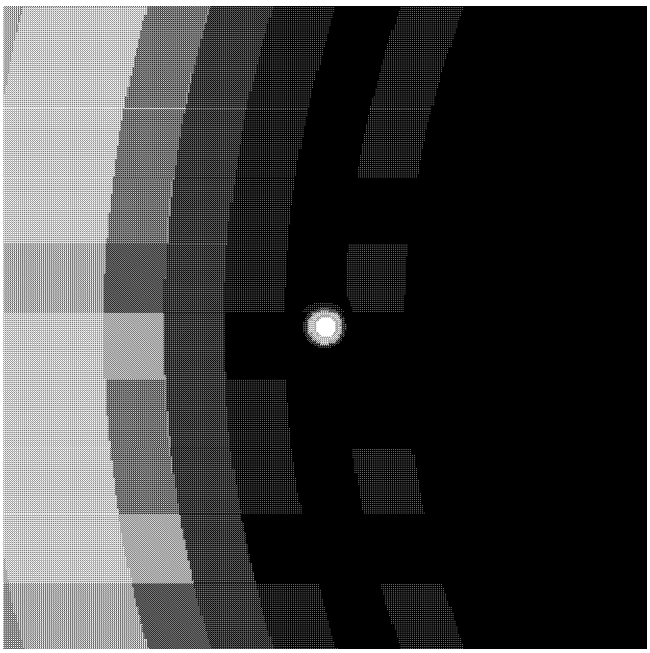
JEMXLIB USERS MANUAL



SDAST software for JEM–X

Document No. IN–PL–JEM–0027

Issue 1, Rev. 0

November 24, 2004

Picture on front page: Strong source on a skew background.

JEM–X
JEMXLIB USERS MANUAL

IN–PL–0027

Issue 1, Rev. 0

SDAST software for JEM–X

Prepared by:
Niels J. Westergaard, Peter Kretschmar,
Niels Lund, Stefan Larsson, Sami Maisala,
Carol Anne Oxborrow, and Silvia Martinez
DSRI

Date: November 23, 2004

Approved by: Date:

Approved by: Date:

DSRI

Table 1: REVISION NOTICE

Document Revision History			
Issue	**Rev.**	**Date**	**Changes**
1	0	December 14, 2004	Initial Release

List of Acronyms

BKG	Instrument Background
DXB	Diffuse X-ray Background
FIFOV	Fully Illuminated FOV
FOV	Field of View
FWHM	Full Width at Half Maximum
IROS	Iterative Removal of Sources
ISDC	INTEGRAL Science Data Center
ISSW	Instrument Specific Software
OBT	On Board Time
PIFOV	Partially Illuminated FOV
SDAST	Science Data Analysis Software Team

Contents

1 Introduction

The JEM–X shared library, JEMXLIB, is an effort with two objectives: 1) to facilitate the software maintenance and 2) to provide the developers with tools that already have been tested well.

1.1 IMOD data reading

The data from IMOD is organized in two C-structures (`JEMXdetectorData` and `JEMXregPixData`) in order to minimize the number of arguments in the function calls and to avoid large function changes when a new data item is to be used in a function.

`JEMXdetectorData` is the more general structure with mask and detector information and `JEMXregPixData` is focussed on the regularization process of the corrected detector pixels. The latter holds a large quantity of data and it should not be read in unless explicitly required.

2 Practical organization

The current version of the library can be found in
/home/njw/jemx/jemxlib/current

When a function is to be updated the developer should get it from the above mentioned directory and at the same time notify the entire SDAST. After changes the function must be delivered to NJW e.g. by email in order to avoid having several versions in play simultaneously.

3 Functions

3.1 General Functions

This is an entire section

3.1.1 JEMXLIBarrayGet

```
int JEMXLIBarrayGet(dal_element *group, char *arrayName, dal_dataType arrayType,
          int numAxes, long *dimAxes, dal_element **arrayPtr, void **array,
          int chatter, int callingStatus)
```

Description:
This is a general function to read a linear array with the data of the named child in the given group.

The array buffer is allocated within this function and must be freed in the calling function after the data has been mapped to some other array which includes reformatting to multiple dimensions.

Parameters:

`dal_element *group`:

Pointer to the group

`char *arrayName`:
String with data structure name

`dal_dataType arrayType`:
Data type

`int numAxes`:
Number of axes in the FITS array

`long *dimAxes`:
Dimensions of axes in the FITS array

`dal_element **arrayPtr`:
Pointer to the requested data structure

`void **array`:
Returned pointer to the data buffer

`int chatter`:
Level of chattiness

`int callingStatus`:
Status parameter at time of call

3.2 IMOD Reading Functions

3.2.1 JEMXLIBdetectorGetMap

```
int = JEMXLIBdetectorGetMap( dal_element *instModGrp, Instrument jemxInstr,
          unsigned short detMap[J_NUM_X_PIXELS][J_NUM_Y_PIXELS], double *onAxisArea,
          int chatter, int callingStatus)
```

Description:

Find the detector map for the given JEM-X instrument in the instrument model group and read in the data.

