SPECTRAL DATA BASE HEADER DEFINITION—VERSION 6

Mark Strickman, NRL

May 22, 1998

NRL Document 0926-159
Rev. 2.00-1

Contents

1 Introduction

The Spectral Database format is a format for the storage of OSSE (or other) spectra in a such a manner that all relevant information required to analyze the data is present with the spectrum. The data and related information are organized and stored in a random access file on disk. Data are located through keyed access. This document concerns itself primarily with the format of the auxiliary data, known as the SDB header. It also discusses the requirements for utility software for SDB manipulation.

The version 6 header has substantial changes when compared to previous versions. These changes are detailed in the sections following. In particular, the key structure has been changed substantially.

As in previous versions, a fundamental subset of the header items shall be identified which will be required for every spectral data set. The most fundamental subset (kernel) of the SDB header items is that which is absolutely required for SDB utility programs to read the data. An example of a kernel item is the size of the spectrum. Kernel items shall always be at the top of the header.

Another major change in SDB format from version 5 to version 6 is the ability to store auxiliary data in addition to spectra and uncertainties under a single header. Such auxiliary data might include channel edge energies, channel widths, channel representative energies and/or other calibration information.

Version 6 of SDB also has the capability to support other than Real*4 data, with the data type indicated in the header.

A number of smaller changes to the header, mostly aimed at streamlining, have been made. In particular, references to detectors other than the detector producing the data have been eliminated. Much of the calibration data has been moved out of the header and into auxiliary arrays as described below.

2 Overall Design Considerations

The following overall SDB design considerations influence header structure and contents. Many of these are new to version 6 of SDB.

2.1 Data Types

In previous versions of SDB (version 5, version E4B and earlier), the data consisted of spectra from a single detector (but possibly multiple spectral memory ranges) and, optionally, uncertainties. Data were written in records unrelated to the number of channels available. The only structure requirements were that data structure and channel numbers be identical to those for uncertainties and that data and uncertainties each start at the beginning of a record. Storage types were guaranteed to be identical since all data were stored R$*$4. Storage information such as record size and numbers of channels were stored in the header record, which was fixed in size and structure.

Version 6 of SDB largely preserves this scheme, but adds the capability to include data arrays other than the primary data under a single header. The primary goal is to optionally include calibration or other information with each spectrum ("local" storage) or to place this information under a single header referred to by some or all spectra in a given file ("remote" storage). The key structure defines a set of auxiliary data types, each of which has its own key space. In this manner, it is possible to index directly into the desired auxiliary type knowing only its type and tagword.

Some of the data types are always (or usually) associated with spectra on a one-to-one basis, whereas others refer to a large number of spectra. The former are usually accessed with the spectra they accompany, while the latter are usually accessed as a group. To make the access of the latter type faster, two classes of auxiliary data type have been defined. Those accompanying spectra are referred to as "dense", while those not accompanying spectra are called "sparse". Sparse auxiliary data types are always pointed to by a tagword pointer in the SDB header. In this fashion, a spectrum carries with it pointers to all the sparse information that is relevant to it.

Structurally, sparse data types are identified in a different part of the key than dense data types so that the sparse types can be more easily accessed in sequential mode.

The data types, including an indication of whether they are sparse or dense, are as follows:

Type # Array Data Type
(Bit #)

0. [D]Spectrum data
1. [D]Spectrum statistical uncertainties/+ statistical uncertainties
2. [D]Spectrum—statistical uncertainties
3. [D]Spectrum systematic uncertainties/+ systematic uncertainties
4. [D]Spectrum—systematic uncertainties
5. [S]Multi-spectrum data
6. [S]Multi-spectrum statistical uncertainties/+ statistical uncertainties
7. [S]Multi-spectrum—statistical uncertainties
8. [S]Multi-spectrum systematic uncertainties/+ systematic uncertainties
9. [S]Multi-spectrum—systematic uncertainties
10. [S]Detector resolution (FWHM(energy units)/E(energy units) vs. channel)
11. [S]Channel lower edge energies
12. [S]Channel lower edge energy uncertainties
13. [S]Channel widths
14. [S]Channel width uncertainties
15. [S]Channel representative energies

16. [S]Channel representative energy uncertainties
17. [S]Calibration line information
18. [S]PSD information
19. [S]Analyzable unit definition
20. [S]Orbit history
21. [S]Associated spectra
22. [D]Footnote (textual comment)
23. [S]Source list
24. [D]Estimated background uncertainties
25. [D]Estimated background
26. [D]Analyzable unit identification
27. unassigned
28. [D]Pulsar window rates
29. [D]TSF window rates
30. [D]Reserved for Monte Carlo DB
31. [D]Reserved for Monte Carlo DB
32. [S]Instrument response matrix
33. [S]Photopeak response vector
34. [S]Selection record (truncated header)
35. [S]Selected spectra list
36. unassigned
37. [D]Compound response coefficients

The maximum number of auxiliary data types allowed shall be 64, as limited by the descriptive bit picks in the header. None of the auxiliary types shall be required and the presence of each is technically independent of the presence of any of the others.

As in earlier versions, data from each detector shall be included under a separate header, with the exception of data summed together from multiple detectors. Low, Medium, and High range spectra shall be treated as a single spectrum (i.e. all three spectra are concatenated). Spectra other than spectral memory low, medium, and high spectra shall be stored as a low spectrum with all associated header variables stored in low spectrum positions in the header.

In the case of an entry with no spectrum data, the header data shall be undefined with the exception of the kernel information, start and stop times, units, live times and comment.

In addition to support of various data types included with each spectrum, referred to as "local" storage of data, the header shall support pointers to data common to a number of spectra, referred to as "remote" storage of data. These data shall be stored as an SDB entry in the same file with any allowed tagword. They shall be stored with the above data type numbers. In general, data types stored locally (i.e. under the same header) shall take precedence over data types stored remotely.

The auxiliary data type record lengths shall be described in a header at the beginning of each auxiliary type. The header key value shall have a record sequence number of zero in each case. The first 20 bytes of the header shall be the same for all data types, consisting of the following:

Description	Name	Type	# bytes
File key	FILE_KEY	Byte	8
Size of header	SIZEHDR	Integer	2
# bytes/record	NBYTES_PER_RECORD	Integer	2
# records	NUMREC	Integer	2
# of entities	NUMENT	Integer	2
# bytes/entity	SIZEENT	Integer	2
Data type version	DATAVERSION	Integer	2

This structure shall apply to the first 20 bytes of the main SDB header (data type = 1) as well as all other data types.

An entity is defined as a fundamental unit of the data type. In the case of data types defined by arrays (e.g. channel edge energies, uncertainties, etc.) the entity is the array element (usually congruent with a spectrum channel) and the number of entities is equal to the number of elements. String arrays are a special case in that the size of each string must also be specified. In the case of data types defined by structures (e.g. calibration line information), the entity is the structure, and the number of entities is the number of structure records corresponding to that structure (e.g. the number of calibration lines). The multi-spectrum data types, which consist of spectrum arrays embedded in structures, are treated as structures in this respect. The number of entities is the number of structures which, in this case, is also the number of spectra (see below for a description of multi-spectrum data types).

In general, a physical file record as defined in the record size shall contain some whole number of entities. This number may be determined from the auxiliary data type header by dividing the number of entities by the number of records and truncating if necessary. Alternatively, it may be necessary to have some entities that span multiple physical records (e.g. large response matrices). Applications will know how to interpret each auxiliary data record by knowing which type it is reading.

When writing auxiliary data types, the normal mode of operation is for the application to pass the number of entities and the entity size. The write utility then computes a reasonable record size and number of records. In a few cases (multi-spectrum and response matrices), where the user may want direct control over record size, the user can specify the record size, number of records and number of entities and the write utility will comply if possible. In general, entities always begin on a record boundary. Since each auxiliary data type may have a different associated data structure, there shall be no requirements placed on the form of the data in the various types. In particular, they do not have to be the same as the primary data (e.g. all integer∗2, etc.).

2.2 Data Type Descriptions

2.2.1 Spectrum Data and Uncertainties (0–4)

Entity type: i*2, i*4, r*4 (first 2 data only as specified in SDB header)

A spectrum can include any combination of the three OSSE spectrum ranges concatenated and treated as a single spectrum.

Spectrum +/− uncertainties refer to cases where uncertainties are asymmetric about the best estimate data point. In the case where uncertainties are symmetric, the − uncertainties array shall not be present and the bit pick indicating present arrays shall be zero for − uncertainties.

2.2.2 Multi-Spectrum Data and Uncertainties (5–9)

Entity type: Structure (includes spectrum array)

The multi-spectrum data type is similar to the spectrum data type except that it supports more than one spectrum (a spectrum, as before, is defined as any combination of the three OSSE spectrum ranges concatenated together) under a single SDB header. The data type consists of an SDB header kernel (top 20 bytes), some additional header information appended to the kernel, and a set of structures, each of which includes a small amount of information specific to that spectrum, and the spectrum itself. The SDB header kernel is identical to that in the spectrum data type, except that the variables referring to data type entities refer to the number of spectrum structures rather than to the number of array elements in the single spectrum.

2.2.2.1 Header

The multi-spectrum header contains the following information:

Description	Name	Type	# bytes
File key	FILE_KEY	Byte	8
Size of header	SIZEHDR	Integer	2
# bytes/record	NBYTES_PER_RECORD	Integer	2
# records	NUMREC	Integer	2
# of entities	NUMENT	Integer	2
# bytes/entity	SIZEENT	Integer	2
Data type version	DATAVERSION	Integer	2
Number of channels	NCHAN	Integer	2
Number of low channels	NCHANL	Integer	2
Number of med. channels	NCHANM	Integer	2
Number of high channels	NCHANH	Integer	2
Accumulation time	ACCUMTIM	Real	4
Seconds/TSAC	TSAC	Real	4
Spectrum flags	SPECFLGS	Bits	2

For multi-spectrum data, the number of records and the number of entities (or spectra), shall be required to be equal. In other words, each spectrum shall occupy one record.

The multi-spectrum data type shall optionally be combined with a standard SDB header and a summed spectrum under a single tagword. The standard SDB header associated with the summed spectrum will describe averages over the summing interval. Quantities such as SPECSRC, CALR-PHA, SPECUNIT and others that describe the contents of the spectra shall be valid for both the summed spectrum, if present, and the multi-spectra present under that tagword.

Variables in the multi-spectrum header shall apply to all the multi-spectrum records (i.e. spectra) stored under that header.

2.2.2.2 Structure Description

Description	Name	Type	# bytes
Start time	STRTTIM	Integer	4
Start time fraction	STRT_FRAC	Integer	2
Spec. mem. low live time/ single spectrum live time	LTL	Real	4
Spec. mem. med. live time	LTM	Real	4
Spec. mem. high live time	LTH	Real	4
Low mem. load rate (4/TSAC)	MLL(4)	Integer	8
Med. mem. load rate (4/TSAC)	MLM(4)	Integer	8
High mem. load rate (4/TSAC)	MLH(4)	Integer	8
Shield 1 LLD rate (8/TSAC)	S1LD(8)	Integer	16
Shield 2 LLD rate (8/TSAC)	S2LD(8)	Integer	16
Shield 3 LLD rate (8/TSAC)	S3LD(8)	Integer	16
Shield 4 LLD rate (8/TSAC)	S4LD(8)	Integer	16
Low dead time rate	DTL	Integer	2
Med. dead time rate	DTM	Integer	2
High dead time rate	DTH	Integer	2
Spectrum	SPEC_ARRAY		

2.2.2.3 Variable Descriptions

The structure variables are all defined in the same fashion as the same variables in the SDB header. Since they are generally single TSAC quantities, the rate variables do not have the _SUM suffix as in the SDB header.

If the multi-spectrum header variables ACCUMTIM and TSAC indicate that more than one TSAC/spectrum accumulation interval is in use, the memory load and shield rates, which are normally collected 4/TSAC and 8/TSAC respectively, will be collected 4 or 8 times per accumulation interval.

The spectrum variable shall contain a variable number of elements and be either I*2, I*4 or R*4 as specified by the SPECFLGS in the multi-spectrum header.

2.2.3 Channel Calibration Data (10–16)

Entity type: r*4

Channel calibration data types describe various calibration-derived results (e.g. channel widths) on a channel-by-channel basis. In general, the spectrum ranges and channels specified in the SDB header also describe the channel calibration data available.

2.2.3.1 Detector Resolution (10)

This is the relative spectrum point spread function (expressed as a ratio of the FWHM in energy units of a line caused by incident photons at a given energy to the incident energy). The resolution on a channel-by-channel basis is just this quantity for the representative energy for each channel.

2.2.3.1.1 Header

The header for detector resolution contains the 20-byte kernel plus the following:

Description	Name	Type	# bytes
Detector ID	DETID	Bits	2
Resolution model ID	RESMODID	Character	8
Number of model parameters	NRESMODPARAM	Integer	2
Resolution model parameters	RESMODPARAM	Real(10)	40

2.2.3.2 Variable Descriptions

DETID Bits 2 bytes

Detector ID as described in SDB header section.

RESMODID Character 8 bytes

Identifier of model function used to describe FWHM/E vs. E. A typical example might be:

FWHM/E = (A + B/E)**0.5

The identifier is broken down as follows:

MMMMMMVV

where MMMMMM is the actual model ID and VV is a major/minor version number.

NRESMODPARAM Integer 2 bytes

The number of parameters in the resolution model (maximum 10).

