

NeXus at Diamond

steve.collins@diamond.ac.uk

peter.chang@diamond.ac.uk: DLS NIAC rep.



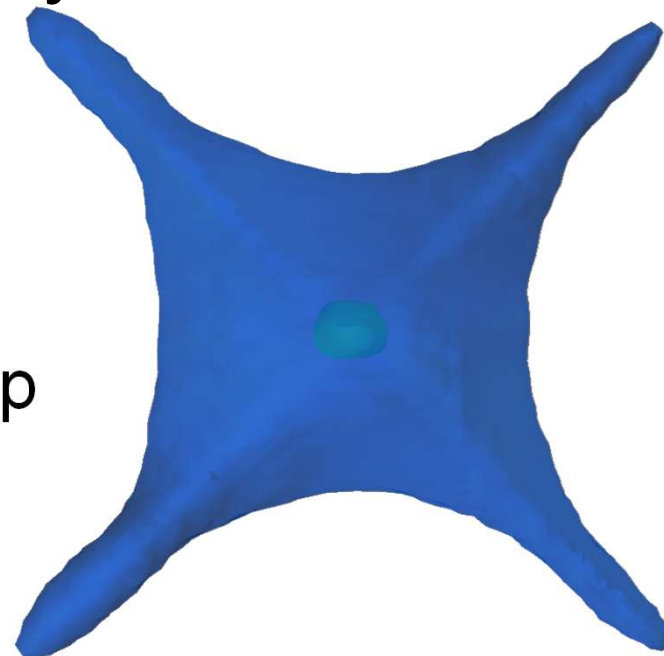
- NeXus adoption at Diamond ~ 80%
- Working towards improving metadata writing and the use of NeXus base classes and application definitions
- NeXus Working Group to improve standardization of NeXus writing
- Aim for more data processing workflows to be based on NeXus classes
- Next steps: Aim to develop ICAT NeXus ingestion
- Next steps: Expand use of processed data NeXus files
- Expect processed NeXus files to be ‘discovered’, not raw data

Standard workflows require well-defined metadata.

Example: *hkldmapper* processor should work with any NXmx compliant NeXus file and is fully automated.



DAWN screen dump showing results of hkl map from I16 NeXus file (four magnetic satellites)



hkl map from CuMnAs thin film (mayavi)

But ...

link_to_userscripts_steve/ x test_classic_sc (2) - JupyterLab x +

https://hub.gke2.mybinder.org/user/spc93-nexus_test_conversions_tmp-1hot78uj/lab/tree/test_classic_scan_...

Apps Photos - Google Ph... Jupyter Hub Diamo... Outlook Web App OneDrive for Busin... 365 Family OneDrive SharePoint Other bookmarks

File Edit View Run Kernel Tabs Settings Help

Filter files by name

Name Last Modified

780431.nxs 4 months ago

815893.nxs 4 months ago

pdnx.py 4 months ago

README.md 4 months ago

requiremen... 4 months ago

test_classic... seconds ago

Untitled.ipyn... a minute ago

Untitled.ipynb x test_classic_scan_conversions X Python 3 (ipykernel)

```
[1]: import sys
sys.path.append('/dls_sw/i16/software/python')
from pdnx import *
from matplotlib.pyplot import *

# use new modified Nexus file with NXmx in a second subentry
p = '%i.nxs'
d = '%i.dat'
e = '%i.xls'

n = pdnx(p % 815893, entry = None, data = None) # Load NeXus file into pdnx (pandas/NeXus) object
== Importing dlstools package

***

[ ]:

[3]: n # display pandas dataframe (all columns in scan)

[3]:
```

	TimeFromEpoch	TimeSec	beamOK	count_time	delta_axis_offset	eta	ic1monitor	kap	kdelta	kgam	kmu	kphi	kth	maxval	maxc
0	1.582424e+09	361029.445210	1.0	1.0	8.8	30.968032	2160.062235	-135.597754	64.693146	0.0	0.0	64.357226	88.555967	4.0	270.0
1	1.582424e+09	361031.738913	1.0	1.0	8.8	30.988032	2161.649749	-135.597754	64.693146	0.0	0.0	64.357226	88.575967	5.0	200.0
2	1.582424e+09	361034.027178	1.0	1.0	8.8	31.008032	2160.733886	-135.597754	64.693146	0.0	0.0	64.357226	88.595967	6.0	100.0
3	1.582424e+09	361036.325025	1.0	1.0	8.8	31.028032	2160.489628	-135.597754	64.693146	0.0	0.0	64.357226	88.615967	5.0	395.0
...
77	1.582425e+09	361229.074869	1.0	1.0	8.8	32.508032	2165.007912	-135.597754	64.693146	0.0	0.0	64.357226	90.095967	4.0	380.0

```
[4]: print(n.nx.tree)

root:NXroot
@HDF5_Version = '1.10.4'
@file_name = '/dls/science/users/spc93/misc_nexus_data/modif...'
@file_time = '2020-05-18T15:56:17.710273'
@h5py_version = '2.9.0'
@nexusformat_version = '0.4.18'
entry1:NXentry
@default = 'scan'
before_scan:NXcollection
@target = '/entry1/before_scan'
PPR:NXcollection
oochi = -44.999841517

[5]: n.nx.plot() # default NeXus plot

scancn eta 0.02 81 BeamOK pil3_100k 1 roi2
```

```
[6]: n.plot('eta', 'sum') # pandas plot
```

Simple 0 2 Python 3 (ipykernel) | Idle Mem: 220.67 / 2048.00 MB Mode: Command Ln 1, Col 1 test_classic_scan_conversions_binder_demo.ipynb

... also need to maintain the ability to manipulate classic *ad-hoc* scan data and convert to legacy formats (.dat, csv, xls) and work with generic data.

