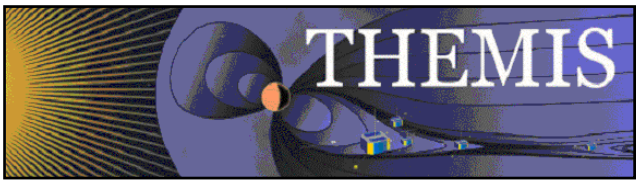


THEMIS
L1 File Definitions

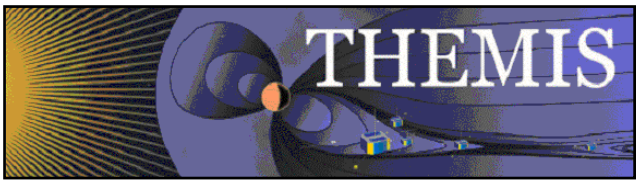
THM-SOC-124
July 2009

Jim Lewis, THEMIS Science Software Engineer

Jim McTiernan THEMIS Science Software Engineer

David King, THEMIS Science Software Manager

Vassilis Angelopoulos, THEMIS Principal Investigator



Document Revision Record

Rev.	Date	Description of Change	Approved By
-	2/21/2008	Draft	-
-		Signature release	

Distribution List

Name	Email
Jim Lewis, U.C. Berkeley	jwl@ssl.berkeley.edu
Dr. Tai Phan, U.C. Berkeley	phan@ssl.berkeley.edu
Dr. Ellen Taylor, U.C. Berkeley	ertaylor@ssl.berkeley.edu
Dr. Uli Auster, TUBS	uli.auster@tu-bs.de
Dr. Alain Roux, CETP	Alain.Roux@cetp.ipsl.fr
Dr. Krishan Khurana, UCLA	kkhurana@igpp.ucla.edu
Dr. Dave Sibeck, NASA GSFC	david.g.sibeck@nasa.gov

TBD List

Identifier	Description

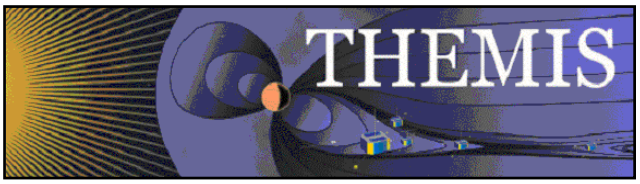
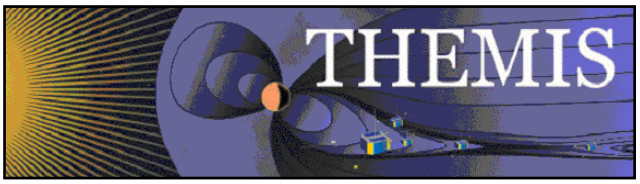


Table of Contents



1. Introduction

1.1 Purpose and Scope.

The THEMIS project has adopted the CDF file format for distributing "Level 1 (L1)" and "Level 2 (L2)" data products. The L1 products consist of time-tagged, uncalibrated data in instrument coordinates, while the L2 products contain key parameter data, calibrated and transformed into geophysically relevant coordinate systems. There is also a set of "Level 0 (L0)" products which are not stored as CDF files, but as raw packet data.

This document describes the structure and contents of the THEMIS L1 CDFs, and the processing steps involved in producing them.

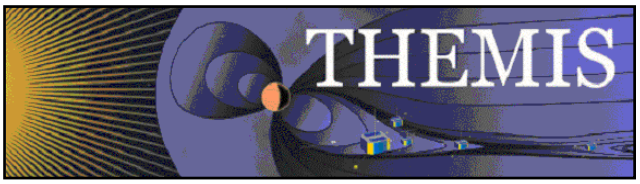
1.2 Applicable Documents.

1.	THM_SYS_012_PDMP	THEMIS Project Data Management Plan
2.	THM_SOC_101_TIME	THEMIS TIME Definition
3.	THM_SOC_105_FIELDS_L1_VARNAMES	THEMIS FIELDS Variable Name Def's
4.	THM_SOC_108_PARTICLES_L1_VARNAMES	THEMIS ESA/SST Variable Name Def's
5.	THM_SOC_108_GMAG_L2_VARNAMES	THEMIS GMAG Variable Name Def's
6.	THM_SOC_109_ASI_L1_VARNAMES	THEMIS ASI Variable Name Definitions
7.	THM_SOC_110_COORDINATES	THEMIS Coordinate Systems Definitions
8.	THM_SOC_111_SUNSENSPROC	THEMIS SUN SENSOR Science Processing
9.	THM_SOC_112_ATTPAIPROC	THEMIS Science ATT & Inertia Determ.
10.	THM_SOC_113_FGM_CALPROC	THEMIS FGM CAL File and Processing
11.	THM_SOC_114_SCM_CALPROC	THEMIS SCM CAL File and Processing
12.	THM_SOC_115_EFI_CALPROC	THEMIS EFI CAL File and Processing
13.	THM_SOC_116_ESA_CALPROC	THEMIS ESA CAL File and Processing
14.	THM_SOC_117_SST_CALPROC	THEMIS SST CAL File and Processing
15.	THM_OGS_431M	OGS Ephemeris File Definition
16.	THM_SOC_135_time_conventions	THEMIS timekeeping, leap seconds

2. CDF Format and Online Resources

All CDF documentation, precompiled libraries, source code, and online conversion tools are available from NASA-GSFC under the following URL (as of this writing, Dec 7 2007): http://cdf.gsfc.nasa.gov. The actual on-disk layout of CDF files is not documented. All manipulation of CDF files takes place via the CDF library API. The library routines (or subsets thereof) are callable from C, C++, Fortran, Perl, IDL, and Java programs (and potentially in any other language which supports linking with libraries written in C). THEMIS CDF data processing requires version 3.1 or later of the CDF library. Most IDL users will have to upgrade the CDF library and associated IDL DLM file shipped with IDL, depending on the platform and IDL version. The replacement library and DLM files are available from Goddard at the above URL.

The CDF command line tools include a "skeletontable" utility, which outputs a human-readable ASCII representation of the structure of a binary CDF file (and optionally, the data values as well). Another tool, "skeletoncdf", converts the opposite way, from an ASCII skeleton table to a binary CDF. The THEMIS L1 master CDFs are stored in our Subversion code repository in SKT format, and converted to binary CDFs and published to the THEMIS L1 data directories whenever a new version of the L0->L1 processing tools ("tmtools") is installed.



CDF files contain variables (the data), and attributes (the metadata).

Variables can be defined as scalars (of various primitive types: signed or unsigned bytes, 16-bit and 32-bit integers, 32-bit and 64-bit floating point values), or arrays (where each variable record is an array (up to 32 dimensions) of some CDF primitive type). All records of a given variable must have the same "shape" -- array sizes or number of dimensions cannot vary from one sample to the next.

Attributes may be global in scope (pertaining to the entire file), or be associated with individual variables. Global attributes might include metadata such as the project name, PI, file version, or processing date. Variable attributes are used to describe properties such as units, display types (e.g. line plot or spectrogram), or relationships with other variables (e.g. the DEPENDS_TIME attribute which associates a data variable with another variable containing sample times).

