SPEDAS training session by courtesy of Arase PWE and MGF team – beginners' course–

M. Teramoto[1], M. Shoji[1], S. Matsuda[2], T. Hori[1], T. F. Chang[1], S. Kurita[1], K. Keika[3], Y. Miyoshi[1]

[1] ERG Science Center, ISEE, Nagoya University
[2] Nagoya University
[3] The University of Tokyo

Goal of this training

- To get familiar with how to download, read, and plot ERG satellite data.
  - Download and plot ERG satellite orbit data
  - Download and plot the MGF and PWE/OFA l2 and PWE provisional CDF data of ERG satellite
  - Create and plot multi-tplot variables

  - Download and plot other scientific instrument data of ERG satellite(Additional)
  - Combine other satellite/ground data with ERG satellite data (Additional)
  - Coordinate transformation from satellite coordinates (Additional)

Space Physics Environment Data Analysis Software (SPEDAS)

Data repository

CDF

THEMIS

CDF

Van Allen Probes

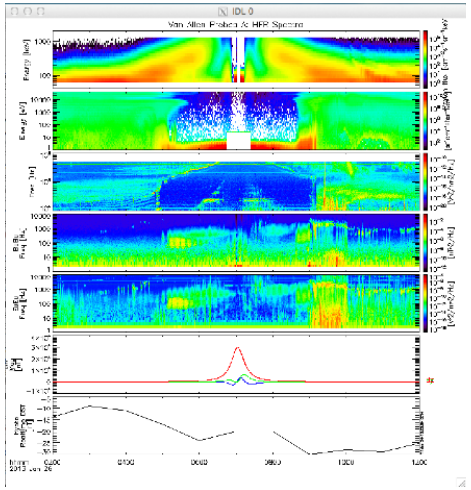
CDF

MMS

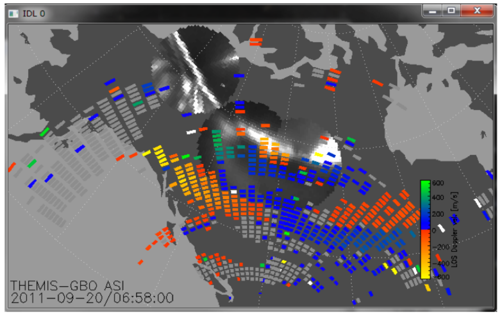
CDF

IUGONET

SPEDAS framework



Each project develops and releases APIs to automatically download and read each project data



CDF

ERG satellite data

CDF

ERG-related ground observation network data

CDF
HDF?

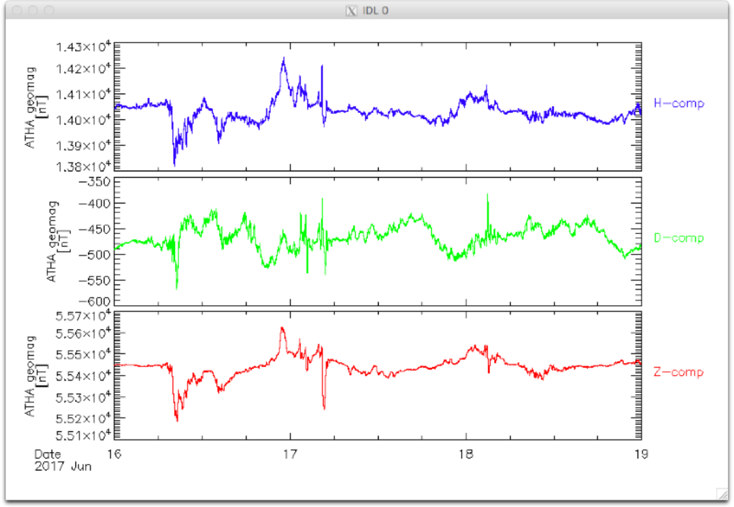
ERG simulation / modeling data

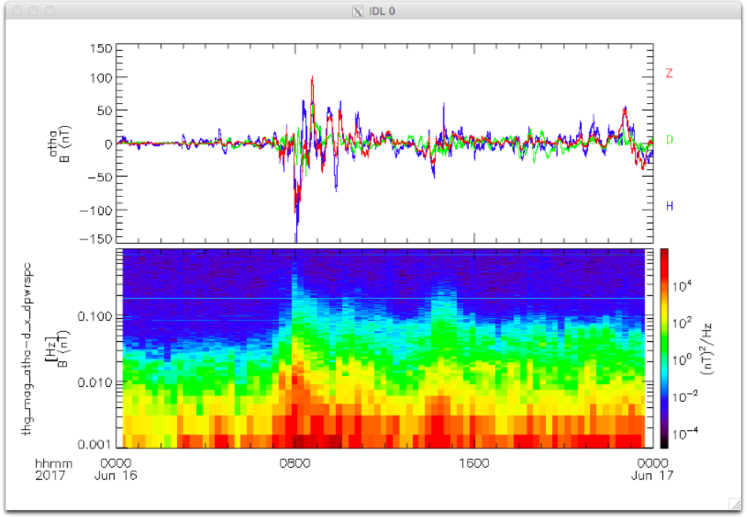
ERG-SC develops data archives and APIs to read the data from SPEDAS

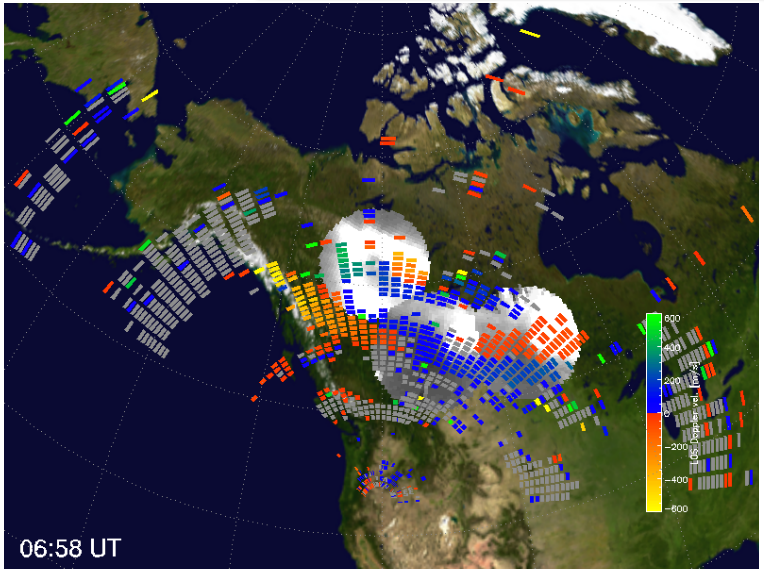
What can you do with SPEDAS?

ERG
Exploration of energization and Radiation in Geospace

- time-series plots
- filtering of data
- frequency analysis
- mapping to the ground maps
- …







Preparation: before using SPEDAS

A few basics of IDL before entering SPEDAS…

- Insert a comma (,) between a command, its arguments, and keywords.

```
IDL> tplot , 1 , title='New plot'
```

- A string is expressed as a text sandwiched by delimeters (') or (").

```
IDL> print, 'This is a text.'
```

- An array is expressed as comma-separated elements that are bracketed.

```
IDL> arr1 = [ 2, 3, 4, 5 ]
IDL> string_arr1 = [ 'text1', 'text2', 'text3' ]
```

- Typical errors beginners often encounter:

```
% Attempt to call undefined procedure: '????'.
```

→ command/routine name (????) is misspelled.

```
% Syntax error.
```

→ , ' ( ) [ ] is missing or mismatched in most cases.

- **Use Up arrow key to reuse previously typed commands**. You can edit them with Left/Right arrow, Backspace keys and execute!

Preparation:
Initial setting of SPEDAS for ERG data

Clear out IDL's variables and compiled files

```
IDL> .full_reset_session
```

Initialize

```
IDL> erg_init, remote_data_dir =
'https://ergsc.isee.nagoya-
u.ac.jp/data/ergsc_training/nagoya_201803/'
```

Setup the username and password to access the ERG Data

```
ERG> uname = ‘?????'

ERG> pass = ‘???????'
```

memo

ERG satellite orbit data

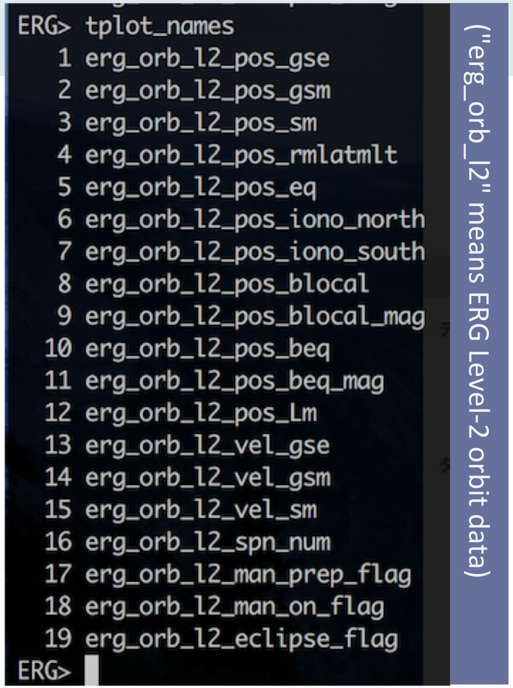
Orbit data:
Set time range and load ERG orbit data

Setup the time range (‘YYYY-MM-DD/hh:mm:ss’)

```
ERG> timespan,'2017-03-28/00:00:00’,3,/day
```

Load orbit data

```
ERG> erg_load_orb
ERG> tplot_names
```



```
ERG> tplot_names
   1 erg_orb_l2_pos_gse
   2 erg_orb_l2_pos_gsm
   3 erg_orb_l2_pos_sm
   4 erg_orb_l2_pos_rmlatmlt
   5 erg_orb_l2_pos_eq
   6 erg_orb_l2_pos_iono_north
   7 erg_orb_l2_pos_iono_south
   8 erg_orb_l2_pos_blocal
   9 erg_orb_l2_pos_blocal_mag
  10 erg_orb_l2_pos_beq
  11 erg_orb_l2_pos_beq_mag
  12 erg_orb_l2_pos_Lm
  13 erg_orb_l2_vel_gse
  14 erg_orb_l2_vel_gsm
  15 erg_orb_l2_vel_sm
  16 erg_orb_l2_spn_num
  17 erg_orb_l2_man_prep_flag
  18 erg_orb_l2_man_on_flag
  19 erg_orb_l2_eclipse_flag
ERG>
```

