SSRL/LCLS Annual Users' Meeting & Workshop, Oct 8th, 2014

Detector calibration and metrology

How to manage the calibration steps, how to use the detector geometry during analysis

Mikhail Dubrovin

SLAC

Outline

- Calibration store
- Calibration Manager
- Intensity correction
- Detector geometry

Calibration store

File system-based calibration store:

```
/reg/d/psdm/INS/<exp>/<ftype> /<calib-version>  /<data source>      /<type>   /<file>
/reg/d/psdm/CXI/cxi12345/xtc
                         calib/CsPad::CalibV1   /CxiDs1.0:Cspad.0    /geometry /0-end.data
                               CsPad2x2::CalibV1/XppGon.0:Cspad2x2.0/center    /10-end.data
                               CsPad::CalibV1   /XppGon.0:Cspad.0    /pedestals/5-8.data
```

+ simple structure, no special tool needed to browse/edit/copy text files
− does not support file versions (but you may use local version of calib-dir), no bookkeeping of modifications, hard to type correctly all fields in the path
± easy add/remove files, made/correct typos in manual operations

Old style calibration file producers:

- psana modules to produce calibration types *pedestals, pixel_rms, pixel_status*
  - cspad_mod::CsPadPedestals
  - ImgAlgos::NDArrCalib
- python scripts for metrology file processing of CSPAD with moving/fixed quads, 2x2 to produce *center, tilts*
- python scripts for image-based geometry alignment to produce all other calibration files with geometry parameters

Needed in a tool which simplify manipulation with all these scripts and files

Calibration manager

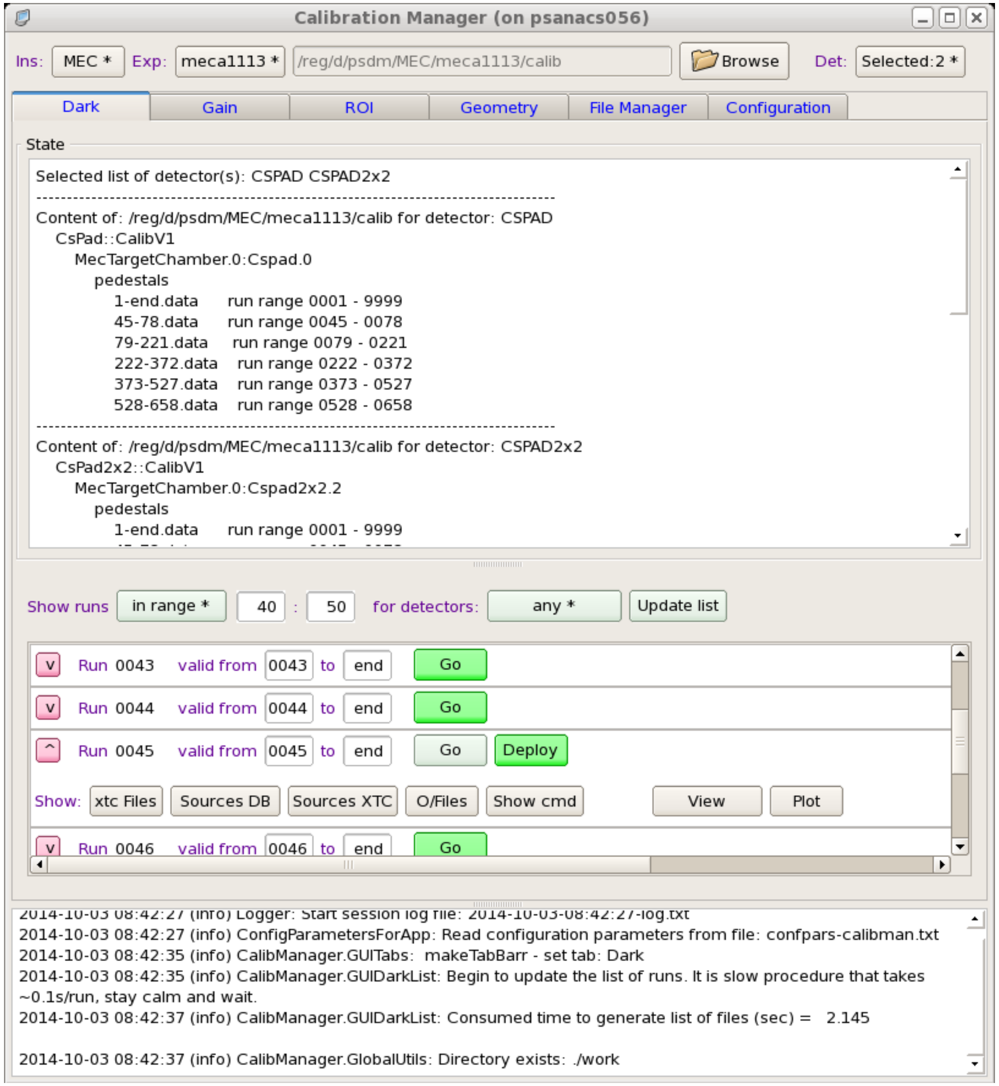
Calibration Manager GUI components

- Destination for calib files
- Tool selection tabs
- Status panel
- Functional panel
- Output information panel

