV` Fringe voltage.

`RIESMEPOS12V` RIE/SME positive 12V.

`RIESMENEG12V` RIE/SME negative 12V.

`RIESMEPOS5V` RIE/SME positive 5V.

`LASERCURRENT` Laser current.

`LEDCURRENT` LED current.

`IDSPOS15V` IDS positive 15 volts.

`IDSNEG15V` IDS negative 15 volts

`IDSPOS5V` IDS positive 5 volts.

`PHOTDIODE` Photodiode monitor.

`HTR80KCURR` 80K heater current.

`HTROP_ACURR` Optics assembly heater current.

`HTRP_MRCURR` Primary mirror heater current.

`HTRS_MRCURR` Secondary mirror heater current.

`HSECOOLTEM` Housing temperature at cooler interface near Ge lens.

`FIRPOLRIZTEM` Far-IR (FIR) inout polarizer temperature.

`HSEABVMECHTEM` Housing temperature on radiator above scan mechanism.

`FP1DETTEM` FP1 detector temperature.

`MECHHSETEM` Scan mechanism housing temperature.

`SECMIRRTEM` Secondary mirror back side temperature.

`PRIMIRRTEM` Primary mirror back side edge temperature.

`SECMIRBAFFTEM` Secondary mirror baffle back side temperature.

`PMSUNSHDMLITEM` Primary mirror MLI side sun-shade temperature.

`PMSUNSHDRADTEM` Primary mirror radiator side sun-shade temperature.

`CONERIMTEM` Cone rim temperature.

`FPATEM` Focal Plane Assembly (FPA) temperature.

`THERMISTORDRV` Thermistor drive.

`IDSCALIB` IDS Calibration.

`IDSNEG5V` IDE negative 5 volts.

3.6.5 IHSK tables

These tables store interpolated housekeeping data. Selected quantities from the HSK housekeeping tables are interpolated to the times of the actual scans, for use in calibration. Fields:

`SCET` Spacecraft Event Time (UT) seconds from 1/1/70.

`HSECOOLTEM` Scan mechanism housing cooler temperature.

`FIRPOLRIZTEM` Temperature of the FIR polarizer.

`ZINSTRADTEM` Z-axis instrument radiator temperature.

`FP1DETTEM` Temperature of the FP1 detector.

`MECHHSETEM` Scan mechanism housing temperature.

`SECMIRRTEM` Temperature of the secondary mirror.

`PRIMIRRTEM` Temperature of the primary mirror.

`SECMIRBAFFTEM` Temperature of the secondary mirror baffle.

`PMSUNSHDMLITEM` Temperature of the primary mirror sunshade MLI.

`PMSUNSHDRADTEM` Tempature of the primary mirror sunshade radiator.

`FPATEM` Temperature of the MIR focal plane assembly.

3.6.6 DIAG tables

These tables contain diagnostic information determined regarding the raw interferograms, including the presence or absence of noise interferences.

`SCET`

The Spacecraft Event Time of start of scan (seconds from epoch 1/1/1970 neglecting leap seconds).

`DET`

The number of the detector which recorded the IFM (see table 1).

`BIURTI`

RTI offset of BIU interference.

`noise`

Noise status identification. Bits are set as follows:

2^0	Normal
2^1	DC Level
2^2	Drift
2^3	(Big) Spikes
2^4	Std Dev
2^5	Long
2^6	Short
2^7	Insuf.Samples
2^8	Avg Ifm
2^9	Invalid Time Stamp
$2^1$0	Shutter
$2^1$1	ZPD Position

`DSCAL`

Set to 1 for DSCAL and 0 for no DSCAL. This identifies whether a given IFM occurs inside a DSCAL request, or not. DSCAL requests are dedicated observations of deep space for calibration purposes. Note however that space IFMs may occur in other request types.

`LASER`

Arbitrary laser mode number:

1 [4.95 - 5.25V]
2 [5.55 - 5.85V]
3 [5.95 - 6.25V]
0 the rest RIE[2]

3.6.7 GEO tables

These tables store the 'geometry', the relative positions and orientations of the spacecraft, primary body, and object bodies (which may include the primary, e.g. Saturn), usually at the time of the zero path difference (ZPD) of the interferogram (when most of the signal is collected). One entry is generated for every body in the target list.

Fields:

`SCET`

The SCET of the beginning of a scan is stored as an integer containing "Unix time", the whole number of seconds past January 1, 1970 UT, *neglecting leap seconds*. This means that for every leap second added (subtracted) the "zero of time" actually shifts forward (backward) with respect to UT. This is ordinarily not a problem, but has to be taken into account when converting SCET to ephemeris time.

`SCET_FRACTIONAL_SECONDS`

The fractional part of the SCET.

`SCLK`

The SCLK.

`PARTITION`

The partition number of the SCLK. See the NAIF toolkit documentation for a description of SCLK partitions.

`TIME_ZPD`

The time of the ZPD, which is a few seconds after `scet`. This is also a "Unix time".

`BODY_ID`

The NAIF id of the body, for example, 699 (Saturn).

`EPHEMERIS_TIME`

The ephemeris time used to obtain the following entries from the SP kernels. This ephemeris time is computed from `TIME_ZPD`.

`BODY_SPACECRAFT_RANGE`

The distance from the center of the body to the spacecraft, in km.

`BODY_POSITION`

The Cartesian coordinates of the body as seen from the spacecraft, in km, in the J2000 coordinate system. This is computed at the `EPHEMERIS_TIME`.

`BODY_SUB_SPACECRAFT_LONGITUDE`

The planetographic sub-spacecraft longitude at `SCET`, in degrees west, in the IAU coordinate system defined for the body. The body can be Saturn or Jupiter or one of their satellites. The NAIF ellipsoid for the surface of the body is used to determine the sub-spacecraft point. For Saturn and Jupiter, this is the ellipsoid defining the 1-bar level, for the satellites, including Titan, their surfaces. The longitude is that of the point nearest to the spacecraft on the NAIF ellipsoid.

`BODY_SUB_SPACECRAFT_LATITUDE`

The planetographic sub-spacecraft latitude, in degrees north. The latitude is that of the point nearest to the spacecraft on the NAIF ellipsoid.

`BODY_SUN_RANGE`

The distance from the Sun to the body, in km. This is computed at the `EPHEMERIS_TIME` minus one light travel time from the spacecraft to the body.

`BODY_SUN_RIGHT_ASCENSION`

The right ascension of the Sun as seen from the body, in degrees, in the J2000 coordinate system.

`BODY_SUN_DECLINATION`

The declination of the Sun as seen from the body, in degrees, in the J2000 coordinate system.

`BODY_SUB_SOLAR_LONGITUDE`

The longitude, in degrees west, of the sub-solar point on the body.

`BODY_SUB_SOLAR_LATITUDE`

The planetographic latitude, in degrees north, of the sub-solar point on the body.

`BODY_PHASE_ANGLE`

The angle in degrees between the radial vector of the spacecraft and the radial vector of the Sun, from the center of the body.

`BODY_ANGULAR_SEMIDIAMETER`

The equatorial angular radius of the body as seen by the spacecraft, in milliradians.

`BODY_ORBITAL_LONGITUDE`

The longitude of the body, in degrees, as measured from the anti-solar point on the primary. If the body and primary are the same body, this is set to -200.

`BODY_SYS3_LONGITUDE`

The longitude of the body in the primary's body fixed IAU coordinate system. If the body is the primary, this is set to -200.

`PRIMARY_ID`

The NAIF id of the primary. This will be the same as `BODY_ID` if the primary is the body.

`PRIMARY_SPACECRAFT_RANGE`

The distance from the primary to the spacecraft, in km.

`PRIMARY_SUB_SPACECRAFT_LONGITUDE`

The planetographic sub-spacecraft longitude, in degrees west, in the IAU coordinate system defined for the primary. The NAIF ellipsoid for the primary of the body is used to determine the sub-spacecraft point. For Saturn or Jupiter, this is the ellipsoid defining the 1-bar level.

`PRIMARY_SUB_SPACECRAFT_LATITUDE`

The planetographic primary sub-spacecraft latitude, in degrees north.

`PRIMARY_SUN_RANGE`

The distance from the Sun to the primary, in km.

`PRIMARY_SUB_SOLAR_LONGITUDE`

The longitude, in degrees west, of the sub-solar point on the primary.

`PRIMARY_SUB_SOLAR_LATITUDE`

The planetographic latitude, in degrees north, of the sub-solar point on the primary.

`PRIMARY_PHASE_ANGLE`

The angle in degrees between the radial vector of the spacecraft and the radial vector of the Sun, from the center of the primary.

`PRIMARY_ANGULAR_SEMIDIAMETER`

The equatorial angular radius of the primary as seen by the spacecraft, in milliradians.

`SCET_STRING`

The `SCET` in a human readible string form in ISO format. For example, an `SCET` of 1091318406 corresponds to an `SCET_STRING` of "2004-214T00:00:06".

3.6.8 POI tables

These tables store the pointing for each active detector at the ZPD of the IFM. One POI entry is generated for each detector and each non-rings non-primary target in the field of view. An entry is always generated for the primary as the target, whether or not it is in the field of view. Rings are described in a separate RIN table rather than as entries in the POI tables.

Some of the values are stored as array[9]. For these quantities, not only are the values at field of view center stored, but at other locations of the projected field of view as well, known as *Q points*. For FP1, the Q points are each of the eight cardinal compass points around the edge of the circular FOV are used. For the square mid-IR detectors, the corners of the FOV and midpoints on the lines connecting the corners are used (see Figure 4). Q point 5 is always the boresight of the detector. The NAIF routine **getfov_c** is used to get the vectors correspoinding to the Q points. All values, unless otherwise noted, are computed at `TIME_ZPD`.

Fields:

`TARGET_ID`

The NAIF id of the target. See table 3.

`PRIMARY_ID`

The NAIF id of the primary. This may be the same as `TARGET_ID`.

`SCET`

The SCET of the beginning of the scan.

`TIME_ZPD`

The time of ZPD.

`TIME_END`

The time of the end of the scan.

`DET`

The detector id. See Table 1.

`ALL_Q_ON`

A boolean argument, 1 if all 9 Q points for a given detector hit the target. If one or more miss, it's 0.

`LATITUDE_ZPD`

This array of nine values, in degrees north, gives the planetographic latitude of the "ray periapsis" for each of the individual Q points of a particular detector at `TIME_ZPD`. If the Q point is on the target, then this is the latitude of the intersection of the ray with the "surface". Otherwise, it is the latitude of the point on the ray closest to the surface.

