NeXus Manual

COLLABORATORS			
	TITLE : NeXus Manual		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY	Peter F. Peterson	December 14, 2009	

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

Contents

List of Figures

List of Tables

Preface

Note

Need to provide basic information about this work here.

Chapter 1

Purpose of NeXus definitions

1.1 Simple plotting

This is a paragraph.

1.2 Unified format for reduction and analysis

This is a paragraph.

Chapter 2

MediaWiki Manual

2.1 NeXus Introduction

In recent years, a number of scientists and computer programmers working in neutron and synchrotron facilities around the world came to the conclusion that a common data format would fulfil a valuable function in the scattering community. As instrumentation becomes more complex and data visualization become more challenging, individual scientists, or even institutions, have found it difficult to keep up with new developments. A common data format makes it easier, both to exchange experimental results and to exchange ideas about how to analyze them. It promotes greater cooperation in software development and stimulate the design of more sophisticated visualization tools. For additional background information see the chapter titled "Brief history of the NeXus format."

This section is designed to give a brief introduction to NeXus, the data format that has been developed in response to these needs. It explains what a modern data format such as NeXus is and how to write simple programs to read and write NeXus files.

2.1.1 What is NeXus?

The NeXus data format has four components:

A set of subroutines to make it easy to read and write NeXus files.

A set of design principles to help people understand what is in them.

A set of instrument definitions to allow the development of more portable analysis software.

A set of low-level file formats to actually store NeXus files on physical media.

2.1.1.1 A Set of Subroutines

In the past, a data format was defined by a document describing the precise location of every item in the data file, either as row and column numbers in an ASCII file, or as record and byte numbers in a binary file. In modern data formats, such as NeXus, the user does not need to know where the data are stored, just what they are called. It is the job of the subroutine library to retrieve the data.

For example, in NeXus, a program to read in the wavelength of an experiment would contain lines similar to the following:

Example 2.1 Simple example of reading data

```
NXopendata (fileID, "wavelength");
NXgetdata (fileID, lambda);
NXclosedata (fileID);
```

In this example, the program requests the value of the data that has the label `wavelength`, storing the result in the variable lambda. `fileID` is a file identifier that is provided by NeXus when the file is opened.

We shall provide a more complete example when we have discussed the contents of the NeXus files.

2.1.1.2 A Set of Design Principles

NeXus data files contain two types of entity: data items and data groups.

Data Items These can be scalar values or multidimensional arrays of a variety of sizes (1-byte, 2-byte, 4-byte, 8-byte) and types (characters, integers, floats). Extra information required to describe a particular data item, such as the data units, can be stored as a data attribute.

Data Groups These are like folders that can contain a number of data items and/or other groups.

In fact, a NeXus file can be viewed as a computer file system. Just as files are stored in folders (or subdirectories) to make them easy to locate, so NeXus data items are stored in groups. The group hierarchy is designed to make it easy to navigate a NeXus file.

2.1.1.2.1 Example of a NeXus File

The following diagram shows an example of a NeXus file represented as a tree structure.
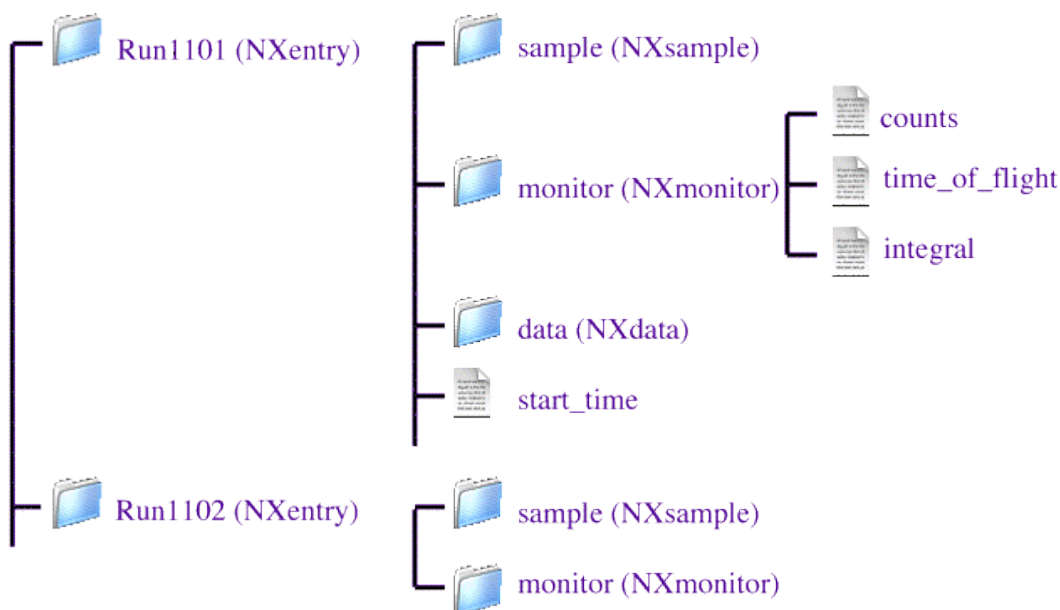
Figure 2.1: Example of a NeXus file

Note that each data item is identified by a name, e.g., counts, but each group is identified both by a name and, in parentheses, a class identifier, e.g., monitor (NXmonitor). The class names, which all begin with NX, define the sort of data items that the group should contain, in this case, counts from a beamline monitor. The hierarchical design, with data items nested in groups, makes it easy to identify information if you are browsing through a file.

2.1.1.2.2 Important Classes

Here are some of the important classes found in nearly all NeXus files. A complete list can be found in the NeXus Design section

NXentry The top level of any NeXus file contains one or more groups with the class NXentry. These contain all the data that is required to describe an experimental run or scan. Each NXentry typically contains a number of groups describing sample information (class NXsample), instrument details (class NXinstrument), and monitor counts (class NXmonitor).

NXdata Each NXentry group contains one or more groups with class NXdata. These groups contain the experimental results in a self-contained way, i.e., it should be possible to generate a sensible plot of the data from the information contained in each NXdata group. That means it should contain the axis labels and titles as well as the data.

NXsample A NXentry group will often contain a group with class NXsample. This group contains information pertaining to the sample, such as its chemical composition, mass, and environment variables (temperature, pressure, magnetic field, etc.).

NXinstrument There might also be a group with class NXinstrument. This is designed to encapsulate all the instrumental information that might be relevant to a measurement, such as flight paths, collimations, chopper frequencies, etc.
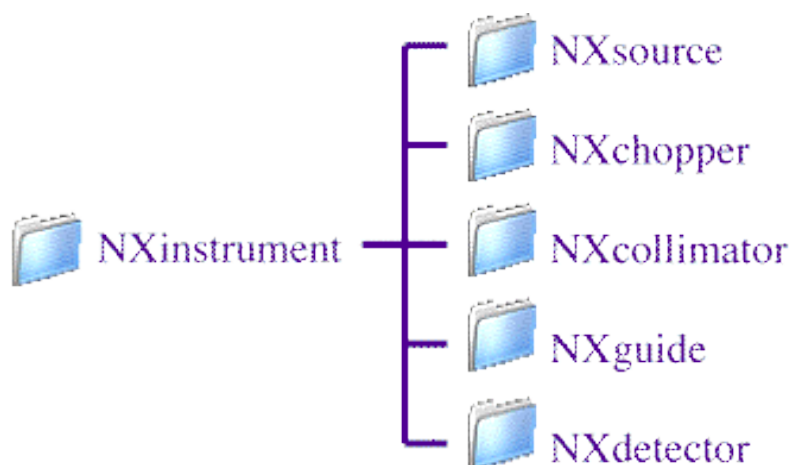
Figure 2.2: NXinstrument excerpt

Since an instrument can comprise several beamline components each defined by several parameters, they are each specified by a separate group. This hides the complexity from generic file browsers, but makes the information available in an intuitively obvious way if it is required.

2.1.1.2.3 Simple Example

NeXus data files do not need to be complicated. In fact, the following diagram shows an extremely simple NeXus file that could be used to transfer data between programs.

Figure 2.3: Simple Example

This illustrates the fact that the structure of NeXus files is extremely flexible. It can accommodate very complex instrumental information, if required, but it can also be used to store very simple data sets.

2.1.1.3 A Set of Instrument Definitions

If the design principles are followed, it will be easy for anyone browsing a NeXus file to understand what it contains, without any prior information. However, if you are writing visualization or analysis software, you will need to know precisely what information is contained in advance. For that reason, NeXus provides a way of defining the format for particular instrument types, e.g., time-of-flight small angle neutron scattering. This requires some agreement by the relevant communities, but would allow the development of much more portable software.

These instrument definitions are being formalized as XML files, using a specially devised syntax that specifies the names of data items, and whether they are optional or required. The following is an example of such a file for the simple NeXus file shown above.

Example 2.2 *verysimple.xml*: A very simple NeXus file

```
1  <?xml version="1.0" ?>
2  <!--
3  URL: http://www.neutron.anl.gov/nexus/xml/simple.xml
4  Editor: Ray Osborn <ROsborn@anl.gov>
5
6  A very simple NeXus file
7
8  -->
9  <NXentry name="{Name of entry}">
10     <NXdata name="{Name of data}">
11       <time_of_flight units="microseconds" type="NX_FLOAT32[i]">{Time-of-flight}</ ↩
             time_of_flight>
12       <data type="NX_INT32[i]" axes="time_of_flight"> {Counts} </data>
13     </NXdata>
14  </NXentry>
15
```

If you want to define the format of a particular type of NeXus file for your own use, e.g. as the standard output from a program, you are encouraged to "publish" the format using this XML format.

2.1.1.4 A Set of Low-Level File Formats

To actually store NeXus files on physical media, different low-level file formats are available, namely HDF4, HDF5, and XML. The NeXus library may be configured to support all of them, or any nonempty subset. Applications that create NeXus files need to decide (or let the user decide) in which low-level format data shall be stored. Generic data analysis applications should be able to read any low-level format.

2.1.2 The NeXus Application Programming Interface (API)

The NeXus API consists of routines to read and write NeXus data files.

2.1.2.1 How do I write a NeXus file?

The NeXus Application Program Interface (API) provides a set of subroutines that make it easy to read and write NeXus files. These subroutines are available in C, Fortran 77, Fortran 90, Java, and IDL. Access from other languages, such as Python, is anticipated in the near future. It is also possible to read NeXus files in a number of data analysis tools, such as LAMP, ISAW, and Open GENIE.

The API uses a very simple "state" model to navigate through a NeXus file. When you open a file, the API provides a file "handle", which then stores the current location, i.e. which group and/or data item is currently open. Read and write operations then act on the currently open entity. In the following, we walk through some parts of a typical NeXus program written in C. See the NeXus API chapter for a more complete version.

First, it is necessary to open the file, specifying whether we want read or write access.

Example 2.3 Open a file

```c
#include "napi.h"

 int main()
 {
   NXhandle fileID;
   NXopen ('NXfile.nxs', NXACC_CREATE, &fileID);
```

The file is opened with "create" access (implying write access), and the API returns a file identifier of type NXhandle. Next, we create an NXentry group to contain the scan using NXmakegroup and then open it for access using NXopengroup.

Example 2.4 Create an entry

```c
NXmakegroup (fileID, "Entry", "NXentry");
NXopengroup (fileID, "Entry", "NXentry");
```

The plottable data is contained within an NXdata group, which must also be created and opened.

Example 2.5 Create data group

```c
NXmakegroup (fileID, "Data", "NXdata");
NXopengroup (fileID, "Data", "NXdata");
```

To create a data item, call NXmakedata, specifying the data name, type (NX_FLOAT32), rank (in this case, 1), and length of the array (n_t). Then, it can be opened for writing.

Example 2.6 Create data array

```c
 NXmakedata (fileID, "time_of_flight", NX_FLOAT32, 1, &n_t);
  NXopendata (fileID, "time_of_flight")
```

Then write the data using NXputdata.

Example 2.7 Write the data

```c
  NXputdata (fileID, t);
```

With data item is still open, we can also add some data attributes, such as the data units, which are specified as a character string (type NX_CHAR) that is 12 bytes long.

Example 2.8 Add an attribute

```c
NXputattr (fileID, "units", "microseconds", 12, NX_CHAR);
```

Then we close the data item before opening another. In fact, the API will do this automatically if you attempt to open another data item, but it is better style to close it yourself.

Example 2.9 Close the data array

```
NXclosedata (fileID);
```

The remaining data items in this group are added in a similar fashion. Note that the indentation whenever a new data item or group are opened is just intended to make the structure of the NeXus file more transparent.

Example 2.10 The rest of the data group

```
1      NXmakedata (fileID, "polar_angle", NX_FLOAT32, 1, &n_p);
2        NXopendata (fileID, "polar_angle");
3          NXputdata (fileID, polar_angle);
4          NXputattr (fileID, "units", "degrees", 7, NX_CHAR);
5        NXclosedata (fileID);
6        dims[0] = n_t;
7        dims[1] = n_p;
8        NXmakedata (fileID, "counts", NX_INT32, 2, dims);
9        NXopendata (fileID, "counts");
10         NXputdata (fileID, counts);
11       NXclosedata (fileID);
```

Finally, close the groups (NXdata and NXentry) before closing the file itself.

Example 2.11 Cleanup

```
    NXclosegroup (fileID);
    NXclosegroup (fileID);
    NXclose (&fileID);
    return;
}
```

2.1.2.2 How do I read a NeXus file?

Reading a NeXus file is almost identical to writing one. Obviously, it is not necessary to call NXmakedata since the item already exists, but it is necessary to call one of the query routines to find out the rank and length of the data before allocating an array to store it.

Here is part of a program to read the time-of-flight array from the file created by the example above.

Example 2.12 File read example

```
1   NXopen ('NXfile.nxs', NXACC_READ, &fileID);
2     NXopengroup (fileID, "Entry", "NXentry");
3       NXopengroup (fileID, "Data", "NXdata");
4         NXopendata (fileID, "time_of_flight");
5           NXgetinfo (fileID, &rank, dims, &datatype);
6           NXmalloc ((void **) &tof, rank, dims, datatype);
7           NXgetdata (fileID, tof);
8         NXclosedata (fileID);
9       NXclosegroup (fileID);
10    NXclosegroup (fileID);
11  NXclose (fileID);
```

NeXus files can also be viewed by a command-line browser, NXbrowse, which is included with the NeXus API. The following is an example session of using NXbrowse to view a data file from the LRMECS spectrometer at IPNS. The following commands are used (see the NXbrowse web page):

Table 2.1: NXbrowse Command Description

Command	Description
dir	List the contents of the current group
open Histogram1	Open the NeXus group "Histogram1"
read title	Print the contents of the NeXus data labelled "title"
close	Close the current group
quit	Quit the browser

Example 2.13 Using NXbrowse

```
1  %> nxbrowse lrcs3701.nxs
2
3  NXBrowse 3.0.0. Copyright (C) 2000 R. Osborn, M. Koennecke, P. Klosowski
4      NeXus_version = 1.3.3
5      file_name = lrcs3701.nxs
6      file_time = 2001-02-11 00:02:35-0600
7      user = EAG/RO
8  NX> dir
9    NX Group : Histogram1 (NXentry)
10   NX Group : Histogram2 (NXentry)
11 NX> open Histogram1
12 NX/Histogram1> dir
13   NX Data  : title[44] (NX_CHAR)
14   NX Data  : analysis[7] (NX_CHAR)
15   NX Data  : start_time[24] (NX_CHAR)
16   NX Data  : end_time[24] (NX_CHAR)
17   NX Data  : run_number (NX_INT32)
18   NX Group : sample (NXsample)
19   NX Group : LRMECS (NXinstrument)
20   NX Group : monitor1 (NXmonitor)
21   NX Group : monitor2 (NXmonitor)
22   NX Group : data (NXdata)
23 NX/Histogram1> read title
24   title[44] (NX_CHAR) = MgB2 PDOS 43.37g 8K 120meV E0@240Hz T0@120Hz
25 NX/Histogram1> open data
26 NX/Histogram1/data> dir
27   NX Data  : title[44] (NX_CHAR)
28   NX Data  : data[148,750] (NX_INT32)
29   NX Data  : time_of_flight[751] (NX_FLOAT32)
30   NX Data  : polar_angle[148] (NX_FLOAT32)
31 NX/Histogram1/data> read time_of_flight
32   time_of_flight[751] (NX_FLOAT32) = [ 1900.000000 1902.000000 1904.000000 ...]
33     units = microseconds
34     long_name = Time-of-Flight [microseconds]
35 NX/Histogram1/data> read data
36   data[148,750] (NX_INT32) = [ 1 1 0 ...]
37     units = counts
38     signal = 1
39     long_name = Neutron Counts
40     axes = polar_angle:time_of_flight
41 NX/Histogram1/data> close
42 NX/Histogram1> close
43 NX> quit
```

The source code provides an example of how to write a NeXus reader. The test programs included in the NeXus API may also be useful to study.

2.1.3 NeXus Mailing Lists

blah, blah

2.1.4 Where to go from here?

The other sections of this document give much more detailed descriptions of the NeXus data format.

NeXus Design This describes NeXus' hierarchical design, and gives more information on how to construct valid groups.

NeXus API This describes the NeXus programming interface, with lists of all the subroutines and supported languages

NeXus Mailing Lists NeXus mailing lists

2.2 NeXus Design

The structure of NeXus files is extremely flexible, allowing the storage both of simple data sets, e.g., a single data array and its axes, and also of highly complex data, e.g., the simulation results of an entire multi-component instrument. This flexibility is achieved through a hierarchical structure, with related data items collected together into groups, making NeXus files easy to navigate, even without any documentation. NeXus files are self-describing, and should be easy to understand, at least by those familiar with the experimental technique.

The logical design is distinct from the underlying format used to store the NeXus file on disk, which are written using the NeXus Application Program Interface (API). Refer to the API section for more details.

2.2.1 NeXus Objects

NeXus data files contain two types of elementary object: data items and data groups. In addition, metadata required to describe a data item, e.g. its physical units, can be attached to the data as data attributes.

2.2.1.1 Data Items

Data items contain the essential information stored in a NeXus file. They can be scalar values or multidimensional arrays of a variety of sizes (1-byte, 2-byte, 4-byte, 8-byte) and types (integers, floats, characters). The items may store both experimental results (counts, detector angles, etc), and other information associated with the experiment (start and end times, user names, etc). Data items are identified by their names, which must be unique within the group in which they are stored.

2.2.1.2 Data Attributes

Attributes are extra (meta-)information that are associated with particular data items. They are used to annotate the data, e.g. with physical units or calibration offsets, and may be scalar numbers or character strings. In addition, NeXus uses attributes to identify plottable data and their axes, etc. Finally, NeXus files themselves have global attributes that identify the NeXus version, file creation time, etc.. Attributes are identified by their names, which must be unique in each data item.

2.2.1.3 Data Groups

NeXus files consist of data groups, which contain data items and/or other groups to form a hierarchical structure. This hierarchy is designed to make it easy to navigate a NeXus file by storing related data items together. Data groups are identified both by a name, which must be unique within a particular group, and a class. There can be multiple groups with the same class.

2.2.2 NeXus Classes

Data groups often describe objects in the experiment (monitors, detectors, monochromators, etc.), so that the contents (both data items and/or other data groups) comprise the properties of that object. NeXus has defined a set of standard objects, or classes, out of which a NeXus file can be constructed. Each data group is therefore identified by a name and a class. The group class, which always has "NX" as a prefix, defines the type of object and the properties that it can contain, whereas the group name defines a unique instance of that class. These classes are defined in XML using the NeXus Definition Language (NXDL) format.

Not all classes define physical objects. Some refer to logical groupings of experimental information, such as plottable data, sample environment logs, beam profiles, etc.

The following table shows the hierarchy of a standard NeXus file, and where groups of a particular class are located. There can be multiple instances of each class. On the other hand, a typical NeXus file will only contain a small subset of the possible classes.

2.2.2.1 NeXus Data

One of the aims of the NeXus design was to make it possible to separate the measured data in a NeXus file from all the metadata that describe how that measurement was performed. In principle, it should be possible for a plotting utility to identify the plottable data automatically (or to provide a list of choices if there is more than one set of data). In order to distinguish the actual measurements from this metadata, it is stored separately in groups with the class NXdata. These groups encapsulate all the information required to produce a meaningful plot, including any error arrays and axis scales, i.e. the physical values corresponding to the data dimensions.

The NXdata groups have to be flexible enough to cope with data of arbitrary rank and provide a mechanism for associating axis scales with the appropriate dimension of data. We use data attributes to accomplish this. Here are the main rules that must be followed in constructing an NXdata group.

- Each NXdata group will consist of only one data set containing plottable data and their standard deviations.

- The data set will be identified by an attribute of "signal" given a value 1.

- This data set may be of arbitrary rank.

If available, the standard deviations of the data are to be stored in a data set of the same rank and dimensions, with the name "errors".

- For each data dimension, there should be a one-dimensional array of the same length.

- These one-dimensional arrays are the "dimension scales" of the data i.e. the values of the independent variables at which the data is measured e.g. scattering angle or energy transfer.

There are two methods of linking each data dimension to its respective dimension scale.

1. The first method is to define an attribute of each dimension scale called "axis". It is an integer whose value is the number of the dimension, in order of fastest varying dimension. i.e. if the array being stored is data, with elements data[j][i] in C and data(i,j) in Fortran, where i is the time-of-flight index and j is the polar angle index, the NXdata group would contain :

Example 2.14 Old way of denoting axes

```
1  <NXdata name=" data " >
2          <time_of_flight axis= 1 primary= 1 > 1500.0 1502.0 1504.0 ... </time_of_flight>
3          <polar_angle axis= 2 primary= 1 > 15.0 15.6 16.2 ... </polar_angle>
4          <data > 5 7 14 ... </data>
5  </NXdata>
```

This attribute must be defined for each dimension scale.

2. The second method is to define an attribute of the data itself called "axes". It contains the names of each dimensions scale as a comma- or colon-delimited list in the order they appear in C. Optionally, the list can be enclosed in brackets, but should not contain any spaces, e.g.

Example 2.15 Preferred way of denoting axes

```
1  <NXdata name=" data " >
2          <time_of_flight > 1500.0 1502.0 1504.0 ... </time_of_flight>
3          <polar_angle > 15.0 15.6 16.2 ... </polar_angle>
4          <data axes="polar_angle:time_of_flight" > 5 7 14 ... </data>
5  </NXdata>
```

The second method is required when the dimension scale is used in more than one NXdata group in a different context, e.g. it is used as the x-axis in one group and the y-axis in another.

The first method was historically the first to be used, but the second is now recommended for future applications. However, both will be supported in NeXus utilities that identify dimension scales, e.g. NXUfindaxis.

There are limited circumstances in which more than one dimension scale for the same data dimension can be included in the same NXdata group. The most common is when they are the three components of an (hkl) scan. In order to handle this case, we have defined another attribute of type integer called "primary" whose value determines the order in which the scale is expected to be chosen for plotting, i.e. 1st choice: primary = 1 2nd choice: primary = 2 etc. If there is more than one scale with the same value of the "axis" attribute, one of them must have the "primary" attribute set to 1. Defining the "primary" attribute for the other scales is optional.

N.B. The "primary" attribute can only be used with the first method of defining dimension scales discussed above. In addition to the signal data, this group could contain a data set of the same rank and dimensions called "errors" containing the standard deviations of the data.

It is often (usually) necessary to associate the data and/or axis scales with other metadata stored in other groups, e.g. the NXsample group or components of the NXinstrument group. For example, it may be necessary to perform corrections for the detector efficiency using information stored in the associated NXdetector group. In this case, it is recommended that the relevant arrays are initially stored in those groups, and then linked to the NXdata group. The API will provide a mechanism for identifying the parent group so that the relevant metadata can be accessed.

Here is a simple example to illustrate the concept:

Example 2.16 Abbreviated NeXus file

```
1  <NXentry name="entry">
2      <NXsample name="sample">
3          <magnetic_field link="/entry/sample">10.0</magnetic_field>
4      </NXsample>
5      <NXinstrument name="instrument">
6          <NXdetector name="detector">
7              <data axes=" time_of_flight:magnetic_field " link="/entry/instrument/detector"> ↩
                    5 7 14 etc </data>
8              <time_of_flight link="/entry/instrument/detector ">1500.0 1502.0 1504.0 etc </ ↩
                    time_of_flight>
9          </NXdetector>
10     </NXinstrument>
11     <NXdata>
12         <data axes="time_of_flight:magnetic_field" link="/entry/instrument/detector">{link ↩
                to values in NXdetector}</data>
13         <time_of_flight link="/entry/instrument/detector">{link to values in NXdetector}</ ↩
                time_of_flight>
14         <magnetic_field link="/entry/sample">{link to values in NXsample}</magnetic_field>
15     </NXdata>
16  </NXentry>
```

The general principle is that physical quantites are stored in the groups that they refer to (e.g. counts in NXdetector, temperature in NXsample) and these quantities are then linked into NXdata for interpretation. In this example, there are two axis scales, "magnetic_field" and "time_of_flight", which are stored in NXsample and NXdetector groups respectively. A program is able to use the information in the "link" attribute to locate the respective groups. One corollary of this is that there should be one NXdetector group for each NXdata group, e.g. one for each detector bank in a multi-bank instrument.