Implemented tools

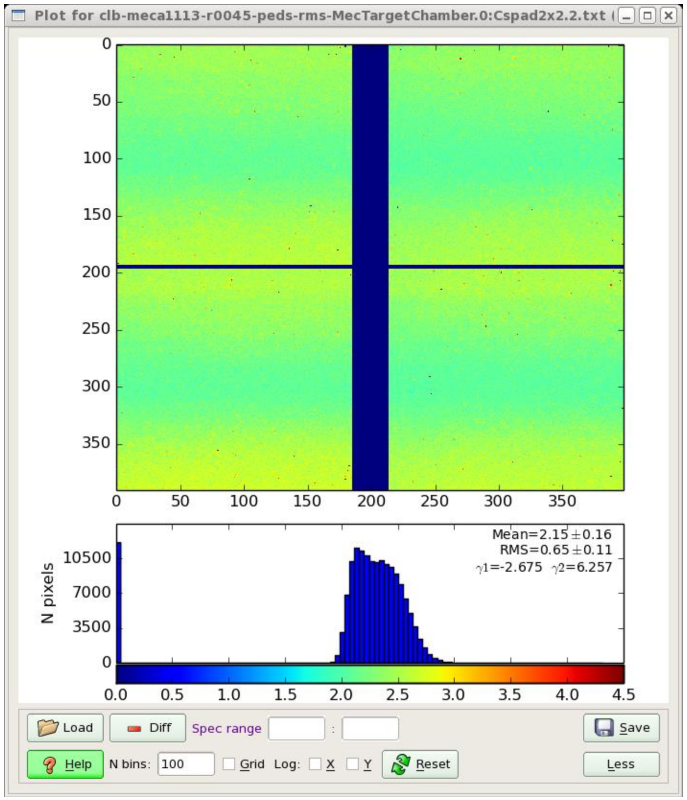
- Dark run processing
- ROI Mask Editor
- Geometry – metrology processing
- File manager for single and group
- Configuration parameters setup

