Instrument Specific Software for JEM-X

Architectural Design Document

Version 7.1

Carol Anne Oxborrow
Danish Space Research Institute

with contributions by
Peter Kretschmar (ISDC)
Sami Maisala (Helsinki)
Niels Jørgen Westergaard, DSRI (Copenhagen)
Stefan Larsson (Stockholm)

August 21, 2002

DSRI

Danish Space Research Institute
Juliane Maries Vej 30
DK-2100 Copenhagen

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	26	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.
		xi	Removed list of TBD's which was only used by CAO and SAH
		All	Rearranged order of components to comply with MOU.
		164, **??**, 182	Edited contributions from SL to include components from data dictionary.
14/8/98		28	Added diagram of all the JEM-X ISSW components and subcomponents.
		16, 17, 18	Added diagrams from Peter Kretschmar
		??	Added a list of pending data structures.
24/8/98	3.0	231	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
		xi	Reinstated TBD list, with inclusions from other authors
2/9/98		161, 179,	Added PK's new components to JDataBinning to handle non-imaging background determination.
		161, 179	Moved components from JBackground to JDataBinning to handle non-imaging background analysis.
3/9/98		208	Renamed JBackground to JIntBkg for interactive, offline background analyses.
		208	Renamed and added components for interactive background analysis.
		??, **??**	Added components into image generation (QL and SA), for automatic background analysis

3/9/98		28	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	93	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		**??**, **??**, 225	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	93	CAO: Added JMXi-ELPS-CAL as input to JCorrection and JPerEventCorrections
16/10/98	3.0	21	Added new section to system overview to introduce background handling throughout the JEM-X ISSW
16/10/98	3.0	211	Included new data dictionary from PK that includes some auxillary and background data structures
	RID11	149, 171	PK: Edits to JDataBinning input list
	RID13	149, 171, 161	PK: Edits to 'dependancies' of JDataBinning and JBkgSpectrum
16/10/98	RID14	161, 179	PK: Update I-O lists of JBkgSpectrum and JBkg-Lightcurve
	RID20	64, **??**	PK: j-data-preparation,JCreateSpectrumMode: Update output lists
	RID34	**??**	PK: JMXi-FRSS-CCR, Update remarks

	RID37	**??–??**	PK:JMXi-MODE-RAW, JMXi-CHAN-COR Consolidate design remarks on use of data structures across different telemetry modes.
	RID39	171	PK: Extend scope of JLightcurveBinning
19/10/98	RID26	211	CAO: Added paragraph to intro. about grouping of idividual source files for each ScW.
	RID26	93	CAO: Added intermediate data files throughout the data correction components.
	RID31	21	NJW: Added section on background handling and image reconstruction.
20/10/98	RID02	137	NJW: Added JMXi-EVWT-BKG to inputs of image construction components.
	RID03	140	NJW: Added design remarks about handling of restricted imaging data.
	RID08	211	NJW: Added pending New Data Structures and intermediate files to list in introduction.
	RID30	137	NJW: Updated inputs/outputs
	RID07	All	ALL: Updated all components to include 'TBD' or 'None' in all empty fields as appropriate.
	3.0	28	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	208	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	189	KM: Add functionality to display JMXi-ALRT-LIM.
	RID25	208	KM: Update the 'processing' field. Closes AI04.
	RID27	**??**	KM: JFiltAutoBkg, Correct inconsistencies in the text. Closes AI06
	RID29	118, 208	KM: Added missing inputs belonging to subordinate components.
	RID31	208, **??**, **??**	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	**??**	KM: Clarified use of JMXi-GAIN-PER. This closes AI07.
	3.0	189, 26	KM: Removed JPerfAlertGeneration, JPerfFunctions, JPerfAlertLimits from JPerformance. CAO: Removed these components from overview figure and system decomposition.
2/11/98	RID23	21	Added SL's contribution to description of background handling in System Overview. This completes RID23.
	RID40	164, 182	SL: Added design remarks about determining background probabilities for photons.
	3.0	149, 171	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

		195, 93	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:
		7	Separate chapter for Software Requirements overview
		1, All	Purpose of ADD expanded to include DD
		51, 28, 26	Complete rearrangement of all components to fit into ISDCLEVL architecture
		19	Added section on ISDC levels
		36	Added chapter on detailed design
		51	Added a new figure and component for every pipeline application script, to conform with initial ISDC V2 integration of the software
		195	j_known_src_calib (KSC) becomes j_ical
		211	Data dictionary updated to ISDC template release 2.0
21/05/01	6.1	28, 29	Updated figures and overview to merge j_cor_gain_events and j_cor_gain_spectral
		51	Update of all component descriptions delivered for V2 consolidation
		46	Added error code lising to DD section
27/07/01	6.2	13, 51	Updated V2 component descriptions for latest consolidated deliveries
		48	Added section about detector coordinates
		51	Added indexing and cross-referencing of input parameters for every component description
		211	Data dictionary brought up to date with ISDC RP3.0 release templates
		??	Added a list of proposed tools that could be envisaged for the future
26/11/01	6.3	189 – 208	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:
		36	Added missing parameters, so all parameters can be indexed and cross-referenced
		214	Added missing data structures

31/03/02		51	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
		118	Automatic background handling has new improved figures
		36	Script parameters added to list of input parameter names
		214	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
August 21, 2002	7.1	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 TBD's

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].

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 z-axis. It is necessary to treat the two instruments completely separately since they will have 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 developpers and ISDC
- It formed the basis of the Detailed Design Phase of software developement
- 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.

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 delivery of code to be integrated into each ISDC version: the current version represents the state of knowledge at the time of V2 consolidation and V3 coding. There has been much effort ensuring that all components and subcomponents documented here actually reflect a piece of software code delivered to ISDC for integration.

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 Acronyms and Abbreviations

ADC Analog to Digital Converter

ADD Architectural Design Document

AH Allan Hornstrup

API Application Programming Interface

BKG Background

CAO Carol Anne Oxborrow

CTS ISDC Coding and Testing Standards document.

DAL Data Access Layer (ISDC Support software library)

DFEE Digital Front-End Electronics

DPE Data Processing Electronics

DSRI Danish Space Research Institute

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

GUI Graphical User Interface

HK Housekeeping Data

IC Instrument characteristics

IJD INTEGRAL Julian date

IRAF Format used for input parameter files

IROS Iterative Removal Of Sources

ISDC INTEGRAL Science Data Centre

ISOC INTEGRAL Science Operations Centre

ISSW Instrument Specific Software

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

JEM-X Joint European Monitor for X-rays

JH Juhani Huovelin

JMX1, JMX2 Prefix on JEM-X data structures

KM Kirsi Malmivaara

KSC Known Source Calibration

LM Logical Model

FOBT Full On-Board Time

L-OBT Local On-Board Time

MOC Mission Operations Centre

MOU Memorandum of Understanding

NJW Niels Jørgen Westergaard

OBSW On-Board Software

OBT On-Board Time

OSM Operations System Monitoring

PCFOV Partially Coded Field of View

PIL Parameter Inteface Library

PK Peter Kretschmar

QLA Quick Look Analysis

RATE Countrate data taken simultaneously with REST and SPEC data

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

SA Standard Analysis

SDAST Science Data Analysis Software Team (PK, CAO, NJW, SM and SL)

SL Stefan Larsson

SM Sami Maisala

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

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

SRD 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 Primary stage of ISDC processing (preProcessing, data preparation, automatic corrections)

V2 Secondary stage of ISDC processing: Science Analysis Pipeline, either as QLA or SA

V3 Tertiary stage of ISDC processing: advanced pipeline analysis and offline interactive tools

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 SRDs 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.

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

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

18.3 Doesn't actually contain any requirements

18.4.1 j-calib-gain-fitting (p. 84) 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. 218), including JMXi-CALB-MOD (p. 215) are corrected for ADC non-linearity, which ensures that all events, scientific and calibration, are likewise corrected.

18.4.3 All derived detector gain values are saved in an index table JMXi-GAIN-CAL-IDX with the OBT of the middle of the spectral integration period.

18.4.4 Executable j-cor-gain (p. 96) in application j-correction (p. 93) 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. 84) to provide a running update of the detector gain.

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

18.5.1 Executable j-ical-gain (p. 205) 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.

18.5.3 Inflight position corrections are derived by executable j-ical-collimator (p. 199) using the collimator footprint shadowgram.

18.5.4 Spatial resolution over the area of the detector is also determined by j-ical-collimator (p. 199).

18.6 Doesn't contain any specific requirements.

18.7.1 Derived detector gain parameters are monitored by j-performance (p. 189).

18.7.2 Changes in gain parameters between the different detector anodes are monitored by j-performance (p. 189) and studied interactively by j-ical-frss (p. 196).

18.7.3 Changes in position corrections are monitored offline by the instrument team, though preliminary checks are done interactively by j-ical-collimator (p. 199).

18.7.4 j-performance (p. 189) 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 and to some extent by j-ical-collimator (p. 199).

18.7.6 Interactive monitoring of the response matrix is provided by j-performance (p. 189).

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. 149) and j_bin_lc j-bin-lc (p. 171), 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. 217), JMXi-RESB-MOD (p. 220), JMXi-SPCB-MOD (p. 222), are corrected for ADC non-linearity.

18.10.2 Executable j-cor-gain (p. 96) 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. 103), using preflight calibration tables and if necessary inflight calibration tables determined by j-ical-collimator (p. 199).

18.10.4 Executable j-cor-gain (p. 96) 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. 205).

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.

18.10.7 Both executables in j-correction (p. 93) 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. 107) handles all determination of good time intervals (GTIs).

18.10.8 Executables j-cor-gain (p. 96) and j-cor-position (p. 103) handle both FULL and REST events.

18.11.1 Use of effective PHA binning table JMXi-FULB-MOD (p. 217) ensures ADC non-linearity correction of all spectral-timing events.

18.11.2 Executable j-cor-gain (p. 96) 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. 222) ensures that all SPEC spectra are corrected for ADC non-linearity.

18.13.2 Executable j-cor-gain (p. 96) performs temporal gain corrections on the SPEC spectra.

18.14 Contains no specific requirements.

18.15.1 Executable j-bin-evts-spectra (p. 153) 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. 175) 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. 175) are also energy-resolved. The binning formats are determined by input parameters.

18.17.1 Application j-bkg-model (p. 118) calculates a background estimate automatically for every pointing. An iteractive tool, j-bkg-interactive (p. 208) allows offline fine-tuning of background models.

18.17.2 Executable j-bkg-spec-select (p. **??**) finds suitable spectral background models for each pointing. New background spectra are created from observations by j-bin-bkg-spectra (p. 161).

18.17.3 Executable j-bkg-spat-select (p. **??**) provides suitable spatial background models automatically for every pointing.

18.18.1 Application j-imaging (p. 137) 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. 137) are stored in data structures with the same format, JMXi-SKY.-IMA (p. 221) and the different images indexed in table JMXi-SKY.-IMA-IDX (p. 221).

18.18.3 It is not clear yet to what extent the JEM-X image data structures conform to those for SPI and IBIS, especially where pixel size and image size are concerned.

18.18.4 Both j-ima-basic-recon (p. 140) and j-ima-refined-recon (p. 147) correct for spatial and spectral background counts as provided in models from j-bkg-model (p. 118).

18.19.1 Application j-imaging (p. 137) produces sky images, found source lists and estimates of uncertainties on found source positions.

18.19.2 Application j-image-bin (p. 131) 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-refined-recon (p. 147) is an iterative method for producing high quality sky images and point source estimates.

18.19.4 Application j-imaging (p. 137) produces full FOV images which include all found sources.

18.19.5 The IROS procedure that is part of j-ima-refined-recon (p. 147) returns shadowgrams and image of events left over after removal of all found sources.

18.20.1 Application j-imaging (p. 137) produces refined images using j-ima-refined-recon (p. 147).

18.20.2 Details of the collimator, mask support structure etc. are applied during image processing in application j-imaging (p. 137).

18.20.3 Neighbouring images can be merged by using the ISDC generic tool for image mosaicing.

18.21.1 An initial list of found point sources is created by j-ima-src-find (p. 142) prior to refined image processing, for use in QLA.

18.21.2 Coarser, quicker, shadowgram binning is done by j-image-bin (p. 131) for use by QLA.

18.21.3 During QLA only j-ima-basic-recon (p. 140) is used to make coarse sky images quickly.

18.21.4 This requirement cancelled – not iterative processing of images is done during QLA, it is simply too time consuming.

18.21.5 Sky images are produced in QLA including all the detected sources, though these have coarser time and energy resolution and are of lower quality than for SA.

18.21.6 This requirement also cancelled–since no iterative removal of sources is done as part of QLA, neither is there a returned residual image (see 18.21.4).

18.22.1 Executable j-ima-cor-intensity (p. 145) 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. 167) 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. 167).

18.24.2 Executable j-src-spectra (p. 167) 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. 161) and j-bin-bkg-lc (p. 179).

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. 80).

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. 75)

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.

Inflight dead time monitoring The dead time of the instrument for various hardware trigger rates is calculated offline by j-ical-dead (p. 202).

ADC Linearity Monitoring Automatic pipeline component j-calib-adc (p. 89) checks that all 43 amplifier chains are still working linearly and produces alerts when any of them show non-linear changes in behaviour.

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 gain
- To remove detector fingerprints from the received shadowgrams
- 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 and lightcurves 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 update fingerprint correction tables if this is needed.

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 discoverd (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 developpers 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 on-board 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 etc. 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 j-data-preparation (p. 64) 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_prp_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 only place where distinctly different processing is used for SA and QLA is in the image reconstruction applications, where the output usually created for QLA is used as input to the iterative process which creates high quality images in Standard Analysis (j_ima_refined_recon). This latter executable is relatively time-consuming and cannot be used in QLA.

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 below.

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 is clear that some evolution of the ISSW can be expected after the launch of INTEGRAL, especially in the Commissioning Phase. It is 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 no separate 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 is improved however, by simply mosaicing the sky images produced by individual science windows. This will be done by a generic ISDC tool, and so does not appear in this ISSW ADD.

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.

3.3 ISDC System Context

The figures on pages 16, 17 and 18 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. Sun/Solaris) on several workstations. Software has been developed either in the intended environment at ISDC, or on similar machines elsewhere.

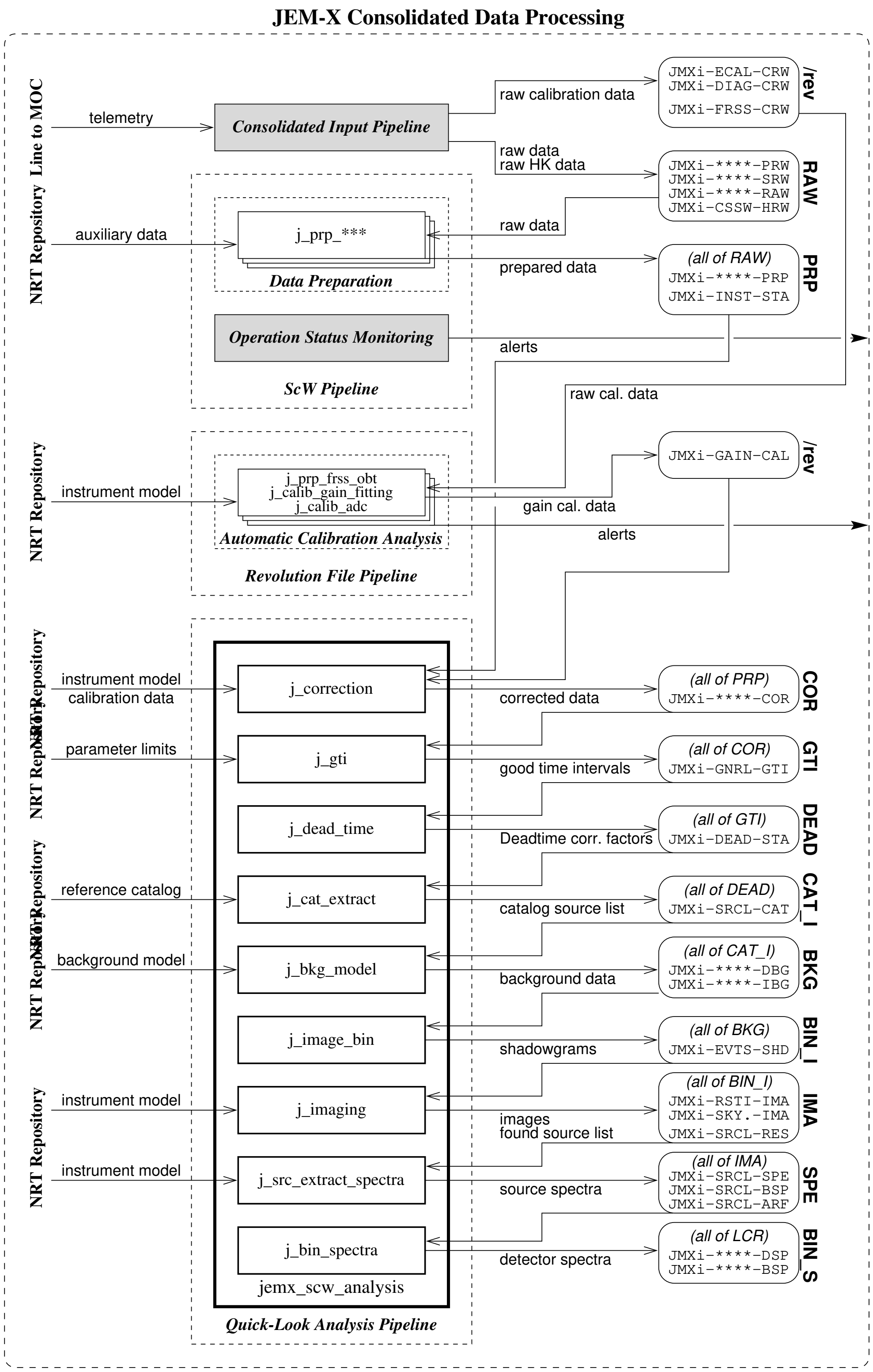

Figure 3.1: JEM-X Near real time data processing

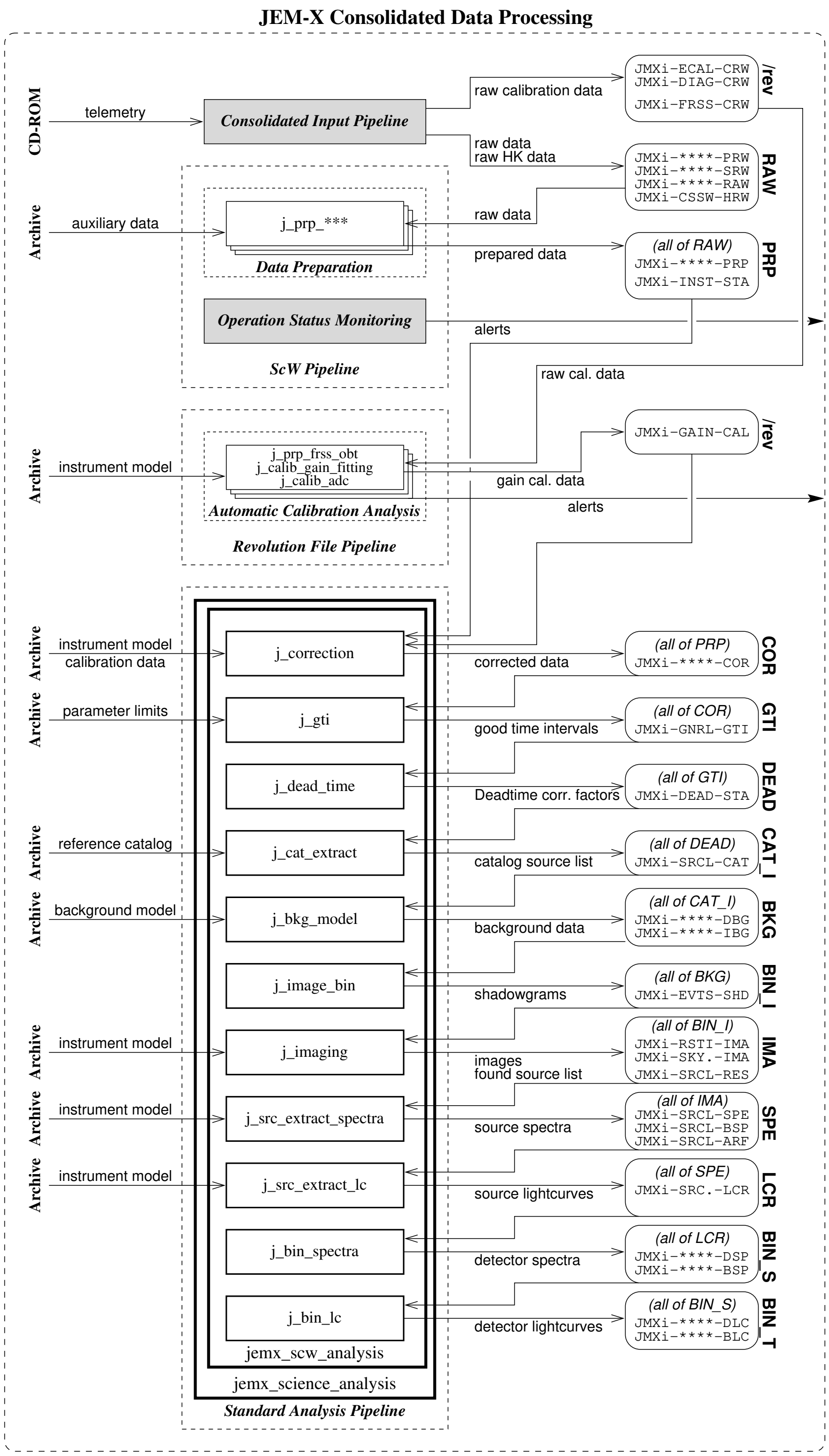

Figure 3.2: JEM-X Consolidated data processing

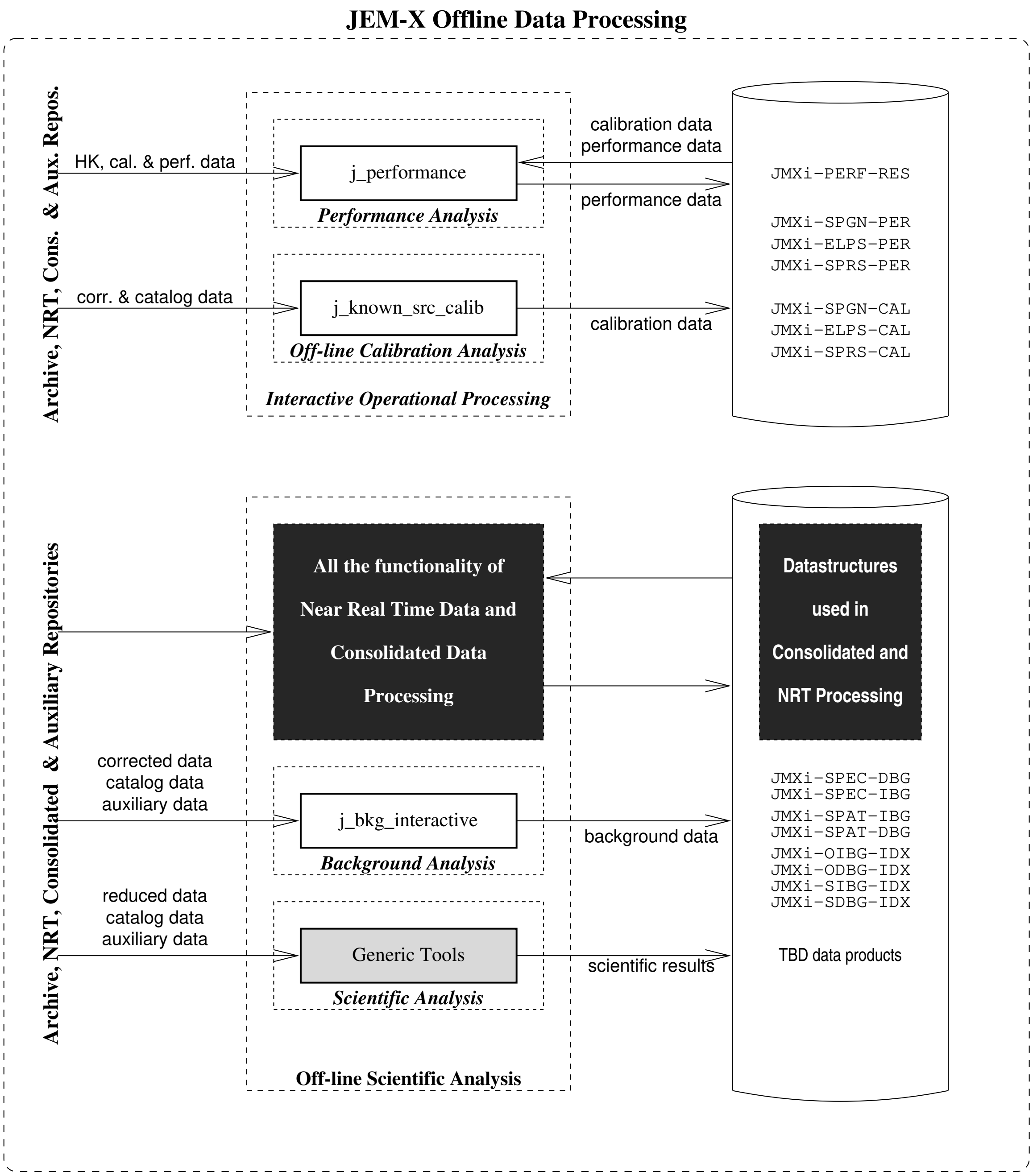

Figure 3.3: JEM-X Offline data processing

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.

For JEM-X the following levels are defined:

Science Window Pipeline	
RAW	Raw data as produced by PreProcessing
PRP	Prepared data, including Full OBTime information
Science Analysis	
COR	Corrections for instrumental fingerprints
GTI	Good-Time Interval calculation
DEAD	Deadtime history calculation
CAT_I	Catalog 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

Table 3.1: Analysis Levels defined for JEM-X

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

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`.

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.

3.6 The Background Problem

One of the most difficult problems faced in analysing the science data from JEM-X will be that of how to correct for 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. Major part of the instrument background does not come via the mask. It arises when photons and particles interact with the bulk material near the detector. The signal from the sky background, on the other hand, is processed and modified similarly with the signal from the observed targets.

In this section we discuss how the background problem will be 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 selected 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 observations.

- Spectral background models are selected by **j-bkg-spec-select**. These models are background catalogue index tables. Each detector pixel has an index value which corresponds to a spectrum model. The selection criteria are same as in *j*-bkg-spat-select.

- 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 will be 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 actual verification of the methodology requires realistic simulations of observations including instrumental and sky background. The instrumental background estimates will be acquired with the aid of the mass model of INTEGRAL.

- The usage of the blank field observation alternative requires that a well determined background observing programme will be executed during the in-flight calibrations of *INTEGRAL.*

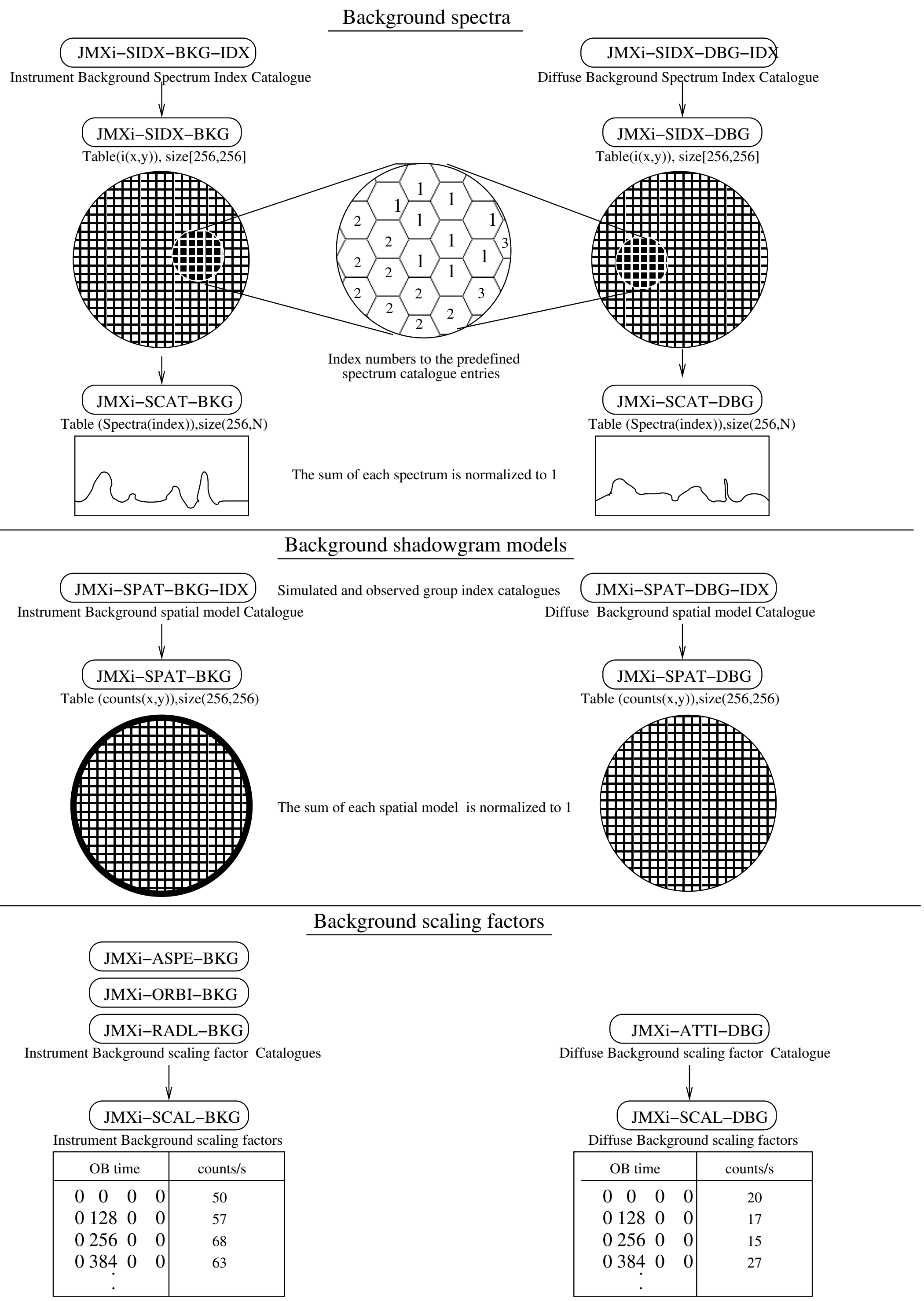

Figure 3.4: Background handling overview

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
- 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 remove the flat background flux. Spatial variations in background flux over the detector will 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

The figures on pages 28 and 29 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.

3.7.1 PRP: j_data_preparation

This pipeline application is the last level of the Science Window Pipeline and continues the pre-processing work begun by ISDC to turn raw telemetry into raw data packaged into fits extensions. By the end of PreProcessing and j_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.