The syntax of the "link" attribute requires a bit of explanation. Under HDF4 you can only create, what would be called under UNIX, "hard links". Hard links have the characteristics that:

- The name of the entity must be the same in both the original and linked groups

- The attributes of both the origial entity and the linked one are the same

- You cannot distinguish the original entity from the linked one

- You cannot follow a link - it is like an inode in a filesystem and just points at the data

To overcome this and allow us to link from NXdata to, say, NXsample and to know that the original data belongs to NXsample we write the "link" attribute that contains the path of the original group containing it. All linked entities will share this "link" attribute and thus can use it to locate the original source group. We are effectively using the "link" attribute to simulate "symbolic links". So in the above example both the original "time_of_flight" and the linked one will share a link attribute containing the text "/entry/instrument/detector" because "/entry/instrument/detector/time_of_flight" is the original instance.

2.2.2.2 NeXus Attributes

Table 2.2: Global Attributes

Name	Type	Description
file_name	NX_CHAR	File name of original NeXus file to assist in identification if the external name has been changed
file_time	ISO 8601	Date and time of file creation
file_update_time	ISO 8601	Date and time of last file change at close
NeXus_version	NX_CHAR	Version of NeXus API used in writing the file
creator	NX_CHAR	Facility or program where the file originated

Table 2.3: Data Attributes

Name	Type	Description
units	NX_CHAR	Units of data which must conform to the standard defined by the Unidata UDunits utility (in particular, see udunits.dat)
signal	NX_INT32	Defines which data set contains the signal to be plotted - set to 1 for main signal
axes	NX_CHAR	Defines the names of the dimension scales for this data set as a comma-delimited array, optionally surrounded by brackets (see a longer discussion in the section on NXdata structure) i.e. if the array being stored is `data`, with elements `data[j][i]` in C and `data(i,j)` in Fortran, with dimension scales `time_of_flight[i]` and `polar_angle[j]`, `data` would have an attribute called "axes" with the following value : `[polar_angle,time_of_flight]`

Table 2.3: (continued)

Name	Type	Description
axis	NX_INT32	As an alternative to using the "axes" attribute, this defines the rank of the signal data for which this data set is a dimension scale in order of the fastest varying index (see a longer discussion in the section on NXdata structure) i.e. if the array being stored is `data`, with elements `data[j][i]` in C and `data(i,j)` in Fortran, "axis" would have the following values: ith dimension (axis = 1), jth dimension (axis = 2), etc.
primary	NX_INT32	Defines the order of preference for dimension scales which apply to the same rank of signal data - set to 1 for preferred dimension scale
long_name	NX_CHAR	Defines title of signal data or axis label of dimension scale
calibration_status	NX_CHAR	Defines status of data value - set to "Nominal" or "Measured"
histogram_offset	NX_FLOAT32	Defines the offset from the first data point to its bin boundary. i.e. left_bin = data[1] - histogram_offset - set to 0 if the data are not histograms. The points themselves should be set to the bin centers. For reasoning behind this design, see note on histograms.
checksum	NX_INT32	Sum of data array acting as a check on data integrity
version	NX_CHAR	Version of XML DTD file or schema on which the NeXus file is based. Should only be used with the "analysis" data item in an NXentry group.
URL	NX_CHAR	The URL of the XML DTD file or schema on which the NeXus file is based. Should only be used with the "analysis" data item in an NXentry group.

2.2.3 NeXus Coordinate System

Nexus provides two coordinate systems: a NXgeometry based system for physical coordinates of beamline components and a polar coordinate based system for scattering coordinates. The usage of these two systems can be seen by considering a 3He gas tube detector:

- The NXgeometry system represents true spatial location and would describe a cylinder at a certain distance from the sample that never changes from one run to another

- The polar system would describe the scattering, rather than actual, geometry. For example, the *distance* coordinate would refer to the distance from the sample to an effective measurement point within the gas tube, which would depend on neutron energy; lower energy neutrons would tend to penetrate a smaller distance within the tube, and so have a shorter secondary flight path.

2.2.3.1 Simple (Spherical Polar) Coordinate System

In this system the instrument is considered as a set of components through which the incident beam passes. The variable **distance** is assigned to each component and represents the effective beam flight path length between this component and the sample. A sign convention is used where -ve numbers represent components pre-sample and +ve numbers components post-sample.

For angular information, the quantities *polar_angle* and *azimuthal_angle* are used and these quantities correspond exactly to the usual spherical polar coordinate definitions i.e. the polar_angle is the *zenith angle* and measured with respect to a *z* axis and the azimuthal_angle to the *x* axis in the xy plane. The direction of these local axes may be different for each component: *z* is the incident beam direction for the **previous** component and we then follow *McStas* for *x* and *y* i.e. the *x* axis is perpendicular to the beam in the horizontal plane pointing left as seen from the source, and the *y* axis points upwards (see diagram below). The *z* axis thus represents the direction of the beam if it was un-deviated by the previous component, and so the polar_angle and azimuthal_angle for a component indicate how much the beam was bent/scattered by the previous component.

If we consider an NXdetector element placed directly after an NXsample, the *z* axis would be in the direction of the beam incident on NXsample. The polar_angle for the NXdetector would be the angle between the scattered beam and this *z* axis and

so correspond to the *Bragg angle* or *two theta* even for out-of-plane scattering. The azimuthal_angle would be the angle between the positive x-axis and the scattered beam projected onto the xy-plane - scattering to the left as seen from the source would have azimuthal_angle=0 and scattering to the right azimuthal_angle=pi. The distance would correspond to what is often called the *secondary flight path length* or *L2*.

2.2.3.2   NXgeometry based system

This coordinate system is based more fully on the *McStas coordinate system* . The instrument is given a global, absolute coordinate system where the *z* axis points in the direction of the incident beam, the *x* axis is perpendicular to the beam in the horizontal plane pointing left as seen from the source, and the *y* axis points upwards. Each beamline component also has a local coordinate system, which is defined by the NXgeometry object. The local *z* direction for a component is taken as the incident beam direction, with *x* and *y* defined as before (*i.e.* the *x* axis is perpendicular to the beam in the horizontal plane pointing left as seen from the source, and the *y* axis points upwards). Information about these coordinate systems and the placement of components is described by the NXgeometry class via its NXtranslation and NXorientation members.
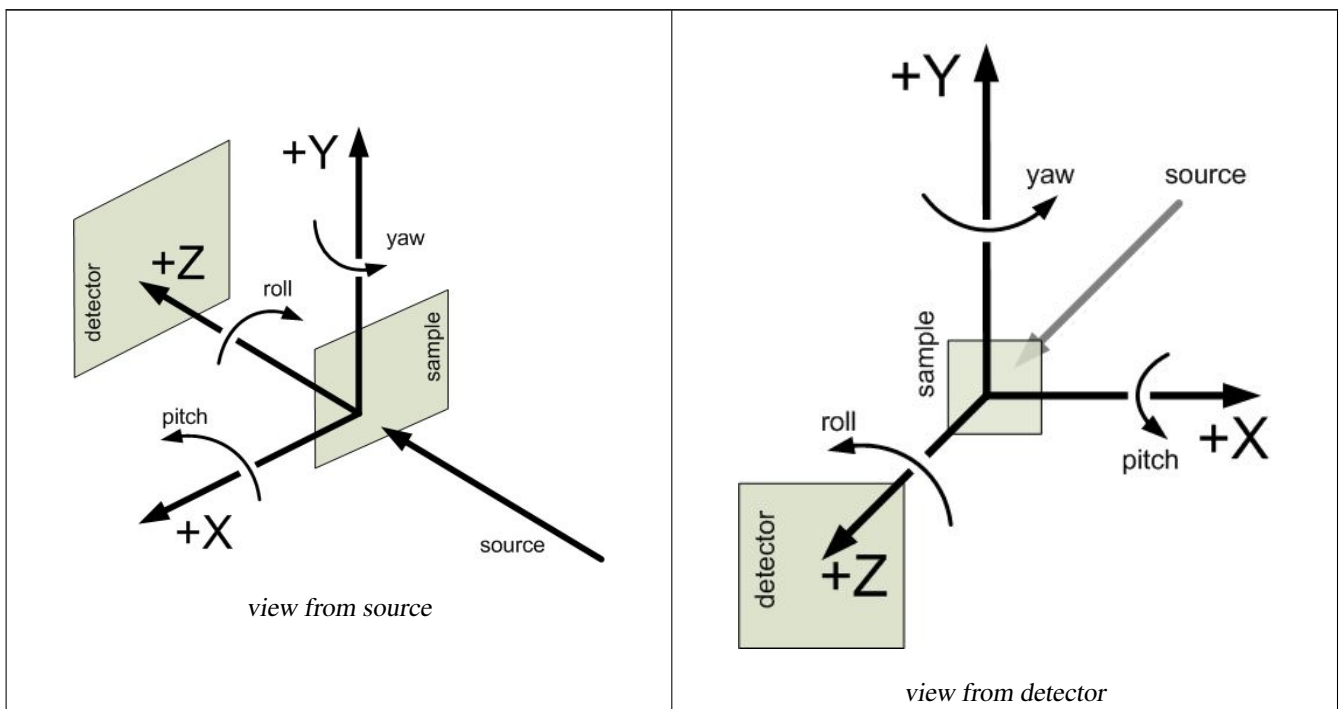
Figure 2.4: Definition of translation and orientation geometry

When computing a transformation, NXtranslation is applied before NXorientation. All of our axes are right handed and orthogonal. Orientation information is stored as direction cosines. The direction cosines will be between the local coordinate directions and the reference directions (to origin or relative NXgeometry). Calling the local unit vectors (x',y',z') and the reference unit vectors (x,y,z) the six numbers will be [x' dot x, x' dot y, x' dot z, y' dot x, y' dot y, y' dot z] where "dot" is the scalar dot product (cosine of the angle between the unit vectors). The unit vectors in both the local and reference coordinates are right-handed and orthonormal: with this restriction we only ned to store 6 rather than 9 direction cosines as the z' axis can be obtained by the vector cross product of x' and y'.

The origin of coordinates is arbitrary, but all components in the file must either agree on its absolutelocation or use relative positioning. To allow for this generality, an origin member can be defined in NXentry; its use will be detailed shortly.

We choose as our absolute the origin the scattering center, which is where a perfectly aligned sample would be. Note that the centre of the sample itself may not always be at this point if the sample is being scanned across the beam. With an origin at the *scattering centre* the spherical polar coordinate specifications of the detector positions conveniently relates to scattering angles and lengths for direct geometry instruments.

Individual components of the instrument (e.g. jaws) will have their own set of local axes (x,y,z) which will be fixed to their body in a way defined by their shape. These local axes will probably not coincidewith the global instrument axes and so a set of rotation angles will also need to be stored. For this an NXgeometry class is defined, along with NXtranslation and NXorientation; the hope is to provide ageneral enough method for relating the location of any object with respect to another object. The mechanism also allows for specifying one position relative to another component: a NeXus file link is made in one instance of an NXgeometry object to another NXgeometry object and a program can then traverse the chain of links to calculate an absolute position.

NeXus does not need to define absolutely where to place the *origin*. All components can instead bedeclared with a relative position that ultimately follows a chain back to one object; this will be named*origin1*, be of class NXgeometry and a member of NXentry. The real space location of this origin is chosen for convenience and should be mentioned in the description attached to *origin1*. If the origin is taken at the sample, then *sample.geometry.distance* will always be (0,0,0) relative to *origin1*; if the origin is taken elsewhere this will not be so, but everything will still work. It may be convenient to define extra origins (similar to *arms* in *McStas*) at other parts of the instrument. For example,defining one at the centre of a circular array of detectors would allow their positions to be conveniently specified in spherical polar coordinates. Another possibility would be to define the sample relative to*origin1* and the detectors to *origin2*; the detectors could then be rotated by a rotation of *origin2* without modifying NXdetector.

As well as specifying the component location, it is also necessary to specify the beam direction. Unless otherwise given in an NXbeam member of the component, the incident beam is assumed to be travelling along (0,0,+z) in the coordinate system of the object (or origin) our position was defined relative to. Thus, for a component with absolute positioning the beam will always be in the incident beam direction unless specified by an NXbeam member.

Size and Shape (NXshape)

! **Warning**
Need some editing at this point!

Many instrument components define *height* and *width* variables to specify their size when rectangular, a *radius* variable for when circular etc. Rather than all these different names, an alternative scheme is proposed based on the *shape* of the object and the local coordinate axes this shape defines. All object would just need to specify a shape (*cuboid*, *cylinder* etc.) and a size array. Specifying size[3] would give the dimensions of the object along its local (+-x,+-y,+-z) axes; specifying size[6] would give the extent along (+x,+y,+z,-x,-y,-z) and allow for e.g. asymmetric jaws where the reference point may not be the centre of the rectangle. For example take shape=*cylinder*: the NXtranslation variable of position would define the location of the reference point for the origin of the local axes: *z* in the direction of the cylinder axis, *x* and *y* in plane. With no rotation the object would be oriented with its local axes pointing in the direction of axes of the object it was defined relative to, but this can be altered with the NXorientation variable within position. If a size[3] array variable was specified, the reference point must be the centre of the cylinder and the dimension are size[0]=size[1]=radius, size[2]=length/2. If size[6] was specified then the reference point would be elsewhere in the object, with its distance from the cylinder edges along the various axes given byelements of the size[6] array.

2.2.4   NeXus units

placeholder

2.2.5   NeXus array dimensions

Here are a couple of examples to get this section started:

Example 2.17 Example of array dimensions.

```
1  <field name="data">
2    <dimensions size="3"/>
3    <!-- other definitions might appear -->
4  </field><field name="time-of-flight">
5    <dimensions size="1">
6      <dim index="1" ref="data" refindex="3" incr="1"/>
7    </dimensions>
8    <!-- other definitions might appear -->
9  </field>
```

In some programming language, this would make data[i,j,k] and time-of-flight[k+1].

2.2.6 NeXus Data Types

placeholder

2.3 NeXus Instruments

The NeXus Design page lists the group classes from which a NeXus file is constructed. They provide the glossary of items that could, in principle, be stored in a standard-conforming NeXus file (other items may be inserted into the file if the author wishes, but they won't be part of the standard). If you are going to include a particular piece of metadata, consult the class definitions to find out what to call it. However, to assist those writing data analysis software, it is useful to provide more than a glossary; it is important to define the required contents of NeXus files that contain data from particular classes of neutron, x-ray, or muon instrument.

As part of the NeXus standard, we have identified a number of generic instruments that describe an appreciable number of existing instruments around the world. Although not identical in every detail, they share enough common characteristics, and more importantly, they require sufficiently similar modes of data analysis, to make a standard description useful. They are in the process of being defined for the NeXus standard. The definitions will be in XML using the NeXus Definition Language (NXDL) format.

2.4 Frequently Asked Questions

This is a list of commonly asked questions concerning the NeXus data format.

1. *How many facilities use NeXus?*

   This is continually evolving. It has been used as the instrument format for several years on some or all instruments at a number of facilities including PSI (Switzerland), LLB (France), and LANSCE (USA). It will be used on all future instrumentation at ISIS (UK), NIST (USA), and ANSTO (Australia). Finally, it has been formally adopted by major facilities under construction, the SNS (USA), JPARC (Japan) and Diamond Light Source (UK).

2. *NeXus files are only useful for archiving instrumental data, aren't they?*

   NeXus files can be used to store both extremely simple data, e.g. a single (x,y) array, and highly complex instrument descriptions. In fact, the original intention of the NeXus data format was to provide a way of interchanging data between facilities and their user communities. However, the power of NeXus hierarchical design has led to its adoption as a standard archiving format by several major facilities, such as ISIS, LANSCE, and the SNS.

3. *Why aren't NXsample and NXmonitor groups stored in the NXinstrument group?*

   A NeXus file can contain a number of NXentry groups, which may represent different scans in an experiment, or sample and calibration runs, etc. In many cases, though by no means all, the instrument has the same configuration so that it would

be possible to save space by storing the NXinstrument group once and using multiple links in the remaining NXentry groups. It is assumed that the sample and monitor information would be more likely to change from run to run, and so should be stored at the top level.

4. *How do I identify the plottable data?*

   Any program whose aim is to identify plottable data should use the following procedure:

   (a) Open the first top level NeXus group with class NXentry.

   (b) Open the first NeXus group with class NXdata.

   (c) Loop through NeXus data items in this group searching for the item with attribute "signal" =1. This is the plottable data.

   (d) Check to see if this data item has an attribute called "axes". If so, the names are defined as a comma-delimited string within this attribute in the C-order of the data array, and you can skip the next two steps.

   (e) If the "axes" attribute is not defined, search for the one-dimensional NeXus data items with attribute "primary" = 1.

   (f) These are the dimension scales to label the axes of each dimension of the data.

   (g) Link each dimension scale to the respective data dimension by the "axis" attribute (= 1,2,...,rank of data).

   (h) If necessary, close the NXdata group, open the next one and repeat steps 3 to 6.

   (i) If necessary, close the NXentry group, open the next one and repeat steps 2 to 7.

   Consult the NeXus API section, which describes the routines available to program these operations. In the course of time, generic NeXus browsers will provide this functionality automatically.

5. *Why are the NeXus classes so complicated? I'll never store all that information*

   The NeXus classes are essentially glossaries of terms. If you need to store a piece of information, consult the class definitions to see if it has been defined. If so, use it. However, it is not compulsory to include every item that has been defined if it is not relevant to your experiment. On the other hand, if there is an NeXus definition for your instrument, you are recommended to include all the compulsory items if you want to use standard software to analyze your data.

6. *I want to produce an application definition. How do I go about it?*

   Read the NXDL Tutorial. If you encounter any problems because the classes are not sufficient to describe your configuration, please contact the NIAC Executive Secretary explaining the problem, and post a suggestion at the relevant class wiki page. The NIAC is always willing to consider proposals to amend the base classes. The procedures are defined in the NIAC constitution.

Note

We might copy the tutorial into this document or at least post the PDF on the WWW and link to it from here. The latter is probably the better option.

Chapter 3

Generic features of NeXus files

3.1 Links

This is a paragraph.

3.2 Linking data to detectors

This is a paragraph.

3.3 Monitors as a special kind of data

This is a paragraph.

Chapter 4

Verification and validation of files

4.1 Definitions of these terms

This is a paragraph.

4.2 Files implement multiple definitions

This is a paragraph.

4.3 Validation techniques

This is a paragraph.

Chapter 5

The NeXus Application-Program Interface

5.1   Description of the NeXus Application-Program Interface

blah blah

Chapter 6

The NeXus Definition Language

6.1 Description of the NeXus Definition Language

The intent of NXDL is to provide an easier, and rules-based method for defining a NeXus data file that is specific to either an instrument (where NeXus has been for years) or an area of scientific technique or analysis.

NeXus wiki page: NXDL[1]

NXDL is the new NeXus definition language - it will replace the Meta-DTD format[2] as the way for specifying the content of NeXus data files. NXDL is based on XML schema technology and follows on from provisional work on NeXus schema NeXus schema[3] in Summer 2008.

At NIAC 2008,[4] it was agreed that writing definitions directly as schema would probably be too verbose and instead a new language (NXDL) was devised that would then be translated into XML schema for eventual validation. The files written in the NXDL language are XML files in their own right and have an associated XML schema - this means a schema-aware XML editor can be used and accurate definition creation is much easier than it was with the old Meta-DTD.

For the latest NXDL definitions see the svn repository.[5]

An NXDL description will be a true (not pseudo) XML file which structure can be validated by a schema. See below for a draft example from the working repository. Since the NXDL specification is not complete, expect that some aspects of this example might change. NXDL is not intended to change the location of information stored in existing NeXus files, only to change (and simplify) the way the file would be arranged for a specific instance such as instrument or technique.

There are several different elements used in the NXDL. These are:

definition

> Root element of an NXDL specification. This is the basic definition of a new type. Components must derive from an existing type and optionally provide documentation.

group

> A group element refers to the definition of an existing NX object: either a base class (such as `NXdata` or `NXgeometry`), application definition, contributed definition, or a locally-defined component.

field

> A field declares a new element in the component being defined.

doc

> Items can be documented (but this documentation is not printed in the manual. It only appears in the NXDL file.)

---

[1]http://www.nexusformat.org/NXDL
[2]http://www.nexusformat.org/Metaformat
[3]http://www.nexusformat.org/Schema
[4]http://www.nexusformat.org/images/b/b2/NIAC2008_minutes.pdf
[5]https://svn.nexusformat.org/definitions/trunk

link

A link to another item.

attribute

attribute of `field`

Note
Application definition is the minimum set of data necessary for data analysis software.

6.1.1 `NXxrot.nxdl.xml`: example NXDL specification

```xml
1  <?xml version="1.0" encoding="UTF-8"?>
2  <!--
3  # NeXus - Neutron and X-ray Common Data Format
4  #
5  # Copyright (C) 2008 NeXus International Advisory Committee (NIAC)
6  #
7  # This library is free software; you can redistribute it and/or
8  # modify it under the terms of the GNU Lesser General Public
9  # License as published by the Free Software Foundation; either
10 # version 2 of the License, or (at your option) any later version.
11 #
12 # This library is distributed in the hope that it will be useful,
13 # but WITHOUT ANY WARRANTY; without even the implied warranty of
14 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the GNU
15 # Lesser General Public License for more details.
16 #
17 # You should have received a copy of the GNU Lesser General Public
18 # License along with this library; if not, write to the Free Software
19 # Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA   02111-1307   USA
20 #
21 # For further information, see http://www.nexusformat.org
22
23 -->
24 <definition name="NXxrot" extends="NXxbase" type="group"
25   xmlns="http://definition.nexusformat.org/nxdl/3.1"
26   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
27   xsi:schemaLocation="http://definition.nexusformat.org/nxdl/3.1 ../nxdl.xsd ">
28   <doc>This is the application definition for raw data from a rotation camera. It extends ↩
        NXxbase,
29     so the full definition is the content of NXxbase plus the data defined here.</doc>
30   <group type="NXentry" name="entry">
31     <field name="definition">
32       <doc> Official NeXus DTD or NXDL schema to which this file conforms </doc>
33       <enumeration>
34         <item value="NXxrot"/>
35       </enumeration>
36     </field>
37     <group type="NXinstrument" name="instrument">
38       <group type="NXdetector" name="detector">
39         <field name="polar_angle" type="NX_FLOAT" units="NX_ANGLE">
40           <doc>The polar_angle (two theta) where the detector is placed.</doc>
41         </field>
42       </group>
43     </group>
44     <group type="NXsample" name="sample">
45       <field name="rotation_angle" type="NX_FLOAT" units="NX_ANGLE"
46         axis="1">
```

```
47        <doc>
48          This is an array holding the sample rotation angle at each
49          scan point
50        </doc>
51        <dimensions size="1">
52          <dim index="1" value="np" />
53        </dimensions>
54      </field>
55    </group>
56    <group type="NXdata" name="name">
57      <link name="rotation_angle" target="/NXentry/NXsample/rotation_angle"></link>
58    </group>
59  </group>
60 </definition>
```