- Can be extended for other apps

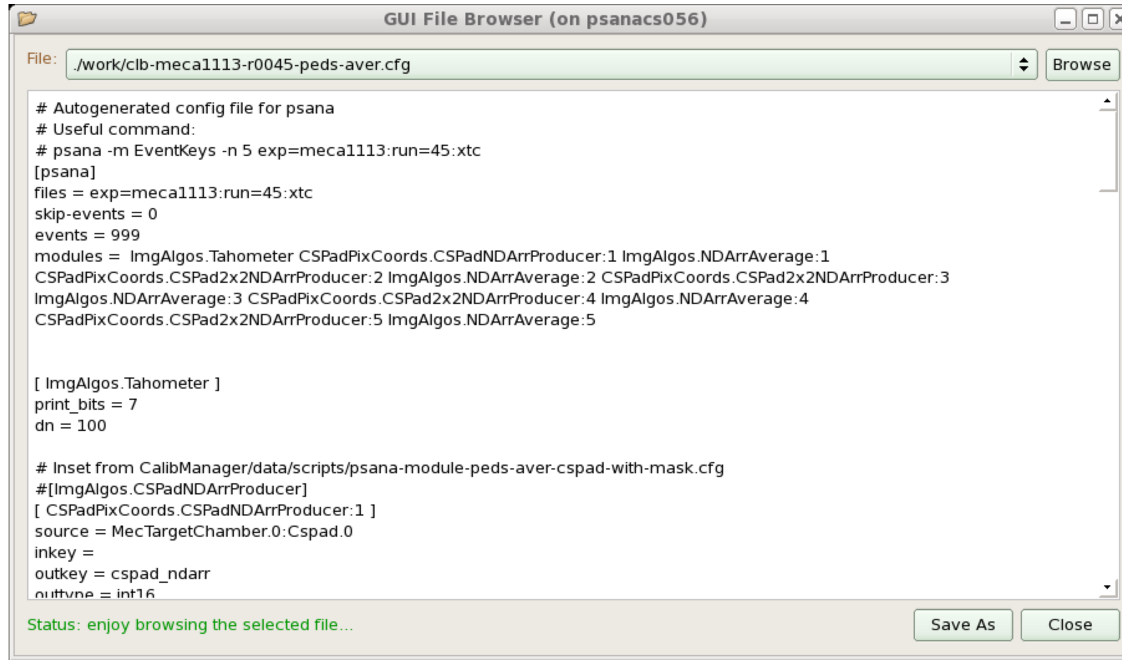


Calibration Manager Service Tools

Plot buttons open image viewer



View buttons open text file viewer



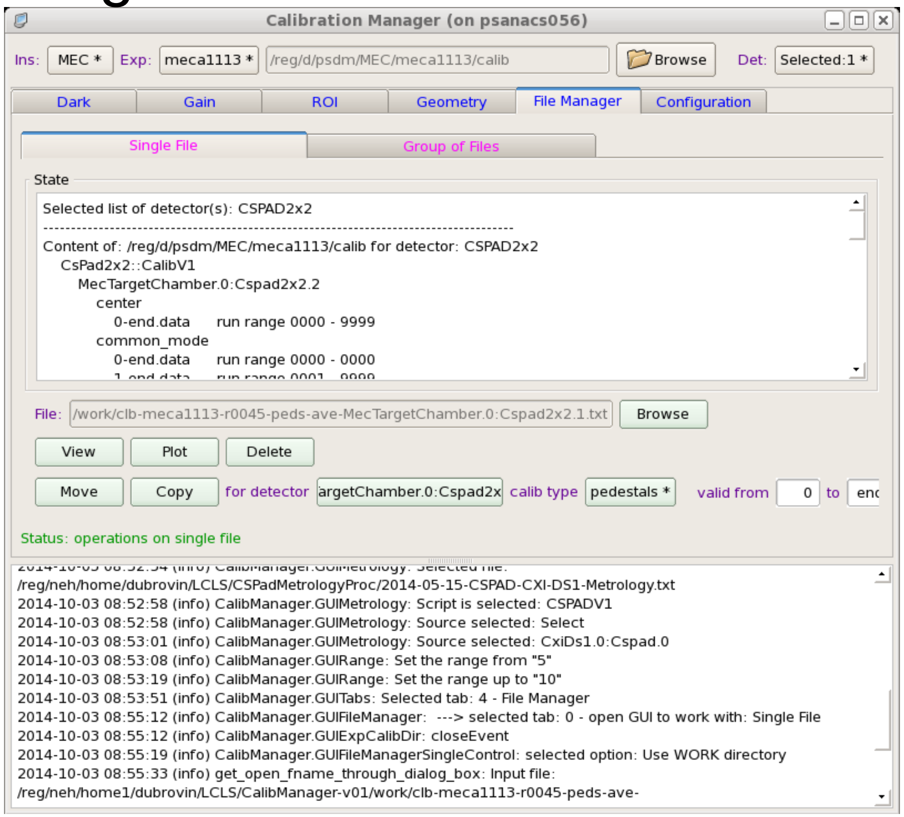
% plims <file-name>

- command launches standalone application

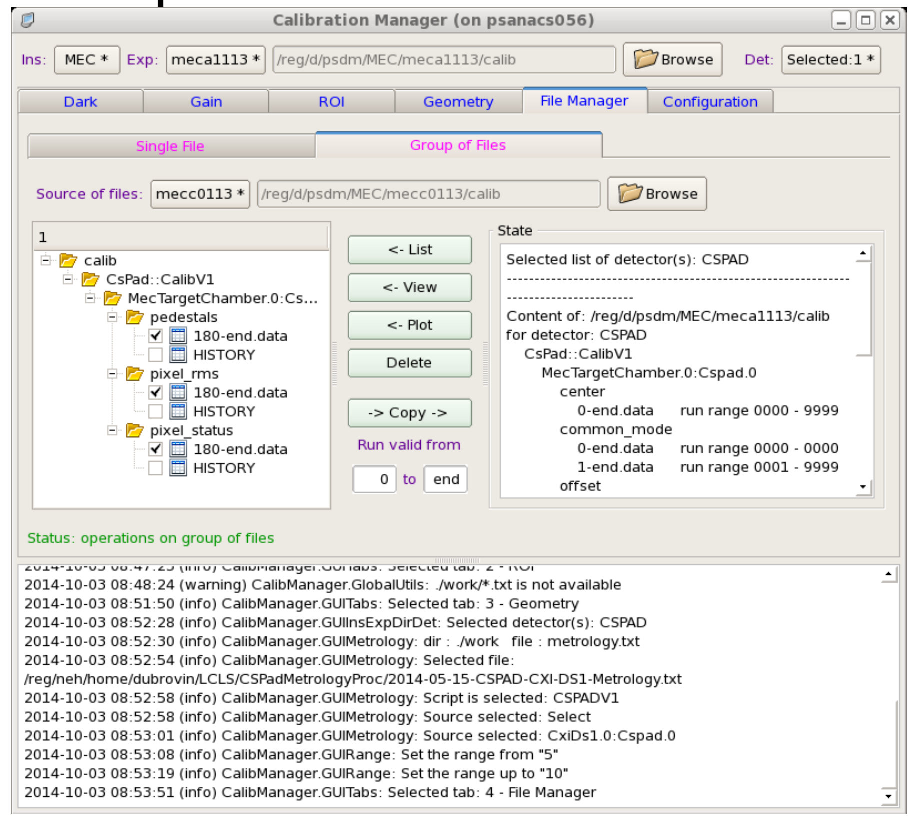
Calibration Manager
File Management Tools