THEMIS CDF files are designed to use attribute and variable names and interpretations following a set of recommendations by the ISTP (International Solar and Terrestrial Physics) program. The ISTP requirements can be found at the following URL:

http://spdf.gsfc.nasa.gov/istp_guide/istp_guide.html

A CDF file's compliance with ISTP standards can be tested with a Java applet available here:

http://sscweb.gsfc.nasa.gov/skteditor/

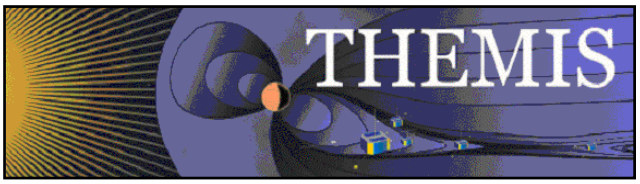
The ISTP standard uses variables of type CDF_EPOCH and CDF_EPOCH16 to represent timestamps. THEMIS has found it more convenient to work with double-precision floating point timestamps following the POSIX convention: number of seconds since the Unix epoch 1970-01-01 00:00:00 GMT (commonly referred to as "Unix time"). It should be noted that the usual Unix-like timekeeping functions do not account for leap seconds (and this behavior is actually written into the POSIX standard, therefore unlikely to change). As of this writing (2009-07-10), there has been one leap second introduced since the THEMIS launch, at the end of 2008 (23:59:60 UTC on 2008-12-31). There may be slight time discrepancies between probe data and ground data for a short period after 2009-01-01, lasting until the next routine adjustment of the probes' onboard clocks to synchronize them with the Unix-based times on the ground.

A few additional CDF variables and variable attributes have been established to connect the THEMIS timestamps to the CDF_EPOCH values required by the ISTP standard:

DEPEND_TIME is the THEMIS analogue to the ISTP DEPEND_0 attribute. DEPEND_0 points to a CDF_EPOCH variable, while DEPEND_TIME points to the CDF_REAL8 variable used by TDAS.

VIRTUAL is an attribute with values of "true" or "false". If true, the variable values do not appear in the CDF; rather they are calculated via some external code from other non-virtual variables present in the CDF. The name of this function is taken from the FUNCT attribute. This function takes its argument variables from the attributes COMPONENT_0, COMPONENT_1, etc.

A concrete example will best illustrate the usage of ISTP and THEMIS variables and attributes. We will use tha_fb1, where each record represents a 1-dimensional, 6-element array of unsigned bytes from Filter Bank 1 on probe THEMIS-A. Here is an excerpt from the skeleton table corresponding to the THEMIS-A master FBK L1 CDF:



Variable Name	Data Type	Number Elements	Dims	Sizes	Record Variance	Dimension Variances
tha_fb1	CDF_UINT1	1	1	6	T	T

Attribute Name	Data Type	Value
CATDESC	CDF_CHAR	Spectral Lines
DEPEND_0	CDF_CHAR	tha_fbk_epoch
DEPEND_1	CDF_CHAR	tha_fbk_freqno
DISPLAY_TYPE	CDF_CHAR	time_series
FIELDNAM	CDF_CHAR	Raw Spectra, ADC units
FILLVAL	CDF_UINT1	255
FORMAT	CDF_CHAR	i6
LABL_PTR_1	CDF_CHAR	tha_fbk_labl
UNITS	CDF_CHAR	ADC
VALIDMIN	CDF_UINT1	0
VALIDMAX	CDF_UINT1	255
VAR_TYPE	CDF_CHAR	data
SCALETYP	CDF_CHAR	linear
DEPEND_TIME	CDF_CHAR	tha_fbk_time
DEPEND_EPOCH0	CDF_CHAR	tha_fbk_epoch0

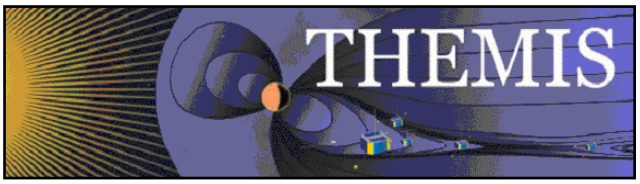
In the above attribute definitions, DEPEND_TIME refers to tha_fbk_time, which is the POSIX timestamp used by the TDAS tools. DEPEND_0 is the ISTP equivalent, pointing to tha_fbk_epoch. tha_fbk_epoch is defined as follows:

Variable Name	Data Type	Number Elements	Dims	Sizes	Record Variance	Dimension Variances
tha_fbk_epoch	CDF_EPOCH	1	0		T	

Attribute Name	Data Type	Value
CATDESC	CDF_CHAR	tha_fbk_epoch
FIELDNAM	CDF_CHAR	tha_fbk_epoch
FILLVAL	CDF_EPOCH	31-Dec-9999-23:59:59.999
LABLAXIS	CDF_CHAR	UT
VALIDMIN	CDF_EPOCH	01-Jan-2001 00:00:00.000
VALIDMAX	CDF_EPOCH	31-Dec-2100.23:59:59.999
VAR_TYPE	CDF_CHAR	support_data
VIRTUAL	CDF_CHAR	TRUE
FUNCT	CDF_CHAR	Comp_themis_epoch
COMPONENT_1	CDF_CHAR	tha_fbk_time

Here we see that tha_fbk_epoch is marked as VIRTUAL. The routine to calculate this value is called "comp_themis_epoch", as specified by the FUNCT attribute. The arguments to comp_themis_epoch are given by COMPONENT_0 and COMPONENT_1: tha_fbk_epoch0 and tha_fbk_time. tha_fbk_epoch0 is the POSIX epoch expressed as a CDF_EPOCH value. So comp_themis_epoch combines the POSIX epoch (tha_fbk_epoch0) with a double-precision offset in seconds (tha_fbk_time) to obtain the CDF_EPOCH data values for tha_fbk_epoch.

By using the VIRTUAL, FUNCT, COMPONENT_0 and COMPONENT_1 attributes in this fashion, we can avoid the overhead of storing the timestamps in two different formats, while satisfying the ISTP requirement to provide CDF_EPOCH timestamp data.



Any users with a need to develop their own tools for working with THEMIS CDFs are strongly encouraged to install and gain familiarity with the CDF utilities distributed by Goddard. Many aspects of the THEMIS L1 CDFs (for example, lesser used variables and attributes) are touched on briefly, if at all, in this document for the sake of clarity. The ability to generate a human-readable skeleton table from any CDF file is a very useful self-documenting property of the format.

3. Overview of Frame and Pocket-Level Telemetry

While the intention of the L1 data products is to abstract away many of the details of the THEMIS low-level packet formats, some aspects of the L0 packet data are worth describing here to set the stage for describing the L1 CDF products.

The basic unit of telemetry produced by the THEMIS probes consists of CCSDS "source packets". For THEMIS, each packet has a maximum size of 4096 bytes. Different types of packets are distinguished by their "ApId" (Application Process IDentifier), a field in the 6-byte CCSDS primary header.

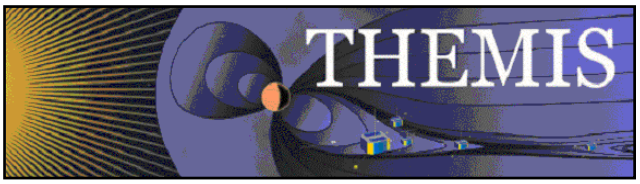
Apids in the range 0x300 to 0x3FF are produced by the Bus Avionics Unit (BAU). Apids 0x400 and 0x401 are used for command packets and memory load packets respectively, directed at the Instrument Data Processing Unit (IDPU). Apids 0x404 through 0x461 represent state of health, housekeeping, and various types of science data sent by the IDPU. Apid 0x7FF is a fill packet used to pad out partially filled transfer frames.

