spi_obs_point

User Manual

Version 1.9.0
8 November 2004

Jürgen Knödlseder
Centre d'Etude Spatiale des Rayonnements
knodlseder@cesr.fr
http://www.cesr.fr./∼jurgen/index.html

Note to the user

This software has been written to analyse data of the SPI telescope onboard INTEGRAL. Particular care has been taken in making the software user friendly and well documented. If you appreciated this effort, and if this software and User Manual were useful for your scientific work, the author would appreciate a corresponding acknowledgment in your published work.

Contents

1   Introduction

The executable `spi_obs_point` is part of the SPI scientific analysis preparation software component (SAP). It is a more versatile equivalent to the `spipoint` task, and produces a pointing definition structure of HDU type `SPI.-OBS.-PNT` for an observation group. In contrast to `spipoint`, `spi_obs_point` also accepts missing pointing information in an observation group, and constructs pointing information from an auxiliary ASCII text file.

`spi_obs_point` is written in the ANSI C++ language and has been developped under ISDC support platform 6.3. It requires `spi_toolslib` version 4.2.0 or higher.

2   Getting started

Before installing `spi_obs_point`, make sure that the ISDC support platform 6.3 or higher is installed on your system, and that you have installed the library `spi_toolslib` version 4.2.0 or higher.

After downloading the `spi_obs_point.tar.gz` file, step into a directory that should hold the distribution, move the `spi_obs_point.tar.gz` file into this directory and type after the UNIX prompt `$` (don't type this prompt):

```
$ gunzip spi_obs_point.tar.gz
$ tar xvf spi_obs_point.tar
```

The first command uncompresses the distribution file, the second unpacks the files.

Before configuration, the distribution needs to be reset to a clean state. To do this, type

```
$ make distclean
```

Then, configure the distribution. It is assumed here that you have previously installed the ISDC support platform, thus you should type

```
$ ~/bin/ac_stuff/configure
```

Finally, build the distribution by typing

```
$ make global_install
```

To perform a unit test, type

```
$ make test
```

Make sure that the test data `spi_test_data-1.0.tar.gz` are available at your site (they should reside in a directory whose name is defined by the `ISDC_TEST_DATA_DIR` environment variable).

3   Parameter file

```
################################################################################
#                                                                              #
#                  Centre d'Etude Spatiale des Rayonnements                     #
#                         (in collaboration with ISDC)                          #
#                                                                              #
#                        SPI OBS pointing definition                            #
#                                                                              #
# ---------------------------------------------------------------------------- #
#                                                                              #
#  File:      spi_obs_point.par                                                 #
#  Version:   1.7.0                                                             #
#  Component: SAP                                                               #
#                                                                              #
#  Author:    Juergen Knoedlseder                                              #
#             knodlseder@cesr.fr                                                #
#             CESR                                                             #
#                                                                              #
#  Purpose:   Parameter file of the SPI pointing definition                     #
#                                                                              #
#  History:   1.7.0  11-Dec-2003  First release (Rev. 7)                         #
#                                                                              #
################################################################################
#
# The input DOLs/filenames
#=========================
ingrpDOL,s,q,"og_spi.fits",,, "Input Observation Group DOL or filename"
intxtDOL,s,q,            "",,, "Pointing definition ASCII file"
#
# The output DOLs/filenames
#==========================
outgrpDOL,s,q,"og_spi_pnt.fits",,,"Output Observation Group DOL or filename"
outpntDOL,s,q, "pointings.fits",,,"Pointings DOL or filename"
#
# Task parameters
#================
useAUX,  b,h, yes,,,"Try using pointing definition from AUX data ?"
useHist, b,h, yes,,,"Try using pointing definition from historic AUX data ?"
useSWG,  b,h, yes,,,"Try using pointing definition from SWG ?"
useSlew, b,h,  no,,,"Use slew data ?"
xtol,    r,h, 1.0,,,"Tolerance on the stability of X axis (arcmin)"
ztol,    r,h, 1.0,,,"Tolerance on the stability of Z axis (arcmin)"
#
# Standard parameters
#====================
clobber,b,hl,yes,,,"Overwrite existing output data ?"
verbose,i,hl,2,0,3,"Information logging level"
```

Instead of specifying complete DOLs (Data Object Locations), which are composed of a filename plus the data structure extension (HDU), **spi_obs_point** accepts also simple filenames and adds the appropriate data structure extensions. This means that **specified data structure extensions are ignored**.

The parameters have the following meaning:

- `ingrpDOL` (optional) specifies the DOL or filename of the input Observation Group (HDU `[GROUPING]`) for which the pointing definition should be derived. The input group has to be of level `COR`.

  If an output Observation Group has been specified (parameter `outgrpDOL`), the specification of this parameter is optional. If the parameter is left blank, the output Observation Group will then be used as input Observation Group. Otherwise, the input Observation Group will be copied into the output Observation Group.

- `intxtDOL` (optional) specifies an ASCII file that provides pointing information in case that no pointing information is available in the input Observation Group. This file will only be used if the `useTXT` is set to `yes`. If left blank, set `useTXT = no`.

- `outgrpDOL` (optional) specifies the DOL or filename of the output Observation Group (HDU `[GROUPING]`). The output Observation Group will be a copy of the input Observation Group plus the resulting pointing definiton data structure `SPI.-OBS.-PNT` attached.