`LONGITUDE_ZPD`

The array containing the longitudes, in degrees west, of the ray periapses for the Q points.

`ALTITUDE_ZPD`

The array containing the altitudes of the ray periapses for the Q points at `TIME_ZPD`. If a Q point is on target, this is zero. Otherwise, it's the altitude of the ray periapsis in km.

`RIGHT_ASCENSION`

An array containing the right ascension towards which each of the Q points is pointing, in degrees, in the J2000 coordinate system at `TIME_ZPD`.

`DECLINATION`

An array containing the declination towards which each of the Q points pointing, in degrees, in the J2000 coordinate system at `TIME_ZPD`.

`SPACECRAFT_TO_IMAGE_POINT_ZPD`

The distance along the Q points to either the "surface" of the target or the ray periapsis, in km.

`LATITUDE_END`

The planetographic latitude, at `TIME_END`, of the boresight (Q point 5) ray periapsis or intersection point on the target, in degrees north.

`LONGITUDE_END`

The longitude, at `TIME_END`, of the boresight (Q point 5) ray periapsis or intersection point on the target, in degrees west.

`ALTITUDE_END`

The altitude, at `TIME_END`, of the boresight (Q point 5). If the boresight is on target, this is zero. Otherwise, it's the altitude of the ray periapsis in km.

`SMEAR`

The `SMEAR` is defined in the following manner. The position of the boresight on the target at `TIME_ZPD` is computed and saved. Then, at `TIME_END`, the angle between that point and the position of the boresight at `TIME_END` is computed. `SMEAR` is the ratio of this angle with the angular width of the field of view of the detector, or in other words, is the fraction of the field of view that the original intersection point of the boresight has moved between `TIME_ZPD` and `TIME_END`. If the boresight at either `TIME_ZPD` or `TIME_END` is off the target, then this is set to -200.

`EMISSION_ANGLE`

This array contains the angle in degrees between the normal at the point on the surface at which the Q point intersects and the spacecraft direction vector, in degrees. If the Q point doesn't intersect the surface, this is set to 90 degrees.

`EMISSION_ANGLE_FOV_AVERAGE`

This is an average of the values of the emission angle over the field of view. It is currently computed using a Monte Carlo integration with 1000 points randomly distributed in the field of view. If a given point is off the target, it is rejected and a new one is tried. If the target isn't in the field of view, or more than 80% of the points attempted are off the target this is set to -200.

`FILLING_FACTOR`

This contains the fraction of the FOV filled by the target (between 0.0 and 1.0), computed during the same Monte Carlo integration as `EMISSION_ANGLE_FOV_AVERAGE`. For FP3 and FP4 this is the geometric filling factor, since their response functions are uniform. For FP1, the response function is approximately a gaussian (the full width at half maximum used is 2.42 mrad), and this is used in the computation of `EMISSION_ANGLE_FOV_AVERAGE` to select points preferentially towards the center of the detector where it is more sensitive. This means that `FILLING_FACTOR` is not geometric, but instead the center of the detector is given more weight than the edges.

`EMISSION_AZIMUTH_ANGLE`

This is the projection of the spacecraft position vector on the surface of the target and north, in degrees, for the boresight only. This projection is done for the ray periapsis if the boresight doesn't hit the surface, and so `ALTITUDE_ZPD[4]` should be checked to see if this contains a useful value.

`SOLAR_ZENITH`

The angle, for each Q point, between the direction vector of the Sun and the normal to the surface at point at which the Q point intersects the surface, in degrees.

`SOLAR_AZIMUTH`

The angle, for the boresight only, between the projection of the direction vector of the Sun and north on the surface of the target, in degrees.

`SOLAR_PHASE`

The angle, for the boresight only, between the Sun and the boresight at the point at which the boresight intersects the target, in degrees. If the boresight is off the target, this will be inaccurate.

`LOCAL_TIME`

The local solar time for each Q point is encoded as

$$\texttt{LOCAL\_TIME} = 100 \times \texttt{hour} + \texttt{min} + \texttt{sec}/100 \tag{1}$$

If the Q point doesn't intersect the surface, this is the local time of the ray periapsis, and so is probably of limited usefulness.

NAIF ID	BODY	NAIF ID	BODY
599	Jupiter	699	Saturn
501	Io	601	Mimas
502	Europa	602	Enceladus
503	Ganymede	603	Tethys
504	Callisto	604	Dione
505	Amalthea	605	Rhea
506	Himalia	606	Titan
507	Elara	607	Hyperion
508	Pasiphae	608	Iapetus
509	Sinope	609	Phoebe
510	Lysithea	610	Janus
511	Carme	611	Epimetheus
512	Ananke	612	Helene
513	Leda	613	Telesto
514	Thebe	614	Calypso
515	Adrastea	615	Atlas
516	Metis	616	Prometheus
		617	Pandora
		618	Pan

Table 3: Selected NAIF IDS of relevance to Cassini CIRS.

3.6.9 RIN tables

One entry is generated for each detector at all times. For the purpose of this table, the rings are assumed to fill the equatorial plane of the primary ($z = 0$ in the coordinate system of the primary with z along the rotation axis) out to a selected cutoff radius. If a line of sight intersects the primary body before intersecting the rings, or if the line of sight falls outside the cutoff radius, the values for most fields for that line of sight are set to -200.

As for the `POI` table, many entries in the `RIN` tables are array[9]. These contain individual entries for the Q points, as discussed above.

`PRIMARY_ID`

The NAIF id of the primary.

`SCET`

The SCET of the beginning of the scan.

`TIME_ZPD`

The time of ZPD, a few seconds after `scet`.

`TIME_END`

The time of the end of the scan.

`DET`

The detector ID.

`RING_RADIUS_ZPD`

An array containing the radius, in km, at which each Q point intercepts the equatorial plane of the primary. Set to -200 if the ray falls outside the rings or the primary is in front of the rings.

`RING_SPACECRAFT_RANGE_ZPD`

An array containing the distance, in km, from the spacecraft to the intercepts of the Q points on the equatorial plane. Set to -200 if the ray falls outside the rings or the primary is in front of the rings.

`RING_EMISSION_AZIMUTH_ANGLE`

This array contains the angles, in degrees, between the Q point vectors projected into the ring plane and the outward radial away from the primary's center. The Q point vectors are vectors along the Q point directions from the focal point of the instrument (center of the focal plane) to the ring plane. Set to -200 if the ray falls outside the rings or the primary is in front of the rings.

`RING_SOLAR_PHASE`

The angle, in degrees, between the Sun's position vector and the Q point's line of sight vectors. Set to -200 if the ray falls outside the rings or the primary is in front of the rings.

`RING_EMISSION_ANGLE`

An array containg the angles in degrees between lines of sight of the Q points and the rotation axis of the primary, which is assumed to be normal to the ring plane. Set to -200 if the ray falls outside the rings or the primary is in front of the rings.

`RING_LONGITUDE_ZPD`

This entry, for the boresight only, conforms to the PDS standard definition of a longitude on a ring. Longitudes are measured eastwards in an inertial frame from the "ascending node of the intersection of the ring plane with J2000". Entries here are in degrees. Set to -200 if the ray falls outside the rings or the primary is in front of the rings.

`RING_SOLAR_AZIMUTH`

This is the angle, in degrees, between the Sun's position vector projected into the ring plane and the outward radial from the center of the primary. This is computed for the boresight (Q point 5) only. Set to -200 if the ray falls outside the rings or the primary is in front of the rings. The angle increases counterclockwise when viewed from above the ring plane.

`RING_SOLAR_ZENITH`

The angle, in degrees, between the Sun's position vector and the rotation axis of the primary at the intercept point of the boresight. Set to -200 if the ray falls outside the rings or the primary is in front of the rings.

`RING_LONGITUDE_END`

The longitude, in degrees, for the boresight only, at `TIME_END`. This is also measured eastwards in an inertial frame from the "ascending node of the intersection of the ring plane with J2000". Set to -200 if the ray falls outside the rings or the primary is in front of the rings.

`RING_RADIUS_END`

The radius, in km, for the boresight of its intercept with the ring plane at `TIME_END`. Set to -200 if the ray falls outside the rings or the primary is in front of the rings.

`RING_SPACECRAFT_RANGE_END`

The distance, in km, from the spacecraft to the ring plane intercept point of the boresight at `TIME_END`. Set to -200 if the ray falls outside the rings or the primary is in front of the rings.

`RING_SMEAR`

This is computed much like `smear` in the pointing tables. The position of the boresight on the ring plane at `TIME_ZPD` is computed and saved. Then, at `TIME_END`, the angle between that point and the position of the boresight at `TIME_END` is computed. `RING_SMEAR` is the ratio of this angle with the angular width of the field of view of the detector, or in other words, is the fraction of the field of view that the original intersection point of the boresight has moved between `TIME_ZPD` and `TIME_END`. If the boresight at either `TIME_ZPD` or `TIME_END` is edge on to the rings (in the equatorial plane), then this is set to -200. These computations are done in the J2000 inertial frame, so the rotation of the primary does not contribute to the smear. This also means that the Keplerian motion of the ring particles does not contribute to the smear. Set to -200 if the ray falls outside the rings or the primary is in front of the rings.

`RING_LOCAL_TIME`

This contains the ring local hour angle in the primary's coordinate system for the intercept of the Q points in the ring plane. It is encoded by

$$\texttt{RING\_LOCAL\_TIME} = 100 \times \texttt{hour} + \texttt{min} + \texttt{sec}/100 \tag{2}$$

Set to -200 if the ray falls outside the rings or the primary is in front of the rings.

3.6.10 TAR tables

These tables describe which, if any, of the various targets are in the field of view of a given detector. Individual fields (e.g. TITAN) are set to 1 or 0 according to whether the body appears in the FOV or not. Any, all or none of these fields may be specified individually to select data with a particular subset of bodys in the FOV. Additionally, the field FOV_TARGETS contains the same information as the individual byte fields, except stored as bits in a single 4-byte integer. This may provide a useful shorthand for selecting an exact target body combination. Fields:

`SCET`

The SCET of the beginning of the scan.

`DET`

The detector ID. See Table 1.

