Instrument Specific Software for JEM-X

Architectural Design Document

Version 10.0

Carol Anne Oxborrow (DNSC)
Jérôme Chenevez (DNSC)
Peter Kretschmar (ESAC)
Stefan Larsson (Stockholm)
Sami Maisala (Helsinki)
Niels Jørgen Westergaard, (DNSC)

March 8, 2006

DANISH NATIONAL
SPACE CENTER

Contents

Change History Table

Date	Revision	Pages	Remarks
24/9/97	0.0.1	All	Skeleton based on [5] and [4]
22/6/98	1.0	All	First configuration control version released
12/8/98	2.0	28	Added a top-down decomposition of the components to the System Overview chapter.
13/8/98		**??**	Removed component JTimeCorrection since all time correction now done by ISDC.
		??	Removed list of TBD's which was only used by CAO and SAH
		All	Rearranged order of components to comply with MOU.
		171, **??**, 181	Edited contributions from SL to include components from data dictionary.
14/8/98		31	Added diagram of all the JEM-X ISSW components and subcomponents.
		18, 19, 20	Added diagrams from Peter Kretschmar
		??	Added a list of pending data structures.
24/8/98	3.0	273	Changed add.sty to include index of components and data items
25/8/98		All	Input and output environments created to include full list of inputs and outputs in the index of data structures
		??	Reinstated TBD list, with inclusions from other authors
2/9/98		203, 216,	Added PK's new components to JDataBinning to handle non-imaging background determination.
		203, 216	Moved components from JBackground to JDataBinning to handle non-imaging background analysis.
3/9/98		**??**	Renamed JBackground to JIntBkg for interactive, off-line background analyses.
		??	Renamed and added components for interactive background analysis.
		??, **??**	Added components into image generation (QL and SA), for automatic background analysis

3/9/98		31	Updated figure to include new non-imaging background components
	3.0	i	Update add.sty to make table of contents showing every component and data structure.
		All	All components updated to reflect delivery dates in MOU
		All	Included newly edited components from all authors.
8–9/10/98	3.0	All	Document reviewed by virtual meeting on IRC channel # jemxadr
15–21/10/98	3.0	All	Corrections arising from internal RIDs. See details below. See SDAST/ADD-RD for RIDs.
	Closed/rejected RIDs: 5,6,12,17,21,24,28,35,36		
15/10/98	RID38	108	CAO: Remove JBulgingCorrection all references to it elsewhere.
	RID07	All	ALL: All components edited to remove empty fields and add 'none' or 'TBW' where applicable
RID32,8,9,10		**??**, **??**, 247	CAO: Introduction to the Data Dictionary was updated to take out a a couple of pending data structures that never get used and to a other New Data Structures waiting for ISDC approval. A new list of Intermediate Data Structures also created and external data structures list was added to.
	RID15	108	CAO: Added JMXi-ELPS-CAL as input to JCorrection and JPerEventCorrections
16/10/98	3.0	**??**	Added new section to system overview to introduce background handling throughout the JEM-X ISSW
16/10/98	3.0	228	Included new data dictionary from PK that includes some auxillary and background data structures
	RID11	191, 206	PK: Edits to JDataBinning input list
	RID13	191, 206, 203	PK: Edits to 'dependancies' of JDataBinning and JBkgSpectrum
16/10/98	RID14	203, 216	PK: Update I-O lists of JBkgSpectrum and JBkg-Lightcurve
	RID20	**??**, 196	PK: j-data-preparation,JCreateSpectrumMode: Update output lists
	RID34	237	PK: JMXi-FRSS-CRW, Update remarks

	RID37	**??–??**	PK:JMXi-MODE-RAW, JMXi-CHAN-COR Consolidate design remarks on use of data structures across different telemetry modes.
	RID39	206	PK: Extend scope of JLightcurveBinning
19/10/98	RID26	228	CAO: Added paragraph to intro. about grouping of idividual source files for each ScW.
	RID26	108	CAO: Added intermediate data files throughout the data correction components.
	RID31	**??**	NJW: Added section on background handling and image reconstruction.
20/10/98	RID02	161	NJW: Added JMXi-EVWT-BKG to inputs of image construction components.
	RID03	**??**	NJW: Added design remarks about handling of restricted imaging data.
	RID08	228	NJW: Added pending New Data Structures and intermediate files to list in introduction.
	RID30	161	NJW: Updated inputs/outputs
	RID07	All	ALL: Updated all components to include 'TBD' or 'None' in all empty fields as appropriate.
	3.0	31	Updated overview figure to include JCatBkg and JScal under JModelAutoBkg and added appropriate text.
26/10/98	3.0	**??**	CAO: Added pending data structures from the Helsinki group to New Data Structures section.
	RID01	**??**	KM: JQlAutoBkg,Corrected input list.
	RID04	**??**	KM: Add comments about use of JMXi-CHAN-COR data.
	RID18	**??**	KM: j-known-src-calib, Correct input list to use uncorrected data.
	RID19	**??**	KM: j-ksc-gain-table, Correct output list to include JMXi-SPGN-PER.
	RID22	223	KM: Add functionality to display JMXi-ALRT-LIM.
	RID25	**??**	KM: Update the 'processing' field. Closes AI04.
	RID27	**??**	KM: JFiltAutoBkg, Correct inconsistencies in the text. Closes AI06
	RID29	141, **??**	KM: Added missing inputs belonging to subordinate components.
	RID31	**??, ??, ??**	KM: JQlAutoBkg, JSaAutoBkg, Added subsection in System Overview to describe how background is handled. Added design remarks in components wherever background is determined. This closes AI03.

26/10/98	RID33	223	KM: Clarified use of JMXi-GAIN-PER. This closes AI07.
	3.0	223, 28	KM: Removed JPerfAlertGeneration, JPerfFunctions, JPerfAlertLimits from JPerformance. CAO: Removed these components from overview figure and system decomposition.
2/11/98	RID23	**??**	Added SL's contribution to description of background handling in System Overview. This completes RID23.
	RID40	171, 181	SL: Added design remarks about determining background probabilities for photons.
	3.0	191, 206	Added PK's edits to JDataBinning due to new output JMXi-SIMU-CAT from JCatBkgSel. Figures also updated.
6/11/98	4.0		Version 4.0, with post-review corrections published
2/3/00	5.0	ALL	Complete revision of document. Changes include: All new component names, lower case with underscores Completely revised overview of background handling New introductory section on GTIs List of data structures incorporated into ISDC reference platform 1.4 (released Feb.2000) Basic templates (column structure) for each data structure not included in RP1.4 All diagrams updated with new naming system Most subordinate scripts removed from AD Automatic background handling moved to separate application j_bkg_model Background handling streamlined to 3 executables after AD work on this area in 1999 JSciAnalysis simplified to executables to create output suitable for use with standard astrophysical analysis tools (e.g. XSPEC) Updated Bibliography to include ISDC support software manuals; SDAST discussion documents

		??, 108	Updated j_calib_auto and j_correction to include INDEX table for calibration spectra and gain calibration results
13/03/01	6.0	ALL	Complete revision of document. ALL sections updated. Major changes are:
		8	Separate chapter for Software Requirements overview
		2, All	Purpose of ADD expanded to include DD
		62, 31, 28	Complete rearrangement of all components to fit into ISDCLEVL architecture
		21	Added section on ISDC levels
		39	Added chapter on detailed design
		62	Added a new figure and component for every pipeline application script, to conform with initial ISDC V2 integration of the software
		??	j_known_src_calib (KSC) becomes j_ical
		228	Data dictionary updated to ISDC template release 2.0
21/05/01	6.1	31, 32	Updated figures and overview to merge j_cor_gain_events and j_cor_gain_spectral
		62	Update of all component descriptions delivered for V2 consolidation
		57	Added error code lising to DD section
27/07/01	6.2	15, 62	Updated V2 component descriptions for latest consolidated deliveries
		??	Added section about detector coordinates
		62	Added indexing and cross-referencing of input parameters for every component description
		228	Data dictionary brought up to date with ISDC RP3.0 release templates
		38	Added a list of proposed tools that could be envisaged for the future
26/11/01	6.3	223 – **??**	Updated all component descriptions for ISDC interactive tools
31/03/02	7.0	All	Complete editorial revision of the text to make document as complete as possible and as self-consistent as possible. Major classes of changes:
		39	Added missing parameters, so all parameters can be indexed and cross-referenced
		232	Added missing data structures

31/03/02		62	All component descriptions improved though no design changes A complete list of changes was submitted to ISDC and can be found at:
			http://www.dsri.dk/˜oxborrow/sdast/Changes.for.7.0.ps
			Performing these corrections closes SCREW–00481
21/8/02	7.1		All components updated to versions delivered for V1 freeze at July 31st 2002
		141	Automatic background handling has new improved figures
		39	Script parameters added to list of input parameter names
		232	Data dictionary had obsolete data structures removed A complete list of changes was submitted to ISDC and can be found at:
			http://www.dsri.dk/˜oxborrow/sdast/ChangesFor7.1.ps
24/10/03	8.0	Title page	Jerome added to list of authors
		Chapter 5	All component descriptions updated to describe versions found in OSA 3.0. Each change in this document corresponds to a SCREW or SPR write against the relevent s/w component
		Chap. 6	All data structure descriptions updated for OSA 3.0
		Sec. 4.1	All parameter names updated for OSA 3.0
		Sec. 1.4	Expanded to include data naming convention abbreviations
		Sec. 3.7.15	Removed all references to j-ical-dead since deadtime is not determined by a trigger-based model any longer
		60	Added Jerome's summary of flux units A complete list of changes was submitted to ISDC and can be found at:
		Chapter 5	Complete revision of SRD references for all components
		63	Section added listing generic ISDC tools used in JEM-X ISSW
		270	Update of publication dates of all documents in the bibliography
			http://www.dsri.dk/˜oxborrow/sdast/ChangesFor8.0.ps

25/06/04	9.0	**??**, **??**, **??**	Added NJW's new j-ima-fine-resol to component descriptions; SRD compliance; system overview figures
		62	All component descriptions updated to describe versions delivered as part of OSA 4.0
		21, 30, 31	Adjusted ISDC levels to include COR in SCW pipeline and use -ALL data structures; in system overview
		232	Added -ALL data structures to data dictionary
		30, 31	Overview figures updated to show COR steps in the SCW Pipeline, as well as OSA
		??	Added dimImage parameter to parameter list
		63	Added short description of evts-pick tool to 'ISDC Generic Components'
		250, 232	Added JMXi-BORE-MOD and JMXi-FMSK-MOD to data dictionary and IMOD listing
		115	Added figure showing method of finfing the keV-per-ADC-Channel gain slope
6/12/04	9.1	30, 31	Updated figures to show different placing of data preparation in CONS/NRT and Offline data analysis
		20	Updated figure to add IMA2 level and remove unrealisable ICAL tools
		21, 22	Added to text and added new table to show how levels can be repeated in offline analysis if so desired
		62	All component descriptions updated to describe software delivered for OSA release 4.2
		252	Description updated to correspond to version 7.25 of jemx.h. Removed Change History Log from the description
		232, 250	Added pixel models and regularised grid data structures to data dictionary and to IMOD description; JMXi-PLLX-MOD, JMXi-PLLY-MOD, JMXi-REGU-MOD
		232	Flagged all IMOD data structures as having level IMOD, and explained this in introductory text

14/02/06	10.0	164, 8	Replaced j-ima-basic-recon, j-ima-src-find, and j-ima-fine-resol by new component j-ima-iros
		219, 8	Added new component j-ima-mosaic
		62	Imaging and mosaicking component descriptions updated to describe software delivered for OSA releases 5.0 and 5.1
		232	Added JMXi-BPL.-GRP and JMXi-BPL.-GRP-IDX as well as JMXi-[DMAP DALL DAST DAWL DAWP MHOL]-BPL to Data Dictionary; updated JMXi-SKY.-IMA-IDX
		39	Added j-ima-iros and j-ima-mosaic parameters; removed j-ima-basic-recon, j-ima-src-find, and j-ima-fine-resol parameters
		18	Updated figure NRT data processing
		31, 32	Updated figures OSA pipeline and Offline overview
		75	Updated figure jemx-obs-analysis
		Title page, All	Changed DSRI by DNSC with new logo
		Title page, 206, 191, etc.	Changed Peter's affiliation to ESAC
March 8, 2006	10.0	All	Submitted PostScript file to WWW

Where ?? appears instead of a page reference, the original component or data structure is no longer contained in this document because of changes made during the evolution of the software architecture and naming conventions.

List of Figures

Chapter 1

Introduction: the JEM-X ISSW ADD

The lay-out of this document, the JEM-X ISSW ADD, follows the template given in the ESA Software Engineering Standards [4]. The architectural design phase is described in [4] chapter 4, pp. 1-29 to 1-37 and the template for this document is given on p. 3-C4. The lay-out of this document is also heavily based upon the ISDC OSM Architectural Design Document [5]. Anyone coming fresh to the JEM-X ISSW should make liberal use of the index which cross-references all the functions, data structures, input parameters etc. Newcomers are also advised to have a copy of the ISSW User Manual [30] close at hand.

The document has also been expanded since version 5.0 to include the Detailed Design documentation required for maintenance and upgrading of the JEM-X ISSW. The requirements for Detailed Design documentation can be found in [4] chapter 5 p. 1-46.

Throughout the text, hyphens in component names can be freely interchanged with underscores. i.e. j-dead-time-calc is the same component as j_dead_time_calc. In the ISDC system only underscores are used, but these cannot be used in labels and other cross-referencing identifiers in LaTeX.

JEM-X actually consists of two identical units that sit side-by-side on the satellite, with the same FOV though rotated by 180° to each other around the spacecraft x-axis. It is necessary to treat the two instruments completely separately since they have slightly different boresight angles and completely different instrument characteristic tables. All data structures therefore come in two different forms: JMX1-ABCD-EFG for JEMX-1 and JMX2-ABCD-EFG for JEMX-2. Throughout the document data structures are call JMXi-ABCD-EFG to indicate these two versions. The two instruments have their own separate pipelines and all processing of data from them is kept entirely separate. The only place where data from both instruments is used together is in the interactive health monitoring and performance tool j_performance where results and data from the two instruments are compared.

1.1 Purpose

This document reflects the JEM-X Science Data Analysis Software Teams's (SDAST) understanding of what Instrument Specific Software (ISSW) should be delivered for integration into the ISDC data handling system. It also details the way components and shared data structures are used by the completed data analysis system. During the evolution of this document individual members of the SDAST and the hardware team can influence and develope the way their components fit into the whole.

Hence the JEM-X ADD has several purposes:

- It serves as a tool for SDAST to communicate internally current understanding regarding the JEM-X ISSW.
- It serves as a means of communication between SDAST, the hardware developers and ISDC
- It formed the basis of the Detailed Design Phase of software development
- It is a tool for the people responsible for integration of the software into the ISDC system
- It is a central repository of design details needed by future programmers not currently part of SDAST or ISDC to understand what we have done and why.
- It contains detailed design information and therefore is the definitive document on JEM-X ISSW. Generally information in this ADD takes precedence over that found in other documents.

1.2 Scope

The architectural design phase of development has produced a physical model of the software components to be written. Hence, this document reflects the current physical model of JEM-X ISSW, as agreed upon by the instrument team members. Changes in this document reflect physical changes in the way the software is realised.

Since there is JEM-X science data analysis software in many different parts of the ISDC system, it has not been attempted to make an overall model of the software system. Instead, it has been attempted to identify the context within ISDC where specific JEM-X ISSW components belong. The assumed context is the system described in the ISDC Top-Level ADD [9].

This document however, is not concerned with the common elements of the general ISDC environment, which are taken for granted: the exact location of data depositories and the running of archives, for example. It is merely assumed that in using the **JMXi-** prefix for data structures, and **j_** as the first letter of all JEM-X ISSW components, this is sufficient to identify the JEM-X software dominion within the ISDC system.

This document is updated with release of the ISDC OSA software to general users: the current version represents the state of the software at the time just before the OSA 4.2 release, planned for December 2005. There has been much effort to ensure that all components and subcomponents documented here actually reflect a piece of software code delivered to ISDC for integration or planned for future deliveries

1.3 The ISDC System

It is assumed that the reader is familiar with the ISDC Support Software Libraries which form the backbone of the ISDC I/O and archiving system. All data access is performed through these libraries. For information on the individual ISDC APIs please see the following user's manuals available from the ISDC website:

Common The common library used for executable initialisation and exit/tidyup [12]

DAL Data Access Library with functions common to all instruments [11]

RIL Reporting Interface Library provides logging/messaging and alert generation facilities [19]

PIL Parameter Interface Library accesses IRAF parameter files [18]

DAL3GEN General ISDC APIs for time conversion etc. [14]

DAL3AUX APIs for auxillary data access [15]

DAL3HK APIs for housekeeping data access [16]

DAL3CAT APIs for catalogue access [17]

DAL3JEMX APIs for access to science data in JEM-X specific formats [13]

isdcROOT Interface with ROOT classes for implementation of interactive tools [20]

1.4 Abbreviations

ADC Analog to Digital Converter

ADD Architectural Design Document

AH Allan Hornstrup, formerly head of SDAST

API Application Programming Interface

ARF Ancilliary Response Function/Files

BKG Background; suffix for data structures containing backgrounds

BTI Bad Time Interval

CAL Suffix for structures containing gain calibration data

CAO Carol Anne Oxborrow, SDAST, DNSC

CAT Suffix for structures containing catalogue data

cfitsio C library for FITS file I/O

CONS Consolidated data processing

COR Suffix for structures containing corrected/calibrated science data

CRW Suffix for structures containing raw calibration data

CTS ISDC Coding and Testing Standards document.

DAL Data Access Layer

DBG Data structure suffix for detector backgrounds

DFEE Digital Front-End Electronics

DPE Data Processing Electronics

DNSC Danish National Space Center (previously DSRI)

DSRI Danish Space Research Institute (now DNSC)

FCFOV Fully Coded Field of View

FITS Flexible Image Transport System

FOV Field of View

FRSS Fixed Radiation Source System

FULL Data obtained with a JEM-X unit in Full Imaging mode

GTI Good time interval. Also suffix for GTI data structures.

GUI Graphical User Interface

HK Housekeeping Data

HRW Suffix for data structures containing raw housekeeping data

IBIS Imager on Board the INTEGRAL Satellite

IC Instrument characteristics

IJD INTEGRAL Julian date

IMA Suffix for structures containing image data; ISDC analysis level

IR Instrument representative at ISDC

IRAF Astronomical image analysis system: Format for parameter files

IROS Iterative Removal Of Sources

ISDC INTEGRAL Science Data Centre

ISOC INTEGRAL Science Operations Centre

ISSW Instrument Specific Software

IT Instrument team

j_ Prefix given to any software component concerned with JEM-X

JC Jérõme Chenevez, SDAST DNSC

JEM-X Joint European Monitor for X-rays

JH Juhani Huovelin, head of background handling team, Helsinki

JMX1, JMX2 Actual prefix on JEM-X data structures

JMXi Wildcard prefix on JEM-X data structures; i = 1,2

KM Kirsi Malmivaara, formerly SDAST Helsinki

KSC Known Source Calibration

LM Logical Model

FOBT Full On-Board Time

LCR Suffix for structures containing lightcurve data

L-OBT Local On-Board Time

MOC Mission Operations Centre

MOD Suffix for structures containing instrument model data

MOU Memorandum of Understanding

NJW Niels Jørgen Westergaard, head of SDAST

NRT Near Real Time data analysis

OBSW On-Board Software

OBT On-Board Time

OSA Offline Science Analysis

OSim Observation simulator

OSM Operations System Monitoring

PCFOV Partially Coded Field of View

PIL Parameter Inteface Library

PK Peter Kretschmar, JEM-X Liason, ISDC

PRP Suffix for prepared data

PRW Suffix for raw packet data

QLA Quick Look Analysis

RATE Countrate data taken simultaneously with REST and SPEC data

RAW Suffix for structures containing raw (unpacked) data

RES Suffix for structures containing found source lists (results)

REST Data taken with a JEM-X unit in Restricted Imaging Mode

SA Standard Analysis

SCP Suffix for structures containing Science Calibrated Parameters

SCW ScW Science Window

SHD Suffix for structures containing shadowgram data

SPE Suffix for structures containing spectral data

SPI SPectrometer on INTEGRAL

SRW Suffix for secondary raw data

STA Suffix for instrument status data

SWG Science Window Group

SDAST Science Data Analysis Software Team

SL Stefan Larsson, SDAST Stockholm

SM Sami Maisala, SDAST Helsinki

SPAG SPAtial Gain variations

SPEC Data taken with a JEM-X unit in Spectral Mode

SPTI Data taken with a JEM-X unit in Spectral Timing Mode

SR(D) Software Requirements (Document)

TBC To Be Confirmed

TBD To Be Decided

TBW To Be Written

TIME Data taken with a JEM-X unit in Timing Mode

TIMM The INTEGRAL Mass Model

TM Telemetry

URD User Requirements document

V1 ISDC Preprocessing and data preparation/ Technical Processing

V2 ISDC Standard Pipeline Scientific Analysis/ Scientific Processing

V3 ISDC Offline Scientific Analysis

Chapter 2

Software Requirements Compliance

The processing performed by the JEM-X ISSW is determined by the ISDC SRD [7] and it is largely the arrangement of these requirements that has determined how the ISSW is broken down into different executables. This chapter outlines how the ISDC/ISSW applications and infrastructure implement the SRs for the JEMX ISSW. In some cases no one application or executable implements a requirement. In many cases several requirements are fulfilled by a single executable, especially where the same or similar requirements occur for the various data-taking modes.

All JEM-X ISSW applications fullfil requirement 18.1.1 of the SRD: coping with autonomous data-taking mode changes. Every executable searches for data of the type(s) for which it is written, and exits without error if it should turn out that there is no data of that type to work on in the given Science Window.

Since the launch of INTEGRAL in October 2002, many changes have been made to the ISSW which improve on the original fulfillment of some requirements so that they are now covered in more ways than before.

The JEM-X ISSW Software Requirements are fulfilled as follows:

18.1.1 All components cope with autonomous data-taking mode changes

18.2.1 Onboard software (OBSW) performs filtering of all measured events for calibration spectra counts from the FRSS. ISSW component j-cor-position (p. 121) also flags all events falling within areas known to be contaminated by calibration spectrum events.

18.2.2 OBSW in the DFEE sorts out calibration counts by their position on the detector—the spectra thus accumulated after 256 seconds are packaged in the HK telemetry by the DPE

18.2.3 OBSW stores calibration counts according to which FRSS source produced them

18.2.4 Uploadable parameters are used by OBSW to define areas on detector where FRSS counts fall. IC file JMXi-DETE-MOD (p. 235) indicates areas on the detector plate known to be contaminated with calibration spectra events.

18.3 Doesn't actually contain any requirements

18.4.1 j-calib-gain-fitting (p. 99) calculates current detector gain values from calibration spectra sent every 256 seconds.

18.4.2 The PHA binning tables provided in the instrument model group JMXi-IMOD-GRP (p. 239), including JMXi-FULB-MOD (p. 237) and JMXi-CALB-MOD (p. 233) are corrected for ADC non-linearity, which ensures that all events, scientific and calibration, are likewise corrected. These corrections are implemented by j-calib-gain-fitting and j-cor-gain.

18.4.3 All derived detector gain values are saved in an index table JMXi-GAIN-CAL-IDX (p. 238), with the OBT of the middle of the spectral integration period, by component j-calib-gain-fitting.

18.4.4 Executable j-cor-gain (p. 112) in application j-correction (p. 108) uses the fitted gain parameters to derive the correct gain of the detector.

18.4.5 All FRSS data is processed automatically by j-calib-gain-fitting (p. 99) to provide a running update of the detector gain stored in JMXi-GAIN-CAL-IDX (p. 238), which is used by j-cor-gain.

18.4.6 OBSW integrates the four FRSS spectra for 4.3 minutes (32 polling cycles of 8 sec.) which provides sufficient counts in each spectrum for the given source strength. See JEM-X User Manual [24] p.116.

18.5.1 Offline tool j-ical-gain (p. **??**) derives spatially distributed spectra and gain fitting parameters, and compares these for time variation, non-linearity etc.

18.5.2 All science data is corrected for ADC non-linearity by using PHA binning tables that are corrected for this problem, should it arise. The corrections are performed via j-calib-gain-fitting and j-cor-gain.

18.5.3 Inflight position corrections will be derived offline by the instrument team at DNSC, though it appears that these may not be needed.

18.5.4 Spatial resolution over the area of the detector is also determined by offline tools and by source extraction components j-src-spectra (p. 175) and j-src-lc (p. 185)

18.6 Doesn't contain any specific requirements.

18.7.1 Derived detector gain parameters are monitored by offline tools j-ical-gain (p. **??**) and j-ical-frss (p. **??**).

18.7.2 Changes in gain parameters between the different detector anodes are monitored by j-performance (p. 223) and studied interactively by j-ical-frss (p. **??**).

18.7.3 Changes in position corrections are monitored offline by the instrument team, using j-ical-collimator (p. **??**).

18.7.4 j-performance (p. 223) provides interactive monitoring of all instrument and data parameters being monitored.

18.7.5 Monitoring the evolution of inflight position corrections are done offline by the instrument team.

18.7.6 Interactive monitoring of the response matrix is provided by j-performance (p. 223).

18.8 Doesn't contain any specific requirements.

18.9.1 Binning data from higher information content modes to formats with less information (spectra and lightcurves) is done by j-bin-spectra (p. 191) and j_bin_lc j-bin-lc (p. 206), not by j_correction as assumed in the SRD.

18.10.1 Effective PHA binning tables in the instrument model group, JMXi-FULB-MOD (p. 237), JMXi-RESB-MOD (p. 241), JMXi-SPCB-MOD (p. 243), are corrected for ADC non-linearity. These data structures are used by j-cor-gain (p. 112).

18.10.2 Executable j-cor-gain (p. 112) corrects all science events for the detector gain on the relevant anode section.

18.10.3 Position measurements of all events are corrected by executable j-cor-position (p. 121), using preflight calibration tables and if necessary inflight calibration tables determined by j-ical-collimator (p. **??**).

18.10.4 Executable j-cor-gain (p. 112) also corrects science events for spatial variation in the detector gain using preflight calibration tables and if necessary inflight tables determined by j-ical-gain (p. **??**) and other offline analyses.

18.10.5 No bulging corrections are performed for the JEM-X data since it has been determined that this cannot be performed reliably for a coded mask instrument: SRD should be updated.

18.10.6 Derivation of On Board Times (OBT) for all data is performed by ISDC and it is their responsiblity to maintain the correct time correlation between the various instrument L-OBTs and Universal Time. This requirement is fulfilled by j-prp-evts-obt (p. 81), j-prp-spec-obt (p. 85), j-prp-rate-obt (p. 83) and j-prp-frss-obt (p. 87).

18.10.7 Both executables in j-correction (p. 108) evaluate the events as they are corrected for various 'goodness' criteria and raise flags if neccessary in the STATUS byte of each JMXi-****-COR data structure. Application j-gti (p. 126) handles all determination of good time intervals (GTIs).

18.10.8 Executables j-cor-gain (p. 112) and j-cor-position (p. 121) handle both FULL and REST events.

18.11.1 Use of effective PHA binning table JMXi-FULB-MOD (p. 237) ensures ADC non-linearity correction of all spectral-timing events. These data structures are used by j-cor-gain (p. 112).

18.11.2 Executable j-cor-gain (p. 112) corrects SPTI events for temporal variations in detector gain.

18.11.3 ISDC is responsible for ensuring correct time correlation of all OBTs for all events and all instruments.

18.12.1 As for 18.11.3.

18.13.1 Use of effective PHA binning table JMXi-SPCB-MOD (p. 243) ensures that all SPEC spectra are corrected for ADC non-linearity. These data structures are used by j-cor-gain (p. 112).

18.13.2 Executable j-cor-gain (p. 112) performs temporal gain corrections on the SPEC spectra.

18.14 Contains no specific requirements.

18.15.1 Executable j-bin-evts-spectra (p. 196) bins SPTI and other event data into spectra, the bin sizes being determined by input parameters in IRAF format.

18.16.1 Executable j-bin-evts-lc (p. 210) bins all event data into lightcurves.

18.16.2 Where the data formats have energy information, binned lightcurves produced by j-bin-evts-lc (p. 210) are also energy-resolved. The binning formats are determined by input parameters.

18.17.1 Application j-bkg-model (p. 141) calculates a background estimate automatically for every pointing. An iteractive tool, j-bkg-interactive (p. **??**) allows offline fine-tuning of background models.

18.17.2 Executablesj-bkg-spec-sel (p. 147) and j-bkg-scal-calc (p. 150) find suitable spectral background models for each pointing. New background spectra are created from observations by j-bin-bkg-spectra (p. 203).

18.17.3 Executables j-bkg-spat-sel (p. 144) and j-bkg-scal-calc (p. 150) provide suitable spatial background models automatically for every pointing.

18.18.1 Application j-imaging (p. 161) reconstructs images from shadowgrams, finds the positions of any found point sources and provides an error estimate on those positions.

18.18.2 All sky images produced by j-imaging (p. 161) are stored in data structures with the same format, JMXi-SKY.-IMA (p. 243) and the different images indexed in table JMXi-SKY.-IMA-IDX (p. 243), by j-ima-iros (p. 164).

18.18.3 This requirement is fulfilled by the image reconstruction component j-ima-iros (p. 164) using the standardised image data structure JMXi-SKY.-IMA (p. 243).

18.18.4 The component j-ima-iros (p. 164) corrects for spatial and spectral background counts as provided in models from j-bkg-model (p. 141).

18.19.1 Application j-imaging (p. 161) produces sky images, found source lists and estimates of uncertainties on found source positions.

18.19.2 Application j-image-bin (p. 154) creates energy- and time-resolved shadowgrams to be processed into images. Time and energy resolution is determined by input parameters.

18.19.3 Executable j-ima-iros (p. 164) is based on an IROS procedure producing high quality sky images and point source estimates.

18.19.4 Application j-imaging (p. 161) can produce full FOV images which include all found sources.

18.19.5 The component j-ima-iros (p. 164) returns images of events left over after removal of all found sources.

18.20.1 Application j-imaging (p. 161) produces refined images using j-ima-iros (p. 164).

18.20.2 Details of the collimator, mask support structure etc. are applied during image processing in application j-imaging (p. 161).

18.20.3 Neighbouring images can be merged by using the image mosaicing tool dedicated to JEM-X j-ima-mosaic (p. 219) including special weighting functions that take into account the strong vignetting of the FOV, not seen in the other instruments that don't have a collimator.

18.21.1 A list of found point sources is created by j-ima-iros (p. 164) for use in QLA.

18.21.2 Coarser, quicker, shadowgram binning can be done by j-image-bin (p. 154) for use by QLA.

18.21.3 This requirement is now obsolete by today's computer performances.

18.21.4 j-ima-iros (p. 164) is run by QLA (see 18.19.3).

18.21.5 Sky images are produced in QLA including all the detected sources.

18.21.6 As a consequence of 18.21.4 a residual image is also returned.

18.22.1 Executable j-ima-iros (p. 164) corrects found source fluxes for systematic detector sensitivity such as that caused by partial coding and collimator shadowing.

18.23.1 For every found point source applications j-src-spectra (p. 175) and j-src-lc (p. 185) create time-resolved spectra and energy-resolved lightcurves.

18.24.1 The binning format of the individual source spectra is determined by the input parameters to j-src-spectra (p. 175).

18.24.2 Executable j-src-spectra (p. 175) produces background spectra for each found point source.

18.25.1 The time resolution of the found source lightcurves is determined by the input parameters to j-src-lc (p. 185).

18.25.2 Executable j-src-lc (p. 185) produces associated background lightcurves for each found point source.

18.26 Scientific analysis of the standard products is performed by generic ISDC tools and other widely available packages such as XSPEC and XRONOS. Data structure format of various end products have been designed to be compatible as input to these packages.

Some requirements that appear in ISDC SRD are no longer applicable to the instrument as we currently understand it (18.10.5, 18.21.4, 18.21.6), and other functionality has been added to accomodate changes in the instrument design. Additional requirements of the JEM-X ISSW are:

Countrate data The original URD and SRD did not forsee the need for countrate data, RATE, hence there are as yet no formal requirements for its processing.

Background binning The need to bin empty fields into background light curves and spectra, to be used to construct a background model catalogue was unforseen in the original SRD. This is done in j-bin-bkg-spectra (p. 203) and j-bin-bkg-lc (p. 216).

Deadtime determination No separate requirement was envisaged for this calculation which was assumed to be a natural part of flux determination wherever this might occur. However, to streamline processing it has been decided to make deadtime calculation a separate executable early in the pipeline, so that the results can be used by all software further downstream.

Verification of prepared data It has been decided to include an executable at the end of data preparation to check that the raw data going into the Scientific Analysis pipeline behaves correctly. This includes checks on packet data versus event data, grey filter changes, OBT evolution, header keywords, out-of-bounds values etc. This will allow us to uncover problems in OBSW, ISDC PreProcessing and Data Preparation as quickly as possible. See j-prp-verify (p. 95).

Status table A central instrument table is needed for each JEM-X unit giving a time-ordered list of grey filter and mode changes. This is done by j-prp-status-table (p. 89)

OEM message display SDAST has provided to ISDC OSM a function that provides warning strings for each On Event Message that can be produced by the housekeeping telemetry, so that these can be displayed by OSM.

ADC linearity monitoring Automatic pipeline component j-calib-adc (p. 104) checks that all 43 amplifier chains are still working linearly and produces alerts when any of them significant changes in behaviour.

Dead anode monitoring Since launch cosmic ray damage to the microstrip plate has caused some anode strips to stop functioning, either wholly or partially. Component j-prp-verify (p. 95) has added functionality to output normalised event populations for each anode which can be used to check anode health throughout the mission.

Monitoring of ECAL data The onboard electronic calibration data is monitored automatically by j-calib-adc (p. 104) in the Revolution File Pipeline one revolution at a time. Longterm trends in this data are monitored by offline tool j-ical-ecal.

Chapter 3

System Overview

3.1 Objectives

The purpose of the JEM-X science data analysis software can be listed as follows:

- To prepare the raw data for processing.
- To check the raw data for anomalies indicating detector or on-board software malfunctions or problems with ISDC PreProcessing.
- To calculate time-dependant detector variables: deadtime and energy gain
- To remove detector fingerprints from the received event lists
- To make shadowgrams from corrected event positions and energies
- To determine the sky background present on each sky image, and in non-imaging data.
- To deconvolve the shadowgrams with the instrument mask to produce images of the sky
- To find bright sources of x-rays on the sky images that may be of interest to astronomers
- To extract data such as spectra, lightcurves and source flux estimates for each source found.
- To perform a running check on the instrument's performance
- To assemble a catalogue of background images for the detectors.
- To monitor raw and corrected data to see if fingerprint correction tables need updating.

Many tasks that are an integral part of Scientific Analysis are performed by ISDC tools that run on data collected by all the INTEGRAL instruments not just JEM-X. They are described in the relevant ISDC documentation:

- To monitor housekeeping data for anomalies and provide GTIs for each instrument (OSM)
- To extract lists of catalogue sources that should appear in the FOV and match them to sources found by the instruments (CAT)
- To perform rapid analysis of image data to see whether new sources or changes to known sources have been discovered (QLA)
- To make mosaics of neighbouring sky images (Observation pipeline)
- To select all data products belonging to just one source (Observation pipeline)
- To take INTEGRAL data products and reformat them for use with standard analysis tools (e.g. XSPEC and XRONOS)

3.2 System Design

Originally, each component from the highest level of the SRD [7] was appointed to a group or person in the SDAST. However, the AD has evolved so that in many cases this distinction between applications and developers is no longer so clear cut. The evolution of the AD, and the ADD, has been driven by several external constraints, which have themselves evolved substantially since ADD5.0: the design of the instrument; the design and function of the onboard software; the ISDC libraries; the output from ISDC PreProcessing ; stylistic demands of ISDC Coding and Testing Standards and the architectural design of the ISDC pipelines.

The ISSW is also dependant on changes in the onboard software that reads in data directly from the instrument in the DFEE[22] and packages telemetry in the DPE[23]. These OBSW modules determine the format of data received for processing: the coarseness (or otherwise) of energy bins; the frequency of new data; time resolution; data-taking mode and grey filter value. The number of bytes (or bits) assigned to each sort of telemetry data is reflected in the data structures which must store the data.

A major change in the architecture since ADD5.0 has been the complete re-arrangement of the individual executables into applications so that the automatic pipeline applications each belong to one of the ISDC Levels (PRP, COR, GTI etc.) which are used across all instruments. Also, generic ISDC tools used for all instruments are now included in the applications. One application, j_sci_format (previously JScientificAnalysis) is now composed entirely of ISDC tools and contains no ISSW.

The ISDC system, especially PreProcessing, determines to some extent the point in the data analysis pipeline at which certain sorts of raw data are available. OBTs for example, cannot be calculated in PreProcessing and so we must wait for Data Preparation to produce these values. Whether data can be treated on a Science Window by Science Window basis is also determined by the ISDC system: in some cases it has been found preferable to treat calibration and configuration data for each Revolution or by Observation Groups. Groups and their implementation in the ISDC system also provide additional constraints on programmers.

One guiding principle of the ISSW design is that no programmer downstream should need to re-assemble data already available further upstream: hence the presence of executables such as j_prp_status_table and j_dead_time_calc both of which assemble information found in several different data structures for use later in the pipeline.

It has generally been assumed that the QLA and SA pipelines use the same executables, in some cases with coarser binning.

The background handling design has been formulated with help from experts who have experience in the field of coded-mask image reconstruction, and with the aid of background simulations using TIMM and the Southampton (Soton) photon generator software. Consequently, applications j_bkg_model and j_bkg_interactive have evolved significantly since versions 4.0 and 5.0 of this document. Background handling procedures are discussed in detail in section 3.6.

While every effort has been made to create an AD that meets all the User Requirements and Software requirements as best possible given the external constraints, it was clear that some evolution of the ISSW should be expected after the launch of INTEGRAL, especially in the Commissioning Phase. It was also forseen that problems that arise with the instrument during the mission will require correction by the ISSW. Consequently, SDAST has tried to build as much flexibility into the AD as possible, especially for calibration, correction and background handling. Interactive, offline analysis of the data allows us to fine-tune these procedures routinely during the mission.

There is only a very simple Observational pipeline for JEM-X since the instrument cannot gain much by merging raw data from neighbouring science windows in a dithering pattern the way the larger instruments can. Improvements in the detection of sources in the PCFOV can be achieved, by simply mosaicing the sky images produced by individual science windows. Currently the observational pipeline collects together useable Science windows containing a desired source, from all the possible pointings, and uses these to make a mosaic image of the sky around this position.

SDAST has chosen to use C as the sole programming language for its V1/V2 pipeline source code. C++ and ROOT are also used for some V3 interactive tools and all of the scripts which link the components for pipeline and standard scientific analysis (OSA).

3.3 ISDC System Context

The figures on pages 18, 19 and 20 show how the JEM-X ISSW components fit into the ISDC system.

All the components of the JEM-X ISSW should be integrated with the ISDC system. How this is achieved is documented in the ISDC Coding and Testing Standards [10]. Code is written in C and input parameter files are written in IRAF format.

It is assumed that the ISDC system, including all ISSW, runs in a UNIX environment (eg.

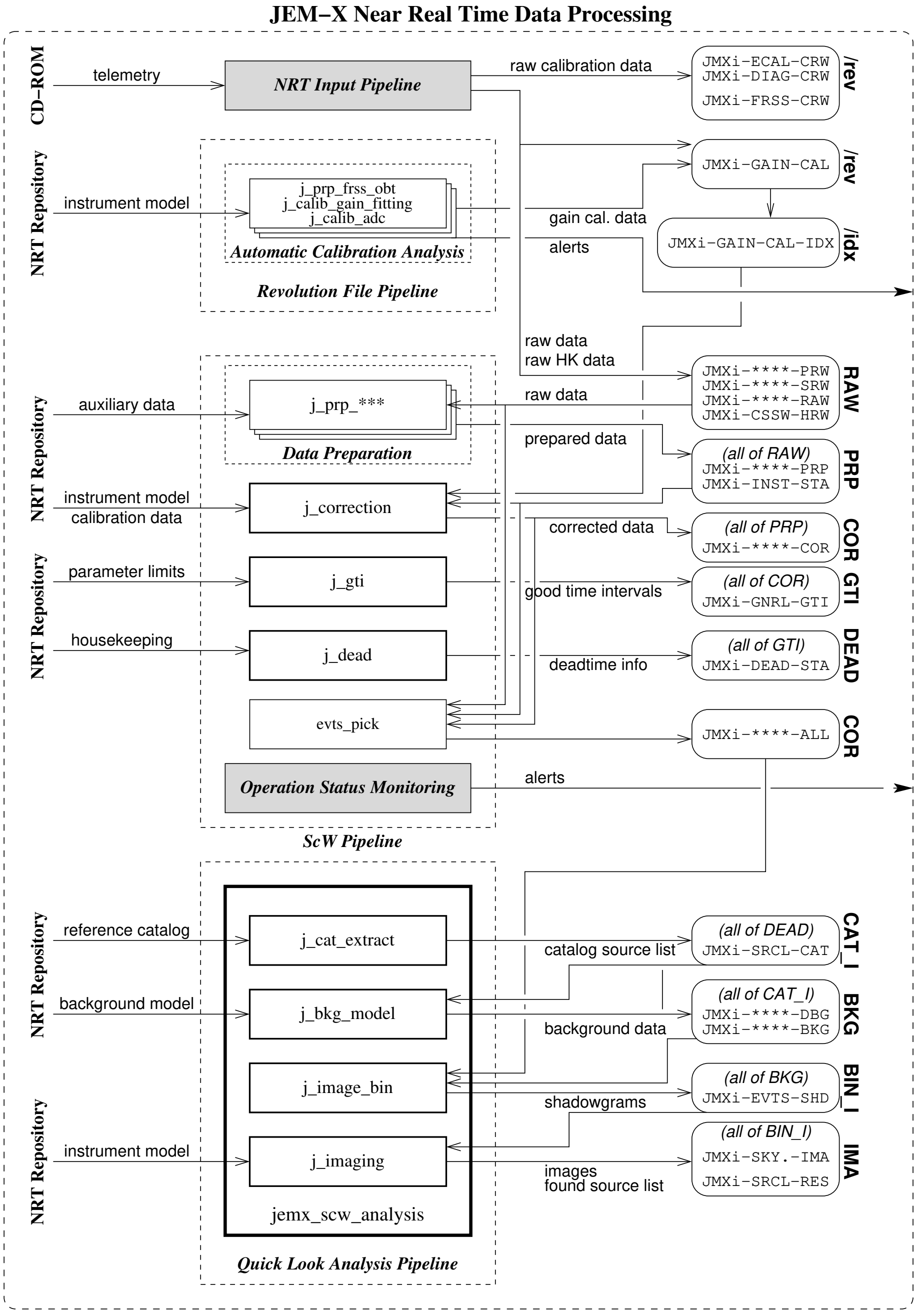

Figure 3.1: ISDC Context: JEM-X Near real time data processing

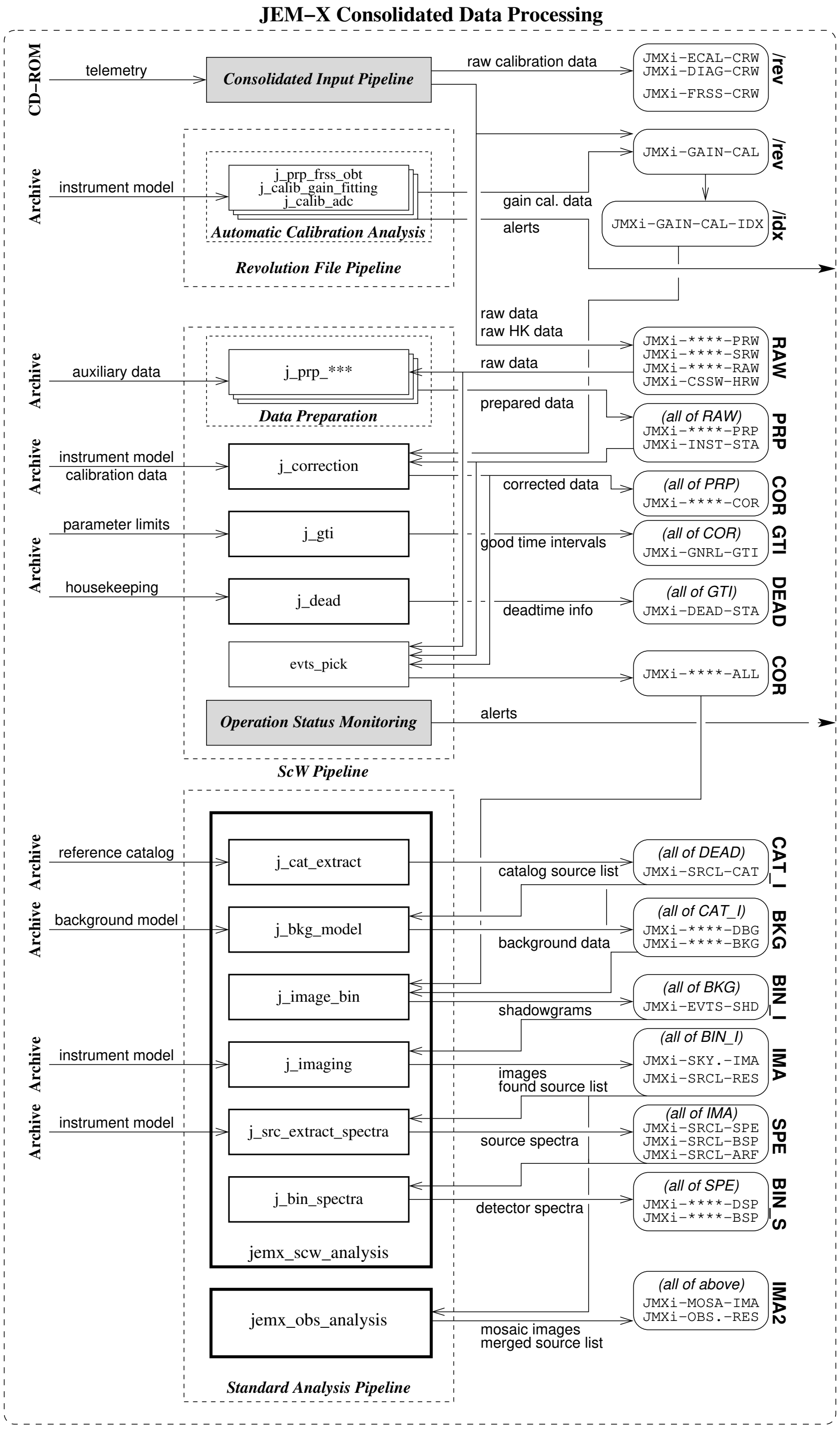

Figure 3.2: ISDC Context: JEM-X Consolidated data processing

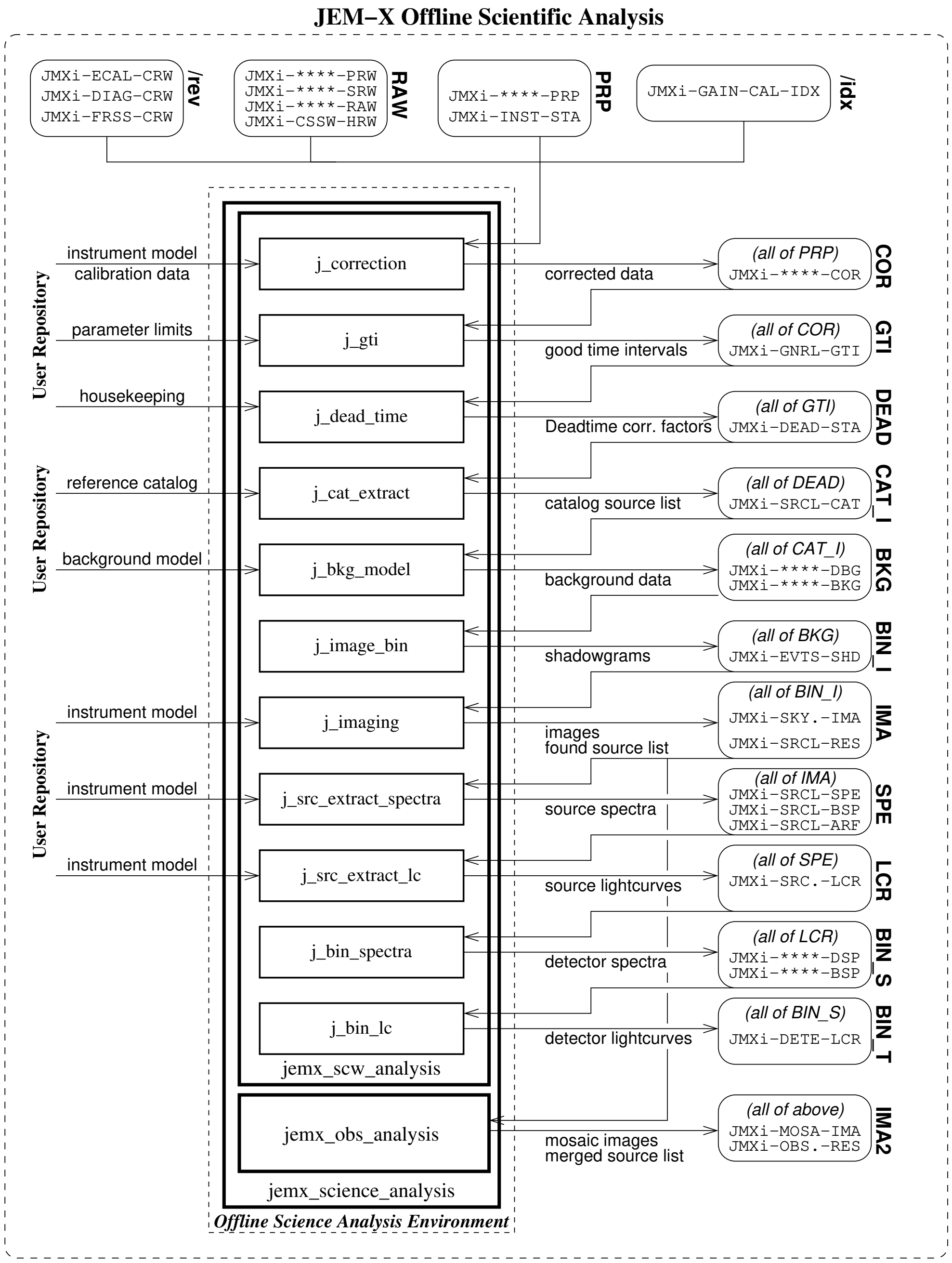

Figure 3.3: ISDC Context: JEM-X Offline Scientific Analysis

Sun/Solaris) on several workstations. Software has been developed either in the intended environment at ISDC, or on similar machines elsewhere.

Executables follow the “ftools” paradigm: callable from a UNIX command line, parameters supplied via IRAF-style parameter files, with overriding of the files via command line input. The automatic QLA/SA pipelines run without any direct user input. All run-time options are set using the parameter files accessed by the ISDC Parameter Interface Library (PIL)[18].

Similarly there are no direct-to-user output for the automatic analyses: messages, errors, warnings and alerts must be sent to log files via the Reporting Interface Library (RIL)[19]. Any alerts that are generated during running of the automatic pipeline are displayed to an operator who will decide on the appropriate course of action. Otherwise, results of a run can only be seen by opening the log file produced by RIL.

The use of DAL, PIL, RIL and the Common library, also supplied by ISDC, is not documented here, but merely assumed as part of the fundamental software structure. Where use of other libraries, such as NAG, SLALIB etc. is required, this is indicated in the specific component entry below.

Some functionality of executables may be put into libraries (lib__.a files in UNIX environment) where this avoids needless repetition, though this has not been done so far.

While all data required to produce the JEM-X user products is corrected using software specific to JEM-X, corrections to the onboard time sent as part of the JEM-X telemetry, and its conversion to universal time, is performed on a one-time only basis by ISDC. The OBTs given in the output data structures (JMXi-$* * **$-PRP) are used unconverted, since they are only needed to identify particular observing periods. For example, periods during which the grey filter had a particular value, or when the instruments were sending data in a particular format. The times used in these calculations do not require any outside reference to a standard time.

3.4 Scientific Analysis Levels

The scientific analysis of the three high-energy instruments onboard INTEGRAL has a degree of commonality, despite the various differences in detail. In a certain step, for example, events are corrected for instrumental fingerprints, in another events are binned into detector maps, and in yet another sky images are derived by image deconvolution.

In order to make this more transparent for scientists working with data from several instruments, so-called *Analysis Levels* were identified by ISDC and designated with unique labels. The order of these levels, the detailed processing and the details of the outputs may differ across instruments, but in general, a given level will mean similar tasks and similar outputs for JEM-X, IBIS and SPI. The list of all levels and more explanations can be found in the ISDC Top Level ADD[9].

Note that Analysis Levels are only defined for science data. Calibration and Instrument Configuration (IC) data have no such levels defined.

ISDC Near Real Time and Consolidated processing both include the data preparation, good

Science Window Pipeline (CONS and NRT)	
RAW	Raw data as produced by PreProcessing
PRP	Prepared data, including Full OBTime information
COR	Corrections for instrumental fingerprints
GTI	Good-Time Interval calculation
DEAD	Deadtime history calculation
Standard Analysis and Quick Look Analysis	
CAT_I	Catalogue source selection for imaging
BKG	Background model selection and derivation
BIN_I	Binning of event data into shadowgrams for Imaging
IMA	Image reconstruction, source flux determination
SPE	Spectra and response extraction for individual sources
LCR	Lightcurve extraction for individual sources
BIN_S	Binning of detector spectra
BIN_T	Binning of detector lightcurves
IMA2	Mosaic images and combined source list for Observation

Table 3.1: Analysis Levels defined for JEM-X NRT/QLA and CONS/SA processing

time intervals and dead time calculation levels so that archived data includes corrected as well as raw events. However, users of the offline science analysis software can choose to repeat these levels of processing on their private copies of the raw data. Equally, users can choose to retain the standard corrections derived by ISDC when processing their own data.

For JEM-X the analysis levels shown in table3.4 are defined.

3.5 Good Time Intervals

Good Time Intervals (GTIs) are used in the analysis to select only those data taken while the detector was considered to work correctly. The corresponding data structures consist simply of a list of start and stop times of those intervals considered “good”. Usually, these intervals are generated based on the values of certain housekeeping or science data parameters.

Within the Scientific Analysis, GTIs are created by a set of standard ISDC tools. These tools create GTI information based on the following data:

1. Housekeeping parameters which are compared with pre-set limits.

2. The satellite stability as recorded in the attitude data.

3. Gaps (lost packets) in the telemetry flow

Science Window Pipeline (NRT and CONS)	
RAW	Raw data as produced by PreProcessing
PRP	Prepared data, including Full OBTime information
COR	Corrections for instrumental fingerprints
GTI	Good-Time Interval calculation
DEAD	Deadtime history calculation
Offline Science Analysis (OSA)	
COR	Corrections for instrumental fingerprints
GTI	Good-Time Interval calculation
DEAD	Deadtime history calculation
CAT_I	Catalogue source selection for imaging
BKG	Background model selection and derivation
BIN_I	Binning of event data into shadowgrams for Imaging
IMA	Image reconstruction, source flux determination
SPE	Spectra and response extraction for individual sources
LCR	Lightcurve extraction for individual sources
BIN_S	Binning of detector spectra
BIN_T	Binning of detector lightcurves
IMA2	Mosaic images and combined source list for Observation

Table 3.2: Analysis Levels defined for JEM-X Offline Science analysis

The different GTIs are recorded in individual tables (`JMXi-GNRL-GTI`) which are grouped by an Index Group (`JMXi-GNRL-GTI-IDX`). The type of GTI is recorded in the attribute `GTI_NAME` of the individual tables. Access to the individual tables is possible transparently via the DAL3HK functions which allow selection on `GTI_NAME`.

In addition, users can create tables with specific good or bad times according to their own choices, e.g., to select only data around a burst. The information from these tables, relevant for a given Science Window are stored in another `JMXi-GNRL-GTI` table with `GTI_NAME = USER`.

Finally, all GTI data for a given instrument are merged into a summary table whose intervals mark time periods for which all checks confirmed "good" status (= a logical AND operation between various good times). This table is also a data structure `JMXi-GNRL-GTI` with keyword `GTI_NAME = MERGED`.

The analysis software will usually use the overall table. If in offline analysis the user wants to change the selection they regenerate the GTI information with their own choices, creating a new overall table for the analysis software.

In addition, certain analysis steps allow to define the JMXi-GNRL-GTI table(s) to be used via special parameters. This is mostly for testing and special circumstances.

3.6 The Background Problem

One of the most difficult problems faced in analysing the science data from JEM-X has been the approximation of the background events, especially since they are expected to arise from a number of different sources: instrument background, diffuse galactic background, unresolved sources etc.

From the observational point of view, the background consists of two main components, the sky background and the instrument background. The former does not come via the mask but from photons and particles interacting in the bulk material near the detector. The particle background is minimized by an on-board rejection algorithm, nevertheless it constitutes the larger background part. The sky background originates in the diffuse X-ray and unresolved sources. This part is most important at lower energies i.e. $< 10\ keV$ and it is modulated by the mask and the collimator.

In this section we discuss how the background problem has been solved by using background models that have been generated from source-free JEM-X science windows or from computational simulations:

- By finding the appropriate models and scaling factors with given selection parameter values.
- By using the spectral background models and scaling factors to produce background spectra and lightcurves for correcting non-imaging data.
- By using the spatial background models and scaling factors to correct the measured shadowgrams for background events before image generation.

3.6.1 Background Determination

The general purpose of the software is to provide an estimate of the background contribution in JEM-X observations. The estimate is the number of background counts at each *time instant* for each *position* and *energy channel* of JEM-X.

The background is handled as two separate components, the diffuse background and the instrument background. The determination of the background is a multistep procedure, which utilises an independent set of source data. There are separate source catalogues for each of the background components. These catalogues are created externally by using,

- Background simulations
- Blank field observations,

and their contents can be updated any time without a need to change the procedure. The generation of the final background estimate consists of the following steps:

- Spatial background models are produced by **j-bkg-spat-select**. The diffuse background model is selected according to the *sky coordinates* of the observation. The instrument background model is selected according to the *orbital phase, attitude parameters* and *radiation monitor level* during the observation. We assume that the spatial model does not change during single science window.
- Spectral background models are produced by **j-bkg-spec-select**. Spectral index maps and the corresponding spectrum models are selected according the multi dimensional array of relative parameters as *phase angle, solar aspect angle* and *radiation level*. Each detector pixel has an index value which corresponds to a spectrum model.
- Background scaling factors are calculated by **j-bkg-scal-calc**. The diffuse background scaling factor is calculated according to the *sky coordinates* and *time interval* of the observation. The instrument background scaling factor is calculated according to the *orbital phase, attitude parameters, radiation monitor level* and *time interval* of the observation. The tables of scaling factor are presented in 8 second periods in OBT format, which is consequence of telemetry interval of the housekeeping parameters.

The final result is an estimate of the diffuse and instrument background counts for the specified time interval in each spectral channel and spatial coordinate on the detector. The consistency between the background estimate and a specific observation will be controlled, and fine tuning, if necessary, will be made by the background software.

For the details of individual components (e.g. the selection of a model), see “design remarks” or “processing” of a specific component. The general functionality of the above tasks requires actions external to the framework of the JEM-X software development team.