6.1.2 `nxdl.xsd` source code

```
1  <?xml version="1.0" encoding="UTF-8"?>
2  <!--
3  ########### SVN repository information ###################
4  # $LastChangedDate: 2009-10-18 15:01:57 -0500 (Sun, 18 Oct 2009) $
5  # $LastChangedBy: Pete Jemian $
6  # $LastChangedRevision: 410 $
7  # $HeadURL: https://svn.nexusformat.org/definitions/trunk/nxdl.xsd $
8  ########### SVN repository information ###################
9  -->
10
11 <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
12   targetNamespace="http://definition.nexusformat.org/nxdl/3.1"
13   xmlns:nx="http://definition.nexusformat.org/nxdl/3.1"
14   version="$Id: nxdl.xsd 410 2009-10-18 20:01:57Z Pete Jemian $"
15   elementFormDefault="qualified">
16
17   <xs:annotation>
18     <xs:documentation>
19       # NeXus - Neutron and X-ray Common Data Format
20       #
21       # Copyright (C) 2008 NeXus International Advisory Committee (NIAC)
22       #
23       # This library is free software; you can redistribute it and/or
24       # modify it under the terms of the GNU Lesser General Public
25       # License as published by the Free Software Foundation; either
26       # version 2 of the License, or (at your option) any later version.
27       #
28       # This library is distributed in the hope that it will be useful,
29       # but WITHOUT ANY WARRANTY; without even the implied warranty of
30       # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the GNU
31       # Lesser General Public License for more details.
32       #
33       # You should have received a copy of the GNU Lesser General Public
34       # License along with this library; if not, write to the Free Software
35       # Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA  02111-1307  USA
36       #
37       # For further information, see http://www.nexusformat.org
38     </xs:documentation>
39   </xs:annotation>
40
41   <!-- ++++++++++++++++++++++++++++++++++++++++++++++++++++++++ -->
42
43   <xs:include schemaLocation="nxdlTypes.xsd">
```

```
44      <xs:annotation>
45        <xs:documentation>
46          Definitions of the basic data types and unit types
47          allowed in NXDL instance files.
48        </xs:documentation>
49      </xs:annotation>
50    </xs:include>
51
52    <!-- ++++++++++++++++++++++++++++++++++++++++++++++++++++++ -->
53
54    <!-- define the document root element -->
55    <xs:element name="definition" type="nx:definitionType" />
56
57    <!-- ++++++++++++++++++++++++++++++++++++++++++++++++++++++ -->
58
59    <xs:complexType name="definitionType">
60      <xs:group ref="nx:groupGroup" minOccurs="0" maxOccurs="unbounded" />
61      <xs:attribute name="name" use="required" />
62      <xs:attribute name="type" use="required" type="nx:definitionTypeAttr" />
63      <xs:attribute name="extends" use="optional" />
64      <xs:attribute name="restricts" use="optional" />
65      <xs:attribute name="svnid" use="optional"/>
66    </xs:complexType>
67
68    <xs:complexType name="attributeType">
69      <xs:annotation>
70        <xs:documentation>
71          A new element may expect or require some attributes.
72        </xs:documentation>
73      </xs:annotation>
74      <xs:sequence>
75        <xs:element name="doc" type="nx:basicType" minOccurs="0" />
76        <xs:element name="enumeration" type="nx:enumerationType"
77          minOccurs="0">
78          <xs:annotation>
79            <xs:documentation>An enumeration specifies the values to be used.</ ←
                 xs:documentation>
80          </xs:annotation>
81        </xs:element>
82      </xs:sequence>
83      <xs:attribute name="name" use="required" />
84      <xs:attribute name="type" type="nx:primitiveType" default="NX_CHAR"/>
85    </xs:complexType>
86
87    <xs:simpleType name="definitionTypeAttr">
88      <xs:restriction base="xs:string">
89        <xs:enumeration value="group" />
90        <xs:enumeration value="definition" />
91      </xs:restriction>
92    </xs:simpleType>
93
94    <xs:group name="groupGroup">
95      <xs:sequence>
96        <xs:element name="doc" type="nx:basicType" minOccurs="0"
97          maxOccurs="1" />
98        <xs:element name="attribute" type="nx:attributeType"
99          minOccurs="0" maxOccurs="unbounded" />
100       <xs:element name="group" type="nx:groupType" minOccurs="0"
101         maxOccurs="unbounded" />
102       <xs:element name="field" type="nx:fieldType" minOccurs="0"
103         maxOccurs="unbounded" />
104       <xs:element name="link" type="nx:linkType" minOccurs="0"
```

```
105            maxOccurs="unbounded" />
106      </xs:sequence>
107   </xs:group>
108
109   <!-- ++++++++++++++++++++++++++++++++++++++++++++++++++++ -->
110
111   <xs:complexType name="basicComponent">
112      <xs:sequence>
113         <xs:element name="doc" type="nx:basicType" minOccurs="0" maxOccurs="1" />
114      </xs:sequence>
115      <xs:attribute name="name" use="required" />
116   </xs:complexType>
117
118   <!-- ++++++++++++++++++++++++++++++++++++++++++++++++++++ -->
119
120   <xs:simpleType name="basicType">
121      <xs:annotation>
122         <xs:documentation>
123            This is the basic definition of a new type.
124            Components must derive from an existing type
125            and optionally provide documentation.
126         </xs:documentation>
127      </xs:annotation>
128      <xs:restriction base="xs:string"></xs:restriction>
129   </xs:simpleType>
130
131   <!-- ++++++++++++++++++++++++++++++++++++++++++++++++++++ -->
132
133   <xs:complexType name="groupType">
134      <xs:annotation>
135         <xs:documentation>
136            A group element refers to the definition of
137            an existing NX object or a locally-defined component.
138         </xs:documentation>
139      </xs:annotation>
140      <xs:group ref="nx:groupGroup" minOccurs="0" maxOccurs="unbounded"/>
141      <xs:attribute name="type" use="required">
142         <xs:simpleType>
143            <xs:restriction base="xs:string">
144               <xs:pattern value="NX.+"/>
145            </xs:restriction>
146         </xs:simpleType>
147      </xs:attribute>
148      <xs:attribute name="minOccurs" use="optional" default="0" type="xs:nonNegativeInteger"/ ↩
          >
149      <xs:attribute name="maxOccurs" use="optional"  default="unbounded"/>
150      <xs:attribute name="name" use="optional" />
151   </xs:complexType>
152
153   <!-- ++++++++++++++++++++++++++++++++++++++++++++++++++++ -->
154
155   <xs:complexType name="dimsType">
156      <!--
157 Peter Peterson writes:
158      Here is a model:
159      <field name="data">
160         <dimensions size="3"/>
161         ...
162      </field><field name="time-of-flight">
163         <dimensions size="1">
164            <dim index="1" ref="data" refindex="3" incr="1"/>
165         </dimensions>
```

```
166          ...
167        </field>
168
169        In old terms this would make data[i,j,k] and time-of-flight[k+1].
170        We need to get to the stage where people do not
171        need to read a manual to understand an existing NXDL.
172
173  +++++++++++++++++++++++++++++++++++
174  then Freddie Akeroyd writes:
175  I think we have three cases to consider here:
176  (1) specifying an explicit value for a dimension
177  (2) specifying a dimension to be the same size as that of an array
178  already defined elsewhere (optionally +- a value)
179  (3) specifying a dimension to be the same as a data value defined
180  elsewhere (e.g. num_scan_points, optionally +- a value)
181
182  For (1) how about
183
184  <dim index="1" value="3" />
185
186  For (2) how about
187
188  <field name="data">
189  <dimensions size="3">
190  <dim index="1" label="num_time_channels"/>
191  < !- - dimensions with no constraints or labels do not need to be mentioned - -  >
192  </dimensions>
193  </field>
194  <field name="time_of_flight">
195  <dimensions size="1">
196  <dim index="1" ref="num_time_channels" incr="1"/>
197  </dimensions>
198  ...
199  </field>
200
201  For (3) how about
202
203  <field name="num_scan_points" type="NX_INT"
204  label="nscanpt" />
205  <field name="data">
206  <dimensions size="3">
207  <dim index="1" label="num_time_channels"/>
208  <dim index="3" ref="nscanpt"/>
209  </dimensions>
210  </field>
211
212  Notes on dim tag:
213
214  use 1 for first element in a sequence, +- to indicate sequence direction
215  (or we could use an attribute instead)
216
217  <dim index="1">   ! refers to first index (fastest varying)
218  <dim index="-1">  ! refers to last index (slowest varying)
219
220  <dim label="i">   ! we should constrain labels to be unique within the
221  file and thus could be referred to
222  from any location. This would mean that "i"
223  could not be used everywhere as a label,
224  but forcing the use of better names like
225  "num_time_channels" makes things clearer. In the
226  first instance by making label an xs:ID and
227  ref an xs:IDREF we can enforce "something" in
```

```
228  label="something" to always be unique and "some_ref" in ref="some_ref"
229  to always point to a
230  valid "label" (We may later want to use xs:key
231  / xs:keyref to do this if we needed a second
232  set of unique labels for another purpose)
233
234  +++++++++++++++++++++++++++++++++++
235  then Mark Koennecke writes:
236  I think the dimensions encoding is tightly integrated with the algorithm we
237  use to validate them later on. I thought about something like:
238
239  <dimensions size="3">
240  <dim level="0" value="25"/>
241  <dim level="1" value="np"/>
242  <dim level="2" path="/entry/blablabla/somefield" dim="0"/>
243  </dimensions>
244  The first case is trivial: validate against a number
245
246  The general idea is that the dimensions validator would initialise np
247  the first place it comes up against it and
248  checks against the value ever after. We must allow expressions like np+1
249  here.
250
251  The third syntax is to explicitly address a specific dimension of
252  another variable.
253
254      -->
255      <xs:annotation>
256        <xs:documentation>dimensions of a data element in a NeXus file</xs:documentation>
257      </xs:annotation>
258      <xs:sequence>
259        <xs:element name="dim" minOccurs="0" maxOccurs="unbounded">
260          <xs:complexType>
261            <xs:attribute name="index" type="nx:NX_CHAR">
262              <xs:annotation>
263                <xs:documentation>Number indicating which axis (subscript) is
264                  being described, ranging from 1 up to "size" (rank of the
265                  data structure).  For example, given an array A[i,j,k],
266                  index="1" would refer to the "i" axis (subscript).</xs:documentation>
267              </xs:annotation>
268            </xs:attribute>
269            <xs:attribute name="value" type="nx:NX_CHAR">
270              <xs:annotation>
271                <xs:documentation>Length (number of values) of this axis.</xs:documentation>
272              </xs:annotation>
273            </xs:attribute>
274            <xs:attribute name="ref" type="nx:NX_CHAR">
275              <xs:annotation>
276                <xs:documentation>The dimension specification is the same as
277                  that in the "ref" field, specified either by a relative path,
278                  such as "polar_angle" or "../Qvec" or absolute path, such as
279                  "/entry/path/to/follow/to/ref/field".
280                </xs:documentation>
281              </xs:annotation>
282            </xs:attribute>
283            <xs:attribute name="refindex" type="nx:NX_CHAR">
284              <xs:annotation>
285                <xs:documentation>The dimension specification is the same as
286                  the "refindex" axis within the "ref" field.
287                  Requires "ref" attribute to be present.</xs:documentation>
288              </xs:annotation>
289            </xs:attribute>
```

```
290    <xs:attribute name="incr" type="nx:NX_CHAR">
291      <xs:annotation>
292        <xs:documentation>The dimension specification is related to
293          the "refindex" axis within the "ref" field by an
294          offset of "incr."  Requires "ref" and "refindex"
295          attributes to be present.</xs:documentation>
296      </xs:annotation>
297    </xs:attribute>
298  </xs:complexType>
299  </xs:element>
300  </xs:sequence>
301  <xs:attribute name="size" type="nx:NX_CHAR">
302    <xs:annotation>
303      <xs:documentation>Rank (number of dimensions) of the data structure.
304        For example: a[5] has size="1" while b[8,5,6,4] has size="4".
305        See http://en.wikipedia.org/wiki/Rank_(computer_programming)
306        for more details.</xs:documentation>
307    </xs:annotation>
308  </xs:attribute>
309  </xs:complexType>
310
311  <xs:complexType name="linkType">
312    <xs:complexContent>
313      <xs:annotation>
314        <xs:documentation>A link to another item.</xs:documentation>
315      </xs:annotation>
316      <xs:extension base="nx:basicComponent">
317        <xs:attribute name="target" use="required">
318          <xs:annotation>
319            <xs:documentation>path to element we link to</xs:documentation>
320          </xs:annotation>
321        </xs:attribute>
322      </xs:extension>
323    </xs:complexContent>
324  </xs:complexType>
325
326  <xs:complexType name="fieldType">
327    <xs:complexContent>
328      <xs:annotation>
329        <xs:documentation>A field declares a new element
330        in the component being defined.</xs:documentation>
331      </xs:annotation>
332      <xs:extension base="nx:basicComponent">
333        <xs:sequence>
334          <xs:element name="dimensions" type="nx:dimsType"
335            minOccurs="0" maxOccurs="1">
336            <xs:annotation>
337              <xs:documentation>dimensions of a data element
338              in a NeXus file</xs:documentation>
339            </xs:annotation>
340          </xs:element>
341          <xs:element name="attribute" type="nx:attributeType"
342            minOccurs="0" maxOccurs="unbounded">
343            <xs:annotation>
344              <xs:documentation>attributes of field</xs:documentation>
345            </xs:annotation>
346          </xs:element>
347          <xs:element name="enumeration" type="nx:enumerationType"
348            minOccurs="0">
349            <xs:annotation>
350              <xs:documentation>A field can specify which
351              values are to be used</xs:documentation>
```

```
352            </xs:annotation>
353          </xs:element>
354        </xs:sequence>
355        <xs:attribute name="units" type="nx:anyUnitsAttr">
356          <xs:annotation>
357            <xs:documentation>
358              String describing the engineering units.
359              The string should be appropriate for the value
360              and should conform to the NeXus rules for units.
361              Can conformance be validated or ensured?
362            </xs:documentation>
363          </xs:annotation>
364        </xs:attribute>
365        <xs:attribute name="signal" type="nx:NX_POSINT">
366          <xs:annotation>
367            <xs:documentation>
368              Presence of the signal attribute means this field is an ordinate.
369
370              Integer marking this field as plottable data (ordinates).
371              The value indicates the priority of selection or interest.
372              Some facilities only use signal="1"
373              while others use signal="2" to indicate
374              plottable data of secondary interest.
375              Higher numbers are possible but not common
376              and interpretation is not standard.
377
378              A field with a signal attribute should not have an axis attribute.
379            </xs:documentation>
380          </xs:annotation>
381        </xs:attribute>
382        <xs:attribute name="axes" type="nx:NX_CHAR">
383          <xs:annotation>
384            <xs:documentation>
385              Presence of the axes attribute means this field is an ordinate.
386
387              This attribute contains a white-space separated list
388              of paths to the names of independent axes when plotting this field.
389            </xs:documentation>
390          </xs:annotation>
391        </xs:attribute>
392        <xs:attribute name="axis" type="nx:NX_POSINT">
393          <xs:annotation>
394            <xs:documentation>
395              NOTE: Use of this attribute is discouraged.  It is for legacy support.
396              You should use the axes attribute instead.
397
398              Presence of the axis attribute means this field is an abcissa.
399
400              Integer marking this
401              field as an axis that is part of the data set.
402              The data set is a field with the attribute
403              signal="1" in the same group.
404              The value can range from 1 up to the number of
405              independent axes (abcissae) in the data set.
406
407              A value of axis="1" indicates that this field
408              contains the data for the first independent axis.
409              For example, the X axis in an XY data set.
410
411              A value of axis="2" indicates that this field
412              contains the data for the second independent axis.
413              For example, the Y axis in a 2-D data set.
```

```
414
415              A value of axis="3" indicates that this field
416              contains the data for the third independent axis.
417              For example, the Z axis in a 3-D data set.
418
419              A field with an axis attribute should not have a signal attribute.
420            </xs:documentation>
421          </xs:annotation>
422        </xs:attribute>
423        <xs:attribute name="primary" type="nx:NX_POSINT">
424          <xs:annotation>
425            <xs:documentation>
426              Integer indicating the priority of selection
427              of this field for plotting (or visualization) as an axis.
428
429              Presence of the primary attribute means this field is an abcissa.
430            </xs:documentation>
431          </xs:annotation>
432        </xs:attribute>
433        <xs:attribute name="type" type="nx:primitiveType" default="NX_CHAR"/>
434        <xs:attribute name="minOccurs" use="optional" default="0">
435          <xs:simpleType>
436            <xs:restriction base="xs:nonNegativeInteger">
437              <xs:enumeration value="0"/>
438              <xs:enumeration value="1"/>
439            </xs:restriction>
440          </xs:simpleType>
441        </xs:attribute>
442        <xs:attribute name="maxOccurs" use="optional"  default="1">
443          <xs:simpleType>
444            <xs:restriction base="xs:nonNegativeInteger">
445              <xs:enumeration value="1"/>
446              <!--
447                The "unbounded" is not valid for nonNegativeInteger
448                But that is what we really mean.
449              -->
450              <xs:enumeration value="99999999999999999999999"/>
451            </xs:restriction>
452          </xs:simpleType>
453        </xs:attribute>
454      </xs:extension>
455    </xs:complexContent>
456  </xs:complexType>
457
458  <xs:complexType name="enumerationType">
459    <xs:sequence>
460      <xs:element name="item" type="nx:enumItemType" minOccurs="1" maxOccurs="unbounded"/>
461    </xs:sequence>
462  </xs:complexType>
463
464  <xs:complexType name="enumItemType">
465    <xs:sequence>
466      <xs:element name="doc" type="nx:basicType" minOccurs="0" maxOccurs="1">
467        <xs:annotation>
468          <xs:documentation>
469            Items can be documented (but this documentation is not
470            printed in the manual.  It only appears in the NXDL file.)
471          </xs:documentation>
472        </xs:annotation>
473      </xs:element>
474    </xs:sequence>
475    <xs:attribute name="value" use="required"/>
```

```
476      </xs:complexType>
477
478  </xs:schema>
```

Chapter 7

Summation and summary

blah blah

Appendix A

NeXus classes

A.1 Overview of NeXus classes

Note

This section intro could be rewritten in the breezy style of the NeXus wiki page: http://www.nexusformat.org/Design

---

Note

Another suggestion is to move the Design section to this location.

Each of the NeXus classes is described in two basic ways. First, a short list of descriptive information is provided as a header, then a table summarizing the fields and groups that comprise the NeXus class is given.

SVN Id Subversion repository checkout identification, stripped of the surrounding dollar signs. (The *Id* is blank on files copied direct from the repository that are not checked out by a subversion client.)

NXDL source Name of the NeXus class and a URL to the source listing in the NeXus subversion repository.

NeXus Definition Language The NeXus Definition Language (NXDL) (described in a separate chapter of this manual) is used to describe the components in the NeXus Base Classes, as well as application and contributed definitions. The intent of NXDL is to provide a rules-based method for defining a NeXus data file that is specific to either an instrument (where NeXus has been for years) or an area of scientific technique or analysis. NXDL replaces the meta-DTD method used previously to define the NeXus base classes.

extends class NeXus class extended by this class. Most NeXus base classes only extend the base class definition (NXDL).

other classes included List (including URLs) of other classes used to define this class.

documentation Description of the NeXus class. No markup or formatting is allowed.

The table has columns to describe the basic information about each field or group in the class. An example of the varieties of specifications are given in the following table using items found in various NeXus base classes.

Table A.1: Example Tabular representation of a NeXus class

Name	Type	Units	Description (and Occurences)
program_name	NX_CHAR		Name of program used to generate this file
@version	NX_CHAR		Program version number Occurences: 1 : *default*

Table A.1: (continued)

Name	Type	Units	Description (and Occurences)
@configuration	NX_CHAR		configuration of the program
thumbnail	NXnote		A small image that is representative of the entry. An example of this is a 640x480 JPEG image automatically produced by a low resolution plot of the NXdata.
@mime_type	NX_CHAR		expected: **mime_type="image/*"**
	NXgeometry		describe the geometry of this class
distance	NX_FLOAT	NX_LENGTH	Distance from sample
mode	"Single Bunch" | "Multi Bunch"		source operating mode
target_material	Ta | W | depleted_U | enriched_U | Hg | Pb | C		Pulsed source target material

In the above example, the fields might appear in a NeXus XML data file as

Example A.1 Example fragment of a NeXus XML data file

```
1  <program_name version="1.0a" configuration="standard">
2      MaxSAS
3  </program_name>
4  <NXnote name="thumbnail" mime_type="image/*">
5      <!-- contents of an NXnote would appear here -->
6  </NXnote>
7  <distance units="mm">125.6</distance>
8  <mode> Single Bunch </mode>
9  <target_material>depleted_U</target_material>
```

The columns in the table are described as follows:

Name Short name of the data field. Name must satisfy both HDF and XML naming rules.

Attributes Additional metadata used to define this variable. Attributes are identified with a leading "at" symbol (@) and belong with the preceding field or group. In the example above, the program_name element has two attributes: version (required) and configuration (optional) while the thumbnail element has one attribute: mime_type (optional).

Type Type of data to be represented by this variable. The type is one of those specified in the NeXus Definition Language.

> ⚠️ **Warning**
> put the URL here
> NXDL needs some documentation

In the case where the variable can take only one value from a known list, the list of known values is presented, such as: [Beryllium | "Pyrolytic Graphite" | Graphite | Sapphire | Silicon | Supermirror]. Selections with included whitespace are surrounded by quotes. See the example above for usage.

Units Type of engineering units to be used. The unit types are those allowed by the NeXus Definition Language.

> **!** **Warning**
> THIS NEEDS REAL WORK HERE

Description A simple text description of the data field. No markup or formatting is allowed.

Occurences The absence of "Occurences" in the item description signifies that both `minOccurs` and `maxOccurs` have the default values.If the number of occurences of an item are specified in the NXDL (through `@minOccurs` and `@maxOccurs` attributes), they will be reported in the Description column similar to the example shown above.

Default values for occurences are shown in the following table. The `NXDL element type` is either a group (such as a NeXus base class), a field (that specifies the name and type of a variable), or an attribute of a field or group. The number of times an item can appear ranges between `minOccurs` and `maxOccurs`. A default `minOccurs` of zero means the item is optional. For attributes, `maxOccurs` cannot be greater than 1.

Table A.2: Default values for occurences

NXDL element type	minOccurs	maxOccurs
group	0	unbounded
field	0	unbounded
attribute	0	1

A.2 NeXus Base Classes

A description of each NeXus Base Class is given.

A.2.1 NXaperture

SVN Id Id: NXaperture.nxdl.xml 373 2009-09-22 16:57:49Z Freddie Akeroyd

NXDL source: NXaperture

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXaperture.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXgeometry

documentation Template of a beamline aperture.

Table A.3: Tabular representation of NXaperture[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
	NXgeometry		location and shape of aperture
material	NX_CHAR		Absorbing material of the aperture
description	NX_CHAR		Description of aperture

A.2.2 NXattenuator

SVN Id Id: NXattenuator.nxdl.xml 373 2009-09-22 16:57:49Z Freddie Akeroyd

NXDL source: NXattenuator

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXattenuator.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: no included classes

documentation Template of a beamline attenuator.

Table A.4: Tabular representation of NXattenuator[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
distance	NX_FLOAT	NX_LENGTH	Distance from sample
type	NX_CHAR		Type of attenuator, e.g. polythene
thickness	NX_FLOAT	NX_LENGTH	Thickness of attenuator along beam direction
scattering_cross-_section	NX_FLOAT	NX_CROSS_SECTION	Scattering cross section (coherent+incoherent)
absorption_cross-_section	NX_FLOAT	NX_CROSS_SECTION	Absorption cross section
attenuator_trans-mission	NX_FLOAT	NX_DIMENSIONLESS	The nominal amount of the beam that gets through (transmitted intensity)/(incident intensity)

A.2.3 NXbeam

SVN Id Id: NXbeam.nxdl.xml 411 2009-10-18 21:36:40Z Pete Jemian

NXDL source: NXbeam

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXbeam.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXdata

documentation Template of the state of the neutron or X-ray beam at any location. It will be referenced by beamline component groups within the NXinstrument group or by the NXsample group. Note that variables such as the incident energy could be scalar values or arrays. This group is especially valuable in storing the results of instrument simulations in which it is useful to specify the beam profile, time distribution etc. at each beamline component. Otherwise, its most likely use is in the NXsample group in which it defines the results of the neutron scattering by the sample, e.g., energy transfer, polarizations.

Table A.5: Tabular representation of NXbeam[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
distance	NX_FLOAT	NX_LENGTH	Distance from sample
incident_energy	NX_FLOAT	NX_ENERGY	Energy on entering beamline component Dimensions: size="1" • dim: index="1" value="i" value=""
final_energy	NX_FLOAT	NX_ENERGY	Energy on leaving beamline component Dimensions: size="1" • dim: index="1" value="i" value=""
energy_transfer	NX_FLOAT	NX_ENERGY	Energy change caused by beamline component Dimensions: size="1" • dim: index="1" value="i" value=""
incident_wavelen-gth	NX_FLOAT	NX_WAVELENGTH	Wavelength on entering beamline component Dimensions: size="1" • dim: index="1" value="i" value=""
incident_wavelen-gth_spread	NX_FLOAT	NX_WAVELENGTH	Wavelength spread FWHM on entering component Dimensions: size="1" • dim: index="1" value="i" value=""
incident_beam_di-vergence	NX_FLOAT	NX_ANGLE	Divergence of beam entering this component Dimensions: size="2" • dim: index="1" value="2" value="" • dim: index="2" value="j" value=""
final_wavelength	NX_FLOAT	NX_WAVELENGTH	Wavelength on leaving beamline component Dimensions: size="1" • dim: index="1" value="i" value=""
incident_polariz-ation	NX_FLOAT	NX_ANY	Polarization vector on entering beamline component Dimensions: size="2" • dim: index="1" value="2" value="" • dim: index="2" value="j" value=""
final_polarizati-on	NX_FLOAT	NX_ANY	Polarization vector on leaving beamline component Dimensions: size="2" • dim: index="1" value="2" value="" • dim: index="2" value="j" value=""
final_wavelength-_spread	NX_FLOAT	NX_WAVELENGTH	Wavelength spread FWHM of beam leaving this component Dimensions: size="1" • dim: index="1" value="i" value=""
final_beam_diver-gence	NX_FLOAT	NX_ANGLE	Divergence FWHM of beam leaving this component Dimensions: size="2" • dim: index="1" value="2" value="" • dim: index="2" value="j" value=""
flux	NX_FLOAT	NX_FLUX	flux incident on beam plane area Dimensions: size="1" • dim: index="1" value="i" value=""

Table A.5: (continued)

Name and Attributes	Type	Units	Description (and Occurrences)
	NXdata		Distribution of beam with respect to relevant variable e.g. wavelength. This is mainly useful for simulations which need to store plottable information at each beamline component.

A.2.4 NXbeam_stop

SVN Id Id: NXbeam_stop.nxdl.xml 373 2009-09-22 16:57:49Z Freddie Akeroyd

NXDL source: NXbeam_stop

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXbeam_stop.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXgeometry

documentation A class for a beamstop. Beamstops and their positions are important for SANS and SAXS experiments.