`FOV_TARGETS`

A bitfield with a single bit for each of the targets that Cassini might observe. If the bit is 0, the object isn't in the field of view. If it's 1, it may or may not be, subject to the following conditions. For the planets and their satellites, a fixed angular width "penumbra" is attached outside their "surfaces" (for satellites, their actual hard surfaces, for Jupiter and Saturn, the 1 bar level in their atmospheres). The angular width of the penumbra is detector and body dependent. If any of the lines of sight of the 9 Q points intersects either the object or the penumbra, this bit is 1. If the center of the object is entirely inside the field of view of the detector, this bit is 1 (this condition is to account for objects small enough to fit entirely inside the detector and miss all of the Q points). A crude ring model is used. If the lines of sight of the Q points hit the equatorial plane of the primary within a fixed radius (129400 km for Jupiter, 140270 km for Saturn), and the primary isn't hit first, then the ring bit is 1. Otherwise, the ring bit is zero.

Bit assignments are as follows:

`00000000000000000000000000000000`	`space (no recognized target)`
`00000000000000000000000000000001`	`Jupiter rings`
`00000000000000000000000000000010`	`Jupiter`
`00000000000000000000000000000100`	`Io`

`00000000000000000000000000001000`	`Europa`
`00000000000000000000000000010000`	`Ganymede`
`00000000000000000000000000100000`	`Callisto`
`00000000000000000000000001000000`	`Saturn`
`00000000000000000000000010000000`	`Mimas`
`00000000000000000000000100000000`	`Enceladus`
`00000000000000000000001000000000`	`Tethys`
`00000000000000000000010000000000`	`Dione`
`00000000000000000000100000000000`	`Rhea`
`00000000000000000001000000000000`	`Titan`
`00000000000000000010000000000000`	`Hyperion`
`00000000000000000100000000000000`	`Iapetus`
`00000000000000001000000000000000`	`Phoebe`
`00000000000000010000000000000000`	`Janus`
`00000000000000100000000000000000`	`Epimetheus`
`00000000000001000000000000000000`	`Helene`
`00000000000010000000000000000000`	`Telesto`
`00000000000100000000000000000000`	`Calypso`
`00000000001000000000000000000000`	`Atlas`
`00000000010000000000000000000000`	`Prometheus`
`00000000100000000000000000000000`	`Pandora`
`00000001000000000000000000000000`	`Pan`
`00000010000000000000000000000000`	`Saturn rings`
`10000000000000000000000000000000`	`one of the stars in the stars file is in the field of view`

Note that the binary bit field must be converted to a decimal number for use in Vanilla queries. As an example, use `FOV_TARGETS 2 2` to search for Jupiter, `FOV_TARGETS 4 4` for Io, `FOV_TARGETS 6 6` for Jupiter and Io etc. The `FOV_TARGETS` is an exact match only, hence the first example would exclude Jovian spectra where a moon was also in the FOV, or the ring plane. Note that because the Jovian ring plane was essentially transparent to CIRS, the numbering system was set up to allow queries such as: `FOV_TARGETS 2 3` which allows Jupiter, or Jupiter + Jovian rings.

`JRING` Set to 1 if the Jupiter rings are in the FOV, 0 otherwise.

`JUPITER` Set to 1 if Jupiter is in the FOV, 0 otherwise.

`IO` Set to 1 if Io is in the FOV, 0 otherwise.

`EUROPA` Set to 1 if Europa is in the FOV, 0 otherwise.

`GANYMEDE` Set to 1 if Ganymede is in the FOV, 0 otherwise.

`CALLISTO` Set to 1 if Callisto is in the FOV, 0 otherwise.

`SATURN` Set to 1 if Saturn is in the FOV, 0 otherwise.

`MIMAS` Set to 1 if Mimas is in the FOV, 0 otherwise.

`ENCELADUS` Set to 1 if Enceladus is in the FOV, 0 otherwise.

`TETHYS` Set to 1 if Tethysis in the FOV, 0 otherwise.

`DIONE` Set to 1 if Dione is in the FOV, 0 otherwise.

`RHEA` Set to 1 if Rhea is in the FOV, 0 otherwise.

`TITAN` Set to 1 if Tian is in the FOV, 0 otherwise.

`HYPERION` Set to 1 if Hyperion is in the FOV, 0 otherwise.

`IAPETUS` Set to 1 if Iapetus is in the FOV, 0 otherwise.

`PHOEBE` Set to 1 if Phoebe is in the FOV, 0 otherwise.

`JANUS` Set to 1 if Janus is in the FOV, 0 otherwise.

`EPHMETHEUS` Set to 1 if Epimetheus is in the FOV, 0 otherwise.

`HELENE` Set to 1 if Helene is in the FOV, 0 otherwise.

`TELESTO` Set to 1 if Telesto is in the FOV, 0 otherwise.

`CALYPSO` Set to 1 if Calypso is in the FOV, 0 otherwise.

`ATLAS` Set to 1 if Atlas is in the FOV, 0 otherwise.

`PROMETHEUS` Set to 1 if Prometheus is in the FOV, 0 otherwise.

`PANDORA` Set to 1 if Pandora is in the FOV, 0 otherwise.

`PAN` Set to 1 if Pan is in the FOV, 0 otherwise.

`SRING` Set to 1 if Saturn's rings are in the FOV, 0 otherwise.

`DEEP_SPACE` Set to 1 if `FOV_TARGETS` is zero, 0 otherwise.

`STELLAR` Set to 1 if a stellar target is in the FOV, 0 otherwise.

3.6.11 ISPM tables

These tables contain the 'interpolated' (resampled) calibrated spectra. The ISPM fragment type is now commonly created, instead of the SPM type (non-resampled), and is the type to be archived. The SPM type created formerly used a 'natural' wavelength index resulting from the instrument, which meant that the final spectrum fell on 'irrational' wavenumber values such 577.3256, 577.7792, instead of 577.250, 577,500, 577.750 etc. The ISPM type contains the same spectra, but interpolated and re-sampled onto a more user-friendly grid. There is no loss of resolution or data in this process.

Fields:

`SCET`

The spacecraft event time of the beginning of scan (seconds from epoch). The `cirs_time` utility may be useful for converting text time to SCET.

`DET`

Detector number (see table 1).

`ISPTS`

The number of points in the output spectrum.

`DS_NAVE`

Number of spectra used in the deep-space calibration average IFM, during the calibration process. This may pose an intrinsic limit to how many calibrated spectra may be co-added. See `SH_NAVE`.

`SH_NAVE`

Number of spectra used in the shutter calibration average IFM, during the calibration process. This may pose an intrinsic limit to how many calibrated spectra may be co-added. For example, assume that the shutter is 170 K and the planet spectrum is similar, and `SH_NAVE`=100. Now, consider that we have 500 planet spectra which were calibated using the same 100-spectrum shutter average. Then, co-adding more than 100 planet spectra will no longer reduce the random noise level any further, because the NESR level of the 100 shutter will become the limiting (biggest) noise source.

`TINSTR`

The instrument temperature (kelvin). For FP1 (detector 0), the **`firpolriztem`** from the IHSK tables is used. For the mid-IR (detectors 1–30) the **`hsecooltem`** from the IHSK is used.

`IWN_START`

The wavenumber of the first point in the spectrum (cm−1). Together with `IWN_STEP` defines the spectrum wavenumber axis.

`IWN_STEP`

The wavenumber increment size between successive points in the spectrum (cm−1). Together with `IWN_START` defines the spectrum wavenumber axis.

`APODTYPE`

The number of the apodization type applied to the final spectrum (if any). See table 4. Apodization is the process of suppressing side-lobes ('ripples') of the ILS by tapering the IFM in the spatial dimension, before FFTing.

`FWHM`

The Full-Width at Half-Maximum of the idealised monochromatic instrumental line shape (ILS) in cm−1. If the spectrum is apodized, then it is the apodized FWHM.

`RAYLEIGH`

The 'Rayleigh' resolution of the spectrum (cm−1). Actually, the distance from the peak of an isolated line to the first 'null' (zero-crossing, not minimum) in the ILS. (Rayleigh resolution is actually defined on the basis of a line doublet being resolved, not individual spectral lines).

No.	Apodisation Function Name
0	Boxcar (no apodisation)
1	Norton & Beer type 1 function
2	Norton & Beer type 2 function
3	Norton & Beer type 3 function
4	Forman type
5	Triangular type
6	Hanning type
7	Hamming type

Table 4: Apodisation functions available for post-processing CIRS spectra.

`NYQUIST`

The Nyquist bin size (cm−1), or intrinsic resolution limit. This is equivalent to the width of the bandpass divided by the number of interferogram samples. This number gives a measure of how far the actual resolution achieved, perhaps after apodization, differs from the 'natural' resolution of the instrument.

`POWER`

The integrated radiance under the power spectrum (W cm^{-2} sr^{-1}).

`DS_SCET`

The mean scet time of the deep space scan block used to create the deep space average IFM used in the calibration of the current target IFM.

`DS_SH_SCET`

The mean scet time of the shutter scan block used to create the shutter average IFM used in the calibration of the current target IFM.

`ISPM`

A pointer to the spectrum in the associated variable-length record (`.VAR`) file. The spectral data points are radiance in units of (W cm^{-2} sr^{-1} $(\text{cm}^{-1})^{-1}$).

4 Software

4.1 Calibration Software

4.2 Utility Software

4.3 Vanilla software

The MGS TES project has produced a software tool that not only reads the PDS table and the variable length records, but is also capable of joining the related records among multiple tables. This piece of software is called 'vanilla' and was offered for use by the CIRS team. Vanilla is included on every volume, along with the *Vanilla Users Guide* (written for MGS-TES, but nevertheless useful). See also the examples in the document `Vanilla Examples with the CIRS database`.

The Vanilla program was developed for use on both LSB and MSB machines.

4.4 PDS software

The CIRS team does not use any PDS software to process or view the data. However the tables are stored using the PDS TABLE standard structure and any tool that understands that structure should be able to read all of the data except the variable length spectra.

A Acronyms and Abbreviations

BIU Bus Interface Unit; interface to spacecraft bus.

CIRS Composite Infrared Spectrometer.

FIR Far Infra-Red.

FOV Field Of View.

FWHM Full-width to half maximum. A measure of spectral line width or FOV width.

IFM Interferogram.

ILS Instrumental Line Shape

LSB Least Significant Bit, data storage convention.

MIR Mid Infra-Red.

MGS Mars Global Surveyor spacecraft.

MSB Most Significant Bit, data storage convention.

NAIF Navigation and Ancilliary Information Facility

NESR Noise Equivalent Spectral Radiance

ODL Object Description Lanaguage.

