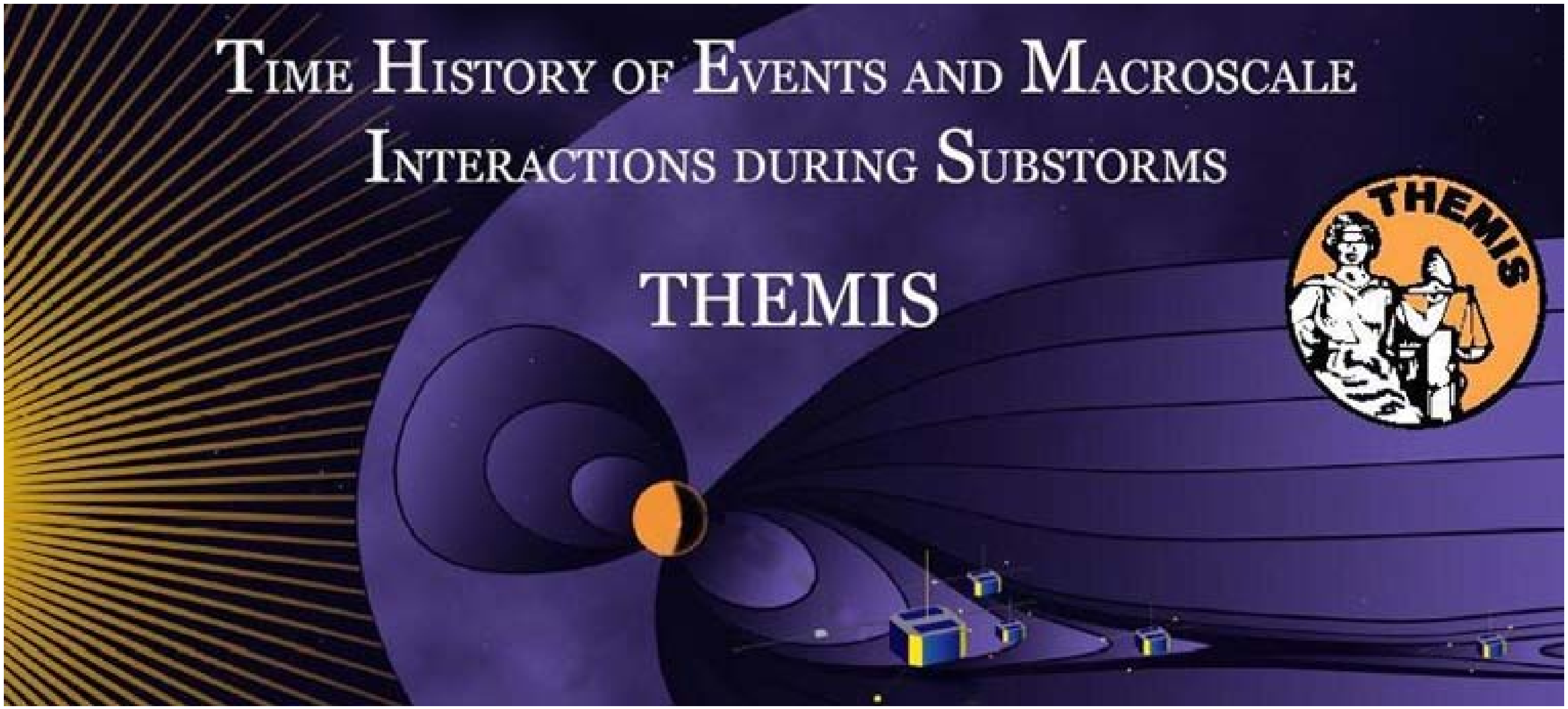


Science Software – v6.0 Tutorial

1. Overview

2. Instruments and Data Sets

3. System
   a) Installation
   b) IDL Configuration

4. Loading Data

5. Plotting Data

6. Analysis Routines
   a) Wavelet Transforms
   b) Coordinate Transforms
   c) Angular Spectra

7. Graphical User Interface

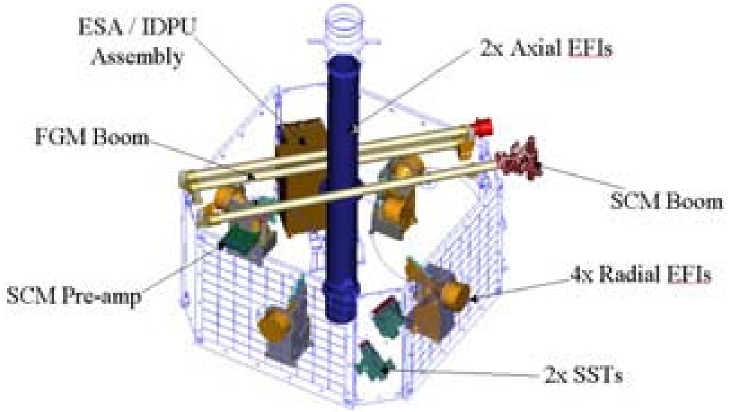
Appendices - Additional Functionality
   a) Load routines
   b) Crib Sheets
   c) Position/Velocity Plots
   d) Trace/Orbit Plots
   e) Ion Electron Velocity Slices
   f) Map/Orbit Plots

Software Objectives

- THEMIS Data Analysis Software (TDAS) Objectives
  - Powerful, Flexible Command Line Interface
  - GUI provides easy access to data, analysis tools, and graphics
  - Crib sheets are available for all data sets and common analysis functions
- IDL based (library of routines –but no main program!)
- Code is free and available to everyone
- The software operates on Level 1 and Level 2 data
  - It is not required to analyze level 2 data.
- Functionally separates the tasks into:
  - Reading
  - Manipulating
  - Plotting
- Platform independent. Works on:
  - Solaris
  - Linux
  - Windows, Vista
  - Mac OS X

FIELDS INSTRUMENTS:

EFI - Electric Field Instruments
FGM - Flux Gate Magnetometer
SCM - Search Coil Magnetometers



PARTICLE INSTRUMENTS:

ESA - Electrostatic Analyzer
SST - Solid State Telescope

GROUND BASED:

ASI – All-Sky Imager Array
GMAG – Magnetometer Array

PROCESSED DATA:

FBK – Filter Bank
FIT – Onboard Spin-Fit
FFT – Fast Fourier Transform
MOM – Onboard Moments
STATE – Spacecraft state vectors

To Download the TDAS software:

1. Download the latest TDAS release .zip file go to by going to the THEMIS web page at http://themis.ssl.berkeley.edu/software.shtml

2. Create a new directory called TDAS into which you will copy the latest software.

2. Move the unzipped folder (tdas_x_xx) into the TDAS directory.

3. The tdas_x_xx directory contains 3 main sub-directories

   a. THEMIS (idl/themis/) **–** routines specific to THEMIS

   b. ssl_general (idl/ssl_general/) **–** general routines

   c. external (idl/external/) **–** external libraries

To Configure IDL:

1. Add the TDAS directory to the IDLPATH.

To Configure IDL 6.4 or older:

a. Start IDL (Windows) or IDLDE (UNIX, Linux, Mac).
b. Go to File->Preferences.
c. Select the “Path” tab.
d. If <IDL_DEFAULT> is not present, press ‘Insert Standard Libraries’.
e. Press “Insert”.
f. Browse to find the TDAS directory.
g. Check the box to indicate “search subdirectories”.

To Configure IDL 7.0 or newer:

a. Start IDL (Windows) or IDLDE (UNIX, Linux, Mac).
b. Go to Window->Preferences.
c. Expand the menu to access the IDL->Paths section.
d. Press “Insert”.
e. Browse to find the TDAS directory. (See section 5.3 above: Installation and Configuration.)
f. Check the box next to the path to indicate “search subdirectories”.
g. Select “Apply”
h. Select “Ok”
i. Type .full_reset_session at the IDL command prompt or restart IDL.

Loading Data

- To load THEMIS data type the following at the IDL command prompt:

idl > **timespan**,‘7-7-23',2,/days

idl > **thm_load_gmag**, site=‘ccnv’, /subtract_average

idl > **tplot, ‘thg_mag_ccnv’**

timespan - sets the time frame (start and stop) of the data to be loaded.

thm_load_gmag - loads the data into memory and stores the CDF file on your disk drive.

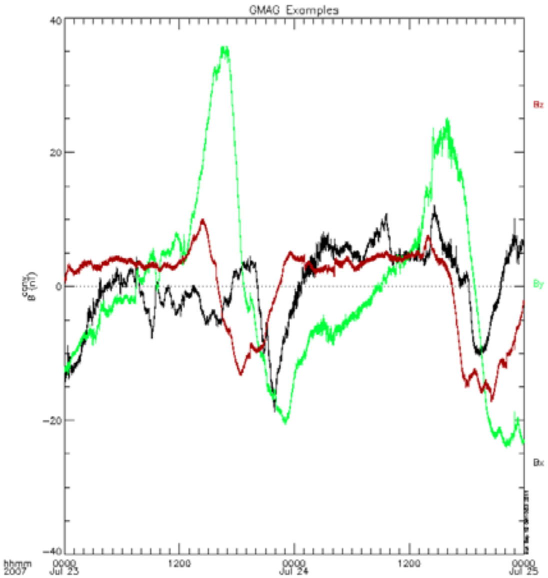
tplot - displays the data

- To add more features to the plot:

idl > **options**,‘thg_mag_ccnv’, labels=[‘Bx’,’By’,’Bz’]

idl > **tplot_options**, ‘title’, ‘GMAG Examples’

idl > **tplot, ‘thg_mag_ccnv’**



NOTE: Load routines are available for all THEMIS science data sets. See appendix slides.

Loading Data (Cont’d)

idl > **timespan**,‘2007-07-23',1

idl > **thm_load_state**, probe=‘a’, /get_support_data

idl > **thm_load_fgm,** probe=‘a’, coord=‘gsm’, datatype=‘fgl’, level=1

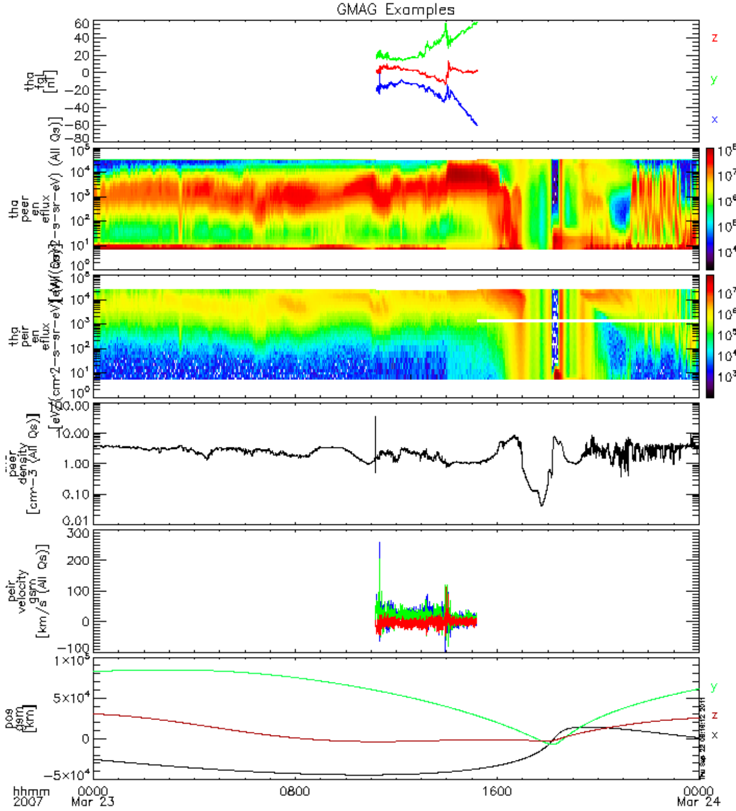
idl > **thm_load_esa,** probe=[‘a’,’b’], datatype=[‘peer_*’,’peir_*’], level=2

idl > **thm_load_efi,** datatype=‘vaw’

idl > **tplot_names**

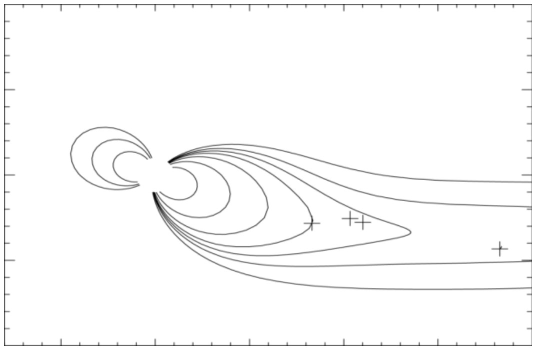
tplot_names - prints names of all tplot variables
currently in memory

idl **>** **tplot**, [‘tha_fgl’, $
‘tha_peer_en_eflux’, $
‘tha_peir_en_eflux’, $
‘tha_peer_density’, $
‘tha_peir_velocity_gsm’, $
‘tha_state_pos_gsm’]