Table A.6: Tabular representation of NXbeam_stop[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
	NXgeometry		engineering shape, orientation and position of the beam stop.
description	circular | rectangular		description of beamstop
size	NX_FLOAT	NX_LENGTH	size of beamstop
x	NX_FLOAT	NX_LENGTH	x position of the beamstop in relation to the detector
y	NX_FLOAT	NX_LENGTH	y position of the beamstop in relation to the detector
distance_to_dete-ctor	NX_FLOAT	NX_LENGTH	distance of the beamstop to the detector
status	in | out		

A.2.5 NXbending_magnet

SVN Id Id: NXbending_magnet.nxdl.xml 373 2009-09-22 16:57:49Z Freddie Akeroyd

NXDL source: NXbending_magnet

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXbending_magnet.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXdata
    NXgeometry

documentation description for a bending magnet

Table A.7: Tabular representation of NXbending_magnet[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
critical_energy	NX_FLOAT	NX_ENERGY	
bending_radius	NX_FLOAT	NX_LENGTH	
spectrum	NXdata		bending magnet spectrum
	NXgeometry		Engineering" position of bending magnet

A.2.6 NXcharacterization

SVN Id Id: NXcharacterization.nxdl.xml 373 2009-09-22 16:57:49Z Freddie Akeroyd

NXDL source: NXcharacterization

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXcharacterization.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: no included classes

documentation Template of the top-level NeXus group which contains all the data and associated information that comprise a single measurement. It is mandatory that there is at least one group of this type in the NeXus file.

Table A.8: Tabular representation of NXcharacterization[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
definition	NX_CHAR		
@version	NX_CHAR		
@URL	NX_CHAR		

A.2.7 NXcollimator

SVN Id Id: NXcollimator.nxdl.xml 373 2009-09-22 16:57:49Z Freddie Akeroyd

NXDL source: NXcollimator

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXcollimator.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXgeometry
    NXlog

documentation Template of a beamline collimator.

Table A.9: Tabular representation of NXcollimator[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
	NXgeometry		position, shape and size

Table A.9: (continued)

Name and Attributes	Type	Units	Description (and Occurrences)
type	Soller | radial | oscillating | honeycomb		
soller_angle	NX_FLOAT	NX_ANGLE	Angular divergence of Soller collimator
divergence_x	NX_FLOAT	NX_ANGLE	divergence of collimator in local x direction
divergence_y	NX_FLOAT	NX_ANGLE	divergence of collimator in local y direction
frequency	NX_FLOAT	NX_FREQUENCY	Frequency of oscillating collimator
frequency_log	NXlog		Log of frequency
blade_thickness	NX_FLOAT	NX_LENGTH	blade thickness
blade_spacing	NX_FLOAT	NX_LENGTH	blade spacing
absorbing_materi-al	NX_CHAR		name of absorbing material
transmitting_mat-erial	NX_CHAR		name of transmitting material

A.2.8 NXcrystal

SVN Id Id: NXcrystal.nxdl.xml 427 2009-10-19 00:06:46Z Pete Jemian

NXDL source: NXcrystal

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXcrystal.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXdata

> NXgeometry

> NXlog

documentation Template of a crystal monochromator or analyzer. Permits double bent monochromator comprised of multiple segments with anisotropic Gaussian mosaic. If curvatures are set to zero or are absent, array is considered to be flat. Scattering vector is perpendicular to surface. Crystal is oriented parallel to beam incident on crystal before rotation, and lies in vertical plane.

Table A.10: Tabular representation of NXcrystal[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
	NXgeometry		Position of crystal
type	PG | Ge | Si | Cu | Fe3Si | CoFe | Cu2MnAl | Multilayer | Diamond		Material of monochromating substance. Use the "reflection" field to indicate the (hkl) orientation. Use the "d_spacing" field to record the lattice plane spacing.
cut_angle	NX_FLOAT	NX_ANGLE	Cut angle of reflecting Bragg plane and plane of crystal surface

Table A.10: (continued)

Name and Attributes	Type	Units	Description (and Occurrences)
unit_cell	NX_FLOAT	NX_LENGTH	Unit cell parameters (lengths and angles) Dimensions: size="2" • dim: index="1" value="n_comp" value="" • dim: index="2" value="6" value=""
unit_cell_volume	NX_FLOAT	NX_VOLUME	Volume of the unit cell
orientation_matr-ix	NX_FLOAT		Orientation matrix of single crystal sample using Busing-Levy convention Dimensions: size="2" • dim: value="3" value="" • dim: value="3" value=""
wavelength	NX_FLOAT	NX_WAVELENGTH	Optimum diffracted wavelength Dimensions: apparent size="1" • dim: value="i" value=""
d_spacing	NX_FLOAT	NX_LENGTH	spacing between crystal planes of the reflection
scattering_vector	NX_FLOAT	NX_WAVENUMBER	Scattering vector, Q, of nominal reflection
reflection	NX_INT	NX_UNITLESS	Miller indices (hkl) values of nominal reflection Dimensions: apparent size="1" • dim: value="3" value=""
segment_width	NX_FLOAT	NX_LENGTH	Horizontal width of individual segment
segment_height	NX_FLOAT	NX_LENGTH	Vertical height of individual segment
segment_thickness	NX_FLOAT	NX_LENGTH	Thickness of individual segment
segment_gap	NX_FLOAT	NX_LENGTH	Typical gap between adjacent segments
segment_columns	NX_FLOAT	NX_LENGTH	number of segment columns in horizontal direction
segment_rows	NX_FLOAT	NX_LENGTH	number of segment rows in vertical direction
mosaic_horizontal	NX_FLOAT	NX_ANGLE	horizontal mosaic Full Width Half Maximum
mosaic_vertical	NX_FLOAT	NX_ANGLE	vertical mosaic Full Width Half Maximum
curvature_horizo-ntal	NX_FLOAT	NX_ANGLE	Horizontal curvature of focusing crystal
curvature_vertic-al	NX_FLOAT	NX_ANGLE	Vertical curvature of focusing crystal
polar_angle	NX_FLOAT	NX_ANGLE	Polar (scattering) angle at which crystal assembly is positioned. Note: some instrument geometries call this term 2theta. Dimensions: apparent size="1" • dim: ref="wavelength" value=""
azimuthal_angle	NX_FLOAT	NX_ANGLE	Azimuthal angle at which crystal assembly is positioned Dimensions: apparent size="1" • dim: ref="wavelength" value=""
bragg_angle	NX_FLOAT	NX_ANGLE	Bragg angle of nominal reflection Dimensions: apparent size="1" • dim: ref="wavelength" value=""
temperature	NX_FLOAT	NX_TEMPERATURE	Average/nominal crystal temperature
temperature_log	NXlog		log file of crystal temperature
reflectivity	NXdata		crystal reflectivity versus wavelength
transmission	NXdata		crystal transmission versus wavelength

A.2.9 NXdata

SVN Id Id: NXdata.nxdl.xml 409 2009-10-18 19:50:56Z Pete Jemian

NXDL source: NXdata

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXdata.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: no included classes

documentation Template of plottable data and their dimension scales. It is mandatory that there is at least one group of this type in each NXentry group. Note that "variable" and "data" can be defined with different names. The "signal" and "axes" attribute of the "data" item define which items are plottable data and which are dimension scales.

Table A.11: Tabular representation of NXdata[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
variable @long_name @distribution @first_good @last_good @axis	NX_NUMBER NX_CHAR NX_BOOLEAN NX_INT NX_INT NX_POSINT		Dimension scale defining an axis of the data Dimensions: size="1" • dim: index="1" value="i" value="" Axis label 0|false: single value, 1:true: multiple values Index of first good value Index of last good value Index (positive integer) identifying this specific set of numbers
variable_errors	NX_NUMBER		Errors associated with axis "variable" Dimensions: size="1" • dim: index="1" value="i" value=""
data @signal @axes @long_name	NX_NUMBER NX_INT NX_CHAR NX_CHAR		Data values Dimensions: size="1" • dim: index="1" value="i" value="" plottable (independent) axis, indicate index number names of dependent axes data label
errors	NX_NUMBER		Standard deviations of data values - the data array is identified by the attribute signal="1". This array must have the same dimensions as the data Dimensions: size="1" • dim: index="1" value="i" value=""
scaling_factor	NX_FLOAT		The elements in data are usually float values really. For efficiency reasons these are usually stored as integers after scaling with a scale factor. This value is the scale factor. It is required to get the actual physical value, when necessary.
offset	NX_FLOAT		An optional offset to apply to the values in data.
x	NX_FLOAT	NX_ANY	This is an array holding the values to use for the x-axis of data. The units must be appropriate for the measurement. Dimensions: size="1" • dim: index="1" value="nx" value=""

Table A.11: (continued)

Name and Attributes	Type	Units	Description (and Occurrences)
y	NX_FLOAT	NX_ANY	This is an array holding the values to use for the y-axis of data. The units must be appropriate for the measurement. Dimensions: size="1" • dim: index="1" value="ny" value=""
z	NX_FLOAT	NX_ANY	This is an array holding the values to use for the z-axis of data. The units must be appropriate for the measurement. Dimensions: size="1" • dim: index="1" value="nz" value=""

A.2.10 NXdetector

SVN Id Id: NXdetector.nxdl.xml 406 2009-10-18 17:12:29Z Pete Jemian

NXDL source: NXdetector

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXdetector.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXcharacterization

> NXdata
>
> NXgeometry
>
> NXnote

documentation Template of a detector, detector bank, or multidetector. The indices require explanation: np - the number of points in a scan. This dimension is only present in scanning measurements. tof - the number of points in a time-of-flight axis. This is only present in time-of-flight measurements. i - the number of pixels in the slowest varying direction. This is only present in missing in the point detector. j - the number of pixels in the fastest varying direction. This exists only in "area" detectors. HOUSEKEEPING: - remove all question marks - polar, azimuthal, distance, and wavelength need description - improve documentation throughout.

Table A.12: Tabular representation of NXdetector[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
time_of_flight	NX_FLOAT	NX_TIME_OF_FLIGHT	Total time of flight Dimensions: size="1" • dim: index="1" value="tof+1" value=""
@axis	NX_INT		
@primary	NX_INT		
@long_name	NX_CHAR		Axis label
@link	NX_CHAR		absolute path to location in NXdetector
raw_time_of_flig‐ht	NX_INT	NX_PULSES	In DAQ clock pulses Dimensions: size="1" • dim: index="1" value="tof+1" value=""
@frequency	NX_FLOAT		Clock frequency in Hz

Table A.12: (continued)

Name and Attributes	Type	Units	Description (and Occurrences)
detector_number	NX_INT		Identifier for detector Dimensions: size="2" • dim: index="1" value="i" value="" • dim: index="2" value="j" value=""
data	NX_NUMBER	NX_ANY	Data values Dimensions: size="4" • dim: index="1" value="np" value="" • dim: index="2" value="i" value="" • dim: index="3" value="j" value="" • dim: index="4" value="tof" value=""
@signal	NX_INT		
@axes	NX_CHAR		[number of scan points,x_offset?,y_offset?,time_of_flight?]
@long_name	NX_CHAR		Title of measurement
@check_sum	NX_INT		Integral of data as check of data integrity
@link	NX_CHAR		absolute path to location in NXdetector
data_error	NX_NUMBER	NX_ANY	Data values Dimensions: size="4" • dim: index="1" value="np" value="" • dim: index="2" value="i" value="" • dim: index="3" value="j" value="" • dim: index="4" value="tof" value=""
@units	NX_CHAR		
@link	NX_CHAR		absolute path to location in NXdetector
x_pixel_offset	NX_FLOAT	NX_LENGTH	offset from the detector center in x-direction Dimensions: size="1" • dim: index="1" value="i" value=""
@axis	NX_INT		
@primary	NX_INT		
@long_name	NX_CHAR		Axis label
@link	NX_CHAR		absolute path to location in NXdetector
y_pixel_offset	NX_FLOAT	NX_LENGTH	offset from the detector center in the y-direction Dimensions: size="1" • dim: index="1" value="j" value=""
@axis	NX_INT		
@primary	NX_INT		
@long_name	NX_CHAR		Axis label
distance	NX_FLOAT	NX_LENGTH	Dimensions: size="3" • dim: index="1" value="np" value="" • dim: index="2" value="i" value="" • dim: index="3" value="j" value=""
polar_angle	NX_FLOAT	NX_ANGLE	Dimensions: size="3" • dim: index="1" value="np" value="" • dim: index="2" value="i" value="" • dim: index="3" value="j" value=""

Table A.12: (continued)

Name and Attributes	Type	Units	Description (and Occurrences)
azimuthal_angle	NX_FLOAT	NX_ANGLE	Dimensions: size="3" • dim: index="1" value="np" value="" • dim: index="2" value="i" value="" • dim: index="3" value="j" value=""
description	NX_CHAR		name/manufacturer/model/etc. information
local_name	NX_CHAR		Local name for the detector
	NXgeometry		Position and orientation of detector element
solid_angle	NX_FLOAT	NX_SOLID_ANGLE	Solid angle subtended by the detector at the sample Dimensions: size="2" • dim: index="1" value="i" value="" • dim: index="2" value="j" value=""
x_pixel_size	NX_FLOAT	NX_LENGTH	Size of each detector pixel. If it is scalar all pixels are the same size Dimensions: size="2" • dim: index="1" value="i" value="" • dim: index="2" value="j" value=""
y_pixel_size	NX_FLOAT	NX_LENGTH	Size of each detector pixel. If it is scalar all pixels are the same size Dimensions: size="2" • dim: index="1" value="i" value="" • dim: index="2" value="j" value=""
dead_time	NX_FLOAT	NX_TIME	Detector dead time Dimensions: size="3" • dim: index="1" value="np" value="" • dim: index="2" value="i" value="" • dim: index="3" value="j" value=""
gas_pressure	NX_FLOAT	NX_PRESSURE	Detector gas pressure Dimensions: size="2" • dim: index="1" value="i" value="" • dim: index="2" value="j" value=""
detection_gas_pa-th	NX_FLOAT	NX_LENGTH	maximum drift space dimension
crate	NX_INT		Crate number of detector Dimensions: size="2" • dim: index="1" value="i" value="" • dim: index="2" value="j" value=""
@local_name	NX_CHAR		Equivalent local term
slot	NX_INT		Slot number of detector Dimensions: size="2" • dim: index="1" value="i" value="" • dim: index="2" value="j" value=""
@local_name	NX_CHAR		Equivalent local term

Table A.12: (continued)

Name and Attributes	Type	Units	Description (and Occurrences)
input @local_name	NX_INT NX_CHAR		Input number of detector Dimensions: size="2" • dim: index="1" value="i" value="" • dim: index="2" value="j" value="" Equivalent local term
type	NX_CHAR		Description of type such as He3 gas cylinder, He3 PSD, scintillator, fission chamber, proportion counter, ion chamber, ccd, pixel, image plate, cmos, ...
efficiency	NXdata		Efficiency of detector with respect to e.g. wavelength Look for special case table **efficiency[NXdata] (within NXdetector[definition])** below.
calibration_date	NX_DATE_TIME		date of last calibration (geometry and/or efficiency) measurements
calibration_meth-od	NXnote		summary of conversion of array data to pixels (e.g. polynomial approximations) and location of details of the calibrations
layout	point | linear | area		How the detector is represented
count_time	NX_NUMBER	NX_TIME	Elapsed actual counting time Dimensions: size="1" • dim: index="1" value="np" value=""
data_file	NXnote		
	NXcharacterization		
sequence_number	NX_CHAR		In order to properly sort the order of the images taken in (for example) a tomography experiment, a sequence number is stored with each image. Dimensions: size="1" • dim: index="1" value="nBrightFrames" value=""

A.2.10.1   Special case table: efficiency[NXdata] (within NXdetector[definition])

Table A.13:   Tabular representation of efficiency[NXdata] (within NXdetector[definition])

Name and Attributes	Type	Units	Description (and Occurrences)
efficiency	NX_FLOAT	NX_DIMENSIONLESS	efficiency of the detector Dimensions: size="3" • dim: index="1" value="i" value="" • dim: index="2" value="j" value="" • dim: index="3" value="k" value=""
wavelength	NX_FLOAT	NX_WAVELENGTH	Dimensions: size="3" • dim: index="1" value="i" value="" • dim: index="2" value="j" value="" • dim: index="3" value="k" value=""

A.2.11 NXdetector_group

SVN Id Id: NXdetector_group.nxdl.xml 373 2009-09-22 16:57:49Z Freddie Akeroyd

NXDL source: NXdetector_group

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXdetector_group.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: no included classes

documentation This class is used to allow a logical grouping of detector elements (e.g. which tube, bank or group of banks) to be recorded in the file. As well as allowing you to e.g just select the "left" or "east" detectors, it may also be useful for determining which elements belong to the same PSD tube and hence have e.g. the same dead time. For example, if we had "bank1" composed of "tube1", "tube2" and "tube3" then group_names would be the string "bank1, bank1/tube1, bank1/tube2,bank1/tube3" group_index would be {1,2,3,4} group_parent would be {-1,1,1,1} The mapping array is interpreted as group 1 is a top level group containing groups 2, 3 and 4 A group_index array in NXdetector give the base group for a detector element.

Table A.14: Tabular representation of NXdetector_group[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
group_names	NX_CHAR		Comma separated list of name
group_index	NX_INT		Unique ID for group. A group_index array in NXdetector give the base group for a detector element. Dimensions: apparent size="1" • dim: value="i" value=""
group_parent	NX_INT		Index of group parent in the hierarchy, -1 means no parent (i.e. a top level) group Dimensions: apparent size="1" • dim: ref="group_index" value=""
group_type	NX_INT		Code number for group type e.g. bank=1, tube=2 etc. Dimensions: apparent size="1" • dim: ref="group_index" value=""

A.2.12 NXdisk_chopper

SVN Id Id: NXdisk_chopper.nxdl.xml 373 2009-09-22 16:57:49Z Freddie Akeroyd

NXDL source: NXdisk_chopper

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXdisk_chopper.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXgeometry

documentation

Table A.15: Tabular representation of NXdisk_chopper[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
type	"Chopper type single" | contra_rotating_pair | synchro_pair		
rotation_speed	NX_FLOAT	NX_FREQUENCY	chopper rotation speed
slits	NX_INT		Number of slits
slit_angle	NX_FLOAT	NX_ANGLE	angular opening
pair_separation	NX_FLOAT	NX_LENGTH	disc spacing in direction of beam
radius	NX_FLOAT	NX_LENGTH	radius to centre of slit
slit_height	NX_FLOAT	NX_LENGTH	total slit height
phase	NX_FLOAT	NX_ANGLE	chopper phase angle
ratio	NX_INT		pulse reduction factor of this chopper in relation to other choppers/fastest pulse in the instrument
distance	NX_FLOAT	NX_LENGTH	Effective distance to the origin
wavelength_range	NX_FLOAT	NX_WAVELENGTH	low and high values of wavelength range transmitted Dimensions: apparent size="1" • dim: value="2" value=""
	NXgeometry		

A.2.13 NXentry

SVN Id Id: NXentry.nxdl.xml 373 2009-09-22 16:57:49Z Freddie Akeroyd

NXDL source: NXentry

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXentry.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXcharacterization

> NXdata
>
> NXinstrument
>
> NXmonitor
>
> NXnote
>
> NXprocess
>
> NXsample
>
> NXuser

documentation Template of the top-level NeXus group which contains all the data and associated information that comprise a single measurement. It is mandatory that there is at least one group of this type in the NeXus file.

Table A.16: Tabular representation of NXentry[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
title	NX_CHAR		Extended title for entry
experiment_ident-ifier	NX_CHAR		Unique identifier for the experiment, defined by the facility, possibly linked to the proposals
experiment_descr-iption	NX_CHAR		Brief summary of the experiment, including key objectives.
experiment_docum-entation	NXnote		Description of the full experiment (document in pdf, latex, . . .)

Table A.16: (continued)

Name and Attributes	Type	Units	Description (and Occurrences)
collection_ident-ifier	NX_CHAR		User or Data Acquisition defined group of NeXus files or NXentry
collection_descr-iption	NX_CHAR		Brief summary of the collection, including grouping criteria.
entry_identifier	NX_CHAR		unique identifier for the measurement, defined by the facility.
definition	NX_CHAR		Official NeXus DTD or NXDL schema to which this file conforms
@version	NX_CHAR		DTD or NXDL version number
@URL	NX_CHAR		URL of DTD or NXDL file
definition_local	NX_CHAR		Local DTD or NXDL schema extended from the file specified in the "definition" tag. This contains the locally defined additional fields in the file.
@version	NX_CHAR		DTD or NXDL version number
@URL	NX_CHAR		URL of DTD or NXDL file
start_time	NX_DATE_TIME		Starting time of measurement
end_time	NX_DATE_TIME		Ending time of measurement
duration	NX_INT	NX_TIME	Duration of measurement
collection_time	NX_FLOAT	NX_TIME	Time transpired actually collecting data i.e. taking out time when collection was suspended due to e.g. temperature out of range
run_cycle	NX_CHAR		
program_name	NX_CHAR		Name of program used to generate this file
@version	NX_CHAR		Program version number
@configuration	NX_CHAR		configuration of the program
revision	NX_CHAR		Revision id of the file due to re-calibration, reprocessing, new analysis, new instrument definition format, ...
@comment	NX_CHAR		
notes	NXnote		Notes describing entry
thumbnail	NXnote		A small image that is representative of the entry. An example of this is a 640x480 jpeg image automatically produced by a low resolution plot of the NXdata.
@mime_type	NX_CHAR		The value should be an "image/*"
	NXcharacterization		
	NXuser		
	NXsample		
	NXinstrument		
	NXmonitor		
	NXdata		
	NXprocess		

A.2.14 NXenvironment

SVN Id Id: NXenvironment.nxdl.xml 373 2009-09-22 16:57:49Z Freddie Akeroyd

NXDL source: NXenvironment

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXenvironment.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXgeometry

NXnote

NXsensor

documentation This class describes an external condition applied to the sample

Table A.17: Tabular representation of NXenvironment[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
name	NX_CHAR		Apparatus identification code/model number; e.g. OC100 011
short_name	NX_CHAR		Alternative short name, perhaps for dashboard display like a present Seblock name
type	NX_CHAR		Type of apparatus. This could be the SE codes in scheduling database; e.g. OC/100
description	NX_CHAR		Description of the apparatus; e.g. 100mm bore orange cryostat with Roots pump
program	NX_CHAR		Program controlling the apparatus; e.g. LabView VI name
position	NXgeometry		The position and orientation of the apparatus
	NXnote		Additional information, LabView logs, digital photographs, etc
	NXsensor		

A.2.15 NXevent_data

SVN Id Id: NXevent_data.nxdl.xml 373 2009-09-22 16:57:49Z Freddie Akeroyd

NXDL source: NXevent_data

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXevent_data.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: no included classes

documentation Time-of-flight events

Table A.18: Tabular representation of NXevent_data[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
time_of_flight	NX_INT	NX_TIME_OF_FLIGHT	A list of time of flight for each event as it comes in. This list is for all pulses with information to attach to a particular pulse located in events_per_pulse. Dimensions: size="1" • dim: index="1" value="i" value=""
pixel_number	NX_INT	NX_DIMENSIONLESS	There will be extra information in the NXdetector to convert pixel_number to detector_number. This list is for all pulses with information to attach to a particular pulse located in events_per_pulse. Dimensions: size="1" • dim: index="1" value="i" value=""

Table A.18: (continued)

Name and Attributes	Type	Units	Description (and Occurrences)
pulse_time @offset	NX_INT NX_INT	NX_TIME	The time that each pulse started with respect to the offset Dimensions: size="1" • dim: index="1" value="j" value="" ISO8601
events_per_pulse	NX_INT	NX_DIMENSIONLESS	This connects the index "i" to the index "j". The jth element is the number of events in "i" that occurred during the jth pulse. Dimensions: size="1" • dim: index="1" value="j" value=""
pulse_height	NX_FLOAT	NX_DIMENSIONLESS	If voltages from the ends of the detector are read out this is where they go. This list is for all events with information to attach to a particular pulse height. The information to attach to a particular pulse is located in events_per_pulse. Dimensions: size="2" • dim: index="1" value="i" value="" • dim: index="2" value="k" value=""

A.2.16 NXfermi_chopper

SVN Id Id: NXfermi_chopper.nxdl.xml 411 2009-10-18 21:36:40Z Pete Jemian

NXDL source: NXfermi_chopper

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXfermi_chopper.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXgeometry

documentation Description of a Fermi chopper, possibly with curved slits.