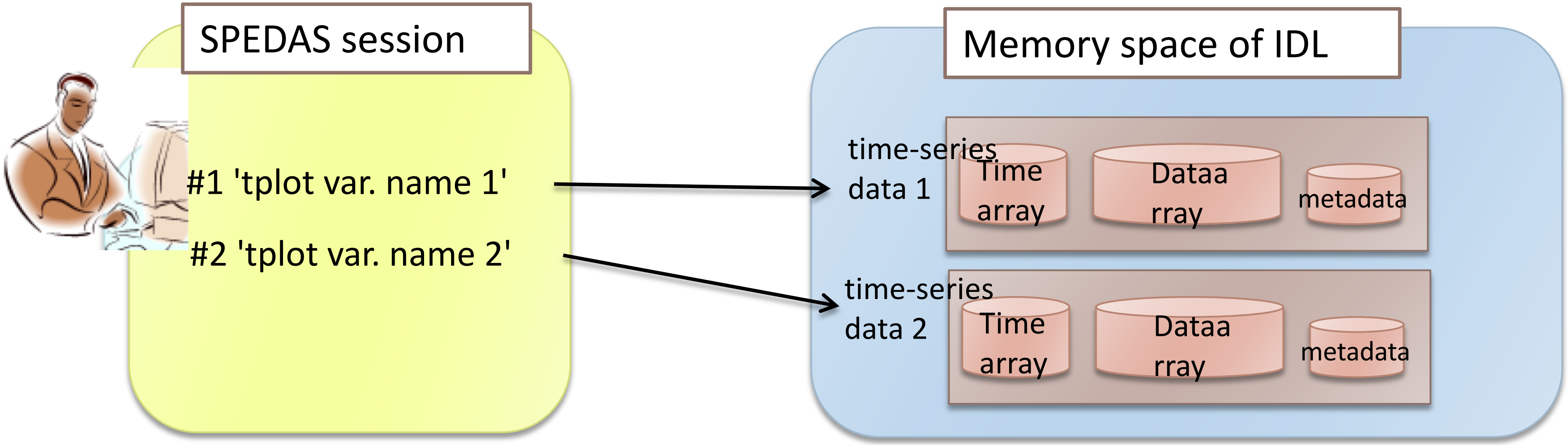
("erg_orb_l2" means ERG Level-2 orbit data)

(*) Using the IGRF model

pos_gse/gsm/sm	s/c position [Re] in GSE, GSM, SM coordinates
pos_rmlatmlt	Radial distance [Re], magnetic latitude [deg], local time [hr] of s/c position
pos_eq	s/c position mapped to the magnetic equator
pos_iono_north/south	Geographic latitude and longitude [deg] of s/c footprints at 100 km altitude in the northern/southern hemisphere
pos_blocal / blocal_mag	model B-field vector (blocal) and B-field strength (blocal_mag) [nT] at s/c position
pos_beq / beq_mag	model B-field vector (beq) and B-field strength (beq_mag) [nT] at s/c position mapped to the magnetic equator
pos_Lm	McIlwain's L-parameter of s/c position for pitch angles of 90, 60, and 30 deg
vel_gse/gsm/sm	s/c orbital velocity [km/s] in GSE, GSM, and SM
man_prep/man_on/eclipse_flag	flag for maneuver preparation (man_prep), maneuver on/off (man_on), and solar eclipse (eclipse)

tplot variable as the primary data model

- ’erg_orb_l2_pos_???' in prev. page is called ***tplot variable***.
- ”tplot variables" bind an indexed name-string to a data structure on IDL containing time-series data with metadata.



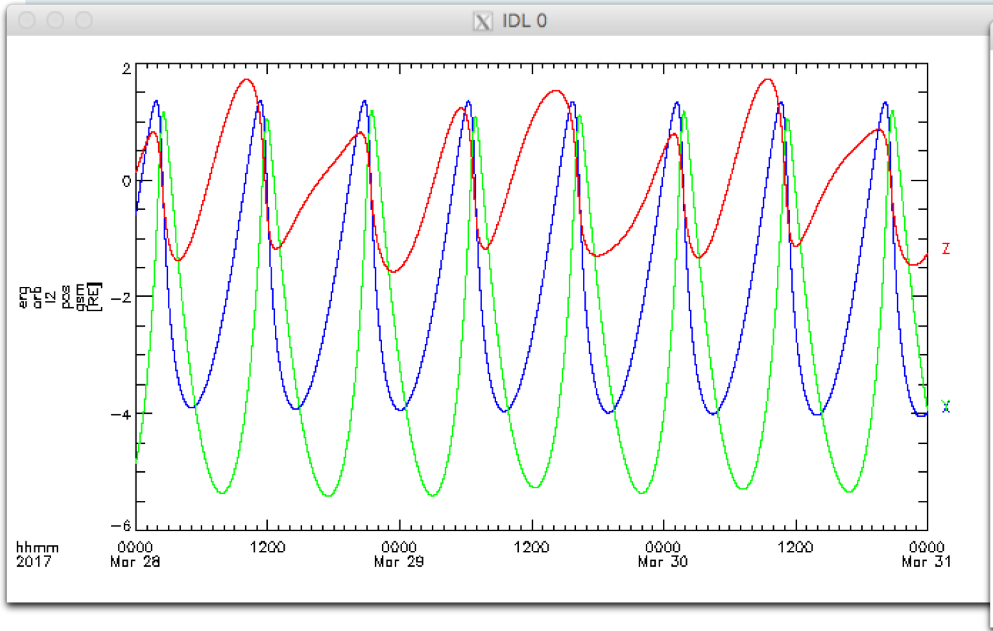
Orbit data:
Definitive orbit as a time series plot: *tplot, tplotxy*

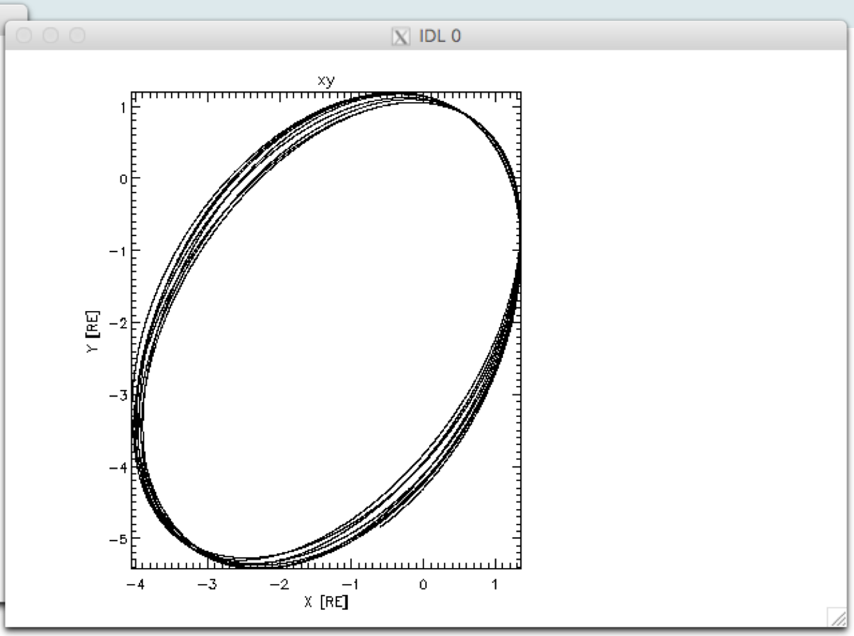
Plot orbit time series data

```
ERG> tplot, 'erg_orb_l2_pos_gsm'
```

Plot orbit data in the X-Y plane.

```
ERG> tplotxy, 'erg_orb_l2_pos_gsm'
```





Set a date/time range: *timespan*

```
ERG> timespan, timestr, N, option
```

`timestr` : a string expressing a particular date/time in UTC in the format of 'yyyy-mm-dd/hh:mm:ss'

`N` : number of time length (Default: 1)

`option` : unit (/day, /hour, /min, /sec,   Default: /day)

For 1 day from 2017-03-28/00:00:00 UTC

```
ERG> timespan, '2017-03-28'
```

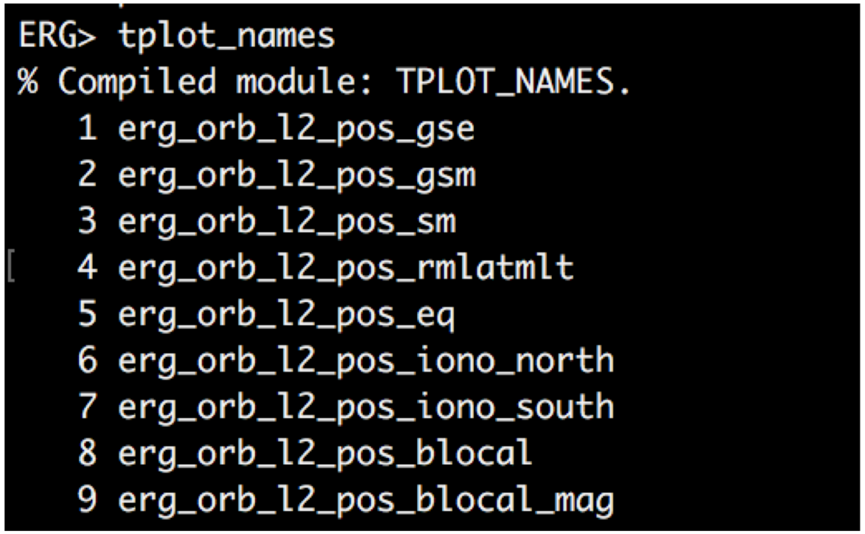
For 10 min from 2017-03-28/01:31:41 UTC

```
ERG> timespan, '2017-03-28/01:31:41', 10, /min
```

Listing tplot data & viewing the content: *tplot_names* *print_tinfo*

```
ERG> tplot_names


ERG> print_tinfo, ’erg_orb_l2_pos_gse'
```



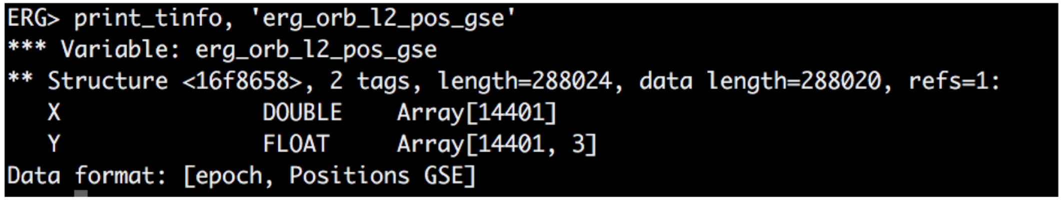
```
ERG> tplot_names
% Compiled module: TPLOT_NAMES.
   1 erg_orb_l2_pos_gse
   2 erg_orb_l2_pos_gsm
   3 erg_orb_l2_pos_sm
   4 erg_orb_l2_pos_rmlatmlt
   5 erg_orb_l2_pos_eq
   6 erg_orb_l2_pos_iono_north
   7 erg_orb_l2_pos_iono_south
   8 erg_orb_l2_pos_blocal
   9 erg_orb_l2_pos_blocal_mag
```

.

.

.

All tplot variables are listed with unique index numbers



```
ERG> print_tinfo, 'erg_orb_l2_pos_gse'
*** Variable: erg_orb_l2_pos_gse
** Structure <16f8658>, 2 tags, length=288024, data length=288020, refs=1:
   X               DOUBLE    Array[14401]
   Y               FLOAT     Array[14401, 3]
Data format: [epoch, Positions GSE]
```

The actual data structure bound to tplot variable is shown.