SLAC

Single file



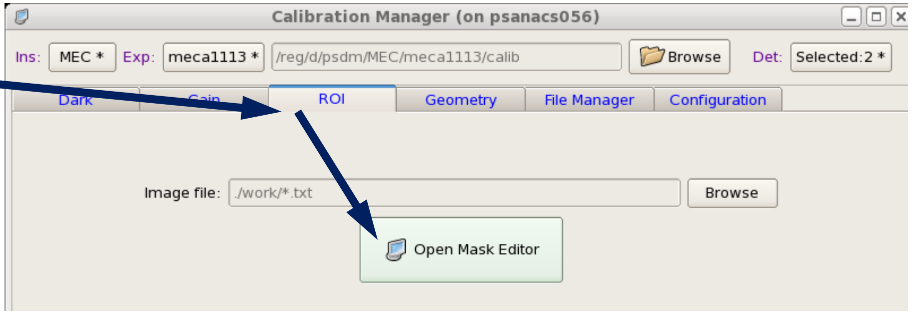
Group of files

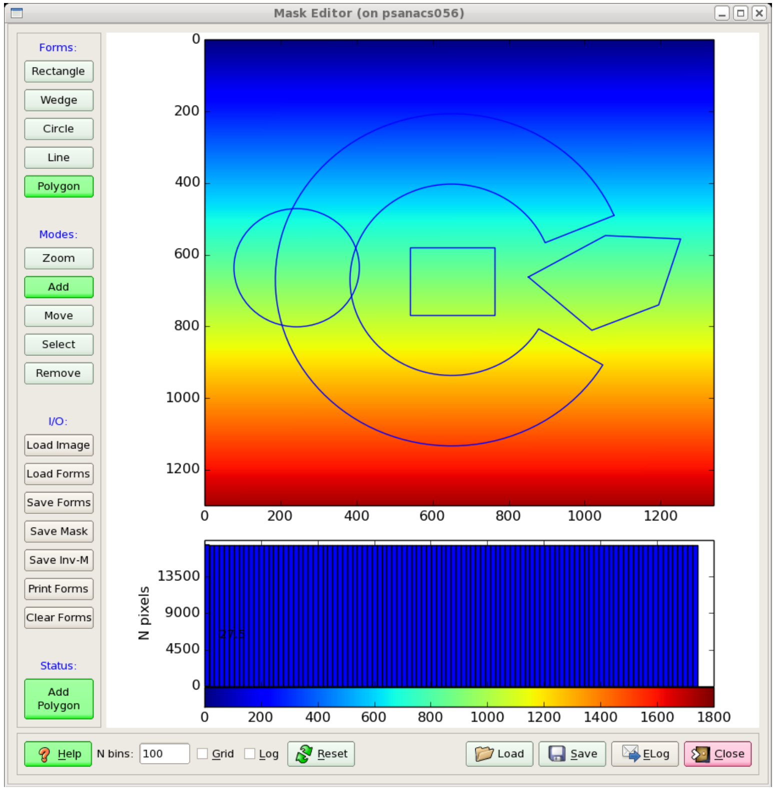


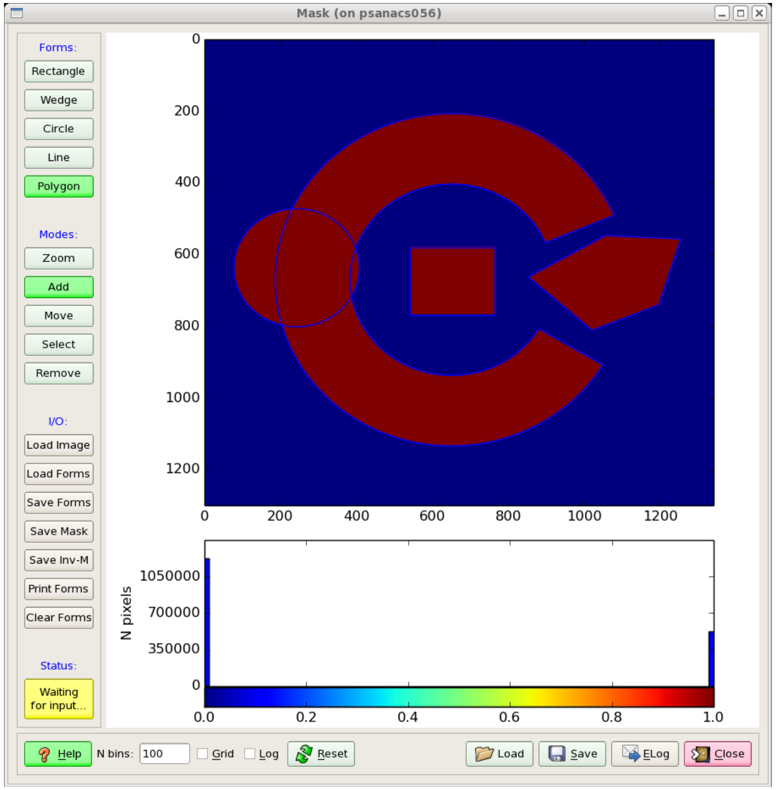
Calibration Manager
ROI Mask Editor

Can be launched from *calibman* or

% med - command to run in standalone mode



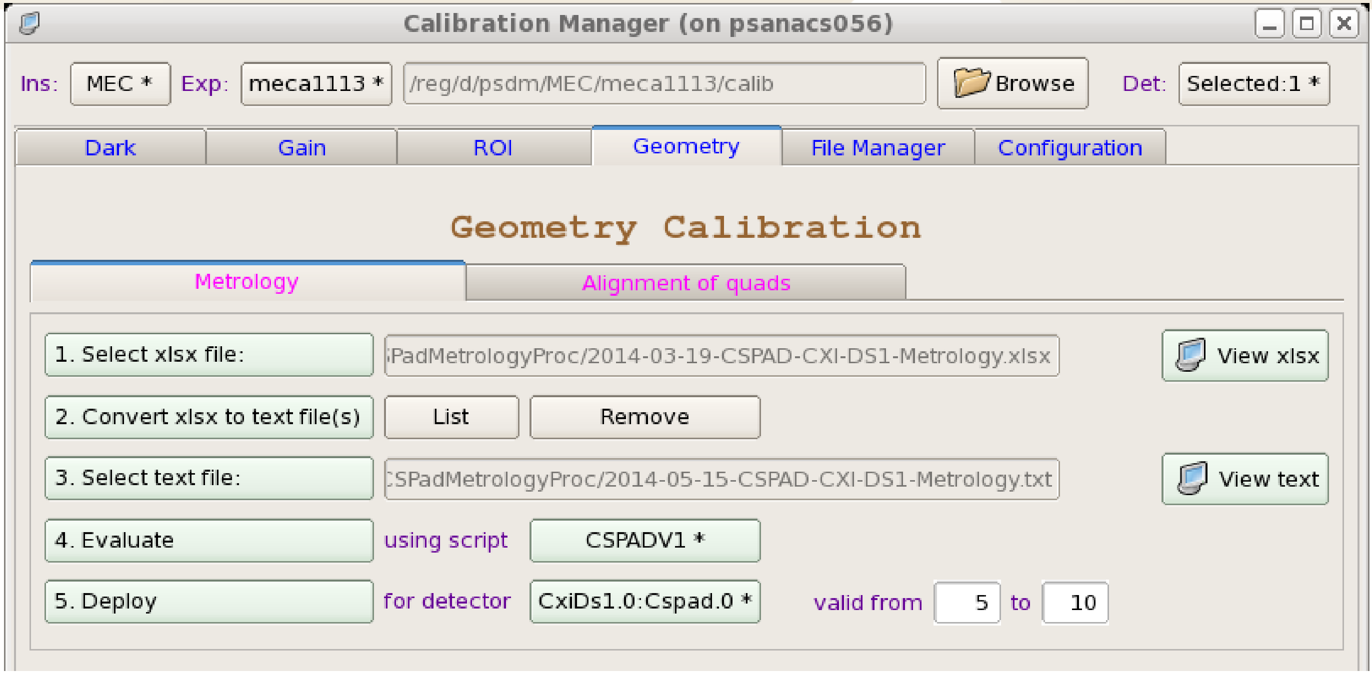




Calibration Manager
Geometry – Metrology Processing

Geometry panel sub-tabs:

- **Metrology** – expert level tool - do not use it without complete understanding of what you are doing and how to check results
- **Alignment of quads** – is not implemented yet



How to apply intensity correction

1. Make sure that necessary calibration files are available in the default or your custom **calib** directory and path to the calibration files has correct structure.