Table A.19: Tabular representation of NXfermi_chopper[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
type	NX_CHAR		Fermi chopper type
rotation_speed	NX_FLOAT	NX_FREQUENCY	chopper rotation speed
radius	NX_FLOAT	NX_LENGTH	radius of chopper
slit	NX_FLOAT	NX_LENGTH	width of an individual slit
r_slit	NX_FLOAT	NX_LENGTH	radius of curvature of slits
number	NX_INT	NX_UNITLESS	number of slits
height	NX_FLOAT	NX_LENGTH	input beam height
width	NX_FLOAT	NX_LENGTH	input beam width
wavelength	NX_FLOAT	NX_WAVELENGTH	Wavelength transmitted by chopper
	NXgeometry		geometry of the fermi chopper
absorbing_material	NX_CHAR		absorbing material
transmitting_material	NX_CHAR		transmitting material

A.2.17 NXfilter

SVN Id Id: NXfilter.nxdl.xml 409 2009-10-18 19:50:56Z Pete Jemian

NXDL source: NXfilter

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXfilter.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXdata

NXgeometry

NXlog

NXsensor

documentation Template for specifying the state of band pass filters

Table A.20: Tabular representation of NXfilter[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
	NXgeometry		Geometry of the filter
description	Beryllium | "Pyrolytic Graphite" | Graphite | Sapphire | Silicon | Supermirror		
status	in | out		
transmission	NXdata		Wavelength transmission profile of filter
temperature	NX_FLOAT	NX_TEMPERATURE	Average/nominal filter temperature
temperature_log	NXlog		Linked temperature_log for the filter
sensor_type	NXsensor		Sensor(s)used to monitor the filter temperature
unit_cell_a	NX_FLOAT	NX_LENGTH	Unit cell lattice parameter: length of side a
unit_cell_b	NX_FLOAT	NX_LENGTH	Unit cell lattice parameter: length of side b
unit_cell_c	NX_FLOAT	NX_LENGTH	Unit cell lattice parameter: length of side c
unit_cell_alpha	NX_FLOAT	NX_ANGLE	Unit cell lattice parameter: angle alpha
unit_cell_beta	NX_FLOAT	NX_ANGLE	Unit cell lattice parameter: angle beta
unit_cell_gamma	NX_FLOAT	NX_ANGLE	Unit cell lattice parameter: angle gamma
unit_cell_volume	NX_FLOAT	NX_VOLUME	Unit cell Dimensions: size="1" • dim: index="1" value="n_comp" value=""
orientation_matr-ix	NX_FLOAT		Orientation matrix of single crystal filter Dimensions: size="3" • dim: index="1" value="n_comp" value="" • dim: index="2" value="3" value="" • dim: index="3" value="3" value=""
m_value	NX_FLOAT	NX_DIMENSIONLESS	m value of supermirror filter
substrate_materi-al	NX_CHAR		substrate material of supermirror filter
substrate_thickn-ess	NX_FLOAT	NX_LENGTH	substrate thickness of supermirror filter

Table A.20: (continued)

Name and Attributes	Type	Units	Description (and Occurrences)
`coating_material`	NX_CHAR		coating material of supermirror filter
`substrate_roughn-ess`	NX_FLOAT	NX_LENGTH	substrate roughness (RMS) of supermirror filter
`coating_roughness`	NX_FLOAT	NX_LENGTH	coating roughness (RMS) of supermirror filter Dimensions: size="1" • dim: incr="1" value="nsurf" value=""

A.2.18 NXflipper

SVN Id Id: NXflipper.nxdl.xml 373 2009-09-22 16:57:49Z Freddie Akeroyd

NXDL source: NXflipper

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXflipper.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: no included classes

documentation Template of a beamline spin flipper.

Table A.21: Tabular representation of NXflipper[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
`type`	coil | current-sheet		
`flip_turns`	NX_FLOAT	NX_PER_LENGTH	Linear density of turns (such as number of turns/cm) in flipping field coils
`comp_turns`	NX_FLOAT	NX_PER_LENGTH	Linear density of turns (such as number of turns/cm) in compensating field coils
`guide_turns`	NX_FLOAT	NX_PER_LENGTH	Linear density of turns (such as number of turns/cm) in guide field coils
`flip_current`	NX_FLOAT	NX_CURRENT	Flipping field coil current in "on" state
`comp_current`	NX_FLOAT	NX_CURRENT	Compensating field coil current in "on" state
`guide_current`	NX_FLOAT	NX_CURRENT	Guide field coil current in "on" state
`thickness`	NX_FLOAT	NX_LENGTH	thickness along path of neutron travel

A.2.19 NXgeometry

SVN Id Id: NXgeometry.nxdl.xml 412 2009-10-18 21:43:22Z Pete Jemian

NXDL source: NXgeometry

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXgeometry.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXorientation

NXshape

NXtranslation

documentation This is the description for a general position of a component. It is recommended to name an instance of NXgeometry as "geometry" to aid in the use of the definition in simulation codes such as McStas. Also, in HDF, linked items must share the same name. However, it might not be possible or practical in all situations.

Table A.22: Tabular representation of NXgeometry[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
	NXshape		shape/size information of component
	NXtranslation		translation of component
	NXorientation		orientation of component
description	NX_CHAR		Optional description/label. Probably only present if we are an additional reference point for components rather than the location of a real component
component_index	NX_INT		Position of the component along the beam path. The sample is at 0, components upstream have negative component_index, components downstream have positive component_index.

A.2.20 NXguide

SVN Id Id: NXguide.nxdl.xml 431 2009-10-19 00:53:57Z Stuart Campbell

NXDL source: NXguide

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXguide.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXdata

NXgeometry

documentation Template of NXguide

Table A.23: Tabular representation of NXguide[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
	NXgeometry		
description	NX_CHAR		
incident_angle	NX_FLOAT	NX_ANGLE	
reflectivity	NXdata		Reflectivity as function of wavelength [nsurf,i]
bend_angle_x	NX_FLOAT	NX_ANGLE	
bend_angle_y	NX_FLOAT	NX_ANGLE	
interior_atmosph-ere	vacuum | helium | argon		
external_material	NX_CHAR		external material outside substrate

Table A.23: (continued)

Name and Attributes	Type	Units	Description (and Occurrences)
m_value	NX_FLOAT		The m value for a supermirror, which defines the supermirror regime in multiples of the critical angle of Nickel. Dimensions: size="1" • dim: index="1" value="nsurf" value=""
substrate_materi-al	NX_FLOAT		Dimensions: size="1" • dim: index="1" value="nsurf" value=""
substrate_thickn-ess	NX_FLOAT	NX_LENGTH	Dimensions: size="1" • dim: index="1" value="nsurf" value=""
coating_material	NX_FLOAT		Dimensions: size="1" • dim: index="1" value="nsurf" value=""
substrate_roughn-ess	NX_FLOAT	NX_LENGTH	Dimensions: size="1" • dim: index="1" value="nsurf" value=""
coating_roughness	NX_FLOAT	NX_LENGTH	Dimensions: size="1" • dim: index="1" value="nsurf" value=""
number_sections	NX_INT	NX_UNITLESS	number of substrate sections

A.2.21 NXinsertion_device

SVN Id Id: NXinsertion_device.nxdl.xml 413 2009-10-18 21:52:21Z Pete Jemian

NXDL source: NXinsertion_device

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXinsertion_device.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXdata

NXgeometry

documentation This is the description for an insertion device.

Table A.24: Tabular representation of NXinsertion_device[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
type	undulator | wiggler		
gap	NX_FLOAT	NX_LENGTH	
taper	NX_FLOAT	NX_LENGTH	
phase	NX_FLOAT	NX_ANGLE	
poles	NX_INT	NX_UNITLESS	number of poles
length	NX_FLOAT	NX_LENGTH	length of insertion device
power	NX_FLOAT	NX_POWER	total power delivered by insertion device
energy	NX_FLOAT	NX_ENERGY	energy of peak
bandwidth	NX_FLOAT	NX_ENERGY	bandwidth of peak energy
harmonic	NX_INT	NX_UNITLESS	harmonic of peak
spectrum	NXdata		spectrum of insertion device
	NXgeometry		"Engineering" position of insertion device

A.2.22   NXinstrument

SVN Id Id: NXinstrument.nxdl.xml 415 2009-10-18 21:56:22Z Pete Jemian

NXDL source: NXinstrument

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXinstrument.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXaperture

>   NXattenuator
>
>   NXbeam
>
>   NXbeam_stop
>
>   NXbending_magnet
>
>   NXcollimator
>
>   NXcrystal
>
>   NXdetector
>
>   NXdisk_chopper
>
>   NXfermi_chopper
>
>   NXfilter
>
>   NXflipper
>
>   NXguide
>
>   NXinsertion_device
>
>   NXmirror
>
>   NXmoderator
>
>   NXpolarizer
>
>   NXsource
>
>   NXvelocity_selector

documentation Template of instrument descriptions comprising various beamline components. Each component will also be a NeXus group defined by its distance from the sample. Negative distances represent beamline components that are before the sample while positive distances represent components that are after the sample. This device allows the unique identification of beamline components in a way that is valid for both reactor and pulsed instrumentation.

Table A.25:   Tabular representation of NXinstrument[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
name	NX_CHAR		Name of instrument
@short_name	NX_CHAR		short name for instrument, perhaps the acronym
	NXaperture		
	NXattenuator		
	NXbeam		
	NXbeam_stop		
	NXbending_magnet		
	NXcollimator		
	NXcrystal		
	NXdetector		

Table A.25: (continued)

Name and Attributes	Type	Units	Description (and Occurrences)
	NXdisk_chopper		
	NXfermi_chopper		
	NXfilter		
	NXflipper		
	NXguide		
	NXinsertion_device		
	NXmirror		
	NXmoderator		
	NXpolarizer		
	NXsource		
	NXvelocity_selector		

A.2.23 NXlog

SVN Id Id: NXlog.nxdl.xml 373 2009-09-22 16:57:49Z Freddie Akeroyd

NXDL source: NXlog

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXlog.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: no included classes

documentation Definition of logged information, i.e. information monitored during the run. They contain the logged values and the times at which they were measured as elapsed time since a starting time recorded in ISO8601 format. This method of storing logged data helps to distinguish instances in which a variable is a dimension scale of the data, in which case it is stored in an NXdata group, and instances in which it is logged during the run, when it should be stored in an NXlog group.

Table A.26: Tabular representation of NXlog[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
time	NX_FLOAT	NX_TIME	Time of logged entry. The times are relative to the "start" attribute and in the units specified in the "units" attribute.
@start	NX_DATE_TIME		
value	NX_NUMBER	NX_ANY	Array of logged value, such as temperature
raw_value	NX_NUMBER	NX_ANY	Array of raw information, such as thermocouple voltage
description	NX_CHAR		Description of logged value
average_value	NX_FLOAT	NX_ANY	
average_value_error	NX_FLOAT	NX_ANY	standard deviation of average_value
minimum_value	NX_FLOAT	NX_ANY	
maximum_value	NX_FLOAT	NX_ANY	
duration	NX_FLOAT	NX_ANY	Total time log was taken

A.2.24 NXmirror

SVN Id Id: NXmirror.nxdl.xml 432 2009-10-19 00:55:04Z Stuart Campbell

NXDL source: NXmirror

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXmirror.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXdata

    NXgeometry

documentation Template of a beamline supermirror.

Table A.27: Tabular representation of NXmirror[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
	NXgeometry		
description	NX_CHAR		
incident_angle	NX_FLOAT	NX_ANGLE	
reflectivity	NXdata		Reflectivity as function of wavelength
bend_angle_x	NX_FLOAT	NX_ANGLE	
bend_angle_y	NX_FLOAT	NX_ANGLE	
interior_atmosph-ere	vacuum | helium | argon		
external_material	NX_CHAR		external material outside substrate
m_value	NX_FLOAT	NX_UNITLESS	The m value for a supermirror, which defines the supermirror regime in multiples of the critical angle of Nickel.
substrate_materi-al	NX_CHAR		
substrate_thickn-ess	NX_FLOAT	NX_LENGTH	
coating_material	NX_CHAR		
substrate_roughn-ess	NX_FLOAT	NX_LENGTH	
coating_roughness	NX_FLOAT	NX_LENGTH	

A.2.25 NXmoderator

SVN Id Id: NXmoderator.nxdl.xml 373 2009-09-22 16:57:49Z Freddie Akeroyd

NXDL source: NXmoderator

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXmoderator.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXdata

    NXgeometry

    NXlog

documentation This is the description for a general moderator

Table A.28: Tabular representation of NXmoderator[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
	NXgeometry		"Engineering" position of moderator
distance	NX_FLOAT	NX_LENGTH	Effective distance as seen by measuring radiation
type	H20 | D20 | "Liquid H2" | "Liquid CH4" | "Liquid D2" | "Solid D2" | C | "Solid CH4" | "Solid H2"		
poison_depth	NX_FLOAT	NX_LENGTH	
coupled	NX_BOOLEAN		whether the moderator is coupled
coupling_material	NX_CHAR		The material used for coupling. Usually Cd.
poison_material	Gd | Cd		
temperature	NX_FLOAT	NX_TEMPERATURE	Average/nominal moderator temperature
temperature_log	NXlog		log file of moderator temperature
pulse_shape	NXdata		moderator pulse shape

A.2.26 NXmonitor

SVN Id Id: NXmonitor.nxdl.xml 427 2009-10-19 00:06:46Z Pete Jemian

NXDL source: NXmonitor

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXmonitor.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXgeometry

NXlog

documentation Template of monitor data. It is similar to the NXdata groups containing monitor data and its associated dimension scale, e.g. time_of_flight or wavelength in pulsed neutron instruments. However, it may also include integrals, or scalar monitor counts, which are often used in both in both pulsed and steady-state instrumentation.

Table A.29: Tabular representation of NXmonitor[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
mode	monitor | timer		Count to a preset value based on either clock time (timer) or received monitor counts (monitor).
preset	NX_NUMBER	NX_ANY	preset value for time or monitor
distance	NX_FLOAT	NX_LENGTH	Distance of monitor from sample
range	NX_FLOAT	NX_ANY	Range (X-axis, Time-of-flight, etc.) over which the integral was calculated Dimensions: apparent size="1" • dim: value="2" value=""

Table A.29: (continued)

Name and Attributes	Type	Units	Description (and Occurrences)
integral	NX_NUMBER	NX_ANY	Total integral monitor counts
integral_log	NXlog		Time variation of monitor counts
type	"Fission Chamber" | Scintillator		
time_of_flight	NX_FLOAT	NX_TIME_OF_FLIGHT	Time-of-flight Dimensions: apparent size="1" • dim: ref="efficiency" value=""
efficiency	NX_NUMBER	NX_DIMENSIONLESS	Monitor efficiency Dimensions: apparent size="1" • dim: ref="i" value=""
data	NX_NUMBER	NX_ANY	Monitor data The signal and axes attributes take the same definitions as in NXdata: signal: signal="1" means this is the plottable data axes: axes="names" where names are defined as a comma-delimited string within this attribute in the C-order of the data array Dimensions: apparent size="1" • dim: value=""
@signal	NX_INT		
@axes	NX_CHAR		
sampled_fraction	NX_FLOAT	NX_DIMENSIONLESS	Proportion of incident beam sampled by the monitor (0<x<1)
	NXgeometry		Geometry of the monitor
count_time	NX_FLOAT	NX_TIME	Elapsed actual counting time, can be an array of size np when scanning. This is not the difference of the calendar time but the time the instrument was really counting, without pauses or times lost due beam unavailability

A.2.27 NXmonochromator

SVN Id Id: NXmonochromator.nxdl.xml 433 2009-10-19 01:00:56Z Pete Jemian

NXDL source: NXmonochromator

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXmonochromator.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXdata

    NXgeometry

documentation This is a base class for everything which selects a wavelength or energy, be it a monochromator crystal, a velocity selector, a undulator or whatever expected units wavelength: angstrom energy: eV

Table A.30: Tabular representation of NXmonochromator[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
wavelength	NX_FLOAT	NX_WAVELENGTH	wavelength selected
wavelength_error	NX_FLOAT	NX_WAVELENGTH	wavelength standard deviation
energy	NX_FLOAT	NX_ENERGY	energy selected
energy_error	NX_FLOAT	NX_ENERGY	energy standard deviation
distribution	NXdata		
geometry	NXgeometry		

A.2.28 NXnote

SVN Id Id: NXnote.nxdl.xml 373 2009-09-22 16:57:49Z Freddie Akeroyd

NXDL source: NXnote

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXnote.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: no included classes

documentation This class can be used to store additional information in a NeXus file e.g. pictures, movies, audio, additional text logs

Table A.31: Tabular representation of NXnote[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
author	NX_CHAR		Author or creator of note
date	NX_DATE_TIME		Date note created/added
type	NX_CHAR		Mime content type of note data field e.g. image/jpeg, text/plain, text/html}
file_name	NX_CHAR		Name of original file name if note was read from an external source
description	NX_CHAR		Title of an image or other details of the note
data	NX_BINARY		Binary note data - if text, line terminator is [CR][LF].

A.2.29 NXobject

SVN Id Id: NXuser.nxdl.xml 303 2009-01-30 17:55:59Z Freddie Akeroyd

NXDL source: NXobject

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXobject.nxdl.xml

NeXus Definition Language NXDL

extends class:

other classes included: no included classes

documentation This is the base object of NeXus

No fields or groups are defined (nothing to show in a table at this point).

A.2.30 NXorientation

SVN Id Id: NXorientation.nxdl.xml 373 2009-09-22 16:57:49Z Freddie Akeroyd

NXDL source: NXorientation

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXorientation.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXgeometry

documentation This is the description for a general orientation of a component - it is used by the NXgeometry class

Table A.32: Tabular representation of NXorientation[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
	NXgeometry		Link to another object if we are using relative positioning, else absent
value	NX_FLOAT	NX_UNITLESS	The orientation information is stored as direction cosines. The direction cosines will be between the local coordinate directions and the reference directions (to origin or relative NXgeometry). Calling the local unit vectors (x',y',z') and the reference unit vectors (x,y,z) the six numbers will be [x' dot x, x' dot y, x' dot z, y' dot x, y' dot y, y' dot z] where "dot" is the scalar dot product (cosine of the angle between the unit vectors). The unit vectors in both the local and reference coordinates are right-handed and orthonormal. Dimensions: apparent size="2" • dim: value="numobj" value="" • dim: value="6" value=""

A.2.31 NXparameters

SVN Id Id: NXparameters.nxdl.xml 406 2009-10-18 17:12:29Z Pete Jemian

NXDL source: NXparameters

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXparameters.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: no included classes

documentation Container for parameters, usually used in processing or analysis.

Table A.33: Tabular representation of NXparameters[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
term	NX_CHAR		A parameter (also known as a term) that is used in or results from processing. Occurences: 0 : 9999999999999999999999999
@units	NX_CHAR		

A.2.32 NXpolarizer

SVN Id Id: NXpolarizer.nxdl.xml 419 2009-10-18 22:08:10Z Pete Jemian

NXDL source: NXpolarizer

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXpolarizer.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: no included classes

documentation Template of a beamline spin polarizer. This is a first cut. This is NOT the last word.

Table A.34: Tabular representation of NXpolarizer[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
type	NX_CHAR		crystal,supermirror
composition	NX_CHAR		description of the composition of the polarizing material
reflection	NX_INT	NX_UNITLESS	[hkl] values of nominal reflection Dimensions: apparent size="1" • dim: value="3" value=""
efficiency	NX_FLOAT	NX_DIMENSIONLESS	polarizing efficiency

A.2.33 NXpositioner

SVN Id Id: NXpositioner.nxdl.xml 373 2009-09-22 16:57:49Z Freddie Akeroyd

NXDL source: NXpositioner

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXpositioner.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: no included classes

documentation This is the description for a generic positioner.

Table A.35: Tabular representation of NXpositioner[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
description	NX_CHAR		description of positioner
value	NX_FLOAT	NX_ANY	best known value of positioner - need [n] as may be scanned Dimensions: size="1" • dim: index="1" value="n" value=""
raw_value	NX_NUMBER	NX_ANY	raw value of positioner - need [n] as may be scanned Dimensions: size="1" • dim: index="1" value="n" value=""

Table A.35: (continued)

Name and Attributes	Type	Units	Description (and Occurrences)
target_value	NX_FLOAT	NX_ANY	targeted (commanded) value of positioner - need [n] as may be scanned Dimensions: size="1" • dim: index="1" value="n" value=""
tolerance	NX_FLOAT	NX_ANY	maximum allowable difference between target_value and value Dimensions: size="1" • dim: index="1" value="n" value=""
soft_limit_min	NX_FLOAT	NX_ANY	minimum allowed limit to set value
soft_limit_max	NX_FLOAT	NX_ANY	maximum allowed limit to set value
controller_record	NX_CHAR		Hardware device record, e.g. EPICS process variable, taco/tango ...

A.2.34   NXprocess

SVN Id Id: NXprocess.nxdl.xml 406 2009-10-18 17:12:29Z Pete Jemian

NXDL source: NXprocess

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXprocess.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXnote

documentation Template for a process.

Table A.36:   Tabular representation of NXprocess[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
program	NX_CHAR		Name of the program used for reconstruction
version	NX_CHAR		Version of the program used
date	NX_DATE_TIME		Date and time of reconstruction processing.
	NXnote		The note will contain information about how the data was processed. The contents of the note can be anything that the processing code can understand, or simple text. The name will be numbered to allow for ordering of steps.

A.2.35   NXroot

SVN Id Id: NXroot.nxdl.xml 373 2009-09-22 16:57:49Z Freddie Akeroyd

NXDL source: NXroot

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXroot.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXentry

documentation Definition of the root NeXus group.

Table A.37: Tabular representation of NXroot[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
	NXentry		entries Occurences: 1 : *default*

A.2.36 NXsample

SVN Id Id: NXsample.nxdl.xml 420 2009-10-18 22:15:41Z Pete Jemian

NXDL source: NXsample

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXsample.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXbeam

    NXdata

    NXenvironment

    NXgeometry

    NXlog

documentation Template of the state of the sample. This could include scanned variables that are associated with one of the data dimensions, e.g. the magnetic field, or logged data, e.g. monitored temperature vs elapsed time.

Table A.38: Tabular representation of NXsample[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
name	NX_CHAR		Descriptive name of sample
chemical_formula	NX_CHAR		The chemical formula specified using CIF conventions. Abbreviated version of CIF standard: 1. Only recognized element symbols may be used. 2. Each element symbol is followed by a 'count' number. A count of '1' may be omitted. 3. A space or parenthesis must separate each cluster of (element symbol + count). 4. Where a group of elements is enclosed in parentheses, the multiplier for the group must follow the closing parentheses. That is, all element and group multipliers are assumed to be printed as subscripted numbers. 5. Unless the elements are ordered in a manner that corresponds to their chemical structure, the order of the elements within any group or moiety depends on whether or not carbon is present. If carbon is present, the order should be: C, then H, then the other elements in alphabetical order of their symbol. If carbon is not present, the elements are listed purely in alphabetic order of their symbol. This is the 'Hill' system used by Chemical Abstracts.