X: time array containing time labels in decimal UNIX time

Y: data array, in this case, a 2-D array of time x 3-components

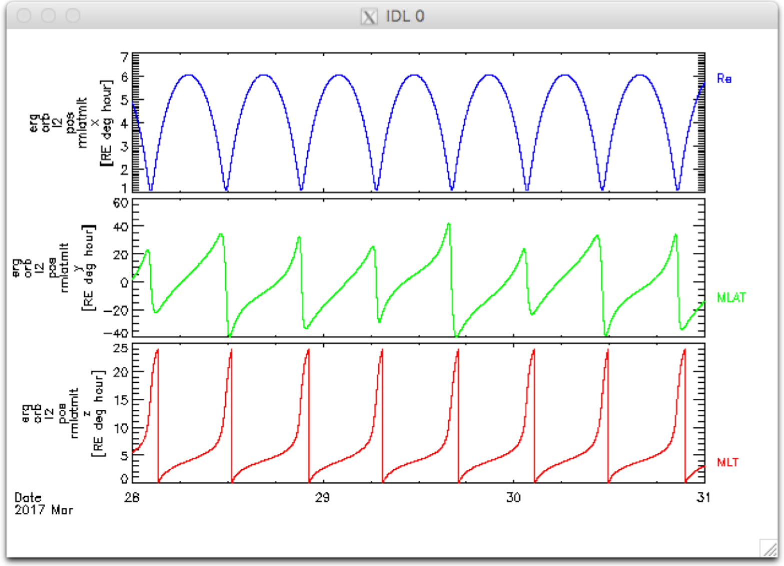
*print_tinfo can be used only for the tplot variables directly created from CDF files.

Separate a tplot variable with vector data: *split_vec*

```
ERG> split_vec, 'erg_orb_l2_pos_rmlatmlt'
STORE_DATA(264): Creating tplot variable: ...
STORE_DATA(264): Creating tplot variable: ...
STORE_DATA(264): Creating tplot variable: ...

ERG> tplot, 'erg_orb_l2_pos_rmlatmlt_?'
```

`split_vec` takes a tplot variable with vector or array data to create new tplot variables containing each component of the vector/array data.


IDL 0
erg orb l2 pos rmlatmlt x [RE deg hour]
7
6
5
4
3
2
1
Re
erg orb l2 pos rmlatmlt y [RE deg hour]
60
40
20
0
−20
−40
MLAT
erg orb l2 pos rmlatmlt z [RE deg hour]
25
20
15
10
5
0
MLT
Date
2017 Mar
28
29
30
31

Orbit data:
Insert orbit values below a time-series plot

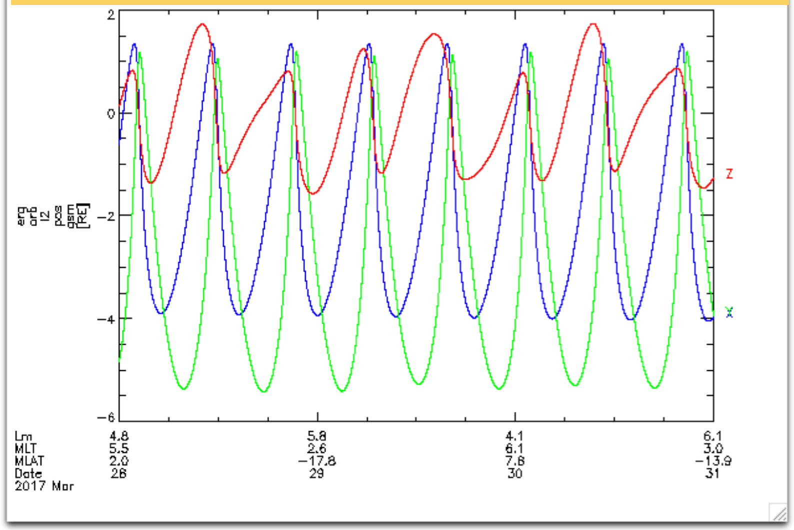
```
ERG> set_erg_var_label
ERG> tplot      ;just type "tplot" to replot the previous panels
```

Using subroutine to add labels (Lm,MLT,MLAT) with time

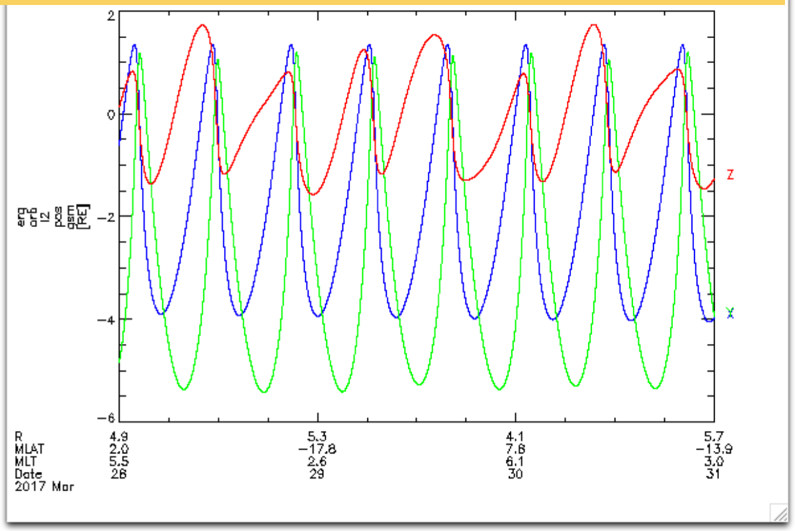
```
ERG> split_vec, 'erg_orb_l2_pos_rmlatmlt'
ERG> options, 'erg_orb_l2_pos_rmlatmlt_x', ytitle='R'
ERG> options, 'erg_orb_l2_pos_rmlatmlt_y', ytitle='MLAT'
ERG> options, 'erg_orb_l2_pos_rmlatmlt_z', ytitle='MLT'
ERG> tplot_options, var_label=['erg_orb_l2_pos_rmlatmlt_z',
'erg_orb_l2_pos_rmlatmlt_y', 'erg_orb_l2_pos_rmlatmlt_x']
ERG> tplot
```

An example to add new labels (R,MLAT,MLT)

(Lm, MLAT, MLT)



(R, MLT, MLAT)



Decorate the plot panel: *options, tplot_options*

Sets options for **each tplot variable.**

```
options, varname, option1='…', option2='…', …
   varname: tplot variable name (wildcards accepted)
   option?: name of tplot variable attribute
```

Sets **global options** for the "tplot" routine.

```
tplot_options, toption1='…', toption2='…', …
   toption?: global options to be set
```

```
ERG> tplot_options, /help
```

Show the current tplot options

memo

Scientific instrument data of ERG satellite: MGF and PWE data

SPEDAS training session, ERG wave data WS @ISEE March 28, 2018

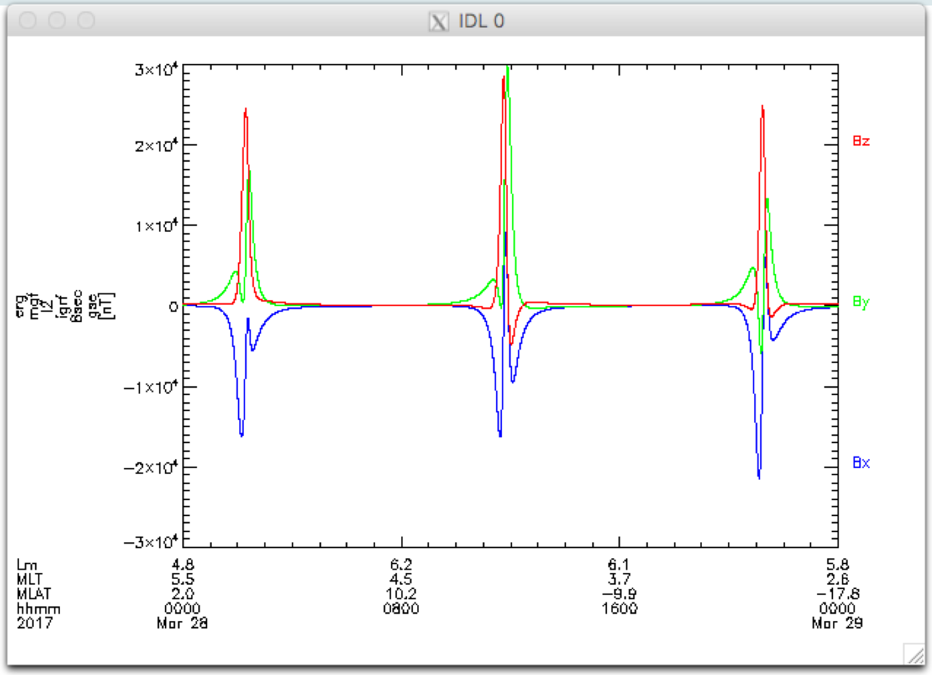
Onboard instrument data: MGF L2 data

Reset time range (No extra keyword: one day)

```
ERG> timespan, '2017-03-28'
```

Load Magnetic field data

```
ERG> erg_load_mgf, uname=uname,pass=pass
ERG> tplot,'erg_mgf_l2_mag_8sec_gse'
```



Change the vertical scale of a plot: *ylim*

ylim, *varname*, *ymin*, *ymax*, *logflag*

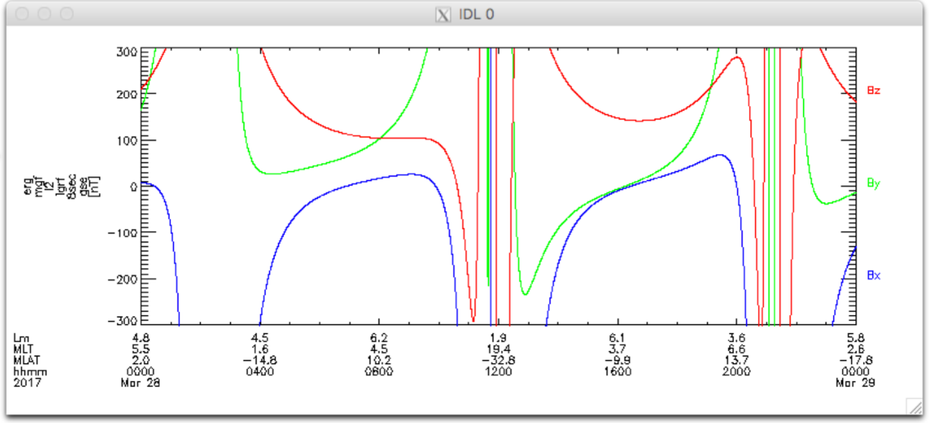
`varname` : variable name(s)

`ymin/ymax` : lower/upper limit along vertical axis
set both to 0 (zero) for plotting with auto-scale

`logflag` : set 0 (zero) for plotting on a linear scale, or 1 for a log scale

```
ERG> ylim, 'erg_mgf_l2_mag_8sec_gse', -300,300,  0
ERG> tplot, 'erg_mgf_l2_mag_8sec_gse'
```

Zoomed in a more limited range in the vertical scale.