RESMODPARAM	Real(10)	40 bytes

The parameters (at most 10) of the resolution model (FWHM/E vs. E) function.

2.2.3.3 Channel Lower Edge Energies and Uncertainties (11–12)

These represent the energy in MeV of the lower edge of each spectrum channel, under the assumption that the channel edge is well-defined. Overlapping channels can be handled by having the lower edge energy plus the width be greater than the next lower edge energy, while gaps can be handled by having the sum be less than the next lower edge energy.

The first element of the edge energy array contains a TBD code indicating the units of the edge energies. As mentioned above, the default is MeV (code=0).

2.2.3.4 Channel Widths and Uncertainties (13–14)

These represent the extent of the channel in energy space expressed in MeV. Channels may be asymmetric, but this is handled by a single width and a representative energy, which may not be centered in the channel. This allows representative energies derived from integration of a model over the channel width to be used.

2.2.3.5 Channel Representative Energies and Uncertainties (15–16)

Channel representative energies (expressed in MeV) are the mean energies for each channel, determined by integrating the energy over the channel weighted by some assumed spectrum. The simplest assumption is that the spectrum is constant over each channel, leading to a representative energy that is the arithmetic mean of the lower and upper edge energies (the arithmetic "center" of the channel). Spectrum manipulation routines that do not integrate over each channel typically assume the average channel flux to be the flux at the representative energy.

2.2.4 Calibration Line Information (17)

Entity Type: Structure

The calibration line information contains data pertaining to gamma ray lines used either in ground based calibration, in-flight calibration (i.e. ^{60}Co), or useful background lines.

2.2.4.1 Structure Description

Description	Name	Type	# bytes
Spectrum Range	RANGE	Integer	2
Official 0. keV channel	OFFSET	Real	4
Centroid	CENTROID	Real	4
Width	WIDTH	Real	4
Error	ERROR	Real	4
Energy	ENERGY	Real	4
Calibration line description	CALDESC	Character	6
Calibration line model ID	CALMODEL	Character	20
Calibration line fit c^2	CALCHI2	Real	4
Calibration line fit number of degrees of freedom	CALNDOF	Integer	2

2.2.4.2 Variable Descriptions

Descriptions of these variables are as follows:

RANGE Integer 2 bytes

Bit pick (bit 0 = low, bit 1 = medium, bit 2 = high) indicating spectral range from which calibration line information was derived.

OFFSET Real 4 bytes

Channel representative of 0 keV (for spectral range specified in RANGE).

CENTROID Real 4 bytes

Fitted centroid (in channels out of number appropriate to RANGE) of calibration line.

WIDTH Real 4 bytes

Fitted full width at half maximum (in channels defined by RANGE) of calibration line.

ERROR Real 4 bytes

Uncertainty (one sigma) in fitted centroid of calibration line in same channel units as CENTROID.

ENERGY	Real	4 bytes

Energy (in MeV) of calibration line center.

CALDESC	Character*6	6 bytes

6 character descriptor of calibration line. Suggested format is EEAAAS where EE is the element, AAA the atomic weight of the isotope and S is a source code (A=atmosphere, D=detector, S=spacecraft, etc.).

CALMODEL	Character*20	20 bytes

Identifies what spectrum model was used to fit calibration line. Compatible with model ID used in FDB and FDR.

CALCHI2	Real*4	4 bytes

Chi-squared per degree of freedom (reduced chi-squared) of calibration line fit.

CALNDOF	Integer*2	2 bytes

Number of degrees of freedom of calibration line fit.

2.2.5 PSD Information (18)

Entity Type: Structure

PSD information contains a structure record for each energy band used by the pulse shape discrimination system. There are sixteen bands defined for each spectral range. For each band, the structure contains information on the digital time window (DTW), the NaI peak position, width and model type, and the estimated acceptance efficiency for NaI events.

2.2.5.1 Structure Description

Description	Name	Type	# bytes
Range	RANGE	Byte	1
DTW energy band #	DTWBAND	Byte	1
Window low channel	DTWLO	Byte	1
Window high channel	DTWHI	Byte	1
NaI peak fit energy band low	FITBANDLO	Byte	1
NaI peak fit energy band high	FITBANDHI	Byte	1
NaI peak centroid channel	NAICENTROID	Real	4
NaI peak width	NAIWIDTH	Real	4
NaI peak model code	NAIMODEL	Integer	2
NaI acceptance efficiency	NAIEFFIC	Real	4
NaI acceptance efficiency uncertainty	NAIEFFICSIG	Real	4

2.2.5.2 Variable Descriptions

Descriptions of these variables are as follows:

RANGE Byte 1 byte

Bit pick of spectrum range (bit 0 = low, bit 1 = medium, bit 2 = high).

DTWBAND Byte 1 byte

Number 0–15 indicating which band within RANGE the data refer to.

DTWLO Byte 1 byte

PSD spectrum channel representing the low edge of the digital time window.

DTWHI Byte 1 byte

PSD spectrum channel representing the high edge of the digital time window.

FITBANDLO	Byte	1 byte

Low channel of energy band over which PSD spectrum was integrated for NaI time peak fit, in channels (0–255). For DTW bands, this variable will always correspond to the hardware DTW band.

FITBANDHI	Byte	1 byte

High channel of energy band over which PSD spectrum was integrated for NaI time peak fit, in channels (0–255). For DTW bands, this variable will always correspond to the hardware DTW band.

NAICENTROID	Real	4 bytes

PSD spectrum channel of the centroid of the NaI peak, in channels (0–255). "Centroid" in this case refers to the peak position parameter of the peak model.

NAIWIDTH	Real	4 bytes

Width of the NaI peak in PSD spectrum channels (0–255). "Width" in this case refers to the peak width parameter of the peak model.

NAIMODEL	Integer	2 bytes

A code pointing to a table of NaI peak models and versions of models.

NAIEFFIC	Real	4 bytes

The probability that a true NaI event would fall within the DTW range for this band. This is derived by integrating the best fit NaI peak model between the DTW boundaries. Value is in range 0.0–1.0.

2.2.6 Analyzable Unit Definition (19)

Entity Type: Structure

Background interpolation will be accomplished via the combination of various background elements associated with a single source. The minimum set of source and background points that can be analyzed is referred to as an analyzable unit.

The analyzable unit definition specifies how many of each background pointing position measurements are required in order to create a background estimate. Typically, each background estimation model requires its own analyzable unit definition.

The analyzable unit definition structure contains information that allows analyzable units to be identified in the data stream. One analyzable unit definition structure record is all that is required for a given analyzable unit definition.

2.2.6.1 Structure Description

Description	Name	Type	# bytes
Source entry identifiers	SRC_ENTRY(0:7,4)	Logical	32
Bkgd entry identifiers	BKG_ENTRY(0:7,4)	Logical	32
Active det. pointing table	DPTBL	Byte	1
The maximum (optimum) number of points for each pos'n	MAX_PTS_PER_POS(0:7,4)	Integer	64
The minimum allowable number of points for each pos'n	MIN_PTS_PER_POS(0:7,4)	Integer	64
The minimum num. bkgds to one side or other of source	MINBKG(0:7,4)	Integer	64
Max. time betw. source and bkgd	SRC_DT(4)	Real	16
Max. time betw. adjacent bkgds at given position	BKG_DT(0:7,4)	Real	128
Search direction flag	SEARCH_FLAG(0:7,4)	Byte	32
Bkgd estimation model ID	MODEL_TYPE	Character	20

2.2.6.2 Variable Descriptions

Descriptions of these variables are as follows:

SRC_ENTRY Logical(0:7,4) 32 bytes

Logical flag indicating the detector positioning table indices to be treated as possible source observations. One element for each positioning table entry (0:7) and detector (4). TRUE indicates that that positioning table entry is to be used as the source for the AU. Note that for any given AU, only a single positioning table index will be allowed, i.e. only one of the 0:7 positioning table entries will be TRUE for each of the 4 detectors.

BKG_ENTRY Logical(0:7,4) 32 bytes

Logical flag indicating the detector positioning table entries to be treated as background observations.

DPTBL Byte 1 byte

Flag indicating the active detector positioning table(s) to use within the AU. Values are specified by a bit code as described for the SDB header variable DPTBL. Valid detector positioning tables will be a bitwise OR of the DPTBL entries. The bit code definitions are:

Bit 0 Primary Table
1 Secondary Table
2 Solar Table
3 Transient Table

All cleared: No code assigned

MAX_PTS_PER_POS Integer(0:7,4) 64 bytes

The optimum (maximum) total number of background observations in the AU for the corresponding detector positioning table entry and detector.

MIN_PTS_PER_POS Integer(0:7,4) 64 bytes

The minimum number of background observations in the AU for the corresponding positioning table entry and detector. Note that this parameter also indicates the maximum order for which time interpolation may be performed.

MINBKG Integer(0:7,4) 64 bytes

The minimum number of background observations before or after the source observation in the AU for the corresponding positioning table entry and detector, i.e. controls how close the source observation may be to the edge of the AU. Note that this parameter allows the support of a sliding window for the background observations.

SRC_DT Real(4) 16 bytes

The maximum time in seconds between the source and all of the background accumulations.

BKG_DT Real(0:7,4) 28 bytes

The maximum time in seconds between adjacent background accumulations for this positioning table entry and detector. In the user interface which generates the AU definition record, this parameter will be specified in units of strategy table cycles and converted into seconds for storage in the AU definition record.

SEARCH_FLAG	Byte(0:7,4)	32 bytes

Flag indicating background observation search method. Currently defined flags are:

Code: 1 Begin search forward in time (DEFAULT)
2 Begin search backward in time
3 Alternate initial search direction

MODEL_TYPE	Character	20 bytes

A character description of the model type to be used in estimating the background. The description will consist of a three character code describing the order and method of estimation (e.g. "2CT1PD" would indicate that the estimation should first be 2nd order Chebychev polynomial in Time and then 1st order Polynomial in DSA.

2.2.7 Orbit History (20)

Entity Type: Structure

One structure record shall correspond to one orbit. Typically, one orbit history record array should serve an entire SDB file, in a fashion similar to the calibration data. The SDB header shall have a tagword pointer to the orbit history tagword for the file.

2.2.7.1 Structure Description

Description	Name	Type	# bytes
Orbit number	ORBIT	Integer	4
Orbit start time	ORBITSTART	Integer	4
Orbit stop time	ORBITSTOP	Integer	4
Orbit integration time	ORBITINTEG	Real	4
SAA entry time	SAAENTRY	Integer	4
SAA exit time	SAAEXIT	Integer	4
SAA integration time	SAAINTEG	Real	4
CPM low flux (orbit)	CPML	Real	4
CPM high flux (orbit)	CPMH	Real	4
CPM low dose (SAA)	CPML_SAA	Real	4
CPM high dose (SAA)	CPMH_SAA	Real	4

2.2.7.2 Variable Descriptions

Descriptions of these variables are as follows:

ORBITSTART Integer 4 bytes

PB5 time of the start of the orbit integration. Accurate to the nearest 16 seconds.

ORBITSTOP Integer 4 bytes

PB5 time (to nearest second) of the end of the orbit integration. Accurate to the nearest 16 seconds.

ORBITINTEG Real 4 bytes

Orbit integration time in seconds.

SAAENTRY Integer 4 bytes

PB5 time (to nearest second) of the mode 4 to mode 3 change marking the entry into the SAA.

SAAEXIT Integer 4 bytes

PB5 time (to nearest second) of the mode 3 to mode 4 change marking the exit from the SAA.

SAAINTEG Real 4 bytes

SAA integration time in seconds.

CPML Real 4 bytes

Average CPM low (actually minimum ionizing) flux (1 MeV $<$ E $<$ 8 MeV) in counts per second over the orbit integration period.

CPMH Real 4 bytes

Average CPM high flux (E $>$ 8 MeV) in counts per second over the orbit integration period.

CPML_SAA Real 4 bytes

Summed CPM low dose in counts during SAA period as defined by SAAENTRY and SAAEXIT above.

CPMH_SAA Real 4 bytes

Summed CPM high dose in counts during SAA period as defined by SAAENTRY and SAAEXIT above.

2.2.8 Associated Spectra (21)

Entity Type: Structure

Each structure record contains a single filespec and tagword along with a short comment. The user may use these records to associate other spectra with any given spectrum.

2.2.8.1 Structure Description

Description	Name	Type	# bytes
File specification	FILESPEC	Character	40
Tagword	TAGWORD	Integer	4
Comment	COMMENT	Character	40

2.2.8.2 Variable Descriptions

Descriptions of these variables are as follows:

FILESPEC Character 40 bytes

Specification of file containing the associated spectrum.

TAGWORD Integer 4 bytes

Tagword of the associated spectrum within the specified file.

COMMENT Character 40 bytes

Comment relevant to the associated spectrum.

2.2.9 Footnote (22)

Entity Type: c*80

Each footnote record contains a single 80-character string containing a text comment.

2.2.10 Source List (23)

Entity Type: structure

A list of the source ID's for a given observation analysis, including the target(s) and confusing sources being considered. The ID's are referred to in the response aux. type. Also included are the source RA and DEC and the collimator position angles (scan and transscan) for each possible pointing position. Positions are usually included for the pointing positions used in the AU definition only. Any pointing positions not used in the AU definition are not included. A maximum of eight positions (potentially from different tables) are allowed.

2.2.10.1 Header

The header includes the 20-byte kernel plus the DETID (I*2) variable.

