NeXus & HDF5

in 10 Minutes

Tobias Richter

2017-03

NeXus

What does NeXus aim for?

NeXus aims to provide a format that can hold (all optional):

- all information for data processing (raw data)
- all data needed for diagnostics
- metadata
- processed data

For all techniques at

- neutron
- muon
- X-ray
- soft X-ray
- VUV

research facilities.

Plan is to replace any formats that require implicit knowledge about the experiment.

NeXus International Advisory Committee NIAC

- Mark Basham, Diamond Light Source, UK
- Herbert Bernstein, CIF (non-facility member)
- Aaron Brewster, Lawrence Berkeley Laboratory, USA
- Stuart Campbell, Brookhaven National Laboratory, USA
- Bjorn Clausen, Los Alamos National Laboratory, USA
- Stephen Cottrell, Rutherford Appleton Laboratory, UK (Muon Representative)
- Ricardo Ferraz-Leal, SNS and HFIR at ORNL, USA
- Jens-Uwe Hoffmann, Helmholtz Zentrum Berlin, Germany
- Pete Jemian, Advanced Photon Source, USA (Documentation Release Manager)
- Mark Könnecke, Paul Scherrer Institut, Switzerland (Executive Secretary)
- David Männicke, Australian Nuclear Science and Technology Organisation, Australia
- Raymond Osborn, Argonne National Laboratory, USA (non-facility member)
- Tobias Richter, European Spallation Source, Scandinavia (Chair)
- Armando Sole, European Synchrotron Radiation Facility, France
- Jiro Suzuki, KEK, Japan
- Benjamin Watts, Swiss Light Source, Switzerland
- Eugen Wintersberger, DESY, Germany (Technical Manager)

Regular guest:

+ Andreas Förster, Dectris, Switzerland

Credit for HDF5 Slides:

+ Elena Pourmal, The HDF Group, USA

Get involved

- http://www.nexusformat.org
  documentation - “wiki”

- join the open Google Hangout session twice a month

- subscribe to the mailing list

- https://github.com/nexusformat

Base Classes

Contain parameters common for particulars type of equipment or sample, user, etc.

```
base_classes$ ls
NXaperture.nxdl.xml                 NXdetector_module.nxdl.xml      NXlog.nxdl.xml                  NXsample.nxdl.xml
NXattenuator.nxdl.xml               NXdisk_chopper.nxdl.xml         NXmirror.nxdl.xml               NXsensor.nxdl.xml
NXbeam.nxdl.xml                     NXentry.nxdl.xml                NXmoderator.nxdl.xml            NXshape.nxdl.xml
NXbeam_stop.nxdl.xml                NXenvironment.nxdl.xml          NXmonitor.nxdl.xml              NXslit.nxdl.xml
NXbending_magnet.nxdl.xml           NXevent_data.nxdl.xml           NXmonochromator.nxdl.xml        NXsource.nxdl.xml
NXcapillary.nxdl.xml                NXfermi_chopper.nxdl.xml        NXnote.nxdl.xml                 NXsubentry.nxdl.xml
NXcharacterization.nxdl.xml         NXfilter.nxdl.xml               NXobject.nxdl.xml               NXtransformations.nxdl.xml
NXcite.nxdl.xml                     NXflipper.nxdl.xml              NXorientation.nxdl.xml          NXtranslation.nxdl.xml
NXcollection.nxdl.xml               NXfresnel_zone_plate.nxdl.xml   NXparameters.nxdl.xml           NXuser.nxdl.xml
NXcollimator.nxdl.xml               NXgeometry.nxdl.xml             NXpinhole.nxdl.xml              NXvelocity_selector.nxdl.xml
NXcrystal.nxdl.xml                  NXgrating.nxdl.xml              NXpolarizer.nxdl.xml            NXxraylens.nxdl.xml
NXdata.nxdl.xml                     NXguide.nxdl.xml                NXpositioner.nxdl.xml           nxdlformat.xsl
NXdetector.nxdl.xml                 NXinsertion_device.nxdl.xml     NXprocess.nxdl.xml
NXdetector_group.nxdl.xml           NXinstrument.nxdl.xml           NXroot.nxdl.xml
```

With those you can build up a fairly complete description of an experiment.