- The usage of the blank field observation is obligatory to the well determined background modelling procedure and the feasibility of the plan is done by the in-flight calibrations of *INTEGRAL.*

3.6.2 Background spectrum and spatial model calculation

The main goal of this task is to elaborate a flexible and an adaptable background handling application. The application **j-bkg-model** generates six data structures to output: Spatial, spectral, and scaling factor data structures of the diffuse and instrumental parts of the background estimate. By making combinations of these data structures it is possible to generate the following background models,

- instrumental or diffuse sky parts of background model or sum of them.
- spectral part of the background model in any pixel or any subsets of pixels on the detector

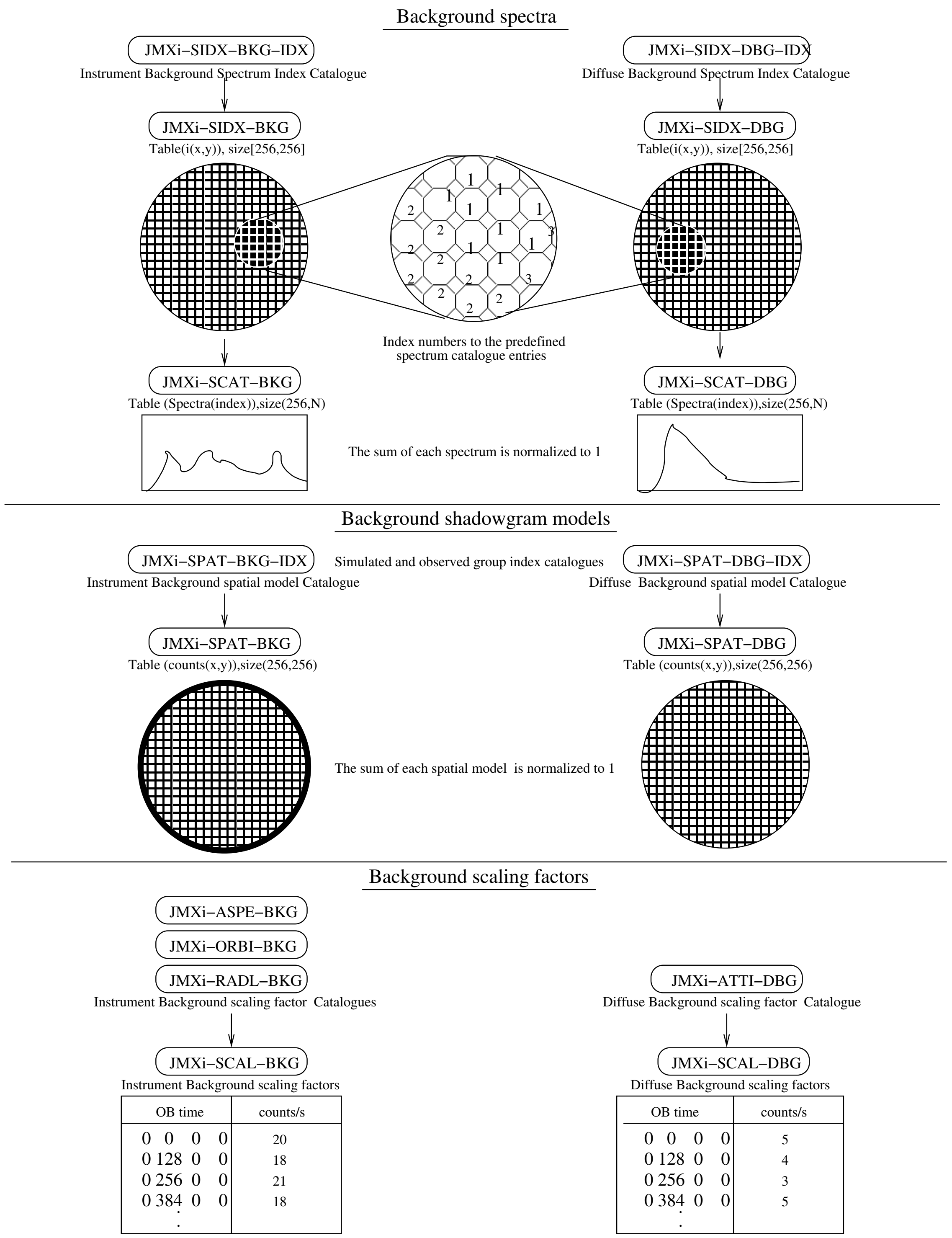

Figure 3.4: Background handling overview

- spatial part of the background model in any pixel or any subsets of pixels on the detector
- Any combination of the above alternatives

The overview of data structures and the background handling definitions are presented in figure 3.6.1. All predefined spectral and spatial models are normalized to 1. The scaling factor is the value which gives the total background counts over the detector per second.

Example 1
If we want to determine the instrumental part of the spatial background in pixel (x, y) on the detector and the time interval is from T_1 to T_2.

1. Read the value *JMXi-SPAT-BKG* in point (x, y)
2. Read the values *JMXi-SCAL-BKG* from T_1 to T_2.

   Total $counts$ = $\sum_{t=T_1}^{T_2}$ *JMXi-SCAL-BKG*(t)
3. Result = $counts\cdot$ *JMXi-SPAT-BKG*(x, y)

where $\sum_{x=1}^{256}\sum_{y=1}^{256}$*JMXi-SPAT-BKG*$(x, y) = 1$.

We assume the spatial model does not change from T_1 to T_2. The diffuse sky background models can be solved by the same method. The full spatial background model is simply the sum of instrumental and diffuse parts.

Example 2
If we want to determine the instrumental part of the background spectrum in pixel (x, y) on the detector and the time interval is from T_1 to T_2.

1. Read the spectrum *JMXi-SIDX-BKG*(i)
2. Read the value *JMXi-SPAT-BKG* in point (x, y).
3. Read the values *JMXi-SCAL-BKG* in points T_1 and T_2.

   Total $counts$ = $\sum_{t=T_1}^{T_2}$ *JMXi-SCAL-BKG*(t)
4. Result = $counts\cdot$ *JMXi-SIDX-BKG*$(i)\cdot$ *JMXi-SPAT-BKG*(x, y)

where *JMXi-SPAT-BKG*(x, y) gives us the fraction of the counts which are assumed to be in that particulary pixel of the detector.

Example 3
When we want to determine the instrumental part of the background spectrum model on the entire detector and the time interval is from T_1 to T_2.

1. Read the values *JMXi-SCAL-BKG* in points T_1 to T_2.

   Total $counts$ = $\sum_{t=T_1}^{T_2}$ *JMXi-SCAL-BKG*(t)

2. The normalized average spectrum can be resolved by equation $Spectrum = \frac{\sum_{x=1}^{256}\sum_{y=1}^{256}}{A}($ *JMXi-SPAT-BKG*$(x,y)\cdot$ *JMXi-SIDX-BKG*$(i(x,y)))$ where, $A = 256 \cdot 256$ is size of detector and i is index number of spectrum.

3. Result= $Spectrum \cdot counts$

3.6.3 Background Handling for binned data

The **j-bin-spectra** and **j-bin-lc** scripts that generate binned spectra and lightcurves covering the whole JEM-X detector, deal with background issues in a different fashion than the image reconstruction processes.

There is no attempt at subtracting the background during the processing, instead for each generated JEM-X spectrum or lightcurve a corresponding background spectrum or lightcurve is produced by binning of the background model information, derived by the methods described in previous sections, using the same binning as the measured spectra and lightcurves.

3.6.4 Background Handling in j-src-extract

The extraction of individual source light curves or spectra will automatically removes the flat background flux. Spatial variations in background flux over the detector however, have a net effect on the extracted source flux. Given the available background model this net contribution is estimated and corrected for. This correction is done for each time/energy bin in which source spectra and light curves are calculated.

3.7 Decomposition of JEM-X ISSW

The delivered software consists of several different parts:

- C source files for each executable and its constituent functions
- Data structure templates agreed between SDAST and ISDC
- makeisdc1.in - an input file for automatic configuration of the ISDC Makefile used to compile and install the software at ISDC
- A header file for each executable
- A parameter file in IRAF format for each executable

- A header file jemx.h common to all JEM-X executables
- Help and documentation files in ASCII format
- Scripts to run unit tests on the compiled software
- Test data to be used with the unit tests
- Sample output from the unit tests
- The Architectural Design Document describing the software for ISDC and incorporating the Detailed Design Document (this document)

All software can be downloaded from :
`http://http://isdc.unige.ch/index.cgi?Software+downloads` The figures on pages 30, 31 and 32 show all of the software components and subcomponents described in this document which are unique to the JEM-X processing. Some levels, such as GTI and CAT_I contain generic ISDC tools that are not described in detail in this document. Please see ISDC documents [5], [26] and [25].

The thick line shows where components must be run in a particular order. However, offline processes j_bkg_interactive, j_performance and j_ical can be run simultaneously or individually in any order, once the fundamental processing has taken place.

All the raw telemetry must pass through the ISDC PreProcessing pipeline which unpacks the raw bit-stream into FITS files, according to the telemetry package structures described in the Instrument Users Manual [24]. After this the raw science, calibration and housekeeping data is processed by either the Science Window Pipeline or the Revolution File Pipeline, depending on whether it is deemed best to divide the data up by revolution (e.g. calibration spectra and electronic calibration data) or by Science Window (e.g. all science data).

From June 2004 onward, the data archive contains the corrected data that has gone through the -COR processing level, as well as all the -RAW data. This means that starting with OSA 4.0, the general user need not do the correction steps at all when running OSA, but start at the CAT_I level. If however, the user wants to repeat the data correction steps using new or customised correction tables, this is also possible in the OSA pipeline.

3.7.1 PRP: Data Preparation

These executables form part ofthe Science Window Pipeline and the Revolution File Pipeline. They continue the pre-processing work begun by ISDC to turn raw telemetry into raw data packaged into fits extensions. By the end of PreProcessing and Data Preparation, all the raw data ready for calibration and correction is contained in data structures with **-PRW, -SRW, -RAW, -PRP, -CRW** suffixes. OBTs cannot be produced in ISDC PreProcessing because the C-OBT/L-OBT correlation is not known at that point.

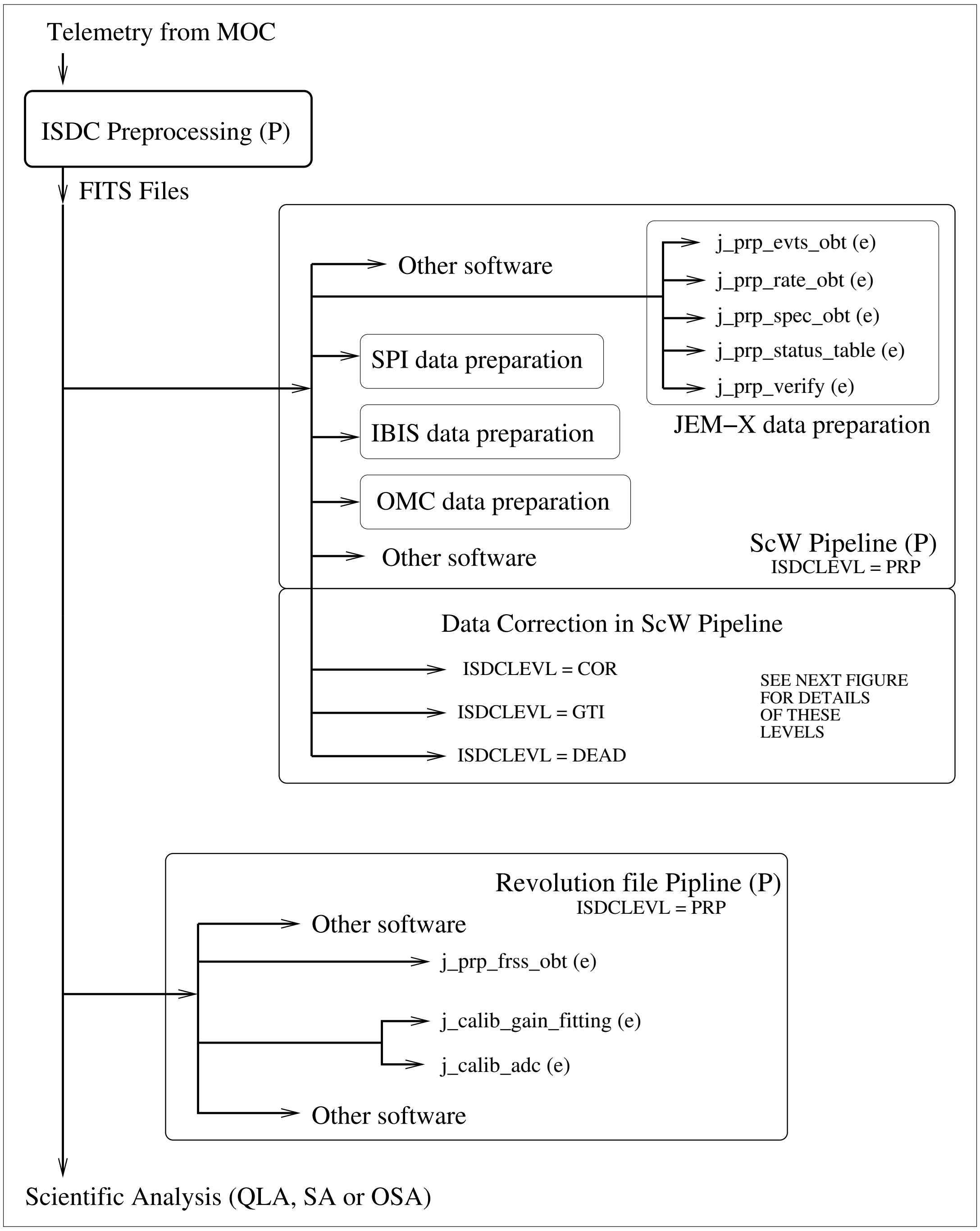

Figure 3.5: JEM-X V1 Data Preparation Overview: 'P' indicates an analysis pipeline; 's' denotes a pipeline script and 'e' is an executable

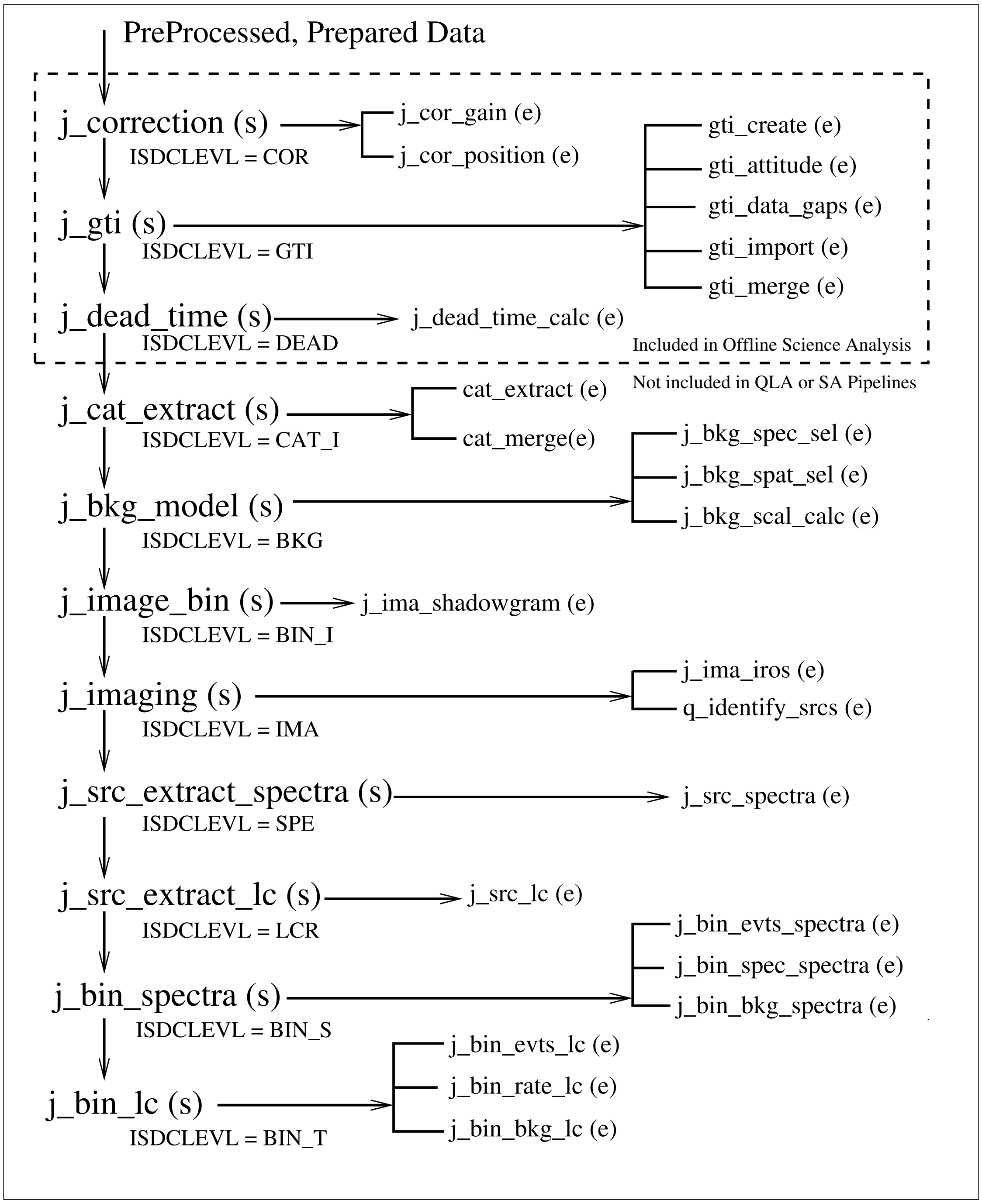

Figure 3.6: JEM-X Science Analysis Pipeline (QLA, SA or OSA): 's' denotes a pipeline script and 'e' is an executable. The dashed box indicates pipeline levels that are part of the science analysis in the Offline Science Analysis, but part of the Science Window Pipeline (data preparation) for Consolidated and NRT processing.

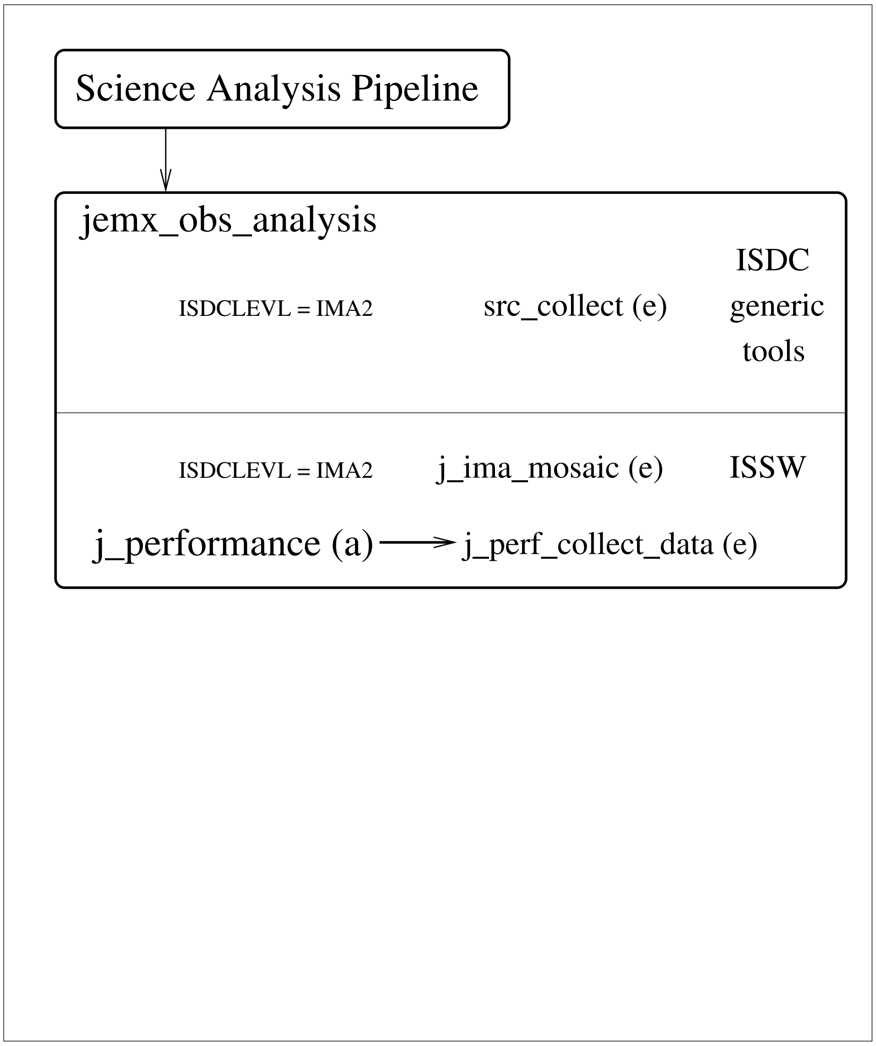

Figure 3.7: JEM-X Offline Analysis Overview: 'a' denotes an interactive application; 'GUI' is a graphical user interface and 'e' is an executable

5

Some data preparation is performed on a science window by science window basis and hence is part of the ScW Pipeline:

j_prp_evts_obt Generates full On-Board Time information for event science data (FULL, REST, SPTI and TIME).

j_prp_rate_obt Generates full OBTs for countrate science data (RATE).

j_prp_spec_obt Generates full OBTs for spectral science data (SPEC).

j_prp_status_table Creates the instrument status table containing data-taking mode and grey filter value as a function of time for each SWG.

j_prp_verify Checks that all science data going into the Science Analysis pipeline is self-consistent and behaves as described in the telemetry and data structure documentation. A list of the normalised event populations for each anode is also made, integrated over the entire science window.

Other executables however, work with files stored according to revolution, and hence are part of the Revolution File Pipeline. Though they are part of Data Preparation, they are not called by j_data_preparation but by the Revolution File Pipeline directly:

j_prp_frss_obt Generates full OBTs for each set of four calibration spectra.

j_calib_gain_fitting Calculates detector gain by fitting calibration spectra to gaussian line shape functions. Energy resolution and source strengths are also produced for performance and health monitoring purposes.

j_calib_adc Checks the linearity of the ADC/amplifier chains on the detector using HK electronic calibration data. Provides alerts if any of the amplifiers are bad.

These components are all part of the ISDC V1 Technical Processing

3.7.2 COR: j_correction

This pipeline application removes detector fingerprints that can be determined quantitatively, for all of the different telemetry modes. Initially, look-up tables of corrections, determined by pre-flight calibration are used. However, if the use of interactive calibration during the flight indicates that the detector behaviour has changed significantly then these tables will be replaced in the instrument model group JMXi-IMOD-GRP (p. 239) by tables produced by j-ical (p. **??**).

At the end of each correction executable every event is graded according to the uncertainty on the determination of its: position, energy, OBT grey filter value and its 'goodness' depending on its position on the detector. Currently a 16-bit flag is used in column STATUS of the JMXi-****-COR data structures to indicate problems with these quantities, or others that can be added later. See "Event Evaluation" on p.56 for further details. This application is part of ISDC V2.

j_cor_gain Corrects energy data (PHA) for changes in the gain of the microstrip detector over time and across the detector area. Corrections are averaged over the four different detector anode sections for all data types. This correction is applied to energy measurements of all events and spectra based on the gain parameters found by j_calib_gain_fitting and the spatial gain model JMXi-SPAG-MOD. Events are flagged for poor gain determination and OBT correlation.

j_cor_position Corrects position data (raw Y and raw Z) for the imaging data modes: FULL and REST. Corrections are performed on the basis of corrected position model tables JMXi-CORX-MOD and JMXi-CORY-MOD in the instrument model group. Each table contains the corrected absolute position of each pixel in millimetres from the centre of the detector. These tables can be updated using output from the interactive tools in j-ical (p. **??**). Events are flagged if they come from: hot spots, 'dead' anodes, areas outside the active detector area and areas contaminated by calibration spectra.

3.7.3 GTI: j_gti

This application creates the tables of Good Time Intervals for the Science Window and is made up of ISDC generic tools in OSM:

gti_create Creates separate lists of GTIs based on different criteria for various housekeeping parameters and their pre-defined limits. GTIs can also be created based on user-defined limits.

gti_attitude Creates a list of GTIs based on the stability of the spacecraft pointing. i.e. its attitude.

gti_data_gaps Creates a list of GTIs based on gaps in the telemetry.

gti_import Imports a list of user GTIs or other GTIs not automatically produced by the tools listed here.

gti_merge Merges the lists of different GTIs produced by gti_create

The application is part of ISDC V2/OSM

3.7.4 DEAD: j_dead_time

This application makes a table of dead times for each polling cycle, i.e. every 8 seconds. Sources with strong variations more rapid than 8 seconds must be corrected for dead time offline with custom-made tools. This component is part of ISDC V2

j_dead_time_calc Creates history table of instrument dead-time based on housekeeping records of the number of software triggers compared to the number of events discarded because of various criteria: full buffers and particle rejection criteria. An effective dead time is also calculated which includes the effect of events discarded due to grey filtering, averaged over the 8 seconds. However, most components further along the pipeline use the instantaneous grey filter values to determine the effective flux, after applying the 8 second dead time.

3.7.5 CAT_I: j_cat_extract

This application makes lists of expected catalogue sources for the instrument's current FOV. It is made up of two generic ISDC tools and is part of ISDC V2:

cat_extract Makes a list of sources within the FOV.

cat_merge Merges different lists of sources that should be observable during the pointing.

3.7.6 BKG: j_bkg_model

This application performs automatic background determination using scaling of background models chosen from a catalogue of spatial and spectral background models. The background thus found for each FOV is subsequently used in the image reconstruction and source data extraction applications. Binning of empty fields into background models is one of the ways that the catalogue is populated. More intricate determination of background is done by the interactive tool j_bkg_interactive as part of the off-line analysis. This application is part of ISDC V2.

j_bkg_spec_sel Generates diffuse and instrument background spectrum catalogue index tables. Tables are same size as detector size (256*256), and each detector pixel has an index value which corresponds to the characteristic background spectrum model. The sum of each spectrum is normalized to 1.

j_bkg_spat_sel Selects automatically the most appropriate instrument and diffuse spatial background models. Spatial distributions are selected from either observed or simulated models. Selection depends on orbit, attitude and radiation level parameters. Simulated and observed shadowgram catalogues are indexed group tables where each model is a separate FITS file. Catalogues can be updated during the mission.

j_bkg_scal_calc Calculates automatically the most appropriate instrument and diffuse background scaling factors. Scaling depends on orbital and attitude data, radiation level and exposure time parameters.

3.7.7 BIN_I: j_image_bin

This application bins the corrected event data into time-resolved shadowgrams with user-defined energy bins given by input parameters. QLA and SA differ in that the former uses coarser energy binning and time resolution. It is part of ISDC V2.

j_ima_shadowgram Selects events in a shadowgram by time and energy range, in preparation for creation of a series of images differing in time and energy.

3.7.8 IMA: j_imaging

This component has totally been restructured for the release of OSA v. 5. It deconvolves the binned shadowgrams to make high quality sky images. This application is part of ISDC V2.

j_ima_iros This executable performs Iterative Removal of Sources (IROS) and produces a found-source list. It returns reconstructed sky images with fine resolution (1.5 arcmin) in celestial coordinates by backprojection. Raw Intensity, Residual and Reconstructed images with sources, as well as Variances sky maps are produced.

q_identify_srcs The tool used by ISDC to identify sources in the found-sources list.

3.7.9 SPE: j_src_extract_spectra

This application extracts time-resolved spectral data for all the found point sources detected in the reconstructed images. Background subtraction is an integrated part of the data extraction process. The application is part of ISDC V2.

j_src_spectra Creates spectral data for each found source in FOV.

3.7.10 LCR: j_src_extract_lc

This application extracts energy-binned lightcurves for each found point source in the FOV. Background subtraction is an integral part of the data extraction. The application is part of ISDC V3.

j_src_lc Creates binned lightcurves for each found source in FOV.

3.7.11 BIN_S: j_bin_spectra

Events that exist in one format are binned into other data formats, so that Full Imaging events, for example, which contain full time and energy information, are binned into spectra. Also empty fields are binned into spectra for use in the background model catalogue. This application is part of ISDC V2.

j_bin_evts_spectra Bins data in FULL, REST and SPTI formats to spectra.

j_bin_spec_spectra Bins data in pure SPEC format to spectra.

j_bin_bkg_spectra Bins background model data to spectra.

3.7.12 BIN_T: j_bin_lc

This application bins event data into lightcurves in different energy bands. The binning format is determined by input parameters to the executables. Empty fields are also binned for use in the background model catalogue. The application is part of ISDC V3.

j_bin_evts_lc Bins JEM-X event data (FULL, REST, SPTI) to lightcurves.

j_bin_rate_lc Bins JEM-X countrate data (RATE) to lightcurves.

j_bin_bkg_lc Bins background model data and empty fields to lightcurves.

3.7.13 IMA_2: jemx_obs_analysis

This is actually an analysis script to run all the observation group tasks like, e.g., combining results of j-imaging (p. 161) at the observation group level. It produces mosaicked images of neighbouring sky images from individual science windows and a compilation list of found sources at the imaging level per science window. If needed in the future, more tasks at the observation group level can be added.

j_ima_mosaic This executable makes weighted mosaics of the sky images produced by j-imaging (p. 161). Intensity, Variance and Significance maps, as well as combined Observation time and effective Exposure time maps are produced.

src_collect Merges found sources lists in the individual science windows.

3.7.14 Offline: j_performance

Performs the long-term monitoring and interactive analysis of the JEM-X detectors. Parameters characterising the detectors are read from the archives (e.g. voltages, pressures, temperatures etc.) and displayed. Alerts occur if the parameters fall outside pre-defined limits. The user can choose to run the performance analysis interactively on displayed parameters. Gain calibration parameters calculated in j-calib-gain-fitting (p. 99) are also displayed and analysized i.e. energy resolution and FRSS source intensities for each anode. This application is part of ISDC V3.

j_perf_collect_data Makes a master table of performance parameters to be viewed over a given time range. The resulting table can then be viewed offline using ftools or other available utilities.

3.8 Dream Tools

While most of the components that will eventually be coded for the JEM-X ISSW are fairly well-defined, there are a number of tools that fall into the low priority category of 'Dream Tools': tools that we would all like to have available eventually, but that will probably not be delivered to ISDC by launch time.

Indeed, the feasibility of some of these tools is questionable given limited time and resources. Some will definitely require in-flight experience with the instruments before either their feasibility or utility can be assessed.

The following is a rundown of tools that could be forseen in the future, in no particular order:

- Tools to merge images/spectra/lightcurves of the two JEM-X units taken at identical times so as to maximise the information content of the units as a single instrument
- A robust spectral-typing tool for rapid survey work with the core programme galactic plane survey data to identify new sources and new types of sources found by JEM-X
- A tool for flagging the operational era of each Science window. Background and other areas of processing can be strongly affected by the onboard settings of certain parameters. Specifically the high voltage supply to the microstrip, the low level descriminator and the particle rejection criteria. Consequently it would be useful if science windows could be flagged by number with the different operational eras when these parameters had particular constant values.
- A tool for finding sources within mosaic'ed sky images. This will depend strongly on each instrument's specific Point Spread Function and so requires a dedicated tool for each one.

Chapter 4

Detailed Design

This chapter describes many different features of the JEM-X ISSW that will need to be understood by those people who maintain the software in the future.

4.1 Common Parameter Names

To ease maintainence of the pipeline scripts all the JEM-X ISSW executables use the same parameter names in their $***$*.par files for the same inputs to the executables. This means that if executable j_cor_position, in ISDCLEVL COR of the automatic pipeline, needs an override DOL for data structure JMXi-REST-COR, it uses the same parameter name for this DOL as executable j_ical in the offline analysis use for this data structure.

All data structures that reside in the SWG should be accessed through inSWG and outSWG—parameters for each data structure within the SWG are for OVERRIDE versions of the particular extension and are included in the parameter file to allow non-standard, non-SWG versions of a particular extension to be used. This means that while all the required parameter names should appear in an executable's parameter file, many of these are without a value, except in unusual circumstances.

Likewise, all the instrument model data structures are accessed using the DOL for the JMXi-IMOD-GRP grouping table with parameter name instMod. There are NO OVERRIDE DOLS FOR THE INSTRUMENT GROUP: the correct instrument group is always provided to the user via time-validity constraints or other considerations, using a generic ISDC tool called by the pipeline script.

Parameter names beginning with capital letters are used by the pipeline scripts. These parameters are usually passed unchanged to the component executables of the script as the parameter whose name appears after the capitals. The part in capitals indicates the processing level of the executables that use the parameters. Similarly, application scripts append the executable name to each parameter.The script parameters are listed under their short names as 'Also known as'.

The following parameters are used by ISDC tools which are not described in this document:

- q_identify_srcs[25]: distFuzz p. 43 , relDist p. 50 , gridNum p. 45 , parrefsearchNum
- q_flag_srcs[25]: errPos p. 44 , logicType p. 47

Below is a list of the parameter names used by the JEM-X ISSW arranged in alphabetical order.

Name	Description
Accuracy	Accuracy of OBT/IJD conversion for GTIs Also known as: GTI-Accuracy
alrtLevel	Maximum level of alerts produced by j-prp-verify (p. 95)
alrtBoth	Alert level produced when ADC/amp offset and slope are bad
alrtOffset	Alert level produced when an ADC/amp offset is bad
alrtOther	Alert level produced when something else wrong with ADC/amp
alrtSlope	Alert level produced when slope of ADC/amp is wrong
alrtZero	Alert level produced when an ADC/amp is dead
angleLimit	Maximum offaxis angle for source acceptance in deg Also known as: IMA-angleLimit
anodePhase	Selection flag for anode phases def:0xFFFFF
anodHis	DOL of override to anode population histogram JMXi-ANOD-PRP
aspeScale	DOL to override JMXi-ASPE-BKG
attiScale	DOL to override JMXi-ATTI-BKG
attTolerance	Allowed attitude variation for GTIs Also known as: GTI-attTolerance
backCorr	Perform background correction? Also known as: BIN-I-backCorr
baseDir	Base directory name string of repository

backRes	Optional DOL for background spectra from event binning
bgrMethod	Choice of background handling method used by j-src-spectra (p. 175): 0=flat; 1=Model Also known as: SPE-bgrMethod LCR-bgrMethod
bkgScale	DOL to override JMXi-SCAL-BKG, Also known as: j-ima-shadowgram-bkgScale , j-bin-bkg-lc-bkgScale , j-bin-bkg-spectra-bkgScale
bkgShdDOL	DOL of background shadowgrams (override if different from ”empty”) Also known as: IMA-bkgShdDOL
bkgSpatial	DOL to override JMXi-SPAT-BKG, Also known as: j-ima-shadowgram-bkgSpatial , j-bin-bkg-lc-bkgSpatial , j-bin-bkg-spectra-bkgSpatial
bkgSpatIdx	DOL to override JMXi-SPAT-BKG-IDX
bkgSpecCat	DOL to override JMXi-SCAT-BKG
bkgSpecIdx	DOL to override instrument background spectrum model index
bkgSpecMap	DOL to override JMXi-SIDX-BKG Also known as: j-ima-shadowgram-bkgSpecMap , j-bin-bkg-lc-dbgSpatial , j-bin-bkg-spectra-bkgSpecMap
binFactor	Performance monitoring binning factor. 0=‘full’
BIN-S-rowSelectEvts	Script parameter string for selecting events
BIN-S-rowSelectSpec	Script parameter string for selecting spectra
BTI-Dol	DOL of Bad Time Interval table Also known as: GTI-BTI-Dol
BTI-Names	BTI filenames to be considered in processing Also known as: GTI-BTI-Names
cdelt	Output pixel size at the center of the mosaic maps [deg.] Also known as: IMA2-cdelt
chanHigh	Array of Upper boundaries of channel bins if nChanBins>0
chanLow	Array of Lower boundaries of channel bins if nChanBins>0

chanMax	Highest PI bin for shadowgram binning
chanMin	Lowest PI bin for shadowgram binning
chatter	Sets degree of verbosity of output (0 to 5)- see below for pre-defined levels of chatiness Also known as: general-chatter
class	Class of objects to select for source catalogue search Also known as: CAT-I-class
clobber	Clobber existing output data structure (y/n) Also known as: general-clobber
clean3size	[pix] Half size of clean3 region
cleanStep	[pix] step for cleaning of stripes
csswCnv	DOL to override JMXi-CSSW-CNV Also known as: j-dead-time-calc-csswCnv
csswHrw	DOL to override JMXi-CSSW-HRW Also known as: j-dead-time-calc-csswHrw
date	IJD of the public data for source catalogue search Also known as: CAT-I-date
dbgGalDbg	DOL of diffuse background database
dbgScale	DOL to override JMXi-SCAL-DBG, Also known as: j-ima-shadowgram-dbgScale , j-bin-bkg-lc-dbgSpatial , j-bin-bkg-spectra-dbgScale
dbgSpatial	DOL to override JMXi-SPAT-DBG Also known as: j-ima-shadowgram-dbgSpatial , j-bin-bkg-lc-dbgSpatial , j-bin-bkg-spectra-dbgSpatial
dbgSpatIdx	DOL to override JMXi-SPAT-DBG-IDX
dbgSpecCat	DOL to override JMXi-SCAT-DBG
dbgSpecIdx	DOL to override diffuse background spectral models index
dbgSpecMap	DOL to override JMXi-SIDX-DBG Also known as: j-ima-shadowgram-dbgSpecMap , j-bin-bkg-lc-dbgSpecMap , j-bin-bkg-spectra-dbgSpecMap

deadTime	DOL to override JMXi-DEAD-SCP Also known as: j-ima-shadowgram-deadTime , j-src-lc-deadTime , j-bin-spec-spectra-deadTime , j-src-spectra-deadTime , j-bin-evts-lc-deadTime , j-bin-rate-lc-deadTime
DECcenter	Output mosaic map center DEC [deg.] Also known as: IMA2-DECcenter
detAccLimit	Acceptance limit in DETE-MOD array Also known as: IMA-detAccLimit
detMapFile	File with detector map (obsolete)
detSigSingle	Detection significance for source acceptance Also known as: IMA-detSigSingle
diagnosticMode	Diagnostic options for source extraction Also known as: j-src-spectra-diagnosticMode , j-src-lc-diagnosticMode
diagnosticParam	Toggle for Diagnostic Parameter Also known as: j-src-spectra-diagnosticParam , j-src-lc-diagnosticParam
diameter	Output mosaic maps diameter [deg.] Also known as: IMA2-diameter
distFuzz	Allowed fuzziness in source-matching distances Also known as: IMA-distFuzz
dolBPL	DOL of JMXi BPL (backprojection list) group (override) Also known as: IMA-dolBPL
doClean3	Use clean3 ?
doPhaseScan	Flag for activation of phase scan
ecalPrp	DOL of JMXi-ECAL-CPR
ecalRaw	DOL of JMXi-ECAL-CRW
edgeEnhanceFactor	Edge enhancement factor Also known as: IMA-edgeEnhanceFactor
eeSelect0	Flag for ee=0 band selection from 16 bands
eeSelect1	Flag for ee=1 band selection from 16 bands

eeSelect2	Flag for ee=2 band selection from 16 bands
eminSelect	Input images selection min. energy [keV] Also known as: IMA2-eminSelect
emaxSelect	Input images selection max. energy [keV] Also known as: IMA2-emaxSelect
eeSelect3	Flag for ee=3 band selection from 16 bands
endLevel	ISDC level at which to end processing—scripts only
errPos	Acceptable variation in identifying a flagged source q-flag-srcs-errPos
evtType	Type of JEM-X event (0=FULL; 1=REST ; 2=SPTI ; 3=TIME ; 4=SPEC ; 5=RATE ; -1=default:all sensible combinations) BIN-I-evtType , SPE-evtType LCR-evtType BIN-S-evtType BIN-T-evtType
findRes	Override DOL of JMXi-FIND-RES
fluxDef	Column used for flux selection during source catalogue search Also known as: CAT-I-fluxDef
fluxLimit	Lower flux limit for alerts on unfound sources Also known as: IMA-fluxLimit
fluxMax	Upper limit for flux selection for source catalogue search Also known as: CAT-I-fluxMax
fluxMin	Lower limit for flux selection for source catalogue search Also known as: CAT-I-fluxMin
fluxScaling	Options for flux scaling to area Also known as: SPE-fluxScaling LCR-fluxScaling
fullCor	DOL to override JMXi-FULL-COR Also known as: j-cor-both-fullCor , j-src-spectra-fullCor j-ima-shadowgram-fullCor , j-bin-evts-spectra-fullCor , j-src-lc-fullCor , j-bin-evts-lc-fullCor
fullPrp	DOL to override JMXi-FULL-PRP Also known as: j-cor-both-fullPrp , j-src-lc-fullPrp , j-ima-shadowgram-fullPrp , j-bin-evts-spectra-fullPrp j-src-spectra-fullPrp , j-cor-gain-fullPrp , j-bin-evts-lc-fullPrp
fullPrw	DOL to override JMXi-FULL-PRW

fullRaw	DOL to override JMXi-FULL-RAW Also known as: j-cor-both-fullRaw , j-bin-evts-spectra-fullRaw
fullSrw	DOL to override JMXi-FULL-SRW
fsTwistJ1	[rad] JEMX1 frontside twist
gainHist	DOL to JMXi-GAIN-CAL-IDX Also known as: COR-gainHist
gainModel	Model options for time-smoothing of detector gain results Also known as: COR-gainModel
gainOut	DOL of interactive gain history output for ICAL tools
glitchOut	DOL of glitch histogram image, for offline analysis
goodTime	DOL to override JMXi-MERG-GTI or JMXi-GNRL-GTI Also known as: j-src-spectra-goodTime , j-bin-spec-spectra-goodTime , j-src-lc-goodTime , j-bin-evts-lc-goodTime , j-bin-rate-lc-goodTime
gridNum	Number of steps on one side of source finding search grid Also known as: IMA-gridNum
gti-data-gaps-RemoveWrongGTIs	Remove overlapping GTIs and GTIs with start time > end time
gti-data-gaps-XXX	Instrument modes for which to make GTIs based on data gaps with the following names: gti-data-gaps-COMPTON-SGLE-Mode gti-data-gaps-COMPTON-MULE-Mode gti-data-gaps-ISGRI-Mode gti-data-gaps-JMX1-Mode gti-data-gaps-JMX2-Mode gti-data-gaps-PICSIT-MULE-Mode gti-data-gaps-PICSIT-SGLE-Mode gti-data-gaps-SPI-Mode
gtiJemxNames	Names of JEM-X GTIs to merge Also known as: GTI-gtiJemxNames
gtiNames	Names of GTI tables to be used in j-src-spectra (p. 175) Also knowns as: BIN-I-gtiNames

gtiScNames	Names of spacecraft GTIs to merge Also known as: GTI-gtiScNames
gtiUser	DOL of user GTI table Also known as: GTI-gtiUser
hkType	Type of housekeeping to use for perforamance table. 0=AVGR, 1=RAW and 2=CONV
hotPixelLimit	Hot pixel limit factor above average Also known as: IMA-hotPixelLimit
IC-Group	DOL of Instrument Model master group table, used by analysis pipeline scripts only
IC-Alias	Selection alias string for Instrument Model group: analysis scripts only
ignoreAnode	Up to 4 numbers 0 to 3 of anode segments to ignore during image deconvolution Also known as: IMA-ignoreAnode
ignoreScwErrors	Script parameter: ignore processing errors?
inCAL	DOL of input calibration spectra index table JMXi-FRSS-CRW-IDX
inDOL	DOL of single data file not in the SWG, e.g. JMXi-FRSS-CRW
indexSWG	DOL of the science window group index GNRL-SCWG-GRP-IDX
inObsGrp	Input observation group DOL
instMod	DOL to JMXi-IMOD-GRP
instStat	DOL to override JMXi-INST-STA Also known as: j-ima-src-find-instStat , j-bin-spec-spectra-instStat , j-src-spectra-instStat , j-src-lc-instStat , j-bin-evts-lc-instStat , j-bin-rate-lc-instStat
inSWG	Input Science Window Group DOL Also known as: LCR-inSWG
interactionDepth	[mm] Mean depth of interaction in the detector Also known as: IMA-interactionDepth
jemxNum	JEM-X Instrument number: 1 or 2; 0='both' i.e. for performance monitoring and comparison. Also known as: general-jemxNum

lcRes	DOL to override output lightcurve JMXi-FULL-DLC or JMXi-SPTI-DLC for j-bin-evts-lc (p. 210)
lcType	Option for binning normal or folded lightcurves
limitTable	DOL of table with alert/GTI limits. If empty table is taken from Instrument Characteristics database Also known as: GTI-limitTable
listRes	Override DOL of JMXi-SRCL-RES
logicType	Logic used for source flagging, 1=AND, 2=OR. Also known as: q-flag-srcs-logicType
loopLimitPeak	Fractional peak power to stop IROS loop Also known as: IMA-loopLimitPeak
mapSelect	Mosaic input images selection - keyword IMADESCR (RAW_RECT or RES+SRC) Also known as: IMA2-mapSelect mosaicselection
makeNewBPL	Do you want to make a new BPL group ? Also known as: IMA-makeNewBPL
maxNumSources	Max number of sources in IROS loop Also known as: IMA-maxNumSources
maxNumTrials	Maximum number of trials in source search Also known as: IMA-maxNumTrials
MergedName	Name of merged GTI table to be used Also known as: GTI-MergedName
minDetLev	Minimum detector level for found source acceptance Also known as: IMA-minDetLev
minDist	Minimum source separation in arcmins for source search Also known as: IMA-minDist
mosaicFileDOL	Script parameter for output mosaic Also known as: IMA2-mosaicFileDOL
mosaicselection	Script parameter string for image selection Also known as: IMA2-mosaicselection

nChanBins	Number of channel bins in output data (e.g. number energy-resolved light curves or images). If set to 0 there will be a single bin with all data. If set to <0 the data will be rebinned by grouping abs(chanBins) channels together. To obtain data with binning at full resolution one would set chanBins= -1 Also known as: BIN-I-nChanBins
newBackProjFile	Name of new BPL group file if requested Also known as: IMA-newBackProjFile
nPhaseBins	Number of bins used in cyclic folding of periodic sources with bins of unequal length an option: if nPhaseBins=0 then no pulse folding is done. If nPhaseBins>0, then this is the number of phase bins to use. if nPhaseBins<0 use $\vert$nPhaseBins$\vert$ equal sized bins for cyclic folding. Size of unequal sized bins is given by vector parameter phaseBins
numPixelSubDiv	Number of pixel subdivisions in j-imaging (p. 161) transforming from scew to detector coordinates Also known as: IMA-numPixelSubDiv
onlyDoShd	Skip sky image making
ogDOL	DOL of observation group—input to observation pipeline
orbiScale	DOL to override JMXi-ORBI-BKG
osimData	Scripts flag for processing on simulated data general-osimData
outBkgSpec	DOL to override JXMi-SRCL-BSP
outCAL	DOL of output gain calibration index table JMXi-GAIN-CAL-IDX
outDOL	DOL of single data file not in the SWG, e.g. JMXi-GAIN-CAL-IDX
outfile	Output mosaic file name prefix Also known as: IMA2-outfile
outputExists	Assume already existing output Also known as: COR-outputExists DEAD-outputExists
outSrcArf	DOL of override JMXi-SLSP-ARF or JMXi-SLLC-ARF
outSrcLC	DOL of override JMXi-SRC.-LCR

outSrcSpec	DOL to override JMXi-SRCL-SPE
outSWG	Output Science Window Group DOL
overrideCollTilt	Collimator tilt Parameter? (if <0 use IMOD) Also known as: SPE-overrideCollTilt LCR-overrideCollTilt
pardesparNamesName of ascii file holding HK parameter names to be included in performance monitoring	
perfTable	Output DOL of performance table
phaseBins	Vector parameter giving size of nPhaseBins bins required for cyclic folding of periodic sources
pix3fact	Amplification factor for pixelfold 3
pixelFold	(obsolete)
pixelFoldFile	File with pixelfold information
precisionLevel	Speed vs. Precision level for j-src-spectra (p. 175) Also known as: SPE-precisionLevel LCR-precisionLevel
qlaCat	DOL of QLA sources catalogue. QLA-qlaCat
qlaMode	Flag—processing for QLA (or SA)? QLA-qlaMode
RAcenter	Output mosaic map center RA [deg.] Also known as: IMA2-RAcenter
radiusLimit	Only events out to this many mm from centre of detector used for imaging
radiusLimit0	[mm] Detector radius limit $E < 6$ keV Also known as: IMA-radiusLimit0
radiusLimit1	[mm] Detector radius limit $6 < E < 12$ keV Also known as: IMA-radiusLimit1

radiusLimit2	[mm] Detector radius limit 12 < E < 20 keV Also known as: IMA-radiusLimit2
radiusLimit3	[mm] Detector radius limit 20 keV < E Also known as: IMA-radiusLimit3
radiusMax	Upper limit of range for catalogue position search Also known as: CAT-I-radiusMax
radiusMin	Lower limit of range for catalogue position search Also known as: CAT-I-radiusMin
radiusSelect	Mosaic input images selection radius [deg.] Also known as: IMA2-radiusSelect
radlScale	DOL to override JMXi-RADL-BKG
ratePrp	DOL to override JMXi-RATE-PRP Also known as: j-bin-rate-lc-ratePrp
ratePrw	DOL to override JMXi-RATE-PRW
rateRaw	DOL to override JMXi-RATE-RAW Also known as: j-bin-rate-lc-rateRaw
rateSrw	DOL to override JMXi-RATE-SRW
rearPhase	Selection flag for rear-side phases def:0xFFFFF
refCat	DOL of ISDC Reference Catalogue of sources for scripts only Also known as CAT-I-refCat
relDist	Allowed uncertainty-scaled distances between matched sources. Also known as: IMA-relDist
response	Script parameter with file of standard ARF and RMF extensions
restCor	DOL to override JMXi-REST-COR Also known as: j-cor-both-restCor , j-ima-shadowgram-restCor , j-bin-evts-spectra-restCor
restPrp	DOL to override JMXi-REST-PRP Also known as: j-cor-both-restPrp , j-cor-gain-restPrp , j-ima-shadowgram-restPrp , j-bin-evts-spectra-restPrp
restPrw	DOL to override JMXi-REST-PRW
restRaw	DOL to override JMXi-REST-RAW Also known as: j-cor-both-restRaw , j-bin-evts-spectra-restRaw

restSrw	DOL to override JMXi-REST-SRW
revolAvg	DOL of ICAL table with revolution averages of gain history
revolStart	First revolution to process with ICAL tools
revolStop	Last revolution to process with ICAL tools
rowSelect	CFITSIO selection string for events Also known as: BIN-I-rowSelect SPE-rowSelect LCR-rowSelect BIN-T-rowSelect
rowSelectEvts	CFITSIO selection string on events (optional)
rowSelectSpec	CFITSIO selection string on spectrum mode data (optional)
rstiRes	Override DOL of JMXi-RSTI-IMA
rsTwistJ1	[rad] JEMX1 readside twist
fsTwistJ2	[rad] JEMX2 frontside twist
rsTwistJ2	[rad] JEMX2 readside twist
searchRad	Range of search around nominal found sources Also known as: IMA-searchRad
shdOutCtrl	Control output of e.g. calculated source shadowgrams
shdType	Type of shadowgram to bin: skew or standard? Also known as: BIN-I-shdType
shdRes	DOL of override JMXi-EVTS-SHD Also known as: BIN-I-shdRes
simulated	Using simulated background models? Also known as: BKG-simulated
skipLevels	Script parameter for Jemx-science-analysis; indicates ISDC levels to skip during processing. e.g. BKG
skipHotSpot	Ignore data from known hotspots? Also known as: SPE-skipHotSpot LCR-skipHotSpot

skipNearDeadAnode	Ignore data near dead anodes? Also known as: SPE-skipNearDeadAnode LCR-skipNearDeadAnode
skyImageDim	Sky image dimension 1→ 255, 2→ 511(prefered) Also known as: IMA-skyImageDim
skyImagesOut	Control output of sky images Also known as: IMA-skyImagesOut
skyRadiusFactor	Sky radius limit factor Also known as: IMA-skyRadiusFactor
skyRes	Override DOL of JMXi-SKY.-IMA
specCor	DOL to override JMXi-SPEC-COR Also known as: j-cor-gain-specCor , j-bin-spec-spectra-specCor
specPrp	DOL to override JMXi-SPEC-PRP Also known as: j-cor-gain-specPrp , j-bin-spec-spectra-specPrp
specRaw	DOL to override JMXi-SPEC-RAW Also known as: j-cor-gain-specRaw , j-bin-spec-spectra-specRaw
specRes	Optional output spectra extension DOL for binning events
sptiCor	DOL to override JMXi-SPTI-COR Also known as: j-cor-gain-sptiCor , j-bin-evts-lc-sptiCor , j-bin-evts-spectra-sptiCor
sptiPrp	DOL to override JMXi-SPTI-PRP Also known as: j-cor-gain-sptiPrp , j-bin-evts-lc-sptiPrp , j-bin-evts-spectra-sptiPrp
sptiPrw	DOL to override JMXi-SPTI-PRW
sptiRaw	DOL to override JMXi-SPTI-RAW Also known as: j-cor-gain-sptiRaw , j-bin-evts-spectra-sptiRaw
sptiSrw	DOL to override JMXi-SPTI-SRW
srcattach	Script parameter: attach result list to observation? Also known as: IMA2-srcattach
srcFileDOL	Script parameter for combined mosaic output source list Also known as: IMA2-srcFileDOL
srcselect	Script parameter for selection string for input sources Also known as: IMA2-srcselect

startLevel	ISDC analysis level on which to begin processing—scripts only
SPE-storePif	Store PIF data structure (1=Events 2=Events detector)?
subDir	Sub directory below the position of the SWG for results
swgDOL	j_scw_analysis: DOL of science window group to analyse
tAccuracy	Time Correlation accuracy in j_src_lc Also known as: LCR-tAccuracy SPE-tAccuracy
threshold	Limit for size of corrections due to vignetting in imaging Also known as: IMA-threshold
timeFormat	Options for possible time formats for GTIs Also known as: GTI-timeFormat
timePrp	DOL to override JMXi-TIME-PRP Also know as: j-bin-evts-lc-timePrp
timePrw	DOL to override JMXi-TIME-PRW
timeSrw	DOL to override JMXi-TIME-SRW
timeStart	Time in IJD (ISDC Julian Date) of *beginning* of first time bin: used as starting point for time binning. Negative values indicate beginning of ScW as default start time. N.B.: All bin times are given by the *centre* time of the bin, except for timeStop and timeStart
timeStep	Integration time in seconds for a single lightcurve bin or a spectrum in time-resolved spectral binning Also known as: SPE-timeStep LCR-timeStep
timeStop	Time in IJD of end of last time bin. If this parameter is negative, last time of ScW is used as default. N.B.: All bin times are given by the *centre* time of the bin, except for timeStop and timeStart
usrCat	DOL of User (private) Catalogue of sources—scripts only Also known as: CAT-I-usrCat
useCatPos	Flag for catalog position to override found position Also known as: IMA-useCatPos
useDeadAnodes	Include dead Anodes ? Also known as: IMA-useDeadAnodes
useHexpos	Flag for employing function hexpos

useRaDec	Use sky coordinates instead of instrument coords.? Also known as: SPE-useRaDec LCR-useRaDec
useTrace	Flag for tracing function calls
viewTime	Create total exposure Time mosaic map? Also known as: IMA2-viewTime
viewIntens	Create raw Intensity mosaic map? Also known as: IMA2-viewIntens
viewVar	Create Variance mosaic map? Also known as: IMA2-viewVar
viewSig	Create Significance mosaic map? Also known as: IMA2-viewSig
vignCorr	Perform vignetting corrections? Also known as: SPE-vignCorr IMA-vignCorr LCR-vignCorr
yes2all	Disregard event selection energy boundaries

Suggested levels of chattiness are given below, and these have been defined as the indicated constants in jemx.h :

J_CHATTY_VERY_QUIET	0	Only system messages and fatal errors
J_CHATTY_QUIET	1	Errors and severe warnings only
J_CHATTY_NORMAL	2	+ important Log messages, extended warnings
J_CHATTY_VERBOSE	3	+ program flow comments
J_CHATTY_VERY_VERBOSE	4	+ miscellaneous program control values
J_CHATTY_MAXIMAL	5	+Everything else you might want for debugging

4.2 Energy Binning: ADC to PI

4.2.1 PHA Binning

The energy values of the events provided in the telemetry are given as a bin number from 0 to 255. These are non-linear groupings of the original 4096 bins of the onboard ADC. The data structures JMXi-***B-MOD indicate the ADC bins covered by a given telemetry PHA bin. These data structures are part of the instrument model group JMXi-IMOD-GRP (p. 239). Each binning table gives a list of the highest ADC channel within each bin.

While we can be certain that under normal conditions the ADC channels are linear to a much higher degree than that used in the telemetry, the PHA bins are designed to be logarithmic so that the energy resolution of the bins parallels that of the detector. The actual grouping of the ADC channels into PHA telemetry bins is determined by the lookup table used by the DPE to pack the telemetry. This table is uploadable, so the values given as examples below cannot be considered fixed, though they should change only once or twice during the mission.

The default values in the DPE-PHA binning table can be found in [23] Here are the first 50 values:

0	1	2	3	4	5	6	7	8	9
10	11	12	13	14	15	17	20	23	25
28	30	33	36	39	42	46	49	53	56
60	64	68	72	77	81	86	90	95	100
105	110	116	121	127	132	138	144	150	156
163	....	...							

The above values are merely default binning values. In the DPE they are integers but in the Instrument model group's data structures the upper channel values are all given as floats. This is to allow us to change the default values we use to effective values if it should be found that non-linearity in the ADC is so great as to cause variations in the PHA binning of the Slow Anode Pulseheight. If, for example, damage to the ADC causes channels 62–69 to register pulseheights that are actually somewhat larger than should be registered by these channels (say, by an amount equal to about half of a normal ADC channel) then the effective upper channel boundaries of PHA bins 21 and 22 are 64.5 and 68.5, not 64 and 68. While it is expected that the boundary values in the instrument model will be identical to those in the DPE binning table, we must at least leave room for the possiblity of using effective binning boundaries in case ADC non-linearity should arise in flight. By using effective binning boundaries we can correct all events and spectra for ADC non-linearities.

Each of the different data-taking modes has its own binning table data structure fitting its telemetry energy resolution: JMXi-CALB-MOD (p. 233) and JMXi-FULB-MOD (p. 237) each have 256 bins for FRSS, FULL and SPTI data; JMXi-RESB-MOD (p. 241) has 8 bins for REST data and JMXi-SPCB-MOD (p. 243) has 64 bins for SPEC data. Tables with fewer than 256 bins are just subsets of the full 256-bin table.

4.2.2 PI Binning

During automatic calibration the original PHA values of the events and spectra are converted into KeV values based on the fitting of the FRSS calibration spectra. The correlation (gain) between the PHA and PI channels varies both in time due to temperature, pressure and voltage fluctuations, and across the surface of the detector. We can expect the biggest spatial gain variations to occur between the four different anode sections of the detector, which is why we have four calibration sources, one for each anode. However, smaller spatial gains can be expected within each anode area and these are mapped by preflight and inflight/interactive calibration, which produce the JMXi-SPAG-MOD (p. 243) data structure of spatial gain model.

The choice of PI bin limits in KeV is largely a matter of taste and is not fixed by the instrument architecture as with the PHA channels. A sensible choice would be 256/64/8 logarithmic channels chosen to cover the entire energy range of the detector (nominally 3–75 KeV). The OGIP standard calls the PI binning table 'EBOUNDS' or EBDS, but since we have several tables, one for each of the three energy resolutions (256, 64 and 8 bins) our data structures are JMXi-FBDS-MOD (p. 236), JMXi-RBDS-MOD (p. 241) and JMXi-SBDS-MOD (p. 242) all of which are in the instrument model group. Again, we expect these tables to be changed very rarely, but the possiblity is always there to make improvements if, for example, there is a sudden change in the energy range of the instruments.

In re-binning our events and spectra from PHA channels to PI channels, we can expect to introduce some inaccuracy into the energy measurements. However, if the PI channels are carefully chosen to reflect the inherent energy resolution of the detector this added error is small compared to the detector energy resolution.

4.3 Event Evaluation

At the end of the various data correction procedures in j_correction, each corrected event and spectrum is examined against a list of 'goodness' criteria. Should the data be found wanting than a flag is raised in the STATUS column of the JMXi-$* * **$-COR being written at that time. The mask values in the STATUS byte have the following meanings:

4 Events fall on an anode segment that is bad

8 Event has uncertain time correlation

16 Event comes from an anode position regarded as bad

32 Events coming from position next to a bad anode

64 Event with bad energy determination due to poor detector gain history determination

128 Event with bad energy determination due to event position

256 Event with bad X position determination

512 Event with bad Y position determination

1024 Event which has probably leaked from a calibration source

2048 Events occurring before SETUP of instrument

4096 Events falling outside active area of the detector

8192 Events coming from a hotspot

A STATUS byte with value zero indicates that the event/spectrum is good for all data analysis purposes. Events that fail in one or more ways can still be used for some purposes: binned spectra don't require good position determinations; uncertainties in grey filter affect only flux determinations etc. Hence it is up to the individual coder to decide what the STATUS byte should be used for.