2.2.10.2 Structure Description

Description	Name	Type	# bytes
Source ID in file	SRCID	Integer	4
Catalog ID (if available) (TARGID)	SRCCATID	Integer	4
Source name	SRCNAME	Character	12
Source RA	SRCRA	Real	4
Source Declination	SRCDC	Real	4
Number of positions	NPOSN	Integer	2
Pointing strategy tbl TW	POINTTBLTW	Integer	4
Active Position Table	DPTBL(8)	Byte	32
Pointing status	FDET(8)	Byte	32
Source scan angle	TARGSCAN(8)	Real	128
Source transscan angle	TARGXSCAN(8)	Real	128

2.2.10.3 Variable Descriptions

Descriptions of these variables are as follows:

SRCID Integer 4 bytes
Index number of source within this file. This is the number used in the target position and response aux. types to refer to the source. In general, this should be made equal to the TARGID used in the catalogs, but this equality is not required.

SRCCATID Integer 4 bytes
This is the TARGID value for the source (if assigned). This value is useful for identifying the source in the catalog database.

SRCNAME	Character	12 bytes

Source name as stored in the catalog database.

SRCRA	Real	4 bytes

Right Ascension (epoch 2000) of the source in degrees.

SRCDC	Real	4 bytes

Declination (epoch 2000) of the source in degrees.

NPOSN	Integer	2 bytes

Number of pointing positions (both source and background) in this analyzable unit definition. This determines how many of the remaining variable array elements are meaningful.

DPTBL	Byte(8)	8 bytes

Identifies pointing table (primary, secondary, solar, burst) referenced by FDET. If two tables have an identical entry, such that this response could refer to either, either table ID could appear here, as long as the combination of FDET and DPTBL identifies the proper pointing parameters.

FDET	Byte(8)	8 bytes

Pointing status byte. Primarily used here to identify entry within pointing table identified in DPTBL.

TARGSCAN	Real(8)	32 bytes

Scan angle of source for each pointing position. The array is indexed by actual pointing position rather than pointing strategy table index. In other words, only pointing positions that are actually used by the AU definition are included in the array. Only the first NPOSN elements of the array are meaningful. The order of the pointing positions in the array is source first, then backgrounds in increasing detector step angle order.

TARGXSCAN	Real(8)	32 bytes

Transscan angle of source for each pointing position. See TARGSCAN.

2.2.11 Estimated Background Uncertainties (24)

Entity Type: array

Similar to the primary (type 0) spectrum data but contains the associated estimated background uncertainties. Header is minimum 20 bytes.

2.2.12 Estimated Background (25)

Entity Type: array

Similar to the primary (type 0) spectrum data but contains the associated estimated background. Header is minimum 20 bytes.

2.2.13 Analyzable Unit Identification (26)

Entity Type: structure

This structure array identifies the spectra associated with a single analyzable unit. For each spectrum, a file and tag pointer and a source/background identifier are included. In addition, the background estimation coefficient Aji is also included for each spectrum. Aji is the coefficient for each background pointing position of the analyzable unit in the linear combination of background measurements that form the background estimate:

$$\text{Bestim i} = \text{Sj Aji Bji}$$

where the index i identifies the analyzable unit and j identifies the pointing position within the analyzable unit. In the simple case of averaged backgrounds symmetric in step angle about the source, the Aji's are just 0.5.

The structure records are stored in the array in a definite order. First comes the source spectrum record. Only one source spectrum is allowed in any given analyzable unit. Next come the background pointing positions sorted first in ascending detector step angle order, then, within each position, in ascending time order. For example, in the case of two times and two background positions:

record	1:	tsrc ,	dsasrc
	2:	t1 bkg1,	dsabkg1
	3:	t2 bkg1,	dsabkg1
	4:	t1 bkg2,	dsabkg2
	5:	t2 bkg2,	dsabkg2

2.2.13.1 Header

The header includes the 20-byte kernel plus the DETID (I*2) variable and the AUDEFTW (I*4) variable that points to the tagword containing the AUDEF aux. data type (type 19).

2.2.13.2 Structure Description

Description	Name	Type	# bytes
File specification	FILESPEC	Character	20
Tagword	TAGWORD	Integer	4
Aji	AJI	Real	4
Positioning table	DPTBL	Integer	1
Detector position status	FDET	Bits	1
Earth center Rt Ascension	EARTHRA	Real	4
Earth center Declination	EARTHDC	Real	4
+ Z Axis RA 2000 epoch	ATTZRA	Real	4
+ Z Axis Dec 2000 epoch	ATTZDC	Real	4
+ X Axis RA 2000 epoch	ATTXRA	Real	4
+ X Axis Dec 2000 epoch	ATTXDC	Real	4
Detector position	DSA	Integer	2

2.2.13.3 Variable Descriptions

Descriptions of these variables are as follows:

FILESPEC Character 20 bytes
Specification of file containing the spectrum.

TAGWORD Integer 4 bytes
Tagword of the spectrum within the specified file.

AJI Real 4 bytes
Background estimate coefficient (Aji) for this pointing position. In the case of multiple records for a given pointing position, the records shall all contain the same value of Aji for that position j.

DPTBL Integer 1 byte
Active positioning table.

FDET Bits 1 byte
Detector positioning status variable. Defined as in SDB header. Applies to spectrum referred to in record.

EARTHRA Real 4 bytes
Right Ascension (epoch 2000) of the earth center as viewed from the spacecraft.

EARTHDC Real 4 bytes
Declination (epoch 2000) of the earth as viewed from the spacecraft.

ATTZRA	Real	4 bytes

+Z Axis RA, epoch 2000, in units of degrees.

ATTZDC	Real	4 bytes

+Z Axis declination, epoch 2000, in units of degrees.

ATTXRA	Real	4 bytes

+X Axis RA, epoch 2000, in units of degrees.

ATTXDC	Real	4 bytes

+X Axis declination, epoch 2000, in units of degrees.

DSA	Integer	2 bytes

Detector step angle.

2.2.14 Unassigned (27)

2.2.15 Pulsar Window Rates (28)

Entity Type: array

The pulsar energy windows are defined in terms of PHA channels in the pulsar strategy table. This auxiliary data type contains the rates summed from the associated spectral memory spectrum over the channels that define the pulsar windows. These are stored as summed counts rather than true rates.

2.2.15.1 Header

The header includes the 20-byte kernel plus the following variables:

Description	Name	Type	# bytes
Lowest window channel	PWINLOCHAN(8)	Integer	16
Highest window channel	PWINHICHAN(8)	Integer	16

Descriptions of these variables are as follows:

PWINLOCHAN Integer(8) 16 bytes

The lowest channel of each of eight possible pulsar windows, numbered from 0–543.

PWINHICHAN Integer(8) 16 bytes

The highest channel of each of eight possible pulsar windows, numbered from 0–543.

2.2.16 TSF Window Rates (29)

Entity Type: array

The Telemetry Scalar File (TSF) energy windows are defined in terms of PHA channels in the command state. This auxiliary data type contains the rates summed from the associated spectral memory spectrum over the channels that define the TSF windows. These are stored as summed counts rather than true rates.

2.2.16.1 Header

The header includes the 20-byte kernel plus the following variables:

Description	Name	Type	# bytes
Lowest window channel	TWINLOCHAN(16)	Integer	32
Highest window channel	TWINHICHAN(16)	Integer	32

Descriptions of these variables are as follows:

TWINLOCHAN Integer(16) 32 bytes
The lowest channel of each of sixteen possible TSF windows, numbered from 0–543.

PWINHICHAN Integer(16) 32 bytes
The highest channel of each of sixteen possible TSF windows, numbered from 0–543.

2.2.17 Instrument Response Matrix (32)

Entity Type: special

The response matrix auxiliary data type holds the instrument response information required to remove the instrument signature from spectra. The interface to the Monte Carlo database from which the responses are derived produces a response matrix for a given set of incident spectrum channels and PHA spectrum channels, for a given veto configuration and instrument resolution function, and for a given incident beam angle. These are referred to as "simple" responses. The processes of estimating background and assessing the effects of sources other than the target ("confusing" sources) require a number of simple responses to be linearly combined into a "compound" response. The compound response represents the response of the instrument to the weighted sum of incident beams from a number of angles. This data type will store both simple and compound response matrices.

Each response under a single tagword (i.e. all responses in the array below) must have the same number of photon channels and the same number of PHA channels. In addition, they must all correspond to the same photon channel and PHA channel energy definitions. These values are stored in the header along with the source ID, which is also required to be the same for all responses under one tagword.

As noted in the entity type above, the structure of the response matrix aux. type is special. This is required by the variable length requirements of the arrays. The header stores the number of rows (NPHACHAN) and number of columns (NPHOTCHAN) for the response matrix. The channel energy and width vectors are of length NPHACHAN for the PHA channel definitions and NPHOTCHAN for the photon channel definitions.

The response aux. data type is different from most in that the first four records are distinct from the rest of the data. They contain the photon channel edge energies, the photon channel edge widths, the PHA channel edge energies and the PHA channel widths. The first two are of size NPHOTCHAN and the last two are of size NPHACHAN. These records will always use record sequence numbers 1–4.

The remainder of the data will be arranged using the record and entity information in the header. One entity will correspond to one response matrix. The variables DPTBL, FDET, SRCSCAN and SRCXSCAN will appear as the first 10 bytes of each response.

2.2.17.1 Header

The header includes the 20-byte kernel plus the following:

Description	Name	Type	# bytes
Detector ID	DETID	Integer	2
Source identification	SRCID	Integer	4
Shield threshold (MeV)	SLD	Real	4
Number of photon channels	NPHOTCHAN	Integer	2
Number of PHA channels	NPHACHAN	Integer	2

2.2.17.2 Data Description

Description	Name	Type	# bytes
Record 1:			
Photon channel edges (MeV)	PHOTEDG	Real	4*NPHOTCHAN
Record 2:			
Photon channel widths (MeV)	PHOTWID	Real	4*NPHOTCHAN
Record 3:			
PHA channel edges (MeV)	PHAEDG	Real	4*NPHACHAN
Record 4:			
PHA channel widths (MeV)	PHAWID	Real	4*NPHACHAN
Records 5–n:			
Active Position Table	DPTBL	Byte	1
Pointing status	FDET	Byte	1
Source scan angle	SRCSCAN	Real	4
Source transscan angle	SRCXSCAN	Real	4
Response (NPHACHAN,NPHOTCHAN) matrices	RESPONSE	Real	4*NPHOTCHAN*NPHACHAN

2.2.17.3 Variable Descriptions

Descriptions of these variables are as follows:

PHOTEDG Real 4*NPHOTCHAN

The lower edge energies of the channels used to store the model incident photon spectrum, expressed in MeV. The response matrix contains probabilities for photons in these channels to appear in the PHA channels of the data.

PHOTWID Real 4*NPHOTCHAN

The widths of the channels used to store the model incident photon spectrum, expressed in MeV.

PHAEDG Real 4*NPHACHAN

The lower edge energies of the PHA channels for the count data, expressed in MeV. These may be the actual PHA channels from the instrument or resampled channels. The response matrix contains probabilities for photons in the photon channels to appear in these PHA channels.

PHAWID Real 4*NPHACHAN

The widths of the channels used to store the count spectrum, expressed in MeV.

DPTBL Byte 1 byte

Identifies pointing table (primary, secondary, solar, burst) referenced by FDET. If two tables have an identical entry, such that this response could refer to either, either table ID could appear here, as long as the combination of FDET and DPTBL identifies the proper pointing parameters.

FDET	Byte	1 byte

Pointing status byte. Primarily used here to identify entry within pointing table identified in DPTBL.

SRCSCAN	Real	4 bytes

Scan angle of given source, in degrees. Same coordinate system as TARGSCAN in SDB spectrum header.

SRCXSCAN	Real	4 bytes

Transscan angle of given source, in degrees. Same coordinate system as TARGXSCAN in SDB spectrum header.

RESPONSE	Real	4∗NPHOTCHAN∗NPHACHAN

The response matrix gives the probability of an incident photon with energy in one of the NPHOTCHAN photon spectrum channels appearing as an event in the detector with pulse height (expressed as an energy) in one of the NPHACHAN PHA spectrum channels. The probability is expressed as an effective detector area in cm^2. Each element of the matrix contains the response of one PHA channel to incident photons of one energy. All photons in the photon spectrum channel are approximated to have the same channel representative energy. This representative energy is computed from the channel edges and widths by the response generation application and the applications that use the response. It is important that they use the same algorithm to calculate a representative energy.

2.2.18 Instrument Photopeak Response Vector (33)

Entity Type: special

The photopeak response vector is similar to the response matrix, but consists of only the probabilities for an incident photon of a given energy to produce a pulse height at that same energy, i.e. full energy deposition in the detector. The data structure is similar to the response matrix except that only the first two records contain channel energy information and each photopeak response, with scalar information prepended as in the response matrix, only occupies a single record.

Unlike the response matrix, the photopeak responses do not include instrument energy resolution. In other words, the probability is for occurrence of the event anywhere in the "photopeak" of the PHA spectrum. Since the photopeak response is derived from the Monte Carlo simulation, it represents only "true" photopeak photons, i.e. those that are fully absorbed in the detector. If the photopeak response were derived from the integral over the photopeak in the calibration data, for example, it would also include a contribution from events that were almost fully absorbed and not resolvable from the true photopeak at instrument resolution.

