



# Basic properties of the INTEGRAL data analysis system OSA 5.1

Igor Telezhinsky,  
Astronomical Observatory of Kiev  
University

# What can you use OSA for?

- Corrections, if needed
- Dead-time calculation, good-time interval selections
- Image reconstruction (including mosaics)
- Source identification and extraction
- Count extraction (spectra, light curves)
- Handling of Instrument Characteristics (IC)

# What can't you use OSA for?

- Image analysis
- Spectral fitting
- Timing analysis (period search,...)
- etc.

# OSA Components

## ➤ **Generic tools**

- Developed and maintained by ISDC
- FTOOL-like

```
dal_list dol=my_file.fits[1]
```

- Written in C

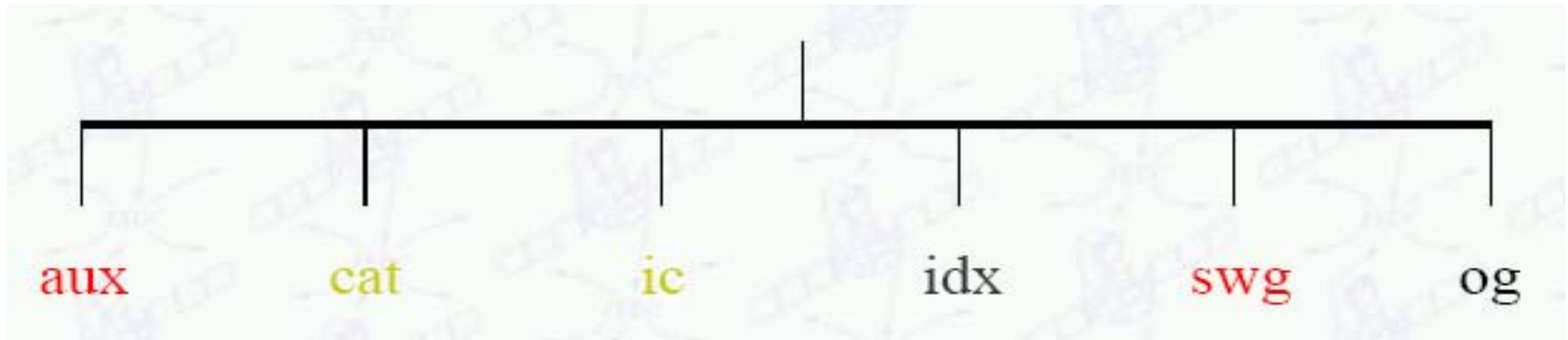
## ➤ **Instrument-specific executables**

- Developed and maintained by Instrument Teams
- FTOOL-like
- Some components written in F90

## ➤ **Analysis scripts**

- Developed and maintained by ISDC/Instrument Teams
- C++ scripts based on ROOT and isdcroot
- Work in script, command-line or GUI modes