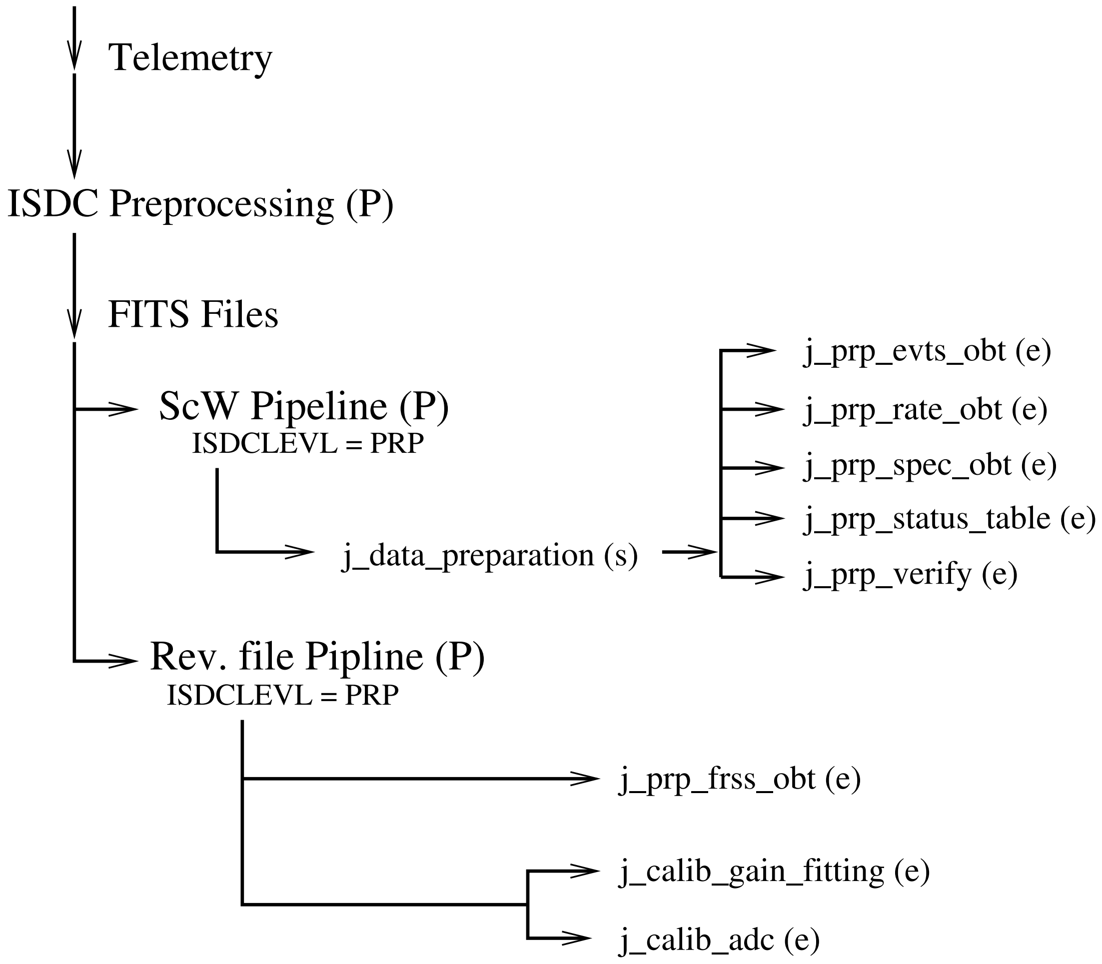

Figure 3.5: JEM-X V1 Data Preparation Overview: ‘P’ indicates an analysis pipeline; ‘s’ denotes a pipeline script and ‘e’ is an executable

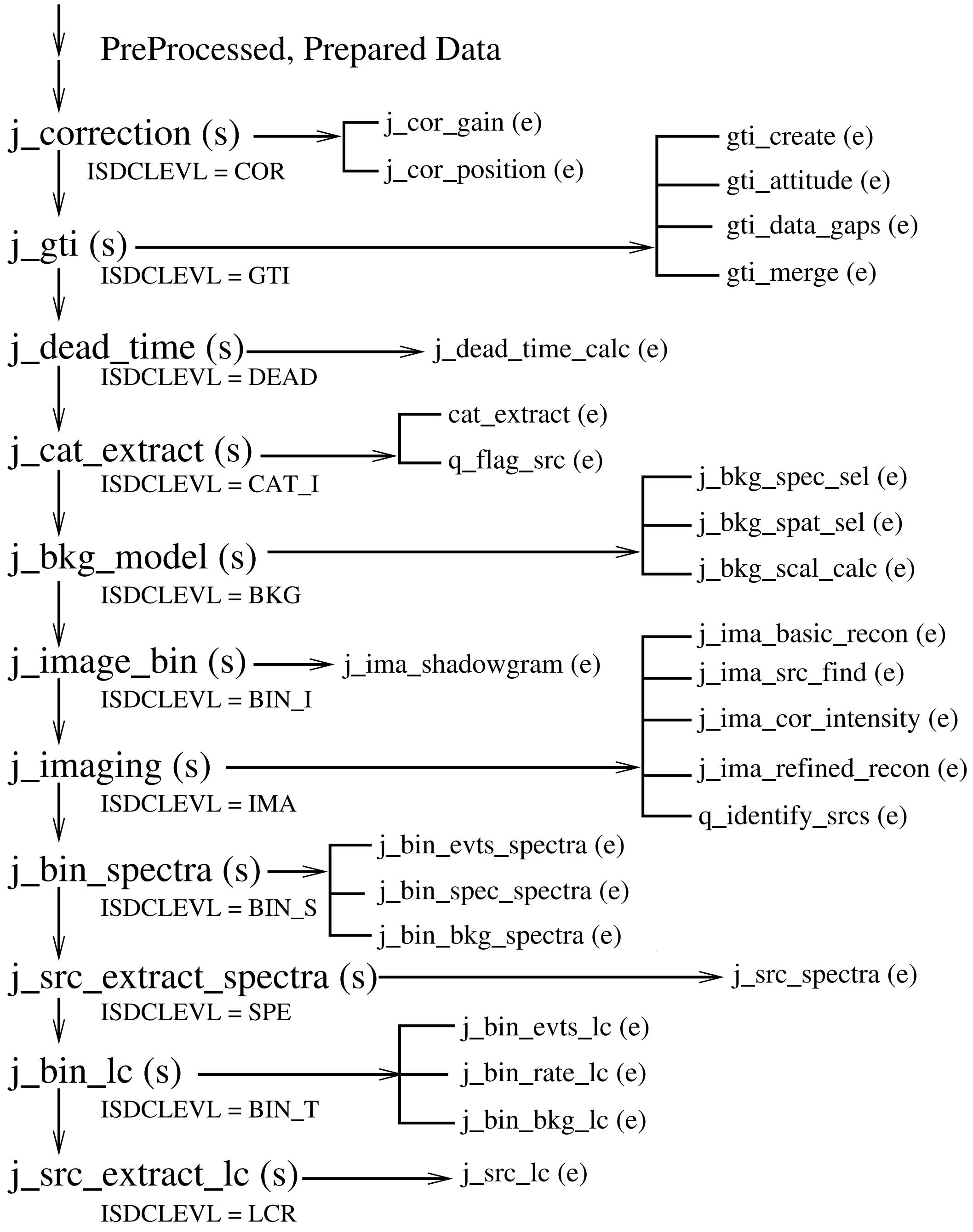

Figure 3.6: JEM-X Science Analysis Pipeline Overview: 'P' indicates an analysis pipeline; 's' denotes a pipeline script and 'e' is an executable

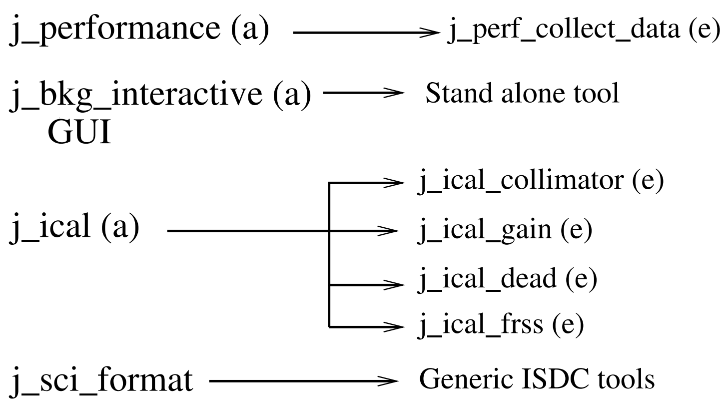

Figure 3.7: JEM-X Offline Analysis Overview: ‘a’ denotes an interactive application; ‘GUI’ is a graphical user interface and ‘e’ is an executable

This application is part of ISDC V1.

Some data preparation is perform 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 data (FULL, REST, SPTI and TIME).

j_prp_rate_obt Generates full OBTs for countrate data (RATE).

j_prp_spec_obt Generates full OBTs for spectral 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.

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 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.

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. 218) by tables produced by j-ical (p. 195).

No corrections are made to the photon positions due to bulging of the detector window because no arrival angle can be assigned to individual photons in a coded mask detector prior to image deconvolution and source finding. This correction however, is expected to be very small. Any systematic errors in event positions are removed by j_cor_position.

At the end of each correction executable every event is graded according to the uncertainty on the determination of its: position, energy, OBT and grey filter value. Currently an 8-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.45 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 done separately for the four different detector anode sections for FULL and REST data, and averaged over the detector area for SPTI and SPEC data. 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.

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 tool j-ical (p. 195).

3.7.3 GTI: j_gti

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

gti_create Creates separate lists of GTIs based on different criteria for various housekeeping parameters

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. It is part of ISDC V2

j_dead_time_calc Creates history table of instrument dead-time based on housekeeping records of number of hardware triggers and the deadtime model table JMXi-DEAD-MOD (p. 215).

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 current FOV

q_flag_src Flags sources in the list which are subject of an observing proposal for the observation.

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_select 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_select 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 deconvolves the binned shadowgrams to make sky images. For QLA only basic reconstruction is performed, followed by source finding and intensity correction. Standard analysis also uses the more time consuming refined image reconstruction which takes the coarser basic images as its input.

This application is part of ISDC V2, except for j_ima_refined_recon which is part of ISDC V3.

j_ima_basic_recon Makes coarse sky images from binned shadowgrams

j_ima_src_find Finds point sources within the sky images

j_ima_cor_intensity Corrects found source flux estimates for known variations in detected intensity due to collimator shadowing of off-axis sources.

j_ima_refined_recon Produces superior image reconstruction using more time-consuming iterative deconvolution techniques, and more detailed detector information. Results from j_ima_basic_recon are used as input. This executable also performs Iterative Removal of Sources (IROS), and updates the found-source list produced by j_ima_src_find.

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

3.7.9 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.

3.7.10 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.

3.7.11 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.

3.7.12 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.

3.7.13 Offline: j_sci_format

Generic ISDC tools format the various end products for all instruments so that they can be used as input for standard analysis tools like XSPEC. Currently this application is very poorly defined and depends largely on what ISDC decides should be done across the board for all instruments. This application is part of ISDC V3

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. 84) 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 using ftools or other available utilities.

3.7.15 Offline: j_ical

This application groups together a number of interactive calibration tools intended for use by the in instrument team to investigate the instruments' behaviour in space. Where significant changes from pre-flight calibration tables are found, new correction tables will be generated by this tool and delivered by the instrument team in a new version of the instrument model group JMXi-IMOD-GRP (p. 218).

These interactive tools will not have a complete GUI since the are not intended for use by general users, but only as a convenience for the instrument team to standardise their inflight calibration. New correction tables will only be issued if the instrument team determines that this is necessary: there is no automatic output of this tool that is used by the ISSW.

This application is part of ISDC V3.

j_ical_frss Allows an instrument specialist to monitor and re-process the gain parameters, FRSS calibration spectra and source intensities and so achieve optimum calibration of the detector. These values can also be monitored over a range of revolutions.

j_ical_collimator Derives new tables of X and Y position corrections based on the collimator footprint shadowgram produced during multiple slews and observations in which the instruments are illuminated from all sides. Creates new versions of JMXi-CORX-MOD (p. 215) and JMXi-CORY-MOD (p. 215).

j_ical_energy Calculates the detector gain across the area of the detector using Xenon lines generated in the detector itself that appear in all data. Slew and empty field data sets will be used for this analysis and new versions of JMXi-SPAG-MOD (p. 222) are created.

j_ical_dead Performs analysis of the event delta-times (time between events) for different hardware trigger rates to determine if there has been any change in the dead time model of the detectors in flight. New versions of JMXi-DEAD-MOD (p. 215) are generated.

3.7.16 Offline: j_bkg_interactive

Calculates the background model for any observation, interactively in the off-line processing. The user can select the background model from a catalogue of background observations, and can also choose a method for finding the scaling factor to be applied to the chosen model background. The various background models, split into spectral and spatial components, are stored in a catalogue, sorted by attitude and orbit parameters. For more details on background handling within different applications see below.

This application is part of ISDC V3

j_bkg_interactive Interactively constructs a background model using scaled catalogued models for use with imaging data of sources.

j_bkg_int_spec_sel Allows a user interactively to examine and select a background spectral model from the catalogue.

j_bkg_int_spat_sel Allows a user interactively to examine and select a spatial background model from the catalogue.

j_bkg_int_scal_calc Allows a user interactively to determine a scaling factor to make the background model chosen from the catalogue applicable to current observation.

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
- Any tools required specifically to analyse data used in proposals generated by the JEM-X consortium (these could be developped in collaboration with the PI on a specific proposal)

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, applicatin 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. 38 , relDist p. 42 , gridNum p. 39 , parrefsearchNum
- q_flag_srcs[25]: errPos p. 39 , logicType p. 40

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

Name	Description
alrtLevel	Maximum level of alerts produced by j-prp-verify (p. 80)
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
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 , SPE-backCorr
baseDir	Base directory string of repository Also known as: general-baseDir
backRes	Optional DOL for background spectra from event binning
bgrMethod	Background subtraction method to use: 0=flat; 1=Model
bkgCor	Perform background correction in j-ima-shadowgram (p. 134)?
bkgScale	DOL to override JMXi-SCAL-BKG, Also known as: j-ima-shadowgram-bkgScale , j-bin-bkg-spectra-bkgScale jemx_science_analysis flag: processing simulated data?
bkgSpatial	DOL to override JMXi-SPAT-BKG, Also known as: j-ima-shadowgram-bkgSpatial , j-bin-bkg-spectra-bkgSpatial
bkgSpatIdx	DOL to override JMXi-SPAT-BKG-IDX
bkgSpecCat	DOL to override JMXi-SCAT-BKG
bkgSpecMap	DOL to override JMXi-SIDX-BKG, Also known as: j-ima-shadowgram-bkgSpecMap , j-bin-bkg-spectra-bkgSpecMap
bgrMethod	Choice of background handling method used by j-src-spectra (p. 167) Also known as: SPE-bgrMethod
binFactor	Performance monitoring binning factor. 0='full'

Name	Description
chanHigh	Array of Upper boundaries of channel bins if nChanBins>0. Also known as: BIN-I-chanHigh
chanLow	Array of Lower boundaries of channel bins if nChanBins>0 Also known as: BIN-I-chanLow
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
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
dbgScale	DOL to override JMXi-SCAL-DBG, Also known as: j-ima-shadowgram-dbgScale , j-bin-bkg-spectra-dbgScale
dbgSpatial	DOL to override JMXi-SPAT-DBG Also known as: j-ima-shadowgram-dbgSpatial , j-bin-bkg-spectra-dbgSpatial
dbgSpatIdx	DOL to override JMXi-SPAT-DBG-IDX
dbgSpecCat	DOL to override JMXi-SCAT-DBG
dbgSpecMap	DOL to override JMXi-SIDX-DBG, Also known as: j-ima-shadowgram-dbgSpecMap , j-bin-bkg-spectra-dbgSpecMap
deadCorr	Perform dead time correction? Also known as: SPE-deadCorr
deadTime	DOL to override JMXi-DEAD-SCP
distFuzz	Allowed fuzziness in source-matching distances Also known as: IMA-distFuzz
ecalPrp	DOL of JMXi-ECAL-CPR
ecalRaw	DOL of JMXi-ECAL-CRW
endLevel	ISDC level at which to end processing—scripts only

Name	Description
errPos	Acceptable variation in identifying a flagged source Also known as: 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) Also known as: BIN-I-evtType , SPE-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
fullCor	DOL to override JMXi-FULL-COR Also known as: j-cor-both-fullCor , j-ima-shadowgram-fullCor , j-bin-evts-spectra-fullCor
fullPrp	DOL to override JMXi-FULL-PRP Also known as: j-cor-both-fullPrp , j-ima-shadowgram-fullPrp , j-bin-evts-spectra-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
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
goodTime	DOL to override JMXi-MERG-GTI or JMXi-GNRL-GTI
gridNum	Number of steps on one side of source finding search grid Also known as: IMA-gridNum
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. 167)
gtiScNames	Names of spacecraft GTIs to merge Also known as: GTI-gtiScNames
hkType	Type of housekeeping to use for perforamance table. 0=AVGR, 1=RAW and 2=CONV

Name	Description
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
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
inObsGrp	Input observation group DOL
instMod	DOL to JMXi-IMOD-GRP
instStat	DOL to override JMXi-INST-STA
inSWG	Input Science Window Group DOL
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. 175)
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
maxNumTrials	Maximum number of trials in source search Also known as: IMA-maxNumTrials
mergedName	Names (keywords) of the GTI extensions to merge 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
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

Name	Description
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 |nPhaseBins| equal sized bins for cyclic folding. Size of unequal sized bins is given by vector parameter phaseBins Also known as: BIN-S-nPhaseBins
numPixelSubDiv	Number of pixel subdivisions in j-imaging (p. 137) transforming from scew to detector coordinates Also known as: IMA-numPixelSubDiv
ogDOL	DOL of observation group—input to observation pipeline
ogID	Observation group label—only needed if no ogDOL given
orbiScale	DOL to override JMXi-ORBI-BKG
osimData	Scripts flag for processing on simulated data Also known as: general-osimData
outBkgLC	DOL to override JMXi-SRCL-BLC for source lightcurves
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
outSrcArf	DOL of override JMXi-SLSP-ARF or JMXi-SLLC-ARF
outSrcLC	DOL of override JMXi-SRCL-LCR
outSrcSpec	DOL to override JMXi-SRCL-SPE
outSWG	Output Science Window Group DOL
parNames	Name 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 Also known as: BIN-S-phaseBins
precisionLevel	Speed vs. Precision level for j-src-spectra (p. 167)
qlaCat	DOL of QLA sources catalogue. Also known as: QLA-qlaCat
qlaMode	Flag—processing for QLA (or SA)? Also known as: QLA-qlaMode
radiusLimit	Only events out to this many mm from centre of detector used for imaging Also known as: IMA-radiusLimit

Name	Description
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
radlScale	DOL to override JMXi-RADL-BKG
ratePrp	DOL to override JMXi-RATE-PRP
ratePrw	DOL to override JMXi-RATE-PRW
rateRaw	DOL to override JMXi-RATE-RAW
rateSrw	DOL to override JMXi-RATE-SRW
refCat	DOL of ISDC Reference Catalogue of sources for scripts only
relDist	Allowed uncertainty-scaled distances between matched sources. Also known as: IMA-relDist
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-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
rowSelect	CFITSIO selection string for events Also known as: BIN-S-rowSelect
rstiRes	Override DOL of JMXi-RSTI-IMA
shdType	Type of shadowgram to bin: skew or standard? Also known as: BIN-I-shdType
shdRes	DOL of override JMXi-EVTS-SHD
searchRad	Range of search around nominal found sources Also known as: IMA-searchRad
simulated	Using simulated background models? Also known as: BKG-simulated
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

Name	Description
specPrw	DOL to override JMXi-SPEC-PRW
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-spectra-sptiCor
sptiPrp	DOL to override JMXi-SPTI-PRP Also known as: j-cor-gain-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
startLevel	ISDC analysis level on which to begin processing—scripts only
swgDOL	j_scw_analysis: DOL of science window group to analyse
threshold	Limit for size of corrections due to vignetting in imaging Also known as: IMA-threshold
timePrp	DOL to override JMXi-TIME-PRP
timePrw	DOL to override JMXi-TIME-PRW
timeRaw	DOL to override JMXi-TIME-RAW
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 Also known as: BIN-I-timeStart , BIN-S-timeStart , SPE-timeStart
timeStep	Integration time in seconds for a single lightcurve bin or a spectrum in time-resolved spectral binning. Also known as: BIN-S-timeStep , SPE-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 Also known as: BIN-I-timeStop , BIN-S-timeStop , SPE-timeStop
usrCat	DOL of User (private) Catalogue of sources—scripts only
vignCorr	Perform vignetting corrections? Also known as: SPE-vignCorr

Suggested levels of chatiness 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. 218). 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. 215) and JMXi-FULB-MOD (p. 217) each have 256 bins for FRSS, FULL and SPTI data; JMXi-RESB-MOD (p. 220) has 8 bins for REST data and JMXi-SPCB-MOD (p. 222) 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. 222) 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. 217), JMXi-RBDS-MOD (p. 220) and JMXi-SBDS-MOD (p. 221) 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:

1 Bad X position determination. Could be due to uncertainty in the position corrections, a bad pixel, missing telemetry or non-imaging data format.

2 Bad Y position determination. Same causes as above.

4 Poor detector gain determination. Could be due to rapid variation in detector gain due to high voltage switch-on; poor or missing calibration spectra; uncertainty in the spatial gain variations or a bad pixel

8 Event occurs on a known bad spot on the detector

16 Event occurs outside the merged GTIs

32 Currently unused

64 Uncertain grey filter value. Could be caused by missing telemetry.

128 Currently unused

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-select	-325,100 to -325,199
		j-bkg-spat-select	-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	j-bkg-int-spec-sel	-341,100 to -341,199
	background tool	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	j-sci-format	-351,900 to -351,999
-361,000	Interactive	j-ical-collimator	-361,100 to -361,199

	calibration	j-ical-energy	-361,200 to -361,299
		j-ical-dead	-361,300 to -361,399
		j-ical-diagnostic	-361,400 to -361,499
		j-ical-frss	-361,500 to -361,599
		j-ical	-361,900 to -361,999
-371,000	JEM-X Performance	j-perf-list-display	-371,100 to -371,199
		j-perf-plot-display	-371,200 to -371,299
		j-perf-analysis	-371,300 to -371,399
		j-performance	-371,900 to -371,999
-380,000	Upper level scripts	jemx-sci-analysis	-381,000 to -381,999
		jemx-scw-analysis	-382,000 to -382,999
		jemx-obs-analysis	-383,000 to -383,999

-390,000 and above: reserved for future components

4.5 Detector positions

Initially the position of an event is given as raw detector pixel numbers, 0–255, where value (0,0) lies outside the active detector area. RAWX and RAWY are integers. During the Automatic Correction and Calibration, these detector values are corrected for known errors in the position determination across the detector area. Then the corrected pixels (floats) are converted to millimeters from the center of the detector. These floating point values are then saved in the corrected data extensions as DETX and DETY.

The scheme for working this out is shown in Figure 4.5.

Each collimater cell, with spacing 7.05 mm, covers 7 raw detector pixels, and the origin of the detector coordinates is at [127.5, 127.5] expressed as corrected pixels. The values used to perform this co-ordinate transformation are saved as keywords in the detector model extension JMXi-DETE-MOD, which also contains an array mapping the detector area and pixel performance. The keywords are the standard ones for fits arrays:

- CRPIX1 = 128.5 Coordinate reference point on axis 1 in pixels
- CRPIX2 = 128.5 Coordinate reference point on axis 2 in pixels
- CRVAL1 = 0.00 Reference point in detector coordinates (mm) on axis 1
- CRVAL2 = 0.00 Reference point in detector coordinates (mm) on axis 2
- CDELT1 = 1.0071428 (7.05/7.0) Coordinate scaling factor along axis 1
- CDELT2 = 1.0071428 (7.05/7.0) Coordinate scaling factor along axis 2

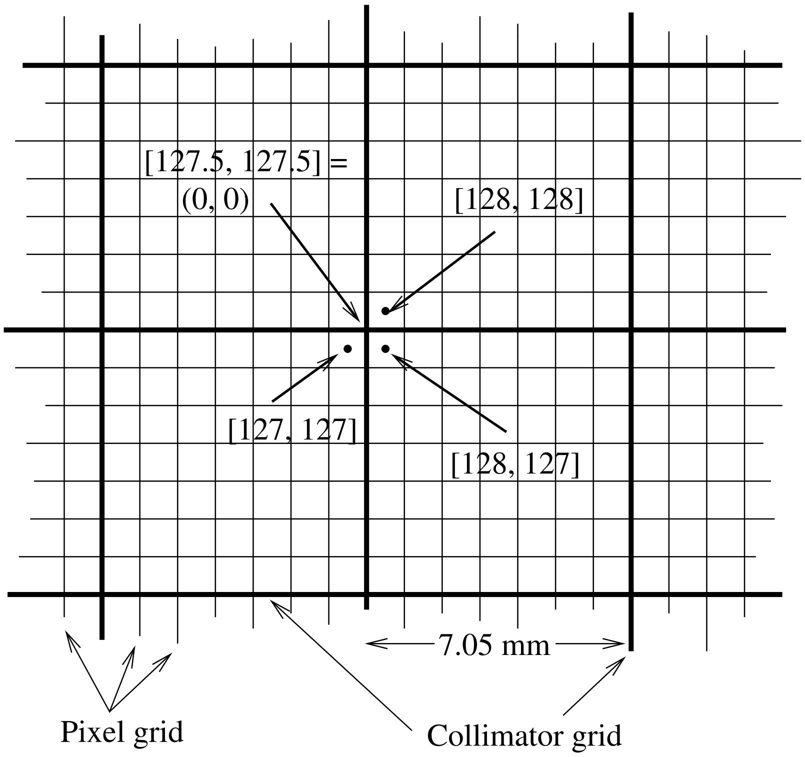

Figure 4.1: Detector co-ordinates: raw pixel positions are shown in square brackets and corrected detector positions (mm) are shown in round brackets

The reference point is at (128.5, 128.5) because the fits convention runs from 1–256, not 0–255, so one must be subtracted from the CRPIX values to get raw detector pixels.

See component j-cor-position (p. 103) for other details of position correction. After this executable, all event positions (including all imaging processing) are treated in millimeters from centre of the detector.

4.6 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'.

4.6.1 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. 224) contains the count-rate spectra of the listed sources.
- JMXi-SRCL-BSP (p. 223) contains the corresponding background spectra.
- JMXi-SRCL-ARF (p. 223) 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-SRCL-LCR (p. **??**) contains the count-rate light-curves of the listed sources and their corresponding background light-curves.
- JMXi-SRCL-ARF (p. 223) 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.

4.6.2 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.

4.6.3 Usage with XSPEC

The X-ray spectrum analysis program XSPEC version 12 can handle the spectra in the output formats of j_src_spectra with no further format change.

4.6.4 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.

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,211.

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. 211. 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.

Pending TBD's Things that are not yet solved/cleared/decided.

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

Delivery date For delivered components this is the delivery date of the version described in this document. Otherwise an estimated date of first delivery is given.

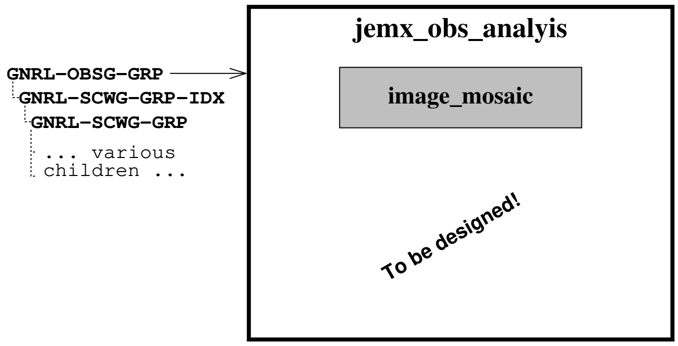

Figure 5.1: Analysis script **jemx_obs_analysis** to run scientific analysis of a single Science Window.

5.2 jemx-obs-analysis

Type: Analysis Script **Responsible:** ISDC (P. Kretschmar)

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: TBW

Outputs: TBW

ogDOL p. 41	DOL of Observation Group to be analyzed
jemxNum p. 40	JEM-X detector number
chatter p. 38	Verbosity level; 2 = normal
nChanBins p. 40	Number of energy-resolved images existing per ScW
IC-Group p. 40	DOL of Instrument Characteristics master group
IC-Alias p. 40	Selection alias for Instrument Characteristics

Resources: See subordinate components

Processing:
For each energy band call image-mosaic selecting this band.

Design remarks:
Still to be developed further

Parent components: jemx-science-analysis(p. 55)

Subordinate components:
image-mosaic ...

Dependencies from other components: jemx-scw-analysis must have run on all Science Windows before.

Completion status: **0%**

ISDC context: Science Analysis

Delivery date: v1.2 30/07/2002

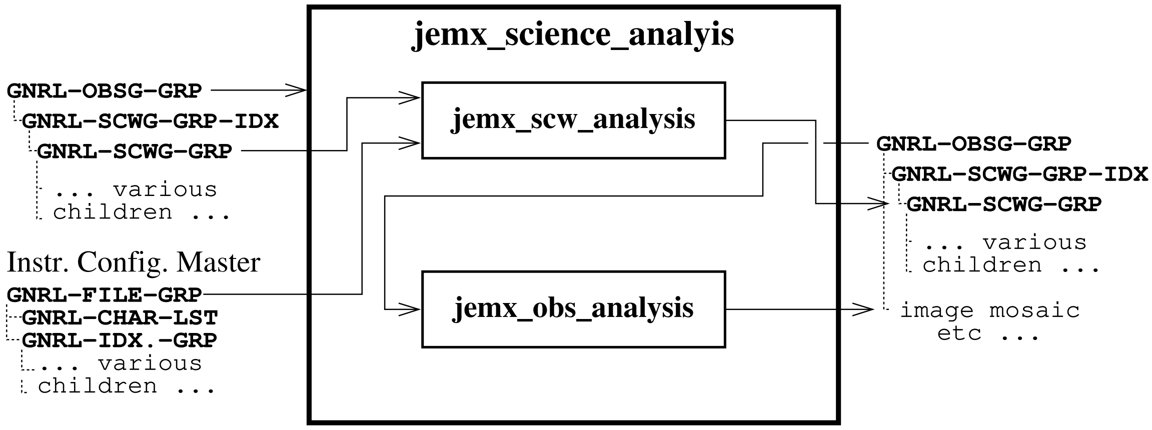

Figure 5.2: Analysis script **jemx_science_analysis** to run scientific analysis of an Observation Group.

5.3 jemx-science-analysis

Type: Analysis Script **Responsible:** ISDC (P. Kretschmar)

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. 41	DOL of Observation Group to be analyzed
ogID p. 41	Label of OG (only required if ogDOL not given)
baseDir p. 37	Base Repository Directory
jemxNum p. 40	JEM-X detector number
startLevel p. 43	Analysis level at which analysis begins
endLevel p. 38	Analysis level at which analysis finishes
chatter p. 38	Verbosity level; 2 = normal
clobber p. 38	Clobber existing output data?
refCat p. 42	DOL of Reference Catalog
usrCat p. 43	DOL of User Catalog
osimData p. 41	Are we using data produced by OSim?