PDS Planetary Data System.

RTI Real Time Interrupt (= 1/8 sec).

TES Thermal Emission Spectrometer (on MGS).

TSDR Time Sequential Data Records.

ZPD Zero-Path Difference; IFM peak.

B Data format files

B.1 OBS.FMT

```
COLUMNS                 = 39
ROW_BYTES               = 51

OBJECT                  = COLUMN
    NAME                = SCET
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 1
    BYTES               = 4
    DESCRIPTION         = "packet time"
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = SCLK
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 5
    BYTES               = 4
    DESCRIPTION         = "spacecraft clock time "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = RTI
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 9
    BYTES               = 2
    DESCRIPTION         = "Length of scan in Real Time Interrupts
                           (1/8 sec) "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = FP3_MODE
    DATA_TYPE           = CHARACTER
    START_BYTE          = 11
    BYTES               = 1
    DESCRIPTION         = "FP3 array pixel mode: O(dd), E(ven),
                          C(enter), P(airs) "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = FP4_MODE
    DATA_TYPE           = CHARACTER
    START_BYTE          = 12
    BYTES               = 1
```

```
    DESCRIPTION          = "FP4 array pixel mode: O(dd), E(ven),
                            C(enter), P(airs) "
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = FIR_OVERFLOW
    DATA_TYPE            = LSB_UNSIGNED_INTEGER
    START_BYTE           = 13
    BYTES                = 1
    DESCRIPTION          = "Overflow detected on FIR accumulation"
END_OBJECT

OBJECT                   = COLUMN
    NAME                 = FP1_OVERFLOW
    DATA_TYPE            = LSB_UNSIGNED_INTEGER
    START_BYTE           = 14
    BYTES                = 1
    DESCRIPTION          = "Overflow on FP1 average buffer"
END_OBJECT

OBJECT                   = COLUMN
    NAME                 = FP3_OVERFLOW
    DATA_TYPE            = LSB_UNSIGNED_INTEGER
    START_BYTE           = 15
    BYTES                = 1
    DESCRIPTION          = "Overflow on FP3 average buffer"
END_OBJECT

OBJECT                   = COLUMN
    NAME                 = FP4_OVERFLOW
    DATA_TYPE            = LSB_UNSIGNED_INTEGER
    START_BYTE           = 16
    BYTES                = 1
    DESCRIPTION          = "Overflow on FP4 average buffer"
END_OBJECT

OBJECT                   = COLUMN
    NAME                 = FP1_COMP_OVERFLOW
    DATA_TYPE            = LSB_UNSIGNED_INTEGER
    START_BYTE           = 17
    BYTES                = 1
    DESCRIPTION          = "Overflow on compression buffer 1"
END_OBJECT

OBJECT                   = COLUMN
    NAME                 = FP3_COMP_OVERFLOW
    DATA_TYPE            = LSB_UNSIGNED_INTEGER
    START_BYTE           = 18
```

```
    BYTES                = 1
    DESCRIPTION          = "Overflow on compression buffer 3"
END_OBJECT

OBJECT                   = COLUMN
    NAME                 = FP4_COMP_OVERFLOW
    DATA_TYPE            = LSB_UNSIGNED_INTEGER
    START_BYTE           = 19
    BYTES                = 1
    DESCRIPTION          = "Overflow on compression buffer 4"
END_OBJECT

OBJECT                   = COLUMN
    NAME                 = FP3_ZERO
    DATA_TYPE            = LSB_UNSIGNED_INTEGER
    START_BYTE           = 20
    BYTES                = 1
    DESCRIPTION          = "Detected 0000H on FP3 raw data"
END_OBJECT

OBJECT                   = COLUMN
    NAME                 = FP3_0FFF
    DATA_TYPE            = LSB_UNSIGNED_INTEGER
    START_BYTE           = 21
    BYTES                = 1
    DESCRIPTION          = "Detected 0FFFH on FP3 raw data"
END_OBJECT

OBJECT                   = COLUMN
    NAME                 = FP1_ZERO
    DATA_TYPE            = LSB_UNSIGNED_INTEGER
    START_BYTE           = 22
    BYTES                = 1
    DESCRIPTION          = "Detected 0000H on FP1 raw data"
END_OBJECT

OBJECT                   = COLUMN
    NAME                 = FP1_0FFF
    DATA_TYPE            = LSB_UNSIGNED_INTEGER
    START_BYTE           = 23
    BYTES                = 1
    DESCRIPTION          = "Detected 0FFFH on FP1 raw data"
END_OBJECT

OBJECT                   = COLUMN
    NAME                 = FP4_ZERO
    DATA_TYPE            = LSB_UNSIGNED_INTEGER
    START_BYTE           = 24
```

```
    BYTES                = 1
    DESCRIPTION          = "Detected 0000H on FP4 raw data"
END_OBJECT

OBJECT                   = COLUMN
    NAME                 = FP4_0FFF
    DATA_TYPE            = LSB_UNSIGNED_INTEGER
    START_BYTE           = 25
    BYTES                = 1
    DESCRIPTION          = "Detected 0FFFH on FP4 raw data"
END_OBJECT

OBJECT                   = COLUMN
    NAME                 = COADD
    DATA_TYPE            = LSB_UNSIGNED_INTEGER
    START_BYTE           = 26
    BYTES                = 1
    DESCRIPTION          = "Coadding enabled"
END_OBJECT

OBJECT                   = COLUMN
    NAME                 = SCIENCE_OVERWRITTEN
    DATA_TYPE            = LSB_UNSIGNED_INTEGER
    START_BYTE           = 27
    BYTES                = 1
    DESCRIPTION          = "Science data overwritten condition detected"
END_OBJECT

OBJECT                   = COLUMN
    NAME                 = FLYBACK_EXCEEDED
    DATA_TYPE            = LSB_UNSIGNED_INTEGER
    START_BYTE           = 28
    BYTES                = 1
    DESCRIPTION          = "Flyback exceeded"
END_OBJECT

OBJECT                   = COLUMN
    NAME                 = SCAN_EXCEEDED
    DATA_TYPE            = LSB_UNSIGNED_INTEGER
    START_BYTE           = 29
    BYTES                = 1
    DESCRIPTION          = "Scan exceeded"
END_OBJECT

OBJECT                   = COLUMN
    NAME                 = MECHANISM_ENABLED
    DATA_TYPE            = LSB_UNSIGNED_INTEGER
    START_BYTE           = 30
```

```
    BYTES               = 1
    DESCRIPTION         = "Mechanism enabled"
END_OBJECT

OBJECT                  = COLUMN
    NAME                = MECH_OUT_OF_PHASE
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 31
    BYTES               = 1
    DESCRIPTION         = "Mechanism out of phase some time"
END_OBJECT

OBJECT                  = COLUMN
    NAME                = LASER_A_ENABLED
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 32
    BYTES               = 1
    DESCRIPTION         = "Laser A enabled"
END_OBJECT

OBJECT                  = COLUMN
    NAME                = LASER_B_ENABLED
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 33
    BYTES               = 1
    DESCRIPTION         = "Laser B enabled"
END_OBJECT

OBJECT                  = COLUMN
    NAME                = MECH_CMDED
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 34
    BYTES               = 1
    DESCRIPTION         = "Enable mechanism cmded"
END_OBJECT

OBJECT                  = COLUMN
    NAME                = WHITE_OVERRIDE
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 35
    BYTES               = 1
    DESCRIPTION         = "White light override"
END_OBJECT

OBJECT                  = COLUMN
    NAME                = OPTICAL_SENSE_MODE
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 36
```

```
    BYTES                 = 1
    DESCRIPTION           = "Optical sensor (0), decs (1) mode"
END_OBJECT

OBJECT                    = COLUMN
    NAME                  = SHUTTER
    DATA_TYPE             = LSB_UNSIGNED_INTEGER
    START_BYTE            = 37
    BYTES                 = 1
    DESCRIPTION           = "Shutter open (0), closed (1)"
END_OBJECT

OBJECT                    = COLUMN
    NAME                  = RIE_LASCMD_LSB
    DATA_TYPE             = LSB_UNSIGNED_INTEGER
    START_BYTE            = 38
    BYTES                 = 1
    DESCRIPTION           = "RIE: LASCMD_LSB Lamp current select"
END_OBJECT

OBJECT                    = COLUMN
    NAME                  = RIE_LASCMD_MSB
    DATA_TYPE             = LSB_UNSIGNED_INTEGER
    START_BYTE            = 39
    BYTES                 = 1
    DESCRIPTION           = "RIE: LASCMD_MSB Lamp current select"
END_OBJECT

OBJECT                    = COLUMN
    NAME                  = RIE_LASCMD_A
    DATA_TYPE             = LSB_UNSIGNED_INTEGER
    START_BYTE            = 40
    BYTES                 = 1
    DESCRIPTION           = "RIE: LASCMD_A Lamp A select"
END_OBJECT

OBJECT                    = COLUMN
    NAME                  = RIE_LASCMD_B
    DATA_TYPE             = LSB_UNSIGNED_INTEGER
    START_BYTE            = 41
    BYTES                 = 1
    DESCRIPTION           = "RIE: LASCMD_B Lamp B select"
END_OBJECT

OBJECT                    = COLUMN
    NAME                  = RAW_NO_SET
    DATA_TYPE             = LSB_UNSIGNED_INTEGER
    START_BYTE            = 42
```

```
        BYTES                  = 2
        DESCRIPTION            = "Count of raw samples without set control bits"
    END_OBJECT

    OBJECT                     = COLUMN
        NAME                   = RAW_FP1_COUNT
        DATA_TYPE              = LSB_UNSIGNED_INTEGER
        START_BYTE             = 44
        BYTES                  = 2
        DESCRIPTION            = "Number of FP1 samples read in this scan"
    END_OBJECT

    OBJECT                     = COLUMN
        NAME                   = RAW_FP3_COUNT
        DATA_TYPE              = LSB_UNSIGNED_INTEGER
        START_BYTE             = 46
        BYTES                  = 2
        DESCRIPTION            = "Number of FP3 samples read in this scan"
    END_OBJECT

    OBJECT                     = COLUMN
        NAME                   = RAW_FP4_COUNT
        DATA_TYPE              = LSB_UNSIGNED_INTEGER
        START_BYTE             = 48
        BYTES                  = 2
        DESCRIPTION            = "Number of FP4 samples read in this scan"
    END_OBJECT

    OBJECT                     = COLUMN
        NAME                   = FIRST_SAMPLE_RTI
        DATA_TYPE              = LSB_UNSIGNED_INTEGER
        START_BYTE             = 50
        BYTES                  = 2
        DESCRIPTION            = "RTI value when 1st sample in scan collected"
    END_OBJECT
```