Table A.38: (continued)

Name and Attributes	Type	Units	Description (and Occurrences)
temperature	NX_FLOAT	NX_TEMPERATURE	Sample temperature. This could be a scanned variable Dimensions: apparent size="1" • dim: value=""
electric_field @direction	NX_FLOAT NX_CHAR	NX_VOLTAGE	Applied electric field Dimensions: apparent size="1" • dim: value=""
magnetic_field @direction	NX_FLOAT NX_CHAR	NX_ANY	Applied magnetic field Dimensions: apparent size="1" • dim: value=""
stress_field @direction	NX_FLOAT NX_CHAR	NX_ANY	Applied external stress field Dimensions: apparent size="1" • dim: value=""
pressure	NX_FLOAT	NX_PRESSURE	Applied pressure Dimensions: apparent size="1" • dim: value=""
changer_position	NX_INT	NX_UNITLESS	Sample changer position
unit_cell	NX_FLOAT	NX_LENGTH	Unit cell parameters (lengths and angles) Dimensions: size="2" • dim: index="1" value="n_comp" value="" • dim: index="2" value="6" value=""
unit_cell_volume	NX_FLOAT	NX_VOLUME	Volume of the unit cell Dimensions: size="1" • dim: index="1" value="n_comp" value=""
sample_orientati- on	NX_FLOAT	NX_ANGLE	This will follow the Busing and Levy convention from Acta.Crysta v22, p457 (1967) Dimensions: size="1" • dim: index="1" value="3" value=""
orientation_matr- ix	NX_FLOAT		Orientation matrix of single crystal sample. This will follow the Busing and Levy convention from Acta.Crysta v22, p457 (1967) Dimensions: size="3" • dim: index="1" value="n_comp" value="" • dim: index="2" value="3" value="" • dim: index="3" value="3" value=""
mass	NX_FLOAT	NX_MASS	Mass of sample Dimensions: size="1" • dim: index="1" value="n_comp" value=""
density	NX_FLOAT	NX_MASS_DENSITY	Density of sample Dimensions: size="1" • dim: index="1" value="n_comp" value=""

Table A.38: (continued)

Name and Attributes	Type	Units	Description (and Occurrences)
relative_molecul-ar_mass	NX_FLOAT	NX_MASS	Relative Molecular Mass of sample Dimensions: size="1" • dim: index="1" value="n_comp" value=""
type	sample | sample+can | can | "calibration sample" | "normalisation sample" | "simulated data" | none | "sample environment"		
situation	air | vacuum | "inert atmosphere" | "oxidising atmosphere" | "reducing atmosphere" | "sealed can" | other		The atmosphere will be one of the components, which is where its details will be stored; the relevant components will be indicated by the entry in the sample_component member.
description	NX_CHAR		Description of the sample
preparation_date	NX_DATE_TIME		Date of preparation of the sample
geometry	NXgeometry		The position and orientation of the center of mass of the sample
	NXbeam		Details of beam incident on sample - used to calculate sample/beam interaction point
component	NX_CHAR		Details of the component of the sample and/or can Dimensions: size="1" • dim: index="1" value="n_comp" value=""
sample_component	sample | can | atmosphere | kit		What type of component we are: "sample | can | atmosphere | kit" Dimensions: size="1" • dim: index="1" value="n_comp" value=""
concentration	NX_FLOAT	NX_MASS_DENSITY	Concentration of each component Dimensions: size="1" • dim: index="1" value="n_comp" value=""
volume_fraction	NX_FLOAT		Volume fraction of each component Dimensions: size="1" • dim: index="1" value="n_comp" value=""
scattering_lengt-h_density	NX_FLOAT	NX_SCATTERING_LENGTH_DENSITY	Scattering length density of each component Dimensions: size="1" • dim: index="1" value="n_comp" value=""
unit_cell_class	cubic | tetragonal | orthorhombic | monoclinic | triclinic		In case it is all we know and we want to record it: "cubic | tetragonal | orthorhombic | monoclinic | triclinic" Dimensions: size="1" • dim: index="1" value="n_comp" value=""

Table A.38: (continued)

Name and Attributes	Type	Units	Description (and Occurrences)
unit_cell_group	NX_CHAR		Crystallographic point or space group Dimensions: size="1" • dim: index="1" value="n_comp" value=""
path_length	NX_FLOAT	NX_LENGTH	Path length through sample/can for simple case when it does not vary with scattering direction
path_length_window	NX_FLOAT	NX_LENGTH	Thickness of a beam entry/exit window on the can (mm) - assumed same for entry and exit
thickness	NX_FLOAT	NX_LENGTH	sample thickness
transmission	NXdata		As a function of Wavelength
temperature_log	NXlog		temperature_log.value is a link to e.g. temperature_env.sensor1.value_log.value
temperature_env	NXenvironment		Additional sample temperature environment information
magnetic_field_log	NXlog		magnetic_field_log.value is a link to e.g. magnetic_field_env.sensor1.value_log.value
magnetic_field_env	NXenvironment		Additional sample magnetic environment information
external_DAC	NX_FLOAT	NX_ANY	value sent to user's sample setup
external_ADC	NXlog		logged value (or logic state) read from user's setup
short_title	NX_CHAR		20 character fixed length sample description for legends
rotation_angle	NX_FLOAT	NX_ANGLE	Optional rotation angle for the case when the powder diagram has been obtained through an omega-2theta scan like from a traditional single detector powder diffractometer

A.2.37 NXsensor

SVN Id Id: NXsensor.nxdl.xml 422 2009-10-18 22:30:16Z Pete Jemian

NXDL source: NXsensor

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXsensor.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXgeometry

NXlog

NXorientation

documentation This class describes a sensor used to monitor an external condition - the condition itself in described in NXenvironment

Table A.39: Tabular representation of NXsensor[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
model	NX_CHAR		Sensor identification code/model number
name	NX_CHAR		Name for the sensor
short_name	NX_CHAR		Short name of sensor used e.g. on monitor display program

Table A.39: (continued)

Name and Attributes	Type	Units	Description (and Occurrences)
attached_to	NX_CHAR		where sensor is attached to ("sample" | "can")
geometry	NXgeometry		Defines the axes for logged vector quantities if they are not the global instrument axes
measurement	temperature | pH | magnetic_field | "electric field" | conductivity | resistance | voltage | pressure | flow | stress | strain | shear | surface_pressure		name for measured signal
type	NX_CHAR		The type of hardware used for the measurement: e.g. Temperature: "J	K	T	E	R	S	Pt100	Rh/Fe" pH: "Hg/Hg2Cl2	Ag/AgCl	ISFET" Ion selective electrode: "specify species; e.g. Ca2+" Magnetic field: "Hall" Surface pressure: "wilhelmy plate"
run_control	NX_BOOLEAN		Is data collection controlled/synchronised to this quantity: 1=no, 0=to "value", 1=to "value_deriv1" etc.
high_trip_value	NX_FLOAT	NX_ANY	Upper control bound of sensor reading if using run_control
low_trip_value	NX_FLOAT	NX_ANY	Lower control bound of sensor reading if using run_control
value	NX_FLOAT	NX_ANY	nominal setpoint or average value - need [n] as may be a vector Dimensions: apparent size="1" • dim: value="n" value=""
value_deriv1	NX_FLOAT	NX_ANY	Nominal/average first derivative of value e.g. strain rate - same dimensions as "value" (may be a vector) Dimensions: apparent size="1" • dim: ref="value" value=""
value_deriv2	NX_FLOAT	NX_ANY	Nominal/average second derivative of value - same dimensions as "value" (may be a vector) Dimensions: apparent size="1" • dim: ref="value" value=""
value_log	NXlog		Time history of sensor readings
value_deriv1_log	NXlog		Time history of first derivative of sensor readings
value_deriv2_log	NXlog		Time history of second derivative of sensor readings
external_field_b-rief	"along beam" | "across beam" | transverse | solenoidal | "flow shear gradient" | "flow vorticity"		

Table A.39: (continued)

Name and Attributes	Type	Units	Description (and Occurrences)
external_field_f-ull	NXorientation		For complex external fields not satisfied by External_field_brief

A.2.38 NXshape

SVN Id Id: NXshape.nxdl.xml 373 2009-09-22 16:57:49Z Freddie Akeroyd

NXDL source: NXshape

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXshape.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: no included classes

documentation This is the description of the general shape and size of a component, which may be made up of "numobj" separate elements - it is used by the NXgeometry class

Table A.40: Tabular representation of NXshape[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
shape	nxcylinder | nxbox | nxsphere | nxcone		general shape of a component
size	NX_FLOAT	NX_LENGTH	physical extent of the object along its local axes (after NXorientation) with the center of mass at the local origin (after NXtranslation). The meaning and location of these axes will vary according to the value of the "shape" variable. nshapepar defines how many parameters. For the "nxcylinder" type the parameters are (diameter,height). For the "nxbox" type the parameters are (length,width,height). For the "nxsphere" type the parameters are (diameter). Dimensions: apparent size="2" • dim: value="numobj" value="" • dim: value="nshapepar" value=""

A.2.39 NXsource

SVN Id Id: NXsource.nxdl.xml 401 2009-10-18 16:37:32Z Pete Jemian

NXDL source: NXsource

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXsource.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXdata

NXgeometry

NXnote

documentation Template of the neutron or x-ray source, insertion devices and/or moderators.

Table A.41: Tabular representation of NXsource[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
distance	NX_FLOAT	NX_LENGTH	Effective distance from sample Distance as seen by radiation from sample. This number should be negative to signify that it is upstream of the sample.
name @short_name	NX_CHAR NX_CHAR		Name of source short name for source, perhaps the acronym
type	"Spallation Neutron Source" | "Pulsed Reactor Neutron Source" | "Reactor Neutron Source" | "Synchrotron X-ray Source" | "Pulsed Muon Source" | "Rotating Anode X-ray" | "Fixed Tube X-ray" | "UV Laser" | "Free-Electron Laser" | "Optical Laser"		type of radiation source
probe	neutron | x-ray | muon | electron | ultraviolet | "visible light"		type of radiation probe
power	NX_FLOAT	NX_POWER	Source power
current	NX_FLOAT	NX_CURRENT	Accelerator current
voltage	NX_FLOAT	NX_VOLTAGE	Accelerator voltage
frequency	NX_FLOAT	NX_FREQUENCY	Frequency of pulsed source
period	NX_FLOAT	NX_PERIOD	Period of pulsed source
target_material	Ta | W | depleted_U | enriched_U | Hg | Pb | C		Pulsed source target material
notes	NXnote		any source/facility related messages/events that occurred during the experiment
pulse_width	NX_FLOAT	NX_TIME	width of source pulse
pulse_shape	NXdata		source pulse shape
mode	"Single Bunch" | "Multi Bunch"		source operating mode
top_up	NX_BOOLEAN		Is the synchrotron operating in top_up mode?
geometry	NXgeometry		"Engineering" location of source

A.2.40 NXtranslation

SVN Id Id: NXtranslation.nxdl.xml 373 2009-09-22 16:57:49Z Freddie Akeroyd

NXDL source: NXtranslation

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXtranslation.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXgeometry

documentation This is the description for the general spatial location of a component - it is used by the NXgeometry class

Table A.42: Tabular representation of NXtranslation[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
geometry	NXgeometry		Link to other object if we are relative , else absent
distances	NX_FLOAT	NX_LENGTH	(x,y,z)}{This field and the angle field describe the position of a component. For absolute position, the origin is the scattering center (where a perfectly aligned sample would be) with the z-axis pointing downstream and the y-axis pointing gravitationally up. For a relative position the NXtranslation is taken into account before the NXorientation. The axes are right-handed and orthonormal. Dimensions: apparent size="2" • dim: value="numobj" value="" • dim: value="3" value=""

A.2.41 NXuser

SVN Id Id: NXuser.nxdl.xml 373 2009-09-22 16:57:49Z Freddie Akeroyd

NXDL source: NXuser

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXuser.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: no included classes

documentation Template of user's contact information. The format allows more than one user with the same affiliation and contact information, but a second NXuser group should be used if they have different affiliations, etc.

Table A.43: Tabular representation of NXuser[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
name	NX_CHAR		Name of user responsible for this entry
role	NX_CHAR		Role of user responsible for this entry. Suggested roles are "local_contact", "principal_investigator", and "proposer"
affiliation	NX_CHAR		Affiliation of user
address	NX_CHAR		Address of user
telephone_number	NX_CHAR		Telephone number of user
fax_number	NX_CHAR		Fax number of user

Table A.43: (continued)

Name and Attributes	Type	Units	Description (and Occurrences)
`email`	NX_CHAR		Email of user
`facility_user_id`	NX_CHAR		facility based unique identifier for this person e.g. their identification code on the facility address/contact database

A.2.42 NXvelocity_selector

SVN Id Id: NXvelocity_selector.nxdl.xml 421 2009-10-18 22:18:00Z Pete Jemian

NXDL source: NXvelocity_selector

URL: http://svn.nexusformat.org/definitions/trunk/base_classes/NXvelocity_selector.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXgeometry

documentation This is the description for a (typically neutron) velocity selector

Table A.44: Tabular representation of NXvelocity_selector[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
`type`	NX_CHAR		velocity selector type
`rotation_speed`	NX_FLOAT	NX_FREQUENCY	velocity selector rotation speed
`radius`	NX_FLOAT	NX_LENGTH	radius at beam centre
`spwidth`	NX_FLOAT	NX_LENGTH	spoke width at beam centre
`length`	NX_FLOAT	NX_LENGTH	rotor length
`num`	NX_INT	NX_UNITLESS	number of spokes/lamella
`twist`	NX_FLOAT	NX_ANGLE	twist angle along axis
`table`	NX_FLOAT	NX_ANGLE	offset vertical angle
`height`	NX_FLOAT	NX_LENGTH	input beam height
`width`	NX_FLOAT	NX_LENGTH	input beam width
`wavelength`	NX_FLOAT	NX_WAVELENGTH	wavelength
`wavelength_spread`	NX_FLOAT	NX_WAVELENGTH	deviation FWHM /Wavelength
`geometry`	NXgeometry		

A.3 NeXus Application Classes

A description of each NeXus Application Class is given.

A.3.1 NXarchive

SVN Id Id: NXarchive.nxdl.xml 350 2009-05-13 14:24:58Z Pete Jemian

NXDL source: NXarchive

URL: http://svn.nexusformat.org/definitions/trunk/applications/NXarchive.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXentry

NXinstrument

NXsample

NXsource

NXuser

documentation This is a definition for data to be archived by ICAT

Table A.45: Tabular representation of NXarchive[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
entry	NXentry		Look for special case table **entry[NXentry] (within NXarchive[definition])** below.
@index	NX_CHAR		

A.3.1.1 Special case table: entry[NXentry] (within NXarchive[definition])

Table A.46: Tabular representation of entry[NXentry] (within NXarchive[definition])

Name and Attributes	Type	Units	Description (and Occurrences)
title	NX_CHAR		
experiment_ident-ifer	NX_CHAR		unique identifier for the experiment
experiment_descr-iption	NX_CHAR		Brief description of the experiment and its objectives
collection_ident-ifier	NX_CHAR		ID of user or DAQ define group of data files
collection_descr-iption	NX_CHAR		Brief summary of the collection, including grouping criteria
entry_identifier	NX_CHAR		unique identifier for this measurement as provided by the facility
start_time	NX_DATE_TIME		
end_time	NX_DATE_TIME		
duration	NX_FLOAT	NX_TIME	
collection_time	NX_FLOAT	NX_TIME	
run_cycle	NX_CHAR		
revision	NX_CHAR		revision ID of this file, may be after recalibration, reprocessing etc.
definition	NXmonopd		Official NeXus DTD or NXDL schema to which this file conforms
program @version	NX_CHAR NX_CHAR		The program and version used for generating this file
release_date	NX_CHAR	NX_TIME	when this file is to be released into PD
user	NXuser		Look for special case table **user[NXuser] (within entry[NXentry])** below.

Table A.46: (continued)

Name and Attributes	Type	Units	Description (and Occurrences)
instrument	NXinstrument		Look for special case table **instrument[NXinstrument] (within entry[NXentry])** below.
sample	NXsample		Look for special case table **sample[NXsample] (within entry[NXentry])** below.

A.3.1.2 Special case table: user[NXuser] (within entry[NXentry])

Table A.47: Tabular representation of user[NXuser] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
name	NX_CHAR		
role	NX_CHAR		role of the user
facility_user_id	NX_CHAR		ID of the user in the facility burocracy database

A.3.1.3 Special case table: instrument[NXinstrument] (within entry[NXentry])

Table A.48: Tabular representation of instrument[NXinstrument] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
	NXsource		Look for special case table **[NXsource] (within instrument[NXinstrument])** below.
name	NX_CHAR		
description	NX_CHAR		Brief description of the instrument

A.3.1.4 Special case table: [NXsource] (within instrument[NXinstrument])

Table A.49: Tabular representation of [NXsource] (within instrument[NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
type	"Spallation Neutron Source" | "Pulsed Reactor Neutron Source" | "Reactor Neutron Source" | "Synchrotraon X-Ray Source" | "Pulsed Muon Source" | "Rotating Anode X-Ray" | "Fixed Tube X-Ray"		
name	NX_CHAR		
probe	neutron | x-ray | electron		

A.3.1.5 Special case table: sample[NXsample] (within entry[NXentry])

Table A.50: Tabular representation of sample[NXsample] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
name	NX_CHAR		Descriptive name of sample
sample_id	NX_CHAR		Unique database id of the sample
description	NX_CHAR		
type	sample | sample+can | "calibration sample" | "normalisation sample" | "simulated data" | none | sample_environment		
chemical_formula	NX_CHAR		Chemical formular fomatted according to CIF conventions
preparation_date	NX_CHAR	NX_TIME	
situation	NX_CHAR		Description of the environment the sample is in: air, vacuum, oxiding atmosphere, dehyrated, etc.
temperature	NX_FLOAT	NX_TEMPERATURE	
magnetic_field	NX_FLOAT	NX_CURRENT	
electric_field	NX_FLOAT	NX_VOLTAGE	
stress_field	NX_FLOAT	NX_UNITLESS	
pressure	NX_FLOAT	NX_PRESSURE	

A.3.2 NXiqproc

SVN Id Id: NXiqproc.nxdl.xml 350 2009-05-13 14:24:58Z Pete Jemian

NXDL source: NXiqproc

URL: http://svn.nexusformat.org/definitions/trunk/applications/NXiqproc.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXdata

> NXentry
>
> NXinstrument
>
> NXparameters
>
> NXprocess
>
> NXsample
>
> NXsource

documentation What is this? Actually this is atemplate to start an application definition from

Table A.51: Tabular representation of NXiqproc[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
	NXentry		Look for special case table **[NXentry] (within NXiqproc[definition])** below.
@entry	NX_CHAR		

A.3.2.1 Special case table: [NXentry] (within NXiqproc[definition])

Table A.52: Tabular representation of [NXentry] (within NXiqproc[definition])

Name and Attributes	Type	Units	Description (and Occurrences)
title	NX_CHAR		
definition	NXiqproc		Official NeXus DTD or NXDL schema to which this file conforms
instrument	NXinstrument		Look for special case table **instrument[NXinstrument] (within [NXentry])** below.
	NXsample		Look for special case table **[NXsample] (within [NXentry])** below.
reduction	NXprocess		Look for special case table **reduction[NXprocess] (within [NXentry])** below.
	NXdata		Look for special case table **[NXdata] (within [NXentry])** below.

A.3.2.2 Special case table: instrument[NXinstrument] (within [NXentry])

Table A.53:   Tabular representation of instrument[NXinstrument] (within [NX-entry])

Name and Attributes	Type	Units	Description (and Occurrences)
	NXsource		Look for special case table **[NXsource] (within instrument[NXinstrument])** below.
name	NX_CHAR		Name of the instrument from which this data was reduced.

A.3.2.3   Special case table: [NXsource] (within instrument[NXinstrument])

Table A.54:   Tabular representation of [NXsource] (within instrument[NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
type	NX_CHAR		
name	NX_CHAR		
probe	neutron | x-ray | electron		

A.3.2.4   Special case table: [NXsample] (within [NXentry])

Table A.55:   Tabular representation of [NXsample] (within [NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
name	NX_CHAR		Descriptive name of sample

A.3.2.5   Special case table: reduction[NXprocess] (within [NXentry])

Table A.56:   Tabular representation of reduction[NXprocess] (within [NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
program	NX_CHAR		
version	NX_CHAR		
input	NXparameters		Input parameters for the reduction program used Look for special case table **input[NXparameters] (within reduction[NXprocess])** below.
output	NXparameters		Eventual output parameters from the data reduction program used

A.3.2.6   Special case table: input[NXparameters] (within reduction[NXprocess])

Table A.57: Tabular representation of input[NXparameters] (within reduction[NXprocess])

Name and Attributes	Type	Units	Description (and Occurrences)
filenames	NX_CHAR		Raw data files used to generate this I(Q)

A.3.2.7 Special case table: [NXdata] (within [NXentry])

Table A.58: Tabular representation of [NXdata] (within [NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
data	NX_INT		This is I(Q). The client has to analyse the dimensions of I(Q). Often, multiple I(Q) for various environment conditions are measured; that would be the first dimension. Q can be multidimensional, this accounts for the further dimensions in the data Dimensions: size="3" • dim: index="1" value="NE" value="" • dim: index="2" value="NQX" value="" • dim: index="3" value="NQY" value=""
variable @varied_variable	NX_CHAR NX_CHAR		Dimensions: size="1" • dim: index="1" value="NE" value="" The real name of the varied variable in the first dim of data, temperature, P, MF etc...
qx	NX_CHAR		Values for the first dimension of Q Dimensions: size="1" • dim: index="1" value="NQX" value=""
qy	NX_CHAR		Values for the second dimension of Q Dimensions: size="1" • dim: index="1" value="NQY" value=""

A.3.3 NXmonopd

SVN Id Id: NXmonopd.nxdl.xml 439 2009-10-27 08:26:48Z Mark Koennecke

NXDL source: NXmonopd

URL: http://svn.nexusformat.org/definitions/trunk/applications/NXmonopd.nxdl.xml

NeXus Definition Language NXDL

extends class:

other classes included: NXcrystal

    NXdata

    NXdetector

    NXinstrument

    NXmonitor

    NXsample

    NXsource

documentation Monochromatic Neutron and X-Ray Powder Diffraction. Instrument definition for a powder diffractometer at a monochromatic neutron or X-ray beam. This is both suited for a powder diffractometer with a single detector or a powder diffractometer with a position sensitive detector.

Table A.59: Tabular representation of NXmonopd[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
title	NX_CHAR		
start_time	NX_DATE_TIME		
definition	NXmonopd		Official NeXus DTD or NXDL schema to which this file conforms
	NXinstrument		Look for special case table **[NXinstrument] (within NXmonopd[definition])** below.
	NXsample		Look for special case table **[NXsample] (within NXmonopd[definition])** below.
	NXmonitor		Look for special case table **[NXmonitor] (within NXmonopd[definition])** below.
	NXdata		

A.3.3.1 Special case table: [NXinstrument] (within NXmonopd[definition])

Table A.60: Tabular representation of [NXinstrument] (within NXmonopd[definition])

Name and Attributes	Type	Units	Description (and Occurrences)
	NXsource		Look for special case table **[NXsource] (within [NXinstrument])** below.
	NXcrystal		Look for special case table **[NXcrystal] (within [NXinstrument])** below.
	NXdetector		Look for special case table **[NXdetector] (within [NXinstrument])** below.

A.3.3.2 Special case table: [NXsource] (within [NXinstrument])

Table A.61: Tabular representation of [NXsource] (within [NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
type	NX_CHAR		
name	NX_CHAR		
probe	neutron | x-ray | electron		

A.3.3.3 Special case table: [NXcrystal] (within [NXinstrument])

Table A.62: Tabular representation of [NXcrystal] (within [NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
wavelength	NX_FLOAT	NX_WAVELENGTH	Optimum diffracted wavelength Dimensions: size="1" • dim: index="1" value="i" value=""

A.3.3.4 Special case table: [NXdetector] (within [NXinstrument])

Table A.63: Tabular representation of [NXdetector] (within [NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
polar_angle	NX_FLOAT		where ndet = number of detectors Dimensions: size="1" • dim: index="1" value="ndet" value=""
data	NX_INT		detector signal (usually counts) are already corrected for detector efficiency Dimensions: size="1" • dim: index="1" value="ndet" value=""

A.3.3.5 Special case table: [NXsample] (within NXmonopd[definition])

Table A.64: Tabular representation of [NXsample] (within NX-monopd[definition])

Name and Attributes	Type	Units	Description (and Occurrences)
name	NX_CHAR		Descriptive name of sample
rotation_angle	NX_FLOAT	NX_ANGLE	Optional rotation angle for the case when the powder diagram has been obtained through an omega-2theta scan like from a traditional single detector powder diffractometer

A.3.3.6 Special case table: [NXmonitor] (within NXmonopd[definition])

Table A.65: Tabular representation of [NXmonitor] (within NX-monopd[definition])

Name and Attributes	Type	Units	Description (and Occurrences)
mode	monitor | timer		Count to a preset value based on either clock time (timer) or received monitor counts (monitor).
preset	NX_FLOAT		preset value for time or monitor
integral	NX_FLOAT	NX_ANY	Total integral monitor counts

A.3.4 NXrefscan

SVN Id

NXDL source: NXrefscan

URL: http://svn.nexusformat.org/definitions/trunk/applications/NXrefscan.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXdata

> NXdetector
>
> NXentry
>
> NXinstrument
>
> NXmonitor
>
> NXmonochromator
>
> NXsample
>
> NXsource

documentation This is an application definition for a scanning reflectometer. It does not have the information to calculate the resolution since it does not have any apertures.

Table A.66: Tabular representation of NXrefscan[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
entry	NXentry		Look for special case table **entry[NXentry] (within NXrefscan[definition])** below.

A.3.4.1 Special case table: entry[NXentry] (within NXrefscan[definition])

Table A.67: Tabular representation of entry[NXentry] (within NXrefscan[definition])

Name and Attributes	Type	Units	Description (and Occurrences)
title	NX_CHAR		
start_time	NX_DATE_TIME		
end_time	NX_DATE_TIME		
definition	NXrefscan		Official NeXus DTD or NXDL schema to which this file conforms
instrument	NXinstrument		Look for special case table **instrument[NXinstrument] (within entry[NXentry])** below.
sample	NXsample		Look for special case table **sample[NXsample] (within entry[NXentry])** below.
control	NXmonitor		Look for special case table **control[NXmonitor] (within entry[NXentry])** below.
data	NXdata		

A.3.4.2 Special case table: instrument[NXinstrument] (within entry[NXentry])

Table A.68: Tabular representation of instrument[NXinstrument] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
	NXsource		Look for special case table **[NXsource] (within instrument[NXinstrument])** below.
monochromator	NXmonochromator		Look for special case table **monochromator[NXmonochromator] (within instrument[NXinstrument])** below.
	NXdetector		Look for special case table **[NXdetector] (within instrument[NXinstrument])** below.