Application Definitions

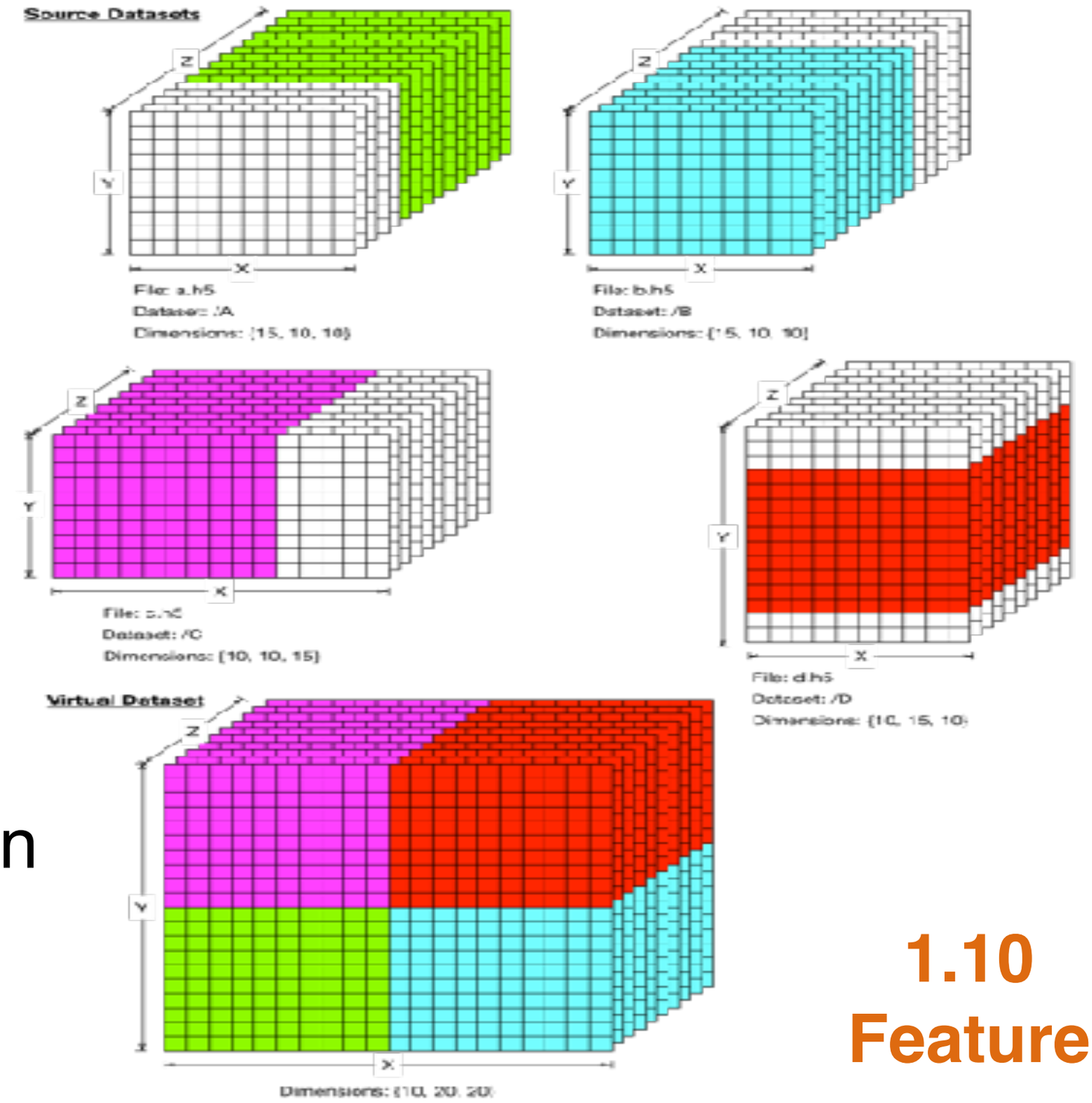
- They guarantee the presence of groups (base classes) and specific fields expected for one type of experiment or measurement
- Defined in XML, for static validation and documentation
- Traditional levels of NeXus adoption:
  - HDF5 container
  - base classes used (all optional)
  - application definition followed (traditionally all required)
- Awesome when it works
- Community adoption is slow - NXmx is a success story