The photopeak response corresponds to the PHA channels. The units of the response are generally cm^2.

2.2.18.1 Header

The header includes the 20-byte kernel plus the following:

Description	Name	Type	# bytes
Detector ID	DETID	Integer	2
Source identification	SRCID	Integer	4
Shield threshold (MeV)	SLD	Real	4
Number of PHA channels	NPHACHAN	Integer	2

2.2.18.2 Data Description

Description	Name	Type	# bytes
Record 1:			
PHA channel edges (MeV)	PHAEDG	Real	4∗NPHACHAN
Record 2:			
PHA channel widths (MeV)	PHAWID	Real	4∗NPHACHAN
Records 3–n:			
Active Position Table	DPTBL	Byte	1
Pointing status	FDET	Byte	1
Source scan angle	SRCSCAN	Real	4
Source transscan angle	SRCXSCAN	Real	4
Response (NPHACHAN) vector	PPKRESPONSE	Real	4∗NPHACHAN

2.2.18.3 Variable Descriptions

Descriptions of these variables are as follows:

PHAEDG	Real	4*NPHACHAN

The lower edge energies of the PHA channels for the count data, expressed in MeV. These may be the actual PHA channels from the instrument or resampled channels. The response matrix contains probabilities for photons in the photon channels to appear in these PHA channels.

PHAWID	Real	4*NPHACHAN

The widths of the channels used to store the count spectrum, expressed in MeV.

DPTBL	Byte	1 byte

Identifies pointing table (primary, secondary, solar, burst) referenced by FDET. If two tables have an identical entry, such that this response could refer to either, either table ID could appear here, as long as the combination of FDET and DPTBL identifies the proper pointing parameters.

FDET	Byte	1 byte

Pointing status byte. Primarily used here to identify entry within pointing table identified in DPTBL.

SRCSCAN	Real	4 bytes

Scan angle of given source, in degrees. Same coordinate system as TARGSCAN in SDB spectrum header.

SRCXSCAN	Real	4 bytes

Transscan angle of given source, in degrees. Same coordinate system as TARGXSCAN in SDB spectrum header.

PPKRESPONSE	Real	4*NPHACHAN

See above for description.

2.2.19 Selection Record (34)

Entity Type: structure

The selection record (formerly called the truncated header or shrunken header) is a subset of the SDB header used for analyzable unit identification and data selection. These are stored as a structure array, one selection record per record.

2.2.19.1 Data Description

Name	Type	# bytes
TargID	Integer	4
Targoff	Integer	2
DetID	Byte	2
SpecSrc	Byte	1
CalRPHA	Byte	1
TlmID	Byte	1
DSFStat	Integer	4
DataQual	Integer	4
Reactor	Byte	1
SpecCat	Byte	1
Low_Width	Real	4
StrtTim	Integer	4
StopTim	Integer	4
Orbit	Integer	4
AccumTim	Real	4
LTL	Real	4
LTM	Real	4
LTH	Real	4
DTL	Real	4
DTM	Real	4
DTH	Real	4
DetRA	Real	4
DetDc	Real	4
EarthSep	Real	4
EarthAz	Real	4
DSA	Integer	2
FDet	Byte	1
DPTbl	Byte	1
Cmd_State_ID_1	Character	8
C6Tmp	Real	4
MLL_Sum	Real	4
MLM_Sum	Real	4
MLH_Sum	Real	4
S1LD_Sum	Real	4
S2LD_Sum	Real	4

S3LD_Sum	Real	4
S4LD_Sum	Real	4
SMD_Sum	Real	4
SUD_Sum	Real	4
CPD_Sum	Real	4
CPML_Rate	Real	4
CPMH_Rate	Real	4
Rigid	Real	4
MCL	Real	4
MCB	Real	4
GROLT	Real	4
GROLG	Real	4
GRORA	Real	4
GRODc	Real	4
GRORad	Real	4
This_OrbitInteg	Real	4
CPML_Frac_Orbit	Real	4
CPMH_Frac_Orbit	Real	4
Last_SAA_Exit	Real	4
Last_SAAInteg	Real	4
CPML_Last_SAA	Real	4
CPMH_Last_SAA	Real	4

2.2.20 Selected Spectra List (35)

Entity Type: structure

The selected spectra list includes pointers to all the spectra selected by a data selection criterion. The criterion or a pointer to the criterion is stored in a character variable in the header.

To avoid redundant storage of file specifications, an array of up to 100 file specs is stored in the header. Each entry in the data contains a tagword and an index into the file spec array.

2.2.20.1 Header

The header contains the 20-byte kernel plus the following variables:

Description	Name	Type	# bytes
Selection criterion	SELCRIT	Character	160
File specifications	FILESPEC(100)	Character	2000

2.2.20.2 Structure Description

Description	Name	Type	# bytes
File spec. array index	FILESPECIDX	Integer	2
Tagword	TAGWORD	Integer	4

2.2.20.3 Variable Descriptions

Descriptions of these variables are as follows:

FILESPECIDX Integer 2 bytes

Points to entry in FILESPEC array in header containing file specification for selected spectrum.

TAGWORD Integer 4 bytes

Tagword of selected spectrum.

2.2.21 Unassigned (36)

2.2.22 Compound Response Coefficients (37)

Entity Type: Structure

The compound response coefficients are the quantities required to produce a compound response matrix from component simple response matrices. Each simple response matrix is computed for a single source direction in the collimator. The compound response matrix takes into account all sources visible at any time. They include the summed background estimation coefficients, the summed visibility coefficients for the target pointing and the summed visibility coefficient-background coefficient products for background pointings.

The summed background coefficients are the coefficients of the weighted summation of observed backgrounds that produce the background estimate:

Best i = Sj AjiBji

where the Bji are background observations within AU i. After background subtraction, when the difference spectra are summed, the deconvolution equation requires the term Si Aji for each background pointing position j used in the AU. These are the summed background estimation coefficients.

These are stored in the same tagword as the summed data. One record can store up to eight summed coefficients (for up to eight pointing positions), but only actually stores the pointing positions used in the AU definition. These are stored indexed by pointing table index, with the indices not used in the AU's marked by zero values.

2.2.22.1 Header

The header includes the 20-byte kernel plus the DETID (I*2) variable.

2.2.22.2 Structure Description

Description	Name	Type	# bytes
Active Position Table	DPTBL	Byte	1
Pointing table index	PTBLINDX(8)	Byte	8
Summed Aji's	SUMAJI(8)	Real	32
Summed ak(tsi) for ea. source k	SUMALPHA(100)	Real	400
Summed ak(tji)Aji for each	SUMALPHAAJI(100,8)	Real	3200
Total number of AU's summed	NUMAU	Integer	4

2.2.22.3 Variable Descriptions

Descriptions of these variables are as follows:

DPTBL	Byte(8)	8 bytes

Identifies pointing table (primary, secondary, solar, burst) referenced by FDET. If two tables have an identical entry, such that this response could refer to either, either table ID could appear here, as long as the combination of FDET and DPTBL identifies the proper pointing parameters.

FDET	Byte(8)	8 bytes

Pointing status byte. Primarily used here to identify entry within pointing table identified in DPTBL.

SUMAJI	Real(8)	32 bytes

Background estimate coefficient (Aji) summed over analyzable units i for each pointing position. Sum over i includes only analyzable units picked by data selection for inclusion in summed difference spectra.

SUMALPHA	Real(100)	00 bytes

Sum over the visibility coefficients ak(tsi) for each source k (maximum 100). These coefficients are equal to 1 if source k is visible during source pointing time tsi (the start time of the source pointing for AU i) and 0 if source k is not visible. For k representing the target (typically k=1), the visibility coefficient is always 1. The array is indexed (k) for a maximum of 100 sources k.

SUMALPHAAJI	Real(8,100)	00 bytes

Sum over the products of visibility ak(tji) for source k at background pointing time t (AU i) and the background estimation coefficient Aji. Array is indexed (j,k) with a maximum of 8 pointing positions j and 100 sources k.

NUMAU	Integer	4 bytes

Total number of AU's summed together.

3 The SDB Header

3.1 Header Structure Overview

A summary of the header variables is included in this section. Detailed descriptions are in the sections that follow.

Header Section	Size (bytes)
KERNEL	54
OBSERVATION DESCRIPTION	67
TIMES AND COLLECTION INTERVALS	40
DETECTOR POINTING	64
TIME SERIES SPECIFIC CONTENTS	10
DETECTOR PARAMETERS	212
RATES	108
ENVIRONMENT	181
REMOTE TAGWORD POINTERS	20
USER DATA	176
MISCELLANEOUS (RESERVE 88)	92
TOTAL	1024

3.1.1 HEADER KERNEL

VARIABLE	NAME	DATA TYPE	BYTES
UNION			
MAP			
File key	FILE_KEY	Real	8
END MAP			
MAP			
Record sequence number	REC_SEQ	Byte	2
Array Data Type	ARRAY_TYPE	Byte	1
Spare key space	SPAREKEY	Integer	1
Tagword	TAGWORD	Integer	4
END MAP			
END UNION			
Size of header	SIZEHDR	Integer	2
# bytes/record	NBYTES_PER_RECORD	Integer	2
# records	NUMREC	Integer	2
# of entities	NUMENT	Integer	2
# bytes/entity	SIZEENT	Integer	2
Data type version	DATAVERSION	Integer	2
Header version number	HDRVERS	Integer	2
Header revision number	HDRREV	Byte	1
# data arrays	NUMARRAY	Byte	1
Array contents flags (1st 32 data types)	CONTFLGS1	Bits	4
Array contents flags (2nd 32 data types)	CONTFLGS2	Bits	4
Spectrum flags (real/integer; I2/I4; SDB/TSDB; Environment/No environment; Autodeglitch/No autodeglitch; Visual deglitch/No visual deglitch; Glitch found/No glitch found; Gain normalized/Not normalized; Deletion flag)	SPECFLGS	Bits	2
Creation software index	SOFTINDEX	Integer	4
Creation s/w major vers.	SOFTMAJOR	Integer	4
Creation s/w minor vers.	SOFTMINOR	Integer	4
Creation date and time VAX format	CREDATE	Real*8	8
Total Header Kernel			54

3.1.2 OBSERVATION DESCRIPTION

VARIABLE	NAME	DATA TYPE	BYTES
Target Name Code	TARGID	Integer	4
Instrument Config. ID	INST_CNFG_ID	Bits	2
Flight ID Code	FLTID	Integer	2
Detector ID	DETID	Bit	2
Spectrum Source Code	SPECSRC	Byte	1
CALPHA/RPHA Mux	CALRPHA	Byte	1
Telemetry format ID	TLMID	Byte	1
DSF stop/event status code	DSFSTAT	Bit	4
Data Quality Flags	DATAQUAL	Bit	4
Reactor event code	REACTOR	Bit	1
Acquisition Hdr	ACHD(25)	Byte	25
Spectral Category Code	SPECCAT	Byte	1
Source/Background Offset	TARGOFF	Byte	1
Units Code	SPECUNIT	Integer	2
Total # of spectrum channels	NCHAN	Integer	2
# of low range channels	NCHANL	Integer	2
# of med. range channels	NCHANM	Integer	2
# of high range channels	NCHANH	Integer	2
Number of ch.s, param. 1	NCHAN1	Integer	2
Number of ch.s, param. 2	NCHAN2	Integer	2
Phase Bin No.	PHBINNUM	Integer	2
Total No. of Phase Bins	NUMPHBIN	Integer	2
Total Observation Description			67

3.1.3 TIMES AND COLLECTION INTERVALS

VARIABLE	NAME	DATA TYPE	BYTES
Start Time	STRTTIM	Integer	4
Start Time Fraction	STRT_FRAC	Integer	2
Stop Time	STOPTIM	Integer	4
Stop Time Fraction	STOP_FRAC	Integer	2
Orbit Number	ORBIT	Integer	4
Time of Slot Header Packet	HSKPTIM	Integer	4
Seconds/TSAC	TSAC	Real	4
Accumulation Time	ACCUMTIM	Real	4
Spec. mem. low live time/ single spectrum live time	LTL	Real	4
Spec. mem. med. live time	LTM	Real	4
Spec. mem. high live time	LTH	Real	4
Total Times and Collection Intervals			40

3.1.4 DETECTOR POINTING

VARIABLE	NAME	DATA TYPE	BYTES
UNION			
MAP			
Detector RA epoch 2000	DETRA	Real	4
Detector decl epoch 2000	DETDC	Real	4
Mean scan pointing uncertainty	DETSIGMA_SCAN	Real	4
Mean transscan pointing uncertainty	DETSIGMA_XSCAN	Real	4
Target RA 2000 epoch	TARGRA	Real	4
Target Dec 2000 epoch	TARGDC	Real	4
Target Latitude (scan angle)	TARGSCAN	Real	4
Target Longitude (transscan)	TARGXSCAN	Real	4
Collimator Position Angle	COLLPOSANG	Real	4
Earth position in FOV (separation)	EARTHSEP(2)	Real	8
Earth position in FOV (rotation angle)	EARTHAZ(2)	Real	8
END MAP			
MAP			
Detector pointing	DETPOINT(13)	Real	52
END MAP			
END UNION			
Detector position, end	DSA	Integer	2
Average Pot Main	AM	Real	4
Average Pot Redn	AR	Real	4
Detector Posit. Status	FDET	Bits	1
Active Position Table	DPTBL	Byte	1
Total Detector Pointing			64

