Jungfrau naming and calibration files

Content

Intro

This page describes dynamic generation of gain correction calibration constants for multi-panel jungfrau detector. This is relevant to calibration types `pixel_gain` and `pixel_offset`, e.g. for single-panel detector

- `pixel_gain`: shape:(3, 1, 512, 1024) size:1572864 dtype:float32
- `pixel_offset`: shape:(3, 1, 512, 1024) size:1572864 dtype:float32

which are combined from 3 gain range per panel arrays of shape (512, 1024).

Naming conventions

Since psana.Jungfrau.ConfigV3 software provides per-segment access to the class ModuleConfigV1 object with three methods

- moduleVersion()
- firmwareVersion()
- serialNumber()

which return 64-bit integer numbers like 1511699, 23403815178277, 3997947, respectively. However, some of these numbers in hexadecimal format looks as dates and times. For better readability the panel Id name can be formatted as '%s-%s-%s', for hexadecimal values with dropped '0x' as hex(v).lsplit ('0x'), e.g. "171113-154920171025-3d00fb".

Then, total Jungfrau Id is a variable-length underscore-separated concatenation of sequential panel Id names, i.g. for two panels of exp=xcsls3716: run=631, XcsEndstation.0:Jungfrau.0 complete detector Id is 170505-149520170815-3d00b0_170505-149520170815-3d00f7.

API

Method returning Jungfrau and its segment Ids is implemented in `Detector/src/UtilsJungfrau.py` and can be used as in example:

```
import psana
from Detector.UtilsJungfrau import id_jungfrau

ds  = psana.DataSource('xpptut15:run=430')
env = ds.env()
src = 'MfxEndstation.0:Jungfrau.0' # or part of the DetInfo name 'Jung' or 'Jungfrau.1' or alias 'jungfrau1M'

print 'id_jungfrau(env, src, 0):', id_jungfrau(env, src, iseg=0) # for segment 0
print 'id_jungfrau(env, src, 1):', id_jungfrau(env, src, iseg=1) # for segment 1
print 'id_jungfrau(env, src)   :', id_jungfrau(env, src)         # for all segments
```

which prints

```
id_jungfrau(env, src, 0): 171113-154920171025-3d00b0
id_jungfrau(env, src, 1): 171113-154920171025-3d00f7
id_jungfrau(env, src)   : 171113-154920171025-3d00b0_171113-154920171025-3d00f7
```

CLI

A few commands which shed light on data with jungfrau detector are listed below.

Command line to get jungfrau psana.EventKey objects:

```
event_keys -d exp=xpptut15:run=430 -m2 -p Jung
```

Command line to find experiment runs with jungfrau:

```
find_detector_runs MFX jungfrau
```

Command line to get jungfrau Id (in release>ana-1.3.51):

```
jungfrau_id     <=== prints help
jungfrau_id exp=xpptut15:run=430 [Jungfrau.0]
```

Available data

Summary of experiments/runs with jungfrau data available on 2018-04-19 is listed below.