B.2 FRV.FMT

```
    COLUMNS                    = 3
    ROW_BYTES                  = 10

    OBJECT                     = COLUMN
        NAME                   = SCET
        DATA_TYPE              = LSB_UNSIGNED_INTEGER
        START_BYTE             = 1
        BYTES                  = 4
```

```
    DESCRIPTION         = "scan time "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = NFRV
    DATA_TYPE           = LSB_INTEGER
    START_BYTE          = 5
    BYTES               = 2
    DESCRIPTION         = "number of fringe voltage samples "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = FRV
    DATA_TYPE           = LSB_INTEGER
    START_BYTE          = 7
    BYTES               = 4
    VAR_DATA_TYPE       = PC_REAL
    VAR_ITEM_BYTES      = 8
    VAR_RECORD_TYPE     = VAX_VARIABLE_LENGTH
    DESCRIPTION         = "pointer to fringe voltage data "
END_OBJECT              = COLUMN
```

B.3 IFGM.FMT

```
COLUMNS                 = 4
ROW_BYTES               = 11

OBJECT                  = COLUMN
    NAME                = SCET
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 1
    BYTES               = 4
    DESCRIPTION         = "SCAN TIME "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = DET
    DATA_TYPE           = LSB_INTEGER
    START_BYTE          = 5
    BYTES               = 1
    DESCRIPTION         = "DETECTOR NUMBER "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = NPTS
```

```
    DATA_TYPE           = LSB_INTEGER
    START_BYTE          = 6
    BYTES               = 2
    DESCRIPTION         = "NUMBER OF POINTS IN INTERFEROGRAM "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = IFGM
    DATA_TYPE           = LSB_INTEGER
    START_BYTE          = 8
    BYTES               = 4
    VAR_DATA_TYPE       = LSB_INTEGER
    VAR_ITEM_BYTES      = 2
    VAR_RECORD_TYPE     = VAX_VARIABLE_LENGTH
    DESCRIPTION         = "POINTER TO INTERFEROGRAM DATA "
END_OBJECT              = COLUMN
```

B.4 HSK.FMT

```
COLUMNS                 = 62
ROW_BYTES               = 402

OBJECT                  = COLUMN
    NAME                = SCET
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 1
    BYTES               = 4
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = SMERIESTAT
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 5
    BYTES               = 2
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = FP3LASTCMD
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 7
    BYTES               = 2
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = FP4LASTCMD
```

```
    DATA_TYPE               = LSB_UNSIGNED_INTEGER
    START_BYTE              = 9
    BYTES                   = 2
END_OBJECT                  = COLUMN

OBJECT                      = COLUMN
    NAME                    = TCMLASTCMD
    DATA_TYPE               = LSB_UNSIGNED_INTEGER
    START_BYTE              = 11
    BYTES                   = 2
END_OBJECT                  = COLUMN

OBJECT                      = COLUMN
    NAME                    = FP1MAX
    DATA_TYPE               = LSB_UNSIGNED_INTEGER
    START_BYTE              = 13
    BYTES                   = 2
END_OBJECT                  = COLUMN

OBJECT                      = COLUMN
    NAME                    = FP3MAX
    DATA_TYPE               = LSB_UNSIGNED_INTEGER
    START_BYTE              = 15
    BYTES                   = 2
END_OBJECT                  = COLUMN

OBJECT                      = COLUMN
    NAME                    = FP4MAX
    DATA_TYPE               = LSB_UNSIGNED_INTEGER
    START_BYTE              = 17
    BYTES                   = 2
END_OBJECT                  = COLUMN

OBJECT                      = COLUMN
    NAME                    = FRINGEMAX
    DATA_TYPE               = PC_REAL
    START_BYTE              = 19
    BYTES                   = 8
END_OBJECT                  = COLUMN

OBJECT                      = COLUMN
    NAME                    = FRINGEMIN
    DATA_TYPE               = PC_REAL
    START_BYTE              = 27
    BYTES                   = 8
END_OBJECT                  = COLUMN

OBJECT                      = COLUMN
```

```
    NAME                 = PENDINGTTAG
    DATA_TYPE            = LSB_UNSIGNED_INTEGER
    START_BYTE           = 35
    BYTES                = 2
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = CODECHK
    DATA_TYPE            = LSB_UNSIGNED_INTEGER
    START_BYTE           = 37
    BYTES                = 2
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = REJECTOP
    DATA_TYPE            = LSB_UNSIGNED_INTEGER
    START_BYTE           = 39
    BYTES                = 2
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = PCA_ATEM
    DATA_TYPE            = PC_REAL
    START_BYTE           = 41
    BYTES                = 8
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = PCA_BTEM
    DATA_TYPE            = PC_REAL
    START_BYTE           = 49
    BYTES                = 8
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = PCA_CTEM
    DATA_TYPE            = PC_REAL
    START_BYTE           = 57
    BYTES                = 8
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = SME_ATEM
    DATA_TYPE            = PC_REAL
    START_BYTE           = 65
    BYTES                = 8
END_OBJECT               = COLUMN
```

```
OBJECT                  = COLUMN
    NAME                = SME_BTEM
    DATA_TYPE           = PC_REAL
    START_BYTE          = 73
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = RIE_ATEM
    DATA_TYPE           = PC_REAL
    START_BYTE          = 81
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = FEE_ATEM
    DATA_TYPE           = PC_REAL
    START_BYTE          = 89
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = FEE_BTEM
    DATA_TYPE           = PC_REAL
    START_BYTE          = 97
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = FEE_CTEM
    DATA_TYPE           = PC_REAL
    START_BYTE          = 105
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = FEE_DTEM
    DATA_TYPE           = PC_REAL
    START_BYTE          = 113
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = TCM_ATEM
    DATA_TYPE           = PC_REAL
    START_BYTE          = 121
    BYTES               = 8
END_OBJECT              = COLUMN
```

```
OBJECT                  = COLUMN
    NAME                = TCM_BTEM
    DATA_TYPE           = PC_REAL
    START_BYTE          = 129
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = BASEPLATETEM
    DATA_TYPE           = PC_REAL
    START_BYTE          = 137
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = PROCESSORTEM
    DATA_TYPE           = PC_REAL
    START_BYTE          = 145
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = BIUTEM
    DATA_TYPE           = PC_REAL
    START_BYTE          = 153
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = ADCTEM
    DATA_TYPE           = PC_REAL
    START_BYTE          = 161
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = MOTORCURRENT
    DATA_TYPE           = PC_REAL
    START_BYTE          = 169
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = DECSPOSN
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 177
    BYTES               = 2
```

```
END_OBJECT                = COLUMN

OBJECT                    = COLUMN
    NAME                  = VELERR
    DATA_TYPE             = PC_REAL
    START_BYTE            = 179
    BYTES                 = 8
END_OBJECT                = COLUMN

OBJECT                    = COLUMN
    NAME                  = PHASEERR
    DATA_TYPE             = LSB_UNSIGNED_INTEGER
    START_BYTE            = 187
    BYTES                 = 2
END_OBJECT                = COLUMN

OBJECT                    = COLUMN
    NAME                  = FRINGEV
    DATA_TYPE             = PC_REAL
    START_BYTE            = 189
    BYTES                 = 8
END_OBJECT                = COLUMN

OBJECT                    = COLUMN
    NAME                  = RIESMEPOS12V
    DATA_TYPE             = PC_REAL
    START_BYTE            = 197
    BYTES                 = 8
END_OBJECT                = COLUMN

OBJECT                    = COLUMN
    NAME                  = RIESMENEG12V
    DATA_TYPE             = PC_REAL
    START_BYTE            = 205
    BYTES                 = 8
END_OBJECT                = COLUMN

OBJECT                    = COLUMN
    NAME                  = RIESMEPOS5V
    DATA_TYPE             = PC_REAL
    START_BYTE            = 213
    BYTES                 = 8
END_OBJECT                = COLUMN

OBJECT                    = COLUMN
    NAME                  = LASERCURRENT
    DATA_TYPE             = PC_REAL
    START_BYTE            = 221
```

```
    BYTES              = 8
END_OBJECT             = COLUMN

OBJECT                 = COLUMN
    NAME               = LEDCURRENT
    DATA_TYPE          = PC_REAL
    START_BYTE         = 229
    BYTES              = 8
END_OBJECT             = COLUMN

OBJECT                 = COLUMN
    NAME               = IDSPOS15V
    DATA_TYPE          = PC_REAL
    START_BYTE         = 237
    BYTES              = 8
END_OBJECT             = COLUMN

OBJECT                 = COLUMN
    NAME               = IDSNEG15V
    DATA_TYPE          = PC_REAL
    START_BYTE         = 245
    BYTES              = 8
END_OBJECT             = COLUMN

OBJECT                 = COLUMN
    NAME               = IDSPOS5V
    DATA_TYPE          = PC_REAL
    START_BYTE         = 253
    BYTES              = 8
END_OBJECT             = COLUMN

OBJECT                 = COLUMN
    NAME               = PHOTODIODE
    DATA_TYPE          = LSB_UNSIGNED_INTEGER
    START_BYTE         = 261
    BYTES              = 2
END_OBJECT             = COLUMN

OBJECT                 = COLUMN
    NAME               = HTR80KCURR
    DATA_TYPE          = PC_REAL
    START_BYTE         = 263
    BYTES              = 8
END_OBJECT             = COLUMN

OBJECT                 = COLUMN
    NAME               = HTROP_ACURR
    DATA_TYPE          = PC_REAL
```

```
    START_BYTE           = 271
    BYTES                = 8
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = HTRP_MRCURR
    DATA_TYPE            = PC_REAL
    START_BYTE           = 279
    BYTES                = 8
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = HTRS_MRCURR
    DATA_TYPE            = PC_REAL
    START_BYTE           = 287
    BYTES                = 8
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = HSECOOLTEM
    DATA_TYPE            = PC_REAL
    START_BYTE           = 295
    BYTES                = 8
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = FIRPOLRIZTEM
    DATA_TYPE            = PC_REAL
    START_BYTE           = 303
    BYTES                = 8
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = HSEABVMECHTEM
    DATA_TYPE            = PC_REAL
    START_BYTE           = 311
    BYTES                = 8
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = FP1DETTEM
    DATA_TYPE            = PC_REAL
    START_BYTE           = 319
    BYTES                = 8
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = MECHHSETEM
```

```
    DATA_TYPE            = PC_REAL
    START_BYTE           = 327
    BYTES                = 8
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = SECMIRRTEM
    DATA_TYPE            = PC_REAL
    START_BYTE           = 335
    BYTES                = 8
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = PRIMIRRTEM
    DATA_TYPE            = PC_REAL
    START_BYTE           = 343
    BYTES                = 8
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = SECMIRBAFFTEM
    DATA_TYPE            = PC_REAL
    START_BYTE           = 351
    BYTES                = 8
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = PMSUNSHDMLITEM
    DATA_TYPE            = PC_REAL
    START_BYTE           = 359
    BYTES                = 8
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = PMSUNSHDRADTEM
    DATA_TYPE            = PC_REAL
    START_BYTE           = 367
    BYTES                = 8
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = CONERIMTEM
    DATA_TYPE            = PC_REAL
    START_BYTE           = 375
    BYTES                = 8
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
```

```
    NAME                  = FPATEM
    DATA_TYPE             = PC_REAL
    START_BYTE            = 383
    BYTES                 = 8
END_OBJECT                = COLUMN

OBJECT                    = COLUMN
    NAME                  = THERMISTORDRV
    DATA_TYPE             = LSB_UNSIGNED_INTEGER
    START_BYTE            = 391
    BYTES                 = 2
END_OBJECT                = COLUMN

OBJECT                    = COLUMN
    NAME                  = IDSCALIB
    DATA_TYPE             = LSB_UNSIGNED_INTEGER
    START_BYTE            = 393
    BYTES                 = 2
END_OBJECT                = COLUMN

OBJECT                    = COLUMN
    NAME                  = IDSNEG5V
    DATA_TYPE             = PC_REAL
    START_BYTE            = 395
    BYTES                 = 8
END_OBJECT                = COLUMN
```

