

**spi\_obs\_add**

# **User Manual**

**Version 2.2.0**  
**8 November 2004**

Jürgen Knödseder  
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**

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Getting started</b>	<b>1</b>
<b>3</b>	<b>Parameter file</b>	<b>2</b>
<b>4</b>	<b>Interface definition</b>	<b>5</b>
<b>5</b>	<b>Algorithms</b>	<b>5</b>
<b>6</b>	<b>Error codes</b>	<b>5</b>

## 1 Introduction

The executable `spi_obs_add` is part of the SPI scientific analysis preparation software component (SAP). It allows to combine a list of Observation Groups, provided in an ASCII file, into a single Observation Group for scientific analysis. Additionally, only a subset of the energy bins available in the input Observation Groups may be selected. Note that the science window group index is stripped in this procedure.

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

## 2 Getting started

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

```
$ gunzip spi_obs_add.tar.gz
$ tar xvf spi_obs_add.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
```

### 3 Parameter file

```
#####
#
#           Centre d'Etude Spatiale des Rayonnements           #
#           (in collaboration with ISDC)                       #
#
#           SPI Observation Addition                            #
#
# -----#
#
# File:      spi_obs_add.par                                   #
# Version:   2.0.0                                           #
# Component: SAP                                             #
#
# Author:    Juergen Knoedlseder                             #
#            knodlseder@cesr.fr                              #
#            CESR                                             #
#
# Purpose:   Parameter file for the SPI Observation Addition  #
#
# History:   2.0.0 29-Jun-2004 Second release                #
#
#####
#
# The input DOLs/filenames
#=====
inlist,s,q,"scw.list",,,"List of Input Observation Groups"
#
# The output DOLs/filenames
#=====
outgrpDOL,  s,q,      "og_spi.fits",,,"Output Observation Group DOL or filename"
outebdsDOL, s,q,"energy_boundaries.fits",,,"Energy boundary DOL or filename"
outpntDOL,  s,q,      "pointing.fits",,,"Pointing Interval DOL or filename"
outgtiDOL,  s,q,      "gti.fits",,,"Good Time Interval DOL or filename"
outdspDOL,  s,q,      "evts_det_spec.fits",,,"Event Spectrum DOL or filename"
outdtiDOL,  s,q,      "dead_time.fits",,,"Deadtime DOL or filename"
outpefDOL,  s,q,      "psd_efficiency.fits",,,"PSD efficiency DOL or filename"
outprfDOL,  s,q,      "psd_response.fits",,,"PSD response DOL or filename"
outbgmidxDOL,s,q, "back_model_index.fits",,,"Background model index DOL or filename"
outbgmDOL,  s,q,      "back_model.fits",,,"Background model DOL or filename"
outswgidxDOL,s,q,      "swg_idx.fits",,,"Science window group index DOL or filename"
#
# Task parameters
#=====
obsid,      s,h,"Test",,,"Observation Identifier"
ogid,       s,h,"Test",,,"Observation Group identifier"
purpose,    s,h,"Test",,,"Observation purpose"
engselect,  s,h,      "",,,"Energy band selection (blank=no selection)"
filterDOL,  s,h,      "",,,"Filter DOL (blank=no filter)"
inclReg01,  s,h,      "",,,"Sky inclusion region 1"
inclReg02,  s,h,      "",,,"Sky inclusion region 2"
inclReg03,  s,h,      "",,,"Sky inclusion region 3"
```

```

inclReg04,s,h,    "",,,,"Sky inclusion region 4"
inclReg05,s,h,    "",,,,"Sky inclusion region 5"
exclReg01,s,h,    "",,,,"Sky exclusion region 1"
exclReg02,s,h,    "",,,,"Sky exclusion region 2"
exclReg03,s,h,    "",,,,"Sky exclusion region 3"
exclReg04,s,h,    "",,,,"Sky exclusion region 4"
exclReg05,s,h,    "",,,,"Sky exclusion region 5"
#
# Standard parameters
#=====
clobber,b,h,yes,,,"Overwrite existing output data ?"
verbose,i,h,3,0,4,"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_add` 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:

- `inlist` specifies an ASCII file that contains a list of observation group filenames or DOLs. The data from these observations groups will be combined.
- `outgrpDOL` specifies the DOL or filename of the output Observation Group (HDU [GROUPING]).
- `outebdsDOL` specifies the DOL or filename of the output Energy Boundary Definitions (HDU [SPI.-EBDS-SET]).
- `outpntDOL` specifies the DOL or filename of the output Pointing Definitions (HDU [SPI.-OBS.-PNT]).
- `outgtiDOL` specifies the DOL or filename of the output Good Time Intervals (HDU [SPI.-OBS.-GTI]).
- `outdspDOL` specifies the DOL or filename of the output Detector Event Spectra (HDU [SPI.-OBS.-DSP]).
- `outdtiDOL` specifies the DOL or filename of the output Deadtimes (HDU [SPI.-OBS.-DTI]).
- `outpefDOL` (optional) specifies the DOL or filename of the output PSD efficiencies (HDU [SPI.-OBS.-PEF]). If the parameter is left blank, no PSD efficiencies will be combined in the output Observation Group.
- `outprfDOL` (optional) specifies the DOL or filename of the output PSD response factors (HDU [SPI.-OBS.-PEF]). If the parameter is left blank, no PSD response factors will be combined in the output Observation Group.
- `outbgmidxDOL` (optional) specifies the DOL or filename of the output Background Model Index (HDU [SPI.-BMOD-DSP-IDX]). If the parameter is left blank, no background model will be combined in the output Observation Group.
- `outbgmDOL` (optional) specifies the DOL or filename of the output Background Model (HDU [SPI.-BMOD-DSP]). If the parameter is left blank, no background model will be combined in the output Observation Group.
- `outswgidxDOL` (optional) specifies the DOL or filename of the output Science Window Group Index (HDU [GNRL-SCWG-GRP-IDX]). If the parameter is left blank, no science window group index will be combined in the output Observation Group.

- **obsid** specifies the observation identifier of the output Observation Group. This string will be written in the respective keyword of the Observation Group and FITS file headers.
- **ogid** specifies the Observation Group identifier of the output Observation Group. This string will be written in the respective keyword of the Observation Group and FITS file headers.
- **purpose** specifies the purpose of the Observation. This string will be written in the respective keyword of the Observation Group header.
- **engselect** (optional) specifies the energy bins that should be used to build the output Observation Group. One may either specify directly the bin indices (starting from 1 as given in the `SPI.-EBDS-SET` data structure) or energies in units of keV (in the latter case the parameter string has to be terminated by the acronym **keV**). A comma separated list has to be specified, where either individual bins or bin ranges should be specified. Examples are: `1,3-5,9,10-20` or `1800.0-1805.0,1813-1818 keV`. If the parameter is left blank, all energies will be selected.
- **filterDOL** (optional) specifies a pointing filter DOL of HDU type `SPI.-OBS.-FLT`. The pointing filter file specifies all pointings (or groups of pointings) that should be skipped while adding the input data. This allows for the implementation of a high-level data filter, avoiding unnecessary re-processing in case of changes in the data filtering.

If the parameter is left blank, no pointing filter is used.

- **inclRegnn** (optional) specifies 5 sky inclusion regions (where *nn* runs from 01 to 05). Only pointings that fall at least in one of the specified sky inclusion regions will be saved into the output Observation Group. This option allows to select pointings corresponding to sky regions from the input data. A sky region may be defined as either a circular region or a box.