IC-Group p. 40	DOL of Instrument Characteristics master group
IC-Alias p. 40	Selection alias for Instrument Characteristics
COR-gainHist p. 39	DOL of Index Group with gain change history
COR-gainModel p. 39	Gain history smoothing model option
GTI-attTolerance p. 37	Accepted attitude variability [arc sec]
GTI-limitTable p. 40	DOL of table with parameter limits (”=take from IC)
GTI-gtiJemxNames p. 39	Names of instrumental GTIs to be included in merged GTI
GTI-gtiScNames p. 39	Names of spacecraft GTIs to be included in merged GTI
QLA-qlaCat p. 41	DOL of catalog with sources of interest for QLA
QLA-qlaMode p. 41	Called for QLA processing?
CAT-I-radiusMin p. 42	Low limit for the position selection
CAT-I-radiusMax p. 42	High limit for the position selection
CAT-I-fluxDef p. 39	Column used for flux selection
CAT-I-fluxMin p. 39	Low limit for flux selection
CAT-I-fluxMax p. 39	High limit for flux selection
CAT-I-class p. 38	Object class of sources to select
CAT-I-date p. 38	IJD for the public data
BKG-simulated p. 42	Use simulated background models?
BIN-I-nChanBins p. 40	Number of energy-resolved shadowgrams to create
BIN-I-chanLow p. 38	Lower channel boundaries for shadowgrams
BIN-I-chanHigh p. 38	Upper channel boundaries for shadowgrams
BIN-I-timeStart p. 43	Start of time interval [IJD] (-1 = use swgDOL bounds)
BIN-I-timeStop p. 43	End of time interval [IJD] (-1 = use swgDOL bounds)
BIN-I-evtType p. 39	Event type (0=FULL, 1=REST, -1=both)
BIN-I-backCorr p. 37	Perform background correction?
BIN-I-shdType p. 42	Type of shadowgram 0 : skew, 1 : standard
IMA-radiusLimit p. 41	[mm] Detector counts will only be used to this radius
IMA-ignoreAnode p. 40	Anode numbers (up to four numbers) to ignore
IMA-threshold p. 43	Vignetting lower limit
IMA-numPixelSubDiv p. 41	Number of pixel subdivisions
IMA-minDetLev p. 40	Minimum detections level for source acceptance
IMA-maxNumTrials p. 40	Maximum number of trials in the source search
IMA-minDist p. 40	Minimum source separation [arcmin]
IMA-relDist p. 42	Maximally allowed value for reldist
IMA-fluxLimit p. 39	Lower flux limit for alerts of unfound sources [Counts/cm/s^2]
IMA-searchRad p. 42	Range of search grid around nominal found source positions
IMA-gridNum p. 39	Number of grid steps on one side of search grid
IMA-distFuzz p. 38	‘Fuzziness’ allowed in relative distances

SPE-timeStart p. 43	Time start in IJD of first spectrum (negative means use SWG start)
SPE-timeStop p. 43	Time start in IJD of last spectrum (negative means use SWG end)
SPE-timeStep p. 43	Length of time bins in seconds
SPE-backCorr p. 37	Correct for background?
SPE-vignCorr p. 43	Correct for vignetting?
SPE-deadCorr p. 38	Correct for instrumental dead time?
SPE-bgrMethod p. 37	Background subtraction method (0=Flat,1=Model)
SPE-evtType p. 39	Event type (0=FULL, 1=REST, -1=both)
BIN-S-timeStart p. 43	Time start in IJD of first spectrum (negative means use SWG start)
BIN-S-timeStop p. 43	Time start in IJD of last spectrum (negative means use SWG end)
BIN-S-timeStep p. 43	Length of time bins in seconds
BIN-S-nPhaseBins p. 41	Number of phase bins (0=no, ¡0=abs(N) equally spaced bins)
BIN-S-phaseBins p. 41	Phase bin sizes, separated by space (e.g: 0.1 0.2 0.3 ...)
BIN-S-rowSelect p. 42	CFITSIO selection string on events (optional)

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
jemx-obs-analysis

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: v2.3 30/07/2002

5.4 jemx-scw-analysis

Type: Analysis Script **Responsible:** ISDC (P. Kretschmar)

Purpose: See subordinate components

Function: Run a full scientific analysis for JEM-X on a single Science Window.

Inputs: See subordinate components

Outputs: See subordinate components

Parameters:

swgDOL p. 43	DOL of Science Window Group to be analyzed
startLevel p. 43	Analysis level at which analysis begins
endLevel p. 38	Analysis level at which analysis finishes
general-jemxNum p. 40	JEM-X detector number
general-chatter p. 38	Verbosity level; 2 = normal
general-clobber p. 38	Clobber existing output data?
general-osimData p. 41	Are we using data produced by OSim?
general-baseDir p. 37	Base Repository Directory
IC-Group p. 40	DOL of Instrument Characteristics master group
IC-Alias p. 40	Selection alias for Instrument Characteristics
COR-gainHist p. 39	DOL of Index Group with gain change history
COR-gainModel p. 39	Gain history smoothing model option
GTI-attTolerance p. 37	Accepted attitude variability [arc sec]
GTI-limitTable p. 40	DOL of table with parameter limits (”=take from IC)
GTI-gtiJemxNames p. 39	Names of instrumental GTIs to be included in merged GTI
GTI-gtiScNames p. 39	Names of spacecraft GTIs to be included in merged GTI
GTI-mergedName p. 40	Name of merged GTI to be used in analysis
refCat p. 42	DOL of Reference Catalog
usrCat p. 43	DOL of User Catalog
QLA-qlaCat p. 41	DOL of catalog with sources of interest for QLA
QLA-qlaMode p. 41	Called for QLA processing?
CAT-I-radiusMin p. 42	Low limit for the position selection
CAT-I-radiusMax p. 42	High limit for the position selection
CAT-I-fluxDef p. 39	Column used for flux selection

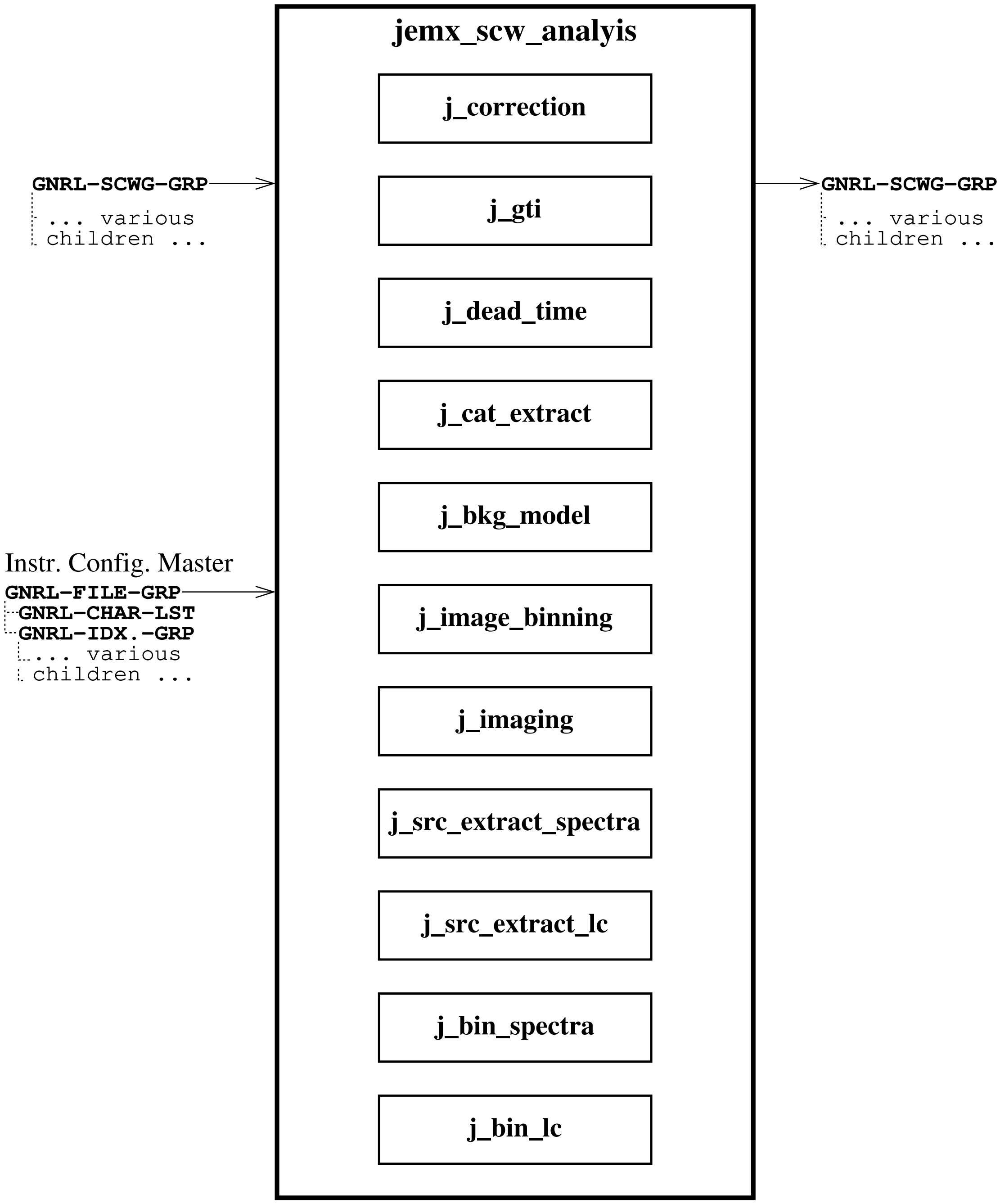

Figure 5.3: Analysis script **jemx_scw_analysis** to run scientific analysis of a single Science Window.

CAT-I-fluxMin p. 39	Low limit for flux selection
CAT-I-fluxMax p. 39	High limit for flux selection
CAT-I-class p. 38	Object class of sources to select
CAT-I-date p. 38	IJD for the public data
BKG-simulated p. 42	Use simulated background models?
BIN-I-nChanBins p. 40	Number of energy-resolved shadowgrams to create
BIN-I-chanLow p. 38	Lower channel boundaries for shadowgrams
BIN-I-chanHigh p. 38	Upper channel boundaries for shadowgrams
BIN-I-timeStart p. 43	Start of time interval [IJD] (-1 = use swgDOL bounds)
BIN-I-timeStop p. 43	End of time interval [IJD] (-1 = use swgDOL bounds)
BIN-I-evtType p. 39	Event type (0=FULL, 1=REST, -1=both)
BIN-I-backCorr p. 37	Perform background correction?
BIN-I-shdType p. 42	Type of shadowgram 0 : skew, 1 : standard
IMA-radiusLimit p. 41	[mm] Detector counts will only be used to this radius
IMA-ignoreAnode p. 40	Anode numbers (up to four numbers) to ignore
IMA-threshold p. 43	Vignetting lower limit
IMA-numPixelSubDiv p. 41	Number of pixel subdivisions
IMA-minDetLev p. 40	Minimum detections level for source acceptance
IMA-maxNumTrials p. 40	Maximum number of trials in the source search
IMA-minDist p. 40	Minimum source separation [arcmin]
SPE-timeStep p. 43	Length of time bins in seconds
SPE-timeStart p. 43	Time start in IJD of first spectrum (¡0 = use SWG start)
SPE-timeStop p. 43	Time start in IJD of last spectrum (¡0 = use SWG start)
SPE-backCorr p. 37	Correct for background?
SPE-vignCorr p. 43	Correct for vignetting?
SPE-deadCorr p. 38	Correct for instrumental dead time?
SPE-bgrMethod p. 37	Background subtraction method (0=Flat,1=Model)
SPE-evtType p. 39	Event type (0=FULL, 1=REST, -1=both)
BIN-S-timeStart p. 43	Time start in IJD of first spectrum (negative means use SWG start)
BIN-S-timeStop p. 43	Time start in IJD of last spectrum (negative means use SWG end)
BIN-S-timeStep p. 43	Length of time bins in seconds
BIN-S-nPhaseBins p. 41	Number of phase bins (0=no, ¡0=abs(N) equally spaced bins)
BIN-S-phaseBins p. 41	Phase bin sizes, separated by space (e.g: 0.1 0.2 0.3 ...)
BIN-S-rowSelect p. 42	CFITSIO selection string on events (optional)

Note: The parameters of this script are mainly passed on to lower level scripts which do the actual analysis calling executables. Parameters with a prefix of “general-” are passed on to all children, parameters beginning with “COR-”, “GTI-”, ... are passed on to the script corresponding to the analysis level COR, GTI, ... Most of these parameters will usually be hidden.

Resources: See subordinate components

Processing:

Read in parameters
Determine which analysis levels are actually executed
if (execute(COR)) call j-correction
if (execute(GTI)) call j-gti
if (execute(DEAD)) call j-dead-time
if (execute(CAT_I)) call j-cat-extract
if (execute(BKG)) call j-bkg-model
if (execute(BIN_I)) call j-image-binning
if (execute(IMA)) call j-imaging
if (execute(SPE)) call j-src-extract-spectra
if (execute(LCR)) call j-src-extract-lc
if (execute(BIN_S)) call j-bin-spectra
if (execute(BIN_T)) call j-bin-lc

Design remarks:
This intermediate level script became necessary to allow a possible parallelization of the analysis (many Science Windows in parallel).

The parameter interface will evolve further

Parent components: jemx-science-analysis(p. 55)

Subordinate components:
j-correction
j-gti
j-dead-time
j-cat-extract
j-bkg-model
j-image-binning
j-imaging
j-src-extract-spectra
j-src-extract-lc
j-bin-spectra
j-bin-lc

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: v2.2 30/07/2002

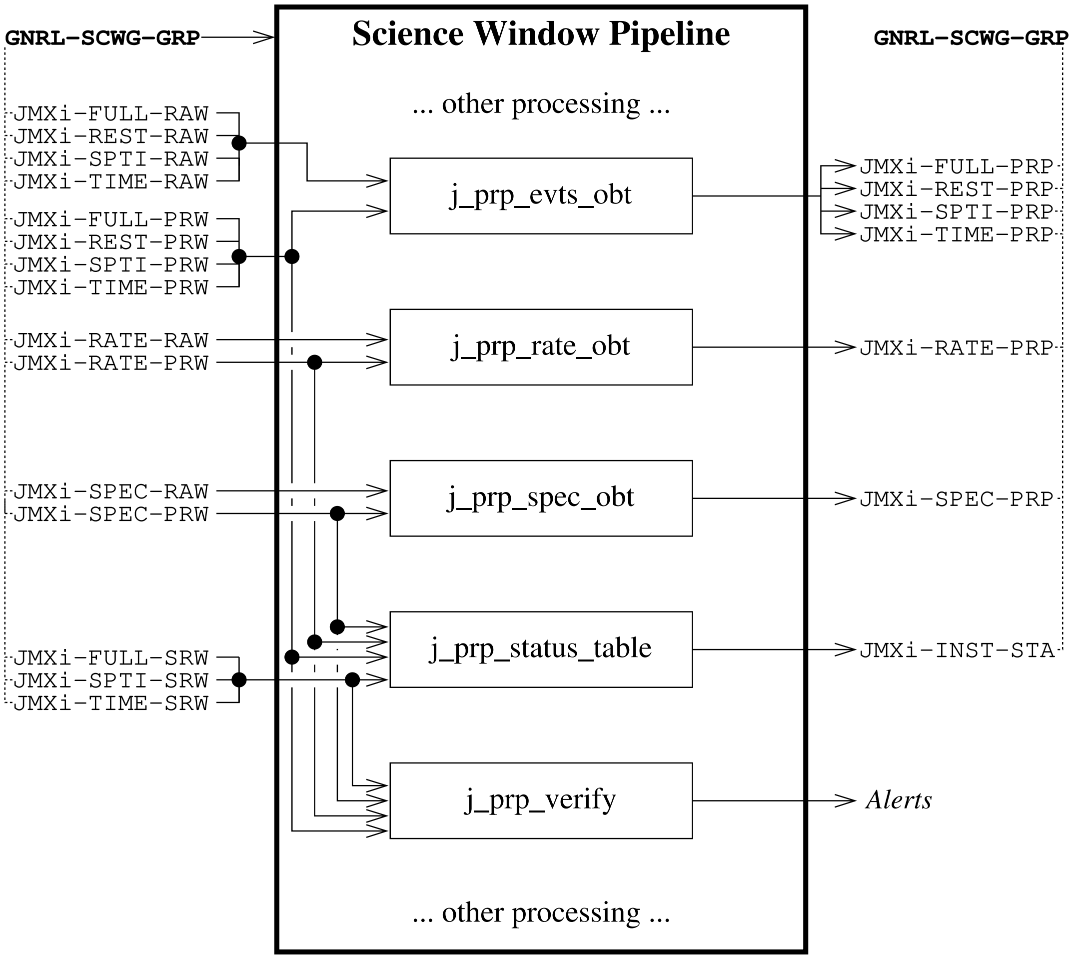

Figure 5.4: j-data-preparation

5.5 j-data-preparation

Type: Part of the overall Science Window Pipeline **Responsible:** ISDC

Purpose: 18.1.1

Function: Create data for subsequent analysis steps that can not be sorted out by ISDC's Pre-Processing.

Inputs:
JMXi-FULL-RAW (p. 218)
JMXi-FULL-PRW (p. 217)

JMXi-REST-RAW (p. 220)
JMXi-REST-PRW (p. 220)
JMXi-SPTI-RAW (p. 223)
JMXi-SPTI-PRW (p. 223)
JMXi-TIME-RAW (p. 224)
JMXi-TIME-PRW (p. 224)
JMXi-RATE-RAW (p. 220)
JMXi-RATE-PRW (p. 220)
JMXi-SPEC-RAW (p. 222)
JMXi-SPEC-PRW (p. 222)
JMXi-FRSS-CRW (p. 217)
JMXi-FULL-SRW (p. 218)
JMXi-SPTI-SRW (p. 223)
JMXi-TIME-SRW (p. 224)

Outputs:
JMXi-FULL-PRP (p. 217)
JMXi-REST-PRP (p. 220)
JMXi-SPTI-PRP (p. 223)
JMXi-TIME-PRP (p. 224)
JMXi-RATE-PRP (p. 220)
JMXi-SPEC-PRP (p. 222)
JMXi-GAIN-CAL (p. 218)
JMXi-INST-STA (p. 219)
JMXi-DEAD-SCP (p. 215)

Parameters:
N/A

Resources: No unusual

Processing:
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-prp-frss-obt to calculate full on-board time for FRSS calibration spectra
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

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. **??**) Revolution File Pipeline(p. **??**)

Subordinate components:
j-prp-evts-obt
j-prp-rate-obt
j-prp-spec-obt
j-prp-frss-obt
j-prp-status-table
j-verify

Dependencies from other components: Pre-Processing must have created a set of raw data

ISDC context: Data Preparation

Delivery date: N/A

5.5.1 j-prp-evts-obt

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

Purpose: Not foreseen in SRD

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

Inputs:
JMXi-FULL-RAW (p. 218)
JMXi-FULL-PRW (p. 217)
JMXi-SPTI-RAW (p. 223)
JMXi-SPTI-PRW (p. 223)
JMXi-TIME-RAW (p. 224)
JMXi-TIME-PRW (p. 224)
JMXi-REST-RAW (p. 220)
JMXi-REST-PRW (p. 220)

Outputs:
JMXi-FULL-PRP (p. 217)
JMXi-SPTI-PRP (p. 223)
JMXi-TIME-PRP (p. 224)
JMXi-REST-PRP (p. 220)

Parameters:

inSWG p. 40	DOL of Science Window Group if not modified
outSWG p. 41	DOL of Science Window Group if modified
jemxNum p. 40	JEM-X detector number
clobber p. 38	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: j-data-preparation(p. 64)

Subordinate components:
None

Dependencies from other components: OBT manipulation functions in DAL3GEN

ISDC context: Data Preparation

Delivery date: V1.9 12/07/2002

5.5.2 j-prp-rate-obt

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

Purpose: Not foreseen in SRD

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

Inputs:
JMXi-RATE-RAW (p. 220)
JMXi-RATE-PRW (p. 220)

Outputs:
JMXi-RATE-PRP (p. 220)

Parameters:

inSWG p. 40	DOL of Science Window Group if not modified
outSWG p. 41	DOL of Science Window Group if modified
jemxNum p. 40	JEM-X detector number
clobber p. 38	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 $(0 \ldots n)T1 = T0 + i * 125$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: j-data-preparation(p. 64)

Subordinate components:
None

Dependencies from other components: None

ISDC context: Data Preparation

Delivery date: V1.6 07/02/2002

5.5.3 j-prp-spec-obt

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

Purpose: Not foreseen in SRD

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

Inputs:
JMXi-SPEC-RAW (p. 222)
JMXi-SPEC-PRW (p. 222)

Outputs:
JMXi-SPEC-PRP (p. 222)

Parameters:

inSWG p. 40	DOL of Science Window Group if not modified
outSWG p. 41	DOL of Science Window Group if modified
jemxNum p. 40	JEM-X detector number
clobber p. 38	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 $(0 \ldots n) T1 = T0 + i * 125\text{ms}$
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: j-data-preparation(p. 64)

Subordinate components:
None

Dependencies from other components: None

ISDC context: Data Preparation

Delivery date: V1.6 07/02/2002

5.5.4 j-prp-frss-obt

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

Purpose: Not foreseen in SRD

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

Inputs:
JMXi-FRSS-CRW (p. 217)

Outputs:
JMXi-GAIN-CAL (p. 218)

Parameters:

inDOL p. 40	DOL of input data (raw FRSS spectra)
outDOL p. 41	DOL of output data (empty gain calibration results table)
jemxNum p. 40	JEM-X detector number
clobber p. 38	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: JEMX-Data-Preparation(p. **??**)

Subordinate components:
None

Dependencies from other components: None

ISDC context: Data Preparation

Delivery date: V1.6 11/04/2002

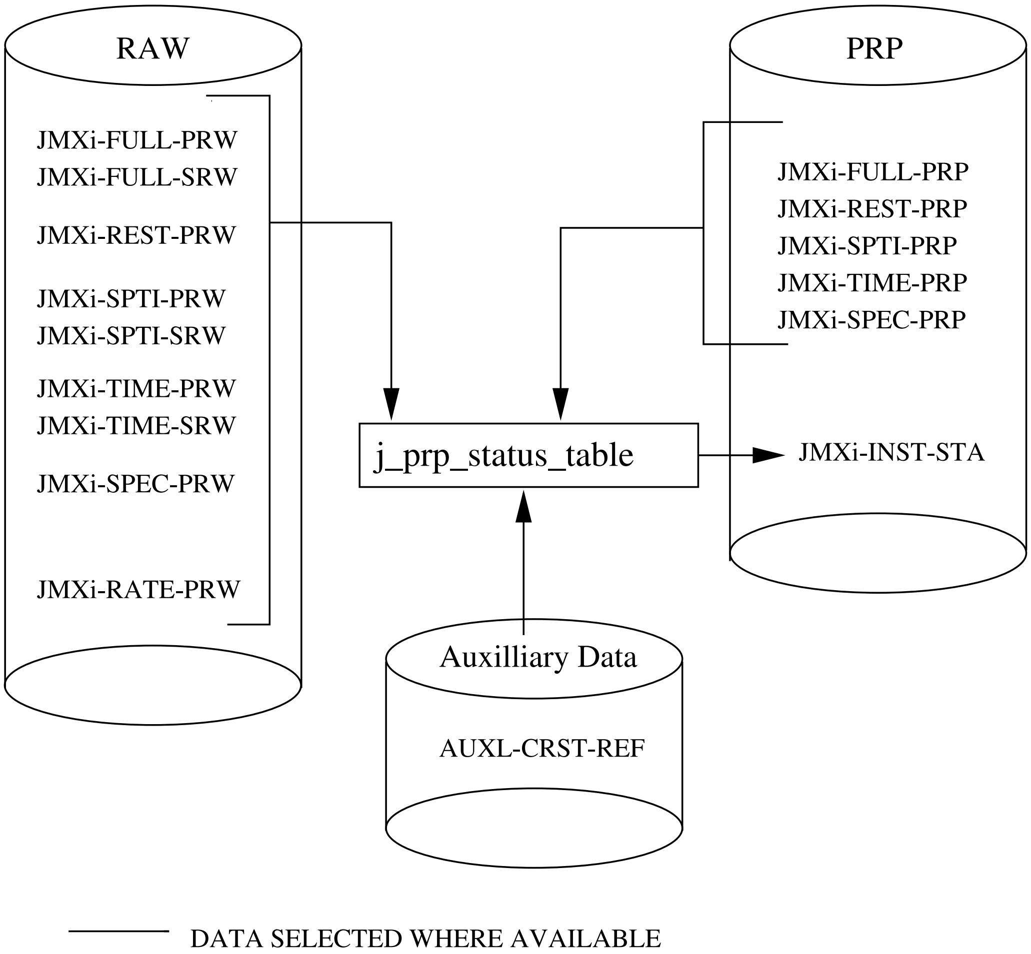

Figure 5.5: j-prp-status-table

5.5.5 j-prp-status-table

Type: Executable **Responsible:** DSRI

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:
JMXi-FULL-PRW (p. 217)
JMXi-FULL-SRW (p. 218)

JMXi-FULL-PRP (p. 217)
JMXi-REST-SRW (p. 220)
JMXi-REST-PRP (p. 220)
JMXi-SPTI-PRW (p. 223)
JMXi-SPTI-SRW (p. 223)
JMXi-SPTI-PRP (p. 223)
JMXi-SPEC-PRW (p. 222)
JMXi-SPEC-PRP (p. 222)
JMXi-TIME-PRW (p. 224)
JMXi-TIME-PRP (p. 224)
JMXi-TIME-SRW (p. 224)
JMXi-RATE-RAW (p. 220)
JMXi-RATE-PRW (p. 220)
JMXi-RATE-PRP (p. 220)
AUXL-CRST-REF (p. 225)

Outputs:
JMXi-INST-STA (p. 219)

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. 40	DOL	Input Software Window Group
outSWG p. 41	DOL	Output Software Window Group
instStat p. 40	DOL	Override output instrument status table
fullSrw p. 39	DOL	Override input FULL event secondary raw data
fullPrw p. 39	DOL	Override input FULL event packet raw data
fullPrp p. 39	DOL	Override input FULL event OBT data
restSrw p. 42	DOL	Override input REST event secondary raw data
restPrp p. 42	DOL	Override input REST event OBT data
sptiSrw p. 43	DOL	Override input SPTI event secondary raw data
sptiPrw p. 43	DOL	Override input SPTI event packet raw data
sptiPrp p. 43	DOL	Override input SPTI event OBT data
specRaw p. 43	DOL	Override input SPEC raw data
specPrp p. 42	DOL	Override input SPEC OBT data

timeSrw p. 43	DOL	Override input TIME event secondary raw data
timePrw p. 43	DOL	Override input TIME event packet raw data
timePrp p. 43	DOL	Override input TIME event OBT data
rateSrw p. 42	DOL	Override input RATE secondary raw data
ratePrw p. 42	DOL	Override input RATE packet raw data
ratePrp p. 42	DOL	Override input RATE OBT data
jemxNum p. 40	int	Number of jemx unit that generated data (1 or 2)
clobber p. 38	boo	Clobber existing output data structures?
chatter p. 38	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 ssc s
    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.

Parent components: j-data-preparation(p. 64)

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 16 and 17

Delivery date: Version 3.1 delivered 2002/07/25

5.5.6 j-prp-verify

Type: Executable **Responsible:** DSRI

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

Inputs:
GNRL-SCWG-GRP (p. 226)
JMXi-FULL-RAW (p. 218)
JMXi-FULL-SRW (p. 218)
JMXi-FULL-PRW (p. 217)
JMXi-FULL-PRP (p. 217)
JMXi-REST-RAW (p. 220)
JMXi-REST-PRW (p. 220)
JMXi-REST-SRW (p. 220)
JMXi-REST-PRP (p. 220)
JMXi-SPTI-RAW (p. 223)
JMXi-SPTI-SRW (p. 223)
JMXi-SPTI-PRW (p. 223)
JMXi-SPTI-PRP (p. 223)
JMXi-SPEC-RAW (p. 222)
JMXi-SPEC-PRW (p. 222)
JMXi-SPEC-PRP (p. 222)
JMXi-TIME-RAW (p. 224)
JMXi-TIME-PRW (p. 224)
JMXi-TIME-PRP (p. 224)
JMXi-RATE-RAW (p. 220)
JMXi-RATE-PRW (p. 220)
JMXi-RATE-PRP (p. 220)
JMXi-IMOD-GRP (p. 218)
JMXi-DETE-MOD (p. 215) In the instrument model group

Outputs: Alerts and logfile output only

Parameters:

inSWG p. 40	DOL of input science window group
instMod p. 40	DOL of the current instrument model group
jemxNum p. 40	Number of jem-x instrument (1 or 2)
alrtLevel p. 37	Maximum level of alerts produced (0,1,2,3)
chatter p. 38	Verbosity of logfile output (0,1,2,3,4,5)

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
    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

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 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.
- Current functionality is very limited and future versions of this executable can expect to do many more checks, though the interface will 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.

Parent components: j-data-preparation(p. 64)

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

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 2.0 delivered 2002/07/01

5.5.7 j-calib-gain-fitting

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

Purpose: 9.17 Ql, 18.4.1

Function: To calculate gain by fitting a spectrum of calibration events to a model of unresolved Gaussian doublets. The calibration spectra will contain two peaks of known energy from which energy offset and scaling can be determined for the detector ADC bins. Other outputs are: calibration spectra re-binned to PI channels; fitting residuals also in PI channels; energy resolution for Ni and Cd doublets and calibration source intensity.

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. 218)
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. **??**) 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 name: JMXi-GAIN-CAL-IDX.