2. In psana config. file use modules <Detector>ImageProducer or <Detector>NDArrProducer to save detector data in the event store as ndarray.
   - Psana - Module Catalog
     https://confluence.slac.stanford.edu/display/PSDM/psana+-+Module+Catalog
   - Psana - Module Examples
     https://confluence.slac.stanford.edu/display/PSDM/psana+-+Module+Examples

3. Use psana modules ImgAlgos::NDArrCalib or cspad_mod.CsPadCalib to get calibrated ndarray and save it the event store.

4. Use calibrated ndarrays in other psana modules or in the python script.

Detector geometry

Old version of detector geometry is available for a couple of years.

Drawbacks of old approach:

- Detector-dependent implementation: was implemented for CSPAD (CXI, XPP, 2x2) only.
- Uses a large number of calibration files (*center, center_global, tilt, offset, offset_corr, tilt, quad_tilt, rotation, quad_rotation, marg_gap_shift*) which meaning is hard to explain.

New detector geometry model:

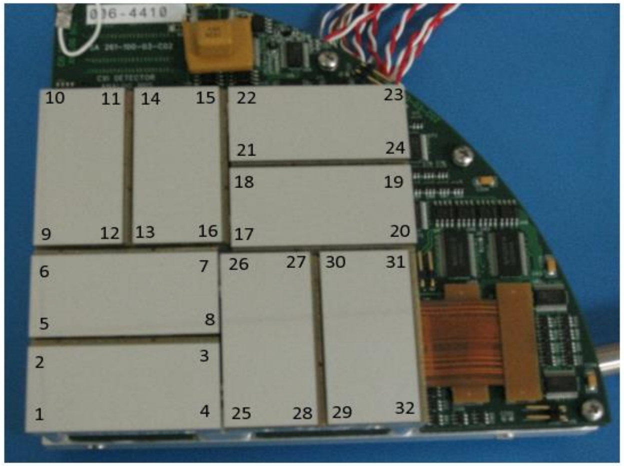
- Almost detector-independent implementation;
  need to implement sensors’ geometry only.
- Single file contains entire geometry information with clean meaning of all parameters, directly based on optical measurements.

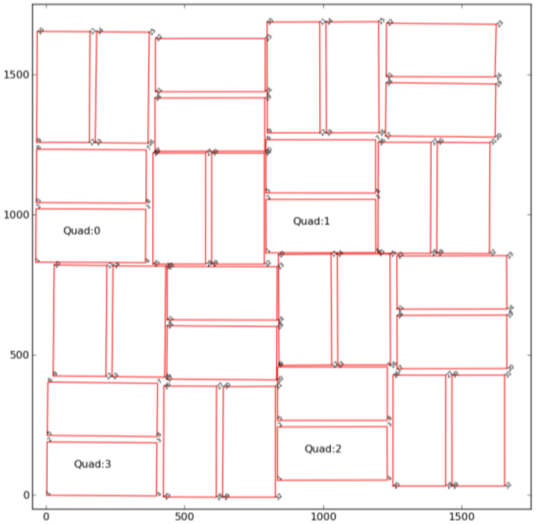
Optical measurements

Optical measurements → table of 3D coordinates of 4 corners for each 2×1 sensor

- in entire XPP-style CSPAD or CSPAD2x2
- in quads of CXI-style CSPAD

Map of Measuring Points





xlsx file → text file → quality check → if necessary, apply corrections → evaluate sensor center coordinates and tilts → save file with geometry parameters

Geometry file format - header

Header consists of a set of comments formatted for dictionary as

<keyword> <string-of-comment>

Hash in the 1st position, keyword is separated by space, empty lines are ignored

```
# TITLE      Geometry parameters of CSPAD-CXI
# DATE_TIME  2014-10-03 12:20:44 PDT
# METROLOGY  /reg/neh/home1/dubrovin/LCLS/CSPadMetrologyProc/2014-05-15-CSPAD-CXI-DS1-Metrology-corr.txt
# AUTHOR     dubrovin
# EXPERIMENT Any
# DETECTOR   CSPAD-CXI
# CALIB_TYPE geometry
# COMMENT:01 Table contains the list of geometry parameters for alignment of 2x1 sensors, quads, CSPAD, etc
# COMMENT:02 All translation and rotation pars of the object are defined w.r.t. parent object Cartesian frame
# PARAM:01 PARENT     - name and version of the parent object
# PARAM:02 PARENT_IND - index of the parent object
# PARAM:03 OBJECT     - name and version of the object
# PARAM:04 OBJECT_IND - index of the new object
# PARAM:05 X0         - x-coordinate [um] of the object origin in the parent frame
# PARAM:06 Y0         - y-coordinate [um] of the object origin in the parent frame
# PARAM:07 Z0         - z-coordinate [um] of the object origin in the parent frame
# PARAM:08 ROT_Z      - object design rotation angle [deg] around Z axis of the parent frame
# PARAM:09 ROT_Y      - object design rotation angle [deg] around Y axis of the parent frame
# PARAM:10 ROT_X      - object design rotation angle [deg] around X axis of the parent frame
# PARAM:11 TILT_Z     - object tilt angle [deg] around Z axis of the parent frame
# PARAM:12 TILT_Y     - object tilt angle [deg] around Y axis of the parent frame
# PARAM:13 TILT_X     - object tilt angle [deg] around X axis of the parent frame

# HDR PARENT IND       OBJECT IND    X0[um]  Y0[um]  Z0[um]   ROT-Z ROT-Y ROT-X     TILT-Z   TILT-Y   TILT-X
```