4.4 Error codes

The assignment of error code ranges to each component in the ISSW is arranged so as to make it easier to track down the exact point in the pipeline where problems are occurring. The range allocated by ISDC for JEM-X ISSW is -300,000 to -399,999. Within this range we have allocated starting error codes for each component as follows:

-300,000 to -309,999: Data Preparation in Revolution File Pipeline			
	PRP	j-prp-frss-obt	-301,100 to -301,199
		j-calib-gain-fitting	-302,100 to -302,299
		j-calib-adc	-303,100 to -303,199
-310,000 to -319,999: Data preparation in Science Window Pipeline			
	PRP	j-prp-evts-obt	-311,100 to -311,199
		j-prp-rate-obt	-311,200 to -311,299
		j-prp-spec-obt	-311,300 to -311,399
		j-prp-status-table	-311,400 to -311,499
		j-prp-verify	-311,500 to -311,599
-320,000 to -329,999: V2 down to and including j-bkg-model			
-321,000	COR	j-cor-gain	-321,100 to -321,199
		j-cor-position	-321,300 to -321,399
		j-correction	-321,900 to -321,999
-322,000	GTI	j-gti	-322,900 to -322,999
-323,000	DEAD	j-dead-time-calc	-323,100 to -323,199
		j-dead-time	-323,900 to -323,999

-324,000	CAT	j-cat-extract	-324,900 to -324,999
-325,000	BKG	j-bkg-spec-sel	-325,100 to -325,199
		j-bkg-spat-sel	-325,200 to -325,299
		j-bkg-scal-calc	-325,300 to -325,399
		j-bkg-model	-325,900 to -325,999
-330,000 to -339,999: V2 after j-bkg-model			
-331,000	BIN_I	j-ima-shadowgram	-331,100 to -331,199
		j-image-bin	-331,900 to -331,999
-332,000	IMA	j-ima-basic-recon	-332,100 to -332,199
		j-ima-src-find	-332,200 to -332,299
		j-ima-cor-intensity	-332,300 to -332,399
		j-ima-refined-recon	-332,400 to -332,499
		j-imaging	-332,900 to -332,999
-333,000	SPE	j-src-spectra	-333,100 to -333,199
		j-src-extract-spectra	-333,900 to -333,999
-334,000	LCR	j-src-lc	-334,100 to -334,199
		j-src-extract-lc	-334,900 to -334,999
-335,000	BIN_S	j-bin-evts-spectra	-335,100 to -335,199
		j-bin-spec-spectra	-335,200 to -335,299
		j-bin-bkg-spectra	-335,300 to -335,399
		j-bin-spectra	-335,900 to -335,999
-336,000	BIN_T	j-bin-evts-lc	-336,100 to -336,199
		j-bin-rate-lc	-336,200 to -336,299
		j-bin-bkg-lc	-336,300 to -336,399
		j-bin-lc	-336,900 to -336,999
-340,000 to -370,000: V3, interactive tools			
-341,000	Interactive background tool	j-bkg-int-spec-sel	-341,100 to -341,199
		j-bkg-int-spat-sel	-341,200 to -341,299
		j-bkg-int-scal-calc	-341,300 to -341,399
		j-bkg-interactive	-341,900 to -341,999
-351,000	Science Analysis	-351,900 to -351,999	
-361,000	Interactive calibration	j-ical-collimator	-361,100 to -361,199
		j-ical-gain	-361,200 to -361,299
		j-ical-ecal	-361,300 to -361,399
		j-ical-frss	-361,400 to -361,499
		j-ical-anodes	-361,500 to -361,599
		j-ical	-361,900 to -361,999
-371,000	JEM-X Performance	j-perf-collect-data	-371,100 to -371,199
		j-performance	-371,900 to -371,999
-380,000	Upper level scripts	jemx-science-analysis	-381,000 to -381,999
		jemx-scw-analysis	-382,000 to -382,999

	jemx-obs-analysis	-383,000 to -383,999
-390,000	Shared functions	

-395,000 and above: reserved for future components

4.5 Spectral Data Handling

The aim of this section is to describe the required executables for ensuring that the derived JEMX spectra for individual sources can be analyzed using 'XSPEC'.

4.5.1 Output data formats

The spectra in the standard analysis are constructed in the analysis script 'j_src_extract_spect' that calls the executable 'j_src_spectra'. The resulting data structures are:

- JMXi-SRCL-SPE (p. 245) contains the count-rate spectra of the listed sources.
- JMXi-SRCL-BSP (p. 245) contains the corresponding background spectra.
- JMXi-SRCL-ARF (p. 245) contains the ARF* (Auxiliary Response Function) of the listed sources.

The light-curves in the standard analysis are constructed in the analysis script 'j_src_extract_lc' that calls the executable 'j_src_lc'. The resulting data structures are:

- JMXi-SRC.-LCR (p. 245) contains the count-rate light-curves of the listed sources and their corresponding background light-curves.
- JMXi-SRCL-ARF (p. 245) contains the ARF (Auxiliary Response Function) of the listed sources. This describes the effective area of the instrument for the direction to the source. The default version contains 500 elements in the array.

4.5.2 Detector Response Matrix

The IC (Instrument Characteristic) file JMXi-RMF.-GRP.fits is a grouping table with three children: JMXi-RMF.-RSP (OGIP type RESPONSE/RSP_MATRIX), JMXi-FBDS-MOD (OGIP type RESPONSE/EBOUNDS), and an on-axis ARF JMXi-AXIS-ARF.

JMXi-RMF.-RSP is the standard detector response normalized to the sum of 1.00 over the detector channels and includes columns defining the input energies. JMXi-FBDS-MOD holds the electronic channels of the detector for its best energy resolution (i.e. in Full Imaging Mode). This data structure is a copy of the IC JMXi-IMOD-GRP.fits data structure with the same name. It is simply repeated for completeness.

4.5.3 Usage with XSPEC

The X-ray spectrum analysis program XSPEC, version 11 and higher, can handle the spectra in the output formats of j_src_spectra with no further format change.

4.5.4 Analysis of light-curves

The light-curve analysis is not as standardized as the spectral analysis. The output format of j_src_lc should be useful with little changes for most analysis packages.

4.6 Flux Units

For both source lightcurves and source spectra (in JMXi-SRCL-LC and JMXi-SRC.-SPE) fluxes are in counts/second/100cm2 (per energy band or channel). This normalisation by the nominal detector area corresponds approximately to the on-axis ARF obtained by multiplying the geometrical detector area (490 cm2) by the coded mask open fraction (0.24) multiplied by the collimator open fraction (0.87). This ideal area corresponds to the sum of the physical areas actually used; if parts of the detector are not used (e.g. dead anodes and/or partial illumination due to off-axis angle) this is corrected by the parameters fluxScaling and vignCorr (which respective default values are 1 and 0, corresponding to the open detector area exposed to a source on axis). For spectra the area is stored in JMXi-SRCL-ARF which is used by XSPEC. For images the flux, also in counts/second/100cm2, is given for each found source in JMXi-SRCL-RES, and in JMXi-OBS.-RES too in case of mosaicking.

4.7 Shadowgram formats

The concept of regularized shadowgram has been introduced since OSA 4.1. Previously the shadowgrams have all been in skew format for simplified handling of the hexagonal mask hole pattern.

The instrument returns raw coordinates (integers 0-255 for each direction x and y). They are roughly in millimeters but offset with respect to the center of the detector which is defined by the central cross of the collimator walls.

The corrected position values are given by IC arrays JMXi-CORX-MOD (p. 234) and JMXi-CORY-MOD (p. 234) (in the JMXi-IMOD-GRP). In other words, when looking up the value of CORX(RAWX,RAWY) you get the average physical position of all the photons returning (RAWX,RAWY).

In the on-ground calibration at the University of Ferrara, Italy, the lower left corners were defined for these pseudo-pixels, and they can be found as JMXi-PLLX-MOD (p. 240) and JMXi-PLLY-MOD (p. 240).

This irregular net of pixels is superimposed on a regular grid and another instrument model table, JMXi-PRED-MOD (p. 240) defines for each pair (RAWX,RAWY) how the measured pixel is distributed in the regular grid, hence the term 'regularized shadowgram'.

The j-ima-shadowgram which makes the shadowgrams has a parameter that defines the output shadowgram type: shdType or BIN_I_shdType in the jemx-science-analysis script. The value is 3 for these regularized shadowgrams but the default choice is 2 for shadowgrams in raw coordinates that are required as input for j-ima-iros (p. 164).

Chapter 5

Component Description

This chapter contains a description of each software component used to analyse the science data from the two JEM-X detectors. Each component is described using the component definitions given below. Data structures used as input and output for the components are catalogued in the Data Dictionary, p,228.

5.1 Component Definitions

Component Piece of software. The JEM-X ISSW is divided into components and sub-components. Components are described in the following way:

Type The class of the component:

- **Application** The highest level ISSW component: a script called by one of the ISDC analysis pipelines: Near Real Time, Consolidated or Offline.
- **Executable** A compiled piece of software which may be called from the command line or invoked by an application script. Executables get *all* their user-defined input from an IRAF-style parameter file. Parameters given on the command line may override parameters from the file (ftools standard behaviour).
- **Script** Intermediate component between a pipeline application and the executables, used to call related executables in a particular order.

Responsible The institution or person responsible for the development, coding and delivery of the component.

Purpose Indicates the relevant part in the Software Requirement Document [7] fulfilled by the component.

Function Short description of the component's functionality.

Inputs List of data structure inputs to the component. Each data structure is described in the Data Dictionary p. 228. Under each application is a list of all the data structures that are involved in the constituent executables.

Outputs List of the data structure outputs from the component. Same remarks apply as for the input list.

Parameters Short description of the control parameters for an executable, or parameters the operator may set during an interactive session.

Resources An estimate of code and data memory, and time taken to process a specified number of events. It should also be indicated how runtime varies with increasing number of events. Special resources the component requires such as CPU speed/clock frequency, RAM, disk space, non-standard libraries must also be noted.

Processing Pseudocode describing the architecture of the component. This should be supplemented with flow diagrams and object/method diagrams for interactive GUIs.

Design remark Description of decisions taken and assumptions used for the current design of the component.

Parent components Listing of components using/calling this component.

Subordinate components Listing of components this component will use/call. In the case of an application this is a list of executables, but for each executables these are all the functions that are delivered along with the main function of the executable.

Dependencies from other components Description of the way this component's development relies on the development of other components, and their input or output.

ISDC context Description of where this component belongs in ISDC subsystem.

Completion box Indicates the estimated degree of completion for components not yet delivered to ISDC.

Delivery date This is the delivery date of the version of the component described in this document.

5.2 ISDC Generic Components

A very few of the executables listed in this chapter are generic functions used for performing certain basic tasks for all the instruments on INTEGRAL. Hence these are not instrument specific like the components described in this document, and we do not attempt to document them here. These components are all part of the ISDC system and are described in the document relevant to each subsystem. The ISDC generic components used in with JEM-X data are:

Observation tools ISDC tools can be used to combine single Science Windows into coherent observations. JEM-X uses only a couple of the available tools: j-ima-mosaic (p. 219) and src-collect (p. 77). A full description can be found in the Offline Scientific Analysis ADD [29]

Catalogue tools These tools find the sources that should be visible in a given pointing and are described in [26]. The JEM-X software uses cat-extract and cat-merge.

Good Time Interval Tools Used to determine GTIs from a number of different criteria, these tools are described in [25]. The tools used by JEM-X are: gti-create, gti-attitude, gti-merge and gti-import.

Quick Look Analysis These tools supplement the usual pipeline software to allow fastest possible viewing of the data for performance and systems monitoring. Some tools are used in the JEM-X pipeline, whether the software is running under QLA or SA: q-identify-srcs and q-flag-srcs. These tools are described in [25]

Event selection tools ISDC provides several library functions and some standalone tools for uses to select events from a science window or observation. The most important of these in the JEM-X context is evts-pick which can be run by the user to create a single -ALL extension which contains all the event information collected from -RAW, -PRP and -COR data structures.

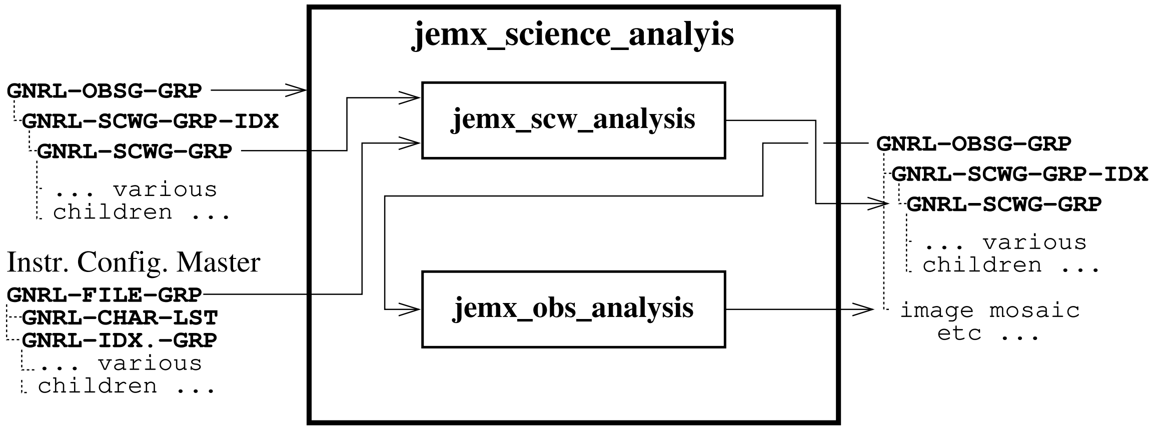

Figure 5.1: Analysis script **jemx_science_analysis** to run scientific analysis of an Observation Group.

5.3 jemx-science-analysis

Type: Analysis Script **Responsible:** ESAC (P. Kretschmar) and Valencia University (S. Martínez-Núñez)

Purpose: See subordinate components

Function: Run a full scientific analysis for JEM-X on an Observation Group.

Inputs: See subordinate components

Outputs: See subordinate components

Parameters:

ogDOL p. 48	DOL of Observation Group to be analyzed
jemxNum p. 46	JEM-X detector number
startLevel p. 52	Analysis level at which analysis begins
endLevel p. 44	Analysis level at which analysis finishes
skipLevels p. 51	Analysis levels to be skipped
chatter p. 42	Verbosity level; 2 = normal
clobber p. 42	Clobber existing output data?
ignoreScwErrors p. 46	Ignore all Science Window processing errors?
osimData p. 48	Are we using data produced by OSim?
nChanBins p. 47	Number of energy-resolved shadowgrams to create

chanLow p. 41	Lower channel boundaries for shadowgrams
chanHigh p. 41	Upper channel boundaries for shadowgrams
timeStart p. 53	Start of time interval [IJD] (-1 = use swgDOL bounds)
timeStop p. 53	End of time interval [IJD] (-1 = use swgDOL bounds)
nPhaseBins p. 48	Number of phase bins (0=no, <0=abs(N) equally spaced bins)
phaseBins p. 49	Phase bin sizes, separated by space (e.g: 0.1 0.2 0.3 ...)
radiusLimit p. 49	Use detector counts up to this radius in mm
IC-Group p. 46	DOL of Instrument Characteristics master group
IC-Alias p. 46	Selection alias for Instrument Characteristics
instMod p. 46	DOL of JEM-X Instrument Model Group (empty=find in IC)
response p. 50	Filename of standard RMF and ARF (empty=find in IC)
COR-gainHist p. 45	DOL of Index Group with gain change history
COR-gainModel p. 45	Gain history smoothing model option
COR-outputExists p. 48	Assume already existing output
GTI-gtiUser p. 46	DOL of a user GTI table (empty=no user GTI)
GTI-timeFormat p. 53	Time format to be read in user GTI table
GTI-BTI-Dol p. 41	DOL of a bad time interval table
GTI-BTI-Names p. 41	Input BTI names to be considered
GTI-attTolerance p. 40	Accepted attitude variability [arc sec]
GTI-limitTable p. 47	DOL of table with parameter limits (”=take from IC)
GTI-gtiJemxNames p. 45	Names of instrumental GTIs to be included in merged GTI
GTI-gtiScNames p. 46	Names of spacecraft GTIs to be included in merged GTI
GTI-MergedName p. 47	Name of merged GTI to be used in analysis
GTI-Accuracy p. 40	Used accuracy for OBT/IJD conversion
DEAD-outputExists p. 48	Assume already existing output
CAT-I-refCat p. 50	DOL of Reference Catalog
CAT-I-usrCat p. 53	DOL of User Catalog
CAT-I-radiusMin p. 50	Low limit for the position selection
CAT-I-radiusMax p. 50	High limit for the position selection
CAT-I-fluxDef p. 44	Column used for flux selection
CAT-I-fluxMin p. 44	Low limit for flux selection
CAT-I-fluxMax p. 44	High limit for flux selection
CAT-I-class p. 42	Object class of sources to select
CAT-I-date p. 42	IJD for the public data
BKG-simulated p. 51	Use simulated background models?

BIN-I-evtType p. 44	Event type (0=FULL, 1=REST, -1=both)

BIN-I-backCorr p. 40	Perform background correction?
BIN-I-shdType p. 51	Type of shadowgram 0 : skew, 1 : standard
BIN-I-shdRes p. 51	DOL of the output shadowgrams
BIN-I-rowSelect p. 51	CFITSIO selection string on events (optional)
BIN-I-gtiNames p. 45	Names of GTI tables to be used
IMA-makeNewBPL p. 47	Are new backprojection lists to be made ?
IMA-newBackProjFile p. 48	Base name of new backprojection file
IMA-detAccLimit p. 43	Acceptance limit in DETE-MOD array
IMA-skyImageDim p. 52	Sky image dimension 1:255, 2:511
IMA-useDeadAnodes p. 53	Include dead Anodes ?
IMA-maxNumSources p. 47	Max number of sources in IROS loop
IMA-edgeEnhanceFactor p. 43	Edge enhancement factor
IMA-loopLimitPeak p. 47	Fractional peak power to stop IROS loop
IMA-detSigSingle p. 43	Detection significance for source acceptance
IMA-skyRadiusFactor p. 52	Sky radius limit factor
IMA-radiusLimit0 p. 49	[mm] Detector radius limit $E < 6$ keV
IMA-radiusLimit1 p. 49	[mm] Detector radius limit $6 < E < 12$ keV
IMA-radiusLimit2 p. 50	[mm] Detector radius limit $12 < E < 20$ keV
IMA-radiusLimit3 p. 50	[mm] Detector radius limit 20 keV $< E$
IMA-interactionDepth p. 46	[mm] Mean depth of interaction in the detector
IMA-hotPixelLimit p. 46	Hot pixel limit factor above average
IMA-skyImagesOut p. 52	Control output of sky images
IMA-dolBPL p. 43	BPL DOL
IMA-bkgShdDOL p. 41	DOL of background shadowgrams
IMA-relDist p. 50	Maximum allowed value for reldist
IMA-fluxLimit p. 44	Lower flux limit for alerts of unfound sources [counts/s^2]
IMA-searchRad p. 51	Range of search grid around nominal found source positions
IMA-gridNum p. 45	Number of grid steps on one side of search grid
IMA-distFuzz p. 43	'Fuzz' allowed in relative distances
SPE-timeStep p. 53	Length of time bins in seconds
SPE-vignCorr p. 54	Correct for vignetting?
SPE-bgrMethod p. 41	Background subtraction method (0=Flat,1=Model)
SPE-evtType p. 44	Event type (0=FULL, 1=REST, -1=both)
SPE-precisionLevel p. 49	Speed - precision level (0 or 20)
SPE-tAccuracy p. 53	Time Correlation accuracy (0=high, 1=low, 3=any)
SPE-rowSelect p. 51	CFITSIO selection string on events (optional)"
SPE-useRaDec p. 54	Use sky coord (Ra,Dec) instead of instr. coord. (y/n)?

SPE-fluxScaling p. 44	Flux scaling (1=Open area,2=Detector area, 3=per square cm)
SPE-skipNearDeadAnode p. 52	Ignore pixels near dead anodes (y/n)?
SPE-skipHotSpot p. 51	Ignore pixels in hotspot area (y/n)?
SPE-overrideCollTilt p. 49	Collimator tilt Parameter? (if <0 use IMOD)
SPE-storePif p. 53	Store PIF data structure (1=Events 2=Events detector)?
LCR-timeStep p. 53	Length of time bins in seconds
LCR-vignCorr p. 54	Correct for vignetting?
LCR-bgrMethod p. 41	Background subtraction method (0=Flat,1=Model)
LCR-evtType p. 44	Event type (0=FULL, 1=REST, -1=both)
LCR-precisionLevel p. 49	Speed - precision level (0 or 20)
LCR-tAccuracy p. 53	Time Correlation accuracy (0=high, 1=low, 3=any)
LCR-rowSelect p. 51	CFITSIO selection string on events (optional)”
LCR-useRaDec p. 54	Use sky coord (Ra,Dec) instead of instr. coord. (y/n)?
LCR-fluxScaling p. 44	Flux scaling (1=Open area,2=Detector area, 3=per square cm)
LCR-skipNearDeadAnode p. 52	Ignore pixels near dead anodes (y/n)?
LCR-skipHotSpot p. 51	Ignore pixels in hotspot area (y/n)?
LCR-overrideCollTilt p. 49	Collimator tilt Parameter? (if <0 use IMOD)
BIN-S-rowSelectEvts p. 41	CFITSIO selection string on events (optional)
BIN-S-rowSelectSpec p. 41	CFITSIO selection string on spectrum mode (optional)
BIN-S-evtType p. 44	Data format to bin (-1=all,0=FULL,1=REST, 2=SPTI,4=SPEC)
BIN-T-rowSelect p. 51	CFITSIO selection string on events (optional)
BIN-T-evtType p. 44	Data format to bin (-1=all,0=FULL,1=REST, 2=SPTI,3=TIME)
IMA2-mapSelect p. 47	Default for Raw intensity maps or, if available, RES+SRC for Residual+Sources maps
IMA2-radiusSelect p. 50	Input images selection radius in degrees
IMA2-eminSelect p. 44	Minimum energy (keV) to select energy interval
IMA2-emaxSelect p. 44	Maximum energy (keV) to select energy interval
IMA2-diameter p. 43	Diameter in degrees of the mosaic image (If <0: not defined but may be used for larger mosaics)
IMA2-cdelt p. 41	Pixel size in degrees at mosaic image center
IMA2-RAcenter p. 49	Center of the mosaic image Right Ascension in degrees (if <0 use the computed middle position of the mosaic)
IMA2-DECcenter p. 43	Center of the mosaic image Declination in degrees (not used if RAcenter <0)
IMA2-outfile p. 48	Prefix of the mosaic FITS file for each energy band
IMA2-viewTime p. 54	Create total exposure Time map? Y/N (if Y $\rightarrow$ OBS_TIME_n.fits, where n is the number of combined input skymaps)

IMA2-viewIntens p. 54	Create raw Intensity map?
IMA2-viewVar p. 54	Create Variance map?
IMA2-viewSig p. 54	Create Significance map?
IMA2-srcFileDOL p. 52	DOL of the output source list mosaic
IMA2-srcselect p. 52	CFITSIO selection string applied to input tables
IMA2-srcattach p. 52	Attach resulting table to group?

Note: The parameters of this script are mainly passed on directly to the lower level scripts. Most of these parameters will usually be hidden.

Resources: See subordinate components

Processing:

Obtain DOLs of Science Window Groups within the given Observation Group
Pass all relevant parameters unchanged to jemx-scw-analysis
Loop over Science Window Groups:
    Call jemx-scw-analysis with DOL of current SWG
Call jemx-obs-analysis for tasks done on the full Observation Group

Design remarks:
This script is mainly a useful wrapper around jemx-scw-analysis to simplify analysis of many Science Windows at once.

Parent components: Offline Scientific Analysis environment

Subordinate components:
jemx-scw-analysis (p. 70)
jemx-obs-analysis (p. 75)

Dependencies from other components: Raw and prepared data must exist in the input swg-DOL.
Access to the Instrument Configuration Master Group and the gain history Index must be possible.

ISDC context: Science Analysis

Delivery date: v4.5.6 19/10/2005

5.4 jemx-scw-analysis

Type: Analysis Script **Responsible:** ESAC (P. Kretschmar) and Valencia University (S. Martínez-Núñez)

Purpose: See subordinate components

Function: Run a full scientific analysis for JEM-X on an Observation Group.

Inputs: See subordinate components

Outputs: See subordinate components

Parameters:

ogDOL p. 48	DOL of Observation Group to be analyzed
general-jemxNum p. 46	JEM-X detector number
startLevel p. 52	Analysis level at which analysis begins
endLevel p. 44	Analysis level at which analysis finishes
skipLevels p. 51	Analysis levels to be skipped
general-chatter p. 42	Verbosity level; 2 = normal
general-clobber p. 42	Clobber existing output data?
general-osimData p. 48	Are we using data produced by OSim?
nChanBins p. 47	Number of energy-resolved shadowgrams to create
chanLow p. 41	Lower channel boundaries for shadowgrams
chanHigh p. 41	Upper channel boundaries for shadowgrams
timeStart p. 53	Start of time interval [IJD] (-1 = use swgDOL bounds)
timeStop p. 53	End of time interval [IJD] (-1 = use swgDOL bounds)
nPhaseBins p. 48	Number of phase bins (0=no, <0=abs(N) equally spaced bins)
phaseBins p. 49	Phase bin sizes, separated by space (e.g: 0.1 0.2 0.3 ...)
radiusLimit p. 49	Use detector counts up to this radius in mm
IC-Group p. 46	DOL of Instrument Characteristics master group
IC-Alias p. 46	Selection alias for Instrument Characteristics
instMod p. 46	DOL of JEM-X Instrument Model Group (empty=find in IC)
response p. 50	Filename of standard RMF and ARF (empty=find in IC)
COR-gainHist p. 45	DOL of Index Group with gain change history
COR-gainModel p. 45	Gain history smoothing model option
COR-outputExists p. 48	Assume already existing output
GTI-gtiUser p. 46	DOL of a user GTI table (empty=no user GTI)
GTI-timeFormat p. 53	Time format to be read in user GTI table

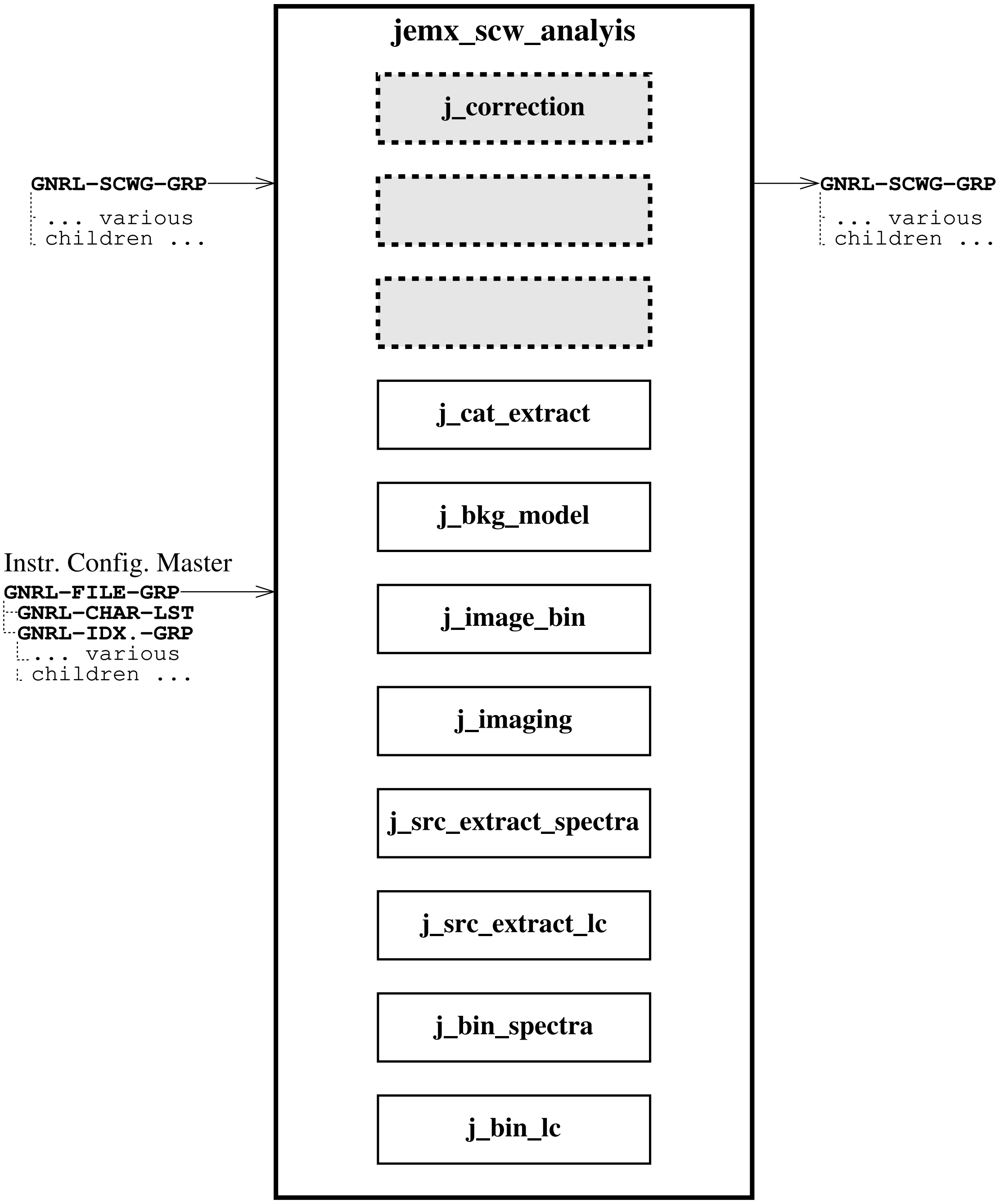

Figure 5.2: Analysis script **jemx_scw_analysis** to run scientific analysis of a Science Window

GTI-BTI-Dol p. 41	DOL of a bad time interval table
GTI-BTI-Names p. 41	Input BTI names to be considered
GTI-attTolerance p. 40	Accepted attitude variability [arc sec]
GTI-limitTable p. 47	DOL of table with parameter limits (”=take from IC)
GTI-gtiJemxNames p. 45	Names of instrumental GTIs to be included in merged GTI
GTI-gtiScNames p. 46	Names of spacecraft GTIs to be included in merged GTI
GTI-MergedName p. 47	Name of merged GTI to be used in analysis
GTI-Accuracy p. 40	Used accuracy for OBT/IJD conversion
DEAD-outputExists p. 48	Assume already existing output
CAT-I-refCat p. 50	DOL of Reference Catalog
CAT-I-usrCat p. 53	DOL of User Catalog
CAT-I-radiusMin p. 50	Low limit for the position selection
CAT-I-radiusMax p. 50	High limit for the position selection
CAT-I-fluxDef p. 44	Column used for flux selection
CAT-I-fluxMin p. 44	Low limit for flux selection
CAT-I-fluxMax p. 44	High limit for flux selection
CAT-I-class p. 42	Object class of sources to select
CAT-I-date p. 42	IJD for the public data
BKG-simulated p. 51	Use simulated background models?
BIN-I-evtType p. 44	Event type (0=FULL, 1=REST, -1=both)
BIN-I-backCorr p. 40	Perform background correction?
BIN-I-shdType p. 51	Type of shadowgram 0 : skew, 1 : standard
BIN-I-shdRes p. 51	DOL of the output shadowgrams
BIN-I-rowSelect p. 51	CFITSIO selection string on events (optional)
BIN-I-gtiNames p. 45	Names of GTI tables to be used
IMA-makeNewBPL p. 47	Are new backprojection lists to be made ?
IMA-newBackProjFile p. 48	Base name of new backprojection file
IMA-detAccLimit p. 43	Acceptance limit in DETE-MOD array
IMA-skyImageDim p. 52	Sky image dimension 1:255, 2:511
IMA-useDeadAnodes p. 53	Include dead Anodes ?
IMA-maxNumSources p. 47	Max number of sources in IROS loop
IMA-edgeEnhanceFactor p. 43	Edge enhancement factor
IMA-loopLimitPeak p. 47	Fractional peak power to stop IROS loop
IMA-detSigSingle p. 43	Detection significance for source acceptance
IMA-skyRadiusFactor p. 52	Sky radius limit factor
IMA-radiusLimit0 p. 49	[mm] Detector radius limit $E < 6$ keV

IMA-radiusLimit1 p. 49	[mm] Detector radius limit $6 < E < 12$ keV
IMA-radiusLimit2 p. 50	[mm] Detector radius limit $12 < E < 20$ keV
IMA-radiusLimit3 p. 50	[mm] Detector radius limit 20 keV $< E$
IMA-interactionDepth p. 46	[mm] Mean depth of interaction in the detector
IMA-hotPixelLimit p. 46	Hot pixel limit factor above average
IMA-skyImagesOut p. 52	Control output of sky images
IMA-dolBPL p. 43	BPL DOL
IMA-bkgShdDOL p. 41	DOL of background shadowgrams
IMA-relDist p. 50	Maximum allowed value for reldist
IMA-fluxLimit p. 44	Lower flux limit for alerts of unfound sources [counts/s^2]
IMA-searchRad p. 51	Range of search grid around nominal found source positions
IMA-gridNum p. 45	Number of grid steps on one side of search grid
IMA-distFuzz p. 43	‘Fuzz’ allowed in relative distances
SPE-timeStep p. 53	Length of time bins in seconds
SPE-vignCorr p. 54	Correct for vignetting?
SPE-bgrMethod p. 41	Background subtraction method (0=Flat,1=Model)
SPE-evtType p. 44	Event type (0=FULL, 1=REST, -1=both)
SPE-precisionLevel p. 49	Speed - precision level (0 or 20)
SPE-tAccuracy p. 53	Time Correlation accuracy (0=high, 1=low, 3=any)
SPE-rowSelect p. 51	CFITSIO selection string on events (optional)”
SPE-useRaDec p. 54	Use sky coord (Ra,Dec) instead of instr. coord. (y/n)?
SPE-fluxScaling p. 44	Flux scaling (1=Open area,2=Detector area, 3=per square cm)
SPE-skipNearDeadAnode p. 52	Ignore pixels near dead anodes (y/n)?
SPE-skipHotSpot p. 51	Ignore pixels in hotspot area (y/n)?
SPE-overrideCollTilt p. 49	Collimator tilt Parameter? (if <0 use IMOD)
SPE-storePif p. 53	Store PIF data structure (1=Events 2=Events detector)?
LCR-timeStep p. 53	Length of time bins in seconds
LCR-vignCorr p. 54	Correct for vignetting?
LCR-bgrMethod p. 41	Background subtraction method (0=Flat,1=Model)
LCR-evtType p. 44	Event type (0=FULL, 1=REST, -1=both)
LCR-precisionLevel p. 49	Speed - precision level (0 or 20)
LCR-tAccuracy p. 53	Time Correlation accuracy (0=high, 1=low, 3=any)
LCR-rowSelect p. 51	CFITSIO selection string on events (optional)”
LCR-useRaDec p. 54	Use sky coord (Ra,Dec) instead of instr. coord. (y/n)?
LCR-fluxScaling p. 44	Flux scaling (1=Open area,2=Detector area, 3=per square cm)
LCR-skipNearDeadAnode p. 52	Ignore pixels near dead anodes (y/n)?
LCR-skipHotSpot p. 51	Ignore pixels in hotspot area (y/n)?

LCR-overrideCollTilt p. 49	Collimator tilt Parameter? (if <0 use IMOD)
BIN-S-rowSelectEvts p. 41	CFITSIO selection string on events (optional)
BIN-S-rowSelectSpec p. 41	CFITSIO selection string on spectrum mode (optional)
BIN-S-evtType p. 44	Data format to bin (-1=all,0=FULL,1=REST, 2=SPTI,4=SPEC)
BIN-T-rowSelect p. 51	CFITSIO selection string on events (optional)
BIN-T-evtType p. 44	Data format to bin (-1=all,0=FULL,1=REST, 2=SPTI,3=TIME)

Note: The parameters of this script are mainly passed on directly to the lower level scripts. Most of these parameters will usually be hidden.

Resources: See subordinate components

Processing:

Loop over a given Science Window from COR to BIN_T, passing parameters unchanged to each child.

Design remarks:
This script runs all the lower-level scripts for a gicen Science Window.

Parent components:
jemx-science-analysis (p. 65)

Subordinate components:
j-correction (p. 108)
j-gti (p. 126)
j-dead (p. 131)
j-cat-extract (p. 138)
j-bkg-model (p. 141)
j-image-bin (p. 154)
j-imaging (p. 161)
j-src-extract-spectra (p. 171)
j-src-extract-lc (p. 181)
j-bin-spectra (p. 191)
j-bin-lc (p. 206)

Dependencies from other components: Raw and prepared data must exist in the input swg-DOL.
Access to the Instrument Configuration Master Group and the gain history Index must be possible.

ISDC context: Science Analysis

Delivery date: v4.5.6 19/10/2005

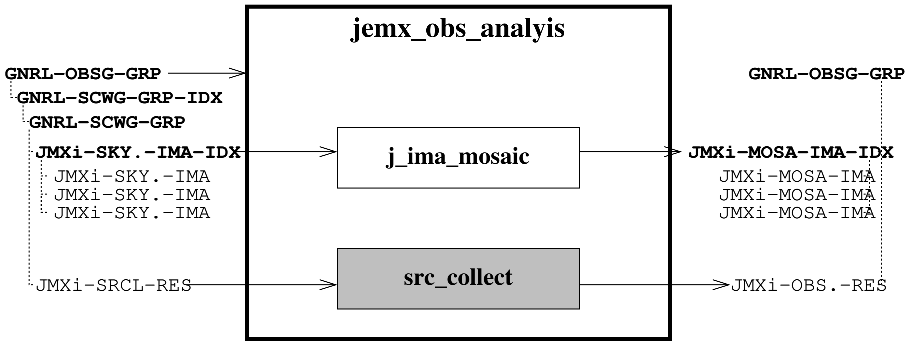

Figure 5.3: Analysis script **jemx_obs_analysis** to run some group tasks on a full Observation Group

5.5 jemx-obs-analysis

Type: Analysis Script **Responsible:** ESAC (P. Kretschmar) and Valencia University (S. Martínez-Núñez)

Purpose: Not foreseen in SRD

Function: Group analysis tasks done on the whole Observation Group, as wrap-up after the actual analysis per ScW.

Inputs:
JMXi-SKY.-IMA-IDX (p. 243)
JMXi-SKY.-IMA (p. 243)
JMXi-SRCL-RES (p. 245)

Outputs:
JMXi-MOSA-IMA-IDX (p. 239)
JMXi-MOSA-IMA (p. 239)
JMXi-OBS.-RES (p. 239)

Parameters:

ogDOL p. 48	DOL of Observation Group to be analyzed
jemxNum p. 46	JEM-X detector number
startLevel p. 52	Analysis level at which analysis begins (currently, only IMA2)
endLevel p. 44	Analysis level at which analysis finishes (currently, only IMA2)
skipLevels p. 51	Analysis levels to be skipped
IC-Group p. 46	DOL of Instrument Characteristics master group
IC-Alias p. 46	Selection alias for Instrument Characteristics
chatter p. 42	Verbosity level; 2 = normal
clobber p. 42	Clobber existing output data?
subDir p. 53	Sub directory below the position of the OG for results
mapSelect p. 47	Default for Raw intensity maps or, if available, RES+SRC for Residual+Sources maps
radiusSelect p. 50	Input images selection radius in degrees
eminSelect p. 44	Minimum energy (keV) to select energy interval
emaxSelect p. 44	Maximum energy (keV) to select energy interval
diameter p. 43	Diameter in degrees of the mosaic image (If <0: not defined but may be used for larger mosaics)
cdelt p. 41	Pixel size in degrees at mosaic image center
RAcenter p. 49	Center of the mosaic image Right Ascension in degrees (if <0 use the computed middle position of the mosaic)
DECcenter p. 43	Center of the mosaic image Declination in degrees (not used if RAcenter <0)
outfile p. 48	Prefix of the mosaic FITS file for each energy band
viewTime p. 54	Create total exposure Time map? Y/N (if Y → OBS_TIME_n.fits, where n is the number of combined input skymaps)
viewIntens p. 54	Create raw Intensity map? Y/N
viewVar p. 54	Create Variance map? Y/N
viewSig p. 54	Create Significance map? Y/N
srcFileDOL p. 52	DOL of the output combined source list
srcselect p. 52	CFITSIO selection string applied to input source lists
srcattach p. 52	Attach resulting source list table to Obs. Group?

Resources: See subordinate components

Processing:
Creates mosaiced images and exposure maps from all available images within the group.
Create a combined list of found sources with src-collect.

Design remarks:

Parent components:
jemx-science-analysis (p. 65)

Subordinate components:
j-ima-mosaic (p. 219)
src-collect (p. 77)

Dependencies from other components: jemx-scw-analysis must have run on all Science Windows before, at least up to level IMA.

ISDC context: Science Analysis

Delivery date: 4.5.6 19/10/2005

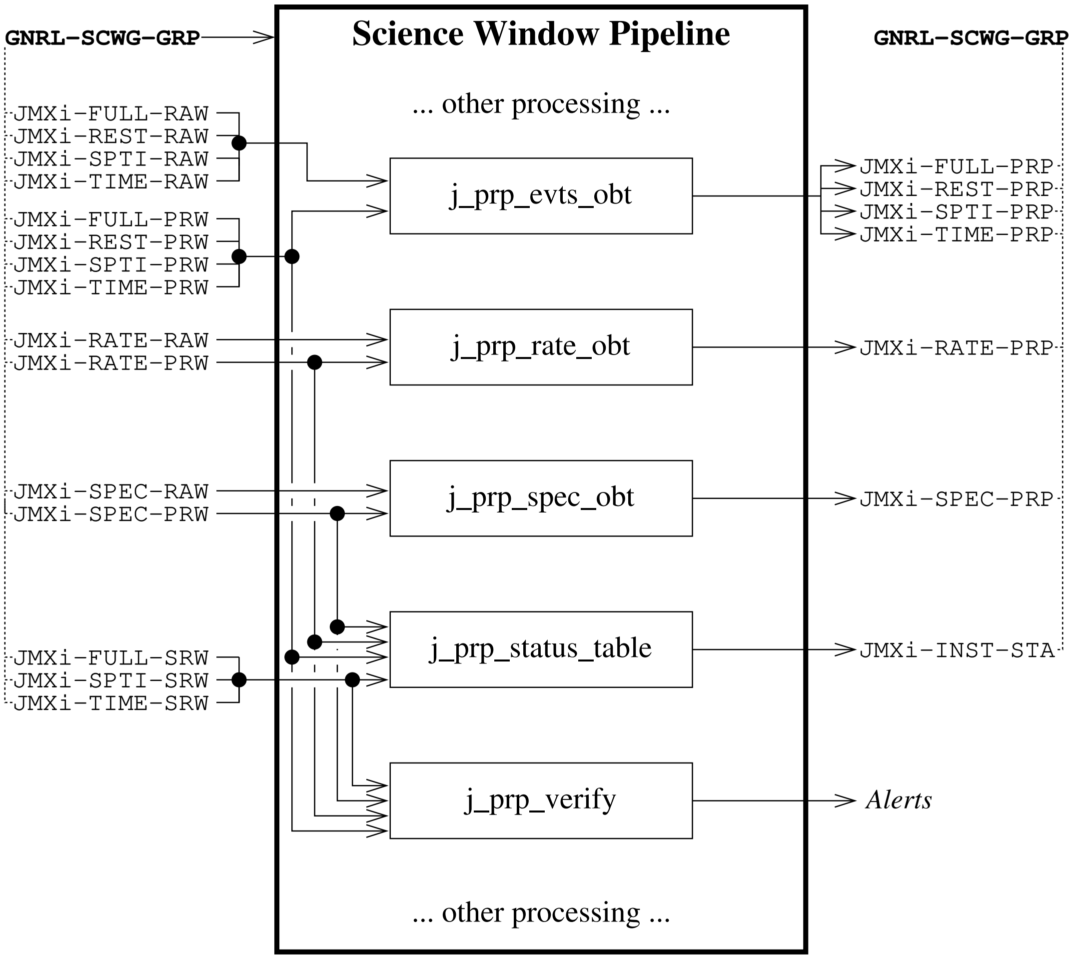

Figure 5.4: Data Preparation

5.6 Data Preparation

Type: Pipeline package **Responsible:** ISDC

Purpose: 18.1.1 18.10.6

Function: Create data for subsequent analysis steps that can not be sorted out by ISDC's Pre-Processing. This is part of ISDC Technical processing

Inputs:
JMXi-FULL-RAW (p. 238)
JMXi-FULL-PRW (p. 238)

JMXi-REST-RAW (p. 242)
JMXi-REST-PRW (p. 241)
JMXi-SPTI-RAW (p. 244)
JMXi-SPTI-PRW (p. 244)
JMXi-TIME-RAW (p. 245)
JMXi-TIME-PRW (p. 245)
JMXi-RATE-RAW (p. 241)
JMXi-RATE-PRW (p. 241)
JMXi-SPEC-RAW (p. 244)
JMXi-SPEC-PRW (p. 244)
JMXi-FRSS-CRW (p. 237)
JMXi-FULL-SRW (p. 238)
JMXi-SPTI-SRW (p. 244)
JMXi-TIME-SRW (p. 246)

Outputs:
JMXi-FULL-PRP (p. 237)
JMXi-REST-PRP (p. 241)
JMXi-SPTI-PRP (p. 244)
JMXi-TIME-PRP (p. 245)
JMXi-RATE-PRP (p. 241)
JMXi-SPEC-PRP (p. 244)
JMXi-GAIN-CAL (p. 238)
JMXi-INST-STA (p. 239)
JMXi-DEAD-SCP (p. 234)

Parameters:
N/A

Resources: No unusual

Processing:

From the ISDC Science Window Pipeline:
  Call j-prp-evts-obt to calculate full on-board time for event data
  Call j-prp-rate-obt to calculate full on-board time for countrate data
  Call j-prp-spec-obt to calculate full on-board time for spectrum data
  Call j-create-status-table to derive a detailed history of the grey filter and TM format changes
  Call j-verify to check the internal consistency of the raw data
From the ISDC Revolution File Pipeline :

Call j-prp-frss-obt to calculate full on-board time for FRSS calibration spectra
Call j-calib-gain-fitting to calculate gain from calibration spectra
Call j-calib-adc to check the onboard electronic calibration data

Design remarks:
This is not strictly speaking a software component but a useful header to group those tools within Data Preparation that are JEM-X specific.

Parent components:
Science Window Pipeline (p. 30)
Revolution File Pipeline (p. 30)

Subordinate components:
j-prp-evts-obt (p. 81)
j-prp-rate-obt (p. 83)
j-prp-spec-obt (p. 85)
j-prp-frss-obt (p. 87)
j-prp-status-table (p. 89)
j-prp-verify (p. 95)
j-calib-gain-fitting (p. 99)
j-calib-adc (p. 104)

Dependencies from other components: Pre-Processing must have created sets of raw science data, as well as unpacking electronic calibration data and calibration spectra.
The time correlation must be available for the entire time period covered so that full OBTs can be calculated. This information is not available during PreProcessing which is why these components must wait until Data Preparation.

ISDC context: Data Preparation in ScW Pipeline and Rev. file pipeline

Delivery date: See individual components

5.6.1 j-prp-evts-obt

Type: Executable **Responsible:** ISDC (N. Produit)

Purpose: 18.10.6 18.11.3 18.12.1

Function: Generate full on-board time information for event data.

Inputs:
JMXi-FULL-RAW (p. 238)
JMXi-FULL-PRW (p. 238)
JMXi-SPTI-RAW (p. 244)
JMXi-SPTI-PRW (p. 244)
JMXi-TIME-RAW (p. 245)
JMXi-TIME-PRW (p. 245)
JMXi-REST-RAW (p. 242)
JMXi-REST-PRW (p. 241)

Outputs:
JMXi-FULL-PRP (p. 237)
JMXi-SPTI-PRP (p. 244)
JMXi-TIME-PRP (p. 245)
JMXi-REST-PRP (p. 241)
JMXi-DEAD-SCP (p. 234)

Parameters:

inSWG p. 46	DOL of Science Window Group if not modified
outSWG p. 49	DOL of Science Window Group if modified
jemxNum p. 46	JEM-X detector number
clobber p. 42	Clobber existing output data?

Resources: Largely dominated by event buffers for input data. For REST events approximately $6600 \times T$ bytes will be allocated for input buffers, where T is the duration of the Science Window in seconds. For other modes the usage is lower.

Processing:
Get the parameters
Get number of packets
For each packets read OBT of packet ($= T0$)
Read all the events delta time ($= DT$) in packet (for REST assume all $DT \equiv 0$)
add $T0$ and DT ($= T1$)

compute full OBT ($= T2$) from $T1$
write all $T2$ to PRP data structure

Design remarks:
The translation of the OBT information is required because the satellites local OBT will cycle approximately every 200 days and may have resets. The translation can not be made as part of PreProcessing, because it can not be guaranteed that all required information is available at that time.

Parent components:
Science Window Pipeline (p. 30)

Subordinate components:

Dependencies from other components: OBT manipulation functions in DAL3GEN

ISDC context: Data Preparation

Delivery date: V2.1 11/11/2002

5.6.2 j-prp-rate-obt

Type: Executable **Responsible:** ISDC (N. Produit)

Purpose: 18.10.6

Function: Generate full on-board time information for countrate data.

Inputs:
JMXi-RATE-RAW (p. 241)
JMXi-RATE-PRW (p. 241)

Outputs:
JMXi-RATE-PRP (p. 241)

Parameters:

inSWG p. 46	DOL of Science Window Group if not modified
outSWG p. 49	DOL of Science Window Group if modified
jemxNum p. 46	JEM-X detector number
clobber p. 42	Clobber existing output data?

Resources: Dominated by buffers for input countrate data — $\sim 75 \times T$ bytes, where T is the duration of the Science Window in seconds. In total negligible.

Processing:
Get the parameters
For each packet read OBT of packet $(= T0)$
for all events in packets $(0 \ldots n)T1 = T0 + i * 125\text{ms}$
compute full OBT $(= T2)$ from $T1$
write all $T2$ to PRP data structure

Design remarks:
The translation of the OBT information is required because the satellites local OBT will cycle approximately every 200 days and may have resets. The translation can not be made as part of PreProcessing, because it can not be guaranteed that all required information is available at that time.

Parent components:
Science Window Pipeline (p. 30)

Subordinate components:

Dependencies from other components: None

ISDC context: Data Preparation

Delivery date: V1.6 07/02/2002

5.6.3 j-prp-spec-obt

Type: Executable **Responsible:** ISDC (N. Produit)

Purpose: 18.10.6 18.12.1

Function: Generate full on-board time information for spectrum mode data.

Inputs:
JMXi-SPEC-RAW (p. 244)
JMXi-SPEC-PRW (p. 244)

Outputs:
JMXi-SPEC-PRP (p. 244)

Parameters:

inSWG p. 46	DOL of Science Window Group if not modified
outSWG p. 49	DOL of Science Window Group if modified
jemxNum p. 46	JEM-X detector number
clobber p. 42	Clobber existing output data?

Resources: Dominated by buffers for input countrate data — $\sim 75 \times T$ bytes, where T is the duration of the Science Window in seconds. In total negligible.

Processing:
Get the parameters
For each packets read OBT of packet ($= T0$)
for all events in packets $(0 \ldots n) T1 = T0 + i * 125$ms
compute full OBT ($= T2$) from $T1$
write all $T2$ to PRP data structure

Design remarks:
The translation of the OBT information is required because the satellites local OBT will cycle approximately every 200 days and may have resets. The translation can not be made as part of PreProcessing, because it can not be guaranteed that all required information is available at that time.

Parent components:
Science Window Pipeline (p. 30)

Subordinate components:

Dependencies from other components: None

ISDC context: Data Preparation

Delivery date: V1.6 07/02/2002

5.6.4 j-prp-frss-obt

Type: Executable **Responsible:** ISDC (N. Produit)

Purpose: 18.10.6

Function: Generate full on-board time information for FRSS spectra.

Inputs:
JMXi-FRSS-CRW (p. 237)

Outputs:
JMXi-GAIN-CAL (p. 238)

Parameters:

inDOL p. 46	DOL of input data (raw FRSS spectra)
outDOL p. 48	DOL of output data (empty gain calibration results table)
jemxNum p. 46	JEM-X detector number
clobber p. 42	Clobber existing output data?

Resources: Negligible as only a few keywords are processed.

Processing:
Get the parameters
Use input attributes PCKSTART and REVOL to calculate times in OBT and IJD
Store attributes OBT_ACQ, INT_TIME, TSTART and TSTOP in output table

Design remarks:
The translation of the OBT information is required because the satellites local OBT will cycle approximately every 200 days and may have resets. The translation can not be made as part of PreProcessing, because it can not be guaranteed that all required information is available at that time.

Parent components:
Revolution File Pipeline (p. 30)

Subordinate components:

Dependencies from other components: None

ISDC context: Data Preparation

Delivery date: V2.2 11/04/2002

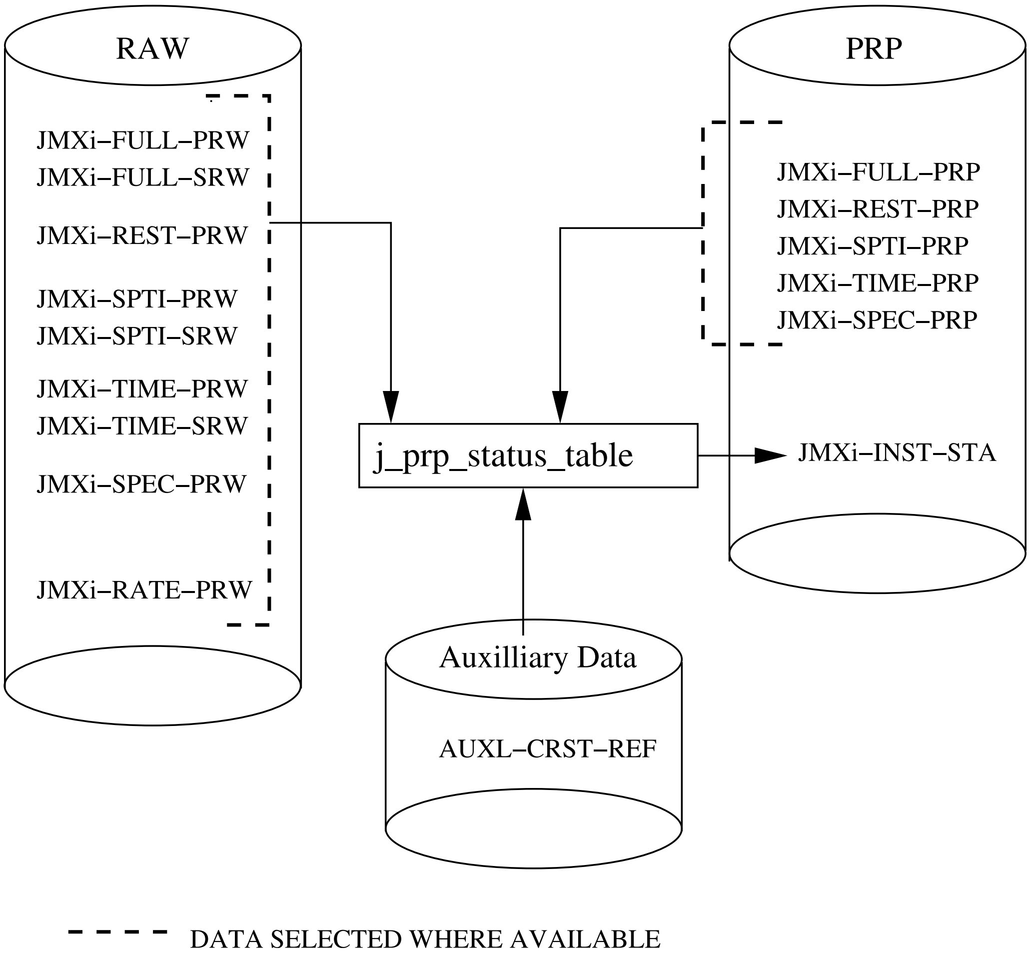

Figure 5.5: j-prp-status-table

5.6.5 j-prp-status-table

Type: Executable **Responsible:** DNSC (C.A. Oxborrow)

Purpose: Not forseen in SRD

Function: To make an instrument status table containing values of the grey filter and instrument data-taking mode as a function of time.

Interfaces: Requires OBTs from previous executables in level PRP

Inputs:
GNRL-SCWG-GRP (p. 249)
JMXi-FULL-PRW (p. 238)

JMXi-FULL-SRW (p. 238)
JMXi-FULL-PRP (p. 237)
JMXi-REST-SRW (p. 242)
JMXi-REST-PRP (p. 241)
JMXi-SPTI-PRW (p. 244)
JMXi-SPTI-SRW (p. 244)
JMXi-SPTI-PRP (p. 244)
JMXi-SPEC-PRW (p. 244)
JMXi-SPEC-PRP (p. 244)
JMXi-TIME-PRW (p. 245)
JMXi-TIME-PRP (p. 245)
JMXi-TIME-SRW (p. 246)
JMXi-RATE-RAW (p. 241)
JMXi-RATE-PRW (p. 241)
JMXi-RATE-PRP (p. 241)
AUXL-CRST-REF (p. 247)

Outputs:
JMXi-INST-STA (p. 239)

Resources: Processing time should be directly proportional to the number of events being processed. For just over 600,000 events, using 83% of the CPU, this takes 0.54 seconds of processor time.
Memory requirements are dominated by fixed-sized arrays for storing cumulative lists of mode and grey changes, currently set unrealistically large to accomodate long sets of lab test data. Currently arrays require 200K of memory, but during routine operations with 1000 sec. science windows, no more than 20K will be needed. Code size is just under 1.6M.

Parameters:

inSWG p. 46	DOL	Input Software Window Group
outSWG p. 49	DOL	Output Software Window Group
instStat p. 46	DOL	Override output instrument status table
fullSrw p. 45	DOL	Override input FULL event secondary raw data
fullPrw p. 44	DOL	Override input FULL event packet raw data
fullPrp p. 44	DOL	Override input FULL event OBT data
restSrw p. 51	DOL	Override input REST event secondary raw data
restPrp p. 50	DOL	Override input REST event OBT data
sptiSrw p. 52	DOL	Override input SPTI event secondary raw data
sptiPrw p. 52	DOL	Override input SPTI event packet raw data
sptiPrp p. 52	DOL	Override input SPTI event OBT data
specRaw p. 52	DOL	Override input SPEC raw data

specPrp p. 52	DOL	Override input SPEC OBT data
timeSrw p. 53	DOL	Override input TIME event secondary raw data
timePrw p. 53	DOL	Override input TIME event packet raw data
timePrp p. 53	DOL	Override input TIME event OBT data
rateSrw p. 50	DOL	Override input RATE secondary raw data
ratePrw p. 50	DOL	Override input RATE packet raw data
ratePrp p. 50	DOL	Override input RATE OBT data
jemxNum p. 46	int	Number of jemx unit that generated data (1 or 2)
clobber p. 42	boo	Clobber existing output data structures?
chatter p. 42	int	Verbosity level of log file (0,1,2,3,4,5)

Processing:
Initialize executable with CommonInit

Read the input parameter file
    Read the input parameters
    Open SWG
    Find instrument number: 1 or 2
    Find or create the need output extension
Find out how many events of each kind of data-taking mode there are
If there FULL events available:
    Find initial packet grey values from JMXi-FULL-PRW
    Read packet event starting rows
    Read packet SSCs
    Read packet start OBTs from JMXi-FULL-PRP
    Add packet starting grey values to grey value list
    Add packet times to OBT list
    Add mode number to mode list
    If there are autonomous grey changes in JMXi-****-SRW:
        Read the grey values and SSC values for each filter change
        Read the position of each grey change within its packet
        Read delta time for each grey change
        Read starting OBT of packet containing the grey change
        Read event OBT at the position of the grey change event
        Find grey OBT from packet OBT, event OBT and delta time
        Add grey change values to grey value list
        Add grey change time to OBT list
        Add mode number to mode list

If there are REST events:
    Read GREY_FIRST values for each REST image from JMXi-REST-SRW
    Read ROW_BEGIN for each image
    Read image OBTs from JMXi-REST-PRP
    Add first grey values in image to grey value list
    Add image start OBTs to OBT list
    Add REST mode to mode list
    Read GREY_LAST values for each REST image from JMXi-REST-SRW
    Read LENGTH_IMAGE value for each image
    Make last grey value in image OBT
    Add last grey values in image to grey value list
    Add last grey value OBTs to OBT list
    Add REST mode to mode list
If there are SPTI events available:
    Processing is same as for FULL events
If there are TIME events available:
    Processing is same as for FULL events
If there are SPEC spectra available :
    Read GREY_FILTER values from JMXi-SPEC-RAW
    Read the OB_TIME value for each spectrum from JMXi-SPEC-PRP
    Add the first grey value to the grey list
    Add the first spectral OBT to the OBT list
    Add SPEC to the mode list
    If there has been a change in the grey filter value:
        Add the grey filter values to the grey value list
        Add the spectra OBTs to the OBT list
        Add SPEC to the mode list
If there is RATE data available:
    Read GREY_FILTER value from JMXi-RATE-RAW
    Read OB_TIME values from JMXi-RATE-PRP
    Add the first grey value to the grey list
    Add the first OBT to the OBT list
    Add RATE to the mode list
    If there has been a change in the grey filter value:
        Add the grey filter values to the grey value list
        Add the OBTs to the OBT list
        Add RATE to the mode list
    Read GREY_FILTER values from JMXi-RATE-PRW
    Read the packet start rows and TIME_BINS
    Read grey filter reports for each packet
    As above add to master lists if there’s been a change in grey filter

Sort the grey, OBT and mode lists by OBT
Prune the master lists to remove repeated information
Save the pruned master list to JMXi-INST-STA columns
Close SWG and tidy up

Design remarks:

- This executable gathers information from a large number of different data structures, but most of the processing performed is sorting and pruning. Only OBTs which must be constructed from various event times and delta times involve any real processing
- The clock reset reference file is automatically located through the ISDC environmental variable AUXL_REF_DIR. Hence there is no DOL parameter for this input, and no override DOL.
- The final output table JMXi-INST-STA contains entries only where changes in grey filter value or data-taking mode occur.
- Program was last updated to expand size of grey filter dummy event delta times to 2 bytes (originally 1 byte), though no such large delta time has yet been seen in the mission data.