B.5 IHSK.FMT

```
COLUMNS                   = 12
ROW_BYTES                 = 92

OBJECT                    = COLUMN
    NAME                  = SCET
    DATA_TYPE             = LSB_UNSIGNED_INTEGER
    START_BYTE            = 1
    BYTES                 = 4
END_OBJECT                = COLUMN

OBJECT                    = COLUMN
    NAME                  = HSECOOLTEM
    DATA_TYPE             = PC_REAL
    START_BYTE            = 5
    BYTES                 = 8
END_OBJECT                = COLUMN
```

```
OBJECT                  = COLUMN
    NAME                = FIRPOLRIZTEM
    DATA_TYPE           = PC_REAL
    START_BYTE          = 13
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = ZINSTRADTEM
    DATA_TYPE           = PC_REAL
    START_BYTE          = 21
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = FP1DETTEM
    DATA_TYPE           = PC_REAL
    START_BYTE          = 29
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = MECHHSETEM
    DATA_TYPE           = PC_REAL
    START_BYTE          = 37
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = SECMIRRTEM
    DATA_TYPE           = PC_REAL
    START_BYTE          = 45
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = PRIMIRRTEM
    DATA_TYPE           = PC_REAL
    START_BYTE          = 53
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = SECMIRBAFFTEM
    DATA_TYPE           = PC_REAL
    START_BYTE          = 61
    BYTES               = 8
END_OBJECT              = COLUMN
```

```
    OBJECT                  = COLUMN
        NAME                = PMSUNSHDMLITEM
        DATA_TYPE           = PC_REAL
        START_BYTE          = 69
        BYTES               = 8
    END_OBJECT              = COLUMN

    OBJECT                  = COLUMN
        NAME                = PMSUNSHDRADTEM
        DATA_TYPE           = PC_REAL
        START_BYTE          = 77
        BYTES               = 8
    END_OBJECT              = COLUMN

    OBJECT                  = COLUMN
        NAME                = FPATEM
        DATA_TYPE           = PC_REAL
        START_BYTE          = 85
        BYTES               = 8
    END_OBJECT              = COLUMN
```

B.6 DIAG.FMT

```
COLUMNS                 = 6
    ROW_BYTES               = 11

    OBJECT                  = COLUMN
        NAME                = SCET
        DATA_TYPE           = LSB_UNSIGNED_INTEGER
        START_BYTE          = 1
        BYTES               = 4
        DESCRIPTION         = "spacecraft clock time "
    END_OBJECT              = COLUMN

    OBJECT                  = COLUMN
        NAME                = DET
        DATA_TYPE           = LSB_INTEGER
        START_BYTE          = 5
        BYTES               = 1
        DESCRIPTION         = "detector number "
    END_OBJECT              = COLUMN

    OBJECT                  = COLUMN
        NAME                = BIURTI
        DATA_TYPE           = LSB_INTEGER
```

```
        START_BYTE          = 6
        BYTES               = 2
        DESCRIPTION         = "RTI offset of BIU interference"
    END_OBJECT              = COLUMN

    OBJECT                  = COLUMN
        NAME                = NOISE
        DATA_TYPE           = LSB_INTEGER
        START_BYTE          = 8
        BYTES               = 2
        DESCRIPTION         = "Description of interferogram:
                               2^0 - Normal
                               2^1 - DC Level
                               2^2 - Drift
                               2^3 - (Big) Spikes
                               2^4 - Std Dev
                               2^5 - Long
                               2^6 - Short
                               2^7 - Insuf.Samples
                               2^8 - Avg Ifm
                               2^9 - Invalid Time Stamp
                               2^10- Shutter
                               2^11- ZPD Position
                            "
    END_OBJECT              = COLUMN

OBJECT                  = COLUMN
        NAME                = DSCAL
        DATA_TYPE           = LSB_INTEGER
        START_BYTE          = 10
        BYTES               = 1
        DESCRIPTION         = "1-DSCAL, 0-rest sequenses"
    END_OBJECT              = COLUMN

OBJECT                  = COLUMN
        NAME                = LASER
        DATA_TYPE           = LSB_INTEGER
        START_BYTE          = 11
        BYTES               = 1
        DESCRIPTION         = "1 - [4.95 - 5.25V]
                               2 - [5.55 - 5.85V]
                               3 - [5.95 - 6.25V]
                               0 - the rest RIE[2]
                              "
    END_OBJECT              = COLUMN
```

B.7 GEO.FMT

```
COLUMNS                 = 30
ROW_BYTES               = 244

OBJECT                  = COLUMN
    NAME                = SCET
    DATA_TYPE           = LSB_INTEGER
    START_BYTE          = 1
    BYTES               = 4
    DESCRIPTION         = "SPACECRAFT EVENT TIME, ENCODED AS AN INT "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = SCET_FRACTIONAL_SECONDS
    DATA_TYPE           = LSB_INTEGER
    START_BYTE          = 5
    BYTES               = 4
    DESCRIPTION         = "FRACTIONAL PART OF SCET, IN 1/100 SECONDS "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = SCLK
    DATA_TYPE           = LSB_INTEGER
    START_BYTE          = 9
    BYTES               = 4
    DESCRIPTION         = "SPACECRAFT CLOCK, IN SCLK FORMAT "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = PARTITION
    DATA_TYPE           = LSB_INTEGER
    START_BYTE          = 13
    BYTES               = 4
    DESCRIPTION         = "SCLK PARTITION "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = TIME_ZPD
    DATA_TYPE           = LSB_INTEGER
    START_BYTE          = 17
    BYTES               = 4
    DESCRIPTION         = "TIME OF ZPD "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = BODY_ID
```

```
    DATA_TYPE            = LSB_INTEGER
    START_BYTE           = 21
    BYTES                = 4
    DESCRIPTION          = "NAIF BODY ID "
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = EPHEMERIS_TIME
    DATA_TYPE            = PC_REAL
    START_BYTE           = 25
    BYTES                = 8
    DESCRIPTION          = "EPHEMERIS TIME CORRESPONDING TO THIS SCET "
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = BODY_SPACECRAFT_RANGE
    DATA_TYPE            = PC_REAL
    START_BYTE           = 33
    BYTES                = 8
    DESCRIPTION          = "KM "
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = BODY_POSITION
    DATA_TYPE            = PC_REAL
    START_BYTE           = 41
    BYTES                = 24
    ITEMS                = 3
    ITEM_BYTES           = 8
    DESCRIPTION          = "KM "
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = BODY_SUB_SPACECRAFT_LONGITUDE
    DATA_TYPE            = PC_REAL
    START_BYTE           = 65
    BYTES                = 8
    DESCRIPTION          = "PLANETOGRAPHIC SUB SPACECRAFT LONGITUDE "
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = BODY_SUB_SPACECRAFT_LATITUDE
    DATA_TYPE            = PC_REAL
    START_BYTE           = 73
    BYTES                = 8
    DESCRIPTION          = "PLANETOGRAPHIC SUB SPACECRAFT LATITUDE "
END_OBJECT               = COLUMN
```

```
OBJECT                  = COLUMN
    NAME                = BODY_SUN_RANGE
    DATA_TYPE           = PC_REAL
    START_BYTE          = 81
    BYTES               = 8
    DESCRIPTION         = "KM "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = BODY_SUN_RIGHT_ASCENSION
    DATA_TYPE           = PC_REAL
    START_BYTE          = 89
    BYTES               = 8
    DESCRIPTION         = "DEGREES "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = BODY_SUN_DECLINATION
    DATA_TYPE           = PC_REAL
    START_BYTE          = 97
    BYTES               = 8
    DESCRIPTION         = "DEGREES "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = BODY_SUB_SOLAR_LONGITUDE
    DATA_TYPE           = PC_REAL
    START_BYTE          = 105
    BYTES               = 8
    DESCRIPTION         = "PLANETOGRAPHIC SUB SOLAR LONGITUDE "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = BODY_SUB_SOLAR_LATITUDE
    DATA_TYPE           = PC_REAL
    START_BYTE          = 113
    BYTES               = 8
    DESCRIPTION         = "PLANETOGRAPHIC SUB SOLAR LATITUDE "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = BODY_PHASE_ANGLE
    DATA_TYPE           = PC_REAL
    START_BYTE          = 121
    BYTES               = 8
    DESCRIPTION         = "PLANETARY PHASE ANGLE "
END_OBJECT              = COLUMN
```

```
OBJECT                  = COLUMN
    NAME                = BODY_ANGULAR_SEMIDIAMETER
    DATA_TYPE           = PC_REAL
    START_BYTE          = 129
    BYTES               = 8
    DESCRIPTION         = "MILLIRADIANS "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = BODY_ORBITAL_LONGITUDE
    DATA_TYPE           = PC_REAL
    START_BYTE          = 137
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = BODY_SYS3_LONGITUDE
    DATA_TYPE           = PC_REAL
    START_BYTE          = 145
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = PRIMARY_ID
    DATA_TYPE           = LSB_INTEGER
    START_BYTE          = 153
    BYTES               = 4
    DESCRIPTION         = "NAIF PRIMARY ID "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = PRIMARY_SPACECRAFT_RANGE
    DATA_TYPE           = PC_REAL
    START_BYTE          = 157
    BYTES               = 8
    DESCRIPTION         = "KM "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = PRIMARY_SUB_SPACECRAFT_LONGITUDE
    DATA_TYPE           = PC_REAL
    START_BYTE          = 165
    BYTES               = 8
    DESCRIPTION         = "PLANETOGRAPHIC SUB SPACECRAFT LONGITUDE "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = PRIMARY_SUB_SPACECRAFT_LATITUDE
```

```
    DATA_TYPE           = PC_REAL
    START_BYTE          = 173
    BYTES               = 8
    DESCRIPTION         = "PLANETOGRAPHIC SUB SPACECRAFT LATITUDE "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = PRIMARY_SUN_RANGE
    DATA_TYPE           = PC_REAL
    START_BYTE          = 181
    BYTES               = 8
    DESCRIPTION         = "KM "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = PRIMARY_SUB_SOLAR_LONGITUDE
    DATA_TYPE           = PC_REAL
    START_BYTE          = 189
    BYTES               = 8
    DESCRIPTION         = "PLANETOGRAPHIC SUB SOLAR LONGITUDE "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = PRIMARY_SUB_SOLAR_LATITUDE
    DATA_TYPE           = PC_REAL
    START_BYTE          = 197
    BYTES               = 8
    DESCRIPTION         = "PLANETOGRAPHIC SUB SOLAR LATITUDE "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = PRIMARY_PHASE_ANGLE
    DATA_TYPE           = PC_REAL
    START_BYTE          = 205
    BYTES               = 8
    DESCRIPTION         = "PLANETARY PHASE ANGLE "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = PRIMARY_ANGULAR_SEMIDIAMETER
    DATA_TYPE           = PC_REAL
    START_BYTE          = 213
    BYTES               = 8
    DESCRIPTION         = "MILLIRADIANS "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = SCET_STRING
```

```
    DATA_TYPE            = CHARACTER
    START_BYTE           = 221
    BYTES                = 24
    ITEMS                = 24
    ITEM_BYTES           = 1
END_OBJECT               = COLUMN
```