Inputs:
JMXi-IMOD-GRP (p. 218)
JMXi-CALB-MOD (p. 215) In the instrument model group
JMXi-FBDS-MOD (p. 217) In the instrument model group
JMXi-FRSS-CRW-IDX (p. 217)
JMXi-FRSS-CRW (p. 217) In the input index table

Outputs:
JMXi-GAIN-CAL-IDX (p. 218)
JMXi-GAIN-CAL (p. 218) In the output index table

Parameters:

inCAL p. 40	DOL	Input calibration spectrum index table DOL JMXi-FRSS-CRW-IDX
outCAL p. 41	DOL	Output extension index table DOL JMXi-GAIN-CAL-IDX
instMod p. 40	DOL	Instrument model group JMXi-IMOD-GRP
jemxNum p. 40	int.	Number of JEMX instrument being used (1 or 2)

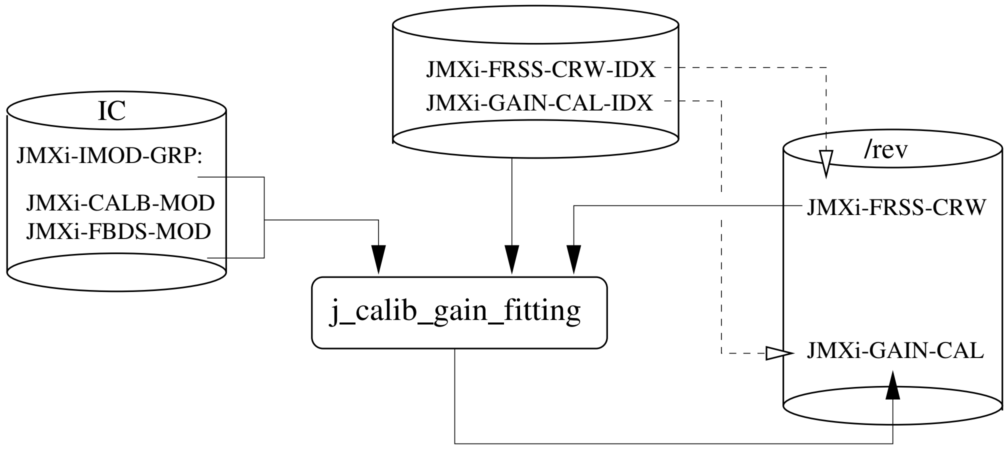

Figure 5.6: j-calib-gain-fitting

clobber p. 38	bool.	Clobber existing data in output data structures?
chatter p. 38	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
  - else go on to next spectral set in the index
- Find if FRSS is Cd/Ni or Fe/Cd
- Make initial guess at fitting parameter values:
  - For Cd/Ni: width of each Ni line
    - width of each Cd line
    - intensity of strongest Ni 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 unresolved 'noise' bump
    - Width of Fe single line
    - Intensity of 'noise' bump
    - 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 at low energy: RES_8_1, RES_8_2 etc.
    - Energy resolution at higher energy: RES_22_1, RES_22_1 etc.
    - Intensity of Ni/Noise line: INT_NI_1, INT_NI_2 etc.
    - Intensity of Cd/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 4.2 minutes) and processing is fast.
- Groups of about 5 calibration spectral sets are input via the input index group so that Revolution Pipeline doesn't need to run this program every 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. 73) so strict one-to-one correspondence of input and output children must be maintained.
- Gaussian lineshapes have been used to model the unresolved Ni and Cd doublets, which should be sufficient given the number of data points in each doublet and detector 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. 218) 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 that be used explicitly by j-cor-gain as the gain history table
- The most important values saved to the gain history table are gain offset (theoretical energy of event appearing in ADC channel 0) and gain slope (linear variation of event energy on ADC channel number). For the four anode segments on each unit these are called G_ZERO_1, G_ZERO_2 etc. and G_CAL_1, G_CAL_2 etc. For a given time, on anode N, the calibrated energy of an event, in KeV, appearing in ADC channel n is therefore G_ZERO_N + n.G_CAL_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/Ni sources and the Fe sources. This does not in the least change the meanings of the main fitting parameters: gain offset and gain slope

Parent components: j-data-preparation(p. 64)

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. 73) 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 5-member version of JMXi-GAIN-CAL-IDX to the master gain history table JMXi-GAIN-CAL-IDX

ISDC context: Automatic Calibration Analysis in the Revolution File Pipeline

Delivery date: Version 6.1 delivered 2002/06/26

5.5.8 j-calib-adc

Type: Executable **Responsible:** DSRI

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

Inputs:
JMXi-ECAL-CRW (p. 216) Stored by revolution
JMXi-ECAL-CPR (p. 216) Stored by revolution
JMXi-IMOD-GRP (p. 218) In Instrument Characteristics
JMXi-ECAL-MOD (p. 216) in the Instrument Model Group

Outputs: Alerts and logfile only

Parameters:

ecalRaw p. 38	DOL of raw electronic calibration data: JMXi-ECAL-CRW
ecalPrp p. 38	DOL of prepared OBTs fro calibration events: JMXi-ECAL-CPR
instMod p. 40	DOL of the current instrument model group
alrtOffset p. 37	Alert level for amps with changed offsets
alrtSlope p. 37	Alert level for amps with changed slopes
alrtBoth p. 37	Alert level for both changed slopes and offsets
alrtOther p. 37	Alert level for other problems with amps
alrtZero p. 37	Alert level for amps with zero slope: dead amps
jemxNum p. 40	Number of the instrument being analysed: 1 or 2
chatter p. 38	Verbosity of log file output

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:
This executable produces only two sorts of output: 1) Logfile entries, including a complete printout of found and model offsets and slopes if verbosity level is 3 or above. 2) Alerts when deviations from model/previous behaviour are detected.
Alert production can be switched off for the individual types of alert by setting the relevant input parameter alrt*** to zero.
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.
The 'null-pulse' data recently 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 two conditions could indicate either temperature sensitivity of the amplifier or damage producing random noise in the offset.
A large RMS variation in the non-null data of an amplifier is probably and 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.

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

ISDC context: Data Preparation in Revolution File Pipeline

Delivery date: Version 1.7 delivered August 25th 2002

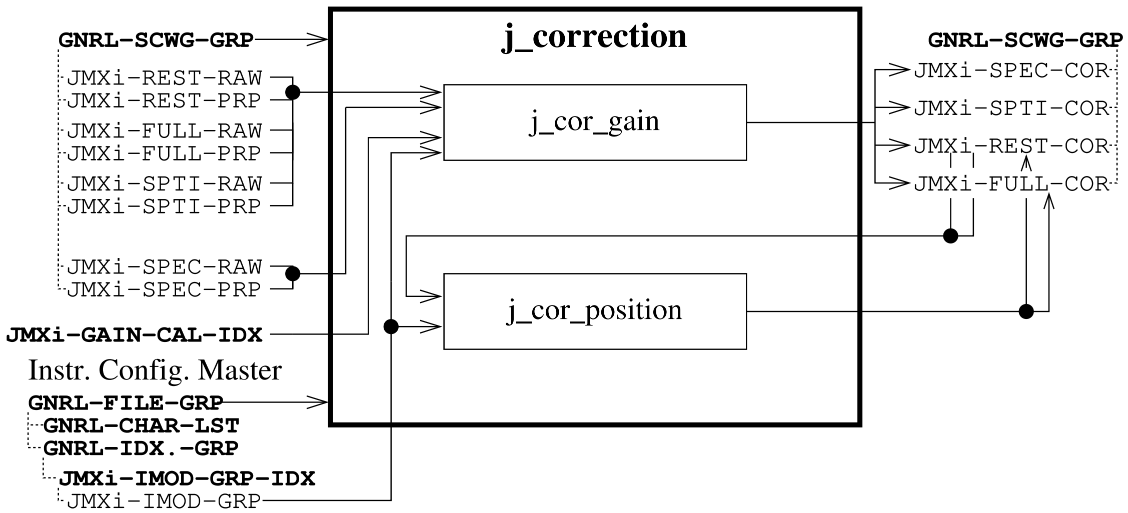

Figure 5.7: j-correction

5.6 j-correction

Type: Analysis Script **Responsible:** ISDC (P. Kretschmar)

Purpose: Fulfill item 18.10 in SRD

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

Inputs:
JMXi-FULL-RAW (p. 218) (event data)
JMXi-FULL-PRP (p. 217)
JMXi-SPTI-RAW (p. 223)
JMXi-SPTI-PRP (p. 223)
JMXi-REST-RAW (p. 220)
JMXi-REST-PRP (p. 220)
JMXi-SPEC-RAW (p. 222) (spectrum mode data)
JMXi-SPEC-PRP (p. 222)
JMXi-IMOD-GRP (p. 218) (instrument description)
JMXi-GAIN-CAL-IDX (p. 218)

Outputs:
JMXi-FULL-COR (p. 217)

JMXi-SPTI-COR (p. 222)
JMXi-REST-COR (p. 220)
JMXi-SPEC-COR (p. 222)

Parameters:

swgDOL p. 43	DOL of Science Window Group to be analyzed
IC-Group p. 40	DOL of Instrument Characteristics master group
IC-Alias p. 40	Selection alias for Instrument Characteristics
jemxNum p. 40	JEM-X detector number
baseDir p. 37	Base Repository Directory
osimData p. 41	Are we using data produced by OSim?
chatter p. 38	Verbosity level; 2 = normal
clobber p. 38	Clobber existing output data?
gainHist p. 39	DOL of Index Group with gain change history
gainModel p. 39	Gain history smoothing model option
j-cor-both-fullRaw p. 39	DOL of FULL imaging raw data (override)
j-cor-both-fullPrp p. 39	DOL of FULL imaging prepared data (override)
j-cor-both-fullCor p. 39	DOL of FULL imaging corrected data (override)
j-cor-both-restRaw p. 42	DOL of REST imaging raw data (override)
j-cor-both-restPrp p. 42	DOL of REST imaging prepared data (override)
j-cor-both-restCor p. 42	DOL of REST imaging corrected data (override)
j-cor-gain-sptiRaw p. 43	DOL of Spectral-Timing raw data (override)
j-cor-gain-sptiPrp p. 43	DOL of Spectral-Timing prepared data (override)
j-cor-gain-sptiCor p. 43	DOL of Spectral-Timing corrected data (override)
j-cor-gain-specRaw p. 43	DOL of Spectrum Mode raw data (override)
j-cor-gain-specPrp p. 42	DOL of Spectrum Mode prepared data (override)
j-cor-gain-specCor p. 42	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:

Parent components: jemx-scw-analysis(p. 59)

Subordinate components:
j-cor-gain
j-cor-position

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: v3.3 30/07/2002

5.6.1 j-cor-gain

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

Purpose: 18.10.2, 18.10.4, 18.11.2

Function: Correct raw events and spectra for variations in detector gain as determined by in-flight calibration spectra, 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 model of the expected spatial variation in gain is also in the instrument model group
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.
7) All output is stored in corrected event data structures JMXi-$* * **$-COR

Inputs:
JMXi-FULL-RAW (p. 218) In input SWG or override DOL
JMXi-REST-RAW (p. 220) In input SWG or override DOL
JMXi-SPTI-RAW (p. 223) In input SWG or override DOL
JMXi-SPEC-RAW (p. 222) In input SWG or override DOL
JMXi-FULL-PRP (p. 217) In input SWG or override DOL
JMXi-REST-PRP (p. 220) In input SWG or override DOL
JMXi-SPTI-PRP (p. 223) In input SWG or override DOL
JMXi-SPEC-PRP (p. 222) In input SWG or override DOL
AUXL-TCOR-HIS (p. 226) In input SWG
JMXi-IMOD-GRP (p. 218)
JMXi-FULB-MOD (p. 217) In instrument model group
JMXi-RESB-MOD (p. 220) In instrument model group
JMXi-SPCB-MOD (p. 222) In instrument model group
JMXi-FBDS-MOD (p. 217) In instrument model group
JMXi-RBDS-MOD (p. 220) In instrument model group
JMXi-SBDS-MOD (p. 221) In instrument model group
JMXi-SPAG-MOD (p. 222) In instrument model group

JMXi-DETE-MOD (p. 215) In instrument model group
JMXi-GAIN-CAL-IDX (p. 218) Output from Revolution file pipeline

Outputs:
JMXi-FULL-COR (p. 217) In output SWG or override DOL
JMXi-REST-COR (p. 220) In output SWG or override DOL
JMXi-SPTI-COR (p. 222) In output SWG or override DOL
JMXi-SPEC-COR (p. 222) In output SWG or override DOL

Parameters:
INPUT GROUPS:
inSWG p. 40 : DOL of input SWG
outSWG p. 41 : DOL of output SWG
instMod p. 40 : DOL of instrument model group JMXi-IMOD-GRP
OTHER INPUTS:
gainHist p. 39 : DOL of gain history table JMXi-GAIN-CAL-IDX
jemxNum p. 40 : Number of the JEM-X instrument: 1 or 2
gainModel p. 39 : Option number for the time-smoothing model
clobber p. 38 : Clobber pre-existing data in extensions?
chatter p. 38 : log file verbosity parameter
OVERRIDES for SWG data structures:
fullRaw p. 39 : Override DOL of JMXi-FULL-RAW
fullPrp p. 39 : Override DOL of JMXi-FULL-PRP
fullCor p. 39 : Override DOL of JXMi-FULL-COR
restRaw p. 42 : Override DOL of JMXi-REST-RAW
restPrp p. 42 : Override DOL of JMXi-REST-PRP
restCor p. 42 : Override DOL of JMXi-REST-COR
sptiRaw p. 43 : Override DOL of JMXi-SPTI-RAW
sptiPrp p. 43 : Override DOL of JMXi-SPTI-PRP
sptiCor p. 43 : Override DOL of JMXi-SPTI-COR
specPrp p. 42 : overide DOL of JMXi-SPEC-PRP
specRaw p. 43 : override DOL of JMXi-SPEC-RAW
specCor p. 42 : 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:

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: gain zero variation over detector area
2nd (256×256) array: gain slope variation over detector area
Find both variation fractions averaged over each anode section
If problems arise with reading/averaging, all variations set to 0.0
Read in the gain history:
Columns: REVOL, OBT_ACQ, G_ZERO_1, G_ZERO_2 etc.,
G_CAL_1, G_CAL_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
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
                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
            Find KeV energy of the event
            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
- Bad pixels are flagged by value 1000.0 added to spatial gain correction for that pixel in JMXi-SPAG-MOD
- 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 for non-imaging data, are found by first correcting the values determined for each anode segment for that segment's average spatial gain variation, then averaging the four corrected values together
- 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.
- Spatial gain variation of gain parameters is given as fractional difference between the value for the pixel and the value for the FRSS calibration spectrum region on a given anode
- 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 rubbish and no real events appear in channel 0. This is stored as G_ZERO_N for anode N, in JMXi-GAIN-CAL-IDX, for each FRSSacquisition time ACQ_OBT.
- Gain zero is typically negative because onboard software adds an offset to the read ADC channels so that zero KeV events appear in about ADC channel 400 (out of 4096 channels). This is the ADC, or amplifier, offset and it should not change much during the mission.

The Gain offset, however, will change quite a bit and is expected to depend heavily on temperature.

- Gain slope is the linear dependance of the energy in KeV of the ADC channels. It is stored as G_CAL_N for anode N in JMXi-GAIN-CAL-IDX.
- For a given time, and event falling on anode N and registering in ADC channel n, the energy in KeV of the event is G_ZERO_$N + n$.G_CAL_N.
- The ADC channel of each event is given by the average ADC channel of the PHA telemetry channel given in the -RAW data extension, according to the PHA binning table appropriate to the data-taking format of the event: JMXi-FULB-MOD, JMXi-RESB-MOD, JMXi-SPCB-MOD
- 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 only two gain models to choose from:

Model 0 Linear interpolation in time between neighbouring values of gain parameters on the gain history table. This is the default model given in j-cor-gain.par. If you want another model you must specify it on the command line.

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.

- 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 integration of the units on the spacecraft it has become clear that spatial gain variation across each anode segment is strongly temperature dependant, and this will have to be addressed in future versions of the executable. Also, it has been found that discharging events on the detector area where FRSS events are found produce anomalous gain fitting results that do not reflect the behaviour of the entire anode for which a particular fixed source should provide gain information. This problem will probably have to be handled with offline analysis of the gain behaviour.

Parent components: j-correction(p. 93)

Subordinate components: Functions of the executable:
JCorGainEvents-Params

JCorGainEvents-Work
JCorGainHistParams
JCorGainSmoothing
JCorGainSolve
JCorGainRowSwap
JCorGainColSwap
JCorGetSpag
JCorGainGetBins
JCorGainAllEvents
JCorGainCorrectEvents
JCorGainAllSpec
JCorGainCorrectSpec

Dependencies from other components:

- The Revolution File Pipeline must run first to provide the master gain history table JMXi-GAIN-CAL-IDX, including OBTs
- The Science Window Pipeline must have run first to provide OBTs for all the raw data events and spectra
- 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 (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).

Delivery date: Version 3.2 delivered 06/08/2002

5.6.2 j-cor-position

Type: Executable **Responsible:** DSRI: 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.

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.
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. 199) 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. 218) In input SWG
JMXi-REST-RAW (p. 220) In input SWG
JMXi-IMOD-GRP (p. 218)
JMXi-CORX-MOD (p. 215) In instrument model group
JMXi-CORY-MOD (p. 215) In instrument model group
JMXi-DETE-MOD (p. 215) In instrument model group

Outputs:
JMXi-FULL-COR (p. 217) In output SWG or override data structure
JMXi-REST-COR (p. 220) In output SWG or override data structure

Parameters:

GROUPS:		
inSWG p. 40	DOL	input SWG with the raw data, ISDC level RAW
outSWG p. 41	DOL	output SWG , ISDC level COR
instMod p. 40	DOL	Instrument model group JMXi-IMOD-GRP
OTHER INPUTS:		
jemxNum p. 40	int.	Number of the JEM-X instrument, 1 or 2

clobber p. 38	bool.	Overwrite pre-existing data in extensions?
chatter p. 38	int.	Log file verbosity parameter (0,1,2,3,4,5)
OVERRIDES:		
fullRaw p. 39	DOL	Override for JMXi-FULL-RAW data structure
restRaw p. 42	DOL	Override for JMXi-REST-RAW data structure
fullCor p. 39	DOL	Override for JMXi-FULL-COR data structure
restCor p. 42	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 pixel conversion factors from JMXi-DETE-MOD
    This is an obsolete procedure since CORX and CORY now contain
    absolute corrected positions of pixels in mm from detector centre
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
        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
    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

End of remarks

Parent components: j-correction(p. 93)

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

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.

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 3.0 delivered 2002/05/03

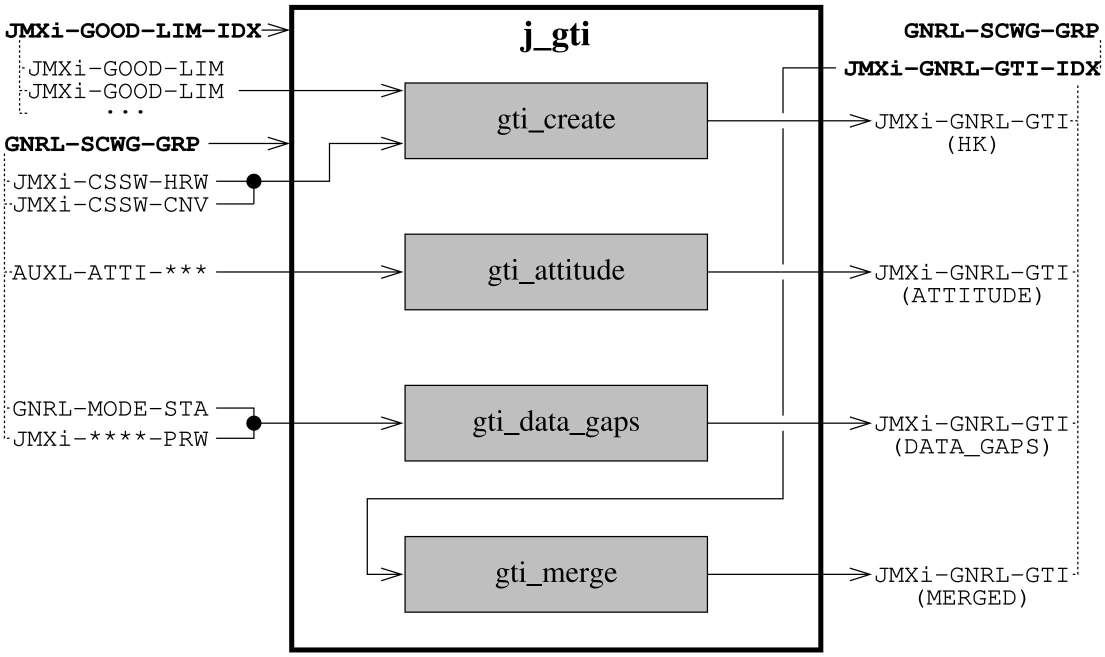

Figure 5.8: j-gti

5.7 j-gti

Type: Analysis Script **Responsible:** ISDC (P. Kretschmar)

Purpose: Not foreseen in SRD

Function: Derive Good-Time Intervals (GTIs) based on housekeeping data, and possibly other parameters and create a merged overall GTI which is used in analysis.

Inputs:
JMXi-CSSW-HRW (p. 215) (housekeeping data)
JMXi-CSSW-CNV (p. 215)
JMXi-GOOD-LIM (p. 218) (limits for parameters)
AUXL-ATTI-PRE (p. 225) (attitude information)
AUXL-ATTI-SNA (p. 225)
AUXL-ATTI-HIS (p. 225)
GNRL-MODE-STA (p. 226) (overall INTEGRAL status)
JMXi-FULL-PRW (p. 217) (packet data)
JMXi-REST-PRW (p. 220)

JMXi-SPTI-PRW (p. 223)
JMXi-TIME-PRW (p. 224)
JMXi-SPEC-PRW (p. 222)
JMXi-RATE-PRW (p. 220)

Outputs:
JMXi-GNRL-GTI (p. 218) (specific GTI for different reason)

Parameters:

swgDOL p. 43	DOL of Science Window Group to be analyzed
IC-Group p. 40	DOL of Instrument Characteristics master group
IC-Alias p. 40	Selection alias for Instrument Characteristics
jemxNum p. 40	JEM-X Instrument number (1 or 2)
baseDir p. 37	Base Repository Directory
osimData p. 41	Are we using data produced by OSim?
limitTable p. 40	DOL of table with parameter limits (”=take from IC)
attTolerance p. 37	Accepted attitude variability [arc sec]
gtiJemxNames p. 39	Names of instrumental GTIs to be included in merged GTI
gtiScNames p. 39	Names of spacecraft GTIs to be included in merged GTI
mergedName p. 40	Name of merged GTI to be used in analysis
chatter p. 38	Verbosity level; 2 = normal
clobber p. 38	Clobber existing output data?

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-merge to merge the different GTIs into a single one used in analysis.

Design remarks:
This script is still only weakly defined, especially concerning the question which conditions will be used to derive GTI.

Parent components: jemx-scw-analysis(p. 59)

Subordinate components:
gti-create
gti-attitude
gti-data-gaps
gti-merge

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: v3.4 30/07/2002

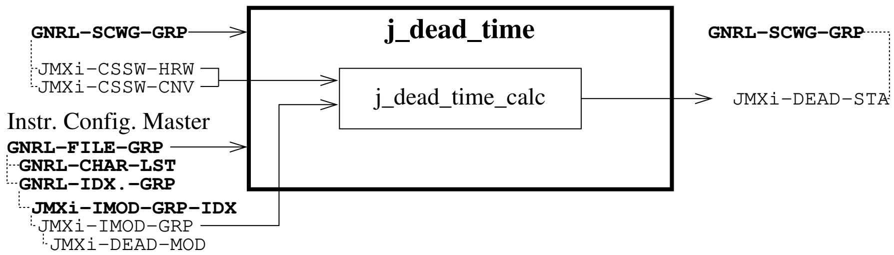

Figure 5.9: Analysis script **j_dead_time** deriving a deadtime history for the given Science Window.

5.8 j-dead-time

Type: Pipeline script **Responsible:** ISDC (P. Kretschmar)

Purpose: Not foreseen in SRD

Function: Calculate dead time history over a science window.

Inputs:
JMXi-CSSW-HRW (p. 215)
JMXi-CSSW-CNV (p. 215)
JMXi-IMOD-GRP (p. 218)
JMXi-DEAD-MOD (p. 215) in group JMXi-IMOD-GRP

Outputs:
JMXi-DEAD-SCP (p. 215)

Parameters:

swgDOL p. 43	DOL of Science Window Group to be analyzed
IC-Group p. 40	DOL of Instrument Characteristics master group
IC-Alias p. 40	Selection alias for Instrument Characteristics
jemxNum p. 40	JEM-X detector number
baseDir p. 37	Base Repository Directory
osimData p. 41	Are we using data produced by OSim?
chatter p. 38	Verbosity level; 2 = normal
clobber p. 38	Clobber existing output data?
j-dead-time-calc-csswHrw p. 38	DOL of raw housekeeping data (override)
j-dead-time-calc-csswCnv p. 38	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:

Parent components: jemx-scw-analysis(p. 59)

Subordinate components:
j-dead-time-calc

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 30/07/2002

5.8.1 j-dead-time-calc

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

Purpose: Not forseen in SRD

Function: To make a history table of instrument dead-time for every 8sec HK interval in a given Science Window. Output table lists packet time (time of first event in the packet) and fraction of time that is dead.

Interfaces: 1) Hardware event triggers 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. 215) In input SWG, ISDC level PRP
JMXi-CSSW-CNV (p. 215) In input SWG: output from Data Preparation
JMXi-DEAD-MOD (p. 215) In instrument model group
JMXi-IMOD-GRP (p. 218)

Outputs:
JMXi-DEAD-SCP (p. 215) In output SWG or override output extension

Parameters:

inSWG p. 40	DOL	Input Science Window Group
outSWG p. 41	DOL	Output Science Window Group
instMod p. 40	DOL	Input instrument model group
csswHrw p. 38	DOL	Override for input extension JMXi-CSSW-HRW
csswCnv p. 38	DOL	Override for input extension JMXi-CSSW-CNV
deadTime p. 38	DOL	Override for output extension JMXi-DEAD-STA
jemxNum p. 40	int.	Number of JEM-X instrument used (1 or 2)
clobber p. 38	bool.	Clobber existing output data structures?
chatter p. 38	int.	Verbosity level of logfile output (0,1,2,3,4,5)

Resources: No special resources needed

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
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 hardward triggers from JMXi-CSSW-HRW
Read in the CPU mode from JMXi-CSSW-HRW
Find how many packets with 8MHz and 16MHz CPU modes
Read the dead time model
Find how many columns there are in the dead time model
Read first column of dead time values for 16MHz mode
If any packets with 8MHz CPU mode:
Read in first column of dead time values for 8MHz mode
Read in column of hardware triggers
If more than 3 columns in deadtime mode
Read each column for 16MHz, and 8MHz if needed
Perform the work for each 8 second period:
Loop over available HK packets
Look up dead time per trigger for given number of triggers
Calculate deadtime as a fraction of time during the packet that is dead
End of loop over packets
Save the packet OBTs and dead times to JMXi-DEAD-STA
Close SWG and tidy up

Design remarks:
1) Currently the dead time model depends only on hardware triggers received during the 8sec period, however the data structure JMXi-DEAD-MOD is arranged so that columns of deadtimes for different ranges of event rates could easily be added, up to 10 pairs of columns for each CPU mode. Keywords of the form RATEMAXn indicate the maximum value for each event rate bin. The program already allows for extra columns to be read, but no functionality has been added for using the extra information. The functionality will only be added if the deadtime can be shown to depend on more than just the hardware trigger rate.

Parent components: j-dead-time(p. 110)

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

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 Near Real Time Science Window Pipeline. See figures on pages 16 and 17

Delivery date: Version 2.2 delivered 2002/07/31

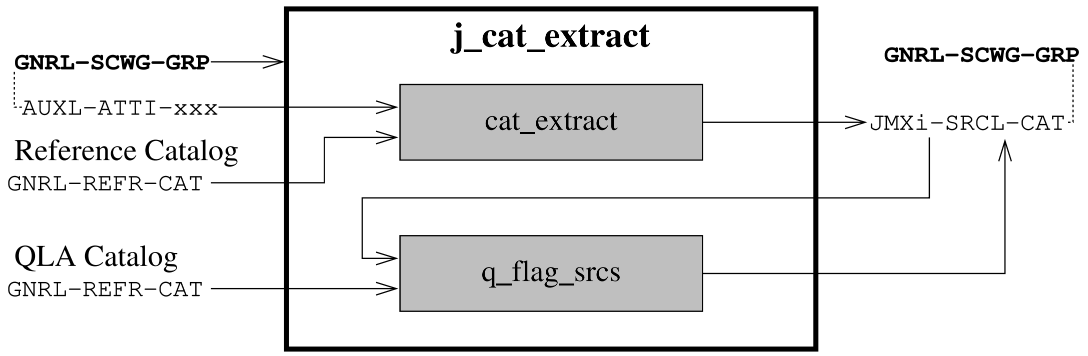

Figure 5.10: j-cat-extract

5.9 j-cat-extract

Type: Pipeline script **Responsible:** ISDC (P. Kretschmar)

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. If run in QLA, flag sources that also appear in a special QLA catalog.

Inputs:
AUXL-ATTI-xxx (p. **??**)
GNRL-REFR-CAT (p. 226) (Reference Catalog)
GNRL-REFR-CAT (p. 226) (QLA Catalog)

Outputs:
JMXi-SRCL-CAT (p. 223)

Parameters:

swgDOL p. 43	DOL of Science Window Group to be analyzed
IC-Group p. 40	DOL of Instrument Characteristics master group
IC-Alias p. 40	Selection alias for Instrument Characteristics
jemxNum p. 40	JEM-X Instrument number (1 or 2)
baseDir p. 37	Base Repository Directory
osimData p. 41	Are we using data produced by OSim?
refCat p. 42	DOL of ISDC Reference Catalog
usrCat p. 43	DOL of User Catalog
qlaCat p. 41	DOL of catalog with sources of interest for QLA
qlaMode p. 41	Boolean: is script called for QLA?
radiusMin p. 42	Lower limits (in deg) of cones for position selection
radiusMax p. 42	Upper limits (in deg) of cones for position selection
fluxDef p. 39	Column used for flux selection
fluxMin p. 39	Low limit for flux selection
fluxMax p. 39	Upper limit for flux selection
class p. 38	Identifier for selection by object class
date p. 38	IJD for public data (¡0 = all public)
chatter p. 38	Verbosity level
clobber p. 38	Clobber existing output data?
q-flag-srcs-errPos p. 39	Acceptable error for source position [deg]
q-flag-srcs-logicType p. 40	Logic for position+name (1=AND, 2=OR)

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
If called for QLA:
    Call q-flag-srcs to flag sources of special interest to QLA

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. 59)

Subordinate components:
cat-extract
q-flag-srcs

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

Completion status: **100%**

ISDC context: Science Analysis

Delivery date: v3.3 30/07/2002

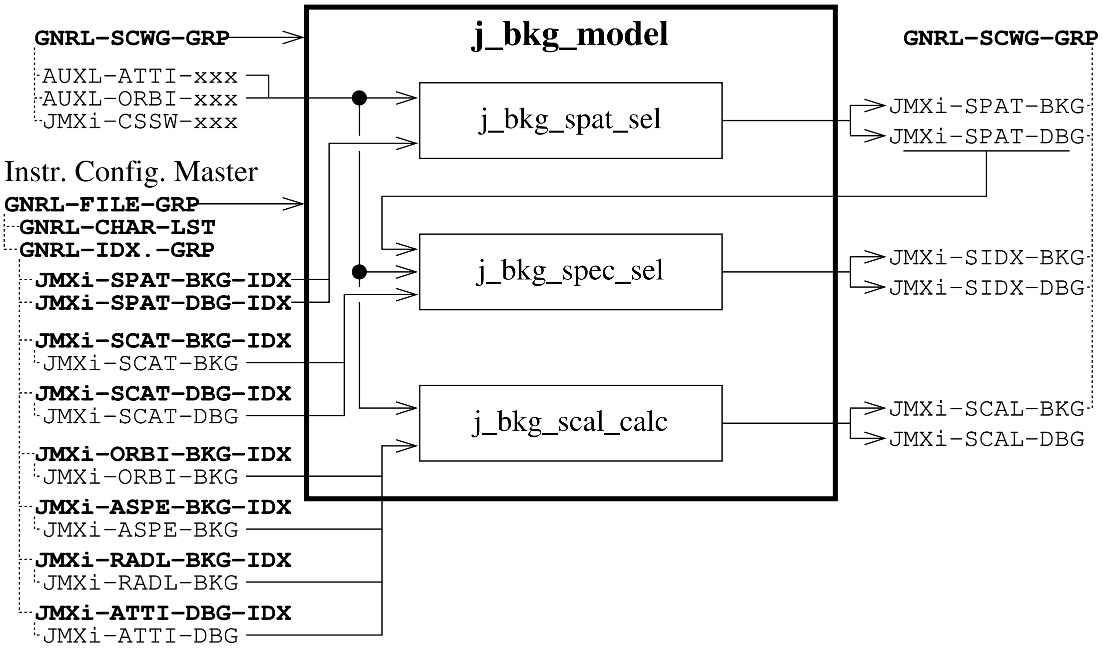

Figure 5.11: j-bkg-model

5.10 j-bkg-model

Type: Analsyis Script **Responsible:** ISDC (P. Kretschmar)

Purpose: Fulfill item 18.17 in SRD

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. 222)
JMXi-SPAT-DBG-IDX (p. 222)
JMXi-SCAT-BKG-IDX (p. 221)
JMXi-SCAT-DBG-IDX (p. 221)
JMXi-ORBI-BKG-IDX (p. 219)
JMXi-ASPE-BKG-IDX (p. 214)
JMXi-RADL-BKG-IDX (p. 219)
JMXi-ATTI-DBG-IDX (p. 214)
AUXL-ATTI-HIS (p. 225) or -SNA, -NRT
AUXL-ORBI-HIS (p. 225) or -SNA, -NRT

JMXi-CSSW-HRW (p. 215) or -CNV, -AVG etc.