Parent components:
Science Window Pipeline (p. 30)

Subordinate components: Functions of the executable:
JCreateStatus-Params
JCreateStatus-Work
JCreateStatus-Save
JCreateStatus-Sort
JCreateStatus-Partition
JCreateStatus-Prune
j-prp-status-FST
j-prp-status-rest
j-prp-status-spec
j-prp-status-rate

Dependencies from other components: Assumes that the other j_data_preparation executables have run first to produce OBTs

ISDC context: Runs as part of j_data_preparation, which is part of the JEM-X Near Real Time Science Window Pipeline. See figures on pages 18 and 19

Delivery date: Version 3.8 delivered 2004/05/26

5.6.6 j-prp-verify

Type: Executable **Responsible:** DNSC (C.A. Oxborrow)

Purpose: Not forseen in SRD or URD

Function: To verify that the raw science data coming out of ISDC preprocessing and j_preparation has all the features of well-behaved JEM-X data and produce alerts if raw data coming out of Science Window Pipeline shows sign of pathologies. Alerts can be switched off by setting input parameter 'chatter' to 0 or 1. Event populations for each anode are also extracted to monitor anode health

Inputs:
GNRL-SCWG-GRP (p. 249)
JMXi-FULL-RAW (p. 238)
JMXi-FULL-SRW (p. 238)
JMXi-FULL-PRW (p. 238)
JMXi-FULL-PRP (p. 237)
JMXi-REST-RAW (p. 242)
JMXi-REST-PRW (p. 241)
JMXi-REST-SRW (p. 242)
JMXi-REST-PRP (p. 241)
JMXi-SPTI-RAW (p. 244)
JMXi-SPTI-SRW (p. 244)
JMXi-SPTI-PRW (p. 244)
JMXi-SPTI-PRP (p. 244)
JMXi-SPEC-RAW (p. 244)
JMXi-SPEC-PRW (p. 244)
JMXi-SPEC-PRP (p. 244)
JMXi-TIME-RAW (p. 245)
JMXi-TIME-PRW (p. 245)
JMXi-TIME-PRP (p. 245)
JMXi-RATE-RAW (p. 241)
JMXi-RATE-PRW (p. 241)
JMXi-RATE-PRP (p. 241)
JMXi-IMOD-GRP (p. 239)
JMXi-DETE-MOD (p. 235) In the instrument model group

Outputs: Alerts

JMXi-ANOD-PRP (p. 232)

Parameters:

inSWG p. 46	DOL of input science window group
outSWG p. 49	DOL of output Science Window Group
anodHis p. 40	DOL of override output anode histogram
instMod p. 46	DOL of the current instrument model group
jemxNum p. 46	Number of jem-x instrument (1 or 2)
alrtLevel p. 40	Maximum level of alerts produced (0,1,2,3)
chatter p. 42	Verbosity of logfile output (0,1,2,3,4,5)
clobber p. 42	Clobber existing output data structure (y/n)?

Resources: Executable memory required is just under 4 MByte
Memory required for arrays at runtime is between 1.0 and 1.5 MBytes for a SWG containing 500,000 FULL imaging events. The executable is designed to read and store only the information about the events needed at one time, and then release the storage again to keep memory requirements to a minimum.
Running time: on a Sun Solaris 8 machine, execution time is 0.15 seconds, using 4% of the CPU, to process about 25,000 events

Processing:

Initialise executable
Read in the parameter file:
    Read the input parameters and find the instrument string
    Open the SWG and the instrument model group
Find the number of FULL, REST, SPTI, SPEC, TIME and RATE events
Loop over data-taking MODES: (FULL, REST, SPTI, SPEC, TIME, RATE):
    Open the -RAW data structure for MODE and find nEvents
    IF there are raw events of this MODE type perform checks:
        else move onto the next data mode
    Check that all the expected files are present and open them:
        For FULL: -PRW, -SRW, and -PRP
        For REST: -PRW, -SRW, -PRP and -RATE-RAW
        For SPTI: -PRW, -SRW, and -PRP
        For SPEC: -PRW, -PRP and -RATE-RAW
        For TIME: -PRW, -SRW and -PRP
        For RATE: -PRW and -PRP
    Check each found data structure contains expected rows
    Check that there's enough packet data for the number of events
    Open the -PRW data structure for the mode
        Read in the packet starting event row numbers
    For imaging modes (FULL and REST) only :
        Read in the detector area model image from JMXi-DETE-MOD
        Read in the RAW_X and RAW_Y values for the mode

Check that each event falls on the active area of the detector
Loop over values of RAW_X (individual anodes)
Sum up the events falling on the anode for the entire SCW
Normalise anode populations to average expected population
Save normalised population histogram to JMXi-ANOD-PRP
For all modes verify the event OBTS:
Read in packet LOBTs from JMXi-****-PRW
Read in event or spectra OBTs from JMXi-****-PRP
Read in event DELTA_TIME from JMXi-****-RAW
Check that OBTs are continuous across packet boundaries
Check that event OBTs agree with packet OBTs and delta OBTs
End loop over the different data-taking modes
Tidy up and exit: close SWG and JMXi-ANOD-PRP

Design remarks:

- If the ‘chatter’ parameter is less than 2 all alerts are suppressed and not sent to the ISDC alert system. However the suppressed alerts are recorded in the log file.
- There are no override input DOLS for the data extensions in the SWG since the point of this tool is to check all the data in a given SWG. It is an automatic tool for checking the health of the pipeline, not an interactive tool for users.
- Every discrepancy in the data will produce an alert with a level appropriate to the severity of the problem, though with highest alert level given by alrtLevel.
- Future versions of this executable can expect to do many more checks, though the interface should remain unchanged.
- The various functions are arranged so that only the part of the data being currently checked is loaded at any one time. When all the checks on that data have been preformed the memory is freed to minimise the amount of memory required at any one time. The executable does not read in all available raw data in the SWG at once.
- The single output extension JMXi-ANOD-PRP is intended for offline analysis only, and is not used further down the analysis pipeline.
- Though the output extension is of level PRP, it is saved in the engineering subdirectory of the SCW directory so that all data, including that from private observations can be viewed for health monitoring.

- Current version of program supercedes versions 3.x which produced automatic alerts if an anode seemed to have fewer events than expected on average. However this system produced too many alerts and such checking is now done offline.
- Program produces the following alerts:

  5000 Invalid data-taking mode

  5001 Missing or invalid combination of raw data extensions

  5002 Dead/dying anode (disabled)

  5008 Event has bad or inconsistent OBT

  5007 Many event positions outside active detector area

Parent components:
Science Window Pipeline (p. 30)

Subordinate components:
j-prp-verify-params
j-prp-verify-files
j-verify-get-pcks
j-verify-xypos
j-verify-obts
j-verify-anodes

Dependencies from other components: 1) OBTs must be calculated in j_data_preparation before this executable can be run. Ideally j_prp_verify should be the last executable in the JEM-X Science Window Pipeline.

ISDC context: Science Window Pipeline

Delivery date: Version 4.1 delivered 2004/01/27

5.6.7 j-calib-gain-fitting

Type: Executable **Responsible:** DNSC (C.A. Oxborrow)

Purpose: 9.17 (QLA), 18.4.1 18.4.2 18.4.3 18.4.5 18.11.1 18.10.1 18.13.1

Function: To calculate gain by fitting a spectrum of calibration events to a model of unresolved Gaussian doublets. The calibration spectra contain an unresolved doublet of known energy from which energy offset and scaling can be determined for the detector ADC bins. Other outputs are: fitting residuals re-binned to PI channels, energy resolution for Fe and/or Cd doublets, fitted peak position in ADC channels, and source intensities.

Interfaces:
1) Uses corrected calibration energy binning (PHA) table and the full imaging PI binning table, both in the instrument model group JMXi-IMOD-GRP (p. 239)
2) The calibration spectra JMXi-FRSS-CRW are passed to the executable via a small index table since these are not processed by science window but individually within the Revolution File Pipeline. The index table is used to group several sets of data at a time so that the pipeline isn't triggered every 4.2 minutes, every time a new set of calibration spectra appears. The results are indexed in another small index table already prepared by the pipeline with entries for the empty extensions waiting to be filled by this executable.
3) Results are passed on to j-cor-gain (p. 112) by means of a master index table of the output JMXi-GAIN-CAL extensions. This index table is made by the pipeline script, not by executable j-calib-gain-fitting. The master index table will cover the entire mission, and should not be confused with the small index table used to group output from one run of this executable, though they will have the same extension name: JMXi-GAIN-CAL-IDX.
4) OBTs for each set of four calibration spectra are created by j_prp_frss_obt before this component runs, and the prepared OBTs are already available in the JMXi-GAIN-CAL-IDX output index at run time.
5) Individual gain history tables of the form JMXi-GAIN-CAL-IDX contain the found Cd/Fe peak positions as tabulated keywords from the JMXi-GAIN-CAL extensions belonging to that revolution. This single revolution table is found in BASE_REP/scw/SSSSSSSS/rev.000/idx and is the table used by j-cor-gain to process science window data from that revolution. Like the other index tables, this is created by the Revolution Pipeline, not by j-calib-gain-fitting.

Inputs:
JMXi-IMOD-GRP (p. 239)
JMXi-CALB-MOD (p. 233) In the instrument model group
JMXi-FBDS-MOD (p. 236) In the instrument model group
JMXi-FRSS-CRW-IDX (p. 237)
JMXi-FRSS-CRW (p. 237) In the input index table

Outputs:
JMXi-GAIN-CAL-IDX (p. 238)

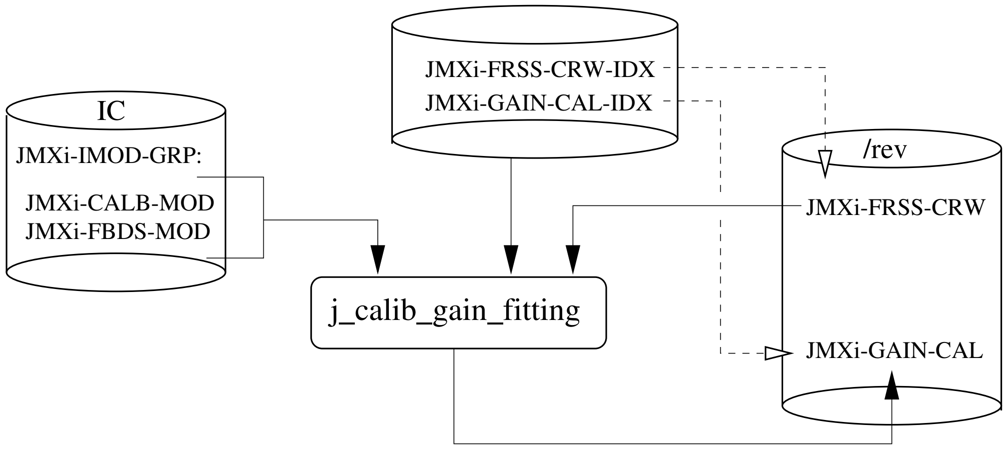

Figure 5.6: j-calib-gain-fitting

JMXi-GAIN-CAL (p. 238) In the output index table

Parameters:

inCAL p. 46	DOL	Input calibration spectrum index table DOL JMXi-FRSS-CRW-IDX
outCAL p. 48	DOL	Output extension index table DOL JMXi-GAIN-CAL-IDX
instMod p. 46	DOL	Instrument model group JMXi-IMOD-GRP
jemxNum p. 46	int.	Number of JEMX instrument being used (1 or 2)
clobber p. 42	bool.	Clobber existing data in output data structures?
chatter p. 42	int.	Verbosity level of logfile output (0,1,2,3,4,5)

Resources: Time: Processing 11 sets of 4 calibration spectra takes 4.62 seconds using 85% of CPU. Total memory requirements for processing one set of 4 spectra is less than 30K for data and below 1.4M for code

Processing:

Initialise executable
Read in parameter file:
    Make data structure prefix for the given instrument
    Make sure all required data structures exist
    Open all the input and output extensions:

```
        JMXi-FRSS-CRW-IDX, JMXi-GAIN-CAL-IDX, JMXi-IMOD-GRP,
        JMXi-CALB-MOD, JMXi-FBDS-MOD, JMXi-
Read in the calibration ADC-binning table and PI channel limits:
    JMXi-CALB-MOD, JMXi-FBDS-MOD
    Find average ADC channel value of each calibration bin
Initialise I/O processes:
    Find and validate all the I/O data structures
    Customise output index fits extensions
Loop over members of input and output index tables:
    Open corresponding member each of input and output index:
        JMXi-FRSS-CRW and JMXi-GAIN-CAL
    Customise output extension
    Read in one set of four raw calibration spectra:
        SSC_BEG, COBT, NUMPART, COUNTS_1, COUNTS_2, COUNTS_3, COUNTS_4
    If we have all packets for 4 spectra continue with fitting
    If number of counts in a spectrum is too small, skip this spectrum
        else go on to next spectral set in the index
    Find if FRSS is Cd or Fe
    Make initial guess at fitting parameter values:
        For Cd:
            width of single Cd line
            intensity of strongest Cd line
            gain offset of 4095 ADC channels
            gain linear dependance (slope) of 4095 ADC channels
        For Fe sources in JEM-X1 parameters are:
            Width of Fe single line
            Intensity of Fe strongest single line
            gain offset of 4095 ADC channels
            gain linear dependance (slope) of 4095 ADC channels
    Use estimated start values and Levenburg-Marquardt least-squares
        minimisation to find best-fit values of fitting parameters
    Save fitting results to output data structures:
        Calculate fitted spectra and then residuals
        Save residuals to JMXi-GAIN-CAL: RESID_1, RESID_2 etc.
        Save fitting results for each anode as keywords in JMXi-GAIN-CAL:
            Gain offset of channel 0: G_ZERO_1, G_ZERO_2 etc.
            Gain slope of channels: G_CAL_1, G_CAL_2 etc.
            Energy resolution for Cd or Fe peak: RES_22_1, RES_22_1 etc.
            Intensity of Cd or Fe line: INT_CD_1, INT_CD_2
    Update index JMXi-GAIN-CAL-IDX to add all new keywords
End loop over members of I/O index tables
```

Close all data structures, tidy up and exit

Design remarks:

- This executable does not run in server mode since it involves relatively small amounts of data (one set of four calibration spectra every 256 seconds) and processing is fast.
- Groups of 8 calibration spectral sets are input via the input index group so that Revolution Pipeline doesn't need to run this program every 256 seconds (4.2 minutes).
- The OBTs of the spectral sets are written directly into the OB_TIME column of JMXi-GAIN-CAL-IDX by j-prp-frss-obt (p. 87) so strict one-to-one correspondence of input and output children must be maintained.
- Gaussian lineshapes have been used to model the unresolved Fe and Cd doublets, which should be sufficient given the number of data points in each doublet and detector resolution. Fitted gaussian widths are output as FWHM values (relative energy resolution).
- The iterative fitting process has been designed to be robust in the presence of noise in the calibration spectra
- JMXi-GAIN-CAL-IDX (p. 238) master index table must be built up by the pipeline script, from all the small versions of JMXi-GAIN-CAL-IDX made by j-calib-gain-fitting. It is the master version, or a single revolution version of this index that is used by j-cor-gain as the gain history table
- The gain offset (GZERO_1, GZERO_2...etc.) for each source is currently zero so long as the electronic calibration indicates that the amplifier offset is being correctly substracted onboard. However if this is not the case we have the capability to take this into account by using non-zero values of GZERO_n.
- The program has had to be expanded to include the use of Fe-55 sources in JEM-X1. This means that fitting parameter names do double duty for both the Cd sources and the Fe sources. This does not in the least change the meanings of the main fitting parameters, only their values.
- Version 7.2 was improved by lowering the number of counts considered too small to fit and improving the first-guess estimates of the fitting parameters.
- As the intensity of the calibration sources decreases with time, and the length of the mission is extended, the program may need to start reading several spectra at a time and adding

them together to get useable spectra for the weakest sources. Alternatively, the very weakest spectra will simply not be used either for fitting or gain correction. This functionality will be implemented in OSA 6.0 at the earliest.

Parent components:
Revolution File Pipeline (p. 30)

Subordinate components: Functions of the executable:
JCalibGuess
JCalibModel
JCalibMatrixSolve
JCalibRowSwap
JCalibChiSq
JCalibMakeMatrix
JCalibResiduals
JGainFittingParams
JGainFitting-Work
JCalibGetSpectra
JCalibGetBins
JCalibLMIteration
JCalibSave
JCalibPIBinning

Dependencies from other components:

- Requires that calibration spectra have been unpacked from HK telemetry by PreProcessing
- Valid instrument model group must be accessible from IC
- Revolution file pipeline must have grouped sets of calibration data together in index table JMXi-FRSS-CRW-IDX, and created a corresponding empty output index table JMXi-GAIN-CAL-IDX pointing to empty data structures JMXi-GAIN-CAL
- j-prp-frss-obt (p. 87) must have calculated OBTs for the indexed spectral sets and put them as keywords in each empty JMXi-GAIN-CAL extension header
- Revolution file pipeline must add small version of JMXi-GAIN-CAL-IDX to the master gain history table JMXi-GAIN-CAL-IDX after each run of this component

ISDC context: Automatic Calibration Analysis in the Revolution File Pipeline

Delivery date: Version 7.2 delivered 2005/12/7

5.6.8 j-calib-adc

Type: Executable **Responsible:** DNSC (C.A. Oxborrow)

Purpose: Not forseen in SRD or URD

Function: To fit a linear model to the electronic calibration data JMXi-ECAL-CRW and determine whether the ADC/amplifier chains on the detector are still working linearly and over their entire range. Alerts produced only. Alerts can be switched off by setting all alert level parameters to zero. Offline analysis of this data is done with j-ical-ecal (p. **??**). For results of offline trend analysis see the JEM-X Forum website.

Inputs:
JMXi-ECAL-CRW (p. 236) Stored by revolution
JMXi-ECAL-CPR (p. 235) Stored by revolution
JMXi-IMOD-GRP (p. 239) In Instrument Characteristics
JMXi-ECAL-MOD (p. 236) In the Instrument Model Group

Outputs: Alerts and logfile only

Parameters:

ecalRaw p. 43	DOL of raw electronic calibration data: JMXi-ECAL-CRW
ecalPrp p. 43	DOL of prepared OBTs fro calibration events: JMXi-ECAL-CPR
instMod p. 46	DOL of the current instrument model group
alrtOffset p. 40	Alert level for amps with changed offsets
alrtSlope p. 40	Alert level for amps with changed slopes
alrtBoth p. 40	Alert level for both changed slopes and offsets
alrtOther p. 40	Alert level for other problems with amps
alrtZero p. 40	Alert level for amps with zero slope: dead amps
jemxNum p. 46	Number of the instrument being analysed: 1 or 2
chatter p. 42	Verbosity of log file output

Parent components:
Revolution File Pipeline (p. 30)

Resources: To run 5 typical complete sets of data, each one for all four anodes, with 160 events per anode, and maximum verbosity, takes 0.7 seconds of CPU time, using 65% percent of the CPU on a Sun Solaris machine running Solaris 8.
Memory requirements are: 1.2Mbytes for code and about 240Kbytes of data.

Processing:

Initialise executable
Read in the IRAF parameter file:
    Read in all the parameters from j_calib_adc.par
    Find the instrument string 'JMX1' or 'JMX2'

- Find pointer to the JMXi-ECAL-CRW data structure
- Find the revolution number keyword from JMXi-ECAL-CRW
- Find pointer to the JMXi-ECAL-CPR data structure
- Find pointer to the JXMi-ECAL-MOD data structure
- Find pointer to the JMXi-IMOD-MOD data structure

Read in the baseline ADC models from JMXi-ECAL-MOD:

- Check there's enough rows in the data structure
- Read in the column of amplifier names
  - Assign each amplifier its correct place in the model arrays
  - Check that every amplifier is present; or error
  - Check that no amplifiers are listed more than once; or error
- Read in the data columns OFFSET, DELTA_OFF, SLOPE, DELTA_SLP, CHISQ
- Sort data columns into model arrays

Do the work to make linear fits to all amplifier data

- Read in all rows of the raw data columns
- Make data input buffers for each data column
- Read in raw data columns:
  - SSC, CAL_AMP_EVT, ANODE_SWITCH, SLOW_ANODE FAST_ANODE, VETO
    - BACK_PLANE, CATHODE
- Check maximum number of calibration events per anode
- Make data arrays for calibration events per amplifier per anode
- Sort input data rows according to amplifier and anode segment :
  - Filter out filler events and count them
  - Read marker events to fill amplitude array
  - Sort real events into correct arrays
- If too many filler events, produce error
- Do linear regression on 20 Backplane amplifiers
  - Data for all four anodes fitted together:
    - no expected variation between anode segments for backplane
  - Null-pulse output average and RMS variation found
- Do linear regression on 11 cathode amplifiers
  - Data for all four anodes fitted together:
    - no expected variation between anode segments for cathode
  - Null-pulse output average and RMS variation found
- Do linear regression on fast anodes for 4 anode segments
  - Data for each anode segment fitted separately
  - Null-pulse output average and RMS variation found
- Do linear regression on slow anodes for 4 anode segments
  - Data for each anode segment fitted separately
  - Null-pulse output average and RMS variation found
- Do linear regression on vetoes for 4 anode segments

Data for each anode segment fitted separately
Null-pulse output average and RMS variation found
Compare model offsets and slopes with found ones for all amps:
If logfile verbosity is big, report all found/model parameters
Count number of amplifiers with very small slopes (nDead)
Count amplifiers with significantly different slope (nBadSlope)
Count amplifiers with different offsets (nBadOffset)
Count amplifiers large RMS variation in their data (nBadAmps)
Count amplifiers different null offsets or large RMS variation
of null offsets (nNoisyAmps)
If nDead>0 produce 'dead amplifiers' alert
If nBadSlope>0 produce 'bad slopes' alert
If nBadOffset>0 produce 'bad offsets' alert
If nBadAmps>0 produce 'amps with bad linearity' alert
If nNoisyAmps>0 produce 'noisy amps' alert
End of work doing linear fitting and analysing results
Close input group and do tidy up

Design remarks:
1) This executable produces only two sorts of output:

- Logfile entries, including a complete printout of found and model offsets and slopes if verbosity level is 3 or above.
- Alerts when deviations from model/previous behaviour are detected.

2) Alert production can be switched off for the individual types of alert by setting the relevant input parameter alrt*** to zero.
3) The executable does not read the OBT data for the electronic calibration. It is assumed that the procedure is performed once per revoluation after coming out of the radiation belts and so each set of results are identified by revolution number only.
4) The 'null-pulse' data introduced between each real pulse event is averaged for each amplifier and its RMS deviation found. If either of these two values are different from model values a 'noisy amp' alert is produced, indicating that the null offset of the amplifier is drifting and/or fluctuating. These conditions could indicate either temperature sensitivity of the amplifier or damage producing random noise in the offset.
5) A large RMS variation in the non-null data of an amplifier is probably an indication that the behaviour deviates more than expected from a straight line. This has been seen in some amplifiers and is reflected in JMXi-ECAL-MOD. Changes in the RMS slope variation could

indicate a worsening of this condition—however, it could also be caused by noise in the data cause by as yet unknown amplfier damage. These different possibilities must be checked offline.
6) The last update of the component was to accomodate smaller amplitude step sizes used in the new ECAL procedure introduced after launch which concentrates on the lower end of each amplifier slope. The new procedure picks up low-end non-linearity more clearly than the old one, and the non-linearity limits in JMXi-ECAL-MOD have had to be adjusted upwards accordingly.
7) Alerts produced by this component are:

5020 Dead amplifier (zero slope)

5021 Amplifier with poor linearity

5022 Amplifier with unexpected slope

5023 Amplifier with unexpected offset/null pulse value

5024 Noisy amplifier

5025 ECAL data not useable; no linear slope found

Parent components: Revolution File Pipeline

Subordinate components:
JCalibADCParams
JCalibADCWork
JCalibADCModel
JCalibADCFit

Dependencies from other components: Assumes PreProcessing has unpacked electronic calibration data in JMXi-ECAL-CRW, and that Data Preparation has calculated the OBTs at the centre of each integration time and put these in JMXi-ECAL-CPR

ISDC context: Data Preparation in Revolution File Pipeline

Delivery date: Version 2.2 delivered 2004/04/19

5.7 j-correction

Type: Analysis Script **Responsible:** ESAC (P. Kretschmar)and Valencia University (S. Martínez-Núñez)

Purpose: 18.10 18.11 18.13

Function: Correct JEM-X science data for instrumental fingerprints and the time variable gain.

Inputs:
JMXi-FULL-RAW (p. 238) (event data)
JMXi-FULL-PRP (p. 237)
JMXi-SPTI-RAW (p. 244)
JMXi-SPTI-PRP (p. 244)
JMXi-REST-RAW (p. 242)
JMXi-REST-PRP (p. 241)
JMXi-SPEC-RAW (p. 244) (spectrum mode data)
JMXi-SPEC-PRP (p. 244)
JMXi-FULL-ALL (p. 237) (event data, new scheme)
JMXi-SPTI-ALL (p. 244)
JMXi-REST-ALL (p. 241)
JMXi-SPEC-ALL (p. 243) (spectrum mode data, new scheme)
JMXi-IMOD-GRP (p. 239) (instrument description)
JMXi-GAIN-CAL-IDX (p. 238)

Outputs:
JMXi-FULL-COR (p. 237)
JMXi-SPTI-COR (p. 244)
JMXi-REST-COR (p. 241)
JMXi-SPEC-COR (p. 243)

Parameters:

swgDOL p. 53	DOL of Science Window Group to be analyzed
IC-Group p. 46	DOL of Instrument Characteristics master group
IC-Alias p. 46	Selection alias for Instrument Characteristics
jemxNum p. 46	JEM-X detector number
instMod p. 46	DOL of JEM-X Instrument Model Group (empty=find in IC)
osimData p. 48	Are we using data produced by OSim?
chatter p. 42	Verbosity level; 2 = normal
clobber p. 42	Clobber existing output data?
subDir p. 53	Sub directory below the position of the SWG for results
gainHist p. 45	DOL of Index Group with gain change history
gainModel p. 45	Gain history smoothing model option
j-cor-both-fullRaw p. 45	DOL of FULL imaging raw data (override)
j-cor-both-fullCor p. 44	DOL of FULL imaging corrected data (override)
j-cor-both-restRaw p. 50	DOL of REST imaging raw data (override)
j-cor-both-restCor p. 50	DOL of REST imaging corrected data (override)
j-cor-gain-fullPrp p. 44	DOL of FULL imaging prepared data (override)
j-cor-gain-restPrp p. 50	DOL of REST imaging prepared data (override)
j-cor-gain-sptiRaw p. 52	DOL of Spectral-Timing raw data (override)
j-cor-gain-sptiPrp p. 52	DOL of Spectral-Timing prepared data (override)
j-cor-gain-sptiCor p. 52	DOL of Spectral-Timing corrected data (override)
j-cor-gain-specRaw p. 52	DOL of Spectrum Mode raw data (override)
j-cor-gain-specPrp p. 52	DOL of Spectrum Mode prepared data (override)
j-cor-gain-specCor p. 52	DOL of Spectrum Mode corrected data (override)

Note: parameters with a prefix of "j_cor_both_" or "j_cor_gain_" are override parameters for data normally found in the Science Window Group, which are passed on to both executables or to **j_cor_gain** respectively. Normally they are hidden and set to empty (no effect).

Resources: See called executables. The script itself uses only negligible resources.

Processing:

Read in parameters
Get DOLs to the appropriate version of JMXi-IMOD-GRP
Call j-cor-gain to correct pulse height information for all data modes (PHA→PI)
Call j-cor-position to correct positions for FULL and REST events

Design remarks:
In the 'new' scheme of 2004 this component can be called both by the Science Window Pipeline (default) behaviour and by the science analysis scripts again, in case the user wants to override the results produced in the pipeline.

Parent components:
jemx-scw-analysis (p. 70)
Science Window Pipeline

Subordinate components:
j-cor-gain (p. 112)
j-cor-position (p. 121)

Dependencies from other components: Raw and prepared data must exist in the input swg-DOL.
The gain history and the instrument description must be accessible.

ISDC context: Science Analysis

Delivery date: v 4.5.2 23/05/2005

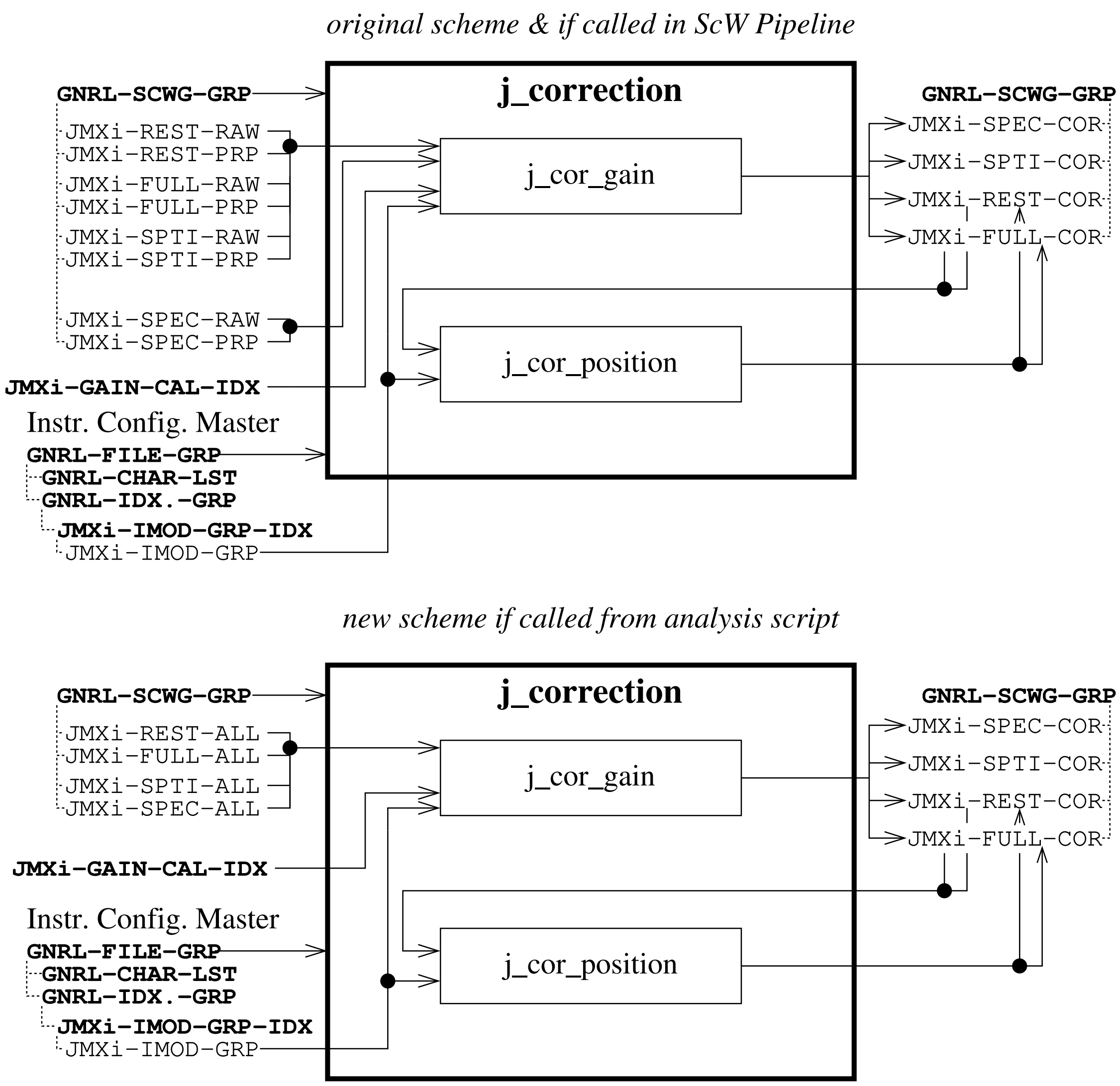

Figure 5.7: j-correction

5.7.1 j-cor-gain

Type: executable **Responsible:** DNSC: C.A. Oxborrow

Purpose: 18.10.1 , 18.10.2 , 18.10.4 , 18.10.7 , 18.10.8 , 18.11.1 , 18.11.2 , 18.13.1 , 18.13.2

Function: Correct raw events and spectra for variations in detector gain as determined by in-flight calibration spectra and spatial gain variation tables.

Interfaces: 1) Executable takes raw event, spectral and OBT data as input from whichever data-taking modes are available for the given SWG (FULL, REST, SPTI, SPEC). All raw data structures can be overriden with versions taken from outside the specified SWG.
2) The energy binning tables for PHA and PI channels are all taken from the instrument model group JMXi-IMOD-GRP.
3) The index table containing the gain fitting parameters is produced by the j-data-preparation pipeline script using the JMXi-GAIN-CAL-IDX data structures output by j-calib-gain-fitting.
4) The table of expected spatial variation in gain is also in the instrument model group, in extension JMXi-SPAG-MOD. The second slice of this table contains spatial gain variations determined for individual anodes after the general SPAG correction is performed (dead anode SPAG)
5) A time correlation table must be provided in the Science Window group so that each event or spectrum can be evaluated for goodness of time correlation. This is AUXL-TCOR-HIS—the DAL3AUX functions find the most appropriate version.
6) The detector model table with listing of the anode segment boundaries is also to be found in the instrument model group, JMXi-DETE-MOD.
7) All output is stored in corrected event data structures JMXi-$* * **$-COR

Inputs:
JMXi-FULL-RAW (p. 238) In input SWG or override DOL
JMXi-REST-RAW (p. 242) In input SWG or override DOL
JMXi-SPTI-RAW (p. 244) In input SWG or override DOL
JMXi-SPEC-RAW (p. 244) In input SWG or override DOL
JMXi-FULL-PRP (p. 237) In input SWG or override DOL
JMXi-REST-PRP (p. 241) In input SWG or override DOL
JMXi-SPTI-PRP (p. 244) In input SWG or override DOL
JMXi-SPEC-PRP (p. 244) In input SWG or override DOL
JMXi-FULL-ALL (p. 237) In input SWG
JMXi-REST-ALL (p. 241) In input SWG
JMXi-SPTI-ALL (p. 244) In input SWG
JMXi-SPEC-ALL (p. 243) In input SWG
AUXL-TCOR-HIS (p. 248) In input SWG
AUXL-CRST-REF (p. 247) In input SWG
JMXi-IMOD-GRP (p. 239)

JMXi-FULB-MOD (p. 237) In instrument model group
JMXi-RESB-MOD (p. 241) In instrument model group
JMXi-SPCB-MOD (p. 243) In instrument model group
JMXi-FBDS-MOD (p. 236) In instrument model group
JMXi-RBDS-MOD (p. 241) In instrument model group
JMXi-SBDS-MOD (p. 242) In instrument model group
JMXi-SPAG-MOD (p. 243) In instrument model group
JMXi-DETE-MOD (p. 235) In instrument model group
JMXi-GAIN-CAL-IDX (p. 238) Output from Revolution file pipeline

Outputs:
JMXi-FULL-COR (p. 237) In output SWG or override DOL
JMXi-REST-COR (p. 241) In output SWG or override DOL
JMXi-SPTI-COR (p. 244) In output SWG or override DOL
JMXi-SPEC-COR (p. 243) In output SWG or override DOL

Parameters:
INPUT GROUPS:
inSWG p. 46 : DOL of input SWG
outSWG p. 49 : DOL of output SWG
instMod p. 46 : DOL of instrument model group JMXi-IMOD-GRP
OTHER INPUTS:
gainHist p. 45 : DOL of gain history table JMXi-GAIN-CAL-IDX
jemxNum p. 46 : Number of the JEM-X instrument: 1 or 2
gainModel p. 45 : Option number for the time-smoothing model
clobber p. 42 : Clobber pre-existing data in extensions?
chatter p. 42 : log file verbosity parameter
OVERRIDES for SWG data structures:
fullRaw p. 44 : Override DOL of JMXi-FULL-RAW
fullPrp p. 44 : Override DOL of JMXi-FULL-PRP
fullCor p. 44 : Override DOL of JXMi-FULL-COR
restRaw p. 50 : Override DOL of JMXi-REST-RAW
restPrp p. 50 : Override DOL of JMXi-REST-PRP
restCor p. 50 : Override DOL of JMXi-REST-COR
sptiRaw p. 52 : Override DOL of JMXi-SPTI-RAW
sptiPrp p. 52 : Override DOL of JMXi-SPTI-PRP
sptiCor p. 52 : Override DOL of JMXi-SPTI-COR
specPrp p. 52 : overide DOL of JMXi-SPEC-PRP
specRaw p. 52 : override DOL of JMXi-SPEC-RAW
specCor p. 52 : override DOL of JMXi-SPEC-COR

Resources: Using 77% of CPU takes 2.37 seconds of processor time to handle 18,500 FULL events + 585,000 REST events, 2600 SPTI events and 20,000 SPEC spectra, over 600,000 'events' in all.
Code size is just less than 1.4M and data memory requirements are limited to a maximum of 1.5M for event data and 800K for SPEC data since events are processed in blocks of 10,000 at a time and spectra are handled 1000 at a time.
Processing is slowed down by using the ISDC-created gain history table in its -IDX format. Offline processing can be performed much quicker by editing a private version of the table to remove the first 5 columns of the table (indexing information never used by j-cor-gain) and changing the header keywords EXTNAME (change to JMXi-GAIN-CAL); BASETYPE (becomes DAL_TABLE); IDXMEMBR and GRPNAME should be deleted. The hidden parameter gainHist in the science analysis GUI should be changed to point to the private gain history table and the end of the DOL should read [JMXi-GAIN-CAL,1] instead of [GROUPING,1]. This problem arises because DAL tries to check that all the indexed files are present, and in some cases attempts to open them all (about 100 files!).

Processing:

Initialise executable
Read parameter file
    Find instrument number
    Open SWG
    Check which raw data structures are present
    Find degree of log verbosity and smoothing model options
If there is either FULL, REST, SPEC or SPTI raw data continue
    else give log message and exit
Read the spatial gain variation table JMXi-SPAG-MOD:
    1st (256×256) array:
        percent peak position variation determined by pre-flight calibration
    2nd (256×256) array:
        percent peak position variation for each anode strip
    Read in the reference Xe, Cd and Fe channels (5 in all) from the SPAG header
    Find both variation fractions averaged over each anode segment
    If problems arise with reading/averaging, all variations set to 0.0
Read in the gain history:
    Columns: REVOL, OBT_ACQ, PKCHAN_1, PKCHAN_2 etc. from JMXi-GAIN-CAL-IDX
    Read in only gain history values for current revolution
    If all gain values for one time are zero or rubbish discard them
    Calculate the Xe-normalised line position of each source based on the
        reference channels, for each 4 minute period

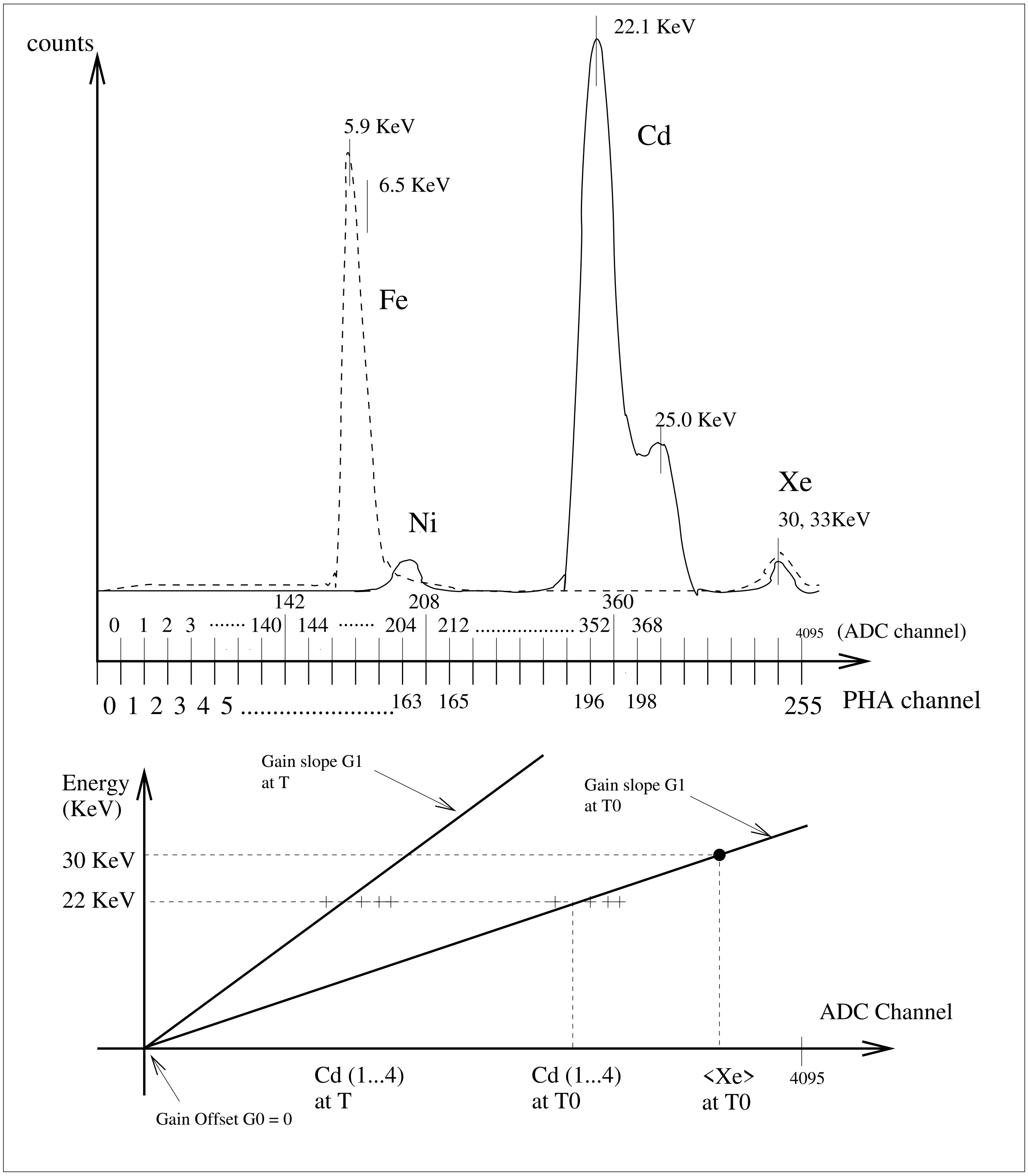

Figure 5.8: Determining the keV-per-ADC-channel gain slope

If gainModel= 2
  Split gain history of revolution into time zones separated by switch-ons
    Find all significant changes in time and gain for each source
    If 3 or more sources find a break at the same place this is start of
      a new time zone, having its own decay, constant and slope
  Determine whether there's a initial partial HV switch-on
    If there's only one point take its value as the gain value
    Otherwise, fit a line to these first data points
  For each of the remaining time zones, fit a decay curve and a short
    harmonic series to the data within the zone
  Store these gain smoothing model parameters for event/spectra correction
Loop over the three event data modes: FULL, REST and SPTI
  If there is a -RAW data extension for the mode continue
    else go on to next mode
  Check there are actually events in raw data structure
    If no events go on to next type of data
  Open or create JMXi-$* * **$-COR:
    stamp it and copy over necessary keywords
  Read in corrected PHA channels for the mode, JMXi-***B-MOD
  Read in PI energy bins for the mode JMXi-*BDS-MOD
  Loop over events in -RAW data extenstion, 10,000 events at a time
    Read in positions RAWX, RAWY of FULL and REST events
    Read in PHA energy channels for (batches of) events
    Read in OB_TIMEs for events from JMXi-$* * **$-PRP
    Convert OBTs to IJD: get goodness of time correlation of events
    Determinine if we have imaging or non-imaging mode data
    Correct one event at a time
      For imaging events, find anode event belongs to
      Set event zero and gain to constant values according to anode
      If using model 0: replace default gain parameters with
          interpolated values according to the event's OBT
      If event falls outside gain history table:
        use default values and raise status flag
      If using model 2: find PI value of each event, finding gain at
        event OBT by using the smoothing model fitting parameters
        For non-imaging modes find average current gain parameters
          over the four anode segments
      Correct gain calibration parameters for spatial variation
        For imaging modes use RAWX and RAWY pixel values
        If imaging events fall on bad pixel raise status flag
        For non-imaging modes use SPAG values averaged over detector

```
            Assign event to correct PI bin given in JMXi-*BDS-MOD
            If event gain parameters fall outside expect bounds raise status flag
            If event time correlation is bad, raise status flag
        End loop over individual events
        Write out PI values and STATUS flag to -COR extension
    End loop of chunks of 10,000 events
End loop over the data modes FULL, REST, SPTI
If there is a SPEC extension in SWG:
    Check there are actually events in raw data structure
        If no events go on to tidy up
    Open/create output extension JMXi-SPEC-COR
        Stamp output extension and copy over keywords
    Read in corrected PHA energy bins, JMXi-SPCB-MOD
    Read in PI energy bins JMXi-SBDS-MOD
    Loop over batches of 1,000 SPEC spectra
        Read in SPECTRUM from JMXi-SPEC-RAW for batch of spectra
        Read in OB_TIMEs from JMXi-SPEC-PRP for batch spectra
        Convert OBTs to IJD to get time correlation of spectra
        Loop over individual spectra in the batch
            Set the gain parameters to averages of the default values
                Average is over the four anodes.
                These default values are also Model 1
            If using model 0:
                By linear interpolation find gain parameters at given OBT
                If spectra fall outside gain history table:
                    Use default gain parameters and raise status flag
                Average the found parameters over the four anode segments
            Find average gain calibration parameters over detector
            Correct gain calibration parameters for average spatial variation
            If average gain parameters out of limits, raise status flag
            Use gain parameters to find KeV values for telemetry (PHA) bins at the given OBT
            Rebin spectrum into PI bins
            Check that all spectral counts are accounted for
            If IJD is NAN raise bad-time-correlation status flag
        End loop over individual spectra
    Write out rebinned spectra and STATUS flag to JMXi-SPEC-COR extension
    End loop over batches of spectra
End SPEC processing
Close SWG, tidy up
If status flag is not ISDC_OK printout DAL error message stack
Exit
```

Design remarks:

- This executable incorporates 2 previous executables: j_cor_gain_events and j_cor_gain_spectral
- All events and spectra are treated in batches of 10,000 or 1000, so no need for huge arrays of event data if there's 500,000 events in a raw data extension
- Conversion of OBTs to IJDs is used only to check time correlation and accordingly set the event evaluation keyword STATUS
- Average gain calibration values are used for non-imaging data without any correction for spatial gain variation of any sort
- Individual events are assigned to PI bins randomly but weighted according to the degree of overlap between the event's raw PHA channel and the corresponding PI channels in the same energy range. The DAL3 random number generators are used to ensure uniformity of results across operating systems.
- Spatial variation of gain parameters is given as percent difference in Xe peak ADC channel and the value measured for the whole detector
- Gain zero, or offset, means the theoretical energy in KeV of an event appearing in ADC channel 0. In practice this channel is used for electronic noise and no real events appear in channel 0. This value is stored as G_ZERO_n for anode n, in JMXi-GAIN-CAL-IDX, for each FRSS acquisition time ACQ_OBT.
- Gain zero is typically zero because onboard software subtracts the pre-flight amplifier offsets from each measurement. If the electronic calibration procedure indicates that the amplifier offsets have changed then either the onboard software tables must be updated, or we can use non-zero values of G_ZERO_n.
- The gain parameters of every event are set to standard constant values given in j-cor-gain.h. These values are retained and used if any problems arise in calculating more accurate values with other models. The status flag for 'poor gain determination' is raised for any event when the default values are used.
- Currently there are three gain models to choose from:

Model 0 Linear interpolation in time between neighbouring values of gain parameters on the gain history table. This is used as backup method if default gain decay model fails.

Model 1 Default values for each parameter and each anode segment of each unit are stored in j-cor-gain.h. They are used as emergency gain values if the gain history table is incomplete in some way, or if there are problems performing the linear interpolation or decay model.

Model 2 This is the default and preferred model. A curve is fitted to the detector gain of a whole revolution and gain for individual event times is calculated from this curve. The model consists of a very brief linear fit corresponding to the first, partial, high voltage switch on. Then there is a series of longer decays each modified by a harmonic series to handle slow temperature and background changes. Up to four of these long decays are available, but usually only one or two are used. The first long decay models gain behaviour after the first full HV switch on. Subsequent decays model the behaviour after emergency switch off/on. This model is preferred because is smooths out statistical noise and local gain glitches.

- Conversion of OBTs to IJD is used to return the goodness of the time correlation, which is used to raise a status flag if the correlation is poor (IJD=NAN)
- Since version 5.3 a single-revolution version of the gain history table, JMXi-GAIN-CAL-IDX, is used to speed up read in, and the program sorts the input which need not necessarily be time-ordered.
- Each event or spectrum is tagged with a value in the 'STATUS' column of the output extensions. See section 4.3
- An ISDC standard random number generator is used for energy bin assignments to ensure consistent results across different platforms
- Program can also handle input from -ALL event lists as well as -RAW/-PRP extensions
- Figure 5.7.1 shows how the KeV per channel, gain slope is determined, using the relative channel positions of the Xe and Cd or Fe peaks at a reference time to anchor the gain measured at the calibration spectrum points (Cd/Fe peaks) to the overall gain of the detector plate (Xe peak). This is necessary to counteract differential gain aging of calibration areas and the plate as a whole.
- Version 6.4 of this component can handle up to 10 time zones, one of which may be the initial partial switch on at the beginning of the revolution. Time zones are determined by time/gain breaks that are the same for 3 out of 4 sources.
- Major additions are expected to this component for OSA 6.0 so that it can use either gain history tables calculated automatically by j-calib-gain-fitting, or tables determined by offline analysis. Also gain correction verification output will be introduced.

Parent components:
j-correction (p. 108)

Subordinate components: Functions of the executable:
JCorGain-Params
JCorGain-Work
JCorGainGetBins
JCorGainAllEvents
JCorGainCorrectEvents
JCorGetSpag
JCorGainAllSpec
JCorGainCorrectSpec
JCorGainGetHistory
JCorGainDecayModel
JCorGainColSwap
JCorGainRowSwap
JCorGainSolve
JCorGainSmoothing
JCorGainDecayChisq
JCorModelGainCalc
JCorGain-Sort
JCorGain-Partition

Dependencies from other components:

- The Revolution File Pipeline must run first to provide the master gain history table JMXi-GAIN-CAL-IDX, including OBTs, from j_calib_gain_fitting and j_prp_frss_obt
- The Science Window Pipeline must have run first to provide OBTs for all the raw scientific data events and spectra: j_prp_evts_obt, j_prp_rate_obt, j_prp_spec_obt.
- The auxilliary time correlation data must be available in the SWG though it is stored by revolution

ISDC context: Scw QLA pipeline in the NRT data processing, both under Automatic Operational Processing. Also, Data Correction and Binning of the consolidated data processing. JEM-X Science Analysis pipeline (OSA).

Delivery date: Version 6.4 delivered 2005/8/4

5.7.2 j-cor-position

Type: Executable **Responsible:** DNSC: C.A. Oxborrow

Purpose: 18.10.3

Function: To correct for small-scale nonlinearities due to variation in the positions of the microstips for both imaging formats and other sources of position errors, such as distortion caused by field ring fields and clumping/rarefication at anode segment boundaries. This is a position (X/Y) correction for imaging modes (FULL and REST) only. Each imaging event is also flagged with a STATUS value reflecting its position on the detector.

Interfaces: 1) The raw event data to be corrected comes from the input SWG
2) The pre-flight and in-flight position correction models come from instrument model group JMXi-IMOD-GRP, JMXi-CORX-MOD and JMXi-CORY-MOD.
3) The output correction data structure is either in the output SWG or in the override data structure
4) Detector gain corrections should be done before this correction since gain corrections are determined by the position where events are actually detected, not by the position where they should have been detected.
5)Position corrections determined by j-ical-collimator (p. **??**) can be used later in the mission, and will appear in the JMXi-IMOD-GRP instead of the pre-flight corrections if this is deemed necessary by the Instrument team, so a switch to in-flight corrections does not require a change of input specifications or interfaces.

Inputs:
JMXi-FULL-RAW (p. 238) In input SWG or override DOL
JMXi-REST-RAW (p. 242) In input SWG or override DOL
JMXi-FULL-ALL (p. 237) In input SWG
JMXi-REST-ALL (p. 241) In input SWG
JMXi-IMOD-GRP (p. 239)
JMXi-CORX-MOD (p. 234) In instrument model group
JMXi-CORY-MOD (p. 234) In instrument model group
JMXi-DETE-MOD (p. 235) In instrument model group

Outputs:
JMXi-FULL-COR (p. 237) In output SWG or override data structure
JMXi-REST-COR (p. 241) In output SWG or override data structure

Parameters:

GROUPS:

inSWG p. 46	DOL	input SWG with the raw data, ISDC level RAW
outSWG p. 49	DOL	output SWG , ISDC level COR

instMod p. 46	DOL	Instrument model group JMXi-IMOD-GRP
OTHER INPUTS:		
jemxNum p. 46	int.	Number of the JEM-X instrument, 1 or 2
clobber p. 42	bool.	Overwrite pre-existing data in extensions?
chatter p. 42	int.	Log file verbosity parameter (0,1,2,3,4,5)
OVERRIDES:		
fullRaw p. 44	DOL	Override for JMXi-FULL-RAW data structure
restRaw p. 50	DOL	Override for JMXi-REST-RAW data structure
fullCor p. 44	DOL	Override for JMXi-FULL-COR data structure
restCor p. 50	DOL	Override for JMXi-REST-COR data structure

Resources: Using 61% of the CPU, requires 1.52 seconds of processing time to handle over 600,000 FULL and REST events. Processing time should scale linearly to the number of events. The code takes up less than 1.4M of memory. Data memory requirements are limited to about 900K by the events being handled 10,000 at a time.

Processing:

Initialise executable
Read in parameter file
    Find instrument number and prefix string
    Open SWG and find pointers to all input data structures
Read in position correction models:
        JMXi-CORX-MOD is 256×256 array giving corrected absolute
            X coordinate of each detector pixel
        JMXi-CORY-MOD is 256×256 array giving corrected absolute
            Y coordinate of each detector pixel
Read in detector model JMXi-DETE-MOD from JMXi-IMOD-GRP
Loop over the two imaging data modes: FULL and REST
    If there are -RAW events of this type, open/create -COR output extension
        else move on to next mode
    Loop over batches of 10,000 events
        Read in RAWX and RAWY position data from JMXi-****-RAW
        Read in current value of STATUS flag from JMXi-****-COR
        Determine whether there are any active hotspots on the plate
            Find the average event coverage per pixel
            Make a map of any exceptionally bright pixels
            Make a hotspot map of clusters of bright pixels
        Loop over individual events

Find CORX/CORY values for event position to its position
If pixel is marked bad (added 1000.00 to position), raise status flag
End loop over individual events
Write corrected position data to JMXi-****-COR: DETX, DETY
Check event's detector pixel value from JMXi-DETE-MOD and
flag the following problems in STATUS output column:
Outside active detector area, bad anode segment,
Dead anodes, dead anode neighbour , or from
calibration spectrum region
If event falls on a hotspot, according to the map, flag it
End Loop over batches of events
End Loop over imaging modes
Close SWG, tidy up and exit

Design remarks:

- The position model arrays JMXi-CORX-MOD and JMXi-CORY-MOD give corrected absolute coordinates of each pixel on the detector, expressed in mm from detector centre. Values are already corrected for microstrip plate irregularities and detector field-induced straying of events. These corrections are determined by pre-flight calibration and offline analysis during the mission.
- Currently, position corrections are not assumed to depend on energy. Nor are they expected to change rapidly with time. We can however expect these corrections to drift with time.
- The first level of position corrections is performed onboard and any corrections factored into CORX and CORY are expected to be small, or unexpected.
- Bad pixels are signaled in the position model tables by having 1000.00 added to their position value
- Events are given non-zero STATUS values as follows, according to region or pixel on detector where it was detected:

  4 Event falls on a bad anode segment

  8 Event with uncertain time correlation

  16 Dead or unstable anode strip

  32 Event from anode next to a dead or bad anode strip

64 Event with uncertain gain parameters

128 Pixel with unstable gain

256 Pixel with poor X value determination

512 Pixel with poor Y value determination

1024 Area contaminated with calibration spectrum photons

2048 Event occurring before start-of-SCW STANDBY condition

4096 Outside of active detector area

8192 Hotspot

- This program flags each event in a 2-byte status word, according to any known problems with the detector area. This means that components further down the pipeline only need to look at the STATUS column of the JMXi-$* * **$-COR data structure for the consolidated ‘goodness’ of each event, rather than assembling this information themselves. It is then up to the individual science analysis component to decide which problems merit excluding the event and which are of little significance for the job in hand.
- Program now flags hotspots dynamically, not by values imported from JMXi-DETE-MOD, since hotspots can arise unexpectedly, and disappear, from revolution to revolution. Also, a given hotspot can pulsate within and between science windows while it is active. All this requires much quicker hotspot reaction than can be provided by instrument model maps.
- Program can also handle input from -ALL event lists as well as -RAW/-PRP extensions
- For OSA 6.0 it is expected that more extensive and efficient hotspot detection will be implemented to mirror the increased incidence of hotspots and hotspot activity.

Parent components:
j-correction (p. 108)

Subordinate components: Functions of the executable:
JCorPostition-Params
JCorPosition-Work
JCorPosGetTable
JCorPosEvents
JCorPosHotspot

Dependencies from other components:

- Energy corrections, j-cor-gain, should be performed before these position corrections.