Plotting Routines

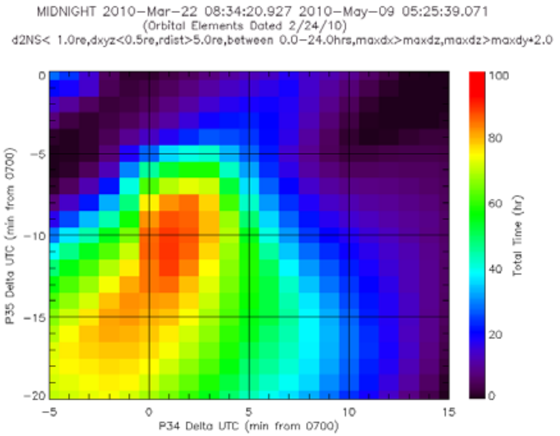
- Plotting routines using tplot variables
  - tplot
  - tplotxy
  - plotxy
  - plotxyz
  - tplot_names
  - tlimit
  - get_data
  - store_data



tplotxy can be used to plot isotropic position plots. Like plots of magnetic field models and spacecraft position

NOTE 1: Crib sheets are available for tplot and other plotting and analysis tools

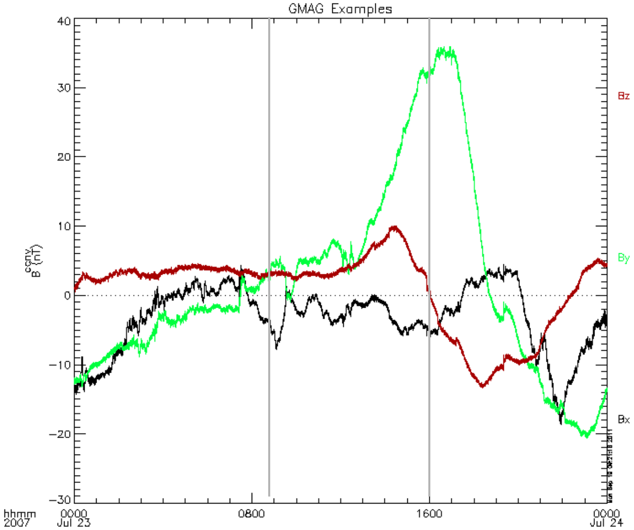
NOTE 2: Functionality of TDAS routines are controlled by the use of keywords. The keywords available for each routine can be found in the header of the source code.



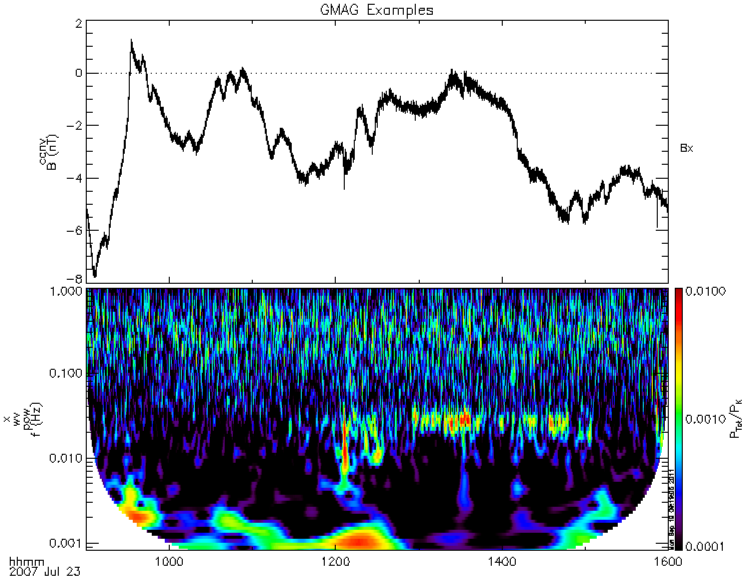
plotxyz can be used to plot 3 dimensional isotropic data, with any axis. (Not restricted to time-series.)

- To perform a wavelet transform on an area of interest

```
; define area of interest
idl >  tr = ['2007-07-23/09:00','2007-07-23/16']
```




```
; split the 3d vector into components
idl >  split_vec,  ' thg_mag_ccnv'
; compute transform of one component
idl >  wav_data,  'thg_mag_ccnv_x',  /kol, $
     trange=tr,  maxpoints = 24l*3600*2
; set color limits (log scale)
idl >  zlim, '*pow',  .0001,  .01, 1
; and plot it.
idl >  tplot,  '*ccnv_x*' ,  trange=tr
```



Analysis Routines

Analytic Coordinate Transformations

- tvector_rotate
- fac_matrix_make
- thm_fac_matrix_make
- minvar_matrix_make
- enp_matrix_make
- rxy_matrix_make
- sse_matrix_make
- gsm2lmn

Tsyganenko Model

- (t)trace2iono
- (t)trace2equator
- (t)t89
- (t)t96
- (t)t01
- (t)t04s

Example:

idl > **tt89**, 'tha_state_pos_gsm', newname='model_field'

idl > **fac_matrix_make**, 'model_field' , other_dim= 'xgse', newname = 'fac_mat'

idl > **tvector_rotate**, 'fac_mat', 'thc_peir_velocity', newname = 'ion_velocity_model_fa'