  If an input Observation Group has been specified (parameter `ingrpDOL`), the specification of this parameter is optional. If the parameter is left blank, the input Observation Group will then be used as output Observation Group.

- `outpntDOL` specifies the DOL of the pointing definition data structure (HDU `[SPI.-OBS.-PNT]`) that will be attached to the output Observation Group.

- `useAUX` specifies if `spi_obs_point` should attempt to find pointing information from auxiliary data that is attached to the Observation Group. If no such data will be found, `spi_obs_point` emits a warning but continues execution. Setting `useAUX=yes` does not harm in the case that no auxiliary data is available. However, if auxiliary data have been found, `spi_obs_point` will in this case not use the pointing information from the Science Window Groups or the ASCII input file.

- `useHist` specifies if historic pointing information should be used of not. This option may allow to force omitting historic pointing information, i.e. in the case that the file is corrupted.

- `useSWG` specifies if `spi_obs_point` should attempt to find pointing information from the Science Window Groups. In this case, `spi_obs_point` will read all relevant keywords for each Science Window Group, and will extract pointing information from these keywords.

- `useTXT` specifies if in case of `useSWG=yes`, `spi_obs_point` should use pointing information from the ASCII file or the Science Window Group keywords. If `useTXT=yes`, ASCII file information has precedence over Group keywords (hence the pointing specified in the ASCII file will logically overwrite the keywords in the Science Window Groups). Only if no pointing information is found for a given Science Window Group in the ASCII file, the keywords will be examined. If also no information can be extracted from the keywords, the pointing will be set to 0 degrees in Right Ascension and Declination.

- `useSlew` specifies if slew Science Window Groups should be used in case of `useSWG=yes`. The Science Window Group type will be determined from the Science Window identifier, and slew Science Windows will be omitted if `useSlew=no`. This is reasonable since we cannot specify a single pointing for a slew in an ASCII file.

- `xtol` specifies the maximum pointing fluctuation tolerance of the INTEGRAL X-axis in units of arcmin that is accepted for a pointing to be considered as stable. This parameter is only used if pointing information is extracted from auxiliary data.

- `ztol` specifies the maximum pointing fluctuation tolerance of the INTEGRAL Z-axis in units of arcmin that is accepted for a pointing to be considered as stable. This parameter is only used if pointing information is extracted from auxiliary data.

- `clobber` specifies if existing output data structures should be overwritten or not. If `yes` is specified, the executable will notify the user about the deletion of any file. If `no` is specified and the executable attempts to overwrite existing data (e.g. an existing output Observation Group or pointing definition structure), the task will exit with an error message.

- `verbose` specifies the verbose level of the executable during preparation of the output Observation Group. For `verbose=0`, no information will be logged in case of an error. For `verbose=1`, only errors will be logged, while for `verbose=2` also actions (such as DOL detachments and attachments and DOL deletion) will be logged.

4 Interface definition

`spi_obs_point` takes the information that is available in either the Observation Group or an auxiliary ASCII file to generate a pointing definition data structure of type `SPI.-OBS.-PNT`.

From the input Observation Group, the following HDUs are required:

- `GNRL-IDX.-GRP` science window group index that groups all science window groups of level `PRP`.

The input Observation Group is copied into the output Observation Group, with the following additional HDUs attached:

- `SPI.-OBS.-PNT` pointing definition.

If no output Observation Group is specified, `spi_obs_point` attempts to make the input Observation Group to the output Observation Group, which is only possible if the `clobber` parameter is set to `yes`.

The ASCII pointing definition file has the following format:

```
900001010022  90.000 +10.000  90.000 -80.000
900001010032  90.000 +10.000  90.000 -80.000
900001010042  90.000 +10.000  90.000 -80.000
...
```

where the first 12 digit integer field specifies the Science Window Identifier, and the following 4 floating-point parameters specify the pointing in degrees, with the meaning

- `RA_X` : SPI pointing axis Right Ascension (`[0, 360]`)

- `DEC_X` : SPI pointing axis Declination (`[-90, 90]`)

- `RA_Z` : SPI secondary axis Right Ascension (`[0, 360]`)

- `DEC_Z` : SPI secondary axis Declination (`[-90, 90]`)

Note that in the actual version the user has to assure himself that the angular separation of the X and Z axis definition amounts to 90 degrees.

5 Algorithm

Please refer to the `DAL3AUX` User Manual for more information about INTEGRAL attitude determination.

6   Error codes

The executable `spi_obs_point` may stop with the following error codes:

```
SPI_OBS_POINT_ERROR_MEM_ALLOC                    -233100
SPI_OBS_POINT_ERROR_INVALID_LIST                 -233101
SPI_OBS_POINT_ERROR_BAD_FILE                     -233102
```

They have the following meaning:

- **SPI_OBS_POINT_ERROR_MEM_ALLOC** : the allocation of dynamical memory has failed. Probable your system resources are too limited to run this task.

- **SPI_OBS_POINT_ERROR_INVALID_LIST** : an invalid list of pointings has been encountered. This error should never happen and indicates that the executable has somewhere a bug. Please report this to the author.

- **SPI_OBS_POINT_ERROR_BAD_FILE** : `spi_obs_point` was unable to open the specified ASCII file given by the parameter `intxtDOL`. Probably, the specified file name was wrong.

In addition, all errors that may occur in calls to ISDC support functions (such as for example DAL, RIL or PIL) are forwarded. Please consult the ISDC web pages for getting information about these error codes.