- Unpacked telemetry must be available in ISDC level RAW - but program will exit without error if no RAW data is available.
- A valid version of JMXi-IMOD-GRP must be available, preferably the most recently updated set of instances

ISDC context: Scw QLA pipeline in the NRT data processing (fig. 2, 3 & 4), both under Automatic Operational Processing (fig. 1). Also, Data Correction and Binning (fig. 9) of the consolidated data processing (fig. 2& 5) in the Science Analysis Pipeline

Delivery date: Version 5.1 delivered 2005/6/2

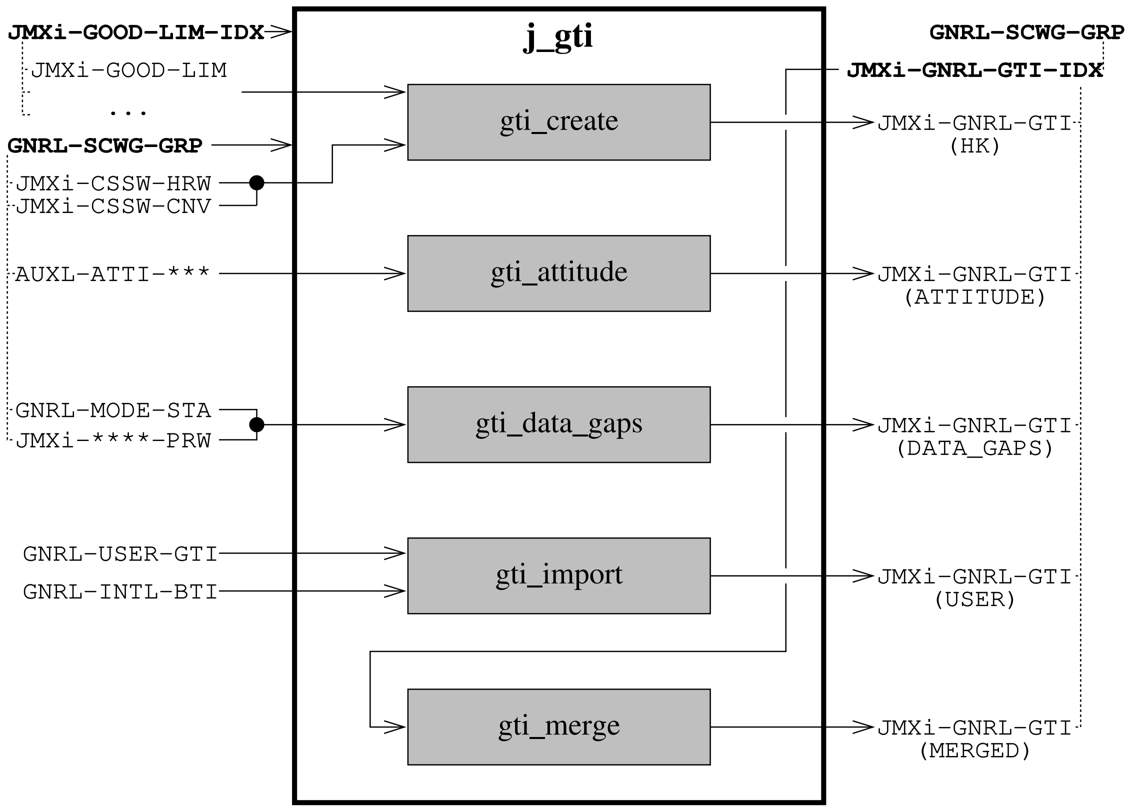

Figure 5.9: j-gti

5.8 j-gti

Type: Analysis Script **Responsible:** ESAC (P. Kretschmar) and Valencia University (S. Martínez-Núñez)

Purpose: Not foreseen in SRD

Function:

Derive Good-Time Intervals (GTIs) based on housekeeping data, information about satellite stability and data gaps and create a merged overall GTI which is used in analysis. The script allows the user to add his/her own GTI table.

Inputs:

JMXi-CSSW-HRW (p. 234) (housekeeping data)
JMXi-CSSW-CNV (p. 234)

JMXi-GOOD-LIM (p. 239) (limits for parameters)
AUXL-ATTI-PRE (p. 247) (attitude information)
AUXL-ATTI-SNA (p. 247)
AUXL-ATTI-HIS (p. 247)
GNRL-MODE-STA (p. 248) (overall INTEGRAL status)
GNRL-INTL-BTI (p. 248) (bad time interval table)
GNRL-USER-GTI (p. 249) (User GTI table)
JMXi-FULL-PRW (p. 238) (packet data)
JMXi-REST-PRW (p. 241)
JMXi-SPTI-PRW (p. 244)
JMXi-TIME-PRW (p. 245)
JMXi-SPEC-PRW (p. 244)
JMXi-RATE-PRW (p. 241)

Outputs:
JMXi-GNRL-GTI (p. 238) (specific GTI for different reason)

Parameters:

swgDOL p. 53	DOL of Science Window Group to be analyzed
jemxNum p. 46	JEM-X Instrument number (1 or 2)
IC-Group p. 46	DOL of Instrument Characteristics master group
IC-Alias p. 46	Selection alias for Instrument Characteristics
limitTable p. 47	DOL of table with parameter limits (”=take from IC)
attTolerance p. 40	Accepted attitude variability [arc sec]
gtiJemxNames p. 45	Names of instrumental GTIs to be included in merged GTI
gtiScNames p. 45	Names of spacecraft GTIs to be included in merged GTI
MergedName p. 47	Name of merged GTI to be used in analysis
gtiUser p. 46	DOL of the user GTI table
timeFormat p. 53	Time format to be used
Accuracy p. 40	Used accuracy for OBT to IJD conversion and viceversa
BTI-Dol p. 41	DOL of a bad time interval table (GNRL-INTL-BTI)
BTI-Names p. 41	Input BTI names to be considered
chatter p. 42	Verbosity level; 2 = normal
clobber p. 42	Clobber existing output data?
osimData p. 48	Are we using data produced by OSim?
subDir p. 53	Sub directory below the position of the SWG for results

gti-data-gaps-SPI-Mode p. 45	SPI modes for which a GTI depending on data gaps should be created
gti-data-gaps-ISGRI-Mode p. 45	ISGRI modes for which a GTI depending on data gaps should be created
gti-data-gaps-PICSIT-SGLE-Mode p. 45	PICSIT modes for which a GTI depending on data gaps should be created
gti-data-gaps-PICSIT-MULE-Mode p. 45	PICSIT modes for which a GTI depending on data gaps should be created
gti-data-gaps-COMPTON-SGLE-Mode p. 45	COMPTON modes for which a GTI depending on data gaps should be created
gti-data-gaps-COMPTON-MULE-Mode p. 45	COMPTON modes for which a GTI depending on data gaps should be created
gti-data-gaps-JMX1-Mode p. 45	JEM-X 1 modes for which a GTI depending on data gaps should be created
gti-data-gaps-JMX2-Mode p. 45	JEM-X 2 modes for which a GTI depending on data gaps should be created
gti-data-gaps-RemoveWrongGTIs p. 45	Remove overlapping GTIs and GTIS with start time > end time

Note: parameters with a prefix of "gti_data_gaps_" are override parameters for data normally found in the Science Window Group, which are passed on to gti_data_gap executable.

Resources: No unusual

Processing:

Read in parameters
Call gti-create to create a GTI table based on housekeeping data.
Call gti-attitude to create a GTI table based on pointing stability
Call gti-data-gaps to create a GTI table based on telemetry gaps.
Call gti-import to imports User GTIs into the ISDC GTI index files ("USER_GTI").
Call gti-merge to merge the different GTIs into a single one used in analysis.

Design remarks:
In the 'new' scheme of 2004 this component can be called both by the Science Window Pipeline (default) behaviour and by the science analysis scripts again, in case the user wants to override the results produced in the pipeline, e.g., to include a user GTI.

Parent components:

jemx-scw-analysis (p. 70)
Science Window Pipeline

Subordinate components:
gti-create (p. 64)
gti-attitude (p. 64)
gti-data-gaps (p. 34)
gti-merge (p. 64)
gti-import (p. 64)

Dependencies from other components: Raw and converted housekeeping data must exist in the swgDOL.
The tables with limits must have been defined.

ISDC context: Science Analysis

Delivery date: v4.5.6 19/10/2005

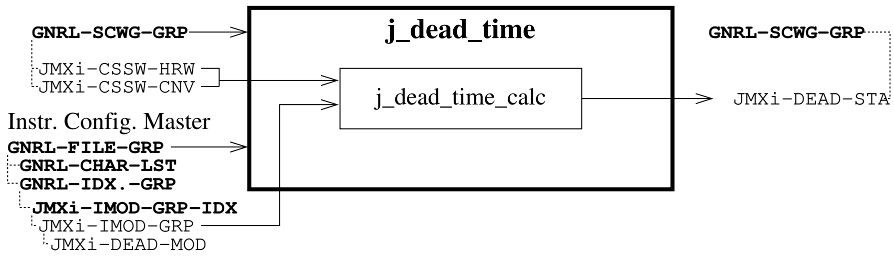

Figure 5.10: Analysis script **j_dead** deriving a deadtime history for the given Science Window.

5.9 j-dead

Type: Pipeline script **Responsible:** ESAC (P. Kretschmar) and Valencia University (S. Martínez-Núñez)

Purpose: Not foreseen in SRD

Function: Calculate dead time history over a science window.

Inputs:
JMXi-CSSW-HRW (p. 234)
JMXi-CSSW-CNV (p. 234)
JMXi-IMOD-GRP (p. 239)
JMXi-DEAD-MOD (p. 234) in group JMXi-IMOD-GRP

Outputs:
JMXi-DEAD-SCP (p. 234)

Parameters:

swgDOL p. 53	DOL of Science Window Group to be analyzed
jemxNum p. 46	JEM-X detector number
IC-Group p. 46	DOL of Instrument Characteristics master group
IC-Alias p. 46	Selection alias for Instrument Characteristics
instMod p. 46	DOL of JEM-X Instrument Model Group (empty=find in IC)
outputExists p. 48	Assume already existing output
osimData p. 48	Are we using data produced by OSim?
chatter p. 42	Verbosity level; 2 = normal
clobber p. 42	Clobber existing output data?
subDir p. 53	Sub directory below the position of the SWG for results
j-dead-time-calc-csswHrw p. 42	DOL of raw housekeeping data (override)
j-dead-time-calc-csswCnv p. 42	DOL of converted housekeeping data (override)

Note: parameters with a prefix of "j_dead_time_calc" are override parameters for data normally found in the Science Window Group, which are passed on to **j_dead_time_calc**. Normally they are hidden and set to empty (no effect).

Resources: No significant resource usage.

Processing:

Read in parameters
Call j-dead-time-calc

Design remarks:
In the 'new' scheme of 2004 this component can be called both by the Science Window Pipeline (default) behaviour and by the science analysis scripts again, in case the user wants to override the results produced in the pipeline.

Parent components:
jemx-scw-analysis (p. 70)
Science Window Pipeline

Subordinate components:
j-dead-time-calc (p. 134)

Dependencies from other components: Raw and converted HK data must exist in the swgDOL. The instrument description must be accessible.

ISDC context: Science Analysis

Delivery date: v1.3.0 03/09/05

5.9.1 j-dead-time-calc

Type: Executable **Responsible:** DNSC: C.A. Oxborrow

Purpose: Not forseen in SRD

Function: To make a history table of instrument dead-time for every 8 sec. HK interval in a given Science Window. Output table, JMXi-DEAD-SCP, lists packet time OBT (time of first event in the packet) and fraction of time that is dead. Input delay dead times, the sum of all the input processing losses, are saved in DEADTIME column. The DEAD_EFF column contains the total deadtime due to input delays, buffer flushing and grey filter losses. RATE_COR is the factor by which all event rates must be multiplied to correct for input delays and buffer losses, but NOT for grey filter. All values are given as decimal fractions.

Interfaces: 1) Onboard event processing counter come from the raw housekeeping for the SWG
2) OBTs for HK packets from previous j_data_preparation output
3) Instrument mode history comes from j_data_preparation output
4) Instrument dead-time model JMXi-DEAD-MOD comes from the instrument model group JMXi-IMOD-GRP

Inputs:
JMXi-CSSW-HRW (p. 234) In input SWG, ISDC level PRP
JMXi-CSSW-CNV (p. 234) In input SWG: output from Data Preparation
JMXi-IMOD-GRP (p. 239) Instrument model group
JMXi-DEAD-MOD (p. 234) In instrument model group
JMXi-IMOD-GRP (p. 239)

Outputs:
JMXi-DEAD-SCP (p. 234) In output SWG or override output extension

Parameters:

inSWG p. 46	DOL	Input Science Window Group
outSWG p. 49	DOL	Output Science Window Group
instMod p. 46	DOL	Input instrument model group
csswHrw p. 42	DOL	Override for input extension JMXi-CSSW-HRW
csswCnv p. 42	DOL	Override for input extension JMXi-CSSW-CNV
deadTime p. 42	DOL	Override for output extension JMXi-DEAD-STA
jemxNum p. 46	int.	Number of JEM-X instrument used (1 or 2)
clobber p. 42	bool.	Clobber existing output data structures?
chatter p. 42	int.	Verbosity level of logfile output (0,1,2,3,4,5)

Resources: No special resources needed: processing is simple and fast, and there are no large

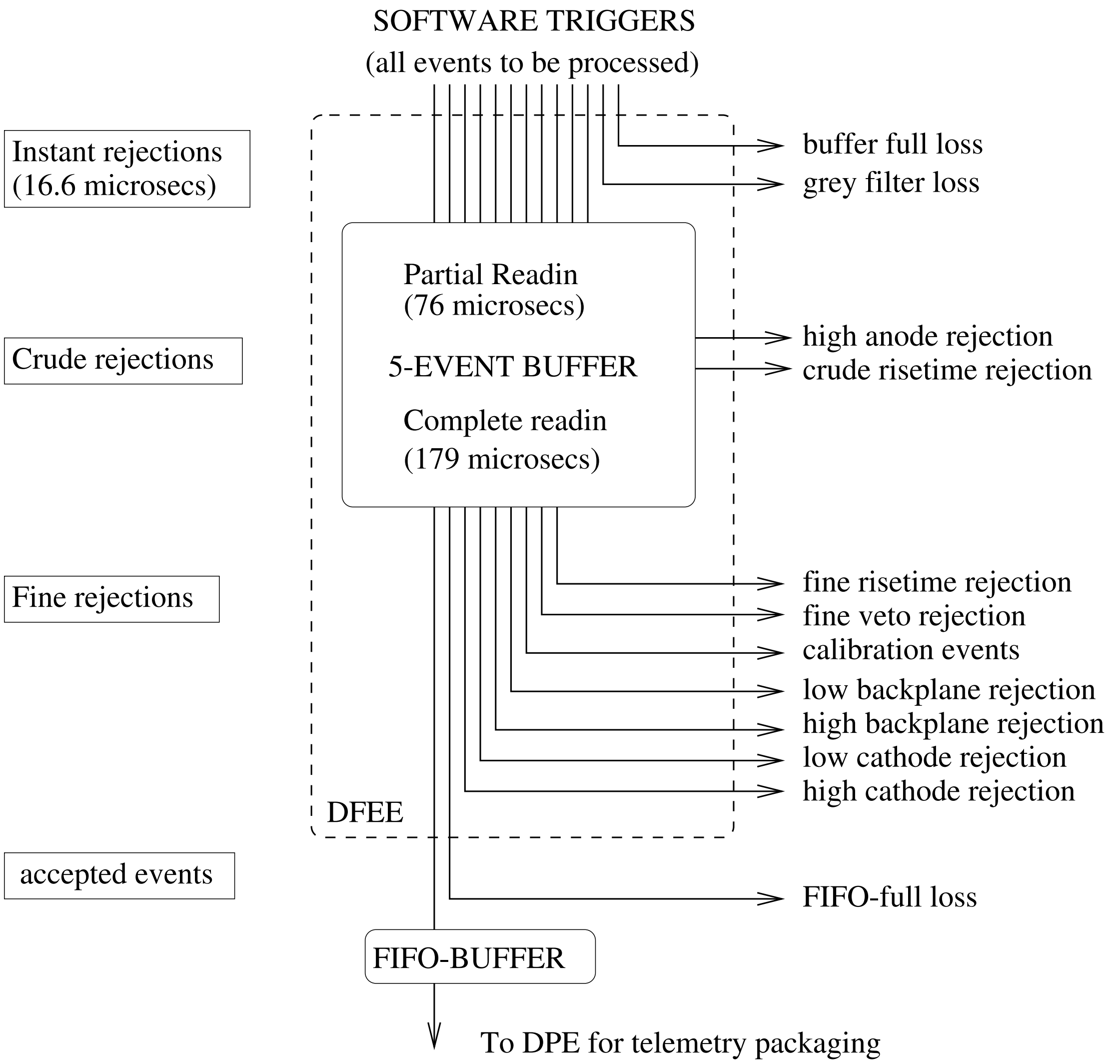

Figure 5.11: DFEE event processing and dead time: This figure shows how the dead times due to event rejection, buffer losses and grey filter are calculated

tables to read in. Processing scales linearly with number of events

Processing:

Initialize executable with CommonInit
Read the input parameter file
    Open SWG, check for all the needed input extensions
    Find instrument number: 1 or 2 and make prefix string
    Open the dead time model JMXi-DEAD-MOD
Read in the HK packet information
    Read in OBT for each packet from JMXi-CSSW-CNV
    Read in the packet PERIOD keyword from raw JMXi-CSSW-HRW
    Read in the event processing counters from JMXi-CSSW-HRW
    Read in the CPU mode from JMXi-CSSW-HRW
Read the dead time model
    Read first column of dead time values for 16MHz mode
Perform the work for each 8 second period:
Loop over available HK packets
    Calculate input dead time by summing up buffer readin delays for
        events processed in different ways
    Multiply processing dead time by appropriate CPU_MODE factor
    Calculate additional deadtimes due to buffer losses
    Calculate event rate correction factor due to input deadtime and
        buffer losses
    Calculate deadtime due to grey filter losses
    Calculate total deadtime due to input processing, buffer losses and
        grey filter losses End of loop over packets
Save the packet OBTs and dead times to JMXi-DEAD-STA
    DEADTIME contains the instrument read in (processing) deadtime only
    RATE_COR contains correction factor for event rates due to both
        input delays and buffer losses
    DEAD_EFF contains total effective dead time due to all factors
Close SWG and tidy up

Design remarks:

1. All deadtimes are calculated from the relative number of events in the different counters produced by the onboard software to keep track of the fate of each event. This strategy

has been used since it was found that some hardware triggers are double triggers so that a simple equation like 'AcceptedEvents/HardwareTriggers' is not valid

2. Time taken to read in events for each kind of processing are stored in JMXi-DEAD-MOD. The values were determined by on ground measurements by Søren Brandt at DNSC. Though there is room in the table for both 16 MHz and 8 MHz values, only the 16 MHz (zero waitstate) values are read in. All processing slower than this is calculated using multiplicative factors: 50% extra for switching to non-zero waitstate and 100% extra for using 8 MHz instead of 16 MHz.

3. Event counters used are: accepted events, events rejected by grey filter, 5-event (input) buffer loss, high anode rejected events, low risetime rejected events, high risetime rejected events, veto rejected events, calibration events, high backplane rejected events, low backplane rejected events, high cathode rejected events, low cathode rejected events, number of software triggers, and number of FIFO-buffer losses

4. Program can now handle mixed 8 MHz and 16 MHz onboard processing in a single science window

Parent components:
j-dead-time (p. **??**)

Subordinate components: Functions of the executable:
JDeadTimeParams
JDeadTime-Work
JDeadTimeSave
JDeadGetCounters

Dependencies from other components: Assumes that FOBT is available for each HK packet from Data Preparation

ISDC context: Runs as part of j_dead_time, which is part of the JEM-X Science Analysis Pipeline. See figures on pages 18 and 19

Delivery date: Version 4.0 delivered 2004/10/8

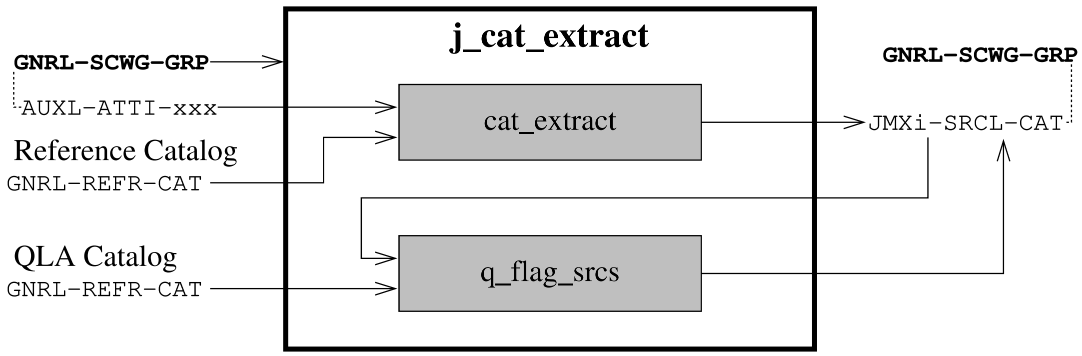

Figure 5.12: j-cat-extract

5.10 j-cat-extract

Type: Pipeline script **Responsible:** ESAC (P. Kretschmar) and Valencia University (S. Martínez-Núñez)

Purpose: Not foreseen in SRD

Function: Extract a list of relevant sources — possibly including strong sources outside the actual FOV — from the ISDC Reference Catalog and merges differents catalogs if the user wants.

Inputs:
AUXL-ATTI-PRE (p. 247)
AUXL-ATTI-SNA (p. 247) or
AUXL-ATTI-HIS (p. 247) or
GNRL-REFR-CAT (p. 248) (Reference Catalog)

Outputs:
JMXi-SRCL-CAT (p. 245)

Parameters:

swgDOL p. 53	DOL of Science Window Group to be analyzed
IC-Group p. 46	DOL of Instrument Characteristics master group
IC-Alias p. 46	Selection alias for Instrument Characteristics
jemxNum p. 46	JEM-X Instrument number (1 or 2)
osimData p. 48	Are we using data produced by OSim?
refCat p. 50	DOL of ISDC Reference Catalog
usrCat p. 53	DOL of User Catalog
radiusMin p. 50	Lower limits (in deg) of cones for position selection
radiusMax p. 50	Upper limits (in deg) of cones for position selection
fluxDef p. 44	Column used for flux selection
fluxMin p. 44	Low limit for flux selection
fluxMax p. 44	Upper limit for flux selection
class p. 42	Identifier for selection by object class
date p. 42	IJD for public data (¡0 = all public)
chatter p. 42	Verbosity level
clobber p. 42	Clobber existing output data?
subDir p. 53	Sub directory below the position of the SWG for results

Resources: Resource usage is mostly driven by the size of the ISDC Reference catalogue, as this could in principle be read in in its entirety. For each source ~ 5 kByte are used.

Processing:

Call cat-extract to extract sources from Reference Catalog
Call cat-merge to merge the extract sources from Reference Catalog and the given by the user

Design remarks:
This script uses no instrument-specific executables. Its detailed design may still evolve significantly depending on the requirements of ISDC for SA, QLA and OSA.

Parent components:
jemx-scw-analysis (p. 70)

Subordinate components:
cat-extract (p. 64)
cat-merge (p. 64)

Dependencies from other components: Attitude data must exist in the swgDOL.
The catalogs must existed and be filled with sensible data.

ISDC context: Science Analysis

Delivery date: v4.5.6 19/10/2005

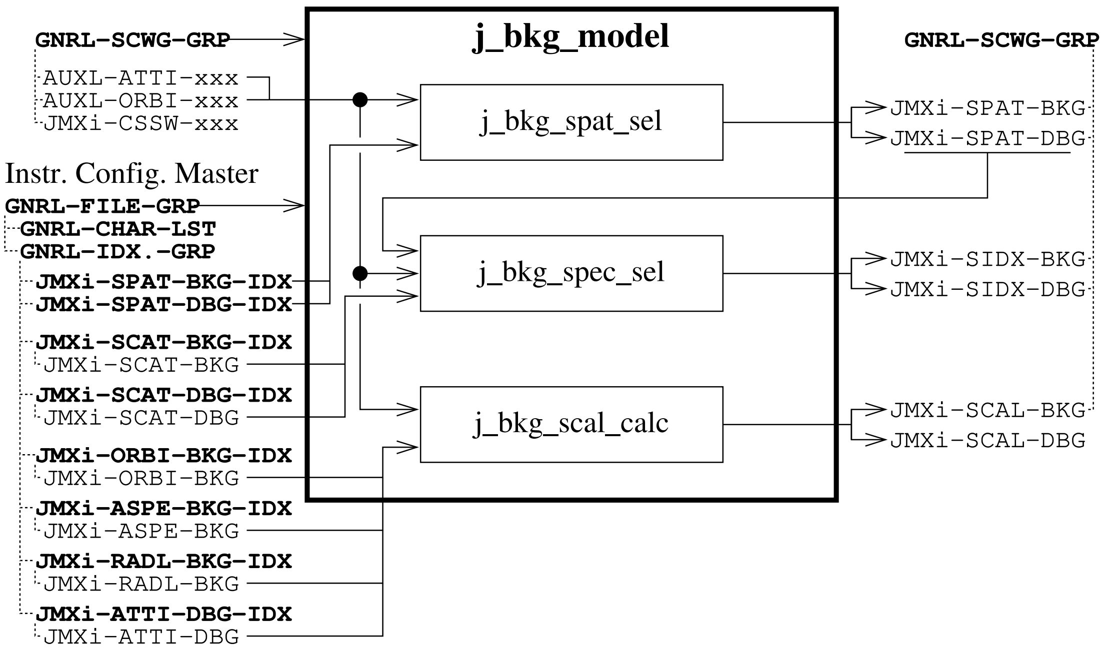

Figure 5.13: j-bkg-model

5.11 j-bkg-model

Type: Analsyis Script **Responsible:** ESAC (P. Kretschmar) and Valencia University (S. Martínez-Núñez)

Purpose: 18.17.1 18.17.2 18.17.3

Function: Extract and calculate background model data adapted to the current Science Window from a database of background model data.

Inputs:
JMXi-SPAT-BKG-IDX (p. 243)
JMXi-SPAT-DBG-IDX (p. 243)
JMXi-SCAT-BKG-IDX (p. 242)
JMXi-SCAT-DBG-IDX (p. 242)
JMXi-ORBI-BKG-IDX (p. 240)
JMXi-ASPE-BKG-IDX (p. 232)
JMXi-RADL-BKG-IDX (p. 241)
JMXi-ATTI-DBG-IDX (p. 233)
AUXL-ATTI-HIS (p. 247) or -SNA, -NRT

AUXL-ORBI-HIS (p. 247) or -SNA, -NRT
JMXi-CSSW-HRW (p. 234) or -CNV, -AVG etc.

Outputs:
JMXi-SPAT-BKG (p. 243)
JMXi-SPAT-DBG (p. 243)
JMXi-SIDX-BKG (p. 242)
JMXi-SIDX-DBG (p. 243)
JMXi-SCAL-BKG (p. 242)
JMXi-SCAL-DBG (p. 242)

Parameters:

swgDOL p. 53	DOL of Science Window Group to be analyzed
IC-Group p. 46	DOL of Instrument Characteristics master group
IC-Alias p. 46	Selection alias for Instrument Characteristics
jemxNum p. 46	JEM-X detector number
osimData p. 48	Are we using data produced by OSim?
simulated p. 51	Use simulated background models?
chatter p. 42	Verbosity level; 2 = normal
clobber p. 42	Clobber existing output data?
subDir p. 53	Sub directory below the position of the SWG for results

Resources: See called executables

Processing:

Read in parameters
Get DOLs to all the background Index Groups from the master IC group
Call j-bkg-spat-sel to select models of bkg spatial intensity distribution
Call j-bkg-spec-sel to determine spatial distribution of bkg spectra
Call j-bkg-scal-calc to derive bkg scaling factors

Design remarks:
The background model data is extracted in a generic manner, independent of the JEM-X telemetry formats used, energy binning etc. It is up to the subsequent analysis tools to make use of this model data.

Currently, the background models are not realistic, and then they are not used by the other components.

Parent components:
jemx-scw-analysis (p. 70)

Subordinate components:
j-bkg-spat-sel (p. 144)
j-bkg-spec-sel (p. 147)
j-bkg-scal-calc (p. 150)

Dependencies from other components: Auxiliary and housekeeping data must exist in the swg-DOL. The background database must exist and be filled with sensible entries.

ISDC context: Science Analysis

Delivery date: v4.5.6 19/10/2005

5.11.1 j-bkg-spat-sel

Type: executable **Responsible:** S. Maisala

Purpose: 18.17

Function: This component selects automatically the most appropriate spatial instrument and diffuse background models. Spatial distributions are selected from either observed or simulated models. Instrument backcground selection depends on *orbital, attitude* and *radiation level* parameters. Diffuse background is selected according to the *sky coordinates* of the observation. Simulated and observed shadowgram catalogues are indexed group tables where each model is a separate image. The first blank field observations are analyzed and observed background models are installed in ISDC system. The models will be updated during the mission. **Instrumental background:** *JMXi-SPAT-BKG-IDX* is a list of individual models of normalized background shape shadowgrams. Models are categorized as sub-groups where each sub-group has a selection of shadowgram shapes which are proportional to parameter values. Decisive parameters are *orbital phase, sun aspect angle* and *radiation level*. Sub-group separation is done by setting the selection keywords for the other parameters to the disregard values. Each sub-group has only one decisive parameter and the others are set to the disregard values orbital phase (-1), sun angle (-1) and radiation level (0). This scheme makes possible to reduce storage requirements compared to multidimensional array where for every orbital phase we have a list of all possible combinations of solar angle and radiation level models. The selection is a multistep procedure where *j-bkg-spat-sel* selects the two closest models for each sub-group corresponding to the parameters values and observation and makes a weighted interpolation between them.

The first step is to use the sub-group where we are using information of the orbital phase. In this case we select the two closest models from 'orbital phase sub-group' by selecting the shortest distance between orbital phase and the database parameter values. These selected models are interpolated to the one 'orbital phase sub-group' model by using information of the models spatial distribution and they are weighted with the coeffecients which are propotional of the differences between models distances and the real orbital phase. Second step is to use 'sun angle sub-group' and as above we select the two closest models and make interpolation between them. Third step is to select 'radiation level sub-group' model and the selection method is same.

After this three selection steps *j-bkg-spat-sel* has a three interpolated models from each sub-group and the final step is to add models wich each other and make a normalization to create the appropriate background model.

Diffuse background: *JMXi-SPAT-DBG-IDX* is a list of normalized shadowgrams for diffuse background. *JMXi-ATTI-DBG* data structure contains information of *gal. longitute, gal. latitude, tilt angle* and *indexed diffuse background spectra*. *j-bkg-spat-sel* selects first the closest shadowgram model from *JMXi-SPAT-DBG-IDX* by longitude and latitude. This model is not suitable for direct usage in data analysis and it must be modified by tilting and rotating. The rotation is defined by the **INTEGRAL** roll angle and the tilt parameters are read from *JMXi-ATTI-DBG* (tilt strength and slope).

Inputs:
JMXi-SPAT-BKG-IDX (p. 243) (background spatial models, in the [ic] files)
JMXi-SPAT-DBG-IDX (p. 243)
AUXL-ATTI-HIS (p. 247) (auxliary data)
AUXL-ATTI-PRE (p. 247)
AUXL-ATTI-SNA (p. 247)
AUXL-ORBI-HIS (p. 247)
AUXL-ORBI-PRE (p. 248)
JMXi-CSSW-HRW (p. 234) (house keeping data)
JMXi-CSSW-CNV (p. 234)
JMXi-ATTI-DBG (p. 233) (diffuse background attitude parameter database, in the [ic] files)

Outputs:
JMXi-SPAT-BKG (p. 243) (spatial models)
JMXi-SPAT-DBG (p. 243)

Parameters:

inSWG p. 46	Input SWG
outSWG p. 49	Output SWG (override)
jemxNum p. 46	JEM-X instrument number (1,2)
clobber p. 42	Clobber existing output data structure (y/n)?
chatter p. 42	Verbosity of logfile output (0,1,2,3,4,5)
simulated p. 51	simulated (y/n)
bkgSpatIdx p. 41	Instrument background spatial model catalogue DOL (override)
dbgSpatIdx p. 42	Diffuse background spatial model catalogue DOL (override)
bkgSpatial p. 41	Instrument background spatial model DOL (override)
dbgSpatial p. 42	Diffuse background spatial model DOL (override)
dbgGalDbg p. 42	Diffuse background database DOL

Resources: No significant resource usage.

Processing:
- check parameter simulated (y/n)
  - call DAL3AUX funtions to get attitude and orbital phase
  - call DAL3HK funtions to get radiation level
- make model selection:
  - Instrument background
    - open JMXi-SPAT-BKG-IDX
    - database model validity time selection
    - select orbital BKG model
    - select sun angle BKG model
    - select radiation level BKG model
    - combine and normalize model

return JMXi-SPAT-BKG = selected BKG model
Diffuse background
open JMXi-SPAT-DBG-IDX
open JMXi-ATTI-DBG
select DBG model
select tilt and slope parameters
calculate model
return JMXi-SPAT-DBG = selected DBG model
write JMXi-SPAT-BKG and JMXi-SPAT-DBG to SCW

Parent components:
j-bkg-model (p. 141)

Subordinate components:
j-bkg-spatGetParams
j-bkg-spatProcessData
j-bkg-spatDbgSelection
j-bkg-spatBkgSelection
j-bkg-spatDbgTangle

Dependencies from other components: Auxiliary and housekeeping data must exist in the swgDOL. The catalog databases must exist and be filled with sensible entries.

Pending TBDs: Model selection and database updating during mission.

ISDC context: Scientific Analysis

Delivery date: Version 1.5.1 delivered 09/06/2004

5.11.2 j-bkg-spec-sel

Type: Executable **Responsible:** S. Maisala

Purpose: 18.17

Function: This component generates diffuse and instrument background spectrum catalogue index tables. Tables are the same size as detector size (256*256), and each detector pixel has an index value which corresponds to the characteristic background spectrum model. The sum of each spectrum is normalized to 1.

Instrumental background: *JMXi-SIDX-BKG-IDX* is a multi dimensional list of individual spectrum map models over detector. Each spectral map is proportional to parameter values. Since it makes no real sense to interpolate spectrum index numbers we have to use multi dimensional array of relative parameters as *phase angle, solar angle* and *radiation level*. *j-bkg-spec-sel* selects the closest model corresponding to the parameter values and observation to create the appropriate spectral index map.

Diffuse background: The *JMXi-SIDX-BKG* will have the same selected spectrum index value for the whole detector area. The selected spectrum index number is taken from database by using longitude and latitude information to select the closest spectrum model.

Inputs:
JMXi-SCAT-BKG (p. 242) (background spectra catalogues, in the [ic] files)
JMXi-SCAT-DBG (p. 242)
JMXi-SIDX-BKG-IDX (p. 243) (background spectrum map model catalogues, in the [ic] files)
JMXi-SIDX-DBG-IDX (p. 243)
AUXL-ATTI-HIS (p. 247) (auxliary data)
AUXL-ATTI-PRE (p. 247)
AUXL-ATTI-SNA (p. 247)
AUXL-ORBI-HIS (p. 247)
AUXL-ORBI-PRE (p. 248)
JMXi-CSSW-HRW (p. 234) (house keeping data)
JMXi-CSSW-CNV (p. 234)

Outputs:
JMXi-SIDX-BKG (p. 242) (spectral maps)
JMXi-SIDX-DBG (p. 243)

Parameters:

inSWG p. 46	Input SWG
outSWG p. 49	Output SWG (override)
jemxNum p. 46	JEM-X instrument number (1,2)
clobber p. 42	Clobber existing output data structure (y/n)?
chatter p. 42	Verbosity of logfile output (0,1,2,3,4,5)
bkgSpecIdx p. 41	Instrument background spectrum map models DOL(override)
dbgSpecIdx p. 42	Diffuse background spectrum map models DOL (override)
bkgSpecMap p. 41	Instrument background spectrum map DOL (override)
dbgSpecMap p. 42	Diffuse background spectrum map DOL (override)

Resources: No significant resource usage.

Processing:

call DAL3AUX funtions to get attitude and orbital phase
call DAL3HK funtions to get radiation level
make model selection:
  Instrument background
    open JMXi-SIDX-BKG-IDX
    database model validity time selection
    select closest model by orbital,sun angle and radiation level information
    return JMXi-SIDX-BKG
  Diffuse background
    open JMXi-SIDX-DBG-IDX
    database model validity time selection
    select closest model by longitude and latitude information
    return JMXi-SIDX-DBG
write JMXi-SIDX-BKG and JMXi-SIDX-DBG to the SCW.

Design remarks:
None

Parent components:
j-bkg-model (p. 141)

Subordinate components:
j-bkg-specGetParams
j-bkg-specProcessData
j-bkg-specDbgSelection
j-bkg-specBkgSelection

Dependencies from other components: Auxiliary and housekeeping data must exist in the swgDOL. The catalog database must exist and be filled with sensible entries.

Pending TBDs: Model selection and database updating during mission.

ISDC context: QLA and SA

Delivery date: Version 1.5.0 delivered 14/06/2004

5.11.3 j-bkg-scal-calc

Type: executable **Responsible:** S. Maisala

Purpose: 18.17

Function: This component calculates automatically the most appropriate instrument and diffuse background scaling factors. Scaling depends on orbital and attitude data, radiation level and exposure time.

Inputs:
AUXL-ATTI-HIS (p. 247) (auxliary data)
AUXL-ATTI-PRE (p. 247)
AUXL-ATTI-SNA (p. 247)
AUXL-ORBI-HIS (p. 247)
AUXL-ORBI-PRE (p. 248)
JMXi-CSSW-HRW (p. 234) (house keeping data)
JMXi-CSSW-CNV (p. 234)
JMXi-ASPE-BKG (p. 232) (background scaling factor databases, in the [ic] files)
JMXi-ORBI-BKG (p. 239)
JMXi-RADL-BKG (p. 240)
JMXi-ATTI-DBG (p. 233)

Outputs:
JMXi-SCAL-BKG (p. 242) (scaling factors)
JMXi-SCAL-DBG (p. 242)

Parameters:

inSWG p. 46	Input SWG
outSWG p. 49	Output SWG (override)
jemxNum p. 46	JEM-X instrument number (1,2)
clobber p. 42	Clobber existing output data structure (y/n)?
chatter p. 42	Verbosity of logfile output (0,1,2,3,4,5)
bkgScale p. 41	Instrument background scaling factor DOL (override)
dbgScale p. 42	Diffuse background scaling factor DOL (override)
attiScale p. 40	Attitude scaling factor DOL (override)
aspeScale p. 40	Sun aspect angle scaling factor DOL (override)
orbiScale p. 48	Orbital phase scaling factor DOL (override)
radlScale p. 50	Radiation level background scaling factor DOL (override)

Resources: No significant resource usage.

Processing:
call DAL3AUX funtions to get radiation information from housekeeping
call DAL3AUX funtions to get attitude and orbital phase

open:
  JMXi-ASPE-BKG
  JMXi-ORBI-BKG
  JMXi-RADL-BKG
  JMXi-ATTI-DBG
loop for exposure time. Step by every 8th second
  calculate instrument and diffuse background scaling factors
end loop
return JMXi-SCAL-BKG and JMXi-SCAL-DBG

Parent components:
j-bkg-model (p. 141)

Subordinate components:
j-bkg-scalGetParams
j-bkg-scalProcessData

Dependencies from other components: Auxiliary and housekeeping data must exist in the swgDOL. The catalog database must exist and be filled with sensible entries.

Pending TBDs: Model selection and database updating during mission.

ISDC context: QLA and SA

Delivery date: Version 1.5.0 delivered 10/06/2004

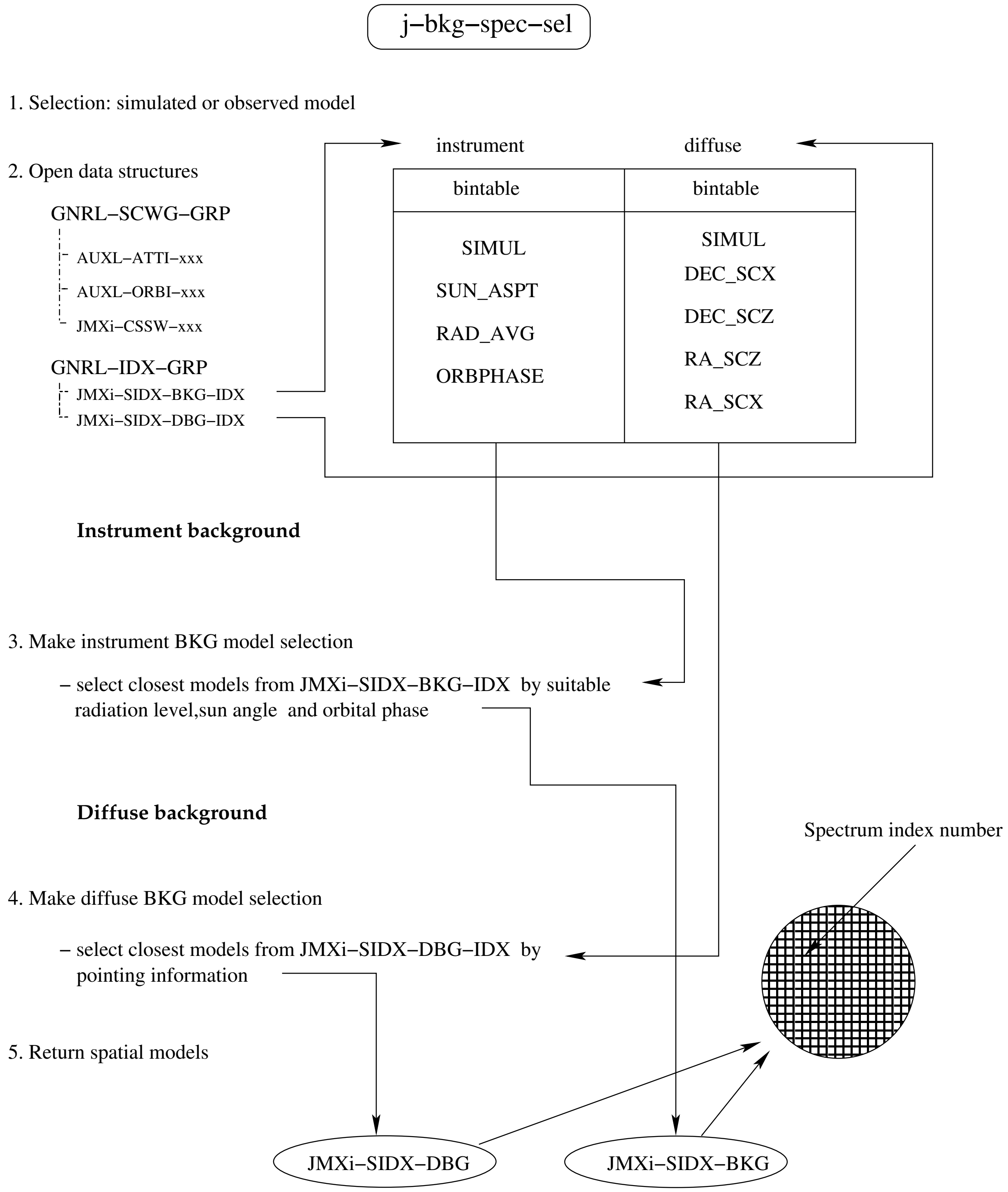

Figure 5.14: j-bkg-spec-sel

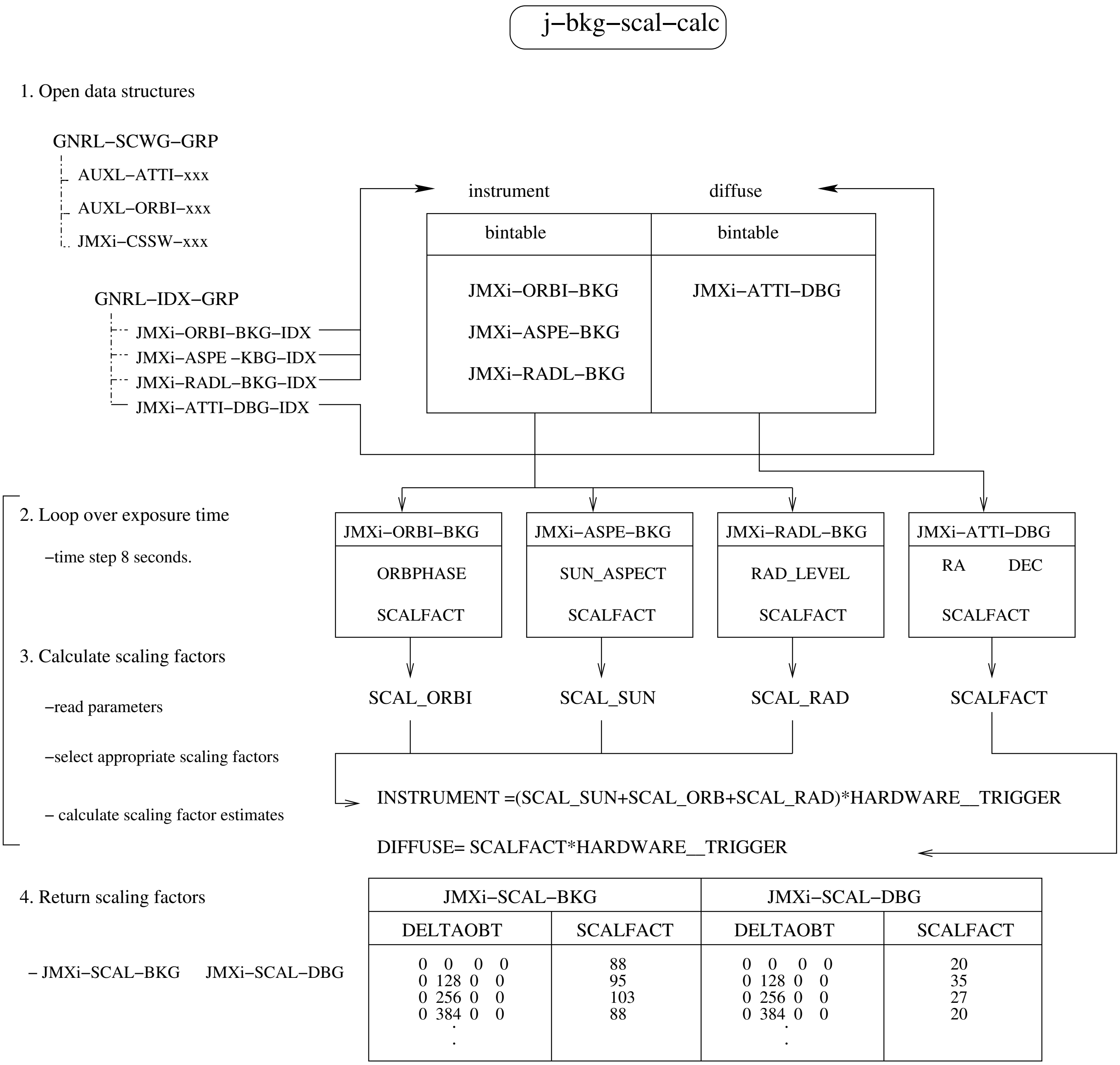

Figure 5.15: j-bkg-scal-calc

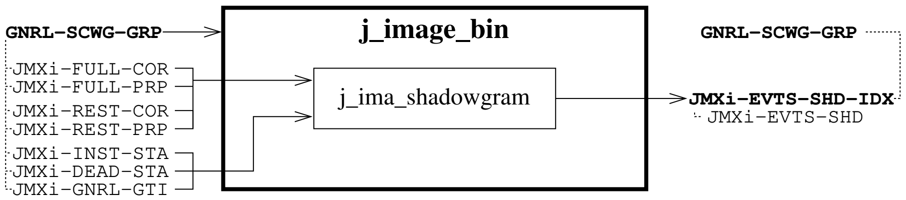

Figure 5.16: Pipeline application script j_image_bin

5.12 j-image-bin

Type: Analysis Script **Responsible:** ESAC (P. Kretschmar) and Valencia University (S. Martínez-Núñez)

Purpose: Fulfill item 18.19.2 and 18.21.2 in SRD

Function: Bin JEM-X imaging data, i.e. Full Imaging or Restricted Imaging events to shadowgrams.

Inputs:
JMXi-FULL-PRP (p. 237) (event data)
JMXi-FULL-COR (p. 237)
JMXi-REST-PRP (p. 241)
JMXi-REST-COR (p. 241)
JMXi-FULL-ALL (p. 237) (event data, new scheme)
JMXi-REST-ALL (p. 241)
JMXi-INST-STA (p. 239) (status & grey filter information)
JMXi-DEAD-SCP (p. 234) (deadtime data)
JMXi-GNRL-GTI-IDX (p. 238)
JMXi-GNRL-GTI (p. 238) (good-time interval, MERGED usually)

Outputs:
JMXi-EVTS-SHD-IDX (p. 236) (shadowgram index)
JMXi-EVTS-SHD (p. 236) (multiple shadowgrams)

Parameters:

swgDOL p. 53	DOL of Science Window Group to be analyzed
jemxNum p. 46	JEM-X detector number
IC-Group p. 46	DOL of Instrument Characteristics master group
IC-Alias p. 46	Selection alias for Instrument Characteristics
instMod p. 46	DOL of JEM-X Instrument Model Group (empty=find in IC)
nChanBins p. 47	Number of energy-resolved shadowgrams to create
chanLow p. 41	Lower channel boundaries for shadowgrams
chanHigh p. 41	Upper channel boundaries for shadowgrams
timeStart p. 53	Start of time interval [IJD] (-1 = use swgDOL bounds)
timeStop p. 53	End of time interval [IJD] (-1 = use swgDOL bounds)
evtType p. 44	Event type (0=FULL, 1=REST, -1=both)
backCorr p. 40	Perform background correction?
shdType p. 51	Type of shadowgram 0 : skew, 1 : standard
shdRes p. 51	DOL of the output shadowgrams
radiusLimit p. 49	[mm] Detector counts will only be used to this radius
rowSelect p. 51	Additional row filter string e.g. abs(DETX) ¡ 110.0
gtiNames p. 45	Names of GTI tables to be used
osimData p. 48	Are we using data produced by OSim?
chatter p. 42	Verbosity level; 2 =4 normal
clobber p. 42	Clobber existing output data?
subDir p. 53	Sub directory below the position of the SWG for results
j-ima-shadowgram-fullPrp p. 44	DOL of FULL imaging prepared data (override)
j-ima-shadowgram-fullCor p. 44	DOL of FULL imaging corrected data (override)
j-ima-shadowgram-restPrp p. 50	DOL of REST imaging prepared data (override)
j-ima-shadowgram-restCor p. 50	DOL of REST imaging corrected data (override)
j-ima-shadowgram-bkgSpecMap p. 41	DOL of Instrument bkg spectral map (override)
j-ima-shadowgram-dbgSpecMap p. 42	DOL of Diffuse bkg spectral map (override)
j-ima-shadowgram-bkgSpatial p. 41	DOL of Instrument bkg spatial map (override)
j-ima-shadowgram-dbgSpatial p. 42	DOL of Diffuse bkg spatial map (override)
j-ima-shadowgram-bkgScale p. 41	DOL of Instrument bkg scaling factor (override)
j-ima-shadowgram-dbgScale p. 42	DOL of Diffuse bkg scaling factor (override)
j-ima-shadowgram-deadTime p. 43	DOL of input table JMXi-DEAD-SCP (override)

Note: parameters with a prefix of "j_ima_shadowgram" are override parameters for data normally found in the Science Window Group, which are passed on to **j_ima_shadowgram**. Normally they are hidden and set to empty (no effect).

Resources: See called executables. The script itself uses minimal resources.

Processing:

Read in parameters
For i=1,nChanbins do:
    Call j-ima-shadowgram with chanLow(i), chanHigh(i) as boundaries

Design remarks:

Parent components:
jemx-scw-analysis (p. 70)

Subordinate components:
j-ima-shadowgram (p. 157)

Dependencies from other components: Corrected events with position information (FULL or REST) must be available.

ISDC context: Science Analysis

Delivery date: v4.5.6 19/10/2005

5.12.1 j-ima-shadowgram

Type: Executable **Responsible:** DNSC (Niels J. Westergaard)

Purpose: 18.19.2 18.21.2

Function: Produce a shadowgram from event file by selecting events according to arrival time and energy from a single Science Window. If the parameter "backCorr" is set to "yes" then the estimated background will be subtracted from the shadowgram.

The background estimate is calculated from the JMXi-SCAT-XXX (IC file) and JMXi-SPAT-XXX, JMXi-SCAL-XXX, and JMXi-SIDX-XXX in the SWG results. ("XXX" stands for either "BKG" (instrument background) or "DBG" (diffuse background)). The two components are added.

The shadowgram output format is defined by the "shdType" parameter:

shdType	Format of result
0	Skew i.e. pixels correspond to mask
1	Regular based on corrected positions
2	Regular based on raw event positions
3	Regularized i.e. mapped to regular grid

When background correction is requested and the parameter 'shdType' is given as 0 (skew), then the resulting images in the index table 'JMXi-EVTS-SHD-IDX' are: Original shadowgram ('SHDTYPE = 30'), estimated BKG shadowgram ('SHDTYPE = 20'), estimated DBG shadowgram ('SHDTYPE = 10'), and finally the background corrected shadowgram ('SHDTYPE = 0'). This scheme ensures that j-ima-iros will pick the correct shadowgram (i.e. with SHDTYPE == 2) for image making. For 'SHDTYPE = 1' the corresponding types will become: 31, 21, and 11.

In case of mixed mode i.e. Full Imaging (FULL) and Restricted Imaging (REST) modes occurring within the selected time interval the Restricted Imaging mode time and energy binning will apply. If several selections (in PI and/or timewise) are wanted then j-ima-shadowgram must be run several times.

The deadtime correction is made when the data structure JMXi-DEAD-SCP if present the the SWG. A simple average of the deadtime-fraction is used.

Inputs:
JMXi-FULL-COR (p. 237) (event data)
JMXi-REST-COR (p. 241) (event data)
JMXi-FULL-PRP (p. 237) (event data)
JMXi-REST-PRP (p. 241) (event data)
JMXi-IMOD-GRP (p. 239)
JMXi-FBDS-MOD (p. 236) (binning table)
JMXi-RBDS-MOD (p. 241) (binning table)
JMXi-GNRL-GTI (p. 238) (GTI table (OBT))

JMXi-SIDX-DBG (p. 243) (background data)
JMXi-SPAT-DBG (p. 243) (background data)
JMXi-SCAL-DBG (p. 242) (background data)
JMXi-SIDX-BKG (p. 242) (background data)
JMXi-SPAT-BKG (p. 243) (background data)
JMXi-SCAL-BKG (p. 242) (background data)
JMXi-DEAD-SCP (p. 234) (deadtime data)

Outputs:
JMXi-EVTS-SHD (p. 236) (the shadowgram)
JMXi-EVTS-SHD-IDX (p. 236)

Parameters:

inSWG p. 46	DOL of input SWG
outSWG p. 49	DOL of output SWG (override)
instMod p. 46	DOL of JMXi-IMOD-GRP
clobber p. 42	Clobber previous data structures?
chatter p. 42	Level of logging output
jemxNum p. 46	JEMX number (1 or 2)
evtType p. 44	Requested event type
chanMin p. 42	Lowest PI value
chanMax p. 41	Highest PI value
radiusLimit p. 49	Upper radius limit in mm
rowSelect p. 51	Additional row filter string
timeStart p. 53	Start of time interval (in IJD)
timeStop p. 53	End of time interval (in IJD)
gtiNames p. 45	Names of GTI tables to be used
backCorr p. 40	Perform background correction
shdType p. 51	Type of shadowgram
bkgSpecCat p. 41	DOL of Instrument bkg spectral catalog
dbgSpecCat p. 42	DOL of Diffuse bkg spectral catalog
fullPrp p. 44	DOL of FULL imaging prepared data (override)
fullCor p. 44	DOL of FULL imaging corrected data (override)
restPrp p. 50	DOL of REST imaging prepared data (override)
restCor p. 50	DOL of REST imaging corrected data (override)
bkgSpecMap p. 41	DOL of Instrument bkg spectral map (override)
dbgSpecMap p. 42	DOL of Diffuse bkg spectral map (override)
bkgSpatial p. 41	DOL of Instrument bkg spatial map (override)
dbgSpatial p. 42	DOL of Diffuse bkg spatial map (override)
bkgScale p. 41	DOL of Instrument bkg scaling factor (override)

Name	Description
dbgScale p. 42	DOL of Diffuse bkg scaling factor (override)
shdRes p. 51	DOL of output (resulting) shadowgram (override)
deadTime p. 42	DOL of dead time table (input) (override)

Resources: No unusual requirements.

Processing:
Read the GTI table and combine with timeStart and timeStop
Get dead time table (if it is present)
Adjust the PI limits if REST events are requested
Setup the selection criteria
if FULL or both FULL and REST is requested {
    Use DAL3JEMX calls to select and get FULL events
    Update the shadowgram }
if REST or both FULL and REST is requested {
    Use DAL3JEMX calls to select and get REST events
    Update the shadowgram }
Apply the dead time correction factor
Write shadowgram to new instance of JMXi-EVTS-SHD in index table JMXi-EVTS-SHD-IDX

Design remarks:
If more PI intervals are requested then the executable must be called several times. This is handled by j-image-bin.

Parent components:
j-image-bin (p. 154)

Subordinate components:
j-ima-shadowgram-params (Standard)
j-ima-shadowgram-work (Standard)
j-isg-save-shd Sets the PI limits if REST events are requested
j-isg-bkg Calculate the predicted background
j-isg-chanlimits Sets the PI limits if REST events are requested
j-isg-get-hex-cell Find appropriate hexagonal cell
j-isg-index-clean Erase previous shadowgrams
j-isg-select-interval Get proper OBT interval
j-isg-cor2raw Transform corrrect positions to raw values
j-isg-build-01-shd Build shadowgram from corrected positions
j-isg-build-2-shd Build shadowgram from raw positions

j-isg-get-gtis Get the current Good Time Intervals

Dependencies from other components: Prepared and corrected data must exist in the SWG. The -BKG and -DBG data structures must have been created already by j-bkg-model

Pending TBDs: None

ISDC context: Scientific Analysis

Delivery date: Version 2.0.1/2004-12-01

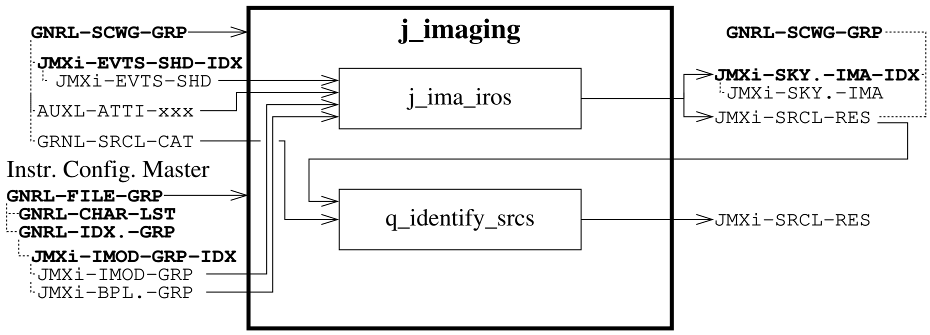

Figure 5.17: Pipeline application script j_imaging

5.13 j-imaging

Type: Analysis Script **Responsible:** ESAC (P. Kretschmar) and Valencia University (S. Martínez-Núñez)

Purpose: 18.18 18.19 18.20 18.21

Function: Creates sky images and variance maps for each shadowgram, searchs for sources, derives the fluxes of the found sources and indetifies them as known or new.

Inputs:
JMXi-EVTS-SHD (p. 236) (multiple shadowgrams)
AUXL-ATTI-PRE (p. 247) or
AUXL-ATTI-HIS (p. 247) (attitude data)
JMXi-IMOD-GRP (p. 239) (instrument description)

Outputs:
JMXi-SKY.-IMA (p. 243) (reconstructed images in sky coordinates)
JMXi-SRCL-RES (p. 245) (list of found sources with fluxes)

Parameters:

swgDOL p. 53	DOL of Science Window Group to be analyzed
jemxNum p. 46	JEM-X detector number
IC-Group p. 46	DOL of Instrument Characteristics master group
IC-Alias p. 46	Selection alias for Instrument Characteristics
instMod p. 46	DOL of JEM-X Instrument Model Group (empty=find in IC)
osimData p. 48	Are we using data produced by OSim?
chatter p. 42	Verbosity level; 2 = normal
clobber p. 42	Clobber existing output data?
subDir p. 53	Sub directory below the position of the SWG for results
makeNewBPL p. 47	Are new backprojection lists to be made ?
newBackProjFile p. 48	Base name of new backprojection file
detAccLimit p. 43	Acceptance limit in DETE-MOD array
skyImageDim p. 52	Sky image dimension 1:255, 2:511
useDeadAnodes p. 53	Include dead anodes ?
maxNumSources p. 47	Max number of sources in IROS loop
edgeEnhanceFactor p. 43	Edge enhancement factor
loopLimitPeak p. 47	Fractional peak power to stop IROS loop
detSigSingle p. 43	Detection significance for source acceptance
skyRadiusFactor p. 52	Sky radius limit factor
radiusLimit0 p. 49	[mm] Detector radius limit $E < 6$ keV
radiusLimit1 p. 49	[mm] Detector radius limit $6 < E < 12$ keV
radiusLimit2 p. 49	[mm] Detector radius limit $12 < E < 20$ keV
radiusLimit3 p. 50	[mm] Detector radius limit 20 keV $< E$
interactionDepth p. 46	[mm] Mean depth of interaction in the detector
hotPixelLimit p. 46	Hot pixel limit factor above average
skyImagesOut p. 52	Control output of sky images
dolBPL p. 43	BPL DOL
bkgShdDOL p. 41	DOL of background shadowgrams
relDist p. 50	Enter the maximally allowed value for reldist
fluxLimit p. 44	Lower flux limit for alerts of unfound sources [Counts/s^2]
searchRad p. 51	Range of search grid around nominal found source positions
gridNum p. 45	Number of grid steps on one side of search grid
distFuzz p. 43	'Fuzz' allowed in relative distances

Resources: See called executables

Processing:

Read in parameters.