Coordinate Transforms

- Geophysical Coordinate Systems supported by TDAS
  - SPG Spinning Probe Geometric
  - SSL Spinning SunSensor L-vectorZ
  - DSL Despun SunSensor L-vectorZ
  - GEI Geocentric Equatorial Inertial
  - GSE Geocentric Solar Ecliptic
  - GSM Geocentric Solar Magnetospheric
  - SM Solar Magnetic
  - GEO Geographic Coordinate System
  - SSE Selenocentric Coordinate System
  - SEL Selenographic Coordinate System

- thm_cotrans
  - transforms to/from any of the following coordinate systems
  - updates metadata in output.
  - knows coordinate system of input from metadata

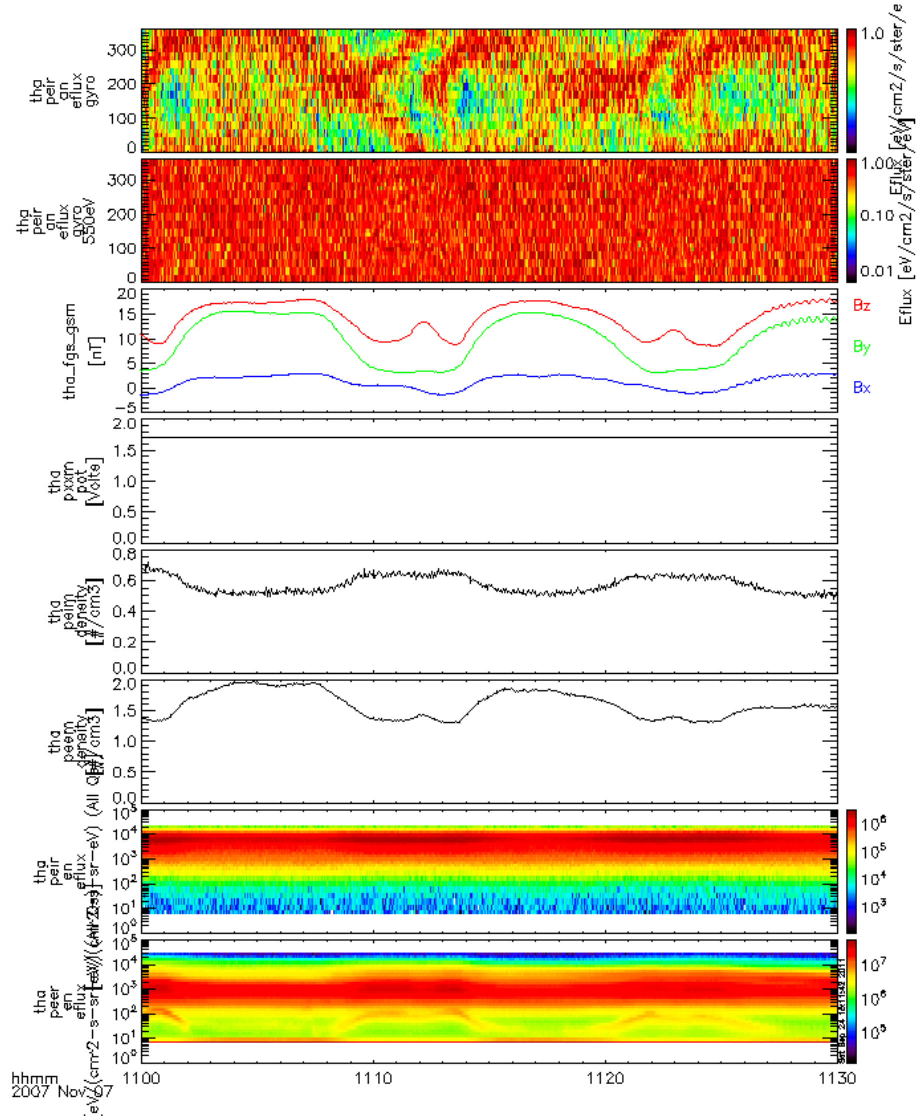
- Example (previously loaded FGM and STATE data)

idl > **thm_cotrans**, 'th?_fg?', out_coord='geo‘, ouf_suffix = ‘geo’