B.8 POI.FMT

```
COLUMNS                  = 25
ROW_BYTES                = 752

OBJECT                   = COLUMN
    NAME                 = TARGET_ID
    DATA_TYPE            = LSB_INTEGER
    START_BYTE           = 1
    BYTES                = 4
    DESCRIPTION          = "NAIF TARGET ID "
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = PRIMARY_ID
    DATA_TYPE            = LSB_INTEGER
    START_BYTE           = 5
    BYTES                = 4
    DESCRIPTION          = "NAIF ID OF THE PRIMARY "
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = SCET
    DATA_TYPE            = LSB_INTEGER
    START_BYTE           = 9
    BYTES                = 4
    DESCRIPTION          = "SPACECRAFT EVENT TIME ENCODED AS AN INTEGER "
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = TIME_ZPD
    DATA_TYPE            = LSB_INTEGER
    START_BYTE           = 13
    BYTES                = 4
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = TIME_END
    DATA_TYPE            = PC_REAL
```

```
    START_BYTE          = 17
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = DET
    DATA_TYPE           = LSB_INTEGER
    START_BYTE          = 25
    BYTES               = 4
    DESCRIPTION         = "DETECTOR ID "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = ALL_Q_ON
    DATA_TYPE           = LSB_INTEGER
    START_BYTE          = 29
    BYTES               = 4
    DESCRIPTION         = "BOOLEAN, 1 IF ALL 9 Q POINTS ARE ON THE
                           TARGET, 0 IF 1 OR MORE IS OFF "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = LATITUDE_ZPD
    DATA_TYPE           = PC_REAL
    START_BYTE          = 33
    BYTES               = 72
    ITEMS               = 9
    ITEM_BYTES          = 8
    DESCRIPTION         = "PLANETOGRAPHIC LATITUDE AT ZPG, 1 BAR LEVEL "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = LONGITUDE_ZPD
    DATA_TYPE           = PC_REAL
    START_BYTE          = 105
    BYTES               = 72
    ITEMS               = 9
    ITEM_BYTES          = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = ALTITUDE_ZPD
    DATA_TYPE           = PC_REAL
    START_BYTE          = 177
    BYTES               = 72
    ITEMS               = 9
    ITEM_BYTES          = 8
END_OBJECT              = COLUMN
```

```
OBJECT                  = COLUMN
    NAME                = RIGHT_ASCENSION
    DATA_TYPE           = PC_REAL
    START_BYTE          = 249
    BYTES               = 72
    ITEMS               = 9
    ITEM_BYTES          = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = DECLINATION
    DATA_TYPE           = PC_REAL
    START_BYTE          = 321
    BYTES               = 72
    ITEMS               = 9
    ITEM_BYTES          = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = SPACECRAFT_TO_IMAGE_POINT_ZPD
    DATA_TYPE           = PC_REAL
    START_BYTE          = 393
    BYTES               = 72
    ITEMS               = 9
    ITEM_BYTES          = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = LATITUDE_END
    DATA_TYPE           = PC_REAL
    START_BYTE          = 465
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = LONGITUDE_END
    DATA_TYPE           = PC_REAL
    START_BYTE          = 473
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = ALTITUDE_END
    DATA_TYPE           = PC_REAL
    START_BYTE          = 481
    BYTES               = 8
END_OBJECT              = COLUMN
```

```
OBJECT                  = COLUMN
    NAME                = SMEAR
    DATA_TYPE           = PC_REAL
    START_BYTE          = 489
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = EMISSION_ANGLE
    DATA_TYPE           = PC_REAL
    START_BYTE          = 497
    BYTES               = 72
    ITEMS               = 9
    ITEM_BYTES          = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = EMISSION_ANGLE_FOV_AVERAGE
    DATA_TYPE           = PC_REAL
    START_BYTE          = 569
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = FILLING_FACTOR
    DATA_TYPE           = PC_REAL
    START_BYTE          = 577
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = EMISSION_AZIMUTH_ANGLE
    DATA_TYPE           = PC_REAL
    START_BYTE          = 585
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = SOLAR_ZENITH
    DATA_TYPE           = PC_REAL
    START_BYTE          = 593
    BYTES               = 72
    ITEMS               = 9
    ITEM_BYTES          = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
```

```
        NAME                   = SOLAR_AZIMUTH
        DATA_TYPE              = PC_REAL
        START_BYTE             = 665
        BYTES                  = 8
    END_OBJECT                 = COLUMN

    OBJECT                     = COLUMN
        NAME                   = SOLAR_PHASE
        DATA_TYPE              = PC_REAL
        START_BYTE             = 673
        BYTES                  = 8
    END_OBJECT                 = COLUMN

    OBJECT                     = COLUMN
        NAME                   = LOCAL_TIME
        DATA_TYPE              = PC_REAL
        START_BYTE             = 681
        BYTES                  = 72
        ITEMS                  = 9
        ITEM_BYTES             = 8
    END_OBJECT                 = COLUMN
```

B.9 RIN.FMT

```
    COLUMNS                    = 18
    ROW_BYTES                  = 512

    OBJECT                     = COLUMN
        NAME                   = PRIMARY_ID
        DATA_TYPE              = LSB_INTEGER
        START_BYTE             = 1
        BYTES                  = 4
        DESCRIPTION            = "NAIF ID OF THE PRIMARY "
    END_OBJECT                 = COLUMN

    OBJECT                     = COLUMN
        NAME                   = SCET
        DATA_TYPE              = LSB_INTEGER
        START_BYTE             = 5
        BYTES                  = 4
        DESCRIPTION            = "SPACECRAFT EVENT TIME ENCODED AS AN INTEGER "
    END_OBJECT                 = COLUMN

    OBJECT                     = COLUMN
        NAME                   = TIME_ZPD
        DATA_TYPE              = LSB_INTEGER
```

```
    START_BYTE          = 9
    BYTES               = 4
    DESCRIPTION         = "TIME ZPD "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = TIME_END
    DATA_TYPE           = PC_REAL
    START_BYTE          = 13
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = DET
    DATA_TYPE           = LSB_INTEGER
    START_BYTE          = 21
    BYTES               = 4
    DESCRIPTION         = "DETECTOR ID "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = RING_RADIUS_ZPD
    DATA_TYPE           = PC_REAL
    START_BYTE          = 25
    BYTES               = 72
    ITEMS               = 9
    ITEM_BYTES          = 8
    DESCRIPTION         = "RADIUS OF INTERCEPT "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = RING_SPACECRAFT_RANGE_ZPD
    DATA_TYPE           = PC_REAL
    START_BYTE          = 97
    BYTES               = 72
    ITEMS               = 9
    ITEM_BYTES          = 8
    DESCRIPTION         = "DISTANCE OF SPACECRAFT FROM RING INTERCEPT "
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = RING_EMISSION_AZIMUTH_ANGLE
    DATA_TYPE           = PC_REAL
    START_BYTE          = 169
    BYTES               = 72
    ITEMS               = 9
    ITEM_BYTES          = 8
END_OBJECT              = COLUMN
```

```
OBJECT                  = COLUMN
    NAME                = RING_SOLAR_PHASE
    DATA_TYPE           = PC_REAL
    START_BYTE          = 241
    BYTES               = 72
    ITEMS               = 9
    ITEM_BYTES          = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = RING_EMISSION_ANGLE
    DATA_TYPE           = PC_REAL
    START_BYTE          = 313
    BYTES               = 72
    ITEMS               = 9
    ITEM_BYTES          = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = RING_LONGITUDE_ZPD
    DATA_TYPE           = PC_REAL
    START_BYTE          = 385
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = RING_SOLAR_AZIMUTH
    DATA_TYPE           = PC_REAL
    START_BYTE          = 393
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = RING_SOLAR_ZENITH
    DATA_TYPE           = PC_REAL
    START_BYTE          = 401
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = RING_LONGITUDE_END
    DATA_TYPE           = PC_REAL
    START_BYTE          = 409
    BYTES               = 8
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
```

```
    NAME                    = RING_RADIUS_END
    DATA_TYPE               = PC_REAL
    START_BYTE              = 417
    BYTES                   = 8
END_OBJECT                  = COLUMN

OBJECT                      = COLUMN
    NAME                    = RING_SPACECRAFT_RANGE_END
    DATA_TYPE               = PC_REAL
    START_BYTE              = 425
    BYTES                   = 8
END_OBJECT                  = COLUMN

OBJECT                      = COLUMN
    NAME                    = RING_SMEAR
    DATA_TYPE               = PC_REAL
    START_BYTE              = 433
    BYTES                   = 8
END_OBJECT                  = COLUMN

OBJECT                      = COLUMN
    NAME                    = RING_LOCAL_TIME
    DATA_TYPE               = PC_REAL
    START_BYTE              = 441
    BYTES                   = 72
    ITEMS                   = 9
    ITEM_BYTES              = 8
END_OBJECT                  = COLUMN
```