The CCSDS Transfer Frame is the basic unit of communication between the THEMIS probes and the ground. Each transfer frame is 1100 bytes long, and includes a header, trailer, and data area. Packets are stored one after another within the data area of each transfer frame. THEMIS packets are potentially larger than the transfer frame data area, so packets may span multiple frames.

Just as packets are distinguished by their apids, transfer frames are assigned to distinct functions corresponding to eight possible Virtual Channels (VCs). THEMIS uses the following VC assignments:

VC Function
0 Real time engineering data
1 Playback engineering data
2 Real time science data
3 Playback science data
4 Thruster performance playback
5 Not used
6 Events (BAU flight software debug messages)
7 Fill

When a probe to ground contact occurs, the ground station sends several files to the THEMIS Science Operations Center (SOC), one file per virtual channel, containing the telemetry received during that contact. These files are colloquially known as "VC files", and are the starting point for all the processing that follows.



4. Overview of VC->L0 processing.

VC files arrive at the THEMIS SOC in the form of a binary file of "SMEX frames", each of which comprises a 4-byte "sync word", a 10-byte SMEX header, and 160-byte Reed-Solomon code block trailer, surrounding an 1100-byte transfer frame. The first stage of VC file processing is to strip the headers, trailers, and sync words, leaving only the transfer frames. Each transfer frame contains a "spacecraft ID" header field which uniquely identifies the spacecraft from which it came. The value of this field is noted for later use in merging this file's data with the appropriate probe's Level 0 packet archive.

The next step is to extract the CCSDS source packets from the transfer frames. This process makes use of both the transfer frame headers (to find where a new packet starts) and packet headers (to find the expected length of the packet). The fact that THEMIS packets can be longer than a single transfer frame introduces some complications: packets (and even packet headers) may span several transfer frames, so that a given frame may not contain the start of a new packet. The transfer frame contains a trailing 4-byte Command Link Control Word, which must not be mistaken for part of a packet header or packet body. Poor link margins or other sources of interference may result in uncorrectable bit errors, manifesting themselves as missing transfer frames; the packet extraction process must take care to detect such dropouts (by observing the VC sequence count in the transfer frame headers) and discard any partially telemetered packet that might have been in progress at the point of the dropout. Under certain circumstances, race conditions in the flight software telemetry playback modules may result in very old (days or weeks) packet header timestamps as compared to the transfer frame timestamps; severely out of date packets are rejected at this stage as probable playback errors.

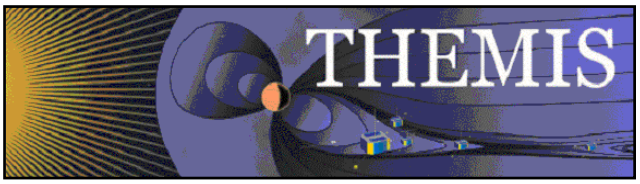
When a packet is successfully extracted from the frame telemetry, the header is checked to see if any of the various packet-level compression algorithms were applied. At this writing (2009-07-10), the following compression algorithms are supported:

Lossless single-channel delta modulation (used for waveforms)
Lossless Huffman coding (used for FFT spectra and particle distributions)
Lossless 3-channel or 6-channel delta modulation (waveforms)
Lossy 8:1 decimation of moment data (apid 0x453)
Lossy hybrid moment compression (truncation plus 63-channel delta modulation)

Any compressed packets are decompressed prior to being written to disk. Whenever possible, internal consistency checks are performed to detect and discard packets with apparent compression errors. Apart from a few FSW bugs early in the mission, most of these types of errors are actually the result of telemetry playback issues alluded to above -- old packet headers can be inadvertently replayed with garbage data, which will usually result in some sort of obvious failure when attempting to decompress the packet.

All packets that were successfully extracted and decompressed from the frame telemetry are then sorted into multiple files, one per apid, which are then sorted by UTC date and (approximately) by header time. (Contiguous extents of packets may have consecutive sequence numbers in their headers, but out-of-order timestamps, usually due to routine adjustments of the spacecraft UTC clock. Sequence number trumps header time when it comes to correct time ordering of packets).

Finally, the extracted, apid- and time-sorted packet files are compared to the corresponding files in the appropriate probe's Level 0 archive. Any duplicate packets (as determined by header time, apid, and sequence numbers...not the data contents) are removed from the new packet telemetry. Whatever remains is merged into the Level 0 archives. The final result is a set of directories, one per UTC day, with each directory containing one time-ordered, overlap-deleted, decompressed packet file per apid.
For detailed information on the structure and interpretation of THEMIS packet data, please refer to the Command and Telemetry (CTM) spreadsheet. The current revision, as of this writing (2009-07-10) is thm_fsw_003_ctm_v4.91.xls.



5. Overview of L0->L1 processing

After incoming telemetry files are processed to L0 packet data, the next step is to create the L1 CDF files using the L0 packets. The L0 data is structured so that all packet files from a given probe on a given date are grouped in a single directory. L0 to L1 processing is performed one directory at a time, whenever it is determined that new packet data has been added to that directory.

Each level 0 packet contains a single timestamp, in the packet header, denoting the number of seconds (a 32 bit quantity) and subseconds (a 16 bit quantity) since the THEMIS mission operations epoch of 2001-01-01 00:00:00 UTC. (All times appearing in packet data are referenced to this epoch.) Individual samples are not time tagged at the packet level, so an important aspect of the L1 processing is to use the packet header time and some data-type-dependent logic to calculate sample time tags. The epoch in the L1 CDF timestamps is also changed (by adding a constant offset) to the POSIX epoch 1970-01-01, which is a convenient choice for plotting the L1 data in IDL.

For apids containing science data (0x410 and above), the packet header contains 4 bytes of instrument-specific configuration data. This typically consists of a number of bit fields denoting sample rates, source selection, scale factors, compression status, mode settings, and other such information. This data is not normally needed by users of the L1 CDFs, but for troubleshooting purposes, the entire 16-byte packet header is stored in the relevant L1 CDF. The L1 CDF variables whose names contain the string "_hed" refer to these packet headers. The packet header times are extracted, converted to the POSIX epoch, stored in the CDF in variables with suffix "_hed_time", and linked to the header data records via the DEPEND_TIME attribute.

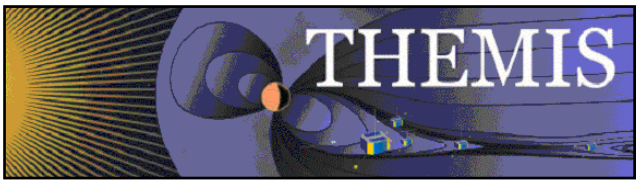
The sample time tag calculations for science data can be divided into two categories: the field instruments (FGM, SCM, and EFI) have their sample timing synchronized to a 1 Hz hardware signal generated by the probe bus and distributed to the IDPU and sensor hardware. The particle instruments (ESA and SST) are synchronized to the probe's spin period via a phase-locked loop implemented in the IDPU, using a hardware "sun pulse" generated by the BAU.