3.1.5 TIME SERIES SPECIFIC CONTENTS

VARIABLE	NAME	DATA TYPE	BYTES
Interval duration	INTDURAT	Integer	2
Interval livetime	INTLIVTIM	Integer	2
Interval duration units code	INTDURUNITS	Integer	2
Number of quantities per interval	NOINTQUANT	Integer	2
Number of parameters per interval	NOINTPARAM	Integer	2
Total Time Series			10

3.1.6 DETECTOR PARAMETERS

VARIABLE	NAME	DATA TYPE	BYTES
Command state table ID, start	CMD_STATE_ID_1	Character	8
Command state table ID, end	CMD_STATE_ID_2	Character	8
Detector config. table ID, start	DET_CNFG_ID_1	Integer	4
Detector config. table ID, end	DET_CNFG_ID_2	Integer	4
Bilevels 0–4 (Start of acquisition)	STRTBILVL(5)	Bits	10
Bilevels 0–4 (End of acquisition)	ENDBILVL(5)	Bits	10
Shield 1 LLD Thresh.	S1LD	Real	4
Shield 2 LLD Thresh.	S2LD	Real	4
Shield 3 LLD Thresh.	S3LD	Real	4
Shield 4 LLD Thresh.	S4LD	Real	4
^{60}Co LLD Thresh.	C6LD	Real	4
CPD LLD Thresh.	CPD	Real	4
Low Energy LLD Thresh.	ELDL	Real	4
Low Time LLD Thresh.	TLDL	Real	4
Low Time ULD Thresh.	TUDL	Real	4
Medium Energy LLD Thresh.	ELDM	Real	4
Medium Time LLD Thresh.	TLDM	Real	4
Medium Time ULD Thresh.	TUDM	Real	4
High Energy LLD Thresh.	ELDH	Real	4
High Time LLD Thresh.	TLDH	Real	4
High Time ULD Thresh.	TUDH	Real	4
Cal. PHA LLD Thresh.	CLLD	Real	4
HV/DAC PMT 1	HVP1	Real	4
HV/DAC PMT 2	HVP2	Real	4
HV/DAC PMT 3	HVP3	Real	4
HV/DAC PMT 4	HVP4	Real	4
HV/DAC PMT 5	HVP5	Real	4
HV/DAC PMT 6	HVP6	Real	4
HV/DAC PMT 7	HVP7	Real	4
Shield 1 HV	HVS1	Real	4
Shield 2 HV	HVS2	Real	4
Shield 3 HV	HVS3	Real	4
Shield 4 HV	HVS4	Real	4
^{60}Co/CPD HV	HVC6	Real	4
Pin Diode Thresh.	PIN	Real	4
LED Level	LED	Real	4
PMT Mount Plate Temp.	PTMP1	Real	4
Mag. Shield –Z Temp.	PTMP2	Real	4

Elect. A1 Temp.	ETMP1	Real	4
Elect. A4 Temp.	ETMP2	Real	4
Xtal Housing Cover Temp.	S1TMP	Real	4
Mag. Shield Out. Temp.	S2TMP	Real	4
Coll./^{60}Co Temp.	C6TMP	Real	4
CPD Temp.	CPDTM	Real	4
High Voltage Supply Temp.	HVTMP	Real	4
Low Voltage Supply Temp.	LVTMP	Real	4
Outboard Spider Temp.	TCTMP	Real	4
A3 Temp.	A3TMP	Real	4
Total Detector Parameters			212

3.1.7 RATES

VARIABLE	NAME	DATA TYPE	BYTES
Rate collection time	RATE_TIM	Real	4
Low energy			
LLD Energy Rate	ELDL_SUM	Real	4
ULD Energy Rate	EUDL_SUM	Real	4
Analog Window Rate	ATWL_SUM	Real	4
ULD Time Rate	TUDL_SUM	Real	4
Memory Load Rate	MLL_SUM	Real	4
SML Dead Time	DTL_SUM	Real	4
Medium energy			
LLD Energy Rate	ELDM_SUM	Real	4
ULD Energy Rate	EUDM_SUM	Real	4
Analog Window Rate	ATWM_SUM	Real	4
ULD Time Rate	TUDM_SUM	Real	4
Memory Load Rate	MLM_SUM	Real	4
SMM Dead Time	DTM_SUM	Real	4
High energy			
LLD Energy Rate	ELDH_SUM	Real	4
ULD Energy Rate	EUDH_SUM	Real	4
Analog Window Rate	ATWH_SUM	Real	4
ULD Time Rate	TUDH_SUM	Real	4
Memory Load Rate	MLH_SUM	Real	4
SMH Dead Time	DTH_SUM	Real	4
Shield 1 LLD Rate	S1LD_SUM	Real	4
Shield 2 LLD Rate	S2LD_SUM	Real	4
Shield 3 LLD Rate	S3LD_SUM	Real	4
Shield 4 LLD Rate	S4LD_SUM	Real	4
Shield MLD Rate	SMD_SUM	Real	4
Shield ULD Rate	SUD_SUM	Real	4
^{60}Co LLD Rate	C6LD_SUM	Real	4
CPD LLD Rate	CPD_SUM	Real	4
Total Rates			108

3.1.8 ENVIRONMENT

VARIABLE	NAME	DATA TYPE	BYTES
ARRAY(START TIME,STOP TIME)			
X Coord. of s/c earth cent inert. epoch 2000	GROX(2)	Real	8
Y Coord. of s/c earth cent inert. epoch 2000	GROY(2)	Real	8
Z Coord. of s/c earth cent inert. epoch 2000	GROZ(2)	Real	8
Latitude of sub-s/c point	GROLT(2)	Real	8
Longitude of sub-s/c point	GROLG(2)	Real	8
RA of s/c in earth cent inert. epoch 2000	GRORA(2)	Real	8
Decl. of s/c in earth cent inert. epoch 2000	GRODC(2)	Real	8
Altitude of s/c above earth cent inert.	GRORAD(2)	Real	8
+ Z Axis RA 2000 epoch	ATTZRA(2)	Real	8
+ Z Axis Dec 2000 epoch	ATTZDC(2)	Real	8
+ X Axis RA 2000 epoch	ATTXRA(2)	Real	8
+ X Axis Dec 2000 epoch	ATTXDC(2)	Real	8
McIlwain L	MCL(2)	Real	8
McIlwain B	MCB(2)	Real	8
Rigidity	RIGID(2)	Real	8
Radial Component of Mag. Field Vector	BRAD(2)	Real	8
Polar Angle Component of Mag. Field Vector	BPOL(2)	Real	8
Azimuthal Angle Component of Mag. Field Vector	BAZ(2)	Real	8
MEAN VALUE:			
Day/Night Code	DAY_NIGHT	Byte	1
CPM low rate	CPML_RATE	Real	4
CPM high rate	CPMH_RATE	Real	4
SAA INFO. FOR CURRENT ORBIT			
Integration time from start of orbit (seconds)	THIS_ORBITINTEG	Real	4
CPM low dose since start of orbit (counts)	CPML_FRAC_ORBIT	Real	4
CPM high dose since start of orbit (counts)	CPMH_FRAC_ORBIT	Real	4
Time since last SAA exit (seconds)	LAST_SAA_EXIT	Real	4

CPM integration time for last SAA (seconds)	LAST_SAAINTEG	Real	4
CPM low dose during last SAA	CPML_LAST_SAA	Real	4
CPM high dose during last SAA	CPMH_LAST_SAA	Real	4
Total Environment			181

3.1.9 REMOTE TAGWORD POINTERS

VARIABLE	NAME	DATA TYPE	BYTES
Calib Tag Word	CALTW	Integer	4
Standard Calib Tag Word	STDCALTW	Integer	4
Orbit History Tag Word	ORBITTW	Integer	4
Associate Spectra Tag Word	ASSOCSPECTW	Integer	4
Footnote Tag Word	FOOTNOTETW	Integer	4
Total Remote Tagword Pointers			20

3.1.10 USER DATA

VARIABLE	NAME	DATA TYPE	BYTES
User spectrum screening flags	USRSCRNFLG	Bits	4
Effective area	EFFECAREA	Real	4
Divisor	DIVISOR	Real	4
Gain shift	GSHIFT	Real	4
Comment	COMMENT	Character	160
Total User Data			176

3.1.11 MISCELLANEOUS

VARIABLE	NAME	DATA TYPE	BYTES
Low range channel width, keV/Channel (use (1332 – 1173)/ (High ^{60}Co line ch.−Low ^{60}Co line ch.))	LOW_WIDTH	Real	4
Reserve	RESERVE	Integer	88

3.2 Header Variable Descriptions

The following descriptions include the variable summary, a description and any applicable codes or bit assignments.

Several of the SDB header variables are codes that must be defined externally. A few of these coded variables involve bit assignments (i.e., each bit is assigned an independent meaning), while the rest are numerical codes or mixed bit and numerical.

The definitions of code values are stored in an .ICL (`CM_DEV:[OSSECM.ICL]SDB_HEADER_CODES.ICL`) file that may be compiled with code. The code structure is described in the comments in the .ICL file.

3.2.1 HEADER KERNEL

3.2.1.1 FILE_KEY

```
UNION
MAP
   FILE_KEY      Real     8 bytes
END MAP
MAP
   REC_SEQ       Byte     2 bytes
   ARRAY_TYPE    Byte     1 byte
   SPAREKEY      Integer  1 byte
   TAGWORD       Integer  4 bytes
END MAP
END UNION
```

The primary key is composed of 8 bytes divided into four fields. The uppermost four bytes are the spectrum tagword, which uniquely identifies the spectrum within the file and also may be encoded such as to do spectrum bookkeeping within a file. The next two bytes are spare key space reserved for future use and currently zeroed. The lowest two bytes are the array type and record sequence number. The array type specifies which of the (32 maximum) array types (or the header) the record contains. It is the same for all records of that array. The array type is zero for the main SDB header. The most significant bit of the array type is a flag that, if set (i.e. if byte is negative), indicates that the record is in an alternate key space projected for use for an audit trail. The spare bytes may be used in this mode as well. The least significant byte is the record sequence number. It counts sequentially from one to the number of records for each array. It is always one for the first record of an array and zero for the array header.

In summary, a single SDB tagword consists of one or more data array types. It always contains an SDB header at minimum. Other than the SDB header, all data array types have an array header preceding them, as described in a previous section. The key structure is summarized in the following table:

Item	Tagword	Spare	Array Type	Sequence #
SDB Header	TW	0	0	0
Data Array Header	TW	0	N	0
Data Array Records	TW	0	N	1–M

where TW is the tagword in question, N is the data array type and M is the number of records in the data array.

3.2.1.2 SIZEHDR **Integer** **2 bytes**

Size of the header in bytes.

3.2.1.3 NBYTES_PER_RECORD **Integer** **2 bytes**

Number of bytes in a single data record.

3.2.1.4 NUMREC **Integer** **2 bytes**

Number of data records in this data array.

3.2.1.5 NUMENT **Integer** **2 bytes**

Number of entities, in the case of spectrum data, channels, in this data array.

3.2.1.6 SIZEENT **Integer** **2 bytes**

Size in bytes of each entity, in this case, spectrum channel.

3.2.1.7 DATAVERSION **Integer** **2 bytes**

Version number of the entity structure of this type of data (not expected to change for spectrum data).

3.2.1.8 HDRVERS **Integer** **2 bytes**

Physical version of header. Headers with different version numbers are not necessarily software compatible.

3.2.1.9 HDRREV — Byte — 1 byte

Header revision number. Incremented each time interpretation of information in header is changed without change of physical version (see HDRVERS).

3.2.1.10 NUMARRAY — Byte — 1 byte

Number of data arrays stored under this header, including primary data array. Maximum number is 32, minimum is 0 (header only).

3.2.1.11 CONTFLGS1 — Bits — 4 bytes

Bit flags indicating which arrays are present, 1st 32 data types.

```
CONTFLGS1       Bit  n          0 = DATA ARRAY TYPE n NOT PRESENT
                                1 = DATA ARRAY TYPE n PRESENT

                where n = 0-31
```

3.2.1.12 CONTFLGS2 — Bits — 4 bytes

Bit flags indicating which arrays are present, 2nd 32 data types.

```
CONTFLGS1       Bit  n          0 = DATA ARRAY TYPE n+32 NOT PRESENT
                                1 = DATA ARRAY TYPE n+32 PRESENT

                where n = 0-31
```

3.2.1.13 SPECFLGS — Bits — 2 bytes

Bit flags indicating various spectrum information as indicated in the codes section above, including real or integer data, 2-byte or 4-byte data, spectra or time history, glitch information, and whether or not the spectrum has been flagged for deletion. The latter may be tested by testing for a negative value of the variable.

```
SPECFLGS         Bit 0          0 = REAL DATA            0
                                1 = INTEGER DATA
                     1          0 = 2 BYTE INTEGER       0
                                1 = 4 BYTE INTEGER
                     2          0 = SDB                  0
                                1 = TIME SERIES DB
                     3          0 = DSF PRODUCT          0
```

	1 = DERIVED PRODUCT	
4	0 = ENVIRONMENT DATA NOT FILLED	0
	1 = ENVIRONMENT DATA FILLED	
5	0 = AUTOMATIC DEGLITCH NOT RUN	0
	1 = AUTOMATIC DEGLITCH RUN	
6	0 = VISUAL DEGLITCH NOT RUN	0
	1 = VISUAL DEGLITCH RUN	
7	0 = NO GLITCH FOUND DURING SCANNING	
	1 = GLITCH FOUND DURING SCANNING	
8	1 = POST-MERGE PROCESS RUN	
	0 = POST-MERGE PROCESS NOT RUN	
9	1 = SPECTRUM GAIN-NORMALIZED	0
	0 = SPECTRUM NOT GAIN-NORMALIZED	
15	0 = DATA NOT FLAGGED FOR DELETION	0
	1 = DATA FLAGGED FOR DELETION	

3.2.1.14 SOFTINDEX **Integer** **4 bytes**

Index into table containing types of software products used to create and/or modify spectra.