Tips:

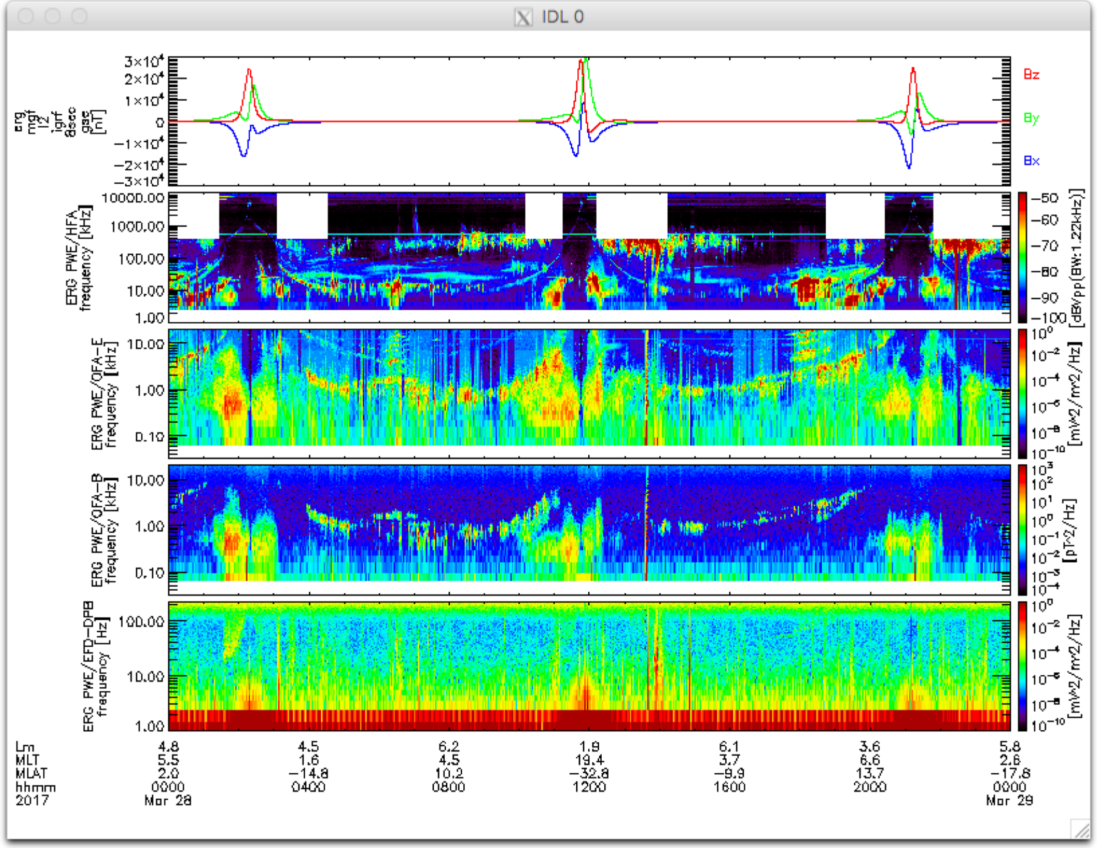
Putting 0 for both ymin and ymax sets the y range to auto-scale.

```
ERG> ylim, 'erg_mgf_pr_mag_8sec_gse', 0, 0
```

Onboard instrument data:
PWE and MGF data

Load PWE data

```
ERG> erg_load_pwe_pre, uname=uname, pass=pass
ERG> tplot, [ 'erg_mgf_l2_mag_8sec_gse' , 'erg_pwe_pre_HFA-merged' ,
'erg_pwe_pre_OFA_E_spectra_132' , 'erg_pwe_pre_OFA_B_spectra_132' ,
'erg_pwe_pre_EFD_DPB_spectra' ]
```



Change the time range of a plot: *tlimit*

Select a time period by mouse-clicks on the plot window

```
ERG> tlimit
```

Specify a time period explicitly

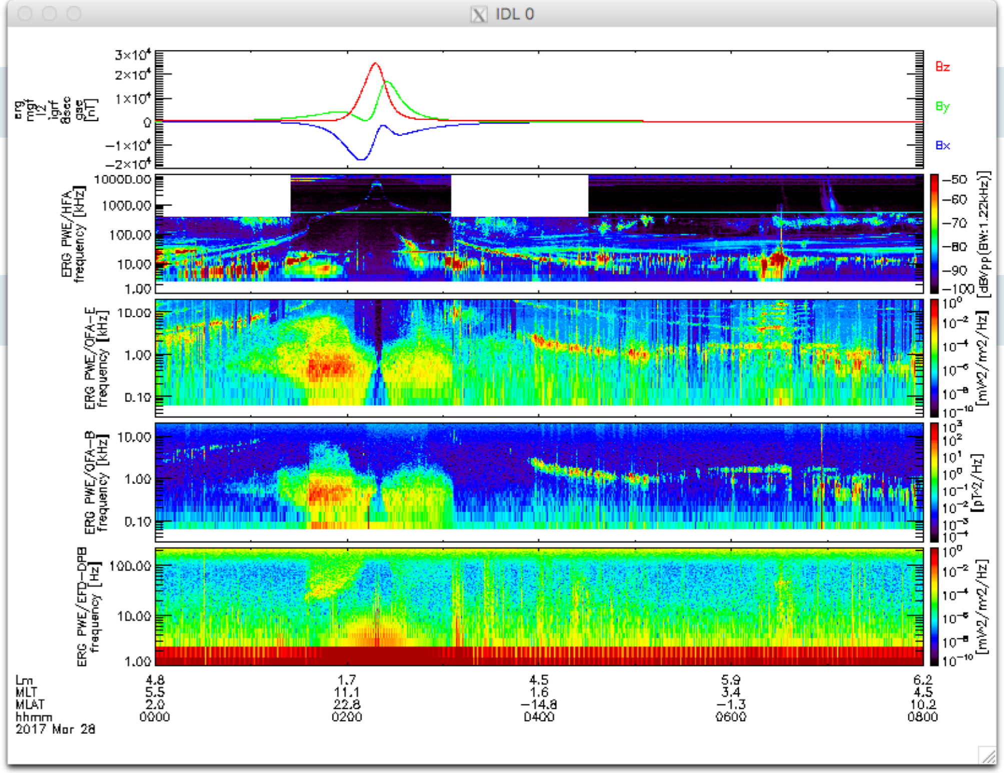
```
ERG> tlimit, '2017-03-28/00:00' , '2017-03-28/08:00'
```

Back to the last plot period

```
ERG> tlimit, /last
```

Restore the original plot period that was set by `timespan`

```
ERG> tlimit, /full
```

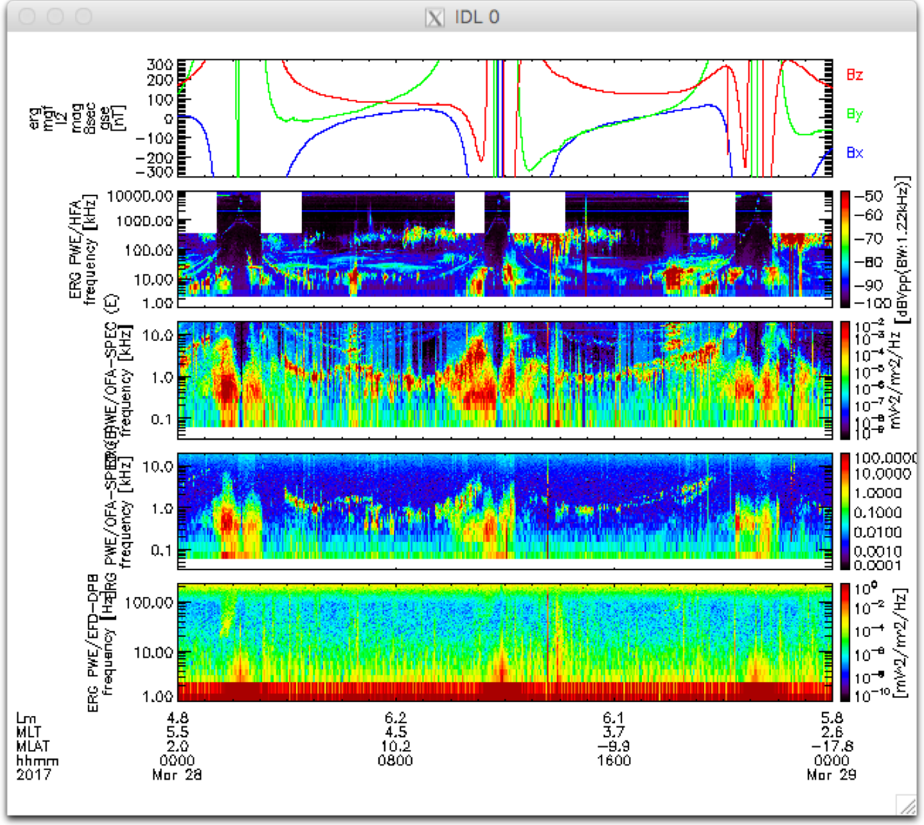


Onboard instrument data:

PWE/OFA L2 data

Load PWE/OFA L2 data

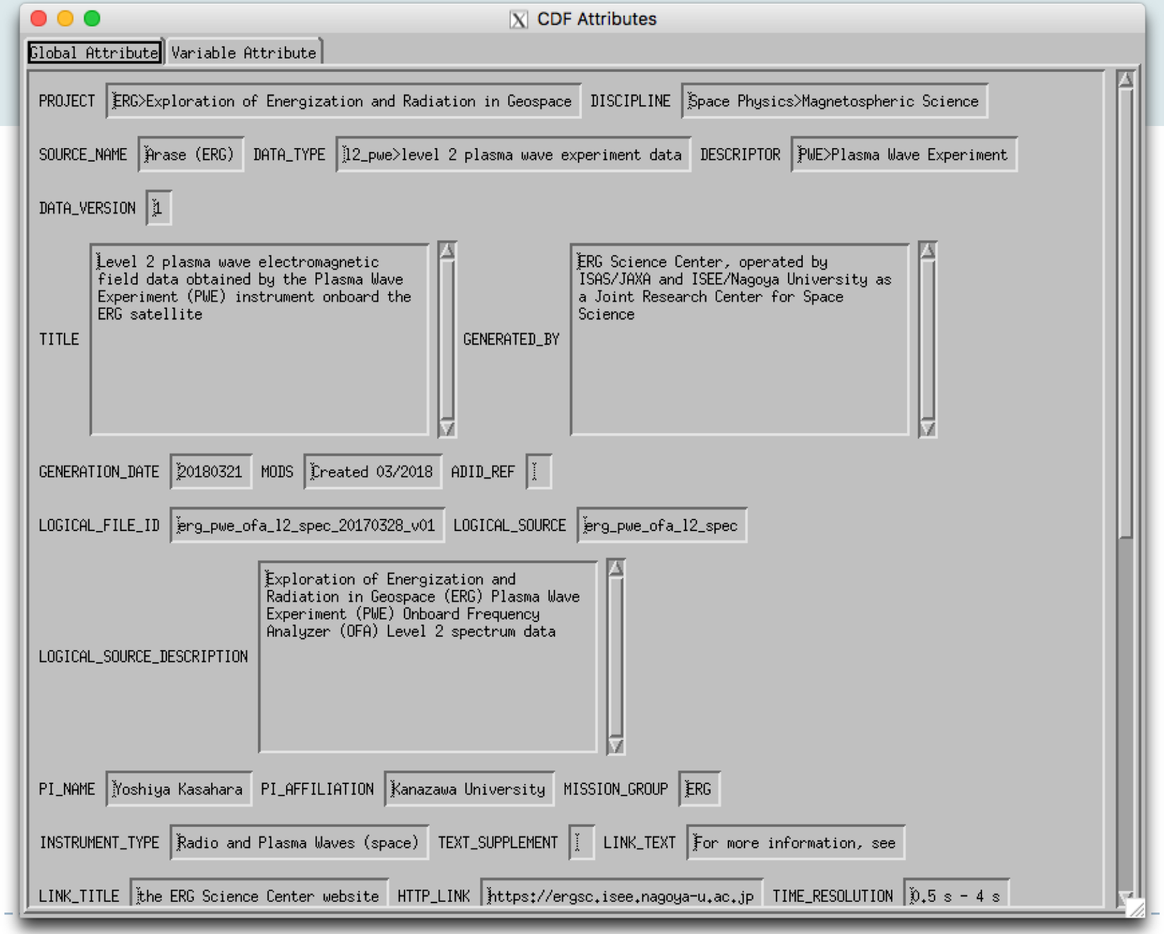
```
ERG> erg_load_pwe_ofa, uname=uname, pass=pass
ERG> tplot, [ 'erg_mgf_l2_mag_8sec_gse' , 'erg_pwe_pre_HFA-merged' ,
'erg_pwe_ofa_spec_l2_E_spectra_132' ,
'erg_pwe_ofa_spec_l2_B_spectra_132, 'erg_pwe_pre_EFD_DPB_spectra' ]
```