Check if imaging data are available, if not, the executables will not be ran.
Call j-ima-iros to create deconvolved images, to find sources in the images and to estimate their fluxes.
Call q-identify-srcs to identify the sources.

Design remarks:

Parent components:
jemx-scw-analysis (p. 70)

Subordinate components:
j-ima-iros (p. 164)
q-identify-srcs (p. 163)

Dependencies from other components: Shadowgrams must have been created previously.

ISDC context: Science Analysis

Delivery date: v4.5.6 19/10/2005

5.13.1 j-ima-iros

Type: Executable **Responsible:** DNSC (Niels Lund and Niels J. Westergaard)

Purpose: 18.18 18.19

Function: Return reconstructed sky images in celestial coordinates by backprojection and iterative removal of sources.

Inputs:
JMXi-EVTS-SHD (p. 236) (shadowgram)
JMXi-EVTS-SHD-IDX (p. 236)
JMXi-IMOD-GRP (p. 239)
JMXi-BPL.-GRP (p. 233) (back projection lists)
AUXL-ATTI-HIS (p. 247) or
AUXL-ATTI-PRE (p. 247) (INTEGRAL attitude information)

Outputs:
JMXi-SKY.-IMA-IDX (p. 243)
JMXi-SKY.-IMA (p. 243) (celestial coord. system)
JMXi-SRCL-RES (p. 245)

Parameters:

inSWG p. 46	DOL of input SWG
outSWG p. 49	DOL of output SWG (override)
skyRes p. 52	DOL of resulting JMXi-SKY.-IMA (override)
listRes p. 47	DOL of resulting JMXi-SRCL-RES (override)
instMod p. 46	DOL of JMXi-IMOD-GRP (IC file)
detMapFile p. 43	File with detector map (obsolete)
dolBPL p. 43	DOL of JMXi BPL (backprojection list) group (override)
makeNewBPL p. 47	Do you want to make a new BPL group ?
newBackProjFile p. 48	Name of new BPL group file if requested
chatter p. 42	Level of debugging output
jemxNum p. 46	Number of JEM-X unit (1 or 2)
detAccLimit p. 43	Acceptance limit in DETE-MOD array
skyImageDim p. 52	Sky image dimension 1 $\rightarrow$ 255, 2 $\rightarrow$ 511, the latter is strongly adviced
bkgShdDOL p. 41	DOL of background shadowgrams (override if different from "empty")
useDeadAnodes p. 53	Include dead Anodes ?
maxNumSources p. 47	Max number of sources in IROS loop
edgeEnhanceFactor p. 43	Edge enhancement factor
loopLimitPeak p. 47	Fractional peak power to stop IROS loop

detSigSingle p. 43	Detection significance for source acceptance
yes2all p. 54	Disregard event selection energy boundaries
pixelFold p. 49	3:NL 1-D (obsolete)
pixelFoldFile p. 49	File with pixelfold information
anodePhase p. 40	Selection flag for anode phases def:0xFFFFF
rearPhase p. 50	Selection flag for rear-side phases def:0xFFFFF
doPhaseScan p. 43	Flag for activation of phase scan
pix3fact p. 49	Amplification factor for pixelfold 3
skyRadiusFactor p. 52	Sky radius limit factor
radiusLimit0 p. 49	[mm] Detector radius limit $E < 6$ keV
radiusLimit1 p. 49	[mm] Detector radius limit $6 < E < 12$ keV
radiusLimit2 p. 49	[mm] Detector radius limit $12 < E < 20$ keV
radiusLimit3 p. 50	[mm] Detector radius limit 20 keV $< E$
interactionDepth p. 46	[mm] Mean depth of interaction in the detector
hotPixelLimit p. 46	Hot pixel limit factor above average
fsTwistJ1 p. 45	[rad] JEMX1 frontside twist
rsTwistJ1 p. 51	[rad] JEMX1 readside twist
fsTwistJ2 p. 51	[rad] JEMX2 frontside twist
rsTwistJ2 p. 51	[rad] JEMX2 readside twist
shdOutCtrl p. 51	Control output of e.g. calculated source shadowgrams
skyImagesOut p. 52	Control output of sky images
onlyDoShd p. 48	Skip sky image making
doClean3 p. 43	Use clean3 ?
clean3size p. 42	[pix] Half size of clean3 region
cleanStep p. 42	[pix] step for cleaning of stripes
eeSelect0 p. 43	Flag for ee=0 band selection from 16 bands
eeSelect1 p. 43	Flag for ee=1 band selection from 16 bands
eeSelect2 p. 43	Flag for ee=2 band selection from 16 bands
eeSelect3 p. 44	Flag for ee=3 band selection from 16 bands
useHexpos p. 53	Flag for employing function hexpos
useTrace p. 54	Flag for tracing function calls

Resources: Uses about 500 MB of memory. The processing time depends mainly on remote access to the backprojection lists stored as separate IC files `JMXi-BPL.-GRP`. It is thus advised to have these files on the same disk as where the program is run.

Processing:

The *j_ima_iros* software package provides two main outputs:

- sky images (and corresponding variance maps) derived from JEM–X shadowgrams.

- a list of source candidates extracted from the sky images during the IROS-phase.

j_ima_iros can process up to 15 shadowgrams from one science window at a time. (The different shadowgrams will typically correspond to different energy channels). For an initial survey of the sky field it is recommended to divide the data into a small number (∼4) of energy bands to increase the sensitivity for faint sources.

j_ima_iros generates its sky images by a simple backprojection algorithm. This requires an Aperture Response Function which for every shadowgram pixel describes which sky pixels may illuminate this shadowgram pixel. It is a lengthy process to calculate the full set of Aperture Response Functions for all 511×511 sky pixels, and therefore this set is calculated off line by a set of ray tracing subroutines. The same set of routines is invoked (but only for one specific sky position) for every new source candidate treated in the IROS phase.

Even when stored in compressed format the Aperture Response Function "backprojection lists" are of substantial size: about 375 Mbyte for each JEM–X unit for sky images with 511×511 pixels (Pixel size: 1.5 arcmin). But the advantage is that the backprojection process (which must be repeated many times during the IROS-process) is quite fast — and the duration of the backprojection step does not depend on the complexity of the instrument model.

The backprojection lists are stored as separate IC files `JMXi-BPL.-GRP`.

5.13.1.1 IROS step by step

The *work* subroutine is the backbone of *j_ima_iros*. After calling a number of initialization routines it reads all the shadowgrams to be processed.

It then calls the *irosloop* routine which go through the following steps:

1. Call the *backpro_d* routine, which performs backprojection of the raw input shadowgram to form the variance map - this is a sky image which contains in each pixel the total number of shadowgram photons which geometrically could have arrived from the corresponding sky point.

2. Initialize two illumination functions describing background components. One of these is an empty field shadowgram averaged over many science windows without strong sources. The other background component is a constant, its primary purpose is to guarantee that the mean value of the residual shadowgram after background subtraction is zero. Having a mean value of zero for the shadowgram counts reduces the magnitude of the large scale features in the sky images.

3. Call the *imagefit* routine, which fits the raw shadowgram with the two background components. *imagefit* returns the solution, i.e. the power in each of the two components and a residual shadowgram with a mean of zero counts.

4. Call the *backpro_d* routine again, now with the residual shadowgram to form the first real sky image.

5. Call the *clean* routine to remove the large scale features from the sky image.

The backprojection performed in *j_ima_iros* is straightforward, and it produces sky images with significant large scale structures even when working from a residual shadowgram with a mean value of zero counts. The large scale features must be removed before a sensitive search for source peaks can be performed on the image. This is done in the *clean* stage of the IROS process.

At this stage, before any source finding has taken place, a useable sky image and a variance map exist. If there are strong sources in the field of view, it will be better, also for mosaics, if the contributions from the strong sources are removed from the images. The final images will have a reduced level of systematic noise. The statistical noise, however, always remains and its level will be given by the original variance map.

Thus the residual sky image with the fitted strength of the two background components subtracted is the starting point for the Iterative Removal Of Sources phase:

6 Call the *findpeak* routine to identify the highest peak in the cleaned sky image. By default *findpeak* will work on a 'significance' map, where the sources at the edge of the image are enhanced because the absolute variance is decreasing towards the edge (the relative variance is strongly increasing).

7 Check if the new peak is close to a peak found in a previous IROS cycle. If so, the position of the peak is adjusted to take into account the new detection of the same peak. A very strong source may continue to reappear in *findpeak* due to inaccurate modelling of the detector response; after five recurrences the peak position (in the sky image - not in the shadowgram!) will be marked as "dont use" for future IROS cycles on this dataset. The source is then safely detected and the position will not improve significantly with further adjustments.

8 Call the *detectmap* routine which generates a "pixel illumination function" for the new source (position) via a full raytracing of the detector illumination from this source position. (Note that even if the raytracing is done ahead of time for all pixels in the standard 511×511 pixel sky image, it must be done from scratch for every new source position because these new positions are in general not coincident with a standard pixel position).

9 Call the *imagefit* routine with the full set of background and source components identified up to this level. In every cycle the fit is done based on the original shadowgram. *imagefit* returns the solution, with the fit power of all the components and a residual shadowgram with a mean of zero counts.

10 If the power associated to the latest source is less than a few percent (determined by parameter `IMA_loopLimitPeak`) of the total number of counts in the shadowgram the IROS iteration process will be terminated. It will also be terminated when 20 IROS cycles have

been executed. When the IROS cycle is terminated the control passes back from *irosloop* to the *work* subroutine.

11 If the IROS process is not terminated for any of the reasons given above *backpro_d* will be called again with the updated residual shadowgram to generate a new sky image.

12 *clean* is called to remove the large scale features from the sky image.

13 Go back to point 6 and repeat the IROS cycle.

5.13.1.2 Final image and source list generation

When control passes back from *irosloop* to *work* the results of the analysis of this shadowgram will be saved and *work* will check if there are further shadowgrams to process for the current science window.

When all shadowgrams have been processed the *work* routine will go over the results from the individual shadowgrams and will check for multiple detections of the same source. All multiply detected sources will be kept in a list of "reliably detected" sources (regardless of their strength). Additionally, sources which have been detected only in one energy band, but with a strength exceeding a given limit (defined by parameter `IMA_detSigSingle`) will be placed also in the list of reliably detected sources.

The *work* routine will then do a final *imagefit* of all shadowgrams using the full set of reliably detected sources. It will do a final backprojection of all the residual shadowgrams, and a final cleaning of the resulting sky images. To restore the peaks (but not the consequent image artefacts) of those sources which have removed during this process, the peaks from the first image (from step 5 in the above description) will be literally copied back into the final images. The resulting images are the recommended input for further mosaic images.

It must be realized that none of the methods for choosing the "reliably detected" sources are foolproof, random double detections do happen, and particularly strong sources can produce artefacts exceeding an otherwise safe threshold for source detection. So one should be extremely cautious about the strong single detections - particularly when other strong sources are in the field of view.

The fit results for all safely detected sources will also be output. It frequently happens that during the last run of *imagefit* with the full set of reliably detected sources some of the sources are assigned a zero flux or even a negative flux in certain energy bands. This is perfectly legal and statistically acceptable. There is no guarantee that a given source has a nonzero flux in all energy bands and statistical fluctuations may produce negative fluxes for weak sources. Get the mask description (JMXi-FMSK-MOD) from instMod (JMXi-IMOD-GRP)
Find and open the SWG
Find and open all shadowgram data structures (JMXi-EVTS-SHD) found
with keyword SHDTYPE == 2 (two)
for all shadowgrams do {

Call function j-ifr-a-recon for reconstruction by convolution (function j-ifr-a-convol)
Write JMXi-SKY.-IMA to disk
}
Close files

Parent components:
j-imaging (p. 161)

Subordinate components:
j-ima-iros-params (standard)
j-ima-iros-prepare
j-ima-iros-work (standard)
logprint.c
j-iir-get-shds.c
j-iir-dpos-attlen.c
backpro8-init.c
backpro8-d.c
j-iir-read-bpl.c
j-iir-get-radec-init.c
j-iir-get-bkgshds.c
backsky-init.c
pixelfold.c
rmscalc.c
clean.c
clean2.c
peaktransfer.c
alignment-init.c
outfit-isdc.c
irosloop.c
fun.c
findpeak.c
imagefit.c
gaussj-nr.c
skyval.c
kwds-funcs.c
kwds-write.c
general-funcs.c
fits-test-output.c
skymap-gen.c
skymap.c

skymap-init.c
mask-test.c
snapshot1.c
snapshot2.c
j-iir-write-srclres.c

Functions from jmx_lib_pif
getxy.c
j-iir-get-radec-init.c
jemxlib-get-iros-imod-data.c
jemxlib-detector-get-map.c
jemxlib-get-mask.c
jemxlib-array-put.c
detectmap.c
getRAdec.c
sh-xy.c
hotpix-chk.c
backsky-init.c
shapefold.c
sh-depth.c
shapecomb.c
sign.c
m-strukdef.c
hexpos.c

Dependencies from other components: One or more shadowgrams must exist before this executable is called.

ISDC context: Scientific Analysis

Delivery date: Version 1.5.0/2005-07-14

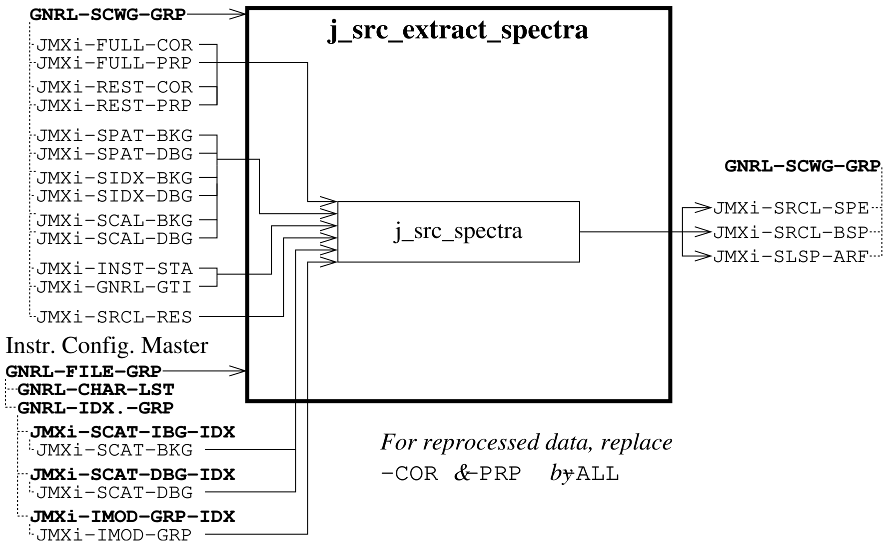

Figure 5.18: Pipeline application script j_src_extract_spect

5.14 j-src-extract-spectra

Type: Analysis Script **Responsible:** ESAC (P. Kretschmar) and Valencia University (S. Martínez-Núñez)

Purpose: Fulfill items 18.23.1 and 18.25 in SRD

Function: Derive spectra, background spectra and ancillary response functions for individual sources.

Inputs:
JMXi-FULL-PRP (p. 237) (event data)
JMXi-FULL-COR (p. 237)
JMXi-REST-PRP (p. 241)
JMXi-REST-COR (p. 241)
JMXi-FULL-ALL (p. 237) (event data, new scheme)
JMXi-REST-ALL (p. 241)
JMXi-INST-STA (p. 239) (instrument status & grey filter)
JMXi-GNRL-GTI (p. 238) (good-time intervals)
JMXi-IMOD-GRP (p. 239) (instrument description)
JMXi-SIDX-DBG (p. 243) (background model data)
JMXi-SPAT-DBG (p. 243)
JMXi-SCAL-DBG (p. 242)

JMXi-SIDX-BKG (p. 242)
JMXi-SPAT-BKG (p. 243)
JMXi-SCAL-BKG (p. 242)
JMXi-SCAT-DBG (p. 242) (background spectra catalogs, *not* in the swgDOL)
JMXi-SCAT-BKG (p. 242)
JMXi-SRCL-RES (p. 245)

Outputs:
JMXi-SRCL-SPE (p. 245) (spectra for multiple sources)
JMXi-SRCL-BSP (p. 245) (corresponding bkg spectra)
JMXi-SRCL-ARF (p. 245) (Ancillary response functions for sources)

Parameters:

swgDOL p. 53	Science Window Group
jemxNum p. 46	JEM-X detector number (1 or 2)
IC-Group p. 46	Instrument Configuration Master Group
IC-Alias p. 46	Instrument Configuration version alias
instMod p. 46	DOL of JEM-X Instrument Model Group (empty=find in IC)
evtType p. 44	Event type (0=FULL, 1=REST, -1=both)
response p. 50	Filename of standard RMF and ARF (empty=find in IC)
osimData p. 48	Are we using data produced by OSim?
chatter p. 42	Verbosity level (2=normal)
clobber p. 42	Clobber existing output data?
subDir p. 53	Sub directory below the position of the SWG for results
timeStep p. 53	Length of integration time bins
timeStart p. 53	Time start in IJD of first spectrum (<0 = use SWG start)
timeStop p. 53	Time start in IJD of last spectrum (<0 = use SWG start)
bgrMethod p. 41	Background subtraction method (0=Flat,1=Model)
vignCorr p. 54	Correct for vignetting?
gtiNames p. 45	Names of GTI tables to be used
nPhaseBins p. 48	Number of phase folding bins
phaseBins p. 49	Fractional length of phase bins
precisionLevel p. 49	Speed - precision level (1-100, 0=Default)
tAccuracy p. 53	Time Correlation accuracy (0=high, 1=low, 3=any)
rowSelect p. 51	CFITSIO selection string on events (optional)
radiusLimit p. 49	[mm] Use events within this detector radius
useRaDec p. 53	Use sky coord (Ra,Dec) instead of instr. coord. (y/n)?
fluxScaling p. 44	Flux scaling (1=Open area,2=Detector area,3=Per square cm)

skipNearDeadAnode p. 51	Ignore pixels near dead anodes (y/n)?
skipHotSpot p. 51	Ignore pixels in hotspot area (y/n)?
overrideCollTilt p. 49	Collimator tilt Parameter? (if <0 use IMOD)
storePif p. **??**	Store PIF data structure (1=Events 2=Events detector)?
j-src-spectra-fullPrp p. 44	Full Imaging Prepared Data DOL (override)
j-src-spectra-fullCor p. 44	Full Imaging Prepared Data DOL (override)
j-src-spectra-instStat p. 46	Instrument Status Data DOL (override)
j-src-spectra-deadTime p. 43	Deadtime History Table DOL (override)
j-src-spectra-goodTime p. 45	Good-Time Interval Table DOL (override)
j-src-spectra-diagnosticMode p. 43	Diagnostic Mode (0=Normal, ¿1=Diagnostic)
j-src-spectra-diagnosticParam p. 43	Diagnostic Parameter?

Note: parameters with a prefix of "j_src_spectra" are override parameters for data normally found in the Science Window Group, which are passed on to **j_src_spectra**. Normally they are hidden and set to empty (no effect).

Resources: See called executable

Processing:

Read in parameters
Check if we have spectral data, if not, the executable will be not ran
Get DOLs to the JMXi-SCAT-... data structures from master IC group
Get DOL to the JMXi-IMOD-GRP data structure from master IC group
Call j-src-spectra to extract spectra, bkg spectra and ARFs.

Design remarks:
Currently, this makes no use of the background model data.

Parent components:
jemx-scw-analysis (p. 70)

Subordinate components:
j-src-spectra (p. 175)

Dependencies from other components: Events must have been corrected.
Background model data must have been extracted.

ISDC context: Science Analysis

Delivery date: v4.5.6 19/10/2005

5.14.1 j-src-spectra

Type: Executable **Responsible:** Stockholm

Purpose: Fulfill item 18.24 in SRD

Function: Extract background subtracted count rate spectra for individual sources in imaging data.

Interfaces: Input data (FULL and/or REST mode) is from swgDOL. Also status, greyfilter, dead time, and GTI data are from swgDOL.
The output datastructures are either in the outswgDOL or created according to the override DOLs (outSrcSpec, outBkgSpec, outSrcArf

Inputs:
JMXi-IMOD-GRP (p. 239)
JMXi-DPOS-MOD (p. 235) In JMXi-IMOD-GRP
JMXi-VIGN-MOD (p. 246) In JMXi-IMOD-GRP
JMXi-MASK-MOD (p. 239) In JMXi-IMOD-GRP
JMXi-DETE-MOD (p. 235) In JMXi-IMOD-GRP
JMXi-FBDS-MOD (p. 236) In JMXi-IMOD-GRP
JMXi-COLL-MOD (p. 233) In JMXi-IMOD-GRP
JMXi-INST-STA (p. 239) In input SWG
JMXi-DEAD-SCP (p. 234) In input SWG
JMXi-SCAL-BKG (p. 242) In input SWG
JMXi-SCAL-DBG (p. 242) In input SWG
JMXi-SPAT-BKG (p. 243) In input SWG
JMXi-SPAT-DBG (p. 243) In input SWG
JMXi-SCAT-BKG (p. 242)
JMXi-SIDX-BKG (p. 242) In input SWG
JMXi-SCAT-DBG (p. 242)
JMXi-SIDX-DBG (p. 243) In input SWG
JMXi-GNRL-GTI (p. 238)
JMXi-CORX-MOD (p. 234) In input SWG
JMXi-CORY-MOD (p. 234) In input SWG
JMXi-SRCL-RES (p. 245) In input SWG
JMXi-FULL-COR (p. 237) In input SWG
JMXi-REST-COR (p. 241) In input SWG
JMXi-FULL-PRP (p. 237) In input SWG
JMXi-REST-PRP (p. 241) In input SWG
JMXi-AXIS-ARF (p. 233) In input response

Outputs:
JMXi-SRCL-ARF (p. 245) In output SWG

JMXi-SRCL-SPE (p. 245) In output SWG
JMXi-SRCL-BSP (p. 245) In output SWG

Parameters:
INPUT GROUPS:
inSWG p. 46 DOL of input SWG
outSWG p. 49 DOL of output SWG
instMod p. 46 DOL of instrument model group JMXi-IMOD-GRP
bkgSpecCat p. 41 DOL of instr bkg spectral catalogue JMXi-SCAT-BKG
dbgSpecCat p. 42 DOL of diffuse bkg spectral catalogue JMXi-SCAT-DBG
goodTime p. 45 DOL of Good-Time Interval Table (override)
deadTime p. 42 DOL of Deadtime History Table (override)
instStat p. 46 DOL of Instrument Status Data (override)
fullPrp p. 44 DOL of full Imaging Prepared Data (override)
fullCor p. 44 DOL of full Imaging Corrected Data (override)
dbgSpecMap p. 42 DOL of diffuse backgr. spectral indices
dbgSpatial p. 42 DOL of diffuse backgr. spatial distribution
dbgScale p. 42 DOL of diffuse backgr. scaling factors
bkgSpecMap p. 41 DOL of instrument backgr. spectral indices
bkgSpatial p. 41 DOL of instrument backgr. spatial distribution
bkgScale p. 41 DOL of instrument backgr. scaling factors
OTHER INPUTS:
jemxNum p. 46 Number of JEM-X unit (1 or 2)
bgrMethod p. 41 Background subtraction method (0=Flat,1=Model)
diagnosticMode p. 43 Diagnostic Mode (0=Normal, ¿1-4=Diagnostic)
diagnosticParam p. 43 Diagnostic Parameter
evtType p. 44 0 for FULL events, 1 for REST events.
fluxScaling p. 44 Flux scaling to area (1=Open 2=100cm-square 3=cm-square)
timeStart p. 53 Time at beginning of (first) spectrum.
timeStop p. 53 End time of data used for extraction.
timeStep p. 53 Time integration step (seconds). (¡0 to use all data)
nPhaseBins p. 48 Number of bins in phaseBins.
phaseBins p. 49 Length of phase bins (in case of cyclic binning).
precisionLevel p. 49 Precision level, 0-100, Only 20 and 25 in use.
skipNearDeadAnode p. 51 Ignore pixels next to dead anodes
skipHotSpot p. 51 Ignore pixels in hotspot area?
radiusLimit p. 49 [mm] Use events within this detector radius
response p. 50 Path and name of file with standard RMF and ARF
rowSelect p. 51 CFITSIO selection string on events (optional)
tAccuracy p. 53 Time Correlation accuracy (0=high, 1=low, 3=any)
vignCorr p. 54 Should program perform vignetting corrections?

gtiNames p. 45 Names of GTI tables to be used
useRaDec p. 53 Use sky coord (Ra,Dec) instead of instr. coord?
chatter p. 42 level of debugging output
clobber p. 42 Clobber existing data in output data structures?
OVERRIDES:
outBkgSpec p. 48 DOL of override JMXi-SRCL-BSP
outSrcSpec p. 48 DOL of override JMXi-SRCL-SPE
outSrcArf p. 48 DOL of override JMXi-SRCL-ARF

Resources: Execution time depends mostly on the number of sources. precisionLevel=25 takes approx. 50% longer than precisionLevel=20. Memory requirement is dominated by input event data.

Processing:
Read input parameters
Read instrument model
Read instrument status
Read source parameters
Loop over sources
    Loop over energy bins
        Calculate instrument response for source
        (Pixel Illumination Fraction)
        Select Open-Closed PIF cut value
    End loop over energy bins
End loop sources
Read background model
Loop over sources
    Loop over energy bins
        Calculate net background contribution (spatial and energy parts)
    End loop over energy bins
End loop over sources

If(evtType == 0) Read FULL event data
If(evtType == 1) Read REST event data

Calculate time ranges for spectra.

Loop over events
    Apply weight for greyfilter and deadtime
    Loop over sources
        Add event to open or closed counts in time-energy bin.

End loop sources
End loop over events

Loop over time bins and sources
Loop over energy bins
Solve for net source and background rate
End energy bins
End loop over time bins and sources

Normalize rates to per $100\mathrm{cm}^2$ Write source spectra to JMXi-SRCL-SPE
Write backgr. spectra to JMXi-SRCL-BSP

Design remarks:
The extraction is made by analysing the flux difference between the detector elements that are illuminated by a source and those that are not. For a given source position those areas are deduced by ray tracing through the instrument. The contrast between the illuminated (open) and non-illuminated (closed) detector area is reduced by the limited resolution of the detector. This ”leakage” is corrected for.

For each source a pixel illumination map of the detector is constructed. Pixels with an illumination fraction (PIF) larger than a cut value will be regarded as open elements and those with PIFs below a lower cut value will be regarded as closed elements and used for the background estimate. Each detector pixel, with proper size and shape is divided into 16 sub pixels. For each sub pixel the line of sight towards the sources is followed through the collimator and coded mask. When a source is illuminating a sub pixel a probability distribution for event detection is added to the nearby pixels. By scanning over the full detector area, excluding non-used areas (e.g. dead anodes) a detector illumination pattern is built up for each source. Source extraction then proceeds by counting the number of events in the pixels with PIF values above and below the cut levels. After area normalization, background subtraction and leakage correction, the extracted count rate per time and energy bin is stored as source spectra in JMXi-SRCL-SPE. Background spectra are stored in JMXi-SRCL-BSP for evaluation purposes. From version 4.0.0 the lower and upper PIF cut levels are identical and selected so that the net leakage out of the selected open area will be approximatly 25 %.

PIF-maps are computed at a number of energies, 10 for precisionLevel=20 and 18 for precisionLevel=25. PrecisionLevel 20 has been found to be sufficient in all tested cases.

Either FULL or REST imaging events will be used. The present version ONLY allow the FULL imaging option.
Phase binning is not yet implemented.
The possibility to choose alternative background methods (e.g. fitting) is forseen, but not imple-

mented. Only the flat background option is provided. Background subtraction is intrinsic to the extraction process.

Error estimates for both extraction methods are based on the assumption of poisson statistics and many counts per bin. For spectra computed with integration times << a typical science window length the number of counts per bin can be small and these error estimates are less reliable. Sources in the outer field of view are more sensitive to this since they only illuminate a small fraction of the detector.

Subordinate components: Functions of the executable:
j-src-work
j-src-params
j-src-array-all
j-src-mask
j-src-mask-find-element
j-src-oddeven
j-mask-findxy
j-eventweight
j-src-through-mask
j-src-through-coll
j-src-background
j-src-nspec
j-src-nspec-times
j-src-grey-correction
j-src-vignetting
j-src-interpol
j-src-interval-ave
j-src-fill-gauss
j-src-in-hole
j-src-add-array
j-src-leakage
j-src-maskhole-sample
j-src-maskhole-average
j-src-leak-coefficients
j-src-WriteLC
j-src-wcs-rsti
j-src-crossprod
j-src-randXY
j-src-openfraction
j-src-event-add

j-src-event-add-0
j-src-event
j-src-pixel-area
j-src-pif-sort
j-src-sort
j-src-dete-index
j-src-get-bit
j-min-range
j-gti-overlap
j-read-GTI
j-ssp-GetFileName
j-src-get-pointing
j-src-pif-interpol
j-src-factors-interpol
j-src-ext-crab
j-src-pif-create20
j-src-solve-rate
j-src-pifcut-interpol

Dependencies from other components: The following components should be run before j-src-spectra:
j-data-preparation
j-calib-auto
j-correction
j-bkg-model (or select flat background)
j-image-sa-generation
j-src-find

ISDC context: Scientific Analysis

Delivery date:
Version 1.0 delivered 7 Sep 2000
Version 4.1.0 delivered 15 Jun 2004

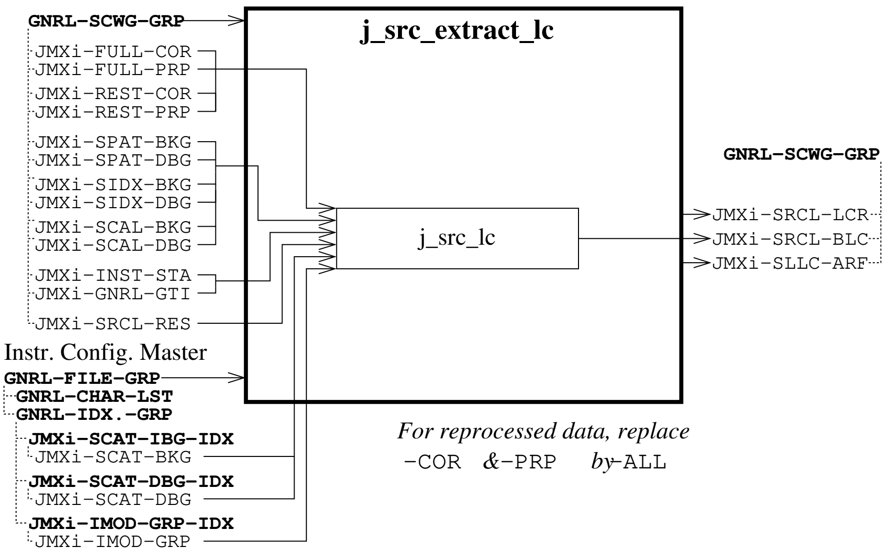

Figure 5.19: Appplication script j_src_extract_lc

5.15 j-src-extract-lc

Type: Analysis Script **Responsible:** ESAC (P. Kretschmar) and Valencia University (S. Martínez-Núñez)

Purpose: Fulfill items 18.23.1 and 18.24 in SRD

Function: Derive individual source and background lightcurves

Inputs:
JMXi-FULL-PRP (p. 237) (event data)
JMXi-FULL-COR (p. 237)
JMXi-REST-PRP (p. 241)
JMXi-REST-COR (p. 241)
JMXi-FULL-ALL (p. 237) (event data, new scheme)
JMXi-REST-ALL (p. 241)
JMXi-INST-STA (p. 239) (instrument status & grey filter)
JMXi-GNRL-GTI (p. 238) (good-time intervals)
JMXi-IMOD-GRP (p. 239) (instrument description)
JMXi-SIDX-DBG (p. 243) (background model data)
JMXi-SPAT-DBG (p. 243)
JMXi-SCAL-DBG (p. 242)
JMXi-SIDX-BKG (p. 242)

JMXi-SPAT-BKG (p. 243)
JMXi-SCAL-BKG (p. 242)
JMXi-SCAT-DBG (p. 242) (background spectra catalogs, *not* in the swgDOL)
JMXi-SCAT-BKG (p. 242)
JMXi-SRCL-RES (p. 245)

Outputs:
JMXi-SRC.-LCR (p. 245) (lightcurves for multiple sources)

Parameters:

swgDOL p. 53	Science Window Group
jemxNum p. 46	JEM-X detector number (1 or 2)
IC-Group p. 46	Instrument Configuration Master Group
IC-Alias p. 46	Instrument Configuration version alias
instMod p. 46	DOL of JEM-X Instrument Model Group (empty=find in IC)
response p. 50	Filename of standard RMF and ARF (empty=find in IC)
evtType p. 44	Event type (0=FULL, 1=REST, -1=both)
gtiNames p. 45	Names of GTI tables to be used
osimData p. 48	Are we using data produced by OSim?
chatter p. 42	Verbosity level
clobber p. 42	Clobber existing output data?
subDir p. 53	Sub directory below the position of the SWG for results
timeStart p. 53	Time start in IJD of first spectrum (<0 = use SWG start)
timeStep p. 53	Length of integration time bins
timeStop p. 53	Time start in IJD of last spectrum (<0 = use SWG start)
nChanBins p. 47	Number of channel bins (0=bin all, -1=full resolution)
chanLow p. 41	Lower channel bounds (numbers separated by spaces)
chanHigh p. 41	Upper channel bounds (numbers separated by spaces)
nPhaseBins p. 48	Number of phase folding bins
phaseBins p. 49	Fractional length of phase bins
precisionLevel p. 49	Speed - precision level (1-100, 0=Default)
tAccuracy p. 53	Time Correlation accuracy (0=high, 1=low, 3=any)
bgrMethod p. 41	Background subtraction method (0=Flat,1=Model)
vignCorr p. 54	Correct for detector vignetting?
rowSelect p. 51	CFITSIO selection string on events (optional)
radiusLimit p. 49	[mm] Use events within this detector radius
useRaDec p. 53	Use sky coord (Ra,Dec) instead of instr. coord. (y/n)?

fluxScaling p. 44	Flux scaling (1=Open area,2=Detector area,3=Per square cm)
skipNearDeadAnode p. 51	Ignore pixels near dead anodes (y/n)?
skipHotSpot p. 51	Ignore pixels in hotspot area (y/n)?
overrideCollTilt p. 49	Collimator tilt Parameter? (if <0 use IMOD)
j-src-lc-fullPrp p. 44	Full Imaging Prepared Data DOL
j-src-lc-fullCor p. 44	Full Imaging Prepared Data DOL
j-src-lc-instStat p. 46	Instrument Status Data DOL
j-src-lc-deadTime p. 43	Deadtime History Table DOL
j-src-lc-goodTime p. 45	Good-Time Interval Table DOL
j-src-lc-diagnosticMode p. 43	Diagnostic Mode (0=Normal, >1=Diagnostic)
j-src-lc-diagnosticParam p. 43	Diagnostic Parameter?

Note: parameters with a prefix of "j_src_lc" are override parameters for data normally found in the Science Window Group, which are passed on to **j_src_lc**. Normally they are hidden and set to empty (no effect).

Resources: Nothing special required

Processing:

Read in parameters
Check if we have lightcurve data, if not, the executable will be not ran
Get DOLs to the JMXi-SCAT-... data structures from master IC group
Get DOL to the JMXi-IMOD-GRP data structure from master IC group
Call j-src-lc to extract lightcurves.

Design remarks:
Currently, this makes no use of the background model data.

Parent components:
jemx-scw-analysis (p. 70)

Subordinate components:
j-src-lc (p. 185)

Dependencies from other components: Events must have been corrected.

ISDC context: Science Analysis

Delivery date: v4.5.6 19/10/2005

5.15.1 j-src-lc

Type: Executable **Responsible:** Stockholm

Purpose: Fulfill item 18.25 in SRD

Function: Extract background subtracted light curves for individual sources in imaging data.

Interfaces: Input data (FULL and/or REST mode) is from swgDOL. Also status, greyfilter, dead time, and GTI data are from swgDOL.
The output datastructures are either in the outswgDOL or created according to the override DOLs (outSrcLC

Inputs:
JMXi-IMOD-GRP (p. 239)
JMXi-DPOS-MOD (p. 235) In JMXi-IMOD-GRP
JMXi-VIGN-MOD (p. 246) In JMXi-IMOD-GRP
JMXi-MASK-MOD (p. 239) In JMXi-IMOD-GRP
JMXi-DETE-MOD (p. 235) In JMXi-IMOD-GRP
JMXi-FBDS-MOD (p. 236) In JMXi-IMOD-GRP
JMXi-COLL-MOD (p. 233) In JMXi-IMOD-GRP
JMXi-INST-STA (p. 239) In input SWG
JMXi-DEAD-SCP (p. 234) In input SWG
JMXi-SCAL-BKG (p. 242) In input SWG
JMXi-SCAL-DBG (p. 242) In input SWG
JMXi-SPAT-BKG (p. 243) In input SWG
JMXi-SPAT-DBG (p. 243) In input SWG
JMXi-SCAT-BKG (p. 242)
JMXi-SIDX-BKG (p. 242) In input SWG
JMXi-SCAT-DBG (p. 242)
JMXi-SIDX-DBG (p. 243) In input SWG
JMXi-GNRL-GTI (p. 238)
JMXi-CORX-MOD (p. 234) In input SWG
JMXi-CORY-MOD (p. 234) In input SWG
JMXi-SRCL-RES (p. 245) In input SWG
JMXi-FULL-COR (p. 237) In input SWG
JMXi-REST-COR (p. 241) In input SWG
JMXi-FULL-PRP (p. 237) In input SWG
JMXi-REST-PRP (p. 241) In input SWG

Outputs:
JMXi-SRC.-LCR-IDX (p. 245) In output SWG
JMXi-SRC.-LCR (p. 245) In output SWG
JMXi-SRC.-FLC-IDX (p. 244) In output SWG

JMXi-SRC.-FLC (p. 244) In output SWG

Parameters:
INPUT GROUPS:
inSWG p. 46 DOL of input SWG
outSWG p. 49 DOL of output SWG
instMod p. 46 DOL of instrument model group JMXi-IMOD-GRP
bkgSpecCat p. 41 DOL of instr bkg spectral catalogue JMXi-SCAT-BKG
dbgSpecCat p. 42 DOL of diffuse bkg spectral catalogue JMXi-SCAT-DBG
goodTime p. 45 DOL of Good-Time Interval Table (override)
deadTime p. 42 DOL of Deadtime History Table (override)
instStat p. 46 DOL of Instrument Status Data (override)
fullPrp p. 44 DOL of full Imaging Prepared Data (override)
fullCor p. 44 DOL of full Imaging Corrected Data (override)
dbgSpecMap p. 42 DOL of diffuse backgr. spectral indices
dbgSpatial p. 42 DOL of diffuse backgr. spatial distribution
dbgScale p. 42 DOL of diffuse backgr. scaling factors
bkgSpecMap p. 41 DOL of instrument backgr. spectral indices
bkgSpatial p. 41 DOL of instrument backgr. spatial distribution
bkgScale p. 41 DOL of instrument backgr. scaling factors
OTHER INPUTS:
jemxNum p. 46 Number of JEM-X unit (1 or 2)
bgrMethod p. 41 Background subtraction method (0=Flat,1=Model)
nChanBins p. 47 Number of energy bands for light curves
chanLow p. 41 Lower PI channels [nChanBins values]
chanHigh p. 41 Upper PI channels [nChanBins values]
diagnosticMode p. 43 Diagnostic Mode (0=Normal, ¿1-4=Diagnostic)
diagnosticParam p. 43 Diagnostic Parameter
evtType p. 44 0 for FULL events, 1 for REST events.
fluxScaling p. 44 Flux scaling to area (1=Open 2=100cm-square 3=cm-square)
gtiNames p. 45 Names of GTI tables to be used
timeStart p. 53 Time at beginning of (first) lightcurve bin.
timeStop p. 53 End time of data used for extraction.
timeStep p. 53 Time integration step (seconds).
bgrMethod p. 41 Choice of backgr. method (0=flat, 1=model).
nPhaseBins p. 48 Number of bins in phaseBins.
phaseBins p. 49 Length of phase bins (in case of cyclic binning).
precisionLevel p. 49 Precision level, 0-100, Only 20 and 25 in use.
skipNearDeadAnode p. 51 Ignore pixels next to dead anodes
skipHotSpot p. 51 Ignore pixels in hotspot area?
radiusLimit p. 49 [mm] Use events within this detector radius

response p. 50 Path and name of file with standard RMF and ARF
rowSelect p. 51 CFITSIO selection string on events (optional)
tAccuracy p. 53 Time Correlation accuracy (0=high, 1=low, 3=any)
vignCorr p. 54 Should program perform vignetting corrections?
useRaDec p. 53 Use sky coord (Ra,Dec) instead of instr. coord?
chatter p. 42 level of debugging output
clobber p. 42 Clobber existing data in output data structures?
OVERRIDES:
outSrcLC p. 48 DOL of override JMXi-SRC.-LCR

Resources: Execution time depends mostly on the number of sources. precisionLevel=25 takes approx. 50% longer than precisionLevel=20. Memory requirement is dominated by input event data.

Processing:
Read input parameters
Read instrument model
Read instrument status
Read source parameters
Loop over sources
    Loop over energy bins
        Calculate instrument response for source
        (Pixel Illumination Fraction)
        Select Open-Closed PIF cut value
    End loop over energy bins
End loop sources
Read background model
Loop over sources
    Loop over energy bins
        Calculate net background contribution (spatial and energy parts)
    End loop over energy bins
End loop over sources

If(evtType == 0) Read FULL event data
If(evtType == 1) Read REST event data

Calculate times for lightcurve bins.

Loop over events
    Apply weight for greyfilter and deadtime
    Loop over sources
        Add event to open or closed counts in time-energy bin.

End loop sources
End loop over events

Loop over time bins and sources
Loop over energy bins
Solve for net source and background rate
End energy bins
End loop over time bins and sources

Normalize rates to per $100\mathrm{cm}^2$ Write source ans backgr. lightcurves to JMXi-SRC.-LCR

Design remarks:
The extraction is made by analysing the flux difference between the detector elements that are illuminated by a source and those that are not. For a given source position those areas are deduced by ray tracing through the instrument. The contrast between the illuminated (open) and non-illuminated (closed) detector area is reduced by the limited resolution of the detector. This ”leakage” is corrected for.

For each source a pixel illumination map of the detector is constructed. Pixels with an illumination fraction (PIF) larger than a cut value will be regarded as open elements and those with PIFs below a lower cut value will be regarded as closed elements and used for the background estimate. Each detector pixel, with proper size and shape is divided into 16 sub pixels. For each sub pixel the line of sight towards the sources is followed through the collimator and coded mask. When a source is illuminating a sub pixel a probability distribution for event detection is added to the nearby pixels. By scanning over the full detector area, excluding non-used areas (e.g. dead anodes) a detector illumination pattern is built up for each source. Source extraction then proceeds by counting the number of events in the pixels with PIF values above and below the cut levels. After area normalization, background subtraction and leakage correction, the extracted count rate per time and energy bin is stored as source lightcurves in JMXi-SRC.-LCR. For evaluation purposes the background lightcurves are also stored in the same data structure. From version 4.0.0 the lower and upper PIF cut levels are identical and selected so that the net leakage out of the selected open area will be approximatly 25 %.

PIF-maps are computed at a number of energies, 10 for precisionLevel=20 and 18 for precisionLevel=25. PrecisionLevel 20 has been found to be sufficient in all tested cases.

Either FULL or REST imaging events will be used. The present version ONLY allow the FULL imaging option.
Phase binning is not yet implemented.
The possibility to choose alternative background methods (e.g. fitting) is forseen, but not implemented. Only the flat background option is provided. Background subtraction is intrinsic to the

extraction process.

Error estimates for both extraction methods are based on the assumption of poisson statistics and many counts per bin. For lightcurves with short time bins the number of counts per bin can be small and these error estimates are less reliable. Sources in the outer field of view are more sensitive to this since they only illuminate a small fraction of the detector.

Subordinate components: Functions of the executable:
j-src-work
j-src-params
j-src-array-all
j-src-mask
j-src-mask-find-element
j-src-oddeven
j-mask-findxy
j-eventweight
j-src-through-mask
j-src-through-coll
j-src-background
j-src-nspec
j-src-nspec-times
j-src-grey-correction
j-src-vignetting
j-src-interpol
j-src-interval-ave
j-src-fill-gauss
j-src-in-hole
j-src-add-array
j-src-leakage
j-src-maskhole-sample
j-src-maskhole-average
j-src-leak-coefficients
j-src-WriteLC
j-src-wcs-rsti
j-src-crossprod
j-src-randXY
j-src-openfraction
j-src-event-add
j-src-event-add-0
j-src-event

j-src-pixel-area
j-src-pif-sort
j-src-sort
j-src-dete-index
j-src-get-bit
j-min-range
j-gti-overlap
j-read-GTI
j-ssp-GetFileName
j-src-get-pointing
j-src-pif-interpol
j-src-factors-interpol
j-src-ext-crab
j-src-pif-create20
j-src-solve-rate
j-src-pifcut-interpol

Dependencies from other components: The following components should be run before j-src-lc:
j-data-preparation
j-calib-auto
j-correction
j-bkg-model (or select flat background)
j-image-sa-generation
j-src-find

ISDC context: Scientific Analysis

Delivery date:
Version 1.0 delivered 7 Sep 2000
Version 4.1.0 delivered 15 Jun 2004

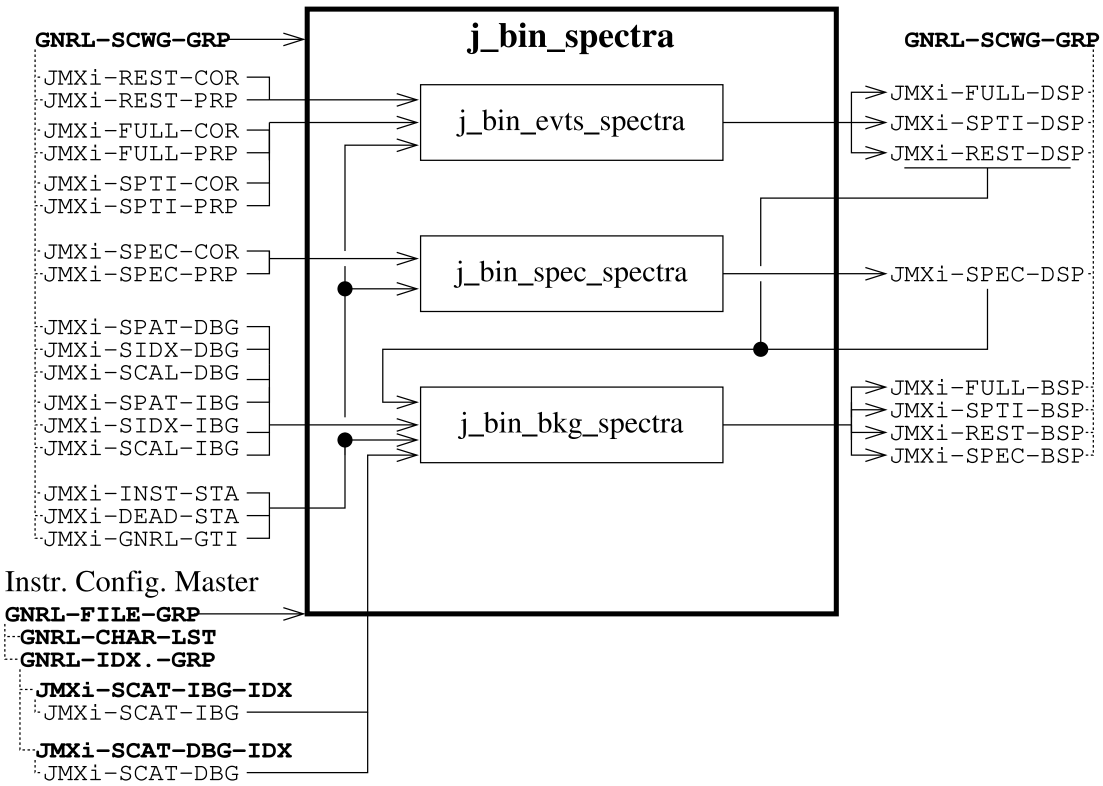

Figure 5.20: j-bin-spectra

5.16 j-bin-spectra

Type: Analysis Script **Responsible:** ESAC (P. Kretschmar) and Valencia University (S. Martínez-Núñez)

Purpose: Fulfill item 18.15 in SRD

Function: Bin JEM-X imaging and non-imaging data to spectra integrated over the whole detector. Create corresponding background data structures.

Inputs:
JMXi-FULL-PRP (p. 237) (event data)
JMXi-FULL-COR (p. 237)
JMXi-SPTI-PRP (p. 244)
JMXi-SPTI-COR (p. 244)
JMXi-REST-PRP (p. 241)

JMXi-REST-COR (p. 241)
JMXi-SPEC-PRP (p. 244) (on-board spectra)
JMXi-SPEC-COR (p. 243)
JMXi-FULL-ALL (p. 237) (event data, new scheme)
JMXi-SPTI-ALL (p. 244)
JMXi-REST-ALL (p. 241)
JMXi-SPEC-ALL (p. 243) (on-board spectra, new scheme)
JMXi-INST-STA (p. 239) (instrument status)
JMXi-GNRL-GTI (p. 238) (good-time intervals)
JMXi-SIDX-DBG (p. 243) (background model data)
JMXi-SPAT-DBG (p. 243)
JMXi-SCAL-DBG (p. 242)
JMXi-SIDX-BKG (p. 242)
JMXi-SPAT-BKG (p. 243)
JMXi-SCAL-BKG (p. 242)
JMXi-SCAT-DBG (p. 242) (background spectra catalogs, *not* in the swgDOL)
JMXi-SCAT-BKG (p. 242)

Outputs:
JMXi-FULL-DSP (p. 237)
JMXi-FULL-BSP (p. 237)
JMXi-SPTI-DSP (p. 244)
JMXi-SPTI-BSP (p. 244)
JMXi-REST-DSP (p. 241)
JMXi-REST-BSP (p. 241)
JMXi-SPEC-DSP (p. 244)
JMXi-SPEC-BSP (p. 243)

Parameters:

swgDOL p. 53	DOL of Science Window Group to be analyzed
jemxNum p. 46	JEM-X detector number
IC-Group p. 46	DOL of Instrument Characteristics master group
IC-Alias p. 46	Selection alias for Instrument Characteristics
instMod p. 46	DOL of JEM-X Instrument Model Group (empty=find in IC)
response p. 50	File with standard RMF and ARF (empty=find in IC)
timeStep p. 53	Binning time for spectra (0.0=bin all data together)
timeStart p. 53	Time in IJD of start of first time bin (<0 = use swgDOL start)

timeStop p. 53	Time in IJD of end of last time bin (<0 = use swgDOL end)
nPhaseBins p. 48	Number of phase bins (N<0 = abs(N) equally spaced bins)
phaseBins p. 49	Phase bin sizes, separated by space (e.g: 0.1 0.2 0.3 ...)
rowSelectEvts p. 51	CFITSIO selection string on events (optional)
rowSelectSpec p. 51	CFITSIO selection string on spectrum mode data (optional)
gtiNames p. 45	Names of GTI tables to be used
evtType p. 44	Data format to bin (-1=all,0=FULL,1=REST,2=SPTI,4=SPEC)
radiusLimit p. 49	Use events within this detector radius
chatter p. 42	Verbosity level; 2 = normal
clobber p. 42	Clobber existing output data?
subDir p. 53	Sub directory below the position of the SWG for results
osimData p. 48	Are we using data produced by OSim?
j-bin-evts-spectra-fullRaw p. 45	DOL of FULL imaging raw data (override)
j-bin-evts-spectra-fullPrp p. 44	DOL of FULL imaging prepared data (override)
j-bin-evts-spectra-fullCor p. 44	DOL of FULL imaging corrected data (override)
j-bin-evts-spectra-restRaw p. 50	DOL of REST imaging raw data (override)
j-bin-evts-spectra-restPrp p. 50	DOL of REST imaging prepared data (override)
j-bin-evts-spectra-restCor p. 50	DOL of REST imaging corrected data (override)
j-bin-evts-spectra-sptiRaw p. 52	DOL of Spectral-Timing raw data (override)
j-bin-evts-spectra-sptiPrp p. 52	DOL of Spectral-Timing prepared data (override)
j-bin-evts-spectra-sptiCor p. 52	DOL of Spectral-Timing corrected data (override)
j-bin-spec-spectra-specRaw p. 52	DOL of Spectrum Mode raw data (override)
j-bin-spec-spectra-specPrp p. 52	DOL of Spectrum Mode prepared data (override)
j-bin-spec-spectra-specCor p. 52	DOL of Spectrum Mode corrected data (override)
j-bin-spec-spectra-instStat p. 46	Instrument Status Data DOL (override)
j-bin-spec-spectra-deadTime p. 43	Deadtime History Table DOL (override)
j-bin-spec-spectra-goodTime p. 45	Good-Time Interval Table DOL (override)
j-bin-bkg-spectra-dbgSpecMap p. 42	Diffuse background spectral indices DOL (override)
j-bin-bkg-spectra-dbgSpatial p. 42	Diffuse background spatial distribution DOL (override)
j-bin-bkg-spectra-dbgScale p. 42	Diffuse background scaling factors DOL (override)
j-bin-bkg-spectra-bkgSpecMap p. 41	Instrm. background spectral indices DOL (override)
j-bin-bkg-spectra-bkgSpatial p. 41	Instrm. background spatial distribution DOL (override)
j-bin-bkg-spectra-bkgScale p. 41	Diffuse background scaling factors DOL (override)

Note: parameters with a prefix of “j-bin-. . . -spectra” are override parameters for data normally found in the Science Window Group, which are passed on to the respective called executable. Normally they are hidden and set to empty (no effect).

Resources: See children

Processing:

Read in parameters
Get DOLs to the JMXi-SCAT-... data structures from master IC group
Call j-bin-evts-spectra for Full Imaging data
Call j-bin-evts-spectra for Spectral/Timing data
Call j-bin-evts-spectra for Restricted Imaging data
Call j-bin-spec-spectra for Spectrum Format data
Call j-bin-bkg-spectra for Full Imaging data
Call j-bin-bkg-spectra for Spectral/Timing data
Call j-bin-bkg-spectra for Restricted Imaging data
Call j-bin-bkg-spectra for Spectrum Format data

Design remarks:
This component is especially necessary if JEM-X is (partially) working in one of its non-imaging modes.

There is no attempt at actual background subtraction in this component, the created background data structures are supposed to be used within scientific analysis - e.g. XSPEC for binned spectra.

At most two of the possible data formats will exist for any given Science window. The script makes no attempt to figure out which data are available. Instead, it is assumed that executables exit peacefully, without error messages, if their corresponding input data is missing.

Note: the creation of the background data was not foreseen in the SRD.

Parent components:
jemx-scw-analysis (p. 70)

Subordinate components:
j-bin-evts-spectra (p. 196)
j-bin-spec-spectra (p. 200)
j-bin-bkg-spectra (p. 203)

Dependencies from other components: Raw and prepared data must exist in the input swg-DOL.
The -BKG and -DBG data structures must have been created already by j-bkg-model.

Pending TBDs: None

ISDC context: Science Analysis

Delivery date: v4.5.6 19/10/2005

5.16.1 j-bin-evts-spectra

Type: Executable **Responsible:** ESAC (P. Kretschmar)

Purpose: Fulfill item 18.15 in SRD

Function: Bin data in Spectral/Timing format to spectra.

Interfaces: 1) The executable finds its input data (FULL, SPTI or REST events) within the swgDOL passed as parameter.
2) Dito for the status & greyfilter table, the deadtime history and GTI data.
3) The output table either already exists as child of the swgDOL or a suitable DOL for creation is passed via the parameter "specRes".

Parameters:

inSWG p. 46	Input SWG
outSWG p. 49	Output SWG (optional)
specRes p. 52	Output spectra DOL (optional)
jemxNum p. 46	JEM-X instrument number
evtType p. 44	Event mode number, must be 0=FULL, 1=REST or 2=SPTI
instMod p. 46	Instrument Model Group
timeStep p. 53	Binning time for spectra (0.0=bin all data together)
timeStart p. 53	Time in IJD of start of first time bin (<0 = use SWG start)
timeStop p. 53	Time in IJD of end of last time bin (<0 = use SWG end)
nPhaseBins p. 48	Number of phase bins (N<0 = abs(N) equally spaced bins)
phaseBins p. 49	Phase bin sizes, separated by space (e.g: 0.1 0.2 0.3 ...)
response p. 50	Path and name of file with standard RMF and ARF
radiusLimit p. 49	Use events within this detector radius
rowSelect p. 51	CFITSIO selection string on events (optional)
gtiNames p. 45	Names of GTI tables to be used
clobber p. 42	Clobber existing output data structures (y/n)?
chatter p. 42	Verbosity of output
fullPrp p. 44	Full Imaging Prepared Data DOL (override)
fullCor p. 44	Full Imaging Corrected Data DOL (override)
restPrp p. 50	Restricted Imaging Prepared Data DOL (override)
restCor p. 50	Restricted Imaging Corrected Data DOL (override)
sptiPrp p. 52	Spectral-Timing Prepared Data DOL (override)

sptiCor p. 52	Spectral-Timing Corrected Data DOL (override)
instStat p. 46	Instrument Status Data DOL (override)
deadTime p. 42	Deadtime History Table DOL (override)
goodTime p. 45	Good-Time Interval Table DOL (override)

Inputs:
JMXi-FULL-PRP (p. 237)
JMXi-FULL-COR (p. 237)
JMXi-SPTI-PRP (p. 244)
JMXi-SPTI-COR (p. 244)
JMXi-REST-PRP (p. 241)
JMXi-REST-COR (p. 241)
JMXi-INST-STA (p. 239)
JMXi-GNRL-GTI (p. 238)

Outputs:
JMXi-FULL-DSP (p. 237)
JMXi-SPTI-DSP (p. 244)
JMXi-REST-DSP (p. 241)

Resources: Memory usage is usually dominated by the input data. For REST events approximately $3000 \times T$ bytes will be allocated for input buffers, where T is the duration of the Science Window in seconds. For FULL and SPTI events this number is lower ($\sim 1200, 2100$). For the output data memory usage will be approximately $3100 \times N$, where N is the number of output spectra. For a typical Science Window of ~1000 sec this means memory usage of a few megabyte only.

The computational effort is very low, most time is usually taken in reading the input data.

Processing:

Read in all parameters, taking default values or parameters of SWG, where necessary
Read in GTI information
Read events falling within GTIs, matching evtType and rowSelect
Determine time bins to use (trivial for linear binning, important for phase resolved binning)
Setup output spectral tables, empty them if necessary and clobber is set
Obtain status and grey filter history
Obtain dead time history
Loop over all required time bins
    Empty local arrays holding spectral information

While events fall within time bin:
 Increase appropriate channel count in local count spectrum
Convert local counts to rates
Calculate mean grey filter correction factor
Correct countrates for grey filter
Calculate mean deadtime correction factor
Correct countrates for deadtime
Calculate fraction of time bin falling within GTIs
Calculate exposure of time bin based on binlength and fraction within GTI
Add local countrates $\times$ exposure to appropriate output spectrum
Add local exposure to total exposure of appropriate output spectrum
Correct total countrates and errors for total exposure
Store spectra

Design remarks:
The functionality described here goes beyond the SRD in including time- and phase-resolved binning, as well as the possibility to bin all types of events with energy information.

The output spectra are supposed to be trivially convertible to OGIP PHA formats or directly usable by XSPEC.

Spectral rebinning is left to generic tools.

The ability to select only part of the events, especially only those falling within a certain radius from the center has been added after launch.