Outputs:
JMXi-SPAT-BKG (p. 222)
JMXi-SPAT-DBG (p. 222)
JMXi-SIDX-BKG (p. 221)
JMXi-SIDX-DBG (p. 221)
JMXi-SCAL-BKG (p. 221)
JMXi-SCAL-DBG (p. 221)

Parameters:

swgDOL p. 43	DOL of Science Window Group to be analyzed
IC-Group p. 40	DOL of Instrument Characteristics master group
IC-Alias p. 40	Selection alias for Instrument Characteristics
jemxNum p. 40	JEM-X detector number
baseDir p. 37	Base Repository Directory
osimData p. 41	Are we using data produced by OSim?
simulated p. 42	Use simulated background models?
chatter p. 38	Verbosity level; 2 = normal
clobber p. 38	Clobber existing output data?

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.

Parent components: jemx-scw-analysis(p. 59)

Subordinate components:
j-bkg-spat-sel
j-bkg-spec-sel
j-bkg-scal-calc

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: v3.2 30/07/2002

5.10.1 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. Selection depends on orbit, attitude and radiation level parameters. Simulated and observed shadowgram catalogues are indexed group tables where each model is a separate image. Catalogues can be updated during 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 (-1). 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. 222) (background spatial models, *not* in the SWG)

JMXi-SPAT-DBG-IDX (p. 222)
AUXL-ATTI-HIS (p. 225) (auxliary data)
AUXL-ATTI-NRT (p. 225)
AUXL-ORBI-HIS (p. 225)
AUXL-ORBI-NRT (p. 226)
JMXi-CSSW-HRW (p. 215) (house keeping data)
JMXi-CSSW-CNV (p. 215)
JMXi-ATTI-DBG (p. 214) (diffuse background database)

Outputs:
JMXi-SPAT-BKG (p. 222) (spatial models)
JMXi-SPAT-DBG (p. 222)

Parameters:

inSWG p. 40	Input SWG
outSWG p. 41	Output SWG (override)
jemxNum p. 40	JEM-X instrument number (1,2)
clobber p. 38	Clobber existing output data structure (y/n)?
chatter p. 38	Verbosity of logfile output (0,1,2,3,4,5)
simulated p. 42	simulated (y/n)
bkgSpatIdx p. 37	Instrument background spatial model catalogue DOL (override)
dbgSpatIdx p. 38	Diffuse background spatial model catalogue DOL (override)
bkgSpatial p. 37	Instrument background spatial model DOL (override)
dbgSpatial p. 38	Diffuse background spatial model DOL (override)
dbgGalDbg p. **??**	Diffuse background database DOL

Resources: Executable memory required is about 3 MBytes.
Memory required for arrays and background models at runtime is approximately 2.5 MB. Running time: on a Solaris 8 machine, execution time is 1.5 seconds, using database with 15 models

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
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. 118)

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.

Completion status: **100%**

ISDC context: Scientific Analysis

Delivery date: v1.4.1 05/07/2002

5.10.2 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. 221) (background spectra catalogues, *not* in the SWG)
JMXi-SCAT-DBG (p. 221)
JMXi-SIDX-BKG-IDX (p. **??**) (background spectrum map models)
JMXi-SIDX-DBG-IDX (p. **??**)
AUXL-ATTI-HIS (p. 225) (auxliary data)
AUXL-ATTI-NRT (p. 225)
AUXL-ORBI-HIS (p. 225)
AUXL-ORBI-NRT (p. 226)
JMXi-CSSW-HRW (p. 215) (house keeping data)
JMXi-CSSW-CNV (p. 215)

Outputs:
JMXi-SIDX-BKG (p. 221) (spectral maps)
JMXi-SIDX-DBG (p. 221)

Parameters:

inSWG p. 40	Input SWG
outSWG p. 41	Output SWG (override)
jemxNum p. 40	JEM-X instrument number (1,2)
clobber p. 38	Clobber existing output data structure (y/n)?
chatter p. 38	Verbosity of logfile output (0,1,2,3,4,5)
bkgSpecIdx p. **??**	Instrument background spectrum map models DOL(override)
dbgSpecIdx p. **??**	Diffuse background spectrum map models DOL (override)
bkgSpecMap p. 37	Instrument background spectrum map DOL (override)
dbgSpecMap p. 38	Diffuse background spectrum map DOL (override)

Resources: Executable memory required is about 3 MBytes.
Memory required for arrays and background models at runtime is approximately 1.5 MB. Running time: on a Solaris 8 machine, execution time is 1.5 seconds, using database with 15 models

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
    select closest model by orbital,sun angle and radiation level information
    return JMXi-SIDX-BKG
  Diffuse background
    open JMXi-SIDX-DBG-IDX
    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. 118)

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.

Completion status: **100%**

ISDC context: QLA and SA

Delivery date: v.1.4.4 28/06/2002

5.10.3 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. 225) (auxliary data]
AUXL-ATTI-NRT (p. 225)
AUXL-ORBI-HIS (p. 225)
AUXL-ORBI-NRT (p. 226)
JMXi-CSSW-HRW (p. 215) (house keeping data)
JMXi-CSSW-CNV (p. 215)
JMXi-ASPE-BKG (p. 214) (background scaling factor databases, *not* in the SWG)
JMXi-ORBI-BKG (p. 219)
JMXi-RADL-BKG (p. 219)
JMXi-ATTI-DBG (p. 214)

Outputs:
JMXi-SCAL-BKG (p. 221) (scaling factors)
JMXi-SCAL-DBG (p. 221)

Parameters:

inSWG p. 40	Input SWG
outSWG p. 41	Output SWG (override)
jemxNum p. 40	JEM-X instrument number (1,2)
clobber p. 38	Clobber existing output data structure (y/n)?
chatter p. 38	Verbosity of logfile output (0,1,2,3,4,5)
bkgScale p. 37	Instrument background scaling factor DOL (override)
dbgScale p. 38	Diffuse background scaling factor DOL (override)
attiScale p. 37	Attitude scaling factor DOL (override)
aspeScale p. 37	Sun aspect angle scaling factor DOL (override)
orbiScale p. 41	Orbital phase scaling factor DOL (override)
radlScale p. 42	Radiation level background scaling factor DOL (override)

Resources: Executable memory required is about 3 MBytes.
Memory required for arrays and background models at runtime is approximately 1.5 MB. Running time: on a Solaris 8 machine, execution time is 1.5 seconds, with 664 s of house keeping data (83 rows)

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. 118)

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

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

Pending TBDs: Model selection and database updating during mission.

Completion status: **100%**

ISDC context: QLA and SA

Delivery date: v1.4.4 09/07/2001

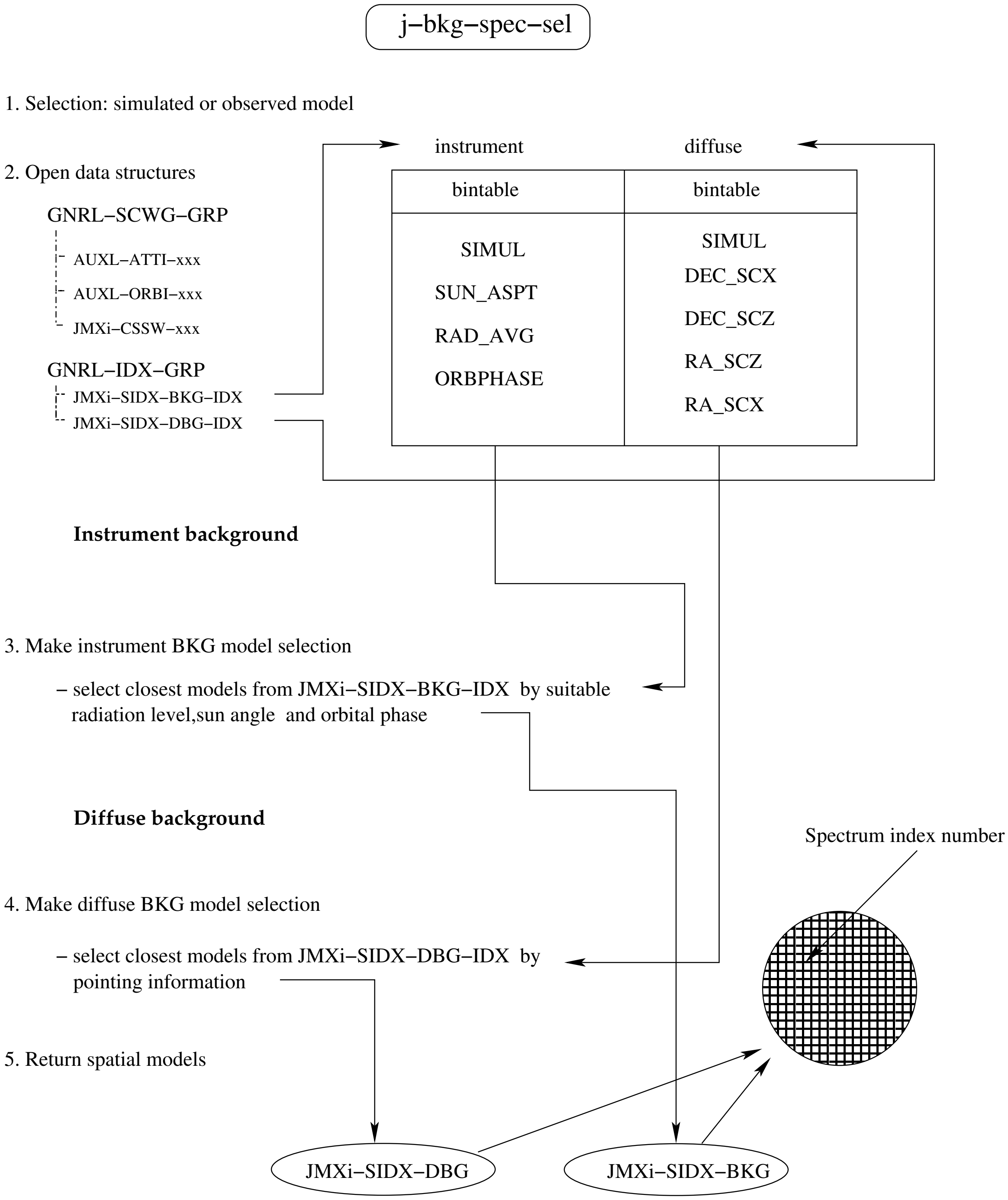

Figure 5.12: j-bkg-spec-sel

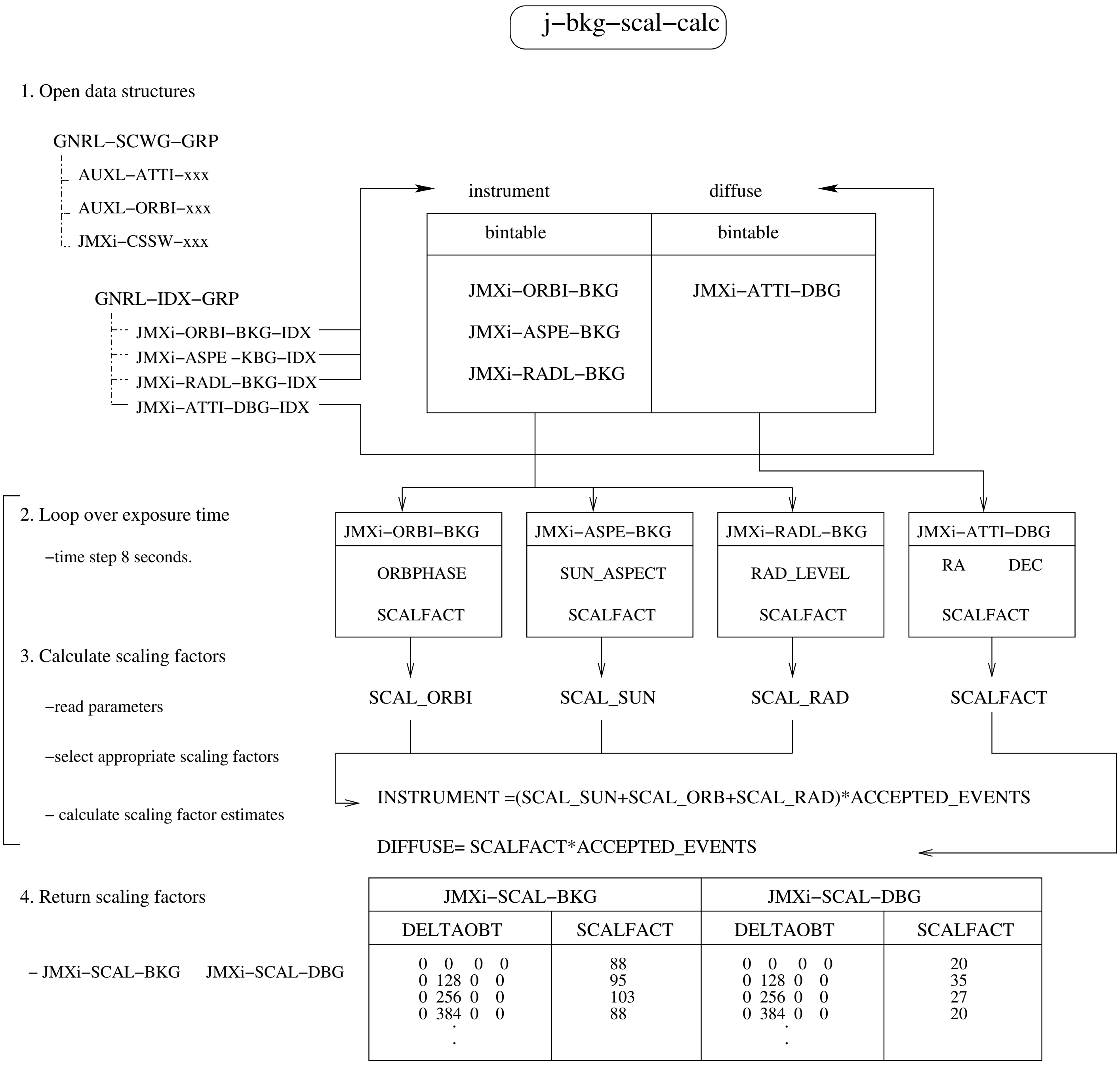

Figure 5.13: j-bkg-scal-calc

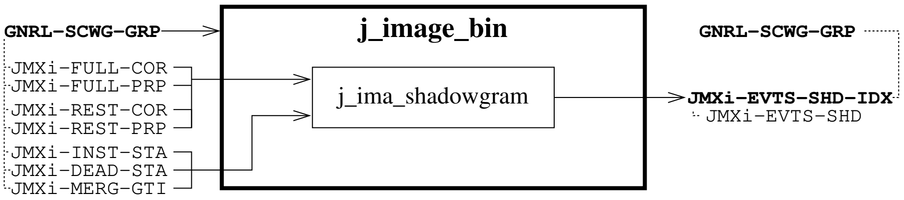

Figure 5.14: Pipeline application script j_image_bin

5.11 j-image-bin

Type: Analysis Script **Responsible:** ISDC (P. Kretschmar)

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. 217) (event data)
JMXi-FULL-COR (p. 217)
JMXi-REST-PRP (p. 220)
JMXi-REST-COR (p. 220)
JMXi-INST-STA (p. 219) (status & grey filter information)
JMXi-DEAD-STA (p. **??**) (deadtime data)
JMXi-MERG-GTI (p. 219) (good-time interval)

Outputs:
JMXi-EVTS-SHD (p. 216) (multiple shadowgrams)

Parameters:

swgDOL p. 43	DOL of Science Window Group to be analyzed
IC-Group p. 40	DOL of Instrument Characteristics master group
IC-Alias p. 40	Selection alias for Instrument Characteristics
jemxNum p. 40	JEM-X detector number
baseDir p. 37	Base Repository Directory
osimData p. 41	Are we using data produced by OSim?
nChanBins p. 40	Number of energy-resolved shadowgrams to create
chanLow p. 38	Lower channel boundaries for shadowgrams
chanHigh p. 38	Upper channel boundaries for shadowgrams
timeStart p. 43	Start of time interval [IJD] (-1 = use swgDOL bounds)
timeStop p. 43	End of time interval [IJD] (-1 = use swgDOL bounds)
evtType p. 39	Event type (0=FULL, 1=REST, -1=both)
backCorr p. 37	Perform background correction?
shdType p. 42	Type of shadowgram 0 : skew, 1 : standard
chatter p. 38	Verbosity level; 2 = normal
clobber p. 38	Clobber existing output data?
j-ima-shadowgram-fullPrp p. 39	DOL of FULL imaging prepared data (override)
j-ima-shadowgram-fullCor p. 39	DOL of FULL imaging corrected data (override)
j-ima-shadowgram-restPrp p. 42	DOL of REST imaging prepared data (override)
j-ima-shadowgram-restCor p. 42	DOL of REST imaging corrected data (override)
j-ima-shadowgram-bkgSpecMap p. 37	DOL of Instrument bkg spectral map (override)
j-ima-shadowgram-dbgSpecMap p. 38	DOL of Diffuse bkg spectral map (override)
j-ima-shadowgram-bkgSpatial p. 37	DOL of Instrument bkg spatial map (override)
j-ima-shadowgram-dbgSpatial p. 38	DOL of Diffuse bkg spatial map (override)
j-ima-shadowgram-bkgScale p. 37	DOL of Instrument bkg scaling factor (override)
j-ima-shadowgram-dbgScale p. 38	DOL of Diffuse bkg scaling factor (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. 59)

Subordinate components:
j-ima-shadowgram

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

ISDC context: Science Analysis

Delivery date: v3.4 30/07/2002

5.11.1 j-ima-shadowgram

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

Purpose: Fulfill item 18.19.2 in SRD

Function: Produce a shadowgram from event file by selecting events according to arrival time and energy in a single Science Window. The shadowgram can be (optional) corrected for the estimated background. 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.

The image output format can either be 'skew' coordinates – a format that is required for the image reconstruction j-ima-basic-recon, or 'standard' detector coordinates i.e. an upright image of the counts in the detector. The former is flagged by keyword 'SHDTYPE = 0' and the latter with 'SHDTYPE = 1'.

Inputs:
JMXi-FULL-COR (p. 217) (event data)
JMXi-REST-COR (p. 220) (event data)
JMXi-FULL-PRP (p. 217) (event data)
JMXi-REST-PRP (p. 220) (event data)
JMXi-FBDS-MOD (p. 217) (binning table)
JMXi-RBDS-MOD (p. 220) (binning table)
JMXi-MERG-GTI (p. 219) (GTI table (OBT))
JMXi-SIDX-DBG (p. 221) (background data)
JMXi-SPAT-DBG (p. 222) (background data)
JMXi-SCAL-DBG (p. 221) (background data)
JMXi-SIDX-BKG (p. 221) (background data)
JMXi-SPAT-BKG (p. 222) (background data)
JMXi-SCAL-BKG (p. 221) (background data)
JMXi-DEAD-SCP (p. 215) (background data)

Outputs:
JMXi-EVTS-SHD (p. 216) (the shadowgram)
JMXi-EVTS-SHD-IDX (p. 216)

Parameters:

inSWG p. 40	DOL of input SWG
outSWG p. 41	DOL of output SWG (override)
instMod p. 40	DOL of JMXi-IMOD-GRP
clobber p. 38	Clobber previous data structures?
chatter p. 38	Level of logging output
jemxNum p. 40	JEMX number (1 or 2)
evtType p. 39	Requested event type
chanMin p. 38	Lowest PI value
chanMax p. 38	Highest PI value
timeStart p. 43	Start of time interval (in IJD)
timeStop p. 43	End of time interval (in IJD)
bkgCor p. 37	Perform background correction
bkgSpecCat p. 37	DOL of Instrument bkg spectral catalog
dbgSpecCat p. 38	DOL of Diffuse bkg spectral catalog
fullPrp p. 39	DOL of FULL imaging prepared data (override)
fullCor p. 39	DOL of FULL imaging corrected data (override)
restPrp p. 42	DOL of REST imaging prepared data (override)
restCor p. 42	DOL of REST imaging corrected data (override)
bkgSpecMap p. 37	DOL of Instrument bkg spectral map (override)
dbgSpecMap p. 38	DOL of Diffuse bkg spectral map (override)
bkgSpatial p. 37	DOL of Instrument bkg spatial map (override)
dbgSpatial p. 38	DOL of Diffuse bkg spatial map (override)
bkgScale p. 37	DOL of Instrument bkg scaling factor (override)
dbgScale p. 38	DOL of Diffuse bkg scaling factor (override)
shdRes p. 42	DOL of output (resulting) shadowgram (override)
deadTime p. 38	DOL of dead time table (input) (override)

Resources: Standard

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. The background correction has not yet been implemented (only prepared for).

Parent components: j-image-bin(p. 131)

Subordinate components:

j-ima-shadowgram-params	(Standard)
j-ima-shadowgram-work	(Standard)
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-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 1.5.1/2002-06-07

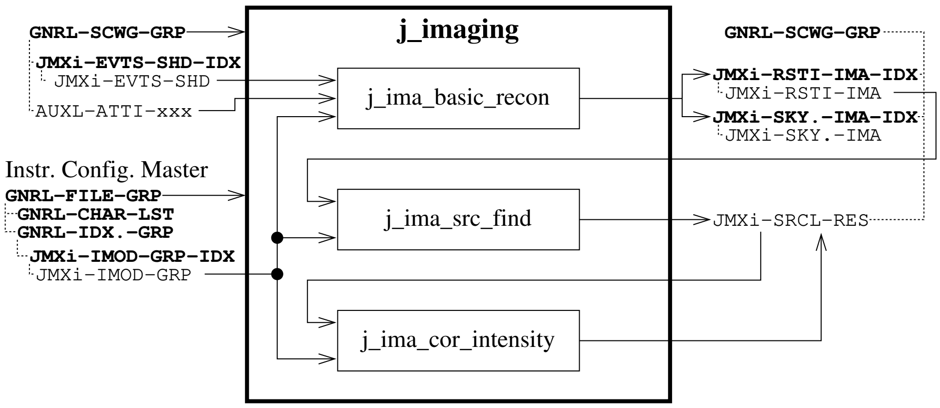

Figure 5.15: Pipeline application script j_imaging

5.12 j-imaging

Type: Analysis Script **Responsible:** ISDC (P. Kretschmar)

Purpose: Fulfill items 18.19.1 and 18.21.1 in SRD

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

Inputs:
JMXi-EVTS-SHD (p. 216) (multiple shadowgrams)
AUXL-ATTI-NRT (p. 225) or -SNA, -NRT (attitude data)
JMXi-IMOD-GRP (p. 218) (instrument description)

Outputs:
JMXi-RSTI-IMA (p. 221) (reconstructed images in skewed coordinates)
JMXi-SKY.-IMA (p. 221) (reconstructed images in sky coordinates)
JMXi-SRCL-RES (p. 224) (list of found sources with fluxes)

Parameters:

swgDOL p. 43	DOL of Science Window Group to be analyzed
IC-Group p. 40	DOL of Instrument Characteristics master group
IC-Alias p. 40	Selection alias for Instrument Characteristics
jemxNum p. 40	JEM-X detector number
baseDir p. 37	Base Repository Directory
osimData p. 41	Are we using data produced by OSim?
chatter p. 38	Verbosity level; 2 = normal
clobber p. 38	Clobber existing output data?
radiusLimit p. 41	[mm] Detector counts will only be used to this radius
ignoreAnode p. 40	Anode numbers (up to four numbers) to ignore
threshold p. 43	Vignetting lower limit
numPixelSubDiv p. 41	Number of pixel subdivisions
minDetLev p. 40	Minimum detections level for source acceptance
maxNumTrials p. 40	Maximum number of trials in the source search
minDist p. 40	Minimum source separation [arcmin]

Resources: See called executables

Processing:

Read in parameters
Call j-ima-basic-recon to create deconvolved images
Call j-ima-src-find to find sources and derive their fluxes
Call j-ima-cor-intensity to correct the fluxes for collimator effects, etc.

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

Parent components: jemx-scw-analysis(p. 59)

Subordinate components:
j-ima-basic-recon
j-ima-src-find
j-ima-cor-intensity

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

ISDC context: Science Analysis

Delivery date: v3.4 30/07/2002

5.12.1 j-ima-basic-recon

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

Purpose: Fulfill item 18.9.4 in SRD

Function: Return sky image both in skew coordinates and in celestial coordinates by correlating shadowgram with mask pattern.

Inputs:
JMXi-EVTS-SHD (p. 216) (shadowgram)
JMXi-MASK-MOD (p. 219) (mask pattern)
JMXi-BORE-MOD (p. 215) (boresight correction)
AUXL-ATTI-LST (p. **??**) (INTEGRAL attitude information)

Outputs:
JMXi-RSTI-IMA (p. 221) (skew coord. system)
JMXi-SKY.-IMA (p. 221) (celestial coord. system)

Parameters:

inSWG p. 40	DOL of input SWG
outSWG p. 41	DOL of output SWG (override)
rstiRes p. 42	DOL of resulting JMXi-RSTI-IMA (override)
skyRes p. 42	DOL of resulting JMXi-SKY.-IMA (override)
instMod p. 40	DOL of JMXi-IMOD-GRP (IC file)
clobber p. 38	Clobber previous data structures?
chatter p. 38	Level of logging output
jemxNum p. 40	JEMX number (1 or 2)
numPixelSubDiv p. 41	Fineness of coordinate conversion between images A value of 20 is recommended
radiusLimit p. 41	[mm] Detector counts will only be used to this radius
ignoreAnode p. 40	Anode numbers (up to four numbers) to ignore

Resources: Standard

Processing:
Get the mask description (JMXi-MASK-MOD) from instMod
Find and open the SWG
Find and open all shadowgram data structures (JMXi-EVTS-SHD) found
with keyword SHDTYPE == 0 (zero)
for all shadowgrams do {
Call function j-ibr-make-rsti for reconstruction by convolution (function j-ibr-convolve)
Write JMXi-RSTI-IMA to disk
Call function j-ibr-make-sky to translate image to an intensity map in WCS coordinates

Write JMXi-SKY.-IMA to disk
}
Close files

Design remarks:
The background subtraction is performed in j-ima-shadowgram, if required.

Parent components: j-imaging(p. 137)

Subordinate components:

j-ima-basic-recon-params	(standard)
j-ima-basic-recon-work	(standard)
j-ibr-make-rsti	Do the image reconstruction
j-ibr-make-sky	Convert image to equatorial coordinates
j-ibr-convolve	Do the mask-shadowgram convolution
j-ibr-crossprod	Vector outer product
j-ibr-get-jemx-arr-short	Get image array
j-ibr-get-pointing	Get pointing of INTEGRAL
j-ibr-index-clean	Erase old reconstructed images
matvmult	Matrix-vector multiplication

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

Pending TBDs: Should the QLA have separate output files to the SA?

ISDC context: Scientific Analysis

Delivery date: Version 1.4.4/2002-06-07

5.12.2 j-ima-src-find

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

Purpose: Fulfill items 18.21.1, 18.18.1 in SRD

Function: Locates sources in the FOV with relative strength above some threshold, based on the intermediate results (RSTI) of basic image reconstruction.

Inputs:
JMXi-RSTI-IMA (p. 221) (deconvolved image)
JMXi-SRCL-CAT (p. 223) (list of catalogue sources)
JMXi-BORE-MOD (p. 215) (boresight correction)
AUXL-ATTI-HIS (p. 225) (INTEGRAL attitude data)
JMXi-INST-STA (p. 219) (boresight correction)

Outputs:
JMXi-SRCL-RES (p. 224) (summary source list)
JMXi-FIND-RES (p. 217) (source list from individual images)

Parameters:

inSWG p. 40	DOL of input SWG
outSWG p. 41	DOL of output SWG (override)
listRes p. 40	DOL of output JMXi-SRCL-RES (override)
findRes p. 39	DOL of output JMXi-FIND-RES (override)
instStat p. 40	DOL of input JMXi-INST-STA (override)
instMod p. 40	DOL of JMXi-IMOD-GRP
chatter p. 38	Level of logging output
jemxNum p. 40	JEMX number (1 or 2)
minDetLev p. 40	Threshold for source acceptance
maxNumTrials p. 40	Maximum number of times of search for a new source
clobber p. 38	Clobber previous data structures
minDist p. 40	Minimum separation between individual sources [in arcmin]

Resources: Standard

Processing:
Find and open SWG
Get the average grey filter level
Find and open sky image file and read the image
Call function j-isf-getsources
do {
    save source parameters
    Call function j-isf-getsources
        Remove all strong peaks in image

Smooth the resulting image with gauss function
Subtract this background from the original image
Define the significance map
Find the highest peak in the significance map and verify
its signicance; remove the peaks to allow finding the next one
} while(there is a new source);
Write result to JMXi-FIND-RES
Compare the sourcepositions to identify coincidences and write the most significant detection to JMXi-SRCL-RES
Close files

Design remarks:
The boresight correction is not yet implemented.
The flux is calculated as

$$\frac{Peaksize \times 1.25 \times (32/greyfilterlevel)}{detectoraperturearea \times masktransparency} \tag{5.1}$$

where the correction factor 1.25 compensates for the loss of counts not attributed to the source.