Onboard instrument data: Checking CDF attributes: *show_cdf_att*

All the global and variable attributes are shown in this panel.

```
ERG> show_cdf_att, tvar
```



PROJECT ERG>Exploration of Energization and Radiation in Geospace

DISCIPLINE Space Physics>Magnetospheric Science

SOURCE_NAME Arase (ERG)

DATA_TYPE l2_pwe>level 2 plasma wave experiment data

DESCRIPTOR PWE>Plasma Wave Experiment

DATA_VERSION 1

TITLE Level 2 plasma wave electromagnetic field data obtained by the Plasma Wave Experiment (PWE) instrument onboard the ERG satellite

GENERATED_BY ERG Science Center, operated by ISAS/JAXA and ISEE/Nagoya University as a Joint Research Center for Space Science

GENERATION_DATE 20180321

MODS Created 03/2018

ADID_REF

LOGICAL_FILE_ID erg_pwe_ofa_l2_spec_20170328_v01

LOGICAL_SOURCE erg_pwe_ofa_l2_spec

LOGICAL_SOURCE_DESCRIPTION Exploration of Energization and Radiation in Geospace (ERG) Plasma Wave Experiment (PWE) Onboard Frequency Analyzer (OFA) Level 2 spectrum data

PI_NAME Yoshiya Kasahara

PI_AFFILIATION Kanazawa University

MISSION_GROUP ERG

INSTRUMENT_TYPE Radio and Plasma Waves (space)

TEXT_SUPPLEMENT

LINK_TEXT For more information, see

LINK_TITLE the ERG Science Center website

HTTP_LINK https://ergsc.isee.nagoya-u.ac.jp

TIME_RESOLUTION 0.5 s - 4 s

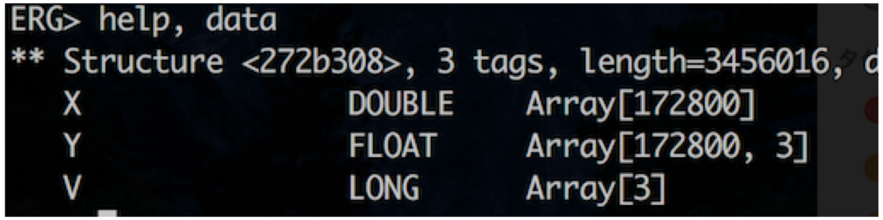
memo

Access the data structure in a tplot variable and create a new tplot variable (get_data, store_data)

Access the data structure in a tplot variable: *get_data*

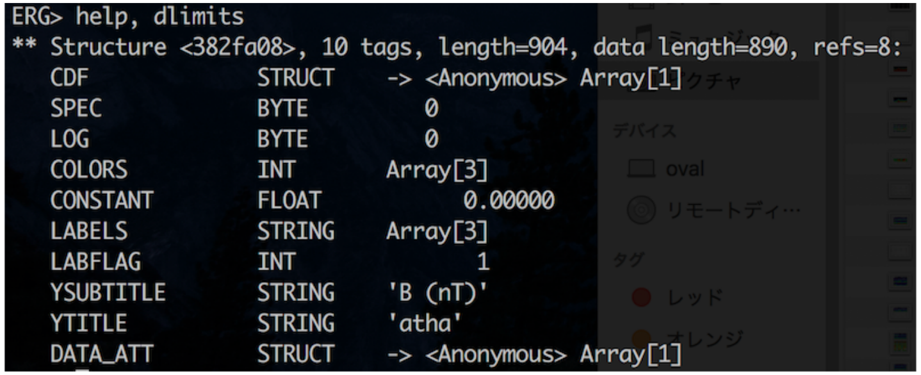
```
ERG> get_data, varname, data=data, dlimits=dlimits, lim=lim
      varname: tplot variable
       data: the data structure is stored
      dlimits: most of metadata are stored
      lim: some plot properties are stored


ERG> help, data.x , data.y
```



```
ERG> help, data
** Structure <272b308>, 3 tags, length=3456016, d
   X               DOUBLE    Array[172800]
   Y               FLOAT     Array[172800, 3]
   V               LONG      Array[3]
```

"get_data" extracts the data structure of a tplot variable and saves in a structure "data" of IDL session in the above case, so that users can access them by referring to as "data.x" or "data.y", for example.



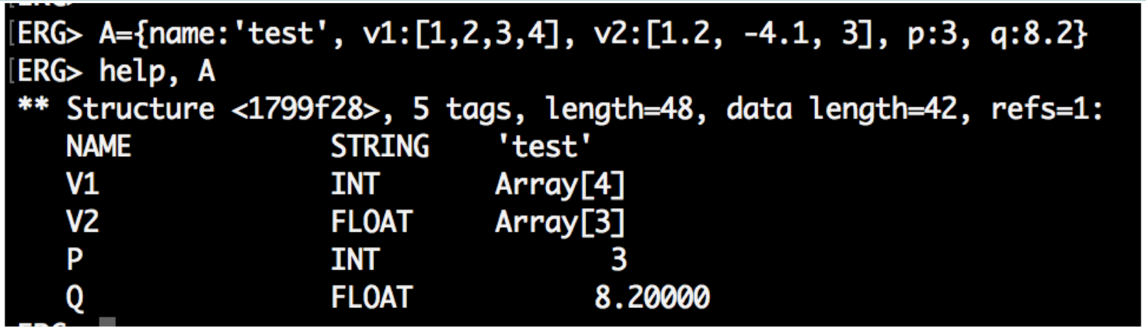
```
ERG> help, dlimits
** Structure <382fa08>, 10 tags, length=904, data length=890, refs=8:
   CDF             STRUCT    -> <Anonymous> Array[1]
   SPEC            BYTE         0
   LOG             BYTE         0
   COLORS          INT       Array[3]
   CONSTANT        FLOAT           0.00000
   LABELS          STRING    Array[3]
   LABFLAG         INT               1
   YSUBTITLE       STRING    'B (nT)'
   YTITLE          STRING    'atha'
   DATA_ATT        STRUCT    -> <Anonymous> Array[1]
```

The information on the original CDF data file (CDF) and metadata, and various plot properties are extracted into a structure "dlimits".

Tips: structure variables in IDL

Creating a structure in IDL

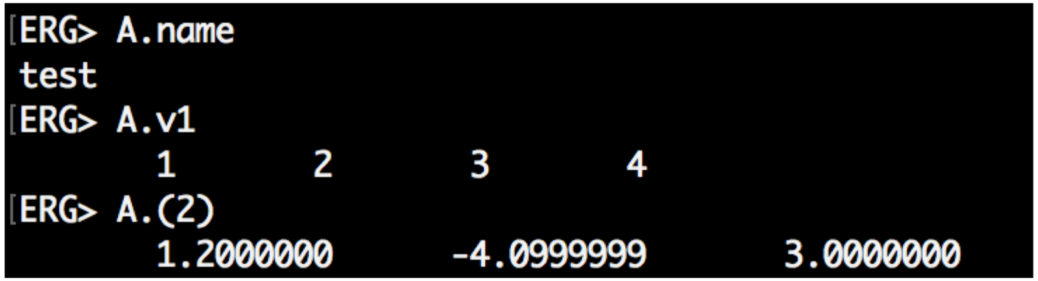
Structure = {tag1: var1, tag2: var2, …}
Structure = create_struct(tag1, var1, tag2, var2, …)



```
[ERG> A={name:'test', v1:[1,2,3,4], v2:[1.2, -4.1, 3], p:3, q:8.2}
[ERG> help, A
** Structure <1799f28>, 5 tags, length=48, data length=42, refs=1:
   NAME            STRING    'test'
   V1              INT       Array[4]
   V2              FLOAT     Array[3]
   P               INT              3
   Q               FLOAT          8.20000
```

Referring to a field within the structure

Structure.tag1
Structure.(tag_num)



```
[ERG> A.name
test
[ERG> A.v1
       1       2       3       4
[ERG> A.(2)
       1.2000000      -4.0999999       3.0000000
```

Create a new tplot variable:*store_data*

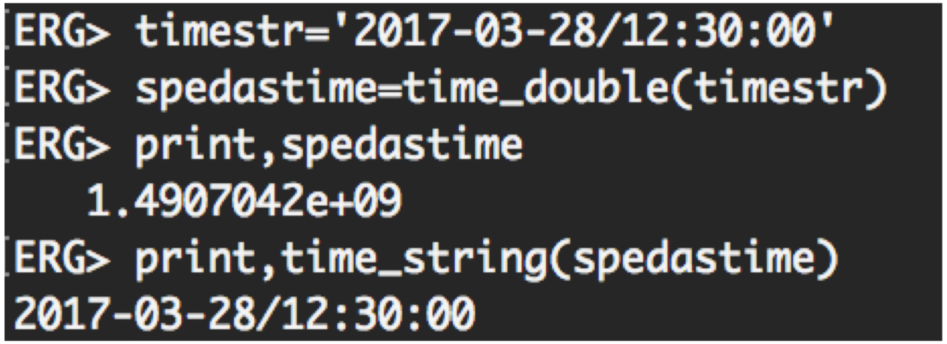
```
ERG> store_data, varname , data = { x:timearr, y:datarr }
```

varname: name of a newly created tplot variable (should be a character)

timearr: 1-D array containing time values in SPEDAS time of time-series data

datarr: 1-D or 2-D array containing the data values of time-series data. The size of 1st dimension should be identical to that of timearr.