The apid 0x404 IDPU housekeeping data contains a 6-byte ONESECMARK value representing the time of the most recent 1 Hz pulse. Apid 0x305 (from the BAU) contains the time of the last sun sensor crossing, counting from the last BAU reset (so-called "Mission Elapsed Time" or MET). This is combined with another field from BAU apid 0x30c, which gives the UTC offset to be added to convert MET values to the THEMIS epoch 2001-01-01.

The first stage of L1 processing is to extract these values from the 0x404, 0x305, and 0x30c packets, and convert them into a form usable for time tagging by later processing stages. The ONESECMARK data extracted from the 0x404 packets is simply written to a text file and used "as is".

The 0x305 and 0x30c data are combined to yield a text file of sun pulse times, along with spin period and phase error estimates from the BAU ACS processing. However, this data requires further processing to be useful for time tagging. There are several known problems with BAU sun sensor crossing times, ranging from occasional 2 msec glitches to completely missed crossings which can cause significant upsets in the ACS spin period and phase error estimates. Even after correcting for known BAU timing issues, the uncertainty in individual crossing times needs to be averaged out somehow in order to achieve the best possible science. This is accomplished by fitting a "spin model" to the raw sun crossing times. The time period covered by this L0 data set is split into a number of segments so that within each segment, the angular velocity can be modeled as a linear function of time, from which the instantaneous spin period and total amount of rotation are easily obtained. The spin model parameters are stored in the L1 STATE CDF, to be described later in this document

Once the ONESECMARK and spin model data have been processed, each packet file is processed (with the tmtools "pkt2cdf" utility) to a preliminary CDF, containing properly time tagged sample data, little or no



metadata, and generic variable names. These preliminary CDFs are then combined with metadata from the appropriate L1 master CDF, and variables are renamed to conform to the THEMIS conventions. The new L1 CDFs are then copied into the appropriate places on the THEMIS file server, where they can be immediately accessed by the public.

The remainder of this document describes the instrument-specific processing used to produce each type of L1 CDF. The variable names used are from the THEMIS-A master CDFs.

6. BAU, HSK: Probe Housekeeping and State of Health

There are two L1 CDF types pertaining to the probe state of health and housekeeping. The "bau" CDF contains a subset of the telemetryvalues from BAU apids 0x302 (solar array and shunt currents), 0x305 (sun sensor crossing times and ACS spin rate information), and 0x30c (offset between UTC and BAU mission elapsed time).

Each quantity is sampled once per packet. The sample times are presumed to be identical to the packet header times.

The BAU CDF contains the following variables:

From apid 0x302:

tha_bau302_time
tha_bau302_hed
tha_bau302_shuntcurr_raw
tha_bau302_psa1curr_raw
tha_bau302_psa2curr_raw
tha_bau302_psa3curr_raw
tha_bau302_psa4curr_raw
tha_bau302_shuntcurr
tha_bau302_psa1curr
tha_bau302_psa2curr
tha_bau302_psa3curr
tha_bau302_psa4curr

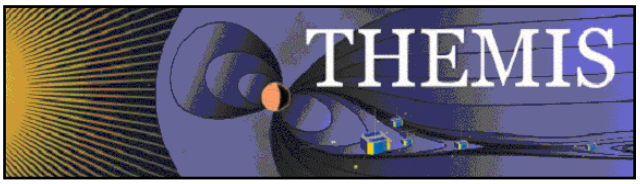
The variables with suffix "_raw" are ADC values as telemetered. The corresponding variables without "_raw" have been converted to amperes.

From apid 0x305:

tha_bau305_time
tha_bau305_hed
tha_bau305_sunpulse_met
tha_bau305_spinper
tha_bau305_sunangle

From apid 0x30c:

tha_bau30c_time
tha_bau30c_hed
tha_bau30c_met
tha_bau30c_utc_offset



The "hsk" CDF contains relatively complete housekeeping and state of health data for the IDPU and instrument suite, from apid 0x404 and 0x406 telemetry. These packets are produced every two to four seconds by the IDPU, but not all these packets are stored -- usually these packets are decimated to a sample rate of approximately one per minute.

Apid 0x404 consists of mostly IDPU status, voltages, currents, and counters. Apid 0x406 is used mostly for instrument state of health. The variable names used in the HSK CDF are nearly identical to the mnemonics used in the CTM spreadsheet; please refer to that document for detailed information about each telemetry point (description, red/yellow limits, unit conversion parameters, bitfield maps, etc.)

FIXME: Add one-line descriptions from CTM spreadsheet to 404 and 406 mnemonics

From apid 0x404:

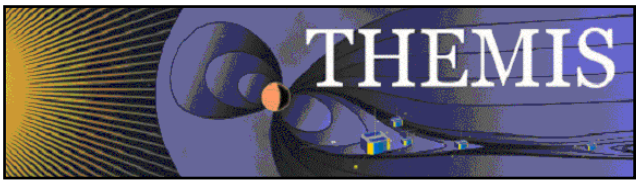
tha_hsk1_time
tha_hsk1_hed

DCB/FSW housekeeping:
tha_idpu_rstctr
tha_ienables
tha_idpu_version
tha_idpu_mode
tha_idpu_fgnd
tha_idpu_errcode
tha_idpu_errdata
tha_idpu_errctr
tha_idpu_cmdreg
tha_idpu_cmdexp
tha_idpu_cmdtot
tha_idpu_loadadr
tha_idpu_dumpadr
tha_io_fpga
tha_io_vc2cnt
tha_io_vc3cnt
tha_io_dcbctl
tha_iscmcal
tha_eepromw
tha_sdram_on
tha_bootrom_pwr
tha_io_auxstat
tha_io_udmastat
tha_itm_enable
tha_itm_errors

Analog housekeeping readbacks:

"_imon" denotes current monitors
"_ivmon" denotes voltage monitors
"p"/"n" denote positive and negative
"va" denotes analog services
"vd" denotes digital services

tha_imon_p10va
tha_imon_n10va



tha_imon_p5va
tha_imon_n5va
tha_imon_n8va
tha_imon_p5vd
tha_imon_p25vd
tha_ignd
tha_ivmon_p10va
tha_ivmon_n10va
tha_ivmon_p5va
tha_ivmon_n5va
tha_ivmon_p8va
tha_ivmon_n8va
tha_ivmon_p5va
tha_ivmon_p25vd
tha_imon_esa28v
tha_ivmon_esa28v
tha_ivmon_act
tha_imon_sma
tha_ivmon_sma
tha_ifge_hsk1
tha_ifge_hsk2
tha_imon_efi_board
tha_imon_efi_x
tha_imon_efi_y
tha_imon_efi_z
tha_imon_idpu
tha_ivmon_idpu
tha_ipcb_fet
tha_idcb_fpgat
tha_idcb_ssrt
tha_idcb_gnd
tha_idcb3v

Power control, one shot actuation, SST attenuation:
tha_ipwrswitch
tha_iactselect
tha_iacttime
tha_iattalloc1
tha_iattalloc2
tha_imtrstat
tha_pwrspare

Solid-state recorder:
tha_issr_mode
tha_issr_eng
tha_issr_quick
tha_issr_survey
tha_issr_burst
tha_itm_fifostat
tha_itm_burst
tha_ieccctrl
tha_ieccadr2
tha_ieccsing
tha_ieccmult