Plotting Angular Spectra

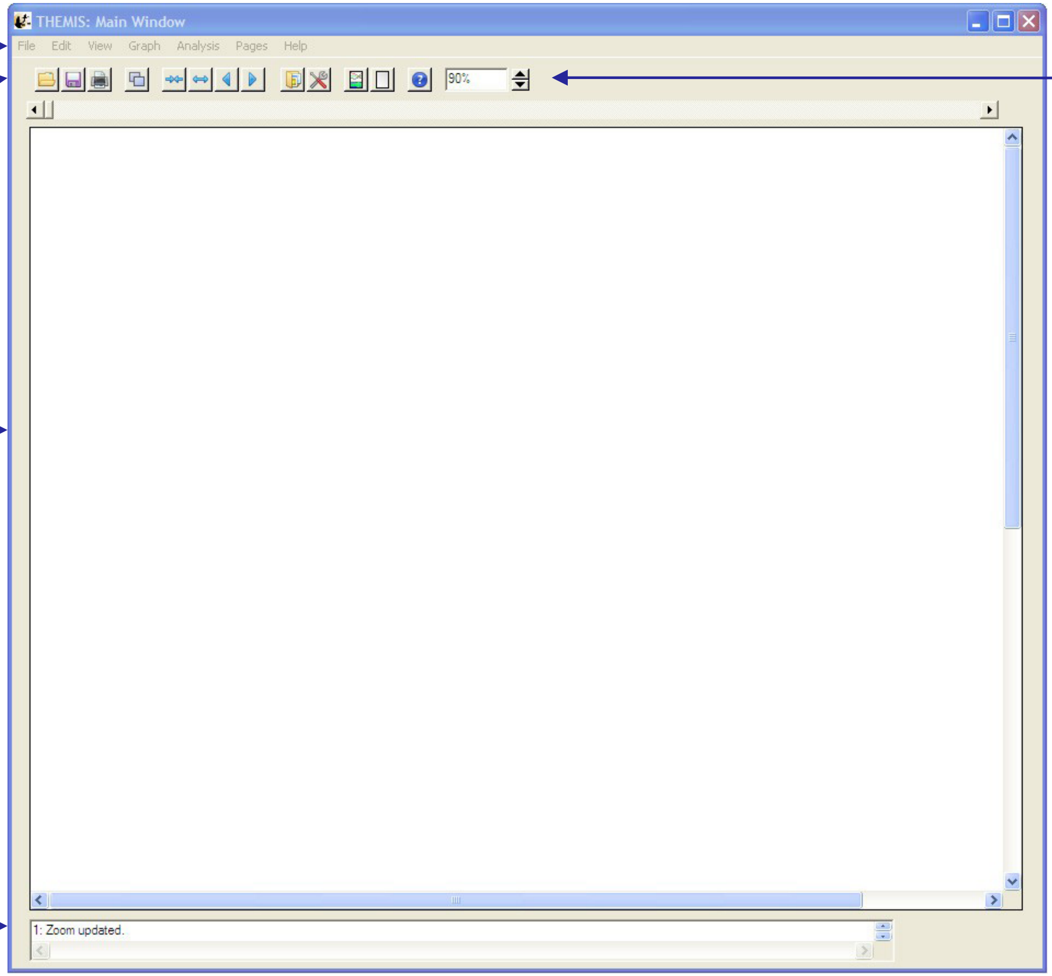
```
idl > timespan,'7 11 07/10',2,/hours
idl > sc='a'
idl > thm_load_state, probe=sc, /get_supp
idl > thm_load_fit, probe=sc, data='fgs',  $
          coord='gsm', suff='_gsm'
idl > thm_load_mom, probe=sc
idl > thm_load_esa,probe=sc
idl > tplot, ['tha_fgs_gsm tha_pxxm_pot',  $
           'tha_pe?m_density',  $
           'tha_pe?r_en_eflux']

idl > trange = ['07-11-07/11:00',' 07-11-07/11:30']
idl > thm_part_getspec, probe=['a'],
            erange=[100,1000],suff='_550eV',
            angle='gyro', pitch=[65,115], $
            other_dim='mPhism', /normalize, $
            data_type=['peir'], regrid=[32,16]
idl > thm_part_getspec, probe=['a'],
            trange=trange,  angle='pa'
            erange=[100,1000],suff='_550eV',
            other_dim='mPhism', /normalize, $
            data_type=['peir'], regrid=[32,16]
idl > tplot, ['tha_peir_an_eflux_pa_550eV' , $
            'tha_peir_an_eflux_gyro',  $
            'tha_peir_an_eflux_gyro_550eV',  $
            'tha_fgs_gsm tha_pxxm_pot', $
            'tha_pe?m_density',  $
             'tha_pe?r_en_eflux']
```



The GUI is the quickest and easiest way to learn TDAS functionality

To run the gui type: idl > **thm_gui**



Menu Bar

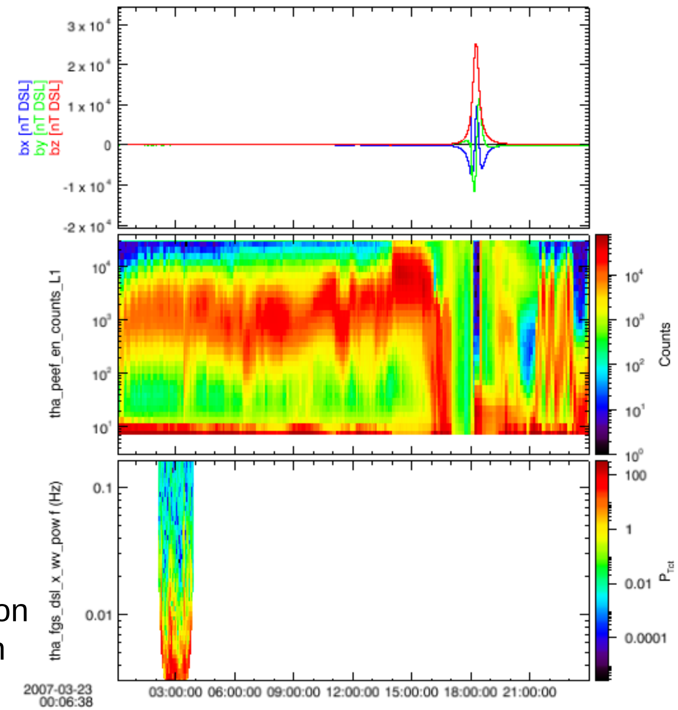
Tool Bar

Window Resize

Main graph area

Status Bar

With a few clicks of the button the user can load, analyze and plot data
For Example:



<u>To Load Data:</u>

Select Load Data under the File menu
Select Instrument Type: fgm, Level2: fgs_dsl, Click Right arrow button
Select Instrument Type: esa, Level1: peef, Click Right arrow button
Click Done button

<u>To Perform Analysis:</u>

Select Data Processing… under the Analysis menu
Select tha_fgs_dsl data, Click right Arrow button
Click Power Spectrum button, Select a 1 hour interval of time, Click OK
Click Done button

<u>To Plot Data:</u>

Select Plot/Layout Options… under the Graph menu
Select tha_fgs_dsl, Click Line button
Click Panels Add button, Select tha_peef_en_counts_L1, Click Spec button
Click Panels Add button, Select tha_fgl_dsl_x_wv_pow, Click Spec button
Click OK button

A. Load Routines

B. Crib Sheets

C. Position/Velocity Plots

D. Trace Orbit Plots

E. Ion Electron Velocity Slices

F. Map/Orbit Plots

- There are load routines for every THEMIS science data set and follow a naming convention:
  thm_load_xxx   where xxx = instrument name
- Each load routine contains keywords that are specific to each instrument or data type.
- The table below contains only a partial listing of routines available to retrieve and load data.