- SPEDAS time is the UNIX time in double-precision floating-point values. UNIX time is the elapsed second since 00:00 UTC on January 1, 1970.
- Usually we use time_double() function to calculate a SPEDAS time value from a time string such as '2017-03-28/12:30:00'.
- SPEDAS time values can easily be converted to time strings with time_string() function.



```
ERG> timestr='2017-03-28/12:30:00'
ERG> spedastime=time_double(timestr)
ERG> print,spedastime
   1.4907042e+09
ERG> print,time_string(spedastime)
2017-03-28/12:30:00
```

Please refer to the SPEDAS wiki at
http://spedas.org/wiki/index.php?title=Time_handling
for more details of the time handling in SPEDAS.

Create and plot multi-tplot variables with PWE data

Get data from tplot Variable and save the data array

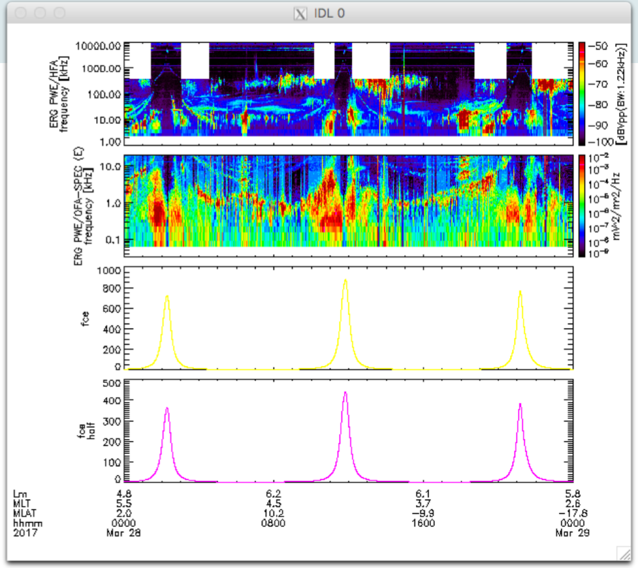
```
ERG> get_data, 'erg_mgf_l2_magt_8sec', data=data
```

Calculate the electron cyclotron frequency

```
ERG> fce = data.y / 10^(9.) * 1.6 * 10^(-19.) / (9.1093D * 10^(-31.)) / 2. / !pi /
1000.
ERG> fce_half = fce / 2.
```

Create and store the data with the new variable name

```
ERG> store_data, 'fce', data={x:data.x, y:fce}, dlim={colors:fsc_color('yellow')}
ERG> store_data, 'fce_half', data={x:data.x, y:fce_half},
dlim={colors:fsc_color(‘magenta')}
ERG> tplot,['erg_pwe_pre_HFA-merged', 'erg_pwe_ofa_spec_l2_E_spectra_132', 'fce',
'fce_half']
```



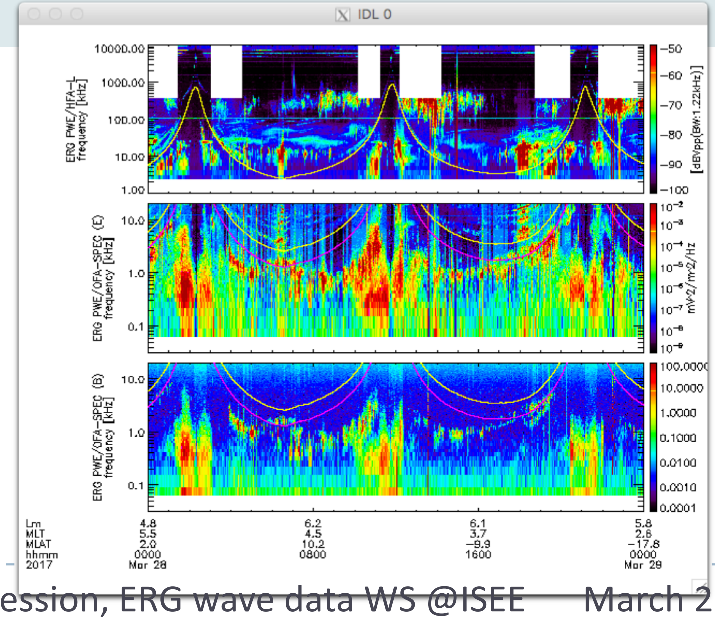
Create and plot multi-tplot variables with PWE data

Create and store the multi-data with the new variable name

```
ERG> store_data, 'erg_pwe_pre_HFA_merged_gyro', data =
['erg_pwe_pre_HFA_L_spectra_ey','erg_pwe_pre_HFA_H_spectra_ey','fce']
ERG> store_data, 'erg_pwe_ofa_spec_l2_E_gyro', data =
['erg_pwe_ofa_spec_l2_E_spectra_132',  'fce','fce_half']
ERG> store_data, 'erg_pwe_ofa_spec_l2_B_gyro', data =
['erg_pwe_ofa_spec_l2_B_spectra_132',  'fce','fce_half']
```

Set the yrange of tplot (min, max, 0 for linear / 1 for log)

```
ERG> ylim, 'erg_pwe_pre_HFA_merged_gyro’, 1, 1e4, 1
ERG> ylim, 'erg_pwe_ofa_spec_l2_E_gyro', 0.032, 20, 1
ERG> ylim, 'erg_pwe_ofa_spec_l2_B_gyro', 0.032, 20, 1
ERG> tplot, ['erg_pwe_pre_HFA_merged_gyro', 'erg_pwe_ofa_spec_l2_E_gyro',
'erg_pwe_ofa_spec_l2_B_gyro’]
```



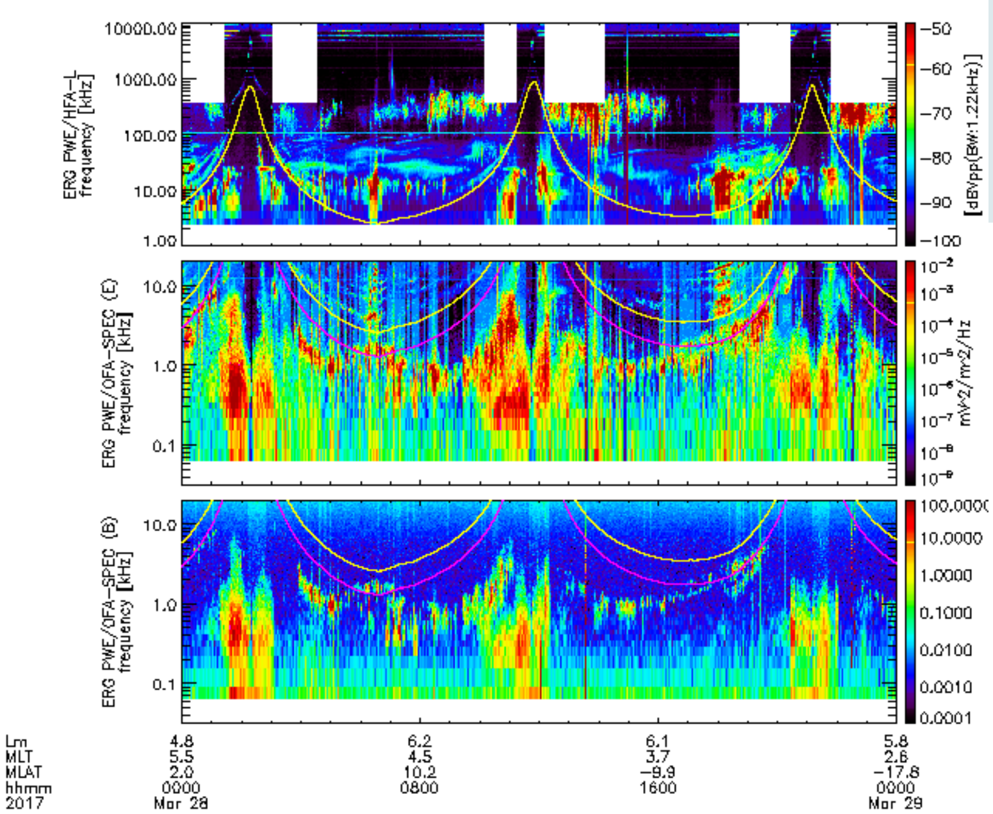
Dump to png and postscript files

To get a png file or postscript file

```
ERG> cwd     ;Display the current directory
CWD(25): Directory changed to: /yyyy/xxxx
ERG> tplot
ERG> makepng, ’erg_pwe_plot'    ;→erg_pwe_plot.png

ERG> popen, ’erg_pwe_plot'
ERG> tplot     ;Redo the last plot
ERG> pclose    ; →atha_plot.ps
```

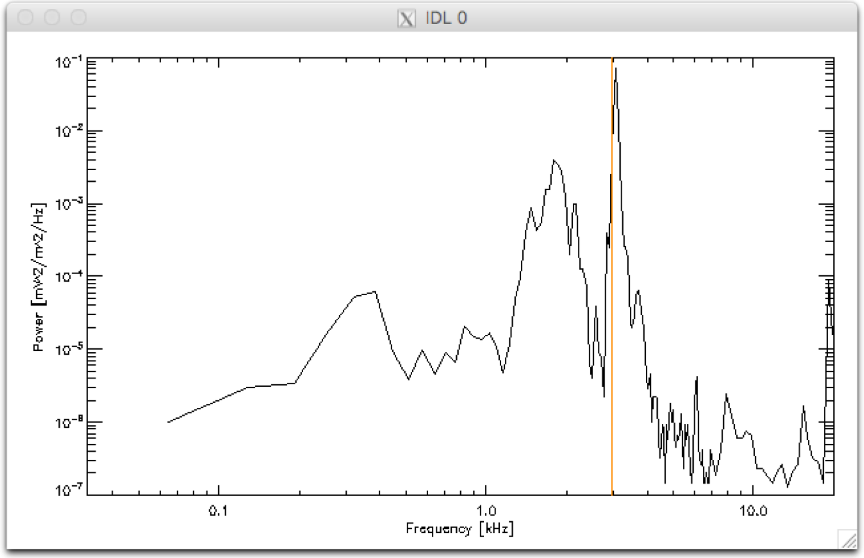
/yyyy/xxxx/erg_pwe_plot.png



Power spectrum at a particular time

```
ERG> t_index = nn('erg_pwe_ofa_spec_l2_E_spectra_132','2017-03-28/18:53')
ERG> get_data, 'erg_pwe_ofa_spec_l2_E_spectra_132’ , data=ofa_e_spec
ERG> tf_index = nn('fce_half ', '2017-03-28/18:53’)
ERG> get_data,'fce_half ', data=data_fce_half

ERG> plot, ofa_e_spec.v, ofa_e_spec.y[t_index,*],$
ERG> /ylog,/xlog,xstyle=1,xrange=[0.032,20],$
ERG> xtitle='Frequency[kHz]',ytitle='Power[mV^2/m^2/Hz]'
 ERG> oplot, [data_fce_half.y[tf_index],data_fce_half.y[tf_index]],[1e-
8,1],color=fsc_color(`green’)
```