3.2.1.15 SOFTMAJOR **Integer** **4 bytes**

Major version number of software specified in SOFTINDEX.

3.2.1.16 SOFTMINOR **Integer** **4 bytes**

Minor version number of software specified in SOFTINDEX.

3.2.1.17 CREDATE **Real*8** **8 bytes**

Date and time of creation of header (8-byte VAX time format).

3.2.2 OBSERVATION DESCRIPTION

3.2.2.1 TARGID — Integer — 4 bytes

Index into catalog of target names.

```
TARGID          Code 0 - 99     Calibration and Test    0
                     0          No target specified
                     1          Y-88
                     2          Co-57
                     3          Na-22
                     4          Cs-137
                     5          Th-228
                     6          Co-60
                     7          Sn-113
                     8          Ce-139
                     9          Hg-203
                     10         Am-241
                     11         Co-56
                     12         Bi-207
                     13         Am-Be
                     14         Cm-C
                     15         K-40
                     16         N030
                     17         N055
                     18         N090
                     19         N120
                     20         N140
                     21         N200
                     22         p+FLUORINE
                     23         p+LITHIUM
                     24         p+ALUMINUM
                     25         p+BORON
                     26         p+TRITIUM
                     27         DECAY
                     28         LED
                     29         RAMP
                     30         BACKGROUND
                     >= 100     FLIGHT
```

3.2.2.2 INST_CNFG_ID — Bits — 2 bytes

Identifies hardware versions used to collect data. Specifies which detector electronics (DE) and which processor electronics (PE) were in use at the time of collection.

INST_CNFG_ID	Bit 0	SCIENCE MODEL DE	BITS 2,6 SET
	1	ENGINEERING MODEL DE	
	2	FLIGHT DE	
	3	PE BRASSBOARD	
	4	BCU	
	5	CE SIMULATOR	
	6	FLIGHT PE	
	ALL CLEARED	NO VALUE ASSIGNED	

3.2.2.3 FLTID — Integer — 2 bytes

Identifies phase of program during which data were collected.

FLTID	Code 100	E/M CBD CAL	0
	200	E/M NEUT CAL	
	300	E/M PHOT CAL	
	400	E/M K-40	
	401	BALLOON FLT	
	402-499	ALL SUBSEQUENT E/M USAGES	
	500	F/M 1987 CAL	
	600	F/M PRE-SHIP	
	700	F/M K-40	
	800	F/M A&T	
	801-899	ALL SUBSEQUENT F/M USAGES	
	900	ALL SUBSEQUENT STORAGE TESTS	
	1000	FLIGHT	
	0	NOT ASSIGNED	

3.2.2.4 DETID — Bit — 2 bytes

Identifies detector(s) that produced spectra. In a number of calibration cases, default value of zero gives false indication of engineering model detector when variable was not filled in.

DETID	Bit 0	DETECTOR 1	ALL CLEARED
	1	DETECTOR 2	
	2	DETECTOR 3	
	3	DETECTOR 4	
	4	BCU	
	5	DETECTOR SIMULATOR	
	6	ENGINEERING MODEL	
	7	SCIENCE MODEL	
	ALL CLEARED	NOT SPECIFIED	

3.2.2.5 SPECSRC **Byte** **1 byte**

Unsigned byte variable indicating PHA/Data channel from which data were taken. The first three bits are reserved for spectral memory low, medium and high ranges. The remaining bits, with the first three bits masked to zero, form a code. Science processes are encoded by their process ID (SPECSRC = PID*8 to allow for first three bits being masked off). The only other science process that makes use of the low three bits is ^{60}Co, which has low and medium range options. These should be represented by the base code (64 for ^{60}Co) summed with the appropriate range bits. In the case of low and medium range being present in the same spectrum either the base code plus 3 (low and medium bits set) or the base code alone are allowable.

For use in applications, this variable must be zero-extended into an i*2.

Other allowable values include vertical (energy spectrum) or horizontal (time spectrum) cuts across two parameter RPHA spectra, and various external data sources.

The data channel is further identified by CALRPHA described below.

```
SPECSRC         Bit  0          SPEC_LOW                7 (BITS 0,1,2)
                     1          SPEC_MED
                     2          SPEC_HIGH

                Code 0          SPEC_ALL                   (PROCESS ID
                                                                   *8)
                     1          SPEC_LOW
                     2          SPEC_MED
                     3          SPEC_L+M
                     4          SPEC_HIGH
                     5          SPEC_L+H
                     6          SPEC_M+H
                     7          SPEC_L+M+H
                     8          CAL
                     16         BURST
                     24         RPHA2
                     48         RPHA1
                     65         CO60_LOW |
                     66         CO60_MED |-- basic Co60 process ID
                     67         CO60_L+M |    + range bits
                     72         PULSAR
                     80         CANBERRA
                     88         CANRPHA1
                     96         CANRPHA2
                     104        EXTERNAL
                     112        MCD1
                     120        MCD2
                     128        VERTICAL CUT RPHA2      |
                     136        HORIZONTAL CUT RPHA2    |--Assume
                     144        VERTICAL CUT RPHA1      |  unsigned
                     152        HORIZONTAL CUT RPHA1    |  i*1;  must
```

```
                                                  zero-extend
                                                  into i*2
                160+1st 3  FLAG TO INDICATE RANGES
                 bits       CONCATENATED WITH OVERLAP REMOVED
```

3.2.2.6 CALRPHA — Byte — 1 byte

In the case of SPECSRC indicating CALPHA, RPHA1, CANRPHA1, or vertical or horizontal RPHA1 cuts, further information concerning data is required. This variable contains a code indicating the CAL MUX address (but not equal to that address—see SDB Header Codes) in the case of CALPHA or a code indicating RPHA mode (low, medium or CAL MUX address if RPHA is pointing at CALPHA). For details, see codes section.

```
CALRPHA (SPECSRC = 8 = CAL)
                Code 0          NOT APPLICABLE          0
                     1          SPEC MEM LOW
                     2          SPEC MEM MEDIUM
                     3          SPEC MEM HIGH
                     4          PIN
                     5          SHIELD 1
                     6          SHIELD 2
                     7          SHIELD 3
                     8          SHIELD 4
                     9          PMT
                     10         CO 60 BETA
                     11         CPD

CALRPHA (SPECSRC = 48,88,144,152 = RPHA1 AND DERIVATIVES)
                Code 0          NOT APPLICABLE          0
                     1          RPHA1 LOW
                     2          RPHA1 MEDIUM
                THE FOLLOWING APPLY TO RPHA POINTING TO CAL:
                     4          PIN
                     5          SHIELD 1
                     6          SHIELD 2
                     7          SHIELD 3
                     8          SHIELD 4
                     9          PMT
                     10         CO 60 BETA
                     11         CPD
```

3.2.2.7 TLMID — Byte — 1 byte

Telemetry format ID of first TSAC summed into spectrum.

3.2.2.8 DSFSTAT **Bits** **4 bytes**

Bit flags indicating the conditions under which DSF produced the spectrum. In particular, the accumulation stop mode and whether any events occurred that caused data to be shifted out of primary data will be indicated. In addition, user-defined events may be recorded. The MSbit will be set (allowing sign tests) if bits 2–8 or any user bits have been set.

		EVENT	RESULT
DSFSTAT	Bit 0	DETECTOR MOTION INCLUDED IN (PRIMARY) ACCUMULATION	INFORMATIONAL
	1	TIMEOUT	STOP ALL DETS
	2	DETECTOR MOTION OCCURRED	REROUTE IF DESIRED
	3	ORBITAL EVENT	STOP ALL DETS AND REROUTE IF DESIRED
	4	TM FORMAT CHANGE (SYSTEM EVENT)	STOP ALL DETS
	5	OPMODE NOT EQUAL TO 4 (SYSTEM EVENT)	STOP ALL DETS
	6	LVLy$Px CHANGES (DETECTOR EVENT)	STOPS THAT DET
	7	LOW/HIGH DEAD TIMES (DETECTOR EVENT)	STOPS THAT DET
	8	DETECTOR POSITIONING DISABLED (DETECTOR EVENT)	STOPS THAT DET
	9	POSTION INDEX CHANGE	STOP ALL DETS
	10	OPMODE RETURNED TO 4	INFORMATIONAL
	11	POINTING VALUE ADJUSTMENTS MADE	
	12-30	USER-DEFINED EVENTS	
	31	EVENT OCCURRED	
	all cleared	NO EVENTS DURING SPECTRUM, DEFAULT MODE STOP	

For the DSFSTAT variable stored in Pulsar Data Files, only bits 1, 4, 5, 9, and 10 are used.

Bit 10 is defined here solely for compatibility with the PDF DSFSTAT variable and should never be set in SDB.

3.2.2.9 DATAQUAL **Bit** **4 bytes**

Bit flags indicating that a glitch or other problem was found during automatic or visual scanning of the data. If no bits are set, no glitch was found. The data quality flags, as set by PDA/DSF are as follows:

DATAQUAL	Code	0	missing packets in accumulation
		1	bad packets during collection (e.g. bad checksum, fill data, etc.)
		2	reactor events
		3	OSSE Burst Trigger
		4	BATSE burst trigger
		5	BATSE flare trigger
		6	OSSE pointing offset error (correctable)
		7	OSSE pointing anomaly (uncorrectable)
		8	s/c reorientation within last 12 hours
		9	DSF Detector event
		10	gain anomaly
		11	single channel discontinuity
		12	electron precipitation event
		13	detector configuration anomaly
		14	instrument configuration anomaly
		15	AGC anomaly
		16	uncategorized transient
		17	solar flare
		18	burst
		19	spectrum overflow
		20	orbital ephemeris anomaly
		21	s/c pointing anomaly
		22	utc error
		23	OSSE pointing guess (usually short sum)

3.2.2.10 REACTOR — Bit — 1 byte

Bit flags indicating that a reactor event was predicted and/or found during the data accumulation.

Bit	none	No reactor event	none
	0	predicted	
	1	observed	
	2	positron event	
	3	electron event	
	4	gamma event	

3.2.2.11 ACHD — Byte(25) — 25 bytes

The Acquisition Header variable contains the following information:

a. CALPHA Acquisition Header (12 bytes)

```
Bytes

0,1    ACQ ENTRY NUMBER, OVERFLOW INDICATOR
2,3    CAL_RECORD_CONTROL (INCLUDES COMPRESSION MODE)
4,5    CLAD$S (CAL MUX ADDRESS)
6,7    CLLD$S (LEVEL COMMAND)
8,9    CLLR$S,CLONE$S,CLOVE$S (CAL MODE DATA)
10,11  CALTS$S (ACQUISITION TSAC COUNT)
```

b. Cobalt 60 Strategy Table (14 bytes)

```
Bytes

0,1    ACQUISITION MODE (DETECTOR SYNCH. OR FREE RUNNING),
NUMBER OF CONSECUTIVE SLOTS
2,3    ACQUISITION TSAC COUNT
4,5    NUMBER OF ACQUISITION CYCLES (DET. SYNCH. MODE ONLY)
6      LOW RANGE ENABLE FLAG
7      LOW RANGE COMPRESSION MODE
8      LOW RANGE NUMBER OF CHANNELS
9      LOW RANGE START CHANNEL
10     MEDIUM RANGE ENABLE FLAG
11     MEDIUM RANGE COMPRESSION MODE
12     MEDIUM RANGE NUMBER OF CHANNELS
13     MEDIUM RANGE START CHANNEL
```

c. RPHA1/RPHA2 Strategy Table (20 bytes)

```
Bytes

0,1    DETECTOR ID/GOVERNOR SETTING
2,3    ACCUMULATION MODE
4,5    ACQUISITION TSAC COUNT OR SMS TIME INTERVAL
6,7    COMPRESSION MODE
8,9    FULL COUNT TERMINATION FLAG
10,11  UPPER THRESHOLD (EXPANDED MODES)
12,13  LOWER THRESHOLD (EXPANDED MODES)
14,15  BUFFER ADDR
16,17  DATA SOURCE COMMAND
18,19  CAL SELECT COMMAND
```

d. Burst Strategy Table (16 bytes)

```
Bytes

0,1    BURST CONTROL WORD
2,3    RATE THRESHOLD
```

```
        4,5    SUCCESSIVE SAMPLE THRESHOLD
        6,7    PRETRIGGER LENGTH
        8,9    TIME OF TRIGGER (LO WORD)
        10,11  TIME OF TRIGGER (HI WORD)
        12,13  TRIGGER LOCATION IN BUFFER
        14,15  TYPE OF TRIGGER

    e.  Spectral Memory Strategy Table (6 bytes)

        Bytes

        0,1    LOW RANGE START CHANNEL/COMPRESSION CODE
        2,3    MEDIUM RANGE START CHANNEL/COMPRESSION CODE
        4,5    HIGH RANGE START CHANNEL/COMPRESSION CODE

    f.  Pulsar (25 bytes)

        Bytes

         0-2   WINDOW 0 EMIN, EMAX, SCNT
         3-5   WINDOW 1 EMIN, EMAX, SCNT
         6-8   WINDOW 2 EMIN, EMAX, SCNT
         9-11  WINDOW 3 EMIN, EMAX, SCNT
        12-14  WINDOW 4 EMIN, EMAX, SCNT
        15-17  WINDOW 5 EMIN, EMAX, SCNT
        18-20  WINDOW 6 EMIN, EMAX, SCNT
        21-23  WINDOW 7 EMIN, EMAX, SCNT
        24     STATUS FLAGS                (BIT 0   = PULSAR
                                                      MODE
                                            BIT 1   = EBE MODE
                                            BIT 2-4 = RATE
                                                      INTERVAL
                                            BIT 5-6 = RATE
                                                      COMPRESS
```