Parent components:
j-bin-spectra (p. 191)

Subordinate components: Functions of this executable:
j-bin-evts-spectraGetParams
j-bin-evts-spectraProcessData
j-bes-ReadEventData
j-bes-SetupOutputSpectra
j-read-GTI
j-read-GreyStatus
j-read-DeadTimes
j-bin-MeanGreyCorrection
j-bin-MeanDeadTimeCorr
j-bin-MeanTimeSeries
j-bin-GoodFraction

Dependencies from other components: The following data should be available in the SWG:
– corrected event data
– greyfilter history
– good time interval data
– deadtime history
but only the first is absolutely mandatory.

Pending TBDs: None

ISDC context: Science Analysis

Delivery date: v2.8.0 01/09/2003

5.16.2 j-bin-spec-spectra

Type: Executable **Responsible:** ESAC (P. Kretschmar)

Purpose: Not foreseen in SRD

Function: Bin data in pure spectrum format to spectra.

Interfaces: 1) The executable finds its input data (SPECtrum Mode data) within the SWG passed as parameter.
2) Dito for the status & greyfilter table, the deadtime history and GTI data.
3) The output table either already exists as child of the SWG or a suitable DOL for creation is passed via the parameter "specRes".

Inputs:
JMXi-SPEC-PRP (p. 244)
JMXi-SPEC-COR (p. 243)
JMXi-INST-STA (p. 239)
JMXi-GNRL-GTI (p. 238)

Outputs:
JMXi-SPEC-DSP (p. 244)

Parameters:

inSWG p. 46	Input Science Window Group DOL
outSWG p. 49	Output Science Window Group DOL (optional)
specRes p. 52	Output spectra DOL (optional)
jemxNum p. 46	JEM-X instrument number
timeStep p. 53	Binning time for spectra (0.0=bin all data together)
timeStart p. 53	Time in IJD of start of first time bin (<0 = use SWG start)
timeStop p. 53	Time in IJD of end of last time bin (<0 = use SWG end)
nPhaseBins p. 48	Number of phase bins (N<0 = abs(N) equally spaced bins)
phaseBins p. 49	Phase bin sizes, separated by space (e.g: 0.1 0.2 0.3 ...)
response p. 50	Path and name of file with standard RMF and ARF
rowSelect p. 51	CFITSIO selection string on input spectra (optional)
gtiNames p. 45	Names of GTI tables to be used
clobber p. 42	Clobber existing output data structures (y/n)?
chatter p. 42	Verbosity of output
specRaw p. 52	Spectrum Mode Raw Data DOL (override)

specPrp p. 52	Spectrum Mode Prepared Data DOL (override)
specCor p. 52	Full Imaging Corrected Data DOL (override)
instStat p. 46	Instrument Status Data DOL (override)
deadTime p. 42	Deadtime History Table DOL (override)
goodTime p. 45	Good-Time Interval Table DOL (override)

Resources: Memory usage is expected to be low and dominated by the input data. Approximately $2100 \times T$ bytes will be allocated for input buffers, where T is the duration of the Science Window in seconds. For the output data memory usage will be approximately $3100 \times N$, where N is the number of output spectra. For a typical Science Window of $\sim$1000 sec this means memory usage of a few megabyte only.

The computational effort is very low, most time is usually taken in reading the input data.

Processing:

Read in all parameters, taking default values or parameters of SWG, where necessary
Read in GTI information
Read Spectrum Mode data falling within GTIs
Determine time bins to use (trivial for linear binning, important for phase resolved binning)
Setup output spectral tables, empty them if necessary and clobber is set
Obtain status and grey filter history
Obtain dead time history
Loop over all required time bins
- Empty local arrays holding spectral information
- While input spectra fall (partially) within current time bin:
  - Add spectra to local count spectrum, weighed with overlap with time bin
  - Convert local counts to rates
  - Calculate mean grey filter correction factor
  - Correct countrates for grey filter
  - Calculate mean deadtime correction factor
  - Correct countrates for deadtime
  - Calculate fraction of time bin falling within GTIs
  - Calculate exposure of time bin based on binlength and fraction within GTI
  - Add local countrates $\times$ exposure to appropriate output spectrum
  - Add local exposure to total exposure of appropriate output spectrum
- Correct total countrates and errors for total exposure
- Store spectra

Design remarks:
The short integration times (0.125 sec) and many energy channels (64) of the spectra generated on-board will usually not allow scientific analysis of individual spectra. Therefore this component allows to create statistically significant spectra binned in time and energy.

Parent components:
j-bin-spectra (p. 191)

Subordinate components: Functions of this executable:
j-bin-spec-spectraGetParams
j-bin-spec-spectraProcessData
j-bss-ReadSpectra
j-bss-SetupOutputSpectra
j-read-GTI
j-read-GreyStatus
j-read-DeadTimes
j-bin-MeanGreyCorrection
j-bin-MeanDeadTimeCorr
j-bin-MeanTimeSeries
j-bin-GoodFraction

Dependencies from other components: The following data should be available in the SWG:
– corrected Spectrum Mode data
– greyfilter history
– good time interval data
– deadtime history
but only the first is absolutely mandatory.

Pending TBDs:

ISDC context: Science Analysis

Delivery date: 2.5.1 16/05/2003

5.16.3 j-bin-bkg-spectra

Type: Executable **Responsible:** ESAC (P. Kretschmar)

Purpose: Not foreseen in SRD

Function: Bin background model data to spectra.

Interfaces: 1) The binned spectrum for which a background model spectrum is required is found within the SWG passed as parameter, if not given explicitly via an override parameter.
2) Dito for the standard spectral model data (spatial, spectral index and scaling history).
3) In addition, two tables holding model spectra for instrumental and diffuse background are passed directly via parameters.
4) The output table either already exists as child of the SWG or a suitable DOL for creation is passed via the parameter "backRes".

Inputs:
JMXi-FULL-DSP (p. 237)
JMXi-SPTI-DSP (p. 244)
JMXi-REST-DSP (p. 241)
JMXi-SPEC-DSP (p. 244)
JMXi-INST-STA (p. 239)
JMXi-GNRL-GTI (p. 238)
JMXi-SIDX-DBG (p. 243)
JMXi-SPAT-DBG (p. 243)
JMXi-SCAL-DBG (p. 242)
JMXi-SIDX-BKG (p. 242)
JMXi-SPAT-BKG (p. 243)
JMXi-SCAL-BKG (p. 242)
JMXi-SCAT-DBG (p. 242)
JMXi-SCAT-BKG (p. 242)

Outputs:
JMXi-FULL-BSP (p. 237)
JMXi-SPTI-BSP (p. 244)
JMXi-REST-BSP (p. 241)
JMXi-SPEC-BSP (p. 243)

Parameters:

inSWG p. 46	Input Science Window Group DOL
outSWG p. 49	Output Science Window Group DOL (optional)
backRes p. 40	Output background spectra DOL (optional)
specRes p. 52	Input detector spectra DOL (override)
jemxNum p. 46	JEM-X instrument number
evtType p. 44	TM format number: must be 0=FULL, 1=REST, 2=SPTI or 4=SPEC
radiusLimit p. 49	Use events within this detector radius
dbgSpecCat p. 42	Diffuse background spectral model table DOL
bkgSpecCat p. 41	Instrument background spectral model table DOL
clobber p. 42	Clobber existing output data structures (y/n)?
chatter p. 42	Verbosity of output
dbgSpecMap p. 42	Diffuse background spectral indices DOL (override)
dbgSpatial p. 42	Diffuse background spatial distribution DOL (override)
dbgScale p. 42	Diffuse background scaling factors DOL (override)
bkgSpecMap p. 41	Instrm. background spectral indices DOL (override)
bkgSpatial p. 41	Instrm. background spatial distribution DOL (override)
bkgScale p. 41	Diffuse background scaling factors DOL (override)

Resources: No unusual

Processing:

Read in parameters
Find input and output data structures
Read in good-time intervals and greyfilter history
Read in background model data
Determine number of output spectra & temporal binning from corresponding detector spectra
Loop over output spectra
    Calculate countrate spectrum from background model data as function of time
        but integrating over whole detector
    Calculate mean grey filter correction factor for time interval
    Correct countrates for grey filter
    Store output spectrum

Design remarks:
The binning is done in real world time units (IJD).

Parent components:
j-bin-spectra (p. 191)

Subordinate components: Functions of this executable:
j-bin-bkg-spectraGetParams
j-bin-bkg-spectraProcessData
j-bin-calcMeanBkgSpectrum
j-bin-getBkgSpatialModel
j-bin-getBkgSpectralIDs
j-bin-getBkgSpectra
j-bin-MeanTimeSeries
j-bin-rebinSpectrum

Dependencies from other components: j-bkg-model must have produced background model data.
j-bin-evts-spectra or j-bin-spec-spectra must have produced the required detector spectra (-DSP) data.

Pending TBDs:

ISDC context: Science Analysis

Delivery date: v2.1.0 08/10/2003

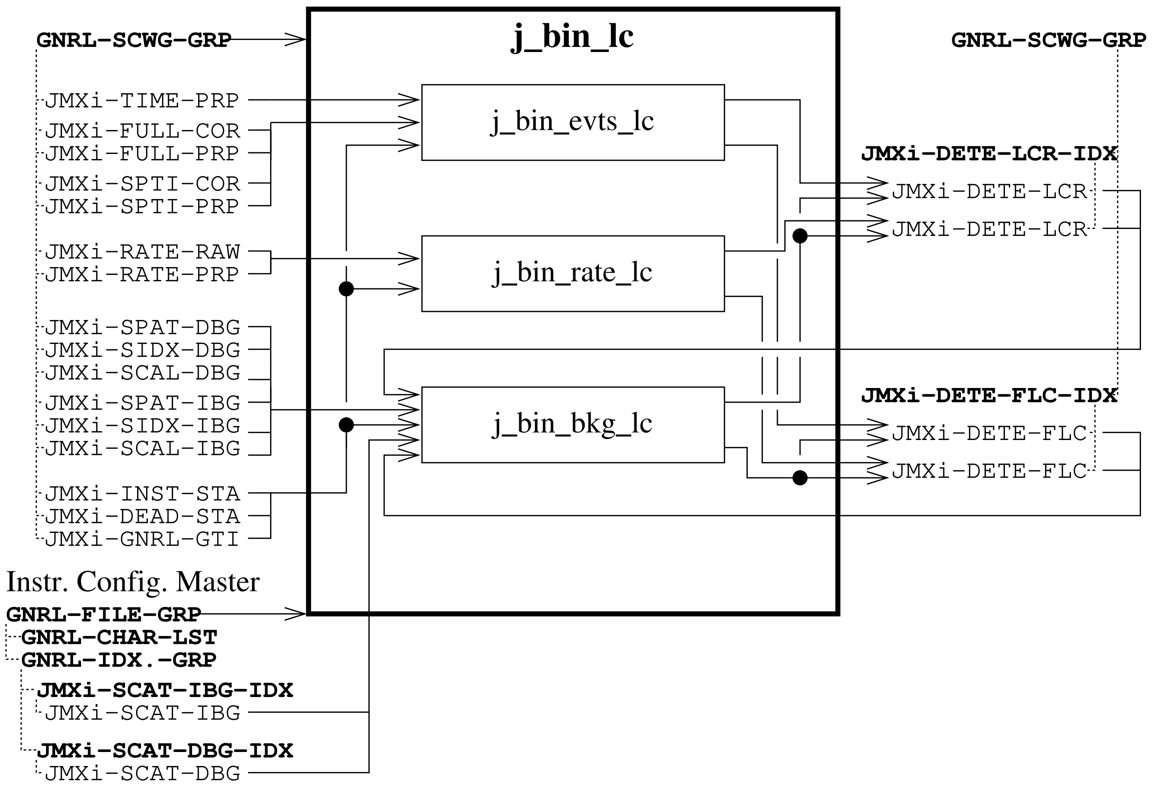

Figure 5.21: j-bin-lc

5.17 j-bin-lc

Type: Analysis Script **Responsible:** ESAC (P. Kretschmar) and Valencia University (S. Martínez-Núñez)

Purpose: Fulfill item 18.16 in SRD

Function: Bin JEM-X imaging and non-imaging data to lightcurves integrated over the whole detector. Create corresponding background data structures.

Inputs:
JMXi-FULL-PRP (p. 237) (event data)
JMXi-FULL-COR (p. 237)
JMXi-SPTI-PRP (p. 244)
JMXi-SPTI-COR (p. 244)
JMXi-TIME-PRP (p. 245)

JMXi-RATE-RAW (p. 241) (countrate data, taken parallel to Restricted Imaging)
JMXi-RATE-PRP (p. 241)
JMXi-FULL-ALL (p. 237) (event data, new scheme)
JMXi-SPTI-ALL (p. 244)
JMXi-REST-ALL (p. 241)
JMXi-INST-STA (p. 239) (instrument status)
JMXi-GNRL-GTI (p. 238) (good-time intervals)
JMXi-SIDX-DBG (p. 243) (background model data)
JMXi-SPAT-DBG (p. 243)
JMXi-SCAL-DBG (p. 242)
JMXi-SIDX-BKG (p. 242)
JMXi-SPAT-BKG (p. 243)
JMXi-SCAL-BKG (p. 242)
JMXi-SCAT-DBG (p. 242) (background spectra catalogs, *not* in the SWG)
JMXi-SCAT-BKG (p. 242)

Outputs:
JMXi-DETE-LCR-IDX (p. 235)
JMXi-DETE-LCR (p. 235)
JMXi-DETE-FLC-IDX (p. 235)
JMXi-DETE-FLC (p. 234)

Parameters:

swgDOL p. 53	Science Window Group
IC-Group p. 46	Instrument Configuration Master Group
IC-Alias p. 46	Instrument Configuration version alias
jemxNum p. 46	JEM-X detector number (1 or 2)
instMod p. 46	DOL of JEM-X Instrument Model Group (empty=find in IC)
timeStep p. 53	Length of integration time bins
timeStart p. 53	Time in IJD of start of first time bin (<0 = use SWG start)
timeStop p. 53	Time in IJD of end of last time bin (<0 = use SWG end)
nChanBins p. 47	Number of channel bins (0=bin all, -1=full resolution)
chanLow p. 41	Lower channel bounds (numbers separated by spaces)
chanHigh p. 41	Upper channel bounds (numbers separated by spaces)
nPhaseBins p. 48	Number of phase bins (N<0 = abs(N) equally spaced bins)
phaseBins p. 49	Phase bin sizes, separated by space (e.g: 0.1 0.2 0.3 ...)
rowSelect p. 51	CFITSIO selection string on events (optional)
gtiNames p. 45	Names of GTI tables to be used

evtType p. 44	Data format to bin (-1=all,0=FULL,2=SPTI,5=TIME)
radiusLimit p. 49	[mm] Use events within this detector radius
chatter p. 42	Verbosity level
clobber p. 42	Clobber existing output data?
subDir p. 53	Subdirectory below the position of the SWG for results
osimData p. 48	Are we using data produced by OSim?
j-bin-evts-lc-fullPrp p. 44	Full Imaging Prepared Data DOL (override)
j-bin-evts-lc-fullCor p. 44	Full Imaging Corrected Data DOL (override)
j-bin-evts-lc-sptiPrp p. 52	Spectral-Timing Prepared Data DOL (override)
j-bin-evts-lc-sptiCor p. 52	Spectral-Timing Corrected Data DOL (override)
j-bin-evts-lc-timePrp p. 53	Timing Prepared Data DOL (override)
j-bin-evts-lc-instStat p. 46	Instrument Status Data DOL (override)
j-bin-evts-lc-deadTime p. 43	Deadtime History Table DOL (override)
j-bin-evts-lc-goodTime p. 45	Good-Time Interval Table DOL (override)
j-bin-rate-lc-rateRaw p. 50	Countrate Prepared Data DOL (override)
j-bin-rate-lc-ratePrp p. 50	Countrate Prepared Data DOL (override)
j-bin-rate-lc-instStat p. 46	Instrument Status Data DOL (override)
j-bin-rate-lc-deadTime p. 43	Deadtime History Table DOL (override)
j-bin-rate-lc-goodTime p. 45	Good-Time Interval Table DOL (override)
j-bin-bkg-lc-dbgSpecMap p. 42	Diffuse background spectral indices DOL (override)
j-bin-bkg-lc-dbgSpatial p. 42	Diffuse background spatial distribution DOL (override)
j-bin-bkg-lc-dbgScale p. **??**	Diffuse background scaling factors DOL (override)
j-bin-bkg-lc-bkgSpecMap p. **??**	Instrm. background spectral indices DOL (override)
j-bin-bkg-lc-bkgSpatial p. 41	Instrm. background spatial distribution DOL (override)
j-bin-bkg-lc-bkgScale p. 41	Diffuse background scaling factors DOL (override)

Resources: No unusual

Processing:

Read in parameters
Get DOLs to the JMXi-SCAT-... data structures from master IC group
Call j-bin-evts-lc for Full Imaging data
Call j-bin-evts-lc for Spectral/Timing data
Call j-bin-evts-lc for Timing data
Call j-bin-rate-lc for Countrate data
Call j-bin-bkg-lc for Full Imaging data
Call j-bin-bkg-lc for Spectral/Timing data
Call j-bin-bkg-lc for Timing data
Call j-bin-bkg-lc for Countrate data

Design remarks:
This component is especially necessary if JEM-X is (partially) working in one of its non-imaging modes.

There is no attempt at actual background subtraction in this component, the created background data structures are supposed to be used within scientific analysis.

At most two of the possible data formats will exist for any given Science window. The script makes no attempt to figure out which data are available. Instead, it is assumed that executables exit peacefully, without error messages, if their corresponding input data is missing.

Note: the creation of the background data was not foreseen in the SRD.

Parent components:
jemx-scw-analysis (p. 70)

Subordinate components:
j-bin-evts-lc (p. 210)
j-bin-rate-lc (p. 213)
j-bin-bkg-lc (p. 216)

Dependencies from other components: Raw and prepared data must exist in the input SWG. The -BKG and -DBG data structures must have been created already by j-bkg-model.

Pending TBDs: None

ISDC context: Science Analysis

Delivery date: v4.5.6 19/10/2005

5.17.1 j-bin-evts-lc

Type: Executable **Responsible:** ESAC (P. Kretschmar)

Purpose: Fulfill item 18.16 in SRD

Function: Bin JEM-X event data to lightcurves.

Parameters:

inSWG p. 46	Input SWG
outSWG p. 49	Output SWG (override)
lcRes p. 46	Output lightcurve DOL
jemxNum p. 46	JEM-X instrument number
evtType p. 44	Event mode number, must be 0=FULL, 2=SPTI or 3=TIME
timeStep p. 53	Binning time for lightcurve (sec)
nChanBins p. 47	Number of channel bins (0=bin all, -1=full resolution)
chanLow p. 41	Lower channel bounds (numbers separated by spaces)
chanHigh p. 41	Upper channel bounds (numbers separated by spaces)
clobber p. 42	Clobber existing output data structures (y/n)?
chatter p. 42	Verbosity of output
radiusLimit p. 49	Use events within this detector radius
gtiNames p. 45	Names of GTI tables to be used
rowSelect p. 51	CFITSIO selection string on events (optional))
fullPrp p. 44	Full Imaging Prepared Data DOL (override)
fullCor p. 44	Full Imaging Corrected Data DOL (override)
sptiPrp p. 52	Spectral-Timing Prepared Data DOL (override)
sptiCor p. 52	Spectral-Timing Corrected Data DOL (override)
timePrp p. 53	Timing Prepared Data DOL (override)
instStat p. 46	Instrument Status Data DOL (override)
deadTime p. 42	Deadtime History Table DOL (override)
goodTime p. 45	Good-Time Interval Table DOL (override)
nPhaseBins p. 48	Number of phase bins
phaseBins p. 49	Specification of phase bins
timeStart p. 53	Start OBT of lightcurve
timeStop p. 53	End OBT of lightcurve

Inputs:

JMXi-FULL-PRP (p. 237)
JMXi-FULL-COR (p. 237)
JMXi-SPTI-PRP (p. 244)
JMXi-SPTI-COR (p. 244)
JMXi-TIME-PRP (p. 245)

JMXi-INST-STA (p. 239)
JMXi-GNRL-GTI (p. 238)

Outputs:
JMXi-DETE-LCR-IDX (p. 235)
JMXi-DETE-LCR (p. 235)
JMXi-DETE-FLC-IDX (p. 235)
JMXi-DETE-FLC (p. 234)

Resources: Not unusual

Processing:

Read in parameters
Find input and output data structures
Read in good-time intervals and greyfilter history
Read in events falling into "good" time intervals
Define starttime=time of first event
Create lightcurve data structure
Foreach event do:
    If time $\leq$ startime+timestep:
        increase bin counter (of appropriate energy channel)
Else move to next time bin
Correct bin counts for grey filter settings

Design remarks:
The functionality described here goes somewhat beyond the SRD in including energy information and thus possibly energy resolved lightcurves in the output data.

The ability to select only part of the events, especially only those falling within a certain radius from the center has been added after launch.

Parent components:
j-bin-lc (p. 206)

Subordinate components:
j_bin_evts_lcGetParams

j_bin_evts_lcProcessData
j_bel_ReadEventData
j_bel_CleanOutLC
j_bel_WriteLC
j_bin_GoodFraction
j_read_GTI
j_read_GreyStatus
j_read_DeadTimes

Dependencies from other components: Raw and prepared data must exist in the input SWG.

Pending TBDs: None

ISDC context: Science Analysis

Delivery date: v2.5.0 01/09/2003

5.17.2 j-bin-rate-lc

Type: Executable **Responsible:** ESAC (P. Kretschmar)

Purpose: Fulfill item 18.15 in SRD

Function: Bin JEM-X event data to lightcurves.

Inputs:
JMXi-RATE-RAW (p. 241)
JMXi-RATE-PRP (p. 241)
JMXi-INST-STA (p. 239)
JMXi-GNRL-GTI (p. 238)

Outputs:
JMXi-DETE-LCR-IDX (p. 235)
JMXi-DETE-LCR (p. 235)
JMXi-DETE-FLC-IDX (p. 235)
JMXi-DETE-FLC (p. 234)

Parameters:

inSWG p. 46	Input Science Window Group DOL
outSWG p. 49	Output Science Window Group DOL (override)
jemxNum p. 46	JEM-X instrument number
timeStep p. 53	Binning time for spectra (0.0=bin all data together)
instMod p. 46	DOL of JEM-X Instrument Model Group (empty=find in IC)
clobber p. 42	Clobber existing output data structures (y/n)?
chatter p. 42	Verbosity of output
rowSelect p. 51	CFITSIO selection string on events (optional)
lcRes p. 46	Output lightcurve DOL (override)
rateRaw p. 50	Countrate Raw Data DOL (override)
ratePrp p. 50	Countrate Prepared Data DOL (override)
instStat p. 46	Instrument Status Data DOL (override)
deadTime p. 42	Deadtime History Table DOL (override)
goodTime p. 45	Good-Time Interval Table DOL (override)
nPhaseBins p. 48	Number of phase bins
phaseBins p. 49	Specification of phase bins
timeStart p. 53	Start OBT of lightcurve
timeStop p. 53	End OBT of lightcurve

Resources: No unusual

Processing:

Read in parameters
Find input and output data structures
Read in good-time intervals and greyfilter history
Read in countrate data falling into "good" time intervals
Define starttime (=time of first countrate bin)
Create lightcurve data structure
Foreach output bin do:
  Determine countrate bins falling completely or partially into output bin
  Add up these countrate bins, appropriately weighted for exposure within output bin
  Correct bin counts for grey filter settings

Design remarks:
Timesteps of <0.125 sec are impossible because the original countrate data is transmitted with this resolution.

Parent components:
j-bin-lc (p. 206)

Subordinate components:
j_bin_rate_lcGetParams
j_bin_rate_lcProcessData
j_brl_ReadRateData
j_brl_CleanOutLC
j_brl_WriteLC
j_read_GTI
j_read_GreyStatus
j_read_DeadTimes
j_bin_MeanTimeSeries
j_bin_GoodFraction
j_bin_MeanGreyCorrection
j_bin_MeanDeadTimeCorr

Dependencies from other components: Raw and prepared data must exist in the input Science Window Group.

Pending TBDs: None

ISDC context: Science Analysis

Delivery date: v2.2.1 16/06/2003

5.17.3 j-bin-bkg-lc

Type: Executable **Responsible:** ESAC (P. Kretschmar)

Purpose: Not foreseen in SRD

Function: Bin background model data to spectra.

Parameters:

inSWG p. 46	Input Science Window Group DOL
outSWG p. 49	Output Science Window Group DOL (override)
dbgSpecCat p. 42	Diffuse background spectral model table DOL
bkgSpecCat p. 41	Instrument background spectral model table DOL
jemxNum p. 46	JEM-X instrument number
lcType p. 47	Work on normal lightcurves (0), folded (1) or both (-1)
rowSelect p. 51	CFITSIO selection string on lightcurves
radiusLimit p. 49	Use events within this detector radius
chatter p. 42	Verbosity of output
clobber p. 42	Clobber existing output data structures (y/n)?
lcRes p. 46	Input detector lightcurve DOL (override)
dbgSpecMap p. 42	Diffuse background spectral indices DOL (override)
dbgSpatial p. 42	Diffuse background spatial distribution DOL (override)
dbgScale p. 42	Diffuse background scaling factors DOL (override)
bkgSpecMap p. 41	Instrm. background spectral indices DOL (override)
bkgSpatial p. 41	Instrm. background spatial distribution DOL (override)
bkgScale p. 41	Diffuse background scaling factors DOL (override)

Inputs:
JMXi-DETE-LCR-IDX (p. 235)
JMXi-DETE-LCR (p. 235)
JMXi-INST-STA (p. 239)
JMXi-GNRL-GTI (p. 238)
JMXi-SIDX-DBG (p. 243)
JMXi-SPAT-DBG (p. 243)
JMXi-SCAL-DBG (p. 242)
JMXi-SIDX-BKG (p. 242)
JMXi-SPAT-BKG (p. 243)
JMXi-SCAL-BKG (p. 242)

Outputs:
JMXi-DETE-LCR-IDX (p. 235)
JMXi-DETE-LCR (p. 235)

JMXi-DETE-FLC-IDX (p. 235)
JMXi-DETE-FLC (p. 234)

Resources: No unusual

Processing:

Read in parameters
Find input and output data structures
Read in good-time intervals and greyfilter history
Read in background model data
Determine number of output lightcurve temporal and spectral bins from detector lightcurves
Loop over output temporal bins
    Calculate countrates (posisibly in several energy bands) from background model data
        as function of time but integrating over whole detector
    Calculate mean grey filter correction factor for time interval
    Correct countrates for grey filter
    Store output spectrum

Design remarks:

Parent components:
j-bin-lc (p. 206)

Subordinate components:
j_bin_bkg_lcGetParams
j_bin_bkg_lcProcessDataj_bbl_setupLCTypes
j_bin_calcMeanBkgSpectrum
j_bin_getBkgSpatialModel
j_bin_getBkgSpectralIDs
j_bin_getBkgSpectra
j_bin_getNumBkgScaling
j_bin_getBkgScaling
j_bbl_AvgOverPhase
j_bin_MeanTimeSeries

Dependencies from other components: j-bkg-model must have produced background model data.
j-bin-evts-lc or j-bin-rate-lc must have produced the required detector lightcurve (-DETE-LCR) data.

Pending TBDs:

ISDC context: Science Analysis

Delivery date: 2.2.1 10/10/2003

5.18 j-ima-mosaic

Type: Executable **Responsible:** DNSC (Jérôme Chenevez)

Purpose: 18.20.3

Function: To merge individual science windows skymaps produced by j-ima-iros. The executable calculates the weighted contribution of each pixel from each input sky image. Mosaic maps of the accumulated raw observation time and of the effective exposure time (taking account of the dead time, grey filter effects and vignetting of the JEM-X instruments) are produced as well as intensity, variance, and significance maps for each of the input energy bands.

Interfaces: Input data are obtained from the Observation Group (ogDOL p. 48) passed as parameter (as a minimum). The list of SWG corresponds to the ScWs in which the images to combine are fetched. The global keywords `OGID`, `TSTART`, `TSTOP`, and `TELAPSE` are copied from the observation group. Other keywords like, e.g., `BUNIT`, `CRVAL`, `CRPIX`, `E_MIN`, `E_MAX`, etc. are read from each individual input map and eventually modified in the processing. The vignetting files are passed via the keyword `VIGNDOL`. The keywords `IMATYPE`, `IMADESCR` and `IMACASE` are used to select the input images. Output mosaic maps are grouped in index (one per energy band plus one optional file grouping the accumulated observation and effective exposure times).

Inputs:
GNRL-OBSG-GRP (p. 248) (ogDOL p. 48)
GNRL-SCWG-GRP-IDX (p. 249)
GNRL-SCWG-GRP (p. 249) (list of ScWs)
JMXi-DMAP-BPL (p. 235) (vignetting from back projection lists)
JMXi-SKY.-IMA (p. 243) (individual images including the celestial coord. system)

Outputs:
JMXi-MOSA-IMA-IDX (p. 239)
JMXi-MOSA-IMA (p. 239) Mosaic maps in the output index table

Parameters:

inObsGrp p. 46	DOL	Input Observation Group (ogDOL p. 48)
mapSelect p. 47	str.	Images selection keyword `IMADESCR` (*RAW_RECT* or *RES+SRC*)
radiusSelect p. 50	real	Input images selection radius [deg.]
eminSelect p. 44	real	Input images selection min. energy [keV]
emaxSelect p. 44	real	Input images selection max. energy [keV]
diameter p. 43	real	Output mosaic maps diameter [deg.]
cdelt p. 41	real	Output pixel size at the center [deg.]

RAcenter p. 49	real	Output mosaic map center RA [deg.]
DECcenter p. 43	real	Output mosaic map center DEC [deg.]
outfile p. 48	str.	Output mosaic file name prefix
viewTime p. 54	bool.	Create total exposure Time mosaic map?
viewIntens p. 54	bool.	Create raw Intensity mosaic map?
viewVar p. 54	bool.	Create Variance mosaic map?
viewSig p. 54	bool.	Create Significance mosaic map?
chatter p. 42	int.	Verbosity level of logfile output (0,1,2,3,4,5)

Resources: Uses about 424 MB of memory. To produce a mosaic map (one energy band) from 60 ScWs takes about 1mn 30s CPU time on a linux PC with 2.80GHz processor.

Processing:
j_ima_mosaic produces a FITS file named (by default) **J_MOSAIC_#.fits** for each energy band (# starting from 0) found in the input Observation Group (ogDOL) unless the user has specifically made a restrictive energy selection by use of the `eminSelect` and `emaxSelect` parameters.

There are three parameters defining the mosaic maps produced in the files **J_MOSAIC_#.fits**, so it can have up to three image extensions (JMXi-SKY.-IMA), respectively from [2] to [4], showing: 1) the weighted intensity (counts/cm2/s) map (if `viewIntens` is set), 2) the weighted variance map (by setting `viewVar`), 3) the significance map (with `viewSig`) defined as the intensity divided by the square root of the variance at each pixel.

An optional output file **OBS_TIME_n.fits** (where n is the number of combined input images – max 999) can also be produced with two extensions (JMXi-SKY.-IMA) containing: 1) a map of the accumulated raw observation times given in seconds and 2) a map of the effective exposure times (in seconds) taking account of the dead time, grey filter effects and vignetting of the JEM-X instruments. The vignetting used to compute the latter map is read from JMXi-DMAP-BPL stored in the IC files `JMXi-BPL.-GRP` via the keyword `VIGNDOL` contained in the input images.

If sky coordinates (`RAcenter`, `DECcenter`) are given as parameters to define the center of the mosaic map, an ordered list of the input images as a function of their respective weights at the given coordinates is displayed for each energy band in the log file.

Initialise the executable
Read in the parameter file
Open the input ogDOL
    Get instrument name to make DS prefix (JMX1 or JMX2).
    List last extensions (JMXi-SKY.-IMA) to get their number.

Read in first member of the list to get the size of input images and check presence of necessary keywords; the program stops with a warning if `IMATYPE` is not found.

- Initialise arrays and variables
- Get vignetting info from JMXi-DMAP-BPL
- Read header attributes of each input Variance map
  - List all `VARIANCE` maps; the program stops with a warning if no `VARIANCE` map is found.
  - Get total number of energy bands
  - Loop over all Variance maps for every ScW and energy band
    - Read `SWID`
    - Get coord. info and energy range of each individual input image
  - End loop over Variance maps
- Define the output mosaic images
  - Find center and determine size of the mosaic skymaps
  - Build ”sky to FITS” transformation matrices
- Compose mosaics by looping over sky images to add for each energy band
  - Initialise the composition of the mosaics
    - List selected input `INTENSITY` skymaps
    - List selected input `VARIANCE` skymaps
  - Loop over energy bands (main loop of the program)
    - Select energy band
    - Refine selection of input `INTENSITY` and `VARIANCE` skymaps
    - Read and smooth corresponding vignetting map
    - Loop over science windows (mosaicking loop)
      - Build ”FITS to sky” transformation matrices
      - Get and read Intensity and Variance input skymaps
      - Loop over individual pixels
        - Define input pixels arrays
        - Loop over sub-pixels
          - Make ”FITS to sky” transformations
          - Transpose to the mosaic image (”sky to FITS”)
          - Compute weights and compose the mosaic maps
        - End loop over sub-pixels
      - End loop over pixels
    - End loop over ScWs
    - Prepare outputs and generate the mosaic FITS file for the given energy band
      - Write `INTENSITY` (I) mosaic map
      - Write `VARIANCE` (V) mosaic map
      - Write `SIGNIFICANCE` map: $S = I/\sqrt{V}$
  - End loop over energy bands
- Generate Observation Time mosaic map and Effective Exposure Time (vignetted) mosaic map.

Design remarks:
In order to achieve backward compatibilty the code has not been cleaned for historical developments and features now become obsolete. As an example, the variable `useA` when <0 makes possible to combine old format images (produced by *j_ima_basic_recon* and *j_ima_fine_resol*). In the same way, the value `useA` =0 may still be used to select optional `RAW_INT` format images produced on demand by the user at the IMA level.

Parent components:
jemx-obs-analysis (p. 75)

Subordinate components:
j-ima-mosaic-params (standard)
j-ima-mosaic-work (standard)
clean
j-vign-matrix
thruput_instr
circ_areal
instrumentGetCode

Dependencies from other components: The application j-imaging (p. 161) must have produced at least one image FITS file containing both an Intensity skymap and a Variance skymap in JMXi-SKY.-IMA extensions.

ISDC context: Scientific Analysis

Delivery date: Version 5.0 12/04/2005 Version 5.1.0 28/10/2005

5.19 j-performance

Type: Application/GUI **Responsible:** S. Maisala

Purpose: 18.6 18.7 18.8

Function: Tool to perform long and short term monitoring of properties of the detector.

- User can take a choice of available parameters and set time interval (IJD).
- Program collects house keeping data either one or two JEM-X units and merge them to one table
- Available parameters: HK-parameters: voltages, currents, pressures, and temperatures and Performance parameters
- Displays parameters vs time or other parameter with IOSM tools

Inputs:
GNRL-OBSG-GRP (p. 248) (observation group)
JMXi-CSSW-HRW (p. 234) (housekeeping data)
JMXi-CSSW-AVR (p. 234)
JMXi-CSSW-CNV (p. 234)
JMXi-INST-STA (p. 239)
GNRL-SCWG-GRP (p. 249)
GNRL-SCWG-GRP-IDX (p. 249)

Outputs: GUI display of chosen parameters for offline analysis only

Parameters:

inObsGrp p. 46	Input observation group
perfTable p. 49	Output table
jemxNum p. 46	JEM-X instrument number (1,2,0=both)
timeStart p. 53	IJD
timeStop p. 53	IJD
hkType p. 46	house keeping type (0=AVG, 1=RAW,2=CONV)
binFactor p. 41	binning factor (0=full)
parNames p. **??**	ascii file of chosen parameters
chatter p. 42	Verbosity of logfile output (0,1,2,3,4,5)

Resources:

Design remarks:

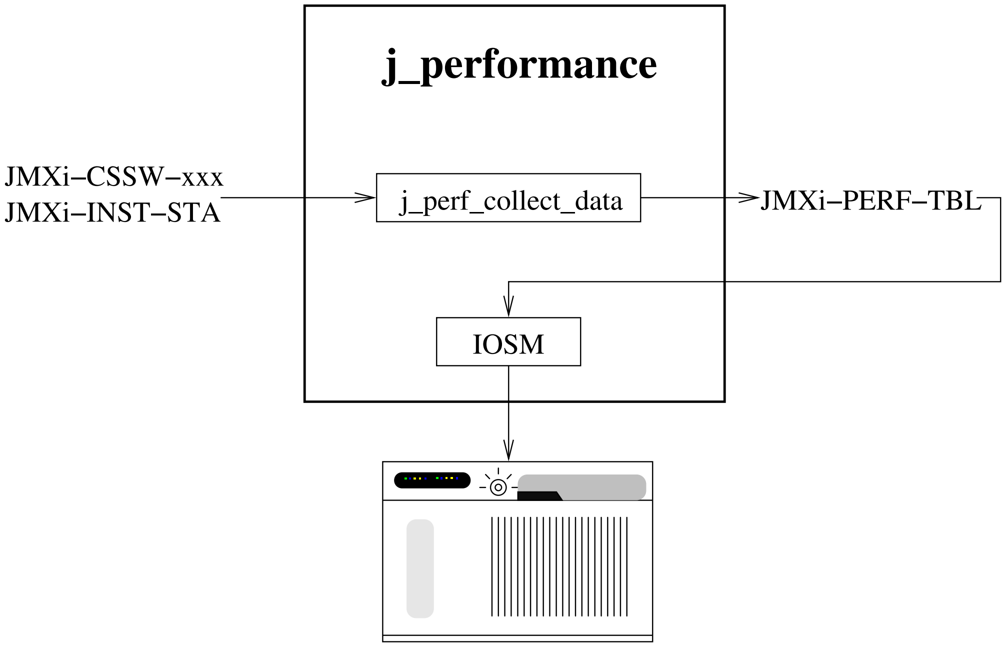

Figure 5.22: j-performance

Limits for the parameters are in JMXi-ALRT-LIM. The purpose of this component is to display the parameter and show predefined limits. if parameter exceeded limit it is not suppose to lead to any further actions by this task or any other task.

Parent components: None

Subordinate components:
j-perf-collect-data (p. 225)

Dependencies from other components: Available observation groups

Pending TBDs: Parameters and resources are TBD.

Prototyping status: None

ISDC context: Off-line processing, Fig. 1

Delivery date: August 2002

5.19.1 j-perf-collect-data

Type: Executable **Responsible:** S. Maisala

Purpose: 18.6

Function: Collects selected data from house keeping and creates one table of all selected parameters by given time period (IJD). User can choose wanted house keeping parameter by creating *default-params.asc* file where is a list of selected house keeping parameters. It is possible to collect data only from JEMX-1 or JEMX-2 but it is also possible to collect data from both detectors at the same time by selecting instrument number as 0 (=both).
Output table contains columns of on board time (OBT), Julian date (IJD) and all selected parameters. All parameter names have added an instrument number suffix string (JMX1-, JMX-2) before actual parameter name. That is a difference between an original house keeping data structure and j-performane table. This arrangement is used to make possible to keep information of the both instruments in the same table. All column data types are in original house keeping data format.

Inputs:
GNRL-OBSG-GRP (p. 248) (observation group)
JMXi-CSSW-HRW (p. 234) (housekeeping data)
JMXi-CSSW-AVR (p. 234)
JMXi-CSSW-CNV (p. 234)
JMXi-INST-STA (p. 239) (Instrument mode and grey filter status)

Outputs:
JMXi-PERF-TBL (p. 240)

Parameters:

inObsGrp p. 46	Input observation group
perfTable p. 49	Output performance table
jemxNum p. 46	JEM-X instrument number (1,2,0 = both)
timeStart p. 53	IJD
timeStop p. 53	IJD
parNames p. **??**	ascii file of chosen parameters
hkType p. 46	Type of housekeeping to use

Resources: Nothing unusual

Processing:
Read parameters.
Create output DOL perfTable from JMXi-PERF-TBL template.
Open input parameter file (default-params.asc).
Read parameter names from input parameter file.
Create parameter suffix names from JEMX instrument number.

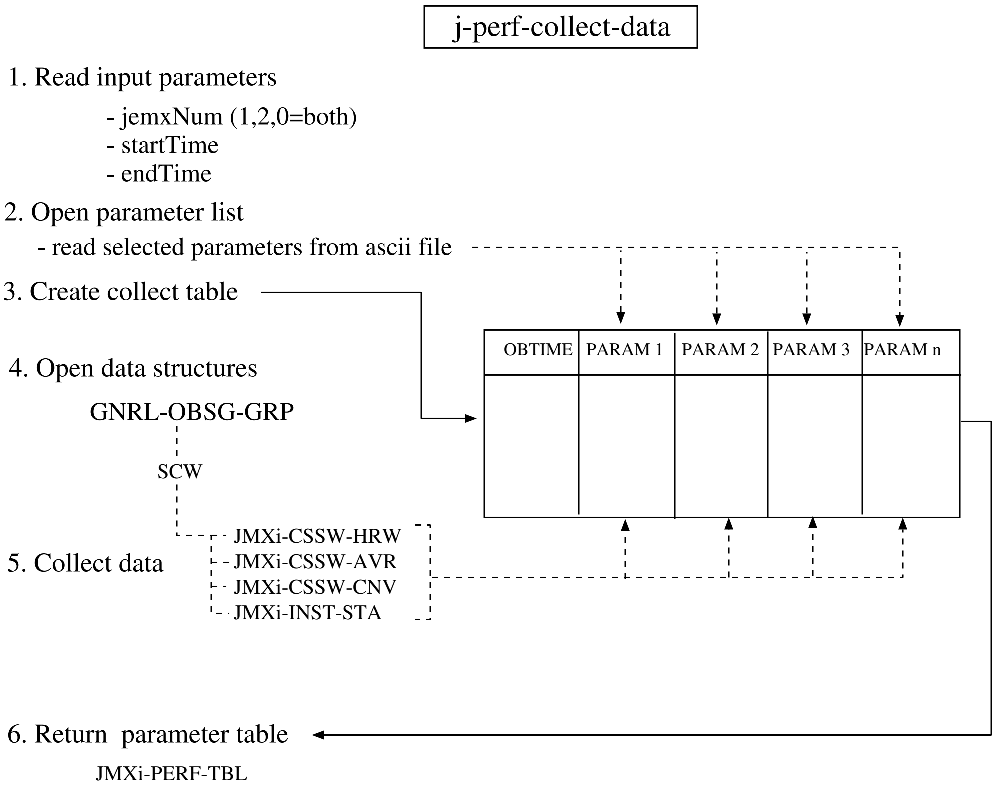

Figure 5.23: j-perf-collect-data

Create column names by combining suffix and parameter names.
Add parameter columns to the table.
Read selected parameters from observation group by given time period.
Write parameter values to the table.

Design remarks:
TBD

Parent components:
j-performance (p. 223)

Subordinate components:
j_perf_collect_GetParams
j_perf_create_table
j_perf_collect_ProcessData

Dependencies from other components: None

Pending TBDs: Parameters and resources are TBD.

Prototyping status:

ISDC context: Off-line processing, Fig. 1

Delivery date: v1.0 30/08/2002

Chapter 6

Data Dictionary

All archived data are stored as FITS extensions, each of which has a format specified in a template file with the *.tpl extension. This chapter contains a complete overview of all the data structure templates available to and used by the JEM-X ISSW. The contents of these data structures are stored in two ways: as keywords in the data header, or as values listed in columns of the table. Multiple columns can be used to make image arrays where this sort of data format is called for.

How data is split up between extensions is determined solely by the vagaries of the ISDC system, especially the PreProcessing and Archiving subsystems, so that without reference to the ISDC context the arrangement of data files may often seem arbitrary and, indeed, inelegant.

It should be remembered that for each data structure here there are actually two versions, one for each JEM-X unit. 'JMXi' actually means 'JMX1' or 'JMX2'.

All telemetry arriving at ISDC from MOC is decoded and packed into FITS extensions by the PreProcessing Pipeline. Hence all the raw data on which the ISSW gets to work, comes from PreProcessing, and is designated by ISDC level RAW. Other data supplied to the sofware comes from ISOC (auxilliary data) and the instrument team (instrument model and characteristics). All the available JEM-X data structures and the applicable general ('GNRL-' prefix) and auxilliary ('AUXL-' prefix) are listed in this chapter. However, several of these are not used at all by the ISSW:

- Some instrument characteristics extensions are used exclusively by the Observation Simulator, OSIM for generating test data and test observations.

- Some RAW data structures produced by Pre-Processing are only used by ISDC Operations System Monitoring (OSM) for real time monitoring purposes.

- Certain housekeeping data structures are used by the instrument team to monitor the performance and health of the instruments, e.g. diagnostic dump and on request housekeeping data.

6.1 Introduction to ISDC data structures

A data structure is a set of data and metadata (data that describe the data) produced by ISDC software (namely the DAL, the Data Access Layer). All the data structures are defined by template files (see DAL user manual). Some data are therefore not data structures in that sense: Most of the files that are received from outside, and some hand-written reports will be free-format files, like ascii text in the case of hand-written reports. The template files can be used by DAL to create the appropriate structure either on disk, in memory, or in shared memory. The syntax defining the template files incorporates most (if not all) the facilities offered by CFITSIO. If the data structure is written on disk, then it will appear in the form of a header data unit (an extension) of a FITS file. The data structures can therefore be thought of as extensions of a FITS file residing somewhere in the ISDC system.

6.1.1 Group data structures

The data structure management will make use of the grouping facilities offered by CFITSIO. A very common example of such groups is the science window group. It is a data structure that contains links to all data structures necessary to the analysis of the science window. In most cases, the science window group will be the only data structure that has to be given to an analysis executable. Groups of science window groups will also be very common. An observation is an example of such a group, but groups of science window groups can be made according to arbitrary criteria (e.g., all science windows where a given object is in the field of view).

6.1.2 Principles

The definition of ISDC data structures is guided by the following principles:

1. No information is lost in the conversion from telemetry. The combination of the raw data structures allow the full reconstruction of the telemetry packets.

2. The data generated by a subsytem (PreProcessing, DataPreparation, etc) is kept unchanged for the rest of the processing, i.e. al changes must lead to the creation of new data structures instead of a modification of the old.

3. Duplication of data is to be avoided as far as possible. The logical connection between entries in different data structures can be obtained by the ISDC grouping mechanism and the order of the entries in the data structure. As an example the N-th line in a table of corrected position data would contain the position of the event described in the N-th line of the raw data table.

6.1.3 Multiple classes of 'raw'

To fulfill the requirement of invertible raw data, but avoid carrying large amounts of unused information further, there are five types of raw data:

Packet raw data is the science data received on a per-packet basis, which is useful to reconstruct the packet from the raw data. It is not supposed to be used beyond DataPreparation. Packet raw data is stored in data structures whose name finishes in -PRW.

Secondary raw data is data that is received neither on a per-packet basis, nor on a per-event basis. For JEM-X the grey filter marker 'events' are a prime example of such data. Secondary raw data is stored in data structures whose name finishes in -SRW.

Raw science data is the actual event, spectrum or countrate data. Raw data is stored in data structures whose name finishes in -RAW.

Calibration Raw data refers to the raw calibration spectra transmitted as part of housekeeping. It can also be the raw electronic calibration data. Such data structures have extension -CRW

Raw Housekeeping data is unconverted HK values, many of which values, such as counters, can be used in their raw state. The raw HK data structures have extension -HRW

Consolidated event lists It is now possible for the user to work, not from the individual event data and -PRP data (OBTs), but to make a single file event list with the extensions -ALL (e.g. JMXi-FULL-ALL (p. 237), JMXi-REST-ALL (p. 241)) in which all the attributes of a particular event are collected in a single row of the -ALL data structure: RAWX, RAWY, DETX, DETY, OBT, PHA, PI etc. The -ALL data structures are created by the ISDC tool evts-pick (p. 64)

6.1.4 Full On Board Time

Since the raw times transmitted in the instrument telemetry do not contain the Central OnBoard Time, these cannot be used as they stand, and hence no FOBTs appear in the raw data. Instead j_data_preparation converts the partial OBTs from the telemetry and HK (in the case of calibration spectra) into Full OBTs, and these then appear in -PRP files.

ISDC is responsible for ensuring the correct time correlation between L_OBT (time stamps from the JEM-X DFEE) and FOBT. Consequently, there is no time correction step in j_correction.

6.1.5 INDEX tables

Where more than one instance of an extension with a given name appears in a SWG, or across SWGs, then each instance is logged in an INDEX file which distinguishes between the different versions using the values of one or more keywords common to all the extensions. A good example of an INDEX table includes the DOL of each instance of a name extension, plus a column of OBTs, one for each extension, found from the headers. This table can then be used to find data contained in a particular sort of extension, but only during a desired time inteval.

INDEX table extension names all end with the extension -IDX. For example, the index table listing all the versions of extension JMX1-FRSS-CRW is called JMX1-FRSS-IDX.

6.1.6 JEM-X specific problems

The JEM-X instrument has the capability to autonomously switch from one telemetry data mode to another within a science window [21]. Software Requirement 18.1.1 requires the analysis software to deal with the possibility that some data types may be present only for parts of the analyzed science window.

At the time of writing, five different science telemetry sub-modes have been identified. Four of them (FULL, REST, SPTI, TIME) send individual 'events' with varying amounts of information, only one (Spectral) contains data binned on board. A complimentary data format for countrate (RATE) is also produced simultaneously with those modes that have poor timing resolution: REST and SPEC.

6.2 JEM-X Data Structures

The following data structures have been defined in the ISDC Reference Platform 6.3 and in the ISDC developement template lists. They are listed here for completeness, but their full description can be found on the ISDC webpage. Each data structure has two versions, one for each JEM-X instrument: JMX1-****-*** or JMX2-****-***. In this way the two instruments are handled separately with completely independant data structures.

Under each data structure name, the ISDCLEVL to which the it belongs is given in italics. The level name is determined by the data processing level that produces the data in a given data structure. Currently, data structures that are not produced by ISSW, but merely delivered by the instrument teams or ISDC do not have any ISDCLEVL - this applies mainly to hardware parameter limits for alerts and GTIs, and Instrument Model data structures. In the list below instrument model data structures are flagged with 'level' IMOD, though this is for user information only and does not actually exist as a level.

'CC' indicates that template is now under configuration control

Name	Contents	Column Names
JMXi-ALRT-LIM-IDX	Index table for alert limits data structure: CC	VERSION VSTART VSTOP DATE RESPONSI
JMXi-ALRT-LIM	Valid limits for monitored HK parameters: CC	PAR_NAME OBT_START OBT_END MIN_VAL MAX_VAL SUB_ASSEMBLY CHECK_MODE ALERT_DELAY
JMXi-ANOD-ICL *ICAL*	Image of anode normalised event populations	X:IJC Y:Anode number
JMXi-ANOD-PRP *PRP*	Normalised event populations of each anode foroffline health monitoring	RAWX COUNTS
JMXi-ASPE-BKG *BKG*	Sun aspect angle background scaling	SUN_ASPECT SCALFACT
JMXi-ASPE-BKG-IDX *BKG*	Index table	VERSION VSTART VSTOP
JMXi-ATAL-MOD *IMOD*	Attenuation data for aluminium	ENERGY ATTLEN
JMXi-ATBE-MOD *IMOD*	Attenuation data for beryllium	ENERGY ATTLEN

JMXi-ATCU-MOD *IMOD*	Attenuation data for copper	ENERGY ATTLEN
JMXi-ATMO-MOD *IMOD*	Attenuation data for molybdenum	ENERGY ATTLEN
JMXi-ATTI-DBG *BKG*	Instrument attitude background scaling factor	RA_SCX DEC_SCX RA_SCZ DEC_SCZ SCALFACT
JMXi-ATTI-DBG-IDX *BKG*	Index table	VERSION VSTART VSTOP
JMXi-ATXE-MOD *IMOD*	Attenuation data for xenon	ENERGY ATTLEN
JMXi-AXIS-ARF *SPE*	Ancilliary response function for an on-axis source	ENERG_LO ENERG_HI SPECRESP
JMXi-BORE-MOD *IMOD*	Transformation matrix connecting instrument boresight to star trackers: make obsolete by misalignment matrices	JMATBORE
JMXi-BPL.-GRP *BPL*	back projection lists	Members: JMXi-[DMAP DALL DAST DAWL DAWP MHOL]-BPL
JMXi-BPL.-GRP-IDX *BPL*	Index table	VERSION VSTART VSTOP
JMXi-CALB-MOD *IMOD*	Telemetry format calibration spectra energy binning tables	CHANNEL
JMXi-COLL-MOD *IMOD*	Collimator model array	FOOTPRINT ARRAY:X;Y
JMXi-CONV-MOD *IMOD*	Conversion curves for HK parameters: CC	PAR_NAME OBT_START OBT_END NUM_VAL PAR_RAW PAR_CONV PAR_UNIT MOC_CURVE
JMXi-CONV-MOD-IDX *IMOD*	Index table of JMXi-CONV-MOD data structures: CC	VERSION VSTART VSTOP DB_VERS DATE RESPONSI

JMXi-CORX-MOD *IMOD*	X position correction table	CORRECTION ARRAY: X;Y
JMXi-CORY-MOD *IMOD*	Y position correction table	CORRECTION ARRAY: X;Y
JMXi-CSSW-AVG *PRP*	Time-averaged Housekeeping Parameters: CC	SWID OB_TIME OB_DURATION
JMXi-CSSW-AVR *PRP*	Time-averaged housekeeping parameters: CC	SWID OB_TIME OB_DURATION
JMXi-CSSW-CNV *PRP*	Converted housekeeping parameters: CC	OB_TIME: other columns created by Data Conversion
JMXi-CSSW-HRW *RAW*	House Keeping Telemetry Packet	Columns defined in instrument database
JMXi-DBXS-MOD *IMOD*	Extra-galactic diffuse x-ray spectrum used by JEM-X Observation Simulator	ENERGY RATE
JMXi-DALL-BPL *BPL*	d_allow array describing the ”aperture response function”	D_ALLOW
JMXi-DAST-BPL *BPL*	d_allow_st array for unpacking of the ”aperture response function”	D_ALLOW_ST
JMXi-DAWL-BPL *BPL*	d_allow_l array for unpacking of the ”aperture response function”	D_ALLOW_L
JMXi-DAWP-BPL *BPL*	d_allow_p array for unpacking of the ”aperture response function”	D_ALLOW_P
JMXi-DEAD-MOD *IMOD*	Dead time model for different event processing paths	TRIGGERS DEAD8_1 DEAD16_1
JMXi-DEAD-SCP *DEAD*	Dead times for input, buffer loss and grey filter	OB_TIME DEADTIME RATE_COR DEAD_EFF
JMXi-DETE-FLC *BIN_T*	Folded detector lightcurve	PHASE PHASEDEL RATE ERROR BRATE BERROR BARYTIME

JMXi-DETE-FLC-IDX *BIN_T*	Index over folded detector lightcurves	DATAMODE CHANMIN CHANMAX E_MIN E_MAX PERIOD PHASZERO
JMXi-DETE-LCR *BIN_T*	Binned detector lightcurve	TIME TIMEDEL RATE ERROR BRATE BERROR BARYTIME
JMXi-DETE-LCR-IDX *BIN_T*	Index of binned lightcurves	DATAMODE CHANMIN CHANMAX E_MIN E_MAX
JMXi-DETE-MOD *IMOD*	Detector model data	ENERGY FWHM TBD
JMXi-DFEE-CRW *RAW*	DFEE dump calibration telemetry: CC	SSC PACK_TIME MEM_ID ADDRESS LENGTH DUMP
JMXi-DFEE-CRW-IDX *RAW*	Index of diagnostic dump HK: CC	REVOL ERTFIRST ERTLAST
JMXi-DIAG-CPR *PRP*	Prepared full OBTs of diagnostic events: CC	OB_TIME
JMXi-DIAG-CRW *RAW*	diagnostic calibration raw data: CC	SSC LOBT_FIRST NUM_EVENTS ANODE_SWITCH HKCYCLE_TIME DFEE_TIME PULSE_HEIGHT SLOW_ANODE CAL_BACK_PLANE CAL_CATHODE CAL_RISE_TIME FAST_ANODE VETO BACK_PLANE CATHODE PROC_STATUS CALIBRATION_MARKER
JMXi-DMAP-BPL *BPL*	Detector maps defining the "aperture response function"	(3D image containing 51 slices)
JMXi-DPOS-MOD *IMOD*	Detector position accuracy determination—for simulations only	ENERGY POSRES1_0 POSRES1_5 POSRES2_0 POSRES3_0 POSRES5_0
JMXi-DUMP-CRW *RAW*	DPE Dump calibration raw data: CC	SSC PACK_TIME MEM_ID ADDRESS LENGTH DUMP
JMXi-DUMP-CRW-IDX *RAW*	Index of dump calibration data: CC	REVOL ERTFIRST ERTLAST
JMXi-ECAL-CPR *PRP*	Prepared full OBTs of calibration events: CC	OB_TIME

JMXi-ECAL-CRW *RAW*	Event calibration raw data: CC	SSC LOBT_EVENT DELTA_LOBT CAL_AMP EVENT_NUM ANODE_SWITCH SLOW_ANODE FAST_ANODE VETO BACK_PLANE CATHODE
JMXi-ECAL-ICL *ICAL*	Output table of electrion calibration fitting	To be decided
JMXi-ECAL-GRP *RAW*	ECAL group extension: CC	
JMXi-ECAL-MOD *IMOD*	Electronic calibration model	TRIGGERS DEAD8_1 DEAD16_1
JMXi-EFLT-MOD *IMOD*	Energy response due to low level cut off	ENERGY EFILTER
JMXi-ENRG-MOD *IMOD*	table of ENERG_LO and ENERG_HI used with JMXi-FBDS-MOD, JMXi-RBDS-MOD or JMXi-SBDS-MOD for spectral response matrix	ENERG_LO ENERG_HI
JMXi-EVTS-PHO *OSIM*	Raw photon-by-photon events for a source—for simulations only	OB_TIME ENERGY DETX DETY SOURCE_ID
JMXi-EVTS-SHD *BIN_I*	Shadowgram of FULL and REST events	COUNTS ARRAY: X;Y
JMXi-EVTS-SHD-IDX *BIN_I*	Index table for full imaging shadowgrams	OBTSTART OBTEND TELAPSE CHANMIN CHANMAX ROWIDEN1 ROWIDEN2 EVNTYPES
JMXi-FBDS-MOD *IMOD*	nominal energy boundaries of the PI channels (0–255) for energy measurements obtained with full and spectral-timing modes	CHANNEL E_MIN E_MAX
JMXi-FIND-RES *IMA*	Found sources from image processing	RA_OBJ DEC_OBJ ERR_RAD DETSIG COSX_JMX COSY_JMX COSZ_JMX PEAKSIZE YPOS ZPOS
JMXi-FIND-RES-IDX *IMA*	Index table of JMXi-FIND-RES data structures	OBTSTART OBTEND TELAPSE CHANMIN CHANMAX ROWIDEN1 ROWIDEN2

JMXi-FMSK-MOD *IMOD*	Map containing fine mask description	Pixels: Mask X, mask Y
JMXi-FOIL-MOD *IMOD*	Transmission curve of kaptan/Al foil	ENERGY TRANSCOEF
JMXi-FRSS-CRW *RAW*	Raw FRSS spectra, obtained on board and transmitted as part of HK: CC	COUNTS_1 COUNTS_2 COUNTS_3 COUNTS_4
JMXi-FRSS-CRW-IDX *RAW*	Index table of gain calibration spectra: CC	ERTFIRST ERTLAST PCKSTART PCKEND COBT0
JMXi-FRSS-ICL *ICAL*	Co-added raw calibration spectra sets for offline analysis	COUNTS_1 COUNTS_2 COUNTS_3 COUNTS_4
JMXi-FULB-MOD *IMOD*	Corrected upper limits of the bins used to convert 4096 ADC channels to 256 full imaging and spectral-timing energy telemetry channels	CHANNEL
JMXi-FULL-ALL *COR*	Complete event list with all RAW, PRP and COR data	RAWX RAWY PHA DELTA_TIME OB_TIME DETX DETY PI STATUS TIME TIMEDEL
JMXi-FULL-BSP *BIN_S*	Background spectral counts for detector in full imaging mode	BACKRATE BACK_ERR
JMXi-FULL-COR *COR*	Corrected event data from the Full Imaging mode: CC	DETX DETY PI STATUS
JMXi-FULL-DSP *BIN_S*	Full imaging detector Spectrum	ROWID CHANNEL RATE STAT_ERR SYS_ERR GROUPING QUALITY EXPOSURE TSTART TSTOP
JMXi-FULL-PRP *PRP*	Full on-board time information for Full Imaging events: CC	OB_TIME

