

Dataset Descriptor, JAIL, Workflows

Tomáš Faragó

Institute For Synchrotron Radiation

Why?

- Currently, datasets are made of number of files which contain images
- For further processing we need to know what do these images represent (dark fields, flat fields, radiograms, ...)
- To store other metadata as well (at least pixel size and angles for tomographic reconstruction)
- To provide minimal addition to existing datasets (no need to convert datasets to other file formats, one just needs to create the descriptor file)
- To have something what can be easily used for describing other dataset types (basically any dataset composed of images)
- Trivial file reading (just use the readers which already exist to read actual pixel values)

Structure

- Similar to Windows .ini files
- Four sections
 - Dataset – basic information about dataset
 - Dimensions
 - pixel size
 - rotation axis
 - Layout
 - string describing the ordering of files on disk (allows to distinguish between different images, e.g. flat, dark, radio, etc.)
 - Specifies in which images are radiograms
 - Contains paths to files with images
 - Static attributes
 - Attributes for the whole dataset (motor names, energy, etc.)
 - Radio attributes
 - Attributes that change for each radiogram (e.g. angle of rotation)
- Options are defined as key=value pairs
- Full specification available at
http://www.ufo.kit.edu/ufo/browser/farago/dsdcreator/doc/descriptor_spec.txt

Example

```
[dataset]
dims = x:1024,y:768,omega:2000
img_dims = x,y
rotation = x
Pix_size = 0.36,0.36

[layout]
layout = (<flat>){10}(<radio>){2000}(<flat>){10}<dark><dark><dark>
dataset_data = radio
dirs = flats1/flat_,radios/radio_,flats2/flat_,darks/dark_

[static_attributes]
motor_names = a,b,c,d
energy = 9

[radio_attributes]
omega = 0,180 #automatically creates a range
timestamp = 1,2,3,4,...,2000 #not a dimension dependent attribute
```

Creation tools

- Python function
(for programmers)
- GUI in PyQt
(for scientists)
- Source code
in repository

Dataset descriptor creator

Directory with dataset: P:\folders\Datasets\DataSet

Prefixes: flats\radio_
radios\radio_

File format: tif

Data type:

Dimensions: x:4008,y:4008,omega:10

Rotation axis: x

Pixel size: x: 0,800 y: 0,800

Layout string: <flat><flat><radio>{10}

Sample name: radio

Dataset attributes

Key	Value
energy	20
date	23.3.2011
sample_name	bug

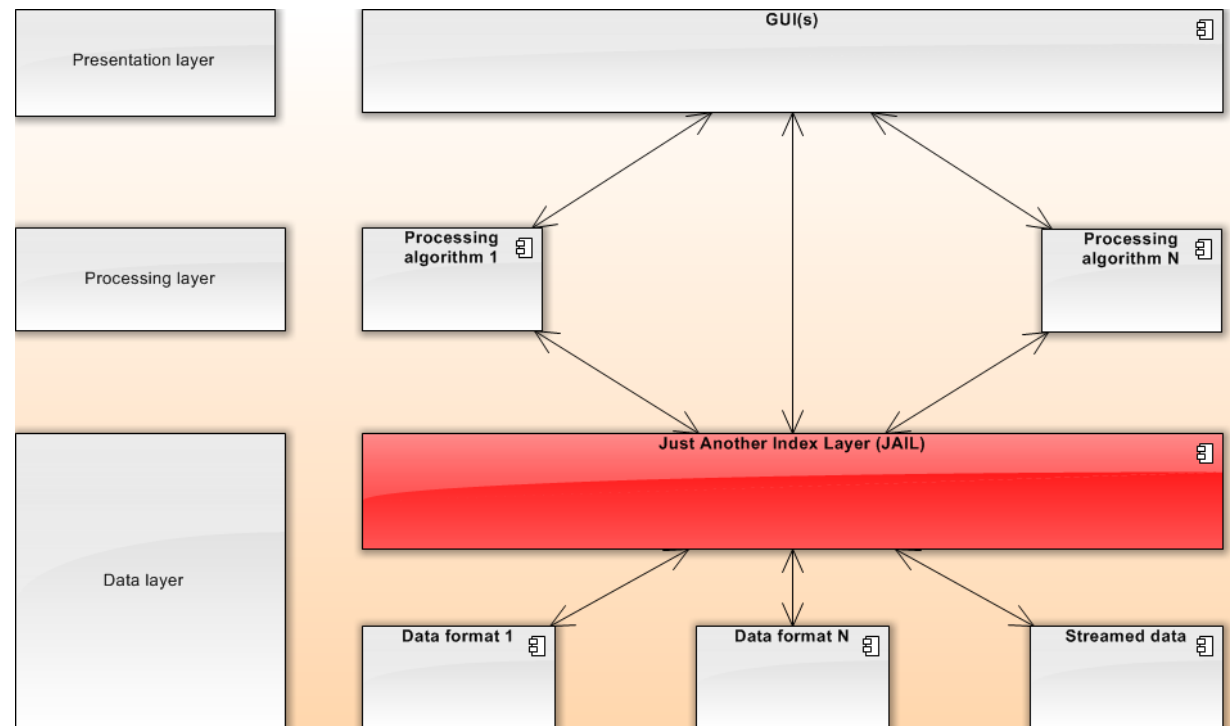
Sample attributes

Key	Value
omega	0,180

Buttons: Select directory, Add prefix, Edit prefix..., Delete prefix, Delete all, Set dimensions..., OK, Add attribute..., Edit attribute..., Delete attribute, Delete all, Add attributes..., Edit attribute..., Delete attribute, Delete all, OK, Cancel

JAIL

- Just Another Index Layer
- File format or even data source in general (streaming) should be transparent for processing layer
- The idea is to provide minimal API for obtaining data
- Processing algorithms do not have to care about data formats then
- File format changes -> reading needs to be implemented but interface stays the same!



Workflow example for programmer (Python)

```
import jail
import processing
import pyhst
import visualization

dataset = jail.FoldersDataSetReader(dsd_filepath)
darks = dataset.get_images("dark")
flats = dataset.get_images("flat")
radios = dataset.get_images("radios")

adj = processing.flat_correction(darks, flats, radios)
reconstructed = pyhst.reconstruct(adj)
visualization.show(reconstructed)
```

Workflow example for scientist