# Data repository

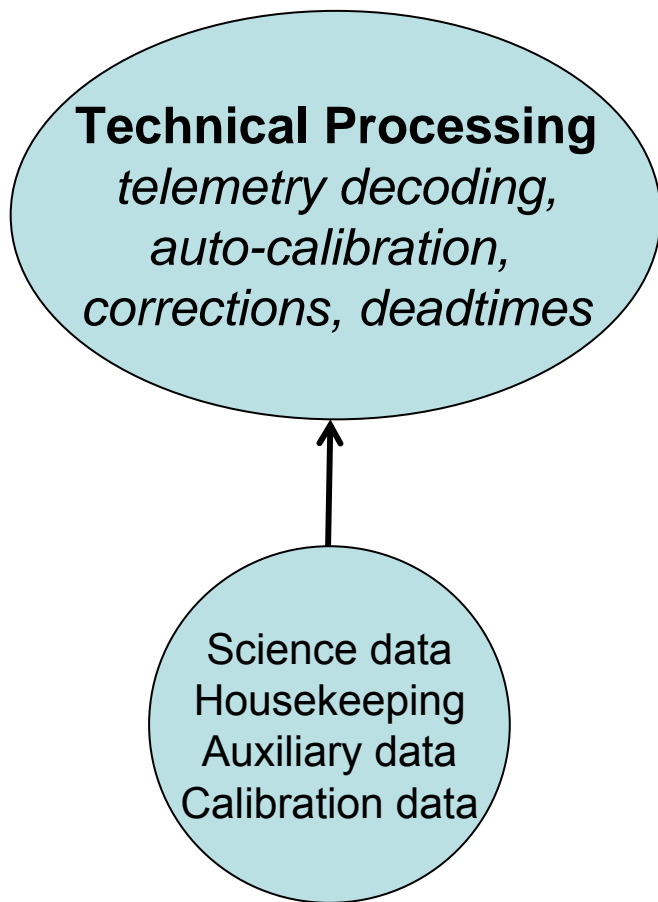


- **aux**: Auxiliary data; attitude, orbit, program, ...  
(downloaded from the Archive)
- **cat**: Source catalogue (downloaded with OSA)
- **ic**: Instrument characteristics; calibration, instrument models, ...  
(downloaded with OSA)
- **idx**: Index tables; sort of databases...
- **swg**: Science Window data; instrument data  
(downloaded from the Archive)
- **og**: Observation data; products generated by OSA