JMXi-FULL-PRW *RAW*	Packet raw data for the full imaging events: CC	SSC LOBT_FIRST ROW_BEGIN ROW_NUM NUM_EVENTS NUM_GREY GREY_FIRST
JMXi-FULL-RAW *RAW*	Raw data from the Full Imaging event mode: CC	RAWX RAWY PHA DELTA_TIME
JMXi-FULL-SRW *RAW*	Grey filter change events transmitted in Full imaging mode: CC	SSC LOBT_FIRST POSITION GREY_FILTER DELTA_TIME
JMXi-GAIN-AVG *ICAL*	Gain history parameters averaged over revolution	REVOL G_ZER0_1 G_CAL_1 RES_8_1 RES_22_1 INT_CD_1 INT_NI_1 G_ZER0_2 G_CAL_2 RES_8_2 RES_22_2 INT_CD_2 INT_NI_2 G_ZER0_3 G_CAL_3 RES_8_3 RES_22_3 INT_CD_3 INT_NI_3 G_ZER0_4 G_CAL_4 RES_8_4 RES_22_4 INT_CD_4 INT_NI_4
JMXi-GAIN-CAL *PRP*	Results of detector gain calibration	RESID_1 RESID_2 RESID_3 RESID_4
JMXi-GAIN-CAL-ICL *ICAL*	Personal copy of JMXi-GAIN-CAL-IDX used for offline analysis	Same as JMXi-GAIN-CAL-IDX, plus others as required
JMXi-GAIN-CAL-IDX *PRP*	Index of detector gain calibration results	OBT_ACQ INT_TIME TSTART TSTOP G_ZER0_1 G_CAL_1 RES_8_1 RES_22_1 INT_CD_1 INT_NI_1 G_ZER0_2 G_CAL_2 RES_8_2 RES_22_2 INT_CD_2 INT_NI_2 G_ZER0_3 G_CAL_3 RES_8_3 RES_22_3 INT_CD_3 INT_NI_3 G_ZER0_4 G_CAL_4 RES_8_4 RES_22_4 INT_CD_4 INT_NI_4
JMXi-GLTC-ICL *ICAL*	Image of collected glitch histograms	TBD
JMXi-GNRL-GTI-IDX *GTI*	General index of JEMX-i gti tables: CC	GTI_NAME
JMXi-GNRL-GTI *GTI*	Standard JEMX-i gti tables: CC	OBT_START OBT_END
JMXi-GOOD-LIM-IDX	Index of parameter limits tables: CC	VERSION VSTART VSTOP DATE RESPONSI

JMXi-GOOD-LIM	All parameters of the JEM-X 1 instrument that have to be monitored and the range that defines a good interval: CC	PAR_NAME OBT_START OBT_END MIN_VAL MAX_VAL GROUP_NAME SUB_ASSEMBLY CHECK_MODE
JMXi-IMOD-GRP *IMOD*	Instrument model group. See page 250 for more details	Members: JMXi-[ATAL ATCU ATMO ATXE BORE CALB COLL CORX CORY DEAD DETE DPOS DXBS ECAL ENRG FBDS FULB MASK PLLX PLLY QEFF RBDS REGU RESB SBDS SPAG SPCB]-MOD
JMXi-IMOD-GRP-IDX	Index table over instrument model groups	VERSION VSTART VSTOP
JMXi-INST-STA *PRP*	Instrument status table: instrument mode and grey filter value as a function of OBT: CC	OB_TIME MODE GREY_FILTER
JMXi-MASK-MOD *IMOD*	Mask array model	MASK ARRAY: X;Y
JMXi-MERG-GTI *GTI*	Merged GTIs	OBT_START OBT_END
JMXi-MHOL-BPL *BPL*	Contains the 'mask hole' array with ideal hole pattern in the JEM-X mask	MASKHOLE
JMXi-MOSA-IMA *IMA2*	Mosaic'ed image	Pixel values: PHI ANGLE (CAR) or X (TAN); THETA ANGLE (CAR) or Y (TAN)
JMXi-MOSA-IMA-IDX *IMA2*	Index table of mosaic'ed images	TSTART TSTOP IMATYPE CHANMIN CHANMAX EVNTYPES E_MIN E_MEAN E_MAX
JMXi-OBS.-RES *IMA2*	Combined source data for many ScWs in observation	SWID; SOURCE_ID; NAME; RA_OBJ; ERROR_RAD; DETSIG; OFFANGLE; EBIN_NUM; E_MIN; E_MAX; FLUX; FLUX_ERR; DEADC; EXPOSURE; TSTART; TSTOP; SEL_FLAG; COSX_JMX; COSY_JMX; COSZ_JMX
JMXi-ORBI-BKG *BKG*	Orbital background scaling factors	XYZPOS SCALFACT

JMXi-ORBI-BKG-IDX *BKG*	Index table	VERSION VSTART VSTOP
JMXi-ORHK-AVG *PRP*	On Request averaged converted HK parameters: CC	SWID OB_TIME OB_DURATION other columns added by dp_average
JMXi-ORHK-AVR *PRP*	On Request averaged raw HK parameters: CC	SWID OB_TIME OB_DURATION other columns added by dp_average
JMXi-ORHK-CNV *PRP*	On Request converted HK parameters: CC	OB_TIME other columns added by DataConversion
JMXi-ORHK-HRW *RAW*	On request raw HK packet	SSC PACK_TIME
JMXi-PERF-TBL	Master table of performance parameters	TBD
JMXi-PHA1-SPE *SPE*	Restored countrate spectrum in PHA1 format	CHANNEL RATE STAT_ERR SYS_ERR QUALITY GROUPING
JMXi-PHA2-SPE *SPE*	Restored countrate spectrum in PHA2 format	SPEC_NUM ROWID CHANNEL RATE STAT_ERR SYS_ERR QUALITY GROUPING BACKFILE RESPFILE ANCRFILE
JMXi-PLLX-MOD *IMOD*	Measured x-corner positions of effective pixels	Image: x-position of lower left corner of pixel at (RAWX,RAWY)
JMXi-PLLY-MOD *IMOD*	Measured y-corner positions of effective pixels	Image: y-position of lower left corner of pixel at (RAWX,RAWY)
JMXi-PRED-MOD *IMOD*	Pixel redistribution from raw shadograms to regular shadowgrams	NUMBER LIST_IDX FRACTION
JMXi-QEFF-MOD	Quantum efficiency models	ENERGY QEFF1_0 QEFF1_5 QEFF2_0 QEFF3_0 QEFF5_0
JMXi-RADL-BKG *BKG*	Radiation level background scaling factors	RAD_LEVEL SCALFACT

JMXi-RADL-BKG-IDX *BKG*	Index table	VERSION VSTART VSTOP
JMXi-RATE-PRP *PRP*	Prepared count rate data (OBT): CC	OB_TIME
JMXi-RATE-PRW *RAW*	Packet raw data of the count rate data: CC	SSC LOBT_FIRST ROW_BEGIN ROW_NUM BIN_NUM GREY_FILTERS TIME_BINS RATE_FIRST
JMXi-RATE-RAW *RAW*	Raw count rate data: CC	DELTA_COUNT COUNT_RATE GREY_FILTER
JMXi-RBDS-MOD	Nominal energy boundaries of the PI channels(0–7) for energy measurements obtained with restricted imaging mode	CHANNEL E_MIN E_MAX
JMXi-RESB-MOD *IMOD*	Corrected upper limits of the bins used to convert original 4096 energy channels into 8 restricted imaging telemetry channels	CHANNEL
JMXi-REST-ALL *COR*	Complete event list with all RAW, PRP and COR data	REST_PHA DELTA_POS RAWX RAWY OB_TIME DETX DETY REST_PI STATUS TIME TIMEDEL
JMXi-REST-BSP *BIN_S*	Background spectra for restricted imaging events	BACKRATE BACK_ERR
JMXi-REST-COR *COR*	Corrected event data from the Restricted Imaging data mode: CC	DETX DETY REST_PI STATUS
JMXi-REST-DSP *BIN_S*	Detector spectrum for restricted imaging events	ROWID CHANNEL RATE STAT_ERR SYS_ERR GROUPING QUALITY EXPOSURE TSTART TSTOP
JMXi-REST-PRP *PRP*	Full on-board time for Restricted Imaging events: CC	OB_TIME
JMXi-REST-PRW *RAW*	Packet raw data for the restricted imaging events: CC	SSC LOBT_FIRST ROW_BEGIN ROW_NUM X_FIRST Y_FIRST SEQ_IMAGE

JMXi-REST-RAW *RAW*	Raw data from the Restricted Imaging event mode: CC	REST_PHA DELTA_POS RAWX RAWY
JMXi-REST-SRW *RAW*	Secondary raw data (image information) for restricted imaging events: CC	SSC_FIRST LOBT_FIRST ROW_BEGIN ROW_NUM LENGTH_IMAGE GREY_FIRST GREY_LAST
JMXi-RMF.-GRP-IDX	Index table	VERSION VSTART VSTOP
JMXi-RMF.-GRP *IMOD*	Redistribution matrix file group	Members: JMXi-RMF.-RSP JMXi-FBDS-MOD
JMXi-RMF.-RSP	Response data for redistribution matrix file, in JMXi-RMF.-GRP	ENERG_LO ENERG_HI N_GRP F_CHAN N_CHAN MATRIX
JMXi-RSTI-IMA-IDX *IMA*	Index of restored sky images	OBTSTART OBTEND TELAPSE CHANMIN CHANMAX ROWIDEN1 ROWIDEN2
JMXi-RSTI-IMA *IMA*	Restored sky images, skew coordinates	COUNTS ARRAY: X;Y
JMXi-SBDS-MOD *IMOD*	Energy boundaries of the PI channels(0–63) for events obtained in spectrum mode	CHANNEL E_MIN E_MAX
JMXi-SCAL-BKG *BKG*	Scaling factors for instrument background spectra: CC	DELTAOBT SCALFACT
JMXi-SCAL-DBG *BKG*	Scaling factors for diffuse background spectra: CC	DELTAOBT SCALFACT
JMXi-SCAT-BKG	Instrument background spectrum catalogue	SPE_INUM SPECTRUM RAD_AVG PIX_COUNTS SUN_ASPECT XYZPOS
JMXi-SCAT-BKG-IDX	Index table	VERSION VSTART VSTOP
JMXi-SCAT-DBG *BKG*	Diffuse background spectrum catalogue	SPE_INUM SPECTRUM RA_SCX DEC_SCX RA_SCZ DEC_SCZ
JMXi-SCAT-DBG-IDX	Index table	VERSION VSTART VSTOP
JMXi-SIDX-BKG *BKG*	Spectral indices array for instrument background: CC	INDEX ARRAY: X;Y

JMXi-SIDX-BKG-IDX *BKG*	Index of instrument background spectral models	VERSION VSTART VSTOP SIMUL RA_SCX DEC_SCX RA_SCZ DEC_SCZ SUN_ASPT SC_POS_X SC_POS_Y SC_POS_Z RAD_AVG ORBPHASE
JMXi-SIDX-DBG *BKG*	Spectral indices array for diffuse background: CC	INDEX ARRAY: X;Y
JMXi-SIDX-DBG-IDX *BKG*	Index of diffuse background spectral models	VERSION VSTART VSTOP SIMUL RA_SCX DEC_SCX RA_SCZ DEC_SCZ
JMXi-SKY.-IMA *IMA*	Sky images	PIXEL: PHI (CAR) or X (TAN); THETA(CAR) or Y(TAN)
JMXi-SKY.-IMA-IDX *IMA*	Index table of sky images	OBTSTART OBTEND IMATYPE IMADESCR IMACASE CHANMIN CHANMAX SHDIDEN EVNTYPES E_MIN E_MEAN E_MAX EXPOSURE CRVAL1 CRVAL2
JMXi-SPAG-MOD *IMOD*	Spatial gain variation of detector area	Arrays X/Y: Gain offset or Gain slope
JMXi-SPAT-BKG-IDX	Index table of Spatial instrument backgrounds	VERSION VSTART VSTOPSIMUL RA_SCX DEC_SCX RA_SCZ DEC_SCZ SUN_ASPT XYZPOS
JMXi-SPAT-BKG *BKG*	Spatial gain model for instrument background: CC	NORMALISATION ARRAY: X;Y
JMXi-SPAT-DBG *BKG*	Spatial gain model for diffuse background: CC	NORMALISATION ARRAY: X;Y
JMXi-SPAT-DBG-IDX	Index table of diffuse background models	VERSION, VSTART VSTOP SIMUL RA_SCX DEC_SCX RA_SCZ DEC_SCZ
JMXi-SPCB-MOD *IMOD*	Corrected upper limits of the bins used to convert the 4096 ADC channels into 64 spectral mode telemetry energy channels	CHANNEL
JMXi-SPEC-ALL *COR*	Complete event list with all RAW, PRP and COR data	GREY_FILTER SPECTRUM OB_TIME SPEC_CORR STATUS TIME TIMEDEL
JMXi-SPEC-BSP *BIN_S*	Time-resolved background spectra	BACKRATE BACK_ERR
JMXi-SPEC-COR *COR*	Corrected data from the spectrum format mode: CC	SPEC_CORR STATUS

JMXi-SPEC-DSP *BIN_S*	Time-resolved diffuse background spectra	ROWID CHANNEL RATE STAT_ERR SYS_ERR GROUPING QUALITY EXPOSURE TSTART TSTOP
JMXi-SPEC-PRP *PRP*	Prepared data (OBT)from the spectrum mode: CC	OB_TIME
JMXi-SPEC-PRW *RAW*	Packet raw data for the spectrum mode: CC	SSC LOBT_FIRST ROW_BEGIN ROW_NUM
JMXi-SPEC-RAW *RAW*	The raw data from the spectrum format mode: CC	GREY_FILTER SPECTRUM
JMXi-SPTI-ALL *COR*	Complete event list with all RAW, PRP and COR data	PHA DELTA_TIME OB_TIME PI STATUS TIME TIMEDEL
JMXi-SPTI-BSP *BIN_S*	Background spectra for Spectral/timing mode	BACKRATE BACK_ERR
JMXi-SPTI-COR *COR*	Corrected event data from the Spectral/Timing event mode: CC	PI STATUS
JMXi-SPTI-DSP *BIN_S*	Detector spectra from Spectral/timing mode	ROWID CHANNEL RATE STAT_ERR SYS_ERR GROUPING QUALITY EXPOSURE TSTART TSTOP
JMXi-SPTI-PRP *PRP*	Prepared data (OBT) from the Spectral/Timing event mode: CC	OB_TIME
JMXi-SPTI-PRW *RAW*	The packet raw data for spectral-timing events: CC	SSC LOBT_FIRST ROW_BEGIN ROW_NUM NUM_EVENTS NUM_GREY GREY_FIRST
JMXi-SPTI-RAW *RAW*	Raw data from the Spectral/Timing event mode: CC	PHA DELTA_TIME
JMXi-SPTI-SRW *RAW*	Automatic grey filter change events transmitted in spectral/timing mode: CC	SSC LOBT_FIRST POSITION GREY_FILTER
JMXi-SRC.-FLC *LCR*	Folded source lightcurve	PHASE PHASEDEL RATE ERROR BRATE BERROR BARYTIME
JMXi-SRC.-FLC-IDX *LCR*	Index of folded source lightcurves	SOURCEID RA_OBJ DEC_OBJ DATAMODE CHANMIN CHANMAX E_MIN E_MAX PERIOD PHASZERO

JMXi-SRC.-LCR *LCR*	Countrate lightcurves for listed sources	OB_TIME EXPOSURE RATE_i STAT_ERR_i, SYS_ERR_i, QUALITY
JMXi-SRC.-LCR-IDX *LCR*	Index of source lightcurves	SOURCEID RA_OBJ DEC_OBJ OFFANGLE DATAMODE CHANMIN CHANMAX E_MIN E_MAX
JMXi-SRCL-ARF *SPE*	Auxilliary Response Function for each of the found sources in the list	ARF_NUM ENERG_LO ENERG_HI SPECRESP
JMXi-SRCL-BSP *BIN_S*	Background spectra for found listed sources	BACKRATE BACK_ERR
JMXi-SRCL-CAT *CAT_I*	Catalogue point sources used as input to ISSW: CC	SOURCE_ID DAY_ID NAME CLASS RA_OBJ DEC_OBJ ERR_RAD SPA_MODL SPA_NPAR SPA_PARS SPE_MODL SPE_NPAR SPE_PARS VAR_MODL VAR_NPAR VAR_PARS COMMENTS SPI_FLUX_1 SPI_FLUX_2 ISGR_FLUX_1 ISGR_FLUX_2 PICS_FLUX_1 PICS_FLUX_2 JEMX_FLUX_1 JEMX_FLUX_2 E_MIN E_MAX FLUX FLUX_ERR SEL_FLAG
JMXi-SRCL-RES *IMA*	Source list resulting from ISSW, identical structure to JMXi-SRCL-CAT, different data	SOURCE_ID DAY_ID NAME CLASS RA_OBJ DEC_OBJ ERR_RAD SPA_MODL SPA_NPAR SPA_PARS SPE_MODL SPE_NPAR SPE_PARS VAR_MODL VAR_NPAR VAR_PARS COMMENTS SPI_FLUX_1 SPI_FLUX_2 ISGR_FLUX_1 ISGR_FLUX_2 PICS_FLUX_1 PICS_FLUX_2 JEMX_FLUX_1 JEMX_FLUX_2 E_MIN E_MAX FLUX FLUX_ERR SEL_FLAG COSX_JMX COSY_JMX COSZ_JMX
JMXi-SRCL-SPE *SPE*	Countrate spectra of each listed source	ROWID CHANNEL RATE STAT_ERR SYS_ERR GROUPING QUALITY EXPOSURE TSTART TSTOP and EVT_TYPE
JMXi-TIME-PRP *PRP*	Prepared OBTs for timing mode events: CC	OB_TIME
JMXi-TIME-PRW *RAW*	Packet raw data for timing mode: CC	SSC LOBT_FIRST ROW_BEGIN ROW_NUM NUM_EVENTS NUM_GREY GREY_FIRST
JMXi-TIME-RAW *RAW*	Raw data for timing mode: CC	DELTA_TIME

JMXi-TIME-SRW *RAW*	Secondary raw data: automatic grey filter changes for timing mode data: CC	SSC LOBT_FIRST POSITION GREY_FILTER
JMXi-TVER-CRW *RAW*	Time verification calibration raw data: CC	SSC PACK_TIME LOCAL_OBT
JMXi-TVER-CRW-IDX *RAW*	Index of time verification tables: CC	ERTFIRST ERTLAST REVOL
JMXi-VIGN-MOD *IMOD*	Vignetting model for source intensity correction	IMAGE: X-AXIS, Y-AXIS

6.3 INTEGRAL data structures

The following data structures are used by the JEM-X ISSW, but are not specific to these instruments. Again, for a fuller description of each data structure consult the ISDC homepage.

Name	Contents	Column Names
AUXL-ATTI-HIS	Historic attitude data: CC	POINTING_ID POINTING_TYPE SRC_ID ATT_SEQ MJD_SLEW MJD_POINTING TIME DURATION OTF_THR RA_SCX DEC_SCX RA_SCZ DEC_SCZ STAR_NUM RA_STAR DEC_STAR SUN_ASPECT CONTINGENCY APD_AMP RA_DIFF DEC_DIFF
AUXL-ATTI-PRE	Planned attitude data: CC	POINTING_ID POINTING_TYPE SRC_ID ATT_SEQ SLEW_TIME POINTING_TIME TIME DURATION OTF_THR RA_SCX DEC_SCX RA_SCZ DEC_SCZ STAR_NUM RA_STAR DEC_STAR SUN_ASPECT CONTINGENCY APD_AMP RA_DIFF DEC_DIFF
AUXL-ATTI-SNA	Snapshot attitude data: CC	POINTING_ID POINTING_TYPE SRC_ID ATT_SEQ SLEW_TIME POINTING_TIME TIME DURATION OTF_THR RA_SCX DEC_SCX RA_SCZ DEC_SCZ STAR_NUM RA_STAR DEC_STAR SUN_ASPECT CONTINGENCY APD_AMP RA_DIFF DEC_DIFF
AUXL-CRST-REF	Clock reset reference data: CC	RESET_NUM TIME
AUXL-ORBI-HIS	historic orbit data: CC	DAYBEG DAYEND EPOCH ORBIN SMAXIS OMOTIN NREC XYZPOS XYZVEL RDIST POLPOS_X POLPOS_Y POLPOS_Z POLVEL_X POLVEL_Y POLVEL_Z

AUXL-ORBI-PRE	Predicted orbital phase: CC	DAYBEG DAYEND EPOCH ORBIN SMAXIS OMOTIN NREC XYZ-POS XYZVEL RDIST POLPOS_X POLPOS_Y POLPOS_Z POLVEL_X POLVEL_Y POLVEL_Z
AUXL-TCOR-HIS	historic time correlation: CC	OB_TIME TIME ERT_CORRECTION COUPLE_VALIDITY OBT0 TIME0 GRADIENT CONSTANT CORRELATION_STATUS
GNRL-INTL-BTI	Table of BTIs across all instruments	OBT_START OBT_END; TSTART; TSTOP; BTI_TYPE; COMMENTS
GNRL-MODE-STA	Master table of instrument mode status	OB_TIME INTL_AOCS_MODE INTL_AOCS_SUBM INTL_AOCS_OTF INTL_AOCS_ESAM INTL_BCP_PID INTL_BCP_DUR INTL_BCP_OTF SPI_MODE SPI_ACS_MODE SPI_AFEE_MODE SPI_DFEE_MODE SPI_PSD_MODE IBIS_MODE IBIS_ISGR_MODE IBIS_PICS_MODE IBIS_VETO_MODE JMX1_MODE JMX1_GREY_FILT JMX2_MODE JMX2_GREY_FILT OMC_MODE OMC_CCD_MODE OMC_LED1_MODE OMC_LED2_MODE IREM_MODE
GNRL-OBSG-GRP	Observation grouping table	
GNRL-REFR-CAT	General reference catalogue for point sources	SOURCE_ID DAY_ID NAME CLASS RA_OBJ DEC_OBJ ERR_RAD SPA_MODL SPA_NPAR SPA_PARS SPE_MODL SPE_NPAR SPE_PARS VAR_MODL VAR_NPAR VAR_PARS COMMENTS SPI_FLUX_1 SPI_FLUX_2 ISGR_FLUX_1 ISGR_FLUX_2 PICS_FLUX_1 PICS_FLUX_2 JEMX_FLUX_1 JEMX_FLUX_2 E_MIN E_MAX FLUX FLUX_ERR SEL_FLAG

GNRL-SCWG-GRP	Group containing DOLs of data structures belonging to a single SWG	GROUP
GNRL-SCWG-GRP-IDX	Index of SWG grouping tables	ISDCLEVL DATE ERTFIRST ERTLAST REVOL SWID SW_TYPE SWBOUND BCPPID OBTSTART OBTEND PCKSTART PCKEND EXPID EXPIDEND PODV PODV_END TSTART TSTOP TELAPSE RA_SCX DEC_SCX RA_SCZ DEC_SCZ POSANGLE SPIMODE IBISMODE JMX1MODE JMX2MODE OMCMODE
GNRL-USER-GTI	Table containing user GTIs	OBT_START OBT_END IJD_START IJD_END UTC_START UTC_END

6.4 Instrument model data structures

All the data structures that describe the instrument (mask, collimator, detector shape etc.) are gathered together in one instrument model table, JMXi-IMOD-GRP, which has the following children:
Materials attenuation:
JMXi-ATAL-MOD JMXi-ATCU-MOD JMXi-ATMO-MOD JMXi-ATXE-MOD JMXi-ATBE-MOD JMXi-FOIL-MOD

ADC channel binning tables for PHA telemetry:
JMXi-CALB-MOD JMXi-FULB-MOD JMXi-RESB-MOD JMXi-SPCB-MOD

PI binning tables:
JMXi-FBDS-MOD JMXi-RBDS-MOD JMXi-SBDS-MOD

Energy response models :
JMXi-ENRG-MOD JMXi-EFLT-MOD

Mask, collimater and detector plate models:
JMXi-COLL-MOD JMXi-DETE-MOD JMXi-DPOS-MOD JMXi-MASK-MOD JMXi-DXBS-MOD, JMXi-VIGN-MOD, JMXi-FMSK-MOD

Event position and pixelisation models: JMXi-CORX-MOD JMXi-CORY-MOD, JMXi-PLLX-MOD, JMXi-PLLY-MOD, JMXi-REGU-MOD

Quantum efficiency:
JMXi-QEFF-MOD

Event correction models:
JMXi-SPAG-MOD JMXi-DEAD-MOD

Electronic calibration model:
JMXi-ECAL-MOD

Pointing corrections:
JMXi-BORE-MOD

Individual descriptions of these data structures can be found in the table above.

It is the responsiblity of the DNSC/SDAST Co-I (currently Niels Jørgen Westergaard) to deliver JMXi-IMOD-GRP complete with all its children whenever one (or more) of the data structures changes. This means a complete redelivery of the entire group of data structures every time there is a change. The ISDC instrument model location tool then ensures that the correct version of the instrument model group is provided to the executable based on the acquisition time of the science window being processed and the validity periods of the various versions of JMXi-IMOD-GRP.

It is felt that this arrangement is simplest for the ISSW and ISDC script developers in that each executable only has to be fed the DOL of the valid JMXi-IMOD-GRP grouping table by the pipeline script, instead of having to locate individual data structures all with different validity periods.

Appendix A

Joint Header jemx.h

The common header file is maintained by CAO and delivered as an individual part of each software package. The newest version is sent to each member of SDAST whenever the file is updated and it is the responsibility of each developer to make sure that the new version is used. The file is designed to be fully backward compatible , i.e. things tend to be added rather than removed. There are plans to deliver the header as part of a common JEM-X library, but this has not been implemented yet.

```
/*
*
* jemx.h V 7.25 2004/11/29
*
* File: jemx.h
*     Version:                7.25
*      Date:                   29th November 2004
* Author: Carol Anne Oxborrow
* Affiliation: DNSC, Copenhagen
* Component: Header for JEM-X Science Analysis
*
*       (Change History log not included here - see jemx.h file)
*/

#ifndef JEMX_H_INCLUDED
#define JEMX_H_INCLUDED

#define J_JEMX_H_EDITION "7.25"    /* Header version number string */

/* Instrument description definitions
```

```
All the definitions here can be found in the JEM-X Users Manual
*/
#define J_EVT_MODE_NUM 4    /* Number of different event modes */
#define J_NONEVT_MODE_NUM 2    /* Number of non-event modes */
#define J_TOT_MODE_NUM 6    /* Total number of data-taking modes */
#define J_MAX_GREY_FILTER 31    /* Maximum value of Grey Filter */
#define J_ORIG_CHAN_NUM 4096    /* Maximum number of energy channels
                            in JEM-X on-board data.*/
#define J_NUM_CALIB_BINS 256    /* Number of JEMX energy channels
                            in the  calibration telemetry*/
#define J_NUM_SCI_BINS 256    /* Number of in the science
                            telemetry: Full imaging, and Spectral-Timing
                            */
#define J_NUM_REST_BINS 8    /* Number of JEMX energy channels in
                            the restricted imaging telemetry */
#define J_NUM_SPEC_BINS 64    /* Number of JEMX energy channels
                            in the spectrum mode telemetry */
#define J_NUM_PI_BINS 256    /* Number of JEMX energy channels in
                            the PI binning scheme */
#define J_NUM_CALIB_SPECTRA 4    /* Number of calibration spectra in
                            each calibration package. One for
                            each instrument anode. */
#define J_NUM_GAIN_COEFFS 2    /* Number of detector gain coefficients
                            to scale calibration spectra */
#define J_RATE_INTEGRATION_TIME  0.125   /* Integration time for Count
                            Rate data in seconds */
#define J_SPEC_INTEGRATION_TIME  0.125   /* Integration time for Spectrum
                            Mode data in seconds */
#define J_NUM_X_PIXELS 256    /* Number of detector pixels on X axis*/
#define J_NUM_Y_PIXELS 256    /* Number of detector pixels on Y axis*/
#define J_MAX_SSC_VAL 16383    /* Max value of the SSC counter value */
#define J_OBT_SEC_TICKS DAL3_OBT_SECOND /* Number of OBT 'ticks' in a
                            second */
#define J_REV_SEC_TICKS 259200*J_OBT_SEC_TICKS
                            /* Number OBT ticks in a revolution */
#define J_COR_PI    3.1415926538979323846 /* Pi */
#define J_DIM_MASK    200 /* Dimension of the mask array*/
#define J_MASK_CLOSED_CELL 0    /* Value of closed mask cell */
#define J_MASK_OPEN_CELL 6    /* Value of open mask cell */
#define J_MASK_OUT_CELL 7    /* Value of out-of-mask cell */
#define J_MASK_OPEN_FRAC 0.24    /* Ratio of (open: all) mask
```

```
                            cells*/
#define J_COLL_OPEN_FRAC  0.876    /* Open fraction of collimator */
#define J_IDEAL_ILLUM     103.20 /* Area in cm^2 of the ideal
                            illuminated detector area for on axis sources.
                            This takes into account area obscured
                             by closed mask cells and collimator*/
#define J_PIXEL_OUT_OF_DETECTOR  0     /* Pixel not in active
                            detector area */
#define J_COLL_DIM 500    /*  Dimension of collimator
                            description array in JMXi-COLL-MOD  */
#define J_COR_BAD_PIXEL_OFFSET 1000.00 /* Value put in instrument model
                            tables to indicate a bad pixel */
#define J_ANODE1_ON 14    /* Value of anode status bits for
                            anode 1 only activated */
#define J_ANODE2_ON 13    /* Value of anode status bits for
                            anode 2 only activated */
#define J_ANODE3_ON 11    /* Value of anode status bits for
                            anode 3 only activated */
#define J_ANODE4_ON 7    /* Value of anode status bits for
                            anode 4 only activated */
#define J_ANODEALL_ON 0    /* Value of anode status bits for all
                            anodes activated
                            (normal data taking)*/
#define J_MAX_FULL_PKT 105    /* Maximum raw events in a FULL
                            mode telemetry packet */
#define J_MAX_REST_PKT 412    /* Maximum raw events in a REST
                            mode
                            telemetry packet */
#define J_MAX_SPTI_PKT 211    /* Maximum raw events in an SPTI
                            mode telemetry packet */
#define J_MAX_SPEC_PKT 10    /* Maximum raw spectra in a SPEC
                            modetelemetry packet */
#define J_MAX_TIME_PKT 563    /* Maximum raw events in a TIME
                            mode telemetry packet */
#define J_MAX_RATE_PKT 406    /* Maximum raw countrates in a RATE
                            mode telemetry packet */
#define J_NUM_ARF_BINS 500    /* Number of ARF bins */
#define J_VIGN_DIM 256    /*  Dimension of vigneting table
                            array in JMXi-COLL-MOD  */
#define J_NORM_AREA 100    /* Ideal detector area in cm^2, for
                            flux normalisation */
```

```
/* Line energies used for gain calibration */

#define J_XE_ENERGY 29.78    /* Energy of Xe major fluoresence line
                            Source: Table of Isotopes p. 563*/
#define J_CD_ENERGY_ONE 22.1    /* Energy of Cd first line in Kev
                            Source: Table of Isotopes p. 563*/
#define J_CD_ENERGY_TWO 25.0    /* Energy of Cd second line in KeV
                            Source: Table of Isotopes p. 563*/
#define J_FE_ENERGY_ONE 5.895    /* Energy of Fe(Mn) first line in KeV
                            Source: Table of Isotopes p. 570 */
#define J_FE_ENERGY_TWO 6.50    /* Energy of Fe(Mn) second line in KeV
                            Source: Table of Isotopes p. 570 */
#define J_CD_INT_RATIO 0.2156862    /* Intensity ratio of Cd doublet
                            lines. Source: Table of Isotopes p. 563 */
#define J_FE_INT_RATIO 0.10596    /* Intensity ratio of Fe doublet lines.
                            Source: Table of Isotopes p. 570 */


/* The following OBT bit shift values can be seen on page 15 of
DAL3GEN users manual
*/
#define J_LOBT_TICK_SHIFT 3    /* JEM-X LOBT uses 2^(-13) as LOBT
                            tick unit, not 2^(-19) seconds
                            which is INTEGRAL standard */
#define J_LOBT_MASK_SHIFT 3    /* Masked bits that do not appear in
                            telemetry LOBTs */
#define J_LOBT_FOBT_SHIFT 1    /* Extra bits in FOBT not used in
LOBT
                            */

/* Instrument mode enumeration from OSM
*/
#define JMX_MODE_OFF            11
#define JMX_MODE_STDBY_CSSW     21
#define JMX_MODE_STDBY_IASW     22
#define JMX_MODE_SAFE           23
#define JMX_MODE_SETUP          31
#define JMX_MODE_DATA           40
#define JMX_MODE_DATA_FULL      41
#define JMX_MODE_DATA_SPTI      42
```

```
#define JMX_MODE_DATA_REST     43
#define JMX_MODE_DATA_TIME     44
#define JMX_MODE_DATA_SPEC     45
#define JMX_MODE_DIAGNOSTIC    51
#define JMX_MODE_MEMORY_DUMP   52
#define JMX_MODE_CALIBRATION   61

/* HK event/trigger/rejection counter IDs
All these values are found in the JEM-X Instrument User Manual
(specifically p.167 of version 5.5 from 2002/10/10) */

#define JMX_SOFT_TRIG_ID 5199
#define JMX_ACCEPTED_EVTS_ID 5377
#define JMX_GREY_LOSS_ID 5121
#define JMX_BUFFER_LOSS_ID 5136
#define JMX_FIFO_LOSS_ID 5135
#define JMX_HIGH_ANODE_ID 5123
#define JMX_CRUDE_RTIME_ID 5128
#define JMX_FINE_RTIME_ID 5127
#define JMX_VETO_REJ_ID 5130
#define JMX_CALIB_EVTS_ID 5100
#define JMX_HIGH_BACK_ID 5134
#define JMX_LOW_BACK_ID 5132
#define JMX_HIGH_CATH_ID 5133
#define JMX_LOW_CATH_ID 5131


/* Event status byte description:
*/
#define J_COR_BAD_ANODE_SEG   4       /* Events fall on an anode segment
                                         where the entire segment is faulty */
#define J_COR_BAD_OBT 8     /* Mask value to indicate event has
                            uncertain time correlation */
#define J_COR_BAD_ANODE 16    /* Mask value to indicate event
                            comes from an anode position regarded as bad */
#define J_COR_BAD_NEIGHBOUR   32      /* Mask value for events coming from
                                         position next to a bad anode */
#define J_COR_BAD_GAIN_PARAMS 64    /* Mask value to indicate bad energy
                            determination due to poor detector
                            gain history determination */
#define J_COR_BAD_GAIN_PIXEL 128    /* Mask value to set in status
```

```
                            bytes to indicate bad energy determination
                            due to event position */
#define J_COR_BAD_XPOS_BIT 256    /* Mask value to set in status byte
                            to indicate bad X position
                            determination */
#define J_COR_BAD_YPOS_BIT 512    /* Mask value in status byte
                            to indicate bad Y position
                            determination */
#define J_COR_BAD_CAL_SOURCE 1024    /* Mask value to indicate event
                            which has probably leaked from
                            a calibration source */
#define J_COR_PRE_SETUP 2048    /* Mask value for events occurring
                            before SETUP of instrument */
#define J_COR_OUTSIDE 4096    /* Mask value for events falling
                            outside active area of the detector  */
#define J_COR_HOTSPOT 8192    /* Mask value for events coming
                            from a hotspot */

/*  Error code offsets
*
* Each start code indicates a range of error codes, relative to the
* ISDC-assigned JEMX start code J_ERROR_CODE_START (value -300000).
* Each executable has 100 error codes assigned to it and each
                     pipeline
* groups together its executables within a particular 1000-value
                        range
* where in the last 100 values are reserved for pipeline/script
                       errors.
*
* Larger blocks of code are assigned error codes as follows:
* -30xxxx => JEM-X V1 ISSW in Rev. File Pipeline
* -31xxxx => JEM-X V1 ISSW in ScW Pipeline
* -32xxxx => JEM-X V2 ISSW up to the BKG level
*           (incl. scripts which have only generic children)
* -33xxxx => JEM-X V2 ISSW from BIN_I to BIN_T
* -34xxxx => JEM-X V3 ISSW - Interactive Background analysis
* -35xxxx => JEM-X V3 - Science Format and other generic tool
                     applications
* -36xxxx => JEM-X V3 ISSW - Offline Calibration Analysis
* -37xxxx => JEM-X V3 ISSW - Performance Monitoring
* -38xxxx => Highest level pipeline scripts
```

```
* -391xxx => JEM-X Shared functions
* -392000+ => Future software expansion for JEM-X
*/


/* Data preparation  error codes for Revolution file pipeline */
#define J_PRP_FRSS_OBT_ERR -1100
#define J_CALIB_GAIN_FITTING_ERR -2100
/* Next useable number here is -2300 */
#define J_CALIB_ADC_ERR -3100

/* j_data_preparation error codes for ScW Pipeline */
#define J_PRP_EVTS_OBT_ERR -11100
#define J_PRP_RATE_OBT_ERR -11200
#define J_PRP_SPEC_OBT_ERR -11300
#define J_PRP_STATUS_TABLE_ERR -11400
#define J_PRP_VERIFY_ERR -11500

/*  j_correction starting error codes */
#define J_COR_GAIN_ERR -21100
#define J_COR_POSITION_ERR -21300
#define J_CORRECTION_SCRIPT_ERR -21900

/* j_gti script starting error code */
#define J_GTI_SCRIPT_ERR -22900

/* j_dead_time starting error codes */
#define J_DEAD_TIME_CALC_ERR -23100
#define J_DEAD_SCRIPT_ERR -23900

/* j_cat_extract script starting error code */
#define J_CAT_SCRIPT_ERR -24900

/* j_bkg_model starting error codes */
#define J_BKG_SPEC_SELECT_ERR -25100
#define J_BKG_SPAT_SELECT_ERR -25200
#define J_BKG_SCAL_CALC_ERR -25300
#define J_BKG_MODEL_SCRIPT_ERR -25900

/* j_image_bin starting error codes */
#define J_IMA_SHADOWGRAM_ERR -31100
```

```
#define J_IMAGE_BIN_SCRIPT_ERR -31900

/*  j_imaging starting error codes */
#define J_IMA_BASIC_RECON_ERR -32100
#define J_IMA_SRC_FIND_ERR -32200
#define J_IMA_COR_INTENSITY_ERR -32300
#define J_IMA_REFINED_RECON_ERR -32400
#define J_IMA_FINE_RESOL_ERR -32500

#define J_IMAGING_SCRIPT_ERR -32900

/* j_src_extract_spectra starting error codes */
#define J_SRC_SPECTRA_ERR -33100
#define J_SRC_SPECTRA_SCRIPT_ERR        -33900

/* j_src_extract_lc starting error codes */
#define J_SRC_LC_ERR -34100
#define J_SRC_EXTRACT_LC_SCRIPT_ERR -34900

/* j_bin_spectra starting error codes */
#define J_BIN_EVTS_SPECTRA_ERR -35100
#define J_BIN_SPEC_SPECTRA_ERR -35200
#define J_BIN_BKG_SPECTRA_ERR -35300
#define J_BIN_SPECTRA_SCRIPT_ERR -35900

/*  j_bin_lc starting error codes */
#define J_BIN_EVTS_LC_ERR -36100
#define J_BIN_RATE_LC_ERR -36200
#define J_BIN_BKG_LC_ERR -36300
#define J_BIN_LC_SCRIPT_ERR -36900

/* j_bkg_interactive starting error codes */
#define J_BKG_INT_SPEC_SEL_ERR -41100
#define J_BKG_INT_SPAT_SEL_ERR -41200
#define J_BKG_INT_SCAL_CALC_ERR -41300
#define J_BKG_INTERACTIVE_SCRIPT_ERR -41900

/* j_sci_format script starting error code */
#define J_SCI_SCRIPT_ERR -51900

/*  j_known_src_calib starting error codes */
```

```
/* Currently it is not known which executables will be in this
pipeline */
#define J_ICAL_COLLIMATOR_ERR -61100
#define J_ICAL_GAIN_ERR -61200
#define J_ICAL_ECAL_ERR -61300
#define J_ICAL_FRSS_ERR -61400
#define J_ICAL_ANODES_ERR -61500
#define J_ICAL_SCRIPT_ERR -61900

/* j_performance starting error codes */
#define J_PERF_COLLECT_DATA_ERR -71100
#define J_PERFORMANCE_SCRIPT_ERR -71900

/* Codes for upper level pipeline scripts jemx_sci_analysis,
jemx_scw_analysis and jemx_obs_analysis
*/
#define J_SCI_PIPE_ERR -81000
#define J_SCW_PIPE_ERR -82000
#define J_OBS_PIPE_ERR -83000


/* Codes for shared JEM-X functions
*/
#define JEMXLIB_ERR                   -90000

/* Data structure definitions
*
* All of these names can be found on the JEM-X Data Definition page
* at http://obswww.unige.ch/isdc/   (ISDC)
* or at the developer's forum site at DNSC:
* http://www.dsri.dk/~oxborrow/sdast/sdast_forum.html  (DNSC)
*
* These partial names can be used for either the JEM-X1 instrument
* or the JEM-X2 instrument depending on the prefix: JMX1 or JMX2.
* The names also refer to both the actual data structure extension
                        name
* in the ISDC fits files, and the template file *****.tpl used to
* create each data structure for the first time.
*
* This list does not include certain index table data structures
                      that
```

```
* are never used directly by the JEMX ISSW.
*
* Under each data structure is the name of the SDAST person
                   responsible
* for maintaining/creating the data structure template.
*
* Entries are ordered alphabetically by the data structure name
*       |
*       |
*      V|V
*       V
*/
#define J_ATTIHIS_DS "AUXL-ATTI-HIS" /* Historic spacecraft attitude
                         data */
/* PK */
#define J_ATTINRT_DS "AUXL-ATTI-NRT" /* Near real time spacecraft
                      attitude */
/* PK */
#define J_ORBIHIS_DS "AUXL-ORBI-HIS" /* Historic orbit data */
/* PK */
#define J_TCORHIS_DS "AUXL-TCOR-HIS" /* Historic time correlation data
*/
/* PK */
#define J_ALRT_LIM "-ALRT-LIM" /* Limits for out of bounds alerts
                            */
/* PK */
#define J_ANOD_HIS_DS "-ANOD-PRP" /* Histogram of anode
               populations*/
/* CAO */
#define J_ASPE_DS "-ASPE-BKG" /* Sun aspect angle scaling factor
                            */
/* SM */
#define J_ALUM_DS    "-ATAL-MOD" /* Aluminium attenuation data
                       table */
/* NJW */
#define J_COPPER_DS  "-ATCU-MOD" /* Copper attenuation data table
                          */
/* NJW */
#define J_MOL_DS     "-ATMO-MOD" /* Molybdenum attenuation data
                        table */
/* NJW */
```

```
#define J_ATTIBKG_DS "-ATTI-BKG" /* Attitude background scaling
                        factor */
/* SM */
#define J_ATTIDBG_DS "-ATTI-DBG" /* Attitude background scaling
                        factor */
/* SM */
#define J_XENON_DS   "-ATXE-MOD" /* Xenon attenuation data  table
                          */
/* NJW */
#define J_AXISARF_DS    "-AXIS-ARF"     /* Axis default ARF */
/* SL */
#define J_CALBCORR_DS "-CALB-MOD" /* Corrected calibration energy
                         bin */
/* NJW                            limits, in ADC channels. */
#define J_COLL_DS "-COLL-MOD" /* Collimator model information */
/* NJW */
#define J_CONV_DS "-CONV-MOD" /* Conversion curves for HK
                     parameters*/
/* NJW */
#define J_CORX_DS "-CORX-MOD" /* X position correction table. */
/* CAO */
#define J_CORY_DS "-CORY-MOD" /* Y-position correction table. */
/* CAO */
#define J_HKRAW_DS "-CSSW-HRW" /* Raw housekeeping data*/
/* CAO */
#define J_HKPRP_DS "-CSSW-CNV" /* Prepared/convered housekeeping
                           data*/
/* CAO */
#define J_DEAD_DS "-DEAD-SCP" /* Dead-time table */
/* CAO */
#define J_DEADMOD_DS "-DEAD-MOD" /* Dead-time model table */
/* CAO */
#define J_DETECTOR_DS "-DETE-MOD" /* Detector instrument model */
/* NJW */
#define J_DIAG_DS "-DIAG-CRW" /* Diagnostic mode raw data */
/* CAO */
#define J_DPOS_DS "-DPOS-MOD" /* Detector position determination
                            */
/* NJW */
#define J_ECAL_MOD_DS "-ECAL-MOD" /* ADC electronic calibration
                        models */
```

```
/* CAO */
#define J_ENRG_DS    "-ENRG-MOD" /* Default line energy table for
                          RMF*/
/* SL */
#define J_EVTSSHD_DS "-EVTS-SHD" /* Shadowgram from FULL or REST
                         events*/
/* NJW */
#define J_PIBINS_DS "-FBDS-MOD" /* PI bin limits for full,
                   spectral- */
/* CAO         timing, and calibration modes.
*/
#define J_FINDRES_DS "-FIND-RES" /* List of found sources */
/* NJW */
#define J_CALSPEC_DS "-FRSS-CRW" /* Raw calibration spectra.  */
/* CAO */
#define J_SCIBCORR_DS "-FULB-MOD" /* Corrected energy binning limits
                            for*/
/* CAO      full and spectral-timing modes
*/
#define J_FULLALL_DS "-FULL-ALL" /* Total event list for FULL
                      events */
/*cao*/
#define J_FULLBLC_DS "-FULL-BLC" /* Background lightcurve for FULL
                           */
/* PK       imaging mode */
#define J_FULLBSPEC_DS "-FULL-BSP" /* Background spectra for detector
                            in */
/* PK                             full imaging mode */
#define J_FULLCOR_DS "-FULL-COR" /* Full imaging corrected data */
/* CAO */
#define J_FULLDLC_DS "-FULL-DLC" /* Full imaging detector
                  lightcurve */
/* PK */
#define J_FULLDSP_DS "-FULL-DSP" /* Full imaging detector spectrum
                           */
/* PK */
#define J_FULLPREP_DS "-FULL-PRP" /* Full imaging prepared data */
/* CAO      (full OBT) ISDC */
#define J_FULLPKT_DS "-FULL-PRW" /* Full imaging raw packet data */
/* CAO */
#define J_FULLRAW_DS "-FULL-RAW" /* Full imaging raw data.  */
```

```
/* CAO */
#define J_FULLDMY_DS "-FULL-SRW" /* Full imaging dummy(grey)
                     events*/
/* CAO */
#define J_GAINHIST_DS  "-GAIN-CAL" /* Calibration gain history  DNSC
                           */
/* CAO */
#define J_GLITCHES_DS "-GLTC-ICL" /* Glitch history image */
/* CAO */
#define J_GMODHIST_DS "-GMOD-ICL" /* Modelled/smoothed gain history
                           */
/* CAO */
#define J_GREVHIST_DS "-GREV-ICL" /* Average gain history per
                     revolution*/
/* CAO */
#define J_IMOD_DS "-IMOD-GRP" /* Instrument model group */
/* NJW */
#define J_STATHIS_DS "-INST-STA" /* Instrument status history table
                            */
/* CAO */
#define J_MASK_DS "-MASK-MOD" /* Mask pattern data structure */
/* NJW */
#define J_GTI_ALL_DS "-MERG-GTI" /* Merged Good Time Interval list
                           */
/* NJW */
#define J_MOSA_DS "-MOSA-IMA" /* Mosaic'ed images */
/* NJW */
#define J_ORBIBKG_DS "-ORBI-BKG" /* Orbital background scaling */
/* SM */
#define J_POSICORR_DS "-POSI-MOD" /* Position accuracydetermined by
                           */
/* NJW      pre-flight calibration */
#define J_POSIKSC_DS "-POSI-CAL" /* Position accuracy refinements
                          */
/* NJW      determined by j_known_src_calib
                            */
#define J_QEFF_DS "-QEFF-MOD" /* Quantum efficiency table */
/* NJW */
#define J_RADBKG_DS "-RADL-BKG" /* Radiation level background
                       scaling */
/* SM    factors */
```

```
#define J_RATEBLC_DS "-RATE-BLC" /* Rate background lightcurve */
/* PK */
#define J_RATECOR_DS "-RATE-COR" /* Countrate corrected data */
/* CAO */
#define J_RATEDET_DS "-RATE-DLC" /* Rate detector lightcurve */
/* PK */
#define J_RATEPREP_DS "-RATE-PRP" /* Countrate prepared data (OBT)
                          */
/* CAO */
#define J_RATEPKT_DS "-RATE-PRW" /* Countrate packet raw data */
/* CAO */
#define J_RATERAW_DS "-RATE-RAW" /* Countrate raw data */
/* CAO */
#define J_RESPI_DS "-RBDS-MOD" /* PI energy binning  table for */
/* CAO      restricted mode data DNSC */
#define J_RESTBCORR_DS "-RESB-MOD" /* Corrected restricted imaging
                         energy*/
/* CAO                            bin limits, in ADC channels. DNSC*/
#define J_RESTALL_DS "-REST-ALL" /* Total event list for REST
                      events */
/*cao*/
#define J_RESTBSP_DS "-REST-BSP" /* Background spectrum for
                    restricted*/
/* PK                            imaging events */
#define J_RESTCOR_DS "-REST-COR" /* Restricted imaging corrected
                         data */
/* CAO */
#define J_RESTDET_DS "-REST-DSP" /* Detector spectrum for
                  restricted */
/* PK                             imaging events */
#define J_RESTPREP_DS "-REST-PRP" /* Restricted imaging prepared raw
                            */
/* CAO                              (event OBT) ISDC */
#define J_RESTPRW_DS "-REST-PRW" /* Restricted imaging packet raw
                          data*/
/* CAO */
#define J_RESTRAW_DS "-REST-RAW" /* Restricted imaging raw events
                          ISDC*/
/* CAO */
#define J_RESTSRW_DS "-REST-SRW" /* Restricted imaging image */
/* CAO                             secondary raw data */
```

```
#define J_RESTORED_DS "-RSTI-IMA" /* 'skew' image made by
                 correlation */
/* NJW                           in basic image reconstruction */
#define J_SPECPI_DS "-SBDS-MOD" /* PI binning table for spectral
                          mode*/
/* CAO */
#define J_IBGSCAL_DS    "-SCAL-BKG" /* Scaling for instrument
                   background */
/* SM */
#define J_DBGSCAL_DS    "-SCAL-DBG" /* Scaling for diffuse background
                           */
/* SM */
#define J_IBGSPEC_DS    "-SCAT-BKG"     /* Instrm. background spectra */
/* SM */
#define J_DBGSPEC_DS    "-SCAT-DBG"     /* Diffuse  background spectra */
/* SM */
#define J_IBGSIDX_DS "-SIDX-BKG"     /* Instr. background spectral
                       model */
/* SM                            index table*/
#define J_DBGSIDX_DS    "-SIDX-DBG"     /* Diffuse background spectral */
/* SM */                           /* model index table  */
#define J_SKYIMA_DS "-SKY.-IMA" /* Sky image */
/* NJW */
#define J_SPATGAIN_DS "-SPAG-MOD" /* Spatial gain corrections
                     determined*/
/* CAO                           by pre-flight calibration */
#define J_SPATKSC_DS "-SPAG-CAL" /* Spatial gain correction
                    refinements*/
/* SM                           determined by j_known_src_calib */
#define J_IBGSPAT_DS    "-SPAT-BKG" /* Instrm. background spatial
                       model */
/* SM */
#define J_DBGSPAT_DS    "-SPAT-DBG" /* Diffuse background spatial
                       model */
/* SM */
#define J_SPECBCORR_DS "-SPCB-MOD" /* Corrected SPEC energy bin */
/* CAO                            limits, in ADC channels. DNSC*/
#define J_SPECALL_DS "-SPEC-ALL" /* Total event list for SPEC
                      events */
/*cao*/
#define J_SPECMODI_DS "-SPEC-BSP" /* Time-resolved instrument
```

```
                     background*/
/* PK                            spectra */
#define J_SPECCOR_DS "-SPEC-COR" /* Spectral mode corrected data */
/* CAO */
#define J_DIFFSPEC_DS "-SPEC-DSP" /* Time-resolved diffuse
                  background */
/* PK                             spectra */
#define J_SPECPREP_DS "-SPEC-PRP" /* Spectral mode packet data */
/* CAO                             events */
#define J_SPECPKT_DS "-SPEC-PRW" /* Spectral mode packet data */
/* CAO                             events */
#define J_SPECRAW_DS "-SPEC-RAW" /* Spectral mode raw data */
/* CAO                            events */
#define J_SPTIALL_DS "-SPTI-ALL" /* Total event list for SPTI
                      events */
/*cao*/
#define J_SPTIBLC_DS "-SPTI-BLC" /* Background lightcurve for SPTI
                           mode*/
/* PK */
#define J_SPTIBSP_DS "-SPTI-BSP" /* Background spectra for SPTI
                        mode*/
/* PK */
#define J_SPTICOR_DS "-SPTI-COR" /* Spectral-timing mode corrected
                           data*/
/* CAO                            events */
#define J_SPTIDLC_DS "-SPTI-DLC" /* Detector background lightcurve
                           for */
/* PK                            SPTI mode*/
#define J_SPTIDSP_DS "-SPTI-DSP" /* Detector background spectra for
                            */
/* PK                            SPTI mode*/
#define J_SPTIPREP_DS "-SPTI-PRP" /* Spectral timing prepared data
                          */
/* CAO                            (full OBT) */
#define J_SPTIPKT_DS "-SPTI-PRW" /* Spectral timing raw packet data
                            */
/* CAO                            events */
#define J_SPTIRAW_DS "-SPTI-RAW" /* Spectral timing raw data.  */
/* CAO                             events */
#define J_SPTIDMY_DS "-SPTI-SRW" /* Spectral timing grey events  */
/* CAO                             events */
```

```
#define J_SRCL_ARF_DS   "-SRCL-ARF"     /* Ancilliary response function
                          */
/* SL */
#define J_SRC_BLC_DS "-SRCL-BLC" /* Background LCs for each source
                           */
/* SL */
#define J_SRC_BSP_DS "-SRCL-BSP" /* Background spectra for each
                        source */
/* SL */
#define J_SRC_CAT_DS "-SRCL-CAT" /* Point sources taken from
                     general*/
/* SL                            catalogue as input to ISSW */
#define J_SRC_LCR_DS "-SRCL-LCR" /* Countrate lightcurves for
                      listed */
/* SL                            sources */
#define J_SRC_LIST_DS "-SRCL-RES" /* List of found sources in FOV */
/* SL */
#define J_SRC_SPEC_DS "-SRCL-SPE" /* Source spectra for each source
                           */
/* SL */
#define J_TIMEALL_DS "-TIME-ALL" /* Total event list for TIME
                      events */
/*cao*/
#define J_TIMEBLC_DS "-TIME-BLC" /* Background countrate lightcurve
                            */
/* PK                            for timing mode */
#define J_TIMEDLC_DS "-TIME-DLC" /* Detector background lightcurve
                           for*/
/* PK                             timing mode */
#define J_TIMEPREP_DS "-TIME-PRP" /* Prepared data (OBT) for timing
                           mode*/
/* CAO                            events */
#define J_TIMEPRW_DS "-TIME-PRW" /* Timing mode packet raw data */
/* CAO */
#define J_TIMERAW_DS "-TIME-RAW" /* Raw timing mode data */
/* CAO     events */
#define J_TIMESRW_DS "-TIME-SRW" /* Autonomous grey filter changes
                           */
/* CAO     during timing mode data taking
                           */
#define J_TIMEVER_DS "-TVER-CRW" /* Time verification data */
```

```
/* PK */
#define J_VIGN_DS       "-VIGN-MOD"     /* Vignetting information */
/* NJW */

/* Message output controls (chatty parameter)
*/

#define J_CHATTY_VERY_QUIET 0  /* Only system messages and fatal errors */
#define J_CHATTY_QUIET      1  /* Errors and severe warnings only */
#define J_CHATTY_NORMAL     2  /* + important Log messages, extend
                    warnings */
#define J_CHATTY_VERBOSE   3  /* + program flow comments */
#define J_CHATTY_VERY_VERBOSE 4 /* + miscellaneous program control values
                           */
#define J_CHATTY_MAXIMAL     5 /* Everything you want */

#endif
```

Bibliography

[1] All documents that relate strictly to the JEM-X ISSW can be downloaded from the SDAST homepage `http://www.dsri.dk/~oxborrow/sdast/sdast_forum.html`. ISDC documents can be downloaded from `http://isdc.unige.ch` Documents concerning the JEM-X instrument can be downloaded from the DNSC (previously DSRI) homepage `http://www.dsri.dk`

[2] The latest version of this document (JEM-X ISSW ADD) may be obtained from `http://www.dsri.dk` or from `http://www.dsri.dk/~oxborrow/sdast/sdast_forum.html`

[3] Software Project Management Plan for the AD Phase, S. A. Hattel (DSRI), `http://www.dsri.dk/integral/doc/SPPMad.ps`

[4] ESA software engineering standards, ESA PSS-05-0 Issue 2, February 1991.

[5] Operation Status Monitoring, Architectural Design Document ISDC/OSMADD, Issue 0.1, August 8, 1997, `http://obswww.unige.ch/~rohlfs/osm_ad/add.ps`.

[6] Optical Monitoring Camera Instrument Specific Software Architectural Design Document, Issue 4.1, 20th August 2002

[7] INTEGRAL Science Data Centre Software Requirements Document, Issue 1.2, 22nd June, 1998,ISDC
`http://obswww.unige.ch/doc/tec/srd/srd-1-2/srd-1-2.ps.gz`

[8] Calibratrion Corrections Applied to Individual PSPC Events, Turner, T. J., George, I. M., Snowden, S. L., Yusaf, R., OGIP Calibration Memo CAL/ROS/95-010

[9] Top Level Architectural Design Document, ISDC, Issue 1.5, 11th July 2002

[10] Coding and Testing Standards, ISDC/CTS, Issue 2.5, 8th March 2002

[11] Data Access Layer Users Guide, ISDC, Issue 2.6, 26th July 2002

[12] Common Library Users Manual, ISDC, Issue 1.3.5, 12th January 2001

[13] DAL3 JEM-X User's Manual, ISDC, Issue 3.5, 21st May 2001

[14] DAL3 General User Manual, ISDC, Issue 5.1, 25th September 2003

[15] DAL3 Auxilliary User Manual, ISDC, Issue 4.4, 25th July 2002

[16] DAL3 Housekeeping Users Manual, ISDC, Issue 3.2.2, 12th Febuary 2003

[17] DAL3 Catalogue User's Manual, ISDC, Issue 2.4, 26th September 2000

[18] Parameter Interface Library Users Manual, ISDC, Issue 1.8.5, 19th September 2003

[19] Reporting Interface Library Users Manual, ISDC, Issue 3.5, 23rd June 2003

[20] isdcroot Users Manual, ISDC, Issue 1.6.0, 9th October 2000

[21] JEM-X Experimental Interface Document Part B, DSRI, Issue 5.6, 8th March 2002

[22] DFEE Software Specification Document. DSRI, Issue 2.0 4th December 2001

[23] DPE Software Specification Document. Space Research Center PAS, Warsaw. Version 1.18 12th November 2001

[24] Joint European X-Ray Monitor JEM-X, X-Ray Monitor for the INTEGRAL Mission: User Manual Issue 5.4 18th June 2002

[25] ISDC Quick Look Analysis Architectural Design Document (ISDC/ADDQLA) Issue 2.2.2 15th August 2002

[26] ISDC Catalogs and Catalog Access Architectural Design Document (ISDC/ADDCAT) Issue 1.0 December 23rd 1999

[27] ISDC Science Window Pipeline Architectural Design Document (ISDC/ADDSCW) Issue 1.6 28th August 2003

[28] ISDC INTEGRAL Burst Alert System Architectural Design Document Issue 2.4 15th may 2002

[29] ISDC Offline Scientific Analysis ADD Issue 1.1 July 31st 2002

[30] JEM-X Analysis User Manual Issue 2.0 18th July 2003

Index