Parent components: j-imaging(p. 137)

Subordinate components:

j-ima-src-find-params	(standard)
j-ima-src-find-work	(standard)
j-isf-avg	Find average of an array
j-isf-box-extract	Extract a rectangular piece of an array
j-isf-box-substitute	Substitute a rectangular piece of an array
j-isf-comb4pix	Add neighboring pixels of array
j-isf-fbox-extract	Extract a rectangular piece of an array
j-isf-fbox-substitute	Substitute a rectangular piece of an array
j-isf-get-imod-table-float	Get array from JMXi-IMOD-GRP
j-isf-get-jemx-arr-short	Get array from JMXi-IMOD-GRP
j-isf-get-pointing	Find the INTEGRAL pointing
j-isf-getsources	Do the search for sources, one by one
j-isf-gfold	Fold array with gaussian function
j-isf-index-clean	Erase previous data structures
j-isf-local-rms	Find local RMS values
j-isf-rem-extremes	Remove extreme values in an array
j-isf-residual	Subtract two arrays using best match
j-isf-rms	Find RMS of array
j-isf-sat-wcs	Convert source coordinates to WCS (Equatorial)
j-isf-crossprod	Outer vector product

matvmult

Dependencies from other components: Sky images must be created by basic imaging first

ISDC context: Scientific Analysis

Delivery date: Version 1.4.1/2002-06-10

5.12.3 j-ima-cor-intensity

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

Purpose: Fulfill igem 18.22 in SRD

Function: Introduce a vignetting correction for the source intensities found in j-ima-src-find, according to position in FOV, as well as a dead time correction. If the correction is too large i.e. when the vignetting is less that the 'threshold' parameter, no correction is done.

Inputs:
JMXi-SRCL-RES (p. 224) (list of found sources)
JMXi-VIGN-MOD (p. 224) (vignetting data)

Outputs:
JMXi-SRCL-RES (p. 224) (updated source list)

Parameters:

inSWG p. 40	DOL of input SWG
outSWG p. 41	DOL of output SWG (override)
instMod p. 40	DOL of JMXi-INST-GRP
chatter p. 38	Level of logging output
jemxNum p. 40	JEMX number (1 or 2)
clobber p. 38	Clean output index table ?
threshold p. 43	Vignetting lower limit

Resources: Standard

Processing:
Find and open SWG
Read the table JMXi-SRCL-RES
Apply corrections according to JMXi-VIGN-MOD
Write back the table JMXi-SRCL-RES

Parent components: j-imaging(p. 137)

Subordinate components:
j-ima-cor-intensity-params
j-ima-cor-intensity-work
j-ici-get-jemx-arr-float
j-ici-lilipol
j-ici-twod-intpol

Dependencies from other components: Must be called after j-ima-src-find

Pending TBDs: None

ISDC context: Scientific Analysis

Delivery date: Version 1.2.0/2002-06-10

5.12.4 j-ima-refined-recon

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

Purpose: Fulfill item 18.20 in SRD

Function: Improved image reconstruction delivering more accurate results by including more detailed detector information and advanced methods.

Inputs:
JMXi-RSTI-IMA (p. 221) (deconvolved image)
JMXi-SRCL-RES (p. 224) (list of found sources)
AUXL-ATTI-HIS (p. 225) (INTEGRAL attitude data)
JMXi-SCAL-DBG (p. 221) (background data)
JMXi-SPAT-DBG (p. 222) (background data)
JMXi-SIDX-DBG (p. 221) (background data)
JMXi-SCAL-BKG (p. 221) (background data)
JMXi-SPAT-BKG (p. 222) (background data)
JMXi-SIDX-BKG (p. 221) (background data)
JMXi-MASK-MOD (p. 219) (mask description)
JMXi-COLL-MOD (p. 215) (collimator description)
JMXi-DEAD-MOD (p. 215) (dead time data)

Outputs:
JMXi-SKY.-IMA (p. 221) (improved sky image)

Parameters:

inSWG p. 40	DOL of input SWG
outSWG p. 41	DOL of output SWG
instMod p. 40	DOL of JMXi-IMOD-GRP
jemxNum p. 40	JEMX number (1 or 2)
other p. **??**	Other parameters are foreseen

Resources: j-ima-refined-recon resources

Processing:
j-ima-refined-recon processing flow

Design remarks:
This component is very likely to include an iterative process where the JMXi-SRCL-RES can act as a first step or to point out the interesting parts in the image.

Parent components: j-imaging(p. 137)

Subordinate components:
Still unknown

Dependencies from other components: TBW

Pending TBDs: A research project has been inititated on possible methods for this component

Prototyping status: Code developed as part of instrument observation simulator

ISDC context: Scientific Analysis

Delivery date: September 2002

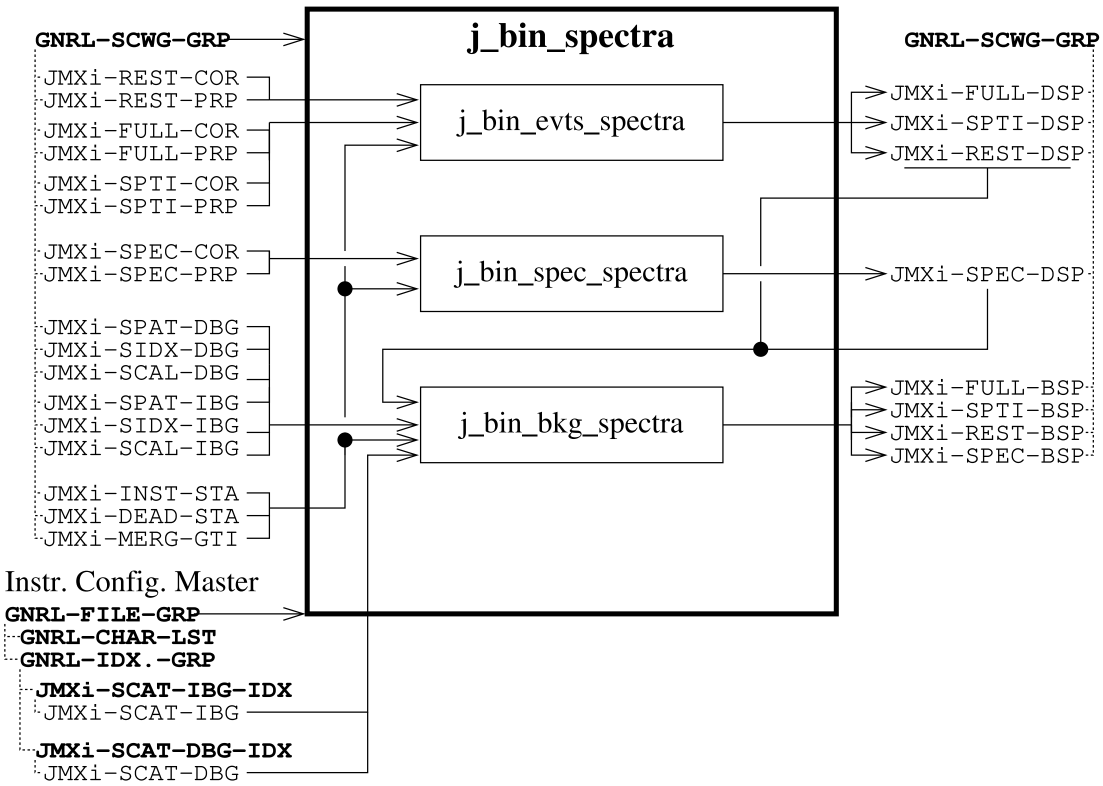

Figure 5.16: j-bin-spectra

5.13 j-bin-spectra

Type: Analysis Script **Responsible:** ISDC (P. Kretschmar)

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. 217) (event data)
JMXi-FULL-COR (p. 217)
JMXi-SPTI-PRP (p. 223)
JMXi-SPTI-COR (p. 222)
JMXi-REST-PRP (p. 220)
JMXi-REST-COR (p. 220)

JMXi-SPEC-PRP (p. 222) (on-board spectra)
JMXi-SPEC-COR (p. 222)
JMXi-INST-STA (p. 219) (instrument status)
JMXi-MERG-GTI (p. 219) (good-time intervals)
JMXi-SIDX-DBG (p. 221) (background model data)
JMXi-SPAT-DBG (p. 222)
JMXi-SCAL-DBG (p. 221)
JMXi-SIDX-BKG (p. 221)
JMXi-SPAT-BKG (p. 222)
JMXi-SCAL-BKG (p. 221)
JMXi-SCAT-DBG (p. 221) (background spectra catalogs, *not* in the swgDOL)
JMXi-SCAT-BKG (p. 221)

Outputs:
JMXi-FULL-DSP (p. 217)
JMXi-FULL-BSP (p. 217)
JMXi-SPTI-DSP (p. 223)
JMXi-SPTI-BSP (p. 222)
JMXi-REST-DSP (p. 220)
JMXi-REST-BSP (p. 220)
JMXi-SPEC-DSP (p. 222)
JMXi-SPEC-BSP (p. 222)

Parameters:

swgDOL p. 43	DOL of Science Window Group to be analyzed
jemxNum p. 40	JEM-X detector number
IC-Group p. 40	DOL of Instrument Characteristics master group
IC-Alias p. 40	Selection alias for Instrument Characteristics
baseDir p. 37	Base Repository Directory
osimData p. 41	Are we using data produced by OSim?
timeStep p. 43	Binning time for spectra (0.0=bin all data together)
timeStart p. 43	Time in IJD of start of first time bin (¡0 = use swgDOL start)
timeStop p. 43	Time in IJD of end of last time bin (¡0 = use swgDOL end)
nPhaseBins p. 41	Number of phase bins (N¡0 = abs(N) equally spaced bins)
phaseBins p. 41	Phase bin sizes, separated by space (e.g: 0.1 0.2 0.3 ...)
rowSelect p. 42	CFITSIO selection string on events (optional)
gtiNames p. 39	Names of GTI tables to be used
chatter p. 38	Verbosity level; 2 = normal
clobber p. 38	Clobber existing output data?

j-bin-evts-spectra-fullRaw p. 39	DOL of FULL imaging raw data (override)
j-bin-evts-spectra-fullPrp p. 39	DOL of FULL imaging prepared data (override)
j-bin-evts-spectra-fullCor p. 39	DOL of FULL imaging corrected data (override)
j-bin-evts-spectra-restRaw p. 42	DOL of REST imaging raw data (override)
j-bin-evts-spectra-restPrp p. 42	DOL of REST imaging prepared data (override)
j-bin-evts-spectra-restCor p. 42	DOL of REST imaging corrected data (override)
j-bin-evts-spectra-sptiRaw p. 43	DOL of Spectral-Timing raw data (override)
j-bin-evts-spectra-sptiPrp p. 43	DOL of Spectral-Timing prepared data (override)
j-bin-evts-spectra-sptiCor p. 43	DOL of Spectral-Timing corrected data (override)
j-bin-spec-spectra-specRaw p. 43	DOL of Spectrum Mode raw data (override)
j-bin-spec-spectra-specPrp p. 42	DOL of Spectrum Mode prepared data (override)
j-bin-spec-spectra-specCor p. 42	DOL of Spectrum Mode corrected data (override)
j-bin-bkg-spectra-dbgSpecMap p. 38	Diffuse background spectral indices DOL (override)
j-bin-bkg-spectra-dbgSpatial p. 38	Diffuse background spatial distribution DOL (override)
j-bin-bkg-spectra-dbgScale p. 38	Diffuse background scaling factors DOL (override)
j-bin-bkg-spectra-bkgSpecMap p. 37	Instrm. background spectral indices DOL (override)
j-bin-bkg-spectra-bkgSpatial p. 37	Instrm. background spatial distribution DOL (override)
j-bin-bkg-spectra-bkgScale p. 37	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. 59)

Subordinate components:
j-bin-evts-spectra
j-bin-spec-spectra
j-bin-bkg-spectra

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: v2.2 30/07/2002

5.13.1 j-bin-evts-spectra

Type: Executable **Responsible:** ISDC (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 swg-DOL 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. 40	Input SWG
outSWG p. 41	Output SWG (optional)
specRes p. 43	Output spectra DOL (optional)
jemxNum p. 40	JEM-X instrument number
evtType p. 39	Event mode number, must be 0=FULL, 1=REST or 2=SPTI
timeStep p. 43	Binning time for spectra (0.0=bin all data together)
timeStart p. 43	Time in IJD of start of first time bin (¡0 = use SWG start)
timeStop p. 43	Time in IJD of end of last time bin (¡0 = use SWG end)
nPhaseBins p. 41	Number of phase bins (N¡0 = abs(N) equally spaced bins)
phaseBins p. 41	Phase bin sizes, separated by space (e.g: 0.1 0.2 0.3 ...)
rowSelect p. 42	CFITSIO selection string on events (optional)
gtiNames p. 39	Names of GTI tables to be used
clobber p. 38	Clobber existing output data structures (y/n)?
chatter p. 38	Verbosity of output
fullPrp p. 39	Full Imaging Prepared Data DOL (override)
fullCor p. 39	Full Imaging Corrected Data DOL (override)
restPrp p. 42	Restricted Imaging Prepared Data DOL (override)
restCor p. 42	Restricted Imaging Corrected Data DOL (override)
sptiPrp p. 43	Spectral-Timing Prepared Data DOL (override)
sptiCor p. 43	Spectral-Timing Corrected Data DOL (override)
instStat p. 40	Instrument Status Data DOL (override)
deadTime p. 38	Deadtime History Table DOL (override)
goodTime p. 39	Good-Time Interval Table DOL (override)

Inputs:
JMXi-FULL-PRP (p. 217)

JMXi-FULL-COR (p. 217)
JMXi-SPTI-PRP (p. 223)
JMXi-SPTI-COR (p. 222)
JMXi-REST-PRP (p. 220)
JMXi-REST-COR (p. 220)
JMXi-INST-STA (p. 219)
JMXi-MERG-GTI (p. 219)

Outputs:
JMXi-FULL-DSP (p. 217)
JMXi-SPTI-DSP (p. 223)
JMXi-REST-DSP (p. 220)

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.

Parent components: j-bin-spectra(p. 149)

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: v2.5.1 25/06/2002

5.13.2 j-bin-spec-spectra

Type: Executable **Responsible:** ISDC (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. 222)
JMXi-SPEC-COR (p. 222)
JMXi-INST-STA (p. 219)
JMXi-MERG-GTI (p. 219)

Outputs:
JMXi-SPEC-DSP (p. 222)

Parameters:

inSWG p. 40	Input Science Window Group DOL
outSWG p. 41	Output Science Window Group DOL (optional)
specRes p. 43	Output spectra DOL (optional)
jemxNum p. 40	JEM-X instrument number
timeStep p. 43	Binning time for spectra (0.0=bin all data together)
timeStart p. 43	Time in IJD of start of first time bin (¡0 = use SWG start)
timeStop p. 43	Time in IJD of end of last time bin (¡0 = use SWG end)
nPhaseBins p. 41	Number of phase bins (N¡0 = abs(N) equally spaced bins)
phaseBins p. 41	Phase bin sizes, separated by space (e.g: 0.1 0.2 0.3 ...)
gtiNames p. 39	Names of GTI tables to be used
clobber p. 38	Clobber existing output data structures (y/n)?
chatter p. 38	Verbosity of output
specPrp p. 42	Spectrum Mode Prepared Data DOL (override)
specCor p. 42	Full Imaging Corrected Data DOL (override)
instStat p. 40	Instrument Status Data DOL (override)
deadTime p. 38	Deadtime History Table DOL (override)
goodTime p. 39	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. 149)

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: 2.3.2 05/07/2002

5.13.3 j-bin-bkg-spectra

Type: Executable **Responsible:** ISDC (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. 217)
JMXi-SPTI-DSP (p. 223)
JMXi-REST-DSP (p. 220)
JMXi-SPEC-DSP (p. 222)
JMXi-INST-STA (p. 219)
JMXi-MERG-GTI (p. 219)
JMXi-SIDX-DBG (p. 221)
JMXi-SPAT-DBG (p. 222)
JMXi-SCAL-DBG (p. 221)
JMXi-SIDX-BKG (p. 221)
JMXi-SPAT-BKG (p. 222)
JMXi-SCAL-BKG (p. 221)
JMXi-SCAT-DBG (p. 221)
JMXi-SCAT-BKG (p. 221)

Outputs:
JMXi-FULL-BSP (p. 217)
JMXi-SPTI-BSP (p. 222)
JMXi-REST-BSP (p. 220)
JMXi-SPEC-BSP (p. 222)

Parameters:

inSWG p. 40	Input Science Window Group DOL
outSWG p. 41	Output Science Window Group DOL (optional)
backRes p. 37	Output background spectra DOL (optional)
jemxNum p. 40	JEM-X instrument number
evtType p. 39	TM format number: must be 0=FULL, 1=REST, 2=SPTI or 4=SPEC
dbgSpecCat p. 38	Diffuse background spectral model table DOL
bkgSpecCat p. 37	Instrument background spectral model table DOL
clobber p. 38	Clobber existing output data structures (y/n)?
chatter p. 38	Verbosity of output
dbgSpecMap p. 38	Diffuse background spectral indices DOL (override)
dbgSpatial p. 38	Diffuse background spatial distribution DOL (override)
dbgScale p. 38	Diffuse background scaling factors DOL (override)
bkgSpecMap p. 37	Instrm. background spectral indices DOL (override)
bkgSpatial p. 37	Instrm. background spatial distribution DOL (override)
bkgScale p. 37	Diffuse background scaling factors DOL (override)
specRes p. 43	Input detector spectra 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. 149)

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: v1.6.1 05/07/2002

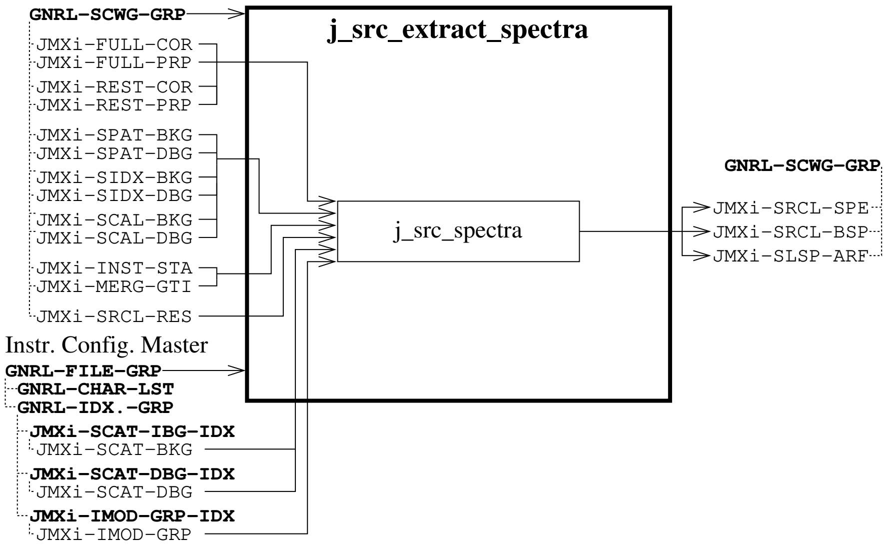

Figure 5.17: Pipeline application script j_src_extract_spect

5.14 j-src-extract-spectra

Type: Analysis Script **Responsible:** ISDC (P. Kretschmar)

Purpose: Fulfill items 18.23.1 and18.25 in SRD

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

Inputs:
JMXi-FULL-PRP (p. 217) (event data)
JMXi-FULL-COR (p. 217)
JMXi-REST-PRP (p. 220)
JMXi-REST-COR (p. 220)
JMXi-INST-STA (p. 219) (instrument status & grey filter)
JMXi-MERG-GTI (p. 219) (good-time intervals)
JMXi-IMOD-GRP (p. 218) (instrument description)
JMXi-SIDX-DBG (p. 221) (background model data)
JMXi-SPAT-DBG (p. 222)
JMXi-SCAL-DBG (p. 221)
JMXi-SIDX-BKG (p. 221)
JMXi-SPAT-BKG (p. 222)
JMXi-SCAL-BKG (p. 221)

JMXi-SCAT-DBG (p. 221) (background spectra catalogs, *not* in the swgDOL)
JMXi-SCAT-BKG (p. 221)

Outputs:
JMXi-SRCL-SPE (p. 224) (spectra for multiple sources)
JMXi-SRCL-BSP (p. 223) (corresponding bkg spectra)
JMXi-SLSP-ARF (p. 221) (Ancillary response functions for sources)

Parameters:

swgDOL p. 43	Science Window Group
jemxNum p. 40	JEM-X detector number (1 or 2)
IC-Group p. 40	Instrument Configuration Master Group
IC-Alias p. 40	Instrument Configuration version alias
baseDir p. 37	Base Repository Directory
osimData p. 41	Are we using data produced by OSim?
chatter p. 38	Verbosity level (2=normal)
clobber p. 38	Clobber existing output data?
timeStep p. 43	Length of integration time bins
timeStart p. 43	Time start in IJD of first spectrum (¡0 = use SWG start)
timeStop p. 43	Time start in IJD of last spectrum (¡0 = use SWG start)
bgrMethod p. 37	Background subtraction method (0=Flat,1=Model)
backCorr p. 37	Correct for background?
vignCorr p. 43	Correct for vignetting?
deadCorr p. 38	Correct for instrumental dead time?
gtiNames p. 39	Names of GTI tables to be used

Resources: See called executable

Processing:

Read in parameters
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:

Parent components: jemx-scw-analysis(p. 59)

Subordinate components:
j-src-spectra

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

ISDC context: Science Analysis

Delivery date: v3.5 30/07/2002

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 (outSrcLC, outSrcArf

Inputs:
JMXi-IMOD-GRP (p. 218)
JMXi-INST-STA (p. 219) In input SWG
JMXi-DEAD-SCP (p. 215) In input SWG
JMXi-SCAL-BKG (p. 221) In input SWG
JMXi-SCAL-DBG (p. 221) In input SWG
JMXi-SPAT-BKG (p. 222) In input SWG
JMXi-SPAT-DBG (p. 222) In input SWG
JMXi-SCAT-BKG (p. 221)
JMXi-SIDX-BKG (p. 221) In input SWG
JMXi-SCAT-DBG (p. 221)
JMXi-SIDX-DBG (p. 221) In input SWG
JMXi-MERG-GTI (p. 219)
JMXi-SRCL-RES (p. 224) In input SWG
JMXi-FULL-COR (p. 217) In input SWG
JMXi-REST-COR (p. 220) In input SWG
JMXi-FULL-PRP (p. 217) In input SWG
JMXi-REST-PRP (p. 220) In input SWG

Outputs:
JMXi-SRCL-ARF (p. 223) In output SWG
JMXi-SRCL-SPE (p. 224) In output SWG
JMXi-SRCL-BSP (p. 223) In output SWG

Parameters:
INPUT GROUPS:
inSWG p. 40 DOL of input SWG
outSWG p. 41 DOL of output SWG
instMod p. 40 DOL of instrument model group JMXi-IMOD-GRP
bkgSpecCat p. 37 DOL of instr bkg spectral catalogue JMXi-SCAT-BKG

dbgSpecCat p. 38 DOL of diffuse bkg spectral catalogue JMXi-SCAT-DBG
goodTime p. 39 DOL of Good-Time Interval Table (override)
deadTime p. 38 DOL of Deadtime History Table (override)
instStat p. 40 DOL of Instrument Status Data (override)
fullPrp p. 39 DOL of full Imaging Prepared Data (override)
fullCor p. 39 DOL of full Imaging Corrected Data (override)
dbgSpecMap p. 38 DOL of diffuse backgr. spectral indices
dbgSpatial p. 38 DOL of diffuse backgr. spatial distribution
dbgScale p. 38 DOL of diffuse backgr. scaling factors
bkgSpecMap p. 37 DOL of instrument backgr. spectral indices
bkgSpatial p. 37 DOL of instrument backgr. spatial distribution
bkgScale p. 37 DOL of instrument backgr. scaling factors
OTHER INPUTS:
jemxNum p. 40 Number of JEM-X unit (1 or 2)
evtType p. 39 0 for FULL events, 1 for REST events.
timeStart p. 43 Time at beginning of (first) spectrum.
timeStop p. 43 End time of data used for extraction.
timeStep p. 43 Time integration step (seconds). (¡0 to use all data)
bgrMethod p. 37 Choice of backgr. method (0=flat, 1=model).
nPhaseBins p. 41 Number of bins in phaseBins.
phaseBins p. 41 Length of phase bins (in case of cyclic binning).
precisionLevel p. 41 speed/precision level (1-100, 0=Default) vignCorr p. 43 Should program perform vignetting corrections?
deadCorr p. 38 Should program perform deadtime corrections?
gtiNames p. 39 Names of GTI tables to be used
chatter p. 38 level of debugging output
clobber p. 38 Clobber existing data in output data structures?
OVERRIDES:
outBkgSpec p. 41 DOL of override JMXi-SRCL-BSP
outSrcSpec p. 41 DOL of override JMXi-SRCL-SPE
outSrcArf p. 41 DOL of override JMXi-SRCL-ARF

Resources: Execution is approximatly linear with number of events. Memmory requirement is dominated by input event data.

Processing:
Read input parameters
Read instrument model
Read instrument status
Read source parameters
Loop over sources

```
    Calculate instrument response for source
    Write Ancillary Response File, JMXi-SRCL-ARF
End loop sources
Read background model
Loop over sources
    Loop over energy bins
        Calculate net background contribution (spatial and energy parts)
    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
    Loop over sources and energy
        Add contribution to source time energy bin
    End loop sources and energy
End loop over events

Loop over time bins and sources
    Loop over energy bins
        Correct countrates for greyfilter and deadtime
        Calculate and correct for background.
    End energy bins
End loop over time bins and sources

Write spectra to JMXi-SRCL-SPE
Write backgr. spectra to JMXi-SRCL-BSP
```

Design remarks:
JMXi-SRCL-SPE will contain background subtracted spectra for individual sources. Background spectra will be stored in JMXi-SRCL-BSP for evaluation purposes.
Either FULL or REST imaging events are used. To extract both, the executable has to be run twice producing two spectra, one based on FULL and one on REST imaging data. a third option is considered (evtType=-1) where both types of events are used, producing a single, merged, spectrum.
The possibility to choose alternative background methods (e.g. fitting) is forseen, but not implemented (will use parameter bgrMethod). The option to select flat or model background will be

introduced in version 3.0.

Subordinate components: Functions of the executable:
j-src-spectra-work
j-src-spectra-params
j-src-spectra-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-min-range
j-gti-overlap
j-read-GTI
j-ssp-GetFileName

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 2.5 delivered 15 Jan 2002

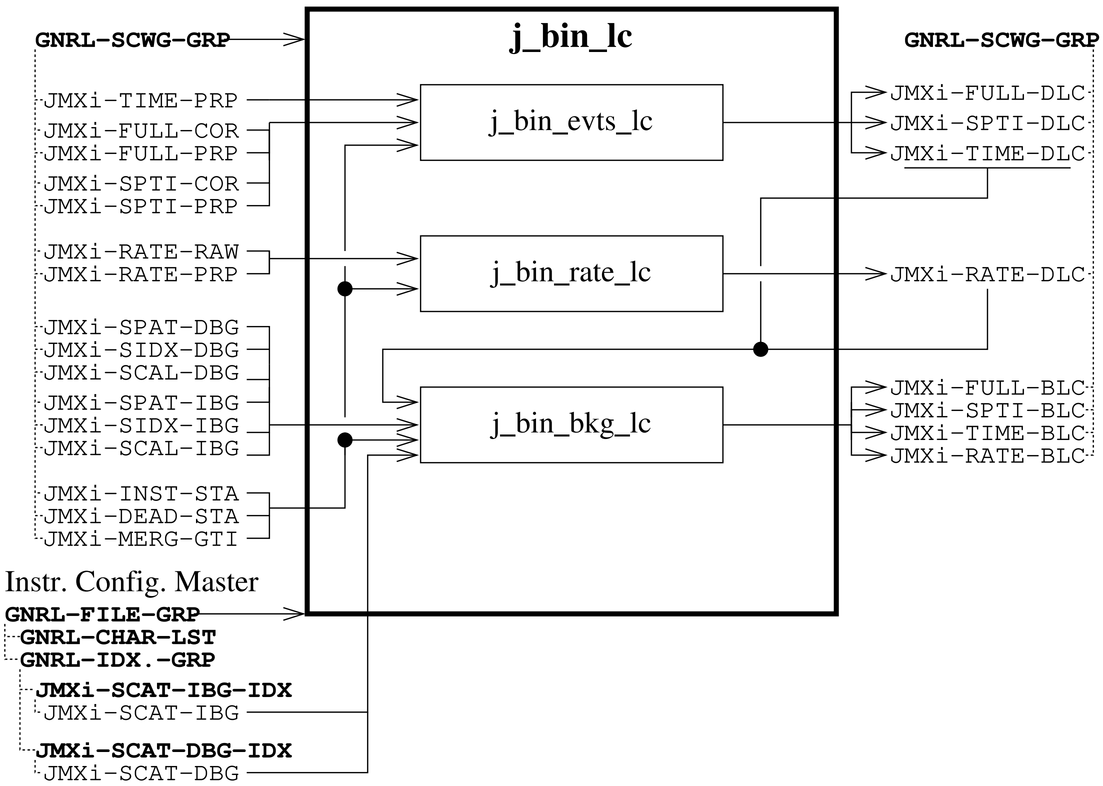

Figure 5.18: j-bin-lc

5.15 j-bin-lc

Type: Analysis Script **Responsible:** ISDC (P. Kretschmar)

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. 217) (event data)
JMXi-FULL-COR (p. 217)
JMXi-SPTI-PRP (p. 223)
JMXi-SPTI-COR (p. 222)
JMXi-TIME-PRP (p. 224)
JMXi-RATE-RAW (p. 220) (countrate data, taken parallel to Restricted Imaging)

JMXi-RATE-PRP (p. 220)
JMXi-INST-STA (p. 219) (instrument status)
JMXi-MERG-GTI (p. 219) (good-time intervals)
JMXi-SIDX-DBG (p. 221) (background model data)
JMXi-SPAT-DBG (p. 222)
JMXi-SCAL-DBG (p. 221)
JMXi-SIDX-BKG (p. 221)
JMXi-SPAT-BKG (p. 222)
JMXi-SCAL-BKG (p. 221)
JMXi-SCAT-DBG (p. 221) (background spectra catalogs, *not* in the SWG)
JMXi-SCAT-BKG (p. 221)

Outputs:
JMXi-FULL-DLC (p. 217)
JMXi-FULL-BLC (p. 217)
JMXi-SPTI-DLC (p. 223)
JMXi-SPTI-BLC (p. 222)
JMXi-TIME-DLC (p. 224)
JMXi-TIME-BLC (p. 224)
JMXi-RATE-DLC (p. 220)
JMXi-RATE-BLC (p. 219)

Parameters:

swgDOL p. 43	Science Window Group
IC-Group p. 40	Instrument Configuration Master Group
IC-Alias p. 40	Instrument Configuration version alias
jemxNum p. 40	JEM-X detector number (1 or 2)
baseDir p. 37	Base Repository Directory
osimData p. 41	Are we using data produced by OSim?
timeStep p. 43	Length of integration time bins
timeStart p. 43	Time in IJD of start of first time bin (¡0 = use SWG start)
timeStop p. 43	Time in IJD of end of last time bin (¡0 = use SWG end)
nChanBins p. 40	Number of channel bins (0=bin all, -1=full resolution)
chanLow p. 38	Lower channel bounds (numbers separated by spaces)
chanHigh p. 38	Upper channel bounds (numbers separated by spaces)
nPhaseBins p. 41	Number of phase bins (N¡0 = abs(N) equally spaced bins)
phaseBins p. 41	Phase bin sizes, separated by space (e.g: 0.1 0.2 0.3 ...)
rowSelect p. 42	CFITSIO selection string on events (optional)
gtiNames p. 39	Names of GTI tables to be used
chatter p. 38	Verbosity level
clobber p. 38	Clobber existing output data?