For a circular region, 3 parameters are required: the centre of the circular region, specified in galactic coordinates, and the radius of the circular region. All quantities are given in units of degrees. For example, `0.0 0.0 10.0` selects a 10 degrees circular region around the galactic centre, or `0.0 +90.0 90.0` specifies the northern galactic hemisphere.

For a box region, 4 parameters are required: the galactic longitude interval and the galactic latitude interval. All quantities are given in units of degrees. For example, `-180.0 +180.0 -20.0 +20.0` specifies a band of  $\pm 20$  degrees along the galactic plane, or `-30.0 +30.0 -30.0 +30.0` specifies a 60 by 60 degrees rectangular region around the galactic centre.

Sky inclusion regions may of course overlap. If the parameter is left blank, no sky inclusion region is specified.

- **exclRegnn** (optional) specifies 5 sky exclusion regions (where *nn* runs from 01 to 05). All pointings that fall at least in one of the specified sky exclusion regions will be skipped. Similar to the inclusion regions (see **inclRegnn**), a sky region may be defined as either a circular region or a box.

For example, `0.0 0.0 30.0` excludes a circular region of 30 degrees from the data, while `-180.0 +180.0 -20.0 +20.0` excludes a band of  $\pm 20$  degrees along the galactic plane.

Sky exclusion regions may of course overlap. If the parameter is left blank, no sky exclusion region is specified.

- **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 background model index), the task will exit with an error message.
- **verbose** specifies the verbose level of the executable. 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. **verbose=3** provides a detailed report about the background model generation.

## 4 Interface definition

Data filtering is implemented in `spi_obs_add` using a file of HDU type `SPI.-OBS.-FLT`. This HDU contains a table with two columns, where the first (`PTID_SPI`) specifies the pointing to exclude, while the second (`REASON`) specifies the reason for exclusion. The exclusion reason will be dumped in the log file.

The pointing identifiers specified in the column `PTID_SPI` may be a full identifier (such as `00150007.000001`) or only a partial identifier (where trailing digits have been skipped). If a full identifier is provided, only the single pointing corresponding to the identifier is excluded. If trailing digits were omitted, all pointings will be excluded which match with the specified digits. For example, if `00150007` is given, all sub-pointings of the pointing `00150007.SSSSSS` will be skipped. If for example `0015001` is given, all pointings of the type `0015001P.SSSSSS` will be skipped, i.e. for example `00150010.000000`, `00150011.000000`, ..., `00150019.000000`. If only the revolution part is specified (such as `0015`), the entire revolution 15 will be skipped.

## 5 Algorithms

... to be written ...

## 6 Error codes

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

<code>SPI_OBS_ADD_ERROR_MEM_ALLOC</code>	-233400
<code>SPI_OBS_ADD_ERROR_BAD_FILE</code>	-233401
<code>SPI_OBS_ADD_ERROR_BAD_PARAMETER</code>	-233402
<code>SPI_OBS_ADD_ERROR_EMPTY_SET</code>	-233410
<code>SPI_OBS_ADD_ERROR_EMPTY_PNT</code>	-233411
<code>SPI_OBS_ADD_ERROR_EMPTY_GTI</code>	-233412
<code>SPI_OBS_ADD_ERROR_EMPTY_DTI</code>	-233413
<code>SPI_OBS_ADD_ERROR_EMPTY_DSP</code>	-233414
<code>SPI_OBS_ADD_ERROR_EMPTY_PEF</code>	-233415
<code>SPI_OBS_ADD_ERROR_EMPTY_PRF</code>	-233416
<code>SPI_OBS_ADD_ERROR_EMPTY_BGM_IDX</code>	-233417
<code>SPI_OBS_ADD_ERROR_EMPTY_BGM</code>	-233418
<code>SPI_OBS_ADD_ERROR_EMPTY_SWG_IDX</code>	-233419
<code>SPI_OBS_ADD_ERROR_EMPTY_SWG</code>	-233420
<code>SPI_OBS_ADD_ERROR_BAD_SET_NUM</code>	-233430
<code>SPI_OBS_ADD_ERROR_BAD_PNT_NUM</code>	-233431
<code>SPI_OBS_ADD_ERROR_BAD_DET_NUM</code>	-233432
<code>SPI_OBS_ADD_ERROR_BAD_GTI_DIM</code>	-233433
<code>SPI_OBS_ADD_ERROR_BAD_DSP_DIM</code>	-233434
<code>SPI_OBS_ADD_ERROR_BAD_PEF_DIM</code>	-233435