Geometry file format - content

# HDR PARENT	IND	OBJECT	IND	X0[um]	Y0[um]	Z0[um]	ROT-Z	ROT-Y	ROT-X	TILT-Z	TILT-Y	TILT-X
QUAD:V1	0	SENS2X1:V1	0	21757	33110	51	0	0	0	0.04474	-0.14079	-0.00274
QUAD:V1	0	SENS2X1:V1	1	21769	10457	18	0	0	0	0.01053	-0.11974	0.00000
QUAD:V1	0	SENS2X1:V1	2	33464	68275	-28	270	0	0	-0.01645	0.10414	0.09737
QUAD:V1	0	SENS2X1:V1	3	10769	68299	18	270	0	0	-0.02828	0.02740	0.13418
QUAD:V1	0	SENS2X1:V1	4	68489	56732	71	180	0	0	-0.05128	-0.11309	0.06303
QUAD:V1	0	SENS2X1:V1	5	68561	79628	-20	180	0	0	-0.03552	0.07104	-0.11788
QUAD:V1	0	SENS2X1:V1	6	77637	21754	-15	270	0	0	-0.33657	-0.00821	0.01183
QUAD:V1	0	SENS2X1:V1	7	54810	21558	-54	270	0	0	-0.06315	0.00000	0.00658
QUAD:V1	1	SENS2X1:V1	0	21757	33329	178	0	0	0	0.08883	0.03158	-0.20830
QUAD:V1	1	SENS2X1:V1	1	21773	10446	61	0	0	0	-0.01448	0.04211	-0.24943
…												
QUAD:V1	2	SENS2X1:V1	6	77482	21698	0	270	0	0	-0.02762	0.00000	0.00000
QUAD:V1	2	SENS2X1:V1	7	54709	21779	0	270	0	0	0.02499	0.00000	0.00000
QUAD:V1	3	SENS2X1:V1	0	21730	33098	102	0	0	0	0.09278	-0.05132	-0.27140
QUAD:V1	3	SENS2X1:V1	1	21755	10477	40	0	0	0	0.06580	-0.02369	-0.32612
QUAD:V1	3	SENS2X1:V1	2	33193	68452	272	270	0	0	0.34083	-0.02192	-0.18687
QUAD:V1	3	SENS2X1:V1	3	10904	68416	270	270	0	0	0.01645	0.00823	-0.15397
QUAD:V1	3	SENS2X1:V1	4	68570	56923	194	180	0	0	0.12435	0.06974	0.07401
QUAD:V1	3	SENS2X1:V1	5	68456	79666	246	180	0	0	0.20857	0.04737	0.12882
QUAD:V1	3	SENS2X1:V1	6	77425	21681	60	270	0	0	0.05264	0.00274	-0.30004
QUAD:V1	3	SENS2X1:V1	7	54648	21761	118	270	0	0	0.01645	-0.00822	-0.22107
CSPAD:V1	0	QUAD:V1	0	-4500	-4500	0	90	0	0	0.00000	0.00000	0.00000
CSPAD:V1	0	QUAD:V1	1	-4500	4500	0	0	0	0	0.00000	0.00000	0.00000
CSPAD:V1	0	QUAD:V1	2	4500	4500	0	270	0	0	0.00000	0.00000	0.00000
CSPAD:V1	0	QUAD:V1	3	4500	-4500	0	180	0	0	0.00000	0.00000	0.00000

Each string contains the object parameters:

- **PARENT, IND** - parent ID
- **OBJECT, IND** - object ID
- **X0, Y0, Z0**[um] – object origin in the parent coordinate frame
- **ROT-X, ROT-Y, ROT-X,**
- **TILT-Z, TILT-Y, TILT-X** [deg] – object frame rotation (by design) and tilt angles in the parent coordinate frame
- Coordinate information comes directly from metrology file

Geometry file format – content exampes

CSPAD with fixed quads, XPP-like

# HDR PARENT	IND	OBJECT	IND	X0[um]	Y0[um]	Z0[um]	ROT-Z	ROT-Y	ROT-X	TILT-Z	TILT-Y	TILT-X
CSPAD:V2	0	SENS2X1:V1	0	51621	112683	153	90	0	0	0.48292	0.00000	0.00263
CSPAD:V2	0	SENS2X1:V1	1	74168	112907	135	90	0	0	0.48886	-0.05755	0.01316
CSPAD:V2	0	SENS2X1:V1	2	16366	124137	107	0	0	0	0.32370	0.00395	-0.01096
CSPAD:V2	0	SENS2X1:V1	3	16415	101350	185	0	0	0	0.15130	0.09867	-0.29055
CSPAD:V2	0	SENS2X1:V1	4	27786	159233	123	270	0	0	0.24474	-0.03015	0.03948
CSPAD:V2	0	SENS2X1:V1	5	4845	159260	106	270	0	0	0.05987	-0.04659	0.01053
CSPAD:V2	0	SENS2X1:V1	6	62758	168631	132	0	0	0	0.44940	0.02895	-0.00548
CSPAD:V2	0	SENS2X1:V1	7	62881	145851	137	0	0	0	0.20068	0.03553	-0.03015
CSPAD:V2	0	SENS2X1:V1	8	107098	129494	165	0	0	0	0.01316	0.01184	-0.02742
CSPAD:V2	0	SENS2X1:V1	9	107067	106699	152	0	0	0	-0.15924	-0.01316	-0.05208
CSPAD:V2	0	SENS2X1:V1	10	118871	164660	152	270	0	0	0.11249	-0.02193	0.04078
CSPAD:V2	0	SENS2X1:V1	11	95879	164632	143	270	0	0	-0.07632	-0.05757	0.03158
CSPAD:V2	0	SENS2X1:V1	12	153905	153051	158	180	0	0	-0.03026	0.00000	-0.04385
CSPAD:V2	0	SENS2X1:V1	13	153838	175841	140	180	0	0	-0.03947	-0.01973	-0.16719
CSPAD:V2	0	SENS2X1:V1	14	163149	117980	156	270	0	0	-0.17500	-0.04385	-0.00526
CSPAD:V2	0	SENS2X1:V1	15	140287	117936	153	270	0	0	0.00000	-0.02741	-0.00790
CSPAD:V2	0	SENS2X1:V1	16	123905	73726	133	270	0	0	0.00263	-0.00548	-0.00921
CSPAD:V2	0	SENS2X1:V1	17	101282	73699	130	270	0	0	0.20725	-0.02740	-0.03948
CSPAD:V2	0	SENS2X1:V1	18	159023	62203	142	180	0	0	-0.05461	-0.00263	-0.03562
CSPAD:V2	0	SENS2X1:V1	19	159024	84885	136	180	0	0	-0.08026	0.01579	-0.06576
CSPAD:V2	0	SENS2X1:V1	20	147998	26847	106	90	0	0	0.03158	-0.08219	0.08947
CSPAD:V2	0	SENS2X1:V1	21	170404	26856	96	90	0	0	-0.07893	-0.00822	0.06972
CSPAD:V2	0	SENS2X1:V1	22	112701	17748	97	180	0	0	0.26908	-0.02500	-0.00548
CSPAD:V2	0	SENS2X1:V1	23	112640	40586	100	180	0	0	0.03685	-0.01448	0.03836
CSPAD:V2	0	SENS2X1:V1	24	68448	57015	119	180	0	0	0.27835	-0.02501	0.09319
CSPAD:V2	0	SENS2X1:V1	25	68292	79701	120	180	0	0	0.37440	-0.03290	-0.04112
CSPAD:V2	0	SENS2X1:V1	26	56919	21965	69	90	0	0	0.18024	-0.06578	0.07894
CSPAD:V2	0	SENS2X1:V1	27	79497	22002	67	90	0	0	0.12040	-0.02740	0.09211
CSPAD:V2	0	SENS2X1:V1	28	21658	33119	72	0	0	0	0.27240	0.03553	-0.12332
CSPAD:V2	0	SENS2X1:V1	29	21748	10489	28	0	0	0	0.09805	0.03027	-0.07946
CSPAD:V2	0	SENS2X1:V1	30	12357	68156	97	90	0	0	0.20919	0.03562	0.05789
CSPAD:V2	0	SENS2X1:V1	31	35247	68250	89	90	0	0	0.09144	-0.04109	0.02368