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. 59)

Subordinate components:
j-bin-evts-lc
j-bin-rate-lc
j-bin-bkg-lc

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: Prototype

5.15.1 j-bin-evts-lc

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

Purpose: Fulfill item 18.16 in SRD

Function: Bin JEM-X event data to lightcurves.

Parameters:

inSWG p. 40	Input SWG
outSWG p. 41	Output SWG (override)
jemxNum p. 40	JEM-X instrument number
evtType p. 39	Event mode number, must be 0=FULL, 2=SPTI or 3=TIME
timeStep p. 43	Binning time for lightcurve (sec)
nChanBins p. 40	Number of channel bins (0=bin all, -1=full resolution)
chanLow p. 38	Lower channel bounds (numbers separated by spaces)
chanHigh p. 38	Upper channel bounds (numbers separated by spaces)
clobber p. 38	Clobber existing output data structures (y/n)?
chatter p. 38	Verbosity of output
rowSelect p. 42	CFITSIO selection string on events (optional))
lcRes p. 40	Output lightcurve DOL (override)
fullPrp p. 39	Full Imaging Prepared Data DOL (override)
fullCor p. 39	Full Imaging Corrected Data DOL (override)
sptiPrp p. 43	Spectral-Timing Prepared Data DOL (override)
sptiCor p. 43	Spectral-Timing Corrected Data DOL (override)
timePrp p. 43	Timing Prepared Data DOL (override)
instStat p. 40	Instrument Status Data DOL (override)
deadTime p. 38	Deadtime History Table DOL (override)
goodTime p. 39	Good-Time Interval Table DOL (override)

Inputs:

JMXi-FULL-PRP (p. 217)
JMXi-FULL-COR (p. 217)
JMXi-SPTI-PRP (p. 223)
JMXi-SPTI-COR (p. 222)
JMXi-TIME-PRP (p. 224)
JMXi-INST-STA (p. 219)
JMXi-MERG-GTI (p. 219)

Outputs:

JMXi-FULL-DLC (p. 217)
JMXi-SPTI-DLC (p. 223)

JMXi-TIME-DLC (p. 224)

Resources: No 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.

Parent components: j-bin-lc(p. 171)

Subordinate components:
None

Dependencies from other components: Raw and prepared data must exist in the input SWG.

Pending TBDs: None

ISDC context: Science Analysis

Delivery date: v1.2.0 15/11/2000

5.15.2 j-bin-rate-lc

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

Purpose: Fulfill item 18.15 in SRD

Function: Bin JEM-X event data to lightcurves.

Inputs:
JMXi-RATE-PRP (p. 220)
JMXi-RATE-COR (p. 219)
JMXi-INST-STA (p. 219)
JMXi-MERG-GTI (p. 219)

Outputs:
JMXi-RATE-DLC (p. 220)

Parameters:

inSWG p. 40	Input Science Window Group DOL
outSWG p. 41	Output Science Window Group DOL (override)
jemxNum p. 40	JEM-X instrument number
timeStep p. 43	Binning time for spectra (0.0=bin all data together)
clobber p. 38	Clobber existing output data structures (y/n)?
chatter p. 38	Verbosity of output
rowSelect p. 42	CFITSIO selection string on events (optional)
lcRes p. 40	Output lightcurve DOL (override)
rateRaw p. 42	Countrate Raw Data DOL (override)
ratePrp p. 42	Countrate Prepared Data DOL (override)
instStat p. 40	Instrument Status Data DOL (override)
deadTime p. 38	Deadtime History Table DOL (override)
goodTime p. 39	Good-Time Interval Table 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 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. 171)

Subordinate components:
None

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: v1.1.0 15/11/2000

5.15.3 j-bin-bkg-lc

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

Purpose: Not foreseen in SRD

Function: Bin background model data to spectra.

Parameters:

inSWG p. 40	Input Science Window Group DOL
outSWG p. 41	Output Science Window Group DOL (override)
jemxNum p. 40	JEM-X instrument number
evtType p. 39	TM format number: 0=FULL,1=REST,2=SPTI,3=TIME or 5=RATE
clobber p. 38	Clobber existing output data structures (y/n)?
chatter p. 38	Verbosity of output
dbgSpecCat p. 38	Diffuse background spectral model table DOL
ibgSpecCat p. **??**	Instrument background spectral model table DOL
lcRes p. 40	Input detector lightcurve DOL (override)
backRes p. 37	Output background lightcurve DOL (override)
dbgSpecMap p. 38	Diffuse background spectral indices DOL (override)
dbgSpatial p. 38	Diffuse background spatial distribution DOL (override)
dbgScale p. 38	Diffuse background scaling factors DOL (override)
ibgSpecMap p. **??**	Instrm. background spectral indices DOL (override)
ibgSpatial p. **??**	Instrm. background spatial distribution DOL (override)
ibgScale p. **??**	Diffuse background scaling factors DOL (override)

Inputs:

JMXi-FULL-DLC (p. 217)
JMXi-SPTI-DLC (p. 223)
JMXi-TIME-DLC (p. 224)
JMXi-RATE-DLC (p. 220)
JMXi-INST-STA (p. 219)
JMXi-MERG-GTI (p. 219)
JMXi-SIDX-DBG (p. 221)
JMXi-SPAT-DBG (p. 222)
JMXi-SCAL-DBG (p. 221)
JMXi-SIDX-BKG (p. 221)
JMXi-SPAT-BKG (p. 222)
JMXi-SCAL-BKG (p. 221)
JMXi-SPEC-DBG (p. **??**)
JMXi-SPEC-BKG (p. **??**)

Outputs:
JMXi-FULL-BLC (p. 217)
JMXi-SPTI-BLC (p. 222)
JMXi-TIME-BLC (p. 224)
JMXi-RATE-BLC (p. 219)

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. 171)

Subordinate components:
None

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 (-DLC) data.

Pending TBDs:

ISDC context: Science Analysis

Delivery date: 1.1.0 10/11/2000

5.16 j-src-extract-lc

Type: Application **Responsible:** Stockholm

Purpose: Fulfill item 18.23 in SRD

Function: Extract light curves and spectra for individual sources in imaging data.

Inputs:
JMXi-INST-STA (p. 219)
JMXi-FULL-COR (p. 217)
JMXi-REST-COR (p. 220)
JMXi-FULL-PRP (p. 217)
JMXi-REST-PRP (p. 220)
JMXi-SRCL-SFR (p. **??**)
JMXi-MASK-MOD (p. 219)
JMXi-WTRN-MOD (p. **??**)
JMXi-CVIG-MOD (p. **??**)
JMXi-DETE-MOD (p. 215)
JMXi-COLL-MOD (p. 215)
JMXi-QEFF-MOD (p. 219)
JMXi-ATXE-MOD (p. 214)
JMXi-ATAL-MOD (p. 214)
JMXi-ATCU-MOD (p. 214)
JMXi-ATMO-MOD (p. 214)
JMXi-MERG-GTI (p. 219)
JMXi-SIDX-DBG (p. 221)
JMXi-SIDX-IBG (p. **??**)
JMXi-SPAT-DBG (p. 222)
JMXi-SPAT-IBG (p. **??**)
JMXi-SCAL-DBG (p. 221)
JMXi-SCAL-IBG (p. **??**)

Outputs:
JMXi-EVNT-WHT (p. **??**)
JMXi-SRCL-LCV (p. **??**)
JMXi-SRCL-BLC (p. 223)
JMXi-SRCL-SPE (p. 224)
JMXi-SRCL-BSP (p. 223)
JMXi-SRCL-ARF (p. 223)

Parameters:
lctimebins,lcenergybins, sptimebins,spenergybins,methodextr,methodmask,methoddet,methodcoll, methodbgrd,delweights,methodweights1,methodweights2

Resources:

Processing:

Read parameters and forward to child parameter files. Call j-src-evt-weights
Call j-src-spectra
Call j-src-lc

If delweights parameter set delete JMXi-EVNT-WHT

Design remarks:
The method used for the source data extraction means that each event is assigned a set of weights (positive or negative) by which it contributes to the different individual sources in the field of view. Light curves and spectra are then created by summing up event contributions in time-energy bins for each source. Background subtraction is an integer part of this procedure.
The event weights (JMXi-EVNT-WHT) are used only by j-src-spectra and j-src-lc. In standard analysis JMXi-EVNT-WHT may be deleted after executing these routines (determind by parameter delweights).

Parent components: None(p. **??**)

Subordinate components:
j-src-evt-weights
j-src-spectra
j-src-lc

Dependencies from other components: The execution of this component must be preceded by background estimation and one of the image generation routines.

Pending TBDs:

Prototyping status: Most of the analysis principles have been tested by prototyping and simulations.

ISDC context:

Delivery date: April 2000

5.16.1 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, outSrcArf

Inputs:
JMXi-IMOD-GRP (p. 218)
JMXi-INST-STA (p. 219) In input SWG
JMXi-DEAD-STA (p. **??**) In input SWG
JMXi-SCAL-BKG (p. 221) In input SWG
JMXi-SCAL-DBG (p. 221) In input SWG
JMXi-SPAT-BKG (p. 222) In input SWG
JMXi-SPAT-DBG (p. 222) In input SWG
JMXi-SCAT-BKG (p. 221)
JMXi-SIDX-BKG (p. 221) In input SWG
JMXi-SCAT-DBG (p. 221)
JMXi-SIDX-DBG (p. 221) In input SWG
JMXi-MERG-GTI (p. 219)
JMXi-SRCL-RES (p. 224) In input SWG
JMXi-FULL-COR (p. 217) In input SWG
JMXi-REST-COR (p. 220) In input SWG
JMXi-FULL-PRP (p. 217) In input SWG
JMXi-REST-PRP (p. 220) In input SWG

Outputs:
JMXi-SRC.-ARF (p. **??**) In output SWG
JMXi-SRC.-LCR (p. 223) In output SWG

Parameters:
INPUT GROUPS:
inSWG p. 40 DOL of input SWG
outSWG p. 41 DOL of output SWG
instMod p. 40 DOL of instrument model group JMXi-IMOD-GRP
bkgSpecCat p. 37 DOL of instr bkg spectral catalogue JMXi-SCAT-BKG
dbgSpecCat p. 38 DOL of diffuse bkg spectral catalogue JMXi-SCAT-DBG
goodTime p. 39 DOL of Good-Time Interval Table (override)

deadTime p. 38 DOL of Deadtime History Table (override)
instStat p. 40 DOL of Instrument Status Data (override)
fullPrp p. 39 DOL of full Imaging Prepared Data (override)
fullCor p. 39 DOL of full Imaging Corrected Data (override)
dbgSpecMap p. 38 DOL of diffuse backgr. spectral indices
dbgSpatial p. 38 DOL of diffuse backgr. spatial distribution
dbgScale p. 38 DOL of diffuse backgr. scaling factors
bkgSpecMap p. 37 DOL of instrument backgr. spectral indices
bkgSpatial p. 37 DOL of instrument backgr. spatial distribution
bkgScale p. 37 DOL of instrument backgr. scaling factors
OTHER INPUTS:
jemxNum p. 40 Number of JEM-X unit (1 or 2)
nChanBins p. 40 Number of energy bands for light curves
chanLow p. 38 Lower PI channels [nChanBins values]
chanHigh p. 38 Upper PI channels [nChanBins values]
evtType p. 39 0 for FULL events, 1 for REST events.
timeStart p. 43 Time at beginning of (first) lightcurve bin.
timeStop p. 43 End time of data used for extraction.
timeStep p. 43 Time integration step (seconds).
bgrMethod p. 37 Choice of backgr. method (0=flat, 1=model).
nPhaseBins p. 41 Number of bins in phaseBins.
phaseBins p. 41 Length of phase bins (in case of cyclic binning).
precisionLevel p. 41 speed/precision level (1-100, 0=Default) vignCorr p. 43 Should program perform vignetting corrections?
deadCorr p. 38 Should program perform deadtime corrections?
chatter p. 38 level of debugging output
clobber p. 38 Clobber existing data in output data structures?
OVERRIDES:
outSrcLC p. 41 DOL of override JMXi-SRC.-LCR
outSrcArf p. 41 DOL of override JMXi-SRC.-ARF
outBkgLC p. 41 DOL to override JMXi-SRCL-BLC

Resources: Execution is approximatly linear with number of events. Memmory requirement is dominated by input event data.

Processing:
Read input parameters
Read instrument model
Read instrument status
Read source parameters
Loop over sources
    Calculate instrument response for source

Write Ancillary Response File, JMXi-SRC.-ARF
End loop sources
Read background model
Loop over sources
Loop over energy bins
Calculate net background contribution (spatial and energy parts)
Loop over energy bins
End loop over sources

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

Loop over events
Loop over sources and energy
Add contribution to source time energy bin
End loop sources and energy
End loop over events

Loop over time bins and sources
Loop over energy bins
Correct countrates for greyfilter and deadtime
Calculate and correct for background.
End energy bins
End loop over time bins and sources

Write lightcurves to JMXi-SRC.-LCR
Write backgr. lightcurve to JMXi-SRC.-LCR

Design remarks:
JMXi-SRC.-LCR. will contain a background subtracted light curve for an individual source and energy band. A background light curve will also be stored in JMXi-SRC.-LCR for evaluation purposes.
Either FULL or REST imaging events are used. To extract both, the executable has to be run twice producing two lightcurves, one based on FULL and one on REST imaging data. a third option is considered (evtType=-1) where both types of events are used, producing a single, merged, lightcurve.
The possibility to choose alternative background methods (e.g. fitting) is forseen, but not implemented (will use parameter bgrMethod). The option to select flat or model background will be introduced in version 3.0. The design and code is close to that of j-src-spectra. j-src-lc will be delivered after completion of j-src-spectra v3.0.

Subordinate components: Functions of the executable:
j-src-lc-work
j-src-lc-params
j-src-lc-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-min-range
j-gti-overlap
j-read-GTI
j-slc-GetFileName

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 2.0 delivered 22 Nov 2000

5.17 j-performance

Type: Application/GUI **Responsible:** S. Maisala

Purpose: 18.6 18.7 18.8

Function: Interactive GUI tool to perform long and short term monitoring of properties of the detector.

- Tool is based on IOSM and ROOT (osm-hk-display)
- User can take a choise of available parameters and set time interval (IJD).
- Program collects house keeping and calibration data and merge them to one table
- Available parameters:
  HK-parameters: voltages, currents, pressures, and temperatures
  and Performance parameters
- Provides a list of scalar parameters and their values and maximum predefined limits.
- Displays parameters vs time or other parameter

Inputs:
GNRL-OBSG-GRP (p. 226) (observation group)
JMXi-CSSW-HRW (p. 215) (housekeeping data)
JMXi-CSSW-AVR (p. 215)
JMXi-CSSW-CNV (p. 215)
JMXi-INST-STA (p. 219) (Instrument mode and grey filter status)
JMXi-ALRT-LIM (p. 214) (alert limits values)

Outputs:
GUI display of chosen parameters. (p. **??**)

Parameters:

inObsGrp p. 40	Input observation group
perfTable p. 41	Output table
jemxNum p. 40	JEM-X instrument number (1,2,0=both)
timeStart p. 43	IJD
timeStop p. 43	IJD
hkType p. 39	house keeping type (0=AVG, 1=RAW,2=CONV)
binFactor p. 37	binning factor (0=full)
parNames p. 41	ascii file of chosen parameters
chatter p. 38	Verbosity of logfile output (0,1,2,3,4,5)

Resources:

Design remarks:
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(p. **??**)

Subordinate components:
j-perf-collect-data

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: June 2002

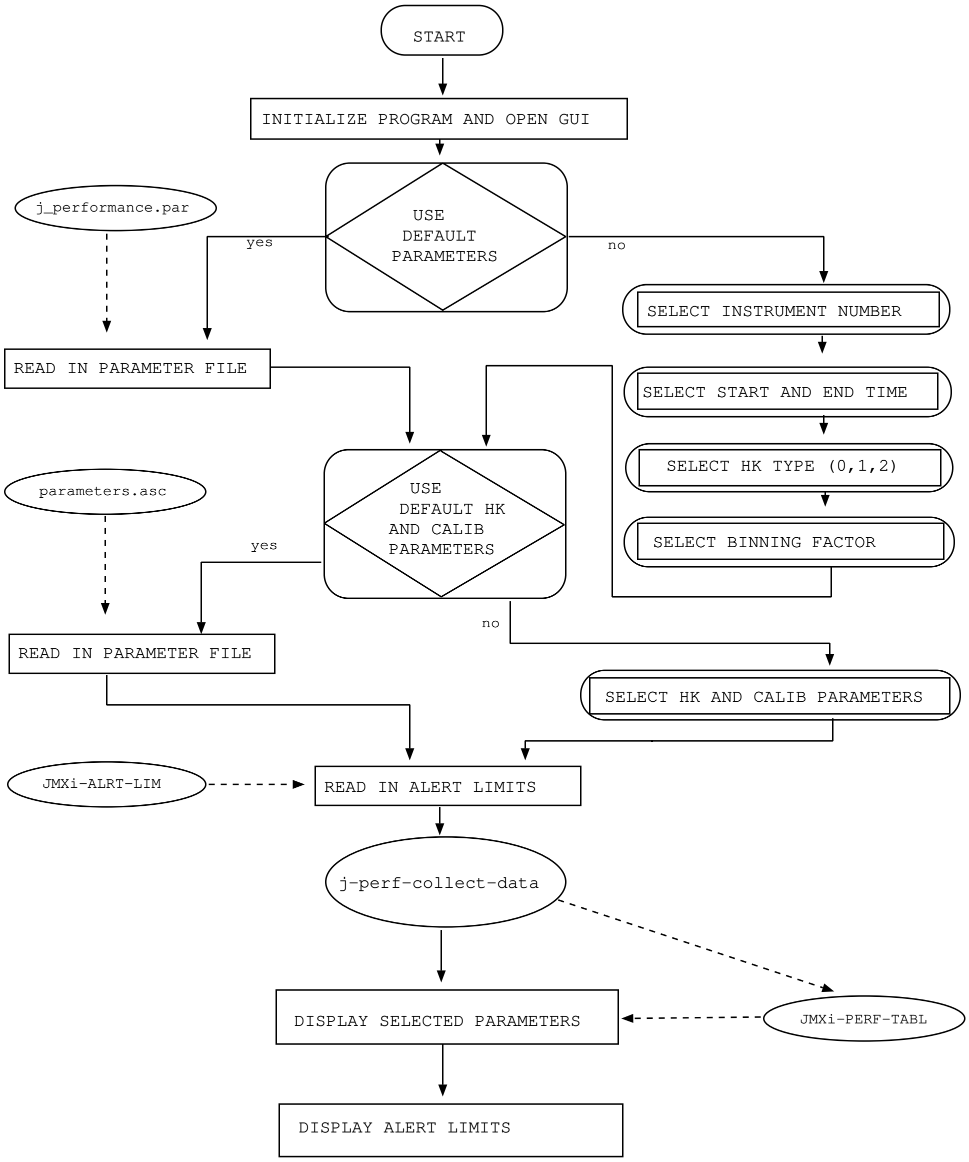

Figure 5.19: j-performance

5.17.1 j-perf-collect-data

Type: Executable **Responsible:** Helsinki

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. 226) (observation group)
JMXi-CSSW-HRW (p. 215) (housekeeping data)
JMXi-CSSW-AVR (p. 215)
JMXi-CSSW-CNV (p. 215)
JMXi-INST-STA (p. 219) (Instrument mode and grey filter status)

Outputs:
JMXi-PERF-TBL (p. 219)

Parameters:

inObsGrp p. 40	Input observation group
perfTable p. 41	Output performance table
jemxNum p. 40	JEM-X instrument number (1,2,0 = both)
timeStart p. 43	IJD
timeStop p. 43	IJD
parNames p. 41	ascii file of chosen parameters

Resources:

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.
Create column names by combining suffix and parameter names.

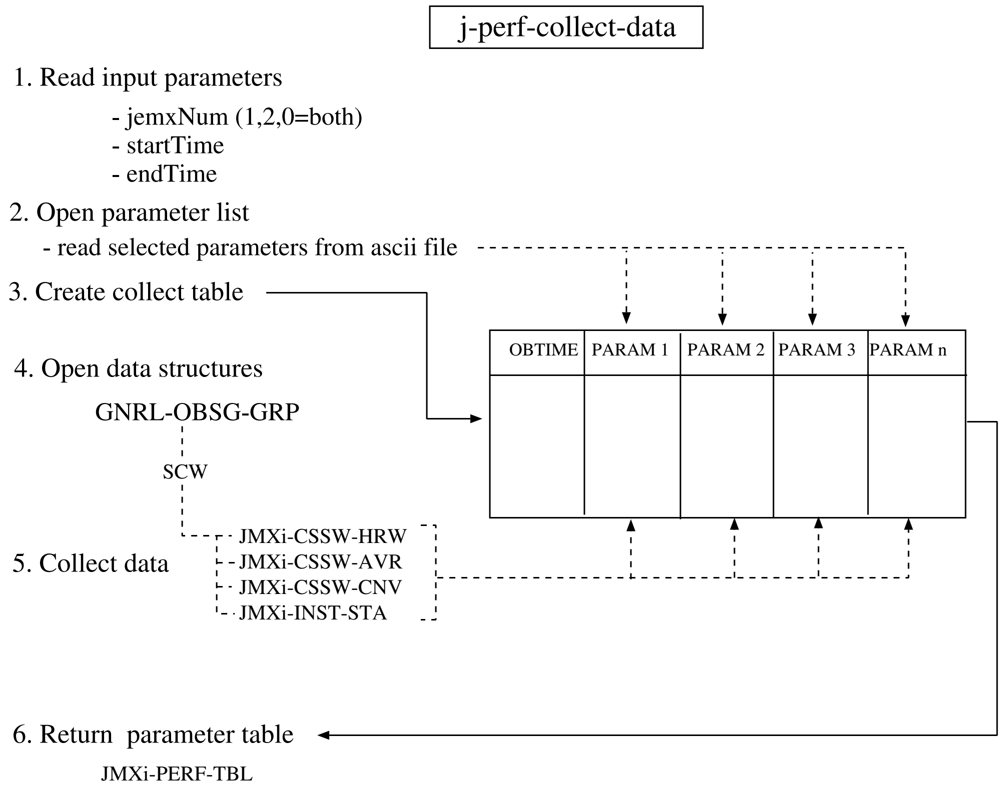

Figure 5.20: j-perf-collect-data

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. 189)

Subordinate components:
none

Dependencies from other components: None

Pending TBDs: Parameters and resources are TBD.

Prototyping status: working executable exist

ISDC context: Off-line processing, Fig. 1

Completion status: **99%**

Delivery date: August 2002

5.18 j-ical

Type: Interactive tools **Responsible:** DSRI

Purpose: 18.7.1, 18.7.2 though much of this functionality not forseen in SRD or URD

Function: This is a collection of interactive tools to allow the IT or Instrument Representative to monitor and re-calculate various calibrations and corrections usually performed automatically. In the case of the FRSS data the user can fit interactively individual sets of spectra which have given problems. The other tools allow the user to re-calculate various instrument model tables used to perform energy, position and deadtime corrections. If the difference between the new and currently used tables is considered significant the IT can resubmit the Instrument Model Group with the new table(s).
This component is not an actual deliverable, but just a way of referring to the collection of interactive calibration tools.

Subordinate components:
j-ical-frss
j-ical-gain
j-ical-collimator
j-ical-dead

Dependencies from other components: 1)It is assumed that the Standard Analysis Pipeline has run many times over before invoking these tools to monitor results for several revolutions. In practice there will be data from many revolutions to monitor and all revolutions from beginning of the mission to the time of the latest set of consolidated data should be available to the user.

ISDC context: ISDC V3

5.18.1 j-ical-frss

Type: Interactive tool **Responsible:** DSRI

Purpose: 18.7.1, 18.7.2 though much of this functionality not forseen in SRD or URD

Function: This is an interactive tool to allow the IT or Instrument Representative to monitor the results of automatic gain calibration performed using the FRSS spectra. If severe problems are found the user can re-process a particular set of spectra interactively

Inputs:
JMXi-GAIN-CAL-IDX (p. 218)
JMXi-GAIN-CAL (p. 218) From the index table
JMXi-FRSS-CRW-IDX (p. 217)
JMXi-FRSS-CRW (p. 217) From the index table
JMXi-IMOD-GRP-IDX (p. 218)
JMXi-IMOD-GRP (p. 218) From the index table
JMXi-DETE-MOD (p. 215) In the instrument model group
JMXi-FBDS-MOD (p. 217) In the instrument model group
JMXi-CALB-MOD (p. 215) In the instrument model group
JMXi-GAIN-AVG (p. 218)

Outputs:
JMXi-GAIN-AVG (p. 218)
JMXi-GAIN-CAL-IDX (p. 218) Personal version
JMXi-GAIN-CAL (p. 218) Personal version

Parameters:

gainHist p. 39	DOL of gain calibration table index
inCAL p. 40	DOL of calibration spectra data index
instMod p. 40	DOL of instrument model table index
jemxNum p. 40	Number of jemx unit (1 or 2)
chatter p. 38	Verbosity of logfile output (0,1,2,3,4,5)

Resources: No special resources forseen

Processing:
Please see processing diagram on page 197: Ovals indicate data stuctures; round-cornered boxes indicate user input; dashed lines show data structure I/O; striped lines indicate data structure referencing in groups and index tables

Design remarks:
1) Initially the calibration source intensities are displayed over the entire mission to date since it is the very long term decay of these values over years that is of interest. The user can choose other time ranges to study these results after the full display.
2) Very few parameters are required for this tool since most input is prompted from the user.

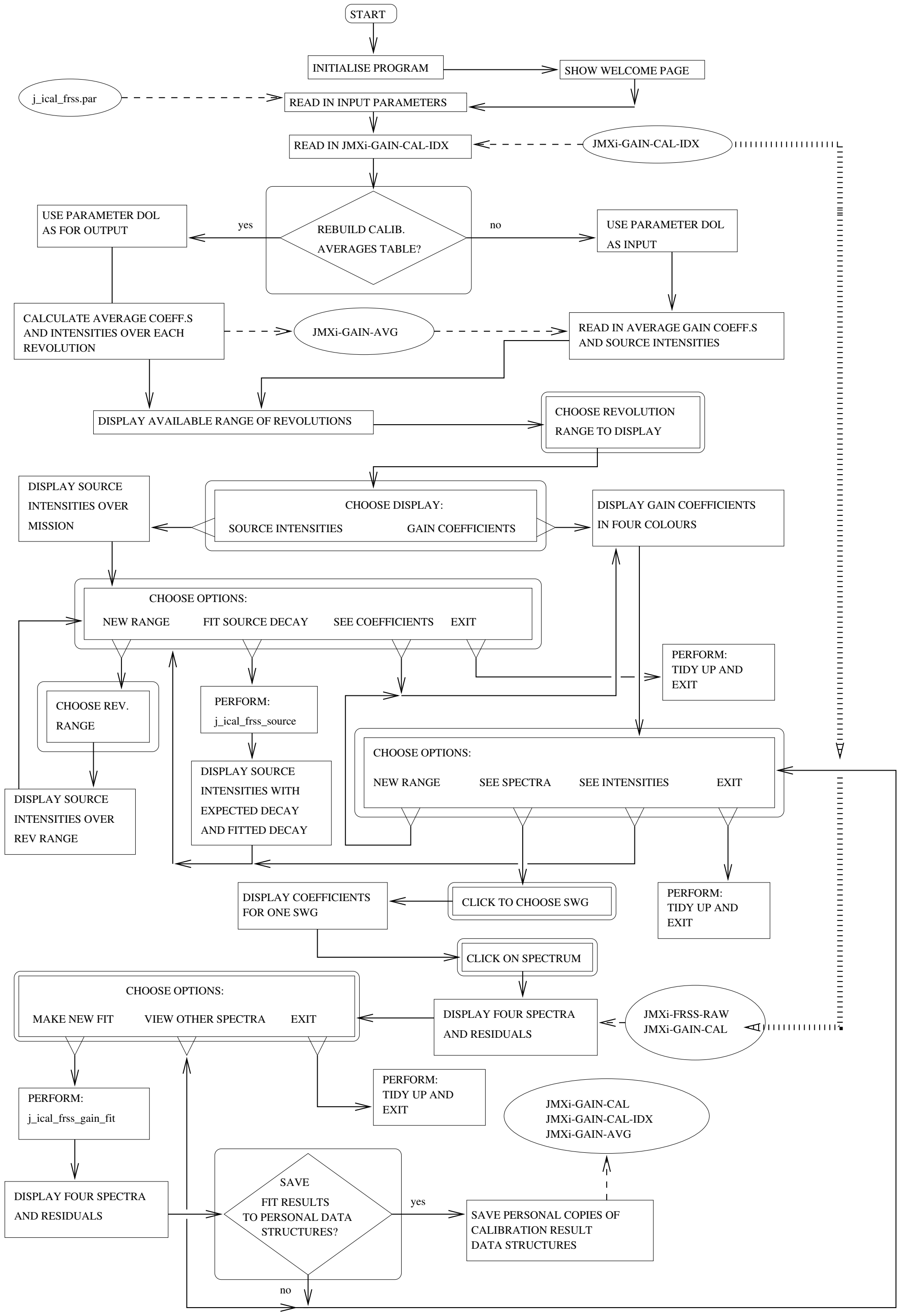

Figure 5.21: j-ical-frss

Where parameters are missing or empty these are requested from the user.
3) This is still only a very rough sketch of this tool for which no code yet exists.

Parent components: None - stand-alone tool

Subordinate components: None

Dependencies from other components: 1)It is assumed that the Revolution File Pipeline has run many times over before invoking the tool to provide calibration results for one or more revolutions. In practice there will be data from many revolutions to monitor and all revolutions from beginning of the mission to the time of the latest set of consolidated data should be available to the user.

ISDC context: ISDC V3

Delivery date: Version 1.0 will be delivered early/mid 2002

5.18.2 j-ical-collimator

Type: Interactive tool **Responsible:** DSRI

Purpose: Replacement for Known Source Calibration expected in 18.5.3

Function: The collimator footprint on the detector during multiple slew Science Windows is used to determine position corrections for FULL and REST events across the entire detector area, and build new position correction models JMXi-CORX-MOD, JMXi-CORY-MOD should these be needed.