<code>SPI_OBS_ADD_ERROR_BAD_PRF_DIM</code>	-233436
<code>SPI_OBS_ADD_ERROR_BAD_BGM_IDX_NUM</code>	-233437
<code>SPI_OBS_ADD_ERROR_BAD_BGM_DIM</code>	-233438
<code>SPI_OBS_ADD_ERROR_NO_MINIMUM</code>	-233450
<code>SPI_OBS_ADD_ERROR_NO_MAXIMUM</code>	-233451
<code>SPI_OBS_ADD_ERROR_NO_BINS_SELECTED</code>	-233452

They have the following meaning:

- `SPI_OBS_ADD_ERROR_MEM_ALLOC` : the allocation of dynamical memory has failed. Probable your system resources are too limited to run this task.
- `SPI_OBS_ADD_ERROR_BAD_FILE` : unable to open ASCII input file.
- `SPI_OBS_ADD_ERROR_BAD_PARAMETER` : bad input parameter.
- `SPI_OBS_ADD_ERROR_EMPTY_SET` : `SPI . -EBDS-SET` data structure in one of the input observation groups is empty.
- `SPI_OBS_ADD_ERROR_EMPTY_PNT` : `SPI . -OBS . -PNT` data structure in one of the input observation groups is empty.
- `SPI_OBS_ADD_ERROR_EMPTY_GTI` : `SPI . -OBS . -GTI` data structure in one of the input observation groups is empty.
- `SPI_OBS_ADD_ERROR_EMPTY_DTI` : `SPI . -OBS . -DTI` data structure in one of the input observation groups is empty.
- `SPI_OBS_ADD_ERROR_EMPTY_DSP` : `SPI . -OBS . -DSP` data structure in one of the input observation groups is empty.
- `SPI_OBS_ADD_ERROR_EMPTY_PEF` : `SPI . -OBS . -PEF` data structure in one of the input observation groups is empty.
- `SPI_OBS_ADD_ERROR_EMPTY_PRF` : `SPI . -OBS . -PRF` data structure in one of the input observation groups is empty.
- `SPI_OBS_ADD_ERROR_EMPTY_BGM_IDX` : `SPI . -BMOD-DSP-IDX` data structure in one of the input observation groups is empty.
- `SPI_OBS_ADD_ERROR_EMPTY_BGM` : `SPI . -BMOD-DSP` data structure in one of the input observation groups is empty.
- `SPI_OBS_ADD_ERROR_EMPTY_SWG_IDX` : `GNRL-SCWG-GRP-IDX` data structure in one of the input observation groups is empty.
- `SPI_OBS_ADD_ERROR_EMPTY_SWG` : `GNRL-SCWG-GRP` data structure in one of the input observation groups is empty.
- `SPI_OBS_ADD_ERROR_BAD_SET_NUM` : invalid number of energy bins found in one of the input observation groups.
- `SPI_OBS_ADD_ERROR_BAD_PNT_NUM` : invalid number of pointings found in one of the input observation groups.
- `SPI_OBS_ADD_ERROR_BAD_DET_NUM` : invalid number of detectors found in one of the input observation groups.
- `SPI_OBS_ADD_ERROR_BAD_GTI_DIM` : invalid Good Time Interval dimension found in one of the input observation groups. Maybe the columns `OBT_START`, `OBT_END`, `TSTART` and `TSTOP` do not have identical dimensions.
- `SPI_OBS_ADD_ERROR_BAD_DSP_DIM` : invalid Detector Spectra dimension found in one of the input observation groups. Maybe the columns `COUNTS` and `STAT_ERR` do not have identical dimensions.
- `SPI_OBS_ADD_ERROR_BAD_PEF_DIM` : invalid PSD efficiency dimension found in one of the input observation groups.
- `SPI_OBS_ADD_ERROR_BAD_PRF_DIM` : invalid PSD response dimension found in one of the input observation groups. Maybe the columns `PSD_FP` and `PSD_FB` do not have identical dimensions.

- **SPI\_OBS\_ADD\_ERROR\_BAD\_BGM\_IDX\_NUM** : an inconsistent number of background model components have been found in the input observation groups.
- **SPI\_OBS\_ADD\_ERROR\_BAD\_BGM\_DIM** : invalid background model dimension found in one of the input observation groups.
- **SPI\_OBS\_ADD\_ERROR\_NO\_MINIMUM** : in the specification of the energy interval selection, no minimum has been found in an interval definition. Please verify the task parameters.
- **SPI\_OBS\_ADD\_ERROR\_NO\_MAXIMUM** : in the specification of the energy interval selection, no maximum has been found in an interval definition. Please verify the task parameters.
- **SPI\_OBS\_ADD\_ERROR\_NO\_BINS\_SELECTED** : no energy bins have been selected. Probably the specified energy selection does not cover the energy bins of the input observation groups.

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.