CSPAD2x2

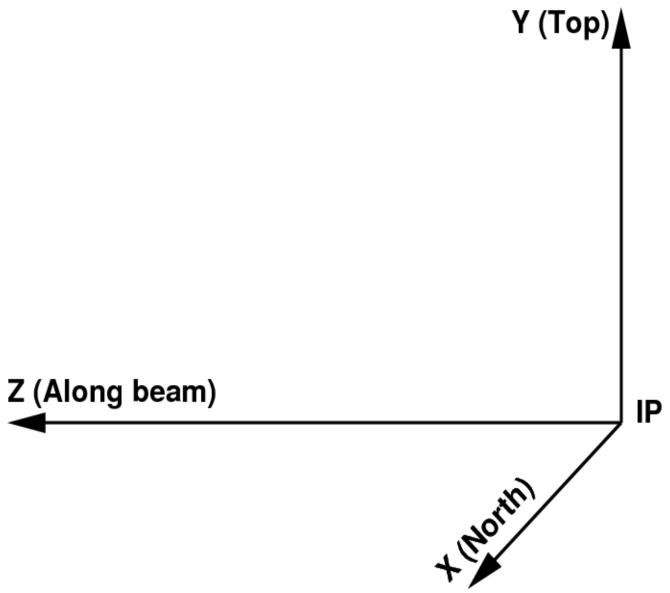
# HDR PARENT	IND	OBJECT	IND	X0[um]	Y0[um]	Z0[um]	ROT-Z	ROT-Y	ROT-X	TILT-Z	TILT-Y	TILT-X
CSPAD2X1:V1	0	SENS2X1:V1	0	21848	10490	6	180	0	0	-0.00197	-0.01049	0.03004
CSPAD2X1:V1	0	SENS2X1:V1	1	21943	33908	4	180	0	0	-0.16127	-0.00262	-0.01639

Set detector coordinates relative to IP

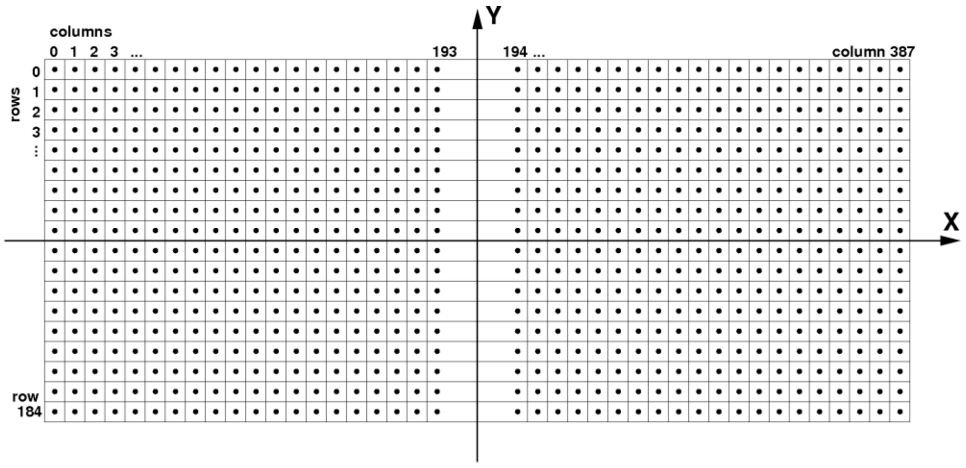
IP 0 CSPAD2X1:V1 0 0 0 1000000 0 0 180 0. 0. 0.