Inputs:
GNRL-SCWG-GRP-IDX (p. 226)
GNRL-SCWG-GRP (p. 226)
JMXi-FULL-RAW (p. 218) In SWG
JMXi-FULL-COR (p. 217) In SWG
JMXi-REST-RAW (p. 220) In SWG
JMXi-REST-COR (p. 220) In SWG
JMXi-IMOD-GRP (p. 218)
JMXi-COLL-MOD (p. 215) In the instrument model group
JMXi-CORX-MOD (p. 215) In the instrument model group
JMXi-CORY-MOD (p. 215) In the instrument model group

Outputs:
JMXi-CORX-MOD (p. 215) Personal copy, NOT in standard group
JMXi-CORY-MOD (p. 215) Personal copy, NOT in standard group

Parameters:

inSWG p. 40	DOL of science window group index GNRL-SCWG-GRP-IDX
instMod p. 40	DOL of the current instrument model group
jemxNum p. 40	Number of jemx unit (1 or 2)
chatter p. 38	Verbosity of logfile output (0,1,2,3,4,5)

Resources: No special resources forseen

Processing:
Please see processing diagram on page 200: Ovals indicate data stuctures; round-cornered boxes indicate user input; dashed lines show data structure I/O; striped lines indicate data structure referencing in groups and index tables

Design remarks:
1) Very few parameters are required for this tool since most input is prompted from the user. Where parameters are missing or empty these are requested from the user.

Parent components: None — stand-alone tool

Subordinate components: None

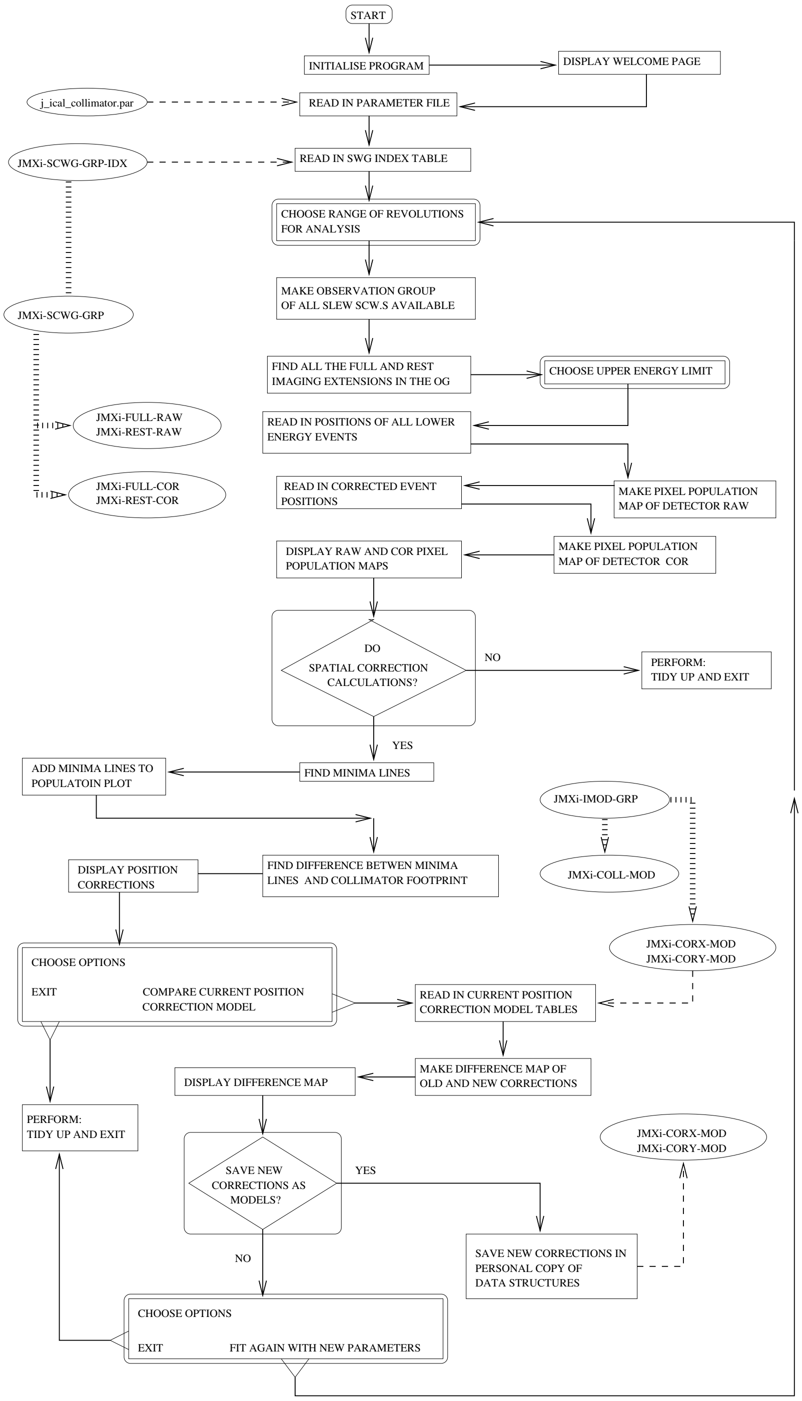

Figure 5.22: j_ical_collimator

Dependencies from other components: 1) This tool can only be used after ISDC PreProcessing (PP), Data Preparation (PRP) and Data Correction (COR) have run on the data to be analysed, since we need both raw and corrected full imaging data.

ISDC context: ISDC V3

Delivery date: Version 1.0 will be delivered in early 2002

5.18.3 j-ical-dead

Type: Interactive tool **Responsible:** DSRI

Purpose: Not forseen in SRD or URD

Function: To calculate instrument dead time as a function of CPU mode, hardware trigger rate and measured event rate, and to compare these results with the dead time model currently used in the instrument model group. If significant differences are found between the model and the results just calculated, the user can choose to store the results so that they can be used as an updated version of the dead time model.

Inputs:
GNRL-SCWG-GRP-IDX (p. 226)
GNRL-SCWG-GRP (p. 226)
JMXi-FULL-RAW (p. 218) In SWG
JMXi-SPTI-RAW (p. 223) In SWG
JMXi-TIME-RAW (p. 224) In SWG
JMXi-CSSW-HRW (p. 215) In SWG
JMXi-IMOD-GRP (p. 218)
JMXi-DEAD-MOD (p. 215) In the instrument model group

Outputs:
JMXi-DEAD-MOD (p. 215) Personal copy

Parameters:

inSWG p. 40	DOL of science window group index GNRL-SCWG-GRP-IDX
instMod p. 40	DOL of the current instrument model group
jemxNum p. 40	Number of jemx unit (1 or 2)
chatter p. 38	Verbosity of logfile output (0,1,2,3,4,5)

Resources: No special resources forseen

Processing:
Please see processing diagram on page 203: Ovals indicate data stuctures; round-cornered boxes indicate user input; dashed lines show data structure I/O; striped lines indicate data structure referencing in groups and index tables

Design remarks:
1) One value of dead time is calculated for each packet and the results saved according to CPU mode, hardware triggers received in 8 seconds (duration of the packet) and number of processed events received during the packet, corrected for grey filter.

2) Very few parameters are required for this tool since most input is prompted from the user. Where parameters are missing or empty, these are requested from the user.

3) This is only a very rough sketch of this tool, for which no code yet exists.

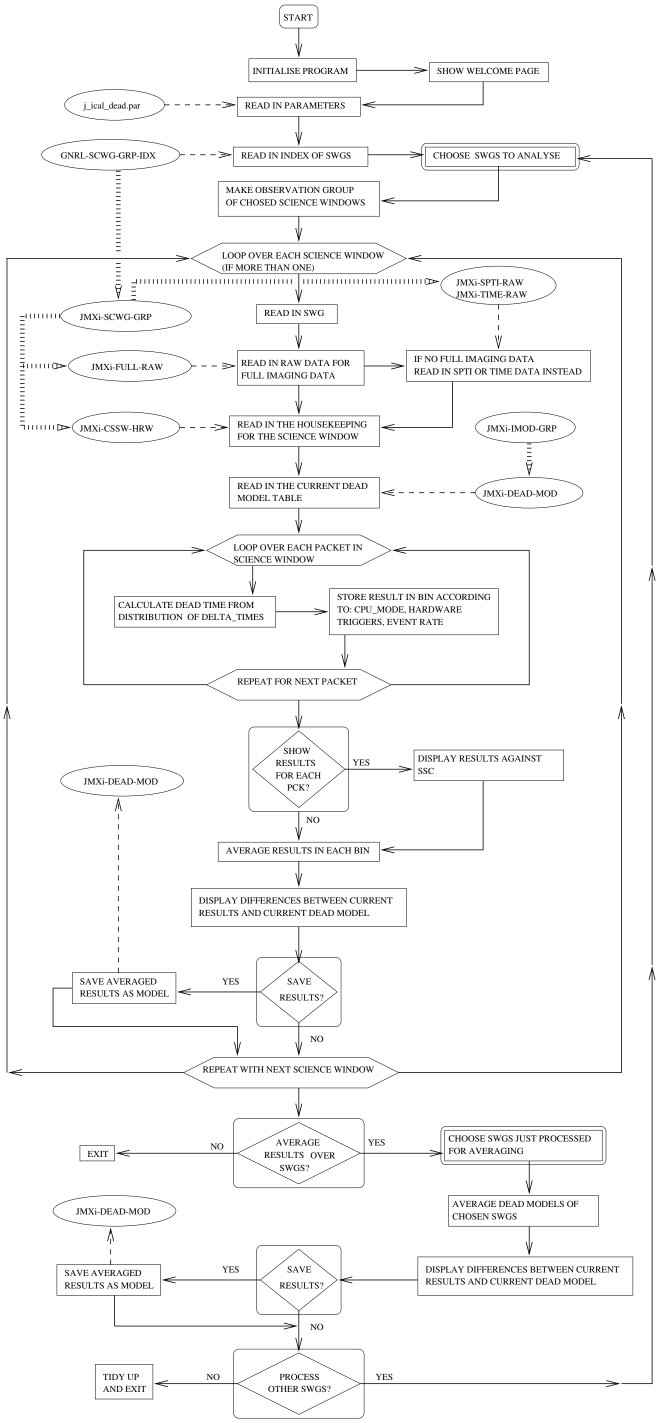

Figure 5.23: j_ical_dead

Parent components: None — stand-alone tool

Subordinate components: None

Dependencies from other components: 1) The PreProcessing and the Science Window Pipeline must have completed their work on the raw data before this tool can be used. However, no analysis in the Science Analysis Pipeline needs to have been done.

ISDC context: ISDC V3

Delivery date: Version 1.0 will be delivered in February 2002

5.18.4 j-ical-gain

Type: Interactive tool **Responsible:** DSRI

Purpose: Not forseen in SRD or URD

Function: To calculate the detector gain for pixel bins all over the detector using known lines either from the sky background and/or from the xenon fluorescence in the detector. Comparisons are made with the fractional variation of the gain compared to each anode's gain determined by FRSS fitting in j_calib_gain_fitting. If significant changes are found the user can choose to do a full pixel-by-pixel analysis and save the result for use as the new spatial gain variation table.

Inputs:
GNRL-SCWG-GRP-IDX (p. 226)
JMXi-GAIN-CAL-IDX (p. 218)
JMXi-FULL-RAW (p. 218) In SWG
JMXi-FULL-PRP (p. 217) In SWG
JMXi-IMOD-GRP (p. 218)
JMXi-SPAG-MOD (p. 222) In the instrument model group

Outputs:
JMXi-SPAG-MOD (p. 222)

Parameters:

inSWG p. 40	DOL of science window group index GNRL-SCWG-GRP-IDX
instMod p. 40	DOL of the current instrument model group
jemxNum p. 40	Number of jemx unit (1 or 2)
chatter p. 38	Verbosity of logfile output (0,1,2,3,4,5)

Resources: No special resources forseen

Processing:
Please see processing diagram on page 206: Ovals indicate data stuctures; round-cornered boxes indicate user input; dashed lines show data structure I/O; striped lines indicate data structure referencing in groups and index tables

Design remarks:
1) Very few parameters are required for this tool since most input is prompted from the user. Where parameters are missing or empty, these are requested from the user.
2) This is a very rough sketch of this tool for which no code yet exists.

Parent components: None — stand-alone tool

Subordinate components: None

Dependencies from other components: 1) The tool must have access to both raw and prepared data since the OBT of every event must be known.
2) The results for and entire revolution of FRSS calibration spectra must have been calculated

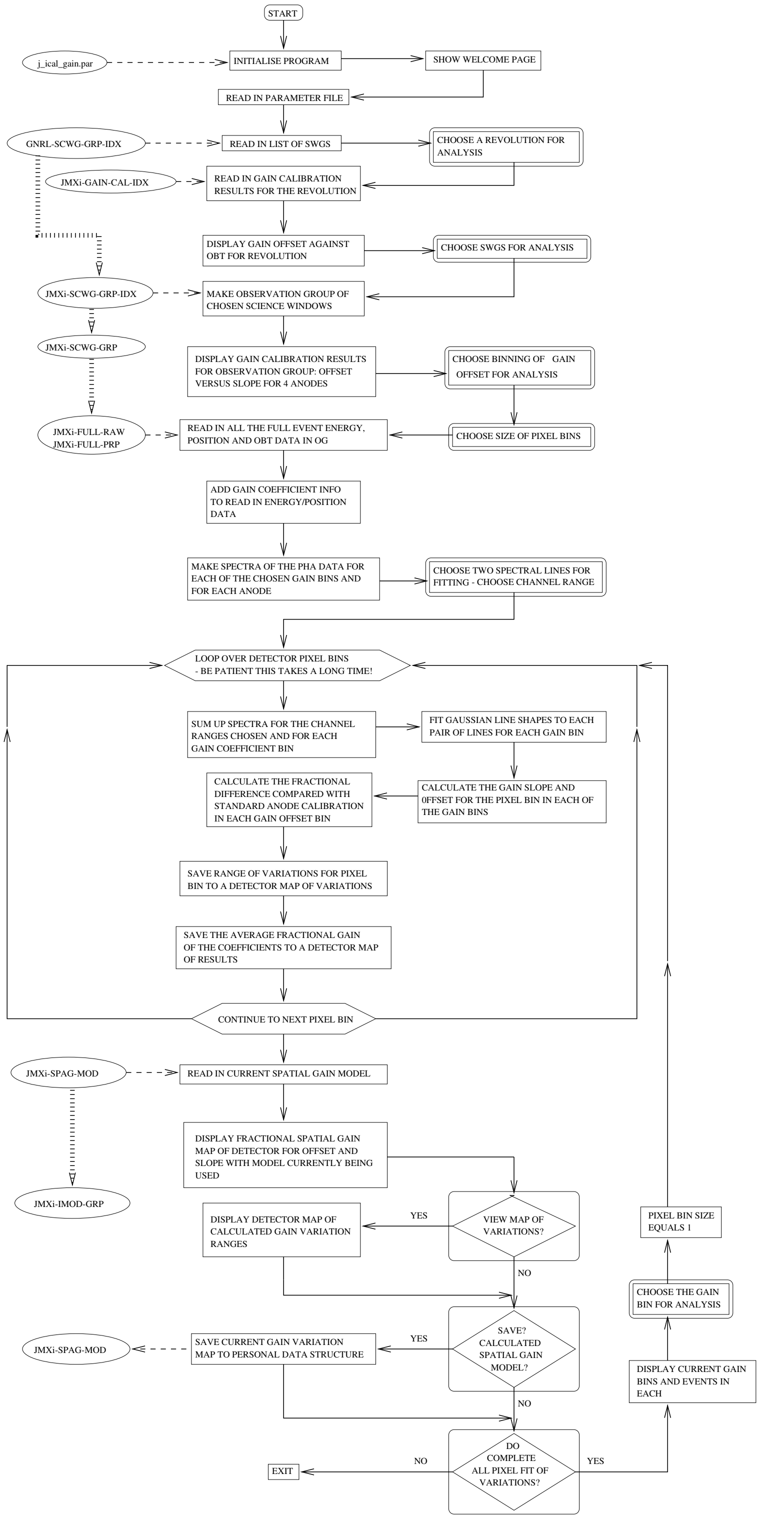

Figure 5.24: j-ical-gain

by j_calib_gain_fitting, so that the tool has comparison nominal gain coefficients for each anode segment.

ISDC context: ISDC V3

Delivery date: Version 1.0 will be delivered in mid 2002

5.19 j-bkg-interactive

Type: Interactive tool
Responsible: Helsinki

Purpose: 18.17

Function: To examine and modify existing background catalogues. Spatial images, spectras, scaling factor databases and all included keywords and parameters can be examined by GUI. The user can modify models and parameters and save results to the user catalogue. These catalogues can be used to override standard analysis default background models. Optional feature is to add new models to the background model catalogues

Inputs:
GNRL-SCWG-GRP (p. 226)
JMXi-EVTS-SHD (p. 216)
JMXi-SPAT-BKG-IDX (p. 222) (background spatial models)
JMXi-SPAT-DBG-IDX (p. 222)
JMXi-SIDX-BKG-IDX (p. **??**) (spectral index maps)
JMXi-SIDX-DBG-IDX (p. **??**)
JMXi-SCAT-BKG (p. 221) (background spectra catalogues)
JMXi-SCAT-DBG (p. 221)
JMXi-ASPE-BKG (p. 214) (background scaling factor databases)
JMXi-ORBI-BKG (p. 219)
JMXi-RADL-BKG (p. 219)
JMXi-ATTI-DBG (p. 214)
AUXL-ATTI-HIS (p. 225) (auxliary data)
AUXL-ATTI-NRT (p. 225)
AUXL-ORBI-HIS (p. 225)
AUXL-ORBI-NRT (p. 226)
JMXi-CSSW-HRW (p. 215) (house keeping data)
JMXi-CSSW-CNV (p. 215)

Outputs:
JMXi-SPAT-BKG-IDX (p. 222) (spatial models)
JMXi-SPAT-DBG-IDX (p. 222)
JMXi-SIDX-BKG-IDX (p. **??**) (spectral index maps)
JMXi-SIDX-DBG-IDX (p. **??**)
JMXi-ASPE-BKG (p. 214) (background scaling factor databases)
JMXi-ORBI-BKG (p. 219)
JMXi-RADL-BKG (p. 219)
JMXi-ATTI-DBG (p. 214)

Parameters:

chatter p. 38	Verbosity of logfile output (0,1,2,3,4,5)

Resources: Resources of this component

Processing:
Look processing diagram

Design remarks:
User can interactively select background model and method for deriving the scaling factor for the model. Selected background model can be either observed or simulated background model.

Parent components: None(p. **??**)

Subordinate components:
none

Dependencies from other components: none

Pending TBDs: Resources and parameters are TBD.

Prototyping status: None

ISDC context: Off-line processing, Fig. 1

Delivery date: June 2002

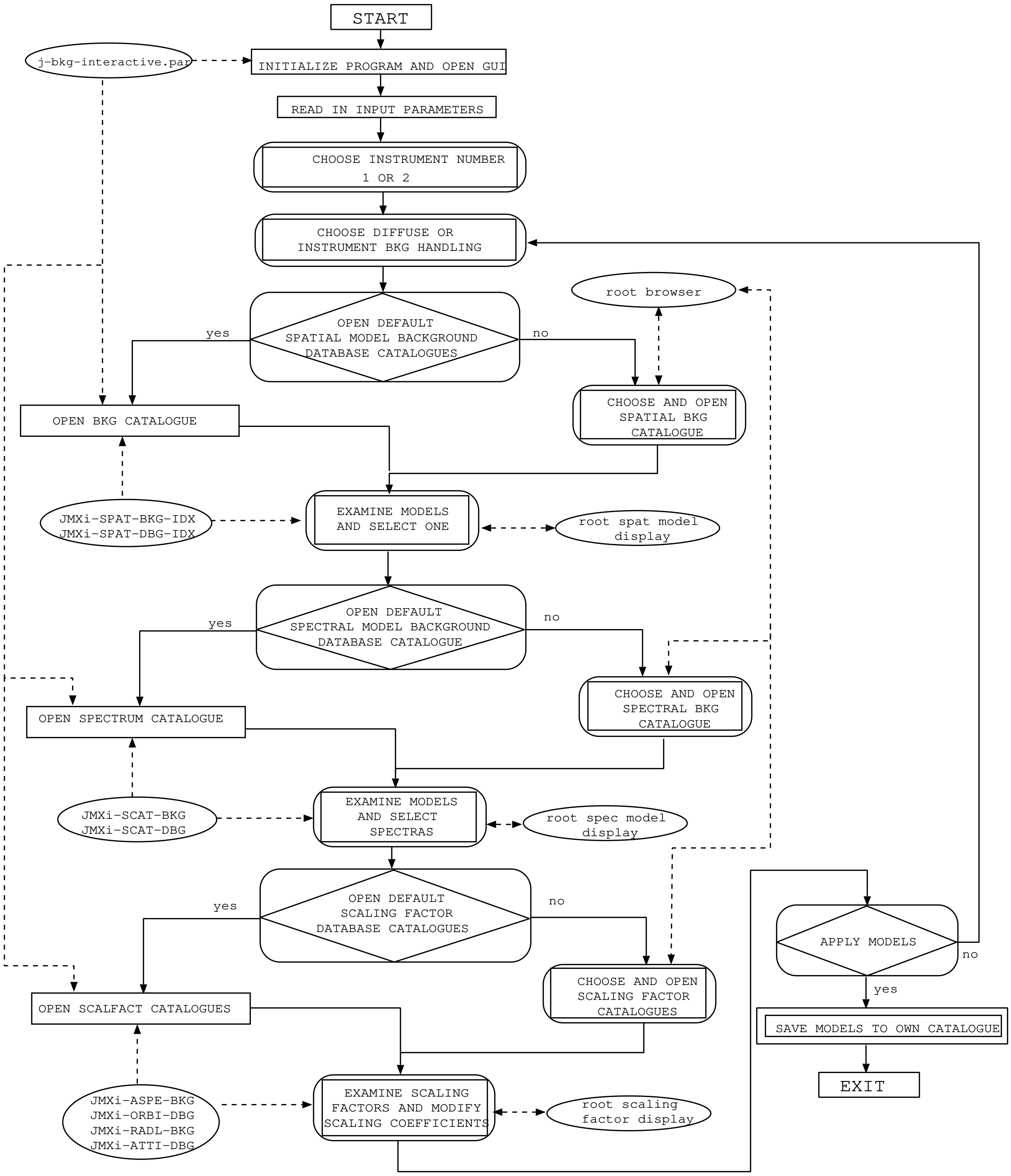

Figure 5.25: j-bkg-interactive

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'.

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

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 3.0 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.

'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-ASPE-BKG *BKG*	Sun aspect angle background scaling	SUN_ASPECT SCALFACT
JMXi-ASPE-BKG-IDX *BKG*	Index table	VERSION VSTART VSTOP
JMXi-ATAL-MOD	Attenuation data for aluminium	ENERGY ATTLEN
JMXi-ATBE-MOD	Attenuation data for beryllium	ENERGY ATTLEN
JMXi-ATCU-MOD	Attenuation data for copper	ENERGY ATTLEN
JMXi-ATMO-MOD	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	Attenuation data for xenon	ENERGY ATTLEN

Name	Contents	Column Names
JMXi-BORE-MOD	Alignment matrix between instrument and star trackers	JMATBORE
JMXi-CALB-MOD	Telemetry format calibration spectra energy binning tables	CHANNEL
JMXi-COLL-MOD	Collimator model array	FOOTPRINT ARRAY:X;Y
JMXi-CONV-MOD	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	Index table of JMXi-CONV-MOD data structures: CC	VERSION VSTART VSTOP DB_VERS DATE RESPONSI
JMXi-CORX-MOD	X position correction table	CORRECTION ARRAY: X;Y
JMXi-CORY-MOD	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-DEAD-MOD	Dead time model for various hardware trigger rates	TRIGGERS DEAD8_1 DEAD16_1
JMXi-DEAD-SCP *DEAD*	Dead times	OB_TIME DEADTIME
JMXi-DETE-MOD	Detector model data	ENERGY FWHM TBD
JMXi-DFEE-CRW *RAW*	DFEE dump calibration telemetry: CC	SSC PACK_TIME MEM_ID ADDRESS LENGTH DUMP
JMXi-DIAG-CPR *PRP*	Prepared full OBTs of diagnostic events: CC	OB_TIME

Name	Contents	Column Names
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-DPOS-MOD *OSIM*	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-DBXS-MOD	Extra-galactic diffuse x-ray spectrum used by JEM-X Observation Simulator	ENERGY RATE
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-MOD	Electronic calibration model	TRIGGERS DEAD8_1 DEAD16_1
JMXi-ENRG-MOD	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

Name	Contents	Column Names
JMXi-FBDS-MOD	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-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-FULB-MOD	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-BLC *BIN_T*	Background lightcurve counts for detector in full imaging mode	BACKRATE BACK_ERR
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-DLC *BIN_T*	Full imaging detector lightcurve	OB_TIME EXPOSURE RATE, STAT_ERR, SYS_ERR, QUALITY
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

Name	Contents	Column Names
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	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-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-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	Instrument model group. See page 227 for more details	Members: JMXi-[ATAL ATCU ATMO ATXE BORE CALB COLL CORX CORY DEAD DETE DPOS DXBS ECAL ENRG FBDS FULB MASK QEFF RBDS RESB SBDS SPAG SPCB]-MOD
JMXi-IMOD-GRP-IDX	Index table over instrument model groups	VERSION VSTART VSTOP

Name	Contents	Column Names
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	Mask array model	MASK ARRAY: X;Y
JMXi-MERG-GTI *GTI*	Merged GTIs	OBT_START OBT_END
JMXi-MOSA-IMA *IMA*	Mosaic'ed image	Pixel values: PHI ANGLE (CAR) or X (TAN); THETA ANGLE (CAR) or Y (TAN)
JMXi-MOSA-IMA-IDX *IMA*	Index table of mosaic'ed images	IMATYPE CHANMIN CHANMAX ROWIDEN1 ROWIDEN2 EVNTYPES E_MIN E_MEAN E_MAX
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-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-BLC *BIN_T*	Rate background lightcurve	BACKRATE BACK_ERR
JMXi-RATE-COR *COR*	Corrected countrate data: CC	RATE_CORR STATUS

Name	Contents	Column Names
JMXi-RATE-DLC *BIN_T*	Rate detector lightcurve	OB_TIME EXPOSURE RATE STAT_ERR SYS_ERR QUALITY
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	Corrected upper limits of the bins used to convert original 4096 energy channels into 8 restricted imaging telemetry channels	CHANNEL
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

Name	Contents	Column Names
JMXi-RMF.-GRP	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	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-DBG *BKG*	Spectral indices array for diffuse background: CC	INDEX ARRAY: X;Y
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 CHANMIN CHANMAX ROWIDEN1 ROWIDEN2 EVNTYPES E_MIN E_MEAN E_MAX EXPOSURE
JMXi-SLSP-ARF *SPE*	Ancilliary spectral response for sources in list	SPECRESP

Name	Contents	Column Names
JMXi-SPAG-MOD	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	Corrected upper limits of the bins used to convert the 4096 ADC channels into 64 spectral mode telemetry energy channels	CHANNEL
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 : Start of the integration time interval in IJD 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-BLC *BIN_T*	Background lightcurves for Spectral/timing mode	BACKRATE BACK_ERR
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

Name	Contents	Column Names
JMXi-SPTI-DLC *BIN_T*	Lightcurves for Spectral/timing mode over entire detector	OB_TIME EXPOSURE RATE STAT_ERR SYS_ERR QUALITY
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.-LCR *LCR*	Countrate lightcurves for listed sources	OB_TIME EXPOSURE RATE_i STAT_ERR_i, SYS_ERR_i, QUALITY
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-BLC *LCR*	Background lightcurves for listed sources	BACKRATE_i, BACK_ERR_i
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

Name	Contents	Column Names
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-BLC *BIN_T*	Background countrate lightcurves for timing mode	BACKRATE BACK_ERR
JMXi-TIME-DLC *BIN_T*	Detector background lightcurves for timing mode	OB_TIME EXPOSURE RATE STAT_ERR SYS_ERR QUALITY
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-VIGN-MOD	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-NRT	near real time 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 MJD_PREVALID UPDATED DUBIOUS CONFIRMED
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

Name	Contents	Column Names
AUXL-ORBI-NRT	Near real time orbit data: CC	Same as AUXL-ORBI=HIS
AUXL-TCOR-HIS	historic time correlation: CC	OB_TIME TIME ERT_CORRECTION COUPLE_VALIDITY OBT0 TIME0 GRADIENT CONSTANT CORRELATION_STATUS
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

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

ADC channel binning tables for PHA telemetry:
JMXi-CALB-MOD JMXi-FULB-MOD JMXi-RESB-MOD

PI binning tables:
JMXi-FBDS-MOD JMXi-RBDS-MOD JMXi-SBDS-MOD JMXi-SPCB-MOD-MOD

Response matrix:
JMXi-ENRG-MOD

Mask and detector models:
JMXi-COLL-MOD JMXi-DETE-MOD JMXi-DPOS-MOD JMXi-MASK-MOD JMXi-BORE-MOD JMXi-DXBS-MOD, JMXi-VIGN-MOD

Quantum efficiency:
JMXi-QEFF-MOD

Event correction models:
JMXi-CORX-MOD JMXi-CORY-MOD JMXi-SPAG-MOD JMXi-DEAD-MOD

Electronic calibration model:
JMXi-ECAL-MOD

Individual descriptions of these data structures can be found in the table above.

It is the responsiblity of the DSRI/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.

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 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 2.0, 21st October 1998

[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 1.3, 21st December 2000

[10] Coding and Testing Standards, ISDC/CTS, Issue 2.3, 13th September 2000

[11] Data Access Layer Users Guide, ISDC, Issue 1.4.3, 17th July 2000

[12] Common Library Users Manual, ISDC, Issue 1.3.5, 12th January 2001

[13] DAL3 JEM-X User's Manual, ISDC, Issue 3.4, 4th February 2001

[14] DAL3 General User Manual, ISDC, Issue 3.3.5, 4th February 2001

[15] DAL3 Auxilliary User Manual, ISDC, Issue 3.3, 28th December 2000

[16] DAL3 Housekeeping Users Manual, ISDC, Issue 2.6.1, 28th September 2000

[17] DAL3 Catalogue User's Manual, ISDC, Issue 2.4, 28th September 2000

[18] Parameter Interface Library Users Manual, ISDC, Issue 1.3, 26th June 1999

[19] Reporting Interface Library Users Manual, ISDC, Issue 3.0, 8th June 2000

[20] isdcroot Users Manual, ISDC, Issue 1.6.0, 9th October 2000

[21] JEM-X Experimental Interface Document Part B, DSRI, Issue 5.2, 17th March 2000

[22] DFEE On-Board Software Specification Document. DSRI, Issue 1.0 25th September 1999.

[23] IASW for JEM-X /DPE -Software Specification Document. Space Research Center PAS, Warsaw. Version 1.11 16/06/00.

[24] Joint European X-Ray Monitor JEM-X, X-Ray Monitor for the INTEGRAL Mission: User Manual Issue 4.2 February 2001.

[25] ISDC Quick Look Analysis Architectural Design Document (ISDC/ADDQLA) Issue 2.1 April 4th 2001

[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 0.3 November 30th 2000

[28] ISDC INTEGRAL Burst Alert System Architectural Design Document Issue 2.0 October 6th 2001

Index

DSRI

DSRI