ERG / Arase
Online information

- - Quick Looks
  https://ergsc.isee.nagoya-u.ac.jp/cef/erg_1day.cgi
- - ERG-SC
  https://ergsc.isee.nagoya-u.ac.jp/
- - CIDAS system
  http://cidas.isee.nagoya-u.ac.jp/kyodo/cidas.shtml.ja

memo

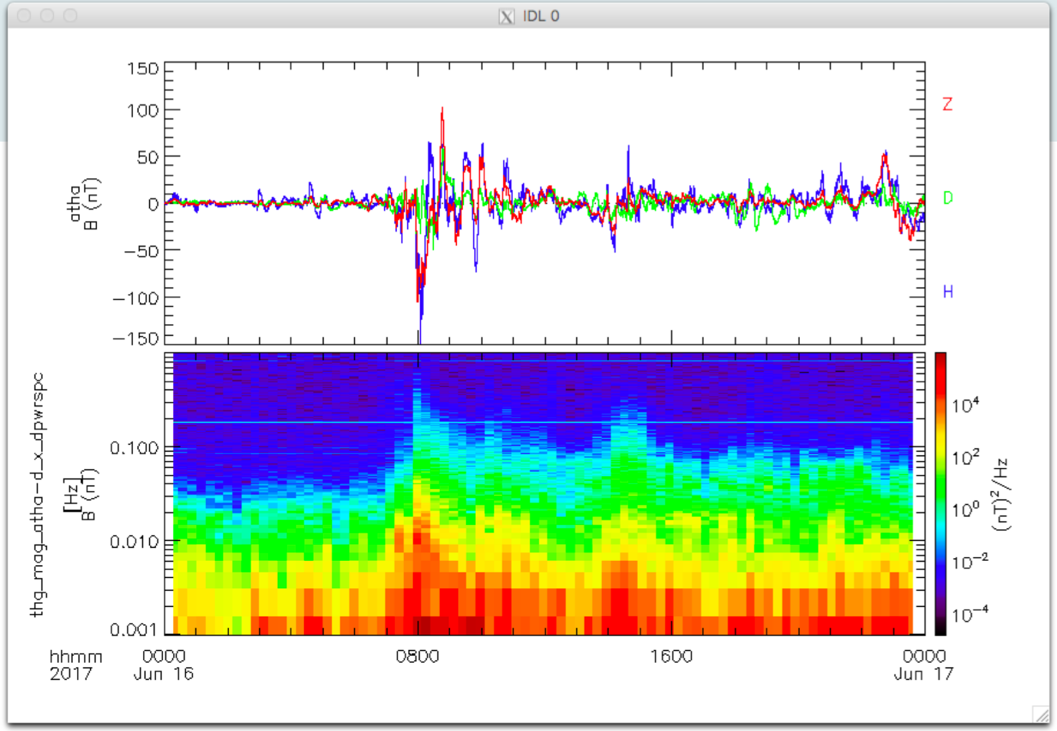
Frequency analysis of tplot data

SPEDAS training session, ERG wave data WS @ISEE March 28, 2018

Dynamics spectra – tdpwrspc–

```
tdpwrspc, 'varname'
  varname : tplot variable name(s)
```

```
ERG> tdpwrspc, 'thg_mag_atha-d'
ERG> tplot_names
    … … … …
     5 tha_mag_atha-d_hpfilt
    … … … …
     9 thg_mag_atha-d_x_dpwrspc
ERG> tplot, [ 5, 9 ]
```



FFT with the hanning window is applied to derive dynamic frequency spectra of the data.

Wavelet analysis – wav_data –

```
wav_data, 'varname'
    varname : tplot variable name(s)
```

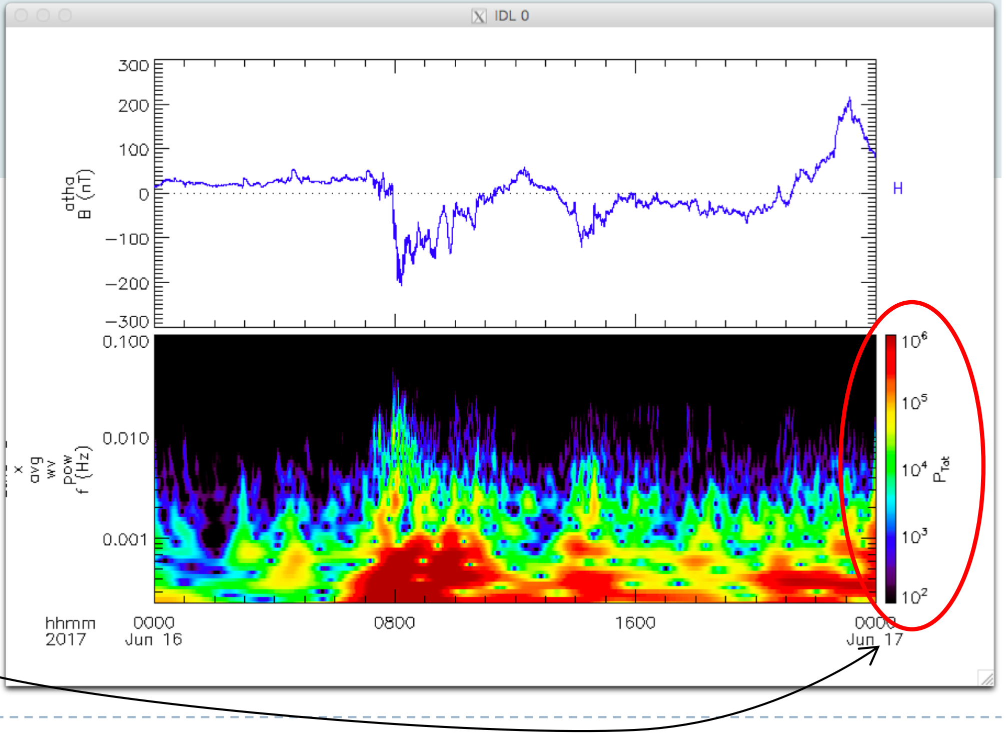
```
ERG> split_vec, 'thg_mag_atha-d'
ERG> avg_data, 'thg_mag_atha-d_x', 5.
ERG> wav_data, 'thg_mag_atha-d_x_avg'
STORE_DATA(264): Creating tplot variable: 13 thg_mag_atha-d_x_avg_wv_pow
ERG> tplot, 'thg_mag_atha-d_x_avg*'

ERG> zlim, 13, 100, 1000000, 1
ERG> tplot, 'thg_mag_atha-d_x_avg*
```

wav_data accepts data with less than 32768 samples. The number of data points is reduced as done with avg_data in this case.

Wavelet analysis is applied to derive dynamic spectra of the data.

zlim is similar to "ylim" command, but set the lower/upper limit of the color scale for a spectrum-type plot.



Other information sources for SPEDAS

- SPEDAS wiki
  - http://spedas.org/wiki/index.php?title=Main_Page
    - User's guide, Plug-in developer's guide, tips and tricks, The list of available crib sheets, …
- Change log of the source repository for the bleeding edge of SPEDAS
  - http://spedas.org/changelog/
- Crib sheets for TPLOT in Your_SPEDAS_dir/idl/general/examples/
  - **crib_tplot.pro** -- basic tplot intro
  - **crib_tplot_annotation.pro** -- How to control annotations in tplot (labels, text, etc...)
  - **crib_tplot_export_print.pro** -- How to export tplot data and tplot plots
  - **crib_tplot_layout.pro** -- How to control tplot plot layouts
  - **crib_tplot_range.pro** -- How to control the range and scaling of tplot plots
  - **crib_tplot_ticks.pro** -- How to control tplot plot ticks. (location, size, etc...)

Scientific instrument data of ERG satellite: Particle data

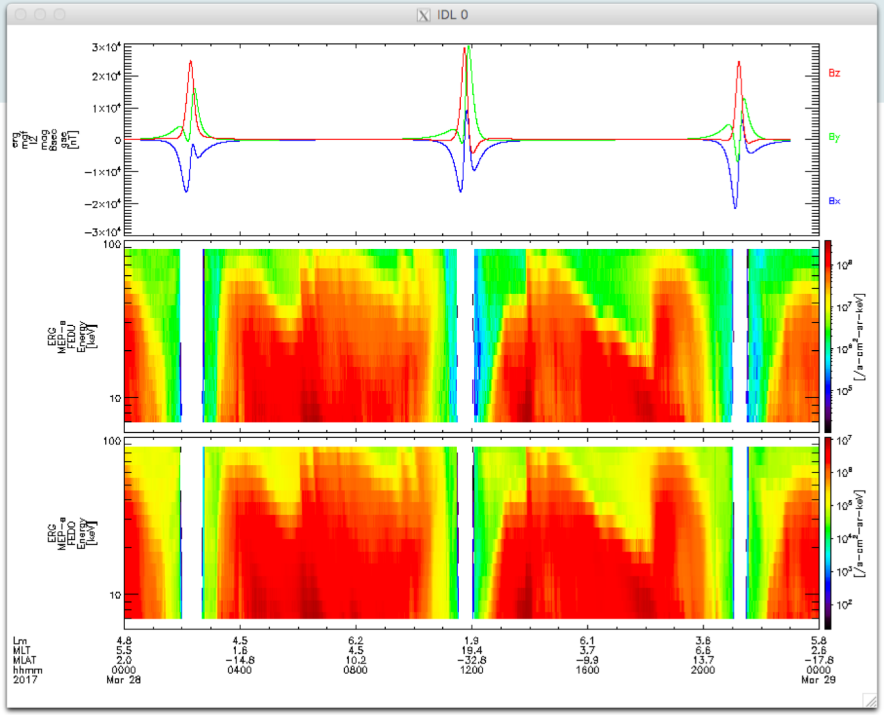
Onboard instrument data: MEP-e 3-D flux data

Load MEP-e data

```
ERG> erg_load_mepe, uname=uname, pass=pass
```

Plot magnetic field and MEP-e data

```
ERG> tplot,['erg_mgf_l2_mag_8sec_gse',
'erg_mepe_l2_3dflux_FEDU’, 'erg_mepe_l2_3dflux_FEDO']
```



Onboard instrument data:
Load the other particle data

Loading LEP-i, MEP-e, MEP-i, HEP and XEP data

```
ERG> erg_load_lepi_pre, uname=uname, pass=pass
ERG> erg_load_lepe_pre_omniflux, uname=uname, pass=pass
ERG> erg_load_mep_pre, species='i', datatype='omni', uname=uname, pass=pass
ERG> erg_load_hep_pre_omniflux, uname=uname, pass=pass
ERG> erg_load_xep_pre, uname=uname, pass=pass
ERG> tplot_names
```

```
… …
26 erg_mgf_l2_mag_8sec_dsi
… …
60  erg_mepe_l2_3dflux_FEDO
61 erg_lepi_pre_FEDO
62 erg_lepe_pre_FPDO

… …
66 erg_mepi_omni_FPDO
… …
70 erg_hep_pre_FEDO_L
71 erg_hep_pre_FEDO_H
72 erg_xep_pre_count
… …
```

Please remember the index numbers of these tplot variables **on your screen**:
they could differ for different person, depending on in what order you have loaded variables.