B.10 TAR.FMT

```
COLUMNS                     = 31
ROW_BYTES                   = 40

OBJECT                      = COLUMN
    NAME                    = SCET
    DATA_TYPE               = LSB_INTEGER
    START_BYTE              = 1
    BYTES                   = 4
    DESCRIPTION             = "spacecraft event time encoded as an integer "
END_OBJECT                  = COLUMN

OBJECT                      = COLUMN
    NAME                    = DET
    DATA_TYPE               = LSB_INTEGER
    START_BYTE              = 5
```

```
    BYTES               = 4
    DESCRIPTION         = "detector ID "
  END_OBJECT            = COLUMN

  OBJECT                = COLUMN
    NAME                = FOV_TARGETS
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 9
    BYTES               = 4
    DESCRIPTION         = "A bitfield, with the following assignments:

                  2^0  Jupiter ring
                  2^1  Jupiter
                  2^2  Io
                  2^3  Europa
                  2^4  Ganymede
                  2^5  Callisto
                  2^6  Saturn
                  2^7  Mimas
                  2^8  Enceladus
                  2^9  Tethys
                  2^10 Dione
                  2^11 Rhea
                  2^12 Titan
                  2^13 Hyperion
                  2^14 Iapetus
                  2^15 Phoebe
                  2^16 Janus
                  2^17 Epimetheus
                  2^18 Helene
                  2^19 Telesto
                  2^20 Calypso
                  2^21 Atlas
                  2^22 Prometheus
                  2^23 Pandora
                  2^24 Pan
                  2^25 Saturn ring
                  2^31 A stellar target from the "stars" file
If the corresponding bit is on, the object is in the field of view,
otherwise it isn't. "
  END_OBJECT            = COLUMN

  OBJECT                = COLUMN
    NAME                = JRING
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 13
    BYTES               = 1
  END_OBJECT            = COLUMN
```

```
OBJECT                  = COLUMN
    NAME                = JUPITER
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 14
    BYTES               = 1
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = IO
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 15
    BYTES               = 1
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = EUROPA
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 16
    BYTES               = 1
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = GANYMEDE
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 17
    BYTES               = 1
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = CALLISTO
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 18
    BYTES               = 1
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = SATURN
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 19
    BYTES               = 1
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = MIMAS
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 20
    BYTES               = 1
```

```
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = ENCELADUS
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 21
    BYTES               = 1
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = TETHYS
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 22
    BYTES               = 1
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = DIONE
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 23
    BYTES               = 1
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = RHEA
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 24
    BYTES               = 1
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = TITAN
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 25
    BYTES               = 1
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = HYPERION
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 26
    BYTES               = 1
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = IAPETUS
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 27
```

```
    BYTES               = 1
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = PHOEBE
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 28
    BYTES               = 1
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = JANUS
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 29
    BYTES               = 1
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = EPHMETHEUS
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 30
    BYTES               = 1
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = HELENE
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 31
    BYTES               = 1
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = TELESTO
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 32
    BYTES               = 1
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = CALYPSO
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
    START_BYTE          = 33
    BYTES               = 1
END_OBJECT              = COLUMN

OBJECT                  = COLUMN
    NAME                = ATLAS
    DATA_TYPE           = LSB_UNSIGNED_INTEGER
```

```
    START_BYTE           = 34
    BYTES                = 1
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = PROMETHEUS
    DATA_TYPE            = LSB_UNSIGNED_INTEGER
    START_BYTE           = 35
    BYTES                = 1
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = PANDORA
    DATA_TYPE            = LSB_UNSIGNED_INTEGER
    START_BYTE           = 36
    BYTES                = 1
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = PAN
    DATA_TYPE            = LSB_UNSIGNED_INTEGER
    START_BYTE           = 37
    BYTES                = 1
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = SRING
    DATA_TYPE            = LSB_UNSIGNED_INTEGER
    START_BYTE           = 38
    BYTES                = 1
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = DEEP_SPACE
    DATA_TYPE            = LSB_UNSIGNED_INTEGER
    START_BYTE           = 39
    BYTES                = 1
END_OBJECT               = COLUMN

OBJECT                   = COLUMN
    NAME                 = STELLAR
    DATA_TYPE            = LSB_UNSIGNED_INTEGER
    START_BYTE           = 40
    BYTES                = 1
END_OBJECT               = COLUMN
```

B.11 ISPM.FMT

```
COLUMNS                   = 16
ROW_BYTES                 = 53

OBJECT                    = COLUMN
    NAME                  = SCET
    DATA_TYPE             = LSB_UNSIGNED_INTEGER
    START_BYTE            = 1
    BYTES                 = 4
    DESCRIPTION           = "spacecraft clock time "
END_OBJECT                = COLUMN

OBJECT                    = COLUMN
    NAME                  = DET
    DATA_TYPE             = LSB_INTEGER
    START_BYTE            = 5
    BYTES                 = 1
    DESCRIPTION           = "detector number "
END_OBJECT                = COLUMN

OBJECT                    = COLUMN
    NAME                  = ISPTS
    DATA_TYPE             = LSB_INTEGER
    START_BYTE            = 6
    BYTES                 = 2
    DESCRIPTION           = "number of points in regridded spectrum"
END_OBJECT                = COLUMN

OBJECT                    = COLUMN
    NAME                  = DS_NAVE
    DATA_TYPE             = LSB_INTEGER
    START_BYTE            = 8
    BYTES                 = 2
    DESCRIPTION           = "number of spectra in deep space average"
END_OBJECT                = COLUMN

OBJECT                    = COLUMN
    NAME                  = SH_NAVE
    DATA_TYPE             = LSB_INTEGER
    START_BYTE            = 10
    BYTES                 = 2
    DESCRIPTION           = "number of spectra in shutter average"
END_OBJECT                = COLUMN

OBJECT                    = COLUMN
    NAME                  = TINSTR
```

```
        DATA_TYPE           = PC_REAL
        START_BYTE          = 12
        BYTES               = 4
        DESCRIPTION         = "instrument temperature in calibration (K)"
    END_OBJECT              = COLUMN

    OBJECT                  = COLUMN
        NAME                = IWN_START
        DATA_TYPE           = PC_REAL
        START_BYTE          = 16
        BYTES               = 4
        DESCRIPTION         = "wavenumber of first regridded spectral
point (cm-1)"
    END_OBJECT              = COLUMN

    OBJECT                  = COLUMN
        NAME                = IWN_STEP
        DATA_TYPE           = PC_REAL
        START_BYTE          = 20
        BYTES               = 4
        DESCRIPTION         = "wavenumber step size of regridded data (cm-1)"
    END_OBJECT              = COLUMN

    OBJECT                  = COLUMN
        NAME                = APODTYPE
        DATA_TYPE           = LSB_INTEGER
        START_BYTE          = 24
        BYTES               = 2
        DESCRIPTION         = "numeric type of apodisation function"
    END_OBJECT              = COLUMN

    OBJECT                  = COLUMN
        NAME                = FWHM
        DATA_TYPE           = PC_REAL
        START_BYTE          = 26
        BYTES               = 4
        DESCRIPTION         = "FWHM of instrument line shape (cm-1)"
    END_OBJECT              = COLUMN

    OBJECT                  = COLUMN
        NAME                = RAYLEIGH
        DATA_TYPE           = PC_REAL
        START_BYTE          = 30
        BYTES               = 4
        DESCRIPTION         = "Rayleigh resol.
(= dist to first null of ILS) in cm-1"
    END_OBJECT              = COLUMN
```

```
    OBJECT                    = COLUMN
        NAME                  = NYQUIST
        DATA_TYPE             = PC_REAL
        START_BYTE            = 34
        BYTES                 = 4
        DESCRIPTION           = "Nyquist bin size (intrinsic resol. limit)
in cm-1"
    END_OBJECT                = COLUMN

    OBJECT                    = COLUMN
        NAME                  = POWER
        DATA_TYPE             = PC_REAL
        START_BYTE            = 38
        BYTES                 = 4
        DESCRIPTION           = "integrated radiance under the spectrum
in W cm-2 sr-1"
    END_OBJECT                = COLUMN

    OBJECT                    = COLUMN
        NAME                  = DS_SCET
        DATA_TYPE             = LSB_UNSIGNED_INTEGER
        START_BYTE            = 42
        BYTES                 = 4
        DESCRIPTION           = "Deep Space SCET "
    END_OBJECT                = COLUMN

    OBJECT                    = COLUMN
        NAME                  = DS_SH_SCET
        DATA_TYPE             = LSB_UNSIGNED_INTEGER
        START_BYTE            = 46
        BYTES                 = 4
        DESCRIPTION           = "Deep Space Shutter Closed SCET "
    END_OBJECT                = COLUMN

    OBJECT                    = COLUMN
        NAME                  = ISPM
        DATA_TYPE             = LSB_INTEGER
        START_BYTE            = 50
        BYTES                 = 4
        VAR_DATA_TYPE         = PC_REAL
        VAR_ITEM_BYTES        = 4
        VAR_RECORD_TYPE       = VAX_VARIABLE_LENGTH
        DESCRIPTION           = "pointer to regridded spectra data. Spectra data
has units W cm-2 sr-1 (cm-1)-1"
    END_OBJECT                = COLUMN
```