3.2.2.12 SPECCAT — Byte — 1 byte

Type of data in the spectrum data array referred to by the header (i.e. PHA spectrum, incident photon spectrum, time series data, etc.). In the case of raw data, all will be PHA and PSD spectra.

```
SPECCAT         Code   0        NO CODE ASSIGNED        0
                       1        ENERGY/PHA SPECTRUM
                       2        DIFFERENCE PHA SPECTRUM
                       3        ESTIMATED INCIDENT PHOTON SPECTRUM
                       4        ESTIMATED BACKGROUND
                      20        DIFLIN
                      21        DIFLINC (CLEANED DIFLIN)
```

```
 40        TIME  (PSD SPECTRUM)
 41        TIMDIF (DIFFERENCE PSD
           SPECTRUM)
 60        THIST (TIME OR PHASE HISTOGRAM)
100        FOUR (FOURIER)
120        CHI2_MAP
```

3.2.2.13 TARGOFF — Integer — 1 byte

TARGOFF contains the angle, truncated to integer quarter degrees, between the designated target (of which one or more shall be defined for each pointing table) and the current pointing position (i.e. TARGSCAN truncated to integer degrees). This shall be defined such that an offset from the target in the direction of the spacecraft x-axis is positive and an offset in the direction of the spacecraft z-axis is negative. In either case, the variable TARGID shall identify the target from which the offset is referenced.

3.2.2.14 SPECUNIT — Integer — 2 bytes

Bit flags that specify units of data. Raw data from archives will almost always be in counts. The most significant bit of the 2-byte word is a flag that indicates that the data have been resampled into a new bin structure. In general, the low order byte (bits 0–7) represents the denominator of the units, while the high order byte (bits 8–15) represents the numerator and the resample flag.

```
SPECUNIT     Bit  0       1/SEC                     bits 0,8 set
                  1       1/CM2 (effective)
                  2       1/MEV
                  3       1/KEV
                  4       1/DIVISOR (SEE DIVISOR VARIABLE)
                  5       1/RADIAN
                  6       1/STERADIAN
                  7       RESERVED FOR EXPANSION
                  8       COUNTS
                  9       PHOTONS
                  10      MEV
                  11      KEV
                  12      ERG
                  13      RESERVED FOR EXPANSION
                  14      RESERVED FOR EXPANSION
                  15      =0  DATA NOT RESAMPLED
                          =1  DATA RESAMPLED
```

3.2.2.15 NCHAN — Integer — 2 bytes

Total number of spectrum channels for all ranges present. This is valid for all spectrum data types.

3.2.2.16 NCHANL **Integer** **2 bytes**

Number of spectrum channels for low range only. This is valid for spectral memory and ^{60}Co data only.

3.2.2.17 NCHANM **Integer** **2 bytes**

Number of spectrum channels for medium range only. This is valid for spectral memory and ^{60}Co data only.

3.2.2.18 NCHANH **Integer** **2 bytes**

Number of spectrum channels for high range only. This is valid for spectral memory and ^{60}Co data only.

3.2.2.19 NCHAN1 **Integer** **2 bytes**

In the case of a two-parameter spectrum, equal to the number of channels in parameter 1, where parameter 1 is defined as the most rapidly changing address (i.e. the spectrum may be viewed as NCHAN2 spectra, each of length NCHAN1, concatenated).

In the case of a one-parameter spectrum, equal to the number of channels.

3.2.2.20 NCHAN2 **Integer** **2 bytes**

In the case of a two-parameter spectrum, equal to the number of channels in parameter 2.

In the case of a one-parameter spectrum, set equal to 0 (although logically this should be set to 1 in this case, setting it to 0 allows the variable to be used as a logical flag where one-parameter = FALSE and two-parameter = TRUE).

3.2.2.21 PHBINNUM **Integer** **2 bytes**

Pulsar pulse phase bin number.

3.2.2.22 NUMPHBIN **Integer** **2 bytes**

Number of current pulse phase bins.

3.2.3 TIMES AND COLLECTION INTERVALS

3.2.3.1 STRTTIM Integer 4 bytes

Start time of actual spectrum acquisition represented as the first four bytes of the PB5 time format (i.e. 15 bits of truncated Julian day followed by 17 bits of seconds of day; same format as UTC).

For slot spectra, the slot header contains the TMAFC of the beginning packet of the acquisition, from which the time shall be derived. This time may deviate from the actual start of the acquisition by 1 TSAC.

For other forms of data (e.g. burst, time series), the intent is that this time represents as closely as possible the start time of acquisition of the data at the detector (as opposed to the time the data were telemetered).

For burst data in particular, this time shall represent the burst trigger time tag.

3.2.3.2 STRT_FRAC Integer 2 bytes

This shall be the fractional seconds of day corresponding to STRTTIM (same format as PKTMS).

3.2.3.3 STOPTIM Integer 4 bytes

Stop time of spectrum acquisition represented as the first four bytes of the PB5 time format (i.e. 15 bits of truncated Julian day followed by 17 bits of seconds of day; same format as UTC).

For slot spectra, the slot header contains the TMAFC of the beginning packet of the acquisition and the number of TSACS of the acquisition, from which the time shall be derived. This time may deviate from the actual stop time of the acquisition by 1 TSAC.

For other forms of data (e.g. burst, time series), the intent is that this time represents as closely as possible the stop time of acquisition of the data at the detector (as opposed to the time the data were telemetered).

For burst data in particular, this time shall represent the burst trigger time tag plus the number of burst channels times the channel duration.

3.2.3.4 STOP_FRAC Integer 2 bytes

This shall be the fractional seconds of day corresponding to STOPTIM (same format as PKTMS).

3.2.3.5 ORBIT **Integer** **4 bytes**

Orbit number as defined by FOT. Also in orbit history auxilliary data structure

3.2.3.6 HSKPTIM **Integer** **4 bytes**

UT time representative of housekeeping data in header. This is important for slot data since the housekeeping data are not necessarily sampled at the same time as the slot data. In general, housekeeping data for summations is taken from the last acquisition of the summation.

For Spectral Memory spectra, HSKPTIM is the time of the last TSAC of acquisition. Depending on whether the housekeeping data are delayed along with the spectra or transmitted immediately (e.g. pointing), HSKPTIM is offset from the last telemetered TSAC time or not.

For slot data, HSKPTIM is the telemetered (i.e. not offset) time of the first slot header of a summation.

3.2.3.7 TSAC **Real** **4 bytes**

Number of seconds comprising a single TSAC. If an accumulation includes multiple formats with different TSAC lengths, this is not valid.

3.2.3.8 ACCUMTIM **Real** **4 bytes**

Accumulation time for spectrum in seconds of wall clock time. Equal to TSAC times the number of TSAC's summed into the accumulation.

3.2.3.9 LTL **Real** **4 bytes**

Live time seconds. In the case of spectral memory and ^{60}Co data, this is the live time for low range.

In the case of all other data, this is the best estimate available of the live time regardless of range. In general, best estimate means that spectral memory dead times for the appropriate time period have been applied to the relevant acquisition time.

Precision of live time is seven significant digits maximum, as dictated by VAX real*4. This should be adequate under any conditions. However, this means that precision of dead time subtraction may be lost under some circumstances (this should never be significant, though).

3.2.3.10 LTM **Real** **4 bytes**

Live time seconds. In the case of spectral memory and ^{60}Co data, this is the live time for medium range.

In the case of all other data, this is set to 0.

3.2.3.11 LTH **Real** **4 bytes**

Live time seconds. In the case of spectral memory data, this is the live time for high range.

In the case of all other data, this is set to 0.

3.2.4 DETECTOR POINTING

3.2.4.1 DETRA **Real** **4 bytes**

Detector Right Ascension (epoch 2000) measured from gamma-ray axis. For pointing interval spectra from production analysis, measured as of end of the interval (generally guaranteed constant over the interval). In degrees.

3.2.4.2 DETDC **Real** **4 bytes**

Detector Declination (epoch 2000) measured from gamma-ray axis. For pointing interval spectra from production analysis, measured as of end of the interval (generally guaranteed constant over the interval). In degrees.

3.2.4.3 DETSIGMA_SCAN **Real** **4 bytes**

Mean pointing uncertainty in the detector scan direction. May be higher than normal if spacecraft using other than star trackers for aspect solution. One standard deviation in degrees.

3.2.4.4 DETSIGMA_XSCAN **Real** **4 bytes**

Mean pointing uncertainty in the detector transscan direction. May be higher than normal if spacecraft using other than star trackers for aspect solution. One standard deviation in degrees.

3.2.4.5 TARGRA **Real** **4 bytes**

Right Ascension (epoch 2000) of the current target location in degrees. In the case of point source observations, the target location is the position of the primary source of interest in the field of view and is not necessarily the pointing direction. For example, in the case of observations attempting to position a source, several observations at offset angles will be made. The value of TARGRA and TARGDC shall be the same, namely the RA and DEC of the source of interest, for all of these spectra. Similarly, in a normal chopping observation, the value of TARGRA and TARGDC shall be the same for all source and background pointed spectra since they all reference the same source of interest.

In the case of diffuse source mapping observations, the target location shall coincide with the pointing direction for target points and the target pointing direction for background points.

3.2.4.6 TARGDC Real 4 bytes

Declination (epoch 2000) of the current target location in degrees. In the case of point source observations, the target location is the position of the primary source of interest in the field of view and is not necessarily the pointing direction. For example, in the case of observations attempting to position a source, several observations at offset angles will be made. The target location should be the same for both source and background pointings.

In the case of diffuse source mapping observations, the target location shall coincide with the pointing direction for target points and the target pointing direction for background points.

3.2.4.7 TARGSCAN Real 4 bytes

Position of the target in detector collimator coordinates. This is equivalent to the scan angle. The origin of the coordinate system is defined as the geometric center of the detector. The latitude angle is defined as the angle between the target vector and the detector x-y plane (where the detector x-axis is the gamma-ray axis and the y-axis is the detector pointing axis). TARGSCAN increases in the counterclockwise direction when viewing the detectors from the spacecraft +y axis (i.e. increases away from the spacecraft +z axis when the detector is between spacecraft +z and +x). In units of degrees.

3.2.4.8 TARGXSCAN Real 4 bytes

Position of the target in detector collimator coordinates (see TARGSCAN for coordinate system origin definition). This is equivalent to the transscan angle. The longitude is defined as the angle between the projection of the target vector on the detector x-y plane and the detector x-axis. TARGXSCAN increases in the counterclockwise direction when viewed from the detector +z axis (see coordinate system definition under TARGSCAN). In units of degrees.

3.2.4.9 COLLPOSANG Real 4 bytes

Collimator position angle. The angle between the north celestial pole and the long axis of the collimator. Increases CCW from the pole (i.e. increases from north through east). Range is 0–180 degrees.

3.2.4.10 EARTHSEP Real(2) 8 bytes

Separation angle between the center of the FOV and the center of the earth, in degrees. EARTHSEP and EARTHAZ are in a coordinate system where the center of the FOV is on the z-axis with the scan direction along the x-z plane. Sampled during the first and last packets of the accumulation.

3.2.4.11 EARTHAZ — Real(2) — 8 bytes

Rotation (azimuth) angle between the FOV center-earth direction plane and the FOV center-scan direction plane ("x-z plane", see EARTHSEP above). Zero when the earth is on the +x-z plane and increases CCW when viewed from the detector FOV center (z axis). Sampled during the first and last packets of the accumulation.

3.2.4.12 DSA — Integer — 2 bytes

Detector step angle at the end of the accumulation.

3.2.4.13 AM — Real — 4 bytes

Average pot value for main pot.

3.2.4.14 AR — Real — 4 bytes

Average pot value for redundant pot.

3.2.4.15 FDET — Bits

Detector positioning status. Offset index and active pot are those in effect at the end of the accumulation. The positioning error flag is set if a positioning error occurred anytime during the accumulation. The calibrated flag is cleared if the drive is not calibrated anytime during the accumulation. The slewing flag is set if slewing occurred anytime during the accumulation, and the positioning enabled flag is set if positioning was enabled anytime during the accumulation.

```
FDET           Bit 0-2        OFFSET # (0-7)         SAME AS TM
                   3          ACTIVE POTENTIOMETER   VARIABLE
                   4          POSITIONING ERROR
                   5          CALIBRATED
                   6          SLEWING
                   7          POSITIONING ENABLED
```

3.2.4.16 DPTBL — Byte — 1 byte

Integer code indicating which positioning table was active for accumulation.

```
DPTBL          Code 0         PRIMARY TABLE          ALL CLEARED
                    1         SECONDARY TABLE
                    2         SOLAR TABLE
```

3	TRANSIENT TABLE
ALL CLEARED	NO CODE ASSIGNED

3.2.5 TIME SERIES SPECIFIC CONTENTS

These variables will be described in detail in a later revision.