A.3.4.3 Special case table: [NXsource] (within instrument[NXinstrument])

Table A.69: Tabular representation of [NXsource] (within instrument[NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
type	NX_CHAR		
name	NX_CHAR		
probe	neutron | x-ray | electron		

A.3.4.4 Special case table: monochromator[NXmonochromator] (within instrument[NXinstrument])

Table A.70: Tabular representation of monochromator[NXmonochromator] (within instrument[NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
wavelength	NX_FLOAT	NX_WAVELENGTH	

A.3.4.5 Special case table: [NXdetector] (within instrument[NXinstrument])

Table A.71: Tabular representation of [NXdetector] (within instrument[NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
data	NX_INT		Dimensions: size="1" • dim: index="1" value="NP" value=""

Table A.71: (continued)

Name and Attributes	Type	Units	Description (and Occurrences)
polar_angle	NX_FLOAT	NX_ANGLE	Dimensions: size="1" • dim: index="1" value="NP" value=""

A.3.4.6 Special case table: sample[NXsample] (within entry[NXentry])

Table A.72: Tabular representation of sample[NXsample] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
name	NX_CHAR		Descriptive name of sample
rotation_angle	NX_FLOAT	NX_ANGLE	Dimensions: size="1" • dim: index="1" value="NP" value=""

A.3.4.7 Special case table: control[NXmonitor] (within entry[NXentry])

Table A.73: Tabular representation of control[NXmonitor] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
mode	monitor | timer		Count to a preset value based on either clock time (timer) or received monitor counts (monitor).
preset	NX_FLOAT		preset value for time or monitor
data	NX_FLOAT	NX_ANY	Monitor counts for each step Dimensions: size="1" • dim: index="1" value="NP" value=""

A.3.5 NXreftof

SVN Id

NXDL source: NXreftof

URL: http://svn.nexusformat.org/definitions/trunk/applications/NXreftof.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXdata

NXdetector

NXdisk_chopper

NXentry

NXinstrument

NXmonitor

NXsample

documentation This is an application definition for raw data from a TOF reflectometer.

Table A.74: Tabular representation of NXreftof[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
entry	NXentry		Look for special case table **entry[NXentry] (within NXreftof[definition])** below.

A.3.5.1 Special case table: entry[NXentry] (within NXreftof[definition])

Table A.75: Tabular representation of entry[NXentry] (within NXreftof[definition])

Name and Attributes	Type	Units	Description (and Occurrences)
title	NX_CHAR		
start_time	NX_DATE_TIME		
end_time	NX_DATE_TIME		
definition	NXreftof		Official NeXus DTD or NXDL schema to which this file conforms
instrument	NXinstrument		Look for special case table **instrument[NXinstrument] (within entry[NXentry])** below.
sample	NXsample		Look for special case table **sample[NXsample] (within entry[NXentry])** below.
control	NXmonitor		Look for special case table **control[NXmonitor] (within entry[NXentry])** below.
data	NXdata		

A.3.5.2 Special case table: instrument[NXinstrument] (within entry[NXentry])

Table A.76: Tabular representation of instrument[NXinstrument] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
name	NX_CHAR		
chopper	NXdisk_chopper		Look for special case table **chopper[NXdisk_chopper] (within instrument[NXinstrument])** below.
detector	NXdetector		Look for special case table **detector[NXdetector] (within instrument[NXinstrument])** below.

A.3.5.3 Special case table: chopper[NXdisk_chopper] (within instrument[NXinstrument])

Table A.77: Tabular representation of chopper[NXdisk_chopper] (within instrument[NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
distance	NX_FLOAT	NX_LENGTH	Distance between chopper and sample

A.3.5.4 Special case table: detector[NXdetector] (within instrument[NXinstrument])

Table A.78: Tabular representation of detector[NXdetector] (within instrument[NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
data	NX_INT		Dimensions: size="3" • dim: index="1" value="xsize" value="" • dim: index="2" value="ysize" value="" • dim: index="3" value="nTOF" value=""
time_of_flight	NX_FLOAT	NX_TIME_OF_FLIGHT	Array of time values for each bin in a time-of-flight measurement Dimensions: size="1" • dim: index="1" value="nTOF" value=""
distance	NX_FLOAT	NX_LENGTH	
polar_angle	NX_FLOAT	NX_ANGLE	
x_pixel_size	NX_FLOAT	NX_LENGTH	
y_pixel_size	NX_FLOAT	NX_LENGTH	

A.3.5.5 Special case table: sample[NXsample] (within entry[NXentry])

Table A.79: Tabular representation of sample[NXsample] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
name	NX_CHAR		Descriptive name of sample
rotation_angle	NX_FLOAT	NX_ANGLE	

A.3.5.6 Special case table: control[NXmonitor] (within entry[NXentry])

Table A.80: Tabular representation of control[NXmonitor] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
mode	monitor | timer		Count to a preset value based on either clock time (timer) or received monitor counts (monitor).
preset	NX_FLOAT	NX_ANY	preset value for time or monitor
integral	NX_INT		Total integral monitor counts
time_of_flight	NX_FLOAT	NX_TIME_OF_FLIGHT	Time channels
data	NX_INT		Monitor counts in each time channel

A.3.6 NXsas

SVN Id

NXDL source: NXsas

URL: http://svn.nexusformat.org/definitions/trunk/applications/NXsas.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXcollimator

NXdata

NXdetector

NXentry

NXgeometry

NXinstrument

NXmonitor

NXmonochromator

NXsample

NXshape

NXsource

documentation This is an application definition for 2-D small angle scattering data collected with a monochromatic beam and an area detector. It is meant to be suitable both for neutron SANS and X-ray SAXS data.

Table A.81: Tabular representation of NXsas[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
	NXentry		Look for special case table **[NXentry] (within NXsas[definition])** below.
@entry	NX_CHAR		NeXus convention is to use "entry1", "entry2", ... for analysis software to locate each entry

A.3.6.1 Special case table: [NXentry] (within NXsas[definition])

Table A.82: Tabular representation of [NXentry] (within NXsas[definition])

Name and Attributes	Type	Units	Description (and Occurrences)
title	NX_CHAR		
start_time	NX_DATE_TIME		
end_time	NX_DATE_TIME		
definition	NXsas		Official NeXus DTD or NXDL schema to which this file conforms
instrument	NXinstrument		Look for special case table **instrument[NXinstrument] (within [NXentry])** below.

Table A.82: (continued)

Name and Attributes	Type	Units	Description (and Occurrences)
sample	NXsample		Look for special case table **sample[NXsample]** **(within [NXentry])** below.
control	NXmonitor		Look for special case table **control[NXmonitor]** **(within [NXentry])** below.
data	NXdata		

A.3.6.2 Special case table: instrument[NXinstrument] (within [NXentry])

Table A.83: Tabular representation of instrument[NXinstrument] (within [NX-entry])

Name and Attributes	Type	Units	Description (and Occurrences)
source	NXsource		Look for special case table **source[NXsource]** **(within instrument[NXinstrument])** below.
monochromator	NXmonochromator		Look for special case table **monochromator[NXmonochromator] (within instrument[NXinstrument])** below.
collimator	NXcollimator		Look for special case table **collimator[NXcollimator] (within instrument[NXinstrument])** below.
detector	NXdetector		Look for special case table **detector[NXdetector]** **(within instrument[NXinstrument])** below.
name	NX_CHAR		Name of the instrument actually used to perform the experiment

A.3.6.3 Special case table: source[NXsource] (within instrument[NXinstrument])

Table A.84: Tabular representation of source[NXsource] (within instrument[NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
type	NX_CHAR		type of radiation source
name	NX_CHAR		Name of the radiation source
probe	neutron | x-ray		

A.3.6.4 Special case table: monochromator[NXmonochromator] (within instrument[NXinstrument])

Table A.85: Tabular representation of monochromator[NXmonochromator] (within instrument[NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
wavelength	NX_FLOAT	NX_WAVELENGTH	The wavelength of the radiation
wavelength_spread	NX_FLOAT		delta_lambda/lambda. The wavelength spread. Important with neutrons to calculate resolution.

A.3.6.5 Special case table: collimator[NXcollimator] (within instrument[NXinstrument])

Table A.86: Tabular representation of collimator[NXcollimator] (within instrument[NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
geometry	NXgeometry		Look for special case table **geometry[NXgeometry] (within collimator[NXcollimator])** below.

A.3.6.6 Special case table: geometry[NXgeometry] (within collimator[NXcollimator])

Table A.87: Tabular representation of geometry[NXgeometry] (within collimator[NXcollimator])

Name and Attributes	Type	Units	Description (and Occurrences)
shape	NXshape		Look for special case table **shape[NXshape] (within geometry[NXgeometry])** below.

A.3.6.7 Special case table: shape[NXshape] (within geometry[NXgeometry])

Table A.88: Tabular representation of shape[NXshape] (within geometry[NXgeometry])

Name and Attributes	Type	Units	Description (and Occurrences)
shape	nxcylinder | nxbox		
size	NX_FLOAT	NX_LENGTH	The collimation length

A.3.6.8 Special case table: detector[NXdetector] (within instrument[NXinstrument])

Table A.89: Tabular representation of detector[NXdetector] (within instrument[NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
data	NX_INT		This is area detector data, of number of x-pixel versus number of y-pixels. Since the beam center is to be determined as a step of data reduction, it is not necessary to document or assume the position of the beam center in acquired data. Dimensions: size="2" • dim: index="1" value="nXPixel" value="" • dim: index="2" value="nYPixel" value=""
distance	NX_FLOAT	NX_LENGTH	The distance between detector and sample
x_pixel_size	NX_FLOAT	NX_LENGTH	Physical size of a pixel in x-direction
y_pixel_size	NX_FLOAT	NX_LENGTH	Size of a pixel in y direction

A.3.6.9 Special case table: sample[NXsample] (within [NXentry])

Table A.90: Tabular representation of sample[NXsample] (within [NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
name	NX_CHAR		Descriptive name of sample
thickness	NX_FLOAT	NX_LENGTH	sample thickness

A.3.6.10 Special case table: control[NXmonitor] (within [NXentry])

Table A.91: Tabular representation of control[NXmonitor] (within [NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
mode	monitor | timer		Count to a preset value based on either clock time (timer) or received monitor counts (monitor).
preset	NX_FLOAT		preset value for time or monitor
integral	NX_FLOAT	NX_ANY	Total integral monitor counts

A.3.7 NXscan

SVN Id

NXDL source: NXscan

URL: http://svn.nexusformat.org/definitions/trunk/applications/NXscan.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXdata

  NXdetector

  NXentry

  NXinstrument

NXmonitor

NXsample

documentation Application definition for a generic scan instrument. This definition is more an example then a stringent definition as the content of a given NeXus scan file needs to differ for different types of scans. This example definition shows a scan like done on a rotation camera: the sample is rotated and a detector image, the rotation angle and a monitor value is stored at each step in the scan. In the following I use the symbol NP as a placeholder for the number of scan points. These are the rules for storing scan data in NeXus files which are implemented in this example: - Each value varied throughout a scan is stored as an array of length NP at its respective location within the NeXus hierarchy. - For area detectors, NP is the first dimension, example for a detector of 256x256: data[NP,256,256] - The NXdata group contains links to all variables varied in the scan and the data. This to give an equivalent to the more familiar classical tabular representation of scans. These rules are there for a reason: HDF allows the first dimension of a data set to be unlimited. This means the data can be appended too. Thus a NeXus file built according to the rules given above can be used in the following way: - At the start of a scan, write all the static information. - At each scan point append new data from varied variables and the detector to the file.

Table A.92: Tabular representation of NXscan[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
	NXentry		Look for special case table **[NXentry] (within NXscan[definition])** below.

A.3.7.1 Special case table: [NXentry] (within NXscan[definition])

Table A.93: Tabular representation of [NXentry] (within NXscan[definition])

Name and Attributes	Type	Units	Description (and Occurrences)
title	NX_CHAR		
start_time	NX_DATE_TIME		
end_time	NX_DATE_TIME		
definition	NX_CHAR		Identifies the NXDL that describes the structure of this data file
@URL	NX_CHAR		Uniform Resource Locator for the base_class, application, or contributed NXDL that describes the structure of this data file
	NXinstrument		Look for special case table **[NXinstrument] (within [NXentry])** below.
	NXsample		Look for special case table **[NXsample] (within [NXentry])** below.
	NXmonitor		Look for special case table **[NXmonitor] (within [NXentry])** below.
	NXdata		

A.3.7.2 Special case table: [NXinstrument] (within [NXentry])

Table A.94: Tabular representation of [NXinstrument] (within [NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
	NXdetector		Look for special case table **[NXdetector] (within [NXinstrument])** below.

A.3.7.3 Special case table: [NXdetector] (within [NXinstrument])

Table A.95: Tabular representation of [NXdetector] (within [NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
data	NX_INT		Dimensions: size="3" • dim: index="1" value="NP" value="" • dim: index="2" value="xdim" value="" • dim: index="3" value="ydim" value=""

A.3.7.4 Special case table: [NXsample] (within [NXentry])

Table A.96: Tabular representation of [NXsample] (within [NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
rotation_angle	NX_FLOAT		Dimensions: size="1" • dim: index="1" value="NP" value=""

A.3.7.5 Special case table: [NXmonitor] (within [NXentry])

Table A.97: Tabular representation of [NXmonitor] (within [NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
data	NX_INT		Dimensions: size="1" • dim: index="1" value="NP" value=""

A.3.8 NXtas

SVN Id Id: NXtas.nxdl.xml 350 2009-05-13 14:24:58Z Pete Jemian

NXDL source: NXtas

URL: http://svn.nexusformat.org/definitions/trunk/applications/NXtas.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXcrystal

NXdata

NXdetector

> NXentry
>
> NXinstrument
>
> NXmonitor
>
> NXsample
>
> NXsource

documentation This is an application definition for a triple axis spectrometer. It is for the trademark scan of the TAS, the Q-E scan. For your alignment scans use the rules in NXscan.

Table A.98: Tabular representation of NXtas[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
entry	NXentry		Look for special case table **entry[NXentry] (within NXtas[definition])** below.

A.3.8.1 Special case table: entry[NXentry] (within NXtas[definition])

Table A.99: Tabular representation of entry[NXentry] (within NXtas[definition])

Name and Attributes	Type	Units	Description (and Occurrences)
title	NX_CHAR		
start_time	NX_DATE_TIME		
definition	NXtas		Official NeXus DTD or NXDL schema to which this file conforms
	NXinstrument		Look for special case table **[NXinstrument] (within entry[NXentry])** below.
	NXsample		Look for special case table **[NXsample] (within entry[NXentry])** below.
	NXmonitor		Look for special case table **[NXmonitor] (within entry[NXentry])** below.
	NXdata		One of the ei,ef,qh,qk,ql,en should get a primary=1 attribute to denote the main scan axis

A.3.8.2 Special case table: [NXinstrument] (within entry[NXentry])

Table A.100: Tabular representation of [NXinstrument] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
	NXsource		Look for special case table **[NXsource] (within [NXinstrument])** below.
monochromator	NXcrystal		Look for special case table **monochromator[NXcrystal] (within [NXinstrument])** below.

Table A.100: (continued)

Name and Attributes	Type	Units	Description (and Occurrences)
analyzer	NXcrystal		Look for special case table **analyzer[NXcrystal] (within [NXinstrument])** below.
	NXdetector		Look for special case table **[NXdetector] (within [NXinstrument])** below.

A.3.8.3   Special case table: [NXsource] (within [NXinstrument])

Table A.101: Tabular representation of [NXsource] (within [NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
name	NX_CHAR		
probe	neutron | x-ray		

A.3.8.4   Special case table: monochromator[NXcrystal] (within [NXinstrument])

Table A.102:   Tabular representation of monochromator[NXcrystal] (within [NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
ei	NX_FLOAT	NX_ENERGY	Dimensions: size="1" • dim: index="1" value="np" value=""
rotation_angle	NX_FLOAT	NX_ANGLE	Dimensions: size="1" • dim: index="1" value="np" value=""

A.3.8.5   Special case table: analyzer[NXcrystal] (within [NXinstrument])

Table A.103: Tabular representation of analyzer[NXcrystal] (within [NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
ef	NX_FLOAT	NX_ENERGY	Dimensions: size="1" • dim: index="1" value="np" value=""
rotation_angle	NX_FLOAT	NX_ANGLE	Dimensions: size="1" • dim: index="1" value="np" value=""
polar_angle	NX_FLOAT	NX_ANGLE	Dimensions: size="1" • dim: index="1" value="np" value=""

A.3.8.6   Special case table: [NXdetector] (within [NXinstrument])

Table A.104: Tabular representation of [NXdetector] (within [NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
data	NX_INT		Dimensions: size="1" • dim: index="1" value="np" value=""
polar_angle	NX_FLOAT	NX_ANGLE	Dimensions: size="1" • dim: index="1" value="np" value=""

A.3.8.7 Special case table: [NXsample] (within entry[NXentry])

Table A.105: Tabular representation of [NXsample] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
name	NX_CHAR		Descriptive name of sample
qh	NX_FLOAT	NX_DIMENSIONLESS	Dimensions: size="1" • dim: index="1" value="np" value=""
qk	NX_FLOAT	NX_DIMENSIONLESS	Dimensions: size="1" • dim: index="1" value="np" value=""
ql	NX_FLOAT	NX_DIMENSIONLESS	Dimensions: size="1" • dim: index="1" value="np" value=""
en	NX_FLOAT	NX_ENERGY	Dimensions: size="1" • dim: index="1" value="np" value=""
rotation_angle	NX_FLOAT	NX_ANGLE	Dimensions: size="1" • dim: index="1" value="np" value=""
polar_angle	NX_FLOAT	NX_ANGLE	Dimensions: size="1" • dim: index="1" value="np" value=""
sgu	NX_FLOAT	NX_ANGLE	Dimensions: size="1" • dim: index="1" value="np" value=""
sgl	NX_FLOAT	NX_ANGLE	Dimensions: size="1" • dim: index="1" value="np" value=""
unit_cell	NX_FLOAT	NX_LENGTH	Dimensions: size="1" • dim: index="1" value="6" value=""
orientation_matr-ix	NX_FLOAT	NX_DIMENSIONLESS	Dimensions: size="1" • dim: index="1" value="9" value=""

A.3.8.8 Special case table: [NXmonitor] (within entry[NXentry])

Table A.106: Tabular representation of [NXmonitor] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
mode	monitor | timer		Count to a preset value based on either clock time (timer) or received monitor counts (monitor).

Table A.106: (continued)

Name and Attributes	Type	Units	Description (and Occurrences)
preset	NX_FLOAT		preset value for time or monitor
data	NX_FLOAT	NX_ANY	Total integral monitor counts Dimensions: size="1" • dim: index="1" value="np" value=""

A.3.9 NXtofraw

SVN Id Id: NXtofraw.nxdl.xml 350 2009-05-13 14:24:58Z Pete Jemian

NXDL source: NXtofraw

URL: http://svn.nexusformat.org/definitions/trunk/applications/NXtofraw.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXdata

NXdetector

NXentry

NXinstrument

NXmonitor

NXsample

NXuser

documentation This is a application definition for raw data from a generic TOF instrument

Table A.107: Tabular representation of NXtofraw[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
entry	NXentry		Look for special case table **entry[NXentry] (within NXtofraw[definition])** below.

A.3.9.1 Special case table: entry[NXentry] (within NXtofraw[definition])

Table A.108: Tabular representation of entry[NXentry] (within NXtofraw[definition])

Name and Attributes	Type	Units	Description (and Occurrences)
title	NX_CHAR		
start_time	NX_DATE_TIME		
definition	NXtofraw		Official NeXus DTD or NXDL schema to which this file conforms
duration	NX_FLOAT		
run_number	NX_INT		
user	NXuser		Look for special case table **user[NXuser] (within entry[NXentry])** below.

Table A.108: (continued)

Name and Attributes	Type	Units	Description (and Occurrences)
	NXinstrument		Look for special case table **[NXinstrument] (within entry[NXentry])** below.
	NXsample		Look for special case table **[NXsample] (within entry[NXentry])** below.
	NXmonitor		Look for special case table **[NXmonitor] (within entry[NXentry])** below.
data	NXdata		

A.3.9.2 Special case table: user[NXuser] (within entry[NXentry])

Table A.109: Tabular representation of user[NXuser] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
name	NX_CHAR		

A.3.9.3 Special case table: [NXinstrument] (within entry[NXentry])

Table A.110: Tabular representation of [NXinstrument] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
detector	NXdetector		Look for special case table **detector[NXdetector] (within [NXinstrument])** below.

A.3.9.4 Special case table: detector[NXdetector] (within [NXinstrument])

Table A.111: Tabular representation of detector[NXdetector] (within [NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
data	NX_INT		Dimensions: size="2" • dim: index="1" value="ndet" value="" • dim: index="2" value="ntimechan" value=""
detector_number	NX_INT		Dimensions: size="1" • dim: index="1" value="ndet" value=""
distance	NX_FLOAT	NX_LENGTH	distance to sample for each detector Dimensions: size="1" • dim: index="1" value="ndet" value=""

Table A.111: (continued)

Name and Attributes	Type	Units	Description (and Occurrences)
time_of_flight	NX_FLOAT	NX_TIME_OF_FLIGHT	Dimensions: size="1" • dim: index="1" value="ntimechan" value=""
polar_angle	NX_FLOAT	NX_ANGLE	polar angle for each detector element Dimensions: size="1" • dim: index="1" value="ndet" value=""
azimuthal_angle	NX_FLOAT	NX_ANGLE	azimuthal angle for each detector element Dimensions: size="1" • dim: index="1" value="ndet" value=""
arrangement	NX_INT		This is new!! This array maps detector numbers to a position relative to a possible area geometry of the detector. A linear arrangement is covered too; then ny is 1 Dimensions: size="2" • dim: index="1" value="nx" value="" • dim: index="2" value="ny" value=""

A.3.9.5 Special case table: [NXsample] (within entry[NXentry])

Table A.112: Tabular representation of [NXsample] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
name	NX_CHAR		Descriptive name of sample
nature	powder | liquid | "single crystal"		

A.3.9.6 Special case table: [NXmonitor] (within entry[NXentry])

Table A.113: Tabular representation of [NXmonitor] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
mode	monitor | timer		Count to a preset value based on either clock time (timer) or received monitor counts (monitor).
preset	NX_FLOAT		preset value for time or monitor
distance	NX_FLOAT	NX_LENGTH	
data	NX_INT		Dimensions: size="1" • dim: index="1" value="ntimechan" value=""
time_of_flight	NX_FLOAT	NX_TIME_OF_FLIGHT	Dimensions: size="1" • dim: index="1" value="ntimechan" value=""

A.3.10 NXtomo

SVN Id

NXDL source: NXtomo

URL: http://svn.nexusformat.org/definitions/trunk/applications/NXtomo.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXdata

    NXdetector

    NXentry

    NXinstrument

    NXmonitor

    NXsample

    NXsource

documentation This is the application definition for x-ray or neutron tomography raw data. In tomography first some dark field images are measured, some bright field images and, of course the sample. In order to properly sort the order of the images taken, a sequence number is stored with each image.

Table A.114: Tabular representation of NXtomo[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
entry	NXentry		Look for special case table **entry[NXentry] (within NXtomo[definition])** below.

A.3.10.1 Special case table: entry[NXentry] (within NXtomo[definition])

Table A.115:   Tabular representation of entry[NXentry] (within NXtomo[definition])

Name and Attributes	Type	Units	Description (and Occurrences)
title	NX_CHAR		
start_time	NX_DATE_TIME		
end_time	NX_DATE_TIME		
definition	NXtomo		Official NeXus DTD or NXDL schema to which this file conforms
instrument	NXinstrument		Look for special case table **instrument[NXinstrument] (within entry[NXentry])** below.
sample	NXsample		Look for special case table **sample[NXsample] (within entry[NXentry])** below.
control	NXmonitor		Look for special case table **control[NXmonitor] (within entry[NXentry])** below.
data	NXdata		

A.3.10.2   Special case table: instrument[NXinstrument] (within entry[NXentry])

Table A.116:   Tabular representation of instrument[NXinstrument] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
	NXsource		Look for special case table **[NXsource] (within instrument[NXinstrument])** below.
bright_field	NXdetector		Look for special case table **bright_field[NXdetector] (within instrument[NXinstrument])** below.
dark_field	NXdetector		Look for special case table **dark_field[NXdetector] (within instrument[NXinstrument])** below.
sample	NXdetector		Look for special case table **sample[NXdetector] (within instrument[NXinstrument])** below.