Datasets with jungfrau data and detector Id

```
Dataset                 Runs     DetInfo name                   Panel/detector Id or configuration object
version
-----------------------------------------------------------------------------------------------------------------------
-------------
exp=mfx00616:run=8      1-8      MfxEndstation.0:Jungfrau.0     psana.Jungfrau.ConfigV1
exp=mfx05516:run=16     16       MfxEndstation.0:Jungfrau.0     171113-154920171025-3d00b0_171113-154920171025-
3d00f7
exp=mfxlr1716:run=295   1-295    MfxEndstation.0:Jungfrau.0     171113-154920171025-3d00b0_171113-154920171025-
3d00f7
exp=mfxlr5016:run=197   194-197  MfxEndstation.0:Jungfrau.0     171113-154920171025-3d00b0_171113-154920171025-
3d00f7
exp=mfxx20515:run=2     1-2      MfxEndstation.0:Jungfrau.0     170505-149520170815-3d00b0_170505-149520170815-
3d00f7
exp=mfx11116:run=697    563-697  MfxEndstation.0:Jungfrau.1     171113-154920171025-3d00fb
exp=mfxls0816:run=193   1-193    MfxEndstation.0:Jungfrau.1     171113-154920171025-3d00fb

exp=xcs11116:run=18     4-18     XcsEndstation.0:Jungfrau.0     psana.Jungfrau.ConfigV2
exp=xcsx22015:run=555   503-555  XcsEndstation.0:Jungfrau.0     psana.Jungfrau.ConfigV2
exp=xcsls3716:run=631   1-631    XcsEndstation.0:Jungfrau.0     170505-149520170815-3d00b0_170505-149520170815-
3d00f7
exp=xcslr6316:run=17    4-17     XcsEndstation.0:Jungfrau.1     171113-154920171025-3d00fb

exp=cxi11216:run=54     4-54     CxiEndstation.0:Jungfrau.0     psana.Jungfrau.ConfigV1
exp=detdaq17:run=68     3-68     DetLab.0:Jungfrau.0            171113-154920171025-3d00fb
exp=xppls2416:run=247   1-247    XppEndstation.0:Jungfrau.1     171113-154920171025-3d00b0_171113-154920171025-
3d00f7
exp=mecls3216:run=226   2-226    MecTargetChamber.0:Jungfrau.0  170505-149520170815-3d00b0_170505-149520170815-
3d00f7
```

Panel calibration arrays

For now, calibration arrays for `offset` and `gain` is produced per panel named as M044, M068, M088, etc., e.g. directory /reg/g/psdm/detector/gains /jungfrau/M088/ contains:

```
g0_gain.npy
g0_offset.npy
g1_gain.npy
g1_offset.npy
g2_gain.npy
g2_offset.npy
```

Content of calibration directory for panels

For auto-generation of calibration files it is suggested to

- keep panel calibration files under /reg/g/psdm/detector/gains/jungfrau/M044, M068, M088, etc.
- use panel Id with timestamp (version) as a reference to one of above directories
- in M044, M068, M088, etc. keep the same naming conventions for gain range and type (g1_gain.npy, g1_offset.npy, g2_gain.npy, etc.).

To distinguish calibration versions the time-stamp in format YYYYMMDDHHMMSS is dash-concatenated at the end of the panel Id name. This time stamp stands for start time of the constants validity range.

Assignment of panel ids to directories with calibration files

```
Panel Ids in hexadecimal format
171113-154920171025-3d00b0   <- this may be reference to one of the M0## directories
171113-154920171025-3d00f7
170505-149520170815-3d00b0
170505-149520170815-3d00f7
171113-154920171025-3d00fb

Philip's assignment:
M044:    for exp=mfx11116:run=689 171113-154920171025-3d00fb-20180306000000

M088,M068: exp=xcsx22015:run=513, psana.Jungfrau.ConfigV2, XcsEndstation.0:Jungfrau.0       run 503 on Oct 25,
2017
Later experiment: exp=xcsls3716:run=631, id=170505-149520170815-3d00b0_170505-149520170815-3d00f7   run 1 on
Dec 5, 2017 16:14
M068:    170505-149520170815-3d00b0-20171025000000
M088:    170505-149520170815-3d00f7-20171025000000
```

Assignment of panel ids to directories with calibration files can be listed by the command:

```
ls -l /reg/g/psdm/detector/gains/jungfrau
```

```
lrwxrwxrwx 1 dubrovin ps-data   4 Apr 30 11:56 170505-149520170815-3d00b0-20171025000000 -> M068
lrwxrwxrwx 1 dubrovin ps-data   4 Apr 30 11:56 170505-149520170815-3d00f7-20171025000000 -> M088
drwxr-sr-x 2 dubrovin ps-data   2 Apr 30 12:01 171113-154920171025-3d00b0 -> M068
drwxr-sr-x 2 dubrovin ps-data   2 Apr 30 12:01 171113-154920171025-3d00f7 -> M088
lrwxrwxrwx 1 dubrovin ps-data   4 Apr 30 11:55 171113-154920171025-3d00fb-20180306000000 -> M044
drwxr-sr-x 2 dubrovin ps-data  12 Mar 27 12:11 M044
drwxr-sr-x 2 dubrovin ps-data  11 Mar 27 12:10 M068
drwxr-sr-x 2 dubrovin ps-data  11 Mar 27 12:10 M088
```

At these conventions calibration file for particular detector can be generated dynamically using panel Ids from configuration data. Thus generated file can be deployed in the calib or "generic" repository.