3.2.5.1 INTDURAT Integer 2 bytes

3.2.5.2 INTLIVTIM Integer 2 bytes

3.2.5.3 INTDURUNITS

```
INTDURUNITS     Code 0         TSACS                  0
                     1         MSEC/8
                     2         MSEC/4
                     3         MSEC/2
                (IN GENERAL, (2**(CODE-1))*(MSEC/8))
```

3.2.5.4 NOINTQUANT Integer 2 bytes

3.2.5.5 NOINTPARAM Integer 2 bytes

3.2.6 DETECTOR PARAMETERS

Descriptions of all the detector parameters are found in the OSSE Command and Telemetry Document (NRL Document 0926-001).

3.2.6.1 CMD_STATE_ID_1 Character 8 bytes

An index into the INGRES Command State Table specifying the command state valid at the end of the first DSF-summed interval of the accumulation. This table contains the command state vector, a collection of ID's into tables of configuration file names.

3.2.6.2 CMD_STATE_ID_2 Character 8 bytes

An index into the INGRES Command State Table specifying the command state valid at the end of the last DSF-summed interval of the accumulation. This table contains the command state vector, a collection of ID's into tables of configuration file names.

3.2.6.3 DET_CNFG_ID_1 Integer 4 bytes

An integer index into the INGRES DEn_CFG table valid at the end of the first DSF-summed interval of the accumulation, where n is the detector addressed by the header (1–4). This table contains the detector configuration file name.

3.2.6.4 DET_CNFG_ID_2 Integer 4 bytes

An integer index into the INGRES DEn_CFG table valid at the end of the last DSF-summed interval of the accumulation, where n is the detector addressed by the header (1–4). This table contains the detector configuration file name.

3.2.6.5 STRTBILVL Bits 10 bytes

Bilevels 0–4, sampled at the end of the first TSAC of the accumulation.

3.2.6.6 ENDBILVL Bits 10 bytes

Bilevels 0–4, sampled at the end of the last TSAC of the accumulation.

3.2.6.7 Analogs

For Spectral Memory data, these are sampled at the end of the last TSAC of the accumulation.

For slot data, these are sampled at the end of the first acquisition (i.e. first dump) of the accumulation.

3.2.7 RATES

Descriptions of all the rates are found in the OSSE Command and Telemetry Document (NRL Document 0926-001). Rates are stored in counts. Divide by RATE_TIM to obtain actual rates. Exceptions are DTx_SUM, the dead times, which are in units of 0.000250 seconds. Rates are summed over the same interval as the spectra.

3.2.7.1 RATE_TIM Real 4 bytes

Acquisition time appropriate to header rates (which are not stored as rates but as count sums). This is computed from the number of TSAC's in the rate summation times TSAC.

3.2.8 ENVIRONMENT

3.2.8.1 GROX Real(2) 8 bytes

X coordinate of spacecraft in earth-centered inertial coordinate system, epoch 2000, in kilometers, sampled during the last packet of the first and last DSF-summed interval of the accumulation.

3.2.8.2 GROY Real(2) 8 bytes

Y coordinate of spacecraft in earth-centered inertial coordinate system, epoch 2000, in kilometers, sampled during the last packet of the first and last DSF-summed interval of the accumulation.

3.2.8.3 GROZ Real(2) 8 bytes

Z coordinate of spacecraft in earth-centered inertial coordinate system, epoch 2000, in kilometers, sampled during the last packet of the first and last DSF-summed interval of the accumulation.

3.2.8.4 GROLT Real(2) 8 bytes

Geographic latitude of sub-spacecraft point, in degrees.

3.2.8.5 GROLG Real(2) 8 bytes

Geographic longitude of sub-spacecraft point, in degrees.

3.2.8.6 GRORA Real(2) 8 bytes

RA of s/c in earth-centered inertial coordinates, epoch 2000

3.2.8.7 GRODC Real(2) 8 bytes

Declination of s/c in earth-centered inertial coordinates, epoch 2000

3.2.8.8 GRORAD Real(2) 8 bytes

Altitude of spacecraft from origin of earth-centered inertial coordinate system used above, in km.

3.2.8.9 ATTZRA Real(2) 8 bytes

+Z Axis RA, epoch 2000, sampled during the first and last packets of the accumulation. In units of degrees.

3.2.8.10 ATTZDC Real(2) 8 bytes

+Z Axis declination, epoch 2000, sampled during the first and last packets of the accumulation. In units of degrees.

3.2.8.11 ATTXRA Real(2) 8 bytes

+X Axis RA, epoch 2000, sampled during the first and last packets of the accumulation. In units of degrees.

3.2.8.12 ATTXDC Real(2) 8 bytes

+X Axis declination, epoch 2000, sampled during the first and last packets of the accumulation. In units of degrees.

3.2.8.13 MCL Real(2) 8 bytes

McIlwain L parameter. Computed from ephemeris data from first and last packets of accumulation.

3.2.8.14 MCB Real(2) 8 bytes

McIlwain B parameter. Computed from ephemeris data from first and last packets of accumulation.

3.2.8.15 RIGID Real(2) 8 bytes

Magnetic vertical cutoff rigidity. Computed from ephemeris data from the first and last packets of accumulation.

3.2.8.16 BRAD Real(2) 8 bytes

Component of the Earth's magnetic field vector in an earth-centered polar coordinate system radial direction. The coordinate system is the same as that used in the GEOMAG_GRID package. Sampled during the first and last packets of the accumulation.

3.2.8.17 BPOL Real(2) 8 bytes

Component of the Earth's magnetic field vector in the polar direction. Sampled during the first and last packets of the accumulation.

3.2.8.18 BAZ Real(2) 8 bytes

Component of the Earth's magnetic field vector in the azimuthal direction. Sampled during the first and last packets of the accumulation.

3.2.8.19 DAY_NIGHT Byte 1 byte

Code indicating whether the sun was above or below the horizon during the accumulation.

```
DAY_NIGHT       Code 0          ALL NIGHT
                     1          DAY AT START OF ACCUMULATION
                     2          DAY AT END OF ACCUMULATION
                     3          DAY FOR ENTIRE ACCUMULATION
```

3.2.8.20 CPML_RATE Real 4 bytes

CPM low rate averaged over accumulation.

3.2.8.21 CPMH_RATE Real 4 bytes

CPM high rate averaged over accumulation.

3.2.8.22 THIS_ORBITINTEG Real 4 bytes

Integration time from start of current orbit to end of this accumulation (seconds).

3.2.8.23 CPML_FRAC_ORBIT Real 4 bytes

CPM low dose from the start of the current orbit to the end of the accumulation (counts).

3.2.8.24 CPMH_FRAC_ORBIT Real 4 bytes

CPM high dose from the start of the current orbit to the end of the accumulation (counts).

3.2.8.25 LAST_SAA_EXIT — Real — 4 bytes

Time (in seconds) since the last mode 3 to mode 4 transition marking the exit from SAA.

3.2.8.26 LAST_SAAINTEG — Real — 4 bytes

Integration time of last SAA period (seconds).

3.2.8.27 CPML_LAST_SAA — Real — 4 bytes

Summed CPM low (minimum ionizing) dose (1 MeV $<$ E $<$ 8 MeV) in counts during the last SAA period, as defined by the period during which the instrument was in mode 3.

3.2.8.28 CPMH_LAST_SAA — Real — 4 bytes

Summed CPM high dose (E $<$ 8 MeV) in counts during the last SAA period, as defined by the period during which the instrument was in mode 3.

3.2.9 REMOTE TAGWORD POINTERS

Each spectrum header shall be able to point to several data arrays located under another tagword in the same file. These are referred to as "remote" data. Remote data are generally superseded by "local" data, i.e. arrays located in the same tagword as the spectrum header in question. Remote data are intended to reduce redundant storage of data that do not change rapidly from spectrum to spectrum.

3.2.9.1 CALTW — Integer — 4 bytes

Tagword of a spectrum containing "remote" calibration data, that is, calibration data valid for multiple spectra and not located under the same header as those spectra. The calibration spectrum must be in the same file as the current spectrum. If CALTW is less than or equal to zero, no remote calibration data is referenced (however calibration data may still be present under the same headers as the data).

Generally, a "remote" calibration data tagword will contain the following data arrays:

- Detector resolution
- Channel lower edge energies
- Channel lower edge energy uncertainties

- Channel widths
- Channel width uncertainties
- Channel representative energies
- Channel representative energy uncertainties
- Calibration line information
- DTW information

3.2.9.2 STDCALTW — Integer — 4 bytes

The standard calibration tagword is similar in function to the calibration tagword but points to arrays that are derived for a standard channel set into which a spectrum is resampled via gain normalization or via the computation of an average gain for some time interval. The tagword to which STDCALTW points will contain the following AUX types:

Edges, widths, etc. (types 10–16)
Calibration line information (type 17)
DTW information (type 18)
Simple response matrices (type 30)
Target positions (type 27)

although not all of the above types will be present in all cases. If the standard channel set is determined by the gain of another spectrum, as might be the case if a background spectrum is gain normalized to match a source spectrum, this tagword shall point to the relevant calibration information for the target configuration. If the standard channel set is artificial, for example a standard set of uniform channels into which all spectra are sampled, a new set of calibration records shall be created. In this case, the calibration line information and DTW information are not relevant.

3.2.9.3 ORBITTW — Integer — 4 bytes

Tagword containing "remote" orbit history data.

3.2.9.4 ASSOCSPECTW — Integer — 4 bytes

Tagword containing "remote" associated spectra data.

3.2.9.5 FOOTNOTETW — Integer — 4 bytes

Tagword containing "remote" footnotes.

3.2.10 USER DATA

The following are variables typically assigned by the user.

3.2.10.1 USRSCRNFLG **Bits** **4 bytes**

User controlled bit flags indicating screening of spectrum. Bits 0–31 can be assigned by users for screening purposes. Specifically, any queries that result in a subset of spectra being selected may cause bits in USRSCRNFLG to be set. Future programs may then determine which spectra to process by examining a single bit.

```
USRSCRNFLG      Bit  0-31       USER SCREENING FLAGS    All Cleared
                ALL CLEARED     NO FLAGS SET
```

3.2.10.2 EFFECAREA **Real** **4 bytes**

The effective area is the projected geometric area times the efficiency for detection in the photopeak. Dividing the count rate under a line photopeak by the effective area results in units of photons cm^{-2} sec^{-1}.

3.2.10.3 DIVISOR **Real** **4 bytes**

In some cases, it is useful to be able to roll a processed spectrum (i.e. one no longer represented by raw counts) back to a count spectrum. DIVISOR is the value the raw spectrum was divided by to obtain the processed spectrum. This is only valid in cases where all channels were divided by the same constant value.

DIVISOR may also be a user-supplied unit denominator in which the spectrum is expressed (i.e. counts/divisor). See SPECUNIT.

3.2.10.4 GSHIFT **Real** **4 bytes**

GSHIFT contains the mean fractional shift in channel edge energies for a spectrum that has been gain normalized. If the zero-energy point remains constant, this represents a first-order gain shift (i.e. assuming linear response).

3.2.10.5 COMMENT **Character** **160 bytes**

Freeform information concerning the spectrum.

3.2.11 MISCELLANEOUS

3.2.11.1 LOW_WIDTH **Real** **4 bytes**

Channel width of equivalent low range spectrum. Computed from:

$$(1332 - 1173)/(\text{High } ^{60}\text{Co line ch.} - \text{Low } ^{60}\text{Co line ch.})$$

where the difference in line centroid channels is corrected to the low range if gain is such that the lines are actually out of that range. In keV/channel.

3.2.11.2 RESERVE **Byte** **88 bytes**

Space not currently in use.

4 SDB User Interface

Previous versions of SDB have been accessed via a library of FORTRAN-callable routines that read and write spectra and perform other utility functions. These routines will not support version 6 of SDB directly, but shall serve as a guide for the version 6 routines. In particular, entry points shall be supplied into the version 6 routines that have identical calling sequences to the equivalent previous version routines (see NRL Doc. 0926-043, Spectral Database Library Users Guide).

The new utilities will, of course, require modified structures for the SDB header. In addition, any applications accessing the header will need to be recompiled with the new structures and will have to be examined carefully for variable names.

The major difference between version 6 utilities and previous versions, besides the new header format, is the multi-array capability of version 6. This shall be dealt with via the REC_TYPE variable in the header. Each call to the user read utility (e.g. SDB_SPEC_GET) shall be able to access two arrays, a data array and, usually, an uncertainty array. The arrays to access will be specified by the REC_TYPE variable, one array per read primitive. There shall be one user read utility (or entry point) for each data/uncertainty pair (spectrum, spectrum uncertainties; channel edge energies, uncertainties; etc.). These utilities will have a calling sequence identical to the current SDB_SPEC_GET. In addition, a new utility in which the user specifies the REC_TYPE pair explicitly in the argument list will allow access to user-defined arrays.

Although data may be stored as 2- or 4-byte integer or 4-byte real, the standard utilities shall convert data to 4-byte real for output. Special utilities for integer output may be supplied if required.