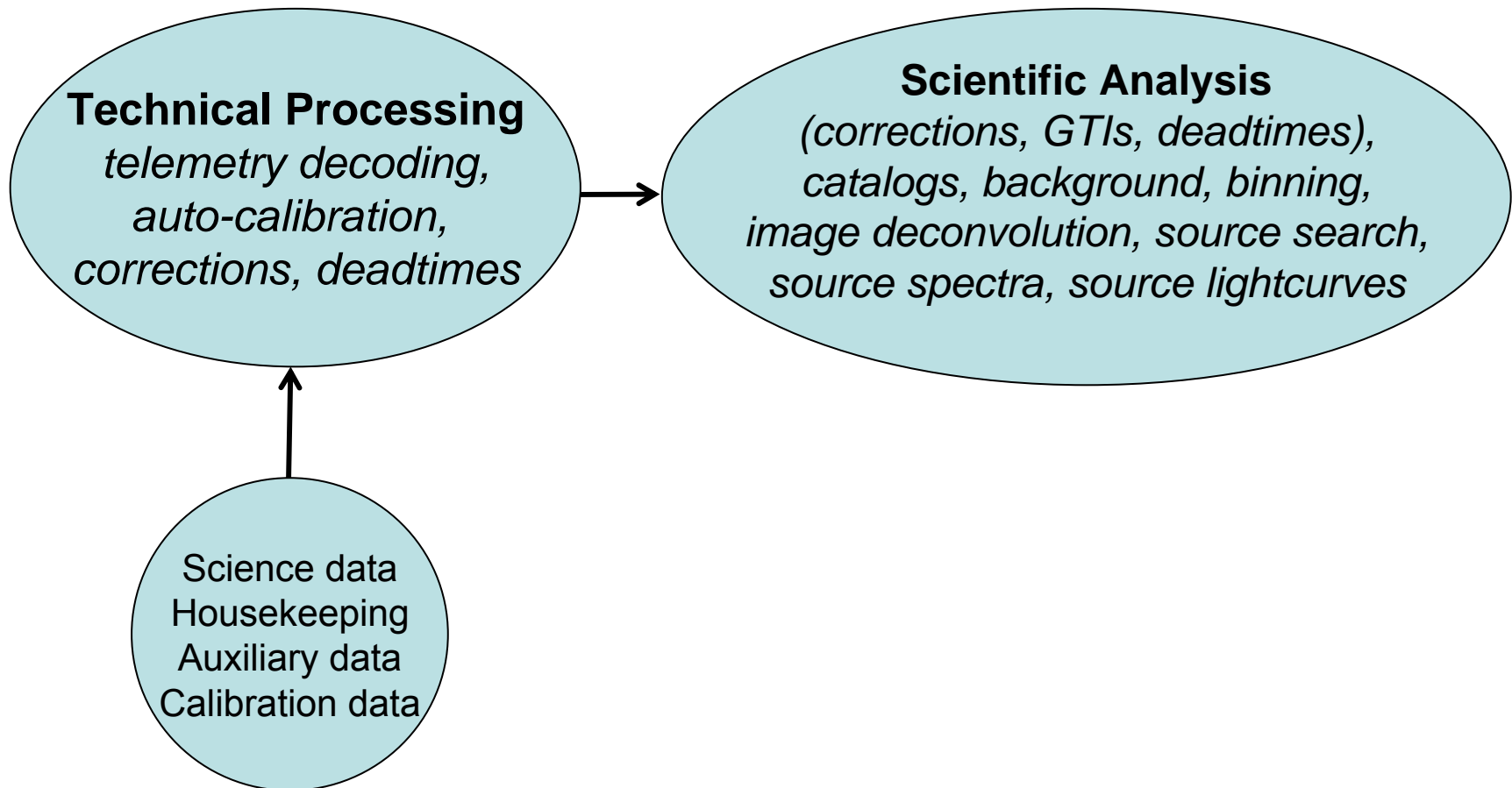
# Data way of life till final results

**Technical Processing**  
*telemetry decoding,  
auto-calibration,  
corrections, deadtimes*

# Data way of life till final results

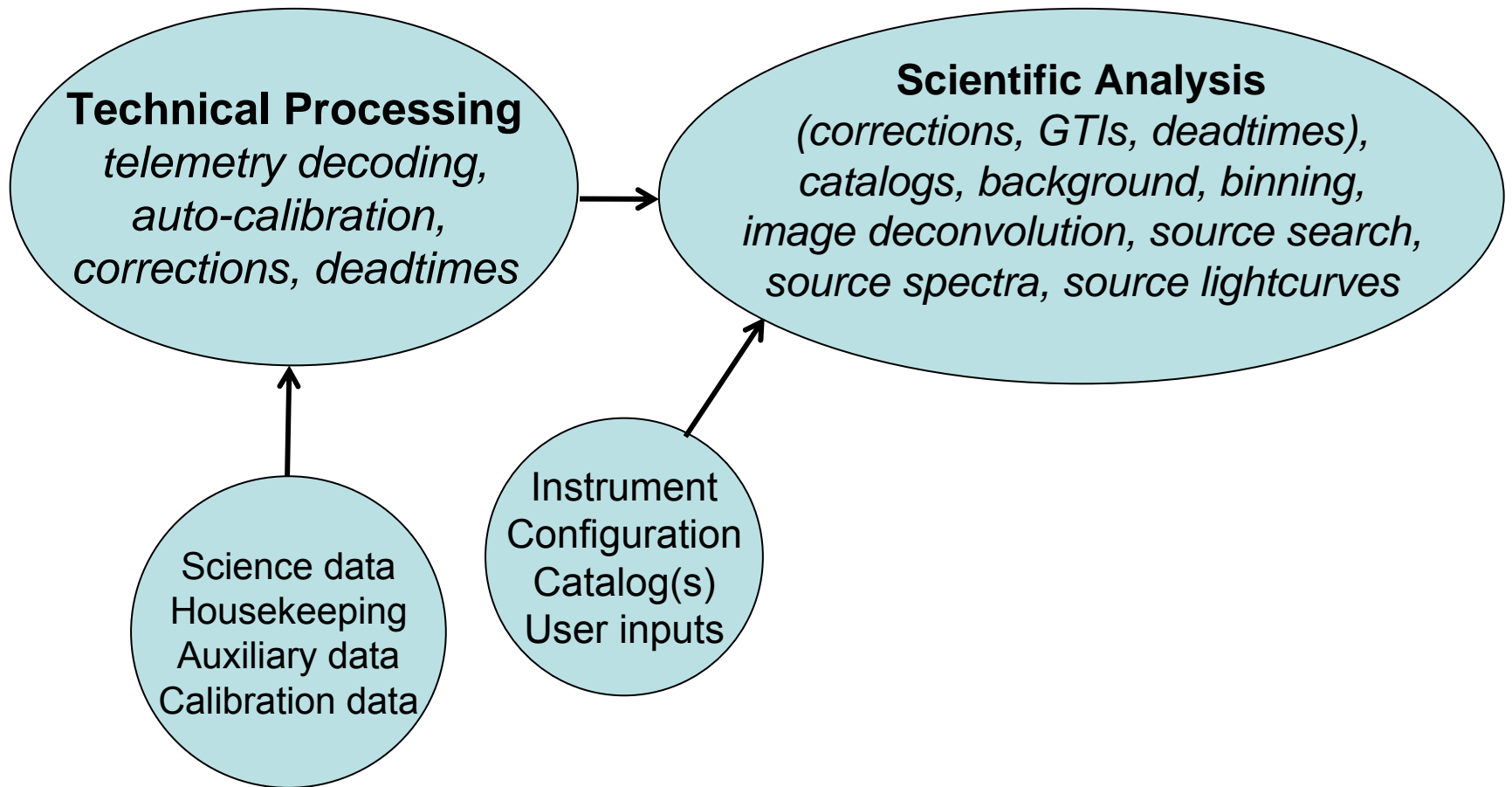


# Data way of life till final results

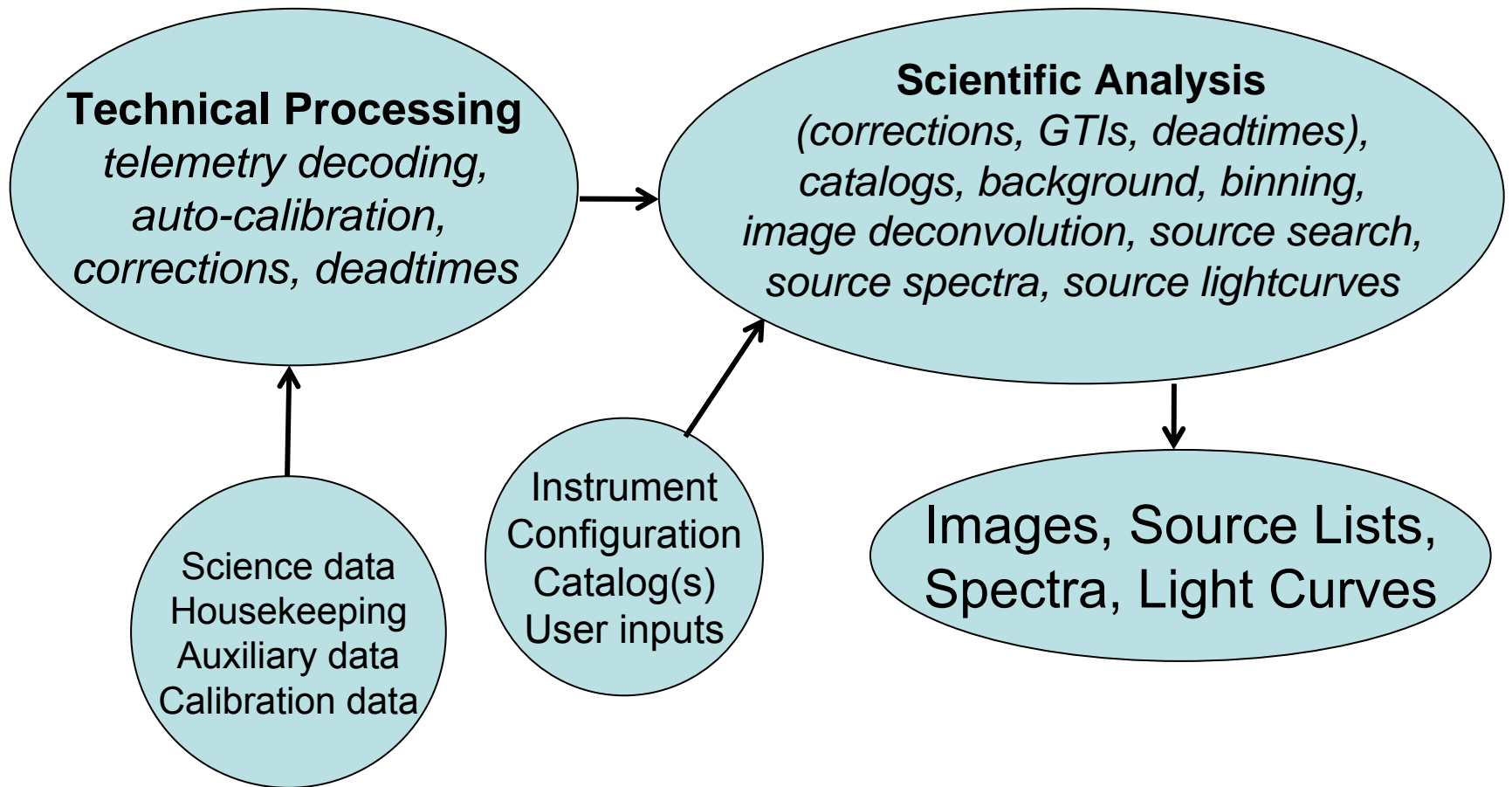




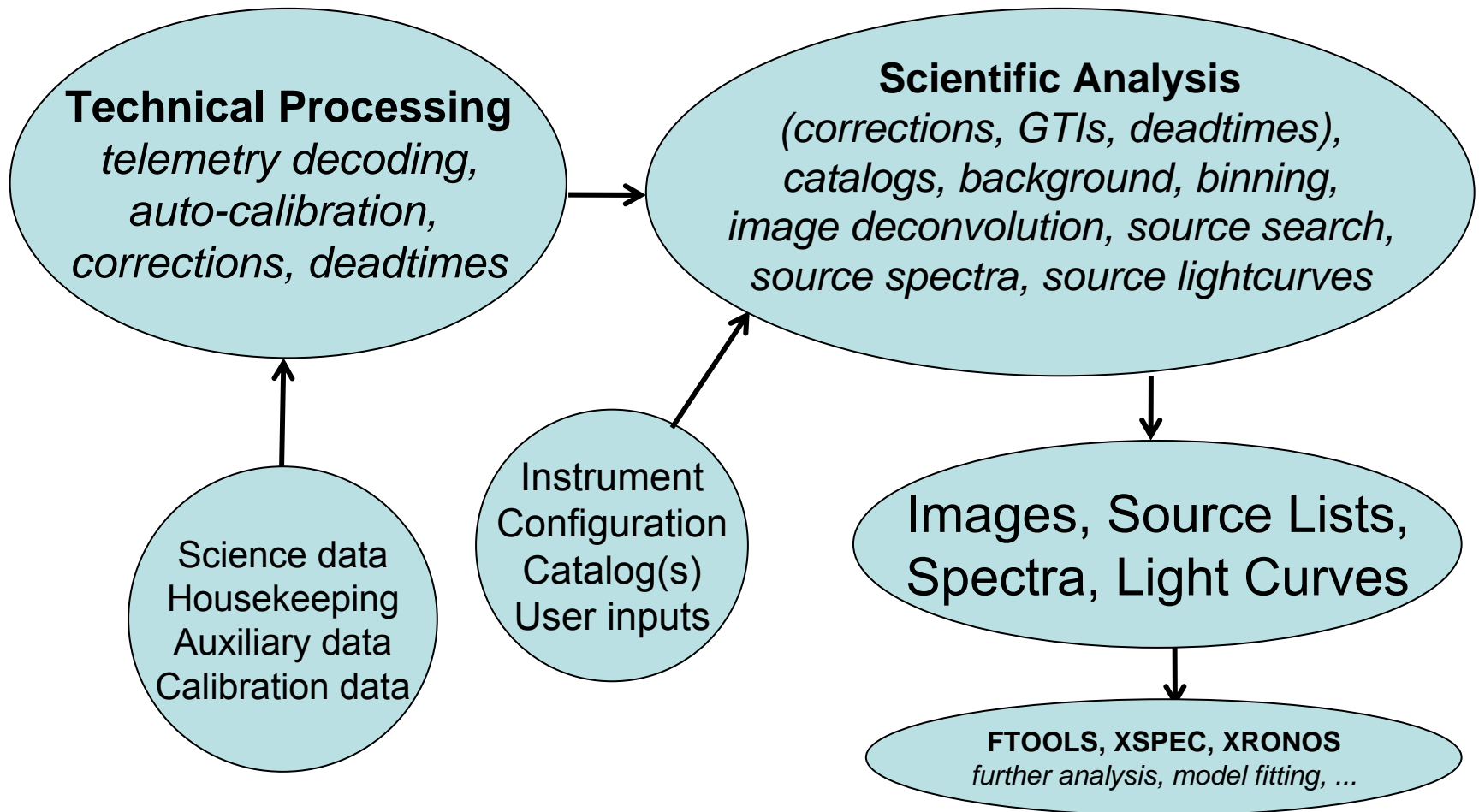
# Data way of life till final results



# Data way of life till final results



# Data way of life till final results



# The way OSA works

- One concept for all instruments.
- Data analysis pipeline is a chain of executables each having a parameter file (so called Parameter Interface Layer implementation).
- Parameters can be queried or hidden
- Parameters can be set on the command line