CLI for calibration

Two commands are available in releases >ana-1.3.51

Command line to get jungfrau Id:

```
jungfrau_id     <=== prints help
jungfrau_id exp=xpptut15:run=430 [Jungfrau.0]
```

This command dumps full name of the detector found in configuration object, number of panels and detector Id:

result of jungfrau_id

```
Finding Jungfrau Id in dataset "exp=xpptut15:run=430" using name pattern Jungfrau
Found source: MfxEndstation.0-Jungfrau.0, number of panels: 2
Jungfrau id: 171113-154920171025-3d00b0_171113-154920171025-3d00f7
```

Command line to generate and deploy gain correction constants (pixel_gain and pixel_offset) for specified experiment and detector:

```
jungfrau_gain_constants [-h]   <=== prints help
jungfrau_gain_constants -d exp=xcsls3716:run=631 -D [-s Jungfrau.0] <=== generate and deploy constants
```

This command

- retrieves detector Id, defines panel Ids
- retrieves gain correction files for panel Ids from specialized repository `/reg/g/psdm/detector/gains/jungfrau`
- merge them in arrays shaped as `(3,<number-of-pannels>,512,1024)` for entire detector
- saves array in files in local directory with names like `xcsls3716-r0631_XcsEndstation.0:Jungfrau.0_170505-149520170815-3d00b0_170505-149520170815-3d00f7_pixel_gain.txt`
- and deploy them under `/reg/d/psdm/XCS/xcsls3716/calib/Jungfrau::CalibV1/XcsEndstation.0:Jungfrau.0` for calibration types `pixel_gain` and `pixel_offset` with name `631-end.data` for validity range beginning from run specified in dataset description.

Maintenance of calibration constants

Panel-calibration files

It is assumed that panel calibration constants for `pixel_gain` and `pixel_offset` **are supplied by the detector group** for all known panels. These constants should be available under directory `/reg/g/psdm/detector/gains/jungfrau/` in files like

`/reg/g/psdm/detector/gains/jungfrau/<panel-id>[-<time-stamp>]/g<index-of-gain-range>-<gain-or-offset>.npy.`

Generator of constants

Example of the command to produce calibration constants for entire detector:

`jungfrau_gain_constants -d exp=xcsls3716:run=631 [-D] [-U] [-s Jungfrau.0]`

- constants are saved in local directory in files with long name containing fields of experiment, run, source, detector Id, constant type and extension txt or npy,
- option -d specifies dataset containing jungfrau data and configuration to define detector numeric Id, e.g. 170505-149520170815-3d00b0_170505-149520170815-3d00f7,
- option `-D` deploys calibration constants in the calib directory for specified experiment and beginning of the validity range defined by the specified run number,
- option `-U` shows commands (`dcs add ...`) to upload constants from local directory to global repository under `/reg/d/psdm/detector/calib/jungfrau,`
- option `-s` is required to distinguish between multiple detectors if specified data set has more than one jungfrau detector.

Upload constants to repository

Use command `jungfrau_gain_constants` with option `-U` to see commands for deployment (`dcs add ...`) . Then, apply these commands to add calibration constants to the detector associated hdf5 file located under directory

- `/reg/d/psdm/detector/calib/jungfrau/` - real repository or
- `/reg/g/psdm/detector/calib/jungfrau/` - test repository

Questions

- format of integers in the panel name fields: hex(v).lsplit('0x'), e.g. "171113-154920171025-3d00fb"
- calibration versions: use time-stamp in format YYYYMMDDHHMMSS dash-concatenated to the panel Id, e.g. "171113-154920171025-3d00fb-2018030600000"

References

- Jungfrau