

# simplest transformation example

import fabio # library for reading ESRF Data Format (.edf)

import pygix

pg = pygix.Transform()

pg.load(‘calibration.poni’)

pg.set\_incident\_angle(0.2)

raw = fabio.open(‘image\_0000.edf’).data

I, qxy, qz = pg.transform\_reciprocal(raw)

# transformation including lots of corrections

import fabio

import pygix

pg = pygix.Transform()

pg.load(‘calibration.poni’)

pg.set\_maskfile(‘detector\_mask.edf’)

pg.set\_darkfile(‘dark\_current.edf’)

pg.set\_flatfile(‘flat\_field.edf’)

pg.set\_sample\_orientation(2) # vertical scattering

pg.set\_incident\_angle(0.2)

pg.set\_tilt\_angle(-1.21)

raw = fabio.open(‘image\_0000.edf’).data

I, qxy, qz = pg.transform\_reciprocal(raw, unit=‘q\_A^-1’)

I, q, chi = pg.transform\_polar(data)

# line profile examples

I, q = pg.profile\_sector(raw, 1000, chi\_pos=0, chi\_width=20)

I, chi = pg.profile\_chi(raw, 360, radial\_pos=2.98, radial\_width=0.05)

I, qxy = pg.profile\_ip\_box(raw, 1000, op\_pos=0.1, op\_width=0.05)

I, qz = pg.profile\_op\_box(raw, 1000, ip\_pos=1.98, ip\_width=0.05)

# example for passing geometry directly:

import pygix

# In this example, roatation angles are zero, therefore poni=direct beam position.

# rotation angles are zero by default.

# distances are given in m, rotation angles in rad, wavelength in angstrom.

pg = pygix.Transform(dist=0.192, poni1=0.56, poni2=0.32, wavelength=1.2398)