A.3.10.3   Special case table: [NXsource] (within instrument[NXinstrument])

Table A.117:   Tabular representation of [NXsource] (within instrument[NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
type	NX_CHAR		
name	NX_CHAR		
probe	neutron | x-ray | electron		

A.3.10.4   Special case table: bright_field[NXdetector] (within instrument[NXinstrument])

Table A.118:   Tabular representation of bright_field[NXdetector] (within instrument[NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
data	NX_INT		Dimensions: size="3" • dim: index="1" value="nBrightFrames" value="" • dim: index="2" value="xsize" value="" • dim: index="3" value="ysize" value=""
sequence_number	NX_CHAR		In order to properly sort the order of the images taken in (for example) a tomography experiment, a sequence number is stored with each image. Dimensions: size="1" • dim: index="1" value="nBrightFrames" value=""

A.3.10.5   Special case table: dark_field[NXdetector] (within instrument[NXinstrument])

Table A.119:   Tabular representation of dark_field[NXdetector] (within instrument[NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
data	NX_INT		Dimensions: size="3" • dim: index="1" value="nDarkFrames" value="" • dim: index="2" value="xsize" value="" • dim: index="3" value="ysize" value=""
sequence_number	NX_CHAR		Dimensions: size="1" • dim: index="1" value="nDarkFrames" value=""

A.3.10.6 Special case table: sample[NXdetector] (within instrument[NXinstrument])

Table A.120:   Tabular representation of sample[NXdetector] (within instrument[NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
data	NX_INT		Dimensions: size="3" • dim: index="1" value="nSampleFrames" value="" • dim: index="2" value="xsize" value="" • dim: index="3" value="ysize" value=""
sequence_number	NX_CHAR		Dimensions: size="1" • dim: index="1" value="nSampleFrames" value=""
x_pixel_size	NX_FLOAT	NX_LENGTH	
y_pixel_size	NX_FLOAT	NX_LENGTH	
distance	NX_FLOAT	NX_LENGTH	Distance between detector and sample

A.3.10.7 Special case table: sample[NXsample] (within entry[NXentry])

Table A.121:   Tabular representation of sample[NXsample] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
name	NX_CHAR		Descriptive name of sample
rotation_angle	NX_FLOAT	NX_ANGLE	Dimensions: size="1" • dim: index="1" value="nSampleFrames" value=""

A.3.10.8 Special case table: control[NXmonitor] (within entry[NXentry])

Table A.122:   Tabular representation of control[NXmonitor] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
integral	NX_FLOAT	NX_ANY	Total integral monitor counts for each measured frame. This is in order to correct for fluctuations in the beam between frames Dimensions: size="1" • dim: index="1" value="nDarkFrames + nBrightFrames + nSampleFrame" value=""

A.3.11 NXtomophase

SVN Id

NXDL source: NXtomophase

URL: http://svn.nexusformat.org/definitions/trunk/applications/NXtomophase.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXdata

> NXdetector
>
> NXentry
>
> NXinstrument
>
> NXmonitor
>
> NXsample
>
> NXsource

documentation This is the application definition for x-ray or neutron tomography raw data with phase contrast variation at each point. In tomography first some dark field images are measured, some bright field images and, of course the sample. In order to properly sort the order of the images taken, a sequence number is stored with each image.

Table A.123: Tabular representation of NXtomophase[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
entry	NXentry		Look for special case table **entry[NXentry] (within NXtomophase[definition])** below.

A.3.11.1 Special case table: entry[NXentry] (within NXtomophase[definition])

Table A.124: Tabular representation of entry[NXentry] (within NXtomophase[definition])

Name and Attributes	Type	Units	Description (and Occurrences)
title	NX_CHAR		
start_time	NX_DATE_TIME		
end_time	NX_DATE_TIME		
definition	NXtomophase		Official NeXus DTD or NXDL schema to which this file conforms
instrument	NXinstrument		Look for special case table **instrument[NXinstrument] (within entry[NXentry])** below.
sample	NXsample		Look for special case table **sample[NXsample] (within entry[NXentry])** below.
control	NXmonitor		Look for special case table **control[NXmonitor] (within entry[NXentry])** below.
data	NXdata		

A.3.11.2 Special case table: instrument[NXinstrument] (within entry[NXentry])

Table A.125: Tabular representation of instrument[NXinstrument] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
	NXsource		Look for special case table **[NXsource] (within instrument[NXinstrument])** below.
bright_field	NXdetector		Look for special case table **bright_field[NXdetector] (within instrument[NXinstrument])** below.
dark_field	NXdetector		Look for special case table **dark_field[NXdetector] (within instrument[NXinstrument])** below.
sample	NXdetector		Look for special case table **sample[NXdetector] (within instrument[NXinstrument])** below.

A.3.11.3 Special case table: [NXsource] (within instrument[NXinstrument])

Table A.126: Tabular representation of [NXsource] (within instrument[NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
type	NX_CHAR		
name	NX_CHAR		
probe	neutron | x-ray | electron		

A.3.11.4 Special case table: bright_field[NXdetector] (within instrument[NXinstrument])

Table A.127: Tabular representation of bright_field[NXdetector] (within instrument[NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
data	NX_INT		Dimensions: size="3" • dim: index="1" value="nBrightFrames" value="" • dim: index="2" value="xsize" value="" • dim: index="3" value="ysize" value=""
sequence_number	NX_CHAR		Dimensions: size="1" • dim: index="1" value="nBrightFrames" value=""

A.3.11.5 Special case table: dark_field[NXdetector] (within instrument[NXinstrument])

Table A.128: Tabular representation of dark_field[NXdetector] (within instrument[NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
data	NX_INT		Dimensions: size="3" • dim: index="1" value="nDarkFrames" value="" • dim: index="2" value="xsize" value="" • dim: index="3" value="ysize" value=""
sequence_number	NX_CHAR		Dimensions: size="1" • dim: index="1" value="nDarkFrames" value=""

A.3.11.6 Special case table: sample[NXdetector] (within instrument[NXinstrument])

Table A.129: Tabular representation of sample[NXdetector] (within instrument[NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
data	NX_INT		Dimensions: size="4" • dim: index="1" value="nSampleFrames" value="" • dim: index="2" value="nPhase" value="" • dim: index="3" value="xsize" value="" • dim: index="4" value="ysize" value=""
sequence_number	NX_CHAR		Dimensions: size="2" • dim: index="1" value="nSampleFrames" value="" • dim: index="2" value="nPhase" value=""
x_pixel_size	NX_FLOAT	NX_LENGTH	
y_pixel_size	NX_FLOAT	NX_LENGTH	
distance	NX_FLOAT	NX_LENGTH	Distance between detector and sample

A.3.11.7 Special case table: sample[NXsample] (within entry[NXentry])

Table A.130: Tabular representation of sample[NXsample] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
name	NX_CHAR		Descriptive name of sample
rotation_angle	NX_FLOAT	NX_ANGLE	Dimensions: size="1" • dim: index="1" value="nSampleFrames" value=""

A.3.11.8 Special case table: control[NXmonitor] (within entry[NXentry])

Table A.131: Tabular representation of control[NXmonitor] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
integral	NX_FLOAT	NX_ANY	Total integral monitor counts for each measured frame. This is in order to correct for fluctuations in the beam between frames Dimensions: size="1" • dim: index="1" value="nDarkFrames + nBrightFrames + nSampleFrame" value=""

A.3.12 NXtomoproc

SVN Id

NXDL source: NXtomoproc

URL: http://svn.nexusformat.org/definitions/trunk/applications/NXtomoproc.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXdata

> NXentry
>
> NXinstrument
>
> NXparameters
>
> NXprocess
>
> NXsample
>
> NXsource

documentation This is an application definition for the final result of a tomography experiment: a 3D construction of some volume of physical properties.

Table A.132: Tabular representation of NXtomoproc[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
entry	NXentry		Look for special case table **entry[NXentry] (within NXtomoproc[definition])** below.

A.3.12.1 Special case table: entry[NXentry] (within NXtomoproc[definition])

Table A.133: Tabular representation of entry[NXentry] (within NXtomoproc[definition])

Name and Attributes	Type	Units	Description (and Occurrences)
title	NX_CHAR		
definition	NXtomoproc		Official NeXus DTD or NXDL schema to which this file conforms

Table A.133: (continued)

Name and Attributes	Type	Units	Description (and Occurrences)
	NXinstrument		Look for special case table **[NXinstrument] (within entry[NXentry])** below.
	NXsample		Look for special case table **[NXsample] (within entry[NXentry])** below.
reconstruction	NXprocess		Look for special case table **reconstruction[NXprocess] (within entry[NXentry])** below.
data	NXdata		Look for special case table **data[NXdata] (within entry[NXentry])** below.

A.3.12.2 Special case table: [NXinstrument] (within entry[NXentry])

Table A.134: Tabular representation of [NXinstrument] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
	NXsource		Look for special case table **[NXsource] (within [NXinstrument])** below.

A.3.12.3 Special case table: [NXsource] (within [NXinstrument])

Table A.135: Tabular representation of [NXsource] (within [NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
type	NX_CHAR		
name	NX_CHAR		
probe	neutron | x-ray | electron		

A.3.12.4 Special case table: [NXsample] (within entry[NXentry])

Table A.136: Tabular representation of [NXsample] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
name	NX_CHAR		Descriptive name of sample

A.3.12.5 Special case table: reconstruction[NXprocess] (within entry[NXentry])

Table A.137: Tabular representation of reconstruction[NXprocess] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
program	NX_CHAR		Name of the program used for reconstruction
version	NX_CHAR		Version of the program used
date	NX_DATE_TIME		Date and time of reconstruction processing.
parameters	NXparameters		Look for special case table **parameters[NXparameters] (within reconstruction[NXprocess])** below.

A.3.12.6 Special case table: parameters[NXparameters] (within reconstruction[NXprocess])

Table A.138: Tabular representation of parameters[NXparameters] (within reconstruction[NXprocess])

Name and Attributes	Type	Units	Description (and Occurrences)
raw_file	NX_CHAR		Original raw data file this data was derived from

A.3.12.7 Special case table: data[NXdata] (within entry[NXentry])

Table A.139: Tabular representation of data[NXdata] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
data	NX_INT		This is the reconstructed volume. This can be different things. Please indicates in the unit attribute what physical quantity this really is. Dimensions: size="3" • dim: index="1" value="nx" value="" • dim: index="2" value="nx" value="" • dim: index="3" value="nz" value=""
scaling_factor	NX_FLOAT		The elements in data are usually float values really. For efficiency reasons these are usually stored as integers after scaling with a scale factor. This value is this scale factor. It is required to get the actual physical value, when necessary.
offset	NX_FLOAT		An eventuell offset to apply to the data in data
x	NX_FLOAT	NX_ANY	This is an array holding the values to use for the x-axis of data. The units must be appropriate for the measurement. Dimensions: size="1" • dim: index="1" value="nx" value=""
y	NX_FLOAT	NX_ANY	This is an array holding the values to use for the y-axis of data. The units must be appropriate for the measurement. Dimensions: size="1" • dim: index="1" value="ny" value=""

Table A.139: (continued)

Name and Attributes	Type	Units	Description (and Occurrences)
z	NX_FLOAT	NX_ANY	This is an array holding the values to use for the z-axis of data. The units must be appropriate for the measurement. Dimensions: size="1" • dim: index="1" value="nz" value=""

A.3.13 NXxbase

SVN Id

NXDL source: NXxbase

URL: http://svn.nexusformat.org/definitions/trunk/applications/NXxbase.nxdl.xml

NeXus Definition Language NXDL

extends class: NXobject

other classes included: NXdata

    NXdetector

    NXentry

    NXinstrument

    NXmonitor

    NXmonochromator

    NXsample

    NXsource

documentation This definition covers the common parts of all monochromatic single crystal raw data application definitions

Table A.140: Tabular representation of NXxbase[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
entry	NXentry		Look for special case table **entry[NXentry] (within NXxbase[definition])** below.

A.3.13.1 Special case table: entry[NXentry] (within NXxbase[definition])

Table A.141: Tabular representation of entry[NXentry] (within NXxbase[definition])

Name and Attributes	Type	Units	Description (and Occurrences)
title	NX_CHAR		
start_time	NX_DATE_TIME		
end_time	NX_DATE_TIME		
definition	NXxbase		Official NeXus DTD or NXDL schema to which this file conforms

Table A.141: (continued)

Name and Attributes	Type	Units	Description (and Occurrences)
instrument	NXinstrument		Look for special case table **instrument[NXinstrument] (within entry[NXentry])** below.
sample	NXsample		Look for special case table **sample[NXsample] (within entry[NXentry])** below.
control	NXmonitor		Look for special case table **control[NXmonitor] (within entry[NXentry])** below.
	NXdata		The name of this group id data if there is only one detector; if there are several the names will be data1, data2, data3 and will data will point to the corresponding detector groups in the instrument hierarchy.

A.3.13.2   Special case table: instrument[NXinstrument] (within entry[NXentry])

Table A.142:   Tabular representation of instrument[NXinstrument] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
source	NXsource		Look for special case table **source[NXsource] (within instrument[NXinstrument])** below.
monochromator	NXmonochromator		Look for special case table **monochromator[NXmonochromator] (within instrument[NXinstrument])** below.
detector	NXdetector		The name of the group is detector if there is only one detector, if there are several, names have to be detector1, detector2, ...detectorn Occurences: 1 : 99999999999999999999 Look for special case table **detector[NXdetector] (within instrument[NXinstrument])** below.

A.3.13.3   Special case table: source[NXsource] (within instrument[NXinstrument])

Table A.143:   Tabular representation of source[NXsource] (within instrument[NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
type	NX_CHAR		
name	NX_CHAR		
probe	neutron | x-ray | electron		

A.3.13.4 Special case table: monochromator[NXmonochromator] (within instrument[NXinstrument])

Table A.144:   Tabular representation of monochromator[NXmonochromator] (within instrument[NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
wavelength	NX_FLOAT	NX_WAVELENGTH	

A.3.13.5 Special case table: detector[NXdetector] (within instrument[NXinstrument])

Table A.145:   Tabular representation of detector[NXdetector] (within instrument[NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
data	NX_INT		The area detector data, the first dimension is always the number of scan points, the second and third are the number of pixels in x and y. The origin is always assumed to be in the center of the detector. maxOccurs is limited to the the number of detectors on your instrument Dimensions: size="3" • dim: index="1" value="np" value="" • dim: index="2" value="number of x pixels" value="" • dim: index="3" value="number of y pixels" value=""
@signal	NX_CHAR		
x_pixel_size	NX_FLOAT	NX_LENGTH	
y_pixel_size	NX_FLOAT	NX_LENGTH	
distance	NX_FLOAT	NX_LENGTH	

A.3.13.6 Special case table: sample[NXsample] (within entry[NXentry])

Table A.146:   Tabular representation of sample[NXsample] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
name	NX_CHAR		Descriptive name of sample
orientation_matrix	NX_FLOAT		The orientation matrix according to Busing and Levy conventions. This is not strictly necessary as the UB can always be derived from the data. But let us bow to common usage which includes thie UB nearly always. Dimensions: size="2" • dim: index="1" value="3" value="" • dim: index="2" value="3" value=""
unit_cell	NX_FLOAT		The unit cell, a, b ,b, alpha, beta, gamma. Again, not strictly necessary, but normally written. Dimensions: size="1" • dim: index="1" value="6" value=""

A.3.13.7 Special case table: control[NXmonitor] (within entry[NXentry])

Table A.147: Tabular representation of control[NXmonitor] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
mode	monitor | timer		Count to a preset value based on either clock time (timer) or received monitor counts (monitor).
preset	NX_FLOAT		preset value for time or monitor
integral	NX_FLOAT	NX_ANY	Total integral monitor counts

A.3.14 NXxeuler

SVN Id

NXDL source: NXxeuler

URL: http://svn.nexusformat.org/definitions/trunk/applications/NXxeuler.nxdl.xml

NeXus Definition Language NXDL

extends class: NXxbase

other classes included: NXdata

NXdetector

NXentry

NXinstrument

NXsample

documentation This is the application definition for raw data from a four circle diffractometer with an eulerian cradle. It extends NXxbase, so the full definition is the content of NXxbase plus the data defined here. All four angles are logged in order to support arbitray scans in reciprocal space.

Table A.148: Tabular representation of NXxeuler[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
entry	NXentry		Look for special case table **entry[NXentry] (within NXxeuler[definition])** below.

A.3.14.1 Special case table: entry[NXentry] (within NXxeuler[definition])

Table A.149: Tabular representation of entry[NXentry] (within NXxeuler[definition])

Name and Attributes	Type	Units	Description (and Occurrences)
definition	NXxeuler		Official NeXus DTD or NXDL schema to which this file conforms

Name and Attributes	Type	Units	Description (and Occurrences)
instrument	NXinstrument		Look for special case table **instrument[NXinstrument] (within entry[NXentry])** below.
sample	NXsample		Look for special case table **sample[NXsample] (within entry[NXentry])** below.
name	NXdata		

A.3.14.2 Special case table: instrument[NXinstrument] (within entry[NXentry])

Table A.150:   Tabular representation of instrument[NXinstrument] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
detector	NXdetector		Look for special case table **detector[NXdetector] (within instrument[NXinstrument])** below.

A.3.14.3 Special case table: detector[NXdetector] (within instrument[NXinstrument])

Table A.151:   Tabular representation of detector[NXdetector] (within instrument[NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
polar_angle	NX_FLOAT	NX_ANGLE	The polar_angle (two theta) where the detector is placed at each scan point. Dimensions: size="1" • dim: index="1" value="np" value=""

A.3.14.4 Special case table: sample[NXsample] (within entry[NXentry])

Table A.152:   Tabular representation of sample[NXsample] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
rotation_angle	NX_FLOAT	NX_ANGLE	This is an array holding the sample rotation angle at each scan point Dimensions: size="1" • dim: index="1" value="np" value=""
chi	NX_FLOAT	NX_ANGLE	This is an array holding the chi angle of the eulerian cradle at each scan point Dimensions: size="1" • dim: index="1" value="np" value=""

Table A.152: (continued)

Name and Attributes	Type	Units	Description (and Occurrences)
phi	NX_FLOAT	NX_ANGLE	This is an array holding the phi rotation of the eulerain cradle at each scan point Dimensions: size="1" • dim: index="1" value="np" value=""

A.3.15   NXxkappa

SVN Id

NXDL source: NXxkappa

URL: http://svn.nexusformat.org/definitions/trunk/applications/NXxkappa.nxdl.xml

NeXus Definition Language NXDL

extends class: NXxbase

other classes included: NXdata

> NXdetector
>
> NXentry
>
> NXinstrument
>
> NXsample

documentation This is the application definition for raw data from a kappa geometry (CAD4) single crystal diffractometer. It extends NXxbase, so the full definition is the content of NXxbase plus the data defined here.

Table A.153: Tabular representation of NXxkappa[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
entry	NXentry		Look for special case table **entry[NXentry] (within NXxkappa[definition])** below.

A.3.15.1   Special case table: entry[NXentry] (within NXxkappa[definition])

Table A.154:   Tabular representation of entry[NXentry] (within NXxkappa[definition])

Name and Attributes	Type	Units	Description (and Occurrences)
definition	NXxkappa		Official NeXus DTD or NXDL schema to which this file conforms
instrument	NXinstrument		Look for special case table **instrument[NXinstrument] (within entry[NXentry])** below.
sample	NXsample		Look for special case table **sample[NXsample] (within entry[NXentry])** below.
name	NXdata		

A.3.15.2 Special case table: instrument[NXinstrument] (within entry[NXentry])

Table A.155: Tabular representation of instrument[NXinstrument] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
detector	NXdetector		Look for special case table **detector[NXdetector] (within instrument[NXinstrument])** below.

A.3.15.3 Special case table: detector[NXdetector] (within instrument[NXinstrument])

Table A.156: Tabular representation of detector[NXdetector] (within instrument[NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
polar_angle	NX_FLOAT	NX_ANGLE	The polar_angle (two theta) at each scan point Dimensions: size="1" • dim: index="1" value="np" value=""

A.3.15.4 Special case table: sample[NXsample] (within entry[NXentry])

Table A.157: Tabular representation of sample[NXsample] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
rotation_angle	NX_FLOAT	NX_ANGLE	This is an array holding the sample rotation angle at each scan point Dimensions: size="1" • dim: index="1" value="np" value=""
kappa	NX_FLOAT	NX_ANGLE	This is an array holding the kappa angle at each scan point Dimensions: size="1" • dim: index="1" value="np" value=""
phi	NX_FLOAT	NX_ANGLE	This is an array holding the phi angle at each scan point Dimensions: size="1" • dim: index="1" value="np" value=""
alpha	NX_FLOAT	NX_ANGLE	This holds the inclination angle of the kappa arm.

A.3.16 NXxnb

SVN Id

NXDL source: NXxnb

URL: http://svn.nexusformat.org/definitions/trunk/applications/NXxnb.nxdl.xml

NeXus Definition Language NXDL

extends class: NXxbase

other classes included: NXdata

> NXdetector
>
> NXentry
>
> NXinstrument
>
> NXsample

documentation This is the application definition for raw data from a single crystal diffractometer measuring in normal beam mode. It extends NXxbase, so the full definition is the content of NXxbase plus the data defined here. All angles are logged in order to support arbitray scans in reciprocal space.

Table A.158: Tabular representation of NXxnb[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
entry	NXentry		Look for special case table **entry[NXentry] (within NXxnb[definition])** below.

A.3.16.1 Special case table: entry[NXentry] (within NXxnb[definition])

Table   A.159:   Tabular   representation   of   entry[NXentry]   (within NXxnb[definition])

Name and Attributes	Type	Units	Description (and Occurrences)
definition	NXxnb		Official NeXus DTD or NXDL schema to which this file conforms
instrument	NXinstrument		Look for special case table **instrument[NXinstrument] (within entry[NXentry])** below.
sample	NXsample		Look for special case table **sample[NXsample] (within entry[NXentry])** below.
name	NXdata		

A.3.16.2 Special case table: instrument[NXinstrument] (within entry[NXentry])

Table A.160: Tabular representation of instrument[NXinstrument] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
detector	NXdetector		Look for special case table **detector[NXdetector] (within instrument[NXinstrument])** below.

A.3.16.3 Special case table: detector[NXdetector] (within instrument[NXinstrument])

Table A.161:   Tabular representation of detector[NXdetector] (within instrument[NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
polar_angle	NX_FLOAT	NX_ANGLE	The polar_angle (gamma) of the detector for each scan point. Dimensions: size="1" • dim: index="1" value="np" value=""
tilt_angle	NX_FLOAT	NX_ANGLE	The angle by which the detector has been tilted out of the scattering plane. Dimensions: size="1" • dim: index="1" value="np" value=""

A.3.16.4 Special case table: sample[NXsample] (within entry[NXentry])

Table A.162:   Tabular representation of sample[NXsample] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
rotation_angle	NX_FLOAT	NX_ANGLE	This is an array holding the sample rotation angle at each scan point Dimensions: size="1" • dim: index="1" value="np" value=""

A.3.17   NXxrot

SVN Id

NXDL source: NXxrot

URL: http://svn.nexusformat.org/definitions/trunk/applications/NXxrot.nxdl.xml

NeXus Definition Language NXDL

extends class: NXxbase

other classes included: NXdata

> NXdetector
>
> NXentry
>
> NXinstrument
>
> NXsample

documentation This is the application definition for raw data from a rotation camera. It extends NXxbase, so the full definition is the content of NXxbase plus the data defined here.

Table A.163:   Tabular representation of NXxrot[definition]

Name and Attributes	Type	Units	Description (and Occurrences)
entry	NXentry		Look for special case table **entry[NXentry] (within NXxrot[definition])** below.

A.3.17.1   Special case table: entry[NXentry] (within NXxrot[definition])

Table A.164:   Tabular representation of entry[NXentry] (within NXxrot[definition])

Name and Attributes	Type	Units	Description (and Occurrences)
definition	NXxrot		Official NeXus DTD or NXDL schema to which this file conforms
instrument	NXinstrument		Look for special case table **instrument[NXinstrument] (within entry[NXentry])** below.
sample	NXsample		Look for special case table **sample[NXsample] (within entry[NXentry])** below.
name	NXdata		

A.3.17.2   Special case table: instrument[NXinstrument] (within entry[NXentry])

Table A.165:   Tabular representation of instrument[NXinstrument] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
detector	NXdetector		Look for special case table **detector[NXdetector] (within instrument[NXinstrument])** below.

A.3.17.3   Special case table: detector[NXdetector] (within instrument[NXinstrument])

Table A.166:   Tabular representation of detector[NXdetector] (within instrument[NXinstrument])

Name and Attributes	Type	Units	Description (and Occurrences)
polar_angle	NX_FLOAT	NX_ANGLE	The polar_angle (two theta) where the detector is placed.

A.3.17.4   Special case table: sample[NXsample] (within entry[NXentry])

Table A.167:   Tabular representation of sample[NXsample] (within entry[NXentry])

Name and Attributes	Type	Units	Description (and Occurrences)
rotation_angle	NX_FLOAT	NX_ANGLE	This is an array holding the sample rotation angle at each scan point Dimensions: size="1" • dim: index="1" value="np" value=""

A.4 NeXus Contributed Classes

A description of each NeXus Contributed Class is given.

Appendix B

Roadmap

blah blah

Appendix C

Brief history of the NeXus format

August 1994 Jon Tischler (ORNL) proposed an HDF-based format as a standard for data storage at APS

June 1994 Mark Koennecke (PSI) made a similar proposal using netCDF for the European neutron scattering community while working at ISIS

1995 and 1996 This was the basis for the current designed which was developed at SoftNeSS 1995 (at NIST) and SoftNeSS 1996 (at ANL)

August 1996 NeXus Abstract Programmer Interface (NAPI) released

October 1996 Przemek Klosowski (NCNR) produced a first draft of the NeXus proposal drawing on ideas from both sources

July 1997 SINQ at PSI started writing NeXus files to store raw data

summer 2001 MLNSC at LANL started writing NeXus files to store raw data

September 2002 NeXus API version 2.0.0 is released

October 2003 NeXus International Advisory Committee (NIAC) formed and first meeting held at CalTech

July 2005 NeXus API version 3.0.0 is released

May 2007 NeXus API version 4.0.0 is released

October 2007 NeXus API version 4.1.0 is released