```
dal_list dol=my_file.fits[1]
```
- Or through the scripts and GUIs
- For the users, OSA allows easy control of the output (screen vs log file), whether in interactive or in batch mode (so called Reporting Interface Layer implementation).

# The way OSA works

- OSA uses FITS files for any kind of data (images, tables, spectra etc.)
- FITS allows special data structures that do not contain any data, but only points to other data structure.
- The basic building block of data for the analysis system is a GROUP.
- **Science Window groups** contain all the data generated by INTEGRAL during a Science Window (pointing, slew,...).
- **Observation groups** group Science Windows on the basis of user defined criteria and all the results from OSA run.

# Scripts

- The analysis of the data from a single INTEGRAL instrument requires a few tens of executables
- Scripts are necessary to help users to run an entire analysis in the correct order
  - Steps are called “Level”: COR, DEAD, GTI, IMA, SPE, LCR, ...
  - There is complete flexibility in the Level sequence, except order
- ISDC scripts are actually C++ code with isdcroot/ROOT classes
- C++ is not mandatory... Advanced users can develop their own scripts in shell, Perl, Python, tcl, ..., but, for the moment, the need to provide scripts in more user-friendly languages that users can reuse, modify is not that obvious...

# GUIs

- GUIs exist for all instruments.
- Simplify parameter filling for interactive sessions, with little loss of capabilities.
- Hides “hidden” parameters not to scare anyone 😊
- With OSA 5.1, GUIs really become parameter editors.
  - Save command lines.
  - Load parameters from scripts.
  - Can also be used as regular GUI's!

GUIs standard analysis usage




# Just before the analysis...

- Set the Environment...

- `setenv ISDC_ENV directory_of_OSA_sw_installation`
- `setenv ISDC_REF_CAT`  
`"$REP_BASE_PROD/cat/hec/gnrl_refr_cat_0023.fits[1]"`
- `source $ISDC_ENV/bin/isdc_init_env.csh`

- ... and create observation group

- `cd $REP BASE PROD`
- `og_create idxSwg=isgri_gc.lst ogid=isgri_gc`  
`baseDir="." instrument=IBIS`

A blue oval callout with a white border and a blue arrow pointing from the oval to the `idxSwg=isgri_gc.lst` part of the command line above.

