The State of the NeXus Data Format

NeXus International Advisory Committee *

Abstract

NeXus is an effort by an international group of scientists to define a common data exchange format for neutron, muon and x-ray scattering. NeXus has six levels: a physical file format, a file structure, rules for storing individual data items in a file, a dictionary of names, instrument definitions and an application programmers interface to NeXus files. The authors will present the large steps forward which have been made both with instrument definitions and the NeXus-API.

Authors to NeXus are: *Freddie Akeroyd, ISIS Facility,Rutherford Appleton Laboratory, UK; Stuart Cambell, Diamond Light Source, UK; Stephen Cottrell, ISIS Facility,Rutherford Appleton Laboratory, UK; Matthias Drochner, Forschungszentrum Jülich, Germany; Nick Hauser, Australian Nuclear Science and Technology; Organisation, Australia; Ron Ghosh, Institute Laue Langevin, Grenoble, France; Andrew Götz, European Synchrotron Radiation Facility, France; Hartmut Gilde, FRM2, Technische Universität München, Germany; Przemek Klosowski, National Institute of Standards and Technology, USA; Mark Könnecke, Paul Scherrer Institute, Switzerland; Ray Osborn, Intense Pulsed Neutron Source, ANL, USA; Toshiya Otomo, KEK, Japan; Peter Peterson, Spallation Neutron Source, USA; Thomas Proffen, Lujan Neutron Scattering Center, USA.*

Key words: data format, data analysis data management

1. Introduction

As of now, most major facilities choose to store the data measured at neutron, x-ray or muon instruments in a number of different home grown data formats. This situation makes the life of the travelling scientist more difficult as its needs to be because she has to cater for so many different data formats while performing data reduction and analysis. Moreover, the existence of many different data formats is a hurdle to sharing data reduction and analysis software. The NeXus team set out to improve this situation by proposing a single, common data exchange format for both raw and preprocessed data. This proposal is NeXus, the NEutron, X-ray, μ (muon) Science data format.

* Corresponding Author: Mark Könnecke, Laboratory for Development and Methods, Paul Scherrer Institute, 5232 Villigen-PSI, Switzerland, Phone: +41-56-3102512, Fax: +41-56-3102939,Email: Mark.Koennecke@psi.ch

2. NeXus Guiding Principles

The NeXus team tried to follow a couple of guiding principles while designing NeXus:

- Portability across common computing platforms.
- Self describing. It must be possible to deduce the content of the file from data in the file alone.
- Extensible. It must be possible to add data to the file without breaking code based on the standard.
- Flexibility in data organisation. This is required to cope with the plethora of different instruments addressed.
- Upcoming high powered neutron and x-ray sources demand efficient data storage.
- Completeness. All data necessary for common data treatment tasks should be stored in one file.
- Easy Access. The user should be protected from writing low level parsers.
- Facilitate Automatic Plotting.

– Availability in the public domain.

3. NeXus Overview

In order to meet these guidelines the NeXus team developed a proposal consisting of six levels:
– A Physical File Format
– An API to access data files
– Rules for storing data items in a file.
– A file organisation
– A collection of instrument component definitions
– A collection of instrument definitions.

4. Physical File Format and the NeXus-API

With NeXus–API version 3.0 (released: July 2005) three different physical file formats are supported:
– HDF–4[1]
– HDF–5[2]
– XML
Rather then inventing yet another binary file format the NeXus team choose to use an existing scientific data format, HDF, as its underlying physical file format. When a newer version of HDF, HDF-5, appeared support for this format was added, too. These two binary file formats allow for the efficient storage of large data sets and support transparent on the fly compression and decompression of data while reading or writing. HDF file formats are natively supported by many commercial and freeware data treatment packages. Such tools can instantly be used to treat NeXus files. XML is a structured ASCII file format. Support for XML was added in order to cater for those scientists who wish to be able to edit their data manually. All three file formats allow for structured data storage in the file. Not only scientific datasets (multi-dimensional arrays of numbers) are supported but also grouping constructs which allow for ordering elements in a hierarchical manner, much like in a file system.

Access to all three physical file formats is provided through the NeXus–API. With the about 30 functions of this API a user can construct and navigate a NeXus file hierarchy, write and read data and inquire meta information without having to even know about the underlying physical file format. The NeXus–API is written in portable ANSII-C. Language bindings exist for FORTRAN-77, FORTRAN-90, Java, phython, Tcl and through a SWIG[3] interface to a plethora of common scripting languages. The NeXus–API comes with a small set of utilities including a file browser and nxconvert, a utility which allows to convert between all three physical file formats.