Basic geometry model elements

- Coordinate frame for setup

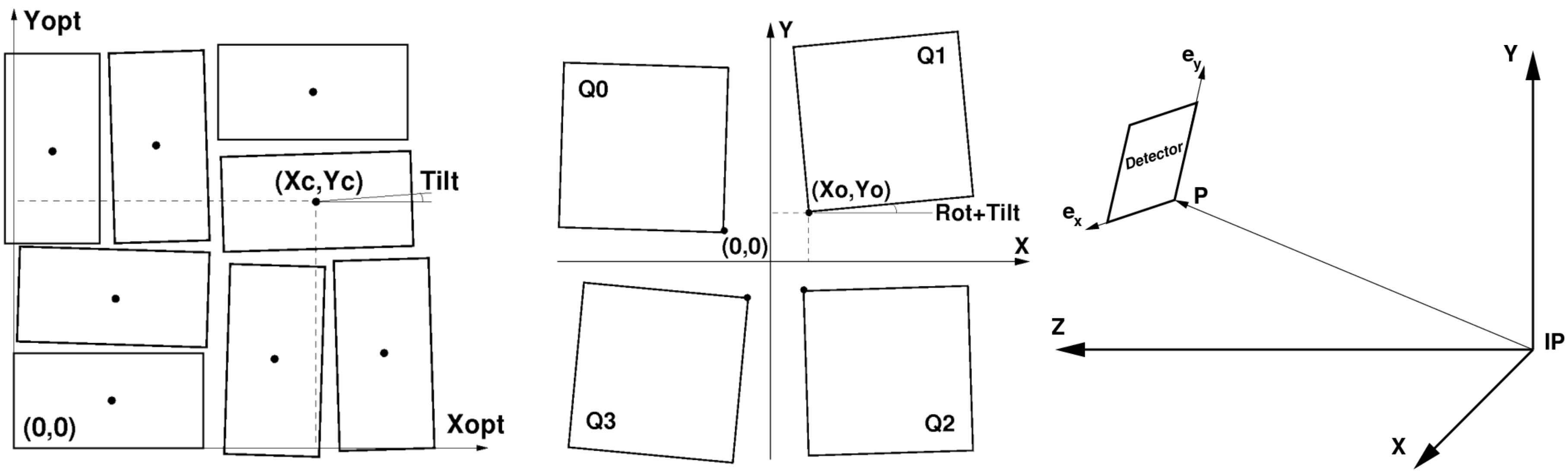


- Coordinate frame of sensor
  - for example, cspad 2x1 is a non-uniform matrix – needs in parameterization



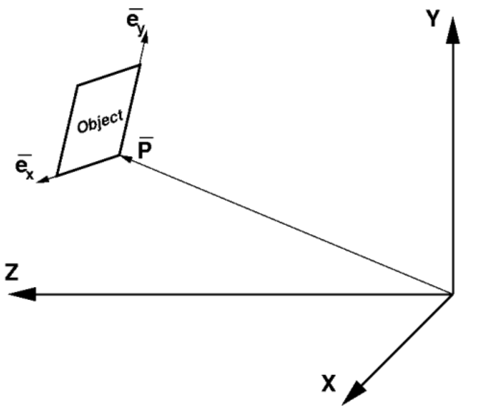
Detector composition example

2x1-sensors → Quads → CSPAD → Setup



Coordinate transformation

- Transformation of child object coordinates in the parent frame



$$C_i = R_{ij} \cdot c_j + P_i$$

$$R(\alpha, \beta, \gamma) = R_x(\alpha) \cdot R_y(\beta) \cdot R_z(\gamma)$$

Order of operations is defined by the code

```
def transform_geo_coord_arrays(self, X, Y, Z, do_tilt = True) :
        angle_z = self.rot_z + self.tilt_z if do_tilt else self.rot_z
        angle_y = self.rot_y + self.tilt_y if do_tilt else self.rot_y
        angle_x = self.rot_x + self.tilt_x if do_tilt else self.rot_x

        X1, Y1 = rotation(X,  Y,  angle_z)
        Z2, X2 = rotation(Z,  X1, angle_y)
        Y3, Z3 = rotation(Y1, Z2, angle_x)

        Zt = Z3 + self.z0
        Yt = Y3 + self.y0
        Xt = X2 + self.x0

        return Xt, Yt, Zt
```

```
def rotation_cs(X, Y, C, S) :
    Xrot = X*C - Y*S
    Yrot = Y*C + X*S
    return Xrot, Yrot
```

$$R_x(\alpha) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{pmatrix} \quad R_y(\beta) = \begin{pmatrix} \cos\beta & 0 & -\sin\beta \\ 0 & 1 & 0 \\ \sin\beta & 0 & \cos\beta \end{pmatrix} \quad R_z(\gamma) = \begin{pmatrix} \cos\gamma & \sin\gamma & 0 \\ -\sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Implementation of hierarchical model

Implemented in package PSCalib for C++ and Python

- classes SegGeometry* – support pixel geometry information for all sensor types; coordinates, dimensions, sizes, areas, mask, etc.
- class GeometryObject – describes one object in hierarchical structure (has parent and the list of children) and its position relative to the parent frame.
- class GeometryAccess – supports hierarchical structure and access to the geometry parameters.

How to use geometry calibration

1. Make sure that the *geometry* calib file is available in the expected default or local place.

2. Use psana module ImgAlgos::PixCoordProducer to evaluate and save pixel coordinate arrays in the calibStore.

3. Use pixel coordinate arrays in other modules or in python code.

--- OR ---

2. Use class PSCalib.GeometryAccess directly in the python script.

References

- Psana - Module Catalog
  https://confluence.slac.stanford.edu/display/PSDM/psana+-+Module+Catalog

- Psana - Module Examples
  https://confluence.slac.stanford.edu/display/PSDM/psana+-+Module+Examples

- Calibration manager
  https://confluence.slac.stanford.edu/display/PSDM/Calibration+Management+Tool

- Mask editor
  https://confluence.slac.stanford.edu/display/PSDM/Mask+Editor

- PSCalib API in Doxyden (C++)
  https://pswww.slac.stanford.edu/swdoc/releases/ana-current/doxy-all/html/group__PSCalib.html
  in Sphinx (Python)
  https://pswww.slac.stanford.edu/swdoc/releases/ana-current/pyana-ref/html/PSCalib/#module-PSCalib