Attitude control:
tha_isuntime
tha_raw_suntime
tha_ispinperiod
tha_iphaserr
tha_idpu_pageadr
tha_onesecmark

From apid 0x406:

tha_hsk2_time
tha_hsk2_hed

EFI analog housekeeping readbacks:
tha_iefi_ibias
tha_iefi_usher
tha_iefi_guard
tha_iefi_braid
tha_ibeb_temp
tha_ispb_temp
tha_iaxb_temp

ESA analog housekeeping readbacks:
tha_iesa_imcp
tha_iesa_emcp
tha_iesa_iswp
tha_iesa_eswp
tha_iesa_igain
tha_iesa_egain
tha_iesa_enables
tha_iesa_pulser
tha_iesa_eswpv
tha_iesa_ehvi
tha_iesa_emcpi
tha_iesa_emcpv
tha_iesa_iswpv
tha_iesa_ihvi
tha_iesa_imcpi
tha_iesa_imcpv

ETC board housekeeping:
tha_ietc_config
tha_ietc_table
tha_ietc_scspot
tha_ietc_covers

SST and attenuator analog housekeeping and status:
tha_isst1_temp
tha_isst2_temp
tha_isst_bias
tha_isst_vref
tha_isst_flux
tha_iatt_state

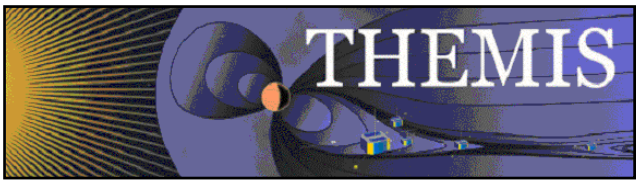
tha_iatt_timer
tha_idhsk_state
tha_idhsk_wait

FGM configuration:
tha_ifgm_config
tha_ifgm_message
tha_ifgm_control
tha_ifgm_phase
tha_ifgm_sampling
tha_ifgm_xy
tha_ifgm_zr

EFI configuration and boom deployment status:
tha_iefi_config
tha_ifbanksel
tha_ifilter1
tha_filter2
tha_ideppair
tha_idepstat
tha_iboombits
tha_idlena
tha_idlenb
tha_ideplimit
tha_ifitmode

Compression and science:
tha_icmp_mode
tha_icmp_seg
tha_icmp_packet
tha_ipb_thresh
tha_iwb_thresh
tha_ipb_evalmax
tha_iwb_evalmax
tha_iscioptions
tha_pb_recstat
tha_wb_recstat
tha_bu_playstat
tha_bu_cmpstat
tha_pb_rdonly
tha_pb_cmprsd
tha_isci_status
tha_iscript
tha_idfbmode
tha_itrgrecord
tha_isstrecord
tha_eval

BAU reported data (from 1 Hz probe status message):
tha_bau_status
tha_ilvps_sctemp
tha_idpu_sctemp
tha_ispb_sctemp
tha_isst_sctemp



tha_idpu_28i
tha_iact_28i
tha_prmy_htri
tha_scnd_htri

7. STATE: Probe State

The "state" CDFs contain information pertaining to each probe's position, velocity, orientation in space, illumination, maneuver status, and regions of interest (e.g. solar wind region, radiation belts, magnetopause crossings). These quantities are sampled at one-minute intervals.

The state CDF also includes spin model parameters, sampled at irregular intervals. A spin model record (a "segment") consists of start and end times, an initial spin period, an angular acceleration term, and a few additional items pertaining to the quality of fit such as number of sun sensor crossings used in the fitting procedure, maximum gap, and maximum deviation of sun crossing times from the modeled times.
The model parameters can be used to calculate the spin phase and spin period at any given instant, the number of spins between two times, and the time of the Nth spin after a given time. These capabilities are necessary for despinning fields data and time-tagging particle moments and distributions.

The data used to construct the spin model comes from the BAU apids 0x305 (which gives the MSSS crossing times relative to the most recent BAU reset), and 0x30c (which records the offset between UTC and the MSSS crossing times). The MSSS crossing times are pre-processed to remove outliers, which sometimes occur due to known BAU hardware and software anomalies. The cleaned-up sequence of MSSS crossing times is then divided into a contiguous set of segments, using an adaptive algorithm which starts a new segment whenever adding a new crossing time would spoil the fit of the data incorporated into the model so far. For each segment, the sun pulse data is modeled by an initial spin period, and a constant angular acceleration. The spin phase is assumed to be 0.0 deg at each segment boundary.

The list of segments produced by the fitting algorithm is adjusted by adding short (usually 1 spin) "buffer" segments to prevent phase discontinuities at the segment boundaries. The final result is a model with no gaps, no overlaps, and no phase discontinuities over the time period covered by the sun pulse data.

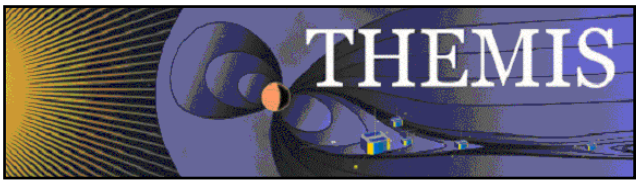
On any given day, there may be several versions of the state file, distinguished by the suffixes V00, V01, V02, V03, etc. Each successive version indicates improvements in the accuracy of ephemeris, attitude, and spin model data. The V00 state file is based on predicted ephemerides produced daily by the THEMIS MOC, and does not include any spin period or spin phase information. The V01 state file is based on the predictive ephemeris, but includes spin period and spin phase information derived from the onboard sun sensor telemetry. The V02 state file is based on a definitive ephemeris calculated from tracking data and fluxgate magnetometer telemetry. V02 also includes spin phase and spin period derived from the probe sun sensor data. It is produced about two weeks after the V00 and V01 state files. The V03 state file incorporates corrections to the spin phase and spin attitude developed from observed and modeled magnetometer data, and may lag the V02 files by several months while the necessary data collection and analysis takes place.

The STATE CDF contains the following variables:

tha_state_time

Probe position and velocity in GEI (a.k.a. ECI) coordinates, true equator and equinox of date:
tha_pos
tha_vel

Probe position and velocity in GSE and GSM coordinate systems (to be released in August 2009):
tha_pos_gse



tha_pos_gsm
tha_vel_gse
tha_vel_gsm

Maneuver flags:
tha_man

Region of interest flags:
tha_roi

Spin axis RA and DEC:
tha_spinras
tha_spindec

Spin axis alpha and beta values: (offset of angular momentum axis from geometric axis)
tha_spinalpha
tha_spinbeta

Spin period and spin phase
tha_spinper
tha_spinphase

The spinper and spinphase variables are filled in during L0->L1 processing using the spin model parameters described in the next set of variables:

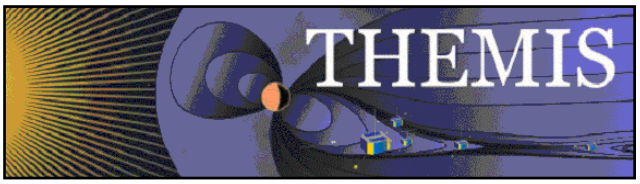
tha_spin_time (segment start time)
tha_spin_tend (segment end time)
tha_spin_spinper (initial spin period)
tha_spin_c (angular acceleration)
tha_spin_nspins (integer number of spins described by this segment)
tha_spin_npts (number of sun crossings used in the modeling process for this segment)
tha_spin_phaserr (maximum difference between observed and modeled sun crossing times)
tha_spin_maxgap (maximum gap between successive sun crossing times for this segment)
tha_spin_correction (angular offset to be applied to spin phase, introduced in V03 processing level)

The spin model variables are sampled at irregular intervals due to the adaptive nature of the modeling process.