Name	Description	Level 1		Level 2
		Raw	Calibrated	
thm_load_esa	ElectoStatic Analyzer			*
thm_load_efi	Electric Fields Instrument	*	*	*
thm_load_fgm	Flux Gate Magnetometer Waveforms	*	*	*
thm_load_gmag	Ground Magnetometer			*
thm_load_scm	Search Coil Magnetometer Waveforms	*	*	*
thm_load_sst	Solid State Telescope	*	(-)	*
thm_load_state	Orbit and Attitude	V3		

(-) data reduction and analysis routines available

Additional Examples:

idl > **thm_load_state**, probe='b', /get_support_data
idl > **thm_load_fgm,** probe=['a','b', 'c'], coord='gei', datatype='fgl', level=1, suffix='gei'
idl > **thm_load_esa,** probe='a'
idl > **thm_load_efi,** datatype='vaf', trange=['2007-03-23', '2007-03-24']

A. Common Load Keywords

probe = Probe name. The default is 'all', i.e., load all available probes. This can be an array of strings, e.g., ['a', 'b'] or a single string delimited by spaces, e.g., 'a b'
datatype = The type of data to be loaded, 'fge', 'fgh', or 'fgl'. 'all' can be passed in also, to get all variables.
trange = Time range of interest (2 element array), if this is not set, the default is to prompt the user. Note that if the input time range is not a full day, a full day's data is loaded
level = the level of the data to read, the default is 'l1', or level-1 data. A string (e.g., 'l2') or an integer can be used. 'all' can be passed in also, to get all levels.
type = 'raw' or 'calibrated'. default is calibrated.
coord = coordinate system of output. default is 'dsl'
suffix = suffix to add to output data quantity (not added to support data)
cdf_data = named variable in which to return cdf data structure: only works for a single spacecraft and datafile name.
varnames = names of variables to load from cdf: default is all.
/get_support_data = load support_data variables as well as data variables into tplot variables.
/download_only = download file but don't read it.
/no_download = use only files which are online locally.
relpathnames_all = named variable in which to return all files that are required for specified timespan, probe, datatype, and level. If present, no files will be downloaded, and no data will be loaded.
/valid_names = if set, then this routine will return the valid probe, datatype and/or level options in named variables supplied as arguments to the corresponding keywords.
files = named varible for output of pathnames of local files.
/verbose = set to output some useful info, set to 0 to or 1 to reduce output.
/no_time_clip = Disables time clipping, which is the default

B. Crib Sheets

- Crib sheets are the easiest way to learn how to load, analyze and plot data.
- Crib sheets can be found in the examples directory under themis.
- The following is only a partial list of the crib sheets available.

thm_crib_efi	thm_crib_sst
thm_crib_esa	thm_crib_state
thm_crib_fgm	thm_crib_tplot
thm_crib_gmag	thm_crib_tplotxy
thm_crib_part_getspec	thm_crib_part_slice2d
thm_crib_scm	thm_crib_dproc

<u>Examples:</u>

idl > **.run thm_crib_fgm**
(or cut and paste, or copy and modify)

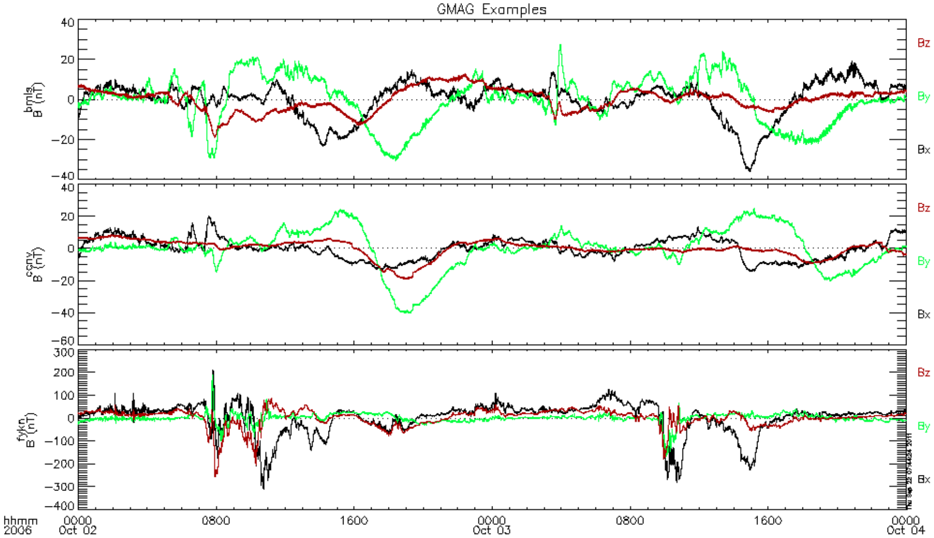
idl > **.compile thm_crib_plotxyz**
idl > **.go**
(run it by compiling in idl and then typing ".go“, or cut and paste)

idl > **thm_crib_part_slice2d**
(this is a procedure and can be just typed at the IDL command prompt)