Parameters:

`dal_element *instModGrp`:
handle to instrument model group

`Instrument jemxInstr`:
instrument number

`unsigned short detMap`:
2D array with DETE-MOD values

`double *onAxisArea`:
active area for an on-axis source

`int chatter`:
Chattiness parameter

`callingStatus`:
error status on input

3.2.2 JEMXLIBdetectorGetCorXY

```
int = JEMXLIBdetectorGetCorXY( dal_element *instModGrp, Instrument jemxInstr,
          double detCorX[J_NUM_X_PIXELS][J_NUM_Y_PIXELS],
          double detCorY[J_NUM_X_PIXELS][J_NUM_Y_PIXELS],
          int chatter, int callingStatus)
```

Description:

Read in the two tables with the corrected pixel positions along the X and Y axis of the detector.

Parameters:

`dal_element *instModGrp`:
handle to instrument model group

`Instrument jemxInstr`:
instrument number

`double detCorX`:
2D array with corrected X positions

`double detCorY`:
2D array with corrected Y positions

`int chatter`:
verbosity control variable

`int callingStatus`:
error status on input

3.3 BKG Related Functions

3.3.1 JEMXLIBgetBkgModels

```
JEMXLIBgetBkgModels(dal_element *theSWG, float **bkgSpatModel,
          float **dbgSpatModel, short **bkgSidxMap, short **dbgSidxMap,
          float **bkgScalFact, float **dbgScalFact, float **bkgSpecCat,
          float **dbgSpecCat, char bkgSpecCatIdx, char dbgSpecCatIdx,
          int *numOBTvalues, OBTime **OBTvaluesScal, int *numSpecCatBkg,
          int *numSpecCatDbg, int jemxNum, int chatty, int status)
```

Description:

Get the models for background

Returns the status value.

Parameters:

`dal_element *theSWG`:
Pointer to the science window group

`float **bkgSpatModel`:
Spatial model array for instrument background

`float **dbgSpatModel`:
Spatial model array for diffuse background

`float **bkgSidxMap`:
Index map array for instrument background

`float **dbgSidxMap`:
Index map array for diffuse background

`float **bkgScalFact`:
Scaling factor array for instrument background

`float **dbgScalFact`:
Scaling factor array for diffuse background

`float **bkgSpecCat`:
Catalog of spectra for instrument background

`float **fdbSpecCat`:
Catalog of spectra for diffuse background

`float **bkgSpecCatIdx`:
Index of catalog of spectra for instrument background

`float **fdbSpecCatIdx`:
Index of catalog of spectra for diffuse background

`int *numOBTvalues`:
Returns number of OBT values

`OBTime **OBTvaluesScal`:
Returns array of scaling factors

`int *numSpecCatBkg`:
Returns number of spectra catalog

`int *numSpecCatDbg`:
Returns number of spectra catalog

`int jemxNum`:
Number of JEM–X unit, 1 or 2

`int chatter`:
Level of chattiness

`int status`:
Status parameter at time of call

3.3.2 JEMXLIBgetBkg

```
int JEMXLIBgetBkg(float *bkgSpatModel,float *dbgSpatModel,short *bkgSidxMap,
          short *dbgSidxMap, float *bkgScalFact, float *dbgScalFact, float *bkgSpecCat,
          float *dbgSpecCat, int Xpos, int Ypos, float *BkgValue, float *DbgValue,
          int chanMin, int chanMax, int numOBTvalues, int numSpecCatBkg,
          int numSpecCatDbg, int numGTI, OBTime OBTstartSWG, OBTime *OBTvaluesScal,
          OBTime *GTIstart, OBTime *GTIstop, int onTime, int chatter, int status)
```

Description:

Applies the background model.

Parameters:

`float *bkgSpatModel`:

Spatial model for instrument background

`float *dbgSpatModel`:
x

`short *bkgSidxMap`:
x

`short *dbgSidxMap`:
x

`float *bkgScalFact`:
x

`float *dbgScalFact`:
x

`float *bkgSpecCat`:
x

`float *dbgSpecCat`:
x

`int Xpos`:
x

`int Ypos`:
x

`float *BkgValue`:
x

`float *DbgValue`:
x

`int chanMin`:
x

`int chanMax`:
x

`int numOBTvalues`:
x

`int numSpecCatBkg`:
x

`int numSpecCatDbg`:
x

`int numGTI`:
a

`OBTime OBTstartSWG`:
a

`OBTime *OBTvaluesScal`:
a

`OBTime *GTIstart`:
a

`OBTime *GTIstop`:
a

`int onTime`:
a

`int chatter`:
: Chattines parameter

`int status`:
: Status parameter at time of call

Index