*list of Science  
Windows!*

# ibis\_science\_analysis

ibis\_science\_analysis

Main

startLevel: COR

endLevel: IMA2

GENERAL\_levelList: COR,GTI,DEAD,BIN\_I,BKG\_I,CAT\_I,IMA,IMA2,BIN\_S,SPE,LCR,COMP,CLEAN

CAT\_refCat: /isdc/arc/rev\_2/cat/hec/gnr1\_refr\_cat\_0020.fits[1][ISGRI\_FLAG==1] browse

SWITCH\_disableIsgrI:  checked: yes

SWITCH\_enablePICsIT:  checked: yes

SCW1\_GTI\_gtiUserI: browse

SCW1\_GTI\_TimeFormat: IJD

ISGRI IMA

ISGRI SPE and LCR

PICsIT analysis

Save As

Load

Reset

Run

Quit

Help

hidden

# isgri\_ima parameters

The screenshot shows the 'ISGRI IMA' software window. The title bar reads 'ISGRI IMA'. The main area is titled 'ISGRI IMAGING'. On the right side, there are 'Ok' and 'Help' buttons. The parameters are organized into several sections:

- Energy Boundaries:** Includes 'IBIS\_II\_ChanNum' (4), 'IBIS\_II\_E\_band\_min' (20 40 60 100), 'IBIS\_II\_E\_band\_max' (40 60 100 200), and 'IBIS\_II\_inEnergyValues' (empty) with a 'browse' button.
- Source Search:** Includes 'OBS1\_SearchMode' (2), 'OBS1\_ToSearch' (15), 'OBS1\_MinCatSouSnr' (6), and 'OBS1\_MinNewSouSnr' (7).
- Mosaic:** Includes 'OBS1\_DuParl2' (1).
- Source Position:** Includes 'OBS1\_PixSpread' (1) and 'OBS1\_SouFit' (0).
- Background map:** Includes 'SCW1\_BKG\_I\_isgrBkgDol' (empty) with a 'browse' button.

# Inside isgri\_sky\_ima.fits file

fv: Binary Table of isgri\_sky\_ima.fits[1] in /scratch/chernyak/OSA4.2

File Edit Tools Help

IMATYPE       E\_MIN       E\_MAX  
 32A                      1E                      1E  
                                  keV                      keV