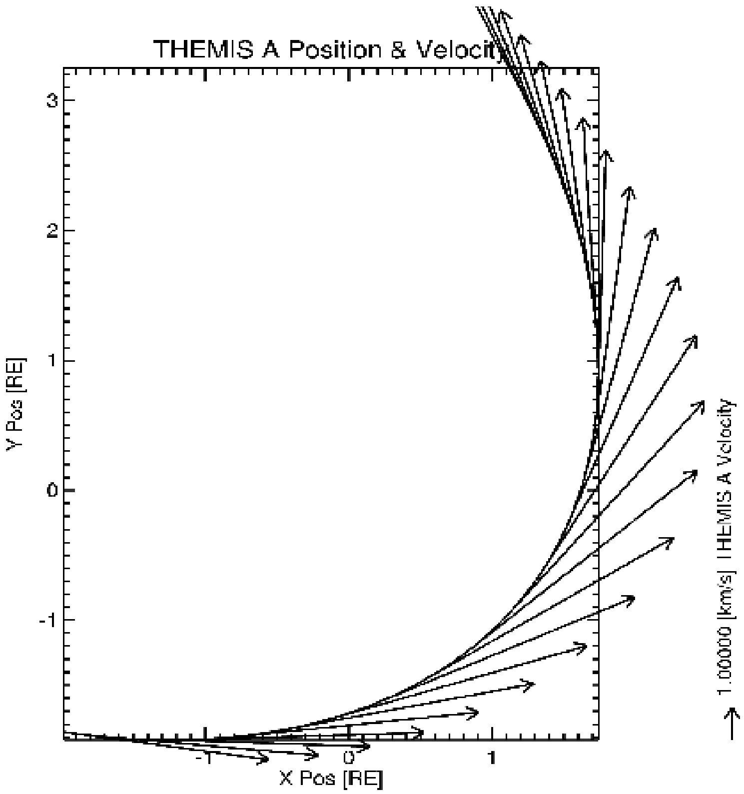
B. Crib Sheet Example

```
;pro thm_crib_gmag
; This is an example crib sheet that will load ground mag data.
; Open this file in a text editor and then use copy and paste to copy
; selected lines into an idl window. Or alternatively compile and run
; using the command: .RUN THM_CRIB_GMAG

pos=ptrace(option=1)        ; Set program trace to display line #.
timespan,'6-10-2',2,/days   ;  Define the time span of interest.
thm_load_gmag,site='bmls ccnv fykn',/subtract_average ;Load data
options,'thg_mag_????',labels=['Bx','By','Bz']
tplot_options, 'title', 'GMAG Examples'
print,'Defined TPLOT Variables;'
tplot_names ,/time          ;  Display all stored variables
print,ptrace(),'Loaded GMAG Data, and Displayed tplot names'
print,ptrace(),'Note that 3 sites were loaded each with 2 days of data.'
print,ptrace(),'All files are downloaded automatically if not found.'
stop
tplot,"thg_mag_????"        ; tplot accepts wildcard characters
                          :
                        etc...
```



Plotxyvec – Position/Velocity Plot



<u>To plot something similar, type the following</u>:

idl > **get_data, 'tha_state_pos', data=d**

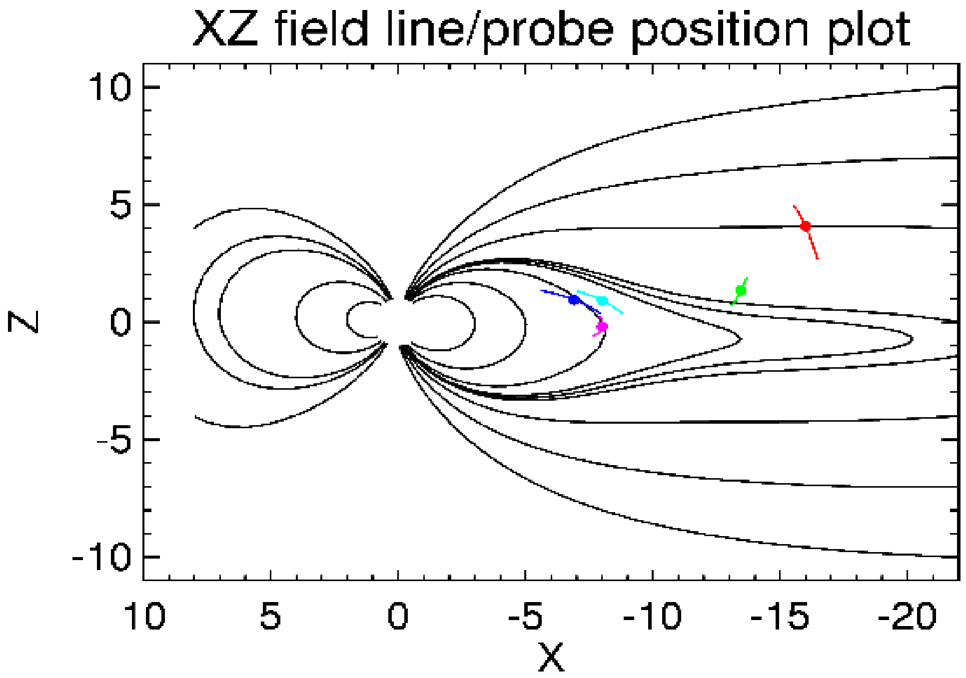
idl > **d_thm_a = d.*y/6374.***

idl > **plotxy, d_thm_a, xsize=600, ysize=600, $**
title="XY orbital plot w/change arrows", $
charsize=1.5, xrange=[-5, 15], yrange=[-15,5]