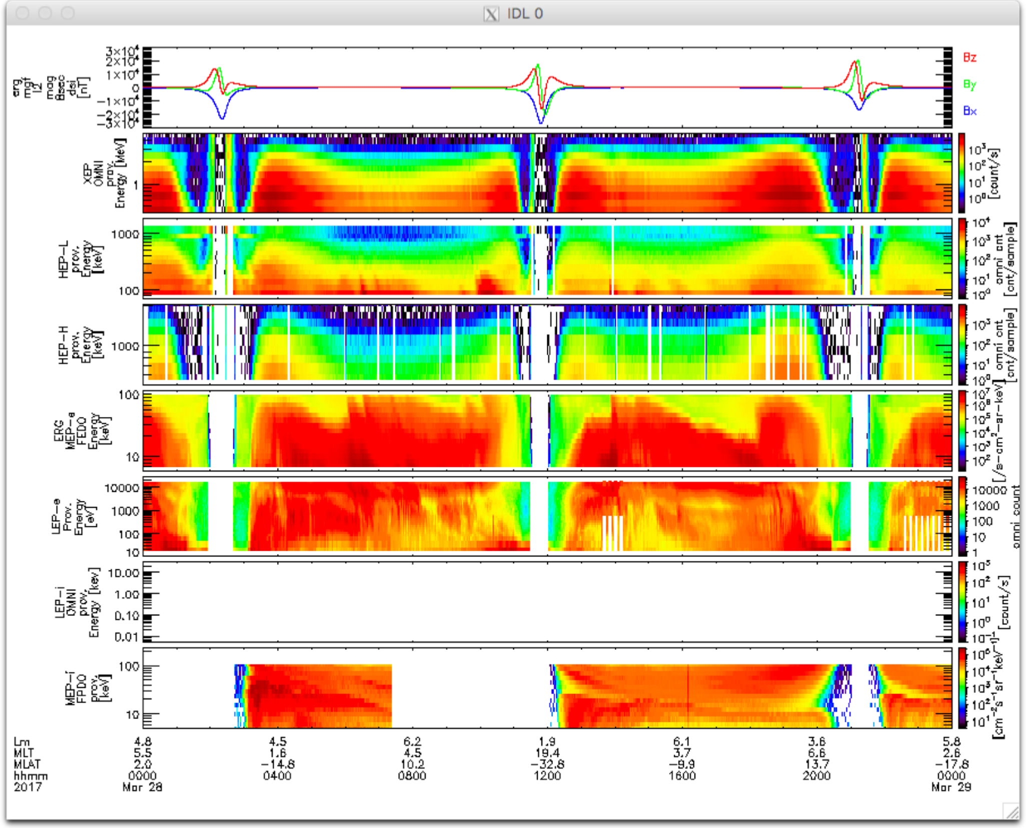
The numbers are used for the plot in the next slide!

Onboard instrument data: Put other particle data together to the plot

```
ERG> tvar = [ 26, 72, 70, 71, 60, 62, 61, 66  ]
```

Please line them up in the order, from top to bottom, of **MGF, XEP, HEP-H, HEP-L, MEP-e, LEP-e, MEP-i, and LEP-i**.

```
ERG> tplot, tvar
```



memo

Sample cases of ERG data analysis

Data analysis:
3-day plot of XEP data and geomag. indices

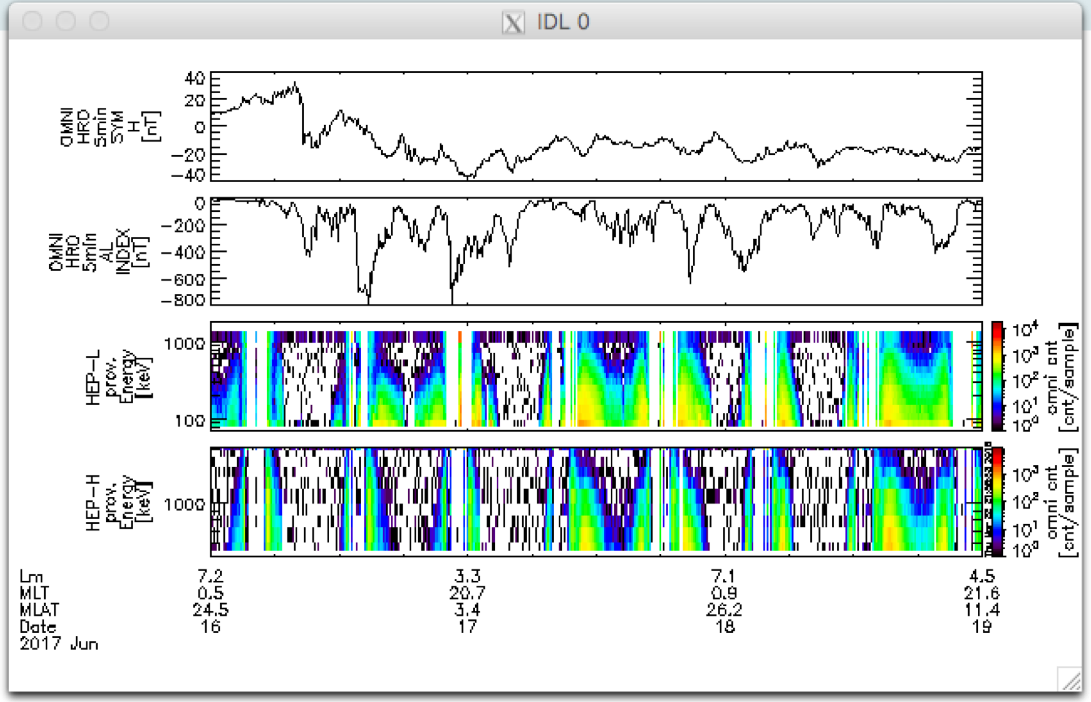
```
ERG> timespan, '2017-06-16', 3, /day
```

Load HEP data

```
ERG> erg_load_hep_pre_omniflux, uname=uname, pass=pass
```

Load other index data

```
ERG> omni_hro_load, /res5min
ERG> tplot,[ 'OMNI_HRO_5min_SYM_H' ,
'OMNI_HRO_5min_AL_INDEX' , 'erg_hep_pre_FEDO_*' ]
```



Onboard instrument data:

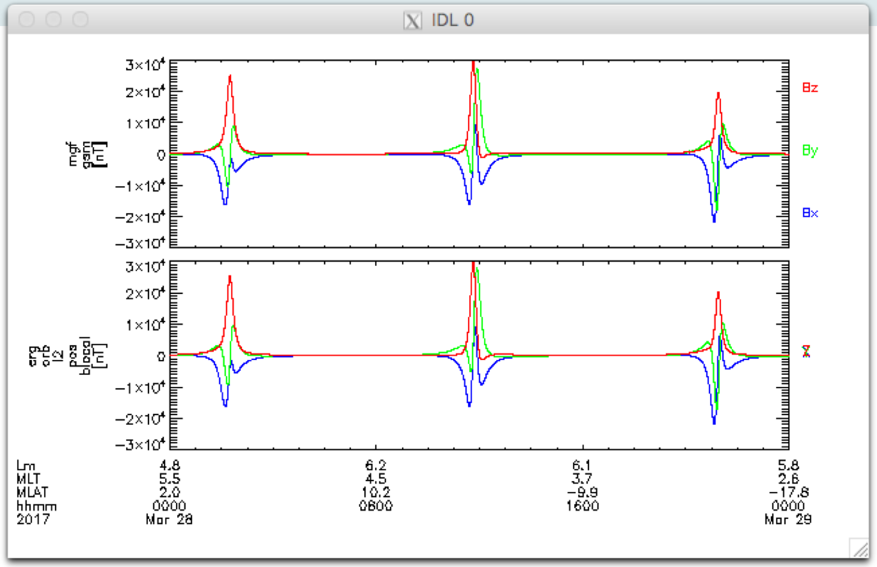
Coordinate transformation of MGF data

Coordinate transformation (SGA, SGI, DSI, J2000)
From J2000 to the geophysical coordinates -------> using “cotrans”

```
ERG> erg_cotrans, 'erg_mgf_l2_mag_8sec_dsi', 'mgf_j2000',
     in_coord='dsi', out_coord='j2000'
```

Coordinate transformation (SM<-->J2000)

```
ERG> spd_cotrans, 'mgf_j2000', 'mgf_gsm', in_coord='j2000',
out_coord='gsm'
ERG> tplot, ['mgf_gsm', 'erg_orb_l2_pos_blocal']
```



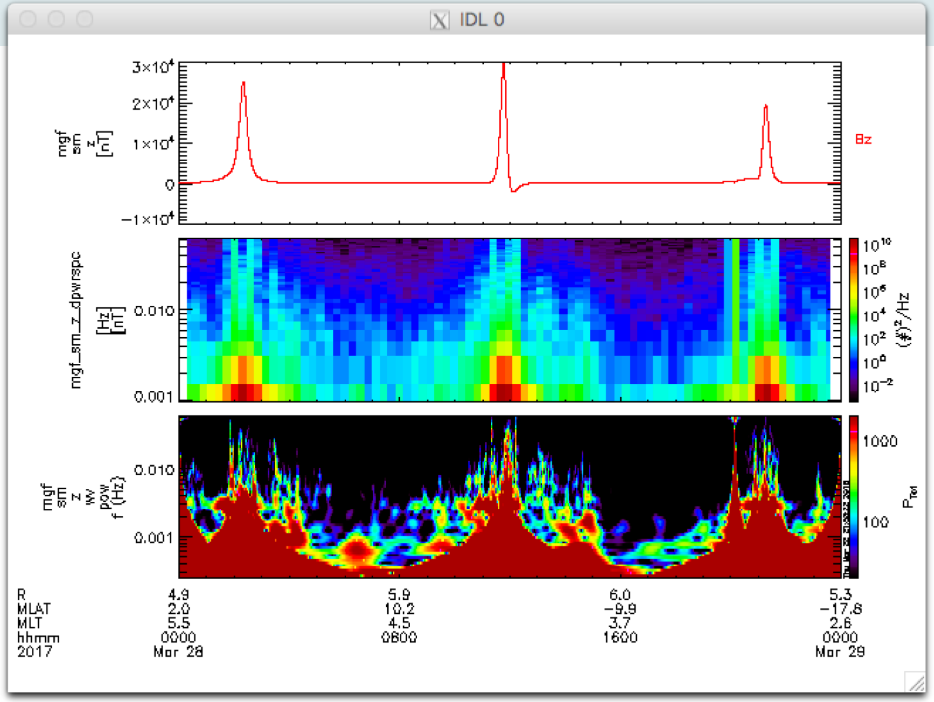
Data analysis:
Frequency analysis of MGF data

Please prepare a tplot variable "mgf_sm" containing the magnetic field vectors in SM coordinates for March 28

```
ERG> timespan, '2017-03-28', 1, /day
ERG> tdpwrspc, 'mgf_sm'
ERG> wav_data, 'mgf_sm_z'
ERG> tplot,'mgf_sm_z*'
```

Dynamic spectra by FFT with the Hanning window

Dynamic spectra by a wavelet analysis



Use `tlimit` to zoom in a period between perigee points!