<table>
<tr><td colspan="4">NXroot</td></tr>
<tr><td rowspan="11">The root level of a NeXus file</td><td colspan="3">NXentry</td></tr>
<tr><td rowspan="11">All data belonging to one scan or run. A given NeXus file can contain multiple related scans or runs</td><td colspan="2">NXinstrument</td></tr>
<tr><td rowspan="7">The data needed to describe an instrument. Contains groups for each relevant instrument component</td><td>NXsource</td></tr>
<tr><td>NXmoderator</td></tr>
<tr><td>NXvelocity_selector</td></tr>
<tr><td>NXcollimator</td></tr>
<tr><td>NXattenuator</td></tr>
<tr><td>NXdetector</td></tr>
<tr><td colspan="2">NXsample
All the information about the sample</td></tr>
<tr><td colspan="2">NXmonitor
Control monitor</td></tr>
<tr><td colspan="2">NXuser
User information</td></tr>
<tr><td colspan="2">NXdata
Links to plottable data in the NXdetector group. One each for each detector bank. This is the support for generating a typical plot automatically</td></tr>
</table>

Table 1
Overview of the structure of a NeXus file

5. NeXus File Structure

For an overview of the structure of a NeXus file, see table 5. Each NeXus group (or directory) has both a class name and a name. NeXus only standardizes the class names. At the root level of each NeXus file there is some global information and one to n NXentry groups. Each NXentry group holds all the data related to one scan or run. NXentry is NeXus support for multiple related data sets in one file. At each entry level, there are further groups: NXinstrument, NXsample, NXuser, NXmonitor and NXdata and possibly additional groups. NXinstrument contains further groups which represent the building blocks of the instrument. Each group will contain data describing the component and ist position within the instrument. The NXdetector group will also contain the counts. The NXsample group holds the sample information. The NXmonitor group the monitor. The NXdata group contains plottable data. This is some data along with axis information which an automatic tool can use to display a default plot of the data. The structure seems to require the duplication of possibly large data sets. This is not

the case as both the file format and the NeXus–API support the concept of links, i.e. references to data elements already written. This is very similar to links in a unix file system.

Additional groups may be present in a NXentry group which describe event based data, logged data items, processing information or the intent of the data.

Within groups, datasets are stored which describe the corresponding component. Datasets can have attributes. A standard attribute which has to be written is the units used. For unit names we use the conventions established by the Udunits[4] unit conversion program. The axis datasets describing the dimensions of a multi dimensional dataset are stored in the same group as the dataset. A convention allows to associate the dimensions of a multi dimensional dataset with the appropriate datasets describing the axis.

For defining the position of a component within an instrument two schemes exist: a simple one where positions are defined by distances to the previous component, polar_angle (in most cases equivalent to two theta) and tilt angle. For the distances, the sample is at zero, towards the source is negative, towards the detector is positive. There also exists a more sophisticated system using NXgeometry and its sub classes. The system is based on the coordiante system of McStas[6] and has been included in order to meet the demands of the instrument simulation community.

6. NeXus Component and Instrument Definitions

A NeXus International Advisory Committee (NIAC) was founded in order to oversee the development of NeXus and further the development of both component definitions and instrument definitions. The following component definitions now exist:

- NXentry
  - NXinstrument
    - NXsource
    - NXmoderator
    - NXcrystal
    - NXchopper
    - NXguide
    - NXcollimator
    - NXaperture
    - NXfilter
    - NXattenuator
    - NXflipper
    - NXmirror
    - NXdetector
    - NXbeam_stop
  - NXsample
  - NXmonitor
  - NXdata
  - NXevent_data
  - NXuser
  - NXprocess
  - NXcharacterizations

In addition there are groups which may appear at any appropriate level in the hierarchy:

NXlog for logging variables, for example temperature

NXnote for free text notes

NXbeam for summarizing the status of the incoming or outgoing neutron beam.

NXgeometry with subgroups NXtranslation, NXshape, NXorientation for defining instrument component positions very accuratetly.

NXenvironment with subgroup NXsensor for handling sample environment controllers.

So far the following instrument definitions have been approved:

- Time of flight neutron powder diffractometer
- Monochromatic triple axis spectrometer
- Monochromatic small angle scattering instrument
- Time-of-flight Neutron Reflectometer.
- Direct geometry time-of-flight spectrometer.

More are to follow.

7. Conclusion

At the last count, 14 major facilities world wide have commited themselves to use NeXus for data storage. More then 30 NeXus aware data treatment programs are already available. This shows that the NeXus file format is gaining widespread acceptance. The NeXus–API has proven mature enough to write and process more then 500.000 files, especially at PSI. For more information, do not hesitate to consult the NeXus WWW–site[5] or to contact the members of the NIAC.

References

[1] http://hdf.ncsa.uiuc.edu/hdf4.html

[2] http://hdf.ncsa.uiuc.edu/HDF5

[3] http://www.swig.org

[4] http://www.unidata.ucar.edu/packages/udunits

[5] http://www.nexus.anl.gov;
http://www.nexus.anl.gov/mediawiki

[6] K. Lefmann, K. Nielsen, Neutron News. 10 (1999) 20