idl > **plotxyvec, [[x[1:n-1]], [y[1:n-1]]],[[u],[v]],/over, $**
charsize=1.5

Note**:** See thm_crib_plotxy.pro and thm_crib_ploxyvec.pro

Trace / Orbit Plots – XZ Plot



Source Code for Trace Plots – XZ Plot

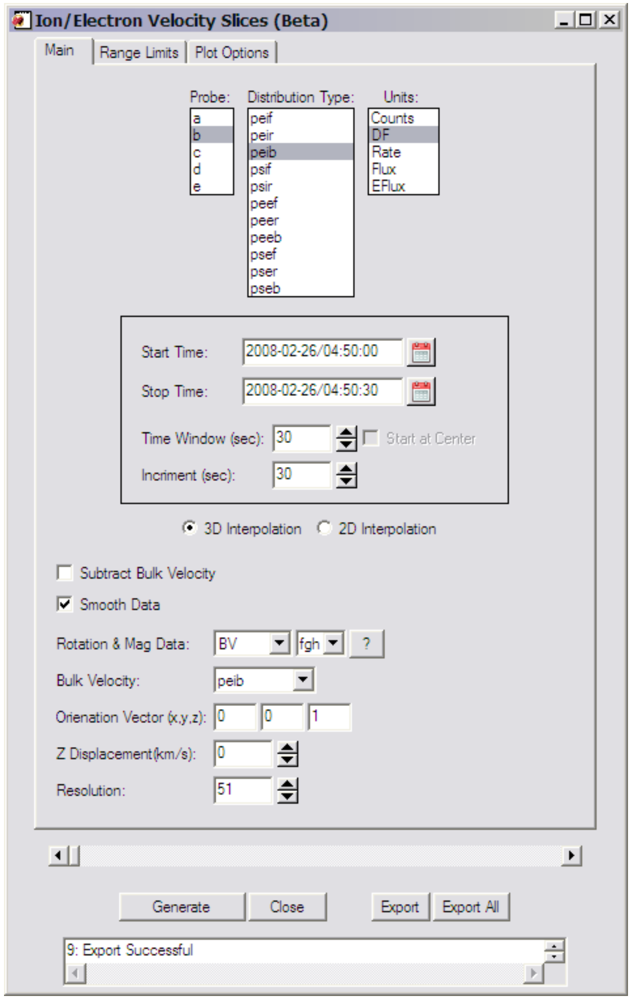
```
; code to plot trace field lines
thm_init
date = '2008-03-27/02:00:00' ;date to be plotted
hrs = 3
sdate = time_double(date)-3600*hrs/2
edate = time_double(date)+3600*hrs/2
timespan,sdate,hrs,/hour
x = [-22,-22,-22,-22,-17,-12,-8,-5,-3,2,4,7,8,8]
y = replicate(0,14)
z = [10,7,4,0,replicate(0,9),4]
times = replicate(time_double(date),14)
trace_pts_north =  [[x],[y],[z]]
trace_pts_south =  [[x],[y],[-1*z]]
store_data,'trace_pts_north',data={x:times,y:trace_pts_north}
store_data,'trace_pts_south',data={x:times,y:trace_pts_south}
ttrace2iono,'trace_pts_north',trace_var_name = 'trace_n',  in_coord='gsm',out_coord='gsm';trace field lines
ttrace2iono,'trace_pts_south',trace_var_name = 'trace_s', in_coord='gsm',out_coord='gsm', /south
window,xsize=800,ysize=600
xrange = [-22,10] ;x range of the xz plot
zrange = [-11,11] ;z range of the xz plot
tplotxy,'trace_n',versus='xrz',xrange=xrange,yrange=zrange,charsize=charsize, $
   title="XZ field line/probe position plot", xthick=axisthick, ythick=axisthick, $
   thick=linethick,charthick=charthick,ymargin=[.15,.1]
tplotxy,'trace_s',versus='xrz',xrange=xrange,yrange=zrange,/over, xthick=axisthick, $
   ythick=axisthick,thick=linethick,charthick=charthick
```

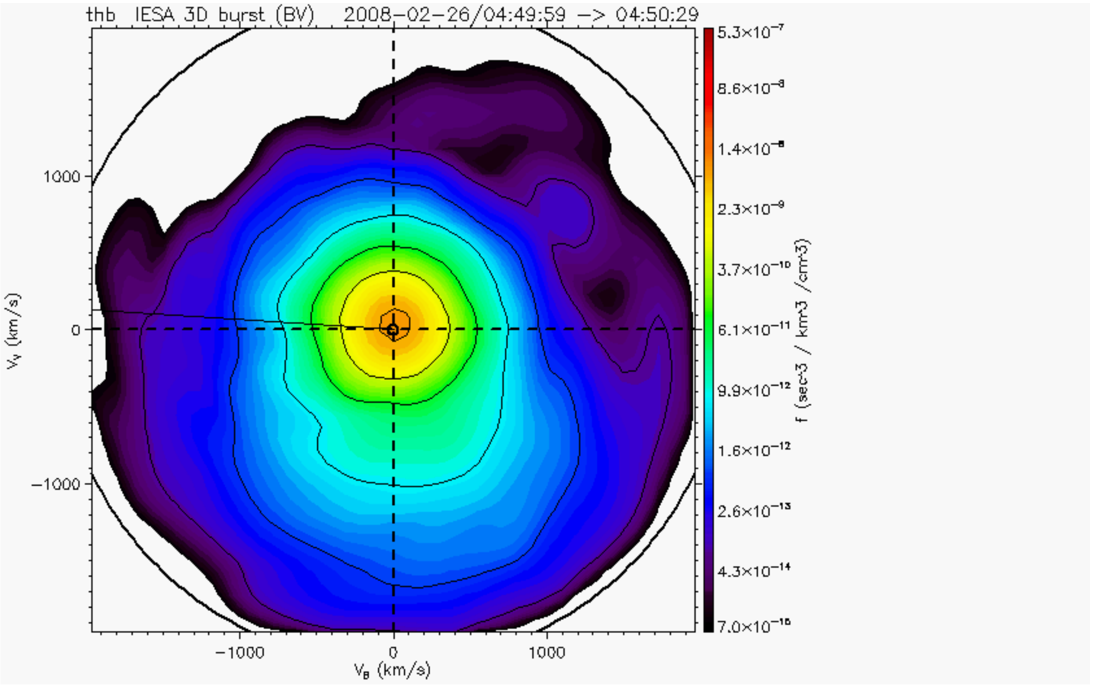
Source Code for Trace / Orbit Plots – Spacecraft Position

```
;code to overplot the probe positions
colors=['m','r','g','c','b']  ;colors for probes
probes = ['a','b','c','d','e'] ;the probes to be marked
A = FINDGEN(17) * (!PI*2/16.) ;makes a circle to mark spacecraft position
USERSYM, COS(A), SIN(A), /FILL
thm_load_state,probe=probes,coord='gsm'
tkm2re,'th'+probes+'_state_pos',/replace

;plot the probe positions
for i = 0,n_elements(probes) - 1 do begin
  probe = probes[i]
  color = colors[i]
  varname = 'th'+probe+'_state_pos'
  get_data,varname,data=d    ;plot position in KM
  if ~is_struct(d) then continue   ;skip if no valid data on day
  tmp = min(abs(d.x - time_double(date)),probe_pos) ;find midpoint
  tplotxy,varname,versus='xrz',/over,color=color,xthick=axisthick,ythick=axisthick, $
      thick=linethick,charthick=charthick  plotxy,reform(d.y[probe_pos,*],1,3),psym=8, $
      color=color,symsize=symsize,versus='xrz',/over,xthick=axisthick,ythick=axisthick, $
      thick=linethick,charthick=charthick
endfor
```

Beta support for Slices of 3d particle Velocity distributions are supported in the bleeding edge. Code can be started by typing: thm_ui_slice2d or can be accessed from the GUI by selecting Analysis->Velocity Slices





Trace/Orbit Plots - AACGM/Iono Trace Plot