8. ESA, SST, MOM: Particle Data

There are three L1 CDF types pertaining to particle data: "esa", containing particle distribution data from the ElectroStatic Analyzer instrument (apids 0x453 through 459), "sst" for the Solid State Telescope particle distribution data (apids 0x45a through 45f), and "mom" for onboard moments calculated from ESA and SST data (apid 0x453).

All particle data is collected on a spin-by-spin basis. The IDPU generates a hardware spin sectoring clock (synchronized to the hardware sun pulse via a phase-locked loop in the IDPU) at a rate of 32 sectors per spin. Particle data collected during one spin is transmitted to the IDPU during the early part of the next spin. If that data happens to start a new packet, the packet header time will be (approximately) the time when the IDPU received the data from the ETC board. That works out to be a little more than one spin after the start of actual data collection began for that sample.



All sample time tags in the ESA, SST, and MOM CDFs are calculated using the spin model parameters calculated from BAU sun sensor telemetry, the packet header time, and the sample cadence (number of spins between samples) for that data type and mode configuration.

Let T_hdr be the packet header time. Using the spin model, it is easy to calculate T_spin (the last time prior to T_hdr when spin phase = 0 degrees), N_zero (a count of spins since some arbitrary zero point), and spinper (the spin period at T_spin). The sample time should represent the midpoint of the data collection interval, or half a spin period before T_spin.

Subsequent sample times are generated by adding N_cadence (the number of spins between samples for this data type and mode) to N_zero, then using the spin model to look up the time and spin period corresponding to that spin number, then correcting by half the spin period to yield the next sample midpoint time.

The mom CDF contains one sample per spin. Each sample consists of a 16 bit ETC status word, a 16 bit spacecraft potential, and 13 32-bit moments for each of four species: ESA ions, ESA electrons, SST ions, and SST electrons respectively:

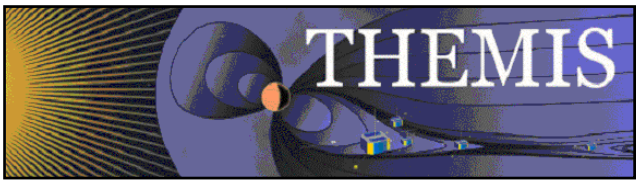
Variable Name	Data Type	Number Elements	Dims	Sizes	Record Variance	Dimension Variances
“tha_mom”	CDF_INT4	1	2	4 13	T	TT
“tha_mom_pot”	CDF_INT2	1	0		T	
“tha_mom_qf”	CDF_UINT2	1	0	T		

tha_mom is the moment data, tha_mom_pot is the spacecraft potential, and tha_mom_qf ("quality flag") is the status word. Bit 0 of the status word indicates whether the ESA data is valid, and bit 1 indicates whether SST is valid.

The 13 moments associated with each species are as follows:

Moment	Description
0	N (Density)
1	NVX (Flux)
2	NVY (Momentum Flux)
3	NVZ
4	MXX
5	MYY
6	MZZ
7	MXY
8	MXZ
9	MYZ
10	QX (Heat)
11	QY
12	QZ

It should be noted that the moment data is subject to lossy compression. One compression method is to decimate the moment packet by a factor of 8, reducing the amount of data contained per packet to 2 samples rather than 16. Another lossy method retains full time resolution, but truncates the least significant 16 bits of most of the moments. (The 9 ESA Ion moment values NVX, NVY, NVZ, MXX,



MYY, MZZ, MXY, MXZ, and MYZ are transmitted with the full 32-bit resolution; all others are truncated to 16 bits).

The distribution data in the ESA and SST CDFs consist of two dimensional arrays of particle counts versus energy bins and angle bins. The instruments can operate in a multitude of different modes, with configurable sample cadences and number of energy and angle bins. Modes are grouped into three categories: "full" distributions with fine resolution in the angle and energy bins, which are sampled every N spins (for mode-dependent values of N). "Reduced" distributions are binned at much coarser resolutions in angles and energies, and may or may not be sampled each spin, depending on the mode. "Burst" distributions are sampled every spin at the full energy and angle resolution.

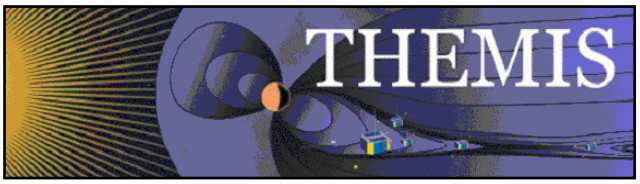
The ESA and SST variables are named with three-letter codes specifying the instrument ('e' for ESA, 's' for SST), the species ('e' for electrons, 'i' for ions), and the distribution type ('f' for full, 'r' for reduced, 'b' for burst). The variable names also distinguish between the various possible angle and energy bin counts for each mode. For example, a reduced distribution of ESA ions at a resolution of 50 angle bins and 24 energy bins from probe A would be named tha_eir_050x24. A full distribution of SST electrons with 32 angle bins and 16 energy bins from probe A would be named tha_sef_032. (All SST distributions have 16 energy bins, so that information is omitted from the SST distribution variable names.)

As of this writing, the following ESA distribution variables are defined:

tha_eif_088x32
tha_eif_176x16
tha_eib_088x32
tha_eib_176x16
tha_eir_001x16
tha_eir_001x32
tha_eir_006x16
tha_eir_006x32
tha_eir_050x24
tha_eir_072x16
tha_eef_088x32
tha_eeb_088x32
tha_eer_001x32
tha_eer_006x16
tha_eer_006x32

The defined SST distribution variables are:
tha_sif_032
tha_sif_064
tha_sif_128
tha_sir_001
tha_sir_006
tha_sef_032
tha_sef_064
tha_sef_128
tha_ser_001
tha_ser_006
tha_seb_064

The values in each distribution (angle,energy) bin are stored in the L1 CDF as they appear in the telemetry: 16 bits of particle counts are reduced to 8 bit quantities via a square root compression algorithm.



Each ESA and SST distribution variable is associated with some support data defining the angle map (phi and theta angles for each angle bin) and energy map (energy level for each energy bin).