Select  
 All

Invert

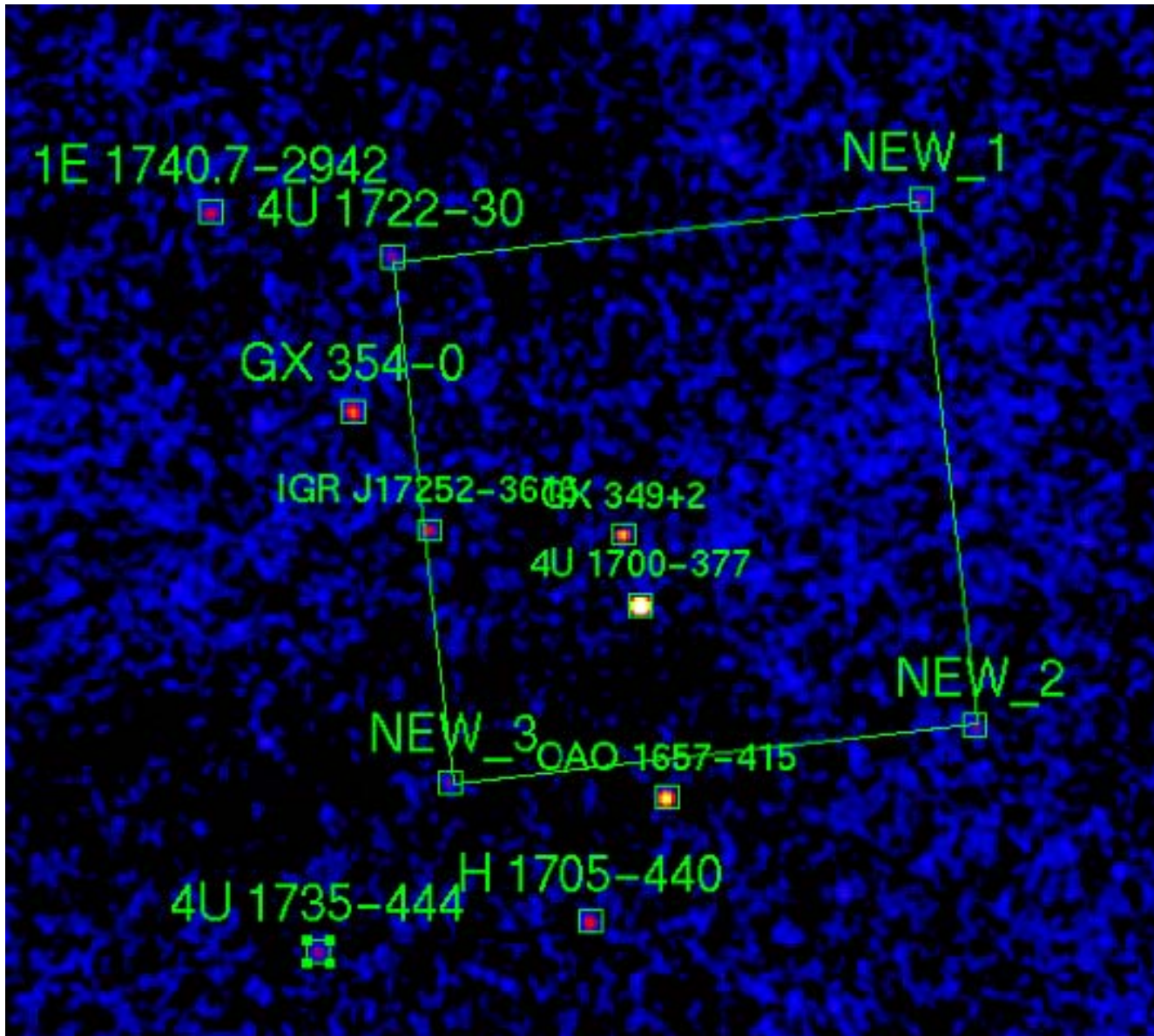
|    |                                     |              |              |              |
|----|-------------------------------------|--------------|--------------|--------------|
| 1  | <input type="button" value="Open"/> | INTENSITY    | 2.000000E+01 | 4.000000E+01 |
| 2  | <input type="button" value="Open"/> | VARIANCE     | 2.000000E+01 | 4.000000E+01 |
| 3  | <input type="button" value="Open"/> | SIGNIFICANCE | 2.000000E+01 | 4.000000E+01 |
| 4  | <input type="button" value="Open"/> | RESIDUAL     | 2.000000E+01 | 4.000000E+01 |
| 5  | <input type="button" value="Open"/> | INTENSITY    | 4.000000E+01 | 6.000000E+01 |
| 6  | <input type="button" value="Open"/> | VARIANCE     | 4.000000E+01 | 6.000000E+01 |
| 7  | <input type="button" value="Open"/> | SIGNIFICANCE | 4.000000E+01 | 6.000000E+01 |
| 8  | <input type="button" value="Open"/> | RESIDUAL     | 4.000000E+01 | 6.000000E+01 |
| 9  | <input type="button" value="Open"/> | INTENSITY    | 6.000000E+01 | 1.000000E+02 |
| 10 | <input type="button" value="Open"/> | VARIANCE     | 6.000000E+01 | 1.000000E+02 |
| 11 | <input type="button" value="Open"/> | SIGNIFICANCE | 6.000000E+01 | 1.000000E+02 |
| 12 | <input type="button" value="Open"/> | RESIDUAL     | 6.000000E+01 | 1.000000E+02 |
| 13 | <input type="button" value="Open"/> | INTENSITY    | 1.000000E+02 | 2.000000E+02 |
| 14 | <input type="button" value="Open"/> | VARIANCE     | 1.000000E+02 | 2.000000E+02 |
| 15 | <input type="button" value="Open"/> | SIGNIFICANCE | 1.000000E+02 | 2.000000E+02 |
| 16 | <input type="button" value="Open"/> | RESIDUAL     | 1.000000E+02 | 2.000000E+02 |

Go to:  Edit cell:





# Let's have a closer look...





*INTEGRAL Science Seminars in Kiev*

**THANKS FOR ATTENTION!**