What does HDF5 do for you?

- HDF5 is well and widely supported
  (h5py, MATLAB, IgorPro, …)
- Saves a lot of work accessing data randomly in many dimensions (less then 32)
- Has lots of clever features that make organising and writing data fast, efficient and flexible.
  - Custom compression filters
  - SWMR
  - Virtual Dataset
- As a rule:
  Don’t solve problems that HDF5 has already solved for you.

Virtual Dataset

- External links on steroids
- Datasets combined from multiple (partial) datasets from multiple files
- Can allow simple natural parallelisation of reads and writes



1.10 Feature

What is/was NAPI?

- Back in the day you could ”Download & Install NeXus”
- That provided and interface abstracting the backends HDF5, HDF4 and XML and some tools.
  - most modern HDF5 are not available in NAPI yet using HDF4 and XML is deprecated
  - not a popular programming model in 2017
  - relatively low adoption
  - in maintenance mode

No software needed?

There is need for a reference implementation and some generic tools, validation, …

- We have a validation tool based on libhdf5.
- We main to maintain a repository for HDF5 compression filters.
  https://github.com/nexusformat/HDF5-External-Filter-Plugins
- Work is going on to implement the newly defined versioning of NeXus definitions.
- Also see next slide…

Project: Define Modular Content

Goal: finer granularity control for how information is stored in the file

- Example: Incident wavelength spectrum on the sample could be encoded:
  - as parameter of the source
  - as parameter of the monochromator (if one exists)
  - as property of incident beam on sample
- This works a bit like a unit test for data.

Example Recipe

```
class recipe:
        """
                A demo recipe for finding the information associated with this demo feature.

                This is meant to help consumers of this feature to understand how to implement
                code that understands that feature (copy and paste of the code is allowed).
                It also documents in what preference order (if any) certain things are evaluated
                when finding the information.
        """

        def __init__(self, filedesc, entrypath):
                self.file = filedesc
                self.entry = entrypath
                self.title = "CIF-style sample geometry"

        def findNXsample(self):
                for node in self.file[self.entry].keys():
                        try:
                                absnode = "%s/%s" % (self.entry, node)
                                if self.file[absnode].attrs["NX_class"] == "NXsample":
                                        return absnode
                        except:
                                pass
                # better have custom exceptions
                raise Exception("no NXsample found")

def process(self):
        dependency_chain = []
        try:
                sample = self.findNXsample()
                # this may need more attention for reading all possible types of string
                depends_on = self.file[sample+"/depends_on"][0]
                while not depends_on == ".":
                        dependency_chain.append(depends_on)
                        # this may need more attention for reading all possible types of string
                        depends_on = self.file[depends_on].attrs["depends_on"]

        except Exception as e:
                raise Exception("this feature does not validate correctly: "+e)

                # better have custom exceptions
        return { "dependency_chain" : dependency_chain }
```

proof of concept

NeXus

~~Roadmap~~ Wishlist

- make progress on modularisation project, should simplify the life of software developers
- cater more for processed data, in some areas raw data may drift out of people's focus
- get more facilities on board, addressing any arising requirements
- compete in who writes the best files and the most general visualisation and reduction code

Thank you.

What did he say?

- NeXus is a general, efficient and versatile data format.
- NeXus is a more sustainable option than a home grown file format.
- We have the will and a process to continuously improve.
- There is a community with lots of knowledge and experience. You can join or just use it.