Here is an excerpt from an ESA L1 skeleton table showing the support data associated with the tha_eif_088x32 distribution type:

Variable Name	Data Type	Number Elements	Dims	Sizes	Record Variance	Dimension Variances
“tha_eif_088x32”	CDF_UINT1	1	2	88 32	T	TT

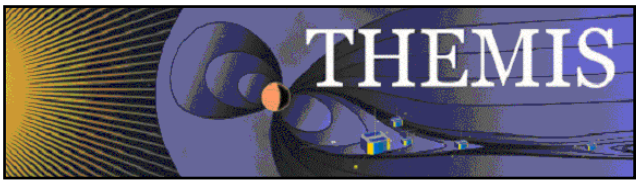
Attribute Name	Data Type	Value
“CATDESC”	CDF_CHAR	"tha_eif_088x32"
“DEPEND_0”	CDF_CHAR	"tha_eif_088x32_epoch"
“DEPEND_1”	CDF_CHAR	"tha_eif_088x32_angleno"
“DEPEND_2”	CDF_CHAR	"tha_eif_088x32_energno"
“DISPLAY_TYPE”	CDF_CHAR	“spectrogram”
“FIELDNAM”	CDF_CHAR	“L1 ESA Ion FDF compressed counts”
“FILLVAL”	CDF_UINT1	255
“FORMAT”	CDF_CHAR	“i6”
“LABL_PTR_1”	CDF_CHAR	“tha_eif_088x32_labl_angle”
“LABL_PTR_2”	CDF_CHAR	"tha_eif_088x32_labl_energy"
“UNITS”	CDF_CHAR	“Ccounts”
“VALIDMIN”	CDF_UINT1	0
“VALIDMAX”	CDF_UINT1	255
“VAR_TYPE”	CDF_CHAR	“data”
“SCALETYP”	CDF_CHAR	“log”
“DEPEND_TIME”	CDF_CHAR	“tha_eif_088x32_time”
“DEPEND_EPOCH0”	CDF_CHAR	"tha_eif_088x32_epoch0"
“DEPEND_SWEEP_MODE”	CDF_CHAR	"tha_eif_088x32_sweep_mode"
“DEPEND_ANGLE_MAP”	CDF_CHAR	"tha_eif_088x32_angle_map"
“DEPEND_ENERGY_TABLE”	CDF_CHAR	"tha_ei_x32_energy_table"
“DEPEND_DENERGY_TABLE”	CDF_CHAR	"tha_ei_x32_denergy_table"
“DEPEND_PHI_MAP”	CDF_CHAR	"tha_ei_088_phi_map"
“DEPEND_DPHI_MAP”	CDF_CHAR	"tha_ei_088_dphi_map"
"DEPEND_THETA_MAP"	CDF_CHAR	"tha_ei_088_theta_map"
"DEPEND_DTHETA_MAP"	CDF_CHAR	"tha_ei_088_dtheta_map"

*RV values were not requested.

9. TRG: Burst Mode Triggers

The "trg" CDF contains a set of quantities, calculated by the IDPU, which are used as a heuristic method of deciding whether to reconfigure the instrument suite for high-resolution burst mode data acquisition. Each sample consists of 8 8-bit unsigned integers. The sample period can range from 1 to 256 seconds between samples.

Variable Name	Data Type	Number Elements	Dims	Sizes	Record Variance	Dimension Variances
"tha_trg"	CDF_UINT1	1	1	8	T	T



The interpretation of the 8 components of each sample is subject to change, as the flight software is refined to improve the triggering heuristics. Some of the components are related to particle fluxes, while others are keyed to fluctuations in the E-field or B-field data.

Trigger data is telemetered in apid 0x451.

10. FGM: Fluxgate Magnetometer Data

The "fgm" CDF contains data from the fluxgate magnetometer instrument. Each sample is a 3-vector of field strength, measured in instrument ADC units and coordinates. There are three data variables, each representing a different time resolution:

Variable Name	Sample rate	Notes
tha_fgh	128 Hz	High time resolution data from apid 0x461
tha_fge	8 Hz	Medium time resolution data from apid 0x405
tha_fgl	4 Hz (variable)	Low time resolution data from apid 0x460

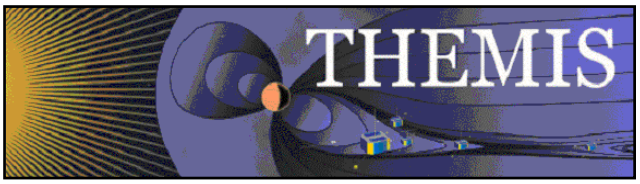
Variable Name	Data Type	Number Elements	Dims	Sizes	Record Variance	Dimension Variances
"tha_fge"	CDF_INT4	1	1	3	T	T

The FGM instrument produces data with 24 bits of resolution per vector component. However, due to telemetry volume constraints, only 16 bits of each component are telemetered. The sensitivity range for FGH and FGL data is usually adjusted during the orbit so the instrument is in its least sensitive range (most significant 16 bits transmitted) near perigee, where the field is strong. Around apogee, where the field is considerably weaker, the most sensitive range (least significant 16 bits transmitted) is usually selected. FGE data (which is used for determination of probe attitude and position) is always in the least sensitive range (where the most significant 16 bits are transmitted).

The L0 packet header data includes the range setting, which is used to shift the sample data back to the native 24-bit ADC range. A range-dependent offset is applied to compensate for a systematic error that would otherwise occur due to the semantics of truncating twos-complement binary data; since this operation always rounds toward negative infinity, the leftmost bit shifted into the 24-bit final value is set to 1 rather than zero. This effectively centers the 24-bit shifted value in the window of possible values that could have produced the 16-bit data that was telemetered.

Since CDF does not support a 24-bit data type, the sample data are stored as 32-bit CDF_INT4 data (with appropriate sign extension from the 24-bit reconstructed samples).

Since the range information (and for FGL, the sample rate) is only transmitted once in each packet header, it is possible for the sensitivity range or sample rate to change in the middle of a packet in a way that is not accounted for during the range correction and time stamping of the L1 CDF data. It may be possible to eliminate these spikes or distortions by taking advantage of the fact that all configuration changes take effect in sync with the onboard 1 Hz hardware pulse -- if the headers indicate a configuration change between two consecutive packets, the L0->L1 processing software could conceivably locate the jump in magnitude or sample rate and correct it in the L1 CDF. This capability is not currently implemented, so spikes are seen in the L1 and L2 data whenever the instrument configuration changes.



11. SCF, SCP, SCW: Search Coil Magnetometer (SCM) Data

There are three types of CDFs which contain data from the Search Coil Magnetometer instrument:

scf SCM fast survey (apid 0x444)
scp SCM particle burst (apid 0x448)
scw SCM wave burst (apid 0x44c)

Variable Name	Data Type	Number Elements	Dims	Sizes	Record Variance	Dimension Variances
"tha_scw"	CDF_INT2	1	1	3	T	T

The variables tha_scf, tha_scp, and tha_scw contain the 16-bit SCM sample data. In wave burst mode, the SCM can be configured to send additional diagnostic information. If present, this data is stored in a variable "tha_scw_dq". Each DQ sample is a 2-element array of 16-bit values, representing ExB and E dot B respectively:

Variable Name	Data Type	Number Elements	Dims	Sizes	Record Variance	Dimension Variances
"tha_scw_dq"	CDF_INT2	1	1	2	T	T

12. FBK: Digital Fields Board Filter Bank Data

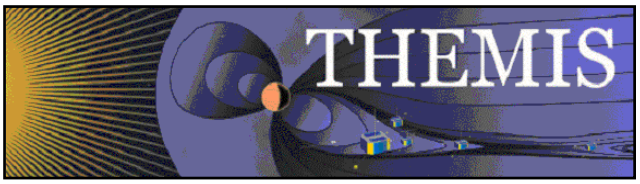
The Digital Fields board contains two banks of digital filters, each covering six frequency bands. Each filter bank can be configured to use a different input source. There is also a high frequency (AKR band) channel dedicated to one of the EFI boom pairs. The peak and average signals in the AKR band are transmitted along with the two filter bank arrays at each sample time. They are defined in the FBK CDF as follows:

Variable Name	Data Type	Number Elements	Dims	Sizes	Record Variance	Dimension Variances
"tha_fb1"	CDF_UINT1	1	1	6	T	T
"tha_fb1_src"	CDF_UINT1	1	0		T	
"tha_fb2"	CDF_UINT1	1	1	6	T	T
"tha_fb2_src"	CDF_UINT1	1	0		T	
"tha_fbh"	CDF_UINT1	1	1	2	T	T

The 6 elements of each filter bank variable roughly correspond to enter frequencies of 2 kHz, 512 Hz, 128 Hz, 32 Hz, 8 Hz, and 2 Hz. The two elements of tha_fbh (the AKR channel) represent the peak and average signal, respectively.

The source selections for fb1 and fb2 are as follows:

Source	Input Channel
0	V1
1	V2
2	V3



3	V4
4	V5
5	V6
6	E12DC
7	E34DC
8	E56DC
9	SCM1
10	SCM2
11	SCM3
12	E12AC
13	E34AC
14	E56AC
15	undefined
16	EDCxB
17	EDCdotB
18	SCMxB
19	SCMdotB
20	EACxB
21	EACdotB

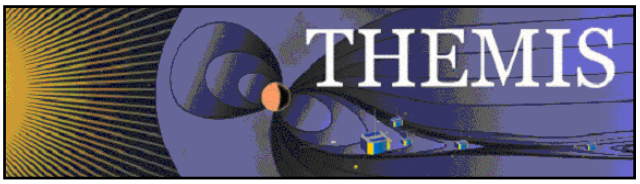
The values stored in the L1 FBK CDF are 8-bit quantities, compressed from the 16-bit outputs of the filter banks.

13. FFP_16, FFP_32, FFP_64, FFW_16, FFW_32, FFW_64: Digital Fields Board Spectra (FFT processors)

The Digital Fields Board contains a set of four FFT processors, which are enabled during wave bursts or particle bursts. The FFT processors can be configured to produce spectra with 16, 32, or 64 frequency bins. Each combination of burst mode and frequency bin count is stored in its own CDF type, yielding the following list of data types:

ffp_16
ffp_32
ffp_64
ffw_16
ffw_32
ffw_64

Each FFT processor can be configured to use a different input source selection, following the same indexing scheme as FBK (described above).



Using ffp_16 as an example, the CDF variables are defined as follows:

Variable Name	Data Type	Number Elements	Dims	Sizes	Record Variance	Dimension Variances
"tha_ffp_16"	CDF_UINT1	1	2	4 16	T	TT
"tha_ffp_16_src"	CDF_UINT1	1	1	4	T	T

Here we see that each sample of tha_ffp_16 contains four 16-bin spectra. tha_ffp_16_src is a 4-element array giving the input source selections for each of the FFT processors.

The spectral data is stored in an 8-bit format, compressed from the 16-bit output of the FFT processors.

14. FIT: Onboard Spin Fits

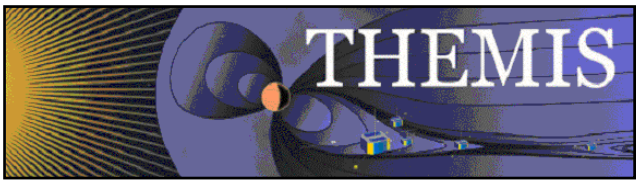
The IDPU monitors a subset of the FGM and EFI data as it is collected, and can perform a fitting procedure to estimate the (despun) ambient field strength and direction, producing one sample per spin. This data is transmitted in apid 0x410, and appears in the L1 "fit" CDF:

Variable Name	Data Type	Number Elements	Dims	Sizes	Record Variance	Dimension Variances
"tha_fit"	CDF_REAL4	1	2	2 5	T	TT
"tha_fit_code"	CDF_UINT1	1	1	2	T	T
"tha_fit_npts"	CDF_UINT1	1	1	2	T	T

Each sample of the tha_fit can be thought of as a pair of 5-element arrays -- one array for E, the other for B. The components of each 5-element array are interpreted as A (X component), B (Y component), C (vector magnitude projected onto spin plane), sigma (goodness of fit), and Z_Avg (estimate of the field component in the Z-axis direction). A and B are scaled to the range [-1, 1], and give the direction of the field vector projected onto the spin plane. C and Z_avg are in ADC units for the EFI or FGM, as appropriate. The spin fit data are in a "pseudo-DSL" coordinate system -- despun, but not necessarily aligned perfectly with the geometric or angular momentum Z axis.

The fit_npts variable gives the number of points (up to 32) that were actually used in the fit. The IDPU can discard outliers as part of its fitting algorithm.

The fit_code variable encodes some additional information about the E and B fits. It is possible for a spin fit packet to contain E-only, B-only, or E and B fits, possibly intermingled in the same packet. A value of 0 in the fit_code variable indicates that no data for that field species was telemetered for that sample, and the data should not be used. For the EFI fits, the fit_code also specifies which spin plane boom pair (E12 or E34) was used by the spin fitter. Downstream processing (L2 and beyond) needs to check the EFI fit codes so that the correct boom lengths and coordinate system are used for coordinate transforms and calibrations.



15. EFF, EFP, EFW, VAF, VAP, VAW, VBF, VBP, VBW: EFI Waveforms

There are several CDF types containing waveform data from the Electric Field instrument (EFI). Voltage group A and voltage group B both contain waveform data for each of the 6 EFI sensor booms. The two groups can be configured separately, to support different sample rates, or AC versus DC coupling configurations. Each sample is a 6-element array of 16-bit ADC values. At the moment only voltage group A is telemetered. The third group represents E-field waveforms, where each sample is a 3-element array of 16-bit ADC values, representing the voltage difference across the three pairs of sensor booms.

Within each group, the file and variable names distinguish between fast survey, wave burst, and particle burst modes:

CDF Type	Notes
eff	E-fields, fast survey (apid 0x443)
efp	E-fields, particle burst (apid 0x447)
efw	E-fields, wave burst (apid 0x44b)
vaf	Voltage group A, fast survey (apid 0x441)
vap	Voltage group A, particle burst (apid 0x445)
vaw	Voltage group A, wave burst (apid 0x449)
vbf	Voltage group B, fast survey (apid 0x442)
vbp	Voltage group B, particle burst (apid 0x446)
vbw	Voltage group B, wave burst (apid 0x44A)

The E-field variables are defined as follows:

Variable Name	Data Type	Number Elements	Dims	Sizes	Record Variance	Dimension Variances
"tha_eff"	CDF_INT2	1	1	3	T	T

Here the three components of each sample are E12, E34, and E56, respectively.

The boom voltage variables are defined similarly:

Variable Name	Data Type	Number Elements	Dims	Sizes	Record Variance	Dimension Variances
"tha_vaf"	CDF_INT2	1	1	6	T	T

The six components of each VAF sample are V1, V2, V3, V4, V5, and V6.

The wave burst E-fields CDF contains an additional variable that may be present in this mode:

Variable Name	Data Type	Number Elements	Dims	Sizes	Record Variance	Dimension Variances
"tha_efw_dq"	CDF_INT2	1	1	2	T	T

The two components of efw_dq represent SCMdotB and SCMxB, respectively.